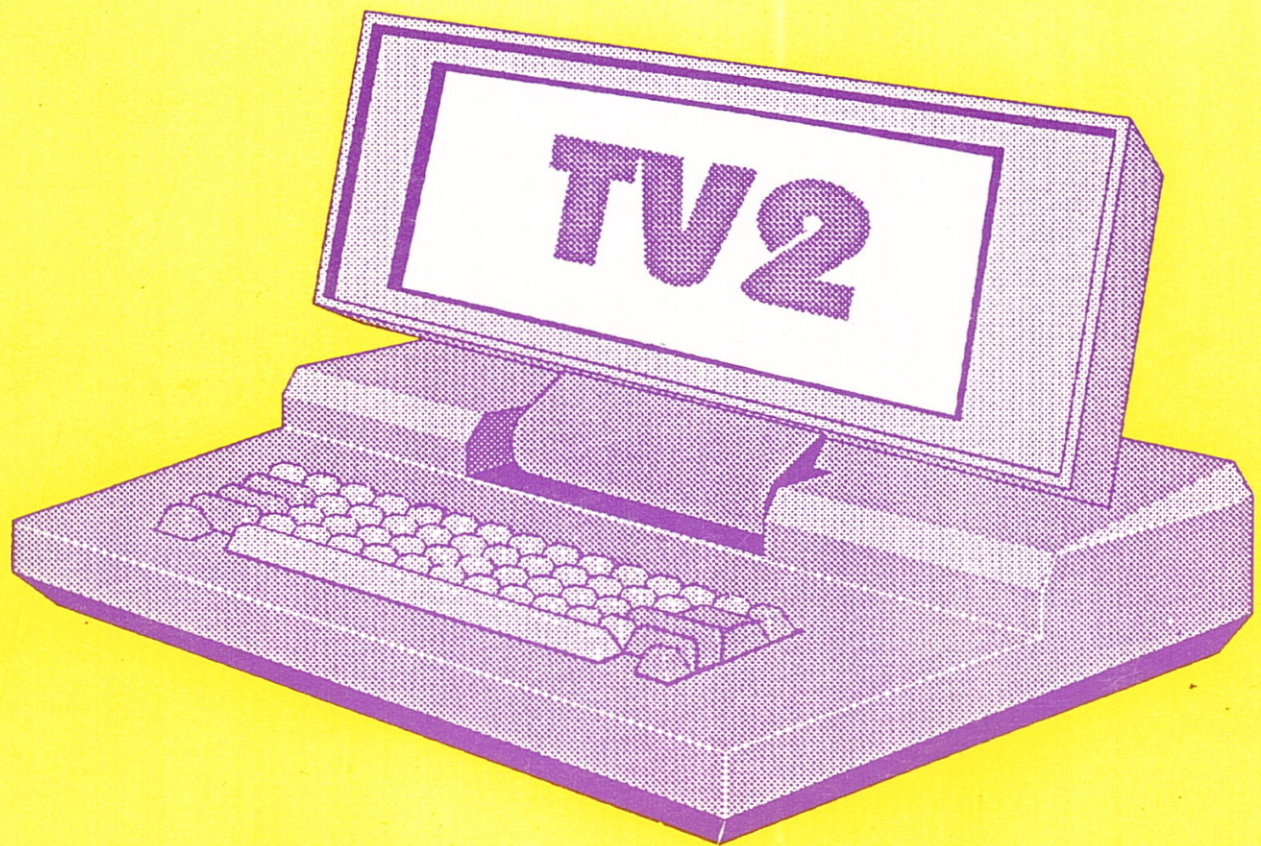


SZÁMÍTÁSTECHNIKA A NAPI GYAKORLATBAN



**MUNKAÜGYI MINISZTERIUM – TV2
TÁVOKTATÁSI PROGRAM**

POLIFON KIADÓ

SZÁMÍTÁSTECHNIKA

A NAPI GYAKORLATBAN

A Munkaügyi Minisztérium és a TV 2
távoktatási programja

Nemzeti Szakképzési Intézet

POLIFON KIADÓ

1992

A könyv megjelentetését az Országos Képzési Tanács Magyar Távoktatási Alapítványa és a Nemzeti Szakképzési Intézet igazgatósága támogatta.

S z e r k e s z t ő:

dr. Simonics István – Munkaügyi Minisztérium

S z e r z ő k:

dr. Bánhidai Ágnes - Központi Statisztikai Hivatal

dr. Brückner Huba - Fulbright Comission

Gerő Judit - Controll Oktatóközpont

Kóczy A. Judit - Controll Oktatóközpont

Nagy Lajos - Kalimera Kft.

Pénzes B. Zuzsa - Controll Oktatóközpont

Reich Gábor - Controll Oktatóközpont

L e k t o r o k:

Füstös János - Veszprémi Egyetem

Helt Ferenc - Központi Statisztikai Hivatal

dr. Szücs Pál - Munkaügyi Minisztérium

©

POLIFON KIADÓ.

A kiadásért felelős: Bakos Zsoltné, az NSZI főigazgató-helyettese. A Kiadó vezetője: Csemus László. Felelős szerkesztő: Török Tivadarné. Készült az SZKI Recognita Rt. tördelése alapján a Rakpart "6" Nyomdában (93/92). Felelős vezető: Galler Péter.

Előszó

1991 júniusában a brit és a magyar Munkaügyi Minisztérium együttműködésében stratégiai jelentőségű konferencia megszervezésére került sor a Budapesti Műszaki Egyetemen. A nyitott és rugalmas oktatás-tanulás kérdésével foglalkozó tudományos tanácskozás a magyar gazdasági átalakulás szempontjából rendkívül lényeges időpontban, és a hosszú távú kérdések megválaszolásának igényével jött létre. E konferencia jellemzője volt, hogy a távoktatás legkorszerűbb eszközrendszerét felvonultatta, és a brit szakemberek segítségével széles szakmai közvélemény tekinthetett be a világszínvonalú oktatástechnikai eszközrendszerbe. A bemutatott technikák és ismertetett eljárások, valamint azok a tapasztalatok, amelyek az egy évtizeddel azelőtt az Egyesült Királyságban végbemenő ipari átalakulás, gazdasági szerkezetváltás és a távoktatás összefüggéseit tették elemezhetővé, rendkívül meggyőzőek voltak.

A szintén 1991-ben megalakult – a munkaadók, a munkavállalók és a kormányzat képviselőiből álló és a szakképzéspolitikai érdekegyeztető fórumaként működő – Országos Képzési Tanács feladatai között megkülönböztetett jelentőséget tulajdonított annak, hogy programjai között a távoktatást megjelenítse az átképzés intézményrendszerében. A múlt év végén létrejött Magyar Távoktatási Alapítvány első jelentős projektje a számítástechnika napi gyakorlatban hasznosítható ismereteinek széleskörű terjesztése. E program célja annak az új típusú tudásnak a megszerzését támogatni, amely ma a munkaerőpiacon sajátos jelleggel képes az elhelyezkedési esélyeket, a hatékony munkavégzést javítani. E szándékot tükrözve formálódott e könyvnek is keretet adó projekt, amely a Televízió és a Munkaügyi Minisztérium együttműködésében, távoktatási program kialakítására tesz kísérletet. A nyomtatott média sajátos információközlő funkcióját kombinálja az egyéni tanulás lehetőségeivel.

A Magyarországon napjainkban zajló munkaerőpiaci folyamatok, a foglalkoztatás átalakuló szerkezete, az új igények az alkalmazottakkal és az általuk képviselt munkakultúrával szemben egyértelműen azt jelzik, hogy a számítástechnika kulcselemmé vált, birtoklása, alkotó alkalmazása nagymértékben növelheti az egyén munkavállalói pozícióját. Ez a felismerés is tükröződik abban, hogy az első távoktatási csomag ezzel a

kérdéskörrel kitüntetett módon foglalkozik. E témakörben kíván a legszélesebb felhasználói kör számára hasznos információkat nyújtani.

A nyitott és rugalmas tanulás magyarországi meghonosítása számos programon keresztül történhet meg. Amikor a Munkaügyi Minisztérium és a TV-2 távoktatási programjának alapkönyvét az olvasó figyelmébe ajánljuk, akkor – az általános szándékon túl – a praktikus ismeretekhez való hozzáférés lehetőségét is biztosítani kívánjuk. A számítógéppel kapcsolatos misztikum, félelem eloszlatása, a napi gyakorlatban is jól használható tudás megszerzése legyen az a közvetlen cél, amely az olvasót vezérli. A könyvben a legkorszerűbb programokkal ismerkedhet meg az érdeklődő, és a számítógéppel folytatandó legáltalánosabb tevékenységekbe nyer bepillantást, amikor a szövegszerkesztés, táblázatkezelés, adatbázis-kezelés rejtelmeibe avatja be a könyv.

Ma már realitás, hogy egyre több munkahelyen, a mindennapi pénzügyi, kereskedelmi gyakorlatban számítógép-hálózatokkal találkozunk, és ezek a rendszerek a legkülönbözőbb technikákat kombinálják a teletextől kezdve a videotexten át az interaktív berendezésekig. A televíziós programokhoz kapcsolódó konzultációk lehetővé teszik a belépést a számítástechnikai oktatásba. A kötet összefoglalja azoknak a szakmai képesítéseknek a körét, amelyek az ilyen típusú tanfolyamokon megszerezhetők. Szándékunk és reményünk szerint hasznos segédletként, további érdeklődést felkeltő és a gyakorlatban is jól alkalmazható könyvként.

Budapest, 1992. március 16.

Benedek András

Néhány gyakorlati tanács a könyv használatához

A könyv elkészítésében egy alkotó közösség vett részt azzal a céllal, hogy az olvasóknak a lehető legtöbb segítséget nyújtsa a közvetlen környezetükben egyre gyakrabban megtalálható számítógépek, a legfontosabb szövegszerkesztő, táblázat- és adatbázis kezelő programok alkalmazásának megismeréséhez, munkájukban minél előbbi felhasználásához.

Ma már számtalan számítástechnikai mű található a könyvpiacra. Könyvünkkel elsőként vállalkoztunk arra, hogy – az alapismereteken túl – együttesen mutassuk be a **Windows** ablak- és menükezelő rendszerprogramot, és e felhasználóbarát környezetben kifejlesztett legfontosabb alkalmazói szoftvereket (1–4. fejezet).

Ugyanakkor fontosnak ítéltük, hogy bemutassunk olyan adatbáziskezelő programot is (5. fejezet), amely ma még nem kapcsolódik közvetlenül a **Windows** környezethez, de a legfontosabb alkalmazói programok körébe beletartozik. Felhívjuk azonban az olvasó figyelmét, hogy az adatbáziskezelés már feltételezi az előző fejezetekben bemutatott módszerek ismeretét, illetve egy kialakult számítástechnikai szemléletet, így e fejezet feldolgozása több időt és gyakorlatot kíván.

A 6. és 7. fejezet azon érdeklődő olvasók számára készült, akik röviden megismerkednének a számítógépes hálózatokkal és a számítógépek ipari alkalmazásával. Ezek a témák azonban számtalan lehetőséget kínálnak, a szerzők csak a legfontosabbakat mutathatták be.

Ha valakinek kedve támad a megszerzett ismeretek kiegészítésére vagy képezés megszerzésére, az tanulmányozza a 8. fejezetben a *Központi Statisztikai Hivatal* képzési- és vizsgarendjét, illetőleg a fejezetek végén levő tanfolyami információs lapokat.

A könyv végén található tartalomjegyzék csak a fő fejezeteket és azok legfontosabb egységeit tartalmazza, az ajánlott irodalom pedig a további elmélyedéshez ad iránymutatót.

Az olvasó tanulását és tájékoztatását segíti a gazdag ábraanyag valamint, az egységes jelölési és kiemelési rendszer:

- **vastagon** szedettek: a gépkezelői részletekben a számítógép **billentyűzetének feliratai**, a képernyőn látható ablak- és menüfeliratok, pl. **Enter, Del, Program Manager, Format** stb.

- *dőltbetűsek* : a szövegben a fontosabb kifejezések vagy meghatározások, pl.
menüsor, adatbázis, közepes sebességű hálózat stb.
- speciális betűvel írottak: a számítógép képernyőjén megjelenő üzenetek, pl.
Search text not found. OK.
The Print Manager cannot write to LPT1.
There may be a printer problem;

A távoktatási program céljának megfelelően a könyvben bemutatott gyakorlati példák részletezésre kerülnek a televíziósorozat adásaiban. A tévésorozat anyaga videokazettán is megvásárolható, így kiegészítő információk is kaphatók az alkalmazásokhoz.

A könyvet a számítógép használata közben célszerű tanulmányozni. Amennyiben nem áll rendelkezésre otthon vagy a munkahelyen számítógép, igénybe vehető az a *konzultációs hálózat*, amelynek listája a mellékletben megtalálható. Ezen szakközépiskolák tanárai vállalkoztak arra, hogy szakértői és gépi szolgáltatást, illetve szervezett tanfolyamokat biztosítanak az érdeklődők számára. Reméljük, mindenki talál lakóhelyéhez közel ilyen központot, és – a megadott szervezőnél személyesen vagy telefonon érdeklődve – részletes információhoz juthat.

A program elsajátításához és a könyv megismeréséhez jó munkát kíván:

a szerkesztő

1. A személyi számítógép és használata; az operációs rendszer

A számítógépekkel ma már szinte mindenfelé találkozunk: a munkahelyeken, sőt otthon is. Ezek azonban már nem ugyanazok, mint amelyeket akár 10-15 évvel ezelőtt a speciálisan felszerelt géptermekekben láthattunk. Az íróasztalunkon is elhelyezhető, elérhető árú, személyi használatú gépek rendkívül sokoldalúan tudnak segíteni a munkánkban.

Sokakban él azonban még mindig az a tévhit, hogy a számítógépet használójának kell felkészítenie a feladatok megoldására, vagyis programozni kell tudnia. Szerencsére ez nem így van. Elég, ha ismeri a gép alapvető tulajdonságait, képességeit és kezelni tudja. *A programok készítése a programozók feladata.* A számítógép-alkalmazók legnagyobb része *kész programokat használ.* Ma már olyan széles ezek választéka, hogy megtalálhatók köztük azok, amelyek kényelmes keretet nyújtanak a feladatok megoldásához.

A megfelelő eredmények eléréséhez bizonyos alapvető ismeretekre van szükség: milyen számítógépet, milyen programokat vásároljunk, és miért azt? Ebben próbál segíteni ez a könyv. A számítástechnika nem a számítástechnikusokért, hanem a felhasználókért van, akiket ki kell szolgálnia. A ma, s még inkább a jövő felhasználója többnyire nem számítástechnikai szakember, s előismeretek nélkül, lehetőleg azonnal szeretné élvezni a számítástechnika előnyeit: a gyorsabb, pontosabb, kényelmesebb, megbízhatóbb feladatmegoldást. Őt azonban csak *felhasználóbarát gépekkel, programokkal lehet megnyerni.*

A számítógépről - röviden

Mit nevezünk számítógépnek?

A számítógép olyan elektronikus berendezés, amely információk feldolgozására (számítási műveletek elvégzésére, tárolására, visszakeresésére stb.) képes a számítógépben tárolt program segítségével.

Ebben a könyvben a mindenki által leginkább elérhető eszköz, a személyi számítógép felépítésével, legtipikusabb alkalmazási lehetőségeivel szeretnénk megismertetni az Olvasót.

A számítógép hardverének nevezzük a gépben található, kézzelfogható egységeket: az alapgépet, a monitort, a billentyűzetet, nyomtatót. A működéshez azonban ez nem elegendő: programra is szükség van, amely a feladat megoldásához vezető lépéseket, utasításokat tartalmazza. Különbőféle feladatokhoz más és más program kell. A programok összefoglaló neve: szoftver.

A számítógép hardverjének legfontosabb része a központi egység, a processzor, a CPU. A programutasításokat ez az egység hajtja végre. Ez nemcsak valamilyen számítási művelet (például két szám összeadásának) elvégzését jelentheti, hanem például a kiszámított eredmény kinyomtatását is. Így a központi egység feladata – a számítási, illetve logikai műveletek elvégzése mellett – a számítógép egyéb egységeivel, például a nyomtatóval való együttműködés biztosítása, az ehhez szükséges vezérlési feladatok ellátása.

A mikroprocesszor olyan integrált áramkör, amely képes erre. Ez az a különleges alkatrész, amelynek kifejlesztése az egész számítástechnikát forradalmasította.

Még a 70-es évek legelején történt, hogy az elektronikai alkatrészgyártásban jónevű amerikai Intel cég egy japán megrendelést az addig megszokottól eltérően teljesített: a kívánt működést megvalósító céláramkör helyett egy általánosabban használható, intelligens alkatrészt állított elő. Ennek konkrét működése csupán attól függött, hogy a vele szoros kapcsolatban álló speciális alkatrészben milyen tevékenységsort rögzítettek. Ez az utóbbi eszköz természetesen a szükséges belső tár, a memória, s benne a működés programját helyezték el. Az elkészített alkatrész, amely képes volt a kívánt működés végrehajtására, tudta mindazt, amit egy központi egységnek tudnia kell: fogadta a memóriából érkező utasításokat, értelmezte őket, eldöntötte, mi a teendője a végrehajtásukkal kapcsolatban, s ezeket meg is tette.

Ha a számítógépnek egy ilyen fontos eleme tömegméretekben gyártható, akkor az elvezet az olcsó, mindenki által hozzáférhető számítógéphez. Így történt. Mindenki hallott a Commodore diadalútjáról, amikor nálunk, pénzsűkében, de felismerve az ügyviteli feladatok gépesítésének elodázhatatlanságát, megjelentek - a hivatalokban. Nem erre készültek. Amikor aztán az IBM a maga egész világra kiterjedő üzlet- és szervizhálózatával megjelent a piacon a szakmai szempontból újdonságot nem tartalmazó, de megbízható, professzionális PC-jével, az óvatos, hosszú távban gondolkodó kis cégek e mellett döntöttek. Nálunk is: a gépesített ügyvitel előnyei mellett tapasztalva a Commodore-ok korlátait, lecserélték a gépeiket. Ha nem is mindenütt az eredeti, IBM gyártmányúra, de legalább ahhoz nagyon hasonlóan működő, vagyis vele kompatibilis gépekre. Ezért ebben a könyvben is az IBM, illetve IBM-kompatibilis személyi számítógépekkel foglalkozunk.

A személyi számítógépnek vagy másképpen mikroszámítógépnek azokat a számítógépeket nevezzük, amelyek központi egysége egy mikroprocesszor. Mikroprocesszorokat nemcsak számítógépekben használnak, hanem szinte mindenütt, ahol valamilyen vezérlési feladatot kell megoldani: pl. a legkülönbébb ipari folyamatokban, gyártásban, ellenőrzésben, de a háztartási gépek közül is jónéhányban, például az automata mosógépekben.

A központi egységek (processzorok) két fontos jellemzője a sebesség és a szóhossz. A központi egység belső működését órajel ütemezi, ezért ennek frekvenciája (egysége Hz, MHz) arányos a műveletvégzések sebességével: minél nagyobb az órajel frekvenciája, annál gyorsabban hajtódik végre a programlépések. A szóhossz az a jellemző, amely megadja, hogy egyszerre mennyi információ kezelésére (fogadására vagy továbbítására, illetve velük műveletvégzésre) képes a központi egység. A szóhosszt bitben mérjük, amely a számítógép által kezelt elemi információ; értéke 0 vagy 1. A számítógép belső folyamatai a kettes (bináris) számrendszerrel modellezhetők. Mind az adatok, mind a programok binárisan kódolt formában ábrázoltak. Ehhez bitcsoportokra van szükség: 8 bit alkot egy bájtot. Ez akkora egység, amely annyi kombinációs lehetőséget ad, hogy 1 bájton ábrázolható például az összes nagy- és kisbetű, számjegy és írásjel kódja. A mikroszámítógépek az ASCII kódrendszert használják. A ma legáltalánosabban használt személyi számítógépek 16 bitesek, de a 32 bitesek is megtalálhatók már.

A központi egységgel legszorosabb együttműködésben az **operatív memória** áll. A felépítésével, funkciójával kapcsolatos követelményeket

Neumann János fogalmazta meg először, akit a korszerű, elektronikus számítógép atyjaként emlegetünk: a memória egy homogén tároló, amely a gép pillanatnyi működéséhez szükséges programokat, adatokat tartalmazza. Nincs elkülönített memóriarész csak az adatok, illetve csak a programok számára. Ez a rugalmasság biztosítja a számítógép sokféle célra történő, vagyis univerzális alkalmazását. A memória tárolókapacitása, kapacitása határozza meg a futtatható programok nagyságát, a gép adatfeldolgozó kapacitását. A személyi számítógépek között ma már ipari szabványnak számító, legjobban elterjedt IBM PC-knél ez legalább 640 kilobájt (1 Kbájt = 1024 bájt, 1 Mbájt = 1024 Kbájt). Az operatív memória alapvetően kétféle fizikai tulajdonságú tárolóelemekből épül fel: ROM és RAM elemekből.

A **ROM** csak olvasható memória, vagyis a benne tárolt információ nem változtatható meg, de felhasználható: az itt tárolt program futtatható, ha adat, feldolgozható: művelet végezhető vele. A ROM-ban tárolódnak a gép alapvető működéséhez szükséges programok.

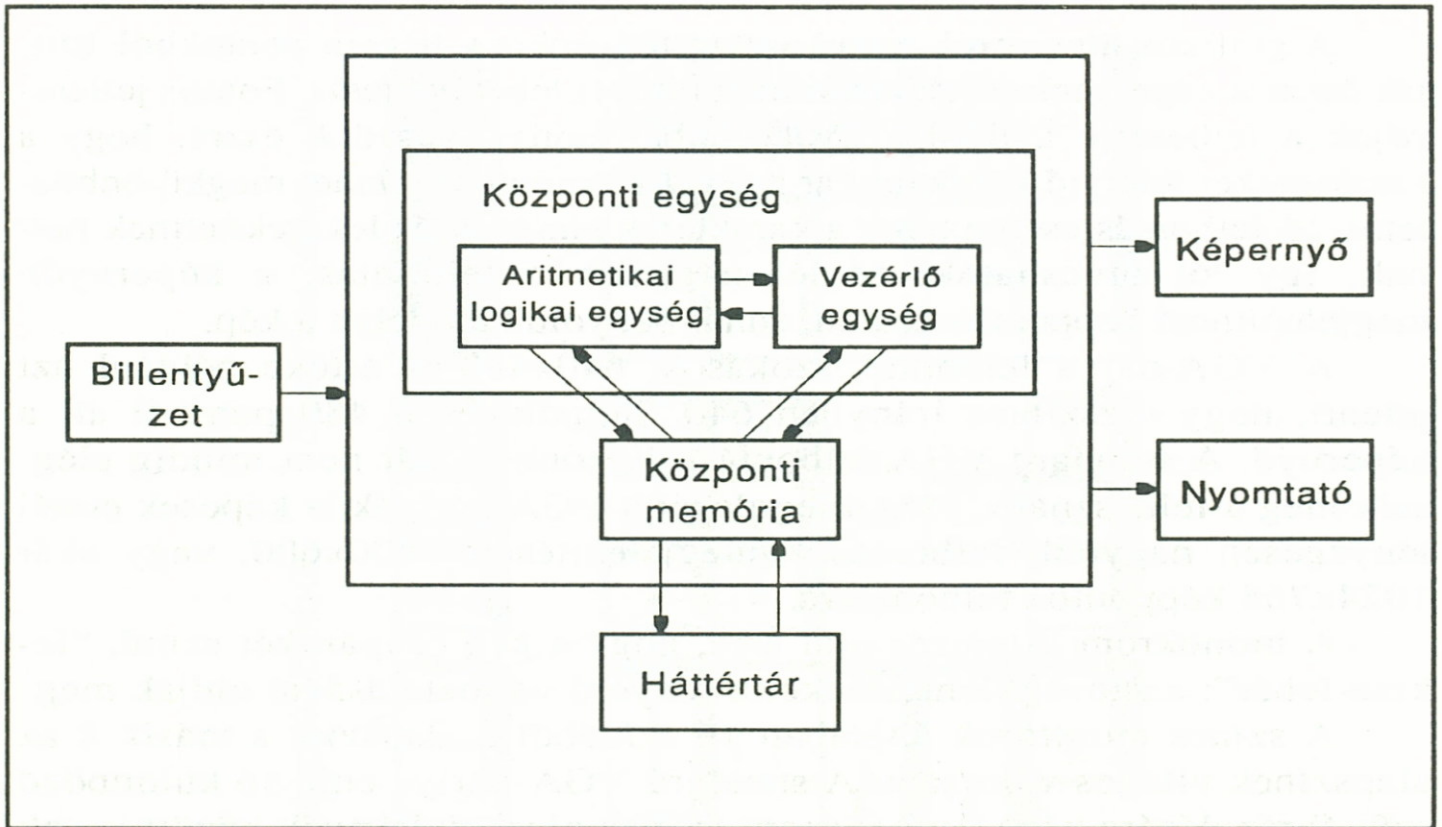
A **RAM**-terület írható és olvasható, ezért ide kerül az a program, amelyet éppen futtatni, végrehajtani akarunk, illetve a program által felhasznált adatok is. Újabb feladat megoldásához a szükséges programot betöltve a RAM-ba, a korábbi tartalom eltűnik, felülíródik. Az itt tárolt adat, program azonban akkor is elvész, ha a számítógépet kikapcsoljuk, s ugyanez történik áramkimaradáskor is. Változó adataink, programjaink tartós tárolására tehát az operatív memória nem alkalmas.

A számítógép és a felhasználó közötti kapcsolattartás eszközei a **perifériák**. A beviteli periférián (pl. billentyűzeten) a gép kezelője adatokat juttathat be, programokat írhat stb. A kiviteli periférián (pl. monitoron) jelennek meg a programok üzenetei, az eredmények.

Különleges perifériális eszközök a **háttértárolók**, amelyek a programok, adatok tartós tárolására szolgálnak. Bizonyos esetekben akkor is használhatók azonban, ha a futó program vagy a feldolgozandó adatok egyszerre nem férnek el a memóriában.

Általában lemezes háttértárolókat használunk, amelyek mágneses elven működnek. Két alaptípusuk a cserélhető lemezes hajlékonylemez (**floppy**) egység, valamint a rendszerint beépített, gyors, nagykapacitású merevlemez egység (**winchester**).

A számítógép legfontosabb rendszertechnikai egységeit, s azok kapcsolatát az (1-1. ábra) mutatja.



1-1. ábra

A legfontosabb perifériatípusok

A monitor

A **monitortalán** a legfontosabb periféria. Ezzel ugyanis valamennyi felhasználó állandóan kapcsolatban áll: megjelennek rajta a programok üzenetei, illetve a bevitt adatokat a monitoron ellenőrizhetjük. A monitor hatással van a számítógéppel végzett munka minőségére, illetve lehetőségeire. Pontos, éles, vibrálás-, villogás- és tükrözésmentes képet kell adnia. A lehetséges videorendszerek (monokróm, CGA, Hercules, EGA, VGA, SVGA) közül az alkalmazáshoz kell kiválasztanunk a megfelelőt: grafikus programokhoz mindenképpen grafikus rendszer szükséges, ma már lehetőleg VGA.

A monokrom monitorok a legolcsóbbak; alfanumerikus és grafikus típusuk is létezik. Az alfanumerikus monitoron csak karakterek (betűk, számok, írásjelek, táblázatok készítéséhez bizonyos grafikus jelek) jeleníthetők meg, 40 vagy 80 karakter széles sorokban, a sorok száma 25.

A grafikus monitorok nem karakterhelyenként, hanem pontokból állítják össze a képet, ami pl. rajzok ábrázolását is lehetővé teszi. Fontos jellemzőjük a felbontás, amiből a gyakorlatban annyit veszünk észre, hogy a karaktereket felépítő képpontokat szabad szemmel alig lehet megkülönböztetni. Jó felbontás esetén tehát a karakterek simának és lekerekítettnek hatnak, így jól olvashatók. Minél nagyobb a felbontás, a képernyőn megjeleníthető képpontok száma, annál bonyolultabb lehet a kép.

A VGA-nál a felbontás szokásos, 640x480-as értéke például azt jelenti, hogy vízszintes irányban 640, függőlegesen 480 pontból áll a képernyő. A standard VGA felbontással azonban már nem mindig elég-szik meg a felhasználó. Még a legolcsóbb VGA kártyák is képesek ennél lényegesen nagyobb felbontások megjelenítésére: 800x600, vagy akár 1024x768 képpontos felbontásra.

A monokróm kifejezés utal arra, hogy a kép csupán két színű, "fekete-fehér"; a szövegkiemeléseket a fényerő változtatásával oldják meg.

A színes monitorok általában 16 színéből 8 alapszín, a másik 8 az alapszínnek világos változata. A standard VGA-kártya eme 16 különböző szín ábrázolására képes, ma viszont már az olcsóbb kártyák között is sok van, amelyik egyszerre 256 színt jeleníthet meg.

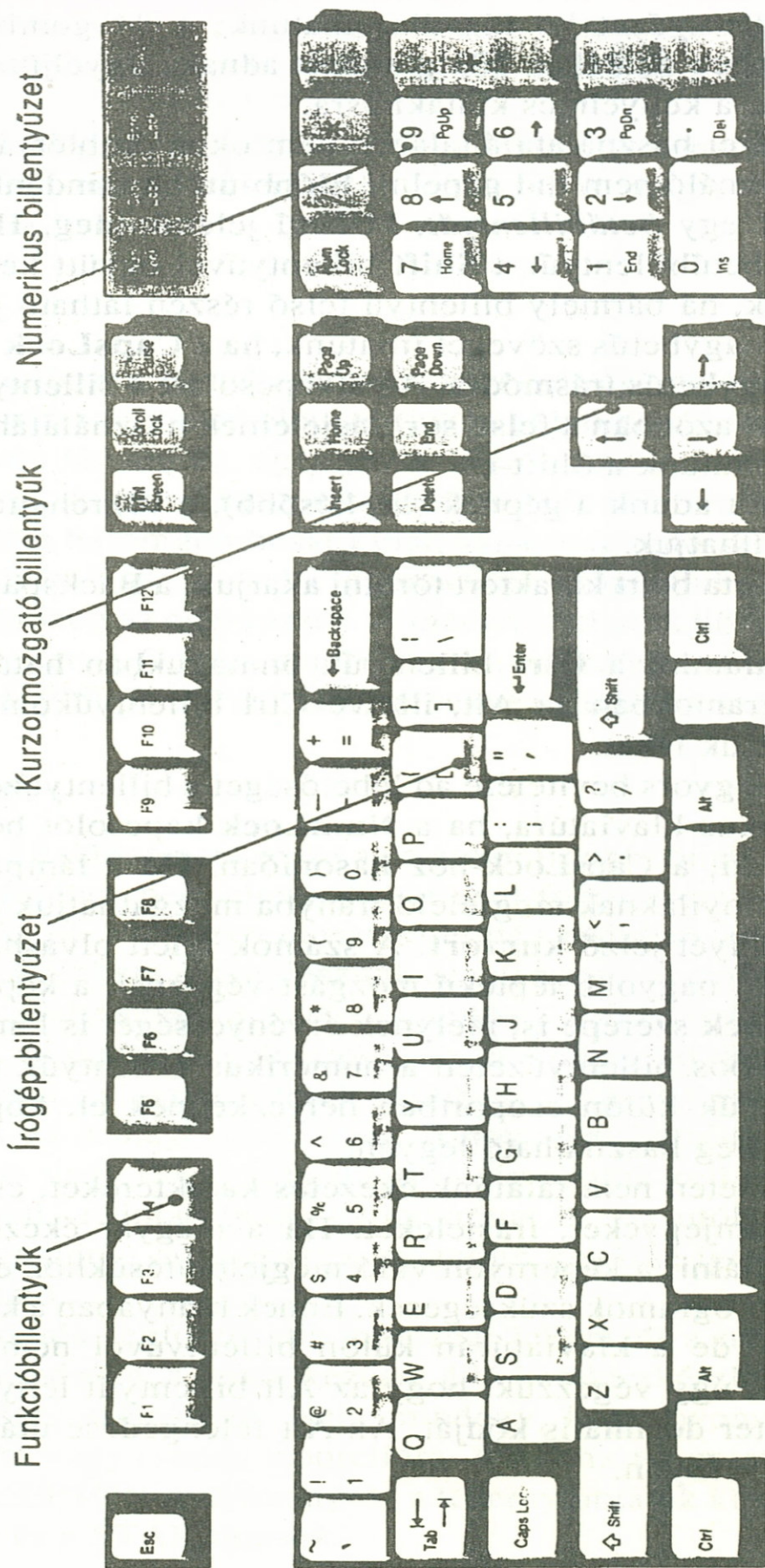
Vannak olyan monitorok is (pl. NEC), amelyek többféle videoszabvánnyal képesek működni. Önállóan felismerik az adott szabványfajtát, és beállítják hozzá a megfelelő üzemi paramétereket.

A nagy felbontású, sok szín megjelenítéséhez nagy képernyőmemória szükséges. Pl. a standard VGA felbontáshoz (640x480 képpont, 16 szín) 150 Kb-ot, míg 1024x768 képpontos felbontáshoz, 256 szín esetén már 750 Kb-ot. A nagy felbontású, sokszínű grafikák ábrázolásához idő kell a leggyorsabb számítógép számára is. A megjelenítési idő rövidítik az ún. gyorsítókártyák. Alkalmazásuk egyelőre nem tekinthető általánosnak.

A billentyűzet

Az adatbeviteli eszközök kiválasztása nagyon fontos, hiszen nincs még egy olyan része a gépnek, amely annyira személyhez kötődne, mint például a billentyűzet és az egér. Kiválasztásuknál szerepet játszhatnak a szubjektív szempontok éppúgy, mint a termék ára vagy a hozzá nyújtott szolgáltatások.

Az a **billentyűzet**, amelyik az egyik felhasználó számára kellemesen rugalmas, a másiknak túlságosan puha (1-2. ábra). Az egér akkor megfelelő, ha belesimul a tenyerünkbe.



1-2. ábra

Sokféle billentyűzet közül választhatunk: a 84 gombostól a 102 gombosig; a géphez általában 101 gombost adnak. Ügyeljünk a hardverkompatibilitásra, a kényelmes kialakításra.

A billentyűzet használata általában nem okoz problémát még akkor sem, ha a felhasználó nem tud gépelni. Előbb-utóbb mindenki belejön.

Ha leütünk egy *betűbillentyűt*, kisbetű jelenik meg. Ha nagybetűt akarunk írni, a betűbillentyűt a **Shift** billentyűvel együtt kell lenyomni. Ugyanezt tegyük, ha bármely billentyű felső részén látható jelet akarjuk leírni. Tartósan nagybetűs szöveget írhatunk, ha a **CapsLock** kapcsolóval átkapcsolunk nagybetűs írásmódra. Az átkapcsolást a billentyűzeten lámpa jelzi. Ilyenkor azonban a felső sor írásjeleinek használatához változatlanul le kell nyomnunk a **Shift**-et.

Ha parancsot adunk a gépnek (ld. később), a végrehajtást az **Enter** billentyűvel indíthatjuk.

Ha az utoljára beírt karaktert törölni akarjuk, a **Backspace** billentyűt üssük le.

Az **Alt**, valamint a **Ctrl** billentyűk önmagukban hatástalanok. A különböző programokban az **Alt**, illetve **Ctrl** billentyűkombinációknak más-más jelentésük lehet.

Számadatok gyors bevitelére ad lehetőséget a billentyűzet jobb oldalán levő numerikus klaviatúra, ha a **NumLock** kapcsolót bekapcsoljuk. Ezt is lámpa jelzi, a **CapsLock**-hoz hasonlóan. Ha a lámpa nem ég, a programokban a nyilaknak megfelelő irányba mozgathatjuk a képernyőn az adatbevitel helyét jelző **kurzort**. A számok felett olvasható feliratok (**Home**, **End** stb) nagyobb léptékű mozgást végeznek a képernyőn. Hasonló a **ScrollLock** szerepe is, melynek érvényességét is lámpa jelzi.

A 101 gombos billentyűzeten a numerikus billentyűk és a kurzormozgató billentyűk külön csoportban helyezkednek el, hogy a kétféle funkció egyidejűleg használható legyen.

A billentyűzeten nem találunk ékezetes karaktereket, csak az angol ABC betűit, számjegyeket, írásjeleket. Ha a magyar ékezetes betűket szeretnénk használni, a képernyőn való megjelenítésükhöz és a nyomtatásukhoz külön programok szükségesek. Ennek hiányában a karakterkészletben szereplő, de a klaviatúrán külön billentyűvel nem rendelkező karakter beírását úgy végezzük, hogy az **Alt** billentyűt lenyomva tartva beütjük a karakter decimális kódját. Az **Alt** felengedése után a karakter megjelenik a képernyőn.

Az **F1-F10** (vagy **F12**) jelű ún. funkcióbillentyűk használata is programfüggő: más és más funkciót rendelnek hozzájuk a különböző programok.

Nem árt még, ha tudjuk, hogy ha például egy vonalat akarunk húzni, egyetlen hosszantartó billentyűlenyomással is megtehetjük a sokszori billentyűleütés helyett.

Az egér

Az **egér** egy doboz, melynek alsó, nyitott részén egy golyó van elhelyezve. Ha az egeret sima felületen elmozdítjuk, a golyó gördülésének megfelelően a képernyőn látható ún. egérkurzor is elmozdul.

Az egerek közül a kétgombos Microsoft vált szabvánnyá, néhány gyártó azonban háromgombosot kínál, saját meghajtó szoftverrel. A legtöbb egér dinamikus, azaz ha gyorsabban mozgatjuk, a képernyőn is gyorsabban mozog az egérkurzor. A piacon kaphatók közül a jó minőségű Microsoft mellett a Genius emelkedik ki, különösen olcsósága miatt. A márkás (IBM, Philips, AT&T stb.) gépekhez általában saját egeret adnak.

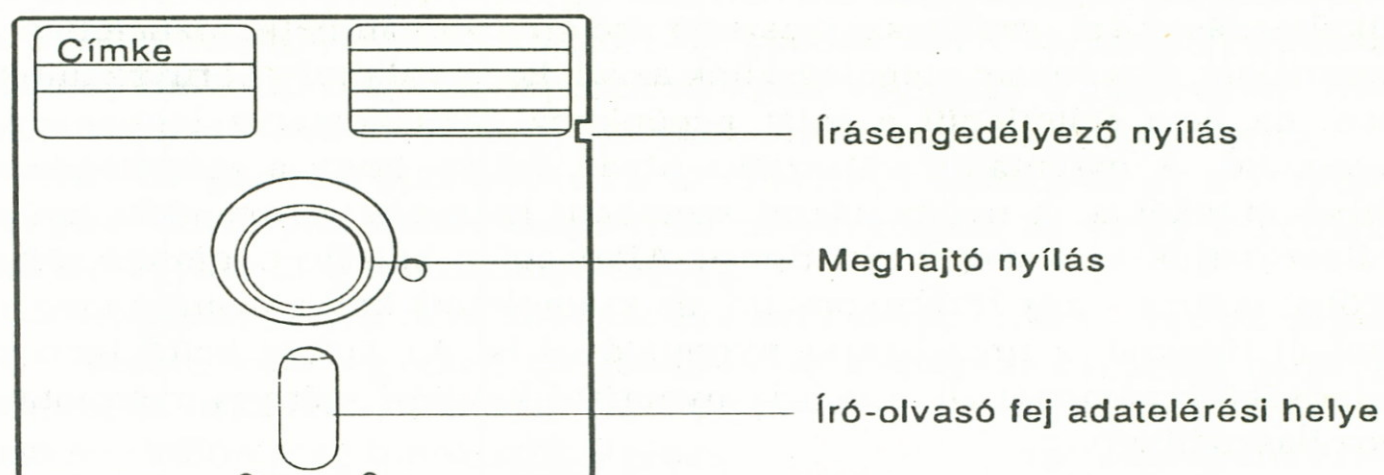
A nyomtató

A **nyomtató** általában nem része az alapkonfigurációnak, hiszen a számítógép működéséhez nem elengedhetetlenül fontos berendezés. Bizonyos alkalmazások (pl. szövegszerkesztés) esetén azonban nélkülözhetetlen a használata. Egyébként megelégszünk azzal, hogy valahol a környezetünkben, de nem feltétlenül a saját gépünkhöz csatlakoztatva legyen egy nyomtató. A nyomtatók választéka olyan széles, hogy a számítógépen folyó munkákat, a nyomtatással szembeni igényeket szem előtt tartva választhatjuk ki a megfelelő típust. Alapvetően kétféle nyomtató jöhet szóba: mátrix- vagy lézernyomtató, de számolnunk kell a nemrég megjelent új típusokkal, a tintasugaras nyomtatóval is. Az utóbbi kettő igen jó minőségű eredményt ad, a mátrix-nyomtató kevésbé igényes nyomtatáshoz használható.

A nyomtatók legfontosabb minőségi jellemzői a nyomtatási sebesség, az írás minősége, a rendelkezésre álló karakterkészlet, a használható papír minősége, szélessége. Nem mindegy, hogy nyomtathatunk-e A4 méretű papírra vagy csupán leporellóra. A mátrix-nyomtatók közül legismertebbek az EPSON-gyártmányok, a lézernyomtatók között a HP (Hewlett Packard) és a STAR típusok.

Lemezes háttértárolók

Adatainkat, programjainkat tartósan mágneslemezes háttértárolókon tároljuk (1-3. ábra). Minden mágneses réteggel bevont lemez egymástól pontosan meghatározott távolságban elhelyezkedő koncentrikus körön (sávokon) tárolja az információt, amelyek azonos középponti szöghöz tartozó körcikkekre, **szektorokra** oszlanak. Egy szektor mérete 512 bájt. A teljes lemez kapacitását a sávok és a szektorok sávonkénti száma határozza meg. Ma már ugyanis csak olyan lemezegységeket használunk, amelyek a lemezek mindkét oldalán tárolnak adatokat. A szokásos méretű, 5 1/4" átmérőjű hajlékonylemezek kapacitása, amelyeket XT típusú gépen használunk, 360 Kbájt (40 sáv, 9 szektor/sáv), az AT-ken használatosaké 1,2 Mbájt (80 sáv, 15 szektor/sáv). A lemez formátuma az ún. formázáskor vagy formattáláskor egy program segítségével kerül kialakításra (ld. később). Egyre inkább terjed a kis helyigényű 3 1/2"-os hajlékonylemez egységek használata is, amelyek kapacitása 720 Kbájt, illetve 1,44 Mbájt.



1-3. ábra

A **hajlékonylemez** minősége a felírási sűrűség miatt döntő. Ez a lemezek dobozán, valamint a lemezeken található gyári címkén van feltüntetve: a DD (Double Density) jelű lemezek a 360 Kbájtos meghajtókhoz használhatók, a HD (High Density), HC (High Capacity) vagy HQ (High Quality) jelűek az 1,2 Mbájtosakhoz.

Az XT gépek **hajlékonylemez egysége** nem tud mit kezdeni az 1,2 Mbájtos lemezekkel, az AT típusú gépen azonban gond nélkül használhatók az XT-n készített lemezek. Sajnos, előfordul azonban, hogy ha egy ilyen lemezt újra XT gépen szeretnénk használni, már nem tudja a lemezegység elolvasni. Nem minden AT-nál előforduló probléma ez, ezért új gép vásárlásakor ki kell próbálni.

A lemezeken tárolt adatok, programok kiválasztásához nyújt segítséget a lemezre ragasztható öntapadós címke, amelyet a lemezt tartalmazó dobozban találunk.

A hajlékonylemezek igen sérülékenyek: óvnunk kell hőtől, nedveségtől, mechanikai hatásoktól, mágneses tértől. A mágneses réteggel bevont műanyaglemez ezért levehetetlen tokban van elhelyezve. Az élettartama azonban így is véges: rendszeres napi használat esetén kb. fél év. A használt, kopott lemezen tárolt adatok másik adathordozóra történő átmásolásával a hasznos információ megőrizhető. A lemezek oldalán látható négyszögletes ún. írásvédelmi nyílás leragasztásával a lemez tartalma megóvható a véletlen törléstől, átírástól. Különösen óvni kell a szennyeződéstől azt a hosszúkás nyílást, amelyen át a csupasz lemez felületét látjuk. A tárolt adatokhoz itt fér hozzá a meghajtóegység író-olvasó fejpárja. A lemez használat közben forgásban van, befogása a lemezforgató egységbe a lemez közepén levő nyíláson keresztül történik. Így az író-olvasó fejeknek csak sugárirányban kell mozogniuk a megfelelő adathelyek keresésekor. A lemez szennyeződésektől való védelmét szolgálja az a papírboríték is, amelybe mindig tegyük bele a lemezt, ha kivettük a meghajtóból.

A hajlékonylemez egységbe a címkékkal felfelé, az író-olvasó nyílással befelé óvatosan, ütközésig kell betolni a lemezt, majd becsukni a lemezegység kilincset. Fontos még, hogy ha a lemezegység lámpája ég, a lemezt kivenni nem szabad. A gépekben általában 1 vagy 2 lemezegység van, újabban esetleg az egyik 3 1/2"-os méretű.

A **winchester** igen nagy kapacitású, rendszerint fixen beépített, nem cserélhető tároló. Az általában 5 1/4"-os lemezek hermetikusan elzárt térben, közös tengelyre felfűzve helyezkednek el. Minden lemez mindkét oldalához tartozik író-olvasó fej. A winchesterműveletek sebessége kb.

10-szerese a floppyénak, ezért a gyakran használt programok, adatállományok praktikus tárolóhelye, s az operációs rendszert is célszerű felvinni rá. A legfontosabb programjainkról, adatainkról hajlékonylemezen biztonsági másolatot kell készítenünk. Használat előtt a merevlemezeket is formázni kell. Fontos még, hogy a winchester egységet tartalmazó gépet szállítani csak úgy szabad, ha az író-olvasó fejek ún. parkolópályára vannak állítva. Ez egy megfelelő programmal lehetséges, amely az író-olvasó fejeket kihúzza az adatterület fölé. Ezután a gép kikapcsolható. A winchester a következő bekapcsoláskor minden külön beavatkozás nélkül újra használható.

Milyen számítógépet vegyünk?

Ha már van gépe, Önnek ez már nem kérdés. Csak akkor olvassa el ezt a fejezetet, ha már lassúnak érzi, vagy azok a programok, amelyekkel dolgozni szeretne, nagyon korlátozottan vagy sehogy sem használhatók rajta. Ekkor el kell gondolkodnia: nem kellene-e kicserélni egy gyorsabb, nagyobb memóriával s más, az Ön számára fontos új tulajdonságokkal rendelkező gépre.

Mások is szeretnék munkájuk hatékonyságát számítógép segítségével növelni. Talán el is mennének vásárolni egyet, csak nem tudják, mi alapján válasszanak. A feladat nehéz: akkor tudjuk megmondani, mire lett volna szükségünk, ha már szűkössé váltak a használt gép lehetőségei. Valahol azonban el kell kezdeni. Sajnos, az újságokban megjelenő hirdetések nem könnyítik meg a dolgunkat. A következőkben megpróbálunk segíteni a választásban: melyet vegyenek?

Természetesen egy PC-t. A hivatalokban használt számítógépek nagy többsége IBM PC vagy azzal kompatibilis gyártmány, ezért ilyet válasszunk. Két számítógépet akkor nevezünk egymással kompatibilisnek, ha a programok ugyanúgy futnak le mindkét gépen. Az IBM-kompatibilis gépek közül (ezeket a hasonló működés miatt a továbbiakban IBM PC-knek nevezzük) az XT és AT típusok jöhetnek szóba, mivel ez az a két típus, amely beszerezhető, s az elvárt professzionális képességekkel rendelkezik. Az XT géptípus központi egysége egy 8088 típusú mikroprocesszor, az AT-ké 80286, 80386 vagy 80486 típusú. A központi egység típusán kívül a felhasználó számára, amint már említettük, a különböző hajlékonylemezek használata szembeötlő: az XT gépekbe 360 Kbájt kapacitású lemezegegyeségeket építettek be, s az AT gépeknél jelent meg a nagykapacitású, 1,2 Mbájtos hajlékonylemez egység. Általában az

AT-k nagyobb (3 Mb-ig bővíthető) operatív memóriával és nagyobb kapacitású (akár 200 Mb-ig) merevlemez egységgel rendelkeznek.

Az IBM PC-k egymással fölülről kompatibilisek, azaz egy korábbi típuson írt program fut az újabb gépeken is. Fordítva azonban nem igaz: az újabb típusok már többet tudnak, s az ezeket a képességeket kihasználó programok a régebbi típusokon nem futtathatók. A számítógép **konfigurációját**, aktuális eszközkészletét úgy kell megválasztani, hogy az a megoldandó feladatokhoz ideális, de legalábbis megfelelő legyen.

Szövegszerkesztésre, kisebb adatbázisok, táblázatok kezelésére akár egy XT gép is megfelel. A ma legnépszerűbb programok azonban már igénylik a nagyobb memóriát, a gyorsabb műveletvégzést, nem csoda hát, ha az XT-k helyét egyre inkább átveszik a különböző AT-k, tipikusan 80286 típusú processzorral. Ha azonban nagy mennyiségű számítást kell elvégezni (pl. CAD/CAM-alkalmazások) feltétlenül gyors, 486-os gépre van szükség.

Hálózati, osztott felhasználásra szintén legalább 386-os gépet célszerű választani. Nem érdemes azonban 486-osat, mert a hálózat központi gépén, a szerveren nincsenek számítási műveletek, sebességben pedig nincs lényeges különbség a 386-os és a 486-os gép között. Olyan gépre van szükség, amely képes a file-műveletek villámgyors elvégzésére, ehhez pedig gyors winchester és ugyanilyen vezérlő szükséges. Hálózati munkaállomás esetén egyértelműen a hálózati csatoló adatátviteli sebessége és a processzor feldolgozási sebessége a meghatározó, miután az ilyesfajta gép a szerveren végzi a winchesterműveletek többségét.

Természetesen mindenki jó és megbízható gépet szeretne, de ha nem tudja megfizetni a legjobb márkákat, olyan szállítót válasszon, aki a kevésbé neves termék mellé különböző szolgáltatásokat kínál: garancia, szerviz stb.). Az előrelátók bővítési lehetőséget is hagynak a rendszerben, hogy az igények növekedésével a gép kicserélése nélkül lehessen majd növelni a kapacitást (cserélhető processzor, memória-, illetve kártyabővítő helyek).

Az operációs rendszer

Szoftver nélkül a számítógép működésképtelen. Valamilyen módon kommunikálnunk kell a géppel, hogy megadjuk például, melyik programot akarjuk futtatni. Ezt billentyűzetten tehetjük meg, de csak akkor, ha a beírt parancsunkat egy program értelmezni, végrehajtani tudja.

Ezt a lehetőséget és egy sereg más nélkülözhetetlen funkciót az operációs rendszer biztosítja. Olyan környezet, amely azon túl, hogy lehetővé teszi a gép működését, gondoskodik a különböző egységek megfelelő együttműködéséről, s közreműködésükről a felhasználó által kért feladat végrehajtásában. Hiszen szinte minden programban vannak olyan funkciók, amelyek programozásához a gép fizikai szintű működésének alapos ismerete szükséges. Ezeket az általános, háztartási problémákat megoldó programokat rendszerprogramoknak nevezzük, s használatukra mindannyiszor sor kerül, amikor egy program például valamilyen perifériaműveletet (adatbevitelt billentyűzeten, nyomtatást) kér. A rendszerprogramok összessége az **operációs rendszer**. Legfontosabb feladatai a programok végrehajtásának ütemezése, perifériaműveletek lebonyolítása, a gép különböző egységei működésének összehangolása, kommunikáció a gép kezelőjével. Eltakarja előlünk a hardver szinten elvégzendő feladatokat, s egy számunkra érthetőbb, kellemesebb képet mutat a számítógépben folyó munkákról. A programok, a parancsaink egy logikai interfésszel, kapcsolódási felülettel vannak közvetlen kapcsolatban, s az operációs rendszer a hívott funkcióhoz hozzárendeli a fizikai végrehajtást meghatározó rutinjait, biztosítva a nélkülözhetetlen fizikai kapcsolódási felületet.

A számítógéppel végzett munka tulajdonképpen az operációs rendszernek adott parancsok sorozatából áll. Ahhoz, hogy parancsunkat a gép végre tudja hajtani, a gép bekapcsolása után rögtön meg kell jelennie ennek a szoftver környezetnek: be kell töltnie az operációs rendszernek a memóriába. A gép kezelőjének billentyűleütéseit egy dinamikus várakozási állapotban várja, s egy speciális karakter (az **ENTER**) leütéséig gyűjti, majd indulhat a felhasználói parancs értelmezése, végül a végrehajtása. A parancs végrehajtásának befejeztével ismét visszatér a várakozási állapotba.

A számítógépes programok másik nagy csoportja hajtja végre a felhasználók különleges feladatait. A programok között az alkalmazói programok igen jelentősek, mivel keretrendszerek egy-egy feladattípus megoldásához. A leggyakoribb ilyen típusfeladatok a szövegszerkesztés, a táblázatkezelés, az adatbáziskezelés stb. Ezek részletes ismertetésére a következő fejezetekben kerül sor. Az egészen speciális, egyedi feladatok megoldására kifejlesztett programokat csak egy szűk felhasználói réteg tudja alkalmazni (pl. autóbusz menetrend készítés, ingatlan ügynökség ügyfeleinek kiszolgálására készített rendszer, vagy éppen a személyi jövedelemadót kiszámító program).

Az operációs rendszerek típusai

Mivel az operációs rendszer ütemezi, felügyeli a gép működését, meghatározza a gép használatának néhány igen fontos jellemzőjét: egyszerre hány program futhat, hogyan történik a nyomtatás stb. A személyi számítógépeken általánosan használt DOS operációs rendszer egyfelhasználós, azaz a számítógép kezelőjének egyetlen programját végrehajtja, majd ha újabb parancsot kap, azt hajtja végre stb. A multiprogramozás elve ad megoldást arra, hogy hogyan lehet kihasználni az egyetlen program futtatása során óhatatlanul jelentkező "üresjáratokat"; a program ugyanis végrehajtása során nem képes folyamatosan leterhelni a központi egységet és a perifériákat. Alapelve a kényszerű várakozási idők kihasználása úgy, hogy a központi egység használatát valamilyen stratégiával átkapcsolja a különböző programok között. A látszólag egyidőben futó programok közösen osztozkodnak a központi egységen, egyszerre azonban csak egyikük használhatja. Ez egyfelhasználós környezetben is megvalósítható: az egymás után indított programok "párhuzamos" feldolgozásával. Ilyen operációs rendszer az OS/2.

Ezzel szemben a többfelhasználós multiprogramozás olyan üzemmód, amely különböző alkalmazásokhoz tartozó programok aszinkron végrehajtására képes, és a futások tervezését és összehangolt lebonyolítását az operációs rendszer (pl. a XENIX) végzi.

Az állomány (a file)

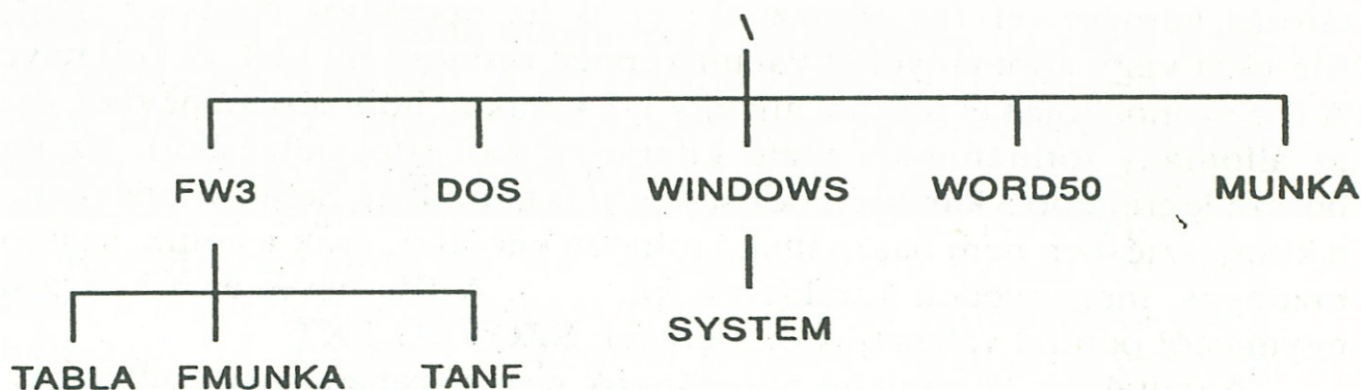
Az adatainkat, a programokat tartósan a háttértárolókon tároljuk. Állománynak (angol nevén file-nak) nevezzük azt a logikai tárolási egységet, amelyet egy feldolgozás során együtt kell kezelni: egy program utasításai, egy feladathoz tartozó adatok, vagy amit a felhasználó, a file készítője összetartozónak ítél. Ezen fogalom bevezetésével nincs szükségünk arra, hogy a fizikai tárolás kérdéseivel foglalkozzunk, ez is az operációs rendszer dolga. A file-okat vagy állományokat valamiképpen azonosítani kell: el kell nevezni. A file azonosítója két részből áll: egy 1-8 karakter hosszúságú névből, és egy, az állomány formátumára utaló kiterjesztésből (típusjelzésből). Ez utóbbi hossza legfeljebb 3 karakter, de esetleg el is maradhat. Sem a filenévben, sem a kiterjesztésben nem használható minden karakter, csak a betűk, számok és bizonyos, megengedett karakterek (pl. -, _). A file nevét és a kiterjesztést egymástól ponttal választjuk el. Például: SZOVEG.TXT

Az általunk létrehozott állományok nevét szabadon megadhatjuk, célszerű azonban olyan nevet választanunk, amely utal a tartalmára. A kiterjesztést rendszerint nem mi, hanem az a program rendeli hozzá a file-névhez,

amellyel az állományt létrehoztuk. A .COM és az .EXE kiterjesztésű állományok futtatható, gépi kódú file-ok. A .BAT típusjelzéssel ellátott file-ok is futtathatók, mert az operációs rendszer parancsait és különleges, ún. batch parancsokat tartalmazó parancsfile-ok. A legtöbb program jónéhány állományból áll, amelyekre a futás során szükség van. Nevüket, kiterjesztésüket ne változtassuk meg!

Az állományok legfontosabb jellemzőinek (azonosító, méret, a létrehozás vagy az utolsó módosítás dátuma és időpontja) nyilvántartása a lemezen tartalomjegyzékben történik. A lemezen formázáskor létrejön egy kötött méretű táblázat, a directory vagy főtartalomjegyzék, amelynek mérete csak a lemez típusától függ: a 360Kb-ajt kapacitású lemezé 112, az 1,2Mb-ajt kapacitásúé 224, a winchesteré kapacitástól függetlenül 512 bejegyzés elhelyezésére ad lehetőséget.

Egy bejegyzés file-ra, vagy további tartalomjegyzékre vonatkozhat. A különböző programrendszerek, felhasználók állományainak elkülönítésére pedig további tartalomjegyzékek hozhatók létre, amelyek file-szerű tulajdonságokkal is rendelkeznek: nincs fix méretük, hanem a file-okhoz hasonlóan bővíthetők/csökkenthetők, ha bennük újabb bejegyzéseket helyezünk el, továbbá azonosítóval rendelkeznek ugyanolyan szabályok szerint, mint az állományok. A kiterjesztésnek ebben az esetben azonban nincs semmi jelentősége. Egy ilyen tartalomjegyzéknek is lehetnek újabb tartalomjegyzékekre vonatkozó bejegyzései. Az ily módon kialakított struktúra a főtartalomjegyzékből kiindulva egyre szétágazóbb ugyanúgy, mint egy fa. Ezért fastruktúrának is nevezik (1-4. ábra).



1-4. ábra

A hétköznapi életben is találkozunk hasonló felépítésű nyilvántartó rendszerekkel. Mindenki által ismert a könyvtár katalógusrendszere, ahonnan a file-ok nyilvántartó rendszerét könyvtári struktúrának, a főtartalomjegyzéket főkönyvtárnak vagy gyökérkönyvtárnak, a további, általunk létrehozott tartalomjegyzékeket alkönyvtárnak nevezzük.

Az állományok azonosításához ilyen körülmények között azt is meg kell adni, hogy melyik könyvtárban található. A könyvtár kiválasztásához nem elegendő csak a név megadása, mert a könyvtári struktúrában más, ugyanilyen nevű alkönyvtár is létezhet. Azt kell megmondani, hogyan, milyen útvonalon lehet eljutni a kívánt alkönyvtárba.

Útvonalon az útbaeső könyvtárak nevét kell rendre felsorolni egymástól \ jellel elválasztva, pl. FW3\FMUNKA. Mivel kijelölhető lemezegységként egy olyan könyvtár, amelyben dolgozni akarunk, vagyis amelyben azok a file-ok találhatóak, amelyekre munkánk során éppen szükségünk van, ezért ezt aktuális könyvtárnak nevezzük. Az útvonal kétféleképpen adható meg: az aktuális könyvtártól és a gyökérkönyvtártól kezdődően. Például: ha az FMUNKA könyvtárban levő állományt keresem, az alkönyvtár azonosítója a gyökérkönyvtártól \FW3\FMUNKA, ha azonban az FW3 az aktuális könyvtár, elegendő annyit írni: FMUNKA. Az első esetben a \ jel a gyökérkönyvtárat jelöli: így nevezzük ezt a minden használható lemezen megtalálható tartalomjegyzéket. Tehát ha egy könyvtári ágon lefelé található alkönyvtárról van szó, elegendő csak az aktuális könyvtártól feltüntetni az útbaeső könyvtárak neveit abban a sorrendben, ahogyan egymás után következnek. Ha azonban egy teljesen más könyvtári ágon levő alkönyvtár file-jait keressük, csak a gyökérkönyvtártól megadott útvonal használható. Egyébként ez mindig egyértelműen kijelöli a keresett alkönyvtárat, függetlenül attól, hogy melyik éppen az aktuális könyvtár.

Az állományok azonosításához tehát a lemezegység azonosítóján, a file nevéen és a kiterjesztésén kívül hozzátartozik az is, hogy melyik könyvtárban található. Ez utóbbi nemcsak az alkönyvtár nevének megadását jelenti, hanem a hozzá vezető útvonalét, ahogy a könyvtárak azonosításánál leírtuk.

P1.: C:\FW3\FMUNKA\SZOVEG.TXT

Az azonosítóból elmaradhat a lemezegységneve, ha az aktuális lemezegységen keressük a file-t, valamint az útvonal, ha az az aktuális könyvtár.

A gép üzembehelyezése, a DOS környezet felépítése

A számítógép berendezéseinek összehuzalozása után a bekapcsolás következik: először a perifériáké, majd az alapgépé. (Kikapcsoláskor fordítva.) Csak földelt hálózatot szabad használni!

Gépeinken rendszerint az egyfelhasználós DOS operációs rendszert használjuk. Ezt a programrendszert eredetileg az amerikai szoftverház, a Microsoft fejlesztette ki az IBM személyi számítógépeihez. A cég azonban megtartotta magának az operációs rendszer forgalmazói jogát, így alakult ki az a helyzet, hogy ha eredeti IBM PC-t veszünk, azzal PC-DOS operációs rendszert kapunk, ha pedig csak egy azzal kompatibilis gépet, akkor általában MS-DOS-t. Bár a két cég a saját operációs rendszerét egymástól függetlenül fejleszti tovább, az eltérések nem a leggyakrabban használt parancsoknál tapasztalhatók. Ráadásul az egyes fejlesztési fázisokat jelentő ún. verziók a korábbiakhoz képest rendre hasonló módosításokat tartalmaznak. E fejezetben a jelenleg legáltalánosabban használt 3.30 verziót vettük figyelembe a legfontosabb parancsok ismertetésénél.

Valahányszor bekapcsoljuk a PC-t vagy megnyomjuk a RESET gombot, inicializáljuk a gépet, amely ily módon egy pontosan meghatározott kiindulási állapotba kerül. Ehhez a processzor a ROM-ban tárolt rutinokat dolgozza föl, amelyek valamennyi PC-ben a BIOS-ban található, s felelnek azért, hogy a processzor számára az összes külső egység üzemképesen inicializálódjék. Amint a számítógép belül már teljesen működőképesé vált, azaz valamennyi egység, például a RAM, a merevlemez és az összes csatolóáramkör üzemképes, akcióba lép a betöltő program (bootstrap loader). Ha ezzel egyidőben az A: hajlékonylemez egységben van floppy, akkor a meghajtó beolvassa az operatív tárba a hajlékonylemez első szektorát. Ha a lemezegységben éppen rendszerlemez van, tehát egy operációs rendszer boot-rekordjáról van szó, akkor ez intézi a betöltési folyamatot.

Ha a bootrekord megkapja a vezérlést, akkor – a gyorsaságtól eltekintve – nincs többé különbség a floppys és a merevlemez indítás között. A bootrekord a rábízott adatterületek alapján meghatározza a gyökérfájelgyár tartalomjegyzékének a helyét, ellenőrzi, benne van-e a két rejtett operációsrendszer-file: az PC-DOS esetén az IBMBIO.COM és az IBMDOS.COM (MS-DOS esetén IO.SYS és MSDOS.SYS). Ha a bootrekord mindkét bejegyzést megtalálja, akkor betölti az IBMBIO.COM (ill. IO.SYS) file-t az operatív tárba. Ez a file tölti be az IBMDOS.COM-ot (ill. MSDOS.SYS-t), feldolgozza a CONFIG.SYS-t,

végül betölti a COMMAND.COM parancsfeldolgozót is. Ez dolgozza fel utolsó lépésként a megfelelő környezetet beállító AUTOEXEC.BAT parancsfile-t, majd kiadja a képernyőre a promptot jelezve, hogy az operációs rendszer betöltése befejeződött.

Parancsok

Az operációs rendszer betöltését követően megjelenik a **prompt** (pl. C:\>), amely tartalmazza annak a lemezegységnek az azonosítóját, ahonnan a betöltés történt: A, ha floppyról, C, ha winchesterről. A prompt után írjuk a **parancsot**. A parancs beírását az **ENTER** billentyű leütésével fejezzük be, ekkor megkezdődik a parancs végrehajtása. A parancs általános formája: **parancsszó paraméterek/opciók** A parancsszó után szóközt kell ütni. A parancs kis- vagy nagybetűkkel egyaránt írható, mert a parancs értelmezésekor a karaktereket nagybetűssé alakítja a parancsfeldolgozó.

A leggyakrabban használt parancsok programjai már az operációs rendszer betöltésekor az operatív tárba kerülnek, így hívásukkor azonnal végrehajthatók. Ezeket ezért **belső parancsoknak** nevezzük. A többi, a **külső parancsok** az azonos nevű programállományokban találhatóak. Hívásukkor az operációs rendszernek először meg kell keresnie, be kell töltenie a programot az operatív tárban, s csak ezután indulhat a végrehajtása.

A következőkben helyhiány miatt csak egészen röviden ismertetjük, inkább csak utalunk a leggyakoribb parancsokra. Felhívjuk azonban a figyelmet az irodalomjegyzékben ajánlott, az operációs rendszerrel részletesen foglalkozó könyvekre.

File-kezelő parancsok

TYPE

A szöveges állományok tartalmának kifrása a képernyőre. A parancs egyetlen paramétere a kilistázandó file azonosítója, amely *, ? karaktereket nem tartalmazhat, így csak egyetlen állományra vonatkozhat.

P1.: TYPE C:\FW3\FMUNKA\SZOVEG.TXT

REN

Az állományok nevének, kiterjesztésének megváltoztatása. A parancsnak két paramétere van: az első a megváltoztatandó file régi (létező) azonosítója, a második az új név, kiterjesztés. Ha nem az aktuális lemezegységen és/vagy azon az aktuális könyvtárban keressük az állományt, akkor az erre vonatkozó file-azonosító rész (lemezegységneve, útvonal) csak az első paraméternél szerepelhet. Ha a file kiterjesztését nem akarjuk megváltoztatni, a második paraméterként megadott file-azonosító kiterjesztéseként * karaktert is írhatunk.

```
Pl.: REN C:\FW3\FMUNKA\SZOVEG.TXT VIZSGA.*
```

DEL

Állományok törlése. A parancs egyetlen paraméterként kell megadni a törlendő állomány azonosítóját. Ha több file-t akarunk törölni, az azonosító file-csoportot is kijelölhet, ha *, ? karaktereket tartalmaz a megfelelő helyeken.

A parancs a törlést megerősítésre való várakozás nélkül elvégzi, kivéve, ha DEL *.* parancsot adunk. Ez az egyetlen eset, amikor a parancs visszakérdez ARE YOU SURE (Y/N)? megerősítésre várva. Formailag eltérő ugyan pl. a DEL FW3 parancs, mégis megkapjuk az ellenőrző kérdést. Előző példáinkban láttuk, hogy FW3 egy alkönyvtár neve, így a parancs jelentése: töröld az FW3 alkönyvtár összes file-ját!

```
Pl.: DEL C:\FW3\FMUNKA\SZOVEG.TXT  
DEL C:\FW3\FMUNKA\*.TXT
```

COPY

File-ok másolása. A parancs első paramétereként kell megadni azt, hogy milyen állományokat másolunk (*, ? használható), a második paraméterben pedig azt, hogy hová. Ha az aktuális lemezegység aktuális könyvtárába másolunk az eredetivel azonos néven, elegendő csak az első paramétert megadni. A parancs hibaüzenettel jelzi, ha a file-t önmagára próbálnánk másolni.

```
Pl.: COPY C:\FW3\FMUNKA\SZOVEG.TXT C:\TANF  
COPY C:\FW3\FMUNKA\SZOVEG.TXT
```

A paraméterek értelemszerűen foglalt eszköznevek is lehetnek. Ezek közül a legfontosabbak: CON (konzol: bemeneti eszközként a billentyűzetet,

kimenetként a monitort értjük alatta), LPT1 vagy PRN (az első párhuzamos nyomtató), AUX vagy COM1 (első soros kimenet). Így a nyomtatás elvégezhető a COPY C:\FW3\FMUNKA\SZOVEG.TXT LPT1 paranccsal, vagy szöveges file készíthető a COPY CON LEVEL.TXT paranccsal.

XCOPY

File-ok, könyvtárak másolása. A parancs a COPY-val azonos formájú. Előnye, hogy a másolandó állományokat nem egyenként olvassa be a memóriába és írja ki a másolati lemezegységre, mint a COPY, hanem amennyit csak tud. Ez különösen akkor gyorsítja meg jelentősen a másolás idejét, ha sok file-t kell másolnunk floppyról floppyra és csak egyetlen hajlékonylemez egységünk van.

Ha egy könyvtár teljes tartalmát akarjuk átmásolni az alkönyvtárai-val együtt, a /S opcióval megtehetjük. Ha ehhez még a /E-t is használjuk, akkor még az üres alkönyvtárakat is létrehozza a másolati helyen.

Pl.: mint a COPY-nál

ATTRIB

File-attributumok (tulajdonságok) lekérdezése, megváltoztatása. A lehetőségekből csak arra mutatunk példát, hogyan tehetjük írásvédetté a file-unkat. Az írásvédett állomány nem törölhető, tartalma nem módosítható, de a file neve és/vagy kiterjesztése megváltoztatható. Ha írásvédett file-t másolunk, a másolat már nem rendelkezik ezzel a tulajdonsággal, azaz törölhető, tartalma változtatható.

Pl.: ATTRIB +R C:\FW3\FMUNKA\SZOVEG.TXT

Az írásvédelem megszüntetéséhez +R helyett -R-t írunk, a lekérdezéshez semmit. A file-azonosítóban *, ? is használható.

Alkönyvtárak használata

DIR

Egy tartalomjegyzék lekérdezése. Paraméter nélkül az aktuális lemezegység aktuális könyvtárának tartalmát jelzi ki a képernyőn: a táblázatban feltünteti a file-ok nevét, kiterjesztését, méretét, valamint létrehozásának vagy utolsó módosításának dátumát és időpontját. Azok a sorok, amelyekben a méret helyén a <DIR> karaktersorozat látszik, nem file-okra, hanem könyvtárakra vonatkoznak.

A parancsnak egyetlen paramétere lehet: egy file-azonosító, amely * és ? karaktereket is tartalmazhat. Ilyenkor ennek az azonosítónak megfelelő file-(ok)ra vonatkozó tartalomjegyzék sor(oka)t kapjuk a képernyőn.

P1.: DIR C:\FW3\FMUNKA*.TXT

Ha csak útvonalat adunk meg, az így elérhető könyvtár tartalomjegyzékét írathatjuk ki.

P1.: DIR C:\FW3

A parancs végére a /P opciót írva, a hosszú listát képernyőnyi oldalakra tördelve jeleníthetjük meg. A /W opcióval a leírtnál tömörebb lesz a lista: csak a file-ok, könyvtárak nevét és kiterjesztését tünteti fel a gép, de egy sorba nem csak egyét, hanem öt bejegyzését írja.

P1.: DIR C:\FW3 /W

MD

Alkönyvtárak létrehozása. A parancs paramétere az útvonal, amelynek végén a létrehozandó könyvtár neve áll.

P1.: MD C:\FW3\TABLA

CD

Aktuális könyvtár kijelölése, lekérdezése. Paraméterként megadható a lemezegységnev, valamint az az útvonal, ahogyan a kérdéses könyvtárhoz eljuthatunk. A lekérdezéshez csak a lemezegység nevét írjuk a parancsszót követő szóköz után, de el is maradhat, ha az aktuális lemezegységről van szó. Minden lemezegységen egy aktuális könyvtár jelölhető ki ezzel a paranccsal, lehetővé téve a különböző lemezegységekben levő lemezek az alkönyvtárak közti műveletekhez a legegyszerűbb, legrövidebb parancsok használatát.

P1.: CD A:

CD C:\FW3\FMUNKA

RD

Alkönyvtár törlése. Paraméterként a lemezegység nevét, valamint a törölni kívánt könyvtárhoz vezető útvonalat adjuk meg. Csak üres könyvtár törölhető. Nem törölhető az aktuális könyvtár és természetesen a főkönyvtár (gyökérkönyvtár).

P1.: RD C:\FW3\FMUNKA

Lemezkezelő parancsok

Az aktuális lemezegység megváltoztatása

A művelet elvégzéséhez nincs parancsszó, csak a lemezegység nevét kell megadni, a végén kettősponttal.

P1.: C:

FORMAT

Lemezformázás. A lemez fizikai ellenőrzése, sávok, szektorok kialakítása: a lemez használhatóvá tétele.

Paraméterként meg kell adni a lemezegység nevét. Általában új hajlékonylemezeket formázunk. Opció nélkül a lemezegység mindig a lehető legnagyobb kapacitású lemezt készíti. Ha azonban az 1,2 Mbájtos lemezegységben a DD vagy 2D jelű lemezt formázunk, a /4 opciót is használjuk: ekkor a formázott lemez kapacitása 360 Kbájt lesz.

A parancs minden adatot, programot töröl a lemezről! Ezért – ha nem üres – formázás előtt győződjünk meg arról, hogy a lemez tartalmára, a továbbiakban valóban nem lesz szükségünk.

A formázást követően a parancs a lemezre rámásolja az operációs rendszer betöltéséhez szükséges file-okat is a /S opcióval.

P1.: FORMAT A: /S/4

DISKCOPY

Teljes lemez másolása. A másolat fizikailag azonos lesz a másolandó lemezzel: az egyes sávok, szektorok tartalma ugyanaz lesz mindkét lemezen, ezért csak azonos formátumú lemezek között használható. Ez az egyetlen parancs, amely formázatlan lemezt is elfogad, s másolás előtt a szükséges formájúra formázza.

P1.: DISKCOPY A: B:

CHKDSK

Adatkiírás a lemezen elhelyezkedő állományokról, könyvtárakról, a lemez tartalmáról. A /V opcióval a lemezen levő összes állomány azonosítóját kifírja.

Pl.: CHKDSK C:*.* /V

LABEL

Lemeznév megadása, megváltoztatása, lekérdezése. A lemez neve – a file-ok, könyvtárak azonosítójához hasonlóan – a 8 karakteres névből és a 3 karakteres kiterjesztésből származó, összesen 11 karakterből állhat. A név a parancs indítása után a képernyőn megjelenő kérdés után frandó, ENTER-rel lezárva.

Pl.: LABEL

Programindítás

A program (.COM vagy .EXE kiterjesztésű file) indításához elegendő a file nevét begépelni. Ekkor a gép megkeresi az aktuális lemezegység aktuális könyvtárában. Ha nem találja, még a PATH paranccsal megadott útvonalakon is. Ha nem akarjuk igénybe venni ezt a lehetőséget, a file neve előtt beírhatjuk a lemezegység nevét, valamint azt, hogy a file-hoz milyen könyvtárakon keresztül juthatunk el. Ebben az esetben csak a megadott helyen keresi a programot az operációs rendszer.

Pl.: WORD

ha a WORD50 az aktuális könyvtár, vagy ha nem, akkor korábban kiadtunk pl. egy PATH C:\WORD50 parancsot.

vagy:

C:\WORD50\WORD

A DOS környezettel kapcsolatos parancsok

PATH

Futtatható állományok végrehajtásához keresési útvonalak megadása, lekérdezése.

A megadott útvonalak a futtatható (.COM, .EXE, .BAT) állományok feldolgozását egyszerűsítik: a program, illetve a parancsfile futtatásához

nem csak az aktuális lemezegység aktuális könyvtárában keresi a file-t, hogy betölthesse a memóriába, hanem a PATH parancsban megadott útvonalakon is. A megadott keresési útvonalak mindaddig érvényesek, amíg a gépet ki nem kapcsoljuk vagy újabb PATH parancsot nem adunk ki.

Pl.: PATH C:\;C:\DOS;C:\FW3

PROMPT

A parancsvárakozási jel módosítása. A parancs paramétereként megadható szabad szöveg és/ vagy a rendszerből vett adat (dátum, időpont, aktuális lemezegység, aktuális lemezegység aktuális könyvtára stb.). Az utóbbiak megadásához speciális karakterpár szükséges, amelynek első tagja \$, a második a kívánt adathoz kapcsolt kötött karakter, pl.:

- t idő
- d dátum
- p aktuális lemezegység és aktuális könyvtára
- n aktuális lemezegység
- g jel

Pl.: PROMPT \$p\$g

a szokásos C:\ prompt beállításához

VER

A működő operációs rendszer verziószámának lekérdezése. Az operációs rendszer módosított, javított változatait a verziószámok különböztetik meg. Minél nagyobb ez a szám, annál későbbi változatról van szó. Bár itt 3.30- as parancsokat ismertetünk, jelenleg a legújabb az 5.0 verzió. A parancs üzenetében a verziószámon kívül megadja az operációs rendszer típusát is: pl. PC-DOS vagy MS-DOS-e. A használt programokat az idők során különböző operációs rendszer verziók alatt fejlesztették. Futtatni egy programot csak ugyanabba az operációs rendszer családba (pl. DOS) tartozó operációs rendszer alatt lehet, továbbá, ha a használt rendszer verziószáma legalább akkora, amely alatt kifejlesztették.

Pl.: VER

DATE

A rendszer dátum beállítása. A parancs indítása után megjelenő formátumban kell megadni a napi dátumot.

XT gépeken minden rendszerindítás után be kell állítani. Az AT-ban a beállított dátum a gép kikapcsolása utáni megőrzését akkumulátor biztosítja.

Pl.: DATE

TIME

A rendszeridő beállítása. A parancs indítása után megjelenő formátumban kell megadni a pontos időt. Elegendő csak az órát és a percet beírni.

XT gépeken minden rendszerindítás után be kell állítani. Az AT-ban a beállított időpont a gép kikapcsolása utáni megőrzését akkumulátor biztosítja.

Pl.: TIME

A "kényelmes" DOS környezet

A betöltésnél említett AUTOEXEC.BAT parancsfile-ba gyűjthetjük azokat a parancsokat, programindításokat, amelyeket minden rendszerindítás után magunknak kellene rendre elvégeznünk. A file előállítását valamilyen szövegszerkesztővel, vagy akár

COPY CON AUTOEXEC.BAT

paranccsal történhet.

A prompt és a path rendszerváltozók beállítását is természetesen az AUTOEXEC.BAT állományban oldjuk meg.

2. Microsoft Windows 3.0

Mi a Windows ?

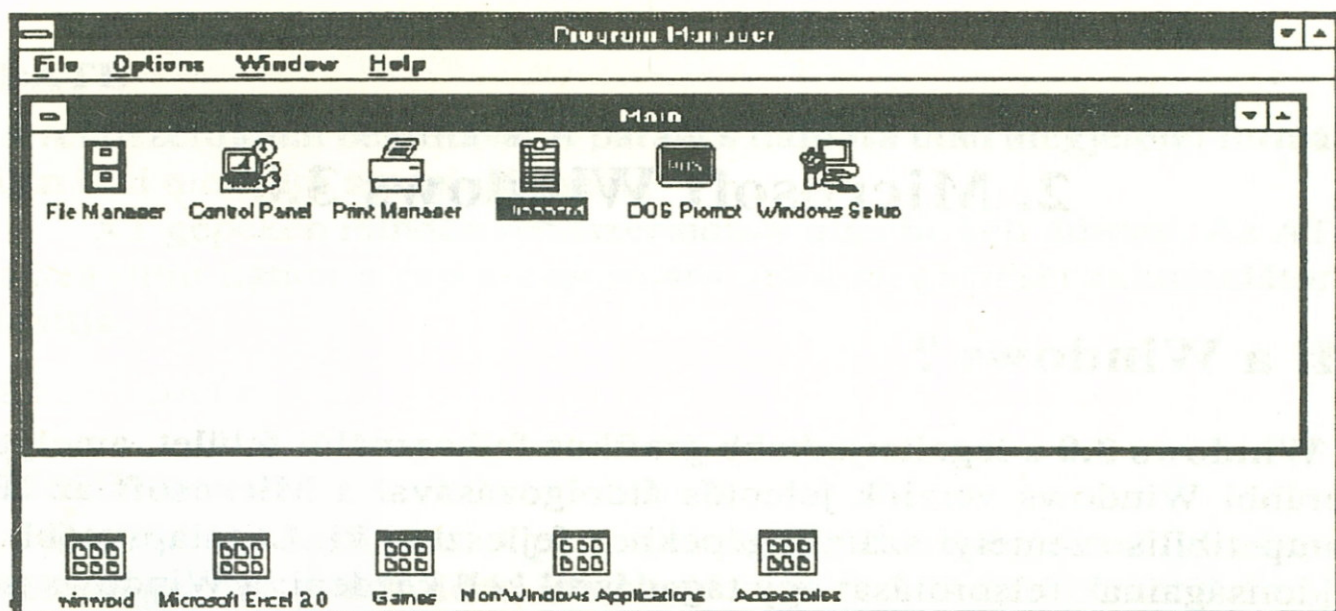
A **Windows 3.0** a legelterjedtebb grafikus felhasználói felület, amelyet a korábbi Windows verziók jelentős átdolgozásával a **Microsoft** az **IBM** kompatibilis személyi számítógépekhez fejlesztett ki. Legalapvetőbb tulajdonságainak felsorolását egy tagadással kell kezdeni: a Windows **nem operációs rendszer**: saját parancs-processzorral nem rendelkezik. Működéséhez szüksége van az operációs rendszerre, ez a Windows esetén a **DOS** vagy az **OS**. Arra épül rá, felhasználja annak szolgáltatásait. Miért volt szükség ezeknek a korábban olyan korszerűnek és könnyen kezelhetőnek tekintett operációs rendszereknek a "megfejelésére"? Az ok a felhasználói kör alapvető átalakulásában rejlik. A korábbi években a számítógépet általában számítástechnikusok kezelték, tőlük az ilyen parancsok:

```
C:\ copy c:\Frame\feladat\Minta.FW3 a:\Frame\munka
```

hibamentes, szintaktikailag korrekt, egy betűközt sem vétő begépelése elvárható volt. Napjainkban azonban az alkalmazók nagyobb része nem számítástechnikus: kereskedő, menedzser, jogász, közgazdász, adminisztrátor, aki igényli, hogy a rendszer zárja ki, vagy legalábbis minimalizálja a téves parancsmegadást, tegye könnyebbé a programok és a rendszer kezelését. Ennek a követelménynek tesz eleget a **Windows**.

A Windows felépítése

A program neve: **Windows**, ablakokat jelent. A futtatandó programok, valamint a rendszer bemenő és kimenő üzenetei jellegükben és kezelésükben hasonló felépítésű ablakokban jelennek meg, ahogy ez a *2-1. ábrán* látható.



2-1. ábra

A rendszer fő ablaka a **Program Manager**, amelyben további ablakok találhatóak kinyitva, mint a **Main**, vagy becsukva, jelképpé, úgynevezett **ikonná** zsugorítva, mint pl: az **Accessories**. A Windows által alapértelmezésben felkínált ablakokon kívül a felhasználó saját ablakot (ablakokat) is létrehozhat.

A rendszer indítása

Amennyiben a rendszert már előzőleg üzembehelyezték, a számítógép bekapcsolása és a DOS bejelentkezése után gépeljük be a **WIN** parancsot.

A rendszer ennek hatására a konfiguráció által megszabott legmagasabb üzemmódba áll be, vagyis a gép lehetőségeit maximálisan kihasználja. Megjelenik a Windows címlapja a verziószámmal és néhány másodperc múlva a **Program Manager**, valamint alapértelmezésben a **Főmenü** (Main) tartalmazó ablak. Ezt szemlélteti a 2-1. ábra.

Az egér használata

A Windows ugyan billentyűzettel is vezérelhető, de a kényelmes munkához az "egér" nélkülözhetetlen. Az egér tulajdonképpen egy elektronikus mutatópálca, amivel kijelölhetjük a kívánt ablakot, a végrehajtandó programot, menüt jeleníthetünk meg vele stb.

Az egerek általában két vagy három nyomógombbal rendelkeznek. A Windows általában a baloldali nyomógombot használja.

Az egeret az asztallapon vagy az egéralátéten mozgatva, változtatni tudjuk a kurzor helyét a képernyőn. Amikor a kurzor a kívánt pozícióba került, nyomjuk le a nyomógombot, ezzel kiválasztottuk a kívánt elemet vagy funkciót.

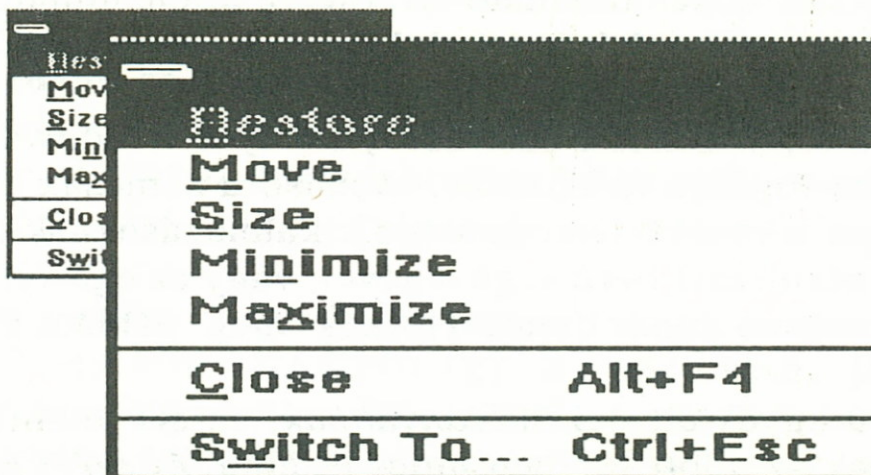
Például mutassunk rá a kurzorral a **Program Manager** ablak jobb felső sarkán található, lefelé mutató nyílhegyre és nyomjuk meg a gombot. Erre az ablakunk a jobb áttekinthetőség végett jelképpé, ikonná zsugorodik, anélkül, hogy funkciója megváltozott volna.

Ilyen ikonok láthatók az 2-1. ábra alján.

Az ikonra rámutatva és a nyomógombot kétszer lenyomva ismét "kinyílik" az ablak.

Ha egy programot akarunk indítani például a pontos időt mutató **Clock**-ot, kinyitjuk az **Accessories** ablakot (vagy rákattintunk, ha már ki volt nyitva) és kétszer rákattintunk az egérrel a **Clock** ikon alatti szövegre.

Amíg még gyakorlatlanok vagyunk a kettős kattintásban, a program indítása helyett sokszor az ún. **Vezérlő (Control)** menü jelenik meg: (2-2. ábra):



2-2. ábra

A Vezérlő menü segítségével a programot a **Restore** parancsra kattintva lehet indítani.

A Program Manager

A **Program Manager** vezérli a Windows működését. Ameddig Windows alatt dolgozunk, akár látjuk az ablakát, vagy az ikonná zsugorított képét, akár nem, a Program Manager vezérli a munkát. Segítségével kiválaszthatjuk az éppen végrehajtandó programcsoportot, illetőleg programot.

A Program Manager ablak-képével érdemes részletesebben is megismerkedni, mivel felépítése és kezelése a többi ablakkal azonos szerkezetű.

Az ablak mérete az egerrel változtatható, úgy, hogy a mutató nyilat az ablak oldalára vagy sarkára helyezzük (*ha a beállítás jó a kurzor kettős nyíl alakot vesz fel*), és a kívánt irányba elmozgatjuk az egeret. Hasonló technikával tudjuk az ablakot mozgatni a képernyő felületén az ablak nevére kattintva az egerrel.

Az ablak neve a felső sorban (címsor) olvasható. A sor bal szélén található "--" jel kiválasztáskor (használjuk az egeret), vagy **Alt + szóköz** lenyomásával a 2. ábráról már ismert Vezérlő menü jelenik meg, aminek **Close** parancsával tudjuk lezárni az ablakot, vagy kilépni a Windows-ból. Ha meggondoltuk magunkat és tovább akarjuk folytatni a munkát, a **Vezérlő** menüt el tudjuk tüntetni az eger - menün kívüli - egyszeri kattintásával. Ez minden menüre igaz, amit be akarunk "csukni".

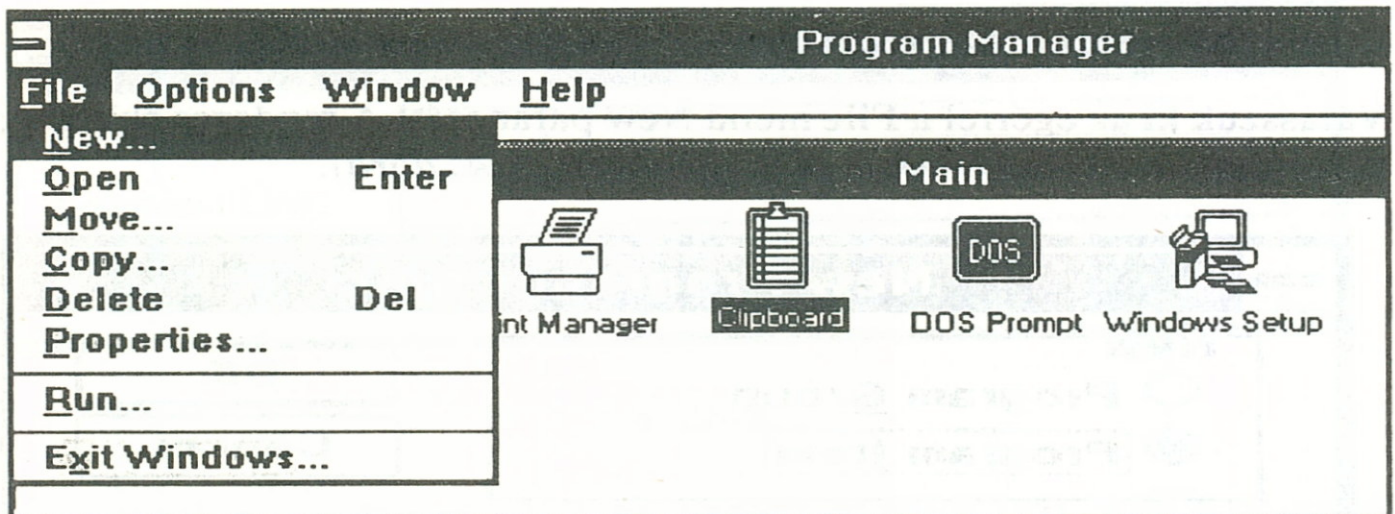
A címsor jobboldalán található fel, illetve lefelé mutató nyíl az ablak nagyítását, illetve zsugorítását teszi lehetővé. Nagyított kép esetén a látható kettős nyíl az eredeti méretet állítja vissza.

A címsor alatt található a **menüsor**, amely a Windows által használható parancsokat foglalja rendszerbe. Menüsora nemcsak a Program Managernek, hanem a később ismerttetendő alkalmazásoknak is van. Az **Alt** billentyű és az **aláhúzott betű** segítségével, vagy az egerrel lehet a menüt kinyitni, más néven legördíteni. Nézzük meg például a **File** menüt (2-3. ábra).

Ha a menü egyes elemei *halványabbak*, ez azt jelenti, hogy az így jelölt parancsnak az adott esetben nincs értelme és ezért *ki sem választható!*

A menükészlet eleme utáni "..." azt jelenti, hogy az elem egy **dialógusablakban** kér további információt, amint ezt hamarosan az új csoportablak létrehozásánál látni fogjuk.

A menüelem előtti "pipa-jelzés" az elem aktív voltát jelzi. Az utána levő billentyűkombináció, pl: az **Open** mellett az "Enter", a billentyűvel való kiválasztás lehetőségét mutatja.



2-3. ábra

A **File** menü segítségével tudunk új csoportablakot létrehozni, illetőleg abba programot helyezni a **New** paranccsal, vagy már létezőt programot megnyitni az **Open**-el.

Ez teszi lehetővé egy program megkeresését (**Find**) és futtatását (**Run**), esetleg törlését (**Delete**). A **Move**, illetve **Copy** paranccsal a kiválasztott programot áthozhatjuk vagy átmásolhatjuk egy másik csoportablakba. Az **Exit Windows** segítségével kiléphetünk a **Windows**-ból.

A **Window** menü segít a nyitott ablakok és az ikonok rendezésével áttekinthetővé tenni az "íróasztalunkat", vagyis a képernyőt.

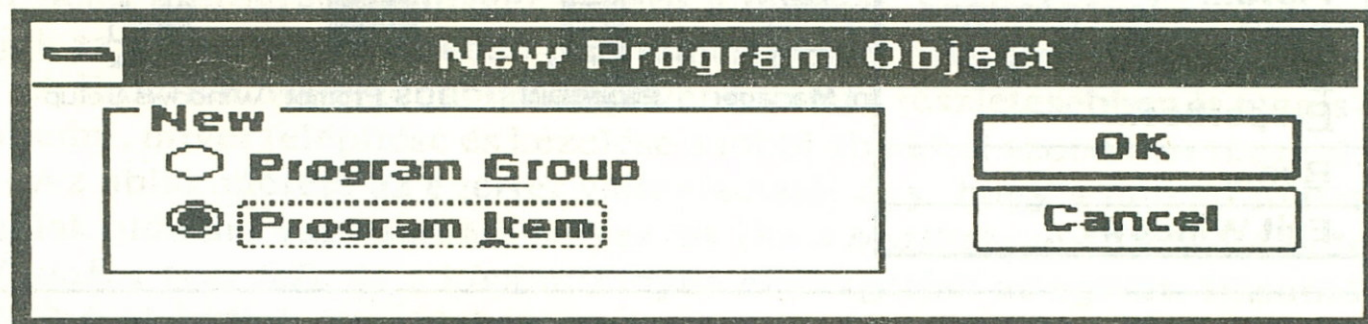
A **Help** menü, illetve az általa lehívható bő kézikönyvi információ azoknak segít a **Windows** működésének és kezelésének megértésében, akik járatosak az angol nyelvben.

Ha egy ablak tartalma a keret megszabta méretet meghaladja, akkor az ablak jobb és/vagy alsó szegélyén egy ún. gördítősáv jelenik meg, amelynek kezdeti és végpontját nyilak jelölik. A gördítősávon mozogva az egérrel az ablakban a szöveg kívánt része jeleníthető meg. A gördítősáv technikáját a **Windows** és a **Windows** alatti programok széleskörben alkalmazzák.

A menüsor alatt (kinyitva vagy ikonná zsugorítva) a csoportablakok találhatóak (pl: **Main**, **Accessories** stb.) A csoportablakok futtatandó programokat, illetőleg dokumentumokat (pl: **Accessories\Write** illetve **Write**-ban írt szövegek), szolgáltatásokat (pl: **Main\DOS Prompt**) vagy rendszerbeállítási lehetőségeket tartalmaznak (pl: **Control Panel\Mouse**).

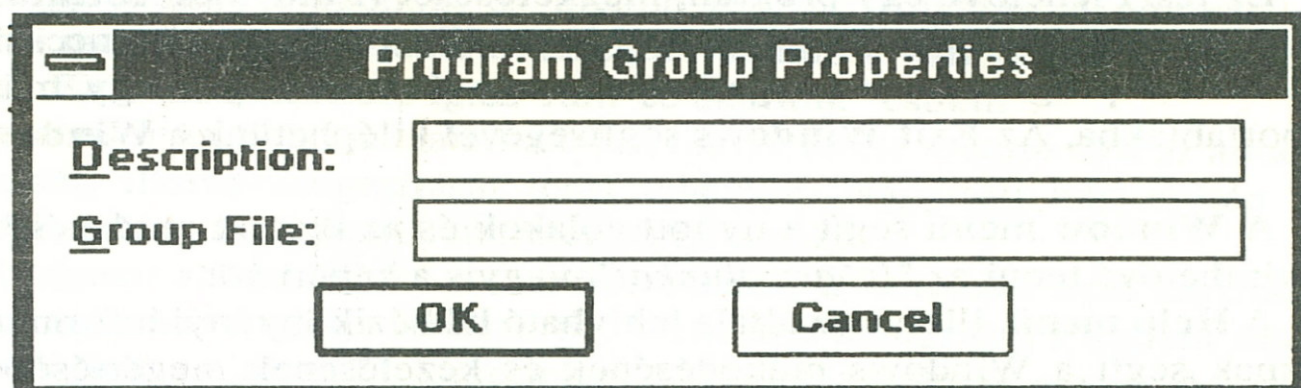
Új csoportablak létrehozása

Válasszuk ki az egérrel a File menü New parancsát! A rendszer ekkor egy dialógusablakban kér további információt (2-4. ábra).



2-4. ábra

Az egérrel válasszuk ki a Program Group-ot, majd az OK- t! Ekkor egy új dialógusablak jelenik meg (2-5. ábra).

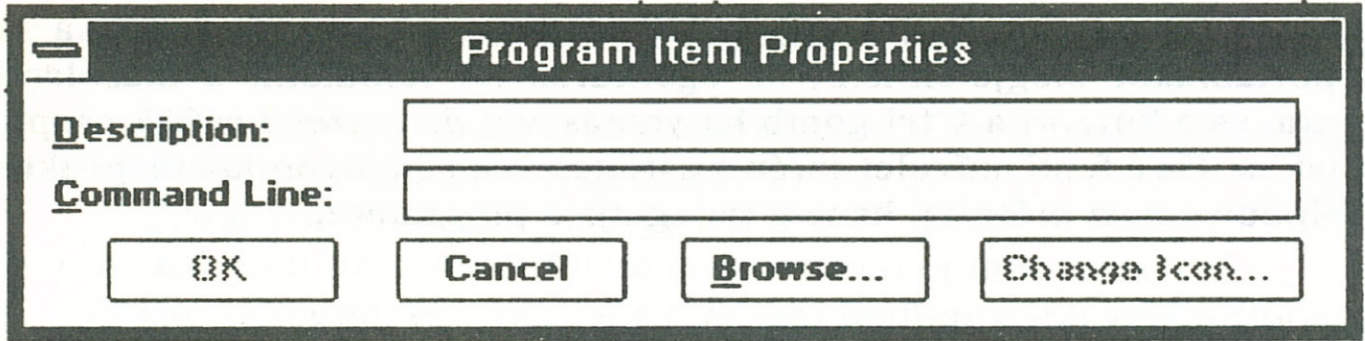


2-5. ábra

Írjuk be a Description mezőbe az új csoportablak nevét, pl: "SAJÁT" és OK! Ennek hatására megjelenik a SAJÁT nevű új csoportablak.

Vigyünk be egy programot az új csoportablakba!

Ismét válasszuk a File\New parancsot! A 4. ábra szerinti dialóguskeretben most a Program Item- et válasszuk ki és az O.K-t. A program nevét a rendszer a következő dialógusablakban kéri (2-6. ábra).



2-6. ábra

A **Description** mezőbe gépeljük be a program nevét (azt, amit a program ikonja alatt akarunk feltüntetni), pl: PONTOS IDŐ, majd az egérrel vagy a Tab billentyűvel belépünk a **Command Line** (parancs sor) mezőbe, és a **Browse** (böngészés) segítségével megkeressük a kiválasztott lemezegység adott könyvtárában azt az **.exe** kiterjesztésű file-t, ami a kívánt programot indítja, pl: C:\Windows\Clock.exe.

A keresést OK-val lezárva, megjelenik a SAJÁT nevű ablakban az általunk adott néven a kiválasztott program.

Mód van arra, hogy egy szűk készletből ikonmintát rendeljünk a programhoz (**Change Icon**).

Ha az ablak létrehozása vagy feltöltése során meggondoljuk magunkat, akkor válasszuk ki a **Cancel**-t, érvénytelenítve az előkészítő lépéseket. A **Cancel** parancsot akkor is használhatjuk, ha egy dialógusablakból ki akarunk lépni.

Ha a programot be akarjuk hívni, akkor az egérrel a kurzort az ikonra helyezzük, és a bal oldali gombot gyorsan kétszer egymás után lenyomjuk. Erre a kiválasztott program végerhajtása elindul.

A programból a vezérlő menü segítségével lehet kilépni.

A csoportablak alkalmazásának célja

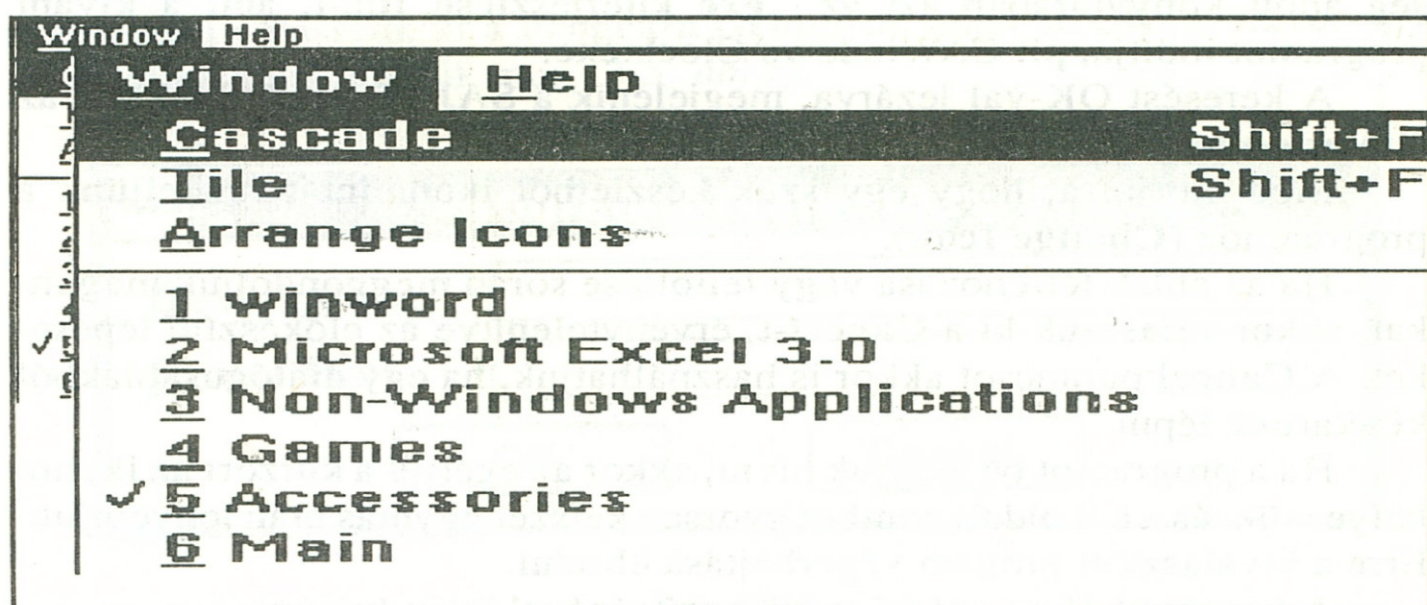
A csoportablak kialakításának technikája után vizsgáljuk meg, hogy milyen előnnyel jár annak alkalmazása. A felhasználó egy nagy rendszer bonyolult könyvtár-struktúrájának megkerülésével a számára fontos programokat, pl. egy szövegszerkesztőt, vagy a szövegszerkesztővel készített dokumentumot egy ablak keretébe foglalja, és a továbbiak során csak ezzel az ablakkal dolgozik. A saját csoportablak létrehozása után az adott ikon kijelölésével indítható a program. Ha egy programot az egyik csoportab-

lakból át akarunk másolni egy másik csoportablakba, elegendő a két csoportablakot megjeleníteni, az egérkurzorral rámutatni a másolandó ikonra, és a kurzort a Ctrl gomb lenyomásával *átvezetni* a másik csoportablakba. Ha a fenti művelet során a Ctrl gombot nem nyomjuk meg, akkor a Windows nem másolja, hanem mozgatja a programot.

Ablakok kiválasztása és rendezése

A képernyőn egyidejűleg több ablak is lehet nyitva vagy ikonná zsugorítva. Nagyszámú ablak esetén az áttekinthetőség csak az egyes ablakok becsukásával, és az ikonok rendezésével biztosítható. Az ablak kinyitásának és becsukásának a módja már ismert.

Rendezni a WINDOW menü segítségével lehet (2-7. ábra).



2-7. ábra

A **Cascade** kiválasztásával az ablakok egymás felett, de a címsorát nem takarva helyezkednek el. Ha a **Tile** elrendezést választjuk, akkor az ablakok csempe szerűen egymás mellé kerülnek úgy, hogy egymást ne takarják. Az ikonokat az **Arrange Icons**-al lehet elrendezni.

A 2-7. ábrán a korábban már beolvasott és most is a memóriában található ablakok nevei is láthatók, megjelölve az éppen aktívát. Az *aktív ablak címsorának színe eltér a többitől*. Ha most egy másik ablakot akarunk kiválasztani, akkor egyszerűen csak ki kell jelölnünk az egérrel

(az ablakon belül egyet kattintva), vagy a sorszámát beütni a számbillen-tyűzeten.

A felsorolásból látható, hogy a Windows nem tesz különbséget a zsugorított és a nyitott ablakok között.

A **Program Manager**, ahogy ez a 2-1. ábrán látható, többféle, a felhasználó által tovább bővíthető csoportablakot tartalmaz.

Main (Főmenü) programjai a rendszer legfontosabb vezérlőfunkcióit látják el.

Accessories (Tartozékok) több, hasznos segédprogramot tartalmaz.

Windows applications (Windows alkalmazások) ablakot a Windows alá kifejlesztett programok számára hozták létre, mint pl: az Excel, a Word for Windows vagy a Works for Windows használatához. Ezek a programok önálló termékek, *nem* a Windows részei, de installálásuk után a már ismertetett módon a csoportablakba helyezhetők, és felhasználják a Windows szolgáltatásait.

A **Non Windows Applications** (nem Windows alkalmazások) tetszőleges, a Windows-tól független programok számára létrehozott ablak. A Windows a beállított üzemmódtól függően több - kevesebb támogatást nyújt ezeknek a programoknak.

Games (Játékok) két játékprogramot tartalmaz.

A továbbiakban a Főmenü (Main) egyes elemeivel és a **Tartozékok** (Accessories) szolgáltatásaival ismerkedünk meg.

File Manager

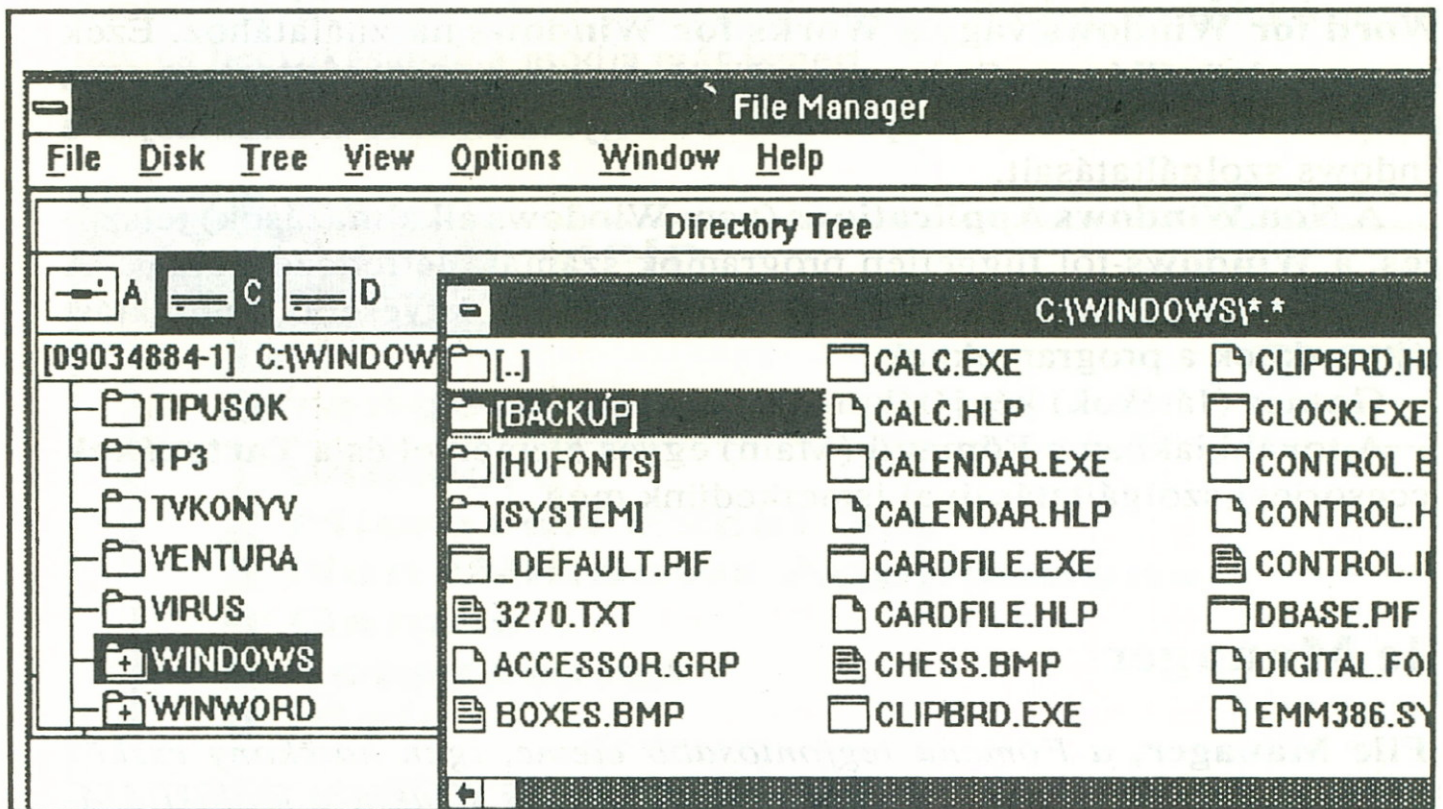
A **File Manager**, a Főmenü legfontosabb eleme, igen hatékony eszköz könyvtárak szervezésében, az áttekinthető, logikus struktúra kialakításában. Szolgáltatásai hasonlóak a DOS nyújtotta szolgáltatásokhoz, de kezelése sokkal egyszerűbb.

Vigyázat! a Program Manager-rel ellentétben, ha valamit meg gondolatlanul hajtunk végre, például kitörlünk egy állományt vagy egy könyvtárat, akkor a hiba a DOS-hoz hasonlóan már nem hozható helyre. A kezdő felhasználó számára a File Manager alatti munka ezért fokozott figyelmet kíván. Ha nem bízunk eléggé magunkban, jobb, ha csak a Program Manager szintjén dolgozunk.

A File Manager két ablaktípust kezel:

- a **Directory Tree** a meghajtókat és a kiválasztott meghajtó könyvtárstruktúráját mutatja,
- a **Directory Window** a kiválasztott könyvtár tartalmát mutatja.

A könyvtári struktúra segítségével kijelölhetjük a kívánt meghajtó ikonját, rákattintással megjeleníthetjük annak főkönyvtárát, majd tetszőleges könyvtári bejegyzésére állva, kettős kattintással az alkönyvtár állományait (2-8. ábra).



2-8. ábra

A könyvtári ikonban látható "+" jel mutatja, hogy alkönyvtár is tartozik hozzá, amit az egérrel lehet kinyitni. A bejegyzések ikonjai eltérőek könyvtár, programállomány, dokumentumállomány, vagy egyéb állomány esetén. A programállomány .com, .exe, .bat vagy .pif kiterjesztéssel rendelkezik.

A File Manager menüSORA

A 2-8. ábrán látható, hogy a Program Manager-hez hasonlóan a File Manager is rendelkezik egy menüSORral.

A File menü segítségével megnyithatunk (**Open**), futtathatunk (**Run**), nyomtathatunk (**Print**) vagy törölhetünk (**Delete**) állományokat. Az állománynév megadásával (*estleg * vagy ? karakterek alkalmazásával*) keresni lehet a rendszerben (**Search**).

A menü lehetővé teszi az állományok átnevezését (**Rename**), paramétereinek módosítását (**Change Attribute**), más könyvtárba való mozgatását (**Move**), másolását (**Copy**), vagy új könyvtár létrehozását (**Create Directory**). Az Associate parancs segítségével adott kiterjesztésű dokumentumokat (pl.:.wri) lehet az azokat kezelő .exe állományokkal (pl: Write.exe) összekapcsolni.

A DISK menüvel hajlékony lemezt formázhatunk (**Format**), másolhatunk (**Copy**), tehetünk rendszerlemezzé (**Make System Diskette**), fix és hajlékony lemezt címkézhetünk (**Label**).

A TREE menüvel a könyvtárstruktúra a kívánt mélységig jeleníthető meg.

A VIEW menüvel megadható, hogy a könyvtár tartalomjegyzékben milyen adatok szerepeljenek (pl. állomán név, méret, a készítés időpontja stb.), és milyen sorrend szerint listázzon.

Állományok gyors másolása és mozgatása

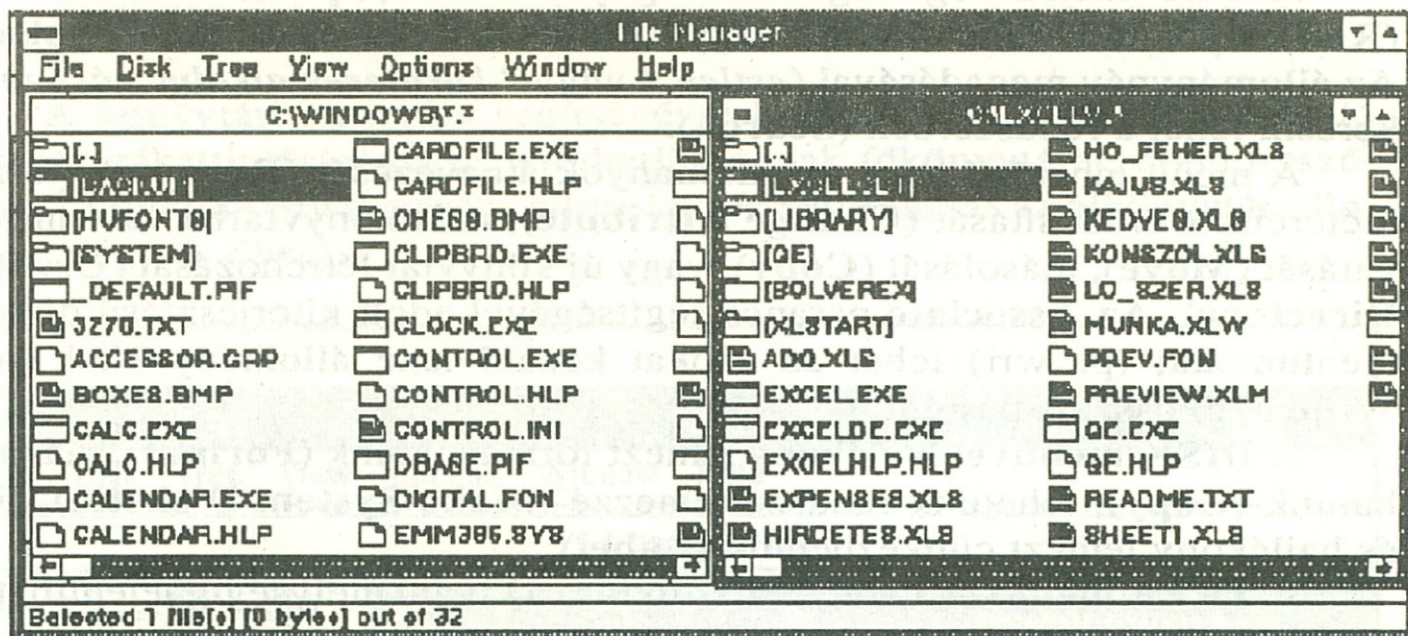
A rendszer az egér segítségével lehetővé teszi az állományok könyvtárak közötti gyors másolását és mozgatását. Ez a módszer különösen több állomány egyidejű áthelyezése esetén hasznos (*de egyúttal sajnos, veszélyes is*).

A File Manager (ill. a Directory Tree) segítségével nyissuk meg a forrás és a cél könyvtárat, és a Window \Tile paranccsal állítsuk egymás mellé (2-9. ábra).

Jelöljük ki az áthelyezendő állományt az egérrel! Több egymás utáni állomány esetén jelöljük ki az elsőt, nyomjuk le a Shift gombot, és az egérrel kattintsunk az utolsó átviendő állományra! Ha nem egymás melletti állományokat akarunk másolni, a Ctrl gombot kell lenyomni, és egyenként rájuk kattintani.

Mozgatás esetén a kijelölt állományokat az egérrel kell átvezetni a célkönyvtárba. Ha másolni akarunk, akkor a Ctrl gombot tartjuk lenyom-

va, miközben a kurzort mozgatjuk. Ha a célkönyvtár más meghajtón van, akkor a Ctrl gombot nem kell lenyomni, de ebben az esetben a mozgatásnál az Alt gomb lenyomása szükséges.

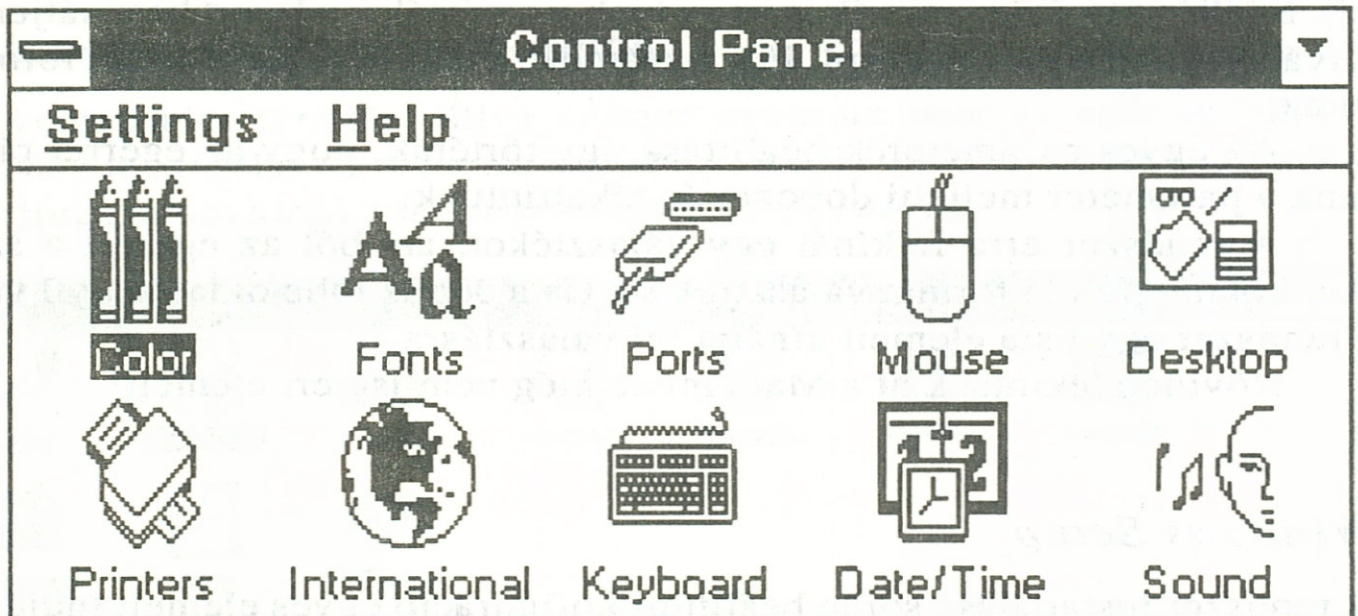


2-9. ábra

Control Panel

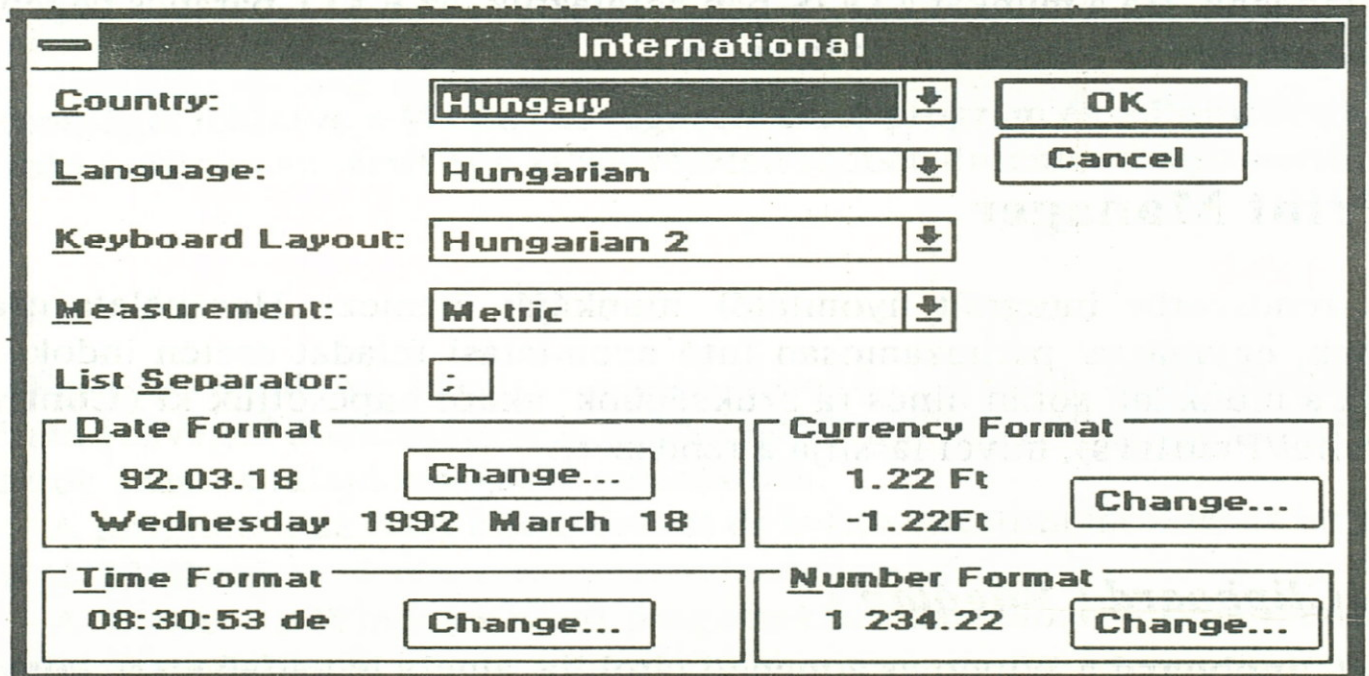
A Control Panel – a File Manager-hez hasonlóan – a Főmenüben található (2-10. ábra). Segítségével a rendszer alapbeállításai módosíthatók. Beállítható többek között:

Color	a képtartalom színkészlete,
Font	a használt betűkészlet,
Port	a soros vonal üzemmódja,
Mouse	az egér üzemmódja,
Printers	a csatolt nyomtatók típusa és paramétere,
Keyboard	a billentyűzet elrendezése,
Date-Time	a rendszeridő,
International	a dátum formátuma, a pénznem, az ezres és a tizedes elválasztó karakter stb.



2-10. ábra

Nézzük meg most egy kicsit részletesebben a Control Panel-on belül az International nyújtotta beállítási lehetőségeket (2-11. ábra)!



2-11. ábra

Az International ablak segítségével megadható a kiválasztott nyelv (a magyart a Windows nem támogatja, erre külön programot kell beszerez-

ni). Beállítható, hogy metrikus vagy inch-es mértékrendszert használjunk. Kiválasztható a pénznem (pl. Ft), a számok, a dátum és az időpont formátuma.

Az egyes paraméterek beállítása úgy történik, hogy az egérrel ráállunk a paraméter melletti dobozra és rákattintunk.

A rendszer erre felkínál egy választékot, amiből az egérrel a számunkra megfelelő formát választjuk ki. Ha a doboz jobb oldalán nyíl van, a rendszer egy lista elemeit kínálja fel választásra.

Röviden tekintsük át a Main menü még nem ismert elemeit!

Windows Setup

A rendszer installálása során beállított konfiguráció egyes elemeit tudjuk a **Windows Setup**-al megváltoztatni (képernyő típust, billentyűzetet stb.)

DOS Prompt

Segítségével visszatérhetünk a DOS-ba anélkül, hogy a Windows-ból kilépnénk. Ha a munkát a DOS-ban befejeztük, az **EXIT** parancs begépelésével lépünk vissza a Windows-ba.

Print Manager

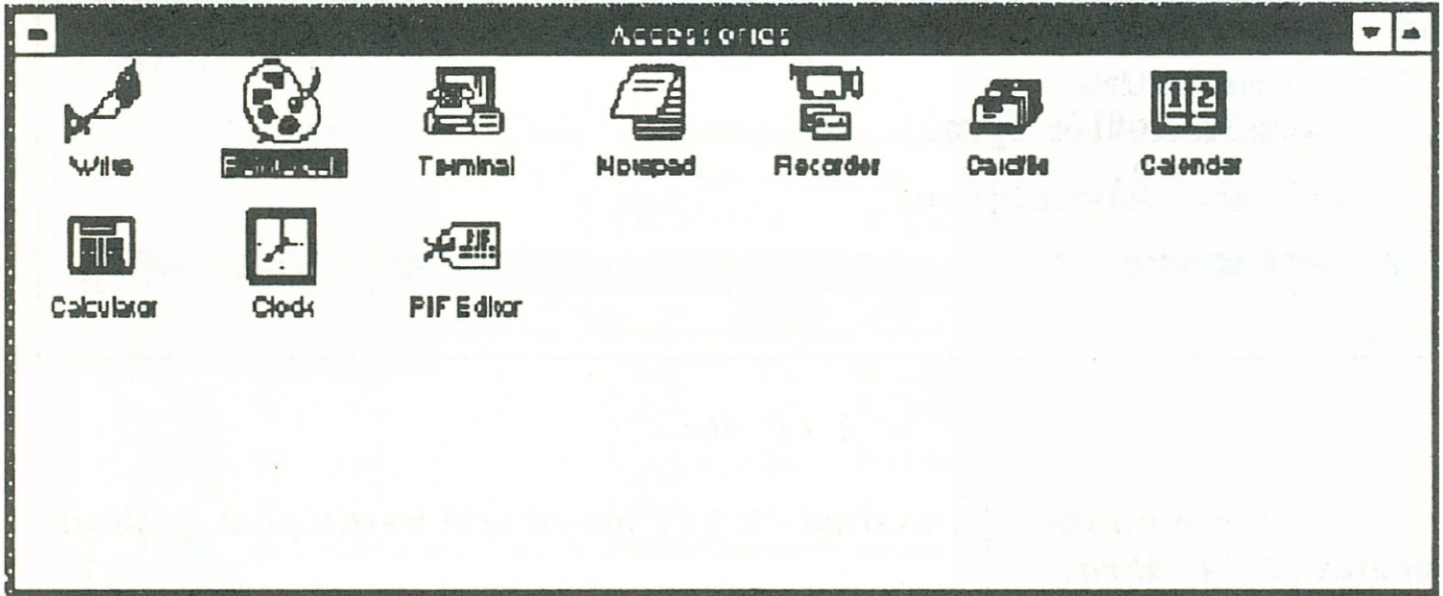
A rendszerbe integrált nyomtatók munkáját ütemezi. Használata csak több, egymással párhuzamosan futó nyomtatási feladat esetén indokolt. Ha a munkánk során nincs rá szükségünk, akkor kapcsoljuk ki (**Control Panel/Printers**), mivel lassítja a rendszert.

A Clipboard (Vágólap)

A **Clipboard** a Windows átmeneti tárolója, amely lehetővé teszi, hogy a felhasználói programok között információt cseréljünk. Használatára a könnyebb érthetőség végett az **Accessories** ismertetése után térünk vissza.

Accessories (Tartozékok) ablak

Az Accessories ikonjára állva kétszer nyomjuk meg az egér gombját. Hatására megnyílik az Accessories ablak, amely néhány kisebb-nagyobb segédprogramot kínál a felhasználóknak (2-12. ábra).



2-12. ábra

Fontosságát tekintve a Write szövegszerkesztő program és a Paintbrush rajzoló program az, amit egy kicsit részletesebben érdemes megismerni.

Write

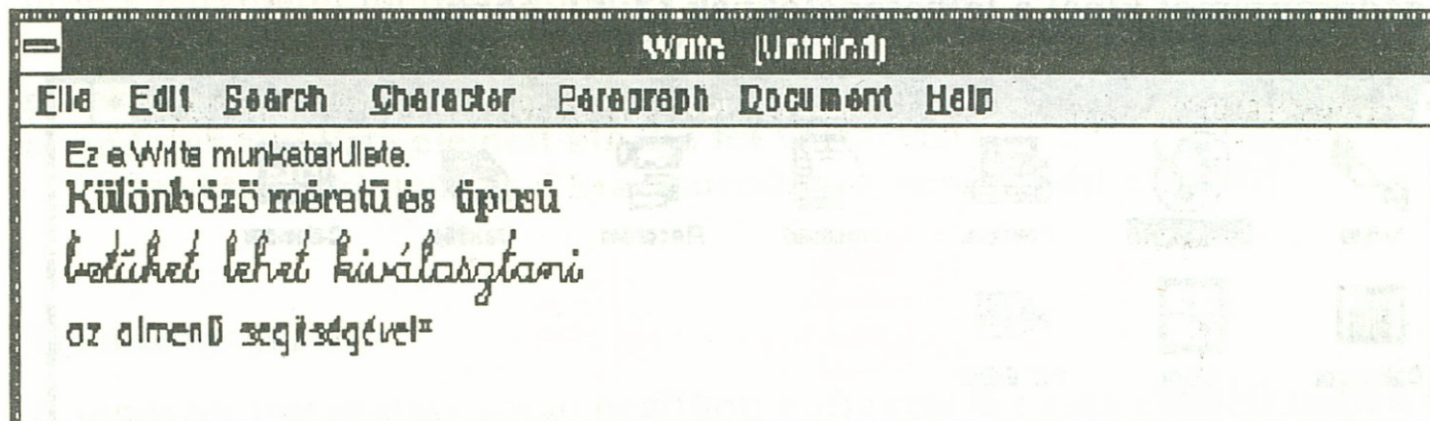
A Write a Windows belső szövegszerkesztője. Kisebb vagy közepes méretű szöveges dokumentumok írására alkalmas, a korszerű szövegszerkesztők számos tulajdonságával rendelkezik.

A programot az ikon kijelölésével és kettős rákattintással lehet indítani (2-13 ábra).

A Write – a Windows alatti programokhoz hasonlóan – saját menükészlettel rendelkezik.

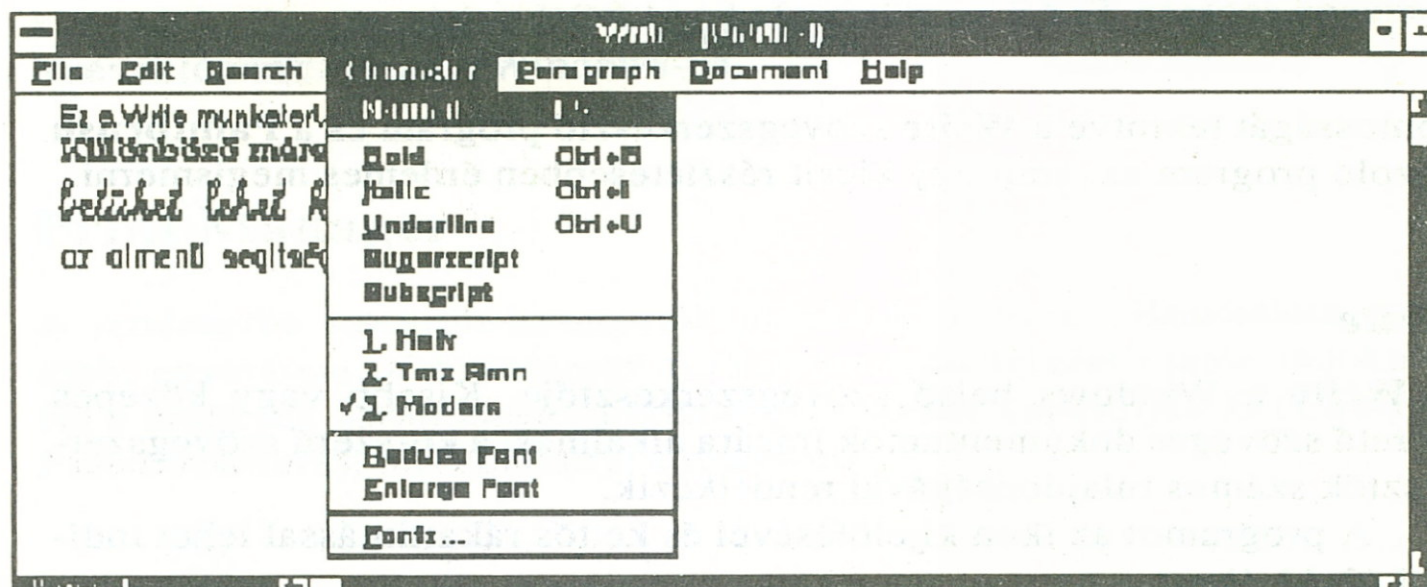
A File menü segítségével meg tudunk nyitni új dokumentumot (New), be tudunk olvasni korábban elkészített szöveget (Open), eltárolhatjuk (Save és Save As...) és kinyomtathatjuk (Print) az elkészített anyagot. A munka befejeztével Exit paranccsal lépünk ki a szövegszerkesztőből.

Az **Edit** menü a kiválasztott információt a vágólapra helyezi (**Cut** és **Copy**),illetőleg onnan új környezetbe másolja (**Paste**). A **Search** menü a szövegben való keresést, valamint a szövegelemek cseréjét teszi lehetővé.



2-13. ábra

A **Character** menüben választhatjuk ki a megfelelő betűtípust és betűméretet (2-14. ábra).



2-14. ábra

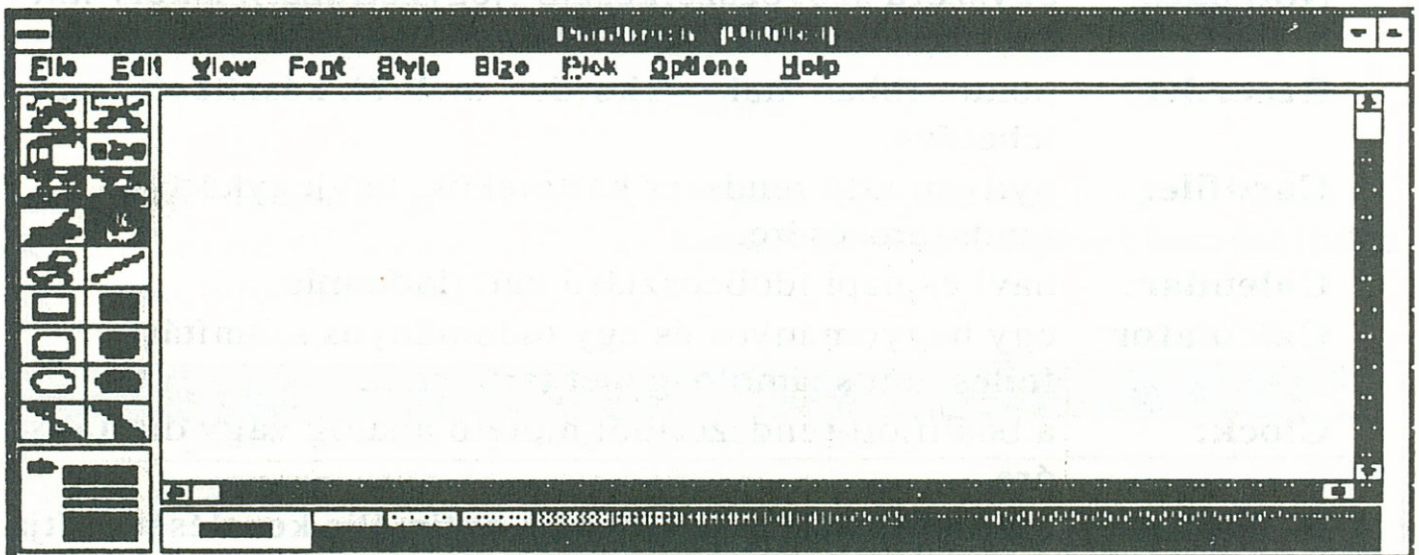
A **Paragraph** menüvel a bekezdést jobbra, balra, illetve középre igazíthatjuk, beállíthatjuk a sorközt, valamint a margótól való távolságot.

A **Document** menüvel többek között fejléctet és lábléctet lehet illeszteni a szöveghez. Beállítható a **tabulátorok** helye, valamint a **margó mérete** is.

Az elkészült dokumentumhoz a rendszer **.wri** kiterjesztést ad.

Paintbrush

A **Paintbrush** a Windows rajzolóprogramja. Behívása az egérrel való kijelöléssel és kettős rákattintással történik (2-15. ábra).



2-15. ábra

A **Paintbrush** ablak bal oldali piktogramm-oszlopán állítható be az üzemmód: szöveg grafikus bevitele, vonalak rajzolása az egérrel, szögletes vagy legömbölyített négyzetek megjelenítése, radírozás, rajzelem kivágása stb.

A **File** és **Edit** menü feladata hasonló a **Write**-nál már ismertetettel.

A **View** menüvel a kiválasztott részletet lehet kinagyítani. A **Font**, a **Style** és a **Size** segítségével a szöveges információ stílusát, méretét lehet beállítani. A **Pick** menüvel és a kijelölő ollóval a megjelenített ábrák tetszőleges része nagyítható, kicsinyíthető, elmozgatható, módosítható. A kijelölő ollóval és az **Edit** menüvel az ábra tetszőleges része vágólapra tehető, és azon keresztül bármely Windows alá fejlesztett programba bemásolható.

Az elkészült dokumentumhoz a rendszer **.bmp** vagy **.pcx** kiterjesztést ad. A Windows rendszer néhány kész ábrát is felkínál, pl.: **Chess.bmp**, melyeket a **File/Open**-el lehet behívni.

Színes monitor esetén az ábra elemei az ablak alján található szín-skála segítségével kiszínezhetők (a "papír" és a "toll" színe egyaránt állítható), és így igen látványos rajzokat készíthetünk.

Az **Accessories** (2-12. ábra) néhány további kisebb-nagyobb szolgáltatást ajánl fel a felhasználóknak:

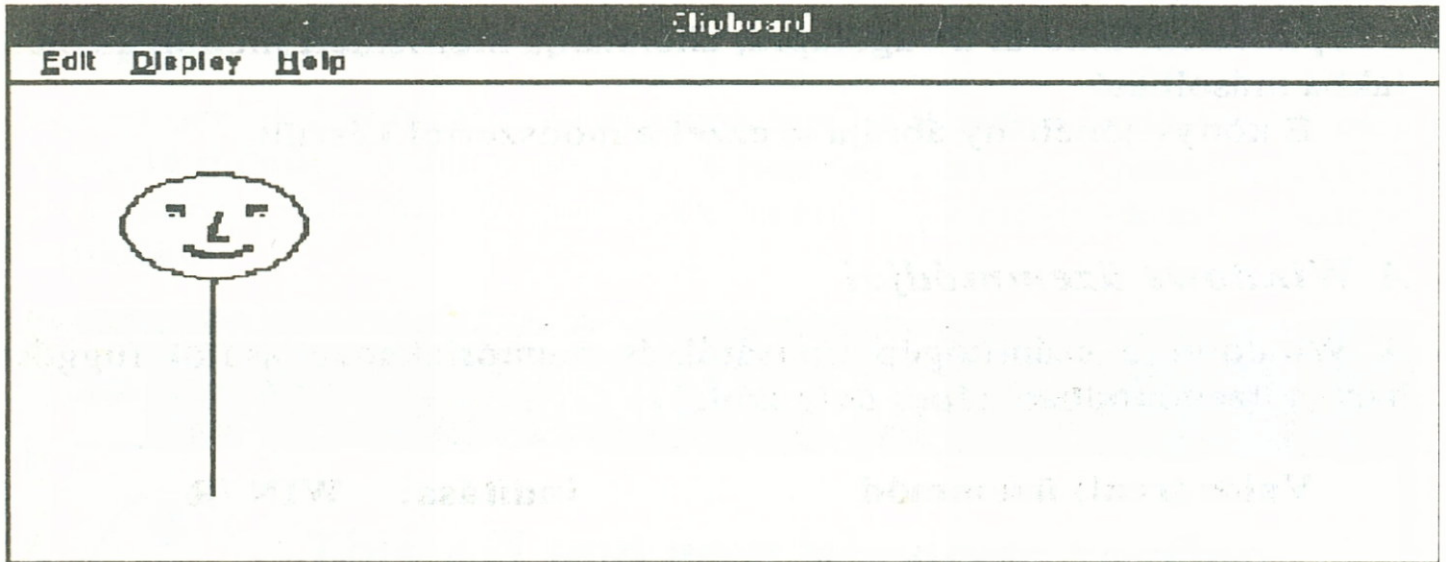
- Terminál:** több számítógép összekapcsolását lehetővé tevő adat-átviteli program.
- Notepad:** egyszerű szövegszerkesztő (jegyzetfüzet), amely **.txt** file-okat hoz létre.
- Recorder:** pontosabban makrorekorder, makrók készítését teszi lehetővé.
- Cardfile:** nyilvántartó rendszer kartotékok, névjegykártyák rendszerezésére.
- Calendar:** havi és napi időbeosztású határidőnapló.
- Calculator:** egy hagyományos és egy tudományos számításhoz fejlesztett számoló-gépet tartalmaz.
- Clock:** a beállított rendszeridőt mutató analóg vagy digitális óra.
- PIF Editor:** a nem Windows programok optimális kezelését állítja be.

A vágólap alkalmazása

A **vágólap** (Clipboard) egy átmeneti belső tároló, amely segítségével szöveges vagy képi információt lehet átvinni egyik alkalmazásból a másikba, illetőleg áthelyezni a dokumentum más helyére.

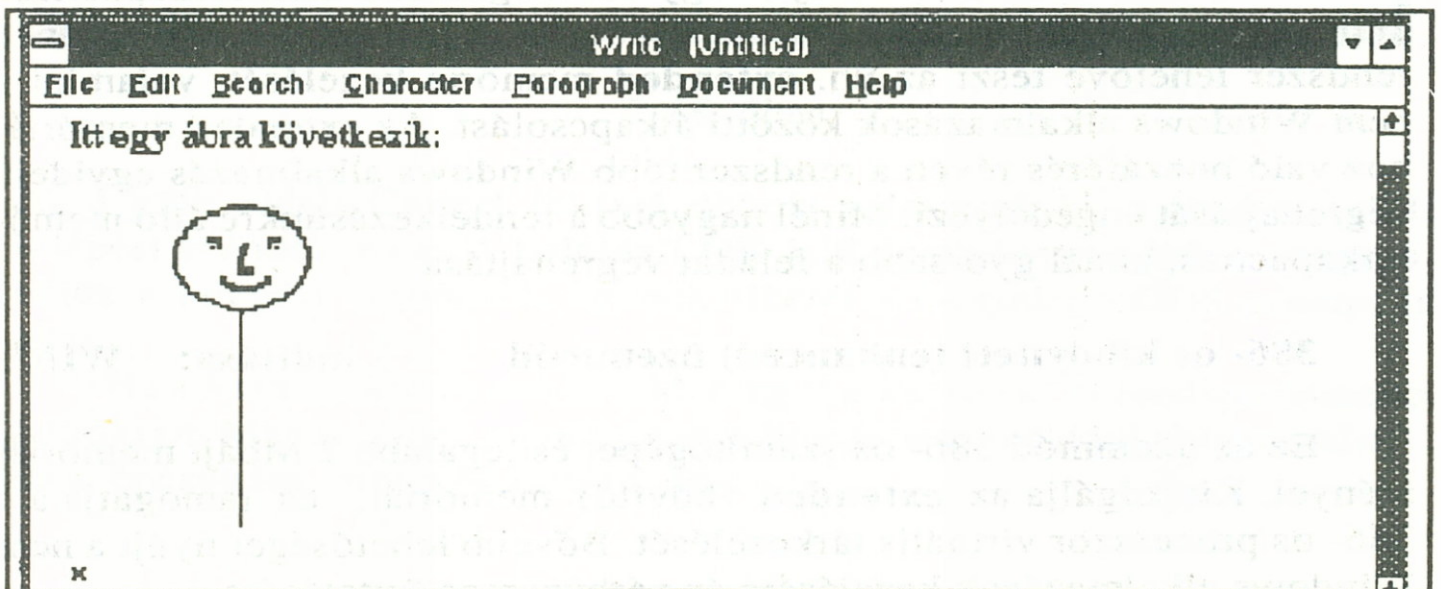
Jelöljük ki az átviendő információt! Például Paintbrush segítségével készítsünk egy ábrát és "vágjuk ki" az ollóval! Az alkalmazás **Edit** menüjéből válasszuk ki a **Copy**, vagy **Cut** parancsot! Ennek hatására az információ a vágólapra kerül.

A Főmenüből (Main) jelöljük ki a Clipboard-ot, és kattintsunk rá kétszer! Erre megjelenik a Clipboard tartalma (2-16. ábra).



2-16. ábra

Ezután a Write szövegszerkesztővel írjunk egy szöveget (2-17. ábra)! A kívánt helyre állva az egérrel, az **Edit/Paste** segítségével bemásoljuk a Vágólapon tárolt információt.



2-17. ábra

Az információ átvitele kétféleképpen történhet: a **Copy** paranccsal úgy, hogy az eredeti helyen is megmarad a szöveg vagy ábra, míg a **Cut** paranccsal nem.

A vágólappal teljes képtartalom is átvihető. Ha a kívánt információ megjelent a képernyőn nyomjuk meg az **Alt** és a **Print Scrn** gombot. Erre

a képtartalom átkerül a vágólapra, ahonnan a már ismert módon a célablakba másolható.

E könyv jónéhány ábrája is ezzel a módszerrel készült.

A Windows üzemmódjai

A Windows a számítógép típusától és memóriakapacitásától függően három üzemmódban képes dolgozni:

Valós (real) üzemmód

indítása: WIN /R

Ezt az üzemmódot akkor alkalmazzuk, ha a számítógép memóriakapacitása nem éri el az 1 Mbájt, illetőleg ha a Windows korábbi verziója (pl: Windows 2.X) alá írt programot akarunk futtatni.

Szabványos (standard) üzemmód

indítása: WIN/S

Ez az üzemmód az 1 Mbájt, vagy ezt meghaladó memóriakapacitású 286-os számítógépek szokásos üzemmódja. Ebben az üzemmódban a rendszer lehetővé teszi az ún. **extended** memória kezelését, valamint a nem Windows alkalmazások közötti átkapcsolást. Az extended memóriához való hozzáférés révén a rendszer több Windows alkalmazás egyidejű végrehajtását engedélyezi. Minél nagyobb a rendelkezésünkre álló memóriakapacitás, annál gyorsabb a feladat végrehajtása.

386- os kibővített (enhanced) üzemmód

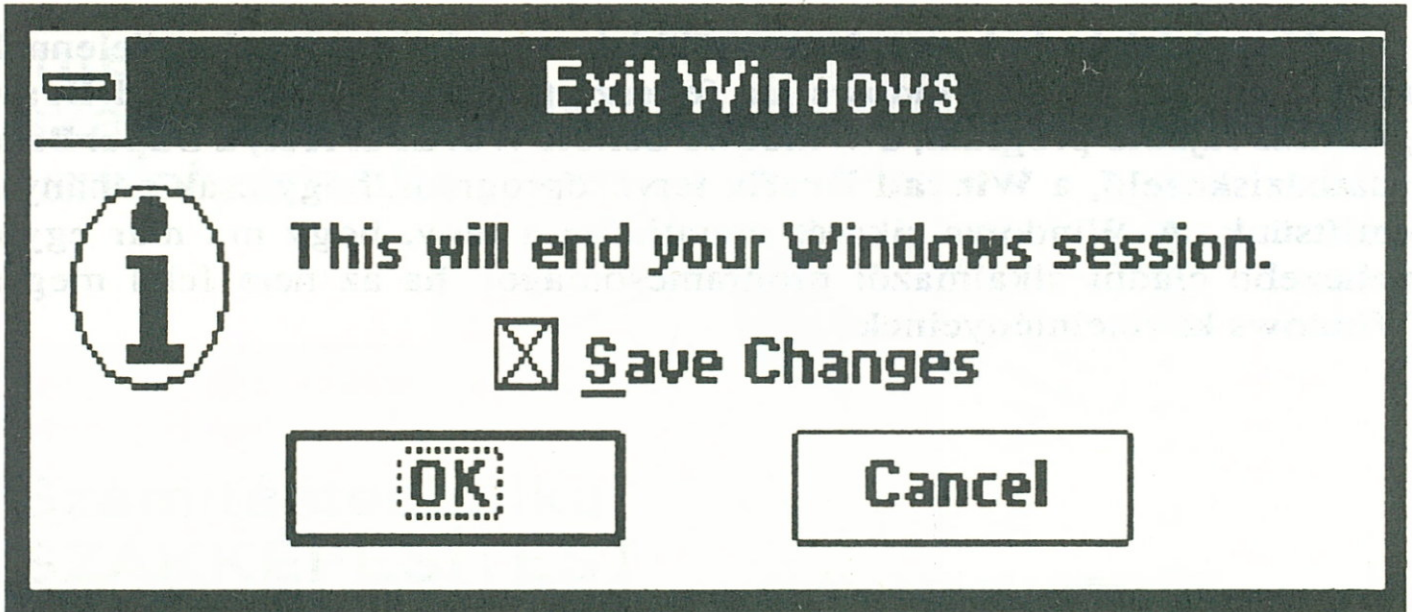
indítása: WIN/3

Ez az üzemmód 386- os számítógépet és legalább 2 Mbájt memóriát igényel. Kiszolgálja az **extended** (bővítő) memóriát, és támogatja a 386- os processzor virtuális tárkezelését. Bővebb lehetőséget nyújt a nem Windows alkalmazások kezelésére és párhuzamos futtatására.

A Windows installálása során a rendszer automatikusan a konfiguráció által megszabott legmagasabb üzemmódba áll be. Ettől eltérni (lefelé) a Windows indításakor a kívánt üzemmód beállításával lehet (pl. WIN/R). Ha az installálás során a rendelkezésre álló szabad memóriakapacitás a szükséges méretet nem éri el, a rendszer alacsonyabb üzemmódot fog beállítani.

Kilépés a Windows-ból

Ha befejeztük a munkát és ki akarunk lépni a Windows-ból, hívjuk meg a **Vezérlő menüt** és válasszuk ki a **Close** parancsot. A rendszer ekkor a 2-18. ábrán látható dialógusablakkal kéri, hogy erősítsük meg a kilépési szándékot:



2-18. ábra

Ha a jobb oldali, **Cancel** feliratú dobozt választjuk ki, a rendszer törli kilépési szándékunkat. A baloldali, **OK** jelű gomb lenyomásával megerősítjük a kilépési szándékot, aminek sikerét a megjelenő DOS prompt is jelzi.

Ha a **Save Changes** dobozt előzőleg "beikszeltük", a rendszer elmenti a Program Manager-ben végrehajtott változásokat (például ablakok kinyitása, becsukása, méret beállítása stb.).

Windows alatti alkalmazói programok

A Windows grafikus felhasználói felület a Windows 3.0 megjelenését követően viharos gyorsasággal terjedt el.

Érthető, hogy a Microsoft volt az első, amely kifejlesztette programtermékeinek Windows alatti változatait:

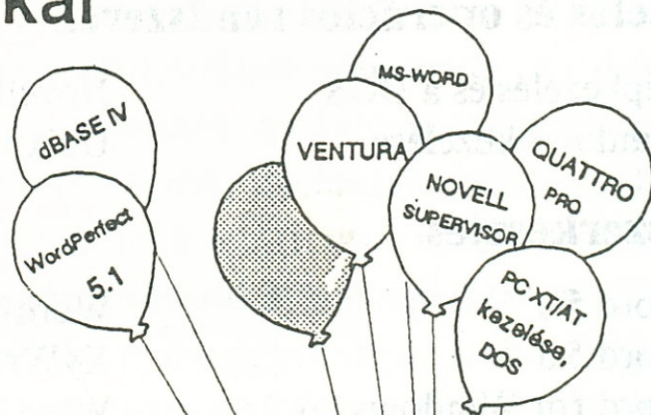
- a **Word for Windows** szövegszerkesztőt,
- az **Excel** táblázatkezelőt,
- a **Works for Windows** integrált programcsomagot.

Nem késlekedtek azonban a többiek sem, és egyre-másra jelennek meg a piacon az új, Windows alá fejlesztett programok: a **Corel Draw** grafikus rajzoló program, a Windows-osított **WordPerfect**, a **SuperBase** adatbáziskezelő, a **Wincad Drafix** tervezőprogram, hogy csak néhányat említsünk. A Windows sikerét mutatja az a tény, hogy ma már egyre nehezebb eladni alkalmazói programcsomagot, ha az nem felel meg a Windows követelményeinek.



MŰSZERTECHNIKA OKTATÁS

**Intenzív (1-2 hetes)
számítástechnikai
tanfolyamok:**



**Számítástechnikai
SZAKKÉPESÍTÉST
adó tanfolyamok:**

- Személyi számítógép
szoftver üzemeltető
(1 éves, középfokú)
- Számítógép programozó
(2 éves, középfokú)
(Szoftver üzemeltetőknek: 1 év)
- Információrendszer programozó
(3 éves, felsőfokú)
(Programozóknak: 2 év)



Jelentkezését várja a Műszertechnika OKTATÁS, 1107 Bp. Szállás u. 21.
☎ 147-5307, 147-1590 Fax: (36-1)157-0284 ✉ 1475 Bp. Pf. 225

számítógépes tanfolyamainak széles választékával áll ügyfelei rendelkezésére

I. Intenzív tanfolyamok

Gépkezelés és operációs rendszerek

Gépkezelés és a DOS
Windows kezelése

Novell hálózati tanfolyamok
Unix kezelése

Szövegszerkesztés

Word 5.0
Word 5.5
Word for Windows

WordPerfect
XyWrite (Írnok)
WordStar

Táblázatkezelők, integrált rendszerek

MS-Excel
MS-Works
Quattro

Lotus
Symphony
Framework

Programnyelvek

C

Pascal

Adatbáziskezelők

dBase III PLUS
dBase IV

Clipper
FoxPro

II. Szakképesítést adó tanfolyamok

Személyi-számítógép szoftverüzemeltető *(középfokú)*

Személyi-számítógép kezelő *(alapfokú)*

Számítógépes kiadványszerkesztő *(alapfokú)*

Számítógépes ügyvitel adminisztrátoroknak, titkárnőknek *(alapfokú)*

III. Kihelyezett tanfolyamok

A tanfolyamokat Önöknél is megtartjuk egyedi megállapodás alapján, egyéni tematikával és beosztás szerint

Keressen meg bennünket!

Bognár Júlia, Gerő Judit, Kóczy A. Judit, Péntes B. Zsuzsa, Reich Gábor, Rudnai Péterné

CONTROLL Oktatóközpont

1027 Csalogány utca. 23. V.em.

Tel: 202-6898 Fax: 201-7292

3. Szövegszerkesztés számítógéppel

Szövegszerkesztőről általában

A írásbeliség, gondolataink papíron történő rögzítése, a kötelező bizonylatolás egyike a legidőigényesebb tevékenységeknek. Az évszázadok során önálló szakmák alakultak ki ennek a tevékenységnek a színvonalas elvégzésére (gépírók, adminisztrátorok, nyomdai előkészítők stb.) A számítógép "csak" egy eszköz, amely a szellemi erőfeszítéstől ugyan nem mentesít, de a megvalósítás gépies munkáját megkönnyíti, és – a tanulási időszakot leszámítva – jelentősen meggyorsítja. Lehetőséget biztosít a korábban elkészített dokumentumok javítására, módosítására, kész szövegrészek "átmentésére" új környezetbe, típuslevelek adott nevekkel és címekkel történő aktualizálására, vagyis arra, hogy az egyszer már leírt gondolatokat ne kelljen ismételten bebillentyűzni.

Az évek során számos szövegszerkesztő program terjedt el, nem egy közülük már magyarul is "beszél". Mindegyik programnak más-más az erőssége (és természetesen a gyengéje is). Sokszor felmerült az a kérdés melyik a legjobb szövegszerkesztő?

Ha a speciális igényeket figyelmen kívül hagyjuk akkor a válasz az, hogy mindenki számára az az ideális szövegszerkesztő, amelynek a kezelését már elsajátította.

A szerzőknek eddigi munkájuk során módjuk volt megismerkedni jónéhány egyszerűbb és bonyolultabb szövegszerkesztő programmal, és ezek közül – a könyv megírásának pillanatában – a Microsoft szoftverház egyik legújabb termékét a **Word for Windows-t** (röviden **WinWord**) találták a legjobbnak. A választás egyik oka, hogy ez a program tud legkényelmesebben együttműködni más, szintén a Windows alá kifejlesztett alkalmazásokkal. (Lásd a Windows ismertetését az első fejezetben!) További okok: a program kényelmes kezelése és – nem utolsósorban – szolgáltatásainak széleskörű választéka.

A program lehetőséget nyújt arra, hogy egy-két órás gyakorlás után egyszerű és szép leveleket készítsünk és ismereteinket tovább bővítve, igényes, a kiadványszerkesztő programok dokumentumaival azonos színvonalú nyomdakész kiadványokat állítsunk elő. Ezeket a programokat úgynevezett

User friendly-felhasználóbarát programoknak nevezzük. Ezt a Windows környezet, a képernyő piktogramok, valamint az egér biztosítja.

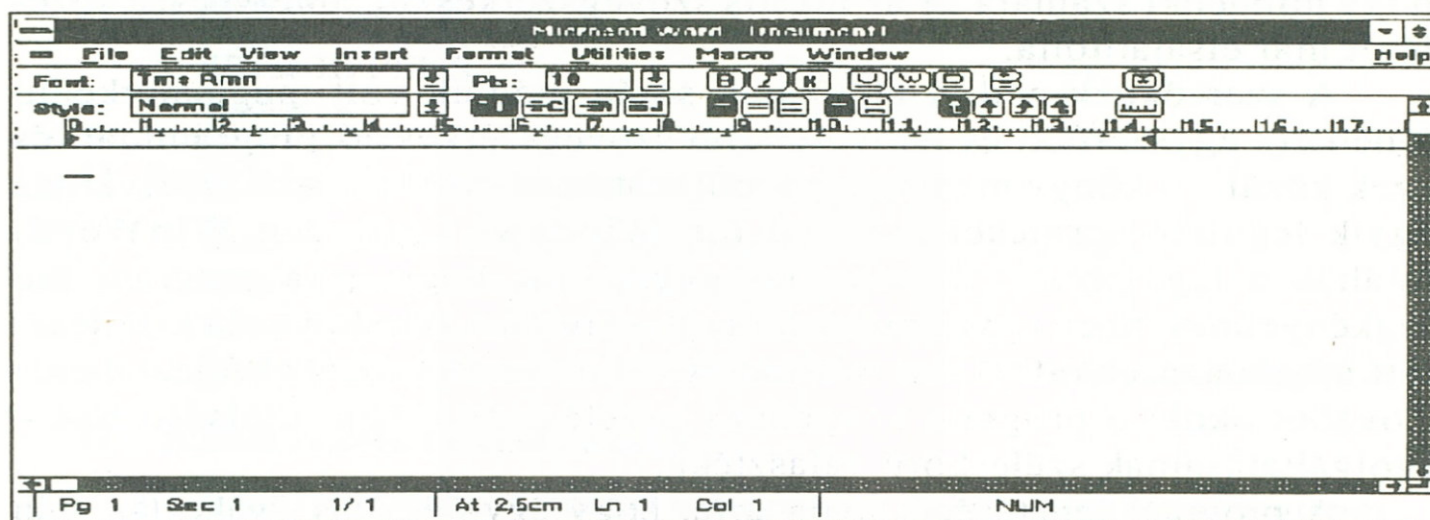
Hogyan készítsünk dokumentumot szövegszerkesztővel

Egy új dokumentum elkészítése során a billentyűzés fáradságos munkáját nem tudjuk megspórolni. Azonban eldönthetjük, hogy a gépelés során a formázási feladatokat is rögtön elvégezzük-e, vagy majd a kész szöveget formázzuk. Mindkét megoldás szokásos. Természetesen a hibák, elgépelések javítása is történhet folyamatosan vagy utólag.

Formázni betűket, szavakat, bekezdéseket, tetszőleges szövegrészeket (szekciókat) és teljes dokumentumot lehet. Egy-egy azonos külalakú bekezdés (címek, felsorolások stb.) formázása történhet egy lépésben is, a közös tulajdonságok egyszeri megadásával. Ennek során először beállítjuk a bekezdés jellemzőit (pl.: betűméret, középreállítás, aláhúzás, sorközök stb.), a beállított tulajdonságot elnevezzük egy úgynevezett *stílusnévvel*, és ettől kezdve, ha egy bekezdést ugyanolyan külalakúnak szeretnénk, csak hozzá kell rendelni az előre elkészített stílust. A hozzárendelés egy listából, – a stílusok listájából – való választást jelent.

A tanulás kezdete

A WinWord elindítása után (a WinWord ikonjára kattintunk a Windows-ban) a következő képernyőképet látjuk. (3-1. ábra)



3-1. ábra

Az alkalmazás címsorában a program neve mellett az aktuális dokumentumnak, állománynak (angolul file) a nevét látjuk. Ez induláskor mindig a *Document1*. Később, az állomány mentése során tudunk saját nevet adni a munkáinknak.

A címsor alatt a főmenü sor van. A menükben megtaláljuk a program kezeléséhez szükséges összes parancsot, tehát nem kell megtanulnunk azok pontos helyesírását, elég csak kiválasztani azokat.

A menükből legkönnyebben az egér segítségével tudunk választani. Ráállunk az egér nyílával a kívánt főmenüpontra, és megnyomjuk az egér bal gombját. (A program általában a bal gombot használja – néhány esettől eltekintve.) Ilyenkor legördül az almenü listája. Az almenükben azok a menüpontok, amelyek az adott pillanatban végrehajthatók, feketén látszanak, amelyek nem azok szürkék. (Például amíg nincs a dokumentumban táblázat, addig a táblázatformázó *Format/Table...* parancsnak nincs értelme.) Azok a menüpontok, amelyek után ... látható, egy-egy újabb dialógusdobozt jelenítenek meg, ahol a paraméterek pontosíthatók, további értékek állíthatók be. A többi menüpont azonnal végrehajtható. Billentyűvel is lehet választani a főmenüből úgy, hogy az ALT billentyűvel együtt lenyomjuk a menüpont aláhúzott betűjét.

Ha az almenüket bármelyik módszerrel legördítettük, a nyílbillentyűk segítségével tudunk függőlegesen haladni. Minden menüponthoz, amelyiken a menükurzor áll, a képernyő alsó részén látható státusz sorban egy magyarázó szöveg jelenik meg.

Néhány menüben a parancsok mellett **billentyűkombinációkat** találunk. Ez azt jelenti, hogy a parancsokat nem csak a menükből választva hajthatjuk végre, hanem ezekkel a billentyűkombinációkkal is. Ezekon kívül még rengeteg billentyűkombináció tartozik a programhoz, de csak igen nagy rutin után jelent ez gyorsabb programkezelést, mint a menüs vezérlés.

A menük alatt láthatjuk a **szalagsort** (Ribbon), ami egyszerű lehetőséget nyújt a kijelölt – tetszőleges hosszúságú – szövegek formázásához (betűtípus, méret, attribútum: vastagítás, dőltbetű stb.) Alatta a **vonalzó** (Ruler) található. Ez utóbbi két részből áll: a bekezdések formázásához tartozó beállítási szolgáltatásokból és egy igazi vonalzóból. Itt is tudunk a bekezdésekhez stílust rendelni, meghatározni az igazításukat (balra, középre, jobbra és kiegyenlítve), beállítani a bekezdések sortávolságát és egymástól való távolságát, valamint a tabulátorokat. A sor végén pedig a vonalzón megjelenő beállítások között tudunk váltani. Az itt

felsorolt fogalmakat a fejezet további részében természetesen részletesen elmagyarázzuk.

A képernyő alján található a **státusz**sorban a dokumentum állapotával kapcsolatos néhány információt követhetünk nyomon: az összes oldal számát, az aktuális (ahol a szövegkurzor éppen van) oldal számát, a sor- és karakterszámot, a kapcsolóbillentyűk állapotát: (NumLock, CapsLock stb)., és esetenként más információt és rendszerüzenetet is.

A képernyő alsó és jobb oldali részén egy-egy **gördítősávot** találunk. Ennek funkciója azonos a Windows-ban leírtakkal, kiegészítve a lapváltási és képernyőosztási lehetőségekkel. Végül de nem utolsósorban, a fennmaradó terület a **munka**terület, ahol a szövegbevitel és a szerkesztés történik.

Amennyiben az Ön képernyőjén mindaz nem látható, amit a fentiekben felsoroltunk, akkor gördítse le a **View** menüt, és ellenőrizze, hogy a felsorolt menüpontok mellett látja-e a pipát? Bekapcsolni, illetve kikapcsolni egyszerű, csak rá kell kattintani az egérrel, és azonnal átvált a megjelenítésből eltüntetésbe és viszont.

A szöveg beírása

A szövegszerkesztés alapja – sajnos még mindig – a szöveg begépelése. Ezt nem tudjuk megspórolni, csak a javítást tudjuk utólag gyorsan elvégezni, és a külalakot bármikor megváltoztatni.

Az írógéppel végzett munkától eltérően, ha szövegszerkesztő programmal dolgozunk, a programra bízhatjuk, hogy a sor végén azt a szót, amelyik nem fér már el, átvigye a következő sorba. Nem kell és nem szabad a sorok végén megnyomni az **ENTER** billentyűt, csak a bekezdések végén! Ilyenkor természetesen új sor is kezdődik.

Miért olyan fontos ez? Ha utólag beszúrunk vagy kitörölünk egy vagy több betűt, illetve szót, vagy megváltoztatjuk a dokumentumunk margóit, egészen máshova esnek a sorok végei. Ha a programra bízunk a sorváltást, akkor az új helyzetnek megfelelően azonnal kialakítja az új sorokat. Ha valamit feltétlenül *új sorba*, de *nem új bekezdésbe* szeretnénk tenni, akkor a **SHIFT** billentyű és az **ENTER** gomb együttes lenyomásával ezt is elérhetjük.

A további tanulás megkönnyítésére gépeljünk be néhány bekezdést, akár ebből a könyvből is!

A szöveg gépelése során a képernyő alján lévő státuszsorban folyamatosan követhetjük, hogy hányadik sor (**Ln**), hányadik karakterét (**Col**)

írjuk, és hányadik oldalon vagyunk (Pg). Ez nagyon hasznos a dokumentumban történő tájékozódásban, mivel a képernyőn megjelenő szöveg kevesebb, mint a papíron megszokott szövegmenyiség.

*Megjegyzés: Amennyiben az elkövetkező feladatok közben bármikor pihenni szeretne, vagy félbehagyni kényszerül a tanulást, el kell mentenie a dokumentumot, hogy a munkát később ott folytathassa, ahol azt abbahagyta. A mentés és visszatöltés lépéseit később, a **Állományműveletek** című részben ismertetjük !*

Speciális karakterek

A szöveg írása közben lehetőségünk van speciális karakterek beírására, amelyek jelentése, hatása eltér a megszokottól. Az alábbi listában sorra vesszük ezeket.

- **Új sor, de nem új bekezdés: SHIFT + ENTER**
Minden olyan esetben ezt használjuk, ha nem akarunk új bekezdést kezdeni, csak új sort.
- **Nem törhető szóköz: CTRL+SHIFT+szóköz**
Erre akkor van szükségünk, ha nem engedjük meg, hogy a program elválassza egymástól, új sorba írja a törhetetlen szóköz két oldalán álló szavakat. Például a 12 000 20Ft leírásakor furcsán nézne ki, ha a 12 után vagy a Ft előtt új sort kezdene a program, pl.:12
000 Ft
- **Új oldal: CTRL + ENTER**
A program automatikusan vált lapot, amikor az aktuális lapméretnek és beállításoknak megfelelő oldal betelik. Abban az esetben, ha mi akarjuk vezérelni a lapváltást, például egy új címnél, táblázatnál stb., a fenti billentyűkombinációval ezt elérhetjük. (Az **Insert/Break Page Break** parancs hatása is ugyanez.)

Elválasztójelek

- **Egyszerű elválasztójel: -**
Mindig megjelenik a szövegben, és ha a sor végére kerül, elválasztójelként is funkcionál

- **Lehetséges (opcionális) elválasztójel beírása: CTRL + -** (a CTRL és a kötőjel együtt.) Ez a jel csak akkor jelenik meg ténylegesen a kötőjelként, ha az adott szó a sor végére kerül, és az túllógna a margón. (A program maga is tud elválasztani (az angol helyesírás szerint) a **Utilities/Hyphenate** parancs segítségével. Ilyenkor felkínálja azt a helyet a szóban, ahol javasolja az elválasztást, és mi vagy megerősíthetjük, vagy új helyet választhatunk.)
- **Fix , elválasztásra nem használható kötőjel: CTRL+SHIFT+-** (a CTRL, a SHIFT és a kötőjel együtt.) Az olyan szavak között használjuk, ahol kötőjelet kell írni, de nem akarjuk, hogy külön sorba kerüljenek: pl. FX-1000-es nyomtató, és *nem* FX-1000-es

A felsorolt speciális karakterek mindegyike látható a képernyőn is, ha a menüsor alatt lévő szalagsor jobb oldalán lévő csillagra rákattintunk. Ez a **Show All** (mindent megmutat) kapcsoló, amivel ezeknek a jeleknek, illetve az eddig még nem tanult dolgoknak a megmutatását tudjuk ki/be kapcsolni. (Ki/be kapcsolni egyesével a **View/Preference...** dialógusdobozban lehet.) Ezek a karakterek igazi karakterek abban az értelemben, hogy ugyanúgy lehet őket törölni, mint minden más karaktert, akár a **DEL**, akár a **Backspace** billentyűvel.

Nem várt események

Mielőtt továbbhaladnánk a tanulásban, fel kell készülnünk váratlan helyzetekre, olyan kérdésekre, amire nem számítunk, amelyekre nem tudjuk a "választ". Biztos nem sikerül itt mindegyiket felsorolni, de megpróbáljuk a legfontosabbakat:

- Véletlenül legördítettük a menüsort, és mégse kellett volna: ilyenkor, ha az egérrel a menüterületen kívül kattintunk egyet, a menü eltűnik. Ha megnyomjuk az **ESC** billentyűt, akkor is. Ez utóbbi billentyű sokszor segít a dolgok érvénytelenítésében (escape = menekülés, szökés).
- Megnyitottunk egy olyan dialógusdobozt, amire nincs szükségünk, vagy nem akarjuk, hogy a beírt dolgok érvényesüljenek. Ilyenkor is segít az **ESC** billentyű, valamint a dialógusdoboz jobb oldalán lévő **Cancel** felíratú gomb. Ha az egérrel erre kattintunk, a dialógusdoboz becsukódik anélkül, hogy történne valami.

- Lenyomtunk egy billentyűt, aminek hatására a képernyő alján a státuszsorban kérdez valamit a program. Ilyenkor is az ESC billentyűt használhatjuk az alapállapot visszaállítására.
- Általában igaz, hogy egy hibás művelet után közvetlenül az **Edit/Undo** paranccsal érvényteleníteni tudjuk annak hatását. Az **Undo** parancs után annak a műveletnek a neve szerepel a menüben, amit utoljára csináltunk. Egyes, például a táblázatokkal kapcsolatos műveletek, nem érvényteleníthetők.

A szöveg javítása

A beírás során azonnal tudjuk javítani az esetleges elgépeléseket. Az egyszerű hibák alapvetően három csoportba sorolhatók, ezek javítását nézzük meg először:

- valami felesleges, azaz törlendő;
- valami hiányzik, azaz ki kell egészíteni, új szöveget beszúrni;
- valami rossz, azaz átírandó.

Törlés

Ha egy vagy több karaktert ki szeretnénk törölni, erre több lehetőségünk is van. Nézzük az egyszerűbbeket!

A törlendő karakter elé visszük a szövegkurzorunkat, a villogó függőleges vonalat, és lenyomjuk a **DEL** billentyűt. A kurzor mögötti karakter törlődik. (A szövegkurzort vagy a nyílbillentyűk segítségével mozgatjuk, vagy az egérrel úgy, hogy az egérkurzort a kívánt helyre visszük, és megnyomjuk a bal gombját.)

Ha a kurzor előtti karaktert akarjuk törölni, akkor pedig a **Back-Space** billentyűt kell leütnünk. (Néhány billentyűzeten csak egy balra mutató nyíl jelzi ezt a billentyűt az **ENTER** felett.)

Beszúrás

Ha a szövegből hiányzik egy-két karakter, ezt egyszerűen pótolhatjuk. A szövegkurzort a hiányhoz visszük, és leütjük a beszúrandó karaktereket. A karakterek a kurzor jobb oldalán jelennek meg. A szöveg további része jobbratolódik, és ha kell, akár a következő sorba is átcsúszik. (Ellenőriz-

zük a státuszsorban, hogy az **OVR** jelző nem látszik-e! Ha igen, nyomjuk meg az **INS** billentyűt az eltűntetéséhez!)

Átírás (felülírás)

Természetesen ki lehet törölni a hibás karaktereket és begépelni a jókat. A két lépés helyett egy lépésben is felülírhatjuk a hibás karaktereket úgy, hogy lenyomjuk a billentyűzet jobb oldalán lévő **INS** billentyűt, ami átállítja a programot "beszűrő" módból "felülíró" módba. Ez azt jelenti, hogy a leütött karakterek "ráíródnak" az ottlévőkre. Ilyenkor a státuszsorban az **OVR** jelző látható, jelezve a felülíró módot. A szövegszerkesztő programok jellegéből fakadóan nem célszerű tartósan felülíró módban használni a programot, csak a javítás idejére. Visszkapcsolni az **INS** billentyű ismételt lenyomásával lehet.

Mozgás a szövegben és a szövegkijelölés

Ahhoz, hogy a dokumentumunk formázását elkezdhessük, meg kell tanulnunk, hogyan kell mozogni a dokumentumunkban, és hogyan lehet kijelölni a szöveget. A szöveg lehet egy karakter, szó, mondat, bekezdés, tetszőleges hosszúságú szöveg, vagy akár az egész dokumentum.

Mozgás a szövegben

Mozogni lehet a kurzormozgató billentyűk segítségével és az egérrel. Terjedelmi korlátaink miatt itt nem vállalkozunk arra, hogy minden lehetőséget bemutassunk, de a legfontosabbakat az alábbi táblázatban foglaltuk össze:

Billentyű	Hatása
nyíl (jobbra, balra,le, fel)	Egy-egy karaktert ugrik jobbra balra, illetve egy-egy sort le vagy fel
CTRL + jobbra, balra nyíl	Egy szót jobbra, balra
Home, End	Aktuális sor elejére, végére
CTRL+ le, fel nyíl	Bekezdés elejére, végére
PgDn, PgUp	Egy képernyőoldalt lapoz lefelé, illetve felfelé
CTRL+Home, CTRL+End	Dokumentum elejére, végére
CTRL+PgDn, CTRL+PgUp	Aktuális képernyőoldal tetejére, aljára

A táblázatban szerepelnek olyan lehetőségek, ahol két billentyűt együtt kell lenyomni. Ilyenkor először a váltóbillentyűre (CTRL, SHIFT, ALT) kell rátenni az ujjunkat, utána egyszer leütni a másikat, majd elengedni a váltóbillentyűt. Ha ezt a módszert követjük, nem kell kínlódnunk az "egyszerre" leütéssel.

Lehetőségünk van egy tetszőleges oldal elejére állni a **GoTo** (ugrás) funkció segítségével. A program fejlesztői a leggyakrabban használt feladatokhoz hozzárendelték a funkcióbillentyűket, valamint az ALT, a CTRL és a SHIFT billentyűk kombinációit. A funkcióbillentyűk a billentyűzet tetején vagy bal oldalán található, feliratuk egy F betű és egy sorszám. Ha leütjük például az **F5**-ös funkcióbillentyűt, akkor a képernyő alján a következő üzenet jelenik meg:

Go to:

A kettőspont után kell beírni annak az oldalnak a számát, ahova ugrani szeretnénk. Ha előjeles számot adunk meg, akkor annyi oldallal **előbbre** (+), vagy **hátrább** (-) ugrik a kurzor, amekkora számot beírunk.

Egérrel mozogni az aktuális képernyőoldalon nagyon egyszerű, az egérkurzort odavisszük a kívánt helyre, és megnyomjuk az egér bal gombját.

Ha a szöveg nincs az éppen látható oldalon, a gördítősáv segítségével tudunk mozogni. (A vízszintes gördítősávot csak akkor kell használnunk, ha a szövegünk szélesebb, mint a képernyőnk). A gördítősáv három részből áll:

- **Alul és felül egy-egy nyíl**, amire ha rákattintunk, egy-egy sorral tudjuk lejjebb, illetve feljebb vinni a kurzort.
- **Sötétszürke (vagy más képernyőtípusnál fehér) négyzet** (csúszka), ami a dokumentumunkon belül az éppen aktuális képernyőoldalt jelöli arányosan, azaz a felette és az alatta lévő terület jelzi, hogy a dokumentumban hol vagyunk. Ha a csúszkára ráállunk, és lenyomott egérgomb mellett mozgatjuk le- fel, a dokumentumunknak tetszés szerinti helyére tudunk jutni. (Kis gyakorlattal egész pontosan a célhoz tudunk érni.)
- Ha a nyíl és a csúszka közötti világosabb területre kattintunk, akkor pontosan egy-egy *képernyőoldallal* tudunk lejjebb vagy feljebb lapozni.

- Ha a **View/Page** be van kapcsolva, akkor a két nyílön kívül a függőleges gördítősávok végén egy-egy *számárfüles papír* is látszik. Erre kattintva tényleges oldalanként tudunk lapozni.

Szövegkijelölés

Szöveget kijelölni lehet az egérrel, a billentyűkkel vagy a kettő kombinációjával. Egyszerre csak egy területet tudunk kijelölni függetlenül annak méretétől. Ha egérrel dolgozunk, akkor a kijelölendő szöveg elejére állunk, lenyomjuk és lenyomva tartjuk az egér gombját, miközben végighúzzuk, az egérkurzort a kijelölendő területen. Ha függőlegesen húzzuk teljes sorokat tudunk kijelölni.

Ha egy szón állva, kétszer gyorsan egymás után kattintunk az egér gombjával, a szó lesz kijelölve; ha előzőleg **CTRL** gombot is lenyomjuk és lenyomva tartjuk, az aktuális mondat. A kijelölt szöveg a képernyőn *inverzen* látszik.

Billentyűzetet használva legegyszerűbben a **SHIFT** és a nyílbillentyűk segítségével tudunk kijelölni szöveget. Az **F8**-as funkcióbillentyű is segít a kijelölésben. Ha egyszer megnyomjuk az **F8**-at, nem kell a **SHIFT**-et nyomni a nyílbillentyűkkel történő kijelöléshez. A lenyomott állapotát a státuszsorban egy **EXT** jelző jelzi. Ha kétszer nyomjuk meg az **F8**-at, az aktuális szót, ha háromszor a mondatot, négyszer a bekezdést, ha ötször a szekciót és hatodszorra az egész dokumentumot jelöli ki a rendszer. Visszafelé, a kijelölést szűkíteni a **SHIFT** és az **F8** együttes lenyomásával lehet. Az egész dokumentumot kijelölni a **CTRL** és a numerikus billentyűzet **5**-ösének lenyomásával is tudjuk. A kijelölést megszüntetni az **ESC** billentyűvel tudjuk, ha már nincsen szükségünk rá, vagy rosszul jelöltük ki.

A szöveg másolása, mozgatása, törlése

Gyakran előfordul, hogy ugyanazt a szöveget többször szeretnénk *megismételni* a szövegben, vagy egy másik dokumentumban már leírtuk, és nem szeretnénk újragépelni. Ekkor segít a másolás.

Ha pedig arról van szó, hogy rossz helyre írtunk valamit, és *át szeretnénk helyezni* akár ugyanabban a dokumentumban, akár egy másikba, a mozgatási műveletet kell végrehajtani.

Másolás (Copy)

Másolni két módszerrel is tudunk, a vágólap (Clipboard) segítségével (lásd Windows fejezetet) és anélkül.

A vágólap a Windows alkalmazásoknak egy olyan közösen használható memóriaterülete a számítógép operatív tárolójában, amelyet minden alkalmazás el tud érni. Bárhonnan rámásolnak egy kijelölt "valamit" (képet, szöveget, táblázatot stb.), azt ugyanabban vagy egy másik alkalmazásban, dokumentumban a kurzor pozíciójába be lehet "ragasztani". A vágólap tartalma mindaddig megmarad, amíg egy másik tartalommal nem írjuk azt felül. Ezért a vágólap tartalmát akárhányszor beragaszthatjuk a kurzor pozíciójába.

Hogyan lehet a vágólapra másolni?

Ki kell jelölni a másolandó szövegrészt, és kiadni az **Edit** menü **Copy** parancsát. A parancs hatására a kijelölt szöveg rákerül a vágólapra, törölve annak esetleges előző tartalmát.

Hogyan lehet a vágólap tartalmát beragasztani?

A szövegkurzort oda kell állítani, ahova a vágólap tartalmát szeretnénk beragasztani, és az **Edit** menüből ki kell választanunk a **Paste** parancsot. A vágólap tartalma beszűrődik a kurzor pozíciójába. Ha több helyre szeretnénk beragasztani, csak az új helyre kell vinni a kurzort, és még egyszer kiadni az utolsó parancsot.

Ha egy másik alkalmazásból vagy dokumentumból, illetve alkalmazásba, dokumentumba szeretnénk átmásolni valamit, csak annyival kell többet tennünk, hogy a vágólapra másolás és a beragasztás között alkalmazást, illetve dokumentumot kell váltani. Ennek technikája azonos a Windows-ban leírtakkal.

A másik módszernél, nem használjuk a vágólapot, mert például nincs szükségünk a vágólap nyújtotta szolgáltatásokra, vagy azért, mert nem akarjuk felülírni annak tartalmát. Ilyenkor egyszerűbb dolgunk, ki kell jelölnünk a szöveget, amit másolni szeretnénk, majd megnyomni a **SHIFT+F2**-es funkcióbillentyűt. Hatására az alábbi üzenet jelenik meg a státuszsorban:

Copy to where?

Hova másoljon? A kurzort, ami most egy szürke vonal, odavisszük, ahova a kijelölt szöveget másolni szeretnénk, és lenyomjuk az **ENTER** billentyűt.

Fordítva is csinálhatjuk, először a célhelyre állunk a kurzorral, lenyomjuk a **SHIFT+F2**-t, ekkor kiírja a státuszsorba, hogy

Copy from where?

Honnan másoljon? Kijelöljük a másolandó szöveget, és **ENTER**-t nyomunk. A kijelölést egy vékony aláhúzásjel mutatja.

Mozgatás (Move)

A másolással ellentétben, mozgatásnál az eredeti helyen nem marad meg a szöveg, hanem áthelyeződik.

Ha a **vágólapot** használjuk, akkora vágólapra az **Edit/Cut** (vágás) paranccsal visszük fel a szöveget. Ennek hatására kivágja (törli) a kijelölt szöveget. A kurzor pozíciójába később bármikor beragaszthatjuk az **Edit/Paste** paranccsal, amíg felül nem írtuk a vágólap tartalmát, pontosan ugyanúgy, amint a másolásnál. Tehát a kettő között csak a vágólapra helyezésben van különbség.

Ha **nem a vágólapot** használjuk szintén hasonló a teendők mint a másolásnál, csak az **F2**-es funkcióbillentyűt most **SHIFT** nélkül kell leütnünk. Tehát kijelöljük a mozgatandó szöveget, leütjük az **F2**-es funkcióbillentyűt, ekkor a képernyő státuszsorában megjelenik a

Move to where?

üzenet: Hova mozgassam? A szürkévé változott kurzort a célhelyre visszük és lenyomjuk az **ENTER** billentyűt.

Itt is eljárhatunk fordítva, először a célhelyre állunk, **F2**-t ütünk, utána kijelöljük, hogy mit mozgasson, és az **ENTER** billentyűvel nyugtázzuk a műveletet. A kijelölés során a második módszernél aláhúzásjel mutatja a mozgatandó szöveget.

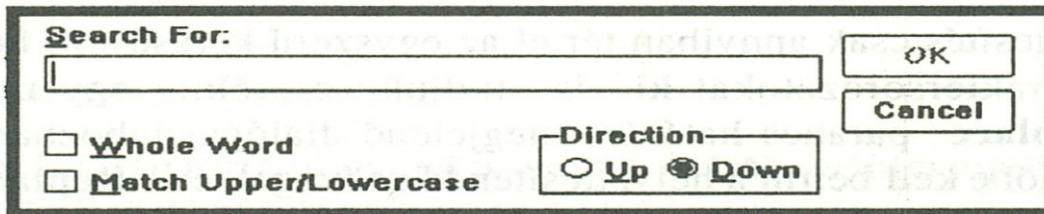
Törlés

Az egyszerű, egy-egy karaktert érintő törlést a *Szöveg javítása* részben már ismertettük. Hosszabb szöveg törlése nagyon fáradságos lenne karakterenként. A törlendő szövegrészt először is ki kell jelölni. Ha úgy gondoljuk, hogy a későbbiekben szükségünk lesz rá, az **Edit/Cut** parancsot használhatjuk, ilyenkor – ahogy ezt az előző részben leírtuk – a vágólapon megmarad a szöveg. Ha nem lesz szükségünk később a szövegre, a **DEL** billentyűt kell lenyomnunk. A törlés megtörténik, és a vágólap tartalma nem változik.

A szöveg keresése, helyettesítése

Keresés

Ha egy hosszú dokumentumban egy adott karaktersorozatot kell megkeresnünk, elég nehéz dolgunk lenne, az **Edit/Search...** parancs nélkül.



3-2. ábra

A parancs hatására megjelenő dialógusdobozban (3-2. ábra) a **Search For** : rubrikába kell írni a keresendő szöveget. Eldönhetjük, hogy a kurzor pozíciójától előre (**Down**), vagy visszafelé (**Up**) keressen. Ha a **Whole Word** kapcsolót bekapcsoljuk, csak akkor találja meg a keresett karaktersorozatot, ha előtte és utána szóköz van, azaz különálló szó. (A négyzet alakú kapcsolók akkor vannak bekapcsolva, ha bennük **X** van. A kapcsolót átállítani egérrel lehet ha rákattintunk, vagy billentyűvel, az **ALT** és az aláhúzott betű leütésével. A **Match Upper/Lowercase** bekapcsolásakor a program megkülönbözteti, a kis- és a nagybetűt. Például a Kovács nem egyenlő a kovács-csal.

Ha lenyomtuk az **ENTER** billentyűt vagy rákattintottunk az **OK** felíratra, rááll az első találatra, azt inverzen jeleníti meg. Ha nem csak egy, hanem több előfordulását keressük a beírt karaktersorozatnak, akkor bármikor tovább kerestethetünk a **SHIFT + F4** -es funkcióbillentyű se-

gítségével. Ha elértük a dokumentumunk alját, és további keresési parancsot adunk ki, a program az alábbi kérdést teszi fel nekünk:

```
Reached end of document. Continue search at
beginning? Yes. No.
```

Elértük a dokumentum végét. Folytassuk-e a keresést az elején? Igen. Nem.

Amennyiben nem talál többet a megadott karaktersorozatból, a következő üzenetet jeleníti meg:

```
Search text not found. OK.
```

A keresendő szöveg nem található. Keresni nemcsak "igazi" karaktereket lehet, hanem speciálisakat is, például: a lehetséges elválasztójelet a [^]-beírásával, vagy az oldaltörés jelet a [^]d -vel.

Helyettesítés

A helyettesítés csak annyiban tér el az egyszerű kereséstől, hogy a megtalált karaktersorozatokat ki is tudjuk cserélni egy másikra. Az **Edit/Replace** parancs hatására megjelenő dialógusdobozban a **Search For** mezőbe kell beírni a helyettesítendő szöveget, és a **Replace With**-be pedig azt, amire ki akarjuk cserélni.

Search For:
[Text Input Field]

Replace With:
[Text Input Field]

Whole Word **Confirm Changes**
 Match Upper/Lowercase

OK
Cancel

3-3. ábra

A helyettesítési művelet mindig *előre* történik, azaz a kurzor pozíciójától a szöveg vége felé. Ha elértük a lap alját megkérdi, hogy folytassa-e a dokumentum elejéről. (Lásd a keresésnél!) A dialógusdobozban megadhatjuk, hogy mielőtt kicserél valamit, kérdezze megengedélyezzük-e? Ha

a **Confirm Changes** kapcsolót bekapcsoljuk, a találatot inverzen kijelöli, és a következő kérdést teszi fel:

Replace selecting? Yes. No. Cancel.

Cserélje-e a kijelölt karaktersorozatot? Ha a **Yes** -t választjuk, kicseréli a találatot, és rááll a következőre, ha a **No**-t választjuk, nem cserél, és szintén rááll a következőre, és végül a **Cancel** kiválasztásakor befejezi a cserélési műveletet. Ha nem szeretnénk, hogy minden találatnál feltegye a kérdést, ebben a dialógusdobozban is kikapcsolhatjuk a **Confirm** kapcsolót, azaz automatikussá tehetjük a cserét.

***Figyelem!** Csak akkor kapcsoljuk ki, ha egészen bizonyosak vagyunk abban, hogy nem cserélhet rosszat! Szavak cseréje során, ha nem különálló szavakat cserélünk, vigyázni kell a ragok és névelők egyeztetésére!*

Állományműveletek

A dokumentum elmentése

A munkánk során a számítógépünk operatív tárában dolgozunk. A szövegszerkesztőből történő kilépéskor, a gép kikapcsolásakor vagy egy hirtelen áramszünet esetén a munkánk elvész. Ahhoz, hogy megmaradjon, el kell menteni valamelyik háttértárra (hajlékonylemezre vagy winchesterre), egy állományba (file-ba). Az állománynak azonosító nevet kell adnunk. Válasszuk ki a **File** menüből a **Save** (mentés) parancsot!

Az alsó dobozban kiválaszthatjuk az aktuális meghajtót, illetve könyvtárat, ahova menteni akarunk. Ha sok könyvtár és meghajtó szerepel a listában, a gördítősávot kell használni a választáshoz. A listában a könyvtárak abc rendben szerepelnek, a lista végén a meghajtók betűjeleit találjuk.

A listában ráállunk a kiválasztandó elemre, és megnyomjuk az **OK** gombot. Kis gyakorlat után gyorsabban is tudunk választani, ha az egérrel egymás után kettőt kattintunk. Az aktuális könyvtár, illetve meghajtó a könyvtári lista felett látható. Egy könyvtári szinttel feljebb úgy juthatunk, hogy a lista tetején lévő két pontra kattintunk. (Ha a dialógusdobozban lévő **Options** parancsra rákattintunk, újabb beállításokat adhatunk meg: például, hogy ne Word for Windows formátumban mentse el a dokumen-

tumunkat a program, hanem más programok számára is olvasható formában. (Ezeket a programokat az installálás során adhattuk meg.) Az állomány nevét a felső keretbe kell beírni. A kiterjesztést nem kell írunk, a program automatikusan a név után írja: **.DOC** (documentum). A file nevének megadása után kattintsunk **OK** parancsra, vagy üssünk **ENTER**-t.

Jegyezzük meg, hogy a dokumentumainkat hova mentjük, mert a winchesteren lévő sok száz vagy ezer file között nehezen találjuk meg, ha nem tudjuk hol keressük! (A keresést megkönnyítheti némileg a **File/Find Search** parancs.)

A program az alábbi dialógusdobozt jeleníti meg mielőtt elmentené az állományt. Ez a doboz az állományhoz tartozó információk összessége (**Summary Info**). Tartalmazza felül az állomány nevét, és a kiválasztott meghajtót és könyvtárat. A **Title** sorba (címsor) írjuk a mű címét, a **Subject** (tárgy) sorba a mű témáját, **Author** (szerző) sorba a program beírja a szerző nevét, amit a program installálásakor megadtunk. Ezt az adatot itt meg tudjuk változtatni. A **Keywords** (kulcsszó) sorba azokat a kulcsszavakat érdemes beírni szóközzel elválasztva, amelyek alapján később vissza akarjuk keresni az állományt. És végül a **Comments** (megjegyzés) rovatba a kiegészítő információkat írhatjuk. A **Statistics** gomb kiválasztásával a dokumentumunkról szinte mindenre kiterjedő statisztikai kimutatást tudunk kapni.

File Name:	TVSZOVI.DOC	OK
Directory:	C:\WINWORD	Cancel
Title:	MTV szövegszerkesztés	Statistics...
Subject:		
Author:	Gerő Judit és Raich Gábor	
Keywords:	MTV szövegszerkesztés WinWord	
Comments:		

3-4. ábra

Ezt a táblázatot nem kötelező kitölteni, elég kiválasztani az **OK**-t. Ha kitöltjük, módunk van a táblázat első négy sora alapján a **File/Find** parancs **Search** funkciója segítségével visszakeresni a dokumentumunkat. Megadhatjuk, hogy mikor készített (**Date Created**), illetve mentett (**Date Saved**) állományokat keresünk. Csak azokon a meghajtókon és

könyvtárakban keres, amelyeket felsoroltunk a **Search List** mezőben. (A listaelemeket vesszővel, illetve pontosvesszővel kell elválasztani, attól függően, hogy a Windows Controll Paneljén az International paraméterek között mi van beállítva.)

A dokumentumot gyakran érdemes menteni, kb. 10-15 percenként. Így nem fordulhat elő, hogy véletlen hiba miatt több órai munkánk elvész. Az első mentést követően, amikor már megadtuk a file nevét és helyét, a mentési parancs kiadása után azonnal végrehajta az utasítást, nem jeleníti meg a dialógusdobozt. Ha más néven, vagy más helyre is szeretnénk elmenteni az irományainkat, akkor a **File/Save As** parancsot kell kiadni. Ilyenkor megjelenik az előbb ismertett dialógusdoboz és összefoglaló képernyő.

A dokumentumok visszatöltése (megnyitása)

Egy elmentett szöveg visszatöltése a memóriába kétféle módon történhet. Az utoljára használt négy állomány neve a **File** menü alján látható. Ha ezek közül szeretnénk egyet megnyitni, az egérrel rákattintunk a nevére, vagy a billentyűzeten a név előtt álló sorszámot leütjük. Ha nem a listából választunk, a **File/Open** parancsot kell kiválasztanunk. A dialógusdoboz közepén lévő keretben kijelöljük azt a meghajtót, illetve könyvtárat, amelyikben a dokumentum van, és a bal oldali keretből kiválasztjuk a dokumentumot. Ez utóbbi listában is abc rendben szerepelnek a dokumentumok, még hozzá csak azok, amelyeknek a kiterjesztése **.DOC**.

A dokumentumok lezárása

Amennyiben egy dokumentummal a munkát be akarjuk fejezni, de a WinWord-ből még nem akarunk kilépni a **File/Close** parancsot használjuk. Megkérdezi a program:

Save Changes to? Yes. No. Cancel.

Elementse-e a megadott állományt? Igen, Nem. Ha a **Cancel** -t választjuk, érvényteleníti a parancsot.

Kilépés a programból

Ha befejeztük a munkát, soha ne tegyük, hogy a programból, illetve a Windows-ból való kilépés előtt kikapcsoljuk a gépet! Megsérülhetnek a file-ok, ha a kilépés "adminisztrációját" a program nem tudja végigcsinálni. Kilépni a File/Exit paranccsal tudunk. A rendszer egyesével megkérdezi minden nyitott dokumentumról, amiben változtattunk az utolsó mentés óta, hogy mentse-e:

Save Changes to "név"? Yes. NO. Cancel

Új dokumentum készítése

Amikor beléptünk a programba, egy üres domumentum-ablakot láttunk a képernyőn. A címsorban *Document1* név látszódott. Az első mentés után ez a név a címsorban az általunk megadott névre cserélődött.

Ha még egy új dokumentumot akarunk készíteni, válasszuk ki a File/New parancsot. A program megkérdezi, hogy dokumentum vagy minta (Template) állományt akarunk-e megnyitni. Mi egyelőre nem tudjuk kezelni a mintaállományokat, így ne változtassunk a dialógusdoboz felkínált beállításán, és maradjunk a dokumentum (Document) mellett. Egy üres képernyőt kapunk, és a címsorban a *Document2* felírat látszik. (Vagy az a sorszám ahányadik dokumentumot nyitottuk meg.)

Több dokumentum együtt kezelése

Anélkül, hogy a részletekbe belemennénk, néhány szót kell szólnunk erről is. Tehát, ahogy az előző pontban leírtuk, meg tudunk nyitni egy új dokumentumot, illetve több régibet egymás után. Minden megnyitási parancs (File/Open) hatására egy-egy új ablak nyílik. Az ablakok (dokumentumok) között a Window paranccsal tudunk váltani. A Window menü alján az összes megnyitott dokumentum neve látszik. Vagy a névre kattintunk az egérrel, vagy a név melletti sorszámot leütjük. Ha több ablakot együtt akarunk látni, a Window/Arrange All -t válasszuk ki. Ha újból csak egy ablakot szeretnénk látni, az aktuális ablak jobb felső sarkában lévő felfelé mutató nyílra kell kattintani az egérrel, vagy például a billentyűzetten a CTRL + F10 -et leütni.

Ahogy leírtuk tehát, másolni, mozgatni adatokat ugyanúgy lehet, mint ahogy egy dokumentumon belül, csak a cél és forráshely kijelölése

között még ablakot is kell váltani a Windows menüben. (Ha a képernyőn látszik mind a két ablak, csak az egérrel kell átkattintani a másikra.)

A szöveg formázása

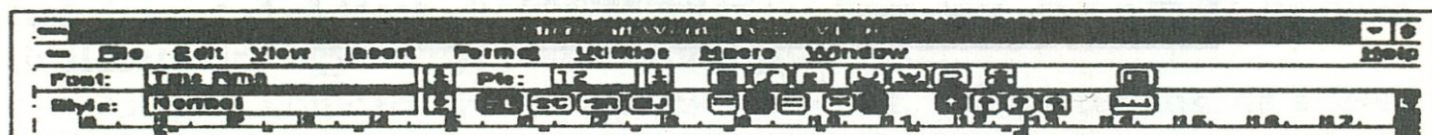
Már tudunk szöveget beírni, elmenteni, visszatölteni, de nem tudjuk formázni. A következő részben a legfontosabb formázási lehetőségeket mutatjuk be. Ugyanazt a hatást többféleképpen is elérhetjük, de helyszűke miatt itt nem ismertetünk minden lehetőséget.

Először is meg kell határozni, hogy mit akarunk formázni. Formázni lehet:

- tetszőleges számú kijelölt karaktert,
- bekezdéseket, bekezdések tabulátorait
- szekciót
- dokumentumot

Karakterek formázása

Ahhoz, hogy egy vagy több karaktert formázni tudjunk, ki kell jelölni azokat. (A kijelölés módszerét a *Szövegkijelölés* részben ismertettük). Leggyorsabban a menüsor alatti szalagsor (Ribbon) segítségével tudunk formázni az egérrel. (3-5. ábra)



3-5. ábra

Nézzük balról jobbra a lehetőségeket. A Font mezőt ha legördítjük (a nyílra kattintva), az adott nyomtatóhoz használható betűtípusok listáját láthatjuk. A gördítősáv segítségével tudunk le-fel mozogni a listában. A kijelölt szöveg felveszi a kiválasztott betű típusát.

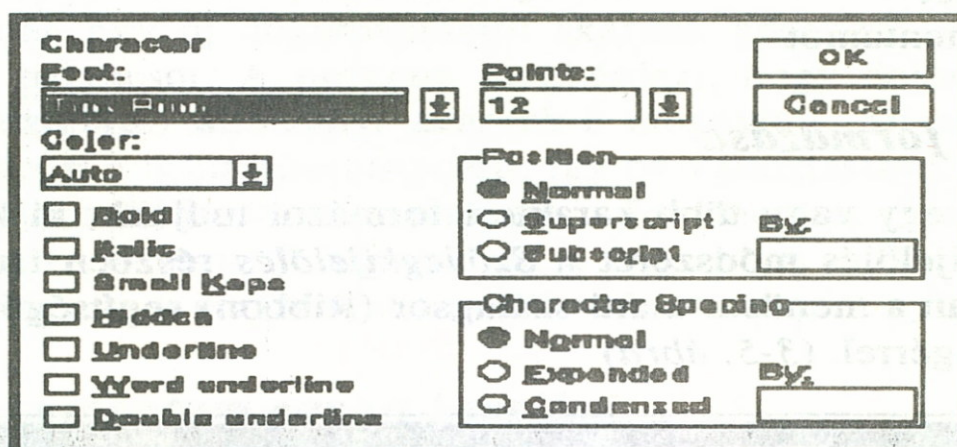
A Pts mezőben a betű méretét tudjuk megadni pontokban. (A pont egy nyomdai mértékegység, kb. 0.3 milliméter. Az újságok általában 8-9, a könyvek 10, a brosrák 12-14 pontos betűvel készülnek.) A méretet a listából válasszuk ki, annak ellenére, hogy beírni is lehet. A listában

megjelenik az összes méret, amit a nyomtatónk a kiválasztott betűtípusban nyomtatni tud.

Ha a **B**-re kattintunk, a karaktereink **Bold** (*vastagítottan*) jelennek meg, ha az **I**-re **Italic** (*dőlt*) formában. Ha a **K** -ra (*Small Kaps*) kattintunk, a betűink kisméretűek maradnak ugyan, de nagybetűkké alakulnak. Az **U** (*underline*) aláhúzást eredményez. A **W** (*Word underline*) szintén aláhúzza a szöveget, de csak a szavakat, a szóközöket nem. A **D** (*double underline*) dupla vonallal húz alá. A **+** (plusz) jellel felsőindexbe kerül a kijelölt szöveg, a **-** (mínusz) jellel pedig alsóindexbe.

Ha az így formázott szöveget eredeti állapotába vissza akarjuk állítani, nyomjuk le a **CTRL + szóköz** billentyűkombinációt.

A felsoroltakon kívül még néhány tulajdonságot tudunk a kijelölt szövegekhez rendelni a **Format** menü **Character...** dialógusdobozá segítségével. (3-6. ábra)



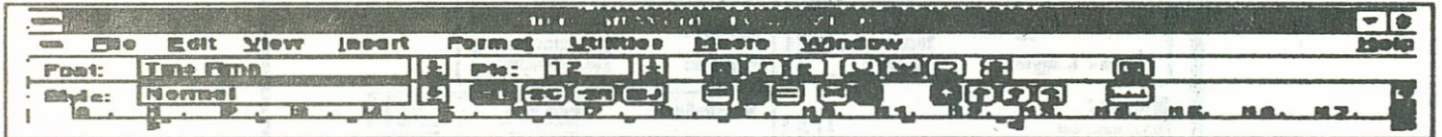
3-6. ábra .

A **Hidden** (láthatatlan) kijelöléssel akkor élünk, ha a nyomtatásban nem akarjuk, hogy a kijelölt szöveg látszódjon.

A képernyőn a **View/Preference** parancsában adjuk meg, hogy látszódjon-e vagy sem a láthatatlan szöveg. Ha látszik, egy vékony szürke aláhúzás jelzi, hogy rejtett tulajdonságú. Használhatjuk például megjegyzésekhez ezt a lehetőséget. A szöveg függőlegesen fel (**Superscript**) és le (**Subscript**) elmozdítható, széthúzható (**Expanded**) és összenyomható (**Condensed**) a **By** mezőben megadott mértékben. A megadást célszerű pontokban (**Pts**) megadni. Növelni vagy csökkenteni az értékeket 0.25 pontonként lehet.

A bekezdések formázása

Bekezdés az, amit a gépelés során egy **ENTER** billentyűvel zártunk le, vagyis két bekezdésvégjel között van. Először megnézzük, hogy hogyan lehet egy-egy, a kurzor pozíciójában lévő bekezdést formázni, illetve több kijelölt bekezdést. A képernyőn a **vonalzósorban** (Ruler – 3-7. ábra) választhatjuk ki az egérrel a leggyakoribb beállításokat, és dialógusdobozból a ritkébbakat.



3-7. ábra

Középen az **L** (left), **C** (center), **R** (right) kockára kattintva, a bekezdés a margóhoz képest balra, középre vagy jobbra igazítható. A **J** (justify) beállítás a kiegyenlítést eredményezi, azaz a szöveg egyaránt illeszkedik a jobb és a bal margóhoz, úgyhogy a kiegyenlítéshez szükséges szóközöket a szavak közé egyenletesen elosztva beszúrja a program. (Ha nem használunk elválasztást, és a magyar nyelv általában hosszú szavait írjuk, a kiegyenlített szöveg általában csúnyán laza lesz.)

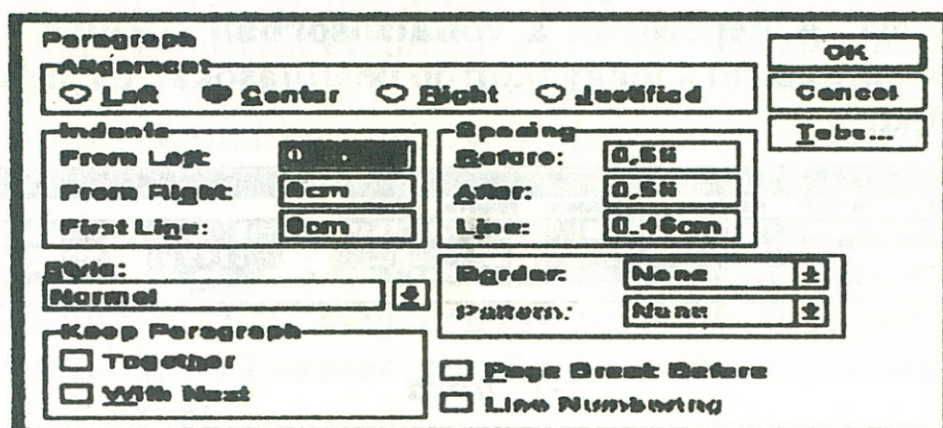
A következő csoport a bekezdésen belüli **sorközöket** állítja. Lehet 1, 1.5 és 2-es sorközt beállítani. Az első bekezdésköz négyzetének hatására a bekezdések előtt nem hagy ki üres sort, a második négyzet kiválasztására egy sornyi távolságot hagy.

Az igazi vonalzósor több funkciót is betölt. Váltani tudunk, ha a felső sor végén lévő kis vonalzócskára kattintunk. Az egyik esetben a sor elején és végén **háromszögek** látszódnak. Ez az aktuális bekezdés margóit jelenti. Ha a jobb oldalra ráállunk az egérrel, és elmozdítjuk a háromszöget, ezzel a bekezdés jobb oldali margóját (ami független a dokumentumunk margójától) tudjuk megváltoztatni. A dokumentum margóját egy függőleges vonal jelzi, bár csak akkor látszik, ha az aktuális bekezdés jobb margó-háromszöge nem takarja. A bal oldali háromszög két fél háromszögből áll. Az alsóval a teljes bekezdés bal oldali margóját tudjuk állítani, a felsővel pedig az első sorét. Ebben az vonalzóállásban a nulla pont a dokumentum margója.

Ha átváltunk a vonalzócskával egy következő vonalzóra, a dokumentum beállított margóit látjuk egy-egy **szögletes zárójellel**. Ha ezeket elmozgatjuk, a teljes dokumentumra fog vonatkozni a beállítás. A nulla

pont a papír szélén van ebben az állásban. (A harmadik vonalzósor típus a táblázatok szerkesztésekor jelenik meg.)

Mielőtt a tabulátorokat megtanulnánk kezelni, nézzük meg a **Format/Paragraph...** dialógusdoboz beállításait! (3-8. ábra)



3-8. ábra

Az **Alignment** sorban a vonalzósor középső piktogrammjaival azonos beállításokat tudunk megadni. Az **Indents** keretben az első sor margóját, illetve a bal és a jobb oldali margókat tudjuk megadni az aktuális bekezdésre. A **Spacing** keretben a bekezdés előtti és utáni távolságot, illetve a sorok közötti távolságot tudjuk tetszőleges méretűre beállítani. (A li a sort jelenti.)

A bekezdések köré vonalak húzhatók (**Border**). Körbe (**Box**), bal oldalra (**Bar**), fölé (**Above**), illetve alá (**Below**). Megadható, hogy milyen vonalat (**Pattern**) húzzon, vékonyat (**Single**), vastagot (**Thick**), duplát (**Double**), és árnyékoltat (**Shadow**). Ez utóbbit csak az aláhúzásnál és a keretezésnél lehet használni.

Ha a **Page Break Before** kapcsolót bekapcsoljuk, az adott bekezdés új oldalra kerül. A **Keep Paragraph Together** kapcsoló hatására a rendszer nem engedi meg, hogy a bekezdés közben lapot dobjon. Ha nem fér el az adott oldalon a bekezdés, az egészet átviszi a következő oldalra. (Vigyázzunk, hogy ne kapcsoljuk be olyan bekezdésben, ami nem fér el egy oldalon, mert a program elkezd keresni egy olyan oldalt, ahová egybefüggően letehetné a bekezdést, és nem hagyja abba a keresést.) A **Keep Paragraph With Next** kapcsolóval elérhetjük, hogy az aktuális bekezdés mindig a következő bekezdéssel azonos lapon jelenjen meg. Minden cím bekezdéshez kapcsoljuk be, nehogy előálljon az a csúnya

eset, hogy egy cím a lap alján van egyedül, és az, aminek a címe, az már a következő oldalra került.

A tabulátorok beállítása

A programban 4 féle tabulátor van: balra, középre, jobbra igazodó és tizedes.

```

tabulátor
|
balra
középre
jobbra
667678.45443 (tizedes)

```

A tabulátor típusa tehát azt szabja meg, hogy a tabulátor pozíciójához képest milyen irányban legyen a szöveg. A vonalzósorban ilyen sorrendben szerepelnek a tabulátorok. Rákattintunk a kiválasztandó tabulátorra, majd a vonalzón odakattintunk, ahova a tabulátort helyezni szeretnénk. A vonalzósorban a beállított tabulátor bal oldaláról törli a normál (egyenletes) tabulátorokat (fejreállított T betű). A normál tabulátorokat a program automatikusan helyezi el a **Format/Document** dialógusdobozban megadott távolságra. Típusa balra igazított. A szöveg írása során, ha lenyomjuk a **TAB** billentyűt, elugrik a szöveg a következő tabulátorpozícióig. Egy-egy tabulátorpozíciót nagyon egyszerűen tudunk törölni: a vonalzósoron ráállunk a törlendő tabulátorra, lenyomjuk az egér gombját, és míg lenyomva tartjuk, "lehúzzuk" a munkaterületre. A **Format/Tabs...** dialógusdobozban pontosan beírhatjuk az egyes tabulátorpozíciókat, típusokat, és kitöltésüket (**Leader**). Ez utóbbi azt jelenti, hogy ahol megnyomtuk a **TAB** billentyűt, onnan a következő tabulátorpozícióig kipontozza, illetve vonalat húz. Tartalomjegyzékben így készítjük a jobbra tabulált oldal-szám és a szöveg közötti pontozást. A tabulátor beállítások mindig egy-egy bekezdésre vonatkoznak.

Stílusok

A bekezdések formázásához célszerű stílusokat (Style) használni. A stílus nem más, mint az eddig ismertetett formátumbeállítások eltárolt összessége. Ha egy bekezdés külalakját beállítottuk a kívánt formátumra, ezeket a beállításokat elnevezhetjük egy stílusnévvel, és a jövőben, ha egy másik bekezdésnek is ilyen külalakot akarunk adni, csak hozzá kell rendelni ugyanazt a stílusnevet. Nézzük meg részletesebben, hogy hogyan is tudunk létrehozni egy stílust, és hozzárendelni egy bekezdéshez.

Stílus létrehozása. Az egyik legegyszerűbb módszer, hogy a vonalzósor **Style** mezőjére kattintunk kettőt, amikor az inverzzé válik, elkezdjük beírni az új stílus nevét, majd leütjük az **ENTER** billentyűt. A program megkérdezi:

Define Style "név" based on selection? Yes. No

Létrehozza-e az aktuális beállításokkal a "név"-vel megadott stílust. Ha **Yes**-t választunk, a névhez hozzárendeli az aktuális bekezdés összes beállítását, a stílust hozzárendeli a bekezdéshez, és a stílusok listájába abc rendbe felveszi a nevet.

Stílus hozzárendelése egy bekezdéshez: Annak a bekezdésnek tetszőleges pontjára állunk, amelyikhez stílust szeretnénk rendelni. Ezután a vonalzósor **Style** mezője melletti nyílra kattintunk, és a listából kiválasztjuk a hozzárendelendő stílus nevét. Ha nem látjuk a listán a nevet, a gördítősávot kell használnunk. Amikor rákattintunk a névre, a hozzárendelés azonnal megtörténik. A bekezdések stílusnevei a képernyő szélén folyamatosan láthatók, ha a **View/Preference** dialógusdobozban a **Style Area Width** mezőbe beírunk egy értéket, például 1.5 cm-t. Ha ez az érték kevésnek bizonyul, és nem látszanak a nevek teljesen, az egérrel az elválasztóvonalra állunk, és a nyíl akakult kurzort jobbra balra mozgathatjuk. (A **View** parancs **Page** neve mellett ne legyen pipa).

Amíg egy bekezdéshez nem rendelünk stílust, az **normal** stílusú lesz. Ha egy bekezdést egyedileg újraformáztunk, és ugyanazt a stílust újból hozzárendeljük, megkérdezi a program, hogy az új beállításoknak megfelelően változtassa-e meg a stílusbeállításokat.

Redefine style "név" based on selection?

Ha **Yes**-t választunk, minden bekezdés külalakja megváltozik, amelyet ugyanezzel a stílusnévvel formáztunk.

A **Format/Define Styles..** dialógusdobozban lépésről lépésre tudjuk a stílus összes tulajdonságát beállítani. (Minden újabb stílus beállítása előtt megkérdezi a program, hogy az előző stílus beállításait tárolja-e.) Az **Options** parancsot kiválasztva, még további beállítási lehetőségeket tudunk megadni. Például megadhatjuk, hogy az adott stílus melyik stílus tulajdonságait vegye át, illetve melyiktől függjön (**Based On**). Azaz, ha a nem specifikus tulajdonságokat az "anya" stílusban megváltoztatjuk, az ebben is meg fog változni. A **Next Style** mezőben azt állíthatjuk be, hogy ha az adott stílussal írt bekezdés után **ENTER**-t ütünk, milyen stílusú bekezdés következzen. A **Merge** segítségével át tudunk venni más dokumentumban, illetve mintaállományban lévő stílusokat az aktuális dokumentumba. A másik dokumentumot a már ismert állomány választási dialógusdobozból lehet kiválasztani. A program figyelmeztet arra, hogy az összefűzés után az azonos stílusnevekből csak egy lesz, és az a behozott stílus beállításait veszi fel.

Ha egy bekezdést egyedileg formáztunk, és utólag szeretnénk, hogy minden egyedi beállítás megszűnjön, csak a stílusban beállított tulajdonságok maradjanak, a bekezdésben állva a **CTRL + X** billentyűkombinációt kell lenyomnunk.

A stílusok használata megkönnyíti a tartalomjegyzék készítését is, ha a címekhez a program által felkínált **Heading 1-9** stílusneveket használjuk. (A formátumokat szabadon állíthatjuk, csak a név kötött.)

A szekciók formázása

Először azt kell tisztázni, hogy mi is az a szekció (**Section**). A szekció nem más, mint a szövegnek egy önállóan formázható egysége. Például szekciókhoz lehet több hasábos nyomtatást, vagy függőleges pozicionálást (fel, középre, elosztott) rendelni. Egy dokumentumon belül akárhány szekció lehet. Új szekciót az **Insert/Break** dialógusdoboz segítségével tudunk kezdeni. Ekkor eldönthetjük, hogy a szekció új oldalon kezdődjön (**Next Page**), páros (**Even Page**), illetve páratlan oldal elején (**Odd Page**), vagy a szövegben folyamatosan jelenjen meg (**Continuous**). A szekció határait dupla szürke vonal mutatja a képernyőn. A a szekció sorszáma a státusz-sorban az oldalszám mellett leolvasható (**Sec 1**).

A szekció formátumbeállításait a **Format/Section...** dialógusdobozban tudjuk megadni (3-9. ábra).

Section OK
Cancel

Columns
Number:
Spacing:
 Line Between

Section Start:
New Page ±

Include Footnotes

Line Numbers
 Line Numbering

Start At #: Per Page
From Text: Per Section
Count By: Continue

Vertical Alignment
 Top Center Justify

3-9. ábra

Több hasábos beállításnál megadhatjuk a hasábok számát (Number) és a hasábok közötti távolságot (Spacing), illetve, hogy húzzon-e vonalat a program a hasábok közé (Line Between). A képernyőn a több hasáb csak akkor jelenik meg, ha a View/Page be van állítva. A Section Start mezőben felül tudjuk bírálni azt, hogy hol kezdődjenek a szekciók. Ha a Line Numbering kapcsolót beállítjuk, nyomtatásnál megszámozza a sorokat.

Dokumentum formázása

A dokumentum formázása során a Format/Document parancs segítségével az egész file-ra vonatkozó beállításokat tudunk megadni (3-10. ábra).

Document OK
Cancel

Page Width: Height:

Default Tab Steps:

Margins
Top: Left: Gutter:
Bottom: Right: Mirror Margins

Footnotes
Print at: ±
Starting Number:
 Restart # Each Section

Template:
 ±
 Widow Control
Set Default

3-10. ábra

Beállíthatjuk az oldal szélességét (**Width**) és magasságát (**Height**), a tabulátorok távolságát, ha nem egyedi beállítást használunk (**Default Tab Stops**).

A margóbeállítások során megadhatjuk a felső (**Top**) és alsó (**Bottom**) margókat. A jobb és a bal margó beállítása attól függ, hogy kétoldalas dokumentumot készítünk-e. Ha igen, a **Mirror Margins** kapcsolót be kell kapcsolni. Ilyenkor a szélső margóbeállítások mellett az **Inside** (belső) és **Outside** (külső) felirat jelenik meg. A **Gutter** (kötés) margó kétoldalas beállításnál a belső margóhoz adódik hozzá, egyoldalas esetén pedig a bal oldali értékhez.

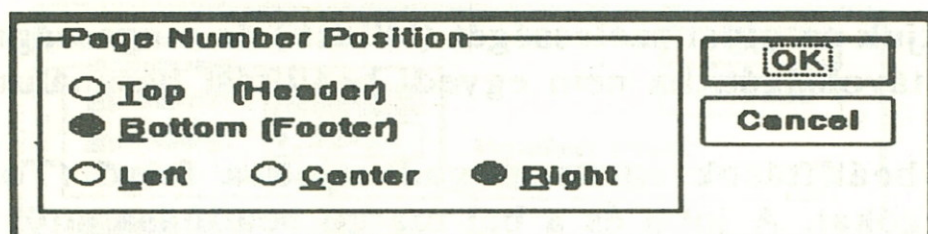
Amennyiben a **Widow Control** (árva sorok) kapcsolót bekapcsoljuk, a program gondoskodik arról, hogy egy bekezdés első vagy utolsó sora ne maradjon egyedül a lap alján vagy tetején. Azaz, ha egy bekezdésből csak egy sor férne el a lap alján, az egész bekezdést átviszi a következő oldalra, illetve, ha csak egy sort kéne átvinni a bekezdésből a következő oldalra, akkor legalább kettőt visz át.

Nyomtatás

A dokumentumunk nyomtatása előtt célszerű megnézni a végleges nyomtatási képet a **File/Print Preview** parancs segítségével. Néhány másodpercnyi várakozás után megjelenik a képernyőn a nyomtatási kép kicsinyített mása. A képernyő tetején kis menügombok jelennek meg. A nyomtatást a **Print** gombbal is kezdeményezhetjük. A **Boundaries** parancs kiválasztásával a képernyőn határoló vonalak jelennek meg a margók, lábjegyzetek, fejlécek és láblécek körül. Ha az egérrel ezekre ráállunk, és lenyomott egérgomb mellett ezeket elmozgatjuk, megváltoztathatjuk a felsorolt elemek helyzetét, méretét. A **Print Preview** képernyőből az **ESC** billentyű lenyomásával tudunk például kilépni.

Az előző bekezdésben néhány új fogalmat is felsoroltunk, amiről eddig még nem beszéltünk. Nézzünk meg néhányat közülük!

A **fejléc** (**Header**) és **lábléc** (**Footer**) a szekciókra jellemző, minden oldal tetején, illetve alján megjelenő szöveg, aminek a tartalmát közösen lehet megadni. Tipikus fejléc például a mű címe és a szerző, vagy az oldalszám. Ezt az **Insert/Page Number** paranccsal tudjuk elhelyezni.

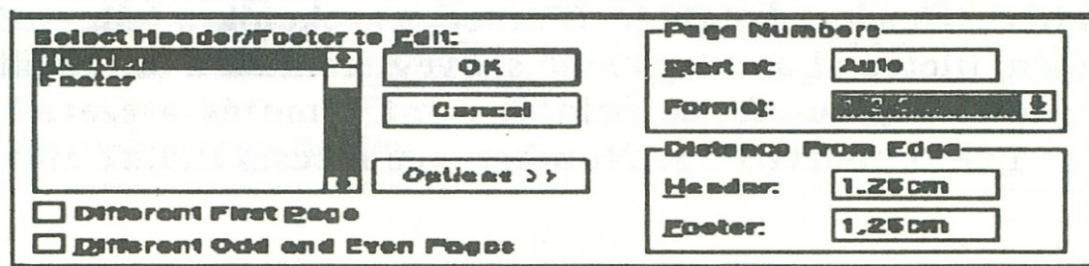


3-11. ábra

A dialógusdobozban (3-11. ábra) beállíthatjuk, hogy a lap-tetején (Top) vagy alján (Bottom) szeretnénk az oldalszámot, valamint azt, hogy vízszintes irányban hova, balra (Left), középre (Center), illetve jobbra (Right). A beállításnak megfelelően minden oldalra ezek után automatikusan felkerül az oldalszám. Az oldalszám a képernyőn csak akkor látszik, ha a View/Page menüszó mellett pipa van. Előfordulhat, hogy a szám helyett { PAGE } látszik. Mi ennek az oka?

A program számtalan feladatot ugynevezett *mezőkkel* old meg. A mezők mindig kapcsos zárójelben vannak. Összesen 52 db. mező van, amit vagy begépelhetünk, vagy az Insert/Fields.. dialógusdobozból választhatunk ki. A mezőkben szerepelhetnek kiegészítő információk, paraméterek, amikkel pontosítani tudjuk a hatásukat. Például az oldalszámnál így tudjuk megadni, hogy római vagy arab számmal jelenjen meg, vagy hogy ne egytől kezdje a számozást. Ennek technikai részleteibe itt nem megyünk bele. Hogyan lehet megszüntetni a "mezős" kijelzést? Az első, amit ellenőriznünk kell, hogy a Show All kapcsoló ki legyen kapcsolva (a Ribon sor végén a csillag ne legyen inverz, vagy a View/Preferences dialógusdobozban a Show All kapcsoló ne legyen bekapcsolva.) A második, hogy a View/Field Codes menü mellett ne legyen pipa, azaz ne legyen bekapcsolva a mező kód megjelenítés.

E kis kitérő után nézzük tovább a fejlécs és láblécet! A fejlécbe, illetve a láblécbe tetszőleges szöveg írható. Először kapcsoljuk ki a View/Page megjelenítést, majd válasszuk ki az Edit/Header/Footer parancsot. Kattintsunk rá az Option gombra, ekkor az alábbi dialógusdoboz jelenik meg. (3-12. ábra)



3-12. ábra

Először is ki kell választani, hogy fejléctet vagy lábléctet frunk. Aztán meg kell határozni, hogy az első oldalt ugyanolyanra szeretnénk-e (**Different First Page**), hogy a jobb és bal oldalakra azonos tartalmat akarunk-e (**Different Odd and Even page**). Ha különbözőt választunk, mindegyiket külön külön tudjuk megadni. Az **Options** segítségével beállíthatjuk a kezdő lapszámot (**Start at**), és a szám formát (**Format**), valamint a fej és lábléc helyét a lap alsó, illetve felső szélétől (**Distance From Edge**). Ha minden beállítottunk, a képernyőnk kettéosztódik, és az alsó részben tudjuk beírni a szöveget. A bal oldali három piktogramm valamelyikére rákattintva rendre az oldalszámot, a dátumot és az időt tudjuk elhelyezni a kurzor pozíciójába. A program automatikusan formázza a fej- és lábléctet a **Header**, illetve a **Footer** stflusokkal. Ha nem tetszik az alapértelmezésbeli külalak, formázhatjuk egyedileg vagy a stflusok segítségével izlésünk szerint.

A tényleges nyomtatást a **File/Print** paranccsal kezdeményezzük. Megadhatjuk, hogy hány másolatot kérünk (**Copies**), és mely oldalakat akarjuk kinyomtatni. Ha az **All-t** választjuk, minden oldal nyomtatásra kerül, egyébként a **From** (megtől), **To** (meddig) mezőkbe kell beírni a nyomtatni kívánt oldalak sorszámát.

Ízelítő

A teljesség igénye nélkül megemlíttünk még néhányat szolgáltatást a **Word for Windows** lehetőségei közül.

- A szövegeinkhez lábjegyzeteket fűzhetünk, melynek helye beállítható.
- Sok címből álló dokumentumnál előre elkészíthetjük a dokumentumunk vázlatát, a címek sorozatát, és utólag írhatjuk közé a szövegeket. Ha használjuk a vázlatkészítőt (**Qutline**), egy-egy parancs segítségével 9 szint mélységű bekezdés számozást tudunk előírni, tartalomjegyzéket készíttetni.
- A program kényelmes tartalomjegyzék és tárgymutató szolgáltatással is rendelkezik, az **Insert** menü parancsai között.
- Tetszőleges beosztású táblázatokat, bonyolult képleteket tudunk készíteni és formázni a program segítségével.
- Gyakran előforduló karaktersorozatokhoz egy rövidítést rendelhetünk, melyeket egy rövidítésszótárban (**Glossary**) tárolhatunk.

Elegendő később a rövidítést begépelni, lenyomni az F3-a funkcióbillentyűt, és a helyén megjelenik a teljes szöveg.

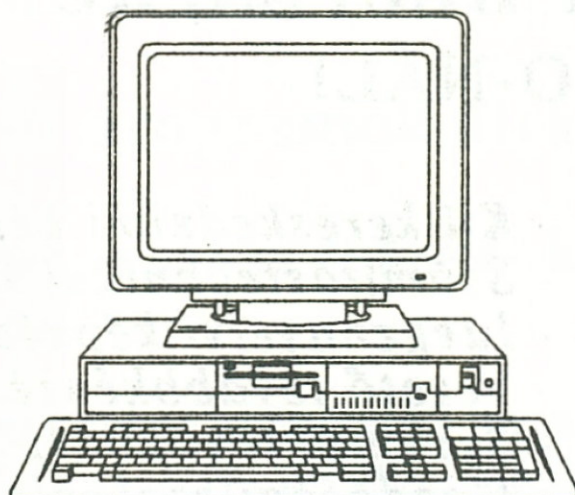
- Ismeri a matematikai logikai műveleteket, függvényeket, akár kis kalkulátornak is használhatjuk.
- **Körleveleket** tudunk nyomtatni tetszőleges adatbázisból. A nyomtatáshoz, bonyolult feltételeket írhatunk elő, hogy melyik rekordot kell kinyomtatnia.
- **Angol helyesírásellenőrző és szinonima szótár** is megtalálható. Az angol elválasztási rendszer elég jól használható magyar szövegek elválasztásához is.
- A szövegeinket, táblázatainkat **rendezni** is tudjuk bármelyik oszlop alapján.



Transzfer

Kereskedelmi és Szolgáltató Kft.

A modern technika használatának lehetőségét kínálja Önnek tanfolyamaival
a TRANSZFER KFT, a CONTROLL RT Kelet-Magyarországi képviselője!



Számítástechnikai szakképesítést nyújtó tanfolyamaink:

- Személyi-számítógép kezelő 1 félév
- Számítógép programozó 3 félév

Egyéb tanfolyamaink:

- Személyi-számítógép kezelése 20 és 50 órás
- WordPerfect 5.1-szövegszerkesztő használata

*Cégek jelentkezése esetén
egyedi igényeknek megfelelő tematikát készítünk!*

Jelentkezésüket az alábbi helyeken várjuk:

NYÍREGYHÁZA

• Transzfer Kft, Hősök tere 7.
Karányi Tibor ☎ (42) 13-843

DEBRECEN

• Kölcsey Ferenc Művelődési Központ, Hunyadi u. 1-3.
Vlajk Tibor ☎ (52) 13-977

MÁTÉSZALKA

• 2. sz. Általános Iskola, Hősök tere 2.
Tóth Józsefné ☎ (44) 10-117



KÜLKERINFO Kft.

MINDENT EGY HELYEN, KOMPLEX KÉPZÉS A KÜLKERINFO-NÁL!

Fő profilunk:

- *Külkereskedelmi képzés*
- *Számítástechnikai képzés*
- *Idegennyelvi képzés*
- *Vezető továbbképzés*
- *Szakmai továbbképzés*
- *Gazdasági szolgáltatás*

**Specialitásunk: a SZÁMÍTÁSTECHNIKAI
KÉPZÉS-en belül**

- *alapfokú személyi-számítógépezői,*
- *személyi-számítógép szoftver-üzemeltető,*
- *középfokú számítástechnikai tanfolyamok.*

Újdonságaink:

- *Idegennyelvű levelezői képzés - PC szövegszerkesztői ismeretekkel*
- *Általános ügyviteli képzés - számítástechnikai ismeretekkel*
- *Titkárnőképzés - kiadványszerkesztői ismeretekkel*
- *Külkereskedelmi áruforgalmi képzés - számítástechnikai és vámügyi ismeretekkel*

TANULJON A KÜLKERINFO-NÁL!

OTK^{Kft.} OKTATÁSI ÉS TOVÁBBKÉPZŐ Kft.

Debrecen, Petőfi tér 12.
Tel.: 52/12-503

Nyíregyháza, Báthory u. 15.
Tel.: 42/18-488

4. Táblázatkezelés számítógéppel

A táblázatkezelő programok általános ismertetése

Milyen feladatokhoz és miért érdemes használni a táblázatkezelő programokat?

A táblázatkezelő programok a számítások, az adminisztráció, a nyilvántartás és mindenféle olyan munka megkönnyítésének eszközei, amelyhez egyébként kockás papírt és kis zsebszámológépet használánk. A táblázatokban, amelyekben bizonyos adatok egymástól is függenek, minden alkalommal, amikor bármelyik adat megváltozott, többé kevésbé az egész táblázatot előlről kezdhettük kiszámolni. Ezt a problémát egycsapásra megoldották a táblázatkezelő programok. Egyetlen billentyűvel az egész táblázatot újraszámíttathatjuk, ahányszor csak szükséges. Minden kiinduló adat változásának hatását egyenként vizsgálhatjuk anélkül, hogy nagyobb erőfeszítést kellene kifejtenünk, és túl sok időt kellene elpocsékolnunk. A táblázat rubrikái, cellái között igen bonyolult összefüggéseket írhatunk elő. Ezeknek a bonyolult összefüggéseknek a leírásához a táblázatkezelő programok több száz függvényt kínálnak segítségül. Mint minden sokat tudó segédeszköz, megtanulása energiabefektetést igényel, de ez megtérül a megtakarított időben, a bosszúságok elmaradásában, a kényelmes munkában, az eredményben.

Mivel a táblázatkezelő programok számtalan szolgáltatást kínálnak, ezért a fejezet megírása során nem törekedhettünk teljességre.

A táblázatkezelő programok alapvető szolgáltatásai

A táblázatkezelő programok szolgáltatásai két csoportba sorolhatók: az alapvető és a specifikus, az adott programra jellemző szolgáltatásokra.

Alapvető szolgáltatás a cellák közötti összefüggések meghatározása, a gyors újraszámolás, az adatok rendezése, a számok megjelenési formátumának beállítása (kerekítések stb.), a sorok és oszlopok beszúrása, törlése, a cellák és cellacsoportok másolása, mozgatása stb.

Specifikus, bonyolultabb szolgáltatások körébe tartozik a konszolidálás azaz az adatok csoportosítása több táblázatból, a keresés és helyettesítés, az adatok elforgatása, a visszafelé számolás (az eredmény alapján, kiszámolja a kiindulási adatokat), a rajzolás és így tovább.

Ebben a rövid kis fejezetben sajnos részletesen csak az alapvető szolgáltatásokról tudunk szólni, egyrészt terjedelmi korlátaink miatt, másrészt ezeket a szolgáltatásokat ma már minden táblázatkezelőben megtaláljuk, nem csak ebben az egy megvalósításban, amiben a könyv példáit készítettük.

Könyvünk az EXCEL for Windows programot mutatja be. Miért pont az EXCEL-t, kérdezhetnénk? Néhány válasz: Azért, mert e könyv szerzői azt a célt tűzték maguk elé, hogy elsősorban a Windows-os alkalmazások körét mutatják be, azért, mert az EXCEL felhasználóbarát, egy bizonyos szintig az elsajátítása gyerekjáték, azért, mert az EXCEL gyönyörű, és nem utolsósorban, mert szolgáltatásainak köre szinte elképzelhetetlenül széles.

A mai táblázatkezelő programok integrált programok

A számítógépek felhasználásának elterjedésével a felhasználókban kialakult az az igény, hogy különböző feladataik megoldásához ne kelljen más és más programot használniuk, megvenniük és megtanulniuk, ne kelljen az adatkonverzióval babilonizálniuk, hanem minél több feladatra legyen alkalmas egyetlen program. Ez az igény készítette a szoftverfejlesztőket az úgynevezett integrált programok elkészítésére, amelyek egyszerre több feladattípus megoldására alkalmasak. Így lehetővé vált, hogy a táblázatkezelő programokkal lehessen üzleti grafikát készíteni, adatbázist kezelni, esetleg szöveget szerkeszteni vagy a gépek között kapcsolatot teremteni. Ilyen programcsomag a LOTUS 1-2-3, a SYMPHONY, a FRAMEWORK és a QUATTRO is.

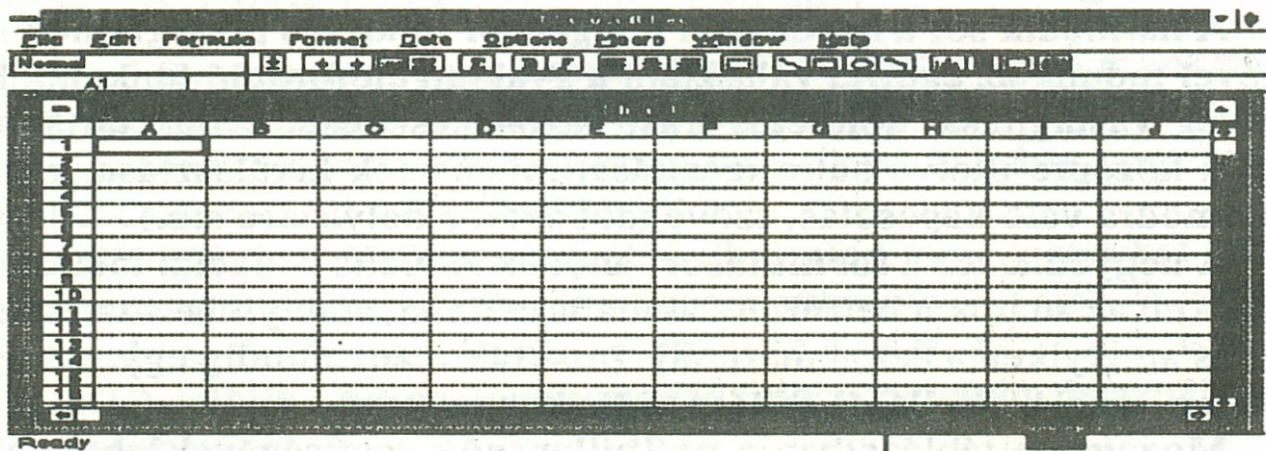
A grafikonok rajzolásához csak ki kell választani a táblázatnak azt a területét, ahol az ábrázolandó adatok szerepelnek, és meghatározni, hogy milyen típusú grafikont szeretnénk (kör-, vonal-, oszlop-, háromdimenziós grafikon stb.) Természetesen a grafikont bármikor tovább lehet csinosítani, feliratokkal, magyarázó szövegekkel ellátni, berácsozni, színeznit, elforgatni, vegyes ábratípust választani stb.

Az adatbáziskezeléshez a táblázat adatainak bizonyos formai követelményeknek eleget kell tenniük, és akkor adatbázisként kezelhetők. Ez

annyit jelent, hogy keresési kritériumokat lehet összeállítani, amelyek alapján a táblázatból kikereshetünk adatokat, kivonatokat készíthetünk, statisztikai számításokat végezhetünk. A kifejezetten adatbáziskezelést igénylő feladatokhoz nem érdemes táblázatkezelő programot használni, hanem csak olyanokhoz, amelyeknél az elsődleges feladat a táblázatkezelés.

A képernyő és a táblázat ismertetése

Tájékozódás, mozgás és választás a képernyőn



4-1. ábra

A EXCEL elindítása után (a Windows-ból az EXCEL ikonjára kattintva kettőt) a 4-1. ábrán szereplő képernyőt látjuk. Külön kell választani az alkalmazói ablakot, amely jelen esetben maga az EXCEL, és a dokumentum ablakot, amely most egy üres táblázat, és a neve egyelőre SHEET1.XLS.

Megjegyzés: Amennyiben az elkövetkező feladatok közben bármikor pihenni szeretne vagy félbehagyni kényszerül a tanulást, el kell mentenie a táblázatát, hogy a munkát ott folytathassa, ahol azt abbahagyta. A mentés és a visszatöltés lépéseit később, a *Táblázatok mentése visszatöltése* című részben ismertetjük!

Néhány szó az EXCEL képernyőjéről. A képernyő felső sora a címsor, ahol az alkalmazás neve szerepel. Alatta a menüsor. A menüből – mint minden Windows alkalmazásban – legkönnyebben az egér nyílának segítségével tudunk választani: rámutatunk a nyíllal a kiválasztandó helyre, és egyet kattintunk az egér bal gombjával. Billentyűzetről a menük kiválasztásához az ALT billentyűt és a menüpontok aláhúzott betűit kell együtt lenyomni.

A legördülő menüből csak azok az almenüpontok hajthatók végre az adott pillanatban, amelyek feketék. A szürke menüpontok nem. (Pl. ha a vágólap üres, akkor nem lehet a vágólap tartalmát beragasztani: Edit/Paste) Bizonyos menük mögött három pont jelzi, hogy kiválasztásuk esetén egy párbeszéddoboz (dialog box) fog megjelenni, ahol a legkülönbözőbb dolgokat lehet beállítani, illetve a beállításokat pontosítani.

A harmadik sor a képernyőn a segédsor, ahol kis piktogramok segítségével tudunk az egérrel választani a gyakran előforduló funkciók közül (szöveg vastagítása, dőltbetűs írás, vázlat készítése, cellák tartalmának jobbra középre illetve balra igazítása, rajzelemek kiválasztása, grafikon üzemmódra való kapcsolás, szövegdobozok elhelyezése stb.)

A negyedik sor a formulasor, ahol az aktuális cellacím melletti üres sorba kell az adatokat beírni. Az aktuális cella az, amelyikbe adatot tudunk írni, és amelyiken a cellakurzor áll. A cellakurzor mindig egy vastagított téglalap, amelyet celláról cellára tudunk mozgatni.

Mozogni a táblázatban a nyílbillentyűk segítségével lehet celláról cellára, képernyőoldalanként le-fel (Page Up, Page Dn), jobbra-balra (CTRL jobb vagy bal nyíl együtt). Mozogni az egérrel pedig úgy, hogy az egér nyílát a kívánt helyre visszük és megnyomjuk a bal gombot. Közvetlenül is lehet egy adott cellára ugrani az ugrási funkcióval (GoTo). Az ugrási parancsot kétféleképpen is használhatjuk. Egyrészt a Formula menüben a Goto-t választva, másrészt az F5-ös funkcióbillentyű segítségével. A megjelenő panelen a Reference sorba kell begépelni a kívánt cella címét, ahova ugrani akarunk, és kiválasztani az OK "nyomógombot". Az aktuális cella a megcímzett cella lesz.

A program fejlesztői a gyorsabb kezelhetőség kedvéért a leggyakrabban használt parancsokat egy-egy funkcióbillentyűhöz, vagy a funkcióbillentyűk kombinációihoz rendelték, hogy ne kelljen mindig a menüből és almenüből választani. A funkcióbillentyűk kombinációja azt jelenti, hogy nemcsak egy billentyűt, hanem kettőt, esetleg hármat együtt kell lenyomni a kívánt parancs végrehajtásához. Ilyen billentyűk a SHIFT,

a CTRL (kontroll) és az ALT billentyűk, amelyekkel együtt lenyomva egy billentyűt más és más parancsot hajt végre a rendszer.

A képernyő alsó sora az üzenetsor, ahol a program pillanatnyi állapotáról és néhány billentyű (pl. INS, END, NumLock stb.) bekapcsolt voltáról tudunk információt szerezni. Itt kapunk a parancsok végrehajtása közben esetleg utasításokat is.

Az eddig vázolt képernyő-elrendezés a segédsor kivételével szinte minden táblázatkezelő programra jellemző, legfeljebb a menüsor nem felül van a képernyőn, hanem alul vagy oldalt, esetleg nem látszik mindig, hanem egy billentyű leütésére (gyakran a / jelre) jelenik meg.

A sorok, oszlopok és cellák azonosítása

A képernyő fennmaradó része a munkaterület, ahol maga a munkatáblázat található. A táblázatok sorokból és oszlopokból állnak. Mindegyiknek van egy azonosítója, ami lehet sorszám vagy betű. Néhány programban, így az EXCEL-ben ezt választani lehet.

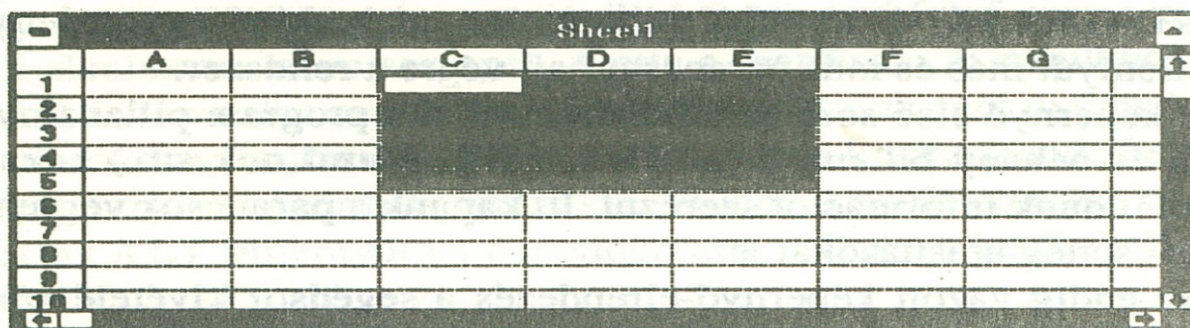
Az EXCEL-ben 16 384 sor és 256 db oszlop van. A sorokat sorszámokkal, az oszlopokat az abc betűivel azonosíthatjuk. Az első 26 oszlopot egy betűvel (A, B, C), majd kettős betűkkel (AA, AB, AC, ...AZ, BA, BC, ...BZ, ...IV) jelölik. Az oszlopok megadásához egyaránt használhatunk kis- és nagybetűket.

A sorok és oszlopok metszéspontjában vannak a cellák. A cellát az oszlop és a sor koordinátájával azonosítjuk. Például a 10. sor 3. oszlopában lévő cella címe (azonosítója) a C10.

Az adatokat cellákba írjuk. Ez a formázás, az adattárolás elemi egysége.

A tartomány fogalma és kijelölése

A táblázatkezelés során sokszor nem csak egy-egy cellával szeretnénk dolgozni, hanem cellák csoportjával. A cellák téglalapalakú csoportját a program tartománynak nevezi, és bármely két átlójának koordinátáival azonosítjuk. Pl. C1:E5 a C és E oszlopok, valamint 1. és 5. sorok által határolt tartomány címe (4-2. ábra). A tartományokat az egérrel lehet kijelölni, miközben nyomjuk a bal gombját, vagy billentyűzetről a nyílak és a SHIFT együttes lenyomásával.



4-2. ábra

Ha egyszerre több tartományt szeretnénk kijelölni, akkor az első tartomány kijelölése után a **CTRL** billentyűt lenyomva kell tartani, és a további tartományokat kijelölni. Egy cella is lehet tartomány, pl.: A1:A1.

Az adatok bevitele és javítása

A táblázatba az adatokat általában gépeléssel visszük be. (Lehet más állományból, táblázatból bemásolni vagy az aktuális táblázat már kitöltött adatait átmásolni.) Az adatbefrás során az adatok a formulasorban látszódnak, miközben a sor elején egy **X** és egy pipa jelenik meg. Az adatbevitt az **ENTER** billentyű vagy valamelyik nyílbillentyű lenyomásával, illetve az egérrel a pipára kattintva fejezhetjük be. Ha az adatbevitt abba szeretnénk hagyni, az **ESC** billentyűt kell leütni, vagy a formulasor **X**-ére kattintani az egérrel.

Egy-egy cellába 255 karakternyi szöveget vihetünk be. Ha megtelt az első formulasor, automatikusan megjelenik a következő. Ez nem jelenti azt, hogy az összes beírt karakter látszik a cellában. *A megjelenő karakterek száma a beállított oszlopszélességtől függ.*

A hibás adatot azonnal javíthatjuk, ha az egérrel a hibás karakterre kattintunk, vagy ha az **F2**-es funkcióbillentyűt leütjük, és töröljük vagy beírjuk a hiányzó karaktereket. Törölni kétféleképpen tudunk: a **DEL** billentyűvel a kurzor jobboldalán lévő karaktert, míg a **BACKSPACE** billentyűvel a kurzor baloldalán lévő. Ha már befejeztük az adatbevitt, amikor rájövünk, hogy valamit javítani szeretnénk, először is aktívvá kell tennünk a javítandó cellát, azaz rá kell állni a cellakurzorral, majd a formátumsorra kell kattintanunk az egérrel, vagy meg kell nyomnunk az **F2**-es funkcióbillentyűt.

A javítás egyik módja lehet, hogy az adatot felülírjuk. Úgy csinálunk, mintha az aktuális cellába nem lenne semmi, és elkezdünk gépelni. Az adatbevitel végén a cella eredeti tartalma törlődik, és az új tartalom fog látszani. Itt célszerű megemlíteni a EXCEL-nek azt a szolgáltatását, hogy az utolsó műveletet, beírást, parancsot stb. – néhány esettől eltekintve – vissza tudja állítani, érvényteleníteni tudja. Ez az Edit menüben az első parancs, az UNDO.

Ha egy cella tartalmára egyáltalán nincsen szükségünk, ki kell törölni. Az nem törlés, ha egy szóközzel "felülírjuk", mert az egy szóköz-tartalmú cella. A törölni a DEL billentyűvel vagy az Edit/Clear paranccsal lehet. A megjelenő párbeszédpanelben eldönthetjük, hogy mindent törölünk (All), vagy csak az egyéni formátumokat (Formats), a képleteket, számokat, szövegeket (Formulas), illetve a megjegyzéseket (Note). Ha nem akarjuk, hogy a párbeszédpanel megjelenjen, a CTRL+DEL billentyűkombinációt kell leütnünk.

A program készenléti állapotát a státusz sor baloldalán a Ready, az adatbevitelt az Enter, míg a javítási üzemmódot az Edit felirat jelzi.

Adattípusok

A bevitt adatokat különböző szempontok szerint csoportosíthatjuk. Az első csoportba a konstansok tartoznak. Konstans lehet szám, szöveg, idő vagy dátum. Az adatok másik csoportja a képletek. Képlet lehet egy-egy egyszerű vagy bonyolult összefüggés vagy függvény.

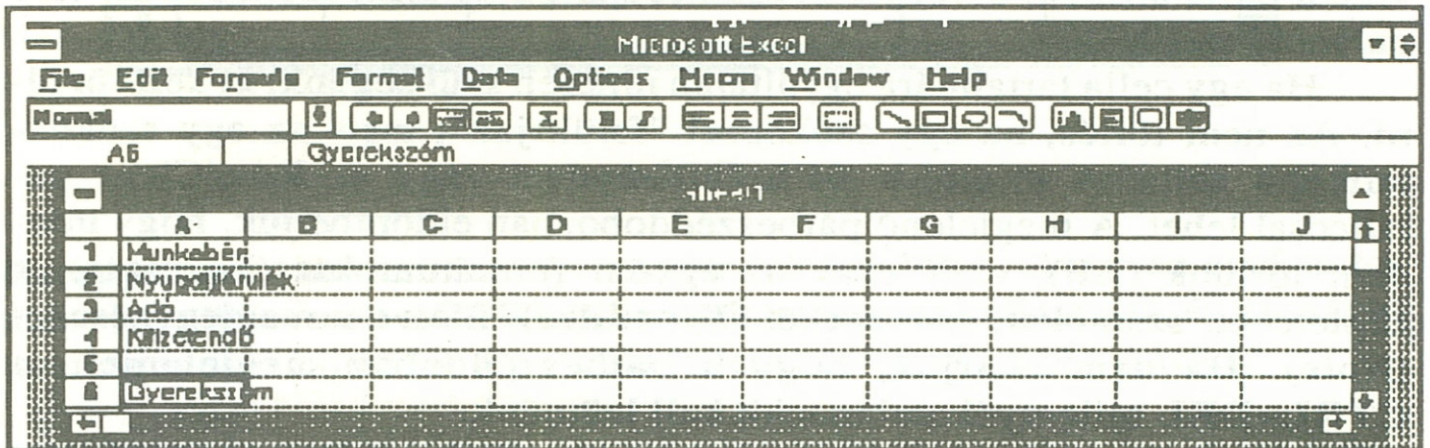
A program az első leütött karakterből tudja, hogy milyen típusú adatot viszünk be. A képletek = jellel vagy a @, +, (, - jel valamelyikével, illetve valamilyen számmal kezdődhetnek. (A program beállítható úgy is, hogy csak az = jelet fogadja el!)

Képletek (formulák) fogalma, beírása

A képletek megértéséhez egy példát hívunk segítségül. A feladat nagyon egyszerű, először egy dolgozónak a munkabéréből kiszámítjuk az adóját, majd a nyugdíjjárulékát és a kifizetendő bérét. Az adószámításnál figyelembe vesszük a gyerekeinek a számát, azaz az adóalapjából levonunk annyiszor 1300 Ft-ot, ahány gyereke van. A példát a megértés kedvéért, most annyira leegyszerűsítjük, hogy egységesen 20%-os adót számolunk. (Természetesen az EXCEL képes arra, hogy egy beírt adótáblából a

jövedelemnek megfelelő adókulcsot és fix összeget kikeresse, és a jövedelemtől függő tényleges adót kiszámolja. De ennek az összefüggésnek a felírásához már egy kicsit többet kell tudni a táblázatkezelőről.)

A táblázat első oszlopába írjuk be a következő magyarázó szövegeket (szövegkonstansokat) (4-3 ábra).



4-3. ábra

A 2. sorban a *Nyugdíjjárulék* szó átlóg a B oszlopba. A program addig engedi meg ezt, amíg a mellette lévő cella üres. Az oszlopok szélességét szabadon állíthatjuk (lásd *A táblázat formázása* fejezetben).

A B1-es cellába beírjuk az első dolgozó munkabérét (pl. 15600). A következő cellába olyan képletet kell írunk, amely kiszámolja a bérből a nyugdíjjárulék értékét.

A képleteket lehet, de nem túl elegáns begépelni, és lehet elegánsan "elmutogatni" a *nyílbillentyűk* vagy az *egér* segítségével. Ez utóbbinál rámutatunk azokra a cellákra, amelyekben szereplő adatokkal számolni szeretnénk.

Nézzük meg mind a három módszert!

Begépeléssel: A kitölteni kívánt cellára (B2) állunk, és a formulasorba írjuk a következőt:

$$=B1*10\%$$

Az adatbevitel végén (ENTER billentyű leütésére, illetve a formulasorban a pipára kattintva), a B2-es cellában megjelenik a számolás eredménye, a cella pillanatnyi értéke, míg a formulasorban – amíg a cellakurzor rajta áll – , a begépelte képlet, a formula látható (4-4. ábra).

The screenshot shows the Microsoft Excel 3.11 interface. The menu bar includes File, Edit, Formula, Format, Data, Options, Macro, Window, and Help. The toolbar contains various icons for file operations and editing. The active cell is B2, which contains the formula =B1*10%. The spreadsheet grid shows the following data:

	A	B	C	D	E	F	G	H	I
1	Munkabér	15600							
2	Nyújtólélek	1560							
3	Adó								
4	Kifizetendő								
5									
6	Gyerekszem								

4-4. ábra

A képlet első karaktere az *egyenlőségjel*, amely jelzi az EXCEL-nek, hogy nem egy B betűvel kezdődő szöveget írtunk be, hanem egy képletet. A B1 azt jelenti, hogy nem igazából a pillanatnyi munkabér (15600) tíz százalékat kell venni, hanem annak a számnak a tíz százalékát, ami most éppen a B1-es cellában van. Ez a táblázatkezelő programok egyik legfontosabb tulajdonsága, azaz nemcsak számokkal, hanem úgynevezett **cellahivatkozásokkal** tudnak dolgozni. Ebből logikusan következik, hogyha megváltoztatjuk a B1-es cella tartalmát, akkor a program automatikusan újraszámolja azokat a cellákat, amelyek a B1-es cellától függenek, példánkban eddig csak a B2-es cellát. Az újraszámolás (recalculation) a második alapvető tulajdonsága a táblázatkezelő programoknak.

Fontos megjegyezni még néhány dolgot. A szorzásjel jele a *, az egyes szám (1) nem keverhető össze a kis l betűvel, mert a program másképp kezeli a számokat és a szövegeket.

A százaléértékeket háromféleképpen is beírhatjuk: 10%, .1 , 0.1 formában. Ez az írásmód feltételezi, hogy a Windows környezet úgy van beállítva (Control Panel), hogy a tizedesjel a pont. (Amennyiben a vesszőt választottuk, figyelni kell arra, hogy a később ismertetendő felsorolásokban nem lehet az elemek elválasztására ezt használni, hanem például a pontosvesszőt.) Végül a számok beírásához még egy megjegyzés: a számokat soha ne válasszuk el ezres helyiértékenként egy szóközzel vagy vesszővel (pl. 12 555.89). Ha ilyen kijelzést szeretnénk látni, ezt a Format/Number dialógus dobozban állíthatjuk be. (Ezt a formázások fejezetben részletezzük.)

A képlet beírása a rámutatással: Ezzel a módszerrel egyrészt rengeteg felesleges billentyűzéstől tudjuk magunkat megkímélni, másrészt kevesebb a hibalehetőség. Nagyobb a valószínűsége, hogy rá tudunk mutatni egy cellára, mint pontosan leolvasni a koordinátáit és utána hibátlanul beírni. Csak a cellahivatkozások beírását tudjuk megspórolni, a számokat, műveleti jeleket nem.

Például a B3-as cellába az alábbi összefüggést kell beírunk:

$$=(B1-(B6*1300))*20\%$$

Nézzük lépésről lépésre a teendőket:

- rá kell állni a kitöltendő cellára: **B3**
- a képletet jelző egyenlőségjelet beírjuk: =
(Bizonyos beállítás mellett az = jelet nem kell kitenni, csak olyan karakterrel kezdeni a beírást, amittől a program meg tudja különböztetni a képletet a szövegtől, pl.: zárójel, szám, előjel stb.)
- az első nyitó zárójelet beírjuk: (
- a felfelé mutató nyílbillentyűvel felmegyünk a **B1**-es cellára. Ahova éppen mutatunk, ott a cella körül egy szaggatott villogó második cellakurzor látható, valamint a formulasorban mindig az éppen mutatott cellának a címe íródik fel. A mutogatás ideje alatt a státuszsorban a **Point** szó látható. Jelen esetünkben kétszer kell a felfelé mutató nyílbillentyűt leütetni, hogy a **B1**-es cellához jussunk.
- mínusz jel beírása: -
- A mínusz jel leütése után a villogó "mutogató" cellakurzor megszűnik!
- második nyitó zárójel beírása: (
- most újból a nyílbillentyűket használva, lemegyünk a **B6**-os cellához. A cella jelen pillanatban üres, azaz 0 értékű. A matematika is megengedi a 0-val való szorzást, tehát 0 gyereknél nem von le a munkabérből semmit.
(Amennyiben osztani akarnánk üres cellával, a **#DIV!0!** hibaüzenet figyelmeztetne a cellában a matematikai hibára. Ha utólag beírunk egy számot az osztó cellájába, a tényleges eredményt

fogjuk kapni, hisz szintaktikailag helyes a képlet, csak éppen az adott pillanatban nem lehet megoldani.)

- következik a műveleti jel, a szorzásjel beírása: *
- begépeljük a levonandó kedvezményt: 1300
- a belső zárójelet bezárjuk: (
(Figyeljük meg, hogy a beírás után néhány másodpercig a zárójelek párok vastagon jelennek meg! Ezzel ellenőrizni tudjuk, hogy minden zárójelek megvan-e a párja, és jó helyen van-e.)
- következik a külső zárójelek bezárása:)
- végül a szorzásjel és a százalék beírása: *20%
- az adatbevitel végén az ENTER billentyűt leütjük, vagy a formulasorban lévő pipára kattintunk az egérrel.

A műveletek eredményét a 4-5. ábra mutatja.

	A	B	C	D	E	F	G	H
1	Munkabér	15600						
2	Nyugdíjjárulék	1560						
3	Adó	3120						
4	Kifizetendő							
5								
6	Gyerekszám							
7								
8								
9								
10								

4-5. ábra

Megnézhetjük, hogyha a B6-os cellába beírjuk a gyerekek számát, akkor az adó értéke hogyan fog változni. (Persze előállhat az az eset, hogy negatív adó jön ki, hisz az egyszerűsítés kedvéért nem figyeltetjük a programmal ezt az esetet.)

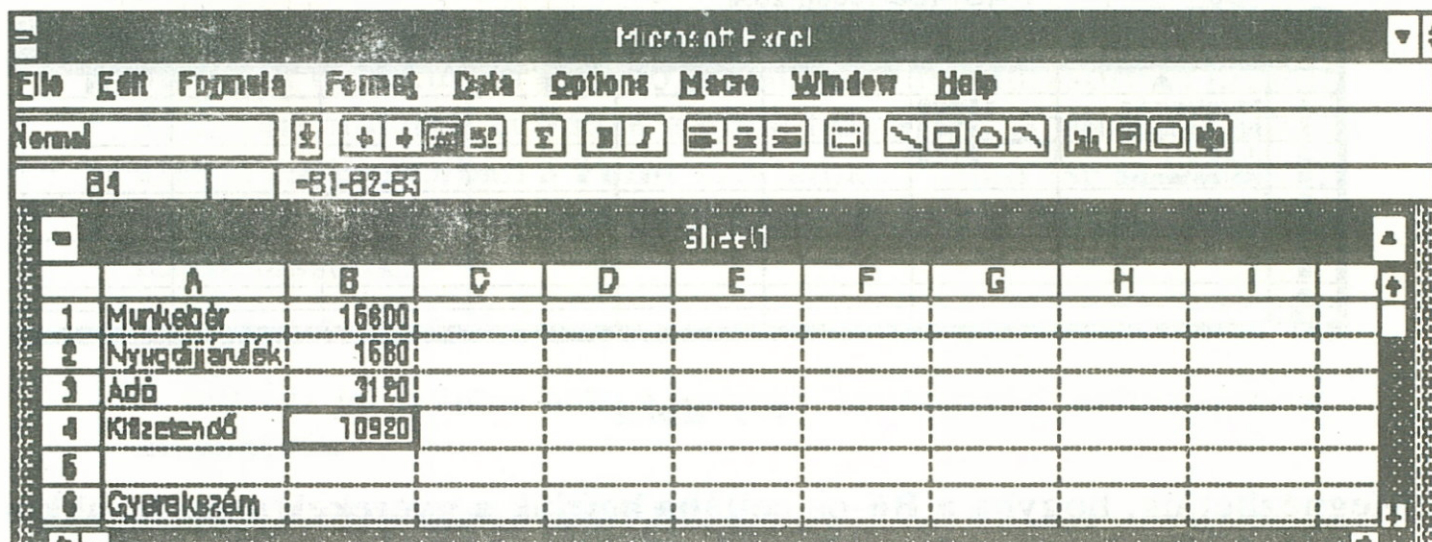
Hátra van a 4. sor kiszámítása, ahol szeretnénk látni a nyugdíjjárulék és az adó levonása után a kifizetendő nettó bért. Ezt a képletet az egérrel fogjuk "elmutogatni". A beírandó képlet:

$$=B1-B2-B3$$

4. Táblázatkezelés számítógéppel

- rá kell állni a kitöltendő cellára: B4
- beírjuk a képletet jelző = jelet
- az egér nyílával elmegyünk a B1-es cellára, lenyomjuk az egér gombját, és a már megismert villogó szaggatott cellakurzor megjelenik.
(Amennyiben a cella, amelyikre mutatni szeretnénk, nincs a képernyőn, akkor akár a gördítősávot, akár a nyílbillentyűket, képernyőváltó billentyűket használhatjuk a cella megkeresésére.)
- a mínusz jelet beírjuk: -
- rámutatunk a B2-es cellára
- a második mínusz jelet is beírjuk: -
- rámutatunk a B3-as cellára
- befejezzük az adatbevitelt

Az eredmény a 4-6. ábrán látható.



The screenshot shows the Microsoft Excel interface. The menu bar includes File, Edit, Formula, Format, Data, Options, Macro, Window, and Help. The toolbar contains various icons for formatting and editing. The active cell is B4, and the formula bar shows the formula =B1-B2-B3. The spreadsheet grid shows the following data:

	A	B	C	D	E	F	G	H	I
1	Munkabér	15800							
2	Nyugdíj járék	1660							
3	Adó	3120							
4	Kifizetendő	10920							
5									
6	Gyerekszám								

4-6. ábra

Az adatbevitel három formája (befrás, rámutatás egérrel vagy billentyűvel) tetszés szerint kombinálható, ahogy szokásaink és a feladat legkényelmesebbé teszi.

Képletek másolása vágólap segítségével

Természetes igény, hogyha eljutottunk idáig a tanulás során, elvárjuk az EXCEL-től, hogy ne csak egyetlen fizetést számoljon ki, hanem akárhányat. Ehhez az kell, hogy a képletek értelemszerűen a többi oszlopban is megjelenjenek.

Először írjunk még 5 fizetést az első sorba (a 4-7. ábra szerint)! Lehet azonosakat is, hogy megfigyelhessük, hogyan befolyásolja a nettó jövedelmet például a gyerekek száma. A gyerekek számát a 6. sorba írjuk.

	A	B	C	D	E	F	G
1	Munkabér	15600	15600	24000	24000	40000	50000
2	Nyugdíj járulék	1560					
3	Adó	3120					
4	Kifizetendő	10920					
5							
6	Gyerekszám		2		2	1	1

4-7. ábra

A képletek másolása a többi oszlopba többféleképpen történhet. Az egyik módszer szerint a másolandó cellákat a vágólapra másoljuk: **Edit/Copy** paranccsal, és a kívánt helyre, helyekre beragasztjuk: **Edit/Paste**.

Kicsit részletesebben. A vágólap a Windows-nak egy olyan tárolóterülete a számítógép memóriájában, amelyre bármelyik Windows-os alkalmazásból "fel lehet tenni" egy kijelölt cellát, tartományt, rajzot, szöveget stb., és ugyanabban vagy egy másik alkalmazásban a kurzor pozíciójába elhelyezni, "beragasztani" azt. A vágólap mindig csak egyetlen egy valami tárolására szolgál, tehát bárholnán valamit ráteszünk, az felülírja a tartalmát. Az, hogy mekkora lehet ez a valami, az a számítógépünk memóriájától függ.

Ha meg akarjuk nézni, hogy az adott pillanatban mi van a vágólapon, akkor vágólapprogramot (Clipboard) elindítjuk a Windows Main ablakból. Az EXCEL vezérlő menüjéből a **Run** parancsot kiválasztva, szintén

megjeleníthetjük a vágólapot. (Itt már találunk olyan lehetőséget a File menüben, hogy a vágólap pillanatnyi tartalmát elmentsük egy önálló állományba, és később használjuk csak fel. Ezeknek az állományoknak a kiterjesztése mindig .CLP)

A vágólapot nem csak másolásra használhatjuk, hanem áthelyezésre is. Ilyenkor az **Edit/Cut** parancsot kell választani. A **Cut** (vágás) hatására az eredeti helyéről kitörli a kijelölt részletet, és a kijelölt célhelyre beteszi, ha az **Edit/Paste** parancsot kiadjuk.

Az EXCEL képes vágólapra helyezni cellákat, tartományokat, képeket, grafikonokat, és amikor a formulasor aktív, annak kijelölt tartalmát is.

A cellák, tartományok másolása, áthelyezése egy kicsit másképp működik, mint általában a Windows-os alkalmazásokban. Eszerint, amikor kijelöljük a másolandó áthelyezendő cellákat, tartományokat, kiválasztjuk az **Edit/Copy** vagy **Edit/Cut** parancsok valamelyikét, és az EXCEL a kijelölt tartományok körül egy villogó szaggatott cellakurzort jelenít meg. Csak addig van a vágólapon a kijelölt tartomány, amíg ez a villogó jel látszik. Ha több másolatot szeretnénk készíteni a vágólap tartalmáról, akkor az **Edit/Paste** parancsot kell használnunk. Ha csak egyet választunk, akkor elég a célhelyet kijelölni az egér vagy a billentyűzet segítségével (ráállni), és megnyomni az **ENTER** billentyűt vagy az **Edit/Paste** parancsot. Ez utóbbinál amíg nem viszünk be újabb adatot, addig a villogó cellakurzor jelzi, hogy még a vágólapon van a kijelölt tartomány, és újabb célhelyet kijelölve, az **Edit/Paste** paranccsal többször elhelyezhető a vágólap tartalma. A **Cut** parancs után azonnal be kell ragasztani a kijelölt tartományt. A **Cut** parancs után csak egyszer tudjuk az **Edit/Paste** parancsot kiadni.

A célhely kijelöléséhez elegendő megadni a célhely bal felső sarkát, vagy többszörös másolásnál a célhelyek bal felső sarkait. Hisz az már megmondtuk a programnak, hogy mekkora az a tartomány, amit át kell másolni vagy mozgatni, most csak azt kell megmondani, hogy hova.

Nézzük meg a másolást lépésről lépésre!

- Jelöljük ki a másolandó területet, a képleteket tartalmazó cellákat, (B2:B4). A kijelölés egérrel a legegyszerűbb, lenyomott egérgomb mellett végighúzzuk a kijelölendő cellákon az egérnyílát. A kijelölendő terület inverz lesz. Billentyűzettel történő kijelölésnél először rá kell állni az első kijelölendő cellára, le-

nyomni és lenyomva tartani a **SHIFT** billentyűt, és a nyílbillentyűkkel bejárni a kijelölendő területet.

- Az **Edit** menüből kiválasztjuk a **Copy** parancsot. (Az egérrel az **Edit** menüsóra állunk, és megnyomjuk az egér gombját, majd a legördülő menüből a **Copy** szóra állunk, és szintén lenyomjuk az egér gombját. Billentyűzetről az **ALT** billentyűvel együtt lenyomjuk az **E** betűt, majd a **C**-t.) A villogó szaggatott vonal jelzi, hogy a kijelölt tartomány a vágólapon van. A státuszsorban az alábbi üzenet jelenik meg:

– Copy (Select destination and press Enter or choose Paste)

Másolás (válaszd ki a célhelyet és nyomd meg az **Enter** billentyűt, vagy válaszd a **Paste** parancsot)

- Ki kell jelölnünk a célhelyek bal felső sarkát. Szeretnénk, ha a képlet a **C2**, **D2**, **E2**, **F2** és **G2**-es celláktól kezdődően bemásolódná a táblázatba. Ezért jelöljük ki a már ismertetett módon a **C2:G2** tartományt.
- Nyomjunk **ENTER**-t, vagy válasszuk ki az **Edit/Paste** parancsot. A villogó keret még megmarad. Ha most azonnal újabb célhelye(ke)t jelölnénk ki, az **Edit/Paste** paranccsal akárhányszor elhelyezhetjük a vágólapon lévő tartományt. Ha valami mást csinálunk, például egy cellába adatot írunk, akkor a villogó keret eltűnik, és a tartomány törlődik a vágólapról. (Ha legördítjük az **Edit** menüt, a **Paste** parancs szürke.) Más alkalmazásba is csak ilyenkor tudunk átvinni táblázatot az **EXCEL**-ből.

Eredményül a táblázatban minden cellában megjelent a jövedelemtől és a gyerekek számától függő eredmény (4-8. ábra).

The screenshot shows the Microsoft Excel interface. The menu bar includes File, Edit, Formula, Format, Data, Options, Macro, Window, and Help. The formula bar displays the formula $= (E1 - (E6 * 1300)) * 20\%$. The spreadsheet below has columns A through H and rows 1 through 6. The data in the spreadsheet is as follows:

	A	B	C	D	E	F	G	H
1	Munkabér	15600	15600	24000	24000	40000	50000	
2	Nyugdíjjárulék	1560	1560	2400	2400	4000	5000	
3	Adó	3120	2600	4800	4280	7740	9740	
4	Kifizetendő	10920	11440	16800	17320	28260	35260	
5								
6	Gyerekszám		2		2	1	1	

4-8. ábra

A relatív és az abszolút cellahivatkozás

Érdekes kérdés, hogy honnan tudta a program kiszámítani például a E oszlopban lévő fizetésből és gyerekszámból a helyes eredményt, amikor a B oszlopra vonatkozóan adtuk meg az összefüggéseket.

A kérdés megválaszolásához álljunk rá például az E3-as cellára! Ha megnézzük a cella tartalmát a formulasorban, azt látjuk, hogy nem pontosan ugyanaz a képlet, mint amit átmásoltunk, hanem:

$$=(E1-(E6*1300))*20\%$$

Ennek az az oka, hogy a cellahivatkozások megadásánál nem azt mondtuk tulajdonképpen a programnak, hogy vegye a B1-es cellát, abból vonja le a B6-os cellának és 1300-nak a szorzatát, és a különbséget szorozza meg 20%-kal, hanem azt, hogy: Vegye a két cellányival felette lévő cella tartalmát, ebből vonja le az alatta három cellányira lévő cella tartalmát szorozva 1300-al, és a különbséget szorozza meg 20%-kal.

Tehát amikor átmásoltuk a fenti összefüggést egy másik cellába, ott is vette az ugyanolyan távolságra és irányban lévő cella tartalmát, és helyesen elvégezte a műveleteket. Ezt a fajta cellahivatkozást **relatív cellahivatkozásnak** nevezzük. Lényege, hogy a cellák másolás után ugyanazzal az összefüggéssel máshol is pontos eredményt adnak anélkül, hogy a képleteket módosítanunk kellene. Gondoljunk bele, ha nem csak

öt dolgozó, hanem többszáz adatát szeretnénk feldolgozni. A relatív cellahivatkozás szintén a táblázatkezelő programok alapvető szolgáltatása.

Vannak azonban esetek, amikor nem szeretnénk, ha a másolás után a cellahivatkozások változnának. Ez akkor fontos például, ha egy cellába beírunk olyan adatot, konstanst, amivel a táblázat legkülönbözőbb helyén szeretnénk számolni, azaz hivatkozunk rá. Példánkban ilyen állandó lehet a nyugdíjjárulék százaléka. Egy cellába, például az A5-be beírjuk: 10%. és szeretnénk, ha minden 2. sorbeli képlet erre a cellára hivatkozna. (Ha megváltoztatják, ne kelljen az összes cellában megváltoztatni, elég legyen ebben az egyben.)

Ha az eddig tanult módon egyszerűen a B2-es cellában lévő képletet módosítanánk a következő szerint:

$$=B1*A5$$

akkor a másolás után a nyugdíjjárulék százalékát a B5-ben, C5-ben stb. keresné. Tehát ez nem jó.

Az EXCEL – és általában a táblázatkezelő programok – az abszolút cellahivatkozás lehetőségével oldják meg ezt a problémát. Ilyenkor a rögzíteni kívánt cellacímbe a sor és az oszlop betűjele elé egy-egy \$ jelet kell tenni (\$A\$5). Ez jelzi, hogy a másolás során a gép ne módosítsa a cellacímeket a relatív helyzetük alapján, hanem mindig "abszolút pontosan" erre a cellára hivatkozunk.

Ha a képleteket valamelyik mutatós módszerrel állítjuk össze, az F4-es funkcióbillentyű többszöri lenyomásával változathatjuk a relatív és abszolút cellahivatkozást. A formulasorban annak a hivatkozásnak a jellegét változtatja, amelyikben a szövegkurzor áll.

Ha ez utóbbit kipróbáltuk látjuk, hogy a \$ jelet a sor, az oszlop, illetőleg egyidejűleg a sor és oszlop azonosítói elé lehet helyezni. Vagyis lehet a relatív és az abszolút cellahivatkozást variálni, vegyes hivatkozást előírni. Lehet, hogy csak az oszlopot "rögzítjük le", azaz csak ahhoz ragaszkodunk, hogy mindig a rögzített oszlopban lévő adattal számoljon (\$A1), míg a sorhivatkozás relatív módon változzon, vagy fordítva, csak a sort rögzítjük és az oszlop változhat (A\$1).

Nézzünk egy példát erre a nem könnyen átlátható, de nagyon hasznos lehetőségre!

Egy kiadóhivatalban, ahol napilap kiadásával foglalkoznak, kalkulálják a várható havi árbevételt. Az árbevétel három dologtól függ, az eladott napi példányszámtól, az adott hónapban lévő napok számától (vannak 28,

4. Táblázatkezelés számítógéppel

29, 30 és 31 napos hónapok) és az újság árától. Mindegyiket változtatva, képet kaphatnak a bevételek alakulásáról. A feltételezések szerint a példányszám 3000 és 10000 között változhat. A kalkulációs táblázat első oszlopába az eladható példányszámot írják 1000-esével növelve. (A program alkalmas tartományok automatikus feltöltésére szám- vagy mértani sorozattal, illetve dátummal a Data/Series parancs segítségével). A felső sorba kerül, hogy hány napos lehet egy hónap és végül egy különálló cellába, például a B16-ba az újság árát írjuk be (4-9. ábra).

	A	B	C	D	E	F	G	H	I
8		28	29	30	31				
9	3 000	1 041 600	1 078 800	1 116 000	1 153 200				
10	4 000	1 388 800	1 438 400	1 488 000	1 537 600				
11	5 000	1 736 000	1 798 000	1 860 000	1 922 000				
12	6 000	2 083 200	2 157 600	2 232 000	2 306 400				
13	7 000	2 430 400	2 517 200	2 604 000	2 690 800				
14	8 000	2 777 600	2 876 800	2 976 000	3 075 200				
15	9 000	3 124 800	3 236 400	3 348 000	3 459 600				
16	Egységár	12.4							

4-9. ábra

Az a feladatunk, hogy egy olyan képletet írjunk be a B9-es cellába, amelyet ha átmásolunk a kalkulációs táblázatunk többi cellájába, mindenhol a helyes eredményt adja, vagyis le tudjuk olvasni, hogy adott példányszám mellett, egy adott hónapban a B16-ba beírt egységár mellett milyen árbevétel várható. Azt is szeretnénk látni, hogyha a B16-ban megváltoztatjuk az egységárat, akár csökkentjük vagy növeljük az árat, ez milyen hatással lehet az árbevétel alakulására. A feladat megoldása során össze kell szorozni a példányszámot a napok számával és az egységárral úgy, hogy a példányszámot rendre az A oszlopból vegye, annak 9., 10. stb sorából, a napok számát a 8. sor B, C, D stb. oszlopából és végül az egységárat mindig a B16-os cellából. A beírandó képlet ezek alapján a következő:

=B\$8*\$A9*\$B\$16

Nézzük meg lépésről lépésre, hogy hogyan lehet a relatív-abszolút cellahivatkozásokat elmutogatni.

- rá kell állni a **B9**-es cellára
- a képletfráshoz szükséges = jelet beírjuk
- ráállunk a **B8**-as cellára, és leütjük az **F4**-es funkcióbillentyűt: ennek hatására a **B8**-as cella **\$B\$8**-ra változik, ha még egyszer lenyomjuk az **F4**-et, a képlet **B\$8** lesz. Nekünk pontosan erre van szükségünk, vagyis a 8. sor rögzítve lett, az oszlop pedig szabadon változhat a másolások után.
- begépeljük a szorzásjelet: *
- rámutatunk az **A9**-es cellára, és háromszor nyomjuk le az **F4**-es funkcióbillentyűt. A formulasorban az aktuális cellahivatkozás a következő lesz: **\$A9**, azaz az **A** oszlop rögzítve van, a sor pedig változhat.
- begépeljük a szorzásjelet: *
- végül ráállunk a **B16**-os cellára, és egyszer lenyomjuk az **F4**-es funkcióbillentyűt: **\$B\$16**, azaz akárhova másoljuk a képletet, mindig erre a cellára hivatkozunk
- befejezzük az adatbevitelt

Ezt az egy képletet kell a kalkulációs tábla összes cellájába átmásolni:

- rajta állunk a **B9**-es cellán
- kiadjuk az **Edit/Copy** parancsot (a **B9**-es cella a vágólapra került, körülötte a villogó szaggatott vonal látszik)
- kijelöljük a célhelyet, ahova másolni szeretnénk: lenyomott egérgomb mellett körberajzoljuk a **B9:E15**-ös tartományt (inverz módon látszik a képernyőn)
- lenyomjuk az **Enter** billentyűt, vagy kiválasztjuk az **Edit/Paste** parancsot

A függvények fogalma, szintaktikája, csoportosítása

Az EXCEL több száz függvényt biztosít a felhasználók gyors és eredményes munkájához. A függvények alapvetően két csoportba sorolhatók: a táblázatkezelő függvényei és a makrózás függvényei. Ez utóbbiak csak a makrókban használhatók. (A makró az EXCEL saját "programnyelve", amellyel automatikussá lehet tenni a munkát, igen bonyolult felhasználói rendszereket lehet írni az EXCEL "alá". Bár a makrózásnak van egy egyszerűbb lehetősége is, amikor gyakran előforduló billentyűleütési sorozatokat megtanítatunk a programmal, és ezt később bármikor végrehajthatjuk, de még ennek ismertetése is túlnyúlik e könyv keretein.)

E kis kitérő után térjünk vissza a táblázatkezelő függvényekhez! A függvény egy segédeszköz a gyorsabb és kényelmesebb munkához. A függvények használatánál néhány szintaktikai szabályt be kell tartani:

függvénynév(argumentumok)

A függvény nevét nem kell megjegyezni és fejből begépelni, hanem egy a képernyőn megjelenő listából választhatjuk ki. Ezt a listát a **SHIFT-F3** együttes lenyomásával vagy a **Formula** menü **PasteFunction...** parancsával jeleníthetjük meg. A listában abc rendben szerepelnek a függvények. A gördítősáv, a nyílbillentyűk vagy a lapozó billentyűk (**PgDn**, **PgUp**) segítségével tudunk le-fel mozogni a listában. Ha leütjük a beragasztani kívánt függvény első betűjét, akkor a kurzor közvetlenül az azzal a betűvel kezdődő első függvényre áll. Innen már hamarabb megtaláljuk azt, amit keresünk. Segítségül a képernyőn láthatjuk a függvény lehetséges argumentumait is.

Az **argumentum** nem más, mint a függvény bemenő adatai (független változói), amivel számolni kell, vagy amit át kell alakítani stb. a függvény jellegéből adódóan. Ha a függvénylista alsó részén látható kis kockába **X**-et teszünk (**Paste Arguments**), akkor a program az argumentumokat is beírja a formulasorba. Ez csak ritkán szerencsés, hisz ekkor külön le kell törölni, és az új argumentumokat beírni. Az argumentumok (listaelemek) elválasztására a **Windows Control Panel**-jén beállított listaelem elválasztó karaktert kell alkalmazni. (Általában a vessző vagy a pontosvessző a szokásos).

Mielőtt a függvények csoportosítását részleteznénk, nézzük meg a leggyakrabban használt függvényt, az összeadó (szumma) függvényt.

Az EXCEL fejlesztői e függvény előhívására külön lehetőségeket is biztosítottak, hogy ki-ki ízlése szerint választhasson. Az előző példánkat felhasználva, ahol a munkabérekől számoltunk adót és nyugdíjjárulékot, nézzük meg, hogy mennyi pénzt kell kifizetni a bankból, hogy mindenkinek ki tudjuk fizetni a fizetését. (Most eltekintünk a címletezési problémától, bár az EXCEL-ben ezt is egyszerű feladat megoldani).

Rá kell állni arra a cellára, ahol az összeget szeretnénk látni, például a H4-esre. Utána a képernyő felső részén lévő piktogrammok közül kiválasztjuk a szumma jelet. A cellában, illetve a formulasorban megjelenik az egyenlőségjel után a SUM(B4:G4) függvény. Honnan tudta a program, hogy mi pontosan ezt a tartományt szeretnénk volna összegezni? Nem tudta, csak felkínált egy logikus lehetőséget. Például a cella melletti összefüggő tartományt, vagy egy előző SUM függvényig terjedő területet stb. A felkínálás során a tartományt rögtön ki is jelölte (inverzen látszik), hogy ne kelljen bűbelődnünk a felkínált tartomány törlésével. Ha mégse ezt a tartományt szeretnénk összegezni, akkor elegendő beírni vagy kijelölni egy másik összegzendő tartományt. De most nem kell semmit törölnünk, módosítani, csak az adatbevitelt befejezni az ENTER billentyű leütésével, vagy a formulasor pipájára kattintani az egérrel (4-10. ábra).

Abban az esetben, ha a függvények nevét mi írjuk be, és nem a listából választjuk ki, akkor természetesen az egyenlőségjelet az első függvény elé nekünk kell beírunk.

The screenshot shows the Microsoft Excel interface. The menu bar includes File, Edit, Formula, Format, Data, Options, Macro, Window, and Help. The toolbar contains various icons, including the Sum (Σ) icon. The formula bar shows the formula =SUM(B4:G4) for cell H4. The spreadsheet below has columns A through H and rows 1 through 6. The data in the spreadsheet is as follows:

	A	B	C	D	E	F	G	H
1	Munkabér	15600	15600	24000	24000	40000	50000	
2	Nyugdíjjárulék	1560	1560	2400	2400	4000	5000	
3	Adó	3120	2600	4800	4280	7740	9220	
4	Kifizetendő	10920	11440	16800	17320	28260	35780	120520
5								
6	Gyerekszám		2		2	1	3	

4-10. ábra

Nézzük meg a függvények csoportosítását, és ennek megfelelően az argumentumokat!

Adatbázis-statisztikai

Dátum és Idő

Információs

Kereső

Logikai

Matematikai

Mátrix

Pénzügyi

Statisztikai

Szöveges

Trigonometriai

A függvény fajtájától függően az argumentumok lehetnek számok, képletek, függvények, cella- és tartománynevek, dátumok és szövegek. A szövegeket mindig *idézőjelek közé* kell tenni, ha függvénybe írjuk. A függvények egymásba ágyazhatók, azaz az egyik függvényben szerepelhet másik függvény.

A függvények ismertetésére itt nem vállalkozhatunk, hisz több száz van, inkább néhányat példákon keresztül mutatunk be, ami segíti a többi megértését is.

A statisztikai függvények csoportjába tartozó átlag **AVERAGE()** függvény az argumentumában felsorolt cellák, tartományok adatainak átlagát számolja ki, de az átlagolandó számokat közvetlenül is fel lehet sorolni benne.

P1.: =AVERAGE(5;7;8;15) esetén a cellába 8.75 kerül.

Ha az átlagos bruttó fizetésre vagyunk kíváncsiak, akkor ráállunk például a H1-es cellára, **SHIFT F3**-al megjelenítjük a függvények listáját, kiválasztjuk az **AVERAGE** függvényt, **OK**-t vagy **ENTER**-t nyomunk. A formulasorban a függvény megjelent, és a gép várja, hogy beírjuk vagy megmutassuk az átlagolandó tartomány(oka)t vagy számokat. Mutassunk rá a **B1:G1**-es tartományra, és az adatbevitel után megjelenik a cellában az eredmény: **28200** (4-11. ábra).

The screenshot shows the Microsoft Excel interface. The menu bar includes File, Edit, Formula, Formát, Data, Options, Macro, Window, and Help. The formula bar displays the formula `=AVERAGE(B1:G1)` for cell H1. The spreadsheet has a single sheet named 'Sheet1' with columns A through H and rows 1 through 6. The data in the spreadsheet is as follows:

	A	B	C	D	E	F	G	H
1	Munkabér	15600	15600	24000	24000	40000	50000	28200
2	Nyugdíjjárulék	1560	1560	2400	2400	4000	5000	
3	Adó	3120	2600	4800	4280	7740	9220	
4	Kifizetendő	10920	11440	16800	17320	28260	35780	120520
5								
6	Gyerekszám		2		2	1	3	

4-11. ábra

Végül nézzük meg a HA IF() logikai függvényt!
A HA függvény mindig három operandusból áll:

`=IF(feltétel;igaz;hamis)`

A függvény először elemzi, hogy az első operandusba beírt feltétel igaz-e, ha igaz, akkor a cellába a második operandus kerül, ha hamis a harmadik. A második és harmadik operandus lehet szám, függvény, képlet, idézőjelek között beírt szöveg stb.

Az IF() függvényt kétféleképpen is bemutatjuk. Először megnézzük a fizetéseket, és egy szöveges értékelést írunk róla: 18000 Ft alatt *kicsi*, 18000-25000 *normál*, 25000-45000 *nagy*, 45000 felett *jelentős*. A fenti szöveges értékeléshez rögtön szükségünk lesz a függvények egymásbaágyazására. Az ilyen feladatoknál különösen ügyelnünk kell arra, hogy minden nyitó zárójelnek legyen párja. Az ellenőrzésben sokat segít a zárójelek megvastagodása, amikor az egyik alatt elmozgatjuk a kurzort, vagy éppen gépeljük a zárózárójelet.

A B7-es cellába írjuk először a függvényt, majd átmásoljuk a C7:G8-as tartományba. Mivel relatív cellahivatkozásokkal dolgozunk, a másolás után mindenhol a saját oszlopában lévő fizetést fogja értékelni a függvény. A beírandó képlet a következő:

`=IF(B1 < 18000;"kicsi";IF(B1 < 25000;"normál";
IF(B1 < 45000;"nagy";"jelentős")))`

Ha a képernyőn megtelik az első formulasor, akkor egy következő sort kezd a program. Nekünk folyamatosan kell gépelnünk, nem szabad semmilyen elválasztó jelet használnunk a sor végén. A beírás során sehol sem szerepelhet szóköz.

Másoljuk át a B7 cella tartalmát a C7:G7-as tartományba! Nézzük meg mi történik, ha a 1. sorban egy fizetést átírnunk! A 7. sorban azonnal vált az új értéknek megfelelő szöveges értékelésre (4-12. ábra).

The screenshot shows the Microsoft Excel interface. The formula bar for cell B7 contains the formula: `=IF(B1<18000;"kicsi";IF(B1<25000;"normál";IF(B1<45000;"nagy";"jelentős")))`. Below the formula bar, the spreadsheet grid is visible, showing a table with columns A through H and rows 1 through 8. The data in the table is as follows:

	A	B	C	D	E	F	G	H
1	Munkabér	15600	15600	24000	24000	40000	50000	282
2	Nyugdíj járulék	1560	1560	2400	2400	4000	5000	
3	Adó	3120	2600	4800	4280	7740	9220	
4	Kifizetendő	10920	11440	16800	17320	28260	35780	1208
5								
6	Gyerekszám		2		2	1	3	
7		kicsi	kicsi	normál	normál	nagy	jelentős	nagy
8								

4-12. ábra

A második példa: Egy vállalat vezetősége elhatározta, hogy a fizetéseket megemeli minimálisan 25000 Ft-ra, és szeretné megtudni, hogy ehhez mennyi pénzre van szüksége. Ezt az IF() függvénnyel egyszerűen kiszámolhatja. A B8-as cellába az alábbi összefüggést kell írni:

$$=IF(B1<25000;25000-B1;)$$

Ezt át kell másolni a C8:G8-as tartományba, majd a H8-as cellába egy szumma függvénnyel összeadni a számokat. Azaz, ha a feltétel igaz, vegye a 25000 és a konkrét fizetés közötti különbséget, ha a feltétel nem igaz, nem kell semmit csinálnia. A cellába ilyenkor nulla kerül. A nullák megjelentetését le lehet tiltani az Options/Display... párbeszédpanelben.

A tartomány gyors feltöltése

A cellák másolásának van egy nagyon gyors, vágólap nélküli módszere is. Ha a másolandó cellát a mellette levő cellá(k)ba szeretnénk másolni, akkor kijelöljük azt a sort vagy oszlopot, ahová másolni szeretnénk úgy, hogy annak az első cellája a másolandó cella legyen. Majd kiadjuk az **Edit/Fill Right** parancsot, ha a jobbra lévő cellákba másolunk; a **Fill Down**-t pedig, ha lefelé. Ha az **Edit** menü kiválasztása közben a **SHIFT** billentyűt is nyomjuk, ugyanezek a menüpontok **Fill Left**, illetve **Fill Up** parancsként jelennek meg, és akkor balra vagy felfelé tudunk másolni. (Van még néhány menüpont, amely más tartalommal jelenik meg, ha a **SHIFT**-et is lenyomva tartjuk a menü kiválasztásakor)

Itt érdemes megjegyezni, hogy van még egy ennél is gyorsabb módszer is: Ha előre kijelöljük a tartományt, amelynek minden cellájába ugyan azt a számot, képletet, szöveget stb. szeretnénk írni, az elsőbe beírjuk, és az **ENTER** billentyű egyszerű lenyomása helyett a **CTRL** billentyűvel együtt nyomjuk le az **ENTER**-t. Hatására a kijelölt tartomány összes cellájába bekerül a beírt konstans vagy képlet.

Sorok oszlopok beszúrása, törlése

Gyakran előfordul, hogy egy táblázat készítése során nem tudtuk előre eltervezni, hogy hány sorra vagy oszlopra lesz pontosan szükségünk. Újabb sorokat, illetve oszlopokat bármikor beszúrhatunk a táblázatba utólag is, és persze fordítva is igaz, bármikor törölhetünk. Nem kell feltétlenül egy teljes sort/oszlopot beszúrni, lehet csak néhány cellát. Ilyenkor a program megkérdezi, hogy a helyén lévő cellákat milyen irányba tolja el. A cellák sorszámára kell kattintani az egérrel, az oszlop kijelöléshez pedig az oszlop betűjére.

Szűrjünk be egy sort a dolgozók nevéhez az első sorba (4-13. ábra):

- kattintsunk az első sor sorszámára (az egész sor inverzen jelenik meg)
- Az **Edit/Insert** parancs hatására az egész táblázat lejjebb csúszik, és az első sor helyén egy üres sor jelenik meg.

Ha most megnézzük a cellahivatkozásokat, az új helyeknek megfelelően átródtak. Tehát valóban probléma nélkül tudunk új sorokat beszúrni, a szükséges adminisztrációt a program elvégzi. Azt viszont nem engedi meg, hogyha a táblázat alján vannak kitöltött cellák, akkor a beszúrások miatt "kitoljunk" valamit a táblázatból.

Írjunk be 6 nevet! Az első beírásánál már azt fogjuk tapasztalni, hogy a név nem fér el a cellába. Lehetőség van arra, hogy a nevet két sorba írjuk és magasabb legyen a sor. Ehhez ki kell jelölnünk azt a tartományt, ahol megengedjük a szélesebb sorokat, azaz a szövegek több sorba írását. Példánkban ilyen az A1:H1 tartomány. A **Format/Alignment...** menüben a **Wrap Text** melletti kis kockába kell X-szet tenni. Ilyenkor a program kettévágja a beírt szöveget, ahol éppen betelt a sor. Ha mi akarjuk eldönteni, hogy hol kezdjen új sort, akkor az **ALT** és **Enter** billentyűt együtt kell leütni. Ilyenkor sort vált már a formulasorban is.

Microsoft Excel

File Edit Formula Format Data Options Macro Window Help

Normal

G1 Aranyos Aranka

Sheet1

	A	B	C	D	E	F	G	H
1		Kellemes Kata	Sovány Edömér	Kiváló Pista	Szép Lajos	Csunya Csilla	Aranyos Aranka	
2	Munkabér	15600	15600	24000	24000	40000	50000	28200
3	Nyugdíj járulék	1560	1560	2400	2400	4000	5000	
4	Adó	3120	2600	4800	4280	7740	9220	
5	Kifizetendő	10920	11440	16800	17320	28260	35780	120520
6								
7	Gyerekszám		2		2	1	3	
8		kicsi	kicsi	normál	normál	nagy	jelentős	nagy

4-13. ábra

A táblázatok elmentése, visszahívása

Munkánk során a számítógépünk operatív tárában dolgozunk. A programból történő kilépéskor, a gép kikapcsolásakor vagy egy hirtelen áramszünet esetén a munkánk elvész. Ahhoz, hogy megmaradjon el kell menteni a munkánkat valamelyik háttértárra (lemezre vagy winchesterre) egy állományba ((file). Az állománynak, hogy azonosítani tudjuk, nevet kell adni. Válasszuk ki a **File** menüből a **Save** (mentés) parancsot! A dialógusdoboz alján kiválaszthatjuk azt a meghajtót és a könyvtárat, ahova menteni akarunk. A két keret között középen az éppen aktuális meghajtót és könyvtárat olvashatjuk le. A felső részen az állomány nevét kell megadnunk. A **Save Worksheet as:** mezőben a **SHEET1.XLS** file név szerepel. Ezt kell felülírni egy általunk választott névvel. Az **OK** vagy az **ENTER** billentyű leütése után a rendszer a kijelölt helyre menti a táblázatot a megadott név alatt.

Azt a néhány szabályt, ami az állományok elnevezésére vonatkozik, be kell tartani (lásd a DOS fejezetet). Az állományok nevéhez nem kell a kiterjesztést beírni (az állománynév utáni három karakter), mert az EXCEL automatikusan a táblázati állományok után odaírja az **.XLS**-t.

A fenti műveletet, (**File/Save**), gyakran, kb. 10 percenként érdemes végrehajtani. Ilyenkor már nem kérdezi meg a gép, hogy hova mentsen és milyen név alatt, hanem az előbb beállított paraméterek alapján csak végrehajtja a mentést, eltárolja a változásokat.

Amennyiben másik néven vagy máshova is szeretnénk mentést készíteni, a **File/Save As...** (mentés mint ...) parancsot kell kiválasztani, hogy új paramétereket adjunk meg. A **Save** parancsokban az **Options...** kiválasztása után tudunk más, a szabványos EXCEL formátumtól eltérő állományokat készíteni. (Pl. **dBASE**, **LOTUS** stb.)

Hogyan lehet egy, a háttértáron lévő állományt behívni, munkára előkészíteni? A **File** menüben van az **Open** (megnyitás) parancs, amely segítségével a kiválasztott állomány betölthető az operatív tárba. Először a jobb oldali keretben kiválasztjuk a meghajtót és a könyvtárat, ahol az állományunk van (vagy sejtjük, hogy van), és a bal oldali keretből választhatjuk ki az állományt. A file listában csak az EXCEL által ismert kiterjesztésű állományok jelennek meg alapértelmezésben.

Az utoljára megnyitott négy file neve a **File** menü alján, az **Exit** parancs felett látható. Ha közülük akarunk egyet megnyitni, csupán a sorszámát kell beírni, vagy a listában rákattintani az egérrel.

A táblázat formázása

Az EXCEL a táblázatok formázásánál igen tág lehetőségek kínál a felhasználónak, hogy ízlésének legmegfelelőbb külalakot érhessen el. Ezek közül ismertetünk néhányat a következőkben.

A sorok oszlopok szélessége, illetve magassága

Egy táblázat elkészítésénél jó, ha minden oszlop ideális szélességű, azaz pont olyan széles, mint a benne szereplő leghosszabb adat. Példánkban a *nyugdíjjárulék* szó nem fért ki, és esetleg más oszlopok szélesebbek, mint kellene. Az oszlopok szélességét legegyszerűbben úgy állíthatjuk be, ha az egér nyílát az állítandó oszlop betűjelének jobb széléhez visszük, ilyenkor a nyíl átalakul egy jobbra-balra mutató nyíllá, és az egér gombjának nyomvatartása mellett jobbra-balra mozogva, az oszlop szélesedik, illetve keskenyedik. A formulasor baloldalán ilyenkor le tudjuk olvasni a pillanatnyi szélességet (*Width:....*). Az ideális szélességet, amikor olyan széles az oszlop, mint a benne szereplő leghosszabb adat, úgy tudjuk gyorsan elérni, hogy az egérrel kettőt kattintunk, amikor a kettős végű nyíl látható az oszlopok elválasztó vonalánál. Ha több oszlopot jelölünk ki egyszerre, a beállítás mindegyikre vonatkozik.

A *Format/Column Width* dialógusdobozban billentyűzetről is be tudjuk állítani a szélességet. Itt a *Hide* paranccsal a kijelölt oszlopokat el tudjuk tüntetni a képernyőről, illetve az *Unhide* paranccsal visszahozni. Az eltüntetés nem törlést jelent, tehát a benne szereplő adatok léteznek, hivatkozni lehet rájuk, csak nem jelennek meg a képernyőn. Szükség lehet erre, ha csak részadatokat akarunk látni vagy nyomtatni.

A sorok magasságát ugyanígy lehet állítani egérrel és a *Format/Row Height* paranccsal a dialógusdobozból.

Szám- és dátumforma

Amikor befrunk egy számot egy cellába, nincs módunk arra, hogy meghatározzuk, hány tizedest jelenítsen meg, elválassza-e ezres helyiértékenként a számokat, kerekítsse-e stb. Ezeket a beállításokat a *Format/Number...* dialógusdobozban adhatjuk meg. Mielőtt ezt a parancsot kiadnánk, ki kell jelölni azt a tartományt, amihez a formátumbeállít

tásokat rendelni akarjuk. A formátumokat egy listából választhatjuk ki, illetve a **Format** sorba magunk készíthetünk egyedi formátumokat. Az aktív cella tartalmát az éppen kiválasztott formátumnak megfelelően a **Sample** sor mutatja. A listában szereplő #, 0 és a ? számhelyet jelöl. A 0 azt jelenti, hogyha nincs értékes számjegy, akkor a helyét kiegészíti 0-val. A ? hatására ilyenkor szóközt tesz, a # csak azt jelöli, hogy számot vár a helyére.

Az általános **General** formátum minden számot jobbra igazít a cellába, annyi tizedesjegyet ír ki, amennyi értékes tizedes van, ha túl nagy a szám átalakítja normál alakra (egy 1 és 10 közötti tizedes számmá és tíz előjeles hatványává). A szövegeket balra jeleníti meg, míg a hibaüzeneteket középre igazítva. Az igazításon tudunk változtatni a következő részben leírtak alapján. Amikor egy új munkatáblázatot kezdünk, minden cellához ez a formátum van rendelve.

A szövegigazítás

Egy táblázat külalakjának lényeges része, hogy a cellákban az adatok hol helyezkednek el. A kijelölt tartomány celláiban lévő adatok helyzetét legegyszerűbben a képernyő tetején lévő segédsor közepén lévő piktogramokkal tudjuk megváltoztatni. Külön jel szolgál a balra-, a középre-, illetve a jobbraigazításhoz. Ezt a **Format/Alignment..** parancs hatására megjelenő dialógusdobozban is be tudjuk állítani, illetve a **Fill** parancs a cella tartalmát megismétli mindig pont annyiszor, hogy kitöltse a cella pillanatnyi szélességét.

A cellák, tartományok vonalazása, keretezése

A táblázatok nyomtatásánál megadhatjuk, hogy a cellák körül minden rácsot kinyomtasson. Ezen felül vagy e helyett minden egyes cellát, tartományt külön-külön megvonalazhatunk. A **Format/Border** parancs segítségével a dialógusdobozból választhatjuk ki, hogy milyen vonalat szeretnénk (**Style**), milyen színnel (**Color**), és hova rajzolni (**Outline** - körbe, **Left** - balra, **Top** - felülre, **Bottom** - alulra).

A cellák, tartományok színezése, satiozása, sraffozása

A cellák, tartományok további kiemelési lehetősége a sraffozás, illetve a színezés. A **Format/Patterns...** menüből választhatjuk ki a mintát az előtér és a háttér színét. A **Sample** mezőben rögtön ellenőrizhetjük a beállítás eredményét.

A számok, szövegek betűinek beállítása

Igen szép hatást tudunk elérni, ha a kijelölt tartományokban a betűk méretét, típusát és az attribútumokat beállítjuk. A **Format/Font...** dialógusdobozban választhatunk a felsoroltak közül. A **Bold** (vastag) és az **Italic** (dőlt) attribútumokat a segédsorból a megfelelő piktogram kiválasztásával közvetlenül is hozzárendelhetjük a kijelölt tartományhoz. Arra sajnos nincs mód, hogy egy cellán belül egy-egy betűhöz, szóhoz más és más attribútumot rendeljünk.

A nyomtatás

Egy táblázat elkészítése sokszor nem ér véget azzal, hogy a képernyőn elkészítettünk egy szép és jól felépített táblázatot. A legtöbbször szükség van arra is, hogy kinyomtassuk a táblázatainkat.

Ha az egész táblázatot, azaz minden cellát, amibe beírtunk valamit ki akarjuk nyomtatni, akkor nem kell megadni a nyomtatandó területet. Egyébként szükség van rá. A nyomtatni kívánt területet kijelöljük, és amikor inverzen látszik az egész terület, az **Options/Set Print Area** parancsot kell kiadni. Ha a nyomtatandó terület nagyobb, mint amekkora táblázat elfér egy lapon, a EXCEL automatikusan felosztja lapokra, ahogy a 4-14. ábra mutatja:

1	4	7
2	5	8
3	6	9

4-14. ábra

Ha mi szeretnénk meghatározni, hogy hol kezdjen új lapot, akkor a cellakurzorral arra a helyre kell állnunk, ahol új oldalt kezdenénk, és az **Options/Set Page Break** parancsot kell kiadni. (Ha mégsincs szükségünk az adott helyen a lapdobásra, vissza kell állni a kurzorral oda, és ugyanezt a parancsot kiadni, ami most a következőképpen néz ki: **Options/Remove Page Break**.)

Ezek után már meg is tekinthetjük a nyomtatási képet a képernyőn a **File/Print Preview** parancs segítségével. A képernyő alján le tudjuk olvasni, hogy összesen hány lapra fért ki a táblázatunk, és ebből éppen hányadik oldalt látjuk. Az oldalak között a következő (**Next**) és az előző (**Previous**) gombokkal tudunk lapozni. A **Zoom** gombbal kinagyíthatjuk, illetve összeszűkíthetjük a képet. (A képernyőn kicsinyített megjelentetés esetén egy nagyító a kurzor, ahova rákattintunk, annak a részletnek a nagyított képét kapjuk.) A **Margins** hatására megjelennek a képernyőn a margóvonalak és az oszlopok elválasztóvonalai. Ha ezekre a vonalakra az egérrel ráállunk és mozgatjuk, változnak az oszlopszélességek és a margók. A **Print**-tel indítható a nyomtatás, míg a **Setup...** segítségével az oldallal kapcsolatos beállítások dialógusdoboza jelenik meg. E két utóbbi parancs a **File** menüből is kiadható.

A nyomtató átállítására, illetve a paraméterek beállítására (File/Printer Setup...) általában nincs szükség, az EXCEL elindításakor a Windows Controll Panel Printer beállításait átveszi.

A Page Setup... dialógusdobozban a nyomtatás külalakjával kapcsolatos beállításokat tudjuk megadni. Megadható a fejléc (Header), illetve a lábléc (Footer), amely minden oldal tetejére, illetve aljára kinyomtatódik. Tetszőleges szövegeket és speciális karaktereket lehet itt megadni (például az oldalszám: &P, a dátum: &d, az idő:&t). Itt adható meg a papír mérete (Paper) és a margók szélessége (Left, Right, Top, Bottom) is. Ha a Center Horizontally és Center Vertically keretbe X-et teszünk, akkor a táblázatot az oldalon vízszintesen és függőlegesen középre helyezi a program. A Row & Column Headings paranccsal elérhetjük, hogy az oszlop- és sorazonosítókat is kinyomtassa, valamint Gridlines-szal azt, hogy a cella határoló vonalait is nyomtassa. A Portrait-tal függőlegesen, a Landscape-pel pedig fekvő lapformátumot tudunk választani.

A Print parancs elindítása után egy újabb dialógusdoboz jelenik meg, ahol egyrészt megadhatjuk, hogy hány másolatot szeretnénk nyomtatni (Copy), minden oldalt (All) vagy csak néhányat nyomtatunk. Ilyenkor meg kell adni, hogy hanyadiktól (From) hanyadikig (To).

Egyéb szolgáltatások

A következőkben ismertetendő lehetőségeket csak kiragadtuk a teljes tárházból. Ha csak azt vesszük, hogy az EXCEL programhoz több ezer oldalas dokumentáció tartozik, érthető, miért kell állandóan szabadkozunk a terjedelmi korlátaink miatt.

Az adatok rendezése

A program képes sorokat és oszlopokat egyaránt rendezni csökkenő és növekvő sorrendben. Megadhatjuk, hogyha a rendezendő adat alapján talál azonosakat, akkor azon belül melyik adat szerint rendezzen. Ezt három szinten adhatjuk meg. (Például, ha városok szerint rendezünk elsődleges szempontként, akkor az azonosokon belül rendezzen a kerületek szerint, és azon belül az utcák abc rendje szerint.)

A rendezési parancs kiadása előtt nagyon pontosan ki kell jelölni azt a tartományt, amit rendezni akarunk. A mi feladatunknál ez a B1:G8-as

tartomány. (4-15. ábra) Nagyon fontos, hogy a kijelölésben minden sor, illetve oszlop benne legyen, ami összetartozik. Ha ugyanis a példánkban a nevek szerint akarunk rendezni, és csak a B1:G5 tartományt jelölnék ki, akkor a gyerekek száma nem menne a rendezés során az eredeti névvel, és máshoz kerülnének tévesen.

A **Data/Sort...** paranccsal jeleníthetjük meg a rendezés paramétereinek beállításához szükséges dialógusdobozt. Először ki kell választani, hogy sorokat (**Rows**) vagy oszlopokat (**Columns**) rendezünk. Az alapértelmezés a sorok szerinti rendezés, ha elfelejtjük átállítani sokszor az **Edit/Undo** parancs sem segít az okozott összevisszaság helyreállításában. Állítsuk be a **Sort by** keretben a **Rows** paramétert! Utána meg kell adni a rendezési kulcsot: annak a sornak bármely celláját, ami szerint rendezni akarunk. Jelen esetben a nevek szerint, ami az első sorban van, tehát az első sor bármely celláját beírhatjuk (pl. A1). Az **Ascending** növekvő a **Descending** csökkenő rendezést jelent. Az eredmény a 4-15. ábrán látható.

Microsoft Excel

File Edit Formula Format Data Options Macro Window Help

Normal

G8 =IF(G2<18000;"kicsi";IF(G2<25000;"normál";IF(G2<45000;"nagy";"jelentős")))

Sheet1

	A	B	C	D	E	F	G	H
		Arányos Aranka	Csunya Csilla	Kellemes Kata	Fiválló Fista	Sovány Edömér	Szep Lajos	
1								
2	Munkabér	50000	40000	15600	24000	15600	24000	28200
3	Nyugdíjjárulék	5000	4000	1560	2400	1560	2400	
4	Adó	9220	7740	3120	4600	2600	4260	
5	Kifizetendő	35780	28260	10920	16800	11440	17320	120520
6								
7	Gyerekszám	3	1				2	2
8		jelentős	nagy	kicsi	normál	kicsi	normál	nagy
9								

4-15. ábra

A visszafelé számolás (az eredményből a kiinduló adatok kiszámítása)

Ez a szolgáltatás lehetővé teszi számunkra, hogy az eredményt adjuk meg, és keressük azt a kiinduló adatot, amiből a kívánt eredményt lehet kapni egy minta alapján. Példánkban megadjuk, hogy mennyi nettó bért szeretnénk, és ehhez mennyi bruttó bér kell. (Ez akkor lenne különösen izgalmas feladat, ha a tényleges adókulccsal számolnánk már az adó sorában. Ehhez azonban meg kellett volna tanulnunk egy igen bonyolult függvényt, a HLOOKUP()-ot.)

Rá kell állni arra a cellára, ahova az új értéket akarjuk megadni, például a B5-re. A Formula/Goal Seek... parancsot kiválaszjuk. A Set cell keretben megjelent a B5-ös cella címe, mert a kurzort előtte ide állítottuk. Ennek a cellának fogunk értéket adni, a To value keretben. Írjuk be, hogy 30000 forintot szeretnénk nettó jövedelemnek! A By changing cell keretbe pedig meg kell adni annak a cellának a címét, aminek az értékén változtathat. A B2-es cella ez a cella, ahol a bér van. Ilyenkor is használhatjuk a rámutatós módszert. Ha a dialógusdoboz útban van, a címsorára állva az egér nyílával, lenyomott gomb mellett arrébb lehet tolni. Az OK után egy státuszképernyő jelenik meg, amiben jelzi, hogy talált-e megoldást a visszafelé számolás során. Ha szeretnénk az új értékkel helyettesíteni a régit, akkor OK-t választunk, ha nem, akkor Cancel-t. Igenlő válasz után a B2-es cellában le tudjuk olvasni, hogy 30000 Ft-os nettó bérhez 42485.71 Ft bruttó bér tartozik.

Az adatok keresése, helyettesítése

Szükségünk lehet rá, hogy egy nagy táblázatban vagy annak kijelölt részében megkeressünk bizonyos tartalmú cellákat. Keresni a Formula/Find paranccsal lehet. Megadható, hogy a keresendő karaktersorozat önálló cellatartalom (Whole) vagy egy cellatartalom része (Part). Elő lehet írni, hogy milyen sorrendben keressen a program, sor- vagy oszlop-folytonosan, illetve hogy képletekben vagy csak számok között keressen. Természetesen be kell gépelni a Find What keretbe, a keresendő karaktersorozatot.

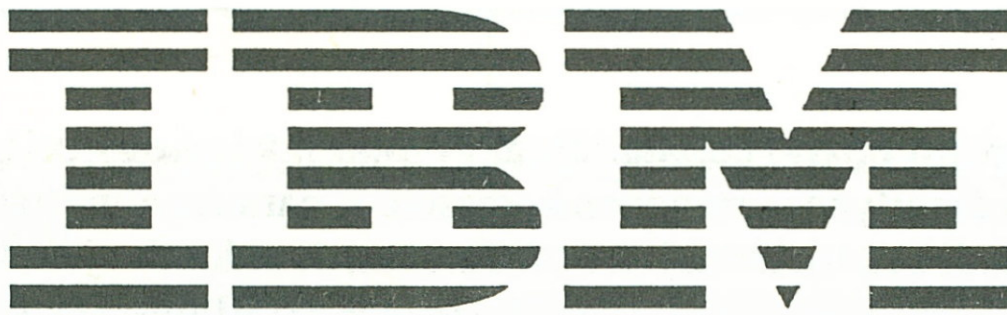
Ha nem csak megkeresni akarunk egy karaktersorozatot, hanem ki is akarjuk valamire cserélni, akkor a Formula/Replace... parancsot kell választanunk. Be kell írunk, hogy mit akarunk cserélni (Find What), mire akarjuk cserélni (Replace With), megadhatjuk, hogy a keresendő

karaktársorozat önálló cellatartalom (**Whole**), vagy egy cellatartalom része (**Part**), és milyen sorrendben keressen. A parancs elindítása után (**Find Next**) celláról cellára láthatjuk a találatokat, és eldönthetjük, hogy az adott találatot kicserélje-e (**Replace**), átugorja-e és álljon-e rá a következőre (**Find Next**), ne kérdezzen tovább, minden találatot cseréljen ki (**Replace All**) vagy fejezze be a cserélést (**Cancel**).

Az ablakkezelés

Az EXCEL szolgáltatásai közé tartozik, a Windows környezetből adódóan, hogy a táblázatokat egy-egy különálló ablakban jeleníti meg. Az egyszerre nyitvatartható (aktív) ablakok számának csak a számítógépünk memóriája szab határt. Az egyes ablakokból az adatok átvitele (másolása, mozgatása), vagy a másik ablakban lévő adatokra történő hivatkozás nagyon egyszerű. Másolásnál például az egyik ablakból rámásoljuk a vágólapra tartományt, ablakot váltunk és beragasztjuk a másikba. Ha mind a két ablak látszik a képernyőn, csak rá kell kattintani bármelyik pontjára, ha nem látszik a célablak, a Windows menü alján találjuk az összes aktív ablak nevét. Vagy a sorszámát ütjük le, vagy rákattintunk a megjeleníteni kívánt ablak nevére. Az ablakkezelés technikája nem tér el a Windows részben tanultakétól.

Egy másik ablak (állomány) cellájára történő hivatkozásnál a cella címe előtt az állomány nevét is meg kell adni: ADO.XLS!A1:K3. Ezt "mutogatásnál" a program beírja.



AZ IBM VÁRJA ÖNT

A TERMÉKEK ÉS A SZOLGÁLTATÁSOK TELJES KÖRÉVEL:

- nagyszámítógépek, minigépek, személyi számítógépek,
- hálózati rendszerek, munkaállomások,
- perifériák, kiegészítők,
- operációs rendszer,
- felhasználói és kommunikációs szoftverek,
- oktatás,
- kábelezés,
- HW- és SW-karbantartás,
- IBM nemzetközi információs hálózat,
- kulcsrakész számítógépterem,
- kulcsrakész számítógépes rendszerek,
- kellékeladás.

KERESSEN BENNÜNKET!

IBM Magyarország
1118 Budapest, Ménesi út 22.
Postacím: 1502 Budapest, Pf. 171.

Tel.: 165-4422
Tx.: 22-4728
Fax: 186-9265



SZTÁV
Átképzést
Támogató Kft
Budapest-Emden

Információ



A SZTÁV-ÁT Kft.

számítástechnikai szaktanfolyamai
új lehetőséget kínálnak az Ön számára!

Személyi számítógép kezelő

- 180 órás alapfokú szaktanfolyam

intenzív formában: 5 óra/nap
2 hónap
tanfolyami díj: 27.000 Ft

Személyi számítógép szoftverüzemeltető

- 400 órás középfokú szaktanfolyam

intenzív formában: 5 óra/nap
4 hónap
tanfolyami díj: 54.000 Ft

Számítógép programozó

- 500 órás középfokú szaktanfolyam

intenzív formában: 5 óra/nap
5 hónap
tanfolyami díj: 67.500 Ft

Részt vehet: alapfokon mindenki
 középfokon, aki középiskolai érettségivel rendelkezik

Képesítések: alapfokú személyiszámítógép kezelő - KSH országos érvényű
 személyi számítógép szoftverüzemeltető - KSH országos érvényű
 középfokú számítógép programozó - KSH országos érvényű

Helyszínek: Budapest és a megyei városok

A tanfolyamokra kérjük írásban jelentkezzen az alábbi címen:

SZTÁV-ÁT Kft

1108 Budapest Gyömrői út 140.
Telefon: (06-1) 127-3499



SZTÁV-PROGRESS

Oktatási, Továbbképző és Szolgáltató Kft.

Minden egyes tanfolyamunk növeli az esélyt az elhelyezkedésre!

SZÁMÍTÁSTECHNIKAI KÉPZÉS

SZAKKÉPESÍTÉST ADÓ TANFOLYAMOK	Időtartam (óra)
1. Személyi számítógép-kezelő (alapfokú)	150
2. Személyi számítógép szoftver-üzemeltető (középfokú)	350
3. Személyi számítógép-kezelő (operátor)	400
4. Személyi számítógép programozó (középfokú)	450
5. Személyi számítógép műszaki karbantartó	400

ALAPOZÓ ÉS TOVÁBBKÉPZŐ TANFOLYAMOK

1. Számítógép-kezelő	40
2. PC DOS + dBASE adatbázis-kezelő	40
3. Clipper adatbázis-kezelő	40
4. (Turbo)PASCAL programnyelv	40
5. VENTURA kiadványszerkesztő	80
6. dBASE IV. adatbázis-kezelő	40/80
7. Szövegszerkesztő (Word 5.0, WordPerfect, ChiWriter)	40
8. Novell Net Ware	40

A képzés heti kétszer 5 órában délelőtt vagy délután, illetve intenzív formában (heti 30 óra) történik. Minden résztvevő külön IBM PC-n dolgozhat! A szükséges alaptankönyvek és oktatási segédanyagok árát a részvételi díj tartalmazza. Részletfizetési kedvezményt adunk!

ÜGYFÉLSZOLGÁLATI IRODÁINK:

3525 Miskolc, Fábrián kapu 11.

3532 Miskolc, Andrássy u. 36.

Postacímünk: 3545 Miskolc, pf. 566

(46) 348-659



(46) 347-618

5. Adatbáziskezelés számítógéppel

A számítástechnika és az adatok nyilvántartása

Az adatbáziskezelés az egyik legrégebbi programeszköz a számítástechnikában. Elsőként fejlesztették ki gazdasági és nem tudományos célokra, arra a feladatra, hogy óriási adattömegeket kezeljen. Adatok ezreinek, tízezreinek kezelése alapvető fontosságú lett az információrobbanás évtizedeiben. Hogyan is igazodnánk el ennyi adat között, ha nem hívnánk segítségül a számítógépeket!

Bármilyen nagyságrendű adatállomány kezelésére bonyolult nyilvántartó programot írtak. Például külön program működtette a dolgozók bér- és munkaidő nyilvántartását. A felhasználóknak egyenként meg kellett tanulni e független szoftverek működését. S ahány fejlesztés annyi program, annyi költség, annyi hibalehetőség s annyi ismétlődő adat. Az adatkezelési tapasztalatokat, trükköket összegyűjtve fejlesztették ki az ún. adatbáziskezelési technikát. Megjelentek az **adatbáziskezelő programrendszerek**.

A programok tipikus felhasználói igényt elégítenek ki, ezért alkalmazói keretprogramok közé soroljuk őket. A szoftver alkalmazóinak most a sokféle program helyett csak egyet, az adatbáziskezelő utasításait kellett megismerni. Ennek az utasításrendszernek a használata leginkább a DOS rendszerprogram kezeléséhez hasonlítható. Begépelünk egy utasítást, s ezt a szoftver azonnal értelmezi és végrehajtja. Ez a működési mód az adatbáziskezelő program használatának egyik módszere.

Majd a személyi számítógépek nagymértékű elterjedésével megjelentek a korszerű menüvezérléssel működő adatbáziskezelők.

Az adatbáziskezelők általános ismertetése

Ha szeretnénk nyilvántartani könyveinket, lemezeinket, netalán receptjeinket, akkor előveszünk egy nagy spirálos füzetet, húzunk néhány függőleges oszlopot, s soronként beírjuk legfontosabb adataikat, például a szerzőjüket, címüket, az étel elkészítési idejét stb. Aztán ha keressük valamelyiket, a sorokat végigböngésszük. Ha nagyon precízek és előrelátóak vagyunk, betűrendben, vagy téma szerint is regisztrálhatjuk az adatokat, ekkor talán könnyebb a visszakeresésük. De mi van akkor, ha betelik a "B" kezdőbetűs lap, vagy a leveseknek szánt oldal! Valahová beszúrjuk az újakat, s reménykedünk, hogy nem felejtjük el hova.

Vegyünk munkahelyi példákat is! Nyilvántartjuk vállalatunk, kft-énk dolgozóit, munkatársait, bedolgozóit stb. Egyszer belépési időrendben, egyszer szoros betűrendben, egyszer fizetési nagyság sorrendjében vagy nyelvtudás szerint szeretnénk sorolni őket. Rengeteg idő és fáradság. Ezek elkerülésében nyújt segítséget az adatbáziskezelő programrendszer, amelyet tetszőleges adatmennyiséget, bármilyen szempont szerint gyorsan és hatékonyan kezelhetünk.

Az adatbáziskezelők szolgáltatásai

Az adatbáziskezelőnek néhány alapvető szolgáltatást tudnia kell.

Ismernie kell:

- a **DOS szolgáltatásokat** (adatállomány másolása, törlése, átnevezése stb.);
- az **adatok feldolgozását** (az új adatok bevitele, a már rögzített adatok módosítása és törlése);
- az **adatok megjelenítését** a képernyőn, a nyomtatón;
- a legkülönbözőbb feltételeknek megfelelő **adatvizakeresést**;
- az **adatok rendezését**.

Az adatbázis fogalma és szerkezete

Mit jelent az adatbázis, adatbáziskezelés kifejezés?

Ha valakinek bemutatunk egy számára ismeretlen személyt akkor leírjuk annak tulajdonságait, alkati, szellemi jellemzőit. Ezt tesszük akkor is, ha nyilvántartást készítünk.

A személyi számítógépeken elterjedt adatbáziskezelő rendszerek az ún. **relációs adatmodellen** alapulnak. Egyszerűen megfogalmazva ez annyit jelent, hogy a feladat számára fontos adatokat egy táblázatba helyezzük el. Az egyes tulajdonságokat oszlopokra bontjuk, s a fejlécbe beírjuk a tulajdonság nevét. A nyilvántartásra kerülő személyeket, tárgyakat viszont az egyes sorokba helyezzük el, ahány személy, annyi sor, akár több százazres nagyságrendig növelve. Az adatbáziskezelés terminológiájában a sorokat **rekordnak**, az oszlopokat **mezőknek** hívjuk. A sorok és oszlopok keresztmetszetében található értéket **adattételnek**, valamennyi adatot összességében **adatállománynak** nevezzük. Sokszor azonban egy adatállomány elegendő a rendszer működéséhez.

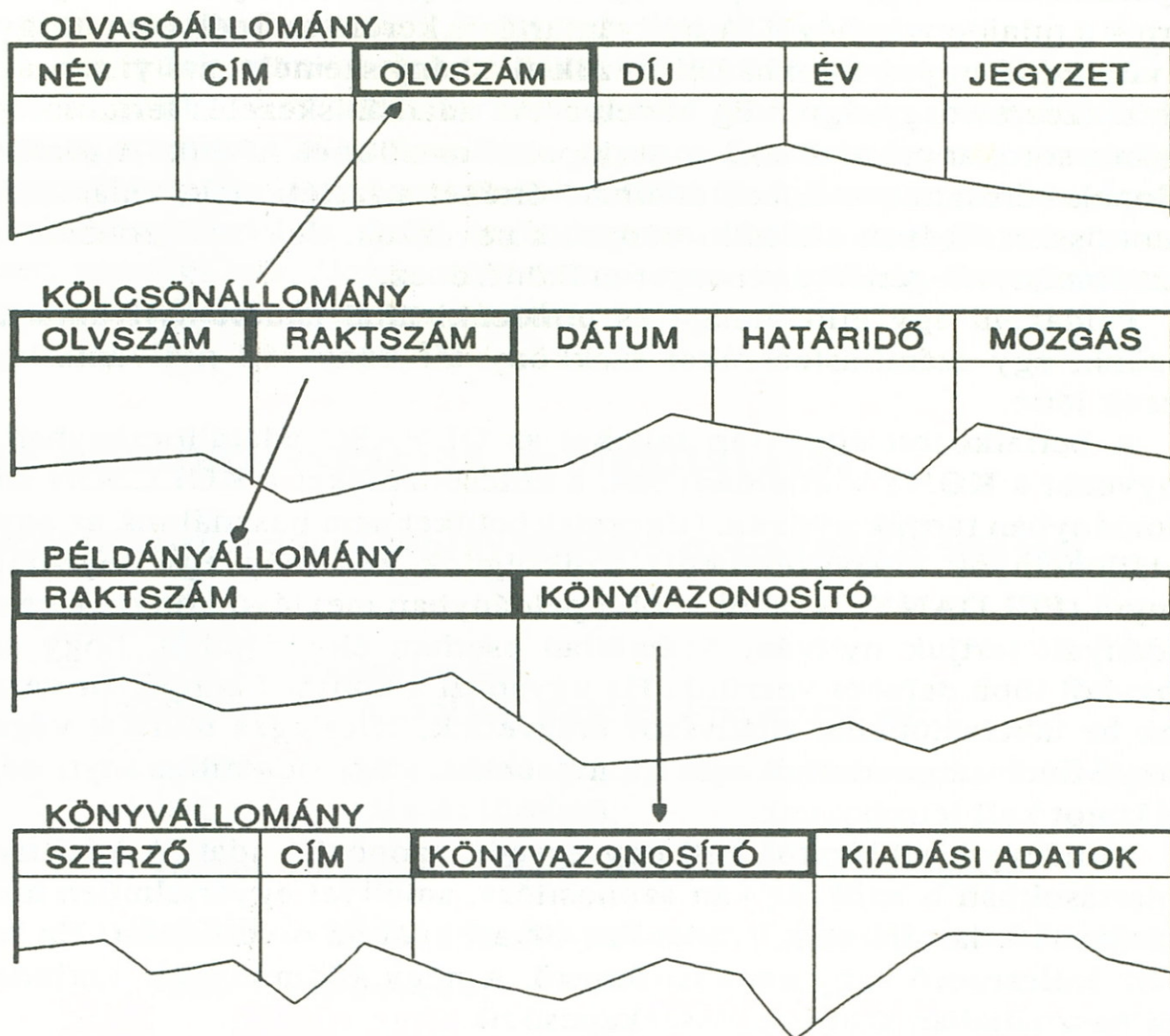
Példaként egy mindennapi és mindenki által ismert nyilvántartást veszünk, egy számítástechnikai szakkönyvtár könyvtári nyilvántartását hozzuk létre.

A beiratkozott könyvtári tagokat az OLVASO adatállományban, a könyveket a KONYV állományban, a kölcsönzéseket a KOLCSON adatállományban tartjuk nyilván. (Ékezetes betűket nem használunk az egyes adatállományok elnevezésében!) Szükségünk van még egy segédállományra (PELDANY) is, ahol a több példányban meglévő könyvek egyes példányait tartjuk nyilván. Számtalan esetben előfordulhat, hogy egy könyvből több darabot veszünk. Ha ugyanazt a könyvet annyiszor vezetnénk be táblázatunkba, ahányszor megvettük, felesleges munkát végeznénk. Miután átgondoltuk nyilvántartásunkat, négy adatállományt, négy táblázatot kell létrehozunk.

Mindegyik táblázatba kell egy egyedi, azonosító adat. A kézi nyilvántartásokban is szükség van azonosítóra, amellyel egyértelműen meghatározzuk rekordjainkat, például az olvasóknál az olvasószám. Ez lesz az ún. **kulcsmező** vagy **azonosítómező**. A négy állományunk a mindennapi használatban szorosan összekapcsódik.

Például bejön az olvasó, kezében két könyvvel, kérjük a beiratkozás-kor kapott olvasószámát. A könyvtáros begépelí a képernyőre ezt a számot, majd megjelennek az olvasó adatai, ehhez viszont szükségünk van az OLVASO állományban tárolt adatokra (az olvasó címe, neve stb.)

Majd szükségesek a kikölcsönzött könyvek adatai is. Így a KOLCSON állományban nemcsak az olvasószámot, hanem a könyvek azonosítószámát, a raktári számot is nyilván kell tartanunk. A raktári számhoz tartozó könyv szerzőjét, címét és egyéb adatait a KONYV állományból hívjuk elő. (A PELDANY állomány közvetítő szerepet játszik, mivel itt tartjuk nyilván, hogy egy könyvféleségből hány darab létezik nyilvántartásunkban.) Tehát élő kapcsolat van az egyes táblázatok között (5-1. ábra)! Ez a kapcsolat logikai kapcsolat a fizikailag létező adathalmazok között.



5-1. ábra

Az egymással logikai kapcsolatban lévő adatállomány-együtteseket **adatbázisoknak** nevezzük. Azokat a szoftvereket, melyek ilyen állományok feldolgozásával, kezelésével foglalkoznak **adatbáziskezelő programoknak** nevezzük. Egy konkrét program, a **dBASE IV** segítségével ismertetjük az adatbáziskezelés legáltalánosabb és legfontosabb tulajdonságait!

A dBASE típusú adatbáziskezelő rendszerek

A személyi számítógépen működő adatbáziskezelők közötti konkurrenciaharcot az *ASHTON-TATE* és a *FOX* szoftverházak "összecsapásai" jellemzik. Az *ASHTON-TATE* cég a **dBASE** családot indította el. Első verzióját (**dBASE II**) még a 8 bites mikroszámítógépre készítette el, majd megjelentek az újabb változatok a **dBASE III**, **dBASE III PLUS**, **dBASE IV** formájában.

A **FoxBASE+**, valamint az újabb verziójú **FoxPRO** a **dBASE** család hatásos konkurrense mind gyorsaságban, mind szolgáltatásaiban. E két adatbáziskezelő rendszer mellett meg kell említenünk a **dBASE** típusú adatbáziskezelői szolgáltatásokkal felszerelt programnyelvet is, a **CLIPPERT**. Kezdő felhasználók részére azonban még programnyelv használatát nem ajánljuk.

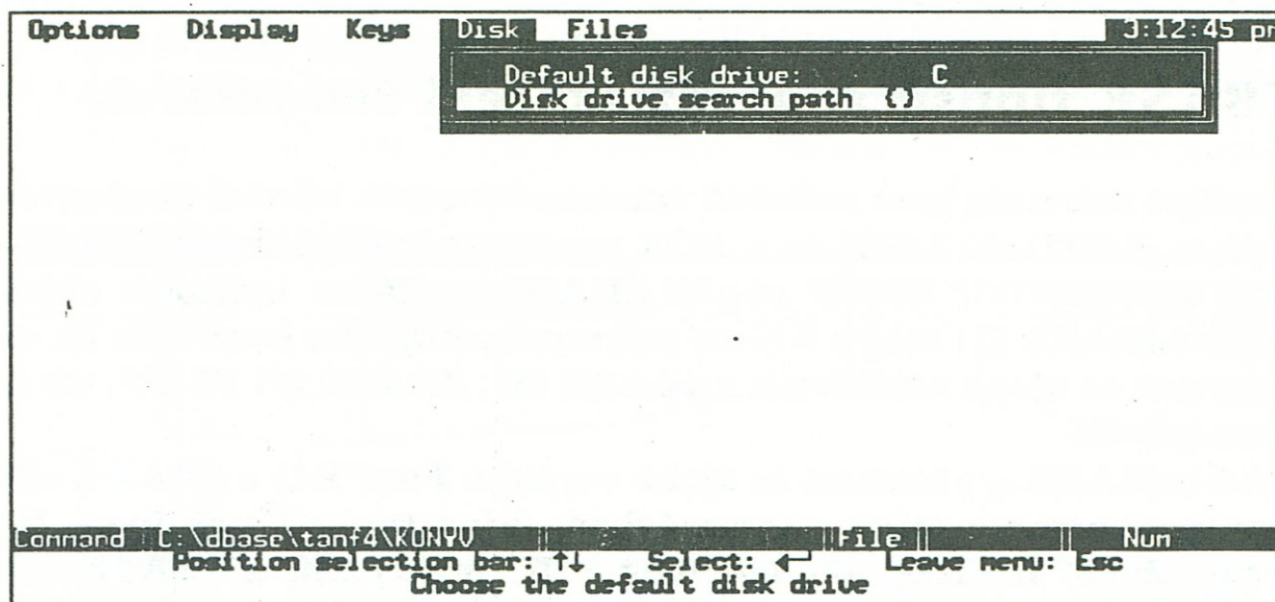
Indítsuk el a programot **dBASE** szó begépelésével és az **ENTER** billentyű leütésével!

A munkaképernyő szerkezete

Indítás után az 5-2. ábrán szereplő képernyőt látjuk, mely négy részből áll. A képernyő sorait 0-24-ig, az oszlopait 0-79-ig sorszámozzuk.

A 0. sorban a menüvezérelt parancsoknál a *menüsor* található, jobb oldalon az *órákijelzés* látszik. Az 1-20 sor az *információs rész*, ahol parancseredmények, képernyők íródnak ki. A 21. sor a *parancssor*, elején a *pontjel*, a **dBASE IV** készletléti jele látható. Ide gépeljük utasításainkat, melyek **ENTER** billentyű leütésével azonnal végrehajthatók. Miután elolvastuk a választ és értelmeztük az eredményt, újból kiadhatjuk a következő parancsot. A 22. sorban a *státuszsor* található, erre a sorra működés közben feltétlenül figyeljünk, mert lényeges információkat tartalmaz. A sorban balról jobbra haladva leolvasható az aktuális képernyő típusa, a feldolgozandó állomány neve, a kurzor pozíciója, a forrásállomány neve,

a kapcsolóbillentyűk (CapsLock, NumLock, Ins) állapotkijelzései. A 23. sor, a *navigációs sor*, az aktuális képernyőn éppen használható legfontosabb billentyűk és funkcióik felsorolására. A 24. sor az *üzenetsor*, a kiválasztott menüre, ill. beviteli mezőre vonatkozó információk megjelenítésére szolgál.



5-2. ábra

A dBASE IV parancsnyelve egy ún. magasszintű nyelv, mely szigorú szintaktikai szabályokat követ. Egy parancs egy sor, a sor hossza maximum 1024 karakternyi lehet, s a sor a képernyőn balra gördülve ad helyet az újabb karaktereknek.

Minden utasítás parancsszóval kezdődik, majd ezt paraméterek, opciók követik. A parancsszavak, függvények 4 karakterre rövidíthetők, míg a dBASE-ben adott paraméterek, állomány- és mezőnevek nem. A kiadott és a kiadásra várakozó utasításokat a *szerkesztőbillentyűk* segítségével szerkeszthetjük. Az adatbáziskezelő lehetőséget ad a legutolsó 20 parancs előhívására, szerkesztésére.

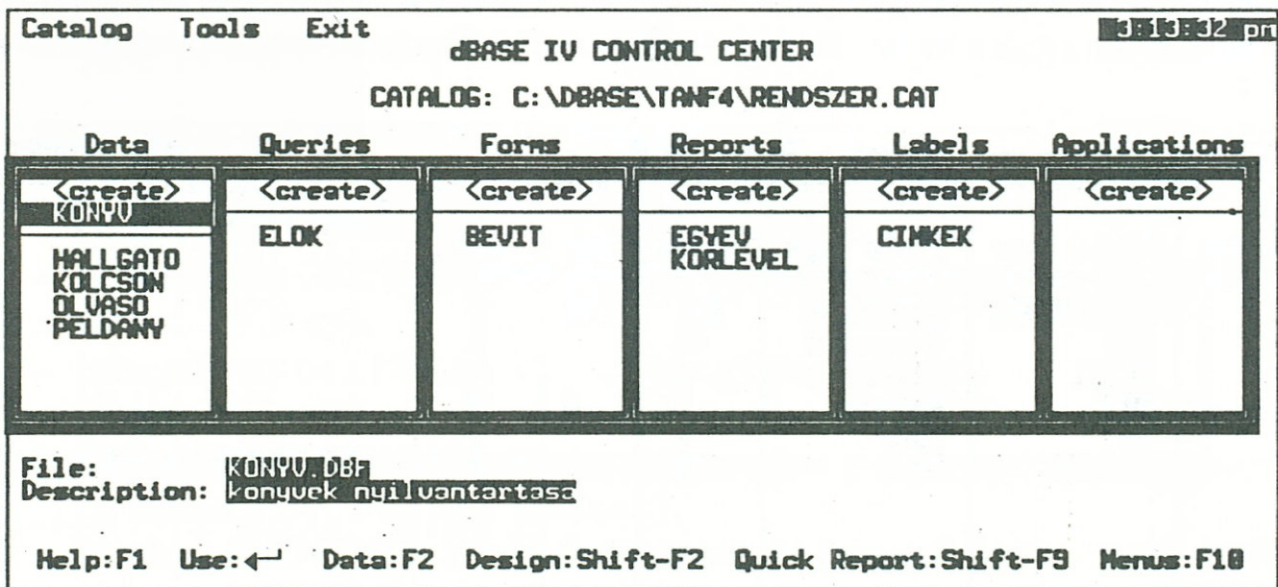
A program a többszintű *menürendszer* egyes *elemeinek* (opciók) kiválasztásával is működtethető. A menürendszer vízszintes sormenük és függőleges rolómenük alkotják. A menüsört F10 funkcióbillentyűvel aktivizáljuk.

A sormenü egyes menüpontjai ALT és az opciók kezdőbetűinek együttes lenyomására, a rolómenüpontok az egyes kezdőbetűk lenyomására aktivizálódnak. Ezeket úgyis kiválaszthatjuk, ha a nyílbillentyűkkel a megfelelő menüopcióra állítjuk a *sávkurzort*, majd lenyomjuk az

ENTER billentyűt. Az egyes szintekről az ESC billentyűvel lépünk ki. A legutolsó szintű menüből kilépve, angol nyelvű üzenet kérdez rá a parancs végleges megszakítására.

Valamennyi adatbáziskezelő szolgáltatás elérhető a vezérlőközponton (CONTROL CENTER) keresztül, az ASSIST parancs kiadásával.

A 5-3. ábrán a legfelső sorban látjuk a *menüsor* elemeit, a sor az F10 billentyűvel aktivizálható. A képernyő *információs területén* találjuk a központ *paneljeit*.



5-3. ábra

Sorrendben a következő *panelek* láthatóak:

- adatbázis-adatállományok oszlopa (DATA)
- adatvisszakeresések, válogatások oszlopa (QUERIES)
- adatbeviteli formátumok oszlopa (FORMS)
- jelentések oszlopa (REPORTS)
- címkék oszlopa (LABELS)
- felhasználói programok oszlopa (APPLICATIONS)

Az állomány szerkezete

A példának vett könyvtári rendszerünkben nyilvántartjuk az olvasókat, a könyveket, az egyes kölcsönzéseket. Létre kell hozni valamennyi táblázatot.

Az adatbázis-adatállományok szerkezetét létrehozzuk a **CREATE** <állománynév>, módosítjuk a **MODIFY STRUCTURE**, kiiratjuk a **DISPLAY STRUCTURE** parancsokkal. Az állománynév max. 8 karakteres, grafikus jeleket nem tartalmazó név lehet.

A 5-4. ábrán megjelenő adatleíró táblázat egy-egy sora egy-egy nyilvántartott adatot ír le. Balról jobbra haladva tekintjük át a táblázatot:

Layout Organize Append Go To Exit						3:16:16 pm
Num	Field Name	Field Type	Width	Dec	Index	Bytes remaining: 3905
1	NEV	Character	36		U	
2	CIM	Character	48		N	
3	OLVSZ	Character	4		N	
4	DIJ	Numeric	3	0	N	
5	EV	Date	8		N	
6	JEGYZET	Memo	10		N	

Database: C:\dbase\tanf\OLVASU field 1/6 Num

Enter the field name. Insert/Delete field:Ctrl-N/Ctrl-U

Field names begin with a letter and may contain letters, digits and underscores

5-4. ábra

Először a **dBASE IV** által automatikusan adott *mezőszámot*, majd az általunk kitalált *mezőnevet*, a *mező típusát*, a *mező szélességét*, a *rendezési igényt* adjuk meg. Tévesztés esetén a táblázat egyszerűen módosítható a szerkesztőbillentyűk segítségével. A **DEL** billentyűvel a kurzor utáni, a **BACKSPACE** billentyűvel a kurzor előtti karaktert törölhetjük. Az **INS** billentyű leütésével a beszúró és a felülíró mód között választunk. Az

oszlopok között a **TAB**, **SHIFT+TAB**, a kitöltött sorok között a nyílbillentyűkkel haladunk. A felesleges sorokat a **CTRL+U** billentyűkombinációval töröljük. Befejezéskor a **CTRL+END** billentyűkombinációval, vagy a menüből kiválasztott **EXIT** opcióval lépünk ki.

Egy adatállomány lefrásnál legfeljebb 255 mező max. 4000 byte összesített mezőhosszal adható meg. A mezőszámot (**NUMBER**) az adatbáziskezelő automatikusan adja. A mezőnév (**NAME**) max. 10 karakteres, betűvel kezdődő, betűket, számjegyeket, aláhúzás-karaktert tartalmazó egyedi, nem ismételheto elnevezés. A mezőtípus (**TYPE**) meghatározza, hogy milyen jellegű adatokat (a megfelelő szököz billentyű leütésével választhatjuk ki) tartalmazhat a mező.

A **dBASE IV** hatféle típust ismer:

- karakteres (**CHARACTER**): valamennyi ASCII karakter (max. 254 karakter),
- numerikus (**NUMERIC**): BCD típusú számjegyek, előjelek (max. 20 jegy),
- lebegőpontos (**FLOAT**): számjegyek, előjelek (max. 20 jegy),
- dátum (**DATE**): ellenőrzött dátumok (8 hosszú év, hó, nap értékek),
- logikai (**LOGIC**): kétértékű adatok (1 hosszú, igen - nem értékek),
- memo (**MEMO**): ASCII karaktersorozatok (10 hosszú)

Az első öt mezőtípus hossza rögzített, csak a memo mező tartalmazhat változó hosszúságú adatokat. A memo típusú adatok nem a 10 karakternyi mezőben tárolódnak. Itt csak egy **.DBT** kiterjesztésű állományra utaló információ van.

A rögzített mezőhosszúságokat (**WIDTH**) a táblázat negyedik oszlopában adjuk meg. A tizedesjegy oszlop (**DECIMALS**) csak a számjegyeket tartalmazó numerikus és lebegőpontos típusnál töltődhet ki. Az elhelyezhető tizedesjegyek maximális számát (18) tárolja, a jegyek hosszúsága, valamint a tizedesvessző beszámít a mezőhosszba.

A bevitt adatok mező szerinti rendezését választhatjuk, ha az index oszlopban (**INDEX**) **Y** betűt írunk. Kilépés (**CTRL+END** vagy **EXIT/SAVE** ... menüpont kiválasztása) után a kért indexállományok létrejönnek.

A szerkezet elkészült, az állomány fogadóképes arra, hogy feltöltsük rekordokkal. Íme feltöltve az első rekord a 5-5. ábrán!

Records	Go To	Exit	3:25:48 pm
NEV	Szűvi Péter		
CIM	1118 Bp. Nagy L. u. 23.		
OLVSZ	0001		
DIJ	110		
EV	45.02.01		
JEGYZET	nemc		

Edit || C:\dbase\tanf4\OLVASO || Rec 1/10 || File || Nun

5-5. ábra

A felkínált *inverz sorokba* (adatcellákba) csak a mezőtípusnak megfelelő tartalmú és formájú adatot írhatunk. A cellák között a **nyílbillentyűkkel** lehet mozogni. Az utolsót kitöltve újból megjelenik az adatbevitelt felajánló kérdés. Válaszunkkal eldönthetjük, hogy folytassuk vagy sem az adatbevitelt.

A memo mező feltöltéséhez aktivizálnunk kell a szövegszerkesztőt a **CTRL+HOME** billentyűkombinációval (5-6. ábra). A mező kitöltését a **CTRL+END** billentyűkkel fejezzük be.

A létrehozott új adatbázis-adatállományunk szerkezetét a **DISPLAY STRUCTURE** parancs írja ki a képernyőre. A mezők részletezése mellett az összrekordszámot (**Number of Data Records**), s az utolsó módosítás időpontját (**Date of Last Update**) is feltünteti.

```

Layout Words Go To Print Exit 3:20:53 pr
-----
Ismétlésként gondoljuk át a következőket:

A CREATE paranccsal vadonatúj adatbázisszerkezetet hozunk létre,
az aktuális munkaterületen.

A SELECT parancs segítségével az egyes munkaterületek sorszámát (1-10)
ill. hivatkozási nevét (a-j) megadva válthatunk közöttük.

A USE parancs az előzőleg kiválasztott munkaterületen megnyitja
az adott adatbázis-adatállományt. Ha volt rajta megnyitott
adatállományt, azt lezárja!

Ezért ha több adatbázis-adatállomány akarunk használni
gondoskodni kell a munkaterület-váltásról, s majd ezután adjuk
ki a használatbavételi parancsot. Illetve a USE parancsot az IN
paraméterrel kell kiadni.

A már megnyitott adatbázis-adatállományok munkaterületei között
szintén a SELECT paranccsal közlekedünk, felhasználva a már
megnyitott állomány hivatkozási nevét (alias-at).
Program C:\text\ZNAF Line:1 Col:1 Nun Ins

```

5-6. ábra

DISPLAY STRUCTURE

Structure for database:C:\DBASE\TANF4\OLVASO.DBF

Number of data records: 29

Date of last update : 92.02.23

Field	Field Name	Type	Width	Dec	Index
1	NEV	Character	30	Y	
2	CIM	Character	40	N	
3	OLVSZ	Character	4	Y	
4	DIJ	Numeric	3	N	
5	EV	Date	8	Y	
6	JEGYZET	Memo	10	N	
** Total			96		

Miután létrehoztuk többi állományunkat is, DIR paranccsal kiirathatjuk a nevüket és kiterjesztésüket. (A DIR parancs másként működik, mint ahogy a DOS rendszerprogramban megszoktuk!)

Adjuk ki a DIR parancsot!

```

DIR
OLVASO.DBF
KOLCSON.DBF
KONYV.DBF
PELDANY.DBF

```

Az adatbázis-adatállományok használata

Az újonnan létrehozott állomány nyitva marad a memóriában. Rekordjait feltölthetjük, adataikkal dolgozhatunk. A programból mindig a QUIT paranccsal lépünk ki, ekkor minden állomány, köztük a munkaállományok is lezáródnak. Így munka kezdetén mindig gondoskodni kell az állományok *megnyitásáról*, ún. *használatbavételéről*.

A USE <állománynév> parancs, s paraméterei szolgálják ezt a tevékenységet. Ha nem adunk állománynévet, tehát a paraméter nélkül adjuk ki az utasítást, akkor az előző nyitott állományt lezárja. Az állománynévvel kiadott parancs szintén lezárja az előzőleg megnyitott állományt, de helyette megnyitja az újat.

Próbáljuk ki!

```
USE  
USE OLVASO
```

A kiadott parancs eredménye a státuszsorban látható, megjelent az OLVASO állomány neve, a program így jelzi, hogy egy megnyitott adatállomány munkaterületén vagyunk.

A megnyitott *adatbázis-adatállományról*, *index segédállományokról* áttekintést nyújtanak a DISPLAY STATUS parancs információi. A megjelenő képernyő rendkívül sok információt tartalmaz, s ráadásul angol nyelven. Először tisztázzuk a fogalmakat. Az angol kifejezés után zárójelben a lefordított, majd az értelmezett kifejezést írtuk.

```
Currently Selected Database:  
Select area: 1, Database in Use: C:\TANF\OLVASO.DBF  
Alias: OLVASO  
Production MDX file: C:\TANF\OLVASO.MDX  
Index TAG: OLVSZ Key: OLVSZ  
Index TAG: NEV Key: NEV  
Index TAG: EV Key: EV
```

Currently Selected Database (jelenleg kiválasztott adatbázis) az éppen használt, vagyis az *aktuális adatbázis*. Select Area (kiválasztott munkaterület) a dBASE 10 munkaterületéből az, amelyiken nyitva van az adatállomány. Ezt *aktuális munkaterületnek* nevezzük. Database in Use (használatba vett adatbázis) adatbázis-adatállomány teljes neve. Alias (hivatkozási név) az állomány *másodlagos neve*. (A USE állománynév ALIAS hivatkozási név parancsban kiadhatunk az állomány nevéből eltérő második nevet is. Ha kiadtuk, akkor már csak ezzel a névvel és nem az eredeti állománynévvel választhatjuk ki a megnyitott állományok közül.) Production MDX file - Index Tag (produkciós index-elemek), a rendezéshez szükséges *segédállomány* és elemei. (Ezekre az állományokra a rendezést tárgyaló alfejezetben visszatérünk.)

Az adatbázis-adatállományokat használat után lezárhatjuk. Az adott munkaterületen megnyitott állományt a USE paraméter nélküli parancs zárja le. A CLOSE DATABASE utasítás valamennyi adatbázis-adatállományt lezárja. A QUIT parancs minden memóriában nyitott állományt, értéket lezár, s kilép a dBASE programból. (Munkánk végén kötelező kiadni ezt a parancsot!)

A dBASE IV egyidejűleg 10 *adatbázis-adatállományt* tud nyitva tartani. Az állományok mindegyike külön-külön, önálló *munkaterületen* helyezkedik el. Minden munkaterületen a többitől független *rekordmutató* mozog, mely mindig az *aktuális rekordra* mutat. Indítás után az adatbáziskezelő mindig az első munkaterületen áll.

Amikor az előbb létrehoztuk az OLVASO állományt, akkor ezt az aktuális, vagyis az egyes számú munkaterületen hajtottuk végre. Azóta már négy állományunk van, használni szeretnénk valamennyit!

Ha több adatbázis-adatállományt akarunk egyszerre használni, gondoskodni kell a *munkaterület-váltásról*. A munkaterületek között a SELECT <terület sorszáma>|<hivatkozási név> parancs segítségével mozgunk. A sorszám 1-10 közötti érték lehet. A hivatkozási név vagy az "a-j" karakterek valamelyike, vagy a USE parancs ALIAS paraméterével megadott *hivatkozási név*. A hivatkozási név az állománynévhez hasonlóan max. 8 karakteres név lehet. Amennyiben nem adtuk meg, úgy használhatjuk az eredeti állománynevet is hivatkozási névként. Az egyszerűbb használat végett ez utóbbi megoldást javasoljuk.

Váltsunk a kettes számú munkaterületre!

SELE 2

Látszólag nem történt semmi, de a státuszszorra pillantva tudjuk, hogy jelenleg üres munkaterületen állunk, mivel semmilyen állománynév nem jelent meg a státuszszorban. Most tehát új adatbázis-adatállományt nyithatunk, anélkül, hogy az OLVASO állományt lezárnánk. Például:

```
USE KONYV
```

Gyorsabb megoldást kínál a használatbavételi parancs és IN paramétere.

USE <állománynév> IN <területszám> formában azonnal kijelölhetjük, hogy mely munkaterületen, mely adatállományt kell megnyitni. Figyeljünk arra, hogy eközben nem váltunk munkaterületet, a státuszszorban kiírt állománynév változatlan marad.

Nyissuk meg valamennyi állományunkat más-más munkaterületen, majd ellenőrizzük az eredményt a **DISPLAY STATUS** paranccsal!

```
USE
USE OLVASO
USE KONYV IN 2
USE KOLCSON IN 3
USE PELDANY IN 4
DISP STAT
```

A képernyőn az alábbi sorok láthatók:

```
Currently Selected Database:
Select area: 1, Database in Use: C:\TANF\OLVASO.DBF
Alias: OLVASO
Production MDX file: C:\TANF\OLVASO.MDX
  Index TAG: OLVSZ Key: OLVSZ
  Index TAG: NEV Key: NEV
  Index TAG: EV..Key: EV

Select area: 2, Database in Use: C:\TANF\KONYV.DBF
Alias: KONYV
Production MDX file: C:\TANF\KONYV.MDX
  Index TAG: SZERZO Key: SZERZO
  Index TAG: PAZON Key: PAZON
```

Select area: 3, Database in Use: C:\TANF\KOLCSON.DBF
 Alias: KOLCSON
 Production MDX file: C:\TANF\KOLCSON.MDX
 Index TAG: RAKTSZ Key: raktsz
 Index TAG: OLVSZ Key: olvsz

Select area: 4, Database in Use: C:\TANF\PELDANY.DBF
 Alias: PELDANY
 Production MDX file: C:\TANF\PELDANY.MDX
 Index TAG: PAZON Key: PAZON
 Index TAG: RAKTSZ Key: RAKTSZ

Ellenőrzésül mindig pillantsunk a státuszsor információira! A továbbiakban az aktuális állomány rekordjaival, mezőivel, adattételeivel dolgozunk.

A rekordok egymás után, a feltöltés sorrendjében állnak az állományban. Használatbavételkor mindig a legelsőként bevitt rekorddal dolgozhatunk, megnyitáskor ezt a rekordot "tartja a kezében" a dBASE. Ez az *aktuális rekord*.

Dolgozhatunk egy *másik munkaterületen* megnyitott adatbázis-adatállomány mezőértékeivel is. Azonban ügyeljünk arra, hogy ott csak az aktuális rekord mezőit érjük el. Ekkor használnunk kell az alábbi formulát. A kívánt állomány hivatkozási nevével rámutatunk egy mezőnévre <hivatkozási név> - > <mezőnév> alakban.

Egyszerre mindig csak egy mezőhöz jutunk el! Ahány mezőértéket szeretnénk megjeleníteni, vagy használni, annyiszor kell kiadnunk ezt a formulát.

Például irassuk a KONYV állomány aktuális rekordjának szerzőjét, címét, kiadási adatait!

?KONYV -> SZERZO
 ?KONYV -> CIM
 ?KONYV -> KIADAS

Az adatvisszakeresés alapjai

Az adatvisszakeresés általános ismeretei

A feltöltés sorrendje a rekordok *fizikai sorrendjét* jelenti. Amíg nem változtatjuk meg mesterségesen ezt a sorrendet, addig minden tevékenység e sorrend szerint hajtódik végre. Minden munkaterületen független rekordmutató az adatbázis-adatállomány fizikai sorrendjében mozog.

A **GO <rekordsorszám> | GOTO <rekordsorszám>, | <rekordsorszám>** utasításokkal azonban tetszőleges rekordra állhatunk. Minden rekordnak van *sorszáma*, amelyet az adatbevitelnél kap automatikusan. A rekordsorszám mindig leolvasható a státuszorból. Az adatállomány végén, az utolsó rekord mögött találjuk az ún. *fantom rekordot*. E rekordnak nincsenek mezőértékei, csak száma, mely szám mindig eggyel nagyobb, mint a legutolsó kitöltött rekord. Ha tovább lépünk, az adatállomány végére érünk, és a státuszorban az EOF (fájl vége) kifejezés jelenik meg. A nem aktuális munkaterületen is mozgathatjuk a rekordmutatót az **IN <területszám | hivatkozási név>** paraméterrel. Rekordmutató pozícionáláskor az adatbáziskezelő válaszként kírja a hivatkozási nevet és a rekordszámot.

Példánkban mozgassuk a rekordmutatót a 3. számú rekordra a PELDANY adatbázis-adatállományban!

```
GO 3 IN PELDANY
PELDANY: Record No 3
```

Az utasítás paramétereiként megadhatók a **BOTTOM** (állomány alja), és a **TOP** (állomány teteje) kifejezések is.

```
GO BOTTOM IN OLVASO
OLVASO:Record No 10
```

Négy állományunk létezik, melyeket még létrehozáskor feltöltöttünk próbarekordokkal. Az OLVASO és a KONYV állományokban 10-10, a KOLCSON állományban 45, a PELDANY-ban 33 rekordunk van. Ezeket a rekordokat most már szeretnénk megtekinteni, listázni. Így néhány alapvető adatmegjelenítési vagy adatbáziskezelői szakkifejezéssel *adatvisszakeresési fogalmat* kell tisztáznunk.

Adatok, adatbázismezők, kifejezések értékeinek megjelenítését hajtja végre a ? <kifejezés[lista]> parancs. Kötelezően legalább egy kifejezést meg kell adnunk. De megadhatunk többet is, ekkor vesszővel elválasztva soroljuk fel a *kifejezések listáját*. A szögletes zárójel érzékelteti a parancsokban, hogy megadható valamely paraméter, de nem kötelező (*opcionális*).

Például az OLVASO állomány aktuális rekordjának NEV értékét jeleníti meg a következő utasítás.

```
USE OLVASO
?NEV
Szlávi Péter
```

Irassuk ki az olvasó nevét és befizetési díját is!

```
?NEV,DIJ
Szlávi Péter 55.00
```

Megjeleníthetjük más állomány adattételeit is a hivatkozási név - mezőnév paraméterrel.

```
GO 5 IN KONYV
KONYV: Record No 5
```

```
?KONYV->SZERZO, KONYV->CIM, KONYV->PAZON
Szlovák Gábor Adatbázis-kezelés dBASE-ben 1. 0005
```

Sok dBASE IV parancsban előforduló kifejezés: az *érvényességi kör* és a *feltétel* paraméter. Az adatmegjelenítő DISPLAY és az adatlistázó LIST parancson mutatjuk be működésüket. Először lássuk teljes "fegyverzetükben" ezeket, majd tekintsük át egyenként a paramétereket!

```
LIST/DISPLAY <kifejezés[lista][<érvényességi kör>]
[FOR <feltétel>][WHILE <feltétel>]
```

A kifejezésbe különböző típusú *adatokat, függvényeket* írhatunk, valamint bonyolultabb esetekben *műveleti jelek* segítségével összetett kifejezéseket is képezhetünk. Olyan formulákra gondoljunk, mint egy matematikai egyenlet. Az adatok kétfélék lehetnek egyrészt *állandók*, ezek értékei nem változnak, másrészt lehetnek *változók*. A változók közé

soroljuk az *adatbázismezőket*, ezek adatai a háttértárolón tárolódnak, és a *memóriaváltozókat*, melyek a memóriában tárolódnak.

Mindegyik adat fontos jellemzője a típusa. Adatmezőknél már az adatbázis létrehozásakor meghatároztuk az adattípust. Abban az esetben mikor nem közvetlenül mezőbe viszünk be adatokat ún. *határolójeleket* használunk típusuk jelzésére. Ezekkel "súgjuk" meg a programnak, hogy milyen adatra is gondoltunk.

Ötféle adattípust ismer a dBASE IV:

- A numerikus (BCD) adatnak nincs határolójele.
- A másik számjellegű adatnak, a lebegőpontosnak sincs határolójele, hanem egy függvényt használunk (FLOAT()).
- A karakteres adatokat " " [] ' ' párokkal keretezzük. Azért választhatunk háromféle karakterből, mert előfordulhat, hogy a keretező karakter magában a szövegben is szerepel ("BNV '92").
- A dátum típusú adatot kapcsos zárójelek határolják ({1992.02.24}).
- A logikai adatok határolójele a pont (.Y.).

A műveleti jelek használatára szigorú szabályok vonatkoznak. A műveleteket csak azonos típusú adatokkal lehet elvégezni.

- A karakteres adatokkal csak a *karakteres műveletek*, az összefűzés (+) és az összevonás (–) végezhető el.
- Az *aritmetikai műveleteket* (hatványozás **; szorzás * ; osztás /; összeadás +) kivonás –) numerikus típusú adatokkal végezhetjük el.
- Dátum típusú adatokkal csak igen szűkkörűen tudunk számolni. Kivonhatjuk őket egymásból, az eredmény napok száma lesz, illetve napokat adhatunk vagy vonhatunk ki belőlük, ekkor viszont az eredmény dátum lesz.
- A *relációs műveletek* azonos típusú adatokat hasonlítanak össze, az eredményük logikai érték (igaz, hamis) lesz. Logikai adatok nem hasonlíthatóak össze. Relációs műveletek a nagyobb mint >; kisebb mint <; kisebb vagy egyenlő <=; nagyobb vagy egyenlő >=; nem egyenlő <=>, #; egyenlő =; van benne operátor \$ (csak karakteres adatoknál használható).

- Végül a *logikai műveletek*, melyek csak logikai típusú kifejezések között állhatnak, és eredményük is mindig logikai. A tagadás **.NOT.**; a logikai "és" **.AND.**; a logikai "vagy" **.OR.**

Ha több műveleti jel sorakozik egy kifejezésen belül, akkor a *végrehajtási sorrend* dönt, a kerek zárójelek figyelembevételével.

Bemutatásul végezzünk el néhány műveletet numerikus adatokkal! Figyeljük meg, hogy több tizedesjegy esetén kerekítés is végrehajtódik!

```
?3.14256*4
```

```
12.56
```

```
?12.345678/4
```

```
12.35
```

```
?DIJ*2
```

```
200
```

Az egyes adattípusok megjelenítése a *környezeti paraméterekkel* befolyásolható. A környezeti paramétereket a **SET** parancsszóval kezdődő utasításokkal állíthatjuk be.

Kapcsolójellegű és értékbeállító környezeti paramétereket ismerünk. A kapcsolóknál **ON** vagy **OFF** érték kiadásával be- és kikapcsolhatjuk hatásukat. Például a fontos státuszinformációkat tartalmazó státuszsort a **SET STATUS ON** parancs kiadásával kapcsolhatjuk be.

A különböző adattípusok megjelenítési formáit is számtalan értékbeállító környezeti paraméterrel változtatjuk. (SET menüvezérelt parancsot is használhatjuk, de ezzel nem minden paramétert éthetünk el.)

A numerikus kifejezések tizedesjegyeinek számát a **SET DECIMALS TO <kifejezés>** parancssal adjuk meg.

```
?12.4/4
```

```
3.1000
```

```
?12.345678/4
```

```
3.0864
```

A program eredeti, amerikai dátumformáját a magyar szokásoknak megfelelő sorrendre a **SET DATE YMD** környezeti paraméterrel változtatjuk meg. Kíráthatjuk az évszázadokat is, a **CENTURY** kapcsoló **ON**-ra kapcsolásával. A dátumértékek közötti elválasztójelet a **SET**

SEPARATOR TO <karakteres kifejezés> karakterei közül választjuk ki.

Lássuk az eredményt a **DATE()** függvény segítségével!

```
?DATE()  
1992.01.29
```

Sok **dBASE** parancs tartalmaz ún. *érvényességi kör paramétert*. Ezzel szabályozható, hogy mely rekordcsoportok vegyenek részt a parancsokban. Ezeknek az utasításoknak meghatározott érvényességi köre van. Például: A **DISPLAY** parancs érvényességi köre az aktuális rekord. Ez azt jelenti, hogy kiadásakor csak az aktuális rekordot jeleníti meg. A **LIST** parancs érvényességi köre valamennyi rekord, tehát eredményül az állomány összes rekordját kilistázza.

Többféle érvényességi kör van.

- A **RECORD** <kifejezés> az adott sorszámú rekordot jelent.
- A **NEXT** <kifejezés> megadott darabszámú rekordon (beleértve az aktuálisat is) hajtja végre a parancsot.
- A **REST** az aktuális rekordtól kezdődően az utolsó rekordig, valamennyi rekordon működik.
- Az **ALL** hatásköre az adatállomány valamennyi rekordja.

Irassunk ki néhány teljes rekordot paraméter nélküli paranccsal, majd adjunk meg különböző kifejezéseket és különböző érvényességi köröket!

```
GO 5  
OLVASO: Record No 5
```

```
DISP  
Record# NEV CIM OLVSZ DIJ EV JEGYZET  
5 Pénzes Katalin 1522, Bp. Háros u. 137. 0005 100  
79.04.12 memo
```

```
DISP NEV,CIM  
Record# NEV DIJ  
5 Pénzes Katalin 100
```

LIST. NEV, DIJ

Record#	NEV	DIJ
1	Szlávi Péter	110
2	Zsakó László	100
3	Bárdos János	50
4	Molnár Máténé	50
5	Pénzes Katalin	100
6	KERTÉSZ ÁDÁM	100
7	Kertész Dóra	20
8	KERTÉSZ TAMÁS	20
9	Kertész Csaba	20
10	Sebök János	100

Feltételhez kötött adatvisszakeresés

Míg az érvényességi kör fizikai válogatást nyújt, addig a **FOR <feltétel>**, **WHILE <feltétel>** paraméterek tartalmi válogatást tesznek lehetővé.

A *feltétel* logikai értékű kifejezés, melyet az adatbáziskezelő kiértékel az adott rekordra lépve. Majd a mező tartalmától függően igaz vagy hamis válaszokat kap, és a választ elbírálva vagy végrehajtja, vagy nem az előírt tevékenységet. A parancsokban **FOR** és **WHILE** paraméter külön-külön és együtt is szerepelhet.

A **FOR <feltétel>** paraméter működése lazább. Kiadásakor a parancs végrehajtódik az adatállomány minden olyan rekordján, melyre a feltétel igaz!

Példaként irassuk ki azokat az olvasóinkat, akik több, mint 50 forintot fizetnek!

```
DISP NEV,DIJ FOR DIJ > 50
      1   Szlávi Péter           55.00
      2   Zsakó László          90.00
      5   Pénzes Katalin        55.00
      6   KERTÉSZ ÁDÁM          90.00
     10   Sebök János           55.00
```

WHILE <feltétel> szigorúbb vizsgálatot tart. Egyrészt nem vizsgál az állomány elejétől kezdve, hanem csak az aktuális rekordtól. Majd csak addig a rekordig hajtódik végre a művelet, amíg a feltétel igaz. Így az első hamis rekordon megáll, ez lesz az aktuális rekord a vizsgálódás befejezése

után. Eltérő működését jól megfigyelhetjük, ha megint az 5. számú rekordtól indulunk olvasóink visszakeresésével.

```
GO 5
```

```
OLVASO: Record No 5
```

```
DISP NEV,DIJ WHILE DIJ > 5
```

5	Pénzes Katalinka	55.00
6	KERTÉSZ ÁDÁM	90.00

Sokszor bonyolultabb kérdések megfogalmazására van szükségünk.

Irassuk ki Kertész családnevű olvasóinkat! Karakteres típusú adat esetén a feltételben használnuk kell a keretező jeleket és az egyenlőség műveleti jelet.

```
DISP NEV FOR NEV="Kertész"
```

```
Record# nev
```

```
7 Kertész Dóra
```

```
9 Kertész Csaba
```

A válasz azonban csak a feltételben megadott formájú családneveket írta ki. De "igaza" van, hiszen a nagybetű és a kisbetű nem azonos betű! Irassuk ki a nagybetűseket is!

```
DISP NEV FOR NEV="KERTÉSZ"
```

```
Record# nev
```

```
7 KERTÉSZ ÁDÁM
```

```
8 KERTÉSZ TAMÁS
```

Ha valamennyi, akár kis-, akár nagybetűs nevet szeretnénk kiírni, akkor csak a karakteres típusú *konverziós* (átalakító) *függvények* használatával (UPPER(), LOWER()) kapunk jó eredményt.

A függvény egy segédeszköz a gyorsabb és kényelmesebb munkához. Használatuknál néhány formai szabályt be kell tartani. A függvény alakja : **Függvéynév(argumentum vagy argumentumok)**. Az *argumentumok* a függvény tárgyai, vagyis bemenő adatai, ezeket kell átalakítani, vagy ezekkel kell számolnia stb. Vannak üres, argumentum nélküli függvények is (DATE()). A függvények eredményét a ? parancs segítségével írhatjuk ki a képernyőre.

Használjuk a nagybetűre alakító UPPER() függvényt a feladat megoldására!

```
DISP NEV FOR UPPER(NEV) = "KERTÉSZ"
```

```
Record#  nev
        6  KERTÉSZ  ÁDÁM
        7  Kertész  Dóra
        8  KERTÉSZ  TAMÁS
        9  Kertész  Csaba
```

Bonyolult, összetett feltételek esetében nemcsak egy, hanem egyszerre több feltételnek is teljesülnie kell a rekordban. Ezt a *logikai ÉS* műveleti jel használatával vizsgálhatjuk. Miért használunk pont logikai típusú jelet? Mert a feltételek kimenő értéke logikai érték, tehát közöttük csakis logikai művelet végezhető el!

Irassuk ki a KOLCSON adatállományból azokat, melyeket 1990.11.04 -e előtt kölcsönöztek ki!

```
SELE KOLCSON
```

```
DISP OLVSZ,MOZGAS,HATIDO FOR MOZGAS='k' .AND.
HATIDO <= {90.11.04}
```

```
Record#  olvsz  mozgás  hatido
        1   0001   k       90.01.08
        2   0002   k       90.01.15
        4   0003   k       90.05.27
        5   0004   k       90.07.16
        6   0004   k       90.07.16
        7   0004   k       90.07.16
        8   0004   k       90.07.16
        9   0004   k       90.07.16
       11   0001   k       90.10.15
       12   0007   k       90.10.15
       13   0007   k       90.10.15
       14   0007   k       90.10.15
```

Az adatkezelés alapjai

Az adatkezelés fogalmára használhatjuk az *adatifeldolgozás*, az *adatkarbantartás* rokonértelmű kifejezéseket. Mindegyiken az adatbeviteli, az *adatjavítási* és az *adattörlési* munkálatokat értjük.

Adatbevétel

Az adatbázis-adatállomány végéhez az **APPEND** utasítással fűzünk új rekordokat. Ekkor a képernyőn megjelenő üres adatmezőkbe írjuk adatainkat, ez a tevékenység már ismerős, mivel első rekordunkat is hasonló felépítésű képernyőn töltöttük ki.

Adatmódosítás

Egy adatrekord módosítására az **EDIT** utasítás és paraméterei szolgálnak.

Módosítsuk az **OLVASO** állomány 10. rekordjának díj értékét 100 forintra. A parancs érvényességi köre az aktuális rekord. Lépünk az 10. számú rekordra és majd a módosításhoz használjuk a szerkesztőbillentyűket! Álljunk a díj mezőre és javítsuk át az 55-öt 100-ra! Javítás után vagy a **CTRL+END** billentyűkombinációval, vagy a felső menüsorban található **EXIT/EXIT** menüponton keresztül lehet a javítást véglegesíteni.

```
SELE OLVASO
```

```
10
```

```
EDIT
```

```
Records Go To Exit
```

```
NEVSebök János
```

```
CIM1023, Bp. Kisbognár u.22.
```

```
OLVSZ0010
```

```
EV49.04.24
```

```
DIJ100.00
```

A már rögzített rekordok között felfelé a **PgUp**, lefelé a **PgDn** billentyűkkel lapozhatunk.

A **BROWSE** paranccsal több adatot, max. 19 rekordot táblázatos formában tekinthetünk át, ill. módosíthatunk. Az egyenkénti és a táblázatos forma között az **F2** billentyűvel váltunk.

A 5-7. ábrán megjelenő táblázat leginkább a kézi nyilvántartásainkra emlékeztet. Az adatbázis-adatállomány táblázatos szerkezetét jól megfigyelhetjük. A függőleges oszlopok a mezők, a vízszintes sorok az egyes rekordok.

Records	Fields	Go To	Exit	0:25:00 pm
NEV	CIM	OLVSZ		
Sárvári István	1118, Bp. Nagy L. u. 23.	0001		
Zsákó László	2345, Makó, Horgos u. 1.	0002		
Bárdos János	1221, Bp. Kistétény u. 32.	0003		
Holnár Máténé	1217, Bp. Csepeli u. 3.	0004		
Pénzes Katalin	1522, Bp. Háros u. 137.	0005		
KERTÉSZ ADAM	3453, Szeged, Folyó u. 1.	0006		
Kertész Dóra	3453, Szeged, Folyó u. 1.	0007		
KERTÉSZ TAMÁS	3453, Szeged, Folyó u. 1.	0008		
Kertész Csaba	3453, Szeged, Folyó u. 1.	0009		
Sebők János	1023, Bp. Kisbognár u. 22.	0010		

Browse C:\dbase\tanf4\OLVASO Rec 1/10 file Num
View and edit fields

5-7. ábra

Az oszlopok és sorok között nyílbillentyűkkel mozgunk. A PgUp és PgDn billentyűkkel egész képernyőt lapozhatunk. A menüsorban kényelmesen beállíthatjuk a megjelenő oszlopok szélességét (FIELDS/SIZE FIELDS), rögzítését (FIELDS/LOCK...), törlését (FIELDS/BLANK...). Az oszlopok tartalmát a szerkesztőbillentyűkkel javíthatjuk, megadott adattételeket megkereshetjük (GO TO/FORWARD SEARCH) stb.

Adattörlés

Az adatok törlését végezhetjük egy lépésben, egyszerre valamennyi rekordot kitörölve. Ekkor a törlendő adatbázis-adatállomány munkaterületén kiadott ZAP utasítással dolgozunk. Amennyiben a biztonsági kérdés feltevését biztosító környezeti paramétert (SET SAFETY) bekapcsoljuk, úgy meggondolhatjuk még az adatbázis rekordjainak törlését.

Több lépéses változatnál megkülönböztetünk *logikai* és *fizikai törlést*. Először a logikai törlést, más szóval *törlésre jelölést* hajtunk végre a kiválasztott rekordokon, a DELETE <érvényességi kör> FOR <feltétel> WHILE <feltétel> parancs alkalmazásával. A törlésre jelöltséget visszaállíthatjuk a RECALL paranccsal és paramétereivel. A végleges fizikai törlést a PACK utasítással indítjuk el.

5. Adatbáziskezelés számítógéppel

Töröljük a KOLCSON adatbázis-adataállományban azokat a kölcsönzési tételeket, melyekben könyvek visszahozását adminisztráltuk, majd listázzunk tizet belőlük!

```
DELE FOR MOZGAS='v'
```

```
15
```

```
KOLCSON: Record No 15
```

```
LIST NEXT 10
```

Record#	OLVSZ	RAKTSZ	DATUM	HATIDO	MOZGAS
15	0010	00001001	91.05.01	91.05.15	k
16	0009	00001001	91.05.01	91.05.15	k
17	0008	00001003	91.05.01	91.05.15	k
18	0007	00001004	91.05.01	91.05.15	k
19	*0004	00000201	91.05.01	. .	v
20	*0004	00000301	91.05.01	. .	v
21	*0004	00000401	91.05.01	. .	v
22	*0004	00000501	91.05.01	. .	v
23	*0004	00000601	91.05.01	. .	v
24	*0001	00000101	91.05.01	. .	v

A dBASE a rekordszám és az első mezőérték közé helyezett csillaggal jelzi a logikai törlést. A törlésre jelölt rekordokat elrejtethjük ideiglenesen további parancsaink előtt a DELETE környezeti paraméter ON állapotra kapcsolásával.

```
SET DELETE ON
```

```
LIST NEXT 10
```

Record#	OLVSZ	RAKTSZ	DATUM	HATIDO	MOZGAS
15	0010	00001001	91.05.01	91.05.15	k
16	0009	00001001	91.05.01	91.05.15	k
17	0008	00001003	91.05.01	91.05.15	k
18	0007	00001004	91.05.01	91.05.15	k
25	0005	00000101	91.05.02	91.05.16	k
26	0005	00000201	91.05.02	91.05.16	k
27	0005	00000301	91.05.02	91.05.16	k
28	0005	00000401	91.05.02	91.05.16	k
29	0005	00000501	91.05.03	91.05.17	k
30	0005	00000601	90.05.03	91.05.17	k

Az elrejtett rekordjaink elérhetőek lesznek, ha a törlésre jelölt rekordra pozícionáljuk a rekordmutatót.

Álljunk a törölt 20. számú rekordra, és irassuk ki!

```
GO 20
```

```
KOLCSON: Record No 20
```

```
DISP
```

```
Record#  OLVSZ      RAKTSZ    DATUM      HATIDO    MOZGAS
      20 *0004  00000301  91.05.01      . .      v
```

A rekordjaink törlésre jelöltségét a **RECALL FOR <feltétel>** paranccsal szüntethetjük meg.

```
RECALL FOR mozgas='v'
15 records recalled
```

Amennyiben véglegesen törölni szeretnénk e rekordokat, úgy a tényleges (*fizikai*) törlést eredményező **PACK** parancsot adjuk ki.

```
PACK
```

```
15 records copied
```

```
Rebuilding index - C:\TANF\OLVASO.MDX Tag: NEV
```

```
100% indexed 15 Records indexed
```

```
Rebuilding index - C:\TANF\OLVASO.MDX Tag: OLVSZ
```

```
100% indexed 15 Records indexed
```

```
Rebuilding index - C:\TANF\OLVASO.MDX Tag: EV
```

```
100% indexed 15 Records indexed
```

Az angol nyelvű üzenetekből leolvasható, hogy a **PACK** utasítás tulajdonképpen másolási műveletet végez el.

Az adatok rendezése

Az adatbáziskezelő egyik legfontosabb szolgáltatása a *rendezés*. Bármikor tetszőleges sorrendbe rakhatjuk adatainkat. A könyvtári rendszerünkben nyilvántartjuk olvasóinkat, könyveinket, és a kölcsönzéseket.

Vegyük példának a könyveket! Jelenleg a könyvek úgy sorakoznak az állományban, ahogy beírtuk adataikat a táblázat soraiba. Ha bármilyen

más szempont szerint szeretnénk látni ezeket, rendezni kell. Rendezhetjük szerzőjük, címük szerint, de felsorolhatjuk kiadási évük, vagy kiadójuk sorrendjében is.

A rendezésnek több célja lehet. Nemcsak a felhasználói igények, hanem az adatbáziskezelő bizonyos szolgáltatásai is megkívánják a rendezettséget. Néhány parancs (pl. gyors keresés) el sem végezhető rendezetlen állománnyal, de sok más parancs is sokkal gyorsabban hajtódik végre rendezettség esetén. Megkülönböztetünk *fizikai rendezést* és *logikai rendezést*.

A fizikai rendezés parancsa `SOR ON [<mező1[/ACD]> [<,mező2[/ACD,...]> TO <új állománynév>`. Az utasítás kiadásával új .DBF kiterjesztésű állományba másoljuk át rendezett rekordjainkat. Új adatbázis-adatállomány jön létre, így használat előtt ne feledkezzünk meg megnyitásáról! E rendezés nagy hátránya, hogy az eredeti állományban történő módosítások a továbbiakban nem "öröklődnek" az új állományban.

Egyszerre több mező szerint is tudunk rendezni. Szintenként megadható a *rendezés iránya* (A, D opciók), *szoros rendezés* is megvalósítható (C opció). A parancs *érvényességi körrel* és FOR, WHILE feltétellel rendelkezik.

Példaként rendezzük olvasóinkat név szerint az új OLVNEV adatbázis-adatállományba, majd tekintsük meg az új sorrendet!

```
SORT TO OLVNEV ON NEV
  100% Sorted 10 Records sorted
```

```
USE OLVNEV IN 6
SELE OLVNEV
```

```
LIST NEV
```

Record#	nev
1	Bárdos János
2	KERTÉSZ ÁDÁM
3	KERTÉSZ TAMÁS
4	Kertész Csaba
5	Kertész Dóra
6	Molnár Máténé
7	Pénzes Katalinka
8	Sebők János
9	Szlávi Péter
10	Zsakó László

Nézzünk példát a szoros betűrendre is!

```
SORT TO OLVNEV ON NEV/AC
  100% Sorted 10 Records sorted
```

```
USE OLVNEV IN 3
SELE OLVNEV
LIST NEV
```

Record#	nev
1	Bárdos János
2	KERTÉSZ ÁDÁM
3	Kertész Csaba
4	Kertész Dóra
5	KERTÉSZ TAMÁS
6	Molnár Máténé
7	Pénzes Katalinka
8	Sebök János
9	Szlávi Péter
10	Zsakó László

A logikai rendezés jellemzői

A fizikai rendezés nagy hátránya, hogy az elvégzett rendezés után bevitt új rekordok nem a helyes sorrendben jelennek meg a rendezett állományban. Ehhez új rendezési parancsot kell kiadnunk. A logikai rendezéssel ez a probléma megoldódik. A memóriában lévő *segédállomány*, mely a rendezettséget biztosítja, adatbevitelkor rögtön aktualizálódik. Azért nevezzük logikai rendezésnek, mert a rekordok fizikai sorrendje érintetlen marad.

Indexeléskor létrejön egy *segédállomány*, melyben csak a rendezett *kulcskifejezés* és a *rekordsorszámok* találhatóak. A parancs után a rekordokat az új, logikai sorrendben látjuk az adatállomány használata során. A parancs szintaktikája a következő: **INDEX ON <kulcskifejezés> TAG <tagnév>**.

Egy indexállományban típusától függően egy vagy több sorrendet tartalmaz(ezeket *tagoknak*, vagy *elemeknek* nevezzük). Az *önálló indexállomány* (.NDX kiterjesztésű) csak egyféle sorrendet tartalmaz. A többszörös indexállománynak (.MDX kiterjesztésű) már max. 47 tagja lehet.

A felsorolt segédállományokat használat előtt pont úgy kell megnyitnunk, ahogy az adatbázis-adatállományokat. Ha nem gondoskodunk megnyitásukról a kialakított sorrend felborul az új rekord feltöltésekor.

Az ún. *produkciós indexállománynak* szintén max. 47 tagja lehet, de már az adatbázis-adatállomány létrehozásánál automatikusan képződik, és a továbbiakban már mindig egyszerre nyitódik, záródik vele. Előnye miatt csak a produkciós indexállomány működéssel foglalkozunk.

A produkciós állomány használata

Először elemezzük az indexelés parancsát! A kulcskifejezés tetszőleges bonyolultságú kifejezést tartalmazhat, lehet benne mező, függvény, műveleti jel. A tagnév a dBASE szabályai szerint képezett egyedi név. A sokféle rendezési sorrend közül azt az egyet, amely az éppen aktuális sorrendet biztosítja *főindexnek* (master index) nevezzük. Új indexelési parancs után mindig az új kulcskifejezés lesz a főindex. Már létrehozott indexelési tagot bármikor kinevezhetünk főindexnek a **SET ORDER TO <tagnév>** utasítással. Végül a feleslegessé vált tag törlését **DELETE TAG <tagnév>** végzi.

Kezdjük a rendezést egy egyszerű esettel, rendezzünk rekordjainkat cím szerint!

```
INDEX ON CIM TAG OLVCIM
100% indexed 10 Records indexed
```

```
LIST CIM,NEV
```

```
Record# cim nev
```

10	1023,	Bp. Kisbognár u.22.	Sebők János
1	1118,	Bp. Nagy L. u.23.	Szlávi Péter
4	1217,	Bp. Csepeli 'u.3.	Molnár Máténé
3	1221,	Bp. Kistétény u.32.	Bárdos János
5	1522,	Bp. Háros u. 137.	Pénzes Katalinka
6	3453,	Szeged, Folyó u.1.	KERTÉSZ ÁDÁM
7	3453,	Szeged, Folyó u.1.	Kertész Dóra
8	3453,	Szeged, Folyó u.1.	KERTÉSZ TAMÁS
9	3453,	Szeged, Folyó u.1.	Kertész Csaba

Figyeljük meg a rekordsorszámokat! Eddigi növekvő sorrend helyett látszólagos rendezetlenséget mutatnak, mivel címsorrendben látjuk a rekordokat!

Valamennyi nyitott segédállományt, tagjaikat, valamint magát a fő-indexet nyilvántartja a **dBASE** a *státuszinformációk* között. Nézzük meg a következő kiíratást!

DISPLAY STATUS

Currently Selected Database:

Select area: 1, Database in Use: C:\TANF\OLVASO.DBF

Alias: OLVASO

Production MDX file: C:\TANF\OLVASO.MDX

Index TAG: OLVSZ Key: OLVSZ

Index TAG: NEV Key: NEV

Index TAG: EV Key: EV

Master Index TAG: OLVCIM Key: CIM

A logikai rendezés kulcskifejezésében megadhatunk függvényt is. Jó példa erre a karakterkezelő függvény, mivel az indexelési parancsnak nincs kisbetűt és nagybetűt összeválogató paramétere.

Rendezzük szoros betűrendbe olvasóinkat!

INDEX ON UPPER(NEV) TAG OLVJONEV

100% indexed 10 Records indexed

LIST NEV

Record#	nev
3	Bárdos János
6	KERTÉSZ ÁDÁM
9	Kertész Csaba
7	Kertész Dóra
8	KERTÉSZ TAMÁS
4	Molnár Máténé
5	Pénzes Katalinka
10	Sebők János
1	Szlávi Péter
2	Zsakó László

Lehet olyan kívánságunk is, hogy egyszerre nem csak egy, hanem több szempont szerint akarjuk látni rekordjainkat. Szükségünk van olyan listára, ahol a tagdíjon belül szoros betűrendben listázzuk olvasóinkat; máskor pedig lakcím szerint válogatunk, de ugyanakkor a névsorrendet is meg akarjuk tartani. Ezt a rendezést többszintű, vagy *összetett indexkulcs* szerinti rendezésnek nevezzük.

Előállítására a kulcskifejezések között a karakteres összevonási (+) műveleti jelet kell használni. Mivel ez a műveleti jel csak karakteres adatokat fűzhet össze, gondoskodni kell a különböző adattípusok konverziójáról. Az adatbáziskezelő program sokféle *konverziós függvényt* ismer. Terjedelmi okok miatt most csak egyet mutatunk be.

A numerikus adattípust az STR (<kifejezés> [,<hossz> [,<tizedesjegyek száma>]]) függvénnyel lehet karakteressé átalakítani.

Rendezzük olvasóinkat befizetési díj értékeken belül szoros névsorba!

```
INDEX ON STR(DIJ)+UPPER(NEV) TAG OLVDIJNE
100% indexed 10 Records indexed
```

```
LIST DIJ,NEV
```

Record#	dij	nev
9	10.00	Kertész Csaba
7	10.00	Kertész Dóra
8	10.00	KERTÉSZ TAMÁS
3	20.00	Bárdos János
4	20.00	Molnár Máténé
5	55.00	Pénzes Katalinka
10	55.00	Sebők János
1	55.00	Szlávi Péter
6	90.00	KERTÉSZ ÁDÁM
2	90.00	Zsakó László

A rendezési parancs után az új sorrendet láthatjuk. Új főindexet a **SET ORDER TO <tagnév>** utasítással választhatunk.

Ha megint az adatbevitel sorrendjében szeretnénk látni rekordjainkat, akkor a paraméter nélkül kiadott **SET ORDER TO** parancsot adjuk ki.

A rekordok logikai sorrendjében a **SKIP <kifejezés>** utasítással léphetünk előre, negatív előjellel pedig hátra. A **GO TOP** paranccsal a logikailag a legelső, a **GO BOTTOM** paranccsal pedig a logikailag a legutolsó rekordra állhatunk.

Álljunk a legelső rekordra név szerinti sorrendben, majd haladjunk előre rekordonként, s végül lépünk az utolsó rekordra!

```
SET ORDER TO NEV
GO TOP
OLVASO: Record No 3
```

SKIP

OLVASO: Record No 8

GO BOTTOM

OLVASO: Record No 2

A státuszsorban leolvasható, hogy az adatállomány végére értünk (megjelenik az adatállomány vége jel az EOF).

A különböző kívánalmaknak, szempontoknak megfelelő rekordokat könnyen megkereshetjük a *keresési parancsokkal, függvényekkel*. A menüvezérlési technika segítségével *komplex kérdésállományokat (QUERY)* is készíthetünk.

Kapcsolat több adatbázis-adatállomány között

Ha több adatbázis-adatállományt együtt szeretnénk használni, akkor mindenképpen gondoskodni kell a *kapcsolatteremtésről*. Ha nincs kapcsolat a másik munkaterületen, akkor mindig csak az éppen "kézben tartott" rekordot érjük el. Térjünk vissza gyakorlati példánkhoz.

A könyvkölcsönzési rendszer központi munkaállománya a KOLCSON állomány. Itt visszük be az új könyvkölcsönzéseket, és vesszük vissza a visszahozott könyveinket. A többi állománnyal logikai kapcsolatot kell teremtenünk. Az OLVASO állománnyal azért, mert bármikor szükségünk lehet egy olvasó nevére, lakcímére. A PELDANY adatbázisállományon keresztül a KONYV állománnyal azért, mert listázni akarjuk a kivitt könyvek szerzőjét, címét. stb. A kapcsolatot az ún. *kapcsolati környezeti paraméterrel* teremtjük meg az egyes munkaterületek között. Ezeket a munkaterületeket az adatbázis-adatállományok hivatkozási nevével azonosítjuk. A kapcsolat eredményeképpen a kapcsolt állományokban összehangolt rekordmutató mozgást tapasztalhatunk.

Ha a munkaállományban egy új rekordot viszünk be vagy egy régi rekordra állunk, ez a másik állomány(ok)ban is rekordmutató mozgatót vált ki. Így a megfelelő rekord mindig megtalálja "párját". A munka adatbázis-adatállományt *aktív*, a hozzákapcsoltat *kapcsolt* adatbázis-adatállománynak nevezzük.

A SET RELATION TO [*<karakteres kifejezés>*] INTO [*<alias>*][,*...*] utasítást használjuk. Lényeges tulajdonsága, hogy kapcsolatteremtő kulcskifejezésnek tartalmilag egyeznie kell a kapcsolt adatbázis-adatállomány főindexében megfogalmazott kulcskifejezéssel.

Kapcsoljuk össze állományainkat!

Először nyissuk meg tanfolyami adatbázis-adatállományainkat főindexek megadásával!

```
USE OLVASO ORDER OLVSZ
Master index: OLVSZ
```

```
USE KONYV IN 2 ORDER PAZON
Master index: PAZON
USE KOLCSON IN 3
USE PELDANY IN 4 ORDER RAKTSZ
Master index: RAKTSZ
```

Építsük fel a a kapcsolatokat!

```
SELE KOLCSON
SET RELA TO RAKTSZ INTO PELDANY, OLVSZ INTO OLVASO
SELE PELDANY
SET RELA TO PAZON INTO KONYV
```

Figyeljük meg, hogyan adminisztrálja a kialakított kapcsolatokat a **dBASE!**

```
DISPLAY STATUS
Currently Selected Database:
Select area: 3, Database in Use: C:\TANF\KOLCSON.DBF
Alias: KOLCSON
Production MDX file: C:\TANF\KOLCSON.MDX
      Index TAG:      RAKTSZ Key: raktsz
      Index TAG:      OLVSZ Key: olvsz
Related into: PELDANY
Relation: raktsz
Related into: OLVASO
Relation: olvsz
```

```
Select area: 4, Database in Use: C:\TANF\PELDANY.DBF
Alias: PELDANY
Production MDX file: C:\TANF\PELDANY.MDX
      Index TAG:      PAZON Key: PAZON
      Master Index TAG:      RAKTSZ Key: RAKTSZ
Related into: KONYV
Relation: pazon
```

A kapcsolatteremtés után kiirathatjuk az összetartozó rekordok megfelelő mezőértékeit a jólismert formula segítségével.

```
LIST RAKTSZ, KONYV->SZERZO, OLVSZ, OLVASO->NEV NEXT 10
Record# raktsz konyv->szerzo olvsz olvaso->nev
  28  0004/1 Bodor Tibor 0005 Pénzes Katalinka
   9  0005/1 Szlovák Gábor 0005 Pénzes Katalinka
  30  0006/1 Szlovák Gábor 0005 Pénzes Katalinka
  31  0002/2 Bognár Júlia 0001 Szlávi Péter
  32  0002/3 Bognár Júlia 0003 Bárdos János
  33  0002/4 Bognár Júlia 0008 KERTÉSZ TAMÁS
```

Nyomtatás

Az adatok kezelése, nyilvántartása sokszor nem ér véget azzal, hogy a képernyőn megjelentek adataink. Legtöbbször szeretnénk azokat kinyomtatni is.

Kétféle nyomtatást különböztetünk meg. Nyomtathatunk egyes parancsok (LIST, DISPLAY) TO PRINT paraméterével. Ekkor kapcsoljuk be a SET PRINTER ON kapcsolót.

Nyomtathatunk strukturáltan is, ez azt jelenti, hogy a lap pontosan meghatározott pontjára kívánuk írni, ez az ún. *pozicionált* nyomtatás. *Pozicionált* nyomtatást végző parancsokkal változatos formájú eredményeket állíthatunk elő. Tablókat, táblázatokat, körleveleket az ún. *jelentésformátum* (REPORT forma), egy- és többhasábos címkeket pedig a *cimkeformátum* (LABEL forma) kialakításával hozhatunk létre. A nyomtatás megtervezéséhez használjuk a CREATE LABEL|REPORT <formátumnév> menüvezérelt utasítást. A kialakított eredményt a REPORT|LABEL FORM<formaállomány neve> TO PRINT utasítással nyomtathatjuk ki a SET DEVICE TO PRINTER környezeti paraméter beállítása után.

ISMÉTLÉS A TUDÁS ANYJA!

... azért a jó

SZEMLÉLTETŐESZKÖZ

se kutya



Forduljon a **CONSULT-EXIM BT**-hez, amely az alábbi oktatástechnikai világcégek kizárólagos magyarországi képviselője:

Phywe (fizika, kémia, biológia),

Lucas – Nülle (technika, elektronika, erősáram, számítástechnika, gépjármű-villamosság),

ASC Electronic (nyelvi labor),

Armfield Ltd. (hidraulika, folyamatszabályozás, élelmiszeripari technológia).

Tel.: 116-7534, 176-1811 * Fax: 176-3218 * 1025 Budapest, Törökvész út 58.



MICROSYSTEM
KARRIER



Szakemberközvetítő
és Tanácsadó Kft.

TANULJON PROFIKTÓL!

A MICROSYSTEM
számítástechnikai tanfolyamain
az egész országban államilag elismert
szakképesítést szerezhet.

Nemzetközi színvonalú szakembereink tanítják meg
a személyi számítógépek kezelésére,
számos szövegszerkesztő,
táblázat- és adatbázis-kezelő programra!

Oktatási programjainkat
az Ön felkészültsége és igényei szerint
állítjuk össze!

Várjuk jelentkezését
a 166-6743 vagy a 185-2128-as telefonon,
vagy levélben címünkön:

MICROSYSTEM KARRIER Kft.
1118 Budapest, Somlói út 46/a.

R-TREND

Számítástechnikai és
Ügyviteli Kft.
Szombathely
Széll Kálmán u. 19.
Tel./fax: (94) 24-033



HARDVER

- Számítógépek és perifériák forgalmazása
- Kellékanyag ellátás

SZOFTVER

- Rendszertervezés, -fejlesztés
- Tanácsadás

ÜGYVITEL

- Komplet, illetve részügyviteli feladatok ellátása társaságoknak

KIADVÁNYSZERKESZTÉS

- Újság, szórólap, hirdetés, prospektus, meghívó, névjegy, embléma, könyv tervezése és kivitelezése

SZAKOKTATÁS

- Alapfokú személyi számítógépkezelői tanfolyam
- Középfokú programozói tanfolyam

Mi a komplett megoldást kínáljuk Önnek

6. Számítógép-hálózatok és alkalmazásaik

A számítógép-hálózatok fejlődéstörténete

Kezdetben a számítástechnikai erőforrások, a processzorok ára meglehetősen magas volt, ezért természetes, hogy a rendszerek középpontja az a számítógép volt, ahová *valamennyi adat és felhasználói igény* befutott. Az információkat nyújtó és igénylő környezetből minden adatot *adathordozókon* (lyukkártyán, mágnesszalagon) fizikailag is a számítóközpontba kellett beszállítani, és az eredményeket – főleg nyomtatványok formájában – vissza kellett juttatni az igénylőkhöz. Ennél – a ritkán még ma is alkalmazott eljárásnál – a feldolgozórendszer távol van a felhasználtól, ami megnehezíti a hibák és más rendkívüli események kezelését, nem teszi lehetővé a folyamatos kapcsolatot a felhasználóval. Az *erőforrások koncentrált elhelyezése* nehezíti az egyes rendszerek együttműködését annál is inkább, mert egységes elvek és eljárások hiányában a működés irányíthatatlanná válik.

Az első számítógép-hálózatok kiépítésének célja mindenképp *a távolság áthidalása* volt. A számítógép néhány adatállomása távgépfő volt, melyek a számítógéptől távol is lehetettek, és a központi géphez telefonvonalon csatlakoztak. A hálózatok legegyszerűbb formája már biztosította a távoli hozzáférést, az adatok küldését és fogadását. Az ilyen hálózatokat egyetlen számítógép köré telepítették, és a számítógép csupán egyféle feladatot látott el, például helyfoglalást vagy banki ügyletek intézését. Mivel az összes vonal egyetlen központba futott be, a hálózat *felépítése, topológiája csillag alakú* volt.

Az *időosztásos üzemű számítógépek* megjelenésekor – amelyeknél egyetlen processzor, erőforrásait az elvégzendő munkák között megosztva, párhuzamosan több feladattal foglalkozhat – épültek ki az első terminálhálózatok.

A korszerű hálózatok már nem egyszeri hozzáférésű hálózatok, és nem is egyetlen számítógépet kiszolgáló rendszerek. A mai tipikus hálózatok egyidejűleg több feladat ellátására is alkalmasak. Valamely felhasználó

nálói igény a hálózaton keresztülhaladó üzenetek egész sorát indítja el. Például a tervezési munkáját végző felhasználó együttműködhet egy olyan számítógéppel, amelynek adatállományából adatokat hívhat le, vagy amely számára a bonyolultabb számítási feladatok elvégzéséhez szükséges számítási kapacitást biztosítja.

A számítógép-hálózatok rendeltetése már nemcsak a távolságok áthidalása, hanem az *erőforrások elosztása és összekapcsolása*, valamint a felhasználók közötti *kommunikáció biztosítása*.

A hálózati környezetben folyó munka tipikusan párbeszédes (interaktív): párbeszéd folyik a gép és a felhasználó között, de párbeszéd folyhat a gépek közvetítésével a felhasználók között is. A hálózatok megjelenése a számítógépek alkalmazási területeit kiszélesíti, a korszerű kereskedelmi online információszolgáltatás, a műszaki tervezés vagy a számítógépes gyártás el sem képzelhető hálózatok nélkül.

A ma alkalmazott hálózatok egyre inkább heterogének, vagyis bennük *különböző gyártók számítógépei és termináljai működhetnek együtt*. Ennek elengedhetetlen előfeltétele a kapcsolatok rendszerének és módjának, az együttműködés előírásainak, protokolljainak egységesítése, szabványosítása. Ma még távolról sem állítható, hogy sikerült elérni a mindenre kiterjedő egységesítést, viszont elmondható, hogy e téren soha nem folyt intenzívebben a szabványosítási munka, mint napjainkban.

Az összetett rendszerek – mint amilyenek a hálózatok is – tervezésének első és talán legkritikusabb fázisa a rendszer felbontása önállóan kezelhető és tervezhető alrendszerekre. Ha sikerül az alrendszerek funkcióit pontosan meghatározni, akkor egyszerűbben és megbízhatóbban dolgozhatók ki a köztük levő kapcsolatok is.

A nemzetközi szabványszervezet, az ISO (International Standard Office) által kezdeményezett OSI (Open Systems Interconnection) célja: olyan nyitott hálózati rendszerek koncepciójának meghatározása, mely hálózatok különböző funkcionális pontjain bármely gyártó termékei telepíthetők, ha azok működése megfelel az OSI előírásainak. A koncepció szerint a rendszert funkcionális alrendszerekre – rétegekre – bontják, a rétegek közötti csatolókat a lehető legegyszerűbbre választják.

A nyílt rendszerek megvalósíthatóságára irányuló törekvés mellett a fejlődés másik fő iránya az *integráltság fokozása*, vagyis hogy egy hálózaton belül különböző szolgáltatások egy időben legyenek igénybe vehetők. Az ISDN (Integrated Service Digital Network – integrált szol-

gáltatású digitális hálózat) rendszerben a beszéd, az adatok, a szövegek és a képek leírása és továbbítása azonos alakban, digitális formában történik, és az adatjelek rendeltetésére vonatkozó *kísérő információ* határozza meg, hogy a feldolgozás helyén a jelfolyamot beszédként vagy éppen képként kell-e megjeleníteni.

Az ISDN rendszereknél a különböző rendeltetésű információk átvitele történhet párhuzamosan, nincs akadálya annak, hogy az átvitt kép mellett beszédet is továbbítsunk (ez lehet például egy képtelefon-szolgáltatás), de lehetőség van arra is, hogy a továbbított adatok rendeltetését beszédes üzenettel közöljük (például űrlapok távolsági kitöltése, árrendelés stb.). Az ISDN is bizonyítja azt a tendenciát, hogy a különböző rendeltetésű *hálózatok* (telefonhálózatok, adatátviteli hálózatok) fokozatosan *közelítenek egymáshoz*.

A hálózatok kiépítése

A hálózat kialakítása szempontjából meghatározó tényező az áthidalt távolság nagysága. Ez alapján helyi, nagyvárosi, körzeti és távolsági hálózatok különböztethetők meg.

A helyi hálózat (LAN – Local Area Network) kiterjedhet egy helyiségre, épületre vagy telephelyre. A körzeti hálózat (RAN – Regional Area Network) jellegzetesen egy intézmény vagy nagyobb vállalat adott földrajzi körzetben fekvő telephelyeit, nem ritkán a telephelyek helyi hálózatait köti össze. A távolsági hálózat (WAN – Wide Area Network) országnyi, földrésznyi, sőt akár az egész földfelületre kiterjedő lehet.

Az egyes hálózattípusok között nehéz éles határvonalat húzni, bár igaz, hogy a helyi hálózatok körzete általában nem nagyobb egy 5 km sugarú kör területénél, a körzeti hálózatoké 25-30 km sugarú kör.

A helyi és a körzeti hálózatok közé sorolhatók az úgynevezett nagyvárosi hálózatok (MAN – Metropolitan Area Network), melyeknél a lefedett terület inkább a körzetre jellemző méretű, a technikai megvalósítás viszont a helyi hálózatok felépítési elveit követi.

A helyi és a körzeti hálózatok legtöbbször vállalatok, intézmények tulajdonában vannak, a távolsági hálózatok állomásai általában több tulajdonoshoz tartoznak; az átviteli közegeket állami vagy társadalmi ellenőrzés alatt álló magánvállalatok birtokolják.

A hálózatok legfontosabb elemei: a szolgáltatásokat nyújtó számítógépek (ezek számításokat, feldolgozást, információkeresést, tranzakciót, adattárolást végezhetnek), a hálózat felhasználóinak adatállomásai, az adatkoncentrátorok és az adatátviteli vonalak (melyek a felhasználói adatállomások egy-egy csoportját az adatátviteli alrendszerre csatolják).

A hálózati elrendezési sémák összefoglalására a topológia elnevezést használjuk. A topológia lehet sín-, fa- és gyűrűszerkezetű, szövevényes, teljesen összefüggő és kombinált felépítésű.

A sínhálózatnál az állomások egyetlen közös csatornán kommunikálnak egymással. A fahálózat jellegzetes típusa a hierarchikus rendszer, amelynek csúcsán található a kiemelt szerepű számítógép. A csillaghálózatnak egyetlen csomópontja van, ehhez futnak be az egyes állomások adatútjai. A gyűrűhálózatok egyik fajtájánál a szomszédos állomásokat közvetlen vonalak kötik össze, a másikonál a vezérállomásból kiinduló és oda visszatérő zárt hurok mentén található az állomások. Ez utóbbit hurokhálózatnak is hívják. A teljesen összefüggő hálózatban minden állomást külön adatút köt össze a többivel. A szövevényes hálózat a teljesen összefüggő hálózat csonka változata. A kombinált topológiai hálózatok különböző formájú részhálózatokból vagy alhálózatokból épülnek fel.

Kezdetben adatátviteli vonalként csak a telefonhálózatot használták. Ismert tény, hogy a telefonközpontok a hívó és a hívott közötti átviteli utat a hívó akaratának (a hívott számnak) megfelelően építik ki. Ez esetben kapcsolt vonalokról beszélünk. Az adatátvitel is végezhető ilyen módon, e megoldás előnye, hogy a kapcsolatot csak addig kell fenntartani, amíg van mit továbbítani. Hátránya viszont az, hogy a régi kapcsolóközpontok esetében a nem megfelelő kontaktusok miatt az átviteli út zajos, az átvitel hibás lehet.

A kontaktushibák elkerülésére alkalmazzák a *bérelt vonalakat*, amikor is a két végpont között állandóan felépített összeköttetéseket alakítanak ki. Ez esetben az átviteli út jobb minőségűvé válik, de ez a megoldás lényegesen költségesebb, mint a kapcsolt vonalak alkalmazása. A bérelt vonalakért ugyanis azok használatán kívül is fizetni kell, nem is beszélve arról, hogy a lekötött vonalak csökkentik a hálózati rendszer átviteli kapacitását. A korszerű elektronikus telefonközpontok korában már egyre ritkábban használják a bérelt vonalakat.

A különböző műszaki megoldásokat követő helyi hálózatok összekapcsolására valók a *hidak*. Ezek lehetnek belső vagy külső hidak. Az első esetben a szerver (vezérlő számítógép), a másodikban egy munkaállomás tölti be a híd szerepét. A hidakkal felépített hálózati rendszerek előnye, hogy bármely munkaállomás – függetlenül attól, hogy eredetileg melyik alhálózathoz csatlakozik – tetszés szerint hozzáférhet a hálózati rendszerben számára engedélyezett összes erőforráshoz.

Egy helyi hálózat és egy nagyszámítógép (és hálózat) közötti közvetlen kapcsolat kiépítésére a kapuk (gateway) szolgálnak. A kapu emulációs szerverből és csatoló hardverből áll, amelyet fizikailag a helyi hálózat egyik szerverén, vagy erre a célra dedikált munkaállomáson valósítanak meg. A nagyszámítógép felől nézve, a kapun keresztül kapcsolódó helyi hálózati elemek a rendszer termináljaiként viselkednek.

A helyi hálózatokra távoli munkaállomások is kapcsolódhatnak, például a telefonvonalon át. Ez esetben az átviteli út két végén *modemet* találunk. Az ilyen módon kapcsolódó munkaállomások átviteli sebessége – az átviteli elemek korlátozott átteresztőképessége miatt – lényegesen kisebb, mint a helyi hálózaton belüli munkaállomásoké.

Átvivőközegek jellemzői

A jelek hordozója lehet fizikai közeg, fémvezeték (érpár vagy koaxiális kábel), fényvezeték (száloptikai kábel), vagy rádióhullámok közvetítésével az éter is, tehát beszélhetünk vezetékes és vezeték nélküli átvitelről. Az átvivőközegen többet értünk, mint magát a fizikai közeget, ideszámítjuk az átvitelnél használt egyéb elemeket (erősítőket, kapcsolókat) is.

A távolsági hálózatoknál bármelyik átvivőközeg előfordulhat. Legtöbbször vegyesen használják ezeket. A felhasználó – például egy helyfoglaló rendszerre kapcsolt terminálnál dolgozó – nem is tudja, hogy hívása milyen úton jut el az információt szolgáltató számítógéphez. Annál is inkább rejtve marad ez számára, mert nagy kiterjedésű hálózatoknál legtöbbször alternatív utakat biztosítanak, s a hívásokat irányító kapcsolóközpontok automatikusan döntenek arról például, hogy egy európai hívás tenger alatti kábelen vagy éppen műholdak közvetítésével jusson el az Egyesült Államokba.

Az átvivőközeg típusa meghatározó az átviteli kapacitásra, a sebességre nézve, vagyis arra, hogy egységnyi idő alatt mennyi információ

A helyi hálózatok szabványosításában szerepet játszó fontosabb intézmények és testületek

6-1. táblázat

	Nemzetközi	Regionális	Nemzeti
A szabványosítással hivatalból foglalkozó intézmények, szervezetek és testületek	ISO IEC ICAO	CEN (CEC) CENELEC (CEC) ESZR	ANSI NBS DIN BSI MSZH
Egy-egy területen monopolhelyzetű szervezetek és testületek	CCITT (ITU)	NATO EURATOM (CEC)	PTT U.S.NIM Committee Katonai vonal
Közös érdekeltségűek csoportosulással	nemzetközi jelentőségű	ECMA	EIA IEEE
Széles körű befolyással rendelkező vállalatok	IBM, AT&T, XEROX, GENERAL MOTORS		

A jelölések jelentése:

ANSI	American National Standard Institute
AT&T (ATT)	American Telegraph & Telephone Company
BSI	British Standard Institute
CCITT	Comité Consultatif pour Telegraphie et Telephonie
CEC	Commission of the European Communities
CEN	Comité Européen de Coordination des Normes
CENELEC	Comité Européen de Normalization Electrotechnique
DIN	Deutsches Institut für Normung
ECMA	European Computer Manufactures Association
EIA	Electronic Industries Association (U.S.A.)
ESZR	Egységes Számítógéprendszer
EURATOM	European Atomic Energy Community
IBM	International Business Machines Corporation
ICAO	International Civil Aviation Organization
IEC	International Electrical Commission
IEEE	Institute of Electrical and Electronic Engineers
ISO	International Organization of Standardization
ITU	International Telecommunications Union
MSZH	Magyar Szabványügyi Hivatal
NATO	North Atlantic Treaty Organization
NBS	National Bureau of Standards (U.S.A.)
NIM	National Instrumentation Methods Committee (U.S.A.)
PTT	Postal Telegraph and Telephone Authorities (általánosságban) Postes Telegraphes et Telephones (a francia posta)

áramolhat a közegen. Az átviteli sebesség szempontjából lassú, közepes és gyors hálózatokat különböztetünk meg.

A leglassúbb hálózatok a telefonhálózatra épülők, melyeket a beszéd frekvenciatartományának (0,3–3,4 kHz) átvitelére alakították ki. Ezeken a hálózatokon a digitális adatjelek eredeti formájukban nem is továbbíthatók, ezért alkalmazzák a *modemeket* (MODulátor és DEModulátor). Feladatuk, hogy a küldő oldalon a digitális jeleket analóg jelekké alakítsák (modulálják), míg a vételi oldalon az analóg jelekből visszaállítsák az eredeti digitális tartalmat (demodulálják). A modulálás során a digitális jel két állapotához (0 és 1) két különböző frekvenciát rendelnek.

A modulálási technikák finomításával elérhető, hogy – a telefonhálózatok kis sáv szélessége ellenére – az átviteli sebesség jelentősen, például 19,2 kbit/s-ig is növelhető.

A közepes sebességű hálózatok lehetnek a nyilvános és a zártkörű távolsági adathálózatok is. A világ egyik legelső nagy kiterjedésű magánhálózata az Egyesült Államokban a főleg katonai célra kidolgozott ARPANET esetében 50 kbit/s az átviteli sebesség.

Gyors hálózatok azok, melyeknél az átviteli sebesség nagyobb 0,5 Mbit/s-nál, és nagyon gyorsak azok, amelyeknél ez az érték meghaladja az 50 Mbit/s-ot is.

A nemzetközi együttműködés érdekében az átviteli sebességeket szabványosították, és a CCITT (Comité Consultatif Internationale de Télégraphique et Téléphonique – Nemzetközi Távíró és Távbeszélő Tanácsadó Bizottság) ajánlásaiban tették közzé (6-1. táblázat).

A különböző átviteli közegek közül a legkisebb átviteli sebesség az érpárral érhető el, nagyobb az átviteli sebessége és így a kapacitása is a koaxiális kábelnek, és hatványozottan nagyobb a fényvezetőnek. Ez utóbbi a legkevésbé érzékeny a külső zavaró tényezőkre, például a mágneses és a villamos terek hatásaira, ami csak tovább növeli jelentőségét. Bár a fénykábeleknél sok gondot okoz a csatlakozások kialakítása (ami nem jelent különösebb feladatot a fémvezetők esetében), mégis egyre inkább terjednek, különösen, mert a korszerű fényvezetőknél akár több kilométeres távolság is áthidalható a jel közbelső erősítése nélkül. Kezdetben elsősorban az országos hálózatok gerincvezetőit készítették száloptikai kábelekkel, napjainkban a kontinenseket összekötő kábelezésnél is használják az optikai átvitelt. Minden valószínűség szerint a fénykábeleket a jövőben a telefonközpontok és a lakások közötti távolság áthidalására is használni fogják.

Az átviteli tartomány lehet alapsávú és széles sávú.

Az analóg telefontechnikában az *alapsávon* a beszédsávot, vagyis a 0,3–3,4 kHz közötti frekvenciatartományt, a digitális beszédátvitelnél pedig a 64 kbit/s-os jelsorozat közvetlen átviteléhez szükséges sávot értik.

A *széles sávú* hálózatokban az információ átvitele modulált szinuszos jellel történik, az átvendő jelet egy vivőjelre ültetik. A vivőhullám frekvenciája sokkal nagyobb, mint a továbbítandó jel frekvenciája. A széles sávú hálózatok jellegzetessége, hogy frekvenciaosztással egymástól függetlenül több csatornát képeznek, melyek mindegyikén önálló adatátvitel alakítható ki. E hálózatok egyesíthetők egyetlen hálózattá is (például egy nagyobb információtartalmú jel továbbításánál vagy az átviteli idő lerövidítése érdekében).

Az üzenet továbbítása

A hálózatokon a kommunikáló partnerek üzenetekkel váltanak egymással információt. Az üzenet általában három részből áll: a *fejrészből*, mely megjelöli az üzenet forrását és rendeltetését, valamint más vezérlőelemeket is tartalmaz; a tulajdonképpeni továbbítandó információból, az *üzenetből*; végül a *zárórészből*, amely az üzenet végét jelzi, és általában ellenőrző információt is tartalmaz.

A hosszú üzeneteket általában célszerűbb kisebb részekre, csomagokra bontva továbbítani. A csomagokat szintén ellátják fejrésszel és zárórésszel. A csomag nem szintaktikai egység, nincs szükségképpen önálló jelentése. A csomagokból a vétel helyén kell előállítani a szemantikailag hibátlan üzeneteket. A korszerű adatátvitel gyakran csomagkapcsolt hálózatokon folyik.

Két állomás (vagy egy állomás és az erőforrást nyújtó számítógép) között a jelek átviteli iránya többféle lehet: egyirányú (szimplex), váltakozó irányú (félduplex) vagy kétirányú (duplex) üzemmódban kommunikálhatnak egymással.

Egyirányú üzemmódban az egyik állomás csak adó, a másik csak vevő lehet, az átvitel mindig csak egy irányban, az adótól a vevő felé folyhat.

Váltakozó irányú átvitelnél mindkét irányban áramolhatnak az adatok, de időben egymástól elválasztva, vagyis amíg az egyik állomás ad, a másik csak vehet, vagy fordítva.

Kétirányú üzemmódban mindkét állomás egy időben betöltheti az adó és a vevő szerepét, vagyis egy időben küldhet adatokat egy másik állomásnak, és fogadhatja annak küldeményét is.

Az adatátvitel az időbeni lefutás szerint lehet aszinkron és szinkron.

Az *aszinkron* vagy más néven *start-stop* üzemmódot a karakterszervezésű üzenetek továbbításánál – például a nemzetközi online információs szolgáltatásoknál – használják. Itt az egyes karakterek továbbításának kezdeti időpontja véletlenszerű. A karakterek továbbítása start bittel (bittekkel) kezdődik, és a karaktert stop bit zárja le. Ezért egy karaktert például 10 bit jellemez, következésképpen egy 300 bit/s átviteli sebességű hálózaton másodpercenként 30 karakternyi információt lehet továbbítani.

A *szinkron* üzemmódnál az üzenetet vivő jel adását és vételét ugyanazon órajel-generátor szinkronizálja. Ezt a módot tipikusan a bitszervezésű üzenetek továbbításánál használják.

Kapcsolási technika szerint megkülönböztetünk vonalkapcsolt, üzenetkapcsolt és csomagkapcsolt hálózatokat.

A vonalkapcsolt hálózatokban adatátvitel csak akkor végezhető két pont között, ha előzőleg a két pont között az *összeköttetés* már felépült. Az átvitel befejezése után az összeköttetést bontani kell, éppen ezért az átvitel során három munkafázist különböztetünk meg: az összeköttetés felépítését hívással, az összeköttetést felhasználó kommunikációt, majd a kapcsolatot bontását.

A kapcsolat felépítésének egyik része a fizikai összeköttetés megvalósítása, másik része a logikai kapcsolat megvalósítása.

A *fizikai kapcsolat* megvalósításához nagyon sokszor a nemzeti posták nyújtanak segítséget (a szükséges átviteli utak és kapcsolóközpontok biztosításával); a *logikai kapcsolatteremtés* során a szolgáltató számítógéppel kell kapcsolatot teremteni, például egy online szolgáltató adatbázis kínálatából kell kiválasztani azt, amit használni akarunk. Ez utóbbi a LOGON folyamat, melynek során a szolgáltató rendszer ellenőrzi a használati jogosultságot (jelszavak alapján), és a felhasználót azonosítva, a felmerülő költségeket annak számlájára terheli.

A nyilvános hálózatú szolgáltatások igénybevétele után a felhasználó két számlát kap(hat), az egyiket az igénybe vett távközlési szolgáltatásért (a postától), a másikat az adatbankok (vagy más szolgáltatások) igénybevételeért a (számítógépes szolgáltatótól).

A hálózaton keresztül végzett munka utolsó szakasza a logikai kapcsolat bontása (ez a LOGOFF) és a fizikai kapcsolat leépítése.

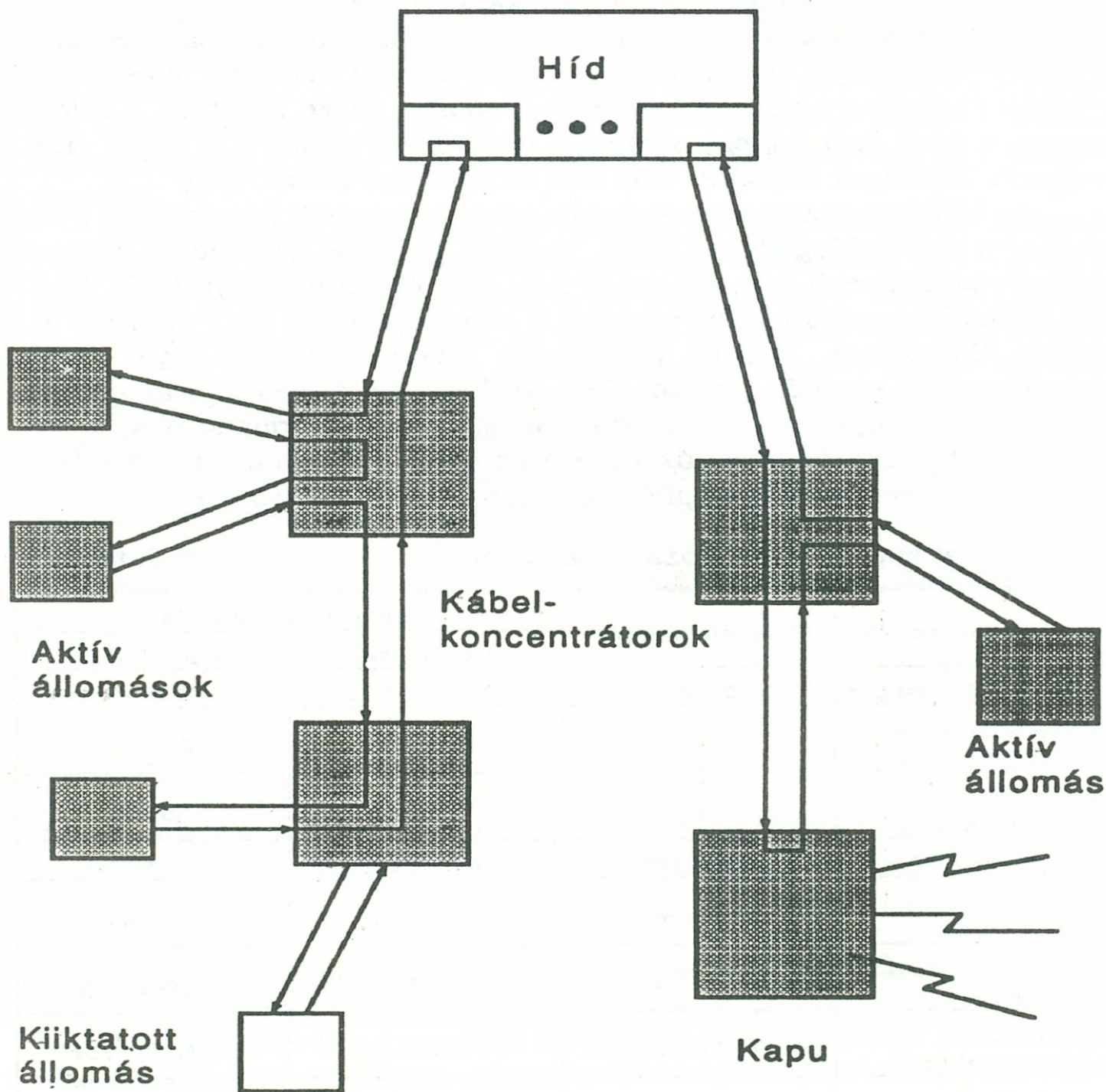
Helyi hálózatok, hálózati szoftverek

Bár a piacon nagyon sokféle helyi hálózattípus jelent és jelenik meg, ma már világosan látszik, hogy alapvetően kétféle változat számíthat hosszú távú sikerre. Ezek a Xerox Palo Alto Research Centerben 1972-ben megkezdett kutatás eredményeként megszületett Ethernet rendszer, valamint az elsősorban az IBM nevéhez fűződő Token-Ring hálózat (6-1. ábra). Mindkettőt (részben) szabványosították, ami tovább erősíti pozíciójukat (6-2. táblázat).

Az Ethernet és a Token-Ring egymáshoz viszonyított főbb előnyei és hátrányai

6-2. táblázat

Jellemző	Ethernet		Token-Ring		Melyik a jobb?
	Előny	Hátrány	Előny	Hátrány	
Szabványosítás	mind a kettő szabványosított				egyforma
Topológia	kevesebb hibalehetőségű			új állomások telepítése problémákkal járhat	Ethernet
Üzenetközvetítés módja		véletlenszerű ütközések	pontos időzítésű	túl részletezett	Token-Ring
Átviteli sebesség	10 Mbit/s		a sebesség nagyon sok felhasználónak elegendő	4 Mbit/s	Ethernet
Kábelezés	egyszerűbb, mint a Token-Ring		olcsó	bonyolultabb, mint az Ethernet	nincs lényeges különbség
Új állomások telepíthetősége	egyszerű			bonyolult és költséges lehet	Ethernet
Költségek	általában olcsóbb, mint a Token-Ring			általában költségesebb, mint az Ethernet	Ethernet



6-1. ábra A Token-Ring hálózat: fizikai csillag-logikai gyűrű

Az Ethernet kísérleti változata után a Xerox, a DEC és az Intel együttműködésével készült el az "Ethernet 1" specifikációja 1980-ban, s az "Ethernet 2" (kissé módosított változatban) 1982-ben.

Az Ethernet alapsávú hálózat. Az átvitel fizikai közege koaxiális kábel, melyen adatsomagok (ezt ez esetben keretnek nevezik) áramlanak. A csomagok átvitele véletlenszerű, az átviteli sebesség maximális értéke 10 Mbit/s. A hálózat szegmensekből épül fel, egy szegmens legnagyobb hossza 500 méter, ennél nagyobb távolságok áthidalása csak több szegmensen oldható meg.

Az Ethernet-típusú hálózatok építéséhez az integrált áramköröket gyártó és adatátvitellel foglalkozó cégek áramkör családokat fejlesztettek ki. Ezek szolgáltatásai kisebb-nagyobb mértékben el is tér(het)nek egymástól. A velük felépített áramköri kártyák személyi számítógépekbe vagy nagyobb gépekbe illeszthetők. Néhány jelentősebb Ethernet-vezérlőkártya, illetve áramkör család készítői: Interlan, Seeq Technology, Advanced Micro Devices, Digital Equipment Corporation, Fujitsu, Intel, Ungermann-Boss.

Az Ethernet-típusú hálózat gyors és sok munkaállomást – terminált – tartalmazó rendszerek kiépítésére használható (6-3. táblázat).

Az Ethernet különböző változatainak jellemzői

6-3. táblázat

Hálózati jellemzők	Kísérleti Ethernet	Ethernet-specifikáció	
		1.0 változat	2.0 változat
Az átviteli sebessége	2,94 Mb/s	10 Mb/s	10 Mb/s
Maximális állomás-állomás távolság	1 km	2,5 km	2,8 km
Maximális szegmenshossz	1 km	500 m	500 m
Jelkódolás	Manchester	Manchester	Manchester
A koaxiális kábel impedanciája	75 ohm	50 ohm	50 ohm
Jelszint a koaxiális kábelben	0-tól + 3 V-ig	0-tól - 2 V-ig	0-tól - 2 V-ig
Adó-vevő kábelcsatlakozók	25 és 15 érintkezős	15 érintkezős	15 érintkezős
Az adatkeret zónáinak hossza			
- az előtag	1 bit	64 bit	64 bit
- a CRC kód	16 bit	32 bit	32 bit
- a cím	8 bit	48 bit	48 bit
Az állomások maximális száma	255	1024	1024

Az IBM Token-Ring helyi hálózat a gyűrűhálózatok családjába tartozik, mely logikai felépítését tekintve gyűrű, fizikai kialakítását tekintve csillag alakú lehet.

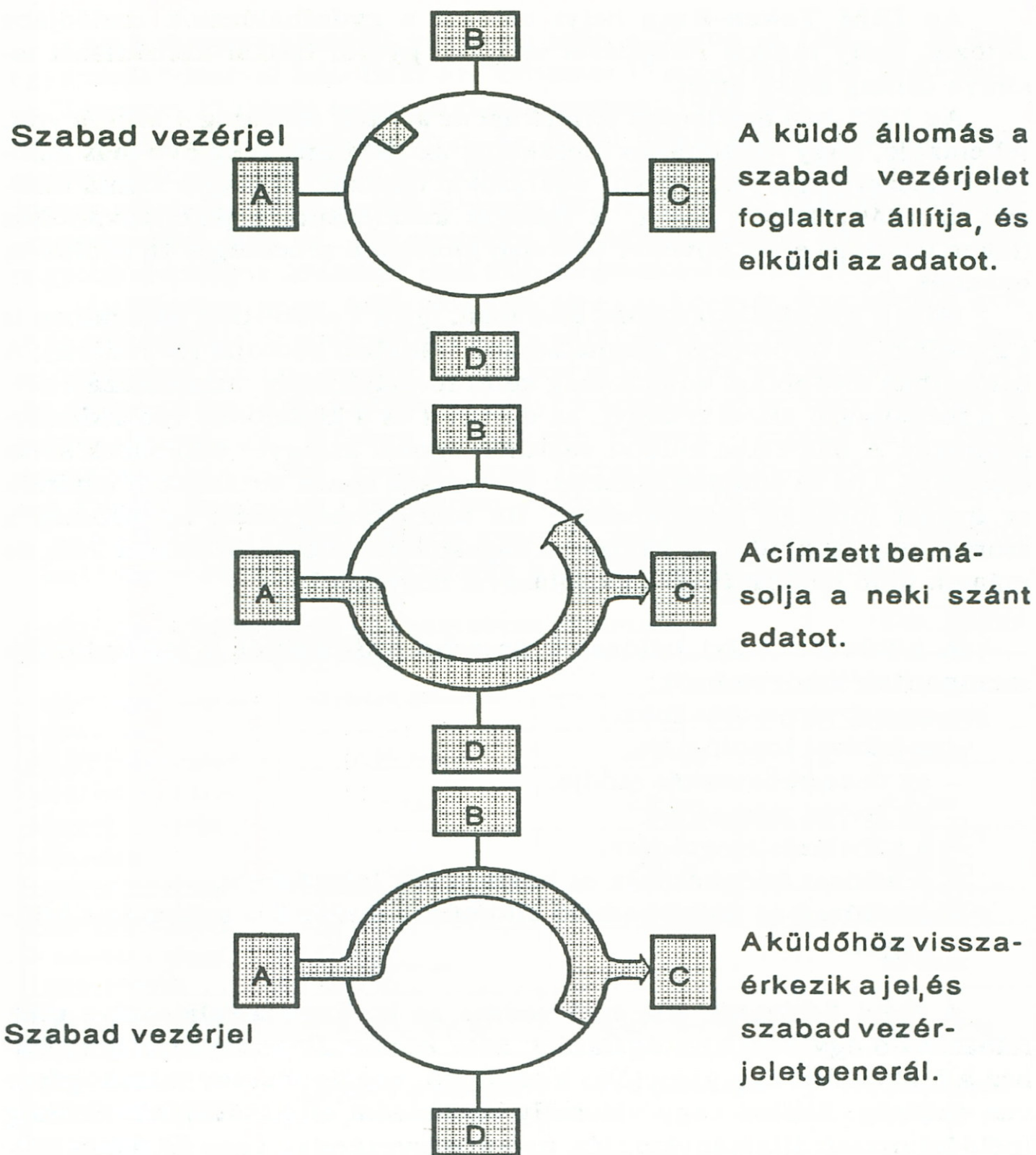
Az 1985-ben bejelentett termékénél az átviteli sebesség 4 Mbit/s volt. Jellemzője, hogy rugalmasan illeszkedik az IBM már meglévő más hálózati rendszereihez, szabványos eljárásokat használ, és magas szintű hálózati szolgáltatásokat nyújt. A hálózat kezelésénél a jelzéstovábbítás (token passing) elvét követik. Újabban az átviteli sebességet 16 Mbit/s-ra növelték.

Mint a gyűrűhálózatokban általában, így a Token-Ring hálózatban is a gyűrűben az információ meghatározott irányban körbejár (6-2. ábra). A hálózatban továbbított adatcsomag keret formájú, mely tartalmazza a cél- és a forgalmazó állomás címét, az üzenetet és a különböző vezérlőinformációkat. A hálózatba küldött adatcsomagokat az egyes állomások sorra érzékelik, s ha az címükre érkezett, kiolvassák annak tartalmát, ellenőrzik az üzenet hibátlan megérkezését. Ha nincs címegyezés, az állomás a csomagot továbbküldi. A hálózatba kapcsolható állomások száma 260, de számuk több hálózat összekapcsolásával növelhető.

A különféle helyi hálózatok összehasonlításánál a legfontosabb szempontok a következők:

- a szabványosítás foka,
- a hálózat topológiája,
- az üzenetközvetítés módja,
- az átvitel sebessége,
- a kábelezés megoldása,
- a hálózat felépítésének és bővítésének lehetőségei,
- a hálózatban működtethető állomások maximális száma és a költségek.

A helyi hálózatok jellemző vonása az is, hogy alkalmasak-e több felhasználó egyidejű kiszolgálására, azaz *osztott állománykezelésre*. Ehhez a funkcióhoz egy vagy több kiszolgáló, vagyis szerver számítógépre van szükség. Ezeket vagy valamely részfeladat ellátásával bízzák meg (például osztott állománykezelés, nyomtatásvezérlés), vagy általános célra használják. Az előbbi esetben dedikált üzemben, az utóbbinál nem dedikált módban működnek.



6-2. ábra A gyűrűn történő forgalmazás változata

A hálózat *teljesítményét* erősen befolyásolják a szerver gépek, amelyek e szempontból legfontosabb jellemzői:

- a processzor típusa és teljesítménye,
- az operatív tároló kapacitása,
- a merevlemez(ek) kapacitása,
- az átviteli adatutak kapacitása.

A helyi hálózati rendszer meghatározóan fontos eleme a **hálózati szoftver**, amelyből nagyon sokfélét fejlesztettek ki az idők folyamán. Ilyenek például a Microsoft Lan Manager, az *IBM PC Network*, a Lantastic és sorolhatnánk. Nálunk a Novell cég *NetWare* operációs rendszere a legnépszerűbb, a helyi hálózati rendszerek több, mint 90%-ánál használják. A népszerűség egyik oka, hogy a NetWare számos helyi hálózati hardverrendszert támogat. Jó a dokumentációja, és egyszerű az üzembe állítása is.

A Novell 1983-ban készítette a NetWare első változatát, amelyet az akkori PC-khez használt Intel 8086 és Intel 8088 processzorokhoz szánt. Az új mikroprocesszorokhoz újabb változatokat készítettek, ilyen az Advanced NetWare 286 is, amelyet az Intel 286-os rendszerekhez kínáltak. Ez már a többszerveres konfigurációkat, több helyi hálózat összekapcsolását is lehetővé teszi (internetworking).

Újabb változatot a **hibatűrő operációs rendszerek** (System Fault Tolerant – SFT) jelentenek, amelyeknél gondoskodnak arról, hogy a szerver hardver meghibásodása ne járjon együtt adatvesztéssel. Előnye, hogy a kiszolgáló rendszerek részleges kiesése nem gátolja a teljes rendszer működőképességét.

A NetWare legújabb változata a *NetWare 386*, amelyet az Intel 386-os és Intel 486-os processzorral felépített számítógépekhez fejlesztették ki. Ez teljesítményében és szolgáltatásaiban minden korábbi változatot felülmúl, például 32 terabájt (elméleti) lemezkapacitású és 250 felhasználó kezelésére képes. Jelszavas titkosságvédelemmel gondoskodik arról, hogy a rendszerben tárolt adatokhoz csak a jogosultak férhessenek hozzá. A NetWare 386 teljesen kompatibilis a korábbi változatokkal, így a már működő rendszerbe minden további nélkül integrálhatók 386-os szerverek is. Az operációs rendszer felépítése moduláris, kialakítása nyílt, hozzákapcsolhatók olyan szoftvermodulok, amelyek funkcionálisan kiegészítik az alapoperációs rendszer képességeit. Ezeket a modulokat NetWare betölthető moduloknak (NLM-nek) nevezik. A koncepciót használva, szerverre alapozott segédprogramok és alkalmazói

A különböző NetWare-verziók jellemzői

6-4. táblázat

Verzió	Adv. 2.15	SFT 2.15	NW 2.2	NW 3.11
Munkahelyek maximális száma	100	100	5, 10, 50, 100	20, 100, 255
Tárolókapacitás	2 gigabájt	2 gigabájt	2 gigabájt	32 gigabájt
Maximális nyitott állomások száma	1000	1000	1000	100 000
Volume-méret	256 megabájt	256 megabájt	256 megabájt	32 terabájt
Volume-ok maximális száma	32	32	32	1024
Maximális állományméret	256 megabájt	256 megabájt	256 megabájt	4 gigabájt
Maximális RAM-kapacitás	12 megabájt	12 megabájt	12 megabájt	4 gigabájt
Dedikált/nem dedikált	mindkettő	dedikált	dedikált	dedikált
Belső híd	igen	igen	igen	igen
Külső híd	igen	igen	igen	igen
Munkahelyek operációs-rendszer támogatása				
DOS 3.X	igen	igen	igen	igen
DOS 4.X	igen	igen	igen	igen
OS/2	igen	igen	igen	igen
Windows	igen	igen	igen	igen
Macintosh	igen	igen	igen	igen
TTS	igen	igen	igen	igen
UPS monitor	igen	igen	igen	igen
SFT szint	I.	I+II.	I+II.	I+II+III.

rendszerek fejleszthetők ki. Alkalmas arra is, hogy különböző protokollok szerinti adatkapcsolatokat alakítsanak ki a felhasználók (6-4. táblázat).

A rendszerek fejlődése révén dinamikusan változik a helyi hálózatok piaca.

A különböző helyi hálózattípusok közül az *európai piacon* az első helyen az Ethernet áll, 48%-os részesedéssel. 1993-ig nem is fog változni a helyzet, sőt az évi növekedés 23%-os lesz. Igaz, a forgalmi értéket tekintve az Ethernet-hálózatok részaránya az idők folyamán viszonylagosan csökkenni fog. Míg 1987-ben 68% volt az értékreszesedésük, 1993-ra 58%-ra csökken. E szegmens legfontosabb szállítója még a DEC, és kisebb részesedésű a 3Com.

A LAN-piacon viszonylagos újdonságnak számító Token-Ring hálózatokhoz 1987-ben 38 150 csomópontot állítottak üzembe, ami akkor a piac 7%-át jelentette. E hálózattípusnál szintén igen gyors fejlődés figyelhető meg. Az évi növekedés elérheti a 65%-ot, s így 1993-ban már 759 000 csomópont üzembe állítása várható egyetlen év alatt, amivel a Token-Ring a piac 43%-át uralja. A szállított rendszerek értékét tekintve a Token-Ring piaci részesedése 1987-ben 6% volt, és 1993-ban már a LAN-piac 38%-át birtokolhatják.

A többi helyi hálózattípus jelentősége egyre inkább csökkenni fog a szabványos Ethernet- és Token-Ring-hálózatok árnyékában.

Ugyancsak lényegi változást hozhat a helyi hálózatok piacán a *UNIX-alapú személyi számítógépek* megjelenése, és a hardverújdonságok közé tartozó, a kifejezetten az ilyen hálózatok termináljaként kifejlesztett, – többek között az IBM, valamint a 3Com cég által forgalmazott – "lemez nélküli" *személyi számítógépek* is. Ezek szolgáltatásainak választéka a "buta terminálokkal" és a teljes kiépítettségű személyi számítógépekkel elérhető közé esik.

A helyi hálózatok elterjedéséhez további ösztönzést adnak az igazán többfelhasználós környezetre szánt alkalmazói szoftverek.

A hálózatkezelő szoftverek területén jelenleg a Novell NetWare termékei jelentik a de facto szabványokat, s ennek megfelelően jelentős a piaci részesedésük is. Valószínű azonban, hogy a jövő években megjelenő új hálózati szoftverek már más koncepciókat követnek, s ezért várhatóan hosszú távon nem a Novellé lesz a vezető szerep. A hosszú távú megoldást jelentő hálózati szoftvereket minden bizonnyal az OSI elvei szerint alakítják ki, valószínű, hogy ezen a területen nőni fog az IBM piaci részesedése. Mindez nem jelenthet a Novell számára a közeljövőben lényeges piacvesztést, hiszen pillanatnyilag nagyon stabil lábakon áll a LAN-piacon.

A távolsági hálózatok

Az információtechnológia társadalmasításával egyre inkább nő az igény nagy kiterjedésű, akár az *országhatárokon is átmenő* adatforgalmat biztosító *hálózatok* iránt. Ma már egyes nagyobb hálózatokra több tízezer terminál is kapcsolódhat. Az igazán fejlett rendszereknél a hálózatok különböző erőforrások, szolgáltató számítógépek elérését teszik lehetővé egy időben. S az is gyakori, hogy ugyanaz a szolgáltató több különböző hálózaton keresztül, sőt különböző átviteli módon (csomagkapcsolt vagy vonalkapcsolt hálózaton) keresztül érhető el. Igaz ez a világ jelenleg legnagyobb online információszolgáltatója, a Dialog Information Services esetében is, melynek székhelye a Palo Altóban van (Kalifornia).

Ilyen nagy rendszerek üzemeltetése csak alaposan kigondolt és a gyakorlatban előzetesen kipróbált elvek szerint, széles körű szabványosítás mellett képzelhető el. Az új elvek megszületése és elterjedése között akár sok év is eltelhet. A fejlesztési idő – nem kismértékben a nemzetközi szabványosítási szervezetek koordinációs tevékenységének köszönhetően – az utóbbi években fokozatosan lerövidül. Az olyan előírások, mint a *CCITT X.25-ös* csomagkapcsolási szabványa, elősegítik a távolsági hálózati rendszerek terjedését. Korábban az IBM-gépek tulajdonosai az IBM SNA – System Network Architecture – előírásait, a Digital Equipment Corporation által készített gépek birtokosai a DNA – Digital Network Architecture – követelményeit követték.

Mint minden hálózat, a távolsági hálózatok is alapvetően két fő elemből, a távolsági kommunikációt biztosító *nyilvános- vagy magánhálózatból* és az *erőforrásgépekből* állnak. Az utóbbiak szolgáltatásait a hálózaton keresztül a felhasználók transzparens módon – vagyis mintha nem is nagy távolságokat áthidalva fordulnának a gépekhez – vehetik igénybe. Az átvitel folyhat analóg vonalakon vagy digitális csatornákon, az átviteli sebesség 110 bit/s-tól 2048 kbit/s-ig változhat. A nemzeti postai hatóságok és a független szolgáltatók csomagkapcsolt adathálózatokat (PSDN – Packet Switched Data Networks) és vonalkapcsolt adathálózatokat (CSDN – Circuit Switched Data Networks) nyújtanak az igénylőknek.

Az erőforrás- vagy szolgáltató gépek, processzorok piacát az IBM uralja. Bár az utóbbi években sokat veszített a piacból, még mindig a legnagyobb szállító, egymaga nagyobb részt birtokol a piacból, mint az összes konkurense együttvéve. A processzorok piaca az 1980-as évek közepéig, értékét tekintve, évi 8%-kal nőtt. A processzorok fajlagos ára-

nak csökkenése és a személyi számítógépek növekvő piaci részesedése miatt a nagygépek piaca most már kisebb lendülettel nő, az 1990-es évek első harmadára már csak az évi 3%-os emelkedés lesz a jellemző. Az átlagot meghaladó növekedés várható e területen Németországban, Angliában, Franciaországban, Olaszországban és Spanyolországban.

A terminálok piaca a processzorokénál dinamikusabban nő, de hosszabb távon itt is a lassulás jelei várhatók.

A modemek típusválasztéka igen gazdag. A CCITT V. sorozatú ajánlásai rögzítik a különböző átviteli sebességeket, a V.21 esetében ez 300 bit/s, a V.33-nál 14,4 kbit/s. Az alacsonyabb átviteli sebességű modemeket a telefonvonalakon át történő adattovábbításnál, míg a nagyobb sebességűeket a bérelt vagy speciális építésű vonalaknál használják. Napjainkban a kisebb sebességű osztályoknál tipikusan 1200 vagy 2400 bit/s-os sebességet, a nagyobb sebességű átvitelnél 9600 bit/s-ot használnak.

Világszerte – és így az európai országokban is – a nemzeti postákra kitüntetett szerep hárul a nyilvános csomagkapcsolt és vonalkapcsolt adatátviteli szolgáltatások biztosítása terén. A szolgáltatások fajtái és minősége függ a telefonhálózat infrastruktúrájától, a rendelkezésre álló analóg és digitális vonalak minőségétől, s még számos más tényezőtől.

A nemzeti nyilvános csomagkapcsolt hálózati szolgáltatás ideális azok számára, akiknek csak időszakosan van szükségük az adatátvitelre, vagyis számukra feleslegesen költséges lenne saját hálózat kiépítése. Európában egy év alatt 30%-kal nőtt az X.25, illetve az X.28 előírásokat követő terminálkapcsolatok száma.

A hálózati szolgáltatások iránt az érdeklődést megfelelő telefonrendszerrel és a szolgáltatások körének bővítésével lehet fokozni. A szolgáltatások köre az elektronikus postai levelezési szolgáltatásokra, különböző adatbázis-szolgáltatók elérésének biztosítására és a legkülönbözőbb emelt szolgáltatású hálózati lehetőségekre (VAN – Value Added Network) – mint automatikus hibaészlelés és -javítás – terjedhet ki.

A szolgáltatások köre azáltal is növelhető, ha a nemzeti csomagkapcsolt hálózaton keresztül más külföldi, illetve nemzetközi hálózatok is elérhetőek. Ezekhez a nemzetközi hálózatokhoz sokszor speciális kapcsolóeszközöket fejlesztenek ki. Ilyen világméretű hálózatok a McDonald Douglas Tymnet rendszere vagy a Telenet.

Nyílt rendszerek – OSI

Napjainkban a számítástechnikai ipar talán legégetőbb problémája *a rendszerek összekapcsolhatóságának – a konnektivitásnak – a megoldása*. Az ipar és az alkalmazók a hálózatokra, a kisgép-nagygép kapcsolatokra, az elosztott adatbázisokra és az együttműködő számítástechnikai rendszerek kiépítésére koncentrálnak. A számítástechnikai erőforrások elosztása, az országok, sőt a kontinensek közötti kapcsolatok kiépítése ma nem egy nemzetközi tevékenységet folytató vállalat és szervezet elsőrendű célja. Olyan egységes koncepció betartása tehát a feladat, hogy heterogén eszközparkból is *egységesen működő rendszereket lehessen felépíteni*.

Ha visszatekintünk a számítástechnika dinamikus fejlődésére, láthatjuk, hogy szinte az élet minden területére kínálnak számítógépes megoldást. De a hallatlanul gyors fejlődési ütem és a komoly versenyharc éppen az egységesítés, *a szabványosítás ellen hatott*. A hardverszektort vizsgálva, látható *az eszközpark széles választéka*, amely a személyi számítógépektől – sőt ezen belül a táskagépektől – a szuper teljesítményű óriásgépekig terjed. Mai eszközállományunk – a világpiac egészét tekintve – több mint 50 ezer nagygépből, 300 ezer minigépből és 50 milliót meghaladó személyi számítógépből áll. Az alkalmazási szoftverek ötvenezres kínálatában minden elképzelhető feladatra találunk valamilyen megoldást. Az igazi problémák a számítógépes rendszerek és alkalmazások integrálásánál kezdődnek. Az alkalmazási rendszerek telepítése az egyik eszközparkból a másikkra, illetve különböző forrásokból származó eszközökből felépített rendszerekben egységes felhasználói környezet kialakítása ma nem egyszerű feladat, de általánosan jelentkező elvárás. A rendszerek – számukat és teljesítményüket tekintve egyaránt – növekedtek, anélkül hogy a gyártók különösebb figyelmet fordítottak volna a szabványosításra. Számos önálló alkalmazási környezet, sziget alakult ki, amelyek önmagukban működőképeseek voltak, de integrálásuk legtöbbször megoldhatatlan, jobb esetben csak körülményesen megoldható feladat.

Ezt felismerve, egyes alkalmazói körök, nemzetközi szabványosítási szervezetek, majd később maguk a számítógépgyártók is szorgalmazták a rendszerek összekapcsolhatóságának megoldását. Javaslataiknál a már meglévő megoldásokat vették alapul, és azok általánosítására törekszenek.

A rendszerek összekapcsolhatóságának legáltalánosabb keretét *a nemzetközi szabványosítási világszervezet, az ISO* égisze alatt dolgozták ki. Az ISO-OSI (Open Systems Interconnections) koncepciója keretében készült *hét réteges modell* alapján épülnek ma a nyílt rendszerek. A szabványosítás

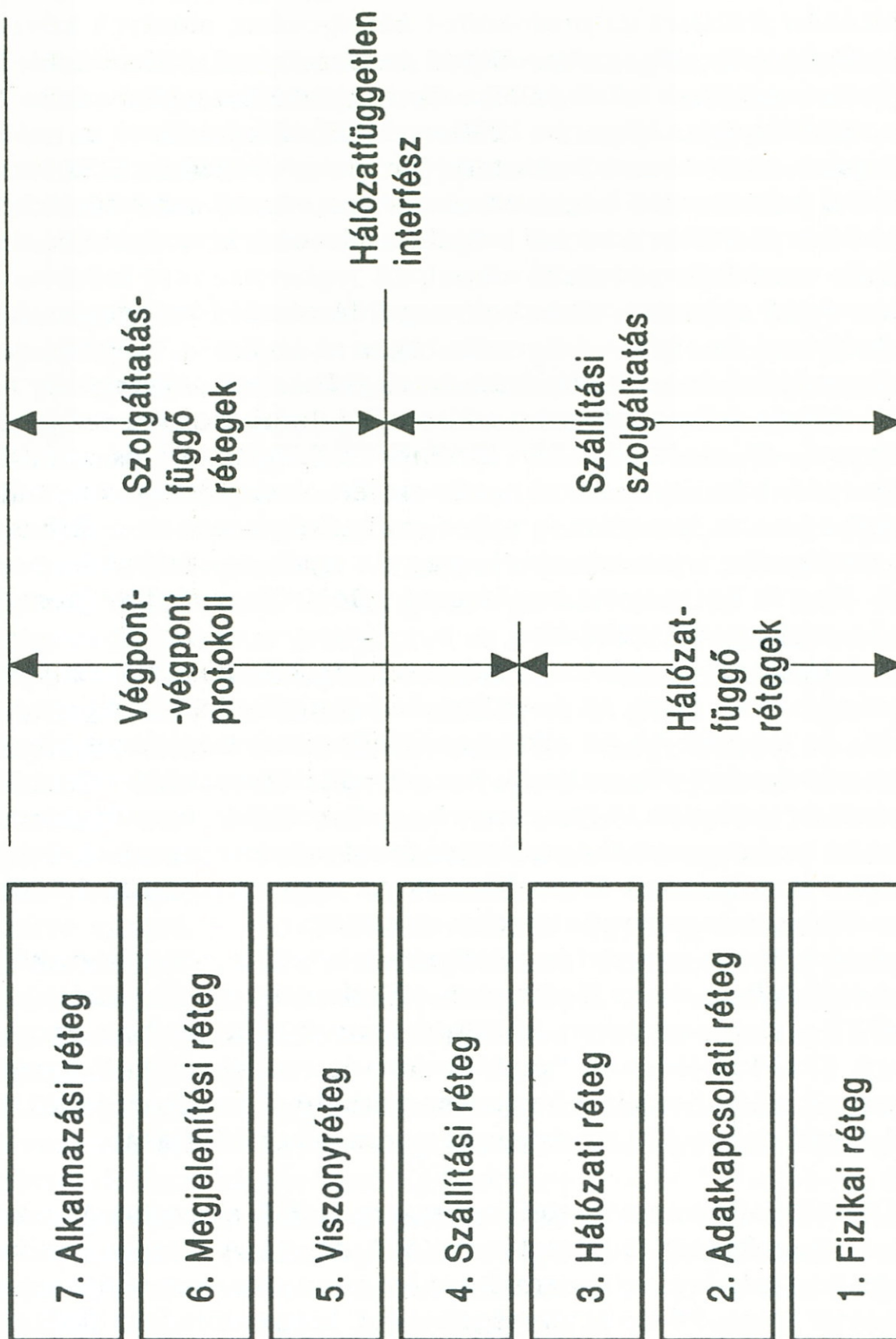
célja nem a gépek belső működésének feltétlen egységesítése, hanem olyan *közös csatlakozási felületek és protokollok kidolgozása*, amelyek követésével a gyártótól függetlenül együttműködő rendszerek alakíthatók ki. Ez a törekvés a felhasználókon kívül találkozik a gyártó szándékával is. Nem túlzott az a megállapítás, hogy az 1990-es években csak azok a szállítók remélhetnek piaci sikereket, akiknek termékei megfelelnek az OSI koncepciójának. Ebből következik, hogy ma az összes vezető számítástechnikai gyártó valamilyen mértékben követi a nyílt rendszerek koncepcióját, és így az OSI modellt veszi fejlesztéseinél alapul.

A felhasználó számára a hardvermegoldásoknál lényegesen fontosabb az a környezet, amelyben dolgozik. Jogos az elvárása, hogy az eszköz forrásától függetlenül *azonos alkalmazási és felhasználói interfészt használhasson*. A közös felhasználói interfész azt jelenti, hogy a számítógép típusától függetlenül azonos módon történhet a felhasználó és a rendszer közötti párbeszéd, bármilyen alkalmazás esetén. A szabványosított alkalmazási és felhasználói interfészek teljes szabadságot adnak a felhasználóknak az eszközök beszerzésénél, vagyis *nyílt rendszerkörnyezetet biztosítanak*. E nyílt környezet természetes velejárója a szoftverportabilitás és a rendszerek együttműködése.

A *portabilitás* az operációs rendszerek és alkalmazási szoftverek azon képességére utal, hogy azok különböző típusú gépeken egységesen használhatók, és jelentik olyan alkalmazásfejlesztési környezet meglétét is, amelyben géptípustól függetlenül, azonos alkalmazási interfészt használó szoftverek készíthetők. A *rendszerek együttműködése* pedig arra utal, hogy heterogén eszközparkkal építhetünk fel olyan rendszereket, amelyeken az adatok tetszőlegesen áramolhatnak az egyes eszközök között. E képességet sokszor interoperativitásnak mondják.

A ma használatos *operációs rendszerek* közül e célra leginkább a *UNIX-alapúak* felelnek meg. Ezért nem véletlen, hogy az előrejelzések szerint az OSI a szoftverpiacon is döntő átstrukturálódáshoz vezet, az egyes gyártók által kifejlesztett "saját operációs rendszerekkel" szemben a szabványosnak tekinthető(k) elterjedése várható. Ez indokolja a UNIX (és a UNIX-alapú operációs rendszerek) gyors piaci hódítását.

Az ISO-OSI referenciamodell egymásra épülő, egymással kommunikálni képes rétegeket különböztet meg (6-3. ábra). A réteges szerkezet értelme, hogy segítségével *csoportosítani lehet az alkalmazói folyamatok együttműködését* biztosító tevékenységeket (pl. kommunikáció fizikai közege – jelátalakítás vagy gondoskodása hibamentes jelátvitelről stb.).



6-3. ábra Az ISO-OSI referenciamodell rétegei

A tevékenységek ésszerű szétválasztása, a feladatok részenkénti megfogalmazása lehetővé teszi a moduláris szabványrendszer kidolgozását.

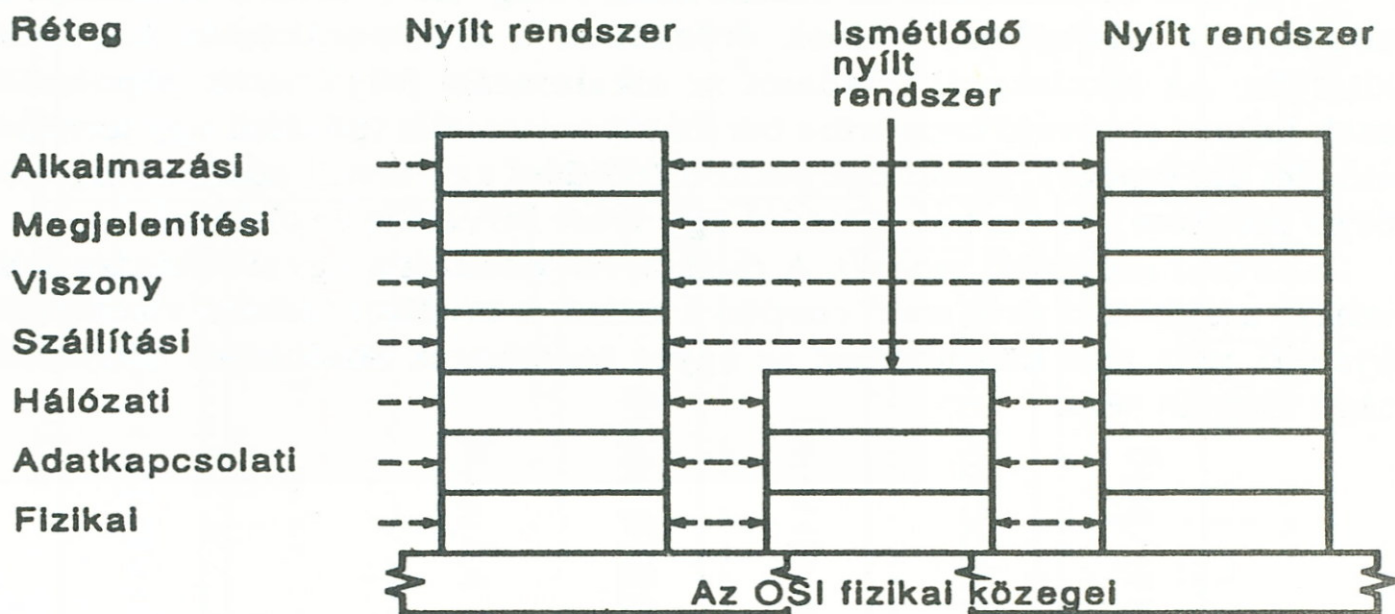
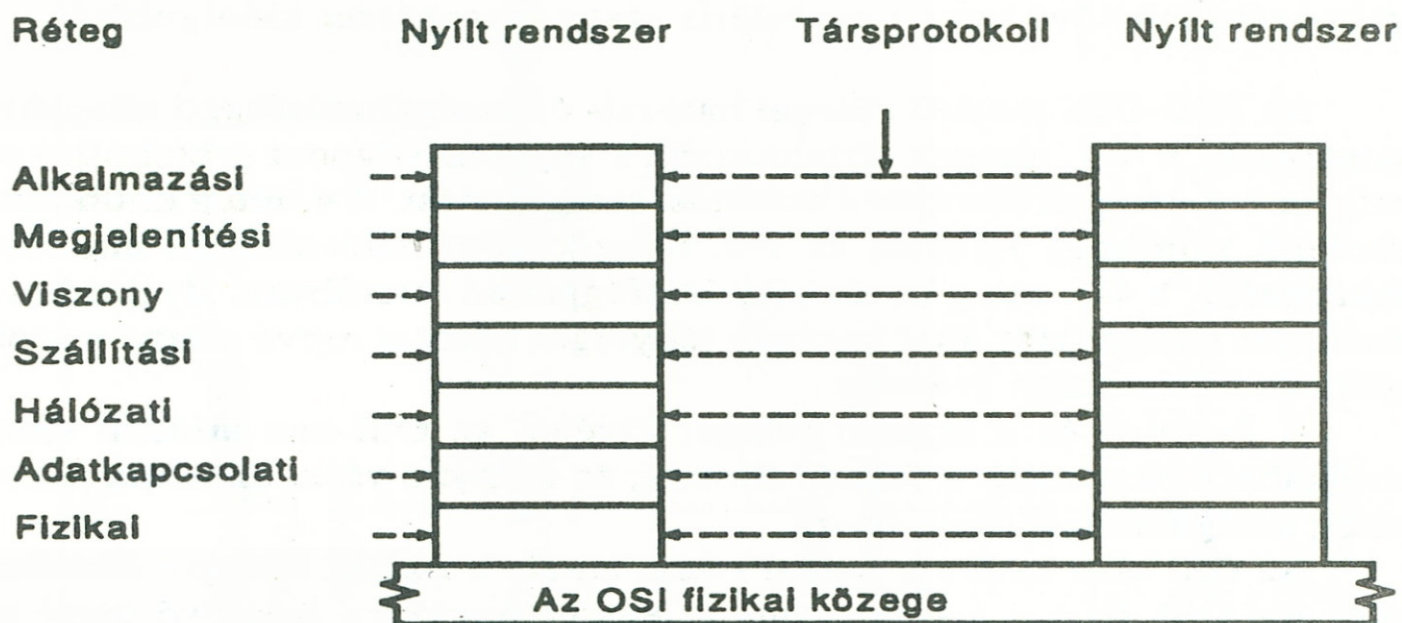
Az ISO-OSI modell rétegei hálózat- és szolgáltatásfüggő rétegekre bonthatók. A 4-7. rétegek tartalmazzák a végpont-végpont protokollokat, míg az 1-4. rétegek alkotják a szállítási szolgáltatást. A 4. réteg kitüntetett szerepű, minthogy feladata az 1-4. réteg közötti hálózatfüggő interfész leképezése, a 4-7. réteg közötti hálózatfüggetlen interfészre. Ily módon a szállítási szolgáltatás által használt tényleges hálózat rejtve marad a szolgáltatás felhasználói számára.

A legfelső és a legalsó réteget kivéve, az OSI-ban minden réteg szolgáltatásokat nyújt a felette állónak, és igénybe veszi az alatta fekvő réteg szolgáltatásait (6-4. ábra).

Az OSI alsó határa a fizikai réteg, amely a fizikai közegre illeszkedik. Az OSI fizikai közege, ami a jeleket hordozva lehetővé teszi az információáramlást, lehet sodrott érpár, árnyékolt érpár, szimmetrikus érnégyes, légvezeték, koaxiális kábel, üvegszál, éter. Sodrott érpár és koaxiális kábel alkalmazása esetén elektromos áramok, üvegszál esetén a fény, az éterben pedig elektromágneses hullámok hordozzák az információt.

Az OSI felső határa az alkalmazási réteg. Ez illeszti a rendszert az alkalmazói feladathoz, aminek érdekében a kommunikációs kapcsolat létrejött. Az alkalmazói feladatot az alkalmazási folyamatok képviselik, ezek három alapvető csoportba oszthatók: manuális (például egy terminál mögött ülő ember), számítógépesített (például egy távoli adatbázist lekérdező program) és fizikai (például egy ipari folyamat vezérlése).

Az OSI absztrakt modell. A rendszerek konkrét megvalósítására semmiféle megkötést nem tesz, csupán a külső, a kívülről látható viselkedést írja elő, nem tesz különbséget az egyes rendszerek különböző bonyolultsága alapján sem.



6-4. ábra Ismétlődő nyílt rendszert is magába foglaló adatközlés

ISDN

Ma a *kommunikáció többféle módon*, beszéd, írott szöveg, adat és/vagy képek továbbításával folyhat. A hírközlésben alkalmazott eszközök választéka már-már kaotikusan bőséges. Használunk képernyővel bővített elektronikus írógépeket, videotex- és teletexterminálokat, távírókat és -másolókat, információt közlünk személyi számítógépes és nagygépes hálózatokon. Nem is beszélve a kommunikáció mindmáig legfontosabb eszközéről, a telefonról!

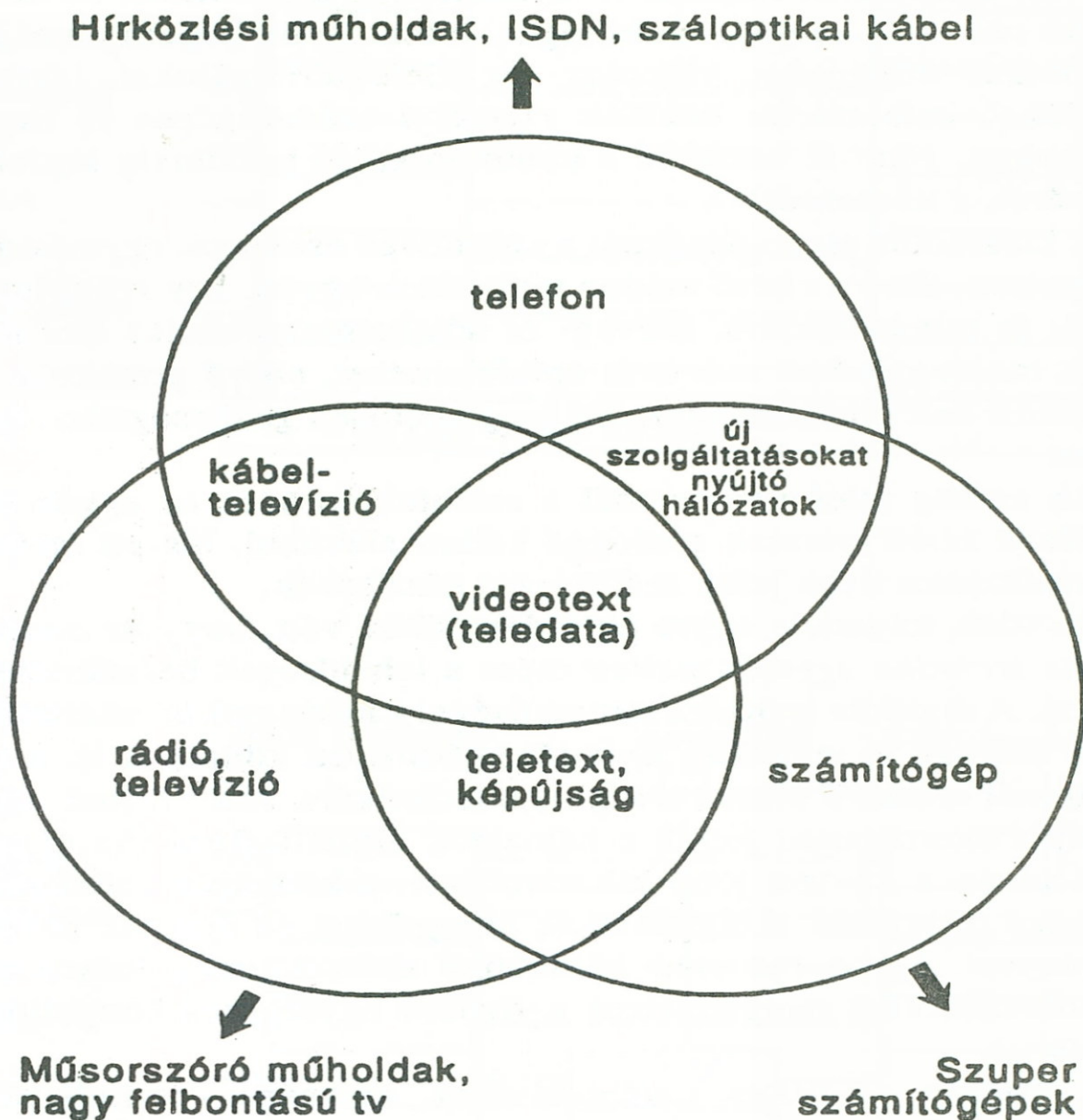
A klasszikus megoldásoknál a különböző eszközök egymástól eltérő hálózatokon, illetve eltérő módon működnek együtt. Így szükségünk van telefon- és telexhálózatra, szöveg- és adatkommunikációs rendszerekre, melyek mindegyikénél más-más építőelemeket, eltérő protokollt és tarifarendszert kell alkalmaznunk. Ez meglehetősen gazdaságtalan és körülményes.

Az analóg telefonhálózatnál a számítógépek és az egyéb digitális jelforrások bitfolyamatait analóggá kellett alakítani, hiszen eredetileg a telefonhálózatot ilyen jelek szállítására készítették.

Az idők folyamán egyre nyilvánvalóbbá vált, hogy *az analóg és a digitális technika egymás mellett élése* a lehetőségek beszűkítését eredményezi. A digitális technika térnyerésével a hálózatok üzemeltetői döntő lépésre szánták el magukat. Belátták, bármilyen költséges is, a telefonrendszerrel célszerű áttérni a digitális technikára. Ma a fejlett országokban teljes intenzitással folyik e hálózatok digitalizálása. Az új technika hosszú távon a hálózat jobb kihasználását, a költségek csökkentését, a különböző hírközlési szolgáltatások integrálását és újabbak bevezetését eredményezi. Így a ma még különböző műszaki megoldásokra épülő hírközlés-technikai szolgáltatások a jövőben egységesen bonyolíthatók le (6-5. ábra).

A beszéd digitális továbbításához szükséges átviteli sebesség 64 kbit/s. Az ilyen sebességű átvitelre alkalmas felhasználói csatornát B (alap-) csatornának nevezzük. Persze a tényleges üzenetek továbbításán túl a zavartalan működéshez egyéb, állapotleíró és -jelző információk átviteléről is gondoskodni kell. Ezek a tudnivalók a jelzőcsatornákon, a D csatornákon áramlanak. A jelzőcsatorna szabványban meghatározott kapacitása 16 vagy 64 kbit/s.

A Nemzetközi Távközlési Unió (az ITU) előírásai szerint egy telefonvonal átviteli kapacitása 144 kbit/s lehet, ami 2B+1D csatornának felel meg. Ezt alap-hozzáférési (Basic Rate Interconnection) csatlakozásnak



6-5. ábra A telefon, a rádió és a televízió, valamint a számítástechnika integrálódása

hívják. Vagyis a digitális technikát használva, egy telefonvonalon két, egymástól független szolgáltatás valósítható meg. Sőt még *16 kbit jelző-információ* is továbbítható másodpercenként. *Ez a szolgáltatások integrálásának, az ISDN kialakításának az alapja.*

Az ISDN-nél tehát a szolgáltatásoknál használt legkisebb átviteli sebesség is 16 kbit/s. Ha ezt összevetjük az átviteli célokra ma használatos értékekkel (az 50, 300, 1200, 2400, 4800 vagy a jelenleg még ugyancsak gyorsnak számító 9600 bit/s-mal), a különbség azonnal szembetűnő. A nagyobb átviteli sebesség az üzenetek gyorsabb továbbítását, illetve időegység alatt több információ átvitelét jelenti. Következésképpen a szolgáltatások minősége ugrásszerűen javulhat, tisztább lesz a telefon hangja, nagyobb a telefax felbontása.

Az ISDN környezetben már a telefon sem a régi. *Számos olyan új telefonos szolgáltatáshoz jutunk, amelyről Bell még álmodni sem mert volna.* Az, hogy a telefonkészülék kijelzőjén látható a dátum, a pontos idő, vagy hogy a készülék akár kalkulátorként is használható, nem rossz. De az ISDN ennél sokkal többet nyújt. A készülék beprogramozható arra, hogy ha nem venném fel a kagylót, jegyezze meg a hívó számát, így, ha akarom, később visszahívhatom. Az ISDN-nél ugyanis mindig tudható, hogy melyik számról hívtak, ami persze védelem a telefonbetyárok ellen is. Ha hivatalomban a szobámat elhagyom, a hívást átirányíthatom abba a másik terembe, ahol éppen tartózkodni fogok. Ehhez nem kell a központ kezelőjének kegyeit kérnem, a parancsot saját telefonomnak kell csak adnom. Ha ezek után kollégám szobájában keresnek, a készülék más hangon fog szólni, ha az eredeti tulajdonost vagy ha vendégét hívják.

Arra is van mód, hogy ha keresnek, de mással beszélek és így a hívónál foglaltat jelez a készülék, saját telefonom – miután az előző beszélgetést befejeztem – automatikusan tárcsázza azt, aki előbb sikertelenül próbálkozott.

Az ISDN telefonnal csak egy gombnyomás a konferenciahívás, vagyis három vagy több különböző helyen lévő készülék összekapcsolása.

Nem szabad megfeledkeznünk arról, hogy az ISDN kapcsolatnál az átviteli sebesség jóval nagyobb a ma használatosnál, amiből két előnyös tulajdonság adódik. A ma használatos felbontás mellett sokkal *gyorsabb egy dokumentum továbbítása*, mint az analóg telefonvonal esetén, illetve a *nagyobb felbontású átvitel* sem nyúlik bosszantóan hosszúra, *rövidebb időtartamú.*

Az ISDN környezetben a *telefax is új lehetőségekkel gazdagodik*. Hiszen például a képátvitel a beszélgetéssel párhuzamosan folyhat.

A CCITT a telefaxkészülékeket – átviteli sebességük és felbontási adottságaik alapján – csoportokba sorolja. A 2. csoportba tartozó készülékekkel egy A/4-es oldal továbbítása körülbelül 3 percet, a 3. csoportba sorolt eszközzel körülbelül 1 percet vesz igénybe. Az ISDN átviteli csatornán ugyanezen dokumentum továbbítása 5 másodpercen belül megoldható.

Nem elhanyagolható a lehetséges *minőségjavulás* sem. Hiszen míg a hagyományos telefaxnál a jellegzetes felbontási érték 200 képelem/hüvelyk (vagyis 8 képelem/mm), addig az ISDN-célokra fejlesztett berendezéseknél ez megduplázható, tehát 16 képelem/mm-re növelhető.

A szolgáltatás kényelme mellett az ISDN használata *takarékos*. Ennek oka részben az, hogy nagyobb az átviteli sebesség, másrészt hogy olcsóbbak a vonalbérlési díjak, nem szólva arról, hogy ugyanazon az érpáron legalább dupla annyi információ továbbítható, mint a hagyományos esetben.

A készüléken megjeleníthető a beszélgetés aktuális költsége, és – mielőtt kiürülne a zsebünk – kérhetjük, hogy a hívott vállalja magára a hívó költségét.

Van, aki az ISDN-t korainak, öncélú műszaki fejlesztésnek tartja, aminek legfőbb haszonélvezői a gyártók lesznek. Mások szerint már most is késésben vagyunk, a felhasználói kívánságok alapján öt évvel ezelőtt sem lett volna koraszülött az ISDN. *Európa nyugati országainak többsége az integrálás pártján áll*. A Közös Piac országainak legtöbbszörében már működnek kísérleti hálózatok, sőt van, ahol napjainkra már teljes jogot nyert az ISDN a szolgáltatások palettáján.

Kétféle fejlesztési stratégia figyelhető meg világszerte. Az egyik az igen intenzív átmenet a hagyományosból a legfejlettebb ISDN-implementációig, a másik a fokozatos fejlesztés, a kis lépések elve. Az első a leggazdagabb és megfelelő háttérparral rendelkező országok jellemzője. A másik koncepciót az óvatosabbak, illetve a tőkehiánnyal küzködők követik. Számunkra is ezt az utat javasolják.

Mint az informatika terén annyiszor, a franciák az ISDN esetében is példát mutatnak. A CCITT előírásait figyelembe véve, tíz évvel ezelőtt kezdték meg a telefonközpontok és a hálózat digitalizálását. A munkák ütemezésénél prioritási sorrendet állítottak fel: a helyi csoportvonalak korszerűsítését tették az első helyre, majd a helyközi vonalak, a helyközi központok és végül a helyi központok következnek. Bár hazánkban pilla-

natnyilag alapvető telefonítség van, a Posta illetékesei már a nyolcvanas évek utolsó harmadában is tudták, hogy a hazai átviteli szolgáltatások fejlesztésénél nálunk is kell az ISDN-nel számolni.

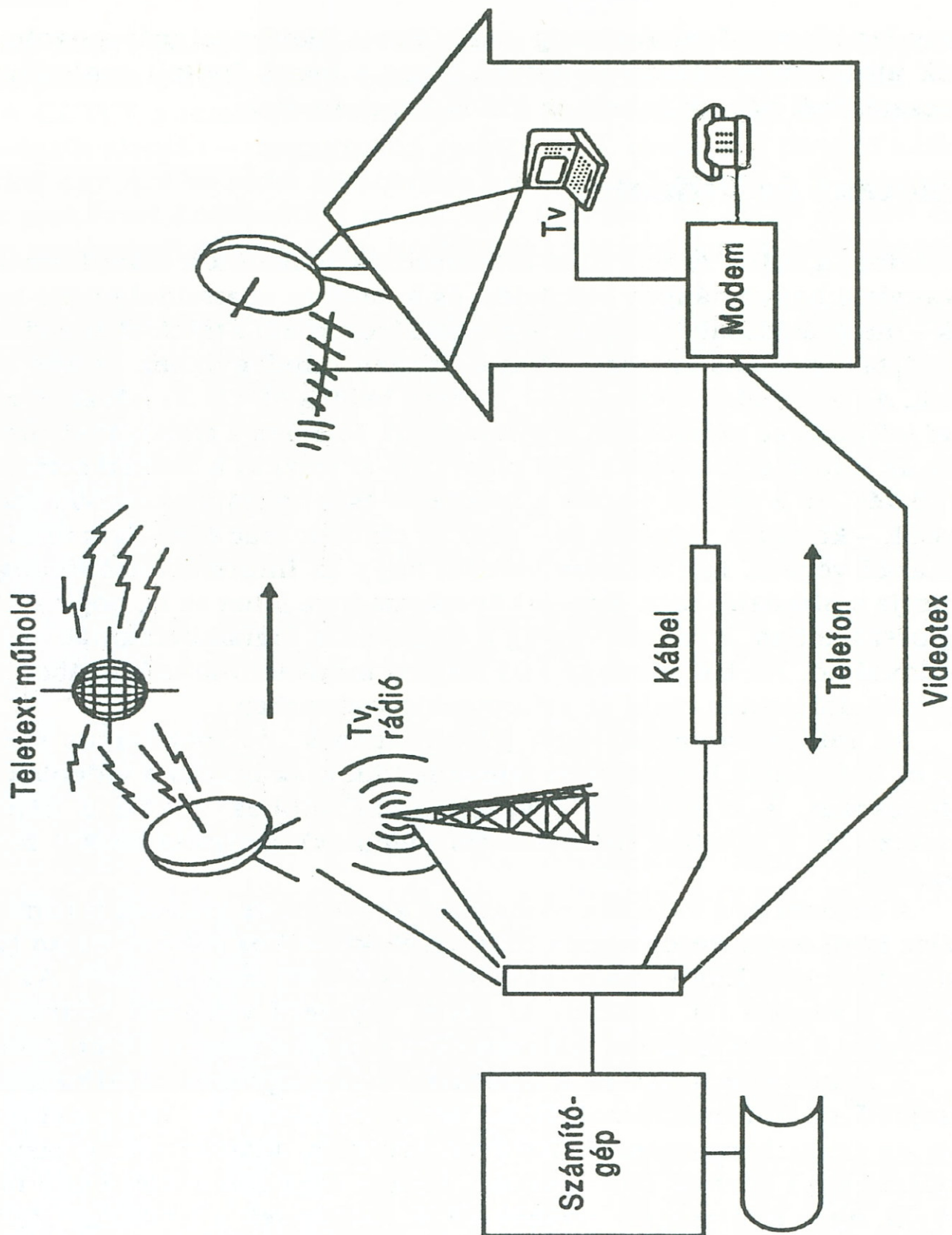
Teletext és videotex

A telefon, a rádió- és televízió-műsorszórás, valamint a számítástechnika – vagyis a hagyományos hírközlési és a modern adatfeldolgozási technikák – integrálódásából születő új megoldások közül a teletext és a videotex (teledata) speciális szerepet játszik, hiszen mindegyikhez szoros szálak fűzik. Az *információtechnológia*, és ezen belül a *hírközlés növekvő gazdasági jelentősége* számokban is kifejezhető. Jelenleg a Közös Piac országában az információtechnológiai piac 15%-át teszi ki a szűken értelmezett hírközlés; de a század végére a hírközlés és a hozzá kapcsolódó szolgáltatások – köztük a videotex is – akár az említett piac 60%-át is uralhatják a század végére. Így biztosra vehető, hogy az információtechnológia fő motorja a hírközlés lesz. Ennek következménye lehet az is, hogy a közönség országában, a század végéig e területen a foglalkoztatottak száma 4 millióval nő. Az EK mintegy 105 milliós munkaerő-állományából 30-60 millióan dolgoznak majd az információs szektorban.

Az információtechnológia jelentőségének, perspektivikus szerepének és hatásainak ismeretében értékelhetjük csak igazán a videotex szolgáltatásokat. A *telematika gyűjtőfogalom*, többek között a képűtség (teletext) és a teledata (párbeszédés videotex) szolgáltatásokat foglalja magában.

A teletext és a videotex rendszerek szöveges és grafikus információk széles körű terjesztését végzik teljesen elektronikus módon, olcsó (gyakran megfelelően kibővített tv-vevőből kialakított) megjelenítőn. Az információt a fogadó fél választja ki olyan egyszerű vezérlési eljárásokkal, amelyeket a szakképzetlen felhasználó is könnyen megért és megtanulhat.

A számítástechnika és a híradástechnika együttes alkalmazásaként létrejövő új információszolgáltató technológiában az igazi újdonságot nem az egyes komponensek (hardver, szoftver, adatbázisok, elszámolási rendszer stb.) jelentik önmagukban, hanem ezek *integrálása egy tömegméretű, nagy információs rendszerré*. E rendszerek a lehetséges felhasználók körét nagyon kiszélesítik, egyrészt az egyszerűen kezelhető eszközök és parancsok, másrészt az általánosan elterjedt információelosztó rendszerek, például a telefonhálózat vagy a műsorszóró, illetve kábeles tévéhálózat használatának köszönhetően (6-6. ábra).



6-6. ábra Videotext/teletext adatátviteli megoldások

A *műsorszóró teletext*nél a szöveges és grafikus információt, adatjeleket digitális formában a videoműsorjel képváltások alatt kihasználatlan tartományaiba keverik, és a hagyományos műsorjellel együtt sugározzák ki. Az eredetileg a hallássérültek számára feliratozott műsor-sugárzáshoz készült megoldás messze túlnőtt a kiindulási célon. A megadott rendszer szerint ismétlődően kisugárzott, számozott (oldalazonosítóval ellátott) oldalakból a felhasználó tetszése szerint választhat, a kiválasztott oldal a tévékészüléken jelenik meg. E megoldásnál a digitális adatjelet a tévéjelbe ültetve továbbítják. A teletext szolgáltatás egyirányú, mert a felhasználó nem küldhet üzenetet az információellátást végző központi számítógépnek; parancsokat csak saját tévékészülékének adhat.

A műsorszóró teletext rendszerek igen olcsón nyújtják szolgáltatásaikat, kicsi a központi beruházási igény, a vevőkészülék kialakítása sem túl drága. A jelek továbbítása a már meglévő tévéadókkal, a kihasználatlan jeltartományban történik. Ezzel szemben korlátozott az információs oldalak száma, és a felhasználónak a kívánt oldal fogadásánál a hosszabb várakozási idővel kell fizetnie az átvitel olcsóságáért.

A *keskeny sávú párbeszédes rendszereknél* a telefonhálózatot használják a jelátvitel céljára, vagyis a felhasználó a kívánt információs oldal fogadására vonatkozó igényét a telefonvonalon keresztül közli a központi számítógéppel, és ezen az úton érkezik a felhasználó megjelenítőjéhez a várt válasz is. Az információáramlás sebessége a két irányban általában különböző: a számítógéptől a felhasználóig 1200 bit/s, a felhasználótól a számítógépig 75 bit/s a ma használatos tipikus sebesség.

Az egyik információs oldalról a másikra csak a felhasználó és a rendszer közötti kétirányú kommunikáció árán léphetünk, ezért ezeket a rendszereket párbeszédes videotex (interactive videotex) rendszereknek nevezzük. A videotex rendszer *főbb funkcionális egységei*:

– Központi számítógép vagy ezek hálózata, amelynek a feladata az informatikai rendszerben az információellátás irányítása és az adatbankokból a szükséges adatok nyújtása.

– Adatátviteli rendszer (átviteli csatorna, illetve hálózat), amelynek a feladata az információk átvitele a felhasználó és a központi számítógép (adatbank) között.

– Felhasználói terminál (például módosított televízióvevő vagy személyi számítógép), amely az információk fogadására, megjelenítésére és küldésére szolgál.

A keskeny sávú párbeszédéses videotex rendszereknél a kezelhető oldalak száma elvileg tetszőleges (már ma is több százezer), a felhasználó mindig csak az őt érdeklőt kapja. Nincs közvetlen kapcsolat az adatátviteli idő és az adatbázis mérete (az elérhető oldalak száma) között, az átviteli idő lényegében az csak az átviteli rendszertől függ, általában néhány szekundum. Ezzel szemben e szolgáltatás költségesebb, mert a tévékészülék módosításán kívül fizetni kell a telefonhálózat használatáért és az információért is. A díjak tipikusan átalánydíjtól és a vonal igénybevételenek tényleges idejétől függnnek, az információs oldalak árát az oldalon fel is tüntetik. Nem elhanyagolható, hogy a központi számítógép itt nagyobb és költségesebb, mint a teletext esetében. Nemcsak azért, mert nagyobb adatbázist kell kezelnie, de azért is, mert gondoskodnia kell a sok, egy időben jelentkező felhasználói kívánság megfelelő kezeléséről is.

A teletext és a videotex rendszereket napjainkban elsősorban információszolgáltatási, információkeresési célokra használják. A rendszerek kialakításának célja az alkalmazók tömegének információellátása volt, ideértve a kis- és középvállalkozókat, de mindezeket túl a háztartásokat, a felhasználók legszélesebb tömegeit.

A videotex szolgáltatások lehetnek nyilvánosak, szólhatnak a felhasználók egyes zárt csoportjaihoz (closed user group í CUG) és lehetnek házi rendszerek.

A nyilvános videotex szolgáltatásra jellemző a Prestel vagy a Teletel rendszer, amelyet magánszemélyek és kisebb-nagyobb intézmények egyaránt használnak. Az információs oldalakért fizetni kell. Adott esetben a felhasználó figyelmeztetést is kaphat az eddig elköltött összegről. A szolgáltatás igénybevételenél nemcsak az oldalak után (tehát magáért az információért), hanem az átviteli hálózat használatáért is kell fizetni (ezért jó, ha a videotexközpont helyi hívással elérhető), sőt a felhasználók "előfizetői átalánydíjat" is fizetnek.

Az információkeresést különböző írott segédletek (felhasználói kézikönyvek) teszik egyszerűbbé és gyorsabbá, ezek használata pénzmegtakarítással is jár. A nyilvános videotex rendszereknél használnak pénzbedobással működő terminálokat is, amelyeket általában adott információs oldalak elérésére programoznak, és mint olcsó informátorállomásokot lehet használni (pl. tájékoztatás színházműsorokról, közlekedési utakról, idegenforgalmi nevezetességekről).

A zárt felhasználói csoportok az érdeklődési kör, érdekezésség vagy egyéb szempont szerint csoportba tömörült felhasználókat jelentik,

akiknek kizárólagos joguk van bizonyos adatok, adatbázisok használatára. Ilyen csoportot alkotnak például a bankok, az utazási irodák, a katalógus-áruházak, az autókereskedők, a kiadók, a könyvkereskedők stb. A zárt csoportba tartozás történhet kötött tagdíj alapján, vagy az igénybevétel arányában történő fizetés szerint. A speciális adatbázisok alkothatják a csoportok tulajdonát, vagy lehetnek külső tulajdonos birtokában is.

A *házi rendszerek* egyes intézmények belső információs igényeit elégítik ki. Ezek lehetnek a szó szoros értelmében véve házon belüli rendszerek, de összekapcsolhatnak egymástól távol lévő telephelyeket is, például nagy nemzetközi vállalkozások esetén.

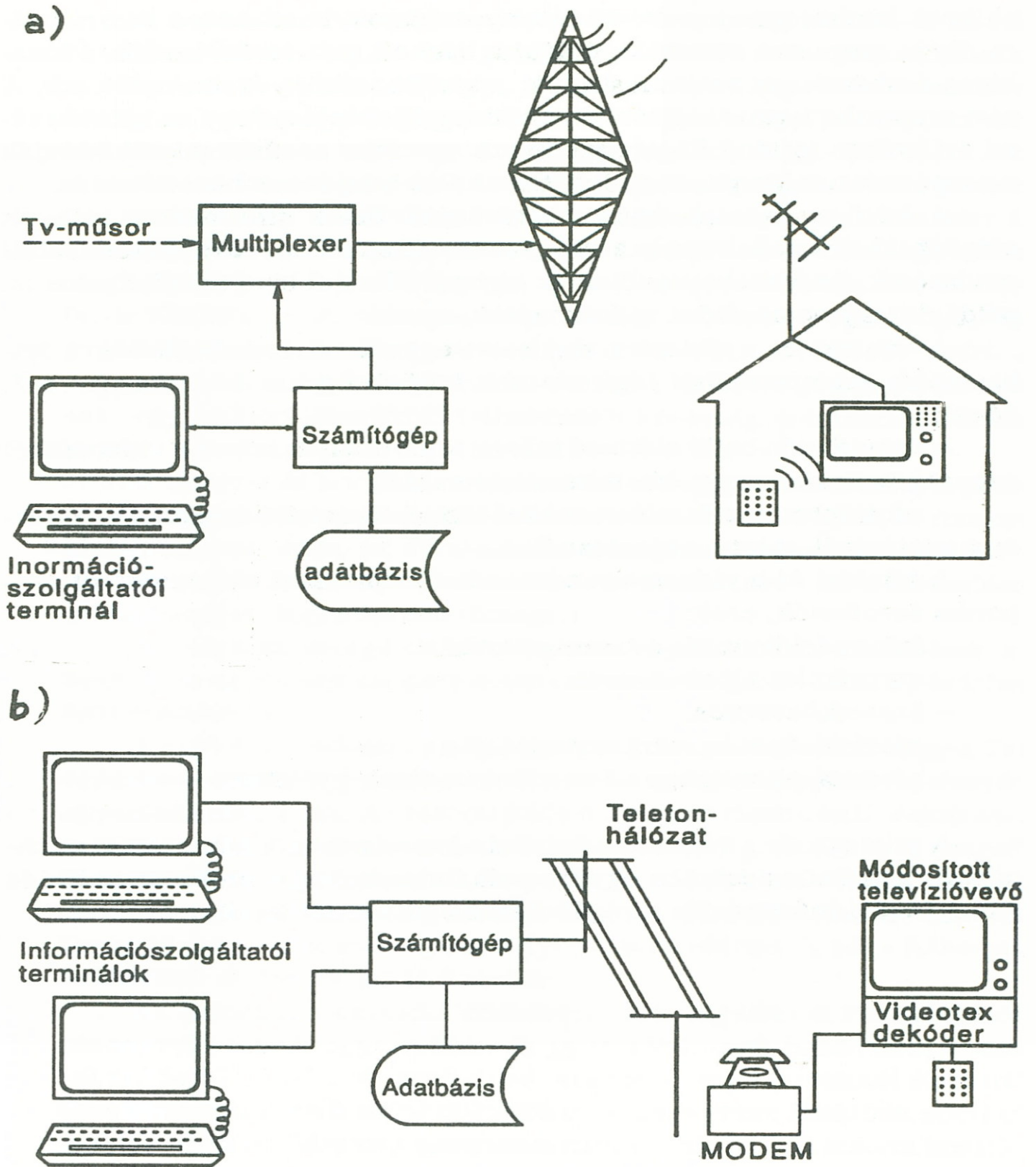
A teletext és a videotex rendszert egymással szembeállítva, a felhasználás szempontjából legfontosabb különbségek a következők (6-7. ábra):

- a rövid időn belül elérhető információs oldalak száma a videotexnél nagyságrendekkel nagyobb, mint a teletextnél;
- adott információs oldalt sokkal több felhasználó fogadhat egyszerre a teletextnél, mint a videotexnél.

A teletext és a videotex rendszerek alkalmazásai alapvetően öt csoportba sorolhatók, ezek:

- információkeresés és -szolgáltatás,
- tranzakciós alkalmazások,
- üzenetközvetítés,
- adatfeldolgozás, szoftverterjesztés,
- háztartások megfigyelése, mérési adatok gyűjtése.

A teletext- és a videotex- (teledata-) rendszerek tehát a híradástechnika és a számítástechnika együttes alkalmazásaival a professzionális és a magáncélú információs szolgáltatások egész sorát teszik lehetővé.



6-7. ábra

INSTALL

Informatikai Stúdió

Pécs, Felsővámház u.72.(DOZSO)

Telefon: (72) 13-648, 24-855

Telefax: (72) 14-600

ÚJ SZAKKÉPESÍTÉST SZEREZHET számítástechnikai tanfolyamainkon.

SZEMÉLYI SZÁMÍTÓGÉP KEZELŐ

alapfokú,

SZEMÉLYI SZÁMÍTÓGÉP SZOFTVERÜZEMELTETŐ

középfokú

SZÁMÍTÓGÉP PROGRAMOZÓ

felsőfokú

INFORMÁCIÓRENDSZER SZERVEZŐ

felsőfokú

szakmai végzettséget adunk.

Nálunk egyetemi és főiskolai oktatók tanítanak,
akik nemcsak számítástechnikai szakemberek, hanem gyakorlott pedagógusok is.
Biztos tudást és sikerélményt nyújtunk Önnek.

Ne halogassa a döntést, biztosítsa jövőjét!

Még ma jelentkezhet az

INSTALL Informatikai Stúdió

Információs szolgálatánál telefonon vagy személyesen.

Rendszeres egyéb tanfolyamaink:

40 órás PC operátor,

80 órás szoftver alkalmazói.

Speciális és céltanfolyamok igény szerint
VÁLLALATOKNAK IS.

Kérje prospektusunkat!

AZ



SZÁMÍTÓGÉPE MINDENT TUD,
AMIT A SZÁMÍTÁSTECHNIKÁBAN
TUDNI KELL!

MINDENKINEK ELÉRHETŐ ÁRON!

A SIKER TITKA



SZÁMÍTÓGÉP!

Eurodata Kft. Budapest,
1138 Váci út 163.

Telefon: 149-7447, 129-7869

Telefax: 149-7447

7. Betekintés a számítógépek ipari alkalmazásába

A gépi intelligencia

Ez a fejezet a számítástechnika ipari alkalmazásának néhány – a szerző által fontosnak ítélt – részletével kívánja megismertetni az olvasót. A témakör ismertetése során nagyszámú szakmai kifejezést, új fogalmat magyaráz meg.

Az ember alkalmazkodó képességét az új vagy egyedülálló helyzetekhez, intelligenciának nevezzük. Ennek során az embernek alkalmaznia kell tudni a már megszerzett ismereteit, hogy befolyásolhassa a környezetét. Ez a folyamat szükségessé teheti új tudás megszerzését, vagy a múltban kialakított tapasztalatok újraértékelését. Természetesen az ember nap mint nap alkalmazkodik, és ez a képessége nagyon egyszerűnek látszik. A gépnek azonban az ember számára egyszerű feladatok is nehezek vagy lehetetlenek.

Az intelligens gép egy sor különféle részegységből áll, amelyek megadott feladatokat látnak el. Ismerkedjünk meg a gép döntéshozó és kommunikációs rendszerével, néhány általános jellegű példa áttekintésével!

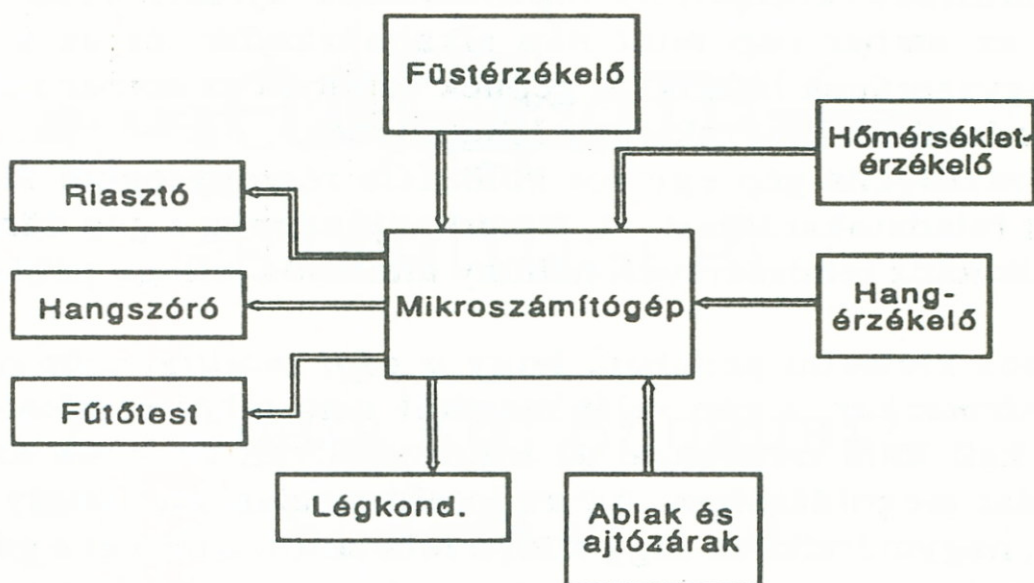
Fontos kiemelni azonban, hogy *a gépi intelligencia csak az embertől származhat*, a gép saját magától nem tehet szert a tudásra. A gépet el kell látni mindazzal az információval, amelyek szükségesek egy feladat megoldásához. Az az intelligenciaszint, amely egy gépet jellemez, nagymértékben függ attól a feladattól, amelyet a géppel végre kell hajtani.

Intelligens gépek

Azokat a gépeket nevezzük intelligens gépeknek, amelyek független tevékenység végzésére programozhatók. Vagyis ezek a gépek programozhatók *a környezeti kölcsönhatások figyelembevételére*, és a környezetüktől nyert információ alapján döntéshozatalra képesek.

Jó példa erre a lakásbiztosításra alkalmazott mikroszámítógép és a hozzá csatlakozó berendezések. Ebben az esetben a mikroszámítógép kölcsönhatásban van egy sor eszközzel, és az ezekből érkező jelektől függően bizonyos akciókat hajt végre.

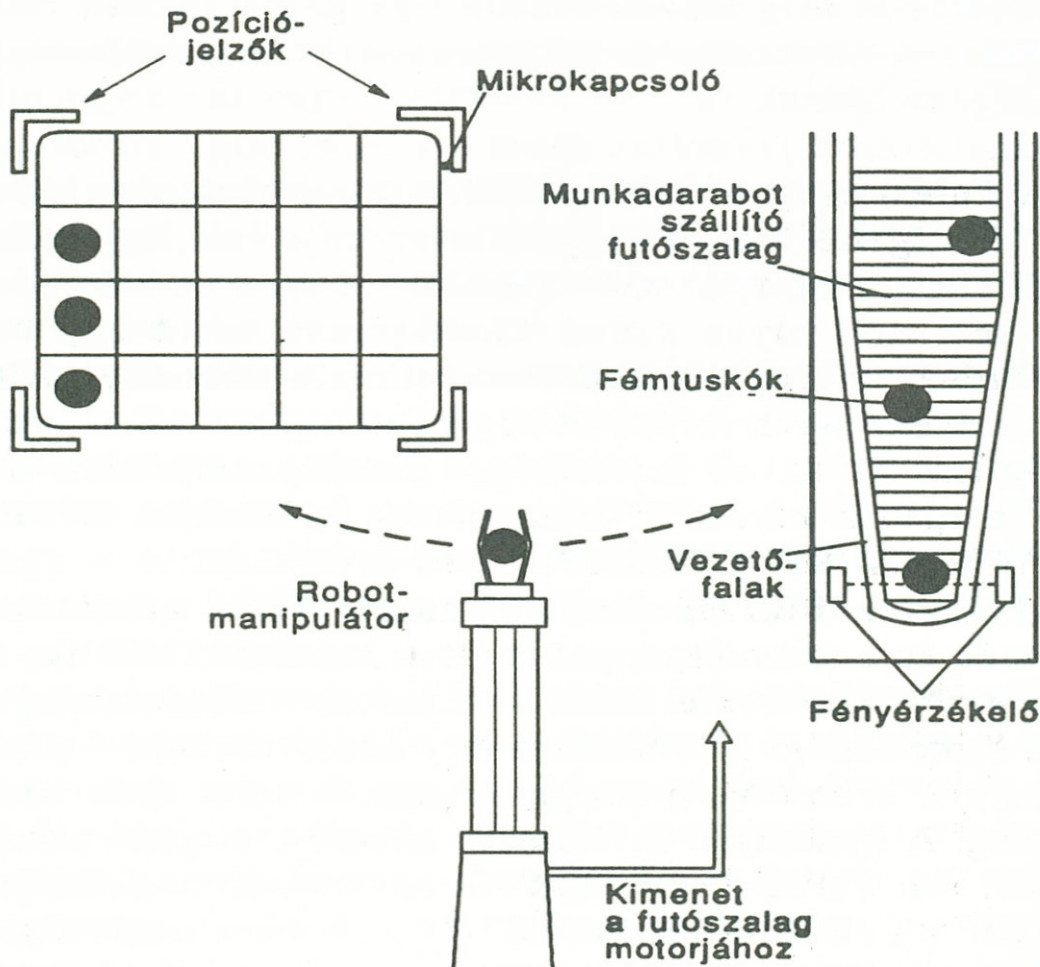
A 7-1. ábra egy ilyen rendszer blokkdiagrammját mutatja be. Ebben a rendszerben a mikroszámítógép több berendezéssel is kapcsolatban van. Bemeneti adatokat kap a füstérzékelőtől, a hőmérsékletérzékelőtől, a hangérzékelőtől, az ajtó- és ablakérintkezőktől. A bemeneti jelektől függően a mikroszámítógép fel tudja hívni a tűzoltóságot vagy a rendőrséget, hangjelzést tud adni a lakásban, a megfelelő hőmérséklet elérésére ki és be tudja kapcsolni a fűtőberendezést vagy a légkondicionálót. Szükség esetén az érzékelő-berendezéseket alap helyzetbe tudja állítani a mikroszámítógép azért, hogy meggyőződjön arról, hogy a jelzett állapot létezik-e. A mikroszámítógép ezeket a feladatokat emberi beavatkozás nélkül látja el, csak a rendszer indításához és a karbantartáshoz kell külső beavatkozás.



7-1. ábra Lakásbiztosító számítógép

Egy másik intelligens gépet mutat be a 7-2. ábra. Itt a robot fémgolyókat vesz le a szállítószalagról, és azokat kartondobozba rakja. Az optikai érzékelő a robot szemeként működik, amelyből jövő jel "mondja

meg" a robotnak, hogy mikor van munkadarab a szállítószalag végén. A robot megfogja a munkadarabot és elhelyezi a kartondobozban.



7-2. ábra Robotmunkahely

A dobozrögzítők egyikében elhelyezett mikrokapcsoló jelzi a robotnak, hogy a doboz a helyén van. Ha a doboz nincs a helyén, a robot megállítja a szállítószalagot, és a munkadarabok elhelyezését sem kísérli meg.

Nyilvánvalóan ez a feladat nem látszik nagyon bonyolultnak, de a gép számára ez nem is olyan egyszerű. A robot ebben az esetben kapcsolatban van a környezetével, döntéseket hoz, módosítja a saját működését a rendszer más elemeitől kapott információk alapján. Ez valóban intelligencia, bár annak egy alacsony foka. A következőkben

olyan intelligens gépeket fogunk megismerni, amelyek magasabb intelligenciafokon dolgoznak.

A mesterséges intelligencia fogalmának meghatározását a szakértők többféleképpen értelmezik.

Néhányuk szerint már egy sakkozógép is mesterséges intelligenciával rendelkezik, míg mások szerint egy gép – hacsak nem tudja a saját mozgását befolyásolni és programozni – nem tekinthető mesterséges intelligenciájúnak.

A nézeteltérések javarészt abból erednek, hogy miből is áll tulajdonképpen a mesterséges intelligencia. A bizonytalanság oka az, hogy az emberi intelligenciának is nagyon sok szintje van. Egy széket eltolni egy másik helyre, bizonyos intelligenciát igényel, de ehhez a művelethez közel sem kell olyan szintű intelligencia, mint egy belsőégésű motor tervezéséhez. Egy gép akkor rendelkezik mesterséges intelligenciával, ha bármely olyan feladatot végre tud hajtani, amely az embertől is intelligenciát kíván. A mesterséges intelligencia ezen definíciója alapján a feladatok három fő kategóriába oszthatók; INTERAKTÍV INFORMÁCIÓCSERE, ÉRZÉKELEÉS és MOZGÁS.

Az interaktív információcsere kategóriájába tartoznak a szakértői rendszerek és a számítógéppel segített tanulás (CAI) is.

A *szakértői rendszer* utánozza az ember problémamegoldó tevékenységét. Ennek megvalósításához egy adott témakör – pl. a belgyógyászat vagy a számítógép programozása – több száz szakemberét kérdezik meg. A gyakorlati kivitelezés ismerete alapján a számítógép-programozó, akit TUDÁSSZAKÉRTŐnek hívnak, megalkotja az ismeretbázist ezekből az információkból. Ez az ismeretbázis algoritmusba foglalt feladatmegoldó módszereket tartalmaz, amely lehetővé teszi, hogy a rendszer "megoldjon" különböző problémákat.

Ilyen a MYCIN nevű program is, amelyet a vér különböző bakteriológiai fertőzéseinek diagnosztizálására fejlesztettek ki. Ez a program 500 heurisztikus szabály segítségével állapítja meg a fertőzést, és gyógyeljárást is javasol. A program működése közben interaktív kapcsolatban van a kezelővel. A szakértői rendszer megadott kérdéseket tesz fel a tünetekről, jelenségekről, és további vizsgálatokat vagy megadott gyógyeljárást javasol. Továbbá a rendszer megmondja a kezelőnek, miért tette az adott javaslatokat, és hivatkozik egy adott szabályra.

A számítógéppel segített tanulórendszerek hasonlóan a szakértői rendszerekhez abban, hogy alapvető szinten lehetővé teszik a tanulóknak, hogy kérdéseket tegyenek fel, és erre válaszokat kapjanak. A

számítógéppel segített tanulórendszer azonban több egy "kérdézz-fel-elek" játéknál, mert valós idejű interaktív rendszer, ami azonnali visszacsatolással rendelkezik, és olyan kérdéseket állít elő, amelyek a tanuló egyéniségéhez szabottak. Ez úgy történik, hogy a rendszer megvizsgálja a tanulónak a korábbi kérdésekre adott válaszait. A tanuló és a rendszer természetes nyelven kommunikál egymással. Röviden az ideális számítógéppel segített tanulórendszer pontosan ugyanazt csinálja, mint egy tanár vagy instruktör, aki egyénileg foglalkozik a diákkal.

Lehetséges kezdetleges tapintó-, képfelismerő- és hangfelismerő rendszereket imitálni aránylag egyszerű érzékelőkkel, amelyek mikroszámítógéppel vannak összekapcsolva. Azonban egy igazi érzékelőrendszert rendkívül nehéz utánozni.

Gondoljunk például az egy-egy másodpercre megjelenő képek problémájára! Bármilyen képen adatbitek ezrei vannak a formára, a színre, a távolságra, az elhelyezkedésre, a méretre vonatkozóan.

Az ember nagyon könnyen fel tud ismerni például egy labdát. Az emberi agy a szem által szolgáltatott képet összehasonlítja azzal a képpel, amelyet a labdáról az agy tárolt. Az összehasonlítás gyakorlatilag egy pillanat műve.

Most gondoljuk meg: mit kell tennie egy intelligens gépnek ahhoz, hogy meg tudja oldani ezt a feladatot! Először is rögzítenie kell a képet. A kép rögzítését ma már nagyon jó minőségű érzékelők is könnyen elvégzik. Ezután a gépnek össze kell hasonlítani ezt a képet a memóriájában tárolt képpel. Ez meglehetősen bonyolult dolog, mivel gyakorlatilag a színnek, az árnyéknak, a formának, a távolságnak, a beállításnak és az aktuális méretnek számtalan kombinációja lehetséges, amelyet tárolni kellene a gép memóriájában az összehasonlításhoz. Vannak módszerek arra, hogy a képet leegyszerűsítsék, azért, hogy a lehetséges kombinációk száma csökkenjen. Azonban egy valódi mesterséges intelligenciájú rendszer nem vesz igénybe ilyen típusú külső segítséget.

Hasonló problémák jelentkeznek mind a képfelismerésnél, mind a tapintórendszereknél. A tapintóérezékelőknél szükséges folyamatos visszacsatolás miatt a tapintóérezékelés valószínűleg a legnehezebben valósítható meg.

Az intelligens gépek mozgása két típusba sorolható: a pozícionálás és a követés.

A pozícionálás adott kezdő- és végpont között nem meghatározott. A robotikában ezt PONTVEZÉRLÉSnek hívják.

A követés gyakorlatilag a pozícionálás továbbfejlesztése. Követésnél a mozgást egy pontsorozat vezérli a kívánt mozgáspályán. Ezek a pontok egymástól kicsi, de meghatározott távolságra vannak. A követést nevezik Vezérlésnek is.

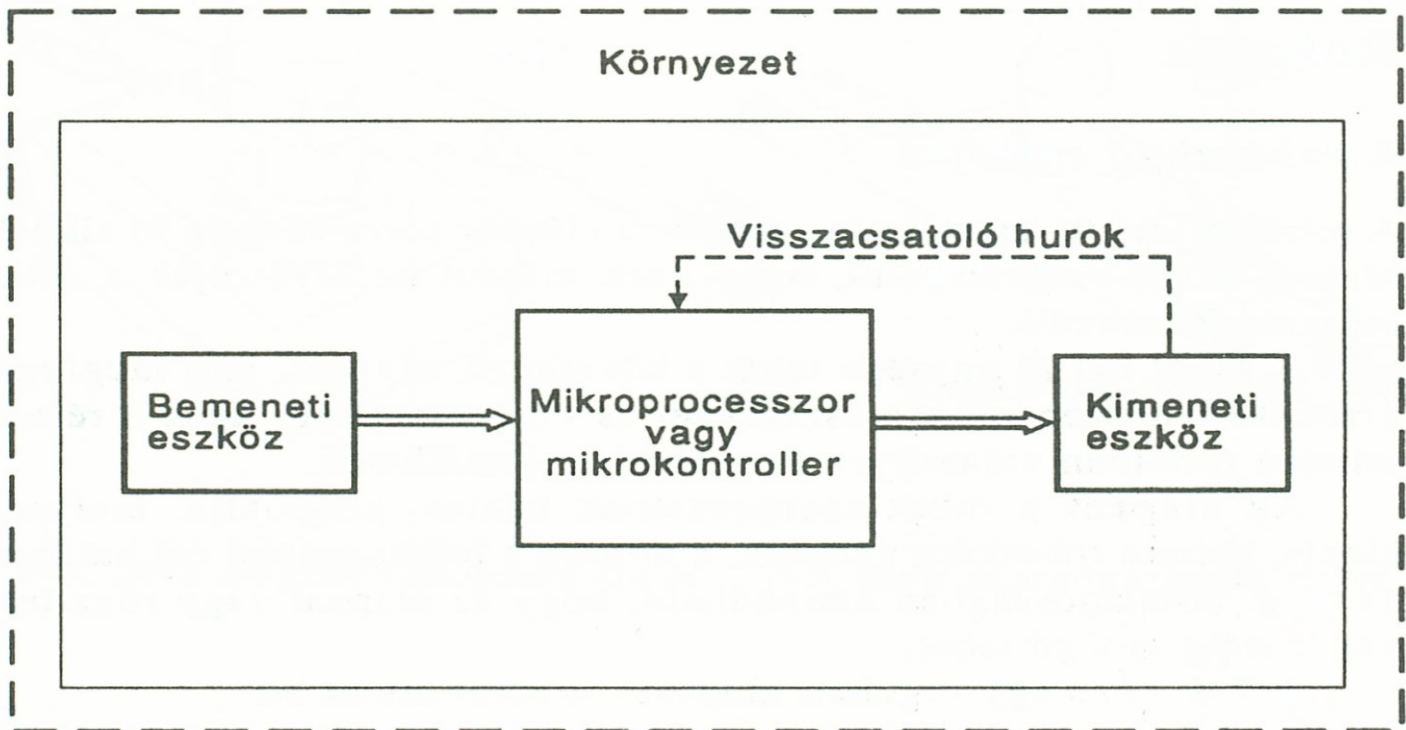
Bármelyik esetet véve, az a gép, amely ilyen feladatok végrehajtására van programozva, nem tekinthető mesterséges intelligenciával rendelkező gépnek. Sokkal inkább azok a gépek, amelyek ki tudnak kerülni akadályokat, a legrövidebb úton haladnak, és biztonságosan működnek. Az intelligens gépnek rendelkeznie kell a környezettel kapcsolatot tartó érzékelőkkel. Az érzékelőkből nyert információkat a gépnek tudnia kell értelmezni, és ha szükséges, meg kell változtatnia a haladási irányát.

Az érzékelőkből nyert információ összetettsége kapcsolatban van azzal a ténnyel, hogy az intelligens gép mozog, és az érzékelőkről ezért folyamatos információt vár a feladat végrehajtásához. Az információk feldolgozását biztosító program összetettsége és a szükséges memória mennyisége – amely a bejövő adatok tárolására és módosításához szükséges – jelentős problémát jelent a mesterséges intelligenciával rendelkező gépek szabályozott mozgásrendszerére vonatkozóan. Ennek részletezésével nem foglalkozunk.

A jellegzetes intelligens gép blokk-sémája a 7-3. ábrán látható. Ezen az ábrán a szaggatott vonalon belüli rész jelképezi a környezetet. KÖRNYEZETnek nevezünk minden olyan külső dolgot, amely az intelligens gép működésére hatással van. Így minden gép környezete más és más. Például a szobavilágítás mértéke nem befolyásolja a számítógéppel segített tanulás programját, de annál intenzívebb hatást gyakorol egy képfelismerő rendszerrel ellátott robotra. Valójában a környezetnek különleges értelme van, amikor robotokkal vagy egyéb intelligens mozgó géppel foglalkozunk.

Az intelligens gép legegyszerűbb formájában tartalmaz egy BEMENETI EGYSÉGet, egy KIMENETI EGYSÉGet és egy MIKROPROCESSZORT vagy VEZÉRLŐt. Az egységek egy BUSZon csatlakoznak egymáshoz, amely bizonyos számú párhuzamos vezetőből áll. Minden kommunikáció az intelligens gép részegységei között a buszokon történik. Az információt a buszon bináris formában visszük át. Ez azt jelenti, hogy az információ binárisan, egyesekre és nullákra van kódolva. Az egyes jelzi a "be" állapotot a nulla jelzi a "ki" állapotot.

A BEMENETI EGYSÉG adatokat vagy információkat gyűjt, és ezeket olyan módon alakítja át, hogy az intelligens gép fel tudja használni. A BEMENETI JEL elektromos formában tartalmazza az információt vagy adatot és ezek a bemeneti egységből a számítógépbe lesznek továbbítva. Nagyon sokféle bonyolult bemeneti berendezést használunk, kezdve egy egyszerű kapcsolótól, ami csak 'be'/'ki' állapotot tudja jelezni, egészen egy összetett képérzékelő rendszerig.



7-3. ábra Intelligens gép

A KIMENETI JEL tartalmazza azt az utasítást vagy információt, amit a mikroprocesszor küld a kimeneti egységnek. A kimeneti egység a mikroprocesszortól kapott információkat felhasználja bizonyos feladatok ellátására, vagy az intelligens gép által feldolgozott adatokat fordítja át valamilyen használható formára. A kimeneti egységek sora a csengőtől egészen a beszéd szintetizáló berendezésig tart.

A 7-3. ábrán látható egy visszacsatolás is, amely a kimeneti berendezésről a mikroprocesszorra csatol vissza. A VISSZACSATOLÁS része a gép kimenetének, és önkorrekciós célra szolgál. A kimeneti jelet a számítógép folyamatosan összehasonlítja a bementettel, egészen addig amíg a kívánt állapotot a rendszer el nem éri. A visszacsatolásnak

két formája van. A POZITÍV VISSZACSATOLÁS felerősíti a gép kimeneti jelét. A NEGATÍV VISSZACSATOLÁS arra szolgál, hogy a kimenőjelet csökkentse.

Az intelligens gép talán legfontosabb része a *mikroprocesszor* (CPU), vagy másnéven szabályozó. Ez az intelligens gép 'agya'. Ez az a hely, ahol az adatokat módosítja az intelligens gép, és itt születnek a döntések is.

Robotok

A robotok fő egységei

A robotok tanulmányozásakor érdemes először sorra venni a fő alkotórészeit, majd megvizsgálni, hogy ezek miként befolyásolják a robot műszaki jellemzőit.

A robot hat fő egysége tehát a következő: alapzat, kar, megfogó, érzékelő rendszer, mozgatószerkezet és vezérlőegység. Ezek a részek minden robotban valamilyen formában megtalálhatók.

Az alapzat a robot szerkezetének bázisa, központja, melynek alakja, mérete robotként változik, s mindig a felhasználási cél határozza meg. Általánosságban elmondható, hogy az alapzat vagy rögzített, vagy pedig mozgó lehet.

A 7-4. ábra egy rögzített alapzatú robotot mutat be.

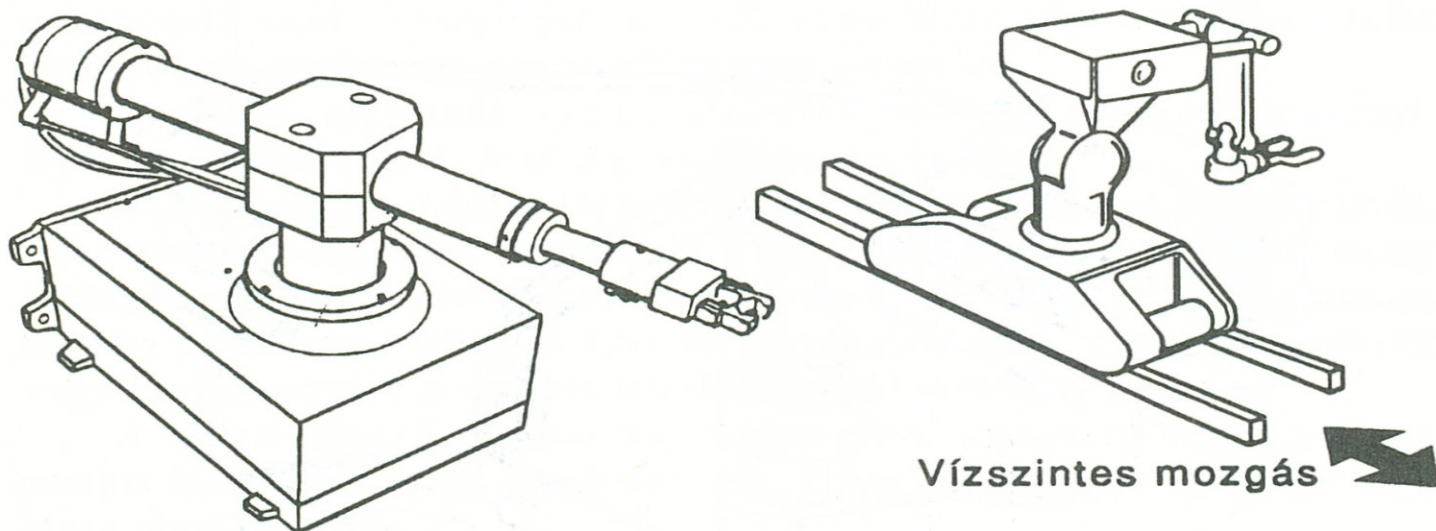
A mozgóalappal kifejezés nem feltétlenül jelenti azt, hogy a robot minden irányba képes elmozdulni. A 7-5. ábrán látható eszköz – a vonathoz hasonlóan – egy sín páron vízszintes irányú mozgást tud végezni. Ebben az esetben az alapzatban kap helyet a mozgatószerkezet is.

A robotkar ízületekkel (csuklókkal, csuszakákkal) összekapcsolt szegmensekből áll, mely az alapzathoz kapcsolódik. A feladata a megfogó szerkezet mozgatása az adott térrészben.

A szegmensek merev részek, melyek ízületekkel kapcsolódnak egymáshoz. Ezek az ízületek teszik lehetővé a szegmensek adott irányú, vagy adott tengely körüli mozgását. Az alapzathoz a vállon keresztül kapcsolódó szegmenst kezdőszegmensnek nevezzük. Azt a karrészt viszont, amelyhez a csukló által a megfogó szerkezet kapcsolódik, végső szegmensnek hívjuk.

A megfogó, amit nevezhetnénk kéznek vagy szerszámnak is, a robot azon része, amely a feladatot elvégzi. A robotkarhoz a csuklón

keresztül kapcsolódik. Napjainkban már igen sokféle megfogó létezik. Egyeseket speciális feladatra terveznek, mint például a festékszóró vagy a hegesztőpisztoly, míg mások általános célúak.



7-4. ábra Rögzített alapzatú robot 7-5. ábra Mozgóalapotú robot

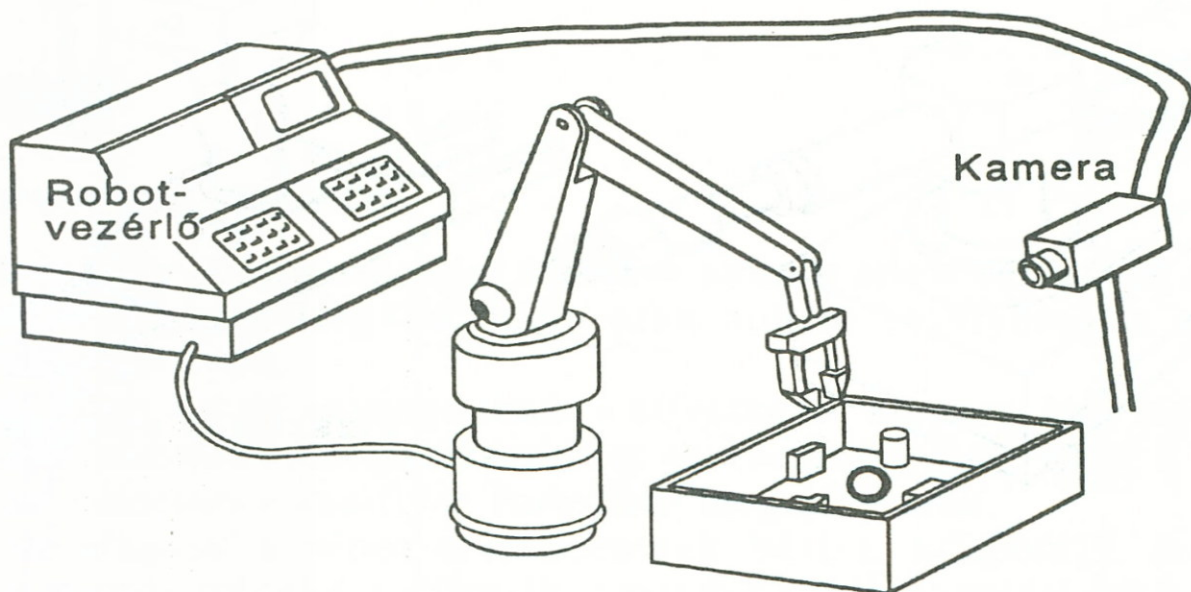
Érzékelőrendszerek

Minden intelligens gépnek szüksége van érzékelőrendszerre. A robotoknak valós idejű kapcsolatot kell fenntartaniuk a környezetükkel, ehhez viszont kifinomult látó- vagy tapintórendszerekre van szükség. Ettől függetlenül azért igen sokszor alkalmaznak bizonyos feladatokhoz kevésbé finom érzékelőket, mint például kapcsolót vagy egyszerű fényérzékelőt.

Az érzékelőrendszer lehet a robot szerves alkotórésze, de gyakori az is, hogy elkülönül tőle, és csak információt küld a robotnak.

A robotok látórendszerét minden esetben ismert és rendezett környezetre tervezik. A számítógépes ipari látórendszerek két nagy csoportra oszthatók, az *alkatrész meghatározás* és az *alkatrész felügyelet* elvén dolgozó rendszerekre. Az *alkatrész meghatározása* nehezebb, mivel látórendszerrel magasabb intelligenciát követel meg.

Például figyelje meg a robot alkatrészfelismerési és osztályozási tevékenységét, amit a 7-6. ábra mutat be. Itt a robotnak a látórendszer segítségével meg kell határoznia a véletlenszerűen elhelyezett tárgyak helyzetét és rá kell raknia azokat egy szállítószalagra. Ez a feladat az ember számára egyszerűen hangzik, de a robot számára bonyolult feladat, amihez a mesterséges intelligencia magas foka szükséges.



7-6. ábra Ipari látórendszer

A robotnak meg kell vizsgálnia a tárgyakat, kiválasztani a szükséges alkatrészt a tárolt képek alapján, meg kell határozni az alkatrészek elhelyezését. Ezek után utasítania kell a megfogót, hogy megfelelően nyúljon az alkatrészhez. A legösszetettebb eljárás az alkatrész azonosítása különböző megvilágítás és elhelyezés esetén. A jelenlegi látórendszerek még csak a fejlődés kezdeti szakaszán vannak.

A robot irányításának tökéletesítése a másik nagy terület, ahol még sok fejlődésre van szükség. Fontos, hogy a látórendszer szabályozni tudja a robot mozgását. A számítógépes látórendszerek fejlődésével a robotok helyettesíteni fogják az embert számtalan veszélyes munkahelyen.

A számítógépes látórendszerek másik nagy területe a gyártmányellenőrzés. Ennek az az oka, hogy sok ellenőrző tevékenységnél 100 százalékos biztonságra kell törekedni. Ez azt jelenti, hogy minden alkatrészt vizuálisan ellenőrizni kell. Az ember nem alkalmas erre a

feladatra, mivel figyelme nem állandó. A számítógépes látórendszerek biztosítják a 100 %-os pontosságot, amely a nagytömegű gyártás megbízhatóságát, így gazdaságosságát eredményezi. (Ilyen berendezéseket alkalmaznak például a nyomtatott áramkörök gyártásánál.)

Igen gyakori, hogy a robotnak fognia kell a tárgyakat. Ez a feladat a különböző *tapintóérzékelők* segítségével valósítható meg. A legegyszerűbb ilyen érzékelő a mechanikus kapcsoló. Ismert azonban, hogy a kapcsoló csak a tárgy jelenlétéről vagy hiányáról tud információt nyújtani, de ez bizony sok esetben nem elegendő.

Ismerünk még más, igen elterjedten használt érzékelőrendszert. Ilyenek a közelség, a távolság, és a hangérzékelők.

A *közelségérzékelő* a tárgy érintése nélkül jelzi annak jelenlétét. Ezek az eszközök tehát nem mérik a tárgy távolságát, csak jelzik, hogy a tárgy egy adott távolságon belül van-e. Ez a távolság maximum néhány centiméter lehet. A közelségérzékelők fény, hang, mágnesség vagy elektrosztatikus vonzás detektálásával működhetnek.

A *távolságmérő* rendszerek segítségével viszont a robot a tárgy pontos távolságát tudja meghatározni. Ilyen eszköz a hanglokátor és a lézeryinterferométer.

A hanglokátor kiküld egy impulzust, ami az érzékelendő tárgyról visszaverődik. Az impulzus terjedési sebességének valamint a kibocsátás és a visszaverődés közötti idő ismeretében a mikroprocesszor ki tudja számítani a tárgy távolságát.

A lézeryinterferométer sokkal pontosabb eszköz. Segítségével nanométernél kisebb távolságok is mérhetők. Ez az érzékelő kibocsátott lézersugár segítségével méri a távolságot, így azonban a környezeti hatások – például füst, köd, por és eső – kedvezőtlenül befolyásolják a működését.

A *hangérzékelők* – mint például a mikrofon – igen gyakoriak a robot érzékelőrendszerében. Igen sokféle méretük és érzékenységük miatt képtelenség lenne felsorolni az összeset.

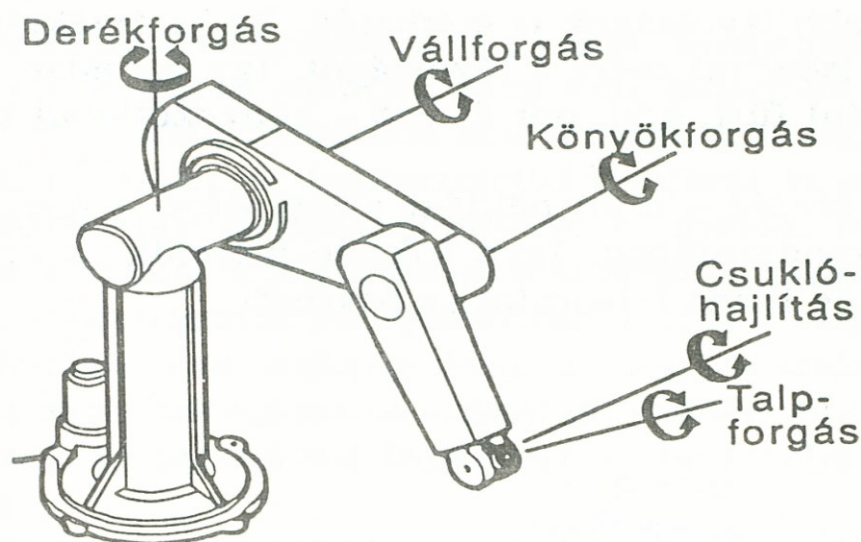
Robotjellemzők

A robotok működésének a leírására több robotjellemzőt használunk, melyek meghatározzák a robot minőségét.

A robot feladata a végrehajtóelem mozgatása a tér előírt pontjába. Ehhez a robotkar minden szegmensének megfelelő mozgatása szükséges. A végrehajtóelem térbeli elhelyezkedését és irányítottóságát 6 szabadsági fokkal tudjuk leírni. Az egyes szabadsági fokok a helyzet és irányítottóság egy-egy komponense. Az első három szabadsági fok az x , y , z , melyek egy térbeli pont koordinátái. A másik három szabadsági fok az a , b , c . Ezek a végrehajtó dőlését, a hajlását és a forgását jelentik, más néven a tér egy pontjának koordinátáihoz viszonyított irányítottóságát. A mi tárgyalásunkban tehát a szabadsági fokok térbeli helyzetet és irányítottóságot jelentenek.

A robotok szabadsági foka a kar és az alapzat tagoltságára utal. Az egyes szabadsági fokok tehát egy-egy mozgást, tengelyt jelentenek. A szabadsági fok meghatározza, hogy a robot milyen mozgásokra képes. Általánosan igaz, hogy minél több szabadsági foka van egy robotnak, annál többre képes.

A 7-7. ábrán egy öt szabadságfokú robot látható. Ez az öt szabadsági fok azt jelenti, hogy a robot öt különböző irányba tud mozogni.



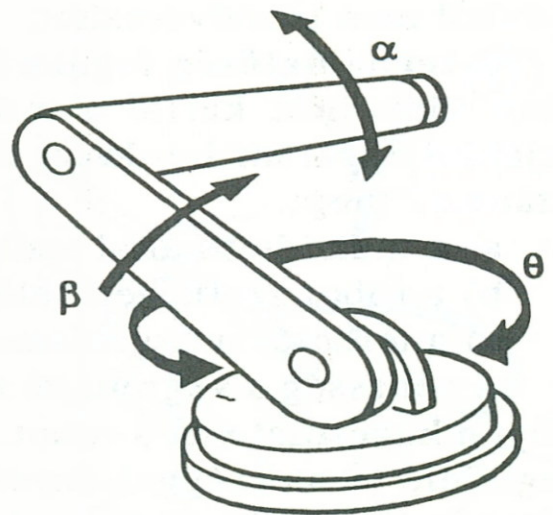
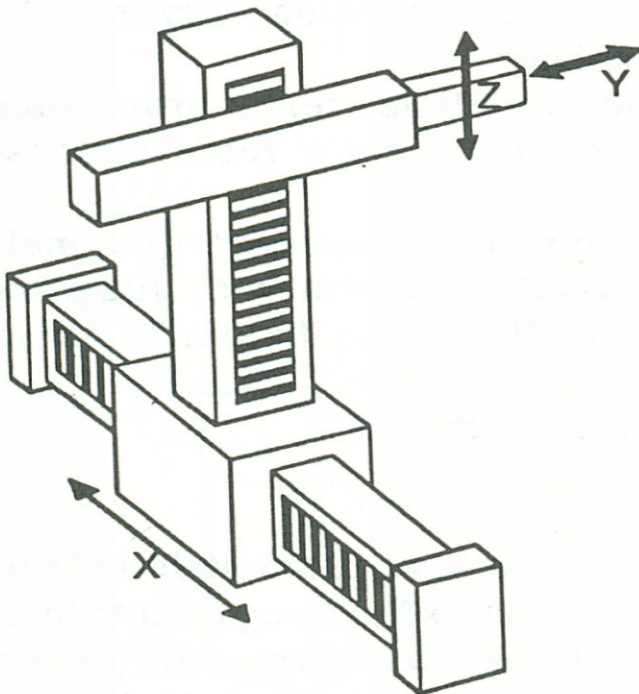
7-7. ábra 5 szabadsági fokkal rendelkező robot

A robotok egyik leggyakoribb osztályozási módszere, hogy a mozgások alapján soroljuk őket csoportokba.

A robotokat a mozgástengelyek és a tengelyeknél végzett *mozgástípusok* alapján csoportosíthatjuk. Két mozgástípust különböztetünk meg: a lineárist és a forgót.

A lineáris, más néven prizmatikus (Descartes koordináta tengelyek mentén történő), mozgást a 7-8. ábra mutatja. A lineáris mozgás során a karszegmens egyenes úton mozog.

A forgómozgás esetén a szegmens egy adott pont (tengely) körül mozog (7-9. ábra).



7-8. ábra Lineáris (prizmatikus) mozgás 7-9. ábra Forgómozgás

A mozgástér vagy munkaterület a háromdimenziós tér azon részét jelöli, melyben a robot a maximális terheléssel a megadott pontossággal

értéken belül tud működni. A terület alakja a szabadsági fokok számától, a kar szegmenseinek hosszától és az egyes tengelyek mozgástípusától függ.

Négy *alapvető robotkonfiguráció* van, amely megszabja a működési terület alakját: a Descartes-koordináták, a hengerkoordináták, a polárkoordináták és a forgáskoordináták.

A Descartes-koordináták szerinti munkaterület a 7-10/A. ábrán látható. A munkaterület 3 dimenziós, hasábalakú. Ahogy a neve is mutatja, a hengerkoordinátás rendszer működési területe egy henger (7-10/B. ábra), míg a polárkoordináták egy felül domború hengert határoznak meg (7-10/C. ábra). Végül a forgáskoordináták szerint mozgó rendszer munkaterülete egy alul levágott gömb, amelyet a 7-10/D. ábra szemléltet.

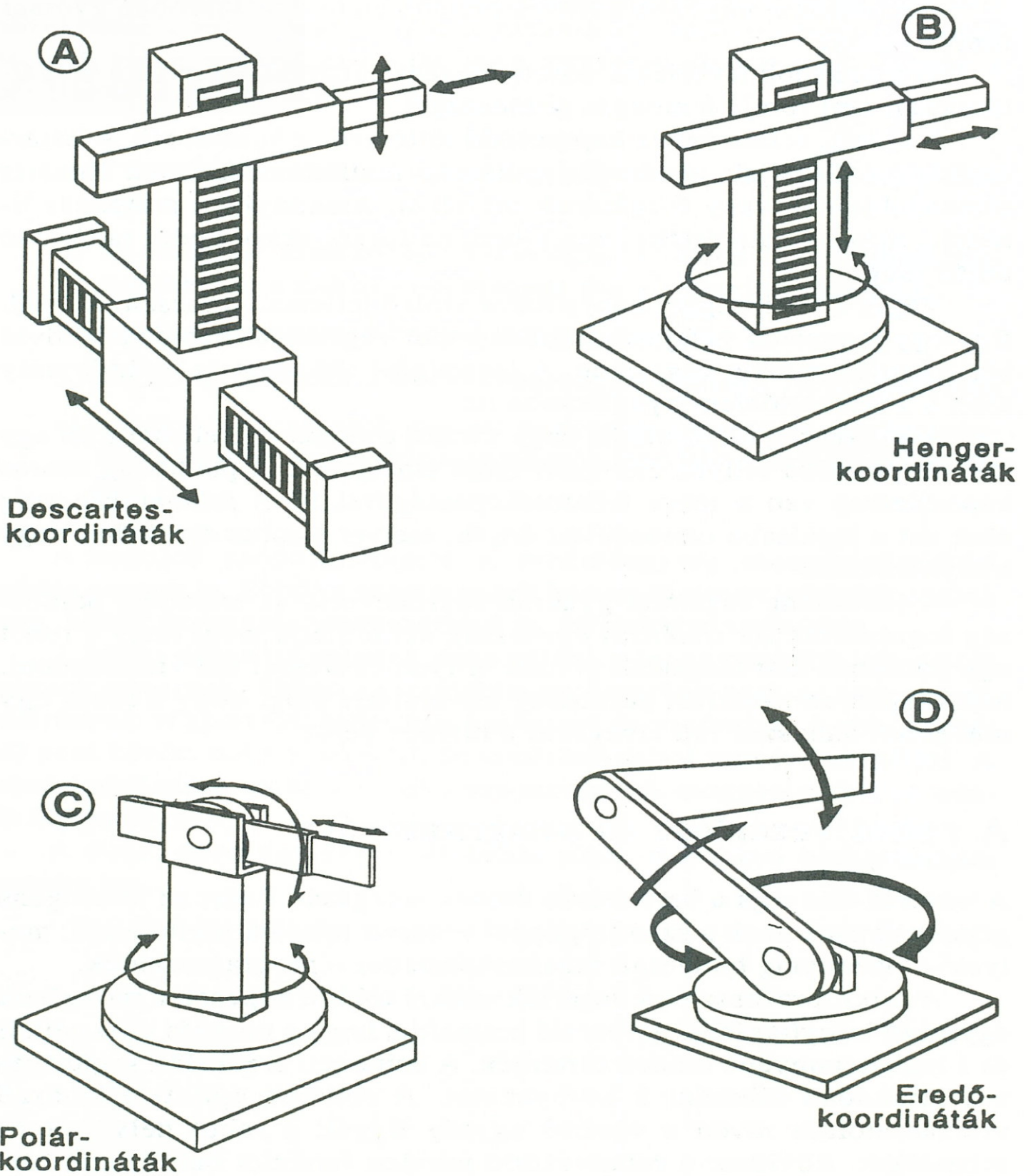
A robotkar felépítését természetesen mindig az adott feladat szabja meg. A Descartes-koordináták alapján mozgó robot például nagyon jól használható két tárgy közötti mozgásra, vagy könnyen bejárható területen munkadarabok szállítására. Amennyiben a robotnak valamilyen tárgy felett kell átnyúlnia, akkor forgás-koordináták szerinti konfiguráció alkalmazható a legelőnyösebben.

A munkahelyeken, a robot működési területén rendszerint szükség van *tiltott zónákra* is. Ezek olyan területek, ahol a robot biztonsági okokból nem manőverezhet.

A *terhelhetőség* fogalmán azt a maximális tömeget értjük, melyet robot kinyújtott karral a működési terület vízszintes és függőleges síkjában egyaránt hordozni tud. A terhelhetőséget három fő tényező határozza meg:

- a) a működtető által szolgáltatott teljesítmény,
- b) a robot szerkezeti felépítése,
- c) a terhelés mozgatásának sebessége.

A sebesség a végrehajtó mozgásának gyorsaságára utaló jellemző. Milyen kapcsolat van a robot maximális sebessége és egy adott feladat megfelelő pontossággal történő elvégzése között? Erre utaló jellemző a *ciklusidő*. A ciklusidő, mely egy adott feladat elvégzésének ideje, a sebességen kívül függ a robot egyéb jellemzőitől és az elvégzendő feladattól is. Gondoljunk csak arra, hogy az ívhegesztés varratának elkészítése sokkal több időt vesz igénybe, mint egy ponthegesztés!



7-10. ábra Robotkonfigurációk

A terheléstől két okból függ a ciklusidő:

a) a robotkar nagyobb terhelés esetén valamivel lassabban gyorsul illetve lassul;

b) a nagyobb terhelések tehetetlensége túllendülést okoz a megállásoknál, ami rontja a mozgás pontosságát.

A másik, sebességhez kapcsolódó jellemző, a *határérték*. A határértékkel definiáljuk maximális sebesség mellett az ízületek lineáris elmozdulásának vagy forgásának mértékét. Amennyiben a mozgás lineáris, akkor a határértéket mm/s-ban, ha forgó, akkor pedig rad/s-ban adják meg.

Végül a sebességgel kapcsolatos utolsó jellemző a *lecsengési idő*. Egy-egy mozzanat befejezése után a robot végrehajtóeleme még rövid ideig tovább mozog vagy leng. A lecsengési idő tehát az az idő, mely alatt a végrehajtóelem nyugalomba jut.

A *pontosság* megmutatja, hogy a robot a végrehajtó elemet a tér egy kijelölt pontjába milyen eltéréssel tudja elhelyezni. A pontosság szoros kapcsolatban van a robot felbontóképességével is. A *felbontóképesség* alatt azt a legkisebb elmozdulást értjük, melyet a robot érzékelni, vagy irányítani tud.

A pontosság fogalmát gyakran összekeverik az ismétlési pontosság fogalmával. Az *ismétlési pontosság* azt mutatja meg, hogy a robot egy korábban már megtalált pontba milyen eltéréssel tud visszamenni. Másképpen az ismétlési pontosság azt mutatja meg, hogy a robot egy műveletet hányszor tud elvégezni a tűrésen belül.

A robot vezérlése és programozása

A vezérlő irányítja a berendezés összes mozgását. Ezért az intelligens gépeknek csak azok a számítógéppel vezérelt robotok tekinthetők, melyek – viszonylag bonyolult utasításkészlettel – programozhatók.

A mikroprocesszoros vezérlők sokkal többre képesek a mozgások egyszerű irányításánál. A vezérlő biztosítja, hogy a mozgás hiba nélkül és a leghatékonyabb módon történjék. A bemeneti érzékelők segítségével állandóan ellenőrzi a környezetet. A robotba épített ellenőrző visszacsatolása révén a vezérlő egység figyeli a robot helyzetét és sebességét. Röviden: a robotvezérlő minden funkciót betölt, ami bármilyen kijelölt feladat elvégzéséhez szükséges.

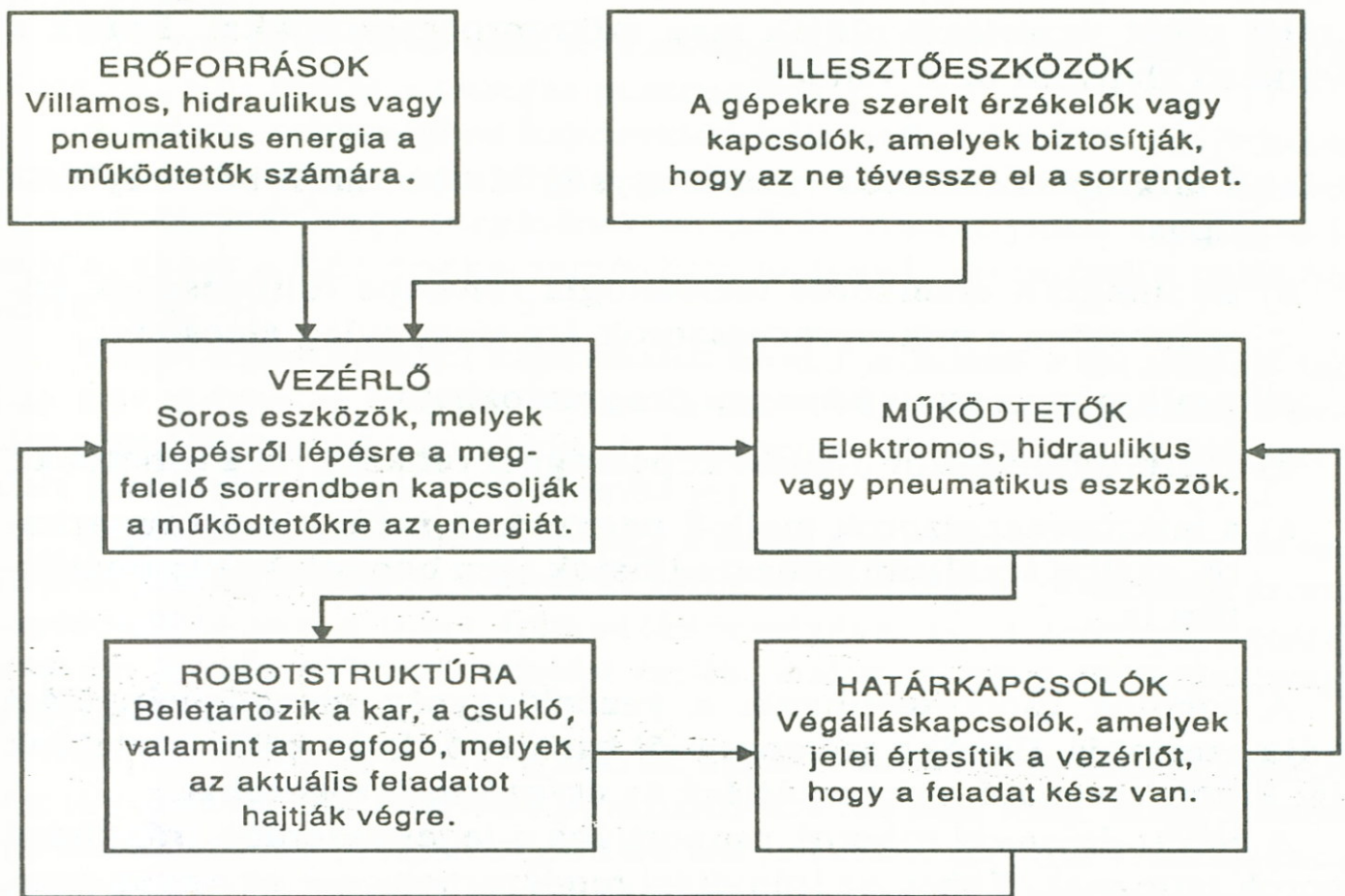
Ahogy nőnek a robotokkal szemben támasztott igények, úgy kell növekednie a vezérlőegység kapacitásának is. Napjainkban egyre több és több robot vezérlését oldják meg mikroprocesszorokkal. Ennek a következő alapvető okai vannak:

- a) a mikroprocesszoros vezérlőegység kisebb, mint bármely más típus;
- b) az integrált áramkörös technológia rohamos fejlődésének köszönhetően a mikroprocesszorok ára viszonylag alacsony;
- c) a mikroprocesszor könnyen programozható;
- d) a mikroprocesszor döntési képességgel ruházta fel a robotokat;
- e) a mikroprocesszorok mellett nagyszámú memória alkalmazható, ezáltal ideálisan felhasználhatók igen bonyolult folyamatokban is.

A robotok csoportosíthatók a vezérlőegység mozgásirányítási módja szerint is. Ebből a szempontból három fő típust különböztetünk meg: kötött útvonalú, pontvezérlésű és útvezérlésű robotokat.

A *kötött útvonalú* robotok csoportjába a legegyszerűbb irányítású robotok tartoznak. Ebben az irányítási rendszerben csak az egyes mozgásirányok végpontjait lehet meghatározni és vezérelni. A robot tehát két pont között tud közlekedni, de útközben sehol sem tud megállni. A feladat elvégzése során a robot ilyen mozgások meghatározott sorozatát hajtja végre.

A 7-11. ábra egy kötött útvonalú robot működési blokkvázlatát mutatja be.

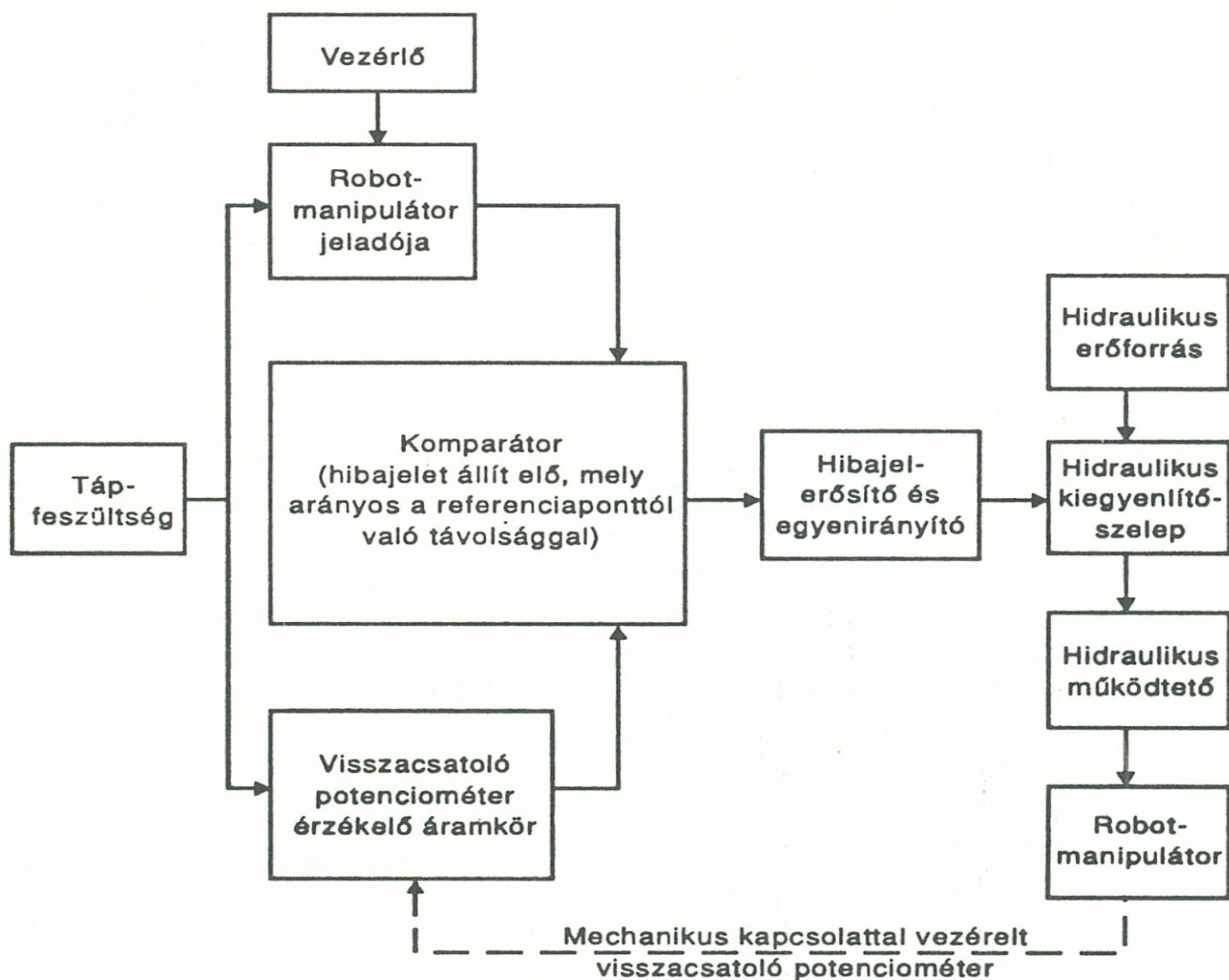


7-11. ábra Kötött útvonalú robot blokkvázlata

A pontvezérlés során az út egymást követő pontjainak koordinátáit elraktározzák a robot memóriájában. A robot pontról-pontra halad, de az egyes pontok elérésének útját nem lehet előre megjósolni. A robot minden egyes szegmense mozog egészen addig, amíg a végrehajtóelem el nem éri a kívánt pozíciót.

A 7-12. ábra egy pontvezérlésű robotmanipulátor működését szemlélteti.

Az útvezérlésű robotok esetében a végrehajtóelem mozgásának útvonalát folyamatosan lehet vezérelni. Az útvezérlés tehát a pontvezérlés továbbfejlesztése. Az útvezérlést akkor használják, amikor speciális úton kell mozgatni a végrehajtót. Igen jó ipari példa a festő és az ívhegesztő robot vezérlése, az útvezérlés segítségével ugyanis bármilyen alak követhető. A rendszer hátránya, hogy nagy memóriakapacitást igényel.



7-12. ábra Pontvezérlésű robot blokkvázlata

Igen sokfajta robotprogramozási mód ismert. Ezek két nagy csoportra oszthatók: az ON-LINE és az OFF-LINE rendszerekre.

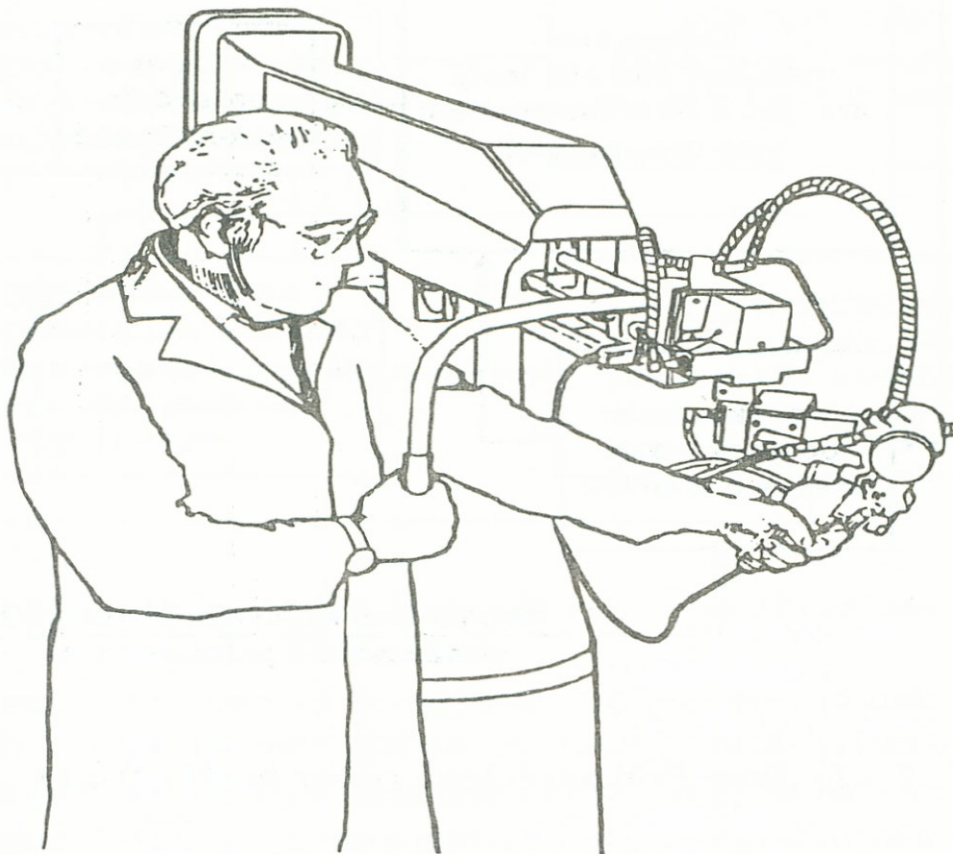
Az ON-LINE programozás során magát a robotot programozzák. A rendszer előnye, hogy a programozó számításba tudja venni a munkaterületen elhelyezkedő tárgyakat, és azonnal ellenőrizni tudja a működést. A hátránya különösen ipari szempontból jelentős: a programozás idejére a robotot le kell állítani, tehát ezalatt nem termel.

Az ON-LINE programozás módjai:

a) a kézi programozás, mely leginkább a kötött útvonalú robotokra jellemző;

b) az átvezető programozás (tanítás mozgatható kezelőpulttal), amelynek lényege, hogy a robotot a kezelő vezérlőkonzol vagy kézi vezérlőkészülék segítségével tanítja meg a szükséges mozgásokra;

c) az átsétáló programozási mód (tanítás kézenfogva), amelynél a programozó kézzel végigviszi a robotmanipulátort az elvégzendő feladat útvonalán (lásd 7-13. ábra), és ennek során a mozgás paramétereit a robot vezérlőegysége önállóan rögzíti.



7-13 ábra Tanítás "kézenfogva"

A programozás másik nagy kategóriája az OFF-LINE módszer. Ennek során a programozó egy számítógép termináljánál, a robottól függetlenül fejleszti ki a működést irányító programot. Ezt követően a programot rögzítik a robot memóriájában.

A robot elmozdításához valamilyen mozgatószerkezetre van szükség. A mozgatórendszer minden robotban három fő egységből tevődik össze. Nevezetesen: erő- vagy energiaforrás, a forrásenergiát mechanikaivá alakító egység és a mechanikai energiát továbbító egység.

Az intelligens gépek erőforrásai pneumatikus, hidraulikus vagy elektromos elven működhetnek.

Az intelligens gépek és a robotok alkalmazásai

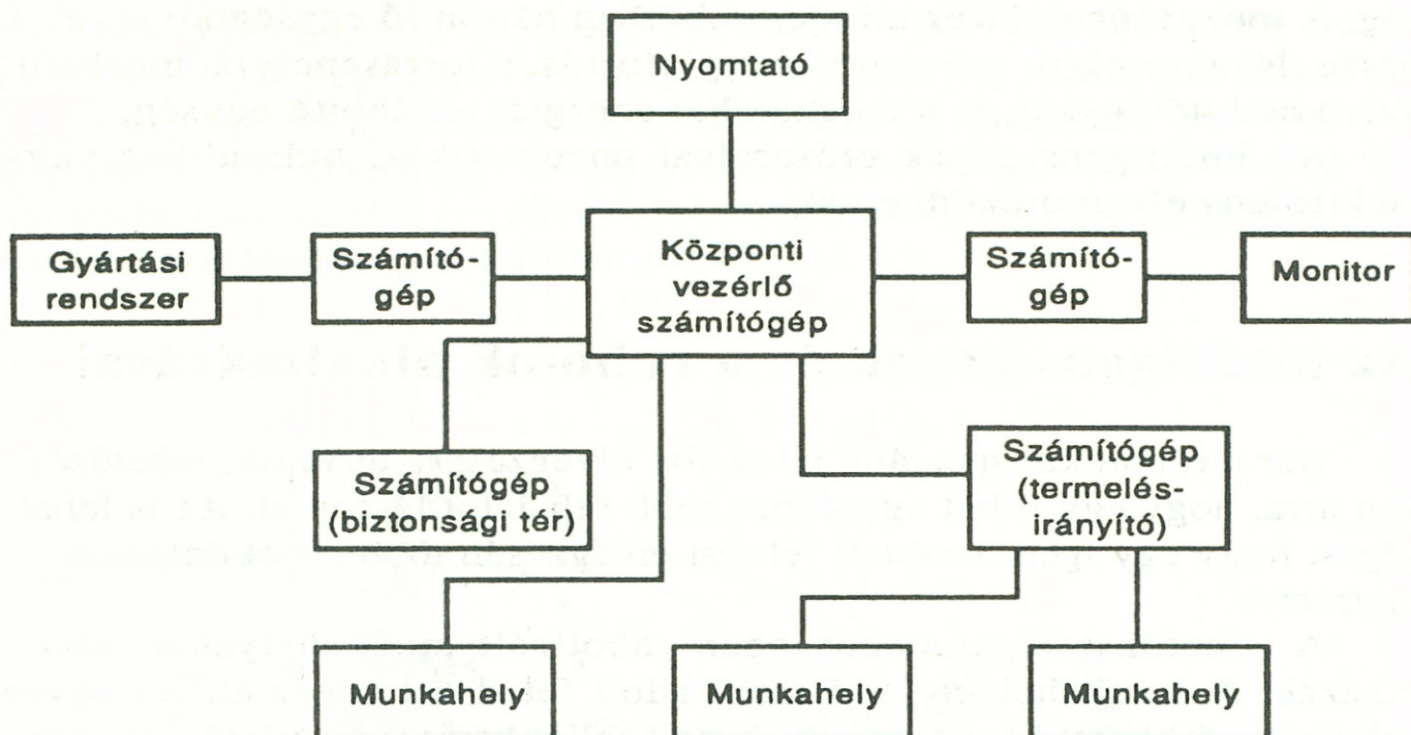
Bár a robotokat speciális feladatok elvégzésére tervezik, lehetőség van arra, hogy egy robot egyidőben több feladatot lásson el. Az is lehetséges, hogy egy igen összetett feladat elvégzésén több robot dolgozik egyszerre.

A robotokat legáltalánosabban robotizált munkahelyeken alkalmazzák. A munkahelyen lévő robot adott feladatot végez el. Az egyes robotokat fel lehet használni ún. gyártócellákban is. A gyártócella több munkahelyből áll, amelyeket azonban egy robot szolgál ki.

A robotok ipari alkalmazása forradalmi változásokat hozott a termelésben. Robotizált munkahelyek, gyártócellák és rendszerek alkalmazásával nagymértékben megváltoztak a termelési módszerek is.

A számítógéppel segített termelés (CAM) a robotok alkalmazásának logikai kiterjesztése. CAM rendszerekben olyan központi számítógép van, mely összekapcsolja a termelési környezet összes intelligens és egyéb gépeit. A CAM sokkal több, mint egyszerűen a robotok vezérlése. A központi számítógép *irányítja* a termelés különféle feladatait, *ellenőrzi* a termelési tervet és elvégzi az energia elosztását. Más szóval a központi számítógép *vezérli* a termeléshez szükséges összes funkciót. A termelés során semmi nem történhet a számítógép vezérlése, tudta nélkül. A CAM-rendszerben a kiszolgáló személyzetnek csak figyelnie kell a számítógép működését és a rendszer berendezéseinek karbantartását kell elvégeznie. A rendszer vázlata látható a 7-14. ábrán.

A számítógéppel segített tervezés (CAD) során a tervező billentyűzet, rajzdigitalizáló, fényceruza és egér (mouse) segítségével alkotja meg a terméket, és egy számítógépes rendszerben raktározza el. A tervező a képernyőn figyelheti a terméket, de nyomtatóval vagy plotterrel másolatot is készíthet róla.



7-14. ábra CAM-rendszer

A rugalmas gyártórendszer (FMS) egyesíti a CAD/CAM-technológiát a robottechnikával, egy teljesen automatikus termelési folyamat létrehozásához. A rugalmas gyártórendszerben a központi számítógép nemcsak a termelést irányítja, hanem a termelési folyamat tervteljesítését is.

Az FMS-ben az új tervek a CAD segítségével emberi beavatkozás nélkül, vagy minimális beavatkozással készülnek. A rendszer meghatározza a nyersanyagot és megtervezi az új terméket. A termékváltás így számítógép közreműködésével zajlik le.

Az FMS technológiát kis és nagy sorozatok gyártásánál egyaránt alkalmazni lehet, bár a nagy rugalmasság inkább a kis sorozatú gyártásra teszi alkalmassá.

Fontos előny, hogy az FMS azonnal reagálni tud a piaci változásokra, méghozzá új és költséges berendezések megvétele nélkül.

Az itt bemutatott példák csak ízelítőt adhatnak a számítástechnika és a robottechnika egyre szélesebb körű alkalmazásáról. Ezeknek az eszközöknek a használata megkönnyíti a munkánkat az ipar területén is.

8. A számítástechnikai szaktanfolyami oktatás és a szakmai képesítő vizsgáztatás rendszere

A számítástechnikai szakfeladatok központi kordinációja

"Aki tudja - csinálja,
Aki nem tudja csinálni - tanítja,
Aki tanítani sem tudja - irányítja."
(Menecken)

Magyarországon a számítástechnika/informatika-alkalmazás-oktatás témakörének állami szintű szakmai felelőse a Központi Statisztikai Hivatal. Ez a kompetencia – miután gyökerei az 1949-es népszámlálásig nyúlnak vissza – tradicionális. Ugyanis ennek a feladatnak az elvégzése indokolta, hogy a "KSH"-ba telepítsék az akkor legkorszerűbb és legnagyobb kapacitású, napjainkban országos hálózattal működő számítógép központot. A kötelező statisztikai adatgyűjtések nagy tömegű adatainak rendszeres feldolgozása szükségessé tette a KSH saját számítástechnikai szakembergárdájának kiképzését; a nagyszámítógépes munkafolyamatok "leképezése" útján alakult ki a Foglalkozások Egységes Osztályozási Rendszerében (FEOR) a közel 80 féle "fizikai" és "szellemi" számítástechnikai foglalkozás.

A "számgyár"– technikai és szellemi bázisait kiaknázva – rövid időn belül már az adatszolgáltatók részére is szervez számítástechnikai tanfolyamokat. Az első időkben az oktatás (egy-egy munkafolyamat vagy konkrét géptípus kezelésének a megtanítása) még csak 1-2 hétig vagy hónapig, később már (szakmai képesítést adva) több féléven át tartott. A tanfolyami záróvizsgák követelményeit sikeresen teljesítő hallgatók – "számítógépprogramozó", "elektronikus gépi szervező", "gépkezelő", "kódoló" stb. – szakképesítő bizonyítványát a KSH Ügyvitelgépesítési Felügyelete állította ki.

Tekintettel arra, hogy e tanfolyamok oktatási tematikáinak összeállításánál messzemenőkig figyelembe vették a hallgatókat beiskolázó vállalatok, intézmények igényeit, a "KSH-iskolából" – az 1950-es évek elejétől, közel húsz éven át – olyan szakemberek kerültek ki, akiknek erőssége elsősorban a számítástechnika gyakorlati alkalmazása volt.

E rövid történeti visszatekintés talán arra is válaszul szolgál, hogy – a tevékenységét döntő mértékben meghatározó statisztikai szakfeladatok mellett, miért tartozik a magyar statisztikai hivatalhoz a számítástechnika-alkalmazás tárgykörébe tartozó ügyek országos koordinációja. A KSH azonban csak az iskolarendszeren kívüli számítástechnikai tanfolyami szakképzés tekintetében gyakorolja *önállóan* a központi (állami) irányítási jogkört; az iskolarendszer keretében megvalósuló számítástechnikai szak- és alkalmazóképzés tekintetében ún. "osztott felelősség"-ben jár el.

Ez konkrétan azt jelenti, hogy szakmai állásfoglalást (pl.: a képzés céljairól, tantervek, szaktárgyi programok fejlesztéséről) vagy szakvéleményt (pl. jogszabály- vagy koncepciótervről, továbbá a különböző forrásokból finanszírozott pályázatok elbírálásának szempontjairól) akkor ad; a számítástechnika-alkalmazás fejlesztés szakmai követelményeit a különböző szakbizottságokban és testületek előtt akkor képviselheti, ha ezt az adott döntések meghozatalára illetékes államigazgatási szervezet, illetve testület nála kezdeményezi.

Mindemellett – főleg az utóbbi években – minden olyan oktatási intézménynek, amely ezt igényli, térítésmentesen adja át a számítástechnikai szaktanfolyamok központi oktatási dokumentációit, de a rendelkezésre álló korszerű külföldi oktatóprogramokat is.

Ami a "hőskorszak"-ot követte: a "szakképesítést nyújtó" tanfolyami oktatás

"Az elemzés olykor megútáltatja velünk részleteiben azt, ami egészében elviselhető volt."

(Valéry)

A kis- és mikrogépek robbanásszerű elterjedése világszerte maga után vonta a számítástechnikai szakemberképzés gyökeres és gyors tartalmi

megújítását. A korszerű technikának erre a kihívására azonban az egyes országok – fejlettségi szintjük, továbbá állami és gazdálkodó szervezeteik érzékenységének, illetve reakciókészségének különbözősége szerint – eltérően reagáltak. Magyarország számítógép állományának átrendeződése az 1984-1988-as időszakban következett be; mégpedig oly módon, hogy miközben az ország számítógépeinek száma – 98%-ban az újonnan üzembehelyezett kis- és mikrogépek hatására! – a 13-szorosára, addig a számítástechnikai (munkakörökben foglalkoztatott és) szakképesítéssel rendelkezők átlagos állományi létszáma mindössze 2%-kal növekedett. Úgy tűnik tehát, nincs sok okunk esodálkozni azon, hogy "A SZÁMÍTÁSTECHNIKA" hazánkban – a gépállomány minőségi átalakulása ellenére – sem hozott (és még ma sem hoz) számottevő eredményeket. Ennek sajnálatos, a társadalmi közérzetre gyakorolt negatív hatásai között elegendő, ha itt csak a különböző intézmények által nyújtott "gépesített ügyfélszolgálat" jelenlegi állapotaira utalunk...

A számítástechnikai szakalkalmazottak számának a 80-as évek végén tapasztalt lassú növekedési üteméből, valamint az egyes tanévek adatait tartalmazó tanfolyami oktatási statisztikák együttes vizsgálatából – többek között – arra lehetett következtetni, hogy a különböző számítástechnikai szakképesítéseket nyújtó intézmények (tudományegyetemek természettudományi karai, 1-2 főiskola, valamint a szakközépiskolák, továbbá az ún. "szakképesítést nyújtó" tanfolyamok szervezését végző két budapesti cég) együtt sem képesek annyi szakembert kibocsátani, hogy ezáltal minden személyi-számítógép mellé megfelelően felkészített szakalkalmazottak kerülhessenek. (Ennek egyenes következménye, hogy – amint a nagyszámítógépeket korábban "nyomdai sokszorosító"-ként, úgy – manapság az új PC-k egy jelentős részét legfeljebb az írógép helyett, illetve unaloműző szórakozásul használják.) Valójában csak a 80-as évek végére tehető annak felismerése is, hogy a számítástechnikai szakképzés – (gyakorlattól idegen) tartalmi hiányosságai jelentős részben a túl szűkre méretezett tanfolyami szervezetre, illetve a közoktatásirányítás merevségére vezethetők vissza. A szakképzési tematikák zöme rendkívül lassan, illetve (a tömeges programozó és folyamatszervező képzést illetően) szinte egyáltalán nem követte a 90-es évek számítástechnika-alkalmazóival szemben támasztott követelményeket; korszerű (IBM-kompatibilis) PC-gépe csak 1-2 középiskolának volt; ezeket a felsőoktatási intézményekben is gondosan őrizték a hallgatók "csínytevései" elől. Az ún. "iskolaszámítógépe-

sítési program" megítélése szakmai körökben mindmáig két véglet köré csoportosul; az azonban tény, hogy a pedagógusok megnyerését és szervezett kiképzését az "elfelejtett szektor" kategóriájába sorolhatjuk.

Eközben, a már említett tanfolyami statisztika adatai szerint, az országban több intézmény szervezett számítástechnikai tanfolyamokat. Ezek között szép számmal akadt olyan is, amely – az oktatott szakismeretek színvonala, a foglalkozások óraszama, illetve a vizsgakövetelmények alapján, korszerű "szakképesítést nyújtó" tanfolyamként is megállta volna a helyét. Az adatszolgáltató intézmények azonban – tudván azt, hogy ilyen jellegű képzés szervezésére állami engedéllyel mindössze két budapesti cég rendelkezik, a kurzus jellegét meghatározó lehetséges válaszok közül, az egyszerűbb utat választva, "továbbképzés"-t jelöltek meg.

A számítástechnikai tanfolyami szakképzés jelenlegi rendszere

"Semmilyen szél nem jó annak,
akinek nincs célul kiszemelt
kikötője."

(Montaigne)

A számítástechnikai tanfolyamokon történő szakképzés rendszerének újjászervezése a KSH-ban – az ismertetett előzmények alapján, 1988 tavaszán kezdődhetett el. Ekkor, a közreműködésre felkért szakértő csoport javaslatainak figyelembevételével, a következő irányelvek alakultak ki:

- a tanfolyami szakoktatás, valamint a képesítő vizsgáztatás egységes tartalmi követelményeinek kidolgozását követően, létre kell hozni a tanfolyami szakképzés versenysemleges piacát ;
- a szakképzési programokat – elsősorban a szakemberek foglalkoztatásában érdekelt gazdálkodó szervezetek (felhasználók), továbbá a munkavállalók (tanfolyami hallgatók) igényeit figyelembe véve – korszerűsíteni kell, majd folyamatosan biztosítani, hogy ezek mindenkori korszerű változatai eljussanak az érdekeltekhez;

- a számítástechnikai szakalkalmazottakkal szemben támasztott követelményeket a **komplex szaktudás igényével** kell megfogalmazni (ezzel összefüggésben a FEOR-t is a ténylegesen megjelenő szakmai feladatoknak megfelelően korszerűsíteni);
- a központi koordináció lehetőségeit kihasználva arra kell ösztönözni, hogy egy-egy konkrét szaktanfolyam oktatási tematikáiban vegyék figyelembe az oktatást szervező **intézmények környezetében felmerülő munkáltatói-alkalmazói igényeket** is, továbbá elő kell segíteni a meglévő hazai számítógépkapacitások gazdaságos(abb) kihasználását és a **szakképzést felelősséggel folytató intézmények országos hálózatának kialakulását**.

Az eddigi tapasztalatok arra utalnak, hogy ezeket az alapvető célkitűzéseket – a szaktanfolyami képzésben részt vevő intézmények tevékeny közreműködésével, szinte maradéktalanul sikerült megvalósítani.

A szaktanfolyamokon való részvétel és/vagy az előírt (későbbiekben ismertetésre kerülő) vizsgakövetelmények teljesítése alapján **jelenleg** megszerezhető számítástechnikai szakképesítéseket az *8-1. melléklet* foglalja össze.

A szaktanfolyamok hallgatói közül – 1989. február és 1991. december 31. között – több százan szereztek alap, közép (és közel ötvenen) felsőfokú számítástechnikai szakképesítést; a folyamatban lévő tanfolyamok hallgatói létszáma meghaladja a kétezer főt.

A napjainkig felgyűlt tapasztalatok, valamint a munkaerőpiacról érkező információk folyamatos értékelése előreláthatólag ismét szükségessé teszi a 11 féle szaktanfolyam újbóli felülvizsgálatát és bizonyos korrekciók végrehajtását.

Amikor ugyanis nem mutatkozik kereslet sem a "**középfokú szervezők**", sem a "**középfokú programozók**" iránt – amit különösképpen nem tartunk meglepőnek –, akkor fel kell, hogy merüljön a kérdés: vajon célszerű-e a (gyakran nehezen megszerzett, ám) piacképtelen szakismereteket állami képesítő bizonyítvánnyal szentesíteni? Ide sorolhatjuk a "**számítógépkezelő (operátor)**" szakmát is, miután ennek eredetileg a nagyszámítógépek környezetében volt (és lenne) létjogosultsága; ilyen számítóközpont azonban hazánkban kevés van, és ez a kevés nem igényli az újonnan kiképzettek "tömegeit". (Ugyanakkor az is kérdéses, mihez kezdhetnek azok a fiatalok, akik középiskolai tanulmányaik végén "középfokú operátor" képesítő bizonyítványt is kaptak;

azonban tényleges szaktudásuk – a fakultációs órakeretben elsajátított számítástechnikai és programozási (alap)ismeretek mellett – kevés kivétellel – messze elmarad az azonos elnevezésű tanfolyami szakképesítés jelenlegi elméleti és gyakorlati követelményei mögött.)

A fentiekkel ellentétben viszont igen kívánatos lenne, hogy minél több, a számítástechnika eszközeit és elméleti módszereit felsőszinten (ugyanakkor az őket megillető helyen és mértékben, de) alkalmazni képes "információrendszer szervező" és "információrendszer-programozó" szakember kerüljön be a jelenlegi – és remélhetőleg hamarosan újonnan létesülő további, munkaalkalmat biztosító – szervezetek vérkeringésébe. (Noha a szükséges szakértelem hiányát általában ritkán számszerűsítjük, könnyű belátni, hogy "az elmaradt haszon" többet árt, mint gondolnánk...) Ezzel összefüggésben – a részben már megvalósult oktatási kísérletek kedvező eredményei alapján – tervezzük az újabb felsőfokú "számítógép-programozó", valamint a "számítógép-rendszer üzemeltető" szakképesítések bevezetését.

Ezek után essék szó a legkeresettebb tanfolyamokról is! Nos, eddig két ilyen "listavezetővel" találkoztunk. Az egyik az alapfokú szakképesítés megszerzésére felkészítő "személyi-számítógép kezelő" (népszerű nevén a PC-operátor), a másik pedig (jó esetben az előbbire ráépített, középfokú szakképesítéshez eljuttató) "személyi-számítógép szoftverüzemeltető". E két szakma közös vonása az abszolút biztonságos gépkezelői, illetve a mélyebb elméleti szakismereteket is feltételező üzemeltetői készség. Ilyen biztonságra pedig – a szükséges mélységű elméleti ismeretanyag elsajátítása mellett – csak alapos és megfelelő időtartamú gyakorlással lehet szert tenni. Ez a magyarázata annak, hogy e két szaktanfolyam esetében a KSH az 1 hallgató/1 gép technikai kapacitás biztosítását kötelezően írja elő.

Minden érdekeltnek tisztában kell lenni azzal is, hogy a központi szakképzési programok keret-tantervei – amint az a táblázatban is olvasható – a képzésnek nem a maximális, hanem a minimális időtartamát tüntetik fel. Ettől némely szervezetek (szem előtt tévesztvén a szakképzés céljait, valamint a hallgatók érdekeit, valamely rosszul értelmezett "mennyiségi mutató" teljesítésére törekedve) szívesen eltekintenek. Holott, egyértelműen a szakképzést szervező intézmény felelős azért, hogy a szakképesítő vizsgára való felkészüléshez minden tanfolyam hallgatója egyenlő esélyt kapjon. Ezért tekinthetjük örvendetesnek, hogy – a kedvezőtlen tapasztalatok ellenére – azoknak a szakintézményeknek a száma növekszik, amelyek – miután megismer-

ték hallgatóik előképzettségét, életkorát, eddigi munkaviszonyának jellegét, az előírtnál lényegesen magasabb óraszámmal szervezik meg az oktatást. Ennek igénye általában az átképzés-jellegű tanfolyamok esetében merül fel; a szakintézmény azonban rendszerint csak akkor realizálhatja elképzeléseit, ha az általa képviselt szakmai követelmények megfelelő súllyal esnek latba a képzést finanszírozó szervezetnél is.

Profik legyünk a számítástechnikában vagy alkalmazók?

"Mindenki csak azt érti meg,
amit önmagában is megtalál."

(Amiel)

Az igazság az, hogy mind a két szakembertípusra szükség van! Valószínűleg igazuk van azoknak, akik szerint a professzionális számítástechnikai szakemberek kibocsátása valójában a középiskolák és a felsőoktatási intézmények feladata. Mindaddig azonban, amíg az iskolarendszer ezt a feladatát nem képes maradéktalanul ellátni, addig – a társadalmi szükségletek kielégítésére tekintettel – nem lehet megkérdőjelezni a tanfolyami szakképzés életben tartásának fontosságát.

Azok dolgát tehát, akik jelenleg bármely okból számítástechnikai szakképesítés megszerzésének lehetőségét fontolgatják, mindenképpen szeretnénk megkönnyíteni. Ezért – a 8-2.,...8-6. számú mellékletben – bemutatjuk néhány szaktanfolyam központi kerettantervét, a legnagyobb hallgatói létszámot vonzó "PC-operátor" szakma esetében a képesítő vizsga szóbeli tételsorát is.

A táblázatok áttekintését követően, ajánlatosnak tartjuk, hogy elgondolkozzék az alábbiakon:

- Ha a tantárgyak többségét már tanulta, netán bármelyikből előző tanulmányai során vizsgát is tett, akkor helyes lenne ismereteit felfrissíteni és tudását szakmai képesítéssel "megkoronázni". (Az egyéni felkészülés alapján történő képesítő vizsgáztatás feltételeiről a továbbiakban lesz szó.)
- Ha valamelyik szaktanfolyam tantárgyai közül csak eggyel vagy eggyel sem találkozott korábban, akkor helyes lenne, ha szak-

tanfolyamon tanulna tovább. (Az oktatást szervező intézményhálózatról is később lesz szó.)

- Ha viszont Önnek a keret-tantervek alaposabb megismerésétől már a kezdet kezdetén elment a kedve, akkor biztos lehet abban, hogy a számítástechnikai szakképesítéseket nem Önnek találták ki. Ez azonban korántsem jelenti azt, hogy nem volna érdemes résztvennie olyan tanfolyamon sem, ahol pl. kizárólag a szöveg- vagy kiadványszerkesztés, a táblázatkezelés vagy egy integrált (több célú, adatkezelő) program alkalmazásának fortélyait ismerhetné meg. Ezek közül ugyanis bármelyik – a nélkül, hogy szakképesítése lenne – igen jól hasznosítható!

Vonjuk le hát bátran a következtetést: **felesleges mindenkinek, mindenáron számítástechnikai szakképesítés megszerzésére törekedni!** Ez elsősorban azoknak ajánlható, akik iskolai tanulmányaik eredményeképpen már rendelkeznek valamilyen szakképesítéssel és a számítástechnikai szakmai ismeretek birtokában magasabb szinten, szélesebb áttekintéssel kívánják megoldani feladataikat; valamint azoknak, akik "puszta" érettségi bizonyítványukkal – piacképes szakma hiányában – kevés eséllyel tudnak a munkaerőpiacon megjelenni.

A szakképesítés megszerzésére felkészítő tanfolyamok szervezett rendszere

"Az intézmények csak annyit érnek,
amennyit azok az emberek,
akik működtetik azokat."

(Amiel)

Amennyiben a tanulás célja egyértelműen számítástechnikai szakképesítés megszerzése, akkor célszerű ezt szakintézményben megkezdeni. Ilyenek azokat tekintjük, amelyeket a KSH (meghatározott személyi és tárgyi feltételeik alapján) felvesz a Központi Számítástechnikai Szakképzési Jegyzékbe (a továbbiakban: SzJ). Az 1992. március 2-ig bejegyzett 110 intézmény felsorolását tartalmazó 8-7. mellékletből látható, hogy az országban ma már szinte mindenki találhat viszonylag elérhető távolságban olyan intézményt, amelyben a szakmai oktatás az országosan egységes képesítési követelmények szerint történik.

Az SzJ-ben nyilvántartott intézményekre is érvényes, hogy tevékenységük színvonalát elsősorban hallgatóik szakmai képesítő vizsgán nyújtott teljesítménye minősíti. Ezzel kapcsolatban – közel négy év elteltével – talán nyugodtan mondhatjuk, hogy a szaktanfolyamok meghatározó többségében igényes a felkészítés.

A számítástechnikai tanfolyami szakképzés központi koordinációja során azt tapasztaltuk, hogy – a mégoly gyors ütemben fejlődő és piacérzékeny számítástechnika esetében is – hozhatók "állami intézkedések" az oktatást/vizsgáztatást végző szakemberek döntő többségének egyetértésével. Talán ennek a módszernek köszönhető az is, hogy a szaktanfolyamok központi oktatási dokumentációira az utóbbi 1.5 évben már számtalan szakközépiskola és gimnázium mellett, egyetemink közül is több felfigyelt. (Az SzJ-ben jelenleg nyilvántartott általános iskolák, közép- és felsőoktatási intézmények részben saját tanulóik szakképesítéshez juttatása, másrészt pedig felnőttek tovább-, illetve átképzése céljából tartanak szaktanfolyamot.)

Miközben ismét leszögezzük: egyáltalán nem mindegy, milyen célból és milyen mélységben akarunk közelebbről megismerkedni a számítástechnikával, szeretnénk emlékeztetni arra, hogy nem csak a különböző ismeretek megtanulására, de ma már ezek oktatására is mindenki szabadon vállalkozhat. Elvileg sem annak, hogy valaki "egyéni" készüljön fel a számítástechnikai szakmai képesítő

vizsga letételére, sem pedig annak, hogy ilyen típusú szakmai oktatást szervezzen, nincs akadálya. Az egyéni felkészülők is széles körből választhatnak, ha egy-egy tananyagrészt elsajátításához olyan szervezettek (vagy magánszemélyek) segítségét kívánják igénybe venni, amelyek (vagy akik) nem szerepelnek a SzJ-ben. A "kötelező óvatosság" azonban arra int, hogy (még a tandíj befizetése előtt) célszerű meggyőződni arról, vajon rendelkezik-e a kiszemelt vállalkozó az eredményes oktatás minden nélkülözhetetlen feltételével...

Mit kell tudni a szakmai képesítő vizsgáztatásról?

"A legtöbb emberrel általában nem az a baj, hogy valamit nem tud, hanem inkább az, hogy amit tud – az nem úgy van."

(Boulding)

A következő, teljességre nem törekvő, ismertetést célszerű a **központi kerettantervek** oldaláról kezdeni. Amint az a 8-2.,...8-6. *mellékletek* áttekintésével megállapítható, szerkezetük **egységes**; az egyes tantárgyak mellett mindegyikben láthatók a "T/1., T/2." stb. jelzések. Az ily módon megkülönböztetett tárgyak oktatása során, az adott képesítés szakmai tartalma szempontjából szükséges alapvető ismereteket lehet, illetve kell elsajátítani.

Fontos tudnivaló, hogy az ún. **"tantárgyi vizsgaköteles"** tananyagból tett eredményes vizsgák meglétét a szakmai képesítő vizsgára történő jelentkezés előfeltételeként – mind a szaktanfolyamok hallgatóinak, mind az egyénileg felkészülőknek – igazolni kell. (A 8. számú mellékletben a szaktanfolyami rendszerben használatos igazoló okmány mintája látható.) Ez a szabály a gyakorlatban úgy érvényesül, hogy a képesítő vizsgára történő jelentkezés alkalmával (ami az adott típusú szaktanfolyamot szervező bármelyik, az SZJ-ben szereplő intézménynél lehetséges), vagy az adott vizsga (szakintézménynél, három éven belül történt) letételét kell igazolni, vagy azt lehet kérni, hogy a képesítő vizsgát szervező intézmény a korábbi (három évnél nem régebben letett) vizsgákat ismerje el egyenértékűnek. Ez utóbbi ese-

tekben az oktatást/vizsgáztatást szervező intézmény – a korábban és a jelentkezés időszakában oktatott tananyag összevetése alapján, az újabb ismeretanyagok pótlása érdekében – különbözeti vizsgák letételeit írja, illetve írhatja elő.

A szakmai képesítő vizsgára való jelentkezés előfeltételeit illetően az iskolai végzettséggel kapcsolatos jelenlegi KSH-állásfoglalás nem tekinthető "örökérvényűnek". Ez egyelőre nem, illetve az alapfokú (PC-kezelő) képesítés megszerzésénél "alulról", vagyis úgy szabályoz, hogy nem köti érettségi bizonyítványhoz. E mellett azonban, most és minden konkrét információrendszer-szervező, illetve programozó, valamint számítóközpont üzemeltetés-vezető szaktanfolyam beindítását megelőzően azt a j á n l j u k, hogy lehetőleg olyan hallgatókat vegyenek fel, akik már rendelkeznek valamilyen felsőfokú iskolai végzettséggel. Ugyanis ezek a szakemberek a mindennapi gyakorlatban (feltehetően többnyire) olyan feladatokkal találkoznak, amelyek megoldásához a szaktanfolyamon elsajátítható ismeretanyag önmagában nem elégséges. Jó tehát még idejekorán szembe néznünk azzal a ténnyel, hogy az iskolarendszerben megszerezhető – általános műveltségbeli és szakismereti – tudást még a legmagasabb színvonalú tanfolyam sem pótolhatja!

Van azonban néhány azonossági pont, amely mind az iskolarendszerű, mind a (számítástechnikai) tanfolyami szakképzés tekintetében azonos kell, hogy legyen; ezek közül talán a legfontosabb a teljesítményképes szakemberek kibocsátásának alapkövetelménye. Ha emellett a (jelzők nélküli!) piacgazdaság várható igényeit is illúziók nélkül vesszük figyelembe, akkor a már észlelhető jelzésekből le kell vonnunk a megfelelő következtetéseket. Így fog sor kerülni – valószínűleg már a következő tanévtől – arra, hogy a szakmai képesítő vizsgáztatás jelenlegi (többek által túlságosan liberálisnak minősített) követelményrendszerét megszigorítsuk. Az "ésszerű korlátok" minden bizonnyal érinteni fogják a "középfokú szervező és programozó" képzést (pl. úgy, hogy a szakmai képesítő vizsga letétele szaktanfolyam elvégzéséhez, a tanfolyam megkezdése pedig bizonyos előző vagy egyidejű gyakorlati munkához kötődik); várható még, hogy a felsőfokú szakmai képesítések megszerzése csak azok számára lesz igazolható, akik már egyetemi, főiskolai oklevéllel is rendelkeznek.

A szakképzés programjain túl, a szakképesítő vizsgák országosan egységes követelményeit, valamint a vizsgáztatás lefolytatásáról, továbbá az oktatást/vizsgáztatást szervező intézmények jogairól és köte-

lezettségéről szóló KSH-szabályzat folyamatos korszerűsítése is az intézmények kezdeményezésére (illetve konkrét tapasztalatok hasznosítása útján) történik.

A közép- és a felsőfokú szakmai képesítő vizsgát 3 tagú, a szaktanfolyamot szervező intézménytől független (és a hallgatók érdekében a tanfolyam helyszínére decentralizált) vizsgáztató bizottság előtt kell letenni. A vizsgabizottságok elnökét, illetve a felsőfokú vizsgáztatás alkalmával mindhárom tagját a KSH delegálja. Ezeken a szinteken a képesítő vizsga írásbeli/gyakorlati és szóbeli részből áll; a felsőfokú szakképesítő vizsga szóbelijén a jelölteknek (önállóan készített!) szakdolgozatukat is meg kell védeni. A vizsgafeladatokat, illetve a szóbeli tételsorokat – amennyiben azok eltérnek a központi dokumentációban közöltektől – a vizsgabizottság elnökével (vagy a KSH-val) a kihirdetés előtt, előzetesen egyeztetni kell.

Az eredményes képesítő vizsga alapján a középfokú szakképesítések megszerzését "Képesítő bizonyítvány", a felsőfokú szakképesítések megszerzését "Képesítő oklevél" elnevezésű okmányon, a vizsgáztató bizottság igazolja, és az illető intézmény bocsátja ki.

A "személyi-számítógép kezelő" alapfokú szakképesítés megszerzését igazoló "Záróbizonyítvány"-t (a központi kerettantervben előírt négy tantárgy eredményes vizsgái alapján) az oktatást szervező intézmény önállóan állíthat ki. (A 8-9.,...8-11. számú mellékletek az említett okmányok mintáit ismertetik.)

Valamennyi szaktanfolyamra érvényes az a szabály, hogy az oktatás beindulásáról, valamint várható befejezéséről a KSH-t értesíteni kell. Az értesítésben az intézményeknek a hallgatók létszáma, az oktatók neve és szakképzettsége mellett, a tantárgyi vizsgák időpontját is be kell jelenteni. Ez utóbbi teremti meg annak a lehetőségét, hogy a tantárgyi vizsgákra esetenként "központi vizsgabiztos" látogasson el. Az ilyen céllal delegált szakembernek az a feladata, hogy személyesen, az oktatás helyszínén győződjön meg a rendelkezésre álló technikai feltételekről és a szakképzési program teljesítésének részeredményeiről; továbbá, hogy bizonyos korrekciók szükségességére idejében felhívja az illetékesek figyelmét.

Eddig még nem esett szó arról – holott a "legjobb családban is előfordulhat" –, mi a teendője annak, akit a szakmai képesítő bizottság első alkalommal kénytelen volt "visszatapsolni". Természetesen a számítástechnikai szakmai képesítő vizsgát is meg lehet ismételni; mégpedig bármely, az SZJ-ben szereplő intézménynél (illetve minősí-

tett esetekben a KSH-ban). Az erre vonatkozó eljárási szabályok – több nyugati országgal szemben – azt sem tiltják meg, hogy valaki ismétlő vizsgán akár jeles osztályzatot is kaphasson.

Úgy tűnik, most érkeztünk el addig a pontig, amelyen túl már a terjedelmes részletkérdések taglalása következne. Miután azonban a további tájékoztatás iránti igény személyenként szükségképpen más-más jellegű, ezért helyesebb, ha ezeket mindenki közvetlenül a legközelebbi számítástechnikai szakmai oktatást/képesítő vizsgáztatást folytató intézményétől kéri meg. A szakképzés első vonalában dolgozó szakemberek, mindenkor készséggel állnak az Önök rendelkezésére. Kérem, forduljanak hozzájuk bizalommal!

Számítógépes tanfolyamok, illetve képesítések

Tanf. típus-száma	Szaktanfolyam/szakképesítés megnevezése	Az oktatás minimális összóraszáma	Képesítési fokozat
1.1.	Számítástechnikai szervező	450	közép
1.2.	Információrendszer-szervező	800	felső
2.1.	Számítógép-programozó	450	közép
2.11	Számítógép-programozó	900	felső
2.21	Számítógéprendszer-programozó	900	felső
2.22	Információrendszer-programozó	900	felső
3.1.	Személyi-számítógép-kezelő	150	alap
3.2.	Személyi-számítógép szoftver-üzemeltető	350	közép
3.3.	Számítógépkezelő (operátor)	400	közép
3.4.	Számítóközpont-üzemeltetés-vezető	750	felső
4.1.	Személyi-számítógép műszaki karbantartó	450	közép
4.2.	Számítógéprendszer műszaki karbantartó	750	felső

1.2 Információrendszer-szervező, felsőfokú számítástechnikai szakképesítés megszerzésére felkészítő szaktanfolyam kerettanterve

Tantárgycsoport megnevezése	Ajánlott minimális óraszám		Előírt tantárgyi vizsga
	elm.+gyak.	együtt	
A/ Alapozó tárgyak			
1. Matematika	-	70	T/1.
2. A számítástechnika alappjai	-	45	T/2.*
3. A programozás alapjai	-	60	
B/ Szaktárgyak			
1. Információs folyamatok	-	80	T/3.
2. Programozás	-	90	T/4.
3. Adatkezelés, adatbázisok tervezése	-	105	T/5.
4. Alkalmazói program- csomagok	-	50	T/6.
5. Rendszerelemzés és rendszertervezés	-	135	T/7.
6. Operációkutatósi mód- szerek, döntéselőkészítés Szakdolgozat előkészítés és konzultáció	-	55 40	- -
C/ Kötelezően választható szaktárgyak	-	120	T/8**
1. A speciális alkalmazói környezet ismerete	-	-	
2. A számítógép-alkalmazás jogi kérdései (adatál- lományok védelme, adat- biztonság, számítógépes bűnözés)	-	-	
3. Alkalmazások gazdasá- gossága	-	-	
4. Szakértői rendszerek	-	-	
5. Projektirányítás	-	-	
D/ Fakultatív tárgyak (Pl.: idegen nyelv, statisztika, korszerű szervezési eszközök, közgazdasági, igazgatás- ügyviteli ismeretek stb.)			

Az A/-C/ tantárgycsoport minimális

összórásszáma: - - 800 (7)

* Az A/2. és A/3. tantárgyból összevont!

** A C/1. és további egy választott tantárgyból a konkrét témaköröket az oktatást szervező intézmény határozza meg.

1.21 Számítógéprendszer-programozó, felsőfokú számítástechnikai szakképesítésre felkészítő szaktanfolyam kerettanterve

Tantárgycsoport megnevezése	Ajánlott minimális óraszám		Előírt tantárgyi vizsga
	elm.+gyak.	együtt	
A/ Alapozó tárgyak			
1. Matematikai és statisztikai ismeretek	-	-	45
2. A számítástechnika alapismeretek	-	-	45
3. A programozás alapjai	-	-	75
4. Operációsrendszer alapismeretek	-	-	30
5. A rendszertervezés alapjai	-	-	30
B/ Szaktárgyak			
1. Programozás és módszertan	-	-	165
2. Integrált adatkezelő rendszerek	-	-	36
3. Adatbázis-ismeretek, adatbáziskezelő rendszerek	-	-	99
4. Operációs rendszerek	-	-	45
5. Rendszerprogramozás (C-nyelven, Assembler)	-	-	60
6. Programozási technológiák Szakdolgozatok előkészítése és konzultációk	-	-	30
C/ Kötelezően választható szaktárgyak			
1. Programnyelv	-	-	110
2. Rendszerprogramozás	-	-	-
3. Hálózati szoftverek	-	-	-
4. Mikroprocesszorok	-	-	-
D/ Fakultatív tárgyak (Pl.: idegen nyelv)			
Az A/-C/ tantárgycsoport minimális összórászáma:	-	-	800
	50%	50%	100%
			(7)

* Egy válaszott tantárgyból, vagy max. két válaszott tantárgyból összevontan.

**A 3.1 Személyi-számítógép-kezelő, alapfokú
számítástechnikai szakképesítés megszerzésére felkészítő
szaktanfolyam kerettanterve**

Tantárgycsoport megnevezése	Ajánlott minimális óraszám		Előírt tantárgyi vizsga	
	elm.+gyak.	együtt		
A/ Alapozó tárgyak				
1. Számítástechnikai alapismeretek	-	-	36	T/1.
2. Gépkezelés, operációs rendszer ismeretei	-	-	42	T/2.
B/ Szaktárgyak				
1. Adatbáziskezelés (dBASE)	-	-	30	T/3.
2. Integrált adat- kezelő rendszerek interaktív használata	-	-	42	T/4.
2.1 FRAMEWORK				
2.2 Symphony				
2.3 QUATTRO				
2.4 WORKS 2.0				
C/ Kötelezően választható szaktárgy(-ak)				(T/4.)
Az oktatást szervező intézmény határozza meg, a B/2.-es tantárgyakból!				
D/ Fakultatív tárgyak (Pl.: a program alapjai, szakkifejezések idegen nyelven stb.)				
<hr/>				
Az A/-C/ tantárgycsoport minimális				
összóraszám:	-	-	150	(4)
	35%	65%	100%	

A 3.2 Személyi-számítógép szoftverüzemeltető, középfokú számítástechnikai szakképesítés megszerzésére felkészítő szaktanfolyam központi kerettanterve

Tantárgycsoport megnevezése	Ajánlott minimális óraszám		Előírt tantárgyi vizsga	
	elm.+gyak.	együtt		
A/ Alapozó tárgyak				
1. Számítástechnikai, matematikai, statisztikai alapismeretek	-	-	58	T/1.*
2. Gépkezelés és operációs rendszer ismeretek	-	-	65	T/2.
B/ Szaktárgyak				
1. Integrált adatkezelő rendszerek interaktív használata B/1.1 FRAMEWORK B/1.2 SYMPHONY B/1.3 QUATTRO B/1.4 Works 2.0	-	-	52	T/1.*
2. Szöveges információk feldolgozása (WORD)	-	-	47	T/3.
3. Lokális számítógép-hálózatok (NOVELL)	-	-	40	T/4.
4. Adatbáziskezelő rendszerek B/4.1 FoxBase+ B/4.2 dBASE III. PLUS	-	-	63	T/5.
C/ Kötelezően választható szaktárgyak A szaktanfolyamot szervező intézmény határozza meg!	-	-	25	-
D/ Fakultatív tárgyak (Pl.: szakkifejezések idegen nyelven, közgazdasági, igazgatási stb. ismeretek!)				
<hr/>				
Az A/-C/ tantárgycsoport minimális összóraszáma:	-	-	350	(5)

*Az A/1. és a B/1. tantárgyból összevont!

A 3.2 Személyi-számítógép szoftverüzemeltető, középfokú szakmai képesítő vizsga szóbeli tételsora

A/ TÉTELEK (szorosán vett üzemeltetői feladatok)

Ezekből a vizsgázónak kettőt (egyét gép mellett) kell megoldani!

- 1./ Mentés készítésének célja; hardver és szoftver eszközei
- 2./ Mentés készítése BACKUP programmal
- 3./ Mentés vissztöltése RESTORE programmal
- 4./ A PCTOOLS program kezelése, felhasználási lehetőségei
- 5./ Lemezek karbantartása PCTOOLS program felhasználásával
- 6./ Boot floppylemez készítése; ennek szükségessége
- 7./ Floppylemezek formázása; lemeztípusok és meghajtók összehasonlítása
- 8./ Winchester particionálása és formázása DM programmal
- 9./ Winchester particionálása és formázása FDISK programmal
- 10./ Winchester-írásvédelem kezelése DM programmal
- 11./ A COPY, XCOPY és DISCOPY parancsok összehasonlítása
- 12./ A könyvtárszerkezet fogalma, karbantartásának eszközei
- 13./ A DOS és a NOVELL fájlkezelő parancsok összefoglalása
- 14./ Adatvédelem biztosításának szoftver és hardver eszközei (DOS és hálózat esetén)
- 15./ Véletlen fájltilés visszaállítása a PCTOOLS programmal
- 16./ DOS parancsok általános alakja; a COMMAND.COM parancs-értelmező működése
- 17./ Rendszerfájlok visszaállítása, azok megsérülése esetén DM programmal
- 18./ Új felhasználó létrehozása és jogainak megadása NOVELL-ben
- 19./ Régi felhasználó jogainak megváltoztatása NOVELL-ben
- 20./ Winchester-tömörítő program használata (SPEED DISK, COMPRESS)
- 21./ Fájl-tömörítő programok használata (PKZIP, LZEXE)
- 22./ Vírusdetektorok használata, installálásuk
- 23./ Egér installálása
- 24./ Printer installálása, kapcsolók beállítása
- 25./ SETUP használata, az eszközök típusának beállítása

- 26./ Hardver tesztprogram használata
- 27./ Új számítógép telepítéskor végrehajtandó feladatok
- 28./ Számítógép szállításakor végrehajtandó feladatok bemutatása (gép, printer)
- 29./ Fájltranzfer végrehajtása gép-gép kapcsolat létrehozásával
- 30./ A DOS prancsok gyengeségei, kiváltásuk utilitykkal

B/ TÉTELEK (a számítástechnikai-elméleti alpműveltség felmérésére)

Ezekből a vizsgázónak egyre kell válaszolni!

- 1./ A kódolás fogalma, szükségessége; számrendszerek, kódkonverziók
- 2./ A programozási munka alapvető fázisai
- 3./ Az IBM PC-k hardver-felépítése; a részegységek feladatai
- 4./ A BOOT folyamat ismertetése DOS esetén
- 5./ Fájl-típusok; a fájlok védelme DOS, illetve hálózat esetében
- 6./ A CONFIG.SYS fájl szerepe; tartalma
- 7./ Az AUTOEXEC.BAT fájl szerepe; tartalma
- 8./ A DOS környezetváltozók (PATH, PROMPT)
- 9./ BATCH-programok írásának alapfogalmai; bemutatás egy példán
- 10./ Bejelentkezési procedúra készítése (jelszó, a használni kívánt program kiválasztása)
- 11./ A hálózatok működésének alapelvei; felhasználásuk területei
- 12./ A BOOT folyamat ismertetése NOVELL esetén
- 13./ A vírusok elleni védekezés eszközei; üzemeltetési fegyelem
- 14./ Táblázatkezelő program (feladata, alkalmazási területei)
- 15./ A táblázatkezelő programok; a grafikus funkciók használata (grafikonok)
- 16./ A táblázatkezelő programok; a függvények használata; adattípusok
- 17./ A táblázatkezelő programok; a fájlkezelés; fájlok konvertálása
- 18./ A táblázatkezelő programok; az adatrendezés lehetőségei; blokkműveletek; mátrixműveletek
- 19./ A táblázatkezelő programok; konfigurálás (printer, monitor stb. beállítása)
- 20./ Adatbáziskezelő program (feladata, alkalmazási területei)
- 21./ Az adatbáziskezelő programok; fájl létrehozása és feltöltése adatokkal

- 22./ Az adatbáziskezelő programok; rendezés és indexelés
- 23./ Az adatbáziskezelő programok; listázás feltételek szerint
- 24./ Az adatbáziskezelő programok; képernyő- és címkeformátum állományok előállítása
- 25./ Az adatbáziskezelő programok; egyszerű programok írása
- 26./ Az adatbáziskezelő programok; kész programok futtatása; hibaüzenetek; környezeti paraméterek (CONFIG.DB)
- 27./ A szövegszerkesztő programok; szövegszerkesztési alapfunkciók
- 28./ A szövegszerkesztő programok; betűtípusok és formátumok használata
- 29./ A szövegszerkesztő programok; a nyomtatási kép előállítása
- 30./ A szövegszerkesztő programok; karaktersorozat keresése, helyettesítése; fájlok összekapcsolása

A 4.1 Személyi-számítógép műszaki karbantartó, középfokú számítástechnikai szakképesítés megszerzésére felkészítő szaktanfolyam központi kerettanterve

Tantárgycsoport megnevezése	Ajánlott minimális óraszám		Előírt tantárgyi vizsga
	elm.+gyak.	együtt	
A/ Alapozó tárgyak			
1. Számítástechnikai alapismeretek	-	-	30
2. A programozás alapjai	-	-	60
3. Elektronikai alap- ismeretek	-	-	30
4. Matematika	-	-	15
B/ Szaktárgyak			
1. Operációs rendszer ismeretei	-	-	30
2. Mikrogepes periféria- vezérlők és perifériák	-	-	30
3. Mikrogepes assembler ismeretek	-	-	30
4. Számítógépek áramkörei	-	-	30
5. Mikroprocesszorok és kiegészítő elemeik	-	-	30
6. Műszer- és mérés- technika	-	-	30
C/ Kötelezően választható szaktárgyak			
1. Személyi-számítógépek karbantartása a/ XT/AT gépek b/ PS/2 gépek stb.	-	-	90
2. Speciális célú szá- mítógépes rendszerek			
D/ Fakultatív tárgyak (Pl.: szakmai idegen nyelv, matematikai, statisztikai, elektrotechnikai ismeretek)			
Az A/-C/ tantárgycsoport minimális összórászáma:	-	-	405
	65%	35%	100%
			(7)

* A B/5. és B/6. tantárgyból összevont.

** A C/1. és a C/2. tantárgyakból választott, illetve meghatározott két tárgyból összevont.

Központi Statisztikai Hivatal
Informatikai főosztály

A
KÖZPONTI SZÁMÍTÁSTECHNIKAI SZAKKÉPZÉSI
JEGYZÉK

– k i v o n a t –

Első bejegyzés: 1989. február 13.
Utolsó bejegyzés: 1992. március 2.

KSzSzJ sorszám	A bejegyzett intézmény neve - székhelye Az indítható szaktanfolyam típus számjele
1.	GATE Mezőgazdasági Gépészmérnöki Kar; Matematikai és Számítástechnikai Intézet – Gödöllő 2.1, 3.2
2.	Bakony Fém- és Elektromoskészülék Művek Oktatási Központ – Veszprém 1.1, 2.1, 3.1, 3.2, 3.3, 4.1
3.	Bolyai János Elektronikai Szakközépiskola – Budapest 3.1, 4.1
4.	EAST - WEST Számítástechnikai és Szolgáltató Kft – Debrecen 2.1, 3.1, 4.1
5.	ELTE Számítóközpont <i>Jogutód:</i> Informatika Oktatási Kft – Budapest 1.1, 2.1, 2.11, 2.21, 3.2, 3.3,
6.	Kempelen Farkas Kísérleti Gimnázium – Budapest 2.1, 3.2
7.	*LSI ÁTSZ <i>Jogutód:</i> LSI OMAK Alapítvány – Budapest 1.1, 1.2, 2.1, 2.21, 3.1, 3.2, 3.4, 4.1, 4.2
8.	Mezőgazdasági Üzemszervezési, Számítástechnikai és Informatikai Rt – Budapest 1.1, 1.2
9.	* Műszertechnika Kiszövetkezet <i>Jogutód:</i> Műszertechnika Számítástechnikai és Kereskedelmi Rt – Budapest 2.1, 2.21, 2.22, 3.2
10.	*Neumann János Közgazdasági Szakközépiskola és Gimnázium – Eger 3.1, 3.2

11. **OKTÁV Ipari Továbbképző Vállalat Budapesti Irodája**
3.1, 3.2
12. **OKTÁV Ipari Továbbképző Vállalat – Esztergom-kertváros**
2.1, 3.1, 3.2
13. **KLTE Matematikai Intézete – Debrecen**
1.2, 2.22, 3.1, 3.2
14. **Számítástechnikai Alkalmazási Vállalat Oktatási Iroda
– Budapest (az 1990/91-es tanévtől)**
1.1, 1.2, 2.1, 1.21, 2.22, 3.1, 3.2, 3.3, 3.4,
4.1, 4.2
15. **KSH-SzÜV Békéscsabai Számítóközpontja**
1.1, 2.1, 3.1, 3.2, 3.3, 4.1
16. **KSH-SzÜV Győri Számítóközpontja**
3.1, 3.2
17. **KSH-SzÜV Kaposvári Oktatóközpontja**
*Jogutód: KSH-SZÜV Kaposvári Számítóközpont Oktatási
Iroda*
1.1, 1.2, 2.1, 3.1, 3.3
18. **KSH-SzÜV Nyíregyházi Számítóközpontja**
1.1, 1.2, 2.1, 2.22, 3.1, 3.2
19. **KSH-SzÜV Székesfehérvári Számítóközpontja
– a SZÁMALK megyei képviselője**
1.1, 3.1, 3.3
20. **KSH-SzÜV Miskolci Számítóközpontja**
1.1, 2.1, 3.1, 3.3
21. **KSH-SzÜV Pécsi Számítóközpontja**
1.1, 1.2, 2.1, 2.21, 2.22, 3.1, 3.2
22. **KSH-SzÜV Salgótarjáni Számítóközpontja**
3.1

23. **KSH-SzÜV Szekszárdi Számítóközpontja**
2.1, 3.1
24. **KSH-SzÜV Szombathelyi Számítóközpontja**
1.2, 2.1, 2.22, 3.1, 3.2
25. ***KSH-SzÜV Soproni Számítóközpontja**
3.1, 3.2
26. **KSH-SzÜV Zalaegerszegi Számítóközpontja**
1.2, 3.1, 2.1, 2.21, 2.22, 3.2, 3.3
27. ****Széchenyi István Közgazdasági Szakközépiskola – Nyíregyháza**
2.1, 3.1, 3.2, 3.3
28. **SZTÁV ÁT BUDAPEST-EMDEN Kft**
3.1, 3.2, 2.1, 3.3
29. **SZTÁV-SZAKSTUDIUM Szakmai Oktatási és Átképzési Kft
– Pécs**
3.1, 3.2, 3.3
30. **KSH-SZÜV Egri Számítóközpontja**
Jogutód: SZÜV RUTIN Számítástechnikai Szolgáltató Kft
3.1, 3.2
31. **SZÜV LÍCIUM Rt – Debrecen**
3.1, 3.2
32. **ÉGSZI DÉLSZÁM Dél-Dunántúli Szervező Leányvállalat**
*Jogutód: ÉGSZI DÉLSZÁM Szervező és Számítástechnikai
Kft – Pécs*
1.1, 2.1, 3.1, 3.2
33. **Veszprémi Vegyipari Egyetem**
Jogutód: Veszprémi Egyetem
2.22, 3.2
34. ***HUNORG Számítástechnikai és Szolgáltató Kft - Békéscsaba**
2.1, 2.22, 3.2

35. **SZTÁV-PROGRESS Oktatási, Továbbképzési és Szolgáltató Kft – Miskolc**
3.1, 3.2, 3.3
36. ***COOPINFORM Számítástechnikai Vállalat – Budapest**
1.1, 1.2, 3.1, 3.4
37. **TALENTUM Gmk**
Jogutód: TALENTUM Oktatásszervező, Szolgáltató-Kereskedelmi Kft – Budapest
3.1, 3.2
38. **SZTÁV Szakmai Továbbképző és Átképző Vállalat – Budapest**
2.1, 3.2, 3.3
39. **ZALASZÁM**
Jogutód: ZALASZÁM Szervezési, Számítástechnikai, Kereskedelmi Kft – Zalaegerszeg
1.1, 2.1, 3.1, 3.2
40. **TRANSZFER Kereskedelmi és Szolgáltató Kft – Nyíregyháza**
2.1, 3.1, 3.2, 4.1
41. **Szakszervezetek Csongrád megyei Könyvtára – Szeged**
3.1, 3.2, 3.3
42. **Pénzügyi és Számviteli Főiskola – Budapest**
1.1, 1.2, 2.22, 3.1, 3.2
- 42.a) **Pénzügyi és Számviteli Főiskola Salgótarjáni Intézete**
2.22
43. **AGYAR HONVÉDSÉG Elektro- és Mechanikai Műszerész Tiszthelyettesképző Szakközépiskola – Tápiószecső**
3.1, 3.2
44. ***KOPINT-DATORG Oktatási Leányvállalat – Budapest**
2.21, 2.22, 3.2, 3.3
45. ***HAJDÚ VOLÁN ELEKTRONIKA Kft – Debrecen**
3.1, 3.2, 4.1

46. **KONTORG Gépipari Tervező és Szervező Kft – Debrecen**
3.1
47. **Agrártudományi Egyetem Mezőgazdasági Kar – Debrecen**
3.1, 1.1, 1.2, 2.1, 2.21, 2.22, 3.2, 3.3, 3.4,
4.1, 4.2
48. ***VIDEOTON Ipari Rt Számítástechnikai Gyára – Székesfehérvár**
1.1, 2.1, 2.21, 3.1, 3.2, 3.3, 3.4, 4.1, 4.2
49. **MICROSYSTEM Műszaki és Számítástechnikai Rt – Budapest**
2.1, 3.1, 3.2, 3.3, 3.4, 4.1
50. **SZOFI BT – Szeged**
2.1, 2.21, 3.2, 3.3
51. **PHOENIX Informatikai Oktató és Szolgáltató BT – Budapest**
1.1, 1.2, 3.1, 3.2, 3.3
52. ***SAVARIA Közlekedésgépészeti Szakközépiskola – Szombathely**
1.1, 2.1, 3.1, 3.2, 3.3, 3.4, 4.1
53. **Közgazdasági Szakközépiskola – Sátoraljaújhely**
1.1, 3.1
54. **Kandó Kálmán Villamosipari Műszaki Főiskola Matematikai és Számítástechnikai Intézete – Budapest**
1.1, 1.2, 2.1, 2.21, 2.22, 3.1, 3.2, 3.3, 3.4,
4.1, 4.2
55. ***Általános Iskola-Bpest. (XIII. Gyöngyösi stny.7.)**
3.1
56. **R-TREND Számítástechnikai és Ügyviteli Kft - Szombathely**
2.1, 3.1
57. **Gödöllői Agrártudományi Egyetem Vállalatgazdasági Üzemmérnöki Intézete – Gyöngyös**
2.1, 3.1, 3.2

58. **Táncsics Mihály Közgazdasági és Kereskedelmi Szakközépiskola – Salgótarján**
3.2
59. **MARKETING Kereskedelmi Ügynökség – Veszprém**
3.1, 3.2, 3.3
60. ***BM Győr-Moson-Sopron megyei Területi Államháztartási és Információs Szolgálat – Győr**
1.1, 1.2, 2.1, 2.21, 2.22, 3.1, 3.2, 3.3, 3.4, 4.1, 4.2
61. **Türr István Közgazdasági és Postaforgalmi Szakközépiskola – Baja**
3.1, 3.2
62. ***ÉGSZI Gazdaságsszervező Intézet – Budapest**
1.1, 1.2, 2.1, 2.21, 2.22, 3.1, 3.2, 3.3, 3.4, 4.1, 4.2
63. ***Gépészeti Szakközépiskola – Békéscsaba**
3.1
64. **Széchenyi István Közgazdasági és Közlekedési Szakközépiskola – Hatvan**
3.2
65. **Táncsics Mihály Építőipari Szakközépiskola és Szakmunkásképző Intézet – Veszprém**
3.1, 3.2
66. ***COMPUTER PRAXIS Kft - Miskolc**
1.2, 2.1, 2.21, 3.1, 3.3, 4.1
67. ***Jász-Nagykun Szolnok megyei Önkormányzat Oktatási és Továbbképző Intézete – Szolnok**
3.1, 3.2
68. **Bánki Donát Ipari Szakközépiskola és Szakmunkásképző Intézet – Ajka**
3.1, 3.2, 4.1

69. **Békéscsabai Tanítóképző Főiskola**
3.1, 3.2
70. **PANNON Agrártudományi Egyetem Állattenyésztési Kar**
– Kaposvár
1.1, 2.22, 3.2
71. ***GATE Vezető- és Továbbképző Intézet – Budapest**
2.22, 3.1, 3.2,
72. **GRAF - GET Kft – Győr**
3.2
73. **BME Villamoskari Számítógép Központ – Budapest**
3.1
74. **Mathiász János Mezőgazdasági Szakközépiskola és Szak munkás-
képző Iskola – Boglárlelle**
2.1, 3.1, 3.2
75. **REND-SZÁM Rendszerszervezési és Számítástechnikai Oktató BT**
– Budapest
1.1, 1.2, 2.1, 2.22
76. ***Északmagyarországi Vegyiművek Humánpolitikai Önálló Osztály**
– Sajóbábony
3.1, 3.2
77. **Rózsa Ferenc Gimnázium – Békéscsaba**
2.1, 3.1, 3.2, 3.3
78. ***Neumann János Számítógéptudományi Társaság Borsod-Aba-Új-
Zemplén megyei Szervezete – Miskolc**
2.1, 2.21, 2.22, 3.1, 3.2,
79. **CONTROL Elektronikai és Számítástechnikai Rt – Budapest**
2.1, 2.21, 2.22, 3.1, 3.2,
80. **Szent István Közgazdasági Szakközépiskola és Kollégium**
– Budapest
3.1, 3.2

81. Apáczai Csere János Nevelési Központ Gimnázium – Pécs
3.1, 3.2
82. Gimnázium, Műszaki Szakközépiskola és Kollégiuma – Újszász
3.1, 3.2
83. *Széchenyi István Közgazdasági és Külkereskedelmi Szakközépiskola – Békéscsaba
3.1, 3.2
84. Kemény Gábor Műszaki Szakközépiskola - Békéscsaba
2.1, 3.1, 3.2
85. RECORD - System Számítástechnikai, Kereskedelmi és Szolgáltató Kft – Kecskemét
1.1, 1.2, 2.1, 2.21, 2.22, 3.1, 3.2, 3.3, 3.4
86. Szombathelyi Közgazdasági Szakközépiskola
3.1
87. SZTÁV Délmagyarországi Oktatásszervező, Kereskedelmi és Szolgáltató Kft – Szeged
3.1, 3.2
88. BM Polgári Védelem Országos Vezetőképző Intézet – Pécel
2.1, 3.1, 3.2
89. Móra Ferenc Szakközép- és Szakmunkásképző Iskola – Szeged
3.1
90. **Műszertechnika Békéscsaba Kft
4.1
91. Mechatronikai Szakközépiskola és Gimnázium – Budapest
3.1, 3.2
92. **GORDEX Adatfeldolgozó, számítástechnikai és kereskedelmi BT – Eger
3.1
93. KÜLKERINFO Oktatási és Továbbképző Kft - Debrecen
3.1, 3.2

94. **PATE GEORGIKON Mezőgazdaságtudományi Kar, Informatikai és Számítástechnikai Központ – Keszthely**
1.2
95. **Tóth Kálmán Szakközépiskola és Hermann Ottó Kollégium – Baja**
3.2
96. **SZÁMALK Menagement Kft – Budapest**
1.1, 1.2, 2.2, 2.21, 2.22, 3.1, 3.2, 3.3
97. **SZÁMALK Kelenföldi Kft – Budapest**
1.1, 1.2, 2.1, 2.21, 2.22, 3.1, 3.2, 3.3
98. **PERFEKT - UNIVERSITAS Oktatási Centrum – Pécs**
3.1, 3.2
99. **CD-2000 Alapítvány a Pályakezdő Fiatalokért – Budapest**
3.2
100. **Gábor Áron Műszaki Középiskola – Miskolc**
3.1, 3.2
101. ****DATACOOOP DEVELOPMENT – Budapest**
3.1, 3.2
102. **Neumann János Közgazdasági Szakközépiskola – Budapest**
3.1
103. **Radnóti Miklós Gimnázium – Jánoshalma**
3.1, 3.2
104. **Mechwart András Gépipari Műszaki Középiskola – Debrecen**
3.1, 3.2
105. **PATRONUS Vállalkozás-Oktatásszervező és Szolgáltató BT – Kazincbarcika**
3.1, 3.2
106. **DIDAKT Betéti Társaság – Budapest**
3.1, 3.2

107. Pollák Mihály Műszaki Szakközépiskola – Szentes

3.1, 3.2

108. GOLD STANDARD Kft – Debrecen

3.1, 3.2

109. HARKÁLY Magániskola – Budapest

3.1

110. INSTALL Informatikai Stúdió – Pécs

1.2, 2.11, 2.22, 3.1, 3.2

Megjegyzés: az 1.1, 1.2, továbbá a 2.11, 2.22 jelű szaktanfolyamok csak az oktatást szervező intézmény által kidolgozott konkrét képzési program előzetes jóváhagyását követően indíthatók.

A * - gal jelzett szervezetek a KSzSzJ-be történt felvételük óta nem jelezték szaktanfolyam beindítását.

A ** -gal jelzett szervezeteknél szaktanfolyamok beindítása eddig, az intézményen kívül álló okból nem volt lehetséges.

Szám:

..... /199

(=törzskönyv, törzslap száma)

A Szakképzési Jegyzékbe felvett intézmény cégbélyegzője

I G A Z O L Á S TANTÁRGYI VIZSGÁK LETÉTELÉRŐL

A –/199 KSH sz. engedély alapján igazolom, hogy
....., akinek személyi száma:, a * megnevezésű számítástechnikai szaktanfolyam kerettantervében előírt tantárgyi vizsgák követelményeit a következők szerint teljesítette:

A tantárgy megnevezése és összóraszám	A v i z s g a			hivatalos bejegyzések
	kelte	eredménye		
		fb./gyak.	szb.	
	xx/	xx/	xxxx/	
		xxx/		

Ez az igazolás az itt megjelölt szaktanfolyamra épülő számítástechnikai szakmai képesítő vizsga három éven belül történő letételére jogosít fel.

..... 19 hó napján

P.H.

.....
(intézményvezető)

- * A szaktanfolyam (típus) megnevezése.
- */ Az írásbeli és szóbeli vizsgára adott érdemjegyekből számítva.
- **/ Az érdemjegy (1-5-ig) és a vizsgáztató tanár névjele.
- *** / A KSH-szabályzat III. rész 1./ g) pontjában szereplő "mentesítés"-t vagy "különbözeti vizsgá"-t a megfelelő tantárgyi vizsgát megjelölve - azzal egy sorban "Fm." vagy "Kv." + az érdemjegy és névjele bejegyzésével - , a vizsgáztató igazolja.
- **** / A vizsgadíj befizetett összege és a tanfolyamfelelős (oktatási előadó) névjele.

Szám:
/19
 (= törzskönyv, törzslap száma)

A szakmai képesítő vizsgát szervező intézmény
 cégbélyegzője

ZÁRÓ BIZONYÍTVÁNY

A – /199 KSH sz. engedély alapján igazolom, hogy
 , akinek személyi száma:
 , a*/ szakmai képesítés
 megszerzésére felkészítő szaktanfolyam kerettantervében kijelölt tantárgyi
 vizsgákat.

	oktató névjele
..... tantárgyból eredménnyel
..... tantárgyból eredménnyel
..... tantárgyból eredménnyel

stb.

letette.

A fentiek alapján minősítéssel **alapfokú számítástechnikai szakképesítést** szerzett.

.....199 hó napján

P.H.

.....
 (intézményvezető)

*/ A szakképesítés megnevezése

Szám: /19
(= törzskönyv, törzslap száma)

A szakmai képesítő vizsgát szervező intézmény cégjegyzője

KÉPESÍTŐ BIZONYÍTVÁNY

....., akinek személyi száma: , a - .. /199 KSH sz. engedély előírásainak megfelelően lefolytatott számítástechnikai szakmai képesítő vizsga követelményeinek

..... minősítéssel tett eleget és ennek alapján */ / középfokú szakképesítést szerzett.

..... 199 hó napján

P.H.

.....
a Szakmai Képesítő Bizottság
elnöke

.....
vizsgáztató

.....
vizsgáztató

*/ A szakképesítés megnevezése

Szám:
/199
 (= törzskönyv, törzslap száma)

A szakmai képesítő vizsgát
 szervező intézmény cégbélyegzője

KÉPESÍTŐ OKLEVÉL

....., akinek személyi száma:
, a – .../199 KSH sz. engedély alapján
 lefolytatott számítástechnikai szakmai képesítő vizsga követelményeinek

..... minősítéssel tett eleget és

ennek alapján */ / felsőfokú
 szakképesítést szerzett.

.....199 hó napján

P.H.

.....
 a Szakmai Képesítő Bizottság
 elnöke

.....
 vizsgáztató

.....
 vizsgáztató

*/
 A szakképesítés megnevezése

SZÁMÍTÁSTECHNIKAI KONZULTÁCIÓS HÁLÓZAT/A

Sz.	Helység	Iskola neve	Címe	Telefonszáma	Szervező neve	Beosztása
1.	Barcs	Erdészeti Vízépítési és Vízgazdálkodási Szakközépiskola	7571 Barcs Lattinca u. 13.	83/12-611	Kovács László	tanár
2.	Békéscsaba	Széchenyi I. Közgazdasági és Kereskedelmi Szakközépiskola	5600 Békéscsaba Andrássy út 1.	66/22-611	Kovács László	tanár
3.	Budapest	Belvárosi Egészségügyi Szakközépiskola és Szaktiskola	1056 Budapest Váczl u. 47.	1/1186-010	Holczinger Istvánné	munkaközösség- vezető
4.	Budapest	Bólyai János Elektronikai Szakközépiskola	1134 Budapest Váci út 21.	1/1208-233	Király László	tanár
5.	Budapest	Zs. Borda András Szakközép- és Szakmunkásképző Iskola	1097 Budapest Tímóti u. 3.	1/1476-341	Fábián Zoltán	tanár
6.	Budapest	Corvin Mátyás Gimnázium és Műszaki Szakközépiskola	1165 Budapest Mátyás király tér 4.	1/2529-900	Dr. Juhász Gábor	igazgató- helyettes
7.	Budapest	Idegenforgalmi Szakközépiskola és Szálloda	1039 Budapest Szt. János u. 16.	1/1886-167	Fenyő Balázs	tanár
8.	Budapest	II. Rákóczi Ferenc Gyakorló Közgazdasági Szakközépiskola	1075 Budapest Wesselényi u. 38.	1/1421-984	Terémyiné Szabó Mónika	iskolafőnök
9.	Budapest	Petrík Lajos Vegyipari Szakközépiskola	1146 Budapest Thököly u.48-54	1/1214-810	Bertalan Zsolt	igazgató- helyettes
10.	Budapest	Vágl István Építőipari Műszaki Középiskola	1015 Budapest Szabó I. u. 2-4.	1/2011-529	Varga László	munkaközösség vezetője
11.	Debrecen	Erdely-Grúz Tilbor Vegyipari Szakközépiskola	4000 Debrecen Csapó u. 29-35.	52/13-023	Kovács Gábor	tanár
12.	Debrecen	Irtinyi János Élelmiszeripari Szakközépiskola	4024 Debrecen Irtinyi u. 1.	52/49-899	Szakáts István	tanár
13.	Debrecen	Mechwart András Műszaki Középiskola	4025 Debrecen Széchenyi u. 58.	52/13-499	Gyehvai Károly	vezetőtanár
14.	Dunaújváros	Rudas László Közgazdász és Külkereskedelmi Szakközépiskola	2403 Dunaújváros Rómal krt 47.	25/23-894	Kondor László	igazgató - helyettes
15.	Eger	Egri Közgazdasági Szakközépiskola	3300 Eger Klapka u. 7.	36/12-304	Katona János	gyakorlati oktatásvezető
16.	Győr	Jedlik Ányos Ipari Szakközép- és Szaktiskola	9021 Győr Szent István u.7.	96/28-842	Hajdú Elemér	igazgató- helyettes

SZÁMÍTÁSTECHNIKAI KONZULTÁCIÓS HÁLÓZAT/B

Sz.	Helység	Iskola neve	Címe	Telefonszáma	Szervező neve	Beosztása
17.	Kaposvár	Munkácsy Mihály Gimnázium és Egészségügyi Szakközépiskola	7401 Kaposvár Kossuth L. u. 48.	82/21-711	Puspán Ferenc	tanár
18.	Kecskemét	Szent-Györgyi Albert Egészségügyi Szakközépiskola	6000 Kecskemét Nyíri u. 73.	76/22-020	Katona Viktor	tanár
19.	Komló	Kazinczy Ferenc Szakközépiskola	7300 Komló Alkotmány u. 2.	72/62-861	Fekete György	tanár
20.	Miskolc	Miskolc Gépipari Műszaki Középiskola	3530 Miskolc Soltész Nagy Kálmán u. 10.	46/345-811	Losonc János	tanár
21.	Nagykanizsa	Cserháti Sándor Mezőgazdasági és Gépészei Szakközépiskola	8802 Nagykanizsa Ady E. u. 7/4/a.	92/73-106	Janzsó Antal	igazgató
22.	Nagykőrös	Toldi Mihály Élelmiszeripari Szakközépiskola	2750 Nagykőrös Cegléd u. 24.	20/51-922	Jánvós László	igazgató- helyettes
23.	Nyíregyháza	Kereskedelmi és Vendéglátóipari Szakközépiskola	4400 Nyíregyháza Krúdi Gy. u. 32.	42/12-244	Fehér Sándor	tanár
24.	Nyíregyháza	Széchenyi István Közgazdasági Szakközépiskola	4400 Nyíregyháza Városmajor u. 4.	42/11-841	Héli Géza	munkaközösség- vezető
25.	Pécs	Zipernovszky Károly Ipari Szakközépiskola	7622 Pécs 48-as tér 2.	72/12-344	Peitkán István	tanár
26.	Salgótarján	211.sz. Ipari Szakmunkásképző és Szakközépiskola	3100 Salgótarján Csokonai u. 21-29.	32/16-233	Farkas László	igazgató- helyettes
27.	Salgótarján	Táncsics Mihály Közgazdasági és Kereskedelmi Szakközépiskola	3100 Salgótarján Május 1. u. 51-58.	32/11-671	Andrásné Nagy Izabella	gyakorlati oktató-vezető
28.	Szeged	Bebribis Lajos Szakközépiskola és Kollégium	6724 Szeged Marx tér 14.	62/21-433	Vida Attila	tanár
29.	Szeged	Csonka János Gimnázium és Szakközépiskola	6726 Szeged Temesvári krt 38.	62/13-185	Horváth Zoltán	tanár
30.	Szeged	Csonka János Gimnázium és Szakközépiskola Gimnáziumi tagozat	6726 Szeged Bai fasor 39-54.	62/13/033	Biró Margit	tanár
31.	Székesfehérvár	Széchenyi I. Műszaki Szakközépiskola	8001 Székesfehérvár Buda u. 45.	22/15-330	Dominó Csaba	tanár
32.	Szelecsárd	505. sz. Ady Endre Szakmunkásképző Intézet és Szakközépiskola	7100 Szelecsárd Széchenyi u. 2-14.	74/12-318	Molnár István Tóth Péter	tanár tanár

SZÁMÍTÁSTECHNIKAI KONZULTÁCIÓS HÁLÓZAT/C

Sz.	Helység	iskola neve	Címe	Telefonszáma	Szervező neve	Beosztása
33.	Szakszárd Potárok	Csapó Dániel Mezőgazdasági Szakközépiskola	7100 Szakszárd-Potárok Pf.: 61.	74/11-277	Ózsa József	tanár
34.	Szolnok	Pótfly János Műszeripari és Vegyipari Szakközépiskola	5001 Solnok Tiszaparti sétány 2.	56/34-955	Szűcs Gyula	igazgató - helyettes
35.	Szolnok	Vásárhelyi Pál Kézagazdasági és Postaforgalmi Szakközépiskola	5000 Solnok Baross G. út 43.	56/42-344	Dr. Kallcz Éva	igazgató - helyettes
36.	Szolnok	Környezetgazdálkodási és Építőipari Szakközépiskola	5000 Solnok Bajcsy-Zs. út 2.	56/30-010	Bertalan István	tanár
37.	Szombathely	Szombathelyi Gépipari Műszaki Szakközépiskola	9701 Szombathely Rohonczi út 1.	94/12-477	Horváth István	munkaközösség- vezető
38.	Tiszaföldvár	Hajmáczy József Gimnázium Pedagógiai és Szociális Asszisztensképző Szakközépiskola	5430 Tiszaföldvár Kossuth út 100-104.	88-38	Szankó Tibor	tanár
39.	Tiszavasvári	115. sz. Vasvári Pál Ipari Szakközépiskola	4400 Tiszavasvári Petőfi u. 1.	42/72-458	Kosztia Mihály	tanár
40.	Tördözenimiklós	Székács Elemér Mezőgazdasági Szakközépiskola	5201 Tördözenimiklós Almásy út 50.	56/30-533	Stell József	tanár
41.	Veszprém	Ipari Szakközépiskola	8201 Veszprém iskola u. 4.	80/24-809	Lázár István	igazgató - helyettes

Ajánlott irodalom

A személyi számítógép és használata; az operációs rendszer

1. *Ferenczy Imre: DOS 3.30 operációs rendszer (PC-DOS és MS-DOS) Referenciakártya. SZÁMALK*
2. *Kocsis Zoltán–Gazsó Zoltán: PC start (Az IBM kompatibilis személyi számítógép kezelése. INNOTECH Műegyetemi Innovációs Park Kft., 1992.*
3. *Lángos István: Bevezetés az IBM PC XT/AT DOS-ba. Novotrade Kiadó Kft.,1992.*
4. *Lebovitsné–Kiss Z.–Dr. Tamás P.–Tóth B.: MS-DOS 5.0. ComputerBooks, 1992*
5. *Peter Norton: DOS-kalauz. Novotrade Kiadó – Brady Books, 1990*
6. *dr. Pajor Gábor: Az IBM PC-ről alapfokon – A hardver. LSI Oktatóközpont, 1991.*

Szövegszerkesztés számítógéppel

1. *F. Ható Katalin: Word 3.0 4.0 5.0 (Computer Books – aPLUS Kft, 1992)*
2. *Molnár Mátyás: Word 5.5 (Computer Books, 1992)*
3. *Szövegszerkesztés WordPerfect-tel (Accent-Software, 1991)*
4. *Szövegszerkesztők az IBM PC számítógéphez: Professional Editor, Personal Editor, Kedit (LSI Oktatóközpont, 1990)*

Táblázatkezelés számítógéppel

1. *Dr. Ferenczy Antal Zoltán: Lépésről-lépésre a QUATTROban (ComputerBooks,1992)*
2. *Ferenczy Imre–Gerő Judit: Quattro (SZÁMALK, 1991)*
3. *Bill Harrison: Bevezetés a Framework III használatába (Novotrade Kiadó, 1991)*
4. *Gerő Judit: Lotus 1-2-3 és SYMPHONY (SZÁMALK, 1986)*

Adatbáziskezelés számítógépen

1. *Bognár Júlia: dBASE III PLUS* (Budapest. SZÁMALK, 1992. 255 p.)
2. *Schellenberger, W.: dBASE IV+dBASE IV 1.1* (Budapest, Műszaki K. 1991, 136 p.)
3. *Szlovák Gábor–Tóth Tibor–Köri Gábor: Adatbáziskezelés, programozás dBASE IV-ben I-II. köt.* (Budapest, LSI ATSZ, 1989. 456 p.)
4. *Sályi János–Szelezsán János: Adatbázisok* (Budapest, SZÁMALK, 1991. 102 p.)

Betekintés a számítógépek ipari alkalmazásába

Horváth Péter–Nagy Lajos: Intelligens gépek, robotok c. sorozat

1. **Az intelligens gép**
2. **Bemeneti jelforrások**
3. **Bemeneti egységek**
4. **Kimeneti egységek**
5. **A robotok**

LSI Alkalmazástechnikai Tanácsadó Szolgálat – Budapest, 1989.
(A 7. fejezet ezen kiadványok alapján készült.)

Tartalomjegyzék

Előszó	1
Néhány gyakorlati tanács a könyv használatához	3
1. A személyi számítógép és használata; az operációs rendszer – Kócy A. Judit	
A számítógépről – röviden	6
A legfontosabb perifériatípusok	9
Milyen számítógépet vegyünk?	16
Az operációs rendszer	17
Parancsok	23
2. Microsoft Windows 3.0 – Reich Gábor	
Mi a Windows?	31
A Program Manager	34
File Manager	39
Print Manager	44
3. Szövegszerkesztés számítógéppel – Gerő Judit–Reich Gábor	
Szövegszerkesztőről általában	53
A szöveg beírása	56
A szöveg javítása	59
Mozgás a szövegben és a szövegekijelölés	60
A szöveg másolása, mozgatása, törlése	62
A szöveg keresése, helyettesítése	65
Állományműveletek	67
A szöveg formázása	71
Nyomtatás	79
Ízelítő	81
4. Táblázatkezelés számítógéppel – Gerő Judit	
A táblázatkezelő programok általános ismertetése	83
A képernyő és a táblázat ismertetése	85
A táblázatok elmentése, visszahívása	109
A táblázat formázása	110
A nyomtatás	112
Egyéb szolgáltatások	114
5. Adatbáziskezelés számítógéppel – Péntes B. Zsuzsa	
A számítástechnika és az adatok nyilvántartása	119
Az adatbáziskezelők általános ismertetése	120
Az adatbáziskezelők szolgáltatásai	120

Az adatbázis fogalma és szerkezete	121
A dBASE típusú adatbáziskezelő rendszerek	123
Az adatbázis-adatállományok használata	130
Az adatvisszakeresés alapjai	134
Az adatkezelés alapjai	141
Az adatok rendezése	145
Nyomtatás	153
6. Számítógép-hálózatok és alkalmazásaik – Brückner Huba	
A számítógép-hálózatok fejlődéstörténete	155
A hálózatok kiépítése	157
Átvivőközegek jellemzői	159
Az üzenet továbbítása	162
Helyi hálózatok, hálózati szoftverek	164
A távolsági hálózatok	172
Nyílt rendszerek – OSI	174
ISDN	179
Teletext és videotex	183
7. Betekintés a számítógépek ipari alkalmazásába – Nagy Lajos	
A gépi intelligencia	189
Robotok	196
Robotjellemzők	200
A robot vezérlése és programozása	204
Az intelligens gépek és a robotok alkalmazásai	209
8. Számítástechnikai szaktanfolyami oktatás és a szakmai képesítő vizsgáztatás rendszere – Bánhidai Ágnes	
A számítástechnikai szakfeladatok központi koordinációja.	211
Ami a "hőskorszak"-ot követte: a "szakképesítést nyújtó" tanfolyami oktatás.	213
A számítástechnikai tanfolyami szakemberképzés jelenlegi rendszere	214
Profik legyünk a számítástechnikában vagy alkalmazók?	218
A szakképesítés megszerzésére felkészítő tanfolyamok szervezett rendszere.	219
Mit kell tudni a szakmai képesítő vizsgáztatásról?	221
Melléklet	224
Számítástechnikai konzultációs hálózat	248
Ajánlott irodalom	251

brill

OKTATÁSTECHNIKA

PROFESSZIONALISTA TECHNIKA AZ OKTATÁSBAN

20 év garancia

a pormentes és krétás
táblafelületekre

Pormentes fehér és színes kréták

Tölthető filctollak a fehér táblákhoz, utántöltő folyadék

Minden táblafelületen használható mágneses tapadó-korong

LC-display a számítástechnika oktatásához

Hidegfényű írásvetítők az LCD-hez

Dönthető nagy fényvisszaverő képességű diffúz vetítőfelületek

Mozgatható táblarendszerek

COOMBER hangtechnika, hangosításhoz, nyelvtanuláshoz

Vezetéknélküli mikrofonok

Telefonok és alközpontok, faxok

Számítógépek, nyomtatók, számítógéphálózatok

Szoftverek (iskolák számára kedvezményel)

Német gyártmányú iskolabútorok

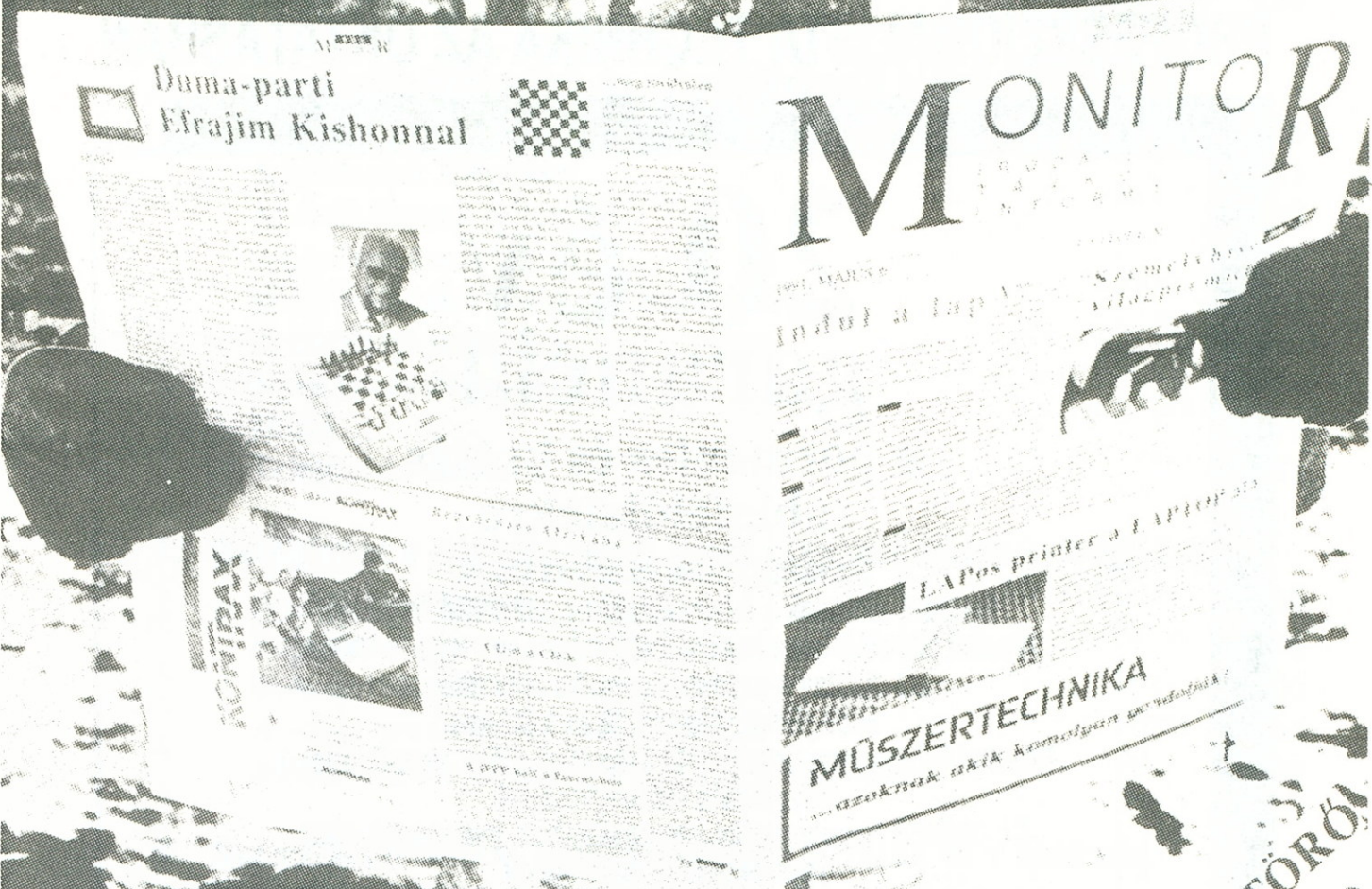
Nagy fényerejű és képernyőjű televíziók

Nagyteljesítményű fénymásolók

Írásvetítő fóliák nyomtatókhoz és fénymásolóhoz

BRILL
OKTATÁS ÉS INFORMÁCIÓTECHNIKA G.M.
H-1136 Budapest, Pannónia u. 48.
telefon: (36-1) 1-299-991
fax: (36-1) 1-400-536

MINDEN SZAKLAP SZÉP, VASTAG ÉS MÉLYENSZÁNTÓ, KIVÉVE A ...



EZ GYORS, OLCSÓ ÉS LÉNYEGRE TÖRŐ

260,- Ft