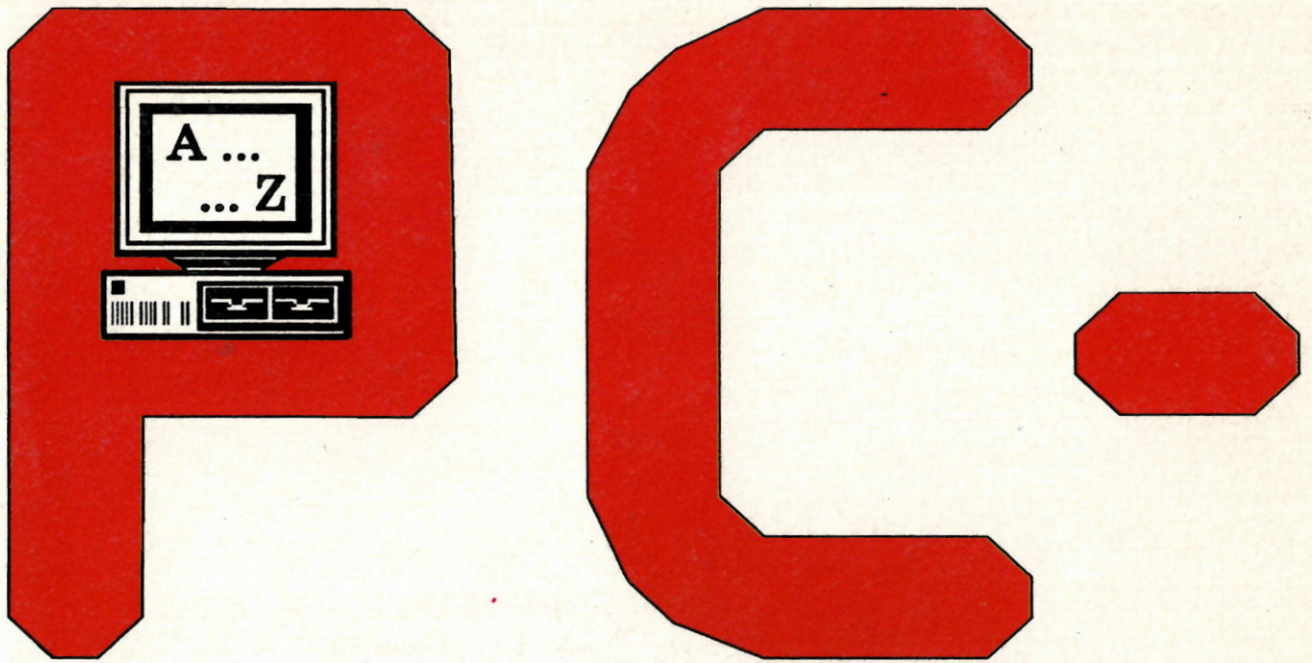


FARKAS ERNŐ

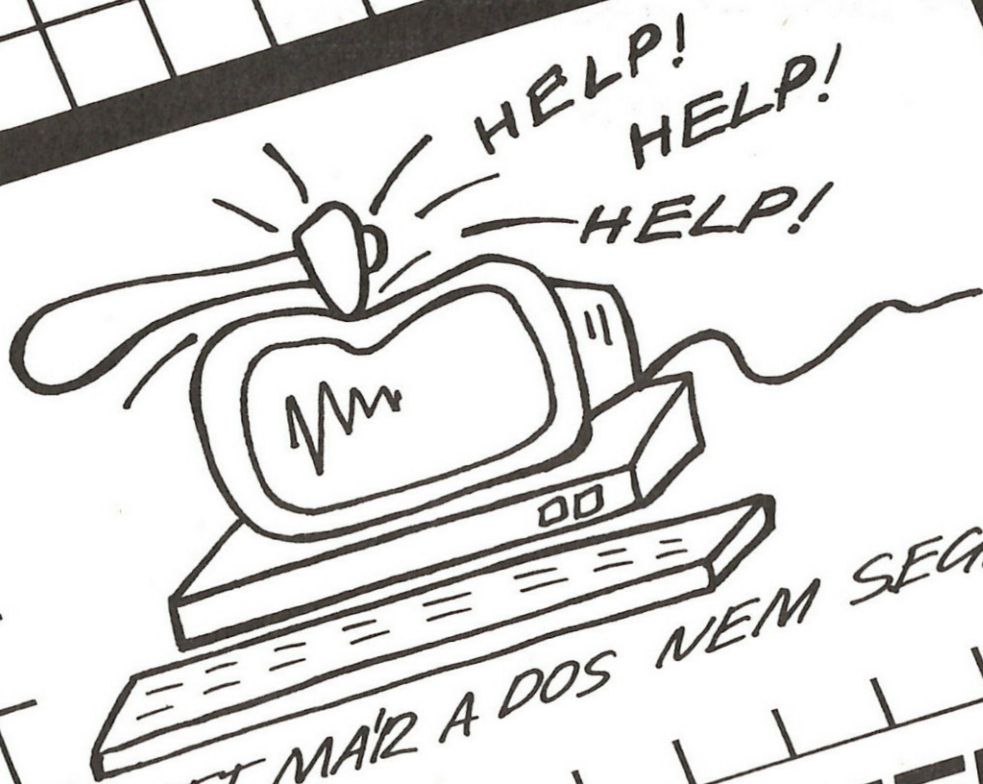


SZÓTÁR

ALAPLAP KÖNYVEK 

FARKAS ERNŐ—CSÓRIÁN SÁNDOR

PC-szótár



PC GYORSSZERVIZ

- ★ CSEREGÉPEKSEL
- ★ ALKATRÉSZBIZTOSÍTÁSSAL
- ★ ÁTALÁNYDÍJAS RENDSZERREL

TRIGON

TRIGON HARDWARE KFT.

BEMUTATÓTEREM ÉS IRODA:
1031 - BP. KADOSA U. 57
TEL: 160-74-57

SZERVIZ:
1112 - BP. BODAJK U. 29
TEL.: FAX: 185-82-93

FARKAS ERNŐ—CSÓRIÁN SÁNDOR

PC-szótár



CÉDRUS KIADÓ

Sorozatszerkesztő: Faklen Pál
Szaklektorok: Brüll Károly, Szentmiklóssy Zoltán
Szerkesztette: Jakab Ágnes

ISSN 0866—434X
ISBN 963 7429 28 X

© Farkas Ernő, Csórián Sándor, 1992



Cédrus Kiadó Kft
Felelős kiadó: Sebestyén Ilona ügyvezető igazgató
Tördelőszerkesztő: Angely Imre
Nyomta és kötötte az Alföldi Nyomda
A nyomdai megrendelés törzsszáma: 8656.66-13-2
Készült Debrecenben az 1993. évben
Felelős vezető: György Géza

Tudnivalók a szótár használatához

Mint számos más szótárban, itt is előfordul, hogy egyes szócikkek nemcsak a címükben foglalt fogalma(ka)t magyarázzák meg, hanem más, azonos jelentésű kifejezés(eke)t, továbbá másutt leírt fogalmakat is. Ilyenkor terjedelmi okokból az illető (kapcsolódó) címszónál nem ismételjük meg a máshol mondottakat, csak annyit közlünk: Lásd Ez a megoldás ebben a szótárban kissé gyakrabban fordul elő, mint a szótárak többségében.

Ha egy szócikk fejében több címszó van megadva vesszővel elválasztva, ez azt jelenti, hogy ezek azonos vagy nagyon közeli fogalmak. A magyar szavak idegen megfelelőjét és fordítva: (ritkábban) az idegen szavak magyar megfelelőjét zárójelben adtuk meg — akárcsak a rövidítések feloldását. Mindezeket a szótárban végig szinonimákként, illetve szinonimaszerűen szerepeltettük, részint azért, hogy a stílus ne legyen túlságosan szegényes, száraz, részint azért, mert ezeket a szakmai köznyelv többé-kevésbé ugyanúgy, valóban befogadta már; — úgy gondoljuk, a szótár forgatása során ez nem okoz nehézséget, sőt: ezáltal is elősegítjük az értelmezések kikristályosodását. (Ahol az idegen szó fordítását nem tettük zárójelbe, ott a visszafordítása nem egyértelmű, vagyis a magyar szóról más szövegkörnyezetben nem lenne feltétlenül megállapítható, hogy milyen idegen szó helyett szerepel.)

A ► mutatót — a szócikk végén — olyképpen és akkor használtuk, ha további információk találhatóak a jelölt címszavaknál. Hogy könnyebb legyen az eligazodás a szótárban, a következő konvenciókat alkalmaztuk.

A szócikk megnevezése(i) a bekezdések elején mindig **vastag** (félkövér) betűvel van(nak) szedve. Ha a szócikken belül további fogalmak is definiálva vannak, akkor erre a tényre azok **vastag**, **dőlt** kiemelése utal.

Természetesen a szócikkek szövegében mindenütt szerepel(het)nek olyan fogalmak is, amelyeket a szótár tartalmaz, de nem itt magyaráz meg, hanem egy másik címszóban. Ezek közül a szótárban leggyakrabban előforduló szavakat/fogalmakat/kifejezéseket nem különbözteti meg semmi a szedésben — azonban ezeket itt alább felsoroltuk. A ritkább, kevésbé ismert, különösen specifikus, a szakterminológia szempontjából nagyon kötött jelentésű rész(ek)re azonban — csak a szócikken belüli első előfordulás(uk)ra — egyszerű **dőlt** betűvel hívjuk fel a figyelmet. (A magyar szavak és a magyaros írású szavak toldalékos alakban is lehetnek; így, ebben a formájukban vannak ilyenkor kurziválva; de természetesen az eredetit — a szótövet, esetleg többet is, ha összetettebb fogalomról van szó — kell keresni; ennek alapján talál rá az olvasó a szótárbeli magyarázatukra.)

Ha tehát ránézünk egy szócikkre, akkor azonnal szembeötlik: a benne szereplő, **dőlt** betűs szavak/fogalmak/kifejezések (toldalék nélküli vagy esetleg másképpen

toldalékolt formában) megtalálhatók a szótárban. A vastag betűsek pedig éppen itt vannak definiálva.

A következő (szak)szavak azok, amelyeket nem jelöltünk meg a szövegekben, mert egyrészt úgy hisszük, az olvasók többsége biztosan ismeri a — mellettük megadott oldalakon közölt — jelentésüket, másrészt rendkívül gyakoriak. (A / jelet a könyvben mindenütt vagy értelemben kell olvasni.)

(adat)állomány/fájl.....	11. old.
adatbázis.....	11. old.
bájt	18. old.
bemenet(i)/kimenet(i) és bemenő/kimenő és bevitel(i)/kivitel(i) és input/output és i/o vagy I/O	19., 54., 83. old.
busz/sín.....	22. old.
CPU/mikroprocesszor/processzor	27. old.
disk/diszk/floppy/lemez	32. old.
display/képernyő/kijelző/megjelenítő/monitor/screen	61. old.
directory/könyvtár.....	64. old.
DOS és operációs rendszer.....	32., 82. old.
felhasználó	40. old.
függvény/művelet/szubrutin	106. old.
grafika/grafikus	44. old.
hardver.....	48. old.
hálózat	48. old.
IBM PC, XT, AT	52. old.
interface/interfész	55. old.
ír(ás)/olvas(ás).....	39. old.
karakter és kód	60., 64. old.
kompatibilis/kompatibilitás	63. old.
memória/tár/tároló	70. old.
modul	73. old.
nyelv/programozási nyelv.....	90., 109. old.
nyomtató/nyomtatás.....	69. old.
parancs/utasítás	114. old.
periféria	88. old.
program és szoftver	90., 105. old.
regiszter	93. old.
rekord.....	94. old.
számítógép (computer).....	104. old.
változat/verzió	116. old.
vezérlés/vezérlő.....	117. old.
védelem	118. old.

Ha valaki úgy gondolja, hogy nem egészen biztos a jelentésükben, akkor — mielőtt belemélyed aktuális problémájának megfejtésébe, vagyis (szótárunk segítségével) éppen utána akar nézni valaminek, tanácsoljuk: — ezekkel kezdje.

Noha a fenti szavak sokszor szóösszetételekben szerepelnek, a magyarázatuk ahhoz már elegendő, hogy az összetétel által keletkezett új fogalom értelmezhető legyen. Néhány esetben ezeknek a — szóösszetétellel keletkezett — fogalmaknak is megtalálható a definíciója, esetleg az eredeti szócikk szövegében; ilyenkor természetesen a vastag, dőlt betűs kiemelés ott azonnal útbaigazít.

A könyv függelékeként az olvasó megtalálja a DOS-parancsok felsorolását egy táblázatban, mely a parancsok és az operációs rendszer változatainak összetartozását és kronológiáját tükrözi.

Természetesen egy ilyen munka hiába is törekedne teljességre, ez eleve reménytelen vállalkozás volna: olyan nagy a mozgás ezen a szakterületen, hogy a követésével nem lehet a szerkesztési és a nyomdai átfutást megelőzni. Arról nem is szólva, hogy senki sem képes az újabb és újabb fogalmakat — a tartalmukban néha nem is különböző, hanem csak az egyes szerzők/műhelyek saját munkanyelvét, divatozását tükröző elnevezéseket — naprakészen összeszedni.

Vannak azonban olyan „hiányosságai” is ennek a szótárnak, amelyek tudatos elhagyással magyarázhatók. A válogatás ilyen szempontból — megint csak természetesen — szubjektív, a szerzők, a lektorok és a szerkesztő megfontolásain alapul. Mindenesetre ajánljuk az olvasóknak, hogy amennyiben valamire itt nem bukkantak rá, vegyék kézbe a kiadó Számítástechnikai ALAP-lexikonának I. kötetét, mert abban az általános fogalmakat sikerrel kereshetik; továbbá e szótárban is szereplő bizonyos címszavak átfogó(bb) jelentése ott olvasható.

Még egy megjegyzés: a sorrendben a magyar ábécét tartottuk szem előtt, de ebben is van azért némi „kavarás”. A kis- és nagybetű között ugyanis nem tettünk különbséget; a sorrendezéskor a betű után következő első pozíción — a második/harmadik/negyedik stb. helyen — szereplő írásjelként nem a betűket néztük először, hanem a különleges eseteket vettük számba: tehát ilyenkor esetlegesen a kötőjelet vagy például az & jelet választottuk, függetlenül attól, hogy azt mi követi. Így szótárunkban például a **K&R** megelőzi a **kalkulátor-t**, a **C (nyelv)** a **cache (memory)-t**; és a szócikk megnevezésében lévő szóközre is ugyanígy kell figyelni: ha van benne, akkor ez előrehoz(hat)ta a címszót a sorrendben az ábécé szerint rendezett, aszerint következő elem(ek) elé.

Mivel ennek a könyvnek a tartalmát szeretnénk folyamatosan aktualizálni, a szócikkeket a PC-s élet változásaival összhangban finomítani, a szükséges újakkal bővíteni, minden érdemi észrevételt, javaslatot szívesen fogadunk, várunk.

A Szerkesztő

Processzorok

8080A: Az Intel cég 1973-ban kibocsátott 8 bites mikroprocesszora. Ez a cég készített először 8 bites mikroprocesszort, a 8008-ast; ennek továbbfejlesztése a 8080A, amely az első, tömeges méretekben alkalmazott CPU volt. A 8 adatvonalon át a 16 címvezetéken max. 64 kb-ot memóriát használhatott. Hátrányai mellett — például háromféle tápfeszültséget (–5 V, +5 V és +12 V) igényelt — már több olyan vezérlővonalal is rendelkezett, amelyek a mai mikroprocesszorokra is jellemzők.

8085: 1977-ben hozta ki az Intel ezt a 8 bites processzort, amely szoftverkompatibilis a 8080A-val, de lényegesen korszerűbb; ma is az egyik legjobb 8 bites CPU.

8086/8088: Az Intel 1978-ban kezdte gyártani első, 16 bites mikroprocesszorát, a 8086-ost. Ennek valamennyi belső regisztere és az adatbusza 16 bites, a címbusz 20 vezetéken max. 1 Mb-ot memóriát tud kezelni. Teljesítménynövekedése — a korábbi 8 bitesekéhez képest — nemcsak a szélesebb adatbusznak, de a hatékony belső felépítésnek is köszönhető. Külön egységet alkot az utasításdekódoló és -végrehajtó, valamint a buszkezelő. *Utasításkészletében* újrándóságot jelentettek a szorzó és osztó utasítások és az ismételhető utasítások. Az 1979-ben bejelentett 8088-as CPU felépítése azonos a 8086-oséval, így szoftver szempontból is kompatibilis vele, de külső adatbusza csak 8 bites. Ez természetesen csökkenti a rendszer sebességét, viszont lényegesen olcsóbb lehet.

A CPU-khoz két *koprocesszort* is kifejlesztettek. A 8087 a matematikai mű-

veleteket, míg a 8089 a perifériavezérlést támogatja.

Az IBM a 8088-ast választotta az IBM PC-hez.

8087: A 8086/8088 típusú processzorokhoz készült *koprocesszor* a matematikai műveletek gyors végrehajtására. A központi CPU *utasításkészletében* csak egész számok összeadása, kivonása, szorzása és osztása szerepel. Minden más függvényt — például: sin, log — ezekből kell előállítani, ami meglehetősen lassú. Ilyen alkalmazásokhoz használható a 8087. Utasításai a CPU utasításai között helyezkednek el a programban, és a CPU hívja le őket, de a 8087-es hajtja végre.

80286: Az Intel 1982-ben jelent meg ezzel a CPU-val. Adatbusza és regiszterei 16 bitesek, akárcsak a 8086-osnál, az újrándóságot a *multiprogramozást* támogató, ún. védett üzemmódja jelenti. Mivel elődje, a 8088-as az IBM PC-ben sikeresnek bizonyult, van ezzel kompatibilis, ún. valós üzemmódja is. A 8088-assal kompatibilis módban a címezhető memóriatartomány 1 Mb-ot, a teljes — max. 16 Mb-ot — memória csak a védett módban használható.

80287: Mivel a 80286-os processzornak védett üzemmódja is van, a korábbi — a 8087-es — matematikai koprocesszor nem alkalmazható mellette. Az Intel a 286-oshoz fejlesztette ki a 80287-est. Mint a központi CPU-nál, itt is szigorúan ügyeltek a szoftverkompatibilitásra a korábbi típusal. (A 8086/8088-hoz készült 8089-es, perifériavezérlő koprocesszort nem fejlesztette tovább az Intel.)

80386, 80386DX, 80386SX: 1985-ben jelent meg az Intel a 32 bitesek közül első mikroprocesszorával, a 80386-ossal. A 8088-assal és 80286-ossal szoftverkompatibilis üzemmódjai

mellett rendelkezik egy új, 386-os móddal, amelyben a címezhető memóriataromány 4 Gbájt. Valódi, korábban csak a nagyobb kategóriájú gépeken használható *virtuális memóriakezelést* tesz lehetővé; a virtuális memóriát max. 64 kbájtos *szegmensekre*, a fizikai memóriát 4 kbájtos lapokra osztja.

Az 1988-ban kibocsátott 80386SX processzor belső felépítése azonos a 80386-ossal — így szoftver szempontból teljesen kompatibilisek —, de külső adatbusza csak 16 bites. Ez lassabb, de olcsóbb rendszer kialakítását teszi lehetővé. Az SX típus megjelenése óta a normál 80386-ost DX jelzéssel látták el. A 386SX típushoz az Intel kifejlesztett megfelelő matematikai koprocesszort, a *80387SX-et*.

80386SSL: 1990-ben jelentették be ezt a típust, amely teljesen kompatibilis a 80386SX processzorral, és kisebb mérete és fogyasztása révén alkalmazása elsősorban hordozható számítógépekben célszerű.

80387, 80387DX, 80387SX: A 80386-os processzort támogató matematikai koprocesszor. Felülről kompatibilis a *80287-essel*: néhány új utasítást tartalmaz, és követi a lebegőpontos szabvány időközben bekövetkezett változásait. Az

SX típus megjelenésével kapott DX jelzést.

A 80386SX processzorhoz illeszkedő matematikai koprocesszor szoftver szempontból teljesen azonos a 80387DX-szel.

80486, 80486SX, 80486DX: A 80486-ost 1989-ben jelentették be. 32 bites, felülről kompatibilis a korábbi típusokkal. Nem tartalmaz alapvető újításokat, mint a *80286-os* vagy *80386-os*, de sok olyan megoldást, amelyek lényegesen javították a teljesítményét a 386-oséhoz képest. Ezek közé tartozik, hogy beépítve tartalmazza a *80387-essel* kompatibilis matematikai egységet és egy 8 kbájtos *cache-t* utasítások és adatok számára, ami különösen gyors buszműveleteket tesz lehetővé.

1991-ben jelent meg az Intel az SX típussal, ami teljesen kompatibilis a 80486-ossal — kivéve, hogy nem tartalmazza a matematikai egységet. Megjelenése óta a normál 80486-ost DX jelzéssel látják el.

80487: A 80486SX típushoz kifejlesztett matematikai koprocesszor.

80x86: Gyűjtőfogalom a fenti típusokon belül: vagy *80286-os*, vagy *80386-os*, vagy *80486-os* processzor; ide értve SX és DX jelűeket is.

A, Á

A/D konverter: Analóg-digitális átalakító. Feladata az *analóg jelek digitális*, azaz bináris bitsorozattal leírható jellel alakítása. A digitális jelek egyrészt számítógéppel jól feldolgozhatók (hiszen a számítógép is egy digitális, bitsorozatokkal dolgozó gép), másrészt az analóg jelnél gyorsabban, megbízhatóbban és olcsóbban továbbíthatók. Így egyre elterjedtebbek például a digitális telefonhálózatok. A digitális jeleket azonban általában nem lehet közvetlenül fizikai eszközökben felhasználni, például a hagyományos telefonban. A jelet az átvitel, illetve a feldolgozás után ismét analóg jellel kell alakítani, ezt a műveletet a *D/A konverterek* (digitális-analóg átalakítók) végzik.

ablak (window): A képernyő egy elkerített része, ami önálló megjelenítőként viselkedik. Arra a célra szolgál, hogy a felhasználó egyszerre láthasson több adatállományt, egy program állapotát több szempontból, vagy *multi-programozott* környezetben több párhuzamosan futó program állapotát. A fejlettebb rendszerekben az ablak mérete és helye változtatható, az ablak megszüntetésekor az képernyőnek az alatta lévő tartalma helyreáll. Az „asztallap metafora” azt a hasonlóságot jelzi, hogy az ablakok olyan helyzetben vannak a képernyőn, mint az iratok egy asztallap tetején. Mások szerint az „ördög találmánya”, hogy az ablakok egymást takarják, mert előfordul, hogy a változás a takart részt érinti.

abnormal termination: Lásd hiba.

abortál, abortálás (abortion): Lásd hiba.

absztrakt adattípus: Lásd adattípus.

AC: Lásd NC 2.

Ada: Az USA hadügyminisztériuma pályázatot írt ki egy olyan nyelvre, amely alkalmas beágyazott *valós idejű* rendszerek írására. A pályázatot a francia J. D. Ichbiach csoportja nyerte meg, majd egy bizottság hosszasan átdolgozta. Végül is belekerült a 70-es éveknek a nyelvekkel kapcsolatos valamennyi elképzelése: *blokkstruktúra, külön fordítás, öröklődés, multitasking, kivételkezelés, reprezentációs specifikáció* stb. A nyelv elkészült, eredeti céljára — rakéták beépített vezérlőprogramjának írására — nemigen felel meg, de logisztikai feladatok megoldására alkalmazzák.

adapter: A IBM PC és vele kompatibilis számítógépek részegysége, amely lehetővé teszi olyan periféria illesztését, amelyik egyébként nem rendelkezik megfelelő csatlakozással. (Lásd például *CGA, EGA videoadapterek*.) Egyetlen adapterkártyán több adapter is lehet. Az IBM PC számítógépcsalád a népszerűségét jelentős mértékben a különböző adapterekkel való bővítési lehetőségeinek köszönheti. ►IBM PC.

adatállomány, állomány (file): Lásd fájl.

adatbázis: Egymással kapcsolatban lévő adatok, amelyeket különböző csoportosításokban lehet felhasználni. Számos egymástól nagyon különböző formájú és célú adatbázis van. A tárolt adatokra vonatkozó műveleteket — a megfelelő szempontok szerint — végrehajtó programok összessége az *adatbázis-kezelő rendszer*.

A *szöveges adatbázis* olyan strukturált szöveg (fejezet-paragrafus-bekezdés avagy rekord-mező-almező), ahol egyrészt a struktúra szerint lehet keresni, másrészt viszont bizonyos elemek tetszőleges hosszúságú szövegek lehetnek, és az ezekben levő szavak vagy szöveg-

minták szerint is lehet válogatni. ► ISIS, Norton Guide.

A hierarchikus adatbázis, mint a neve is mutatja, elsősorban a hierarchiát és bizonyos adatoknak a főcsoportból az alcsoport(ok)ba való leszármazását írja le.

A relációs adatbázis olyan fájlokból áll, ahol egy állományon belül azonos szerkezetű *rekordok* vannak. Egy rekordban összetartozó adatok vannak. Egy relációs adatbázis-kezelő általában a következőket tudja: az egyes fájlok, illetve rekordok módosítása; rekordok törlése, beszúrása, sorba rakása bizonyos mezők szerint; azonos szerkezetű rekordokat tartalmazó fájlok összefűzése.

Szelekció (válogatás): Kiválogatja azon rekordokat egy fájlból, amelyeknél a bennük szereplő mezők között fennáll egy adott összefüggés.

Projekció (vetítés): Egy olyan fájlt készít, amelyben az eredeti rekordoknak csak bizonyos mezői szerepelnek.

Join (összekapcsolás): Ha két különböző felépítésű rekordban vannak közös mezők, akkor az eredménylistában olyan rekordok lesznek, amelyekben a mezők vagy az egyik, vagy a másik fajta rekordban szerepelnek. A rekordokat pedig olyan rekordpárokból képezzük, ahol a megfelelő mezők azonosak. (Például: egyik listában van a dolgozók végzettsége, a másikban a dolgozók fizetése, és olyan listát akarunk, amelyben mindkettő szerepel; tehát azokat a rekordokat kell egyesíteni, ahol a név azonos.)

PC-s terminológiában egyetlen ilyen fájlt neveznek adatbázisnak, bár rendszerint vannak adatbázisok közötti műveletek is. A join műveletet nem minden adatbázis-kezelő tudja. ► dBASE, Clipper, FoxPro, fájl típus.

adatbusz, **adatsín**: Lásd busz.

adatrögzítés: Adatok felvitele fájlba későbbi feldolgozás céljából.

adatsín: Lásd busz.

adattípus: A program által felhasznált adatok osztályokba sorolhatók, amelyeket adattípusoknak nevezünk. Egy adattípust két dolog határoz meg: a felvehető értékkészlet és a rajta elvégezhető műveletek halmaza. Értékkészlet szempontjából az adattípusok két csoportba oszthatók: *elemi adattípusok* és *összetett adattípusok*. Az összetett adatok egyszerűbb adatokra, végső soron elemi adatokra bonthatók.

Más szempontból az adattípusok ugyancsak kétfélék: beépített és felhasználó által definiált adattípusok. A beépített adattípust és a rá érvényes műveleteket a programnyelv szolgáltatja. A szokásos beépített elemi adattípusok: különböző hosszúságú (előjeles) egész számok, előjeltelen egész számok, lebegőpontos számok, bitsorozatok, címek. Ezenkívül a programozási nyelvek az összetett adattípusok létrehozására bizonyos beépített módszereket és — ehhez kapcsolódva — bizonyos beépített műveleteket adnak. Ezek: *tömbképzés és -indexelés*, *rekord- és mezőkiválasztás*, fájl- és fájlkezelő műveletek. A felhasználó által definiált adattípusoknál az adat szerkezeti felépítését (*reprezentációját*) és a rajta elvégzett műveleteket a felhasználó adja meg a beépített típusokkal, módszerekkel és műveletekkel.

Absztrakt adattípusról beszélünk olyankor, ha csak az értékkészletet és a műveleteket vesszük figyelembe, és eltekintünk az adat gépi ábrázolásától és a műveletek megvalósításától. *Erős típusellenőrzést* végez az a nyelv, amelyben a változónak típusa van (és csak ilyen típusú értéket kaphat), az egyenlőséget csak azonos típusú értékek között

lehet vizsgálni, valamint minden *függvénynek* (műveletnek) meghatározott típusú *argumentumai* vannak csak — ezáltal ilyen típusú értéket fogadnak el, és meghatározott típusú értéket adnak eredményül. (A *Pascal* és a *Modula* erős típusellenőrzést végez, a *C* nem.)
 ➤ integer, real, felsorolt típus (enumeration), Bool(ean), lista 2., karakter, halmaz (set), mutató (pointer), tömb, string, rekord, fájl típus.

adattömörítés, adattömörítő programok: Az adattömörítő program rendszerint több állományt zömít össze egyé, amelyet *archív példánynak* is neveznek. A tömörítés egyik következménye, hogy a fájlok végén lévő lyukak eltűnnek. A másik abból ered, hogy a kódot egy tömörebb kódba írjuk át. A tömörítés az ismétlődő karakterek, karakterláncok felismerésén és összevonásán alapulhat, de statisztikai elveken is, amikor a gyakori kódokat rövidebb, a ritkébbakat hosszabb bitsorozattal ábrázoljuk; avagy a kettőt kombinálják.

PC-kre számos tömörítő program született: PKPAK & PKUNPAK, PKARC & PKXARC, LZEXE, PKLITE, PKZIP & PKUNZIP, LHA / LHICE / LHARC, ARJ stb. Egy csomó már kiment a divatból, jelenleg az ARJ a legjobb, amely általában a felénél kisebbre zömíti az információt.

A tömörített fájl *önkicsomagoló* (*SFX*), ha az archívum egy olyan *EXE* fájlba van berakva, amely elindítva kikapolja a benne lévő fájlokat, így a tömörítő program átadására nincs szükség.

Másfajta módszereket használnak digitalizált hangra, képpontokból álló képre, ahol a bitre pontos visszaállítás nem szükséges.

AdLib: Olcsó, 18 csatornás *hangkártya*. A csatornák 3 üzemmódban hasz-

nálhatók: hangszer, hangszer és ütősök, *beszédszintézis*. Egy hangszer hangjának visszaadásához 2 csatorna kell. A csatornák kapcsolhatók egymás mögé (additív szintézis) vagy párhuzamosan (frekvenciamoduláció). Erősítőhöz vagy fejhallgatóhoz csatlakoztatandó. A jobb hangkártyáknak szokott lenni AdLib-kompatibilis üzemmódja.

Adlus Pagemaker: Nagy teljesítményű *kiadványszerkesztő*, grafikus lehetőségekkel. 3.01 verziója *Windows* alatt fut. Hasznos funkciója a formába öntés. Meg kell adni az oldal szélét, a hasáboakat, a képek helyét, és a szöveg automatikusan tördelődik, illetve módosításkor áttördelődik.

adventure: Kalandjátékfajta. A játékosnak egy pályán kell végigmennie az indulástól a célíg, közben ki kell találnia, hogy mit kell csinálni, merre kell továbbmenni. A régebbi játékok szövegesen írták le a helyszínt, a mai játékok grafikát és animációt alkalmaznak. Vannak játékok, amelyek időnként véletlenszerűen viselkednek. Az utasítások nyelve többnyire primitív, legtöbbször egy igéből és a tárgyat jelölő névszóból áll a parancs.

AI (artificial intelligence): Lásd mesterséges intelligencia.

alacsony szintű nyelv (low-level language): Olyan nyelv, amelyben visszük tükröződnek a gép szerkezeti elemei, a végrehajtandó gépi utasítások. Az *assembly nyelv* egyértelműen alacsony szintű. Magasabb szintű nyelvekben is lehetnek alacsony szintű vonások. Ezek egyrészt hatékony programok írását teszi lehetővé, másrészt viszont igen megnehezítik a programok átvitelét, illetve más géptípuson rossz hatékonyságú lesz a program. ➤ magas szintű nyelv.

alaplapp: Ezen az áramköri panelen helyezkednek el a PC kategóriájú számítógépeknek a teljesítményt leginkább meghatározó központi részei. Ide tartozik a CPU, a matematikai *koprocesszor* (ha van), a RAM memória, a ROM memória, a DMA-vezérlő, a megszakítás-vezérlő, az időzítő, a *billentyűzetkezelő* és az ezeket kiszolgáló áramkörök. Néhány típusnál a perifériavezérlő áramkörök egy részét is az alaplappra építik.

alfabetikus: Pusztán betűkből álló. Egyes rendszerekben van egy szokőzpótló karakter is (a UNIX-ban a '_'), ami szintén betűnek számít.

alfanumerikus: 1. Betűkből és számokból álló. Néha az is előírás, hogy az első karakter betű legyen.

2. A képernyő üzemmódjának jelzőjeként azt jelenti, hogy a képernyőre *ASCII karakterek* vannak írva, nem pontokból álló ábrák.

ALGOL, ALGOL-60, ALGOL-68: Elsősorban tudományos/műszaki célú, *magas szintű programozási nyelv*. 1958-ban kezdték megvalósítani, pontos definícióját 1960-ban egy nemzetközi bizottság dolgozta ki, de 1962-ben ezt revideálni kellett.

A bizottság 3 célra szánta. A hivatkozási nyelv előírta, hogy mit kell egy ALGOL-60 *fordítónak* megértenie; az előírás nem volt teljes, az I/O műveleteket a konkrét megoldásra hagyta. Az *implementált* nyelv az ALGOL-60 egy konkrét gépre kidolgozott formája. Később kiderült, hogy a teljes hivatkozási nyelvet nem lehet, illetve nem érdemes megvalósítani.

Mint publikációs nyelv — a nyelvet eredetileg *algoritmusok* publikálására is szánták — sem vált be.

Programozási nyelvként az ALGOL a hatvanas években igen népszerű volt. 1968-ra egy újabb, fejlettebb változat

kidolgozását határozta el az ALGOL bizottság. Az új nyelv mind a *szintaxis*, mind a *szemantika* formalizáltsága tekintetében, mind pedig a programozási koncepciók fejlődése szempontjából nagy lépes volt, de programozási nyelvként nem vált be. Napjainkra mindkét nyelv kiment a divatból.

algoritmus: Műveletek sorozata, amellyel valamilyen feladatot meg lehet oldani. Maga az utasítássorozat mindig adott és véges, de mivel lehet benne *elágazás* és egy részsorozat feltételes ismétlése, különböző kiinduló adatok mellett a végrehajtott műveletsorozat hossza más-más lehet. Az algoritmus azokon az adatokon működik, amelyeknél valahány lépés után a végére ér. Bizonyos adatok esetén előfordulhat, hogy egy ismétlődő részből nem lehet kilépni (*végtelen ciklus* keletkezik); ezen az adatokon az algoritmus nem működik.

Vannak algoritmikusan megoldhatatlan feladatok is. Jó példa erre például egy adott sugarú kör kerületének pontos kiszámítása, hiszen ezt akárhány jegy pontosságig megtehetjük, de a végértéket — mivel végtelen sok jegyből áll — véges számú lépésben nem érhetjük el. Adott hibahatáron belül viszont megoldható algoritmikusan a feladat. Arra a feladatra viszont, miszerint adva vannak egymást helyettesítő betűcsoportok és két szó; kérdés: levezethető-e helyettesítésekkel az első szóból a második, bizonyítható, hogy nincs általános algoritmus (de a helyettesítésekre is előírhatók olyan kikötések, amikor van).

allokálás (allocation): Helyfoglalás a memóriában. Két fő formája van: a *statikus allokálásnál* a program vagy programrész futása előtt lefoglalódik a hely és a program befejezésekor szabadul fel; *dinamikus allokálásnál* akkor és annyi

hely foglalódik le (automatikusan vagy a felhasználó utasítására), amikor és amennyire szükség van, és mihelyt nincs már rá szükség, felszabadul (illetve felszabadítható, újra allokálható).

állomány, adatállomány (file): Lásd fájl.

alprogram: Lásd szubrutin.

alulcsordulás (underflow): Lásd válas típus.

Ami Prof (1.2): Saját grafikával és táblázatkezelővel ellátott szövegszerkesztő. (Csak Windows alatt fut.) Több mint egy egyszerű szövegszerkesztő, de DTP-nek kevés. Belső beszámoló, nagyobb üzleti jelentések készítésére jó.

analóg jel: Adott egy fizikai jellemző (hőmérséklet, nyomás stb.), amelynek értéke bizonyos határok között folyamatosan változhat. Ezt az információt egy elektromos jel, általában feszültség vagy elektromágneses hullám valamelyik folytonosan változtatható jellemzőjével visszük át. A jellemző lehet a jel amplitúdója, frekvenciája vagy fázisa, amelyet az információnak megfelelően modulálnak. Az analóg információtovábbítás előnye az egyszerű megvalósítás, mivel a fizikai világ jellemzői általában folyamatosan változnak. Hátránya, hogy bizonyos pontosság, gyorsaság és megbízhatósági követelmények felett a költségei igen nagyok, a digitális jel/információ továbbításáét messze meghaladják. Általában az információ feldolgozása is digitális formában oldható meg egyszerűbben és olcsóbban.

annotáció: Megjegyzés. *Hypertextben* elhelyezett szövegekhez, műszaki rajzokhoz, formális definíciókhoz fűzött vagy fűzhető szöveg. Hivatalosan nem része a dokumentumnak.

ANSI: (American National Standard Institute) Az USA szabványügyi hivatala.

ANSI.SYS: Egy olyan program (*device driver*), amely ANSI szabvány szerinti interfészt biztosít a képernyőhöz és a klaviatúrához. (Vannak olyan terminálok, amelyek hardverben, illetve *firmware-ben* tudják ezt.) A képernyőt úgynevezett eszköz-szekvenciákkal (*escape sequence*) lehet vezérelni. Ezeknek az első eleme egy ESC karakter, ami azt jelzi, hogy itt egy utasítás jön, az utána következő karaktersorozat pedig leírja az utasítást. Ilyenek például: a képernyő törlése; a sor *kurzor* előtti/utáni részének törlése; a képernyő *scrollozása* fel- és lefelé egy sorral; a *kurzor* pozicionálása, eltüntetése, megjelenítése; az oszlopok és sorok számának megváltoztatása; a képernyő üzemmódjának, és/vagy a színeknek a megváltoztatása, ha erre a berendezés módot ad. A klaviatúra oldaláról lehetőség van a billentyűk átdefiniálására (nemzeti billentyűzet). Egyes gombokhoz hosszabb karaktersorozatokat is rendelhetünk (*billentyűmakró*). Ezeknek a lehetőségeknek a kihasználását a szakkönyvek nem javasolják, mert könnyen összeütközésbe kerülhetünk olyan programokkal, amelyek ezeket a lehetőséget más szinten akarják megoldani.

Az ANSI.SYS biztosítja, hogy karakteres olvasás (nem *scan kód*) esetén minden billentyű gépfüggetlen kódot küldjön be. Ha a beküldött kód *ASCII karakter*, akkor a kód egykarakteres, ha nem, akkor először egy nulla karakter érkezik, majd egy — a billentyűre jellemző — szám. A szabvány leírja, hogy 84 és 101 gombos klaviatúra esetén melyik gombnak mit kell beküldenie, ezt kiterjesztett ASCII-nek is hívják. ► billentyűzet, CONFIG.SYS.

antivírus programok: A *vírusok* felderítésére írt, illetve kiirtásukat végző programok. Szabvánnyá az amerikai

McAfee-féle SCAN/CLEAN/VSHIELD programcsomag vált, amelynek elnevezései a vírushatározás alapjául is szolgálnak. Például xxx vírus az, amelyet a SCAN bizonyos verziótól felfelé annak ismer fel. Ismert antivírus programcsomagok: a kelet-európai vírusok felismerésére specializálódott Friderik Skulasson FPROT rendszere, az angolok VR (vírus remover) egyedi irtói, valamint a szovjet katonai vírusprogram atyjának, Lozinszkijnek AIDSTEST programjai. Nyugat-Európában népszerűek dr. Solomons antivírus programjai is. A korszerű antivírus programok (HTSCAN 1.6 +) képesek a mutációk kiszűrésére is valószínűségi alapon. Az igazi megoldást a hardver jelenti, amikor a gépet immunissá teszik a fertőzés ellen. A tisztességes vírusellenes programok sohasem *másolásvédettek*, és vagy ingyenesek, vagy áruk mindenki számára elfogadható.

API (Application Program Interface): Felhasználói programfelület. A *Windows-ban* használt kifejezés azokra az eszközökre (*ablakok, menük, jelképes nyomógombok, ikonok stb.*), amelyeken keresztül a programokat kezelni lehet. Hasonló értelemben említik a fogalmat más programok kapcsán is (mint a *Novell* vagy az *MS LAN Manager*), tehát általános értelemben azon eszközök összessége, amelyeket a felhasználói programok használhatnak.

APT: NC programozási nyelv. Lásd NC 2.

arcade-game: Pénzbedobós automatákra írt program; illetve PC-kre is készítették amazokéhoz hasonló stílusú, lövöldözős, botkormányal irányítható játékokat.

architektúra (architecture): Egy számítógéptípus hardver felépítése összefoglaló néven.

archiválás, archív példány: Adatok és programok adott állapotának lementése abból a célból, hogy az így regisztrált állapotot vissza lehessen állítani. Rendszerint az adatok tömörítésével, esetleg ellenőrző információk hozzáfűzésével jár. ➤ adattömörítés.

ARCnet (Attached Resource Computer Network): A Datapoint cég helyi hálózati hardvere; *vezérlőjelesbusz-protokoll* jellemző rá, sebessége 2.5 Mbit/sec. A 75 ohmos koaxiális kábel *analóg* modulációval továbbítja a jelet. Topológia szerint van sínszerkezetű (busz) és csillag változata is. ➤ helyi hálózat.

argumentum: Lásd szubrutin.

array: Lásd tömb típus.

artificial intelligence, AI: Lásd mesterséges intelligencia.

ASCII kód/karakter: Lásd karakterkód.

assembler: A gép csak a központi memóriájában elhelyezett *bináris* kódot tudja végrehajtani. Ez a *gépikód*, mely rendszerint két részből áll: a végrehajtandó utasítás kódjából és az *operandusok*, illetve azok memóriacímeinek kódjából. A gépikódot természetesen nem bináris számokként szoktuk megadni, mert így nagyon nehézkes lenne az írása és az olvasása. Helyette különböző nyelveket alkalmazunk, amelyeket a számítógép fordít le gépikódra.

A legegyszerűbb, legalacsonyabb szintű nyelv az *assembly nyelv*, amelyben a gép utasításait és azok operandusait szimbolikusan lehet megadni. Az utasítás kódját egy emlékeztető rövidítés, a *mnemonik* helyettesíti. Az adatoknak és a memóriacímeknek nevet lehet adni — ezek lesznek az operandusok. Az *assembly nyelv fordítója* az assembler.

A szimbolikus utasításokon kívül az assembly nyelvben *direktívákat* is kell adni. A direktíva az assemblernek szóló utasítás, amelyből nem lesz végrehajtható kód. Ilyen az adatok elhelyezése a memóriában; memóriaterületek lefoglalása a futás közbeni adatoknak; annak deklarálása, hogy egy adat *globális* vagy *lokális* legyen; stb. Utasítást adhatunk *listafájl* készítésére, arra, hogy hiba esetén folytassuk-e a munkát, arra, hogy gyűjtsük ki a kereszthivatkozásokat.

A *macro assembler* nyelvben lehetőség van *makrók* definiálására és felhasználására.

A modern assemblerek és általában a fordítók nem fordítanak közvetlenül végrehajtható gépikódra, mert a nagyobb programokat nem egyetlen egységként írják meg, hanem egymással összekapcsolható modulokban. A különböző modulok nyelve különböző lehet. A modulok a *globális azonosítókon* keresztül kapcsolódnak össze. Az összefűzést az *linkage editor*, *linker* (összekapcsoló) program végzi. A különböző magas szintű nyelveken írt programmodulokat mindig össze lehet kapcsolni assembly nyelvű modulokkal, noha a magas szintű nyelveket nem mindig lehetséges. ► magas szintű nyelv, makró, metaassembler, moduláris programfelépítés.

aszinkron adatátvitel: Lásd soros adatátvitel.

asszociatív keresés, asszociatív memória: Memóriatartalom szerinti keresés, illetve ezt elősegítő memória (hardver). A keresendő szót, jelsorozatot adjuk meg és válaszul megkapjuk, hogy hol van. Ez nem egyenkénti végignézésel derül ki, hanem a párhuzamos keresést hardver teszi lehetővé.

Atari: Játékautomatákat és számítógépeket gyártó cég. Ugyanazok a prog-

ramok futnak a játékautomatában és az otthoni gépen. Régebbi házi gépei a Commodore hasonló gépeivel voltak egy kategóriában. Az újabbak az Amigától a nagy PC-ig terjedő gépekkel mérhetők össze.

AT-busz: Lásd IDE 2.

átkódolás: Lásd kód.

attribútum: Tulajdonság. Szűkebb értelemben egy *bináris* érték, amely azt mutatja, hogy az adott objektum, program, eszköz stb. rendelkezik-e az adott tulajdonsággal. Tágabb értelemben az objektumot, programot, eszközt jellemző érték. ► paraméter 2.

attribútum-nyelvtanok: Lásd szintaxis.

AUTOEXEC.BAT: A DOS-os rendszerekben többnyire van egy ilyen parancsfájl, amely a gép elindításakor automatikusan működésbe lép, és behívja a *rezidens programokat*, beállítja a *path-okat*, ahol a gépnek a fájlokat keresnie kell. Elindítja a *shellt*, esetleg automatikus mentést készít vagy betöltést végez, stb. Olyan utasításokat ad ki, amelyeket később is és kézzel is ki lehetne adni, ebben különbözik a *CONFIG.SYS* fájlától.

azonosító (identifíer): Olyan név, amelyet a programozó adott valaminek (*változónak*, típusnak, programrésznek stb.). A felhasználón múlik, hogy mit minek nevez. Vannak olyan azonosítók is, amelyeket a programozási nyelv vagy a hozzá kapcsolódó könyvtárak adtak. Ezek felülbírálnak, azaz a névhez a felhasználó új jelentést köthet, esetleg átnevezheti az eredetit. Vannak ezzel szemben *lefoglalt kulcsszavak*. Ezek nem azonosítók, értelmüket a felhasználó nem bírálhatja felül, azonosítóként használni őket tilos, de legalábbis veszélyes.

B

B-tree: Adatstruktúra (kiegyensúlyozott fa = balanced tree), *indexelt* fájlok visszakeresésénél alkalmazzák. ► fájl.

backup: Visszaállítás. Az óvatos programok a bemenő fájl eredetijét elrakják egy mentési példányba — ez az ún. *backup copy* —, mielőtt bármilyen módosítást végeznének rajta. Így később, ha szükséges, az eredeti állapot visszaállítható.

bal-érték: Lásd BCPL.

BASIC: A Beginner's All-purpose Symbolic Instruction Code rövidítéséből kapott nevű programnyelv. Kezdek számára kialakított, könnyen megtanulható, *párbeszédés üzemmódon* alapuló nyelv. Még a legprimitívebb változatai is tartalmaznak valamilyen programírást, tesztelést, javítást segítő környezetet. A BASIC nyelv felépítése kifejezetten ókonzervatív, nem nyújt lehetőséget az újabb programozási koncepciók megvalósítására, ezért sokan ellenzik oktatását és használatát. Számos egymástól lényegesen eltérő nyelvjárása van, így már a modern programozási elveknek megfelelően módosított változatok is születtek. Az IBM PC-n hosszú ideig a GW Basic szinte szabvány volt, manapság azonban a QuickBasic vagy annak megszorított formája a QBasic a mértékadó. ► IPSE.

BAT: A batch-fájlok szabványos kiterjesztése. ► batch.

batch: Köteg. *Batch-programról, batch-fájlról* beszélünk, amikor az operációs rendszernek szóló parancssorozatot egy *BAT* kiterjesztésű fájlba írjuk, és a *parancsértelmező* ezt hajtja végre. A batch-fájlból olyan utasítások is lehetnek (például *ciklus* vagy *feltéte-*

les utasítások), amelyek párbeszédés üzemmódban értelmetlenek. Régebben voltak olyan gépek, amelyek csak *batch üzemmódban* dolgoztak, azaz a felhasználó egy kártyaköteget adott be, amelyre parancsok voltak írva, hogy milyen programokat kell végrehajtani — esetleg maga a program is, illetve az aktuális input adatok is a csomagban voltak —, és nyomtatásban kapta vissza az eredményt vagy a hibajelzést. ► utasítás.

baud: Az adatátviteli csatorna másodpercenkénti állapotváltásait jelző, az átvitel sebességét mutató szám. Ha a csatorna egyetlen bináris vonalon továbbítja az adatokat, azonos a másodpercenként továbbított bitek számával (*bps*). Egy nyolccsatornás vonalon azonos baud ráta mellett a bps természetesen ennek a többszöröse.

bájt (byte): Nyolcjegyű (nyolcbites) bináris szám. A modern számítógépekben egy bájt a központi memória legkisebb írható és olvasható egysége. Nagyobb egységei (szokásos rövidítéseikkel): kilobájt=kbájt=kB=KB, megabájt=Mbájt=MB, gigabájt=Gbájt=GB (kilo= $2^{10}=1024$, mega= 2^{20} , giga= 2^{30}).

BBS (Bulletin Board System): Elektronikus információs rendszer, melyből számítógép-hálózaton keresztül kaphatók tájékoztatók. A legelterjedtebb, legnépszerűbb forma manapság az egyszerű postai telefonhálózatra telepített, *modemek* segítségével elérhető BBS.

BCD (Binary Coded Decimal): Binárisan kódolt decimális (szám). Egy decimális számjegyet 4 biten (egy tetrádon) ábrázol (0 — 9-ig), a további 6 bitkombináció érvénytelen (áltetrád). A BCD aritmetika ilyen számokkal végzi a műveleteket. (A kettes számrendszerben végzett műveletek kerekítési hibája más, mint a tízes számrendszerben vég-

rehajtott műveleteké. A BCD aritmetika úgy számol, mint az ember papíron.) Ügyviteli programokban, adatbázisokban használatos ez az ábrázolási mód.

BCPL: Rendszerprogramok írására kidolgozott nyelv. Nem ismeri az adattípusokat, memóriaszavakban gondolkodik. Egy *változóhoz* egy *bal-érték* — a címe — és egy *jobb-érték* — a tartalma — tartozik. (Az *értékadó utasítás* bal és jobb oldalán álló értékek.) Az utóbbi elnevezésekkel máshol is találkozunk a szoftverirodalomban.

beégetés (blast, blow, burn): Adatok és programok beírása a nemtörlődő memóriába (*PROM, EPROM, EEPROM*). Ez egy speciális berendezéssel történik. A számítógépbe szerelt részegység adattartalmát a gép nem képes megváltoztatni. ► CPU, memória.

belépési pont: Lásd entry point.

belövés: Lásd debugging.

bemeneti/kimeneti, bemenő/kimenő, b/k, B/K: Lásd input/output.

benchmark: Olyan program(ok gyűjteménye), amellyel számítógépek, szoftverek, perifériák teljesítményeit lehet tesztelni. Ugyanazt a benchmarkot különböző gépeken, más-más programokkal futtatva bizonyos műveletek elvégzésének sebességéről, helyfoglalásáról kapunk tájékoztatást.

beszédszintézis (speech sythesis): Olyan szoftver és hardver együttes, amely képes a gépen tárolt írott szövegből jól érthető beszédet *generálni*. Számos nyelvre, így a magyarra is, kapható ilyen rendszer.

betöltőprogram (loader): Lásd linker, linkage editor.

betűtípus, betűkészlet: Lásd font.

billentyűmakró: Lásd makró.

billentyűzet, klaviatúra (keyboard): A számítógép klaviatúrája önmagában is igen bonyolult eszköz, és számítógépen-

ként és terminálként változik, sőt egy-két éve még az IBM-kompatibilis PC-k is számos különböző klaviatúrával kerültek forgalomba. Újabban az IBM PC-k 84, 101 vagy 102 gombos klaviatúrával készülnek, és a 101 gomboshoz hasonló klaviatúrák más gépekre is egyre inkább jellemzők. A következőkben a 101 gombos billentyűzetről lesz szó.

A klaviatúrán a gombok tömbökben helyezkednek el. A legnagyobb (bal alsó) tömbben helyezkednek el a betűk, a számok és az írásjelek, valamint néhány sötétebb színű vezérlőbillentyű, amelyek szokásos funkciója:

— Enter: kocsivissza (CR=13) karakter; DOS-ban a sor vagy az üzenet végét jelenti, *UNIX-ban* ugyanez a billentyű egy soremelés (LF=10) karaktert küld, mivel ott ezt használják a sor és üzenet végének jelzésére.

— Backspace: a *kurzor* visszalép egyet, és törli a karaktert.

— Tab: a kurzor a képernyő következő kijelölt oszlopára lép.

— Shift+Tab: a kurzor a képernyő előző kijelölt oszlopára lép.

— CapsLock: lenyomása után a kis- és nagybetűs írásmód felcserélődik.

A váltó billentyűk csak másik gombbal együtt hatásosak.

— Shift: kis- és nagybetűváltó, váltás a billentyűn az alulra és felülre írt jelek és számok között.

— Ctrl: a betűk a Control billentyűvel együtt lenyomva 0 — 32 közötti *ASCII vezérlőkódot* küldenek be. A billentyű más gombokkal együtt a program vezérlésére használható kódot küld be, amely azonban szövegfájlban nem helyezhető el.

— Alt: az Alt billentyűt lenyomva tartva a numerikus billentyűzeten egy 0—255 közötti számot írhatunk be. Az Alt billentyű felengedése után ez a szám

ASCII karakterként lesz értelmezve. Így olyan betűket és jeleket is be lehet írni, amelyek nincsenek a klaviatúrán, például nemzeti karaktereket, amelyek 128 és 255 között vannak. Az Alt más billentyűkkel együtt a program vezérlésére alkalmazható.

A jobb szélső tömb az úgynevezett numerikus billentyűzet, amely elrendezésében hasonlít az üzleti számológépeken szokásoshoz. Számok és műveleti jelek gyors bevitelére szolgál. A bal felső sarkában lévő NumLock kapcsolóval kurzorvezérlő módba váltható át. Ilyenkor a NumLock lámpa kialszik. Ilyenkor a billentyűk hatása:

— Nyilak: a kurzor az adott irányban egy egységet lép.

— Home: a kurzor a sor elejére lép.

— End: a kurzor a sor végére lép.

— PageUp: egy lapnyit, azaz kb. egy képernyővel visszalép.

— PageDown: egy lapnyit, azaz kb. egy képernyővel előrelép.

— Del: kitörli azt a karaktert, amelyen a kurzor áll.

— Ins: vált a beszúró és felülíró mód között, más rendszerekben egy karaktert beszúr.

Ezek a gombok nem küldenek be olyan kódot, amelytől mindez a képernyőn automatikusan megtörténne, erről a programozónak kell gondoskodnia.

A két tömb között alul helyezkedik el a négy nyílbillentyű, és felette két sorban az Insert, Home, PageUp és a Delete, End, PageDown. Ezek a gombok pontosan úgy működnek, mint a numerikus billentyűzet megfelelő gombjai kurzorvezérlő módban. Ezek a billentyűk hiányoznak a 84 gombos klaviatúráról.

A klaviatúra felső részén egy sorban további gombok helyezkednek el. Bal oldalt a szélén áll az Esc (Escape), mely egy olyan ASCII vezérlőkódot küld be,

amely egy vezérlőkód-sorozat első karaktere. Ilyen vezérlőszekvenciával adhatunk parancsokat nyomtatóknak, a terminálok képernyőinek, egyéb berendezéseknek. Ezt a gombot gyakran használják a programok befejezésére, funkciók abbahagyására vagy visszavonására.

Mellette áll három csoportban négy négy funkcionális gomb: F1 — F12. Ezeket programok vezérlésére lehet használni (nem ASCII kódot küldenek be). Váltó billentyűkkel párosítva a funkciók száma megsokszorozható. A 84 gombos klaviatúrán csak 10 funkcionális gomb van.

Utána három gomb áll: PrintScreen, ScrollLock, Pause-Break. A Shift+PrintScreen hatására az operációs rendszer kinyomtatja a képernyő tartalmát, a Ctrl+Break gombnyomásra az operációs rendszer megszakítja a program futását. Ehhez hasonlóan a Ctrl+Alt+Del együttesre az operációs rendszer újraindul, de hardverellenőrzést nem végez.

A klaviatúra egy gomb lenyomására egy ún. scan kódot küld be a gépbe. Hasonlóan egy scan kód beküldésével jelzi a gomb elengedését. Ezeket az operációs rendszer összegyűjti és ASCII kóddá alakítja. A programból le lehet kérdezni, hogy van-e beérkezett karakter, kiolvasható az ASCII és a scan kód, és lekérdezhető a váltó billentyűk állása. ➤ billentyűmakró, ASCII kód, ANSI.SYS, scan kód.

billentyűzetpuffer: Lásd puffer.

bináris (szám): A tízes számrendszerben leírt szám jegyei azt adják meg, hogy hányat kell venni tíz megfelelő hatványából (jobbról balra növekvő rendben). A bináris (kettes számrendszerű) szám jegyei (a bitek) azt adják meg, hogy a megfelelő kettőhatványból há-

nyat kell venni. A *bináris bitsorozatban* ez a szám csak nulla vagy egy lehet, ezért a bináris számokat igen könnyen ábrázolhatjuk olyan berendezéseken, amelyek kétállapotú alkatrészekből állnak.

A bináris számok nullákból és egyesekből álló hosszú sorozatok, és így a nagyobb számok igen nehezen olvashatók. Ezért a bináris számok helyett *oktális* (nyolcas számrendszerű) vagy *hexadecimális* (tizenhatos számrendszerű) számokat használunk. Ezekben hármas, illetve négyes bitsoportokat jelölünk egy-egy számjeggyel. A hexadecimális számok 10 és 15 közötti jegyeit az „A” — „F” betűkkel jelöljük.

BIOS (Basic Input-Output System): A PC különböző hardverrészegységeit kezelő *szubrutinok* gyűjteménye. Az egyes *perifériákhoz*, illetve feladatokhoz rendelt rutinok *megszakítások* által érhetőek el, kiszolgálják az operációs rendszert és a felhasználói programokat. A megszakításokhoz a *megszakítási táblában* elfoglalt helyük alapján egy szám rendelhető, így a BIOS megszakításai is számozottak. A BIOS-t a számítógéppel, pontosabban az *alaplappal* együtt szállítja a gyártó, és az alaplapon lévő *ROM* vagy *EPROM* tartalmazza. A BIOS lehetőséget ad a gyártónak a hardver bizonyos fokú megváltoztatására, fejlesztésére, mivel a BIOS-t kezelő szoftvert is ő szállítja. Ez akkor okozhat kompatibilitási problémát, ha egy program közvetlenül és nem a BIOS-on keresztül akarja kezelni a számára ismeretlen hardvert. A többfelhasználós, többtaskos operációs rendszerek (például a *Novell* vagy a *UNIX*) nem használhatják a BIOS-t, ekkor fontos az eredeti IBM típussal való kompatibilitás. A ROM a BIOS mellett tartalmazza a bekapcsoláskor lefutó tesztet is (POST: Power On-

Self Test), amelynek hibajelzései — ha vannak egyáltalán — sajnos nem egységesek.

bit-map: Bittérkép. Valamilyen képszerű információ, például grafikus kép, fotó vagy betűforma egyik tárolási módja a számítógépben. Lényege, hogy a képet kisméretű pontokra bontják, és a pontoknak biteket feleltetnek meg. Egyszínű kép esetén egy bit egy pontot jelöl, színes felbontásnál a színek száma határozza meg az egy ponthoz tartozó biteket. A bittérképes tárolás előnye az egyszerűsége, továbbá hogy grafikus módban a képernyőn és a nyomtatón közvetlenül megjeleníthető. Hátránya, hogy a kép nagyítása rontja az élességet, a kicsinyítés pedig a kép torzulásával jár. További hátránya, hogy az így tárolt kép nagy területet igényel a memóriában, illetve a *winchesteren*.

BLOB (Binary Large Object): Nagy, bináris adat (digitalizált hang, raszteresen ábrázolt kép és hasonló). Adatbázisokban való tárolásuk új keletű probléma. ➤ *hypertext*.

blokkstruktúra: A program blokkokból épül fel. Egy blokk adatok, *függvények*, *szubrutinok deklarációiból* áll, majd ezután egy végrehajtható rész következik, ez használhatja a deklarált eszközöket. A függvény maga is egy blokk, amely *lokális* deklarációkat és végrehajtható részt tartalmaz. A függvényen belül lehetnek lokális függvények, azon belül még lokálisabbak stb. Egy belső blokk végrehajtható részében használhatjuk mindazokat a *változókat* és *függvényeket*, amelyek ebben, illetve a nála magasabb szintű blokkokban lettek deklarállva, de nem használhatjuk az azok belsejében lévő lokális deklarációkat. A modern nyelvekben adott a blokkstruktúra, de a több szint mélységű deklarációt nemigen alkalmazzák, mert a prog-

ramok fő strukturálási elve a modulokra bontás.

BNF: (Backus-Naur Form) Lásd szintaxis.

Bool(ean): Lásd logikai típus.

boot(strap): Betöltő. Az operációs rendszer indító része, amely a gép bekapcsolása után, a merev- vagy hajlékonylemez elejéről, a *bootszektorból* automatikusan a gépbe töltődik.

bootszektor (boot sector): A *floppy-lemez*, illetve a *harddisk-partíció* legelső szektora. Tartalmazza egyrészt a lemezre vagy a partícióra vonatkozó adatokat, másrészt azt a rövid programot, amely az operációs rendszert betölti a memóriába. Ezért a bootszektor felépítése függ az operációs rendszer típusától.

A gép bekapcsolásakor a *BIOS*-ban lévő teszt lefutása után mindig a floppy-lemez/harddisk-partíció első szektora — a bootszektor — töltődik be a memóriába.

botkormány (joystick): Elsősorban játékprogramokhoz való irányító eszköz. A PC kategóriájú gépeken nem terjedt el.

bps (bit per second): Az adatátvitel során egy másodperc alatt továbbított bitek száma; az adattovábbítás sebességét jelzi. ➤ baud.

break point (töréspont): Lásd debugging.

bridge (híd): Két különböző hozzáférési *protokollt* használó helyi hálózat összekapcsolására alkalmas berendezés. A két hálózat összekapcsolását az adatkapcsolati réteg szintjén valósítja meg, így ennek a két hálózaton azonosnak kell lennie. ➤ gateway, repeater, router.

bridgware: Olyan hardver vagy szoftver, amelyik egy gépről egy másfajta gépre visz (alakít) át programokat és adatokat.

browse: Böngészés. Olyan utasítások (parancsok), amelyekkel fájlokban, adatbázisokban, *hypertextben* stb. kereshetünk, és az ott található adatokat megnézhetjük.

BTRIVE (4.10): Többféle környezetben (DOS, *Novell*, *OS/2*) futtatható „record management system”.

Alapjában véve többfelhasználós, bár DOS alatt egyfelhasználós is lehet. Számos nyelvhez (*BASIC*, *Pascal*, *C*, *Modula-2* stb.) és ezek számos változatához van felhasználói *interfésze*. 36 funkciója van, amelyek formája nyelvfüggő, de környezetfüggetlen. Biztosítja a többszörös hozzáférést és a rekordszintű zárolást egy felhasználó részére. *Deadlock-felismerő* mechanizmusa van. ➤ indexszekvenciális fájlkezelés, adatbázis.

buffer: Lásd puffer.

busz (bus), sín: A számítógépen belüli egységek összekötésére szolgáló vezeték összessége. Funkciójuk szerint három csoportra oszlanak. Az adatbiteket párhuzamosan az *adatbusz* továbbítja, a helyét azonosító címinformációt — ami lehet memória- vagy perifériacím — pedig a *címbusz*. Az adatot kísérő vezérlőjelek a *vezérlőbuszon* haladnak. A vezeték számának csökkentése érdekében előfordul, hogy az adatbiteket és a címbiteket fizikailag ugyanazokon a vezetékeken továbbítják, időben szétválasztva. Ezt a megoldást *multiplexelt busznak* nevezik. A buszok osztályozásánál az egyszerre továbbítható adatbajtok számát, illetve azt a tulajdonságot veszik alapul, hogy csak egyetlen vagy több processzor csatlakozhat rá. Különböző cégek által kidolgozott számos, jogilag védett buszrendszer létezik.

byte: Lásd bájt.

C, Cs

C (nyelv): A *UNIX* operációs rendszer első változatát Ken Thompson a B nyelvben fejlesztette. A C nyelv a B nyelv és a *BCPL* rendszerprogramozási nyelv alapján született, és a *UNIX* végső változatát ebben írták meg. Noha a *UNIX* alatt számos más nyelv is van, és a C nyelv számos más operációs rendszer alatt is fut, a *UNIX*-ban és a C-ben annyi fontos közös gondolat van, hogy bizonyos programok természetes nyelve a C. Magát a nyelvet B. W. Kernighan és D. M. Ritchie írta le, a nyelvet azóta több irányban továbbfejlesztették, de a *K&R-ben* leírtakat minden változat tudja. Tudása körülbelül azonos a *Pascal* nyelvével, de koncepciója egészen más, típusellenőrzése sokkal lazább, támogatja a *pointer* manipulációkat, nem támogatja az összetett adatok mozgatását és összehasonlítását. Szintaxisa erősen eltér az *ALGOL*, *Pascal*, *Modula*, *Ada* nyelvek szokásos jelölésrendszerétől.

Bizonyos jelöléseiről nehezen dönthető el, hogy programozási fogás vagy elírás, ezért a modernebb fordítók ilyenkor *figyelmeztetés* (*warning*) küldésével kérik a részletesebb leírást. Például a Turbo C-ben, ha egy *feltételes utasítás*-ba feltételként egy *értékadást* írunk, a fordítóprogram megkérdezi (a *help* parancs során): nem egyenlőséget akarunk-e írni; és kéri: írjuk át a feltételt úgy, hogy vizsgálja meg az adott értéket, nem nulla-e.

Módosított változata az *objektumorientált C++*. Nem egészen kompatibilis a C-vel, szigorú típusellenőrzése van. A C++ nyelvű program magába foglalhat C nyelvű modulokat is.

cache (memory) (gyorsítótár): Az adattovábbítás sebességének növelésére alkalmazott, átmeneti tárolásra szolgáló memória. A központi memória és a CPU között használatos a *processzor-cache*, a memória és a *winchester* között a *harddisk-cache*.

CAD (Computer Aided Design): Lásd CAXX technológiák.

CAE (Computer Assisted Education): Számítógépes tanítás. Lásd CAL.

CAL (Computer Assisted Learning): Számítógéppel segített tanulás. Az oktatóprogramok írásának és használatának tudománya.

CAM (Computer Aided Manufacturing): Lásd CAXX technológiák.

CAPP (Computer Aided Process Planning): Lásd CAXX technológiák.

CAQC (Computer Aided Quality Control): Lásd CAXX technológiák.

CASE (Computer Aided Software Engineering): Bizonyos típusú, nagyméretű, egymáshoz hasonló feladatok megoldását elősegítő szoftverkörnyezet. Fontosabb részei:

Formális feladatspecifikáció: Valamilyen, rendszerint grafikus diagramokat használó nyelven meg kell adni, hogy mi a feladat, milyen részfeladatokra bomlik, azok között milyen adatkapcsolat van. Fontos, hogy a *generált* diagramok a tervezett felhasználók számára is áttekinthetők és véleményezhetők legyenek.

Gyors prototípus-generálás: Előre elkészített elemekből a specifikáció alapján — lehetőleg automatikusan — olyan prototípust állítanak elő, amelyen a felhasználó ellenőrizheti, hogy valóban erre van-e szüksége. A prototípus általában nem elég hatékony a valódi felhasználásra, ezért a program nagy részét újra kell programozni.

A változatok nyilvántartása: A programot mind megtervezése idején, mind az életciklusa során számos változtatás érinti. Ezek számítógépes nyilvántartása — nagy hangsúlyt fektetve arra, hogy mi és miért változott — igen fontos feladat; részint a program karbantartása, részint az újabb alkalmazások megtervezése szempontjából.

Hazánkban a CASE rendszerek elsősorban az adatbázis-alkalmazások és a mérnöki automatizálás területén nyertek teret. ► IPSE.

cartridge: Bármilyen, (műanyag) burkolattal ellátott és cserélhető alkatrész, tartozék (festékszalag, memória, tinta, diszk stb.).

CAXX technológiák: Termékek előállítását segítő számítógépes technológiák. Ha ez a technológia átfogja a teljes folyamatot a tervezéstől a gyártáson keresztül a minőség-ellenőrzésig, számítógépes integrált gyártásnak nevezik — **CIM (Computer Integrated Manufacturing)**. Két pillére a számítógéppel segített tervezés — **CAD (Computer Aided Design)** — és a számítógéppel segített gyártás — **CAM (Computer Aided Manufacturing)**. A CAD lényegében tervrajzokat állít elő számítógépes grafika és technológiai adatbázisok segítségével. A következő lépés a technológia és a gyártási folyamat megtervezése — **CAPP (Computer Aided Process Planning)**. A CAM számjegyzévlésű (**NC**) szerszámgépeket tételez fel, de a teljesen automatizált gyártáshoz ezenkívül számítógéppel vezérelt kiszolgáló robotokra, illetve szállítóberendezésekre is szükség van. A számítógéppel támogatott minőség-ellenőrzés — **CAQC (Computer Aided Quality Control)** — két részből áll: a gyártási folyamat közbeni minőségellenőrzés és a végtermék-ellenőrzés. Az előbbi azért fontos, mert

az így kapott adatok visszahatnak a gyártásra. Az utóbbi kissé másfajta feladat, mert nem elegendő azt ellenőrizni, hogy a termék megfelel-e a terveknek, hanem azt kell nézni, hogy a termék a céljának is megfelel-e. (Az egyes részfeladatokat és az egyes rendszerek közötti adatátadást szabványok és ajánlások írják elő.)

CCITT (Comité Consultatif Internationale de Télégraphique et Téléphonique): Az ENSZ Nemzetközi Távközlési Uniójának (ITU) telefon- és adatforgalmi rendszereket koordináló bizottsága. Műszaki ajánlásait gyakran fogadják el szabványként, a telefonrendszerekre vonatkozóak alkotják a V sorozatot, az adathálózatokra vonatkozóak az X sorozatot.

CD-ROM, CD-ROM meghajtó: Olyan háttértároló, amelynek működési elve megegyezik a CD lejátszóéval. A *bináris* információt tartalmazó lemezt lézersugár olvassa ki. Tároló kapacitása 600-700 Mbájt, de hátránya, hogy a *ROM-hoz* hasonlóan csak olvasható. Főként nagy tömegű adatbázisokat — például szabvány- vagy jogszabálygyűjteményeket — árusítanak CD-ROM-on. A nagy kapacitás lehetővé teszi alkalmazásukat a *hypermédia* rendszerekben, ahol szöveget, képet, hangot tárolnak egy lemezen. A CD-ROM-on a rastergrafikus képeket tömörítve tárolják, így rövidebb animációk is elérnek rajta. ► hypertext.

CDS/ISIS: Lásd ISIS.

Centronics: Lásd párhuzamos port.

CGA (Color Graphics Adapter): Az IBM PC számítógépek első grafikus üzemmódot is lehetővé tévő *képernyőadaptere*. Ezt a típust alkalmazták a később megjelent IBM PC/XT és IBM AT gépek is. Alfanumerikus módban a ma is elterjedt 80x25-ös formátumban

dolgozott. Kétféle grafikus módja közül a 320x200-as felbontással négy színt, a 640x200-as felbontással a fekete mellett egyetlen színt használt. Eredetileg rendelkezett NTSC szabványú tévécsatlakozással is, ezt azonban a legtöbb gyártó később elhagyta. Kis felbontása miatt ma már nem gyártják, az újabb képernyőadaptereknek általában van vele kompatibilis üzemmódja. ► grafika.

char(acter): Lásd karakter típus.

checksum: Lásd kontrollösszeg.

chip: Lásd integrált áramkör.

chip set: Lásd NEAT.

ciklikus puffer: Lásd puffer.

ciklus: Utasítás vagy utasítássorozat ismétlése. ► vezérlésátadó utasítás.

cilinder: Lásd harddisk.

CIM (Computer Integrated Manufacturing): Lásd CAXX technológiák.

címbusz, címsín: Lásd busz.

címke (label): Valaminek az azonosítására használt név vagy kód, amelyet a felhasználó adott neki. Használják adatok, fájlok, adathordozók megjelölésére, és lehet programozási nyelvekben utasítások mellé is. A címkével ellátott utasítások azok a pontok a programban, ahová *vezérlésátadó utasításokkal* ugrani lehet.

CISAM: C nyelvű programkönyvtár, amely *indexszekvenciális* fájlkezelő rendszert valósít meg. (*ISAM* = Index Sequential Access Method.) Többszörös hozzáférést és ennek megfelelő védelmi rendszert is nyújt. ► fájl, konzisztencia.

CISC (Complex Instruction Set Computer): A processzoroknak az a családja, amelynek fejlesztése során a teljesítménynövelés mellett az *utasításkészlet* bővítését is fontosnak tekintik. Jelenlegi két legnagyobb teljesítményű tagja az Intel 80486 és a Motorola 68040. Lásd még *RISC*.

client-server (kliens-szerver, ügyfél-kiszolgáló): Csillagszerű *lokális hálózatokban* alkalmazott stratégia az „intelligencia megosztására”. A kliens egy teljesen kiépített PC, amelyen önmagában is lehet dolgozni (nem egy *buta terminál*, mint a régi *időosztásos* nagygépek rendszereiben). A szerver lehet akár PC, akár nagy gép, akár egyéb számítógép. Minden kliens gép össze van kötve a szerverrel. A szerver gép szolgáltatásával emeli a kliens gépek intelligenciáját. Elsősorban lehetővé teszi bizonyos nagy teljesítményű közös erőforrások — nyomtatók, nagy háttértárak, speciális berendezések — ésszerű kihasználását. Kezeli a közös adatbázisban levő adatokat, és védelmet nyújt egyidejű felhasználásból eredő *hibák* ellen. Hálózati szolgáltatásokat nyújt. Ha a szerver egy nagyobb gép, lehetőséget ad olyan munkák elvégzésére, amelyek meghaladják a PC kapacitását.

Clipper (5.0): A Nantucket Corporation terméke. *dBASE (IV)* adatbázisokat kezelő nyelv és *fordítója*. A nyelv lényegében *C makrókból* áll, amelyek mélyebben elrejtett *függvényekből* épülnek fel, a makrók belső szerkezete kiférthető. *Assembly*, *C*, *C++* nyelvű részek beilleszthetők a nyelvbe. Forrásszinten *nyomkövethető*, ha a *forrásszöveget* a kódhoz kapcsoljuk (sokkal hosszabb lesz). Legalább egy IBM PC/AT kategóriájú, gyors gépet igényel. A *Clipper decompiler* visszafordítja a *EXE* programot Clipper forrásszöveggé.

clone: Lásd klón.

close: Lásd fájlműveletek.

cluster: Lásd FAT.

CMOS: A *MOS* típusú félvezetőgyártás egyik változata. ► integrált áramkör.

CMOS-RAM: Az IBM AT számítógép speciális tárolóegysége, nevét a gyártási technológiáról kapta. Tartalmát

egy kisméretű akkumulátornak köszönhetően a gép kikapcsolt állapotában is megőrzi. Információkat tárol a gép *konfigurációjáról*, és tartalmazza azt az óra áramkört, *timert* is, amely folyamatosan méri az időt és követi a dátumot.

CNC: Lásd NC 2.

COBOL (COmmon Business Oriented Language): Az USA hadügyminisztériuma által kidolgozott és szabványosított nyelv, ügyviteli célokra. 1958-61 között született, azóta többször revideálták, ma is előszeretettel használják. Csak az a COBOL *fordító* tekinthető legálisnak, amelyet *validáltak* (érvényesítettek), azaz egy bizottság megfelelő tesztprogramokkal ellenőrizte, hogy megfelel-e a szabványnak. ► validáció.

code: Lásd kód.

COM: Szabványos fájlkiterjesztés. Egyetlen *szegmensben* (64K) elérő, futtatható program megjelölése.

command (parancs): Lásd utasítás.

COMMAND.COM: Az MS-DOS parancsértelmező programja. Gondoskodik arról, hogy induláskor elinduljon az *AUTOEXEC.BAT*, végrehajtja a beépített parancsokat, kezeli a hardverhibákat ("Abort, Retry, Ignore, Fail?"), értelmezi a *batch-fájlokat*. A beírt sor első szavát megvizsgálja: ha belső parancs, végrehajtja, ha nem az, keres egy ilyen nevű és *COM*, *EXE*, *BAT* kiterjesztésű fájlt. A utóbbit értelmezi, az előbbieket betölti, és átadja nekik a vezérlést.

comment: Lásd kommentár.

Commodore: Otthoni célokra, elsősorban játékokhoz készült számítógépcsalád, illetve azonos nevű gyártó cég.

compatibility: Lásd kompatibilitás.

compiler: Lásd fordítóprogram.

CONFIG.SYS: Az MS-DOS úgynevezett *konfigurációs fájlja*. A rendszer *default* beállítását bírálja felül. Átállítható: a Ctrl+Break figyelése, a fájlok és

bufferek száma, a dátum formája, a képernyő, a *billentyűzet*, a nyomtató, a floppy- és a fixdiszk paraméterei, a rendszerben levő logikai lemezegységek száma; létesíthető *RAMdiszk*, *egérmeghajtó*. Egyes nagyobb programok futtatásához meg kell bizonyos paramétereket változtatni (elsősorban a fájlok és *pufferek* számát). Ha egyszerű alkalmazásokat futtatunk, amelyek semmi különlegeset nem igényelnek, akkor erre a fájltra nincsen szükség. ► ANSI.SYS.

consistent: Lásd konzisztens.

controler: Lásd kontroller.

copyright: A szoftvert mint szellemi alkotást — nálunk és az európai országokban — védi a szerzői jog. A szerző beleegyezése nélkül forgalomba hozni, lemásolni, megváltoztatni tilos. Fontos tudni, hogy a jog az alkotást, nem a benne foglalt tudományos eredményt védi. Az utóbbit mindenkinek módjában áll megismerni és felhasználni, ezek alapján új programot írni.

Egyes szoftverszerzők azért, hogy programjuk elterjedjen, lemondanak a jogvédelemről vagy annak egy részéről. *Public domain* az a szoftver, ahol a szerző minden jogától eláll, így programja megváltoztatható és más programokba beleépíthető. A *freeware* bárki számára ingyen rendelkezésére bocsátatik, de a szerző ragaszkodik ahhoz, hogy a programon ne történjen változtatás és a saját neve alatt fusson. A *shareware-ekhez* általában igen alacsony összegért vagy ingyen hozzá lehet jutni, de a szerző kéri, hogy azok, akik sikeresen használják, juttassák el neki méltányos adományukat (donation fee vagy registration fee). Cserében általában többletszolgáltatásokat ígér: a program részletesebb dokumentációját vagy *forráskódját*, vagy továbbfejlesztett változatát, avagy értesítést az újabb változatokról.

Shareware-ként kínálják sok komoly program nem teljes vagy *demóváltozatát*. A komolyabb és drágább programokat általában regisztráltatni kell. A *regisztrálás* során a vevő jogot kap a program egy vagy több példányának használatára. Támogatást kap a felmerült problémák megoldására, és kedvezményesen juthat hozzá a program újabb változataihoz.

Corel Draw (2.0): Grafikai program, amelynek súlypontja a látványos betűkőn és felirat-elhelyezésen van. (Csak Windows alatt működik.)

coroutin: (korutin, társrutin) Lásd Modula.

CP/M: A 8 bites mikroszámítógépekre kifejlesztett, a maga idejében népszerű, primitív operációs rendszer. 16 bites változata, a *CP/M-86* nem terjedt el.

cpi (character per inch): Nyomtatónál használatos mérték, az egy hüvelykre (inch-re) nyomtatandó karakterek száma.

CPU (Central Processing Unit): A számítógép központi, a gépi utasítások dekódolását és végrehajtását végző része: *processzor*. Az egyetlen *integrált áramkörbe* épített processzort *mikroprocesszornak* is nevezik.

A CPU-k belső kialakítása típusonként különböző, de vannak közös alapelvek. Az adatokat regiszterekben tárolják, ezek száma szerint megkülönböztettek korábban memóriaorientált mikroprocesszorokat — amelyek csak 2-3 ilyen regiszterrel rendelkeztek — és regiszterorientált CPU-kat, ahol a regiszterek száma 5-10. A memóriaorientált processzorok egyre inkább a háttérbe szorultak, a mai típusok inkább regiszterorientáltak tekinthetők. A regiszterek hossza meghatározza az egy művelettel feldolgozható adatok nagyságát, így jelentősen befolyásolja a CPU telje-

sítményét. Az első tömegesen elterjedt processzoroknak (8080A, 8085, Z-80) 8 bites regisztereik voltak, később megjelentek a 16 bites (8086/88, 80286, 68010) majd a 32 bites (80386, 80486, 68040) és a 64 bites (R4000) típusok.

A memória eléréséhez címre van szükség, a memóriacímet külön regiszterek tartalmazzák. Az előállítható memóriacím nagysága megszabja a processzorhoz használható memória maximális méretét. A 8 bites CPU-k általában 64 kb-ot, a mai processzorok több Mb-ot vagy Gb-ot memóriát címeznek. A regiszterek hossza mellett a működtető órajel frekvenciája befolyásolja leginkább a teljesítményt. A kezdeti 1-2 MHz-ről elérték az 50-60 MHz-et.

A mai CPU-kat tervezési alapelveik szerint *RISC* és *CISC* típusok szerint osztályozhatjuk.

CRC (Cyclic Redundancy Code): Adattárolás és adattovábbítás során fellépő hibák, adatvesztések észlelésére szolgáló, valamilyen polinommal előállított kód. Minden tárolt vagy továbbított adatblokkhoz hozzáfűzik a CRC-jét. Kiolvasáskor vagy vételkor az adatblokkból ismét kiszámítják a CRC-t és összehasonlítják a rögzített, illetve a vett CRC-vel. Egyezésük esetén valószínűleg nem történt hiba. Az ilyen kóddal való ellenőrzés biztonsága függ az előállító polinomtól, az adatblokk hosszától és a CRC méretétől.

CRT (Cathode Ray Tube): A katódsugárcső angol nevének rövidítése, gyakran így hívják a *képernyő-adapterekben* a központi vezérlőáramkört is.

csatoló: Lásd adapter.

CTC (Counter Timer Circuit): Számláló és időzítő áramkör. ► timer.

cursor: Lásd kurzor.

cut&paste: Olyan művelet, amely (kivágja és) lementti a képernyőn meg-

jelenítettő kép egy részét (egy fájlba), és képes ezt egy másik képernyőn lévő képbe belemásolni. Ehhez hasonló program a *képernyőmentő*, amely a teljes képernyőt egy fájlba menti vagy ki-

nyomtatja. A képernyőn látható képeket a különböző programok eltérő formában tárol(hat)ják, az intelligensebb programok többféle formátumot ismernek.

D

dBASE: A Vulcan nevű programot az Ashton-Tate cég *dBASE II* néven kezdte forgalmazni. Leszármazottai a *dBASE III* és a *dBASE IV*. A továbbiakban a legutóbbiról lesz szó, először magát az adatbázist ismertetjük.

Egy dBASE adatbázis egyforma rekordokból álló fájl, ezt egy szabványos *header* előzi meg, amely a rekordok szerkezetét írja le, és egy *EOF* karakter zárja be. Egy rekord mezőkből áll, az egyes mezőknek típusuk van, ami maximum 255 bájt hosszú *karakterstring* lehet; vagy dátum; lehet az igaz vagy hamis érték; vagy 20 jegyű exponenssel ellátott szám karakteres alakban, amely a gépben lebegőpontos számként van ábrázolva; vagy egy 20 bájtos *BCD* szám; vagy memó, azaz hosszabb szöveg, ami külön fájlban tárolódik. Az adatbázishoz *indexfájl generálható*, az ebben levő indexek (bizonyos mezők) sorrendje szerint az adatbázis könnyen kezelhető. Az adatbázishoz tartozhat címkefájl (*label file*), ebben az adatbázist ismertető szöveg van. Tartozhatnak hozzá olyan fájlok, amelyek segítségével válogatásokat (*reportokat*) lehet ki-gyűjteni és kinyomtatni az adatbázisból.

A kezelésnek öt eszköze van:

— A *Controll Center interaktív, menüvezérelt* környezet, amelyben adatbázisokat lehet létrehozni, feltölteni, módosítani, lekérdezni, reportokat *generálni*.

— A *Command Interpreter* (népszerű néven Dot Prompt), amely az adatbázis-kezelő parancsokat azonnal végrehajtja. 20 alapparancsa van és mintegy 300 összesen.

— A fenti adatbázis-kezelő parancsokból összeállított programok.

— *SQL* nyelven írt programok.

— Továbbá az *interaktív, menüvezérelt Applications Generatorral* alkalmazási programokat állíthatunk elő. Menüket, képernyőket, *ablakokat* definiálhatunk. Összekapcsolva az adatbázist, az indexfájlt, a menüfájlokat, kezelőprogramokat, egy alkalmazási programot generálhatunk.

Programot a *Controll Center*ből indíthatunk el (és lőhetünk be annak *debuggerével*); vagy közvetlenül a DOS-ból, de a dBASE-en keresztül futtathatjuk. Lehetőség van egyszerre több adatbázis kezelésére, és egy adatbázis közös, egyidejű használatára többek által.

DD (double density): Dupla sűrűségű. Floppy lemezekben alkalmazott jelölés. Kétoldalas, 5.25 inch-es lemezeknél 360 kbájt a tároló kapacitása, 3.5 inch-es lemezeknél 720 kbájt.

deadlock: Patthelyzet. Amikor két program kölcsönösen egymásra vár. Például: Mindkettőnek szüksége van két erőforrásra (diszkre, adatbázisra stb.). Az egyik program lefoglalja az első erőforrást kizárólagos használatra, a másik a másodikat, de mindkettőnek szüksége lenne a másikuk által foglaltra is a munka folytatásához. Csak olyan rendszerekben léphet fel, ahol több program fut, és lehetőség van a kizárólagos használatra. (*UNIX, Novell, Windows* stb.)

debug(ging) (hibakeresés, belövés): Egy program hibáinak felderítése. Ezt számos különböző eszközzel lehet támogatni, például: nyomkövetéssel, töréspontok elhelyezésével, lépésenkénti végrehajtással, a memóriatartalom kifrásával. Egy korszerű *debugger* program ezeket a lehetőségeket egyesíti.

Nyomkövetés (Trace). Az adott pontokon a program kiír bizonyos információkat.

Töréspont (Break point). A ponton a program megáll és lehetőség van a memóriatartalom kiírására, bizonyos értékek kiszámítására, lépésenkénti vagy folyamatos folytatásra; a nyomkövetési pontok és töréspontok újradefiniálására.

Lépésenkénti végrehajtásnál a program minden utasítás után megáll.

A program megszakítása (Ctrl+C, Break vagy Ctrl+Break gombbal) után olyan lehetőségeink vannak, mint a töréspontnál.

A nyomkövetés két szinten valósulhat meg. A végrehajtható programokat a gépi utasítások mentén lehet nyomkövetni. Az ilyen program egyik további funkciója, hogy a *gépikódból* visszaállítja az *assembly* kódot a könnyebb olvashatóság kedvéért. Egyes programfejlesztői környezetek viszont lehetővé teszik, hogy a *magas szintű nyelven* írt programokat megfelelő fordítási opciók beállításával, forrásszinten lőjük be. Így lehetőség nyílik arra, hogy a *forrásnyelv* szintjén helyezzünk el nyomkövetési és töréspontokat, stb. Az ilyen nyomkövetés egyik leghasznosabb eszköze a *függvények* hívási *vermének* kiírása.

decompiler: Visszafordító program. A program futtatható kódjából megpróbálja visszaállítani a *forrásszöveget*. Rendszerint arra használják, hogy a programot *visszafejtsék* és megváltoztassák. A megváltoztatott program átadása, illetve a megismert megoldások szövszerinti lemásolása és továbbadása a szerzői jog szempontjából problematikus. ► copyright.

default: Alapértelmezés. Programok, parancsok stb. esetén lehet olyan lehetőség, hogy egy *paramétert* nem kell megadni, mert ilyenkor a program vala-

milyen alapértelmezést használ. Például: DIR az aktuális *directory* összes állományának nevét kiírja.

deklaráció: A modern programozási nyelvek többségében csak olyan szavakat (*azonosítókat*) lehet használni, amelynek a szövegben előzőleg megadtuk a jelentését. Ezt az elvet *linear readingnek* nevezik. Ezért a programok moduljait deklarációkkal kell kezdeni, ahol megadjuk az azonosítók jelentését. A deklarációkhoz a következők tartoznak: adat-, *függvény-* és *szubrutindeklarációk*, külső hivatkozások, *típus-specifikációk*.

A *külső hivatkozások* olyan adatok, függvények és specifikációk, amelyek más modulokban vannak megadva. A specifikációból nem keletkezik adat vagy kód, csak akkor fejt ki a hatását, amikor egy-egy konkrét adatot kezelünk vagy egy konkrét függvényt hívunk meg. Például leírhatjuk egy rekord szerkezetét, de egyetlen ilyen példányt sem hozunk létre, hanem a specifikációt arra használjuk, hogy megmondjuk: ez és ez a fájl áll ilyen rekordokból.

Az adatok vagy változók, vagy konstansok. A *változó* időről időre új értéket kap, a *konstans* azonban olyan adat, amelybe bele van írva valamilyen konkrét érték, amelyet nem lehet megváltoztatni. A konstansot meg kell különböztetnünk a literáltól. A *literál* a program szövegébe beleírt konkrét érték (szám, *string* stb.), amellyel az adott ponton dolgozik a program. A literált nem kell deklarálni. A konstans olyan *implementációfüggő* adat, amely az adott programban egy fix érték, de a program másik változatában, más gépen, más operációs rendszerben eltérő érték is lehet, és ilyenkor az összes olyan helyen, ahol hivatkoznak rá, meg kell változnia.

A **függvény** (vagy szubrutin) egy utasítássorozat, amely valamilyen jól definiálható feladatot hajt végre, és tartozhatnak hozzá *paraméterek*, valamint *lokális deklarációk*. ► **blokkstruktúra**.

dekódolás: Lásd kód.

delimiter: Elválasztójel. Olyan karakter, amely azt mutatja, hogy a szöveg valamilyen egységének vége van. (Szóköz, vessző, pont stb.)

demó: Demonstrációs program, illetőleg egy valódi program demonstrációs változata. Az első esetben a program megfelel arra, hogy valamilyen eszköz tulajdonságait, funkcióit bemutassa, a másik esetben alkalmas megismerésre, tanulásra, de hiányoznak belőle részek a tartós és kényelmes használathoz.

DESIGNER (3.01): Rajzolóprogram Windows alatt. 42-féle *betűtípust* kezel. Kifinomult görbeillesztő programja van (Bezier-görbék). Feliratokat tud illeszteni görbékre, geometriai idomok szélére. Beépített *szövegszerkesztőjével* a képen elhelyezendő hosszabb szövegrészek is elkészíthetők. Van egy funkciója, amelyik a beolvasott rasztergrafikus képeket vektorgrafikává alakítja át. Sajátos eszköze a színek átfolytatása egyik színből a másikba, egyik árnyalatból a másikba. ► **grafika**.

desktop publishing: Lásd kiadványszerkesztő.

DESQview: Karakteres *ablakkezelő shell* DOS fölé. (Funkciójában a Windows-hoz hasonló, de grafika nélkül, ennek megfelelően sokkal kevésbé látványos.) *Multitaskingot* támogató felhasználói *interfész*.

dialog box: *Ablakkezelő* rendszerekben, grafikus felhasználói *interfész*nél alkalmazott megoldás. A kiadott utasítás hatására beugrik egy kisebb ablak, amelyben a végrehajtandó utasítás *paramétereire* vonatkozó kérdések van-

nak. Például: az egész képet kell-e nyomtatni, vagy a látható részét, milyen minőségben stb. Válasz után ez az ablak eltűnik.

digitális jel: Az információ továbbítására használt elektromos jel — általában feszültség vagy elektromágneses hullám — valamilyik jellemzője; csak két különböző értéket vehet fel, amelyek megfelelnek a *bináris* 0, illetve 1 számnak. A továbbítandó információt ennek megfelelően kettes számrendszerben kell ábrázolni. A fizikai környezet jelei általában folyamatosan változnak, ezek számítógépi kezeléséhez, illetve digitális átviteléhez *A/D és D/A konverterekre* van szükség. Ennek ellenére a digitális jelek továbbítása olcsóbb, gyorsabb és megbízhatóbb, mint az *analóg jelé*.

digitalizáló tábla: Egy nagyobb méretű érzékelő felületen (a rajta elhelyezett ábra fölött) egy vele összekapcsolt mutató eszközt — leggyakrabban speciális tollat vagy fonalkeresztet — mozgatva lehet különböző ábrákat bevinni a számítógépbe. A beviteli eszközön lévő gombot megnyomva a tábla adott pontjának koordinátái kerülnek a gépbe. Az érzékelő felület nagysága és működési elve típusonként változik.

DIL (Dual-In-Line): Lásd DIP.

dinamikus: A működés jelzője, ami arra utal, hogy valami csak akkor érvényesül/történik, amikor kell. Vannak például programok, amelyek induláskor lefoglalják mindazt, amire menet közben szükségük lesz (memóriát, fájlokat, perifériákat stb.), és csak a munka végén engedik el. Mások csak akkor foglalják le az erőforrás(oka)t, amikor az(ok) éppen kell(enek) nekik, és amikor már nem, el is engedik. Egy program jobban ki tudja használni a gépet, több fájlt olvashat/írhat, nagyobb adattömeget

dolgozhat fel, ha mindig csak annyit foglal le, amennyit használ. ► allokálás.

DIP (Dual-In-Line Package): Az *integrált áramkörök* leggyakoribb tokozási formája, amelyben egy műanyag (ritkább esetben kerámia) tok két oldalán sorban helyezkednek el a kivezetések.

DIP switch: A DIP formában elhelyezett, kétállású kapcsolók.

directory (katalógus): Lásd könyvtár.

direkt hozzáférés: Lásd adatállomány.

direktíva: Lásd assembler.

Disk Doctor II: Lásd Norton Utilities.

display: Lásd képernyő.

diszjunkció: (megengedő 'vagy' művelet, logikai összeadás) Lásd logikai típus.

diszk (disk): (Mágnes)lemez. A számítógép háttértárát jelöli, amely lehet hajlékonylemezes *floppy-diszk* vagy merevlemezes *harddiszk*.

DLL (Dinamic Link Library): Könyvtármodul *Windows* alatt. Nincs a programba *beleszerkesztve*, a program futásidőben kapcsolódik hozzá. Egyszerre több program használhatja. ► dinamikus.

DMA (Direct Memory Access): Közvetlen memória-hozzáférés. Olyan műveleteknél, ahol a központi memória és valamely periféria között folyik az adatátvitel, ezt a CPU helyett egy külön áramkör, a *DMA vezérlő* irányítja. Ez gyorsítja a rendszer működését.

DNC: Lásd NC 2.

DOS (Disk Operating System): Számos operációs rendszer neve. A PC-világban ilyenkor rendszerint az IBM PC hivatalos operációs rendszerére, az MS-DOS-ra gondolunk, vagy tágabb értelemben valamelyik ezzel többé-kevésbé

kompatibilis operációs rendszerre (*DR-DOS*, Compaq DOS, Tandon DOS).

DOS-parancs: A DOS-parancsok két csoportba sorolhatók: a *belső (internal) parancsokat* a *COMMAND.COM* program hajtja végre, a *külső (external) parancsok* tulajdonképpen közönséges *EXE* és *COM* végrehajtható programok, csak éppen az operációs rendszer részeként adják őket.

Az MS-DOS 1.0-ban 7, az 1.1-ben 11, a 2.0-ban 36, a 3.0-ban és 3.2-ben 40, a 3.3-ban 42, a 4.0-ban 45, az 5.0-ban 48 belső parancs van. A új verziók során egyes parancsok meg is változtak, a legtöbb (8 darab) a 4.0 kapcsán. Hasonló a helyzet a külső parancsokkal is (a 4.0-ás verzió 20 külső parancsot definiált újra).

dpi (dot-per-inch): Egy kép pontokra bontásának mértékegysége. Ennyi pontból áll egy 1 hüvelykes (2,54 cm-es) vonal. Berendezések, illetve az általuk készített kép finomságának jellemzője. ► kiadványszerkesztő, nyomtató.

DRAM: Lásd memória.

DR-DOS: A Digital Research MS-DOS-szal konkurens, de ugyanakkor azal kompatibilisnek szánt operációs rendszer.

A DR-DOS 6.0 szolgáltatásaiban összemérhető az MS-DOS 5.0-val, bár számtalan ponton különbözik tőle.

drive, driver: Lásd meghajtó.

DTP (desktop publishing): Lásd kiadványszerkesztő.

DS (Double sided): Kétoldalas. Floppy-lemezek burkolatán feltüntetett jelzés, amely szerint a lemez mindkét oldala alkalmas adattárolásra. Személyi számítógépeken már csak kétoldalas floppy-lemezeket használnak.

dummy: Helykitöltő. Olyan *argumentum*, fájlrekord, rekordmező, karakter stb., amelyre az adott esetben nincs

szükség, ugyanis a többi adatnak (argumentumnak, rekordnak, mezőnek stb.) olyan értékei vannak, hogy emiatt azt a program nem akarja felhasználni.

dump: A memóriatartalom kiírása egy fájlba. Különösen régebben a *hibák* ki-

derítésére használták. Ha *visszafejtéséhez* jó segédeszközök vannak, ma is jó szolgálatot tehet.

duplex átvitel: Lásd soros adatátvitel.

dzsóker: Lásd joker.

E

e-mail: Lásd elektronikus levelező-rendszer.

echo: Az a jelenség, amikor a számítógép megismétli (kiírja a képernyőre) a kapott parancsot. Elsősorban a *batch-fájlból* kiadott parancsok esetén használatos; ha megengedjük az echót, látjuk, milyen lépéseket hajt végre a program, ha letiltjuk, nem tájékozódhatunk.

editor: Lásd szövegszerkesztő.

EDLIN: Igen régi és primitív *editor*. A *CP/M*, a *UNIX*, az *MS-DOS* (az 5.0 is) tartalmazza, mert néha ma is használják programozók, amikor ismeretlen környezetbe kerülnek.

EEPROM: Lásd memória.

EGA (Enhanced Graphics Adapter): Az IBM 1985-ben jelent meg ezzel a színes *képernyőadapterrel*, amely lényeges fejlődést jelentett a korábbihoz képest (*CGA*). A *karaktergenerátor* memóriája már nem *ROM*, hanem *RAM*, ezért *alfanumerikus* módban tetszőleges — így nemzeti — karakterek megjelenítését is lehetővé teszi. Grafikus módban a felbontást 640x350 képpontra növelték, 16 egyidejűleg használható színnel, amelyek 64 szín közül választhatók ki. Az EGA-t később több más gyártó is továbbfejlesztette; a 640x480 képpontos grafikus felbontás által helyreállt a képernyő 4:3 szélesség-magasság aránya. Később ennél is nagyobb, 800x600-as felbontású EGA kártyák is megjelentek, amelyek azonban általában nem kompatibilisak egymással, ezért csak a gyártó által mellékelt *meghajtó* szoftverekkel használhatók.

egér (mouse): Adatbevitelre szolgáló eszköz, amellyel egy mutatót lehet a képernyőn mozgatni, és ezzel, valamint

az egéren levő gombok megnyomásával adható parancsok: kiválasztás, rajzolás, mozgatás. Az egerek többsége a *soros portra* csatlakozik, de van külön *adapterkártyát* igénylő típus is. Sajnos külön *meghajtó* szoftver szükséges hozzá, amit gyártó szolgáltat, mivel például a DOS sincs az egér használatára felkészítve. Ezért az egérkezelés lehetőségei nem egységesek. Két legelterjedtebb változat a *Microsoft* egere és a *Mouse System PC* egere, amelyekhez egyre több más gyártó is igazodik. Az *Atari* és *Amiga* gépek saját egerükkel működnek.

egész típus (integer): Az egész típus általában meghatározza, hogy a számokat egészként kell ábrázolni, és hogy bizonyos műveletek érvényesek rájuk, de rendszerint nem írja elő a maximális és minimális értéket. Így lehetőség van arra, hogy azonosnak tekintsük a gépbe beépített előjelesegész-aritmetika által adott számábrázolással és műveletekkel. De mivel a gépek az ábrázolható számok halmazában különböznek, egy nyelv az integer típusának értékkészlete más és más lehet gépenként. Egyes nyelvekben 'short', 'long', sőt 'long long' változatot is megengednek, ezeket nem biztos, hogy a hardver támogatja; lehet olyan *implementáció*, amely két különböző típushoz azonos *reprezentációt* rendel.

Az egész típuson négy vagy öt alapművelet végezhető el (összeadás, kivonás, szorzás, az osztás eredményének egész része és a maradék — ezek gépi műveletek; ha hatványozni is kell, ezt többszörös szorzással oldják meg). Az *összehasonlító (relációs) műveletek* (kisebb, nagyobb, egyenlő stb.) igaz vagy hamis *logikai értéket* adnak. Minden típushoz tartozik egy legnagyobb és legkisebb szám, és ha a művelet eredménye ezt túllépi, *túlcsordulás* következik be. ➤ adattípus.

EISA (Extended Industry Standard Architecture): Az eredeti, 80286-os processzorral működő IBM AT-kben használt ISA busz csak 16 bites adatátvitelt tesz lehetővé az *alaplap* és a különböző *adapterkártyák* között. Az IBM a PS/2 sorozatban már egy új buszrendszert kínál, a *mikrocsatornát*, amely már 32 bit szélességű. Ezt azonban jogilag védetté tette, így más gyártók csak jogdíj fizetése mellett alkalmazhatják saját gépeikben. Ennek elkerülésére a legnagyobb gyártók kidolgozták a szintén 32 bites adatátvitelt lehetővé tevő EISA buszrendszert, amely azonban nem védett, így bármely gyártó által beépíthető. Az EISA előnye, hogy kompatibilis a korábbi ISA busszal, és az ahhoz készült adapterek is megfelelőek hozzá. Speciális üzemmódjában a maximális adatátviteli sebessége 33 Mb/ajts lehet másodpercenként.

elágazás, elágazó utasítás: Lásd vezérlésátadó utasítás.

elektronikus levelezőrendszer (e-mail): A világ jelentős számítóközpontjai több nagy, kontinensnyi méretű hálózatba vannak kapcsolva — ezek egymással is össze vannak kötve —, és kisebb-nagyobb helyi hálózatok kapcsolódnak hozzájuk. E hálózatnak egy fontos szolgáltatása az elektronikus levelezőrendszer.

A nagyobb számítóközpontokban postaközpontok is vannak, ahol az egyes előfizetőknek elektronikus postaládájuk (*mailbox*) van, ide lehet küldeni a leveleket. (Ha nem tudjuk valakinek a címét, de ismerjük a hozzávetőleges címét, érdeklődhetünk az ottani postameszternél.) A postaládából kéri ki a felhasználó a levelet a saját gépébe. A rendszer meglehetősen gyors, igen biztonságos, elég olcsó. A postaláda tárolja a levelet mindaddig, amíg ki nem vesszük belőle.

A rendszer intézkedik az automatikus nyugtázásról; értesítést kapunk, ha az előfizetés megszűnt; ha elmegyünk, átírányíthatjuk a levelet az új címre vagy az intézkedő kollégának.

eljárás (procedure): Lásd szubrutin.

elosztott, osztott: Lásd shared.

EMS: Lásd expanded memória.

emulátor: Olyan hardver és/vagy szoftver, amely egy másik számítógép működését szimulálja. Gyakran írnak emulátort készülékben levő számítógépekre és hasonló berendezésekre, hogy a szoftverírást meg lehessen kezdeni a gép elkészülte előtt; de használják a későbbiekben is, mert az emulátor sok információt tud szolgáltatni a *belövéshez*.

encapsulated: Lásd OOP, EPS.

End Of File: Lásd EOF.

entity: Entitás. Egy *lista* egy eleme, a benne foglalt adatok összessége. ➤ entry, item.

entry: Bejegyzés. Egy *lista* egy eleme, rendszerint egy címszóból és a hozzátartozó adatokból áll. ➤ entity, item.

entry point: Belépési pont. Egy nagyobb programegységben azok a pontok, ahol a végrehajtást el lehet kezdeni. Például: egy *modulban* azok a *függvények*, amelyeket meg lehet hívni.

enumeráció (felsorolás): Lásd felsorolt típus.

environment (környezet): Azon eszközök és adottságok, amelyek között a felhasználó dolgozik, például grafikus környezet, többfelhasználós környezet stb. A *UNIX-ban* és a *DOS-ban* a program a *környezetváltozók* lekérdezésével tájékozódhat arról, hogy milyen környezetben működik (például milyen típusú terminálon fut az adott pillanatban), és ennek megfelelően viselkedhet. A környezetváltozókat beállíthatja az operációs rendszer és a felhasználó is.

EOF (End Of File): Fájlvégjel. Egy *ASCII vezérlőkarakter*, amelyet arra szántak, hogy az *ASCII* karakterekből álló szövegfájl végét jelezze. Mivel tudjuk, hogy normálisan a szövegfájlok végét egy nulla karakter jelzi, ez a két egymásnak ellentmondó jelzés néha összeütközésekhez, hibákhoz vezet. Egy sor program, például néhány *szövegszerkesztő* a szövegfájl végének érzékeli az *EOF (Crlt+Z)* karaktert. Mások nem vesznek tudomást róla.

EPROM: Lásd memória.

EPS (Encapsulated PS): Lásd PostScript.

erős típusellenőrzés: Lásd adattípus.
error: Lásd hiba.

értékadás (assignment): Az a művelet, amivel egy változóba beírunk egy értéket, illetve az ott található értéket felülírjuk.

escape sequence: Lásd *ANSI.SYS*.

ESDI (Enhanced Small Device Interface): A harddiszkek fejlődésével az *ST506/412 interfész* adatátviteli sebessége egyre nagyobb korlátot jelentett, ezért 1983-ban kidolgozták az *ESDI* interfészt, amelyet 1990-ben szabványként fogadtak el.

Az *ESDI* az *ST506/412* továbbfejlesztésének tekinthető, és eszköz szintű interfész, akár az elődje. Előnyei azzal szemben, hogy max. 7 meghajtó csatlakoztatható rá, nem tartalmazza az adatkódoló/dekódoló áramköröket, az adatok lemezre írható formába kódolása a meghajtó feladata. Így a választott kódolás nem befolyásolja az interfészt. A max. adatátviteli sebesség az eredeti szabványnál 10 Mbit/sec, de a mai *adapterek* 24 Mbit/sec sebességre képesek.

eseményvezérelt programozás (event driven programming): Esemény minden olyan külső történés (például felhasználói akció) vagy egy másik

program akciója, vagy a program másik ágának akciója, amelyre az adott programrésznek reagálnia kell. Az eseményvezérelt program állandóan figyeli az események bekövetkezését, és ennek megfelelően fut tovább. Az *OOP* szerint írt programoknak mindig van egy olyan központi része, amely az eseményeket, azaz a beérkezett *üzeneteket* figyeli és továbbítja a megfelelő objektumoknak.

Ethernet: Az ütközéses *helyi hálózati protokollal (CSMA/CD)* működő hálózati rendszer. Az Intel, a DEC és a Xerox cégek összefogásából 1978-ban született egy szabvány, ez alkotta a később elfogadott *IEEE 802.3* alapját. A rendszer sebessége 10 Mbit/sec, 50 ohmos koaxiális kábelben alapsávú jeleket továbbít. Az információt változó hosszúságú (max. 1500 bájt) egységekben viszi át, amely az adatokon kívül szállítási és ellenőrzési információt is tartalmaz. Az *ISO/OSI* szabvány két alsó szintjét valósítja meg.

event driven programming: Lásd eseményvezérelt programozás.

EXAPT: Programozási nyelv, *NC* berendezések vezérlőprogramjainak írásához. ► *NC 2*.

Excel (3.0): A *Microsoft* cég *Macintosh* gépre és *MS-Windows* alá írt táblázatkezelő programja; 256 oszlopos, 16 384 soros táblázatokat is képes kezelni. *Menükkel* és *dialog boxokkal*, valamint egy toolbarnak nevezett szimbolikus (egérrel kezelhető, nyomógombsorozattal vezérelhető). Nyomdai minőségű szöveget, táblázatokat, 68-féle két- és háromdimenziós grafikont képes egy dokumentumba elhelyezni.

Egymásba ágyazhatók és összekapcsolhatók a táblázatok. Adatok lekérhetőek *dBASE* fájlokból és — hálózatban — a *Microsoft SQL szervertől* (teljes *SQL* nyelven). Kompatibilis a *Lotus 1-*

2-3-mal. Hálózati kapcsolatra képes az *MS LAN Manager*, *IBM LAN Server*, *Novell NetWare* és más rendszerekkel. Számos típusú nyomtatót és *plottert* ismer.

exception (kivétel): Lásd kivételkezelés.

exclusive or (kizáró vagy): Lásd logikai típus.

EXE: Szabványos fájlkiterjesztés, amely végrehajtandó programot jelöl. Legtöbb *fordítóprogram* EXE programot állít elő.

expanded memória, EMS, (Expanded Memory System): Kibővített memória.

Az IBM PC és az IBM PC/XT processzora (8088) max. 1 Mbájt memóriát képes címezni, amelyből 384 kbájt a *ROM* és a *képernyő-memória* számára kijelölt címtartomány. Az operációs rendszer — és rajta keresztül a felhasználói programok — max. 640 kbájt *RAM* memóriával rendelkeznek. Ez a programok fejlődésével egyre kevésbé volt elegendő, ezért három vezető cég, a Lotus, Intel és Microsoft (*LIM*) megegyezett egy eljárásban, amellyel további memóriaterület vált a programok számára hozzáférhetővé. Ezt nevezték el EMS-nek. Mivel a CPU címzési kapacitását megnövelni nem lehetett, az EMS-t, ami lehet több Mbájt is, csak max. 4 darab, egyenként 16 kbájtos részben, ún. ablakban lehet elérni a 67-es *megszakításon* keresztül. Az elvből következik, hogy

ezt a memóriát csak adattárolásra lehet felhasználni, program végrehajtására nem. Ráadásul a LIM csak a szoftver specifikációjában egyezett meg, a hardver-megvalósításban nem, így az egyes EMS *adapterek* nem egyformák, mindegyik csak a saját *meghajtó* szoftverjével üzemel biztosan.

Az IBM AT 80286-os CPU *védett módjában* 16 Mbájt a címezhető memória, de a DOS csak a PC/XT-vel kompatibilis *valós módban* használja, ahol az elérhető memória itt is csak 640 kbájt. Ezért, és mert egyre több program tudta kezelni az EMS-t, sok AT *alaplakra* is beépítették, az inkompatibilitás azonban itt is megmaradt.

expert system: Lásd szakértő rendszer.

extended memória, XMS, (Extended Memory System): Kiterjesztett memória. Az IBM AT 80x86-os processzora a korábbi PC/XT-vel kompatibilis módjában csak 1 Mbájt memóriát érhet el. Ebben a módban a többi, az *alaplapon* lévő memóriát nevezik *extended memóriának*. Ez a memória a *védett módban* a korlátozás nélkül használható, a *valós, kompatibilis módban* azonban csak adattárolásra és olyan programok futtatására, amelyek erre fel vannak készítve. Az extended memória első 64 kbájtos *szegmensét* gyakran *HMA* (*High Memory Area*)-nak nevezik.

external (külső): ► deklaráció, moduláris programfelépítés.

F

FAT (File Allocation Table): Állományok helyfoglalási táblája. A DOS operációs rendszer alatt a hajlékonylemezen, illetve a harddisk *partíció*n a fájlok elhelyezkedését és a szabad területeket nyilvántartó táblázat. Fontossága miatt két példányban tárolják a lemezen. A FAT bejegyzései floppyn egy-egy *szektorra* vonatkoznak, azonban a harddisken egy bejegyzés több egymást követő szektorra, azaz egy *clusterre* érvényes.

fatal error (végzetes hiba): Lásd hiba.

faxkártya: Lásd telefaxkártya.

fájl (file): *állomány, adatállomány.* Olyan adatkészlet, amelyet rendszerint diszken, esetleg mágnesszalagon vagy más háttértárolón tárolnak. Az operációs rendszer szempontjából vannak végrehajtható fájlok és adatfájlok. A DOS alatt a *COM* és *EXE* kiterjesztésű fájlok programfájlok, azaz betölthetők a memóriába és végrehajthatók. A *BAT* fájlok olyan, az operációs rendszernek szóló parancsokat tartalmaznak, amelyeket végre lehet/végre kell hajtani. A többi fájl csak valamelyik programhoz tartozó adatok halmaza lehet. Más rendszerek szempontjából más a helyzet. Például a *BASIC interpreter* számára a *BASIC (BAS kiterjesztésű)* programok a végrehajtható programok, a többi fájlt csak adatként veszi figyelembe. ➤ (fájlnév)kiterjesztés.

fájlattribútumok: Az MS-DOS-ban a fájlokhoz a *directory-ban* attribútumok tartoznak. Ezek a következők:

- csak olvasható fájl;
- rejtett fájl (a DIR nem írja ki);

— archív bit (a BACKUP és a RESTORE használja).

Fejlettebb operációs rendszerekben, például *UNIX*-ban ez sokkal komplikáltabb. Háromféle dolog engedélyezhető: írás/olvasás, csak olvasás, végrehajtás. És mindezt három szinten kell megadni: mit tehet a tulajdonos maga; mit azok, akik vele egy csoportba tartoznak; és mit a többi felhasználó.

fájlkereső program: Olyan program, amely megkeresi a állományt valamelyik diszken vagy hierarchikus könyvtárban. A keresés haladhat a fájl neve vagy egy benne levő konkrét bájtsorozat szerint, vagy valamilyen minta szerint. Egyes programok még az összetömörített fájlt is megtalálják az ezt tartalmazó fájlban belül.

fájlkiterjesztés: Lásd (fájlnév)kiterjesztés.

fájlműveletek: A programozási nyelvek szempontjából a fájl egy olyan adattípus, amelynek egyszerre csak egy kis része érhető el. Továbbá a rendszerek többségében egyszerre csak bizonyos számú (kb. 10—35) fájl érhető el. A fájlban a következő műveleteket lehet végrehajtani:

Open: A fájl megnyitása. A külső (például diszken lévő) fájl hozzárendelése a programbeli belső fájlhoz (*stream*, logikai periféria). Meg kell adni, hogy írni vagy olvasni akarunk, hogy egy létező fájlt felülírhatunk-e.

Close: A fájl bezárása. Azért van rá szükség, hogy egyszerre ne kapcsolódjunk túl sok fájlhoz. Továbbá, hogy *multiprogramozott* környezetben más is hozzáférhessen a fájlhoz. Ha a program normális véget ér, automatikusan becsukja a fájlokat. *Abnormális befejezés* esetén a program által létrehozott bezáratlan fájlok elveszhetnek.

Írás/olvasás: A fájlban egy mutató jelzi, hogy hol tartunk. A következő írás vagy olvasás innen, illetve ide aktuális, és a művelettel egyidejűleg a mutató eggyel előbbre lép.

Seek: Direkt pozicionálás. A *közvetlen hozzáférésnél* használatos. Olvasás nélkül az utasításban megadott helyre állítja a mutatót a fájlban.

Search: Keresés. Elsősorban az *indexszekvenciális hozzáférés* művelete. Megkeresi az (első) olyan rekordot, amelyben bizonyos mezők, illetve a kulcsok tartalma egy adott érték, és oda pozicionál. (A terminológia nem egységes, az elnevezések a C nyelvből származnak, a *dBASE*-ben például ez az utasítás a *find* és a *seek*.)

(fájlnév)kiterjesztés (extension): Az MS-DOS operációs rendszerben a fájlnevhez egy pont után hárombetűs kiterjesztést kell/lehet kapcsolni, amely többé-kevésbé meghatározza, hogy milyen célra szolgál a program. Az *EXE* és *COM* kiterjesztés végrehajtható programot jelöl, a *BAT* batch-fájlt, a *TXT* általában szöveget, a *C*, *PAS*, *MOD C*, *Pascal*, *Modula forrásnyelvi* fájlt stb. Az operációs rendszer egyes rutinjai és a programok egy része ellenőrzi a kiterjesztést, mások nem.

fájlszerver: Lásd szerver.

fájl típus: A programozási nyelvek szempontjából a fájl olyan adatkészlet, amelyet rendszerint diszken, esetleg mágnesszalagon vagy más háttértárolón tárolnak, és egyszerre csak egy kis része van a memóriában. Alapvetően két formája van:

A szerkezet nélküli fájlban az adatok egymás után, hézag nélkül helyezkednek el. Ilyen például a *textfájl*: amely *ASCII* karakterek sorozata. A sorokat kocsivissza (CR=13 kód) vagy soremelés (LF=10 kód), vagy mindkettő zárja

le. A sorok hosszúságtól függetlenül egymás után vannak lerakva. A fájl végét egy 0 kódú karakter jelzi (vagy *EOF*).

A *rekordosított fájl*ok azonos hosszúságú rekordokból állnak. A rekordok fix hosszúságú mezőkre oszlanak. Egy adat egy mezőbe kerül. Ha egy mezőbe például stringet akarunk elhelyezni, a mezőt a maximális hosszúságra kell lefoglalni. Amikor egy konkrét string kerül bele, a maradék üresen marad. Egyes nyelvekben *változó hosszúságú rekordok* is vannak, ilyenkor a maximális hosszra foglalunk helyet. A rekordosított fájloknak három kezelési módjuk lehet:

Szekvenciális hozzáférés: A rekordokat a fájl elejétől kezdve lehet elérni, a vezérlés minden beolvasás vagy kiírás után automatikusan a következő rekordra áll.

Direkt (közvetlen) vagy random (tetszőleges) hozzáférés: A rekordokat sorszámuk alapján lehet elérni. A kitöltött rekordok között kitöltetlenek is lehetnek. A fájlkezelés általában észreveszi, ha magasabb *indexű* rekordot akarunk olvasni, mint amilyen egyáltalán létezik, de nem feltétlenül veszi észre, ha egy kitöltetlen rekordot akarunk olvasni, amelyiknél magasabb és alacsonyabb sorszámúak is vannak. A régebbi rendszerek korlátozták az egy fájlban levő rekordok számát.

Indexszekvenciális hozzáférés (ISAM=Index Sequential Access Method): A rekord két részből áll, kulcsokból és tartalomtól. A kulcs egy olyan mező, amely szerint a rekordokat sorba lehet állítani. Az indexelt fájlkezelés a rekordokat nem rakja sorba, hanem olyan *indexfájlt* vagy *-fájlokat generál* hozzá, amelyek megadják, hogy melyik

kulcshoz vagy kulcsokhoz hol található meg a rekord.

A rekordokból álló fájlok kezdete gyakran egy *header* (fejléc), amelyben különböző, a fájlra vonatkozó információk vannak feljegyezve. ► fájlműveletek.

fájlvégjel: Lásd EOF.

FDC (Floppy Disk Controller): A meghajtót a számítógéphez illesztő adapter központi vezérlő *integrált áramkörre*, bár gyakran az egész adaptert is így nevezik. Az IBM PC típusú gépek a NEC PD 765 típusát vagy más gyártó ezzel kompatibilis áramkörét használják.

felhasználó (user): Szűkebb értelemben: aki a gépet éppen valamire használni akarja. Más nézőpontból: a rendszerprogram kategóriával szemben, melybe beletartozik mondjuk egy *linkage editor*, *felhasználói program* például egy bérszámfejtő program.

felhasználó által definiált (user defined): Nem készen kapott, nem szabványos, hanem egyedi.

felsorolt típus (enumeration): Ebben a típusban a felhasználó egy névlistát definiálhat. A gépen az egyes neveket a 0-val kezdődő sorszámokkal ábrázolják. Mindössze négyféle művelet alkalmazható rájuk: *értékkadás*, összehasonlítás, *előrelépés (INCrement)* és *visszalépés (DECrement)*, a sorszám átkonvertálása egész számba (és esetleg fordítva). Implementáció kérdése, hogy az első elem elé lépés vagy az utolsó után lépés *túlsordulást* okoz-e vagy hatástalan. ► adattípus.

feltétel nélküli vezérlésátadás: Lásd vezérlésátadó utasítás.

feltételes utasítás: Olyan utasítás, amelyet csak bizonyos feltétel teljesülése esetén kell végrehajtani. ► vezérlésátadó utasítás.

figyelmeztetés (warning): Lásd hiba. **file:** Lásd fájl.

filter (szűrő): Olyan programok, illetve operációsrendszer-parancsok, amelyek csak a *standard inputról* olvasnak és csak a *standard outputra* írnak, miközben azon valamilyen átalakítást hajtanak végre. Ilyen programokat lehet *pipe-ba* állítani. (Példa a MORE parancs. Az előző parancs, például egy DIR a képernyőre, a standard outputra írna, de annyit, hogy egyszerre nem fér ki. A DIRIMORE parancs elkapja az outputot, és képernyőnyi egységekben írja ki.) ► utasítás.

firmware: A számítógép azon tudása, amely a benne ki nem törölődően elhelyezett programokból származik. ► szoftver, hardver, mikroprogram.

flag: Jelző. Rendszerint két állapot megkülönböztetésére: valaminek a meglétéről vagy hiányáról tudósít, és ezzel vezérli a további tennivalókat.

flipping: Lásd scroll.

floppy(-diszk), (floppy disk), floppy-lemez, hajlékonylemez, hajlékony mágneslemez: Mágneses elven működő adattároló. Egy vékony, hajlékony műanyag — általában poliészter — lemez egyik vagy mindkét oldalát mágnesezhető réteggel vonják be. A kör alakú, közepén a forgató tengely számára lyukas lemezt egy borítóba helyezik, a használat során végig ebben forog. Az adatokat a floppy-meghajtó koncentrikus *sávokban* írja a lemezre, a sávok *szektorokra* oszlanak. Új lemezeknél a sávokat és a szektorokat ki kell alakítani, ezt — a megfelelő szoftverrel végzett műveletet — nevezik *formázásnak* vagy *formattálásnak*.

A személyi számítógépeken használt floppy-lemezek átmérője 5.1/4 inch (minifloppy) vagy 3.1/2 inch (mikrofloppy). A méretmegadásnál szokásos

még a tizedesvesszős alak (5,25 vagy 3,5) és a " (coll) mértékegység feltüntetése is. Az 5.1/4"-ost papír, a 3.1/2"-ost kemény műanyag borítóban forgalmazzák. A lemezt gondosan kell kezelni, nagy melegben a műanyag hordozó deformálódhat, a por, nedvesség tönkretetheti a felületet, külső mágneses hatások letörölhetik az adatokat.

floppy-meghajtó, (floppy drive), hajlékonylemez-meghajtó: A floppy-lemezre adatokat író, illetve arról olvasó berendezés. A lemez mindkét oldalához tartozik egy — a magnófejhez hasonló elven, de *digitális jelekkel* dolgozó — író-olvasó fej. A fejek felfekszenek a közöttük forgó lemezre, ezért a felületnek tisztának kell lennie, nem lehet rajta por vagy ujjlenyomat. A lemez egy fordulata alatt lehet írni vagy olvasni mindkét oldal egy-egy *sávjáról*. A fejek sugárirányban lépkednek a lemezen, így minden sávhoz hozzáférhetnek. A lemez fordulatszáma általában 360 percenként. Az IBM PC típusú személyi számítógépekhez kezdetben csak az 5.1/4 inch-es lemezt kezelő meghajtókat használták. A PC-ben és az XT-ben kétoldalas, oldalanként 40 sávot, sávonként 9 szektort tartalmazó, összesen 360 kb-ot tároló típust alkalmaztak.

Az AT-ben jelent meg az 5.1/4-es, kétoldalas, oldalanként 80 sávot, sávonként 15 szektort tartalmazó, összesen 1,2 Mb-ot tároló típus. Ezeket a használt lemezeknek megfelelően gyakran nevezik *DD*, illetve *HD* floppy-meghajtónak. Az 1,2 Mb-ot típusban a 360 kb-ot DD lemez is jó. A 3.1/2 inch-es lemez-meghajtókat az IBM kezdte alkalmazni a *PS/2* sorozatú gépeiben, és később a hordozható — *laptop, notebook* — számítógépekkel rohamosan elterjedtek. Itt is kétféle típus létezik. A HD-s lemezzel — oldalanként 80 sávot

és sávonként 18 szektort tartalmazva — 1,44 Mb-ot tárol, míg a DD típus oldalanként 80 sávot, sávonként 9 szektortban 720 kb-ot tárolására alkalmas.

Az adatokat minden típus *MFM* kódolással írja a lemezre.

A floppy-meghajtó a harddiszkhöz képest lassú, és egy lemezt tekintve kis kapacitású *háttértár*, mégis fontos, mert a gépek között így lehet a legegyszerűbben adatokat, programokat cserélni.

FM: Frekvenciamoduláció. A mágneses adatrögzítésben használt adatkódolási módszer. Az adatok az *órajellel* ütemezve érkeznek, az FM módszernél az óraimpulzusokat is felírják a lemezre, a bináris adatokból csak az egyeseket. Az FM módszer a rossz hatásfoka miatt már nem használatos.

folder: Iratgyűjtő. A *katalógusok ikonos* ábrázolását nevezik így.

folyadékkristályos képernyő: Lásd képernyő.

font (betűkészlet): A kiadványokban számos betűtípussal lehet az írott szöveg értelmét/értelmezését árnyalni, segíteni. (Nem is szólva az esztétikai funkciókról.) A különböző betűtípusok eltérnek egymástól formájukban, méretükben, egyéb jellemzőkben: dőlt, kövér, aláhúzott, árnyékolt stb. Lehet jelkészletünk görög, arab, japán stb. íráshoz, különféle grafikai jelekhez. A *kiadványszerkesztők* számos fontot tudnak kezelni mind a képernyőn, mind a nyomtatón. Egyes kiadványszerkesztők és a hozzájuk tartozó segédprogramok lehetővé teszik, hogy rasterpontokból maga a felhasználó állíthasson össze fontokat a képernyőre, illetve a nyomtatóra. Ezeket *softfontnak* nevezzük, a programot pedig *softfont editornak*. (Ellentétbe állítva a hardverbe épített fontokkal.)

Vannak olyan programok is, ahol a betűket nem rasterpontokkal, hanem

görbékkel lehet megadni. Ez lehetőséget nyújt arra, hogy a betűket tetszőleges méretre nagyítsuk, sőt a döntés, árnyékolás stb. is geometriai transzformációkkal állítható elő. ► PostScript.

fordító(program) (compiler, translator): A program szövegét, ahhoz hogy végre lehessen hajtani, a memóriában elhelyezett gépi kóddá kell alakítani. Ennek a munkának az első fele, amikor a *forráskódot tárgykóddá* alakítjuk — ez a fordítás. A munka másik fele, amikor a lefordított *modulokból* és az előre elkészített modulokból összeáll a program (ezt a *linker* végzi), a legutolsó fázisban pedig a (*loader*) betölti a memóriába. A fordítás maga is több fázisra bontható: a *lexikális elemzés* a szöveget szavakra bontja, majd megállapítja azok szerepét; a *szintaktikus elemzés* ellenőrzi a szöveg formai helyességét, és megállapítja a kisebb-nagyobb egységeket; a (*statisztikus*) *szemantikai analízis* a deklarációk és hivatkozások egybevetésével, a típusok megállapításával és ellenőrzésével foglalkozik; a *kódgenerálás* kódot állít elő, és a *kódoptimalizálás* az adott gép szempontjából optimalizálja azt. ► lexika, szintaxis, szemantika, kódoptimalizálás, interpreter.

format(ting): Formázás. 1. Lásd formázás. 2. A kimenő adatok olvasható formájának kialakítása.

formázás, formattálás: Az a művelet, amellyel a *floppy-lemezen* és a *harddiszken* kialakítják a *sávokat* és a *szektorokat*. A lemez formátuma a típusán kívül az operációs rendszertől is függ. A DOS alatt a harddiszk formázása két lépcsőben történik. Az alacsony szintű formázásnál (low level format) alakítja ki a sávokat és a szektorokat a lemez teljes felületén, ezt nem a DOS, hanem valamilyen segédprogram végzi. A DOS csak *partíciónkénti* magas szintű formá-

zást végez, amelynek során felírja a *boot sectort*, a főkönyvtárat, a *FAT* táblát. A formázás általában a korábban tárolt információ teljes felülírását, elvesztését okozza. Számos újabb rendszer ismeri a biztonsági formázást, ilyenkor a már használt diszk bizonyos információiról mentés készül, és csak annyi íródik felül, amennyi muszáj. Ilyenkor az újraformázott diszk tartalma helyreállítható.

forrásszöveg, forráskód (source code, source text): A programot tartalmazó szövegfájl. Lehet szerkeszteni, javítani, de a gép nem tudja közvetlenül végrehajtani, előbb le kell fordítani vagy *interpretálni* kell. ► fordító(program), interpreter.

FORTH: 1960-ban született, *interpretált*, úgynevezett kiterjeszhető programozási nyelv. (A programozó egyre fejleszti a nyelv utasításkészletét a feladatai megoldásához, az új utasítás hozzácsatolódik a rendszerhez. Az *objektumorientált* rendszerek egy részében is szokásos a kifejlesztett objektumok hozzácsatolása a rendszerhez. A *moduláris programozásnál* is eszközöket fejlesztünk, de ezek nem épülnek bele automatikusan a nyelvbe, csak bevehetők a programba, amikor kell.)

A FORTH előnye volt, hogy belső ábrázolása rendkívül tömör, igen gyorsan interpretálható. Szintaxisa kissé különc. Népszerűsége a kismemóriájú házi számítógépeken volt a legnagyobb. ► OOP.

FoxPro (2.0): A *FoxBase* nevű, hasonló szoftver és a *FoxPro 1.0* verzió utódja. A *dBASE III*-mal, a *dBASE IV*-gyel és a *FoxBase*-zel kompatibilis adatbázis-kezelő. A fájlok szintjén kompatibilis ezekkel, de a programok nyelve és a *menükezelés* más. Kicsit többet tud. Legfőbb újdonsága az úgynevezett *Rushmore-technikájú indexkezelés*,

ami a korábbi technikáknál sokkal gyorsabb. Alapműködése menü- és egérvezérelt. Sok segédprogramja van *szövegszerkesztő, kalkulátor, határidőnapló, ASCII táblázat, képernyőmentő*. Ismeri az *SQL* nyelvet, tud a rekordokhoz mezőket fűzni (*join*).

Képernyő- és menügenerátora van. A Project Manager kapcsolja össze a menüt, az adatbázist, a képernyőket és a programokat egyetlen projektbe, amelyből egy felhasználói program generálható. Ez lehet *közbülső alak*, amelyet a FoxPro futtathat, és lehet DOS-ból végrehajtható *EXE* program. Az utóbbi használja a *FOXPRO.EXE*-t, amelyet azonban nem szerkeszt hozzá a programhoz, és egyszerre több futó program használhatja. ► relációs adatbázis.

FPU (Floating Point Unit): Lebegőpontos műveleti egység. A mikroprocesszorok nagy részének *utasításkészlete* csak az egész számokkal végzett alapműveleteket tartalmazza. Minden más magasabb rendű vagy nem egész számokkal végzett műveletet ilyen *utasításkészlet* mellett csak bonyolultan és főként lassan lehet csak megoldani. Ezért fejlesztettek ki olyan processzorokat, amelyek speciális *utasításai* ilyen feladatok gyors megoldását teszik lehetővé. A lebegőpontos egység lehet *önálló koprocesszor*, vagy beépítve tartalmazza a CPU.

fragmentáció, fragmentálódás (feldarabolódás, széttöredezés): A számítógép munkája közben területeket foglal le a memóriában és a diszken, majd amikor már nincs rájuk szüksége, elen-

gedi ezeket, majd később újabbakat foglal le. A későbbi lefoglalások a szabad lyukakat telítik; ez azzal jár, hogy egy nagyobb programot vagy fájlt nem lehet folyamatosan elhelyezni, hanem több távolos lyukba kerül. Az ilyen elhelyezés lassítja a programok futását. A memória feldarabolódását különféle kiosztási stratégiákkal lassítják; a gép kikapcsolásakor természetesen a memóriatartalom eltűnik. A diszk feldarabolódása átrendező programok (*Disk Doctor, PC Tools*) futtatásával szüntethető meg.

Framework (III 1.1): Az Ashton-Tate cég úgynevezett integrált szoftvere. *Szövegszerkesztő, táblázatkezelő, grafikonkészítő, adatbázis-kezelő* funkciói vannak. Az adatbázisok, táblázatok, grafikonok összekapcsolhatók. Nyelve a FRED. A táblázatok celláira nemcsak pozíciójukkal, hanem az oszlop és sor nevével is hivatkozhatunk.

freeware: Lásd copyright.

funkcionális programozás (functionális programming): A függvény(szerű) programozási nyelv a programot egy függvényként képzelel el, amelyet egyszerűbb függvényekre vezet vissza, ezeket még egyszerűbbekre, és így tovább, egészen a beépített függvényekig. A program végrehajtása nem más, mint sokszorosan egymásba ágyazott függvényhívás-sorozat. A legújabb nyelvek matematikai háttér tekintetében közel állnak az *objektumorientált* programozáshoz (*öröklődés, polimorfizmus*); de máig legnépszerűbb az 1960-ból származó *LISP*.

függvény: Lásd szubrutin.

G

garbage collection (szemétgyűjtés): A dinamikus memóriakezelésnek két különböző stratégiája van: a *dinamikus allokálásnál* a program kér és visszaad memóriadarabokat, a másik stratégiánál a program csak kéri, csak kéri a memóriát, így hát előbb-utóbb kifogy belőle. Ilyenkor a szemétgyűjtő program megvizsgálja, hogy a program milyen allokált adatokat tud még elérni, és amelyeket már nem, azokat szemétnek minősíti és helyüket felszabadítja. Ez néha azzal is jár, hogy az elérhető adatokat összetömöríti, így védekezvén a szabad memória *fragmentálódása* ellen.

A dinamikus allokálásnál a programozónak oda kell figyelnie, hogy mikor foglal le memóriát és mikor engedi el, de az ilyen program hatékony. A másik stratégia igen kényelmes a felhasználó számára, de a szemétgyűjtő program további memóriát és tetemes futásidőt igényel.

gateway (kapu): Két eltérő (esetleg az *OSI modellt* egyáltalán nem követő) számítógép-hálózatot összekapcsoló berendezés. ► bridge, repeater, router.

gázplazmás képernyő, kijelző: Lásd képernyő.

GEM: Grafikus operációs rendszer, *ikonos* vezérlésű. Első verziója a Macintosh gépeken futott, de készült változat a Macintosh II-re, Amigára, ATARI ST-re és IBM PC-re is. A különböző változatoknak különböző előnyei és hátrányai vannak. (Az ATARI ST-n ez a szokásos operációs rendszer. Amigán, IBM PC-n alig használják.)

generálás, generátor: Létrehozás. Például:

tesztadat-generátor: olyan adatokat állít elő, amelyekkel valamely program könnyen ellenőrizhető.

indexfájl-generátor: egy fájlhoz előállítja az *indexfájlt*.

képernyő-generátor: egy *interaktív* program, amelynek megadhatjuk, hogy mi jelenjen meg a képernyőn, és hogy a kitöltendő részeket hogyan akarjuk kezelni. A program ezek után előállít egy képernyőfájlt és hozzá a kezelő rutinokat.

gépikód: Lásd assembler.

globális: 1. Egy, az egész programban érvényes, bármely programrészben használható.

2. Hálózati környezetben: ami az egész hálózatban érvényes/érvényesül.

globális optimalizálás: Lásd programoptimalizálás.

GPSS: (General Purpose System Simulator) *Szimulációs nyelv*, amely blokkdiagramok formájában képzelettel a folyamat leírását.

grafika, grafikus: A számítógépes grafika (computer graphics) ábrák előállításával, ábrákat képviselő adatok számítógépbe való bevitelével és tárolásával foglalkozik (grafikus output, grafikus input és grafikus adatstruktúrák). Az ábrák a grafikus képernyőn (*display-n*) konstruálhatók, előállításukhoz pedig a grafikus módban is használható pontszteres nyomtató vagy a tollas rajzgép (*pen plotter*) a munkaeszköz. Rajzok elemenkénti bevitelére szolgálnak a *digitalizáló táblák* és kisebb méretű *tabletek*. A *scanner* mint fénypásztás képbeviteli eszköz az alakzatot pontokra bontva viszi be — akár szövegről, akár ábráról van szó. (A szöveget vagy a képelemeket e képpontokból programmal kell fölismerni.)

A számítógéppel előállított dokumentumokban háromféle elem lehet: szöve-

gek, geometriai elemekből álló ábrák és pontraszteres ábrák.

A **vektorgrafika, vonalgrafika** geometriai elemekből építkezik: pontok, szakaszok, görbevonalak együttese adja a képet, amelyen feliratok is elhelyezhetők. A képelemeket koordinátákkal kell megadni, és általában szükséges, hogy az elemeket egy geometriai modellé állítsuk össze. A modellt a „grafikus szerelőszalag” (*graphic pipe-line*) egymás utáni műveletei alakítják ki a képernyőn látható képpé. A 3 dimenziós (3D) modellből a középpontos vagy párhuzamos vetítés síkbeli képet állít elő, ezt követően egy lineáris transzformáció képezi le az adatokat a kép(ernyő)koordináta-rendszerbe, a képkivágás (*clipping*) levágja a kilógó képrészeket, a láthatósági eljárások a testeknek a látható részeit keresik meg (szokták takarási eljárásoknak — *hidden surface elimination* — is nevezni), a pásztakonverzió (*scan conversion*) feladata a geometriai képelemek képpontokká konvertálása.

Térbeli tárgyak és színterek ábrázolásánál a kép minden egyes raszterpontjában a fényvisszaverődés fizikai törvényeinek szimulációjával külön meghatározzák a fényességet és színezetet. A jóval egyszerűbb *drótvázás ábrázolásnál* csak a testeket közelítő sokszöglapok éleit rajzolják meg.

Síkbeli (2D) képek geometriai megadása természetesen sokkal könnyebb.

A **raszter(es) grafika, pontgrafika** esetén a kép képpontokból (*pixelekből*) áll össze. A pontminták elhelyezésekor valamilyen imitált festőeszközt mozgatunk *egérrel* vagy *pozicionáló gömbbel*. Lehetőség van még egyszerűbb geometriai elemek (téglalap, töröttvonal, ellipszis, ívek) elhelyezésére is úgy, hogy egérrel kijelöljük nevezetes pontjaikat,

de ezek is bitmintaként kerülnek a képre. Szabad a kép egy részét kijelölni, ezt lemásolni, elmozgatni, törölni, tükrözni stb. Egyszínű képeknél egy képpont egyetlen bittel, míg többszínű vagy szürkeárnyalatos képnél több bittel jellemezhető.

Rajzokat szekvenciális hozzáférésű képfájlokban tárolnak (például HPGL, DXF, CGM stb. formájú képfájlok), amelyek a geometrikus képelemeket és *attribútumaikat* sorolják föl. Raszteres képek tárolására ugyancsak többféle fájlforma alkalmas (PIC, TIFF stb.). Gyakran használt rajzelemeket könyvtárakba lehet gyűjteni, amelyek hierarchikus szervezésűek is lehetnek (a képi *szubrutinok* többszörös mélységben hivatkozhatnak egymásra).

A munkát segítik bizonyos konvertáló programok, amelyek a különböző képfájlok közötti transzformációra alkalmasak. A raszteres képek vektorgrafikává alakítása nehéz feladat, de a vektorgrafikus képek kicsinyítése és nagyítása során a kép kevésbé torzul.

grafikus felhasználói interfész (GUI): Az ember-számítógép interakció korszerű módszere, amelynél egy nagyfelbontású képernyőn gyorsan változtatható *ablakokkal* és más táblácskák (*widget*) segítik az ember munkáját. Az „asztallap metafora” szerint a teljes képernyő egy íróasztal lapjának felel meg, amelyen az egymást átfedő különböző ablakok egy-egy iratot (munkát, feladatot) mutatnak meg. Az iratok iratrendezőbe (*directory, file holder*) gyűjthetők. A lezárt iratokat és rendezőket szerepükre utaló képszimbólumok, *ikonok* szemléltetik. Az asztalon mozgatott *egér* elmozdulását a képernyőn egy mutató jel, *kurzor* követi, amellyel rá lehet mutatni különböző tárgyakra, vagy a *menük* kívánt parancsára. A menük vagy állan-

dóan jelen vannak (például egy fejléc menü), vagy időlegesen földobott menük (*pop-up menu*), vagy a fejlécből redőnszerűen lehúzott menük (*roll-down menu*).

A szemléletes felhasználói interfészeknél csak a szövegesen megadandó adatokat és számokat kell begépelni, minden más művelet közvetlen manipulációval (*direct manipulation*) — rámutatással, elhúzással stb. — végezhető el. A Microsoft cég GUI Guide című könyve ismerteti az IBM és a Microsoft által használt grafikus eszközkészletet, és hat nyelven megadja ezen eszközök elnevezését. ► GUI.

grafikus szerkesztőprogram: (graphic editor). Képek készítésére alkalmas *interaktív* program. Vagy geometriai alapú (*vektorgrafikus*), vagy képpontokból álló (*rasztergrafikus*) képet állít elő, va-

lamilyen adott fájlformátumban. A fejlettebb programok képesek átvenni képeket, képrészeket más formátumban, sőt egymás között is, a másik típusból is. ► grafika.

gráf: Matematikai struktúra; csomópont (node) és él (arc) elemekből áll. Szemléletesen nagyobb pontokkal és azokat összekötő vonalakkal szokás ábrázolni.

A gráf irányított, ha az éleknek irányuk van (ilyen esetben két pontot két különböző irányú nyíl is összeköthet). Egy gráf fa, ha az egy pontból kiinduló ágak mindig csak szétágaznak, soha nem csatlakoznak össze. Ilyenkor a kiindulópontot a fa gyökerének, a végső pontokat, ahonnan nincs tovább, a fa leveleinek nevezzük.

GUI: (Graphical User Interface) Lásd grafikus felhasználói interfész.

H

hacker: A hack angol igéből származik, amelynek egyik jelentése: belepiszkálni egy program bináris kódjába. A hacker olyan ember, aki azzal szórakozik, hogy programok és adatok védelmét feltöri, esetleg rongál is az állományokban vagy lopkod belőlük. A hackerek kedvenc célpontjai a telefonvonalon elérhető hálózatok.

hajlékonylemez: Lásd floppy-diszk.

halmaz típus (set): A felhasználó egy névlistát ad meg, ebből lehet részhalmazokat kiválasztani, és e részhalmazok lesznek a *változó* értékei. A program a változót egy bitvektorral ábrázolja, és eggyel jelöli azokat az értékeket, amelyek benne vannak az aktuális részhalmazban, és nullával, amelyek nincsenek. A szokásos műveletek: egyesítés (*unio*); közösrészt- vagy metszetképzés (*intersection*); *komplementképzés*, annak tesztelése, hogy egy adott elem benne van-e a halmazban. (Egyes nyelvekben nincs elnevezett elemekből álló set típus, csak bitvektor, és bitenkénti logikai műveleteket kell végrehajtani.)
 ► adattípus, logikai típus.

handbook: Lásd kézikönyv.

handshake: Kézfogás. Lásd párhuzamos adatátvitel.

hang(digitalizáló) kártya: Analóg hangfrekvenciás jeleket *digitálissá* alakít és bináris formában tárol. A tárolt adatokat analóg jellé alakítva vissza is játssza. A digitális formában tárolt zenén különböző manipulációk is végrehajthatók. ► MIDI, AdLib, Sound Bluster, Thunder Board.

harddiszk (hard disk), fix diszk, me-revlemezés háttértár, winchester: Mágneses elven működő *háttértár*. Kör alakú

alumíniumlemez mindkét oldalát bevonják mágnesezhető anyaggal, és a floppy-diszkhez hasonló író-olvasó fej tartozik minden felülethez. A tároló kapacitás növelésére több lemezt is alkalmaznak, közös tengelyre szerelve. A lemezek mérete többféle lehet 8, 5.1/4, 3.1/2 vagy 2.1/2 inch. A fejek *sávokban* írnak vagy olvasnak a lemezről, a sávok *szektorokra* osztottak. A fejeket közös szerelvényre erősítik, és a lemezek felett sugárirányban mozognak. Az egy fejpozícióhoz tartozó, egymás felett elhelyezkedő sávokat nevezik együtt *cilindernek*.

Lényeges eltérés a floppy-mechanizmushoz képest, hogy a lemezek fordulatszámuk nagyobb, leggyakrabban 3600 fordulat/perc. Ezért működés közben a fejek nem érnek hozzá a lemezhez, attól nagyon kis távolságra, néhány tized microméterre repülnek. Kikapcsolt állapotban a fejek a lemezen nyugszanak, bekapcsoláskor a felpörgő lemezek által keltett légörvény emeli fel őket, ezért nem csak szemléletes a repül szó. Kikapcsoláskor a fejek újra leereszkednek a lemezre.

A fejek rendkívül rugalmas felfüggesztése miatt a harddiszkek nagyon érzékenyek a rázkódásra. Kikapcsolt állapotban, ha a gépet szállítják, a fejeket olyan sávon kell letenni, amely nem tárol adatot és keményebb felületű, mint az adatsávok. Ezt a sávot nevezik *parkoló sávnak*, a fejek beállítását pedig *parkolásnak*. Szállítás előtt ezt egy megfelelő programmal lehet végrehajtani. Az újabb típusok kikapcsoláskor automatikusan parkolják a fejeket. A fej és a lemez közti légrés kis méretére tekintettel nem lehet por a környezetben, ezért a lemezeket a fejekkel együtt a gyártás során egy légmentesen lezárt kazettában helyezik el, amit kinyitni

csak megfelelően felszerelt szervizben szabad.

A lemezek formátuma, a sávok és a sávon lévő szektorok száma típusonként változik, a szektorok egységesen 512 bájtosak. A harddisk lényegesen gyorsabb, mint a floppy, a sebességét az átlagos elérési idővel szokás megadni, ami azt jelzi, hogy az adatkéréstől átlagosan mennyi idő telik el az adat megérkezéséig. Az adatok lemezre írását általában *MFM* vagy *RLL* kódolással végzik.

A harddisk kezelését az IBM PC típusú személyi számítógépekben külön *adapterkártya* végzi. Az adapter valamilyen *interfészt* biztosít a harddisk számára, ez az interfész szintén befolyásolja a harddisk teljesítményét. Korábban csak az *ST506/412* interfész volt járatos, ma elterjedtek a jobb teljesítményű *ESDI*, *SCSI* és *IDE* interfészek.

harddisk-cache: Lásd cache.

hardver (hardware): A számítógépbe és a hozzá tartozó eszközökbe beépített alkatrészek összessége, illetve az ebből származó képesség. ➤ szoftver, firmware.

hashing: Tárolási és keresési stratégia. A keresendő szó betűiből egy képlettel egy kódot képezünk, és az azonos kódú szavakat tároljuk egy helyen. A képlet akkor jó, ha a csoportokba körülbelül ugyanannyi szó kerül, és egy csoportban olyan szavak vannak, amelyek már az első karakterekben különböznek (nem kell végignézni őket az összehasonlításkor).

hálózat (network): A számítógépeket összekapcsoló rendszer, amelynek célja a különböző gépeken tárolt információ és a csatlakoztatott eszközök több felhasználó közötti gazdaságos megosztása. A gépek távolságától függően lehet

helyi hálózat (LAN), *városi hálózat (MAN)*, és *távolsági hálózat (WAN)*.

A hálózatok meglehetősen különböznek egymástól; amikor a *hálózat architektúrájáról* beszélünk, arról van szó, hogy milyen berendezésekből épülhet fel, milyen *protokollok* vannak benne, a kommunikáció milyen filozófián alapszik, milyen szabványoknak és ajánlásoknak felel meg.

Az IBM SNA (System Network Architecture) szabványa és az *ISO/OSI (Open System Interconnection)* szintekben definiálja a hálózatot. Egy szint (*layer*) az átvitel egy aspektusát határozza meg.

hálózati szerver: Lásd szerver.

háttértár, háttértároló: Mivel a számítógép *RAM* memóriája a gép kikapcsolt állapotában nem őrzi meg tartalmát, a *ROM* pedig nem változtatható, szükség van olyan eszközökre, amelyek a tápfeszültség megszűnése után is tárolják a programokat és adatokat. A PC-kben használt háttértárak mágneses (*harddisk*ek és *floppy*-k), vagy optikai elven működnek (*optikai tároló*).

HD: 1. (high density): Floppy-lemezeken alkalmazott minőségi jelölés. Kétoldalas, 5.25 inch-es lemezeknél 1.2 Mbájt, 3.5 inch-es lemezeknél 1.44 Mbájt a tároló kapacitása.

2. Néha a harddisk rövidítéseként is előfordul, ez esetben nem jelenti azt, hogy az high density.

header: Lásd fájl típus.

heap: A program által *dinamikusan allokált* memória. Illetve, ha ez egy meghatározott területből történik, akkor ennek a területnek a neve. ➤ allokálás, garbage collection.

help: Segítségkérés, illetve -nyújtás. A help parancsra vagy a help gomb lenyomására megjelenik a *helpszöveg*, amely tájékoztat a program működésé-

ról, illetve *szelektív help* esetén arról, hogy mit lehet az adott esetben csinálni. A korszerűbb rendszerekben a képernyőn egy *helpablak* jelenik meg ilyenkor, vagy a képernyő tartalma helyett a *helpképernyő*. Ebben jelenik meg a helpszöveg, és a *helprendszer* segítségével böngészhetünk a programmagyarázat különböző pontjai között. A böngészés befejeztével a helpszöveg eltűnik, és a képernyő eredeti tartalma helyreáll.

A helpszövegeket a korszerűbb programok a *helpfájlból* veszik. A helpfájl lefordítása lehetőséget nyújt a más nemzetiségű felhasználók támogatására.

helyi hálózat (LAN = Local Area Network): Számítógépek kis — max. néhány kilométer — távolságon belüli összekapcsolására szolgáló rendszer. Nemcsak az egyszerű, gépek közötti adatcserére ad lehetőséget, hanem a működtető hálózati operációs rendszertől függően olyan szolgáltatásokat is nyújt, mint az adatállományok vagy programok megosztása, fizikai eszközök (nyomtató, háttértárak) közös használata, bejelentkezés távoli számítógéphez, egységes fájlrendszer kialakítása, elektronikus posta stb. Megvalósítása igen sokféle lehet.

A gépek fizikai összekapcsolására sodrott érpár, koaxiális kábel vagy üvegszálalás fénykábel szolgál. Az elérhető átviteli sebesség a közegen kívül erősen függ a távolságtól, általában több Mbit/sec, üvegszál esetén tipikusan 100 Mbit/sec.

A hálózat gépeinek — állomásainak — elrendezése (a hálózat topológiája) többnyire 3-féle formájú. Sínszerkezetű hálózatban valamennyi állomás közös vezetékre csatlakozik (busz topológia), mindegyik veheti az összes többi adását. Csillag elrendezés esetén az állomások

egy központi géphez (*szerverhez*) csatlakoznak, csak rajta keresztül kommunikálhatnak egymással. Végül a gyűrűs elrendezésben csak a két szomszédjával áll fizikai összeköttetésben.

A hálózatok úgy is összekapcsolhatók egymással, hogy az egyes részek topológiája különböző is lehet. Az összekapcsoláshoz a hálózatok típusától függően kapcsoló-irányítókra (*routerekre*), ismétlő egységekre (*repeaterre*), hídra (*bridge-re*) vagy kapura (*gateway-re*) van szükség, ezekből esetéenként többre is.

A helyi hálózatok — a csillag topológia kivételével, ahol minden állomás csak a *szerverhez* kapcsolódik — általában üzenetszórásúak, ami azt jelenti, hogy mindegyik állomás veszi a többi adását. Ezért a hálózat működésének egyik legfontosabb kérdése, hogyan szerzik meg az egyes állomások az adás jogát. Amennyiben egy központi gép irányítja a hálózat forgalmát, akkor ő adja ki az engedélyeket. Ez egyszerű megoldás, de a központi gép meghibásodása esetén a hálózat megbénul. Egyenrangú állomások esetén különböző eljárásokat, protokollokat dolgoztak ki a forgalom szabályozására. Ezek közül a három legelterjedtebb az ütközésérzékelő CSMA/CD (Carrier Sense Multiple Access/Collision Detection), a vezérlőjeles busz és a vezérlőjeles gyűrű, amelyeket az IEEE 802 sorozatú szabványai írnak le.

Az ütközéses protokollt (a CSMA/CD-t), amelyet az IEEE 802.3 szabvány ír le, a közös buszt használó hálózatok alkalmazhatják. Amikor egy állomás adatot akar küldeni, figyelni kezdi a hálózatot, folyik-e éppen forgalom. Ha igen, akkor vár, ellenkező esetben elkezd az adást. Ez egybeeshet egy másik állomás adáskezdésével, ekkor az

üzenetek ütköznek, a célállomások nem tudják venni őket. Az ütközést mindkét (illetve az összes) adó érzékeli, és leállítja a küldést. Ezután véletlenszerű ideig várnak, majd ismét figyelni kezdik a csatornát. Az eljárás kis és közepes forgalom esetén hatékony, az állomásoknak nem kell túl sokáig várniuk, az ütközések az előzetes figyelésnek köszönhetően viszonylag ritkák. Előnyös, hogy az állomások ki- és bekapcsolódása a hálózatba vagy az esetleg meghibásodott állomás nem igényel külön kezelést. Nagy forgalom esetén az ütközések gyakorisága nő, a hálózat hatékonysága romlik. Ilyen protokollal működik a közismert *Ethernet* hálózat. Bizonyos alkalmazásokban súlyos hátránya, hogy nem adható meg olyan időkorlát — a legrosszabb esetre —, amelyen belül egy állomás biztosan szóhoz jut.

Ezt oldja meg a *vezérlőjel-továbbító (token passing) protokollt* használó hálózat. Ennek állomásai szintén közös buszra csatlakoznak, és egymás között körben továbbítanak egy vezérlőjeles üzenetet. Egy állomás csak akkor küldhet adatot a hálózatban, ha a vezérlőjelet (token) megkapta. Amint befejezte az adást, a vezérlőjelet — vagyis a használati jogot — továbbítja a következő állomásnak. Ehhez az állomásoknak ismerniük kell szomszédjaik címét a hálózatban — ez nem jelent fizikai, csak címbeli szomszédságot —, hogy tudják, kitől kapják és kinek adják tovább a vezérlőjelet. Ezért külön eljárást igényel az állomások be- és kiléptetése a hálózatban, illetve a meghibásodott állomás kiiktatása. Minden állomás — akárcsak a többi protokollnál — csak meghatározott ideig foglalhatja le a hálózatot, ezért a vezérlőjelet minden állomás egy maximális időn belül — ami természetesen az állomások számától függ — biztosan

megkapja. A hálózat kis forgalom esetén nem hatékony, mert a tétlen állomások egymásnak továbbítják a vezérlőjelet, így a küldeni szándékozó állomásnak ki kell várnia, amíg az hozzá elér. A forgalom növekedésével a hatékonyság is javul. Az IEEE 802.4 szabványa tartalmazza a *vezérlőjelesbusz-protokoll* leírását, ennek felel meg például az *ARC-Net* hálózat is. Mindkét szabvány koaxiális kábelt ad meg átviteli közegként, de az árak csökkentése miatt gyártanak sodrott érpárt használó *Ethernet* és *ARC-Net* hálózati *adaptereket* is. A *vezérlőjeles gyűrű (token ring) protokollt* használó hálózatban minden állomás csak a két szomszédjával áll fizikai kapcsolatban. Az állomások az így kialakított gyűrűben egymásnak adják a vezérlőjelet. Az adni kívánó állomás a neki átadott vezérlőjelet kivonja a hálózatból és elküldi az üzenetet, amely a gyűrűben haladva eléri a címzett állomást. A címzett beolvassa az üzenetet, és — beállítva a „rendben vett” jelzést — továbbküldi azt a hálózatban. Az üzenet így visszaér a feladóhoz, aki ellenőrzi a vételi jelzést, majd visszahelyezve a vezérlőjelet a hálózatba, továbbítja a szomszédjának. Az IEEE 802.5 szabványa tartalmazza a *vezérlőjelesgyűrű-protokoll* leírását, a legelterjedtebb megvalósítás pedig az *IBM Token Ring* hálózata.

A vezérlőjeles gyűrű hálózatot a *MAU* (Medium Access Unit) alkalmazása teszi hatékonyá. Ez köti össze az állomások kábeleit, minden állomás egy adás- és egy vételvezetéssel kapcsolódik hozzá. Így, bár logikailag gyűrű, fizikailag csillag alakú a hálózat, ami megkönnyíti a telepítést, bővítést. A *MAU* automatikusan érzékeli az állomások ki- és bekapcsolását, a meghibásodott állomások pedig egyszerűen kiiktathatók a gyűrűből. Segítségével lehetőség van a háló-

zat tesztelésére, a hibás állomás szofverrel való azonosítására. ► *shared*.

Hercules: Elsősorban *képernyő-adaptereket* gyártó cég az USA-ban. Nevét az 1982-ben megjelent, az IBM PC, IBM PC/XT és IBM AT gépekhez való monokróm, az *MDA*-val kompatibilis, de grafikus üzemmódra is képes videó-adaptere tette ismertté. Felbontása (720x348) még a mai grafikus szoftverekhez is megfelelő. Ez a típus teljesen kiszorította az IBM eredeti *MDA* kártyáját, sok más gyártó is készíti monografikus kártya/videóadapter vagy *MGP* néven.

heurisztikus: Olyan megoldási módszer, amely tapasztalatokon és intuíción alapul, de tudományosan nincs igazolva.

hexadecimális szám: Lásd bináris szám.

hiba (error): A program nem mindig tudja normálisan befejezni a munkáját. Ilyenkor hibaüzenetet küld, és egy nem nulla visszatérő kóddal (a *hibakóddal*) fejezi be munkáját. A hibákat hibaszintekbe soroljuk:

warning, (figyelmeztetés): A program tovább működik, de gyanús, hogy amit csinált, nem volt jó.

(serious) error, (komoly hiba): A program nem tud eredményt produkálni, de folytatja a munkát, további hibák kiderítésére.

fatal error, (végzetes hiba): A program nem képes tovább dolgozni, a futás megszakad, a hívó rendszer visszkapja a vezérlést. Ilyenkor azt is mondjuk, hogy a program *abortál* (*abortion, abnormal termination*).

hibakeresés: Lásd debugging.

hibakód: Lásd hiba.

hierarchikus adatbázis: Lásd adatbázis.

High Sierra specification: *CD ROM-on* tárolt fájloknak és rekordoknak az ábrázolására vonatkozó szabvány, mely minden olyan berendezésre nézve is előírás, amely *CD ROM*-okkal dolgozik.

high-level language: Lásd magas szintű nyelv.

highlighting (kiemelés): Az aktuális szövegrész megváltoztatása a képernyőn — figyelemfelkeltési célzattal. Lehet a fényerősség növelése, a szín megváltoztatása, a háttérszín megváltoztatása, esetleg villogtatás.

hívási szekvencia: Lásd szubrutin.

HMA(High Memory Area): Lásd extended memória.

hot key: Lásd TSR.

hozzáférési mód: Lásd fájl típus.

hypertext: Bekezdésekre bontott szöveg, amelyet nem az elejétől végéig szoktak elolvasni, hanem különböző témakörök szerint, kapcsolatok szerint lehet bejárni. Például egy katalógus, ha benne kor, szerző, téma szerint lehet végignézni az anyagot. Az anyag bizonyos szempontok szerint eleve össze van fűzve, a hypertext általában segíti a keresést, az újabb szempontok szerinti összefűzést, az egyéni példány létrehozását, ahol a leírtakhoz hozzáfűzhetjük saját *kommentárjainkat* is.

Ha az anyag nemcsak szövegből áll, hanem rajzok, programok, hang, animáció is tartozik hozzá, akkor *hypermédiáról* szoktunk beszélni. (Ezt általában *CD ROM*-on tárolják.)

I, Í

IBM PC (IBM Personal Computer): Az IBM 1981 augusztusában jelent meg ezzel a számítógéppel, amely később külön kategóriává vált. Az IBM eredetileg csak egyszerűbb feladatokra — például *programszerkesztésre* — önmagában is elegendő, intelligens terminálnak szánta. Az eredeti PC még tartalmazott csatlakozót kazettás magnóhoz, és csak floppy-meghajtóval lehetett használni, harddiszkkal nem. Emellett több szempontból lényegesen jobb volt akkori versenytársainál. Az Intel 8088-as processzorával működött, amely 1 Mbájtos memóriacím-tartományával és belső (16 bites) felépítésével meghaladta a 8 bites típusok teljesítményét.

Az *alaplapon* helyet biztosítottak az Intel 8087 matematikai *koprocesszorának*, így a lebegőpontos számítások sebességét más, hasonló kategóriájú gépekkel összehasonlítva jelentős mértékben növelni lehetett. Ugyancsak az alaplapon olyan buszcsatlakozókat (slot) alakítottak ki, amelyekbe illeszkedtek a különböző perifériavezérlő *adapterkártyák*, így a monitor és a floppy-vezérlő, a *soros/párhuzamos port* stb. Ez lehetőséget adott a gép bővítésére, akár más gyártóktól származó adapterekkel is. A versenytársak gépei csak nehézkesen és általában csak az eredeti gyártó tartozékaival voltak bővíthetők.

A gép sikeréhez nagyban hozzájárult operációs rendszere, a DOS, amelyet eredetileg a Microsoft fejlesztett ki az IBM-nek. A DOS a 8 bites CPU-khoz való primitív CP/M operációs rendszernél — bár átvette néhány vonását — sokkal fejlettebb volt, már az első verzióban is. A IBM PC kategóriájú gépeket

— így később a PC/XT-t és a PC/AT-t is — kelendőségükre való tekintettel sok más cég is gyártani kezdte, ami ugyan tekintélyes bevételtől fosztotta meg az IBM-et, de olyan árcsökkenést eredményezett, hogy napjainkban szinte mindenki számára elérhetővé váltak a PC-k.

IBM PC/XT (IBM PC/EXtended Technology): A PC sikerét látva az IBM 1983 márciusában hozta ki továbbfejlesztett változatát, az XT-t. Ebből már elhagyták a kazettás magnó csatlakozóját, és alkalmassá tették harddiszkkal való kibővítésre. Az eredeti XT — akár csak a PC — 4.77 MHz-es *órajellel* működött, ezt később 8, 10 és 12 MHz-re növelték.

IBM PC/AT (IBM PC/Advanced Technology): Az IBM 1984-ben jelentette be az AT-t. Gondosan ügyeltek a PC-vel és a PC/XT-vel való szoftverkompatibilitásra, ugyanakkor az AT egy teljesen új, 16 bites számítógép volt, a később *ISA*-nak nevezett *busszal*. A korábbi gépekhez fejlesztett *adapterek* használhatók maradtak az AT-ben is, ami olcsóbbá tette az átállást erre a típusra, bár hamarosan sok cég kínált az AT-hez is új, 16 bites adaptereket. A kompatibilitás miatt választott CPU (az Intel 80286) a *8088-aséval* kompatibilis üzemmódja mellett ún. védett módban is működhet, ami lehetőséget adott egy személyi számítógépen futó többtaszkos és többfelhasználós operációs rendszer használatára. Ilyen operációs rendszerek a géphez képest meglehetősen későn jelentek meg, és a PC-kre *implementált UNIX* kivételével nemigen terjedtek el tömegesen.

A PC/XT-hez hasonlóan az AT-ben is lehet matematikai *koprocesszort* *installálni* (a *80287-est*). Újdonságot jelentett a *CMOS-RAM*, és hogy a harddiszk vezérlését a *BIOS*-ba illesztették, a PC/XT-

ben erre csak az adapterkártyán volt lehetőség. Az eredeti AT 6 MHz-es órajellel működött, de később megjelentek a 8, 12, 16, 20 sőt 25 MHz-et használó típusok. Az AT fejlődésének következő lépcsője a különböző áramkörkészletek (*chip set*) elterjedése volt, elsőként a *NEAT* chip seté. Az IBM 1987-ben, a *PS/2* megjelenésekor leállt az AT gyártásával, az Intel *80386-os* processzort már más gyártók kezdték alkalmazni az AT-kben. A 386-os AT-k felépítése tulajdonképpen azonos a 286-os AT-k felépítésével, de a már 32 bites *adatbuszon* fölgyorsult a memória elérése, így a teljesítménynövekedés ebből és a CPU nagyobb teljesítményéből adódik. Ugyanez áll a *80486-os* processzorral épített AT-kre. Kivételt jelentenek az *EISA* buszos 386-os és 486-os gépek.

IC: Lásd integrált áramkör.

identifier: Lásd azonosító.

IDE: 1. (Integrated Development Environment) A Borland C++ (*Turbo C-hez, Turbo Pascal-hoz* hasonló) fejlesztői környezete. (*Editor, fordító, linker, opciók beállítása, több fájlból álló projekt menüs* vezérlése.) Másik neve Programmer's Platform (programozói felület).

2. (Integrated Device Electronic): AT-busz. Az a harddisk *interfész* típus, amelynél a vezérlő és *illesztő* áramköroket egybeépítik a *meghajtóval*, tehát tulajdonképpen nincs interfész felülete. A harddisk közvetlenül az *alaplapon* lévő buszra — általában *ISA* buszra — csatlakozik. Az *adapterkártya* feladata a harddisk felé csak az engedélyezés, a jelerősítés és a fizikai csatlakozás megvalósítása, de legtöbbször a floppy-vezérlő áramkört is tartalmazza, így a floppy-illesztést is ellátja. Az IDE megoldás esetén a meghajtó adatkódolása független az adapterkártyától, és a meg-

hajtó fizikai *paramétereit* — hány fejet és *cilindert* tartalmaz — sem kell ismerenie. Ez lehetőséget ad a fizikai paraméterek transzformációjára, ami azt jelenti, hogy a meghajtó más paramétereit mutat a számítógép felé, mint amilyenek a valódiak. Ez kétféle előnnyel jár. Egyrészt az IBM AT típusú gépeknél a *CMOS-RAM-ban* kell tárolni a harddisk paramétereit, amelyek egy, a *BIOS-ban* tárolt táblázatból választhatók. Ha a táblázat nem tartalmaz az alkalmazni kívánt harddisknek megfelelő, választható egyéb, a transzformálnak megfelelő paraméter. A másik előny abból a korlátból származik, hogy a *partíciós táblában* max. 1024 cylinder és 16 fej állítható be. A több cilindert tartalmazó harddisk a transzformációval kevesebb cilindert, de több fejet mutat a számítógép felé, így kapacitása teljesen kihasználható. Az IDE interfész 16 bites adatbuszán a maximális sebesség 10 Mbit/sec.

időosztás (time sharing): Lásd ütemezés.

IEEE (Institute of Electrical and Electronics Engineers): Az USA-ban működő szervezet, különböző szabványok elbírálásában vesz részt. Jelentősek a nevével jelzett hálózati szabványok. ➤ helyi hálózat.

ikon (icon): Kicsi, grafikus jelkép. Az objektumorientált programozás kapcsán terjedt el. Valamilyen objektumot, programot vagy adatot jelképez. Ha *egérrel* rámutatunk az ikonra, akkor egy, az objektumhoz kapcsolódó tevékenység indul el. (Például a megfelelő program elindul és megmutatja, mit lehet vele csinálni; megmutatja az adatfájl tartalmát és azt, hogy mit lehet vele kezdeni.) Az ikonos rendszert könnyű kezelhetősége, barátságossága teszi vonzóvá. ➤ OOP, grafikus felhasználói interfész.

illesztő: Lásd adapter.

image processing (képfeldolgozás): Fényképek, rajzok stb. bevitele a számítógépbe. Azok feldolgozása, átalakítása: színek, kontrasztok megváltoztatása, háttérvonalak, kontúrok kiemelése, területek szétválasztása, jelek, tárgyak megkeresése. A részben vagy egészen feldolgozott kép megjelenítése.

implementáció: Megvalósítás.

implikáció: Logikai művelet. Lásd logikai típus.

indent: Beljebb kezdés. Amikor egy sor beljebb kezdődik, mint a margó. ► szövegszerkesztés, kiadványszerkesztés.

index: Olyan adat, amely megadja egy elem helyét egy szekvenciális adatszerkezetben. Két nevezetes fajtája van:

1. **Tömbindex.** Egy *tömb* egy elemét kiválasztó sorszám. Indexként egy megsorszámozható adattípus eleme adható meg.

2. **Indexfájlbeli index.** Ez megadja *kulcsok* alapján, hogy az adatfájlban hol van a keresett rekord. ► fájl típus.

indexfájl: Lásd fájl típus.

indexfájl-generátor: Lásd generálás.

indexlyuk: Lásd sáv.

indexszekvenciális hozzáférés: Lásd fájl típus.

infix operátor: Lásd szubrutin.

informatika: Az információ gyűjtésével, átalakításával, továbbításával, rendszerezésével stb. foglalkozó tudomány neve, amely összefoglalja a különböző szaktudományok ilyen irányú eredményeit. Nem tévesztendő össze a számítástechnikával, kibernetikával és egyebekkel, bár kapcsolatos azokkal.

inicializálás: Az az eljárás, amivel valamit elindításkor működőképesse tesznek. Például: A gép indításakor bizonyos programokat lefuttatnak, paramétereket beállítanak, stb.

inkonzisztencia: Lásd konzisztens.

input/output, I/O, i/o (bemenet/kimenet): **input:** Bevitel. Adatok beolvasása a számítógépbe. **output:** Kivitel. A program üzenetei és eredménye.

Vannak eszközök, amelyek csak az adatok bevitelére szolgálnak: a *klaviatúra, egér, pozicionáló gömb, digitalizáló tábla, szkennner, kamera, mikrofon* (digitalizált kép és hang esetén). Más eszközök a kivitelre valók, ilyen a *képernyő, a nyomtató, a rajzgép (plotter)*, de például a hangszóró is. A háttértárak egyszerre ki- és beviteli eszközök. Ilyen a mágneslemez, mágnesszalag (kazetta), *optikai diszk*. A számítógépek közötti kommunikáció is input/output-műveletek segítségével megy végbe. Ezek szerint beszélhetünk input-, output- és input/output-eszközökről, valamint az ezekhez tartozó input/output-kezelésről, továbbá -kommunikációról. Mivel a berendezések igen sokfélék és nem is hasonlítanak egymásra, az egyes berendezések kapcsán elég különböző dolgokról van szó. ► periféria.

installálás: Üzembe helyezés. Olyan egyszeri munkák végrehajtása, amelyek egy berendezés vagy program üzembe helyezéséhez szükségesek. Az *uninstall* parancs visszaállítja az előbbi állapotot. Kiiktatja az üzembe helyezett berendezést vagy programot.

instruction: Lásd utasítás.

instruction set: Lásd utasításkészlet.

integer: Lásd egész típus.

integrált áramkör: *IC* (Integrated Circuit), *chip*. Különböző áramköri funkciókat egyetlen félvezető — rendszerint szilícium — lapkán megvalósító eszköz. Előnyei a diszkrét, különálló elemekből felépített áramkörökkel szemben a nagyobb megbízhatóság, a kisebb fogyasztás és helyfoglalás. Előállításuk két fő technológiai eljárása sze-

rint megkülönböztetnek *MOS* és *bipoláris* integrált áramköröket. Mindkét család több ágra oszlik. Személyi számítógépekben a *MOS* különböző változatai szerint készült áramköröket alkalmazzák leginkább. A beépített aktív elemek, tranzisztorok száma szerint osztályozhatók:

VLSI (Very Large-Scale Integration): Nagyon nagyfokú integráltság, több mint 100 000 aktív elemet tartalmaz.

LSI (Large-Scale Integration): Nagyfokú integráltság, több mint 10 000 aktív elemet tartalmaz.

MSI (Medium-Scale Integration): Közepes integráltság, több mint 100 aktív elemet tartalmaz.

SSI (Small-Scale Integration): Kis integráltság, 100-nál kevesebb aktív elemet tartalmaz.

integrity: Integritás, sértetlenség. Az a tulajdonság, hogy egy fájl, adatbázis, szabályrendszer, általában véve egy adatkészlet nem sérült meg, hiánytalan és *konzisztens*. Az adatkészlet elvesztheti integritását programozási hiba és hardverhiba miatt. Nagyobb adatrendszerek esetén rendszeres mentéssel vagy a módosítások feljegyzésével védekezünk ellene. ➤ konzisztencia.

interaktív: Lásd párbeszédéses üzemmód.

interfész (interface): Kapcsolódási/csatlakozó felület. Annak leírása, hogy miként kapcsolódhat össze két berendezés, két program; hogyan használhat egy program egy berendezést, egy másik programot; a felhasználó egy programot vagy általában a számítógépet (*user interface*). ➤ GUI.

interlacing (félképes módszer): A monitor egy képet két ciklusban rajzol meg, egyikben a páros, másikban a pá-

ratlan sorokat. A monitor egyszerűbb, olcsóbb, de a kép rosszabb.

interleave: Átlapolás. A harddiszkek *sávjain* belül a *szektorokat* korábban nem a fizikai sorrendben számozták, az 1. után például a 3. következett, és csak ezután a 2. Ennek oka, hogy a szektorokhoz általában egymást követően fért hozzá a gép, de mivel mondjuk az 1. szektor feldolgozása után a 2.-at kérte volna, noha a lemez már elfordult, így meg kellett volna várni, amíg ismét körbe ér, a 2. szektor eléréséhez. Ezen egyszerűen a számozás megváltoztatásával segítettek; a fenti példa a 2:1 interleave. A CPU és főként a harddiszk-vezérlők fejlődésével ma nincs szükség rá, a szektorokat általában fizikai sorrendjük szerint számozzák, ez az 1:1 interleave.

intermediate: Közbülső. Vannak olyan programok, például *fordítóprogramok*, amelyek több menetben állítják elő a bemenő adatokból a kimenő adatot, például a *forrásnyelvű* programból a *tárgykódot*. Az adatoknak azt a formáját, ahogy a egyik menet a másiknak tovább adja őket, *közbülső formának, közbülső kódnak, közbülső nyelvnek* nevezik. ➤ interpreter.

Internet modell: Az *OSI modell*hez hasonló hálózati modell, amely különböző számítógépes hálózatok egységes elvek alapján való felépítését és így az összekapcsolhatóságukat támogatja. Kidolgozása a 70-es években megelőzte a PC-ket és az *OSI* modellt, ezért elsősorban a középkategóriájú gépeken és a *UNIX* operációs rendszer környezetében terjedt el. Az *OSI* modellhez hasonlóan a különböző hálózati funkciókat hierarchikusan egymásra épülő rétegekre osztja.

1. Fizikai réteg. (Magában foglalja a fizikai kapcsolat létrehozásával és fenn tartásával kapcsolatos feladatokat.

Funkcionálisan az OSI modell két alsó — fizikai és adatkapcsolati — szintjével egyezik meg.)

2. Hálózati interfész réteg. (A hálózat fizikai részének *meghajtó* szoftverét tartalmazza.)

3. Internet réteg. (A gép-gép közötti kommunikáció feladatait látja el, ilyen például a címzés és az útvonal-kijelölés. Az OSI modell hálózati rétegének felel meg.)

4. Szállítási réteg. (Az OSI modell megfelelő rétegéhez hasonlóan a program-program kapcsolatot támogatja és hibaellenőrzést végez.)

5. Felhasználói réteg. (Ez jelenti azokat a programokat, amelyek használják a hálózat szolgáltatásait.)

Az OSI modellel összehasonlítva a legfontosabb elvi eltérés, hogy az Internet modellben csak a szállítási réteg végez hibaellenőrzést, míg az OSI modellben ezt minden szint külön külön vizsgálja. A másik eltérés, hogy az OSI modell szinte minden hálózatkezelési feladatot beépít, az Internet modell sokkal egyszerűbb szolgáltatásokat kínál, és a felhasználói programra több feladat hárul.

interoperabilitás: Más programokkal, különféle hardverekkel való együttműködési képesség.

interpreter: Értelmező. Egy programozási nyelven írt program kétféle szemlélettel kerülhet végrehajtásra: vagy egy *fordítóprogrammal* lefordítjuk *gépikódú* utasításokra, és ezt hajtjuk végre, vagy a programot utasításonként elolvassuk és közben végrehajtjuk. Az utóbbit *interpretálásnak* nevezzük, amelyet az interpreter hajt végre. Noha a két szélsőséges megoldásra is van példa, a közbülső megoldások is gyakoriak.

Az interpretált nyelveket manapság nemigen szokták magából a *forrásszö-*

vegből interpretálni. Első beolvasáskor rendszerint valamilyen belső ábrázolásra (*közbülső formára*) alakítják át, ezt értelmezik (és ha kell, visszaalakítják forrásszöveggé). Az átalakítás folyamatának része a *lexikális elemzés* és a *szintaktikai elemzés* (valamilyen szintig vagy teljesen). Hasonlóan: egy sor olyan programnyelv van, amelynél a lefordított programhoz egy meglehetősen nagy futtató csomagot (*runtime package*) is hozzá kell *szerkeszteni*, ami bizonyos értelemben egy csökevényes interpreter. ➤ lexika, szintaxis, szemantika, fordítóprogram, intermediate.

interrupt: Lásd megszakítás.

intersection: Lásd halmaz típus.

invertált fájl: Lásd ISIS.

IPSE (Integrated Program Support Environment): *Programozási környezet:* olyan egymáshoz kapcsolódó eszközök gyűjteménye (lehetőleg egy programba összekapcsolva), amelyekkel könnyen és hatékonyan fejleszthetünk ki programokat. *Forrásszöveg-szerkesztő, forrásszöveg-formázó, fordító, nyelvi szintű nyomkövető* stb. ➤ CASE.

írás/olvasás: Lásd fájlműveletek.

írásvédelem: Hardver vagy szoftver általi megoldás, amely megtiltja valamilyen adat vagy adatkészlet megváltoztatását. ➤ read only file, védelem.

ISA (Industry Standard Architecture): Az IBM AT gépek eredeti buszrendszerre, 16 bites adatátvitelt tesz lehetővé. A 32 bites processzorok (80386 és 80486) alkalmazásával felmerült az igény 32 bites buszrendszer iránt, amelyet az IBM a *mikrocsatornával*, a többi nagy gyártó pedig az *EISA* buszrendszerrel elégített ki. Mivel a 32 bites *adapterek* választéka viszonylag szűk és még áruk is magas, az AT *alaplapon* többsége a 32 bites CPU-k ellenére csak a 16 bites ISA busszal készül. Ezek az alaplapon

csak a memória és a processzor közötti adatátvitel 32 bites, az adapterkártyákkal a 16 bites ISA buszon zajlik a kommunikáció.

ISAM: (Index Sequential Access Method) Indexszekvenciális hozzáférés. Lásd adatállomány.

ISDN (Integrated Services Digital Network): Integrált szolgáltatású digitális hálózat.

Jelenleg a világon egyetlen igazán átfogó, információtovábbításra használható hálózat létezik, a telefonhálózat. Ezt a rendszert azonban eredetileg alacsony sebességű *analóg* hangátviteli célra tervezték, ezért — bár sokat fejlődött az elmúlt 100 évben — nagysebességű számítógépes átvitelre alkalmatlan. Az ISDN olyan — egyelőre csak tervekben, illetve kísérleti szakaszokban létező — digitális hálózat, amely sokféle, a mainál lényegesen gyorsabb információtovábbítást és szolgáltatást tesz lehetővé egyetlen hálózaton.

A hálózat terveit 1984-ben hagyta jóvá a CCITT, 1988-ban öntötték végleges formába. Ezek szerint a telefon új szolgáltatása lesz például, hogy csöngetéskor a hívó számát és nevét — amennyiben nem nyilvános állomásról beszél — kijelzi. Zárt, egyszerre több készüléket bekapcsoló hálózat alakítható ki, ún. konferenciahívás stb.

A számítógépes adatátvitelt, mivel ez digitálizált, közvetlenül támogatja max. 64 kbit/sec-os sebességgel, így *modemekre* nem lesz szükség. Mivel a költségek óriásiak, az ISDN csak fokozatosan valósulhat meg, sokáig él még együtt a korábbi, analóg telefonhálózattal. Az ISDN részének tekinthető a már üzemelő, franciaországi MiniTel rendszer.

ISIS, CDS/ISIS (Computerized Documentation System/Integrated Set of

Information Systems): Szöveges adatbázis-kezelő rendszer, amelyet az UNESCO dolgozott ki.

A tárolandó adatokat a *master fájl* tartalmazza, rekordjai mezőkből és almezőkből állnak. A rekordok tényleges információtartalom szerint változó hosszúságúak és változó szerkezetűek lehetnek.

A gyors visszakereshetőséget az *invertált fájl* vagy *szótárfájl* biztosítja. Ebbe belekerül a szövegben előforduló összes szóalak, kigyűjtve hozzá előfordulásának helyeit a master fájlban. Ennek segítségével a szavak előfordulására kérdéseket tehetünk fel, ezeket 'és'-sel, 'vagy'-gyal és egyéb *logikai* kapcsolatokkal kombinálhatjuk. A szókombinációk előfordulásáról feltett kérdésekre a választ gyorsan megkapjuk, anélkül, hogy az adatokat tartalmazó master fájlban kerestünk volna.

Hatékony rendező és igen rugalmas, kiadvány-előkészítésre alkalmas *listakészítő* formátumnyelve van.

A teljes CDS/ISIS nagyszámítógépre készült, de van kisebb kategóriában (mini: VAX és mikro: PC, WANG) futó változata is (*microISIS*). A változatok között az ISO 2709 szabvány szerinti, ún. export/import formátummal biztosíthatunk adatcserét, például az akár több százezer rekordot tartalmazó nagygépes adatbázis részét letölthetjük *microISIS*-be, vagy mikrogépen készült feldolgozásokat betölthetünk VAX-ra vagy nagygépre.

A *microISIS* saját ISIS-Pascal programozási nyelve lehetővé teszi, hogy programokkal hozzáférjünk a *microISIS* adatállományaihoz, így a szöveges adatbázis-kezelés mellett tetszőleges programozási feladatokat is elvégezhetünk (például készletnyilvántartás).

A CDS/ISIS-szel és változataival könyvtárakban és *informatikai* intézményekben gyakran találkozhatunk, mivel az UNESCO bizonyos változatokat ingyen/igen kedvezményesen terjesztett.

ISO: (International Organisation for Standardization) Nemzetközi Szabványosítási Szervezet. (Figyelem!! — nem Nemzetközi Szabványügyi Hivatal.) Nemzetközi szabványokat állapít meg; 1946-os megalakulása óta több mint 70 országban van tagszervezete. Az információtovábbításra kidolgozott szabványai jelentősek.

ISO/OSI: (Open System Interconnection) A nemzetközi szabványügyi szervezet (ISO) által javasolt *protokollmodell*. Rétegekből épül fel. A 7 réteg (*layer*) egymásra épül:

Fizikai réteg: A hardverkapcsolat mechanikai, elektromos, funkcionális jellemzőinek leírása.

Adatkapcsolati (Data-link) réteg: Kódolás, címzés, adatblokkok hibamentes átvitele két szomszédos állomás között.

Hálózati (Network) réteg: Az átviteli utak kialakítása és az ehhez kapcsolódó üzenetváltás.

Átviteli (Transport) réteg: A szállítás pontosságával, megbízhatóságával foglalkozik vég-vég viszonylatban.

Viszony (Session) réteg: Egy kommunikáció létrehozása, fenntartása, koordinálása.

Megjelenítési (Presentation) réteg: Kódkonverziók, adatformátum-átalakítások az átvitel során.

Alkalmazási (Application) réteg: A felhasználó hogyan juttathatja el adatait programtól programig.

item: A *lista* egy eleme — több közül egy. A szakmában nevezik így, noha az eredeti latin szó jelentései közül csak a 'továbbá' utal valamelyest erre a fogalomkörre. ► entry, entity.

iteráció: Ismétlés. Olyan megoldási mód, amely valamilyen értékből kiindulva, valamilyen eljárást ismételve, az előző eredményt is felhasználva egyre pontosabban közelíti meg a végső megoldást.

J

jelszó: Lásd kulcsszó.

jobb-érték: Lásd BCPL.

join (összekapcsolás): Lásd adatbázis.

joker (*dzsóker*) karakter, *wild card* character: Olyan karakter, amelyik a keresett szó bizonyos részét helyettesíti.

Általában két ilyen karakterrel találkozhatunk: a DOS-ban ?-lel jelöljük, ha az adott pozícióban bármilyen karakter

állhat; *-gal jelöljük, hogy az adott részen tetszőleges jelsorozat lehet (de az előtte és utána álló karaktereknek meg kell egyezniük). A DOS és a *UNIX* parancsai olyan értelemben használják, hogy a parancs végrehajtódik mindazon fájlokra, amelyeknek a neve illeszkedik az így megadott névhez.

joystick: Lásd botkormány.

jumper: A számítógép áramköreit tartalmazó paneleken lévő kapcsoló különböző célokra; beállításának megváltoztatása kívülről nem, általában csak a gép kibontásával lehetséges.

justification: Lásd sorkiegyenlítés.

K

K&R: A C nyelv első leírása. Lásd C nyelv.

kalkulátor: Zsebszámológép, illetve olyan program, amelyik egy zsebszámológépet szimulál. (Rendszerint a numerikus *klaviatúrát* használja, amely elrendezésében azonos a zsebszámológépek *billentyűzetével*.)

kalózkodás: Programok jogtalan lemásolása, továbbadása, eladása. A *kalózmásolat* olyan programpéldány, amelyet jogtalanul használnak. A *kalózverzió* olyan példány, amiből a jogosultság ellenőrzését kiirtották. ➤ copyright.

kapcsolóprogram: Lásd linker, linkage editor.

karakter típus (char(acter)): Egy 0 és 256 közötti szám, amely egy *ASCII karakterkódot* jelöl. A látható karaktereket általában idézőjelek között lehet megadni, a nem látható vezérlőkarakterek megadására nyelvenként különböznek a konvenciók. Az *értékadás* és az összehasonlító műveletek alkalmazhatók. ➤ adattípus.

karaktergenerátor: Lásd képernyőadapter.

karakterkód: A számítógépek az emberekkel jelenleg elsősorban írásban tartanak kapcsolatot. A karakterkészlet azon betűk, számok, írásjelek halmaza, amelyeket a felhasználó a *billentyűzeten* keresztül beadhat, illetve amelyek a képernyőn vagy a nyomtatón megjelennek; továbbá a szöveg formátumát vezérlő karakterek: kocsivissza, soremelés, tabulátor, lapdobás; valamint egyéb vezérlőkarakterek, például: Ctrl-C, End-of-file, Esc stb., amelyek a programoknak, az operációs rendszernek, berendezések

hardverének, adatátviteli eszközöknek és egyébeknek szólnak.

Régebben számos különböző kód volt forgalomban (telexkód, EBCDIC stb.). Mára az *ASCII kód* (American Standard Code for Information Interchange) egyeduralmává vált. Ez egy 8 bites kód, azaz 0-tól 255-ig terjedő számokat rendel a karakterekhez. Az alsó 7 bit értékéhez tartozó kódokat az *ANSI* 1977-ben rögzítette, majd az *ISO* is elfogadta. A nyolcadik bit eredetileg paritásbitként (az átvitel/transzformáció helyességének ellenőrzésére) szolgált. Később igény merült fel további nemzeti karakterekre, egyéb grafikus jelekre és vezérlőkarakterekre. Ezeket a nagy cégek egymástól függetlenül és egymástól különbözően helyezték el a felső 128 értéken. Ezen kódok egyike sem tartalmazta a magyar „ő” és „ű” karaktereket.

Kezdetben Magyarországon is számos egymásnak ellentmondó kód kiosztás született, de mára többé-kevésbé szabványnak számít a CWI kód, amely a Számítástechnika című újság iniciatívájára jött létre. Ennél kisebb mértékben használják a *Ventura kiadványszerkesztő* program magyar karakterkészletét is. Legújabbán a *Windows* kapcsán jelent meg a *852-es karakterkészlet*, amelyben a 128-nál nagyobb értékeken a kelet-európai népek betűi vannak, tehát a magyar betűk is, de nem tartalmazza (vagy máshol tartalmazza) a régebbi karaktereket, így a nyugateurópai betűket és a keretrajzoló karaktereket, továbbá erősen különbözik a korábbi magyar karakterkiosztástól. Egyelőre éles ellenkezést váltott ki.

Az ISO-nál munkák folynak egy olyan karakterkészlet elkészítésre, amely a latin betűkön kívül az egyéb betűírásokat (görög, cirill, arab stb.), sőt a kínai, japán, koreai írást is magában

foglalja. Hasonló szabványkísérlet az amerikai *Unicode*.

karakterisztika: Lásd valós típus.

katalógus (directory): Lásd könyvtár.

katenáció, konkatenáció: Lásd string típus.

Kedit: Az IBM nagygépek XEDIT programjának PC-s megfelelője. Sajátossága, hogy a javítás fájlból is vezérelhető.

keret(program, -rendszer): Lásd shell.

Kermit: *Aszinkron átvitelre* kifejlesztett fájlátviteli *protokoll*. A Columbia Egyetemen fejlesztették ki nagygépek és mikrogépek közötti átvitelre. Az információt max. 96 bájtos egységekben viszi át.

kernel: Mag. A program központi része, amely intézkedik arról, hogy mikor melyik programrész fusson, gazdálkodik az erőforrásokkal. Operációs rendszerekkel, párhuzamosan futó részeket tartalmazó programokkal kapcsolatban használatos szó.

keyboard: Lásd billentyűzet.

keyword: Lásd kulcsszó.

képernyő, kijelző, megjelenítő, display, screen (Visual Display Unit, VDU): A számítógép egyik legfontosabb kimeneti perifériája, erre írja a felhasználó számára az adatokat. A legelterjedtebb típus a katódsugárcsöves megjelenítő, működési elve megegyezik a televíziónál alkalmazottal. Lehet egyszínű (monokróm), ami a gyakorlatban fekete-fehér, sárga-fekete vagy zöld-fekete színpárost jelent, de lehet színes is. A képernyőhöz mindig a megfelelő üzemmódú *képernyőadapert*t kell használni. A katódsugárcső fő hátránya nagy tömege és viszonylag magas energiafogyasztása.

A hordozható számítógépekhez fejlesztették ki a *folyadékkristályos (LCD)*

megjelenítőket. Ezek tömege és energiafogyasztása lényegesen kisebb, felbontásuk pedig már eléri az átlagos katódsugárcső felbontását. Hátrányuk, hogy rosszabban olvashatók, és hogy a színes típusok ára igen magas. A *gázplazmás kijelzők* működése hasonlít a neonrekláméra, a képernyő egy pontjának megfelelő méretű cellába zárt gáz világít. Az *elektrolumineszcens kijelzőkben* pedig fénykibocsátó foszforvegyületeket helyeznek el a képernyő egy pontját jelentő elektródák közé. Mindkét típus felbontása alkalmas EGA szintű grafikus megjelenítésre, olvashatóságuk pedig az aktív fénykibocsátásnak köszönhetően igen jó. Hátrányuk magas árak és a folyadékkristályos képernyőkéhez képest nagy energiafogyasztásuk. Egyelőre csak monokróm változatban léteznek, színesben nem.

képernyőadapter, monitoradapter, videóadapter, videokártya: A számítógép képernyőjét kezelő részegység. Minősége és üzemmódjai meghatározzák az adatok megjelenítési formáját. A képernyőadapterek két fő működési módja az *alfanumerikus* és a *grafikus*. Alf numerikus módban a képernyőre csak olyan betűk, jelek rajzolhatók, amelyeket a karakterek pontjait tároló memória, a *karaktergenerátor* tartalmaz. Ekkor a megjeleníthető sorok és az egy sorba írható betűk számával jellemzik az üzemmódot. Grafikus módban a képernyő képpontonként kezelhető, így tetszőleges ábra rajzolható. Ebben a módban a vízszintesen és függőlegesen megjeleníthető pontok számával adják meg a felbontást.

Az alfanumerikus mód a kétségtelen korlátai mellett kisebb *képernyő-memóriát* igényel, gyorsabb és egyszerűbb képernyőkezelést tesz lehetővé. A grafikus mód nagyobb felbontása látványo-

sabb, de lényegesen több képernyő-memória szükséges hozzá.

képernyő-generátor: Lásd generálás.

képernyő-memória, képernyő-RAM, videómemória, videó-RAM: A képernyőn megjelenő információt tároló memória. A *képernyőadapter* központi vezérlőáramkőre másodpercenként 50-70-szer olvassa ki a képernyő-memóriát, és a tartalmát megjeleníti a képernyőn. Alfabetikus módban a megjelenítendő karakterek *ASCII kódját* tartalmazza, grafikus módban az egy képponthez tartozó bitek a pont színét kódolják, számuk a használható színek számától függ.

képernyőmentő: Lásd cut&paste.

kézikönyv (manual, handbook): A programokról hosszabb-rövidebb leírások készülnek. A programozói kézikönyv a szoftverek munkáját segíti. A felhasználói kézikönyv a potenciális felhasználóknak, a karbantartói kézikönyv a karbantartóknak szól. Azon programok esetében, amelyek valamilyen szabványt valósítanak meg, a könyv gyakran csak az adott *implementáció* sajátosságait, a szabványtól való eltéréseket írja le. ► szabvány, jelentés.

kiadványszerkesztő, DTP (DeskTop Publishing): Ezek a programok hasonlítanak a szövegszerkesztő programokhoz, hiszen a végeredmény itt is többnyire egy szöveg lesz. Akkor beszélhetünk azonban — megkülönböztetésül — kiadványszerkesztésről, amikor az eredmény grafikai megjelenésére legalább akkora gondot fordít a program, mint a szöveg létrehozására. Minden kiadványszerkesztő tartalmaz egy beépített szövegszerkesztőt, amellyel még az utolsó pillanatban is könnyen meg lehet tenni apró változtatásokat, javításokat a szövegben.

A modern kiadványszerkesztők egyik alapelve a *WYSIWYG* (What You See Is What You Get), azaz hogy körülbelül olyan formában tudja megmutatni a szerkesztett dokumentumot, mint amilyen formában az a nyomtatásban megjelenik. Tehát már a képernyőn látni lehet a különböző *betűtípusokat*, méreteket, a képek helyét (esetleg magát a képet is). Ez szükségessé teszi a nagyfelbontású grafika használatát és azt, hogy a dokumentumot leíró fájlba ne csak szöveg kerüljön bele, hanem a formátumot vezérlő utasítások is. A képernyőn és a nyomtatásban megjelenő kép természetesen nem lehet teljesen azonos, hiszen a két berendezésnek bizonyára más a felbontóképessége. A nyomtató berendezések általában nagyobb méretben és sokszorosan finomabb felbontással tudnak dolgozni, mint a képernyő.

A kiadványszerkesztők másik fontos tulajdonsága, hogy képesek alkalmazkodni a különböző berendezések lehetőségeihez, a *mátrixnyomtatótól* a *lézernyomtatóig* keresztül a nyomdai levilágítóig. Újabban ezt segíti a *PostScript* nyelv is, amely a nyomtatandó dokumentumok leírására szolgál, és amelyet egyre több *printer* ért meg. ► font, szövegszerkesztő.

klaviatúra: Lásd billentyűzet.

klón (clone): Utánzat, másolat. Elsősorban a neves cégek gépeiről készült olcsó másolatokat nevezik így. Néha a drága szoftverrendszereket helyettesítő olcsóbb, de eredeti programokat is így hívják.

kommentár (comment): A program *forrásszövegéhez* fűzött magyarázat. Az eleje és a vége meg van jelölve, a *fordítóprogram* figyelmen kívül hagyja. A program forrásszövegével együtt az úgynevezett fejlesztői vagy karbantartó

dokumentációhoz tartozik. Ezért a szabványos programfejlesztő módszerek előírásokat adnak a kommentezésre. Célszerű, ha a program javításakor az eredeti kommentárt meghagyjuk, és újat fűzünk hozzá a hiba okáról és a javításról.

komoly hiba: Lásd hiba.

kompatibilis, kompatibilitás (compatibility): Két berendezés vagy program, vagy egy berendezés és egy program akkor kompatibilis egymással, ha minden változtatás nélkül képesek együttműködni. Egy termékcsalád felfelé kompatibilis, ha az újabb változatok változatlanul képesek használni a régi berendezéseket, programokat, adatokat.

Egy berendezés *szoftverkompatibilis* a másikkal, ha képes annak a programjait futtatni. Hardverek kompatibilitásán azt értjük, hogy helyettesíthetők egymással, illetve összekapcsolhatók egymással. Egy alkatrészt vagy részegység *plug-in kompatibilis* egy másikkal, ha a helyére beszerelhető vagy csatlakozó-zsinórral összeköthető a két berendezés. Egy berendezés teljesen kompatibilis (*true compatibility*) egy másikkal (rendszerint egy nagynevű cég termékével), ha szoftverkompatibilis és a másikkal csatlakoztatható perifériák és egyéb hardverek hozzá is köthetők. Vannak ezzel szemben teljességgel *inkompatibilis* rendszerek is, amelyek egymással semmilyen együttműködésre nem képesek. ➤ portabilitás, változat.

komplemens: Olyan művelet, amelyhez egy részhalmazt kell megadni, és az alaphalmaz azon elemeit adja eredményül, amelyek nem voltak a részhalmazban. ➤ halmaz típus.

koncentrátor: Egy PC-re alapkiépítésben nem túl sok periféria csatlakozhat. A koncentrátor olyan berendezés, amelyet egy periféria helyére bekötve,

ehhez már több (8-16-32) periféria kapcsolható. Többfelhasználós rendszerekben, például *UNIX*-ban elsősorban a terminálok számának növelésére használják. ➤ terminálkoncentrátor.

konfiguráció, konfigurálás: A konfigurálás az az eljárás, amelynek során összeállítjuk azt a konfigurációt, amelyben a számítógép dolgozni fog; azaz megadjuk, hogy a hardvernek milyen (rész)egységei vannak, azokat milyen paraméterekkel és milyen szoftverekkel használjuk.

konfigurációs fájl: Lásd *CONFIG.SYS*.

konjunkció (logikai és): Lásd logikai típus.

konkatenáció, katenáció: Lásd string típus.

konstans (constant): Lásd deklaráció.

kontroller (controller): Valamelyik számítógép-részegység, általában valamilyen periféria vezérlését ellátó áramkör vagy *adapterkártya*.

kontrollösszeg (checksum): Adatállományok összegéből képzett ellenőrző szám. Rendszerint nem a benne lévő bájtok egyszerű összege, hanem valamilyen *iteratív* képlettel számítják. (Az egy bájjal hosszabb sorozat összege valamilyen képlettel számolódik a sorozat összegéből és az utolsó bájt értékéből.) Annak ellenőrzésére való, hogy az adatállományt nem módosították-e. ➤ CRC.

konverzió: Adatátalakítás. Egy adat vagy adatsorozat átalakítása egyik formáról a másikra, valamely kódból a másikba. Például: a *bináris számok* átalakítása olvasható karaktersorozattá, *analog jelek* átalakítása bináris számmá, képeket tartalmazó fájlok átalakítása egyik fajta leírásból a másik fajta leírásba, stb.

konzisztencia, konzisztens (consistent): Egy fájl, adatbázis, szabály-

rendszer, általában véve egy adatkészlet konzisztens, ha nincsenek benne egymásnak ellentmondó adatok, illetve szabályok. Megsértését *inkonzisztenciának* nevezzük. Ilyen hiba könnyen keletkezik akkor, ha az adat több példányban létezik és egyszerre többben módosíthatják. Inkonzisztenciához vezető tipikus eset például, ha egy program *forrásszövegét* egy időben ketten módosítják. A védekezés hatáskörök és *kulcsszavak* definiálásával és az egyidejűség kizárásával oldható meg. A másik gyakori ok, ha miközben az egyik adatot már átírtuk, de a másikat még nem, és éppen ilyenkor egy *végzetes hiba* következik be. ► integritás.

koprocesszor (coprocessor): *Társprocesszor*. Speciális feladatot ellátó, utasításkészlettel rendelkező, processzor szintű áramkör. Az ilyenek önálló működésre nem képesek, utasításaikat a központi, a futó programot végrehajtó CPU-tól kapják — innen az elnevezésük. Speciális felépítésük miatt az adott feladatot gyorsabban oldják meg, mint a CPU, amely ezalatt mással foglalkozhat. Így a teljes rendszer teljesítménye növelhető. PC-kben a lebegőpontos számításokhoz, a képernyő vezérléséhez és a hálózatkezeléshez előnyös a koprocesszor.

kód (code): Valamilyen információ rögzítésére alkalmas jelkészlet. (Lásd például: *karakterkód*.) Kódolásnak nevezzük valamilyen információ megadását kódokkal. Néha a program leírását is kódolásnak nevezzük, megkülönböztendő a programtervezéstől, a *programbelövéstől* és a programozás egyéb fázisaitól. *Átkódolásnak* nevezzük, amikor az információt egyik kódból a másikba átírjuk. Ennek sokféle célja lehet: az információ titkosítása, tömörítése, átvitele egyik rendszerből a másikba, vagy

éppen *redundáns* formába való átírása — a sérülésektől védendő. *Dekódolás*: az átkódolás visszaállítása.

kódoptimalizálás: Lásd programoptimalizálás.

könyvtár (library): Fájlok gyűjteménye. A modernebb operációs rendszerekben a könyvtár hierarchikus szervezésű, azaz egy könyvtárban nemcsak fájlok, hanem alkönyvtárak is lehetnek, és ezeknek is lehet alkönyvtára. Minden könyvtárhoz tartozik egy *katalógus*, más néven *directory*, amely tartalmazza a fájlok nevét, születési időpontját, esetleg az utolsó módosítás és az utolsó hozzájárulás időpontját, továbbá állapotjelzőket, hozzáférési kulcsokat stb. A *gyökérvényvtár* (*root directory*) a hierarchia tetején álló könyvtár (főkönyvtár) katalógusa. A DOS-ban minden egyes diszken (*partícióban*) van egy root directory, ebbe valamilyen szinten (alkönyvtárakon keresztül) minden fájl beletartozik. *Path-nak* (elérési útnak) nevezzük alkönyvtárnevek azon sorozatát, amin keresztül az adott fájl elérhető. Az operációs rendszer lehetővé teszi, hogy bizonyos útvonalakat előre kijelöljünk, és ha egy fájlnevet adunk meg, az azok által kijelölt könyvtárakban is keresni fogja.

körlevélíró (mail merge): Olyan program, amely egy levélséma és egy címlista alapján névre szóló leveleket készít. Lehet *szövegszerkesztőbe* beépítve, de lehet önálló program is.

közbülső forma, közbülső kód, közbülső nyelv: Lásd intermediate.

közvetlen hozzáférés: Lásd fájl típus.

kulcsszó (keyword): 1. Olyan jelsorozat, amelyet a felhasználónak azért kell begépelnie, hogy programokhoz vagy adatokhoz hozzáférhessen. *Jelszónak* (*passwordnek*) is mondják.

2. Egy programozási nyelvben egy olyan szó, amelynek az értelme és szerepe előre rögzített.

kurzor (cursor): A képernyőn megjelenő jel, amely azt mutatja, hol dolgozunk.

Karakteres üzemmódban egy karakterre mutat, és karakterenként lép tovább. Grafikus üzemmódban a képernyő

egy pontjára, illetve az azt magába foglaló ábrára mutatunk rá. A *klaviatúra* nyilakkal jelölt gombjaival, *egérrel*, *pozicionáló gömbbel* mozgatható. A mozgás nem automatikus, attól függ, mi van beírva az adott programba. ► képernyőadapter.

külön fordítás: Lásd moduláris programfelépítés.

L

label: Lásd címke.

LAN (Local Area Network): Lásd helyi hálózat.

laptop: Hordozható, akkumulátorról is működő számítógép. A méretek és a fogyasztás csökkentésével sikerült az asztali (desktop) számítógépek teljesítményét táskameéretű számítógépbe építeni. Akkumulátorról 2,5-4 órán át működtethetők.

layer: Szint, réteg. Lásd ISO/OSI.

LCC (Leadless Chip Carrier): Az *integrált áramkörök* tokozási formája, amelynél a kivezetések a tok kerületének mind a négy oldalán helyezkednek el. Ennek némileg módosított formája a **PLCC (Plastic Leaded Chip Carrier)**.

LCD képernyő: Lásd képernyő.

LED dióda (Light Emitting Diode): Fénykibocsátó dióda. Elterjedt félvezetőeszköz, amely áram hatására a típusától függő színű fénnel világít.

levelezőrendszer: Lásd e-mail.

lexika: A program részint előre megadott *kulcsszavakból* és jelekből épül fel, részint olyan szavakból és számokból, amelyeket a felhasználó ad meg. A lexika szabályai megmondják, hogy milyen fogalmat mivel jelölhetünk, és milyen különböző formákban írhatunk le. Ilyenek azok a szabályok is, amelyek megmondják, hogy bizonyos pozícióban (a sor elején, valamilyen *kulcsszó* után, stb.) mit jelent a szó. A lexikális elemző a *fordítóprogramnak* az a része, amely a szöveget szavakra bontja és megállapítja, hogy melyik mire szolgál (*azonosító, kulcsszó, műveleti jel, szám, címke* stb.). A modern nyelvek lexikai szabályai leírhatók egy *véges automatával*, és

egy ilyen leírásból rendkívül hatékony ellenőrző/analizáló program *generálható*.

lépésenkénti végrehajtás: Lásd debugging.

lézerprinter: Lásd nyomtató.

library: Lásd könyvtár.

LIM: Lásd expanded memória.

linear reading: Folyamatos olvasás. Lásd deklaráció.

linker, linkage editor: Nevezik kapcsolóprogramnak, szerkesztőprogramnak, szerkesztőnek is. (Össze)szerkesztő program. A program általában több *modulból* épül fel. A modulok részben úgy keletkeznek, hogy a felhasználó ezekre bontja a programját, részben úgy, hogy a program automatikusan hivatkozik bizonyos rendszermodulokra (input/output, fájlkezelés stb.), amelyeket a rendszer szolgáltat neki. Ez a rendszerprogram összekapcsolja ezeket, feloldja a modulokban levő külső hivatkozásokat. Az így kapott program még nem olyan, mint a memóriában majdan lévő *bináris* alak, hanem egy úgynevezett *áthelyezhető forma (relocatable form)*, amelyet a rendszer *betöltőprogramja (loader)* visz be és alakít át bináris alakra. ► modularitás, fordítóprogram.

LISP: (LISt Processing language) Szimbólum-manipulációs, függvény-szerű nyelv. 1960-ban született J. McCarthy 1958-as cikke nyomán. Amerikában igen népszerű nyelv volt egyetemi körökben és *MI-alkalmazásokban*. Komolyabb feladatokhoz igen nagy és gyors gépet igényel, speciális gépeket is építettek ilyen céllal. Utóbbi időben népszerűsége picit csökkent.

Elemi adattípusai az ún. atomok, a *szimbólumok* (egy-egy *alfanumerikus* karakter sorozat) és a számok. Egyetlen összetett adattípusa a *lista*. 7 (nagyon egyszerű) listakezelő alapfüggvénye van (+ néhány numerikus), minden más

függvényt, így saját *interpreterét* is — az ún. eval függvényt — ezekből építi fel. A függvények formája is lista (az első elem a függvénynév, a többi *argumentum*). A program egy sor listaként megadott függvénydefinícióból és egyetlen függvényhívásból áll. Képes a futása közben létrehozott listák végrehajtására is.

Több dialektusa van, a legelterjedtebb a Common LISP, másik fontos dialektus az INTERLISP. Az újabb LISP-ek már átmenetet képeznek a szokásos *procedurális nyelvek* és az eredeti tiszta (pure) LISP között.

lista: 1. Sornyomtatón kinyomtatott fájl.

2. Két *pointerből* álló cellákból felépített adatszerkezet. A listának van egy kezdő cellája. Minden cella második *pointere* a lista folytatására mutat, vagy a lista végét jelzi (NIL *pointer*). A cella első *pointere* vagy egy értékre mutat, vagy egy allistára. Vannak általánosításai is: kétirányú lista (a cellában van egy, a lista előző elemére mutató *pointer* is), ciklikus lista vagy gyűrű (a lista vége jelzés egyben a lista elejére is visszamutat).

3. Elemek sorozata, ahol elemenként tulajdonképpen bármit érthetünk, de többnyire összetartozó adatokról van szó.

literál: Lásd deklaráció.

loader (betöltőprogram): Lásd linker.

lock: Lezárás. Egy adatra vagy berendezésre érvényesíthető korlátozás, mely által (szoftveres úton) ideiglenesen elzárjuk a többi felhasználó elől. A műveletek befejezése után felszabadítjuk (*unlock*). ➤ *shared*.

logikai típus (Bool(ean)): Két értéke van, az igaz (True) és a hamis (False). A műveletek: 'és' (*konjunkció*), akkor igaz, ha mindkét *argumentuma* igaz;

megengedő 'vagy' (*diszjunkció*), akkor igaz, ha legalább az egyik *argumentuma* igaz; 'kizáró vagy' (*exclusiv or*), akkor igaz, ha egyik *argumentuma* igaz és a másik nem; tagadás (*negatio*), akkor igaz, ha az *argumentuma* hamis. (Ismeretes, de az adattípus szintjén nem használatos művelet a következtetés (*implikáció*): akkor hamis, ha az első *argumentuma* igaz és a második hamis. Az 'a implikálja b-t' művelet az 'a tagadása vagy b' formulával helyettesíthető. Egy bonyolultabb logikai összefüggés több különböző alakban is megadható.)

➤ adattípus.

logo: Embléma, bejelentkező ábra.

LOGO: Gyerekek programozástani-tására készült nyelv.

lokális: Helyi. Valamilyen korlátozott körben érvényes. Például olyan *változó*, amely csak egy *függvényen* vagy *modulon* belül használható. Amikor azt mondjuk, hogy egy *változó* *külső* és nem *lokális* valamilyen *függvényre* nézve, ez nem jelenti azt, hogy ne lehetne *lokális* valamilyen nagyobb egységre, például egy *modulra* nézve. ➤ *globális*.

lokális optimalizálás: Lásd program-optimalizálás és optimum.

lokális optimum: Lásd optimum.

lokális hálózat: Lásd helyi hálózat.

Lotus 1-2-3: A Lotus Development Corporation *táblázatkezelő* és *üzleti grafikai* programja, igen népszerű, bár meglehetősen tárigényes. Utódja a *Lotus Symphony* *táblázatkezelőből*, *grafikonrajzolóból*, *szövegszerkesztőből*, *adatbázis-kezelőből* és *kommunikációs modulból* áll.

low-level language: Lásd alacsony szintű nyelv.

LQ (Letter Quality): Levélminőségű nyomtatás. A 24 tűs *mátrixnyomtatónknál* választható üzemmód és minőség.

M

Macintosh: Az Apple cég személyi-számítógép-családja. Tagjai a Motorola 68000-es processzorcsaládjára épülnek, a processzorok fejlődésével jelentek meg az egyre nagyobb teljesítményű Macintosh gépek. Sem szoftver, sem hardver szempontból nem kompatibilisek az IBM PC-k családjával. Erősségük a PostScript nyelven alapuló egységes *fontkezelés* és a teljesen grafikus felhasználói interfész, a háttértárolókhoz az *SCSI* interfészt használják. Operációs rendszerüket a gépekben lévő *ROM* tartalmazza, ez jogilag védett — így nincsenek hasonmás (*klón*) típusok, szemben az IBM PC-kkel; áruk ezért viszonylag magas.

macro: Lásd makró.

macro assembler: Lásd assembler.

magas szintű nyelv (high-level language): A magas szintű nyelv olyan fogalomkészletben gondolkozik, hogy ez lehetővé teszi a problémák megoldásának megfogalmazását a konkrét számológéptől eltekintve is. Az már a *fordítóprogram* (vagy az *interpreter*) feladata, hogy a program a lehető leghatékonyabban kerüljön végrehajtásra az adott gépen. A programozási nyelvek többsége figyelembe veszi a mai gépek általános tulajdonságait, és a programíró is gondol ezekre. A *nagyon magas szintű nyelvek* támogatói viszont amellet vannak, hogy elég, ha az ember csak a feladatot fogalmazza meg, a gép majd valamilyen igen általános *algoritmussal* megtalálja a megoldást, a hatékonyság nem számít, a gép olcsóbb az embernél. ➤ alacsony szintű nyelv.

mail merge: Lásd körlevélíró.

mailbox: (levélszekrény) Lásd elektronikus levelező rendszer.

mainframe: Nagyszámítógép. A 80-as évek közepéig a számítógépeket teljesítményük és ezzel arányos méreteik révén hagyományosan mikro-, mini- és nagyszámítógép osztályba sorolták. A nagyszámítógépek külön légkondicionált géptermet igényeltek, és többnyire kötegelt (*batch*) feldolgozással működtek, bár voltak *valós idejű* (*real-time*) alkalmazások is. A modern nagyszámítógépek operációs rendszerei egyaránt támogatják a valós idejű munkákat; az előre beütemezett, batch jellegű munkákat, amelyek nem igényelnek párbeszédet a felhasználóval; és az *interaktív* munkákat a géphez kapcsolt terminálkon és hálózaton keresztül.

Ma is dolgoznak nagyszámítógépekkel (a legismertebb gyártók/típusok: IBM, Honeywell Bull, Borroughs), amelyeket számítóközpontban helyeznek el, ezeket ott külön kezelőszemélyzet működteti (az *operátorok*), hálózati központként szolgálnak, gigantikus méretű adatbázisokat kezelnek, különleges kapacitást igénylő munkákat futtatnak. Ezeket a felhasználók igen távolról, hálózaton keresztül érik el.

maintenance: Karbantartás, fenntartás.

make-file: A *MAKEFILE* nagyobb *C* programokhoz tartozó fájl, amely megkönnyíti a program javítását, újrafordítását, összeszerkesztését. A *MAKEFILE* lényegében azt írja le, hogy melyik *forrásfájl* megváltoztatása után milyen más fájlokat kell újrafordítani.

makró (macro): 1. *Billentyűmakró:* Olyan kényelmi lehetőség, ami által a *klaviatúra* egy billentyűjéhez vagy a billentyűk kombinációjához (például: Alt-F1) egy karaktersorozatot rendelünk. Amikor a billentyűt lenyomjuk, a gép

úgy érzékeli, mintha az egész karakter-sorozatot begépeztük volna.

2. Szövegmakró, programmakró: Egyes nyelvekben mód van arra, hogy makrókat definiáljunk. A *makródefiniációban* meg kell adni egy makrónevet és egy *paraméterlistát*, majd egy *makró-törzset*: egy olyan szöveget, amelyben szerepelhetnek a makró paraméterei. A definíció után, ha leírjuk a makrónevet és egy aktuális paraméterlistát (ezt *makróhívásnak* nevezzük), akkor ennek helyébe bemásolódik a makró törzse, de úgy, hogy a makróparaméterek helyébe az aktuális paraméterek kerülnek. Az így keletkezett szöveget fordítja le a nyelv *fordítóprogramja*. ► szubrutin.

MAN (Metropolitan Area Network): Lásd városi hálózat.

Manchester-kódolás: A helyi hálózatok alapsávú, analóg modulációt nem alkalmazó átvitelénél bevált kódolási forma. A 0 értékű biteket az alacsony-magas jelátmenet, míg az 1-es biteket a magas-alacsony átmenet reprezentálja. A különbségi Manchester-kódolásnál a 0-t a bit elején alacsony-magas átmenet jelzi, 1-nél a bit elején nincs átmenet, a bit közepén minden esetben van jelszint-váltás.

mantissza: Lásd valós típus.

manual: Lásd kézikönyv.

master (mester): Az eredeti példány. Az esetek nagy részében a megvett programot tartalmazó diszket nem fogjuk munkára, hanem amennyiben lehet, írás-védetté tesszük, és a másolataival dolgozunk. Az eredeti példányt gondosan őrizzük. Csak akkor lesz rá szükség, ha a másolat megsérült.

maszk (mask): Egy bitsorozat. Ezzel és egy ugyanilyen hosszú állapotleíró bitsorozattal bitenként 'és' műveletet hajtunk végre. A keletkező állapotvektorban csak azok a bitek maradnak meg,

amelyeket a maszk engedélyezett. Ez a *maszkolás*.

másolásvédelem: Lehetetlenné teszi, hogy a programot másolatról futtassák; de működhet úgy is, hogy csak adott ideig vagy futáásszámig engedélyezi a program használatát. Célja a felhasználó korlátozása abban, hogy a programot tetszése szerint vegye igénybe. Megakadályozza abban, hogy továbbadja, több helyen futtassa, hosszabb ideig alkalmazza, mint amilyenre engedélyt kapott. A programot csak az eredeti lemezzel lehet futtatni, vagy ha valamilyen hardver-védelmieszközt alkalmaznak, akkor csak annyi példánya működhet, ahány ilyen eszközt a forgalmazó a rendelkezésükre bocsát.

A másolásvédett programok általában tiltakoznak a védelem kiiktatása ellen is, és ilyenkor retorziókkal élnek. A leggátálástalanabb másolásvédelmek a retorziók során alkalmazzák a *vírusprogramozás* és az ezzel kapcsolatos technológiai eljárások minden eddigi felfedezését. A romboló másolásvédelem veszélyes, mert el is szabadulhat, a program véletlen sérülését is a másolásvédelem eltávolítási kísérletének minősítheti; egyes védelmek pusztán a másolás tényére is elindulnak. Ezeknek a büntető védelmeknek a hatásai felérnek a vírusok okozta kárral.

A másolásvédelem hátrány és veszély a felhasználónak. Az ilyen programok forgalmazása — ha van olyan helyettesítő termék, amelyik védelem nélküli — többnyire megbukik a felhasználók bojkottján. (Magyarországon a *Quattro* azért tudta elhódítani a *Lotus* és a *Symphony* felhasználóit, mert a Borland cég a *Quattro* minden verzióját védelem nélkül, csak sorszámkóddal forgalmazta.)

mátrixnyomtató: Lásd nyomtató.

MCGA (Multi Color Graphics Array): Kizárólag az IBM PS/2 kategóriájú gépeinek 25-ös és 30-as modelljeibe épített *képernyőadapter*. Kezelése és üzemmódjai — egy kivétellel — megegyeznek a VGA-éval; nincs 320x200-as felbontású, 256 színű grafikus módja. Az MCGA-t nem vették át más gyártók.

MDA (Monochrome Display Adapter): Az IBM PC és IBM PC/XT gépek első monokróm, azaz egyszínű megjelenítést eredményező *képernyőadaptere*. Csak *alfanumerikus* üzemmódja volt 80x25-os formátumban. A grafikus képességek hiánya miatt gyorsan kiszorította a Hercules cég vele kompatibilis *videóadaptere*.

meghajtó: A számítógép valamelyik (perifériáját) működtető eszköz. Legtöbbször a floppy-val vagy a harddiskkel kapcsolatban emlegetik, megkülönböztetve az adathordozóktól. Angol neve ez esetben *drive*. A hardvert — például az egeret — kezelő szoftvert is így hívják (meghajtószoftver, meghajtóprogram). A szóhasználat ebben az esetben *driver*.

megszakítás (interrupt): Esemény, melynek hatására a CPU félbeszakítja a éppen futó program végrehajtását, és a megszakítást kiszolgáló rutin kapja meg a vezérlést. A megszakítások kezelése fejlettebb rendszerekben igen bonyolult feladat, hiszen a megszakításokat előidéző események fontossága/sürgőssége egymáshoz képest nagyon különböző lehet. A megszakításoknak éppen ezért *prioritásai* és szintjei lehetnek, és kezelésük, feldolgozásuk eszerint kerül sorra. A megszakításkezelés után a program folytathatja a munkát.

A megszakítást kiváltó okok a rendkívüli eseménytől a rendszeres munkához tartozó tevékenységekig terjednek. Legfontosabb fajtái: a külső berendezésből

vagy részegységből érkező jel (*külső megszakítás*), a felhasználó által kiadott utasítás (*szoftvermegszakítás*), végrehajthatatlan utasítás (például hibás memóriacím, nullával való osztás stb). A PC-ben a külső berendezésekkel a felhasználó általában megszakítások által lép kapcsolatba (például fájl megnyitása, írás/olvasás stb.).

Különösen *multiprogramozott* rendszerekben vannak olyan szituációk, amikor bizonyos interruptok bekövetkezése megzavarhatná a munkát. Ilyenkor az interruptot *maszkolni* kell. A külső megszakítások érvényesülése letiltható vagy nem letiltható, azaz nem maszkolható (*NMI* = Non Maskable Interrupt).

A megszakítást kiszolgáló rutinok címei — a CPU típusától függően — gyakran a memória egy kitüntetett területén található, a *megszakítási táblában*. Külső megszakítás esetén a megszakítást kérő eszköz egy egybájtos adatot, *megszakításvektort* küld a CPU-nak, amely ez alapján választja ki a megszakítási táblában a megfelelő *megszakításrutin* címét.

memória: A processzor és a perifériák mellett a számítógép harmadik fontos része. Itt vannak tárolva a CPU által végrehajtandó programok és a feldolgozásra váró adatok. A memóriák két fő csoportra oszthatók aszerint, hogy megőrzik-e tartalmukat a tápfeszültség megszűnése után, vagy mindent elfelejtenek: törlődnek.

A tartalmukat őrző típusok legnagyobb hátránya, hogy a processzor csak olvasni tudja őket, írni nem. Innen kapták a nevüket: *ROM* (Read-Only Memory) — csak olvasható memóriák. Tartalmukat a gyártáskor építik be, ez többé nem változtatható. Léteznek a felhasználó által elektromosan írható típusok is, ezek az *EPROM*ok (Electrically Prog-

rammable ROM). Programozásuk azonban nem a számítógépben, hanem egy külön erre a célra készült *EPROM-égetővel* lehetséges. A beírt tartalom ultraibolya fényvel törölhető, ezért van kvarcüveg ablak a token.

Az elektromosan törölhető típusok az *EEPROM*ok (Electrically Erasable PROM). A processzor által írható-olvasható, azaz tetszőleges hozzáférésű memóriák: *RAM*-ok (Random Access Memory). Két csoportra oszlanak. A statikus *RAM*-ok (*SRAM*) tartalmukat a tápfeszültség fennmaradásáig megőrzik. A dinamikus *RAM* (*DRAM*) a beírt információt csak néhány milliszekundumig őrzi meg, ezután újra kell írni, azaz frissíteni. E hátrányuk ellenére a PC-be szinte kizárólag dinamikus *RAM*-okat építenek, mivel a frissítés egyszerűen megoldható, nem rontja lényegesen a rendszer teljesítményét, és hátrányaikat sok előnyös tulajdonsággal ellensúlyozzák. Az egy tokba foglalható kapacitásuk legalább négyszerese a statikus *RAM*-okénak, lényegesen olcsóbbak, kevesebbet fogyasztanak. (A memóriakapacitást bájtban mérik; egy adott számítógépben ennek növelését a CPU címzési tartománya korlátozza.)

menü: A képernyőre felírt lista, amelyből választani lehet. A kijelölés hatására vagy elindul a kiválasztott program, vagy átadásra kerül az adatállomány neve, vagy *betűtípusváltás* következik be, vagy megváltozik az üzemmód, stb. A menünek számos megjelenési formája van: a kép alsó, illetve felső sorába beugró vagy állandóan ott tartózkodó lista, a felső sorról legördülő (roll-down) lista, a képernyőn egy *ablakban* látható felsorolás, stb. A kiválasztás úgy történik, hogy bizonyos gombokkal, esetleg *egérrel* ráállunk a lista egy elemére, és egy újabb gombnyomással ki-

választjuk. Az esetek egy részében a listán be van jelölve az a nyomógomb, amellyel egyetlen gombnyomással is választhatunk.

merevlemez: Lásd harddisk.

merevlemez-cache: Lásd cache.

message: Lásd üzenet.

mesterséges intelligencia, MI (AI = artificial intelligence): A számítástudománynak az az ága, amely az emberi intelligencia olyan vonásainak tanulmányozásával és utánzásával foglalkozik, mint a felismerés, következtetés, tanulás. Bizonyos feladatokat, amelyek egykor nagyon bonyolultnak látszottak, mára elég hatékonyan sikerült megoldani (például sakk, objektumok felismerése rossz minőségű fényképeken). Másoknak, amelyek egyszerűnek látszottak, máig sem született kielégítő megoldása (például: beszélt szöveg felismerése, *természetes nyelvek* közötti fordítás). Mindazonáltal napjainkban a MI-kutatók és -alkalmazások már átszövik a tudományos és technikai szakterületek fejlődését, és a mindennapi életben is egyre több vonatkozásban találkozunk az eredményekkel. ➤ szakértő rendszerek.

mesterséges nyelv: Minden olyan jelrendszer, amely nem az emberi beszéd pontos leírására szolgál. Ilyenek a programozási nyelvek, a matematikai jelrendszerek, a kotta, egy rendszer *ikonjai* stb. Formális nyelv ezek közül az, amelynek matematikailag leírt nyelvtana van, amely megmondja, hogy melyik jelsorozat helyes és melyik helytelen. Az emberi nyelvet *természetes nyelvnek* nevezzük.

metaassembler: Olyan *assembler*, amely több géptípus *gépikódjára* képes fordítani.

METAFONT: Lásd TEX.

metanyelv: Lásd szintaxis.

MFLOPS (Million Floating Point Operation Per Second): Milliós lebegőpontos művelet másodpercenként. A lebegőpontos műveleti egység (*FPU*) teljesítményét jelző szám, az egy másodperc alatt végrehajtott lebegőpontos műveletek száma. Értékeléséhez meg kell adni a működtető *órajel-frekvenciát* is.

MFM: Módosított frekvenciamoduláció. Az *FM* adatkódolási módszer továbbfejlesztett változata, a felírandó adatoktól függően az óraimpulzusok egy részét nem rögzíti, így nagyobb adatsűrűség és tároló kapacitás érhető el.

MGP: Lásd Hercules.

MHz (Megahertz): Milliós hertz. A számítógépben a CPU *órajelének* frekvenciáját szokás MHz-ben megadni. A szabványos órajelek XT-nél: 5, 8, 10 és 12 MHz, AT-nál: 6, 8, 12, 16, 20, 25, 33, 40 és 50 MHz. Ezek az értékek csak a processzor sebességét jellemzik, nem az egész számítógépét. A busz órajele AT-nál általában 6 vagy 8 MHz.

MI: Lásd mesterséges intelligencia.

Micro Channel: Lásd mikrocsatorna.

microISIS: Lásd ISIS.

Microsoft: Szoftverház, termékeiben neve rövidítve MS. Lásd MS.

MIDI (Music Instrumentum Digital Interface): Szabvány a *soros interfészen* keresztül csatlakozó zenei eszközökre. (Hardver és szoftver.) E szabványnak megfelelő számos eszköz, elsősorban szintetizátor van a piacon, amelyeket főleg zeneszerzésre, zene előállítására szoktak használni.

mikrocsatorna (*Micro Channel*): Az IBM az 1987-ben kihozott *PS/2* sorozatú gépeiben kezdte alkalmazni ezt a 32 bites buszrendszert. Az IBM védett szabadalma, más gyártók csak jogdíj fizetése mellett használhatják. Nem kompatibilis a korábbi, az AT-kben használt *ISA* busszal, a régebbi *adapterek* így

nem csatlakoztathatók rá. Maximális adatátviteli sebessége 20 Mbájt másodpercenként. Nem vált sikeressé; bár néhány másodgyártó megvette a jogot, az IBM gépein kívül elenyészően kevés gépben van ilyen. A gépek többségében a másodgyártók megtartották az *ISA* buszt, illetve kifejlesztették az *EISA*-t. A mikrocsatornához illeszkedő adapterkártyák drágák, és választékuk szűkös.

MIPS (Million Instructions Per Second): Milliós utasítás másodpercenként. A CPU teljesítményét jelző mérőszám, az egy másodperc alatt végrehajtott utasítások száma. Mivel egy processzor típus általában többféle *órajel-frekvenciára* is készül, az értékeléshez azt is meg kell adni, hogy milyen órajel mellett nyújtja a megadott teljesítményt. ► MHz.

Mirror: Lásd PC Tools.

MIS: (Management Information System) Számítógépes információs rendszer, amely szervezési és vezetési információkat tart nyilván, kimutatásokat készít, döntési alternatívákat számítki.

mnemonik: Emlékeztető szó vagy rövidítés. Lásd assembler.

modem (modulátor-demodulátor): Távoli számítógépek közötti adatátvitelre — más lehetőség hiányában — gyakran használják a telefonhálózatot. Ez azonban *analóg* beszédtovábbításra készült, *digitális jelek* közvetlen küldésére alkalmatlan. A *bináris* bitsorozatot a modem ülteti át egy analóg jel frekvencia- vagy fázisváltozásává — így már továbbítható a telefonhálózatban. Az elérhető sebesség általában alacsony, 2400 *baud*. A túloldalban szintén egy modem csatlakozik a telefonvonalra, ez alakítja vissza bináris jelsorozattá a kapott adatokat, és azt továbbítja a számítógép felé. A modem a *soros porton* át

csatlakozik a számítógéphez. Az átvitel lehet csak egyirányú (*szimplex*), átkapcsolhatóan kétirányú (*fél duplex*) és kétirányú (*duplex*).

modul: Lásd moduláris programfelépítés.

Modula: Noha a *Pascal* nyelv kapcsán Wirth professzor kifejezetten törekedett arra, hogy a modularitást elkerülje, következő nyelve, a Modula(-1) két alapgondolatra épített: a *moduláris programfelépítésre* és a *multitasking* feladatok megoldására. Ez a nyelv egy sor fontos dolgot kihagyott a *Pascal* nyelvből, bevette viszont a moduláris programozással kapcsolatos legújabb ötleteket és egy csomó *multiprogramozásra* szolgáló eszközt. A nyelv nem igazán vált be. 1979-ben a Xerox cég Mesa nyelvnek tanulmányozása után Wirth professzor megcsinálta a Modula-2 nyelvet. Ebben visszatett a nyelvbe mindent, ami a *Pascal*-ban benne volt, kivette viszont a párhuzamosságot kezelő utasítások nagy részét, és a kezelést az ún. *coroutinokra* alapozta — abból a megfontolásból, hogy egy gépben úgyis csak egy központi processzor van (ez manapság nem feltétlenül igaz), és a legfontosabb feladat a folyamatok közötti gyors váltás.

Coroutin: egy olyan rutin, amely meghívás után úgy tudja visszaadni a vezérlést, hogy nem fejezi be a működését, hanem a legközelebbi meghíváskor ott folytatja, ahol korábban abbahagyta. (Társprogram és nem *alprogram*.) A modularitás tekintetében a nyelv fontos tulajdonsága a modulok *specifikációjának* és végrehajtható részének (*implementációs modul*) kettéválasztása helyileg és fordítási időben.

Svájcban a Modula-2-vel kapcsolatosan megépítették a Lilith kísérleti *munkaállomást*. Ennek nemcsak az az érde-

kessége, hogy minden szoftverje Modula-2-ben íródott, hanem a szokástól eltérően a gép hardverjét tervezték a szoftverhez, azaz úgy, hogy a lehető legjobban kiszolgálja a Modula-2 nyelvet. ► *Pascal*, moduláris programfelépítés, multitasking.

moduláris programfelépítés: A modern programszervezés egyik alapelve. Meg kell állapítanunk, hogy a programban milyen fő- és alfeladatok vannak. Ezeknek a feladatoknak a megoldására össze kell állítani egy-egy eszközkészletet.

Egy *modul* tartalmazza azokat az *alprogramokat* és adatokat, amelyek egy feladat megoldásához szükségesek. A modulok egymástól többé-kevésbé függetlenül lefordíthatók, kipróbálhatók, módosíthatók, más programokban megint felhasználhatók. A modern nyelvek lehetővé teszik a moduláris programozást, és gazdag választékot kínálnak előre elkészített igen hatékony modulokból.

Egyes rendszerek lehetővé teszik, hogy különböző nyelven írt programokat kapcsoljunk össze egyetlen programmá. (Például *Turbo C + Turbo Pascal + Assembler*.) A modulok külső hivatkozással (*external*) kapcsolódnak más modulokhoz, míg azok olyan a nevekre hivatkozhatnak, amelyek a modul látható részében (*visible part*) vannak megadva. Ez a rész egyes nyelvekben egészen elkülönül és *specifikációnak* hívják. Ez a rész *külön fordítható*, és más modulokból azok szintaktikus helyességének ellenőrzésére használható, még mielőtt a modul ténylegesen elkészült volna.

motherboard: Lásd alaplap.

mouse: Lásd egér.

MS: A Microsoft cég termékeinek szokásos márkajele.

MS-DOS, PC-DOS: 1980-ban az IBM cég némi hezitálás után a Microsoft céget választotta a Digital Research-csel szemben, hogy az új 16 bites gépek operációs rendszerének elkészítésével megbízza. A Microsoft Tim Patterson 86-DOS-a alapján kezdte kifejleszteni az MS-DOS 1.0-ást (amelyet az IBM-nél PC-DOS-nak neveznek). A PC-XT-hez már a DOS 2.0-val jöttek ki. Ez már fix diszket is tud kezelni, és a könyvtára hierarchikus (*UNIX*-szerű). A PC-AT-re hozták ki a 3.0-ás változatot, utána többször javították; a 3.3-as változat végül is igen stabilnak bizonyult. A 7 évvel később kibocsátott 4.0-ás úgy jellemezhető: „nagy, drága, új verzió, amit senki sem használ”. 1991 közepén jelent meg az 5.0-ás verzió. Ez számos olyan programot beépített a DOS-ba, amelyek korábban ráépültek a DOS-ra. Ilyenek: biztonságosabb diszkformázás (*undelete, unformat*), a memória felső részének kezelése (ide kerül a DOS jó része és a tárrezisens programok), nagyobb *partíciók*, új *BASIC*, teljes képernyős szövegszerkesztő, *menüvezérelt shell*, *task swapping* (több aktív program van, és váltogatni lehet, hogy melyik fusson) stb.

Vannak olyan programok, amelyek csak valamelyik verziótól fölfelé futhatnak. Az 5.0-áshoz bizonyos programokat át kell konvertálni.

A DOS működéséhez az szükséges, hogy az indító (*bootable*) diszkre három fájl fel legyen írva: az *IBMBIO.COM* (*IO.SYS*), az *IBMDOS.COM* (*MSDOS.SYS*) és a *COMMAND.COM*. Az *IBMBIO* a perifériakezelő programokat (*device drivers*) tartalmazza. Az *IBMDOS* valósítja meg a DOS-funkciókat. A *COMMAND.COM* a *parancsértelmező*; végrehajtja a belső parancsokat, és átadja a vezérlést a programok-

nak. Az operációs rendszer normális működése: a program hívja a DOS-t, a DOS hívja a perifériakezelőt, az hívja a *ROM BIOS*-t, a *ROM BIOS* a hardvert.

MS LAN Manager: A Microsoft cég által kifejlesztett helyi hálózati technológia. DOS, *OS/2* és *UNIX* alatt futó gépeket köthet hálózatba. A *szerver* gép gondoskodik a közös erőforrások használatáról. Osztott alkalmazások futtatását teszi lehetővé *client-server* alapon: a központi gépen (*server*) fut a program magja, ami rendszerint egy *multitasking* program. A *szerver* a hálózaton keresztül tart kapcsolatot a többi gép (*client*) programjaival, amelyek a felhasználó számára magas szintű interfészt nyújtanak a központ erőforrásaihoz.

MS Word: Microsoft cég *szöveg/kiadványszerkesztője*. Menüvezérelt. A Word Document lehetővé teszi, hogy ugyanannak az iratnak a különböző részein többen dolgozzanak; *annotációs* lehetőség. Stíluslapok. A nyomtatandó anyag távlati képének megtekintése (*page preview*). Az MS Word 5.5-ben használható a korábbi változatok menürendszere és egy új lehúzó (*Windows*-szerű) menürendszer.

MS Works: A Microsoft cég integrált szoftvere, azaz: adatbázis-kezelő, *táblázatkezelő*, *szöveg/kiadványszerkesztő*, *üzleti grafika*, óra, határidőnapló, *menüvezérelt shell* felhasználói programok hívására, fájlműveletek, hálózati kapcsolat, fájlátvitel és *terminálemuláció* a szolgáltatások. A táblázatok és az adatbázisfájlok 256 oszlopból, illetve mezőből és 4096 sorból/rekordból állhatnak. Saját oktatóprogramja van, és segítséget ad 200 témában, szabványosított üzleti formanyomtatványokat lehet vele készíteni adatbázist használva. Számos nyomtatót ismer, és hálózatban is alkalmazható.

multiprocesszoros rendszer (több-processzoros rendszer): A klasszikus (Neumann-féle) számítógépmódban egyetlen processzor hajtja végre a memóriában tárolt programot. Több processzor felhasználásával nyilván növelhető a gép teljesítménye. Fontos, hogy multiprocesszoros rendszeren egyenrangú CPU-kkal felépített gépet értenek, nem pedig *segédprocesszorokkal* támogatott központi CPU-t. (Ilyen értelemben egy közönséges AT is többprocesszoros gép lenne, hiszen a programot végrehajtó 286-os vagy 368-os, vagy 486-os CPU mellett használható matematikai *koprocesszor*, külön processzor végzi a *billentyűzetkezelést*, esetleg a harddisk vezérlését is.) A valódi multiprocesszoros rendszerben a CPU-k egyenrangúak, és párhuzamosan hajtják végre a programot. Ezekben a rendszerekben a legfontosabb kérdés a CPU-k egymásközti hatékony kommunikációjának megoldása. Erre jelenleg két alapvető *architektúra* létezik, az *osztott memória* és a *közvetlen kommunikáció*. Az osztott memória esetén a CPU-k — noha lehet saját külön memóriájuk is — egy közös memórián át tartanak kapcsolatot egymással. A közvetlen kommunikációnál a CPU-k közvetlenül egymáshoz kapcsolódnak.

A közvetlen kapcsolódásra csak bizonyos processzorok képesek, ilyenek a *transzputerek*, amelyeket kifejezetten ilyen rendszerekben való felhasználásra fejlesztettek ki. Az osztott memóriájú rendszer egyszerűbben programozható, de teljesítménye csak egy bizonyos fókig növelhető újabb CPU-k rendszerbe kötésével. A közvetlen kommunikáció esetén nincs ilyen korlát, de programozásuk igen nehéz. Általában a multiprocesszoros rendszerek tömeges elterjedését magas árak mellett elsősorban a szű-

kös szoftvertámogatás akadályozza. (Nagyszámítógépeken az osztott memóriájú, többprocesszoros *architektúra* eléggé általános.) ► párhuzamos feldolgozás.

multiprogramozás, multitasking: A számítógép egy időben vagy látszólag egy időben több programot futtat. A modern számítógépben elég sok egymástól függetlenül működni tudó egység van, így a párhuzamosságra meglehetősen sok lehetőség adódik. Az egyetlen központi processzorral rendelkező gépen is lehetőség van arra, hogy amíg egy program például a felhasználó inputjára vagy a nyomtatóra vár, addig egy másik program fusson. Ha egy gépben több CPU van, vagy például vannak *transputer* kártyái, akkor természetes, hogy a gépen egyszerre több programot futtatunk, vagy a programot egymással párhuzamosan futtatható részekre bontjuk.

A több CPU-t a sok számítást és kevés I/O-műveletet végző programok felgyorsítására lehet felhasználni. Multi-programozásról akkor szoktunk beszélni, ha az operációs rendszer szintjén több programot tudunk egy időben elindítani. Régebbi PC-s operációs rendszerekben (CP/M, MS-DOS) erre igen kevés lehetőség volt. Az újabb operációs rendszerek és *shellek* (például a *Windows*) erre is lehetőséget nyújtanak, itt már multi-programozott környezetről beszélhetünk.

A nagyobb gépeken ez a lehetőség már korábban is adott volt, de elsősorban arra használták, hogy a géphez önálló *terminálokkal* kapcsolódó felhasználók között osztották fel a gép kapacitását. (*Multi-user system* = többfelhasználós rendszerek.) A *Windows ablak-rendszere* és hasonló megoldások tették igazán értelmessé, hogy egy felhasználó több programot is futtathasson, és a

programok futását nyomon tudja követni. ►ütemezés.

multisync monitor: A képernyőre a képeket, a képen belül a sorokat különböző időközönként küldik ki a különböző berendezések. A multisync monitor mintavételezéssel felismeri a kép- és sorfrekvenciát, és alkalmazkodik hozzá.

munkaállomás: Lásd workstation.

mutató típus (pointer): Memóriacímek kezelésére szolgál. Elvileg a memóriacímek explicit manipulációjára nincs szükség a felhasználói programok megírásához, de azok a programok, amelyek mégis élnek ezzel, sokkal hatékonyabbak. Ugyanakkor a címek explicit manipulációja meglehetősen

veszélyes, a hibák könnyen vezethetnek a rendszer lerobbanásához (system crash), ezért a programozási nyelvek általában erősen megszorítják az ilyen lehetőségeket. Az értékadás és az egyenlőség vizsgálata minden nyelvben megengedett, más műveleteket csak néhány nyelv, például a C enged meg. A mai nyelvekben a futás közben lefoglalt területek kezelése a pointereken keresztül történik. Programozástechnikai megfontolásból a nyelvek bevezetik a NUL vagy NULL szóval jelölt pointert, amely elvileg nem mutat sehová, és üres pointernek nevezik, értéke az *implementciótól* függ.

művelet: Lásd szubrutin.

N

nagyon magas szintű nyelv: Más néven 4. generációs nyelv (4GL). Lásd magas szintű nyelv.

nagyszámítógép: Lásd mainframe.

NC: 1. Lásd : Norton Commander.

2. Numeric Controlled: Számjegyzérelt (berendezés). Olyan gép, többnyire szerszámgép, amelynek működését vezérlőkódokkal és numerikus értékekkel írják elő. Az adatokat legegyszerűbb esetben lyukszalagról, mágnesszalagról, mágneslemezről kapják (ahova számítógéppel írták fel). A *DNC* (Direct NC) esetén a kódok közvetlenül a számítógépből érkeznek. A *CNC* (Computerized NC) esetén a vezérlő számítógép a szerszámgépbe van építve. Ezekben az esetekben lehetőség van arra, hogy érzékelőkön keresztül a számítógép ellenőrizze a megmunkálást, megváltoztassa a kiadandó értékeket (*AC* = adaptív vezérlés). Például: ha túlmelegedést észlel, csökkenti a megmunkálás sebességét. NC berendezések vezérlőprogramjainak megírására szolgáló nyelvek: APT, ADAPT, EXAPT, MINIAPT, UNIAPT, PROMPT, SPLIT stb.

NEAT (New Enhanced AT): Az IBM AT-nek az *alaplapon* elhelyezkedő perifériáit és kiegészítő áramköreit — kb. 40-50 *IC-t* — először a Chips & Technologies cég integrálta 5 db *IC-készletbe* (*chip set*). Ezzel a készlettel készült alaplapok kapták a NEAT nevet. Azóta sok cég sokféle áramkörkészlettel jelent meg, akad olyan is, amelyik egyetlen *IC-be* integrálta egy AT teljes alaplapjának áramköreit — a processzort, a memóriát és a billentyűzetvezérlőt leszámítva. Az áramkörkészletek előnye a nagyobb sebesség és megbízhatóság,

hátrányuk, hogy nehezen javíthatók. Az árak csökkenésével ez egyre kevésbé probléma, gyakran egyszerűen kidobják a meghibásodott egységet. Az áramkörkészletek nagy részébe beépítik az *EMS*-kezelési lehetőséget, így az alaplapon lévő memóriát lehet *EMS-ként* használni.

negáció (negatio): (tagadás) Lásd logikai típus.

NetBIOS (Network BIOS): A helyi hálózatok szolgáltatásaihoz hozzáférést biztosító, *megszakításon* át hívható szoftverinterfész. 19 parancsa az *OSI modell* adatkapcsolati és viszony rétegével érintkezik. Eredetileg az IBM hálózati *adapterén* lévő *ROM* tartalmazta, innen az elnevezése. A legtöbb hálózati operációs rendszer kompatibilis a NetBIOS-szal, mert átvette a parancsait.

network: Lásd hálózat.

neurális hálózat (neural network): Agysejteket (neuronokat) modellező rendszer. A sejtek viszonylag egyszerű felépítésűek, valamilyen rendszer szerint össze vannak kötve, és üzeneteket (impulzusokat) küldenek egymásnak. A hálózat nincs előre beprogramozva, gyakorlással tanul. Vannak hardverrel és vannak szoftverrel megvalósított modellek.

NeXT: A *munkaállomások* kategóriájába tartozó számítógép, a NeXT Computer Inc. gyártmánya. Kategóriájában az egyik legnagyobb teljesítményű gép. Motorola CPU-val, több 10 Mb-ajt memóriával, speciálisan nagyfelbontású — 1120x832 képpont — *videóadapterrel*, beépített *Ethernet csatolóval* és a zene-, illetve *beszédszintetizálásra* egy *digitális jelprocesszorral* rendelkezik.

NG: Lásd Norton Guide.

NLQ (Near Letter Quality): Közel levélminőségű nyomtatás. A 9 és 24 tús

nyomtatók egyik üzemmódja és nyomtatásminősége.

NMI (Non Maskable Interrupt): Lásd megszakítás.

non-preemptív: Lásd ütemezés.

Norton Commander (3.0): Kéttablakos (itt az *ablakot* panelnak nevezik) *menüvezérelt shell*. Szolgáltatásai: fájlok, könyvtárak, kijelölt csoportok átvitele, másolása, törlése, átnevezése; könyvtárak összehasonlítása; fájlkeresés név szerint; a könyvtárfa bemutatása, könyvtár- és fájlválasztás a fából; felhasználói menük, azokon belül fő- és könyvtárfüggő lokális menük; programindítás a fájlkiterjesztés alapján. Meg tudja mutatni a fájlok tartalmát, nemcsak szövegfájlokét, hanem több adatbázist, *táblázatkezelőt*, sőt egy grafikus formátumot is ismer. Saját egyszerű *editor* van, kezeli az MCI mail *levelezőrendszert*, fájlátvitelt biztosít gépek között. Vegyíti a gombos vagy egeres menüvezérlést a billentyűkombinációkkal való vezérléssel és a parancsbeírással.

Norton Desktop for Windows: A Norton Utilities (NU 6.0) *Windows-os* változata. Minden programja (határidőnapló, *kalkulátor*, *backup*, *sleeper* stb.) kitűnő, csak a *Disk Doctor* gyengébb az eredeténél. A Norton Desktoppal lecserélhető a Windows Program Managere, minden funkcióját helyettesíti, és a felhasználható hely növekszik ilyenkor.

Norton Guide (NG): Szöveges adatbázisok *memóriarezidens* kezelője. E termékhez tartozó adatbázisok az operációs rendszert, a különféle nyelveket és

a programok használatát ismertetik. A felhasználó is hozhat létre saját adatbázist. Átmenet az adatbázis-kezelő, az intelligens *online help* és a *hypertext* között.

Norton Utilities (6.0): Elsősorban diszkkézelő, -karbantartó programok gyűjteménye, noha tartalmaz egy új *shell*t, *batch-programokhoz* egy jobb végrehajtót és hardverteszteket is. A *Disk Doctor II* logikailag és fizikailag ellenőrzi a lemezt, és ha hibát észlel, elkészíti a diagnózist, továbbá megjelöli a gyógy módot. Képes törölt fájlokat, lemezeket, sérült állományokat visszaállítani; a diszkkézelést felgyorsítani, rámutogatással kijelölt alkönyvtár-műveleteket elvégezni. Képes kihasználni az *extended* és *expanded* memóriát, kapcsolódik a *MS-DOS 5.0*-hoz és a *Windows 3.0*-hoz.

notebook: Könyv (szó szerint jegyzetömb) méretű, hordozható számítógép. Általában folyadékkristályos képernyőjű, akkumulátorral működő, az IBM PC-kel kompatibilis gép. Teljesítménye eléri az átlagos asztali (desktop) gép teljesítményét, a hordozható (*laptop*) géptől csak a méretében különbözik.

Novell NetWare: Helyi hálózati operációsrendszer-család. IBM PC és Apple *Macintosh* gépeken fut. Lehetővé teszi a *lokális hálózat* gépeinek kapcsolatát. Egy *szerver* gép irányításával biztosítja közös erőforrások (fájlok, diszkek, *printerek*) használatát.

Numeric Controlled: Lásd NC 2.

Ny

nyílt rendszer: Open System. A hagyományos értelmezés szerint a nyílt rendszer egy olyan rendszer, amelyik egy másik nyílt rendszerrel összekapcsolható. Újabban a következő követelményeket támasztják a nyílt rendszerrel szemben. Legyen:

- könnyen átvihető (*portábilis*),
- továbbfejleszthető,
- képes együttműködni másik rendszerrel (*interoperábilis*).

A nyílt rendszerek összekapcsolását írja le az *ISO/OSI* szabvány.

A nyílt rendszerek első példájának tekintik a *UNIX* rendszert. Az *OSFI* a fenti elveket maradéktalanul megvalósító *UNIX*-szerű operációs rendszer, melyhez az Open System Foundation tagjai kedvezményesen juthatnak.

nyílt szubrutin: Lásd szubrutin.

nyomkövetés (trace): Lásd debugging.

nyomtató: A számítógép adatkiviteli eszköze: szöveg, adat vagy ábra papírra nyomtatására szolgál. Csoportosítják őket működési elveik szerint, mivel ez meghatározza a nyomtatás minőségét is. A legegyszerűbb típus a *hőnyomtató (termoprinter)*, amely meleg tűkkel alakítja ki a hőérzékeny papíron a betűk formáját. Zajtalan, de minősége gyenge és lassú. PC-k mellett már nem használják.

A *mátrixnyomtató* fejében kisméretű tűk vannak, amelyeket elektromágnesek mozgatnak. A tű egy festékszalagra ütve hagy nyomot a papíron. A kinyomtatott betűk és ábrák ezekből a pontokból állnak. A pont mérete meghatározza a nyomtatás finomságát, minél több és vékonyabb tűs a fej, annál jobb a minőség.

A leggyakoribbak a 9, 18 vagy 24 tűs típusok. A nyomtatás minőségét *DPI*-ben adják meg, ennek megfelelően beszélnek *DRAFT*, *NLQ*, *LQ* és *SLQ* minőségről. A mátrixnyomtatóknál az írásminőség mellett fontos paraméter a sebesség, amelyet az egy másodperc alatt nyomtatható betűk számával — *CPS* (character per second) — adnak meg, valamint a zaj. A mátrixnyomtatót gyakran nevezik *sornyomtató*nak is, bár a valódi sornyomtatók általában nagyszámítógépek mellett használatosak. Jellemzőjük, hogy az írógépnél megszokott kalapácsos és festékszalagos módszerrel egy teljes sor betűt nyomtatják egyszerre, így igen nagy sebességet érnek el. Írásminőségük, leszámítva a nagyon drága típusokat, gyenge.

A *tintasugaras nyomtatók* működésekor egy finom szelep tintafoltokat hagy a papíron, ezekből áll össze az íráskép. A foltok kisebbek, mint a mátrixnyomtató tűi, így jobb minőséget adnak. Működése szinte zajtalan, sebessége kisebb, mint a mátrixnyomtatóké.

A *lézernyomtatóknál* egy lézerdióddával előállított lézersugár írja egy szelénbevonatú hengerre a betűket, amely festékpontot visz fel a papírra. Minősége betűk esetén közel nyomdai, felbontása általában 300x300 DPI, vagy jobb. A festéket hővel rögzítik a papíron, mint a fénymásolásnál. Általában műanyag fóliára is lehet vele nyomtatni. Szinte zajtalan, sebességét az egy perc alatt nyomtatható oldalak számával adják meg.

A nyomtatók vezérlése, kezelőprogramja sajnos nem egységes, egy adott típus használatához a programnak ismernie kell a nyomtatót. Ha a program által ismert nyomtatók között nem szerepel az adott típus, akkor nem tudunk mindent kihasználni, amire a nyomtató

képes. (A lézernyomtatóval esetleg egyáltalán nem is tudunk dolgozni.) Mátrixnyomtatóknál az IBM és az Epson típusú vezérlésűek a leggyakoribbak, a kisebb gyártók általában ezeket követik.

A lézernyomtatók kezelése teljesen eltér a mátrixnyomtatókétól; két fő típusuk a Hewlett-Packard cég printeréről elnevezett HP-kompatibilis és a *Post-*

Script. A HP-kompatibilis nyomtatóknál a betűket a nyomtatóban vagy a számítógépen tárolt bittérképből (*bit-map*) állítják elő. Ennek a módszernek a legnagyobb korlátja, hogy a betű mérete nem változtatható, előre megadott méretek vannak. A PostScript típusoknál a betűk tetszőlegesen méretezhetők. A PostScript nyomtatók ára 2-3-szorosa a HP-kompatibilis nyomtatókéknak.

O, Ó, Ö

Oberon: 1988-ban Wirth professzor — több más *magas szintű nyelv* alkotója — újabb programozási nyelvet hozott létre, az Oberont. Ez az *OOP* elveit tükrözi, az *Ö* értelmezésében. A nyelv a *Pascalon* (és a *Modulán*) alapul.

Néhány nyelvi konstrukciót ismét kihúzott, mint amire munkásságában már volt példa — például a *set típust* —, bekerültek viszont a típuskiterjesztés, továbbá az *öröklődés* bizonyos elemei és a szemétygyűjtés (*garbage collection*). Legfőbb előnye, hogy viszonylag egyszerű.

A *SMALLTALK*-hoz hasonlóan itt is összekeverednek a nyelv és a benne megírt rendszer elemei. ➤ Pascal, Modula, OOP.

object code: Lásd tárgykód.

Object Oriented Programming: Lásd OOP.

ObjectVision: *Windows* alá készülő alkalmazási programok írására szolgáló *objektumorientált* grafikus eszköz.

A programírás abból áll, hogy formanyomtatványokat tervezünk, és döntési fákkal leírjuk azok kitöltését. A mezőértékek és a döntési fák leírásához a *Quattro Pro* függvényeit és kifejezéseit használja. A kitöltés során a program adatokat vehet és adhat át *Paradox*, *BTRIVE*, *dBase*, *ASCII* adatbázisoknak, illetve dinamikus adatcserén (DDE-n) keresztül más *Windows*-programoknak. *ObjectVision* alatt a programok futtathatók és továbbfejleszthetők. Az elkészült programok a *ObjectVision Runtime* rendszerrel futtathatók.

Occam: *Transputerek* programozására szolgáló *alacsony szintű nyelv*.

OEM (Original Equipment Manufacturer): Valamilyen számítógép-részegységet — az eredeti gyártó licenc engedélye alapján — készítő cég jelzése, ami megkülönbözteti a hasonmás-készítőktől és a csak összeszereléssel foglalkozóktól.

offline: Lásd online.

offline kiíratás: Lásd online.

offset: Az összetett cím egyik része. Az összetett cím egy nagyobb adategység címéből és az azon belüli relatív címből áll. Az utóbbi az offset. Például a rekord címe (sorszám) a fájlban, és a mező címe (az offset) a rekord elejéhez képest.

Az IBM PC hardverje az utsításokban szereplő címeket bázis + offset alakban várja, ahol a bázis valamelyik bázisregiszter tartalmának 16-tal szorzott értéke.

oktális szám: Lásd bináris szám.

online: Bekapcsolva, rákapcsolva.

online help: Olyan funkciója/képessége a programnak, amikor a segítő információ (*help*) szolgáltatása — az interaktív működése közben — bármikor hívható.

online kapcsolat: A két berendezés össze van kapcsolva, úgy történik az adatok átvitele.

offline: Kikapcsolva, lekapcsolva.

offline kiíratás: A nyomtató nem futó programtól kapja a kiírandó adatokat. Például a program floppyra viszi, és egy másik gépen írják ki.

OOP, objektumorientált programozás (Object Oriented Programming): Objektumokon alapuló programozás.

Különböző filozófiákon alapuló programozási módszerek vannak (*funkcionális programozás, procedurális programozás, moduláris programozás*).

Az OOP viszont úgy képzei, hogy maga a program *objektumok* gyűjtemé-

nye; az objektumok egymástól és a külvilágtól, tehát a felhasználótól is *üzeneteket* kapnak, azt feldolgozzák — lehet, hogy ettől állapotuk megváltozik, és reakcióképpen ők is üzenetet küldenek.

Az objektumok zártak (*encapsulated*), azaz az objektumot felépítő eljárások — itteni névük: *tevékenység (action)* — és *lokális* adatok más objektumok számára nem hozzáférhetők. Az objektumok kizárólag üzenetátadással kommunikálnak egymással. Az üzenet nem más, mint egy kérés bizonyos tevékenység elvégzésére.

Az objektumok osztályokba sorolhatók: az egy *objektumosztályba* sorolt objektumoknak ugyanolyan adataik és tevékenységeik vannak. Az objektumosztályt finomítani lehet és így egy lezármazott osztályt kapunk belőle. A lezármazott osztály *örökli* a *lokális változókat* és a tevékenységeket; ilyenkor további adatokat és tevékenységeket lehet bevezetni, illetve az öröklött tevékenységet meg lehet változtatni.

Többféle objektum is kaphat ugyanolyan üzenetet. Az egyes objektum reakciója függ az objektum osztályától, valamint a pillanatnyi állapotától — ezt a lokális változók írják le.

Amikor egy objektumorientált programot futtatunk, lényegében sok kisméretű program fut egyidejűleg. Mind-egyik figyel a ún. *globális üzenetteret*: ha van neki szóló, akkor elvégzi a megfelelő tevékenységet, saját üzeneteit is ugyanide küldve. Ilyenformán az objektumorientált programnyelvek ideális eszköznek tűnnek *párhuzamos feldolgozó* rendszerek programozására.

A Xerox cégnél 1980-ra fejlesztették ki a *SMALLTALK*-ot, az első objektumorientált programnyelvet. Később meg-

jelent az *Object Pascal*, *Oberon*, *C++*, *Turbo Pascal 5.5/6.0*.

Az objektumorientáltság 3 jellemzője: a zárt objektumok, az öröklődés és az üzenetekkel való kommunikáció. Különösen az öröklés tekintetében vannak nagy különbségek az egyes nyelvek között.

opcionális: Nem kötelező. **Opció:** választható lehetőség.

open: Lásd fájlműveletek.

open system: Lásd nyílt rendszer.

operandus: Lásd szubrutin.

operátor: 1. Gépkezelő.

2. Műveleti jel. (Lásd *szubrutin*).

operációs rendszer (operating system, OS): Olyan rendszerszoftver, amely egyrészt fogadja a felhasználó parancsait, másrészt szolgáltatásaival lehetővé teszi azok végrehajtását.

Az operációs rendszer fő funkciói: kezeli a perifériákat, *multiprogramozott* esetben foglalkozik az időnek és a memóriának a feladatok közötti kiosztásával (ez a rész a *kernel*), fogadja a felhasználó parancsait és értelmezi azokat, majd elindítja a megfelelő tennivalókat (ez a rész a *shell*). ► MS-DOS, UNIX, OS/2.

optikai kábel: Kvarcüvegből készült szál, amelyben fényimpulzusok továbbítják az információt; fényforrásuk *LED dióda* vagy egyre inkább lézercióda.

Az ún. többmódusú szálban a fény a szál belső faláról visszaverődve halad. A 70-es évek végén kifejlesztett egymódusú szál átmérője a fény hullámhosszával azonos nagyságrendű, ezért nincsenek benne visszaverődések, kisebb veszteséggel továbbítja az impulzusokat. Erősítés nélkül több mint 100 km távolságon több száz Mbit/sec sebesség érhető el, a költségei azonban egyelőre magasak.

optikai tároló: Olyan tárolóeszköz, amelyben lézerfényt használnak az információ kiolvasásához.

A *CD-ROM* működési elve megegyezik a CD lemezjátszó működésével. Gyárilag írják, csak olvasható; illetve kapható *CD-ROM MASTERING* egység, amellyel üres lemezre írhatunk. Kapacitása általában 560 Mbájt.

be a lemezre a biteket és olvassa ki, a rögzített adatok nem változtathatóak. A felírás meglehetősen lassú: 22—150 msec/szektor. A diszkek átmérője különböző, kapacitásuk Gbájtban mérhető.

A többször írható mágneses-optikai lemezeknél a felírást mágnesezéssel, az adatok kiolvasását pedig lézerrel oldják meg. Ezeknek a sebessége is elmarad a harddiszkek sebességétől.

Van 5 1/4 inch-es változata; itt egy lemez 594 vagy 650 Mbájt kapacitású, a pozicionálás 70 msec, az írás pedig 90 msec időt igényel. Van 3 1/2 inches változata is; itt a kapacitás kisebb, de az elérés gyorsabb (kb. 10 msec).

optimum: A legkedvezőbb érték.

Optimumról csak akkor beszélhetünk, ha tudjuk, mi a célfüggvény, azaz milyen célt mekkora súllyal kell figyelembe venni, és ismerjük a korlátokat, amelyek között a megoldást keresni kell.

Az *optimalizálás* maga a folyamat, amikor keressük, közelítjük az optimumot.

Lokális optimumnak nevezünk egy megoldást, ha az optimumot csak bizonyos körben kerestük. ► programoptimalizálás.

Oracle: Relációs adatbázis-kezelő, az Oracle cég terméke, amelyet a PC-kig (bár kis PC-n nem elég hatékony).

Adatbázis-kezelő nyelve az *SQL*.

Az Oracle az adatbázis-kezelő programok fejlesztéséhez egy fejlett *CASE* technológiát is ad.

órajel (CLK): A digitális számítógépekben alkalmazott áramkörök tekintélyes része — így a processzorok, különböző vezérlők — ún. ütemezett áramkör. Ez azt jelenti, hogy a működéshez szükség van egy ütemező jelre, minden belső változás eszerint lépdel. Két jel között az áramkör tétlen, nem változtatja az állapotát. Így a működési sebesség és ezen át a teljesítmény végül is az áramkör órajelének frekvenciájától függ.

Természetesen fizikai korlátai vannak, hogy egy áramkör maximálisan mekkora órajellel működtethető; ez a tervezésén és a megvalósítás technológiáján múlik. ► MHz.

OS/2: A Microsoft cég operációs rendszere; a 80x86-os gépeken védett módban fut. *Virtuális memóriakezelést* és *multitaskingot* tesz lehetővé, felkészült a *lokális hálózatok* kezelésére.

A *Presentation Manager* egy grafikus felhasználói interfész (*shell*). Képes MS-DOS programokat futtatni. Végzetes hibája, hogy nem képes a *virtuális mód* előnyeit kihasználni.

1992 tavaszán jelent meg az OS/2 v2.0, amely 386-os, 486-os, 4 Mbájtos gépeken *installálható*. Ez már *Windows-os* programok futtatására is alkalmas (Windows nélkül).

OSI modell: Lásd ISO/OSI.

osztott, elosztott: Lásd shared.

output: Kivitel. A program *üzenetei* és eredménye, illetve az a folyamat, amikor ezeket beírja egy (vagy több) eredményfájlba, vagy kiküldi valamelyik perifériára. ► input/output.

overflow: Lásd túlcscordulás.

overlay: Átfedés. Olyan program-szervezés, amikor a nagyméretű programot egy közös gyökérből kiinduló ágakra bontjuk, és mindig az az ág kerül a memóriába, amelyik éppen fut. Moder-

nebb memóriaszervezés mellett nincs rá szükség.

overloaded: Lásd szubrutin.

öröklődés (inheritance): A programozási nyelvek azon tulajdonsága, hogy amikor egy adattípusból újabb típust definiálunk — azáltal, hogy új típusnevet

adunk neki —, az új típusra a régi típus műveletei és *attribútumai* érvényben maradnak (de újakat is adhatunk hozzá). Hasonló a helyzet a *objektumorientált* programozásban, amikor egy objektumosztályból új objektumosztályt származtatunk. ► OOP.

P

packed (pakolt): Adatábrázolás. Előfordul, hogy egy adat nem tölti ki teljesen azt a memóriarészt, amelyet a *fordítóprogram* a könnyű elérhetőség kedvéért a számára lefoglalt. (Például egy *logikai változó* egy bitet igényel, a programok egy bájtot foglalnak le számára, mert ez a legkisebb elérhető memóriaegység.) Különösen akkor zavaró ez a kapacitáspazarlás, ha változónként már eleve nagy a fel nem használt hely, és ebből a típusú adatból egy testesebb *tömbre* van szükségünk. Ilyenkor a program a memória-megtakarítás érdekében tömörítheti az adatokat, azaz például: ha egy húszelemű logikai vektorra van szükségünk, akkor ezt elhelyezhetjük húsz egymást követő biten is, ami kevesebb 3 bájtnál. Bár logikusnak látszik egy ilyen *reprezentáció*, de azzal jár, hogy ilyenkor egy-egy bit elővételéhez esetleg több utasítás és valószínűleg több idő szükséges — ami megfontolandó.

A 'packed' *kulcsszó* a fordítóprogram számára, hogy a tömör ábrázolást válassza. (Gyakran tömörítik össze az adatokat, mielőtt egy fájlba raknák őket.)

package: 1. Programcsomag. Több egymáshoz kapcsolódó program, amelyek egy nagyobb téma egyes részfeladatait oldják meg.

2. A modulnak megfelelő programrész az *Ada* nyelvben.

packet switching (csomagkapcsolás): Lásd távolsági hálózat.

page preview: A nyomtatandó lap szemrevételezése. Mivel készítés közben rendszerint a kinyomtatandó lapnak csak kicsiny része látható a képernyőn, a félkész nyomtatások és javítások elke-

rülése érdekében a képernyőn megjeleníthető — akár oldalpáronként is — a nyomtatandó lapok kicsinyített képe. Ezt lehet fokozatosan nagyítani is.

Paradox: A Borland cég adatbázisformátuma. Kezelője a *Paradox* vagy a *Microsoft C-hez. Ír, olvas és létrehoz fájlokat; kitöröl, beszúr és felülír rekordokat; indexfájlokat* készít, és támogatja a keresést; segíti a fájlok és rekordok lezárását automatikusan és kézi vezérléssel; kezeli a hibákat. Egyszerre 64 fájl lehet megnyitva.

A Paradox Engine-re épül a *Sidekick*, a *Quattro pro*, a *Paradox 3.5* adatbáziskezelő. De ráépülhet mindenféle alkalmazási program is. *Windows-környezetben* a Paradox Engine *DLL* formában van megadva. Azaz olyan rutinok gyűjteménye, amelyek futás közben kapcsolódnak a programhoz és a párhuzamosan futó programok a közös példányon osztzkodnak. Erre írhatók *Windows-alkalmazások Turbo Pascal-ban, C++-ban,* illetve az *ObjectVisionön* keresztül.

paraméter: 1. A számológép által végrehajtott tevékenységeket kisebb-nagyobb zárt egységekbe lehet sorolni, például program, parancs, utasítás stb. Ritka az olyan tevékenység, amelyet mindig ugyanúgy hajtunk végre, ugyanazzal az eredménnyel. A tevékenység végrehajtásához rendszerint bemenő adatokra (input paraméterekre) van szükség, és eredményül kimenő adatokat (output paramétereket) kapunk. A bemenő paraméterek esetén a paraméter értékére van szükség, így megadható akár maga az érték, akár az a hely, ahonnan az értéket venni kell, viszont a kimenő paramétereknél mindig a címet kell megadni, ahová az érték kerül.

2. Általánosabb értelemben paraméternek nevezünk minden mennyiséget, amely egy rendszert, egy hardvert vagy

egy programot, objektumot stb. jellemez. ► attribútum.

parancs (command): Lásd utasítás.

parancsértelmező: Lásd shell.

parancsnyelv: Lásd utasítás.

parkolás, parkolósáv: Lásd harddiszk.

parser: Lásd szintaxis.

particionálás: Felosztás. Általában véve valaminek részekre való osztása. Speciálisan a PC-knél a nagyobb kapacitású diszkek felosztása.

Egy *partíció* általában valamilyen tárterület; speciálisan PC-knél a *harddiszk* egy része. Egy partíciót az operációs rendszer teljesen önállóan kezel, mintha egy külön *meghajtója* volna. Minden partíciónak saját *boot sectora*, *FAT* táblája és főkönyvtára van. Egy harddiszk max. négy partícióra osztható. Az egyes partíciókra vonatkozó adatok — méret, elhelyezkedés — a *partíciós táblában* vannak, amelyet a harddiszk fizikailag legelső *szektora*, az ún. *master boot* szektor tárol. Kijelölhető az ún. aktív partíció, amelyben egy operációs rendszer helyezkedik el, és ahonnan azt a gép bekapcsolása után betölti.

(A partíciókra osztásra az MS-DOS 3.31 előtti változatainál sok esetben szükség volt, mert az operációs rendszer csak max. 32 Mbájtos partíciót tudott kezelni. Az ennél nagyobb harddiszkeket szükségszerűen több részre kellett osztani.) A particionálás lehetővé teszi, hogy egy harddiszken több operációs rendszert is elhelyezhessünk. (Például: *DOS+XENIX*).

partíció: Lásd particionálás.

partíciótábla, partíciós tábla: Lásd particionálás.

Pascal: N. Wirth, a Zürichi Műszaki Egyetem (ETH) professzora szakított az *ALGOL68* nyelvet tervező bizottsággal, és 1968-ban egy új nyelvet definiált,

amelyet a nagy francia matematikusról nevezett el. A nyelv első *implementációja* 1970-re, pontos definíciója 1974-re datálható. Ugyancsak a zürichi egyetemen készült el a *Pascal-P*, amely a nyelv más gépekre való átvihetőségét (*portabilitását*) célozta meg. Ez egy *fordítóprogram*, de nem egy konkrét *gépkódra* fordított, hanem a *pszeudokódra*: egy fiktív gép kódjára. A fordítóprogram eredetileg Pascalban volt írva, de rögtön átírták pszeudokódba is. Ezek után, ha írtak egy fordítóprogramot, amely a pszeudokódról — ami végül is egy *assembly*-szerű nyelv — egy konkrét gépre fordított, akkor ezen a gépen minden lefordított Pascal programot végre lehetett hajtani, így magát a Pascal fordítót is. Ha a pszeudofordító optimalizálásokat végzett, akkor a programok hatékonyabbak lettek, maga a Pascal fordítóprogram is.

A Pascal nyelv számos programozási koncepciót tisztázott: szétválasztotta az adattípus-*specifikációt* és az *adatdeklarációt*; lehetővé tette a típusok elnevezését; előírta a névszerinti, *erős típusellenőrzést*. Ezt azóta is minden modern nyelv így csinálja. Ugyanakkor számos primitív hibája volt: alig törődött a felhasználói interfésszel és az operációs rendszerhez való interfésszel, és igen keményen a *moduláris programozás* ellen foglalt állást. Ezeket a hibákat a későbbi verziók megszüntették, de ezáltal különböző dialektusok keletkeztek. Wirth professzor azonban ekkor már újabb nyelveit, a Modulát, majd az *Oberont* ajánlotta programozásra. E nyelvből számos új nyelv is született, ilyenek: a Konkurens Pascal, Mesa, Object Pascal, Turbo Pascal 6.0. ► Modula, Oberon, adattípus, OOP.

password (jelszó): Lásd kulcsszó.

path: (útvonal) Lásd könyvtár.

PathMinder: Elsősorban a fájlműveletek megkönnyítésére szolgáló menüvezérelt *parancsértelmező*. Lehetőséget teremt a felhasználó kedvenc programjainak kényelmes indítására. Beépített egyszerű *szövegszerkesztőt* tartalmaz. Továbbfejlesztett változata sokkal nagyobb, de nem sokkal jobb.

párbeszédes üzemmód, interaktív üzemmód: Olyan üzemmód, amikor a felhasználó folyamatosan kapcsolatot tart a programjaival. A képernyőn figyeli a program által küldött *üzeneteket*, és a *klaviatúráról* vezérli a program lefutását.

párhuzamos adatátvitel (parallel transmission): Az adatok továbbításának a számítógépek és a külső perifériák között két módja lehet: a soros és a párhuzamos. A párhuzamos átvitelnél a továbbítandó bájt minden bitje külön vezetéken halad egyidejűleg. Az adatvesztés megakadályozására minimálisan két vezérlőjel szükséges. Az egyiket a küldő adja ki (a szokásos neve STROBE), jelezve, hogy érvényes adat van a vezetéken, a másikkal (ezt általában ACKNOWLEDGE-nek vagy ACK-nak hívják) a fogadó nyugtázza, hogy átvette az adatot, a vezetéket szabaddá lehet tenni a következő adat számára. Ez a módszer az ún. *handshake (kézfogás)* — így említi a szakirodalom.

Ha az átvitelre mindkét irányban sor kerülhet, természetesen akkor is mindig csak egy adat van egyszerre a vezetéken, de átkapcsolhatják az adatirányt. A vezetékek nagy száma miatt az átvitel érzékeny a külső zavarokra, általában csak néhány méteres távolság áthidalására megbízható.

párhuzamos feldolgozás: Olyan működési mód, amikor egy gépben párhuzamosan több munka folyik. Csak több processzor esetén lehetséges. (Nem ke-

verendő össze a *multiprogramozással* és a *multitaskinggal*, ahol csak látszólagos a párhuzamosság, hiszen egyetlen processzor osztja fel idejét több feladat között.) Egyszerűbb esete, amikor a processzorokon más-más munka folyik. Az igazán nehéz feladat egy munkát úgy megszervezni, hogy egyes részei futhassanak párhuzamosan. ► multiprocesszoros rendszer.

párhuzamos port: Párhuzamos adatátvitelre alkalmas csatlakozás. Az IBM PC típusú gépekben kialakított párhuzamos portok a 8 adatbit mellett 5 állapot- és 5 vezérlőbit kezelésére alkalmasak. Leggyakrabban a *printer* illesztésére használják, ezért nevezik *printer portnak* vagy *Centronics portnak* is.

PC Tools (PCT): Keretprogram és szerszámkészlet. 7.0 verziója képes együttműködni a MS-DOS 5.0-val és a *MS-Windows 3.0-val*. Tartalmaz egy *menüvezérelt parancsértelmezőt*, különböző *kalkulátorokat*, editort helyesírás-ellenőrzővel, telefonhívót, faxot, *e-mailt*. Egyik programja kapcsolatot tud teremteni másik géppel helyi hálózaton, *modemmel* vagy anélkül (*soros vonalon*). A háttérben egy másik munka mellett mentést tud készíteni a teljes rendszerről vagy nagy részéről. Képes bizonytalanodó diszkeket és hibás fájlokat javítani, bár ebben a tekintetben nem mérhető össze a hasonló *Norton Utilities* programmal. *Mirror* funkciója biztonsági mentést csinál a harddisk táblázatairól, és Recovery Disk is kérhető a rendszer kulcsinformációról. Vírusfelismerő, -eltávolító és fertőzésmegelőző szolgáltatásai is vannak.

PC-DOS: Lásd MS-DOS.

PCT: Lásd PC Tools.

peek, poke: A személyi számítógépek őskorának két kedvenc utasítása; a peekkel egy memóriarekesz tartalmát lehet

kiolvasni, a poke-kal pedig írni. Trükkökre és a program felgyorsítására használták. Mindkettő feltételezi a memória-rekeszek kiosztásának pontos ismeretét. Csak primitív rendszerekben alkalmazhatók. A komolyabb operációs rendszerek közönséges felhasználónak egyszerűen nem engedik, hogy az operációs rendszerhez, vagy bármely másik programhoz nyúljon, még a saját programjának az utasításkódjaihoz sem férhet hozzá.

periféria: A számítógépek Neumann-féle modelljében a processzor számára a memórián kívül minden más eszköz periféria. A perifériák a külvilággal való kapcsolattartás eszközei, ezeken át kap és küld adatokat a CPU. A gyakorlatban a perifériákat *háttértárakra* és a továbbított adat irányának megfelelően kimeneti és bemeneti perifériákra osztják. Tipikus kimeneti periféria például a *képernyő* vagy a *nyomtató*, bemeneti periféria a *billentyűzet*.

PGA (Pin-Grid Array): Processzorok vagy más nagy bonyolultságú *integrált áramkörök* tokozási módja. A kivezetések nem a tok oldalán helyezkednek el, mint a *DIP* esetén, hanem a teljes felületén, rácsot alkotva.

PIO (Parallel Input-Output) : A *párhuzamos adatátvitelt* vezérlő *integrált áramkör* általános neve. Az IBM PC típusú gépekben az Intel 8255-öset vagy más gyártó ezzel kompatibilis áramkörét használják.

pipe, pipe-line: Szerelőszalag. Lásd utasítás.

pixel (képpont): Lásd grafika, képernyő.

PLCC: Lásd LCC.

plotter (rajzgép): A számítógéppel készített ábrák, műszaki tervek, grafikák stb. megrajzolására alkalmas kimeneti periféria. Általában papírra vagy esetleg

fóliára rajzol tollal. Egyes típusoknál a papír van rögzítve, és a toll mozog mindkét irányban, más esetekben, főként a nagyméretű — akár A0-ás — felületre rajzolóknál az egyik irányban a toll, a másikban a papír mozog (egy hengeren). A tollak váltásával többszínű rajzok is készíthetők.

pointer: Lásd mutató típus.

pointermanipuláció: Lásd mutató típus.

polimorfizmus: Lásd szubrutin.

pop: Lásd stack.

port: A számítógép más eszközökkel való összekötését lehetővé tévő csatlakozó. Az adatok továbbítási módja szerint lehet *párhuzamos port* — amelyet gyakran *Centronics* portnak is neveznek — vagy *soros port*. A párhuzamos esetben egy bájt bitjei egyidejűleg, míg a sorosnál egymást követően továbbítódnak. Párhuzamos port való a nyomtatók többségéhez, míg a soros portot az *egérhez* használják, vagy egy másik számítógéppel való összekötésre.

portabilitás: Átvihetőség. Ugyanarra a programra vagy adatfájltra több különböző gépen, különböző operációs rendszerek alatt lehet szükségünk. A két gépet *kompatibilisnek* nevezzük (az adott program szempontjából), ha a program és az adatok változtatás nélkül átvihetők. Ez ritka szerencse. Egy programot *portábilisnek* nevezünk, ha az átvitel kevés munkával megoldható. A portabilitást úgy érhetjük el, ha ragaszkodunk a szabványosított megoldásokhoz, és minél kevesebb gép- és operációsrendszer-függő utasítást teszünk bele, és ezeket a speciális részeket a többtől jól elválasztott modulokba helyezzük el.

postprocesszor: Utófeldolgozó program, amelyet azért hajtunk végre, hogy egy szoftver eredménye szebb, jobban olvasható, jobban használható legyen.

Így nevezik a rajzoló- és a tervezőprogramok eredményfájljait elkészítő programokat is.

PostScript: *Lézerprinterek* és nyomdai levilágítók vezérléséhez kifejlesztett önálló nyelv, funkciójából adódóan lapleíró nyelvnek is nevezik. Jogilag védett, tulajdonosa az Adobe System Inc. A PostScript nyelven a képszerű információt nem pontokra bontva tárolják, hanem vonalakra, görbékre bontva. A betűket általában Bézier-görbékkel írják le. Ennek a módszernek nagy előnye, hogy a tárolás lényegesen kisebb terület igényel akár a memóriában, akár a háttértárolón. A nagyítása vagy a kicsinyítése nem jár az élesség csökkenésével, illetve a kép torzulásával. A nyomtató vagy a levilágító valójában pontokból állítja elő a képet, így a megjelenítés fázisában a pontok helyét a görbékkel kell kiszámítani. Ez komoly számítási kapacitást igényel. A számítást elvégezheti a számítógép a nyomtatás előtt, de a nyomtatóba beépített hardver is. Ezért lényegesen magasabbak a PostScript nyelvet értő berendezések árai a *bit-térképesekénél*.

A PostScriptet, bár eredetileg nyomdai célokra szánták, egyre gyakrabban alkalmazzák a képernyőn való megjelenítéshez is. Az Adobe az eredeti változatot továbbfejlesztette, és az újabb verzió — a Level-2 — felülről kompatibilis a korábbival.

A nyelv több *betűkészletet* tartalmaz, amelyek mérete tetszőlegesen változtatható. *EPS (Encapsulated PS)* a nevük azoknak az előre elkészített kis ábráknak, grafikáknak stb., amelyeket be lehet komponálni a kiadványba. ► kiadványszerkesztés.

pozicionáló gömb (track ball): Funkciójában az *egérhez* hasonló eszköz. Működési elve is azonos, csak míg az

egérben lévő golyó attól forog, hogy az egeret az asztalon mozgatjuk, itt közvetlenül kell egy nagyobb golyót a foglalatában forgatni az ujjunkkal. Előnye a kis helyigény, a *billentyűzetbe* is beépíthető.

preemptív: Lásd ütemezés.

prefix operátor: Lásd szubrutin.

preprocesszor: Előfeldolgozó program. Olyan program, amelyet bizonyos feltételek mellett egy másik program előtt végre kell hajtani. Egyik tipikus esete, hogy amikor egy nyelvet *szöveg-makrókkal* kibővítünk, ezeket a bővítéseket a fordítás előtt fel kell oldani. Ilyen értelemben használják például a *C nyelv fordítóprogramjának* a legelső menetére. Másik gyakori esete, hogy feloldozás előtt az adatokat valamilyen sorrendbe kell állítani, bizonyos adatokat ki kell szűrni.

Presentation Manager: Lásd OS/2.

printer: Lásd nyomtató.

printer port: Lásd párhuzamos port.

printer puffer: Lásd puffer.

prioritás (elsőbbség): A számítások során gyakran alakul ki olyan helyzet, amikor két vagy több tennivaló közül választani kell. A gép ilyenkor a tennivalókhöz egy-egy számértéket rendel, és közülük azt választja, amelyiknél ez az érték a legnagyobb. A prioritás statikus, ha ez az érték időben nem változik, és dinamikus, ha az előzőleg történtektől függ.

procedurális nyelv, procedurális programozás: Az eljárásokon alapuló — procedurális — nyelvekben a program utasítások sorozata. A logikailag összetartozó, ismétlődő utasítássorozatoknak nevet adunk (ezek lesznek az eljárások), a program elejére a deklarációs részbe visszük őket, és felhasználásukkor csak a nevüket írjuk le.

► szubrutin, moduláris programozás, OOP, funkcionális programozás.

procedúra (procedure, eljárás): Lásd szubrutin.

Processzor: Lásd CPU.

production system: „Ha a feltétel teljesül, tedd...” alakú szabályok halmaza. Azt mondjuk, hogy a szabály „tüzel”, ha feltétele teljesül és végrehajtjuk. Egy rendszerben általában egy adott pillanatban több szabály is tüzelőképes lehet, és a stratégián múlik, hogy melyiket hajtjuk végre. A *mesterséges intelligenciában* alkalmazott feladatmegoldó eszköz.

program: Előírás arra, hogy a számítógép milyen utasítássorozatot hajtson végre. A programot valamilyen *programozási nyelven* lehet megadni. A leírt program általában nemcsak egymást követő végrehajtandó utasításokból áll, és az utasításokat nem a végrehajtás sorrendjében tartalmazza, hanem funkcionális egységekből épül fel. Ezek definiálják a program egyes részfeladatait, a hozzájuk tartozó adatok belső szerkezetét, az egyes adatláncokat stb. A program globális szerkezete pedig összekapcsolja ezeket a részeket.

programcsomag: Lásd package.

programeditor: Lásd szövegszerkesztő.

programmegszakítás (break): Lásd debugging.

programoptimalizálás: Olyan módszerek alkalmazása, amelyek eredményeképpen a program méretét akarjuk csökkenteni és a program futási sebességét pedig növelni. Nagyrészt a *fordítóprogram* feladata. A két cél egyébként általában ellentmond egymásnak.

A programok *lokális optimalizálásán* azt értjük, hogy a fordítóprogram úgy fordítja le a programot, ahogy a programozó megírta, de igyekszik minden egyes utasítást a legrövidebb gépi utasítá-

tássorozattá fordítani — a regiszterek tartalmát és más állapotjelzőket figyelembe véve (*kódoptimalizálás*).

A *globális optimalizálás* nagyobb átalakításokat is megenged, például különböző *változókat* azonos helyre rak el, *rekurzív* hívást *ciklussá* alakít, stb. Ilyenkor, ha a programot visszafordítjuk, már nem az eredeti *forrást* kapjuk vissza. ► optimum.

programozási környezet: Lásd IPSE.

programozási nyelv: Programok leírására szolgáló nyelv. A programozási nyelvek mind formájukban, mind filozófiájukban igen sokfélék. Az *alacsony szintű nyelvek* egy géphez (gépcsaládhoz) kötődnek, a *magas szintű nyelvek* gépfüggetlenek. A különböző magas szintű nyelvek mögött különböző programozói filozófiák vannak. ► moduláris programozás, funkcionális programozás, procedurális programozás, objektumorientált programozás.

programozói felület: Lásd IDE 2.

projekció: Lásd adatbázis.

PROLOG (PROgramming LOGic): A nyelv alapötlete az, hogy a problémát az elsőrendű formális logika egy résznyelvében, az úgynevezett Horn-clause-okban kell megfogalmazni, majd a gép egy igen általános megoldási módszerrel, az úgynevezett rezolúcióval keresse meg a megoldást. Az elméleti alapok kidolgozása után 1973-ban *implementálták* először a nyelvet.

Ez eredeti formájában úgy sikerült, hogy a gép a saját elgondolása szerint kereste a megoldást, és igen kevésbé lehetett befolyásolni. A későbbi fejlesztések programnyelvszerű vonásokat és adatbázis-kezelést, *interaktív* ki- és bevitelt, grafikát kapcsoltak hozzá. A *MI-feladatokban* Európában, különösen Magyarországon sokszor alkalmazták.

prompt: Olyan jel vagy *üzenet*, amellyel a párbeszédés üzemmódban dolgozó program azt jelzi, hogy újabb parancsra vár, esetleg azt, hogy inputot kér. A DOS promptjában számos információt lehet megadni: dátumot, időt, aktuális *meghajtót*, az aktuális könyvtárig vezető utat stb.

proporcionális: Mint jelző a nyomtatásra vonatkozik olyankor, amikor a szélesebb betű, például az „m” több helyet foglal el, mint az „i”. Ilyenkor minden betűhöz és jelhez más szélesség tartozik (attó függően, hogy hány pontból áll). Ha *sorkiegyenlítés* is van, a szavak közti hely egyenlően osztódik szét a sorban.

protected mode: Lásd védett üzemmód.

protection: Lásd védelem.

protokoll (protocol): 1. Egy gép és egy berendezés (illetve másik számítógép) közötti adatátvitel szabályai.

2. Számítógép-hálózatban az egymásra épülő protokollok rendszere határozza meg az *üzenetek*, illetve adatsomagok felépítésének és lebontásának (értelmezésének), valamint továbbításának rendjét. ► ISO/OSI.

PS/1 (Personal System/1): Az IBM 1990-ben egy könnyen kezelhető, kis-méretű — és a cég saját áraihoz képest olcsó — gépnek szánta a PS/1 gépeket. A 80286-os CPU-val, VGA adapterrel és 3.5 inch-es floppy-val működő gép teljesítménye azonban erősen elmaradt a versenytársakétól.

PS/2 (Personal System/2): 1987-ben jelent meg az IBM ezzel a gépcsaláddal, és ezzel egyidejűleg leállt az IBM AT gyártásával. A PS/2 sorozattal vezette be a *mikrocsatornát* és a *VGA képernyőadapert*, kezdte el alkalmazni a

80386-os CPU-t és a 3.5 inch-es floppy-meghajtót. A legkisebb típusok a (PS/2 25 és PS/2 30) a 8086-os processzorral és az *MCGA videóadapterrel* készültek. Az 50-es és 60-as típusok a 80286-os CPU-t, a 70-es és 80-as típusok a 80386-os CPU-t alkalmazták és valamennyi VGA adaptert. Később újabb tagokkal bővült a család, a 90-es és 95-ös típusokban jelent meg az *XGA képernyőadapter*.

pszeudo véletlen: Lásd random.

pszeudokód: Lásd Pascal.

public domain: Köztulajdon. Lásd copyright.

puffer (buffer): Tároló, amely az adatok beérkezése és felhasználása közötti ritmusbeli különbséget szünteti meg.

kimeneti szövegátlapuffer: A program karakterenként írja a kimenő fájlt az output pufferba. Az adatok akkor kerülnek diszkre, amikor egy átvitelhez szükséges mennyiség összegyűlt.

printerpuffer: A nyomtató saját memóriája; ide kerül a kinyomtatandó szöveg, és a nyomtató akkor kér további adatot, amikor ezzel végzett.

billentyűzetpuffer: 15 karaktert képes befogadni (ha nincs átállítva a mérete) úgy, hogy közben a program ettől függetlenül működik. Amikor a program egy karaktert kér, azt a pufferből veszi.

A billentyűzetpuffer egy *ciklikus puffer*, ami a következőket jelenti: az előzőektől eltérően nem akkor kerül sor egy kivitelre vagy behozatalra, amikor betelik, hanem folyamatosan lehet beletenni, amíg van benne üres hely, és folyamatosan lehet belőle kivenni, amíg van benne valami. Amikor kiveszünk belőle, a karakter helye felszabadul.

push: Lásd stack.

Q

QEMM-386 (6.0): A Quarterdeck cég univerzális memóriakezelő programja. A 80386, 80386SX, 80486, 80486SX jelű processzorral bíró PC-k 640K feletti memóriájához fejlesztették ki. Mindent a 640-1024 közötti memóriába pakol, amit lehet (*bufferek, driverek, tárrezidens programok*). Felismeri, hogy a program miként akarja használni a memóriát *expanded* memóriának (*LIM EMS 3.2, EMS 4.0* vagy *EEMS* eljárás szerint), avagy *extended* memóriának (*Microsoft XMS* szabvány vagy a saját — Quarterdeck/Phar Lap-féle — Virtual Control Program Interface (VCPI) szerint), és ennek megfelelően kezeli a memóriát. A *QRAM 2.0* hasonló programcskák gyűjteménye *8088-as, 8086-os* és *80286-os* gépekre.

Quattro (Pro 2.0): Kis tárfoglalású, de nagy tudású *táblázatkezelő*, a Borland cég gyártmánya. 3 dimenziós, menüvezérelt, más adatbázisokkal és *táb-*

lázatkezelőkkel kompatibilis, hálózati kapcsolattal rendelkezik. Magas szintű *kiadványszerkesztői* (szöveg- és képszerkesztési) lehetőségekkel van ellátva.

query language (lekérdező nyelv): Az adatbázisok kezelésének résznyelve, amellyel kiválogathatunk és megmutathatunk adatokat.

queue (sor): Más néven FIFO (First In, First Out) lista. Átmeneti tárolásra szolgáló adatszerkezet; jellemzője, hogy megőrzi a sorrendet, mindig a legrégebben beletett adatot vesszük ki belőle. *Multiprogramozott környezetben* a queue-kat gyakran használják az egymásnak küldött *üzenetek* tárolására.

quit: Befejezés. Kétféle értelemben használják: 1. A program normálisan (híba nélkül) végetért.

2. A programmal dolgoztunk ugyan, de ezt abbahagyjuk, mégpedig úgy, hogy a kapott eredményre nincs szükség. (Például szövegszerkesztésnél: meggondoltuk magunkat, a változtatásainkat mégsem tesszük bele a szövegbe, az maradjon az előző állapotában.)

R

RAM: Lásd memória.

RAMdiszk: Olyan *virtuális diszk-meghajtó*, amely valójában nem egy igazi hardvereszköz, hanem a memóriában van. IBM PC-n a *CONFIG.SYS* fájlban lehet definiálni: *RAMDRIVE.SYS* vagy *VDISK.SYS* eszközdefinícióval (device definition) jön létre. Mérete minimum 64K, maximum 4M, de több is létesíthető. Írható, olvasható, de *formattálni* nem kell és nem lehet. Előnye az igazi harddiszknél gyorsabb működés, de a gép kikapcsolásakor tartalma — akár csak a memóriáé — elvész. Közbülső eredményeket tárolunk benne.

random: Véletlen. Számos program nem viselkedik determinisztikusan, hanem véletlenszerűen választja meg a következő lépését. Ehhez ún. véletlen számokra van szükségük.

A véletlen szám előállítható erre a célra szolgáló hardverrel, a *véletlen-szám-generátorral*; *pszeudo véletlen* (véletlennak látszó) számokat előállító függvényekkel; *interaktív* programoknál pedig a felhasználó reakcióidejéből, a gép órájának állásából vagy ezek kombinációjából képzik.

A statisztikai programok tervezésekor jelentősége van annak, hogy milyen eloszlású a véletlen számok sorozata. (Poisson-, normális, egyenletes eloszlás stb.)

random access, random elérés, random hozzáférés: Lásd adatállomány.

ray tracing (sugárkövetés): Térbeli természetes fényviszonyokat imitáló ábrázolási *algoritmus*. Azon alapul, hogy a nézőpontból kiindulva végigpásztazza a kép minden egyes pontját, és ahol tárgyba ütközik, ott megállapítja az

fényforrásból odaérkező fénysugarak beesési szögét.

read-only: Csak olvasható.

Read-Only Memory, (ROM): Olyan memória, amelybe *beleégették* a programot; csak olvasni lehet.

read-only file: Olyan fájl, amelynek *katalógus-bejegyzésében* beállították a felülírást tiltó bitet.

read-only variable: A *konstansok* neve egyes nyelvekben. ► fájlattribútum, írásvédelem.

real: Lásd valós típus.

real mode: Lásd valós üzemmód.

real-time (valós idejű): Elegendően gyors, hogy valamilyen külső eseményre időben reagáljon. Elsősorban berendezéseket vezérlő programok, illetve ilyen programokat futtató operációs rendszerek jelzője. ► megszakítás.

recovery: Újraélesztés. A rendszer újra működőképes állapotba hozása valamilyen *végzetes hibából* származó leállás után.

redlining: Korrektúrázást segítő eszköz. A szövegben látható lesz, hogy mit húztunk ki és mit írtunk a helyére. Az eredeti és a javított szöveg is megkapható e funkcióval.

redo: Lásd undo.

redundancia, redundáns: Ha az adattömeg többszörözve tartalmazza a benne foglalt információt vagy egy részét, akkor redundáns. A redundancia lehetőséget nyújt a sérülések felismerésére, esetleg javítására is. ► inkonzisztencia.

regiszter: Néhány bájt tárolására alkalmas áramkörü elem. Regisztereket használnak a különböző bonyolultabb részegységekben és a processzorokon belül. Funkció szerint vannak vezérlő, adat- és állapotregiszterek.

regisztrálás: Lásd copyright.

rekord (structure) típus: Egyes nyelvekben nevezik struktúrának. Olyan összetett adat, amely több, különböző típusú részadatból épül fel; ezeket mezőknek nevezzük, és nevet adunk nekik. Ezért inhomogén adatstruktúrának minősül. (Szemben a *tömbbel* vagy a *listával*, amely azonos elemekből épül fel.)

A rekordra alkalmazható művelet a mezőkiválasztás. Ilyenkor a rekord típusú adat neve után pontot teszünk és leírjuk a mező nevét. Az összetett nevet *kiterjesztett névnek* hívjuk. A mezőnevek önmagukban nem használhatók, csak kiterjesztésként.

Változó rekordnak (variable record) nevezik az olyan rekordot, amelyben ha egy mezőnek bizonyos értékei vannak, akkor bizonyos további mezők szerepelnek a rekordban, ha más értékei vannak, akkor más mezők kerülnek bele. A *Modulában* ezt a meghatározó mezőt *szektor mezőnek*, az *Adában* *diszkriminánsnak* nevezik. A fejlettebb nyelvekben az értékadás és az egyenlőség ellenőrzése is beépített művelet. ► adattípus.

rekurzív, rekurzió: Lásd szubrutin.

relációs adatbázis: Lásd adatbázis.

release: Lásd változat.

relocatable form (áthelyezhető forma): Lásd linker.

repeater (ismétlő): Két hálózat összekapcsolását lehetővé tevő eszköz. Erősíti és a másik hálózatra továbbítja a kapott biteket. Ha a két adatátviteli közeg különböző (például réz- és fénykábél), a jelsorozaton nem változtat, csak az adathordozónak megfelelő alakra hozza, megismétli. Csak azonos hozzáférési protokollal és egyeztetett állomáscímekkel működő hálózatok köthetők össze vele fizikai szinten. ► bridge, gateway, router.

report: 1. Jelentés. Általában azoknak a programozási nyelveknek és prog-

ramoknak a definícióját nevezik így, ahol egy bizottság hozta létre a definíciót. ► szabvány, manual.

2. Egy nyilvántartás kiválogatott adataiból készült *lista*. ► adatbázis-kezelés.

reprezentáció (ábrázolás): Egyrészt úgy merül fel, mint probléma, hogy miként ábrázoljunk bizonyos fogalmakat egy adott programozási nyelv eszközeivel. Másrészt: hogyan ábrázolja a gép a nyelv fogalmait. A *reprezentáció-specifikáció* egy olyan nyelvi lehetőség, amellyel előírhatjuk, hogy a nyelv fogalmait miként ábrázolja a gép — és ezen keresztül a mi eredeti fogalmaink gépi ábrázolását is. ► packed.

reset: Visszaállítás, újraindítás.

rezidens program (Terminate and Stay Resident, TSR): Olyan program, amely bemásolódik a memóriába és ott marad. A később indított programok nem írják felül. A rezidens programok más programok futása alatt bizonyos billentyűkombinációk (*hot key*) megnyomásával aktivizálhatók (és inaktivizálhatók). (Néhány ilyen program: a magyar *billentyűzetkezelő*, az *ASCII* kód-táblát vagy az aktuális *directory-t* kiíró, az *egérmeghajtó*, a *képernyőmentő*, a szóhelyesség-ellenőrző, de lehet közöttük például szinonimaszótár-program stb.)

RGB (Red-Green-Blue): Vörös-zöld-kék. A színes képernyő vezérléséhez szükséges, az egyes színek intenzitása szerint lebontott információ. *Analóg* esetben az egyes színek továbbítására külön vezetékek szolgálnak, de lehet az információ *digitálisan* kódolva is.

RISC (Reduced Instruction Set Computer): A mikroprocesszorok fejlesztése során sokáig egyforma fontosságú volt a fizikai teljesítmény növelése és az, hogy törekedjenek a programozás ké-

nyelmének fokozására — újabb és újabb, bonyolultabb utasítások felvételével az adott CPU *utasításkészletébe*.

A 80-as évek közepén azonban nyilvánvalóvá vált, hogy az egyre komplikáltabb utasításokat igazában csak az *assembly* programozók szeretik. A *magas szintű nyelvek fordítóprogramjai* vagy nem tudják a komplikált utasításokat jól kihasználni, vagy hosszadalmas optimalizálási folyamattal találják meg a legjobb utasításkombinációt. Felmerült az igény, hogy inkább egy-két tucat nagyon általános, egyszerű, áttekinthető és nagyon gyors utasításra lenne szükség. Az elv sikeresnek bizonyult, a RISC processzorok teljesítményben megelőzték a CISC típusokat, szinte minden nagy gyártó kínál már RISC alapú gépeket.

A RISC gépek természetesen nem kompatibilisak az IBM PC-vel, de többségükre *implementálva* van a UNIX, így a C nyelvű programok viszonylag könnyen átvihetők rájuk.

RLL (Run Length Limited): A hardiszkek kapacitását döntően befolyásolja az adatok mágneses impulzusokkal való rögzítésének módja. Az FM majd az MFM után a legújabb és leghatékonyabb kódolás az RLL. Mivel a lemezen a fluxusváltozások távolsága nem csökkenthető bizonyos határ alá, a cél az adatokat minél kevesebb mágneses impulzussal (1-es bittel) rögzíteni. Ezért nem az egyes bitekhez, hanem 2, 3 vagy 4 bitből álló bitsoporthoz rendelnek

egy-egy kódot, amely az eredeti bitsoporthal azonos vagy kevesebb számú 1-est tartalmaz, így jobban sűrítendő a lemezen. A kódok hozzárendelése többféle lehet, ezért többfajta RLL-kódolás létezik. Megkülönböztetésükre megadják a két 1-es közé eső 0 bitek minimális és maximális számát. Eszerint a legelterjedtebb a (2,7)RLL-kódolás.

ROM (Read-Only Memory): Lásd memória.

ROM BIOS: Lásd BIOS.

root: Gyökér. Egy irányított fával leírt *gráfban* vagy egy ilyennel leírható hierarchiában a kiinduló pont, a hierarchia legmagasabb pontja.

rounding (kerekítés): Lásd truncating.

router (kapcsoló-irányító, útvonal-kijelölő): Két eltérő topológiájú és hozzáférési protokollt használó hálózat összekapcsolására alkalmas számítógép. A kapcsolatot ez mint konverter az OSI modell 3 alsó hálózati rétegének szintjén valósítja meg, így ezeknek a két hálózatban azonosnak kell lenniük.

RS-232-C: Lásd soros port.

RS-449: Lásd soros port.

runtime: Futásideji. **Runtime error:** futásidőben jelentkező vagy csak akkor felismerhető hiba. **Runtime library:** azon *függvények* gyűjteménye, amelyekre futás közben a programnak szüksége lehet, és amit ezért hozzá kell *szerezni*.

runtime package: Lásd interpreter.

Rushmore-technika: Lásd FoxPro.

S

SAA (System Application Architecture): Az IBM által kidolgozott és bevezetett szabvány, amely előírja, hogy milyen interfészt kell mutatnia egy programnak a felhasználó és az operációs rendszer felé. A cél az, hogy akár nagy-, akár kisgépen, akár PC-n a program ugyanúgy viselkedjen; és a különböző programok viselkedése ugyanazt a sémát kövesse.

sáv: A hajlékony- és a merevlemezek felületén az adatok koncentrikus körökben, sávokban helyezkednek el. A lemezfelületen lévő sávok száma függ a floppy-lemez, illetve a harddisk típusától. A sávra a fej pozicionálásával lehet ráállni. A floppy-nál a sávok kezdetét az *indexlyukkal* jelölik meg. A sávok több — szintén a típustól függő számú — *szektorra* vannak osztva.

scan kód: Az IBM PC típusú gépek *billentyűzetét* vezérlő beépített processzor minden billentyűhöz scan kódokat rendel, és a billentyű lenyomásakor és elengedésekor ezeket a kódokat küldi el soros jelek formájában az *alaplappra*. A központi CPU által beolvasott scan kódhoz az INT 9h *megszakításrutin* rendel szoftver úton *ASCII kódot*. Ez a megoldás teszi lehetővé, hogy az egyes billentyűkhöz rendelt betűk és funkciók — az INT 9h megszakítás átírásával — szoftver által megváltoztathatók. (Vannak olyan programok, amelyek ezt megteszik a saját futásukhoz, de a végén vissza is állítják az eredeti megszakítást.)

scanner: Lásd szkennер.

scheduler, scheduling (ütemező, ütemezés): Lásd ütemezés.

scope (hatáskör): Az a terület, ahol egy felhasználó által deklarált *azonosító* érvényes, felhasználható. ► deklaráció, lokális, globális.

scratch: Olyan memóriaterület vagy fájl, ahová a program olyan közbülső adatokat, részeredményeket ír, amelyekre később nincs szükség, így a program lefutásával elvesznek.

screen: Lásd képernyő.

scroll(ing): Egy dokumentum a képernyőn általában nem látható teljes egészében. (Alul, felül, oldalt kilóghat.) Egy programnak azt a működését, amikor a képernyőt mint egy nyílást feljebb vagy lejjebb, esetleg jobbra vagy balra mozgatjuk a dokumentum felett — soronként vagy annál kisebb egységekben — *szkrollozásnak* nevezzük. A képernyő hardverje segíti a szkrollozást, de teljesen nem oldja meg. A könyvtári *ablakkezelő* rutinok nagyrészt megoldják a problémát.

Amikor egy sornál vagy oszlopnál nagyobb egységekben (például félképernyőnként) mozgunk a képernyő szélére érve, akkor ez az ún. *flipping*.

SCSI (Small Computer System Interface): Az SCSI interfészt 1986-ban fogadta el szabványként az *ANSI*. Magas szintű — eszközfüggetlen — interfész; egy kétirányú, 8 bites adatbuszt és ehhez szükséges vezérlőbuszt tartalmaz. Az eszközfüggetlenségnek megfelelően a buszon parancsokat továbbít, és logikai címezést használ. A parancskészletet és parancstovábbítási eljárást pontosan definiálja a szabvány.

A parancs végrehajtása az *SCSI vezérlő* feladata, amely ehhez elég intelligens — tulajdonképpen egy célszámítógép. Leggyakrabban harddisk-interfészként használják az SCSI-t, de a vezérlőjétől függően bármilyen más perifériához vagy két számítógép összekap-

csolására is alkalmas. Az interfészre 8 vezérlő csatlakozhat, de mindegyik alá rendelhető 256 logikai egység.

Az SCSI busz fizikai megvalósítása kétféle lehet; közös földvezetékekkel — ezt egylezésűnek hívják —, vagy minden vonalhoz külön földdel, ez esetben differenciálisnak nevezik. Az áthidalható távolság egylezésűvel max. 6 méter, a differenciálissal 25 méter. Az adatátvitel lehet *szinkron* vagy *aszinkron*, szinkron módban az elérhető sebesség 10 Mbájt/sec, aszinkron módban 4 Mbájt/sec. Az interfész továbbfejlesztése az *SCSI-2*: ennek nagyobb a parancskészlete, továbbá a szélesebb — 16 vagy 32 bites — adatbusz révén gyorsabb is. Szinkron módban 40 Mbájt/sec érhető el, itt azonban az egylezésű busz hossza csak max. 1.5 méter lehet.

search (keresés): Lásd fájlműveletek.
sector: Lásd szektor.

seek (direkt pozicionálás): Lásd fájlműveletek.

segédprocesszor: Lásd koprocesszor.
segment: Lásd szegmens.

sequential (szekvenciális, sorrendi): Sorban, egyiket a másik után. Fájlokkal kapcsolatban például *szekvenciális hozzáférés*; a szekvenciális gép lépésenként dolgozik; a szekvenciális *algoritmus* egy normális algoritmus — a szó itt azt hangsúlyozza, hogy nincs benne párhuzamosság vagy véletlen elágazás.

Ellentéte: véletlenszerű (random), párhuzamos (paralel).

serial (soros): Az átvitel jelzője, amikor a bitek egyenként mennek át.
► soros átvitel.

serious error: Lásd hiba.

server: Lásd szerver.

set: Lásd halmaz típus.

setup: Beállítás. Az üzembe helyezéshez szükséges kezdőértékek beállítása.

shared: Osztott (memória, adat, változó, fájl, adatbázis). Párhuzamos környezetben használatos kifejezés. Azt jelenti, hogy az adatot több program egyszerre írhatja és olvashatja. Mivel rendszerint nem egyetlen bájtról van szó, hanem egy nagyobb adatról, amelynek teljes beírása, illetve kiolvasása több utasítást igényel, egy ilyen művelet idejére az adatot vagy a megfelelő részét el kell zárni más műveletek elől, mert az adat *inkonzisztens* állapotba kerülhet. Fájloknál, adatbázisoknál a védelem lehet fájl- és rekordszintű.

A dolog másik aspektusa, hogy egy *osztott* adattal, pláne osztott adatbázissal kapcsolatban a különböző felhasználók különböző jogokkal és *prioritásokkal* rendelkeznek. Nagyobb rendszerekben ezek kezelése meglehetősen komplikált. Néha (helytelenül) a *distributed* szót is ilyen értelemben használják. ► lock, multiprogramozás, multiprocesszoros rendszer.

shareware: Szabadon terjeszthető program. Lásd copyright.

shell: 1. *Parancsértelmező*. Az operációs rendszernek az a része, amelyik a parancsokkal foglalkozik, ez fájlok másolását, törlését, áthelyezését, programok indítását teszi lehetővé, továbbá hasonló alapvető dolgokat. A *UNIX*-ban 3-féle shell közül választhatunk (Bourne-shell, C-shell, Visual-shell). Az *MS-DOS* eredeti változatában csak egy primitív, sororientált parancsértelmező volt, de számos jó shellt építettek fölé: *Norton Commander*, *PathMinder*, *PC Tools* stb.

2. *Keret(rendszer)*. Szakértő rendszerekben shellnek nevezik a még kitöltetlen rendszervázat, amelyben megvan a tudásbázis helye, a következtetési szabályok helye, a következtető mechaniz-

mus, de még nincs beletáplálva egyetlen szakterületről sem a felhalmozott tudás.

shift (léptetés): Egy bitvektoron (bináris számon) a bitek elmozdításával végrehajtott művelet. Történhet jobbra és balra.

A művelet **logikai léptetés**, ha a szélén álló kilépő bit elveszik, és a másik szélén egy 0 lép be. Akkor **ciklikus vagy cirkuláris léptetés**, amikor a kilépő bit a másik oldalon belép; és akkor **aritmetikai léptetés**, ha a legbaloldalibb bit (a szám előjele) balra léptetésnél nem mozog, csak az utána lévők lépnek, de jobb oldalon 0 lép be; jobbra lépésnél az előjelbit átmásolódik a következőbe, a többiek pedig jobbra lépnek. Az „aritmetikai léptetés balra” a kettővel való szorzással egyenértékű (feltéve, hogy a szám kétszerese is ábrázolható), az „aritmetikai léptetés jobbra” pedig kettővel való osztás, de így sokkal gyorsabb. A szám egyszerre több bittel is léptethető.

és 'vagy' kapcsolatokat tartalmazó **logikai feltételt** úgy számítunk ki, hogy a kiszámítást azonnal abbahagyjuk, amint egy 'és' kapcsolatban találunk egy nem teljesülő feltételt — ilyenkor azt mondjuk, hogy a konjunkció nem teljesült; illetve a 'vagy' kapcsolatban az első teljesült feltétel esetén igaznak nyilvánítjuk a diszjunkciót.

A rövidzár-kiszámítás majdnem mindig ugyanazt az eredményt adja, mint a formális logikában előírt kiszámítási mód; kivéve, ha a konjunkció második tagjának kiszámítása hibát okoz. Ilyenkor egyik esetben a program lefut, másik esetben hibával ér véget. A rövidzár-kiszámítás az emberi gondolkodás természetes formája. Ha azt mondjuk, hogy „Ha k nem nulla, és a szám k-val osztva 1 maradékot ad ...”, akkor senki nem gondolja, hogy k=0 esetén is meg kellene nézni a másik feltételt, hiszen az

nyilvánvaló módon elvégezhetetlen. Ezért a modern nyelvek rövidzár-kiszámítást alkalmaznak; a régebbi nyelvekben a konjunkció elemeinek kiszámítási sorrendje és az, hogy minden tagot ki kell-e számítani, tisztázatlan volt.
► konjunkció, diszjunkció.

side effect (mellékhatás): Amikor egy függvény nemcsak egy értéket ad vissza, hanem a külvilágát is megváltoztatja. Tipikus példa erre a C nyelv „getc” függvénye, amely az *argumentumként* megadott fájlról egy karaktert olvas be, eközben természetesen a fájlban egy karaktert előre lép, és a következő hívásnál ugyanez a függvényhívás más eredményt ad. Ebből akkor támadhat zavar, ha egy kifejezésben nincs megszabva az argumentumok kiszámításának sorrendje. Ezért például ha a *logikai* kifejezések rövidzár-kiszámítása (*short circuit evaluation*) esetén azt írjuk, hogy: „Ha a getc-vel olvasott karakter 'A' és a (következő) getc-vel beolvasott karakter 'B'”, amennyiben az első feltétel teljesül, azaz a beolvasott karakter 'A', akkor a következő karaktert is be kell olvasni; ha viszont nem volt az, az első karakter után megállunk. De ha nincs előírva a rövidzár, a Jóisten tudja csak, hogy mi történik.

A probléma elvi megoldása az, hogy függvénynek ne lehessen mellékhatása, amelyeknek mellékhatása van, az legyen *szubrutin*. A szubrutin csak önálló utasításként állhat, az utasítások sorrendje viszont megszabott.

SideKick: A Borland cég *memória-rezidens* ún. desktop-organizátora. Az íróasztalon levő holmikat helyettesíti. A 2.0 változatban van 9 *szövegszerkesztő*, 4-féle *kalkulátor* imitált szalaggal, telefonkezelés, cím- és telefonszámnyilvántartás, hálózati kapcsolat, előjegyzési napló automatikus riasztással,

sokféle nyomtató kezelése. A nyomtató anyag előzetesen megtekinthető (*page preview*), és többféle papírméret nyomtatható. Kezeli a DOS-fájlokat és -könyvtárakat.

SIMM (Single Inline Memory Module): Az *alaplapon* lévő RAM memóriák korábban csak *DIP* tokozással készültek; és az egy memóricímhez tartozó 8 bit nem egy tokban volt, hanem kettőben vagy nyolcban. A SIMM egy újabb memória-beépítési, illetve -tokozási forma, amely az előbbi értelemben egy teljes bájt szélességű. Kisebb, tehát ezért is egyre jobban terjed, sok alaplapon már csak ilyen — vagy *SIPP* tokozású — memória használható.

SIMSCRIPT: *Szimulációs nyelv* — diszkrét rendszerekre.

SIMULA: A párhuzamos folyamatok leírására és szimulálására alkalmas nyelv a 60-as évek közepéről. Számos olyan nyelvi konstrukciót vezetett be, amely a modern nyelvek és programozási koncepciók alapja lett (*coroutine* — *Modula-2*; *Class*, *öröklődés* — *OOP*).
► *szimulációs nyelv*.

SIO (Serial Input-Output): Olyan *integrált áramkörök* összefoglaló neve, amelyek *soros, aszinkron adatátvitel* vezérlésére alkalmasak.

SIPP (Single Inline Plugging Package): Azonos funkciójú, mint a *SIMM* tokozás, de az érintkezők kialakítása más, így a kettő nem kompatibilis egymással.

sín: Lásd busz.

SLQ (Super Letter Quality): Különleges levélminőségű nyomtatás. A 24 tűs *mátrixnyomtatóknál* használható üzemmód és nyomtatásminőség.

SMALLTALK: A Xerox cégnél 1980-ban kifejlesztett nyelv, az első *objektumorientált* programnyelv. A Xerox *munkaállomás* teljes operációs rend-

szere is ezen a nyelven íródott, a nyelv és a rendszer erősen összefonódik. A *SMALLTALK*-ot fejlesztői olyan grafikus felhasználói interfésszel látták el, hogy vele a programozáshoz nem értő, de némi képzelőerővel megáldott egyén is tud programot írni. ► *OOP*, *GUI*, *ObjectVision*.

smart: *Intelligens* (agyafúrt).

smart terminal: *Intelligens terminál*. Ellentéte a buta (*dumb*) terminál.

smart linkage: Egy nyelvnek az a tulajdonsága, hogy a *modul külső hivatkozásainál* nemcsak a külső *függvények* nevét, hanem a *paramétereinek* a típusát is meg kell adni, és ezt a *fordító* ellenőrzi. A *Pascal*, *Modula*, *C++* smart, a *C* nyelv nem. ► *adattípus*, *erős típusellenőrzés*.

SNA (System Network Architecture): Az IBM cég hálózati szabványa. Az *ISO/OSI* szabványhoz hasonlóan rétegeket definiál, de máshol húzza meg a rétegek határait. ► *hálózat*.

softfont, **softfont editor**: Lásd font.

software: Lásd szoftver.

sorkiegyenlítés (*justification*): Amikor egy szövegrészt úgy rendezünk át, hogy a széle a jobb és bal margóra kerüljön. Nyomdászati igényű megoldása programmal a szavak elválasztásával és az elválasztottak egyesítésével jár, ennek pontos és helyes végrehajtása nagyon nehéz feladat, mert az elválasztási szabályok feltételezik a szóösszetétel felismerését.

soros adatátvitel (*serial transmission*): Az adatok továbbítása a számítógépek és perifériák között *párhuzamos* vagy *soros átvitellel* halad. Soros átvitelnél a bájtok bitjei egy vezetéken sorban követik egymást. Kevés vezetéket — legalább kettőt — igényel, ezért a párhuzamos átvitelnél nagyobb távolságra, néhányszor 10 méterig is megfe-

lelő. Mivel *handshake* jeleket nem használ, fontos, hogy a vevő oldalon a megfelelő időpillanatokban vizsgálják a vezetéken az érkezett biteket, azaz a vevő és adó oldalnak szinkronban kell működni. Ehhez egyeztetni kell a bitek továbbítási sebességét, vagyis az átvitelt ütemező *órajel* frekvenciáját, és észlelni kell a bájtok kezdetét. Ez két módon biztosítható. *Aszinkron átvitelnél* az adat biteit egy startbit előzi meg, a végén pedig stopbit zárja le. Ha nincs küldeni való adat, a vezeték üres. *Szinkron átvitelnél* nem használnak stop és start biteket, azonban adásszünetekben előre megállapított szinkronkarakterek folyamatos küldésével tartják együtt az adó és vevő oldalt.

A csak egyirányú adatátvitelt *szimplex* üzemmódnak hívják, az átkapcsolhatóan kétirányú átvitelt *félduplexnek*, az automatikusan, két adat között is irányváltásra képes, látszólag kétirányú átvitelt pedig *duplexnek*. ➤ párhuzamos adatátvitel.

soros (serial) port: Az IBM PC típusú gépeknél a *soros, aszinkron adatátvitelre* alkalmas csatlakozót nevezik soros portnak. Elsősorban a számítógép és a *modem* vagy a *terminál* közötti, kétirányú adatátvitelre szolgál. Az IBM PC/XT/AT gépekben az *RS-232-C* szabványú soros port terjedt el, a *CCITT V.24.* ajánlása csak némileg tér el tőle. A szabvány egy 25 pólusú csatlakozót definiál, amellyel 15 méteres távolságon max. 20 kbit/s sebesség érhető el, bár rövidebb távolságon magasabb érték is. Az átvitelhez nem szükséges valamennyi, a szabványban definiált jel, ezért sok esetben csak egy 9 pólusú csatlakozót alkalmaznak. Az *RS-232-C* továbbfejlesztése az *RS-449* szabványú interfész, amelyet néhány kiegészítő áramkörrel láttak el.

SoundBlaster: *Hangkártya* hangszóró-kivezetéssel. Van *AdLibbel* kompatibilis üzemmódja. A *Game Blaster üzemmódban* a hanggal ellátott játékok lejátszására alkalmas. Képes digitalizált hangfelvételre és lejátszásra. Felhasználását segíti a *Blaster Master* nevű program, melynek *shareware* változata is van.

source code, source text: Lásd forrásszöveg.

SPARC (Scalar Processor ARChitecture): *RISC* mikroprocesszor, illetve egy erre épített *munkaállomás*. A SUN Microsystem terméke. A cég ezt a mikroprocesszort mások körében is el akarja terjeszteni.

specifikáció: Valamilyen adattípus vagy eljárástípus tulajdonságainak definíciója anélkül, hogy az adott típusból egyetlen konkrét példányt hoznánk létre. Például megadjuk a rekordtípus nevét, a mezőinek a nevét és típusát, de egyelőre egyetlen ilyen rekordot sem hozunk létre; vagy megadjuk a *függvény* nevét, paramétereinek és visszaadott értékének típusát, de még nem adjuk meg a törzst, azaz a végrehajtandó utasításokat. Egyes nyelvekben *típusdefiníciónak* nevezik.

spell(ing) checker: Szóellenőrző, betűzés-ellenőrző. Olyan program, amely megvizsgálja, hogy a leírt szavak helyesek-e. Angol nyelvre, mivel a szavaknak csak néhány alakjuk van, könnyű készíteni, régóta és sokféle van belőle. Egy magyar szónak több ezer képzett és ragozott alakja van, ezért sokkal nehezebb megoldani, de 1991-ben elkészült a NyelvÉsz, majd a HELYES-E? is. A spell-checker program csak az egyes szavakat nézi, a szövegkörnyezet nélkül, nem veszi észre, hogy a mondat értelmetlen. A szóelválasztást általában tudja kezelni.

SPLIT: NC programozási nyelv. Lásd NC 2.

spooler: Elsősorban nyomtatóknál alkalmazott technika. A nyomtatandó fájlt a spooler programnak küldjük el. Ez más programok mellett, azok zavarása nélkül a háttérben fut, és a kapott fájlokat egymás után kiküldi a nyomtatóra. Célja kettős: egyrészt a programnak nem kell várnia a nyomtatóra — ami egy lassú periféria —; másrészt *multiprogramozott* rendszerekben megakadályozza, hogy a nyomtatóra küldött fájlok összekeveredjenek.

spreadsheet: Lásd táblázatkezelő.

sprite: Olyan hardver lehetőség, amellyel a képernyőn lévő kép kis része felülírható egy téglalap alakú ábrával. (Lehet átlátszó része is, és az alapkép a memóriában nem változik meg.) Helye koordinátákkal adható meg, és gyorsan változtatható.

SQL (Structured Query Language): Olyan adatbázis-kezelő nyelv, amelyik bármely *relációs adatbázis* létrehozására, kezelésére, lekérdezésére alkalmas, amelynek van SQL-interfésze.

Párbeszédés üzemmódban és másik nyelvbe beágyazva egyaránt alkalmazható. A 70-es években az IBM SEQUEL projektje kapcsán született. Szinte az összes modern adatbázishoz van SQL-interfész.

SRAM: Lásd memória.

ST506/412: A Shugart cég fejlesztette ki ezt a harddisk-interfészt az 1970-es évek végén. Nevét a cég által 1980-ban gyártani kezdett 5 Mbájtos harddisk interfészéről kapta, ennek a típuszáma volt az ST506. A Seagate néhány évvel később megjelent ST412 típusú, 10 Mbájtos *winchestere* szintén ezt az interfészt használta.

Az ST506/412 eszköz szintű interfész, ami azt jelenti, hogy a harddisk

minden számára szükséges vezérlőjelet az *adaptertől* az interfészen át kap. A interfész külön kábelen továbbítja a vezérlőjeleket és az adatokat. Az adatok *kódolását-dekódolását* az adapter végzi, az adatkábelen már kódoltan haladnak. Az interfész kezdetben kizárólag *MFM*-kódolást alkalmazott, ezért gyakran nevezték *MFM* interfésznek. Az *MFM*-kódolású adatokat 5 MHz frekvenciával írják a lemezre, ezért az interfész adatátviteli sebessége 5 Mbit/sec. 1986 óta áttértek az *RLL*-kódolásra, ez nagyobb bitsűrűséget tesz lehetővé a lemez felületén; ezzel a kódolással változatlan vezérlőjelek mellett 7.5 Mbit/sec átvitel érhető el. Az interfészre elvileg max. 4 *meghajtó* csatlakoztatható, azonban a legtöbb a adapter csak két meghajtót kezel.

stack (verem): Olyan tároló, amelynek csak a tetejére lehet tenni az új adatot, és kivenni is mindig csak a tetején levő adatot lehet. Elvileg két művelete van: *push* (tedd bele) és *pop* (vegyék ki a veremből). Kiderült azonban, hogy az ilyen stack meglehetősen korlátozott képességű. A gyakorlatban olyan vermet használnak, amely a *push* és *pop* utasításon kívül megengedi a stack-relatív írást és olvasást is. (Azaz megmondhatjuk, hogy a stack tetejéhez képest hányadik elemet akarjuk írni vagy olvasni.)

stand-alone: Önmagában működő. Olyan berendezés vagy program, amely önállóan működik, más programok vagy berendezések segítségével nélkül. (Nincs beleépítve egy nagyobb rendszerbe.)

standard fájlok: A DOS és a *UNIX* operációs rendszerben van 3 standard logikai fájl: a standard input (*stdin*), a standard output (*stdout*) és a standard error (*stderr*) fájl. Az első azonos a *kla-*

viatúrával, másik kettő a képernyővel, hacsak másként nem jelöljük ki.

Az operációs rendszer számos parancsa és egy sor rendszerprogram az inputját az `stdin` fájlból veszi és az `stdout` fájlba küldi az eredményt, a hibajelzések rendszerint a `stderr` fájlba mennek. Így a hibajelzések nem keverednek az outputtal; ha csak *figyelmeztetéseket* kapunk, az output további feldolgozása lehetséges.

Csak a képernyő soronkénti írása esetén ajánlatos, komolyabb képernyőkezelés esetén nem alkalmazható. ► parancs.

statikus: Fix, állandó, nem változtatható. ► dinamikus.

stderr, stdin, stdout: Lásd standard fájlok.

stíluslap (style sheet): Lapok formáját leíró fájl, amely megadja a margókat, az oszlopok szélességét, a paragrafusok bekezdését, a különböző helyre kerülő karakterek alakját. Segítségével több azonos külalakú dokumentum készülhet. Kompatibilis stíluslapok között automatikus formátumkonverzió kérhető.

stream: A *UNIX* (és a *C nyelv* is) megkülönbözteti egy I/O eszköz logikai (belső) nevét — ez a `stream` — és a fizikai (külső) nevét — ez fájl. Az *open* utasítás egy streamhez egy fájlnevet kapcsol (vagy valamilyen berendezést). *Bináris* streamek esetén a kívülről jövő bájt sorozatot változatlanul kapja meg a felhasználó, szövegstreamek esetén azonban bizonyos konverzióra kerül sor. Ilyen például, hogy a fájlban szereplő sorvégjelet (*UNIX*-ban `LF=10`, *DOS*-ban `CR=13` decimális értékű karakter) `CR,LF` sorozattal helyettesíti, ha pedig a fájl végén túl akarunk olvasni, *EOF* karaktereket küld be.

streamer: A magnószalaggal egyező elven működő *háttértár*. A biteket

— mint a floppy vagy a *winchester* — *digitálisan* rögzíti. A használt magnószalag az audió- vagy videokazettához hasonló, a *meghajtóban* cserélhető kazettában helyezkedik el, kapacitása 40 Mbájtól Gbájtig terjed. Az igen lassú hozzáférés miatt csak *archiválásra* való.

string típus (karakter)lánc, (karakter)füzér, karaktorsorozat: Egyes nyelvekben karaktertömb (array of characters), más nyelvekben önálló adattípus. Szokásos műveletei: átmásolás (*értékadás*), összefűzés (*katenáció, konkatenáció*), összehasonlítás.

A stringekkel kapcsolatosan felmerülő kérdés, hogy mi történik olyankor, és az eredmény kerüljön a másodikba. Egyes nyelvek ilyenkor megszabják, hogy az eredményt először meg kell csinálni valahol, és csak azután kerülhet a célváltozóba. Más nyelvekben az eredmény rögtön a célváltozóban képződik, így mire az első string után elővesszük a másodikat, mivel ő egyben a célváltozó is volt, módosult az elsőnek az értékével — ezt másoljuk az eddigi eredményhez. Így az első string változó tartalma fog ismétlődni a második string változó teljes hosszában. (A string szóközzel való feltöltésére szokták ezt használni.)

Az összehasonlításnál az egyik string akkor kisebb a másikonál, ha megegyezik az elejével, vagy ha a megegyező rész után következő karakterkód értéke kisebb.

Régebben két stringábrázolási forma volt elfogadva: az egyiknél a sorozat hosszát kell megadni és magát a sorozatot, a másikonál a sorozat egy 0 értékű karakterrel van lezárva. Az utóbbi időben a második ábrázolási forma vált a nyelvekben elfogadottá.

stub: Olyan *szubrutin*, amelynek üres a törzse, azaz nem csinál semmit. Félig

kész programokba stubok vannak a megíratlan részek helyén, így a programot, legalábbis a megírt részeit már félkész állapotban is ki lehet próbálni.

style sheet: Lásd stíluslap.

swap: Ki- és besöprés. A program olyan szervezése, hogy egy nagyobb programrészhez tartozó adatokat akkor hozunk csak be a diszkról, amikor a programrész futni kezd, és amikor a fu-

tását befejezte, a hozzá tartozó adatokat ismét kivisszük a diszkre; az így felszabadult helyre a következő programrész adatai kerülhetnek.

A memória növekedése és a modernebb memóriaszervezés mellett nincs rá szükség.

switch: Kapcsoló.

syntax analyser: Lásd szintaxis.

system board: Lásd alaplap.

Sz

szakértő rendszer (expert system): A *mesterséges intelligencia* eredményeit alkalmazó piaci program, amely tanácsot ad, döntést hoz vagy valamilyen feladatot old meg. Fő komponensei: a tudott tények (knowledge base = tudásbázis), a következtetési szabályok (inference rules) és a következtetési mechanizmus (inference engine). A tények egy része eleve adva van, a másik részét az adott feladat kapcsán kapja meg a program. A rendszer nemcsak a következtetést vonja le, hanem azt is közölni tudja, hogy milyen lépéseken keresztül jutott ide.

számítógép (computer): Pontosabban: tárolt programú digitális számítógép; Neumann János, Arthur W. Burks és Hermann H. Goldstein által kidolgozott számítóeszköz-koncepció (Neumann-elvű gép). Részei: a memória, amelyben egyrészt számjegyekkel (digitálisan) ábrázolt adatokat lehet tárolni, másrészt végrehajtandó utasítássorozatok kódját; a feldolgozó egység (processzor), amely értelmezi az utasításkódokat és műveleteket hajt végre az adatokon; a perifériák, amelyek a külvilágba továbbítják az eredményeket, és beemelő adatokat kapnak onnan.

szegmens (segment): A memóriának valamilyen célra kijelölt része. Maximális mérete 64 kb-ot lehet; alkalmazásukat az Intel processzorok — 80286, 80386, 80486 — belső felépítése teszi szükségessé.

szektor: A floppy-lemez, illetve a harddisk *sávjai* részekre, szektorokra — geometriailag körcikkekre — vannak osztva. A szektorok adattároló kapacitása egységes: 512 bájt. (Illetve ennél na-

gyobbak, mert azonosító információkat is tartalmaznak, ez azonban a felhasználó számára nem látható; emiatt a *formázott* lemezek adattároló kapacitása kisebb, mint amennyit a formázatlan lemezekre megadnak, azonban formázás nélkül a lemezek nem használhatók.)

szekvenciális hozzáférés: Lásd adatállomány.

szelekció: Lásd adatbázis.

szelektív help: Lásd help.

szemafor (semaphore): Párhuzamosan futó programok vagy programágak szinkronizálási eszköze. Több változata van. Lényege: amikor a program egy kritikus tevékenységbe kezd — amelyet egyszerre csak egy program végezhet —, tilosra állítja a szemafor. Ha más programok is odaérnek és a szemafor tilos, várniuk kell. Amikor a foglalt a kritikus szakaszból kilép, szabadra állítja a szemafor, és a várakozók közül egyet elindít. Az belép, és tilosra állítja a szemafor. Ha nincs várakozó, a szemafor szabad marad.

szemantika: Jelentés. Utasítások, parancsok, programozási nyelvek konstrukcióival kapcsolatban használjuk. Manapság a jelentést többnyire szavakban szokták megadni. *Formálisnak definíciónak* nevezünk egy szemantikai definíciót, ha pontosan megadjuk, hogy ilyen szintaktikai formához milyen jelentés kapcsolódik, és ez a jelentés valamilyen formalizmusban van megadva. A formalizmus lehet valamilyen ismert matematikai apparátus (formális logika, lambda kalkulus stb.), valamilyen programozási nyelv vagy valamilyen, kifejezetten erre a célra kifejlesztett jelölésrendszer. Tulajdonképpen a *fordítóprogram* is egy formális definíció, csak nem embernek, hanem számítógépnek szól.

szerkesztőprogram, összeszerkesztő program: Lásd linker, linkage editor.

szerver (server): Kiszolgáló gép. Olyan gép, amelynek feladata, hogy más gépeket, *munkaállomásokat, terminálokat* kiszolgáljon. A *fájlszerver* olyan gép, amely központilag tartja nyilván a fájlokat, lehetőséget nyújt arra, hogy a munkahelyek használhassák ezeket, ugyanakkor biztosítja az egyidejű felhasználásból származó hibák elkerülését. A *hálózati szerver* az adatok továbbításának szervezésével foglalkozik. (*LAN-okban* a két szerepet gyakran egy gép látja el.)

szimplex átvitel: Lásd soros adatátvitel.

szimuláció, szimulátor, szimulációs nyelv: Valamilyen rendszer, berendezés vagy folyamat dinamikus (időbeni) működésének modellezése, leutánczása. Célja lehet annak vizsgálata, hogy a matematikai modell mennyiben felel meg a valóságnak. Ha biztosak vagyunk benne, hogy a modell jó, olcsón kipróbálható, miként fog a berendezés viselkedni. Tehát ez egy gazdaságos módszer berendezések kezelésének betanítására. (De lehet a cél játék, szórakozás is, ettől még a modell lehet komoly.) Alapjában véve két különböző formája van. Az egyik, amikor megfelelő időközönként előállítjuk a rendszer állapotait a rendszer előrehaladásának és a külső behatásoknak megfelelően. A másik modellben csak akkor állítunk elő új állapotot, amikor azt valamilyen külső esemény megváltoztatja.

A különböző szimulációs feladatokra igen különböző szimulációs nyelveket hoztak létre, közös bennük, hogy időpontokat és időtartamokat kezelnek (nem a konkrét idő szerint). Ismertebb nyelvek a *SIMULA*, *SIMSCRIPT*, *GPSS*, *T-PROLOG* stb. Újabban az *objektumorientált* nyelveket használják szimuláci-

óra. ➤ **OOP**, eseményvezérelt programozás.

szinkron (adat)átvitel: Lásd soros adatátvitel.

szintaxis: Azon szabályok összessége, amelyek alapján egy formailag helyes (*hiba nélkül lefordítható*) programot lehet írni.

Manapság a szintaxist részben vagy egészen formálisan szokták megadni. Erre a célra szolgál a *BNF* (Backus—Naur formalizmus), amely matematikailag a Chomsky-féle ún. CF (context free) nyelvtannal azonos (*metanyelvnek* is nevezik). Illetve a teljes és pontos formális definícióra ennek továbbfejlesztését, az *attribútum-nyelvtanokat* választják eszközül.

Szintaktikai analizáló (*syntax analyzer*) programnak, más néven *parser*-nek nevezik azt programot, amely egy nyelv programjainak formai helyességét ellenőrzi, és a program szövegét egy — a kapcsolatokat ábrázoló — *gráfba* transzformálja.

Ha adva van a nyelv szintaxisának teljes *formális definíciója*, abból egy metanyelvi *fordítóprogram* segítségével előállítható a nyelv parserje (és ebből további munkával a nyelv fordítóprogramja).

szkenkód: Lásd scan kód.

szkenner (scanner): Képi információk számítógépes bevitelére szolgáló eszköz. A képet megvilágítva azt pontokra bontja, így *bit-map* formában tárolható. A szkennerek felbontását *dpi-ben* adják meg; lehetnek egyszínű, megadott számú árnyalatot megkülönböztető, de színes típusok is. A kézi szkennerek kisebb, az asztali szkennerek nagyobb, általában A4-es méretű képeket olvasnak be.

szoftver (software): A számítógép mindazon tudása, amely nincs eleve be-

leépítve, azaz a beíráható programok összessége. ➤ hardver, firmware.

szótárfájl: Lásd ISIS.

szöveges adatbázis: Lásd adatbázis.

szövegmakró (text macro): Lásd makró.

szövegszerkesztő (program) (editor, text editor): Szöveges állományok írására, javítására alkalmas program. Két fő típusuk van: a programeditorok, amelyek programok forrásszövegének szerkesztésére szolgálnak, és a valódi, *természetes nyelvű* szövegek gondozására készült programok, amelyek igen közel állnak a *kiadványszerkesztő* programokhoz. (A kiadványszerkesztők általában nem ASCII kódú szövegfájlokat hoznak létre.)

Minden editor alkalmas arra, hogy vele beírjunk (beszúrjunk), kitöröljünk, átírjunk karaktereket, nagyobb szövegrészeket; kisebb-nagyobb szövegrészeket áthelyezzünk az állományon belül; más (esetleg csak rész)fájlokat csatoljunk az éppen szerkesztett anyaghoz, vagy ennek egy részét valamilyen másik fájlba tegyük ki. Szövegrészeket minta alapján visszakereshetünk, kicserélhetünk.

A *programeditorok* általában segítik, hogy a programot a *blokkstruktúrának* megfelelő bekezdésekkel írjuk. Az egy programnyelvhez kötött editorok a programot szép külalakra hozzák, segítik a *kulcsszavak* beírását, esetleg még bizonyos szintaktikai ellenőrzésekre is képesek.

Az igazi szövegeket szerkesztő editor (*word processor*) a szöveget centírozza (középre helyezi), a bal vagy jobb oldali margóhoz vagy mindkettőhöz illeszti, a bekezdéseket javítás után újraformázza. Az igényesebb editorok beépített helyesírás-ellenőrző, illetve automatikus elválasztó programot is tartalmaznak, bár ezek erősen nyelvfüggő feladatok. Le-

hetnek bennük olyan lehetőségek, mint a személyre szóló körlevelek előállítására címlista alapján és hasonlóknak.

A különböző *betűtípusok* és méretek alkalmazása a szövegben, a többhasábos tördelés, az ábrák beillesztése, színek kezelése stb. már a kiadványszerkesztés körébe tartozik.

Persze nem lehet éles határvonalat vonni, hogy egy adott termék program- vagy szövegszerkesztő-e inkább, avagy esetleg kiadványszerkesztő. (Egyértelműen megkülönböztetendők azonban a *linker, linkage editor* kategóriájú rendszerprogramtól, amelyet szintén hívnak szerkesztőprogramnak is.)

szubrutin, alprogram, eljárás, függvény, művelet: A programok kisebb, logikailag jól elkülöníthető lépésekből épülnek fel. Az is gyakran előfordul, hogy egy ilyen lépés kis eltérésekkel többször is szerepel a programban. A program lerövidítéséért és a jobb áttekinthetőség kedvéért ezeket a részeket kiemeljük, nevet adunk nekik, és a programban pedig — ahol ezekre a tevékenységekre szükség van — nevükkel hivatkozunk rájuk. A szubrutin tehát két elemből áll: a *szubrutindeklaráció* során a program szövegében jól elkülönítve megadjuk a szubrutin nevét és formális *paramétereit*, amelyekből a végrehajtás függ, valamint azt az utasítássorozatot, amelyet végre kell ilyenkor hajtani, a *szubrutin törzsét*. A program törzsében (vagy egy másik szubrutinban) pedig meghívjuk a szubrutint — akkor, amikor alkalmazni kell. Ez a másik elem, a *szubrutinhívás*, ami a szubrutin nevének és az aktuális paramétereknek a megadásából áll.

zárt szubrutin: Más néven *eljárás, (procedúra, procedure)* — amikor a szubrutintörzs egyetlen példányban fordul le, a hívások helyére pedig csak egy

hívási szekvencia fordul, amely a következő lépésekből áll: a paraméterek értékének vagy címének felrakása a *stackre* (különböző nyelvekben különböző szabályok vannak arra, mikor kell értéket és mikor címet átadni); a soron következő utasítás címének feltétele a *stackre*, majd ugrás a szubrutin törzsére. (A szubrutin törzse a paramétereket a *stackből* veszi, majd végül a *stack* tetején levő címre ugrik vissza, és a paramétereket is eltávolítja a *stackről*.)

nyílt szubrutin: Ezt *makrónak* is nevezzük. A szubrutin törzse minden hívás helyére befordul, természetesen úgy, hogy a formális paraméterek helyére az aktuálisak kerülnek. Néhány nyelvben szabályozni lehet, hogy melyik hívás legyen nyílt és melyik zárt.

rekurzív szubrutin: Egy olyan eljárás, amely a törzsében önmagát meghívja. A rekurzív szubrutint természetesen csak zárt szubrutinként lehet megvalósítani.

függvény: Olyan szubrutint, amelynek minden paramétere input paraméter, és egyetlen értéket ad vissza. Megvalósítás szempontjából a függvény egy olyan szubrutin, amelynek első paramétere a *visszatérő érték*, egy névtelen output paraméter, amelyet visszatérés után a *stacken* hagy. A függvény bemenő paramétereit *argumentumoknak* is nevezzük. Míg a szubrutinhívást utasításként lehet alkalmazni, a függvényhívás értéket kiszámító kifejezésekben szerepelhet.

művelet: A nyelvek beépített függvényei között van néhány egy- és kétargumentumú függvény, amelyeket nem névvel, hanem műveleti jelekkel (+, —, *, /, & stb.) jelölünk. Ezeket más néven *operátoroknak* is nevezzük, argumentumaikat pedig *operandusoknak*.

Az olyan egyargumentumú műveletet, ahol a műveleti jelet az operandus elé írjuk, *prefix operátornak* nevezzük, az olyanokat pedig, ahol a műveleti jeletet a két operandus közé írjuk, *infix operátornak* nevezzük.

Az operátorszimbólumok általában *túlterheltek (overloaded)*, azaz egy jel többféle műveletet is jelölhet, és hogy milyen függvényt hajtunk végre, az operandusok típusától függ. (Például a + jel jelölheti egész számok, valós számok, vektorok vagy mátrixok összeadását.) Ezt a jelenséget *polimorfizmusnak* is nevezik.

Néhány nyelv lehetőséget ad rá, hogy a műveleti jelekhez a felhasználó rendeljen függvényeket.

A függvényhívásokat egymásba lehet ágyazni, azaz a függvényhívás argumentumában lehet egy másik függvényhívás. A normális kiszámítási mód ilyenkor (hacsak a nyelv mást nem ír elő): először a belső függvényt számítjuk ki, azután az így kapott értékre a külsőt. Operátorok esetén nem látszik, mi a külső, mi a belső. Itt a kiszámítás sorrendjét az operátorok *prioritása* szabja meg.

T

tablet: Kisebb *digitalizáló tábla*.

target code: Lásd tárgykód.

task (taszk): Feladat.

táblázatkezelő (spreadsheet): Egy táblázat sorokból és oszlopokból áll, egy cellába vagy egy szöveg, vagy egy szám, vagy egy képlet kerülhet. A képlet más cellákra hivatkozik. Amikor a megfelelő cellák kitöltődtek, a képlettel definiált cellákba is beíródik az eredmény. Az adatok megváltoztatásakor a táblázat újraszámolódik. A háromdimenziós esetben a táblázatok több lapon vannak, és a lapok között is lehet hivatkozás.

A táblázatkezelő gyakran kapcsolódik össze *üzleti grafikával*, ilyenkor a táblázatban szereplő bizonyos számsorokat különböző grafikonokkal lehet ábrázolni. Sok program arra is lehetőséget nyújt, hogy a táblázatok alapján olyan számításokat is elvégezzünk, amelyeknek az eredménye nem kerül a táblázatba.

A táblázat többnyire fejlécből, tartalom sorokból és összesen sorból épül fel. A tartalom sorok tulajdonképpen azonos szerkezetű rekordok, így adatbázisba menthetők, illetve onnan olvashatók be. A korszerű táblázatkezelő programok számos adatbázissal biztosítanak kapcsolatot. A táblázathoz többnyire szöveget is kapcsolunk, ezért lehet beépített *szövegszerkesztő* is a programban, sőt a grafikonok kiegészítéséhez *grafikus szerkesztő* is.

tárgykód (object code, target code): Kód, amelyre a programozási nyelveket fordítják. A *fordítóprogramok* nem készítene azonnal végrehajtható kódot, hanem olyat, amely más modulokból, más nyelvekből származó kódokkal

összeköthető. Ez egyrészt lehetővé teszi, hogy hozzákapcsoljuk a program többi modulját, másrészt lehetővé teszi, hogy több különböző programba illesszük bele. Így arra is szükség van, hogy a tárgykód a memóriában különböző helyekre kerülhessen.

társprocesszor: Lásd koprocesszor.

távolsági hálózat (WAN = Wide Area Network): Egymástól nagy távolságra, akár más kontinensen lévő számítógépek összekapcsolását biztosító rendszer. A helyi és a távolsági hálózatok között különbség van az átviteli távolságon kívül más jellemzőkben is, például az átviteli hibák számában és a sebességben is. A *helyi hálózatoknál* ez több, általában néhány száz Mbit/sec, a távolsági hálózatoknál napjainkban tipikusan 64 Kbit/sec.

Vannak műholdas hálózatok. A földi hálózatoknál az átviteli közeg ma még általában *modemes* telefonhálózat, ami jelenleg általában 9600 *baudra* korlátozza a sebességet. A telefonvonal lehet kapcsolt vagy erre a célra bérelt.

A *vonalkapcsolt hálózatban (line switching network)* az adó és a vevő között (közbülső állomásokon keresztül) egy útvonal épül ki, pont-pont összeköttetés jön létre. Ez a kapcsolat a kommunikáció teljes ideje alatt fennáll, és lefoglalja a benne résztvevő vonalakat.

A *csomagkapcsolt hálózatokban (packet switching network)* a küldendő adatokat kisebb csomagokra osztják, és a csomagokat egyenként továbbítják. Ilyenkor nem épül ki kapcsolat az adó és vevő között, adáskor egyszerűen a hálózatra bízva a küldő az adatok továbbítását. A hálózat a csomagokat esetleg különböző útvonalakon, vonalainak optimális kihasználásával próbálja továbbítani a címzetthez.

A csomagkapcsolt hálózatok két gyakran használt eljárási *protokollja* (kapcsolatkiépítés, kommunikáció, kapcsolatbontás stb.) a *CCITT X.25* szabványa és az USA hadügyminisztériuma által kidolgozott TCP/IP (Transmission Control Protocol/Internet Protocol).

► Internet.

telefaxkártya: Olyan *adapter*, amelynek segítségével a telefonvonalon fax küldhető és vehető, közvetlenül a számítógépből. A küldendő faxot egy *szövegszerkesztővel* lehet elkészíteni, és az kinyomtatás nélkül elküldhető. A vétel természetesen csak a bekapcsolt számítógéppel lehetséges, és ilyenkor a vett faxokat automatikusan a harddiszken vagy floppy-n tárolják.

teletexkártya: Olyan *adapter*, amellyel a teletex adás lapjait lehet a képernyőn megjeleníteni, a harddiszken vagy floppy-n tárolni.

teletext: A *CCITT* által tervezett nyilvános számítógép-hálózat, amely a hagyományos telexet váltaná fel.

természetes nyelv: Az emberi beszéd és annak írott formái. Kutatása a *mesterséges intelligencia* problematikájában is az egyik ősi feladat, amelynek két fő célkitűzése a gépi fordítás (MT = Machine Translation), valamint a természetes nyelvű parancsok megértése és végrehajtása a számítógép által. Mindkettőben mérsékelt sikerek vannak.

► mesterséges nyelv.

terminál: Számítógéphez kapcsolódó, legalább egy képernyőből és *klaviatúrából* álló munkahely. A *buta (dumb) terminál* csak e kettőből áll. Karakter-sorozatokat (és eszköz-szekvenciákat) küld be és ki, minden más munkát a számítógép végez el. Az *intelligens (intelligent/smart) terminál* saját processzorral és saját memóriával is rendelkezik, a munka egy részét maga a ter-

minál végzi el. (Például annyira magas szintű grafikus parancsokat kap, hogy rajzolj egy valamilyen görbét, illeszd fölé a következő szöveget stb.; és a képet maga állítja elő.)

Egy számítógépes rendszer csak bizonyos terminálokat tud kezelni. A *terminálemulátor* PC-n vagy *munkaállomáson* futó program, amely a számítógéphez csatlakozva egy terminál viselkedését utánozza. Persze a PC-n vagy a munkaállomáson más programok is futhatnak, akár a terminálemulátorral egy időben is.

terminálkoncentrátor: Amennyiben minden felhasználói terminál külön vezetékkel csatlakozna a számítógéphez, a kábelezés nagyon drága lenne. Esetleg nincs is a gépen elég *soros port*. Ezért terminálkoncentrátorhoz csatlakoznak az egymáshoz közeli, mondjuk egy szobában lévő terminálok, és csak a koncentrátor kapcsolódik a számítógéphez.

A koncentrátor *időosztásosan* engedélyezi az egyes terminálok és a gép kommunikációját. Ez általában megfelelő, mivel a terminálok terhelése nagyon változó. Előfordulhat, hogy minden terminál egyszerre akar a géppel beszélni, ebben az esetben a terminálkoncentrátor a beépített átmeneti tárolójába tölti az érkező információt, és késlelteti a küldést a kisebb terhelésű periódusokra. A koncentrátor memóriáját és őt a számítógéppel összekötő kábel kapacitását a várható forgalomhoz kell méretezni, mert ha a memória kevésnek bizonyul, adatok veszhetnek el.

tesztadat-generátor: Lásd generálás.

TEX: Az egyik első *DTP* program, Donald Knuth közreműködésével készült. Erőssége a matematikai képletek szerkesztése. Műszaki, matematikai cikkek szedésére alkalmazzák egyetemi környezetben. Mivel szinte minden ope-

rációs rendszeren létezik és egységes, számos folyóirat elfogadja beküldési formátumként.

Hátránya, hogy nem egy *interaktív* program — tehát szerkesztés közben nem látható a szöveg várható alakja —, hanem egy nyelv, amelyben vezérlőutasítások és -makrók váltakoznak a beírt szövegekkel. Viszont számos *fontja* (magyar, cirill, görög, japán) van, és a *METAFONT* programmal újabbak is készíthetők. Tipográfiaileg magas színvonalú nyomtatást biztosít legkülönbélebb berendezéseken: a *mátrixnyomtatótól* kezdve a legjobb felbontású nyomdai levilágító berendezésekig.

text editor: Lásd szövegszerkesztő.

Thunder Board: *Hangkártya*, erősítő és hangszóró, mikrofonbemenettel. Digitalizált hang ki- és bemenete, 11 hangcsatornája, programjai vannak. A digitalizált hangfájlokat tömöríti, és a hangokat a tömörített fájlból is visszajátssza. Viszonylag olcsó.

time sharing, time sliceing (időosztás, időszeletelés): Lásd ütemezés.

timer: Időzítési feladatok ellátására alkalmas *integrált áramkör*; sok esetben a felhasználásával készült, bonyolultabb célokra alkalmas *adaptert* is így nevezik. Feladata megfelelő időzítésű jelek előállítására a számítógépen belül — részben különböző vezérlési célokra, részben a felhasználó számára az óra (és ezen keresztül a naptár) funkció megvalósításához. Az IBM PC típusú gépekbe az Intel 8253-as vagy 8254-es időzítő áramkörét építik.

tintasugaras nyomtató: Lásd nyomtató.

token: Vezérlőjel. Lásd helyi hálózat.

Token Ring: Az IBM 1985-ben bejelentett, vezérlőjelesgyűrű-protokollal működő helyi hálózata. Az első változat sebessége 4 Mbit/sec volt, ezt később 16

Mbit/sec-ra növelték. Az átviteli közeg sodrott érpár vagy fénykábel. Az *IEEE 802.5* (token-ring hálózati) szabvány koncepcióját testesíti meg. ➤ helyi hálózat.

toolkit: Szerszámkészlet. Programok, illetve különféle programozási nyelvekből hívható *függvények* gyűjteménye, amelyek olyan alapvető dolgokat oldanak meg, mint a képernyő, az *egér*, a memória, a könyvtárak stb. kezelése.

top down (felülről-lefelé): Programtervezési és programozási elv; a nagy feladatok megoldásának szokásos útja. A tervezés úgynevezett absztrakciós szintekben gondolkozik: első szinten megállapítjuk, hogy mi a program feladata, milyen bemenő adatokat kap, milyen kimenő adatokat szolgáltat. A következő szinten a feladatot részfeladatokra bontjuk, meghatározzuk ezek ki- és bemenő adatait és a részfeladatok egymáshoz kapcsolódását. Majd a részfeladatokra ismételjük az eljárást, amíg a kivitelezés szempontjából áttekinthető méretű feladatokat kapunk.

Nem árt kiegészíteni egy olyan vizsgálattal, hogy a részfeladatok részfeladatai között vannak-e közösek, illetve közös eszközkészlettel megoldhatók, továbbá hogy milyen kész eszközeink vannak.

A felülről-lefele programozásnál először a felső modulokat/ függvényeket kell megírni, amelyek az alsóbb szintű modulokat felhasználják és összekötik. Előnye, hogy mivel az alsóbb szintű modulok egymástól függetlenek, ezeket párhuzamosan lehet megírni, és elkészültük után rögtön ellenőrizhető a programba való beilleszthetőségük. Lehetőségünk van arra is, hogy először a program legérdekesebb része készüljön el, a többi pedig később. Hátránya a módszernek, hogy az egész program viszont

elég hosszú ideig nem próbálható ki, amíg az alsó modulok el nem készültek.

A program megírható alulról-fölfelé is. Ilyenkor az alsó szintű modulokat írják meg először. Az egyes elkészült modulokat hamarabb lehet tesztelni, de a modulok megfelelése csak később derül ki. ➤ stub.

tömb típus (array): Azonos típusú elemek sorozata (ezért homogén adatszerkezetnek is nevezik). A nyelvek megoszlának aszerint, hogy a különböző hosszúságú sorozatokat különböző típusúnak tekintik-e, vagy pedig egy fő típus különböző altípusainak. Az utóbbi fel fogás a korszerű; vannak olyan műveletek, ahol csak a típusellenőrzés van előírva, és vannak olyanok, ahol az altípust is ellenőrizni kell. Például a modern nyelvekben paraméterként bármilyen hosszú tömböt átadhatunk, értékül azonban csak ugyanolyan hosszú (esetleg rövidebb) tömböt adhatunk, mint a változó *deklarált* hossza.

Minden tömbre alkalmazható művelet az *indexelés*. Ekkor egy indexérték segítségével kiválasztjuk a sorozat egy elemét. Azokban a nyelvekben, ahol az index nem csak egy pozitív (vagy nem negatív) szám lehet, ott a tömbindex legkisebb és legnagyobb értékét kell megadni a tömb létrehozásakor. Tömbműveletek: *értékadás*, egyenlőség (esetleg a kisebb, nagyobb is, ha az elemeken értelmezve van). Attribútumai a legnagyobb és legkisebb index, valamint a tömb hossza. ➤ adattípus.

töréspont (break point): Lásd debugging.

trace (nyomkövetés): Lásd debugging.

track: Lásd sáv.

track ball: Lásd pozicionáló gömb.

translator: Lásd fordítóprogram.

transputer: Az Inmos cég által *multiprocesszoros rendszerek* számára kifejlesztett mikroprocesszor. Jellegzetessége, hogy a szokásos processzoregységeken kívül négy darab speciális, nagy sebességű, *soros kommunikációs vonalat* tartalmaz a CPU-k egymáshoz kapcsolódására. Ezeket keresztül — néhány kisegítő áramkör alkalmazásával — egy *párhuzamos működésű* processzorhálózat hozható létre, igen nagy teljesítménnyel.

Az első, 1985-ben megjelent típus — a T414 — a maga idejében a leggyorsabb 32 bites CPU volt. Az 1987-ben kibocsátott T800 már lebegőpontos műveleteket végrehajtó egységet (*FPU*) is tartalmazott. Az Inmos a transzputerrel együtt létrehozta az Occam nyelvet, amelyen a transzputerek párhuzamos működése programozható. Már jelentek a C nyelv párhuzamos programozásra alkalmas változatai is. ➤ párhuzamos feldolgozás.

trójai program: Az ún. trójai programok mások, mint aminek mutatják magukat. Például: miközben a monitoron egy repülő sas animációját látjuk, tönkreteszi a merevlemezt (EAGLE TROJAN). Hasonló botrányt kavaró, büntetőjogi eset volt a Trojan AIDS Information DISK esete, amikor az AIDS információs rendszernek álcázott programcsomag *átkódolta* és használhatatlanná tette a merevlemezt, illetve ennek visszaállításának lehetőségével pénzt zsarolt ki a felhasználóktól. A trójai program a *vírustechnológia* fogásait alkalmazza. A *másolásvédelmek* túlnyomó része trójai programként viselkedik, amikor büntet.

A trójai programot a vírustól az különbözteti meg, hogy nem szaporodik, nem reprodukálja önmagát.

True Type betűkészlet (font): Lásd Windows.

truncating (letörés, csonkolás): A számítógépben a számokat meghatározott számú bináris (vagy *BCD* aritmetika esetén decimális) számjeggyel ábrázolják. Bizonyos műveleteknél az eredmény több jegyű lesz, mint amennyit el lehet tárolni, ilyenkor a törtrészt a megfelelő számú jegyre levágjuk (*truncating*) vagy fel-/lekerekítjük (*rounding*). Géptől függ, hogy mi történik. Letörésnél a pontatlan ábrázolásból eredő hiba nagyobb valószínűséggel halmozódik fel, mint a kerekítésnél.

TSR (Termianate and Stay Resident): Lásd rezidens program.

tudásalapú rendszer: Lásd szakértő rendszer.

túlcsordulás (overflow): Minden numerikus típushoz tartozik egy legnagyobb és legkisebb szám; ha a művelet eredménye ezt túllépi, túlcsordulás következik be. Szinte minden hardverben adott a lehetőség, hogy ilyenkor *megszakítás* következzen be, illetve hogy ezt letiltssuk.

Egy sor nyelv és sok *implementáció* a túlcsordulást figyelmen kívül hagyja. Másokban a *fordítóprogram opciója*, hogy történjen-e ilyenkor hibajelzés vagy sem. Vannak olyan programok, amelyek futásuk közben figyelik a túlcsordulást, és ha bekövetkezik, egy — a kivételes helyzetet (*exception*) kezelő — programrészre adják a vezérlést.

TXT: Szövegfájlok szokásos kiterjesztése.

U

UART (Universal Asynchronous Receiver-Transmitter): Univerzális adó-vevő áramkör. A *soros, aszinkron adatátvitel* vezérlésére alkalmas *integrált áramkör* általános neve. Az IBM PC típusú gépekben az Intel 8250-est vagy ezzel kompatibilis áramkört alkalmaznak.

UMB (Upper Memory Block): A PC-kben a 640 és 1024K közötti címekeket eredetileg a *ROM* és a *képernyő-RAM* számára tartották fenn. Mivel ezek nem foglalják el a teljes tartományt, megfelelő *alaplapon az extended memória* egy-két — 64 Kbájtos — *szegmensét* képezik le a szabadon maradt részre. Ezek a blokkok így DOS alatt is használhatók.

undelete: Az utolsó törlőutasítás visszavonása. Biztonságosan csak közvetlenül a törlőutasítás után adható ki, de később is lehet próbálkozni vele elveszett fájlok visszanyerése céljából — bizonyos operációs rendszerekben (például a DOS 5.0 változatában) vagy diszkezelő programokban (például *Norton Utilities*).

underflow (alulcsordulás): Lásd valós típus.

undo: A utolsó utasítás visszavonása. **Redo**: Az undo visszavonása, azaz az előző változtatást mégis meg kell tenni.

unformat: Amikor egy diszket újraformázunk, tartalma elérhetetlenné válik. Az unformat megpróbálja helyreállítani az eredeti tartalmat. Ez úgy lehetséges, hogy a gépnek különféle táblázatai vannak, a *safe (biztonsági) formázás* bizonyos információkat elment a diszkre, és a *formázás* kapcsán csak olyan keveset töröl, amennyit muszáj.

Unicode: Lásd karakterkódok.

uninstall: Kiiktatás. Lásd installálás.

unio: Lásd halmaz típus.

UNIX: A Bell Laboratóriumban a 70-es évek elején kifejlesztett — többfelhasználós, *multitasking* — operációs rendszer. Születése idején a nagyszámítógépek operációs rendszereivel konkurált, jelenleg a PC-k primitívebb operációs rendszereit szorítja ki. Számos nagyjából kompatibilis változata van (AT&T-, SCO-, SINIX, ULTRIX, XENIX stb.). Munkaállomások operációs rendszereként képes együttműködni a nagygépekkel.

A UNIX alatt futó *X-WINDOWS* korszerű, grafikus, *ikonos vezérlésű* felhasználói felületet biztosít a felhasználóknak.

unlock (felszabadítás): A *lock* megszüntetése. Lásd lock.

update: Aktualizálás. Valamilyen információ naprakész (up to date) állapotba hozatala. Az adatállományban bekövetkezett változások bevitelekor vagy a program legújabb változatának munkába állításakor éppen ez történik.

upgrade: Felfejlesztés. Lásd változat.

upward compatibility: Lásd változat.

USART (Universal Synchronous-Asynchronous Receiver-Transmitter): Univerzális *szinkron-aszinkron* adó-vevő áramkör. Olyan *integrált áramkör* általános neve, amellyel a *soros adatátvitel* mind szinkron, mind aszinkron módon végrehajtható, a programozott beállításától függő paraméterekkel. Az IBM PC típusú gépek megfelelő *adapterein* az Intel 8251-es vagy ezzel kompatibilis típusú USART-ot használnak.

user: Lásd felhasználó.

user defined: Lásd felhasználó által definiált.

utasítás (instruction): A számítógépnek előírjuk, hogy mit hajtson végre. Vannak olyan utasítások, amelyeknek önmagukban nem sok értelmük van, például a *BASIC* nyelv FOR vagy IF utasításai, ezek csak más utasításokkal együtt, programba szervezve alkalmazhatók. Vannak olyan utasítások, amelyeket nemigen szoktak más utasításokkal kombinálni, csak azonnal végrehajtani, ilyen például a *BASIC* RENUMBER utasítása.

parancs (command): Egy programban azokat az utasításokat, amelyeket a program azonnal végrehajt, parancsoknak nevezzük. Van egy sor olyan *interaktív* program, amelyek alapvetően csak parancsokkal dolgoznak. Ilyen például az MS-DOS parancsnyelvét értelmező *COMMAND.COM*, a legtöbb *szövegszerkesztő* program, *kalkulátor* stb. Né-

ha azonban kiderül, hogy itt is jó lenne hosszabb utasítássorozatokot kiadni. Erre több út kínálkozik:

— Az egyik ilyen a *pipe, pipe-line*. A parancs1|parancs2|... hatására végrehajtódik az első parancs, ennek eredménye (*stdout*) nem megy a képernyőre, hanem a parancs2 inputja (*stdin*) lesz stb. ► standard fájlok.

— A *billentyűmakróval* egy gombhoz több parancsot lehet hozzárendelni.

— A parancsok készletét kibővíthetjük *ciklus-, elágazó* és egyéb utasításokkal, így egy olyan *parancsnyelv* jön létre, amellyel meglehetősen összetett parancsokat lehet egy parancsfájlba leírni.

utasításkészlet: Egy adott típusú processzorral végrehajtható elemi utasítások. Minden magasabb szintű műveletet ezekből az utasításokból kell felépíteni. ► RISC, CISC.

Ü

ütemezés (scheduling) : *Multitasking* környezetben a végrehajtandó feladatokat a számítógép különböző stratégiák szerint hajthatja végre. A programnak vagy feladatnak 3 állapota lehet: éppen fut; nem fut, de tudna futni; vár valamire (felhasználói inputra, másik feladat *üzenetére*, valamilyen erőforrásra (memória, diszk, nyomtató, ...), amelyet egy másik program foglal le.)

Az ütemezés lehet *non-preemptív*: ilyenkor a program maga adja át a vezérlést, amikor várni kényszerül; és *preemptív*, amikor megadott idő után vagy nagyobb *prioritású* feladat beérkezéssel a számítógép el is veheti tőle a vezérlést.

Az ütemezés alapulhat *prioritáson*; „round robin” elven (a futóképes a sor végére áll, és mindig az első fut); *időosztáson (time sharing, time sliceing)* (a program csak meghatározott ideig futhat, utána ütemezés jön). Az ütemezés rendszerint ezen elvek valamilyen bonyolultabb kombinációja.

üzenet (message): 1. A számítógép és a *perifériák*, valamint a nagyobb számítógépes rendszerek (például a hálózatok) elemei egymással üzeneteket váltanak, miközben átviszik az adatokat. Az üzenet egyrészt a valódi adatokból, másrészt az üzenet kezelését előíró adatokból áll.

2. A programok, elsősorban az *interaktív* programok munkájuk közben szöveges üzeneteket küldenek a felhasználónak. Az üzenet lehet hibajelzés, tájékoztatás, kérdés stb.

A fejlettebb programok az üzeneteiket egy üzenetfájlból veszik, ami könnyen módosítható. Még fejlettebb formája a *képernyő-generátor* és az általa generált fájlok használata. ► hiba.

3. Lásd OOP.

üzenettér: Lásd OOP.

üzleti grafika (business graphic): Különféle grafikonok (körcikkés, oszlopos, vonalas, térhatású stb.) készítésére és feliratozására szolgáló program.

Az adatokat a program adatbázisokból vagy *táblázatkezelőktől* veheti át.
► Lotus 1-2-3.

V

validáció: Érvényesítés. Az USA hadügyminisztériuma által alkalmazott hitelesítő eljárás. Ennek alapján döntenek el, hogy a nyelv *fordítóprogramja* megfelel-e a katonai szabványokban lefektetetteknek. Ilyenkor egy olyan, több száz programból álló programcsomag készül, amely teszteli, hogy a programok eleget tesznek-e a nyelv előírásainak, különös tekintettel a nehezen megvalósítható vonásokra, valamint az egymást keresztező vonások együttes működésére.

► Ada, COBOL.

valós idejű: Lásd real-time.

valós típus (real): A gép hardverje által meghatározott, illetve szoftverben megvalósított lebegőpontos adattípus.

A modern nyelvekben 'long', esetleg 'short', esetleg még több különböző hosszúságú változatot is megengednek. A hardver által adott lebegőpontos sokszorosan hatékonyabb a szoftverben megvalósítottnál. Bizonyos programok (például a szövegfeldolgozás) a lebegőpontosot egyáltalán nem igénylik, mások (például a grafika) erőteljesen kihasználják.

A lebegőpontos szám mantisszából és karakterisztikából áll. A *mantissza* egy valahány jegyű előjeles szám, ami meg van szorozva egy másik szám pozitív vagy negatív hatványával, a hatványkitevőt a *karakterisztika* adja meg. A lebegőpontos szám tág határok között, de a szám nagyságától függő tűréssel (közelítőleg) ábrázolja a számokat. Így előfordulhat, hogy ha egy nagy számhoz egy túl kicsit adunk hozzá, a szám változatlan marad. Az is megeshet, hogy két szám kivonása vagy osztása után eredményül olyan kis számot kapunk, amely már nem ábrázolható, ezt *alul-*

csordulásnak nevezzük. A legtöbb *implementáció* (de nem mind) az ilyen számot nullává teszi.

A lebegőpontos számok körében elvégezhető a négy alapművelet, viszont a hatványozás mindig közelítő képlettel történik. A kisebb és nagyobb reláció értelmes, de a programozás során az egyenlőségtől óvakodjunk, mert különböző gépeken és implementációkon — a közelítő ábrázolás miatt — eltérő eredményeket kaphatunk. Van legnagyobb és legkisebb szám, ezek túllépése *túlcsordulást* okoz. ► adattípus.

valós üzemmód (real mode): A 80x86-os processzorok *default* üzemmódja, amikor a 8086-os processzort *emulálja*. Kompatibilis üzemmódnak is nevezik, mert ebben az üzemmódban a korábbi DOS-os programok futtatására szolgál.

VAR (Value Added Reseller): Olyan kereskedő, viszonteladó, aki valamilyen módon (például szoftverfejlesztéssel) növeli az általa forgalmazott termék értékét.

variable: (változó) Lásd deklaráció.

variable record (változó rekord): Lásd rekord típus.

változat, verzió (version, release): Amíg egy program él, addig változik, fejlődik. Az előző változat hibáját kijavítják, és új képességeket/lehetőségeket (features) adnak hozzá. A hivatalosan kiadott friss változatokat egy egész és törtrészből álló számmal szokták jelölni. (Például 3.2 — ejtsd: három pont ket-tő/kettes.) Amikor az egész rész változik, az arra utal, hogy a programon jelentős változtatásokat hajtottak végre.

Noha minden fejlesztésnél törekszenek arra, hogy az új változat mindent tudjon, amit az előző, vagyis felülről kompatibilis legyen (*upward compatibility*), de ezt nem mindig sikerül elérni.

Ilyenkor a fejlesztő cég részletes utasítást ad ki, hogyan kell a programokat átalakítani, felfejleszteni (*upgrade*).

változó (variable): Lásd deklaráció.

változó rekord (variable record): Lásd rekord típus.

városi hálózat (MAN = Metropolitan Area Network): A *helyi hálózatok* technológiáját használó, de nagyobb — 50-100 km — kiterjedésű hálózat. Gyakorta kisebb helyi hálózatokat kapcsol össze, az igényeknek megfelelően általában fénykábelben át.

vektorgrafika: Lásd grafika.

Ventura Publisher (2.0): A Xerox cég *kiadványszerkesztő* programja. Magyar nyelvű változata is van, teljes magyar ábécével és magyar szóelválasztással, ezért itt igen népszerű. Kiváló *egérvezérelt, menüs* interfésze és *szelektív helpje* van. Sokféle *szövegszerkesztő*, grafikus program, *táblázatkezelő* kimenetét képes fogadni. Számos nyomtatót és levilágítót kezel, ismeri a *PostScript* nyelvet is. Tulajdonságai: színes nyomtatás; színelbontás és speciális effektusok PostScript készülékekre; grafikák kicsinyítése/nagyítása; formátumleírás *stíluslapokkal*; automatikus lapszámozás, fejezet-, ábra- és táblázatszámozás; automatikus tartalomjegyzék-, tárgymutató-, ábrajegyzék-*generálás*.

Kapható hozzá olyan bővítés, amely képes a 640K feletti memória kezelésére. Erre a változatra jellemző még: táblázatszerkesztés üzleti célra; egyenlet-szerkesztés tudományos célra; a kereszt-hivatkozások (ábra-, lapszám) automatikus újragenerálása. Új képessége a *függőleges kiegyenlítés (vertical justification)*: a program széthúzza a szöveget, hogy a rendelkezésére álló helyet kitöltse.

verem: Lásd stack.

version: Lásd változat.

vertical justification: (függőleges kiegyenlítés) Lásd Ventura Publisher.

verzió: Lásd változat.

vezérlés (control): Maga a folyamat, a számítógépben ez is *digitális jelek* átadásával történik.

vezérlésátadó utasítások: A program *specifikációkból, adatdeklarációkból* és végrehajtandó utasításokból áll. A végrehajtható utasítások a leírás sorrendjében kerülnek sorra, amíg egy vezérlés-átadó utasítás nem jön. A vezérlésátadásnak a következő formái vannak.

Elágazó utasítások: Az utasítás többféle lehetőség közül választja ki a folytatást.

'IF' (feltételes) utasítás: Az utána következő utasítás vagy utasítássorozat csak akkor érvényesül, ha a feltétel teljesül. Lehet 'ELSE' ága is, ez akkor hajtódik végre, ha a feltétel nem teljesül. Egyes nyelvekben az 'IF' után lehetnek 'ELSIF' vagy 'ELIF' (az ELSE IF rövidítése) ágak is; amelyekben újabb feltételek állnak; ilyenkor a feltételeket sorban megvizsgáljuk, és az első olyan ág hajtódik végre, amelynek feltétele teljesül; ha egyik sem teljesül, az 'ELSE' ág érvényesül, feltéve, hogy van.

'CASE' utasítás: Olyan elágazó utasítás, amely aszerint választja ki a végrehajtandó utasításokat, hogy egy kifejezés milyen értéket szolgáltat. Ennek is lehet 'ELSE' ága.

Ciklusutasítások: Egy utasítás vagy utasítássorozat ismétlése. Több formája van.

'WHILE' ciklus: Egy utasítássorozatot addig ismétlünk, amíg valamilyen feltétel teljesül. A feltételvizsgálatra az utasítássorozat előtt kerül sor, tehát ha eleve nem teljesült a feltétel, az utasítássorozatot egyszer sem hajtjuk végre.

'REPEAT' ciklus: A feltételvizsgálatra a ciklus végén kerül sor. Az utasítás-

sorozat tehát legalább egyszer végrehajtódik.

'LOOP' ciklus: Az utasítássorozat látszólag feltétel nélkül ismétlődik, de a ciklustörzsben van (legalább egy) feltételes utasítás, amellyel kiugorhatunk ('GOTO', 'BREAK') a ciklusból.

'FOR': A ciklusnak van egy *ciklusváltozója* (*control variable*). Ez a ciklus elején kezdőértéket kap. A ciklusváltozó a ciklustörzs minden lefutása után egy bizonyos értékkel, a lépésközzel nő vagy csökken. A ciklus addig tart, amíg túl nem léptük a megadott végértéket.

A feltétlen (vezérlésátadó) utasítások: Hatásukra a program nem a soron következő utasítással folytatódik.

'BREAK' utasítás: Elhagyja a (legbelső) ciklust, és az ezt követő utasításra lép.

'GOTO' utasítás: Egy címkét kell megadni a 'GOTO' *operandusaként* — erre a címkére ugrik —, és az utána következő utasításnál folytatja a programot. Matematikailag bebizonyítható, hogy a 'GOTO' utasításra nincs szükség, de a tapasztalat azt mutatja, hogy hiánya bizonyos helyzeteknek, különösen a hibáknak a kezelését megnehezíti, áttekinthetetlenné teszi. Ugyanakkor a 'GOTO' korlátlan alkalmazása szintén áttekinthetetlenné teszi a programot. Ezért alkalmazhatósági körét a programozási nyelvek általában megszorítják. Például ciklusból ki lehet ugrani, de szubrutinból nem.

A vezérlésátadó utasítások formája nyelvenként különböző, de amikor általában beszélünk, a fenti típusneveket emlegetjük.

védelem (protection): A programozás során szükségessé válhat, hogy bizonyos adatokat óvjunk véletlen átírások és törlések ellen. Többfelhasználós rendszerek esetén a jogszerű felhasználás és

a szándékos kártevés ellen is védekezni kell. A védelem történhet hardver- vagy szoftvereszközökkel. A floppy-lemez írásengedélyező nyílásának lezárása vagy egy *porta* elhelyezett *hardverkulcs* még akkor is hardvervédelem, ha megfelelő programok is kellenek hozzá. A hozzáférést tiltó *flagek*, *kontrollösszegek*, *védelmi kulcsszavak elhelyezése és ellenőrzése viszont szoftvervédelem*.

Ilyen eszközökkel történik a *vírusok* terjedése elleni védekezés, valamint az *osztott adatbázisok integritásának* ellenőrzése, és hasonló problémák megoldása is.

védett üzemmód (protected mode): A 80x86-os processzoroknak a memóriavédelem kezelésére felkészült, *multi-programozást* támogató üzemmódja. Az OS/2 és a legtöbb UNIX implementáció ebben az üzemmódban dolgozik.

véges automata: A véges automata egy olyan absztrakt gép, amelynek valahány (de nem végtelen sok) állapota van, és különböző jeleket lehet beadni neki. Az állapotok egy részéről azt mondjuk, hogy végállapot. Ha az automata valamilyen adott állapotban van és egy bizonyos bemenő jelet kap, valamilyen meghatározott állapotba kerül, és egy kimenő jelet ad ki.

A *lexikális* ellenőrzést végző automaták speciális (Medvegyev-féle) automaták, amelyeknek nincsenek kimenő jeleik, csak azt jelzik, hogy az automata végállapotban van-e. A lexikális ellenőrzés úgy történik, hogy az automatát elindítjuk a kezdőállapotból, és minden *kulcsszó*, *azonosító*, *szám* stb. végén megnézzük, végállapotban van-e.

végzetes hiba: Lásd hiba.

véletlen, véletlenszám-generátor: Lásd random.

VGA (Video Graphics Array): AZ IBM a PS/2 sorozatban új *képer-*

nyőadapterrel jelent meg, a VGA-t a PS/2 típusú (50-es és magasabb modellszámú) gépek tartalmazzák, eredetileg nem önálló adapterként, hanem az *alaplapon* beépítve.

A megelőző EGA típushoz képest grafikus módban növelték a felbontást 640x480-asra — 16 színnel —, és megjelent az alacsonyabb, 320x200-as felbontás — 256 egyidejű színnel. A *monitorcsatlakozón* pedig a korábbi *digitális jelek* helyett *analóg* módon adja ki a színjeleket, ami elméletileg korlátlaná teszi az alkalmazható színek számát. Ennek köszönhetően az egyidejűleg használható 16, illetve 256 szín tulajdonképpen 262 144 szín közül választható. Más gyártók továbbfejlesztették a VGA-t, és megjelentek a 800x600 sőt 1024x768 képpont felbontású *Super VGA* típusok. Ezek sajnos általában nem kompatibilisek egymással, a nagy felbontások csak azokkal a szoftverekkel használhatók ki, amelyekhez a gyártó *meghajtót* biztosít.

vezérlő (controller): Lásd kontroller.

videó-RAM: Lásd képernyő-memória.

videóadapter: Lásd képernyőadapter.

videókártya: Lásd képernyőadapter.

videómemória: Lásd képernyő-memória.

virtuális: látszólagos, fiktív.

virtuális memóriakezelés: A gép úgy viselkedik, mintha annyi memóriája volna, amennyi a virtuális memóriában látszik, holott ennél kevesebb van. Ezt úgy éri el, hogy a memóriatartalom egy részét diszken tartja, és szükség esetén betölti vagy kírja a fizikai memóriából. Ha a processzor ezt támogatja, a felhasználói program mindebből semmit sem vesz észre. A korábbi nagygépes rendszereknél terjedt el, de a mikropro-

cesszorok fejlődésével a PC-kben is megvalósítható ez a memóriakezelés.

virtuális terminál, fantom terminál: Olyan terminál, amelyhez nem tartozik képernyő és *klaviatúra*, de amire hivatkozva elindíthatunk olyan munkákat, amelyek egyáltalán nem folytatnak a felhasználóval párbeszédet.

virtuális gép: Képzelt gép, amelynek utasításait könnyen meg lehet valósítani különböző valódi gépeken.

virtuális üzemmód (virtual mode): A 80386-os és 80486-os gépek egyik üzemmódja, amelyben képes egyszerre több 8086-os gépet szimulálni.

vírus: Olyan programkód vagy kód-részlet, amely önmagát automatikusan meg többszörözi, és pusztán ezzel vagy más módon is kárt okoz.

A *trójai programtól* és a *másolásvédelemtől* az különbözteti meg, hogy a vírusok legtöbbször valamely terjedési szakasz után kezdik meg romboló tevékenységüket, valamilyen *aktivációs feltétel (trigger)* teljesülésekor.

Általában más programokba épülnek be, amelyeket a szakirodalomban a biológiából vett kifejezéssel vektoroknak neveznek. A vírusok lehetnek károkozók — akár katonai, akár szabotázs céllal fejlesztették ki őket (Jerusalem sorozat, Saddam) — vagy közvetlen károkozás nélküliek, amelyek csak szaporodnak, és a felhasználók életét keserítik meg (Polimer).

A vírusok ellen hardvereszközökkel vagy *antivírus programokkal* lehet védekezni. A vírusokat nemzetközileg a McAfee-féle, illetve a VIRSUM elektronikus kiadványban alkalmazott lajstromnevükön, illetve a Virus Bulletin elektronikus kiadvány megfelelő jelölésein keresztül azonosítják. A vírusok lezármazási fákat alkotnak *törzs (strain)*, amelyeken belül megkülönböztetünk

egyres változatokat (*mutáció*). 1992 februárjában mintegy 1200 MS-DOS, 400 *Macintosh*, 260 körüli AMIGA vírustörzset tartottak nyilván.

A számítógépes vírusok hasonló terjedési törvényszerűséget mutatnak, mint az emberi vagy állati járványokat okozó vírusok. Egy komputervírus elengedésének a hatása hasonlóképpen kiszámíthatatlan, mint a biológiai hadviselésé.

Az igazi antivírus-szakemberek vírushódót, vírust szívesen elfogadnak, de cserélni, kiadni kívülállóknak nem hajlandók. (Éppen ezért hiányoznak a teljes kódok a nyilvános publikációkból.)

A vírustechnológia a legbeavatottabb, legbonyolultabb programozási eljárásokat alkalmazza, amelyeket a nyilvános dokumentációkból ki sem lehet deríteni.

visible part (látható rész): Lásd moduláris programfelépítés.

visszafejtés (reverse engineering): Ha a programról nincs *forrás* szintű dokumentációnk, megpróbálhatjuk a futtatható kódból kideríteni a működést. Ehhez *nyomkövetőket*, visszafordító programokat vehetünk igénybe. Elvileg ilyesmire nincs szükség, a kérdések megválaszolása, a hibák kijavítása a gyártó feladata; viszont régi programok és távoli gyártók esetén rászorulhatunk.

A védelem kiiktatása, a program megváltoztatása *kalózkodás*. ► copyright.

visszatérő érték (return value): Az az érték, amelyet egy *függvény* szolgáltat. ► szubrutin.

VisualBASIC: Programozási nyelv *Windows*-hoz és fejlesztői környezet, kevés kapcsolata van az eredeti *BASIC* nyelvhez. Egy grafikus képszerkesztő programmal *ablakokat* rendezhetünk be, *menüket*, kapcsolókat, szövegek és számok bevitelére és kiírására szolgáló részeket és egyéb ilyen *Windows*-os eszközöket helyezhetünk el az ablakban. Az egyes eszközökhöz (QBasic-ben) lehet rutinokat megadni. Ezek a rutinok indulnak el az eszköz igénybevételekor. ► eseményvezérelt programozás.

VLSI: 1. Az USA-ban működő félvezetőgyártó.

2. Lásd integrált áramkör.

void: Üres típus. A modernebb nyelvekben annak jelölésére használják, hogy a függvény nem ad vissza értéket, vagy nem kér paramétereket. Például a „void initsrc(void);” azt jelöli, hogy a képernyőkezelés *inicializálását* végző függvény (az „initsrc”), nem kér bemenő paramétereket és nem ad vissza semmilyen értéket.

vonalfrafika: Vektorgrafika. Lásd grafika.

W

wait state: A processzor sebessége a számítógépek nagy részénél meghaladja a memória sebességét. A wait state olyan állapotot jelöl, amikor a processzor tétlenül a memóriára vár. Ez lassítja a rendszert, de olcsóbb — és lassabb — memória használatát is lehetővé teszi. Különböző memóriavezérlő áramkörökkel és a CPU mellett alkalmazott *cache-sel* minimálisra csökkenthető a wait state állapotok száma.

WAN: (Wide Area Network) Lásd távolsági hálózat.

warm boot (meleg indítás): A gépet az áram kikapcsolása nélkül indítjuk újra (IBM PC-n: Ctrl-Alt-Del). A számítógép és a perifériák tesztje elmarad. A munka az operációs rendszer betöltésétől indul.

warning: (figyelmeztetés): Lásd hiba.

Weitek koprocesszorok: A Weitek cég az Intel 80286-os processzorához az 1167-es, a 80386-os CPU-hoz a 3167-es a 80486-hoz 4167-es matematikai processzorait gyártja. Ezek sem az utasításokban, sem a kivezetésekben nem kompatibilisek az Intel 80287-es, 80387-es típusaival. Így csak a megfelelően kialakított *alaplapon* alkalmazhatók, olyan szoftverekkel, amelyek kezelik ezeket a típusokat.

Teljesítményük és az Intel típusok hiányosságai miatt több gyártó is készít ilyen alaplapokat; a nagy teljesítményű grafikus tervezőprogramok használják a Weitek processzorokat.

wild card character: Lásd joker (dzsóker karakter).

winchester: Lásd harddiszk.

window: Lásd ablak.

Windows (3.0): Az MS-DOS fölé épülő grafikus interfészű, *ikonos* vezérlésű operációs rendszer, a Microsoft cég terméke. Lehetőség van arra, hogy egyszerre több *ablakban* több programot futtassunk. DOS programokat is tud futtatni, ha azok nem hajtanak végre olyan trükköt, ami az operációs rendszer alá nyúl. MS-Windows programok fejlesztéséhez nagyobb *konfiguráció* szükséges, mert a programok az interfész miatt meglehetősen nagyok, a gép nem kellő kiépítése esetén lassúak is.

1992 tavaszán jelent meg a 3.01-es verzió. Ebben már úgynevezett *True Type fontok* is vannak, azaz olyan *betűkészletek*, amelyek már vonalakkal vannak megadva, és tetszőleges méretre nagyíthatók torzításmentesen. ► kiadványszerkesztő, PostScript, WYSIWIG.

Winword: A *Word for Windows* beceneve.

Word for Windows (2.0): A Microsoft cég *kiadványszerkesztője* az MS-Windows alá. Ikon- és menüvezérelt, de makrónyelve is van, a WORD BASIC. Utasítható, hogy felismerje a *WordPerfect* billentyűparancsait. Lehetőségei/képességei: *redlining*, *annotáció*, *stíluslap*, *körlevélkészítés*, képletszerkesztés. 14 nyelvre van helyesírás-ellenőrző programja és szinonimaszótára.

A Windows alatti *táblázatkezelők* behívhatók és a táblázatok helyben, a szövegben állíthatók össze vagy módosíthatók. Át tud venni adatokat számos *szövegszerkesztőből* és grafikus fájlközből.

word processor: Lásd szövegszerkesztő.

WordPerfect: A korábbi évek legnépszerűbb *szövegszerkesztője*. Alapjában véve billentyűkkel és billentyűkombinációkkal vezérelhető, de 5.1-es változatában már számos funkciót lehet *egérrel*

és menüvel is irányítani. Számos fontot ismer, fej-, illetve lábsort, a jobb és a bal oldali lapot figyeli, különböző betűméretek és sortávolságok váltakozhatnak. Automatikus elválasztással, szóhelyesség-ellenőrzéssel, szinonimaszótárral segíti a munkát. Sajátos utasítása a **kódfeltárás (Reveal Codes)**, amely megmutatja, hogy milyen formátumvezérlő utasítások vannak elhelyezve a szövegben. Ennek segítségével tudjuk a megfelelő szövegrészt kivágni és áthelyezni úgy, hogy a megfelelő formátumvezérlés vele menjen vagy éppen ne. A WordPerfect for Windows a programnak a Windows alá készített továbbfejlesztése, amely 18 nyelvet ismer. Mindazonáltal kevesebbet tud, mint a Word for Windows, és a kódfeltárás elavult csökevény a Windows-os filozófiában.

WordStar: Igen elterjedt program, szövegek/kiadványok szerkesztésére. Jellemzői: 43/50 soros üzemmód; HP Leserjet III és *PostScript* nyomtatók kezelése; SAA szabvány szerinti *helprendszer*; *többablakos* szerkesztés; fejléc és lábjegyzet lehetőség, kolumnás (oszlopos) tördelés; angol helyesírás-ellenőrző és szinonimaszótár, automatikus elválasztás; szövegformázás, rajzolás és képelhelyezés a szövegben; átjárás más szövegszerkesztők formátumába, és *Lotus*, *dBASE*, *Quattro* állományok olvasása; címnyilvántartás, körlevélkészítés és

címkenyomtatás. *Page preview:* az elkészült lapok alakjának megjelenítése, többfokozatú nagyítás.

Kissé átalakított változata *Windows* alatt is fut. A *kiadványszerkesztéshez* használatos valamennyi funkciója megvan, de kezelése nem egyszerű.

workstation (munkaállomás): A személyi számítógépeknél általában nagyobb teljesítményű, de egyetlen felhasználót kiszolgáló gép.

A sokféle típus közös jellemzője a *RISC* processzor, az alapkiépítésben is néhányszor 10 Mb-ot memória, több száz Mb-ot harddisk-kapacitás, nagyfelbontású grafikus képernyő. Rendszerint hálózatba van kötve, vagy egy nagyszámítógéphez kapcsolódik terminálként.

A munkaállomások legtöbbször a *UNIX* operációs rendszer valamelyik változata alatt működnek, tipikusan műszaki és kutatási alkalmazásokkal.

WORM (Write Once, Read Many): Csak egyszer írható, de aztán korlátlanul olvasható *optikai diszk*.

wrap: Olyan működési mód, amikor egy fájl, képernyő stb. végére érve a megismételt parancs nem vált ki hibajelzést, hanem az elejére visszaugorva hajtódik végre.

write protected: Lásd írásvédelem.

WYSIWYG elv: Lásd kiadványszerkesztő.

X

X.25: Lásd távolsági hálózat.

X-WINDOWS: Lásd UNIX.

XENIX: A Microsoft cég által létrehozott *UNIX*-szerű operációs rendszer, amely képes együttműködni az ugyanazon a gépen lévő, de máskor futó *MS-DOS* operációs rendszerrel. ► particionálás.

XGA (Extended Graphics Array): Az IBM 1990-ben megjelent *képernyőadaptere*, amelyet a *PS/2* sorozat 90-es és 95-ös modelljeibe épített be. A korábbi *VGA*-val kompatibilis üzemmódjai mellett új grafikus módjai az 640x480-os és 1024x768-os felbontás, 256 egyidejű színnel; és a 640x480-os felbontás, 65 536 színnel. *Alfanumerikus* módban új az egy sorban 132 karaktert megjelenítő mód. Az XGA lényegesen intelligensebb a *VGA*-nál, beépített *koprocesszora* több elemi feladat végrehajtására képes, például a képernyő víz-

szintes és függőleges koordinátáinak átszámítására közvetlen *videómémóriacímmé*.

XMS (eXtended Memory System): Lásd extended memória.

XTree: Fájl- és könyvtárkezelő program. Adattömörítést, lemez- és memóriastatisztikát nyújt, beépített — a *WordStarral* kompatibilis — szövegszerkesztője van. A *DOS parancsértelmezőjének* kiegészítőjeként szolgál.

XyWrite: Az egyik igen kedvelt, szövegek szerkesztésére való program. Egyidejűleg 9 állomány és 9 *ablak* kezelhető. Lézernyomtatókkal tud dolgozni, automatikus a laptördelés és a sorozámozás. Van fejlécezési és lábjegyzetelő funkciója, továbbá kolumnás (oszlopos) tördelés, automatikus tartalomjegyzék-készítés is. A *redlining* funkcióval a javítások és az eredeti szöveg is belekerül a fájlba. Angol helyesírás-ellenőrző és szinonimaszótár segíti a szöveggészítést. Saját *menü- és help-rendszer* *installálható* a felhasználó igénye szerint.

Z

Z-80: A Zilog céget az Intel egyik volt főkonstruktőre alapította. A cég 1976-ban hozta ki ezt a 8 bites processzort. Szoftverkompatibilis a 8080A-val, de annál lényegesen korszerűbb CPU.

zárt szubrutin: Lásd szubrutin.

zéróelnyomás: Az adott számú jegyre kinyomtatandó szám elejére nem nullákat teszünk, hanem szóközöket. (Banki alkalmazásnál más karaktereket, például *-ot tesznek a 0 helyére.)

zoom: Egy képernyőn megjelenített dokumentum egy részének kinagyítása vagy kicsinyítése. Grafikus szerkesztőprogramok egyik funkciója, finom változtatásokat tesz lehetővé a képen.

Függelék

(A táblázat értelmezése: a * jel azt mutatja, hogy a parancs ebben a verzióban jelent meg, az m pedig arra utal, amelyikben módosították.)

DOS-parancsok és verziók								
	1.0	2.0	3.0	3.1	3.2	3.3	4.0	5.0
APPEND				*			m	
ASSIGN		*						
ATTRIB			*		m	m		m
BACKUP		*	m			m	m	
BREAK		*						
CALL						*		
CHCP						*		
CHDIR(CD)		*						
CHKDSK	*m	m					m	
CLS		*						
COMP	*	m						
COPY	*							
CTTY		*						
DATE	*		m			m		
DEL	*	m					m	
DEBUG	*m	m						
DIR	*	m					m	m
DISKCOMP	*m	m	m		m		m	
DISKCOPY	*m	m	m		m		m	
DOS								*
DOSKEY								*

Függelék

DOS-parancsok és verziók

	1.0	2.0	3.0	3.1	3.2	3.3	4.0	5.0
DOSSHELL							*	m
ECHO		*				m		
EDIT								*
EMM386								*
ERASE	*m	m					m	
EXIT						*		
FASTOPEN						*	m	
FC								*
FDISK		*				m	m	
FIND		*						
FOR								*
FORMAT	*m	m	m		m		m	m
GOTO		*						
GRAFTABL			*			m	m	
GRAPHICS		*	m				m	
HELP								*
IF								*
JOIN				*				
KEYB					*		m	
LABEL			*		m		m	
LOADHIGH								*
MEM							*	m
MIRROR								*
MKDIR(MD)		*						
MODE	*m					m	m	
MORE		*						

Függelék

DOS-parancsok és verziók

	1.0	2.0	3.0	3.1	3.2	3.3	4.0	5.0
NLSFUNC					*			
PATH		*						
PRINT		*						
PROMPT		*						
RECOVER		*						
RENAME(REN)	*							
REPLACE				*			m	
RESTORE		*	m			m		
RMDIR(RD)		*						
SELECT			*		m		m	
SET		*						
SETVER								*
SHARE			*				m	
SORT		*						
SUBST					*			
SYS	*							
TIME	*					m	m	
TREE		*		m				
TRUENAME							*	
TYPE	*							
UNDELETE								*
UNFORMAT								*
VER		*						
VERIFY		*						
VOL		*					m	
XCOPY					*			

A

Cédrus Kiadó

még kapható ALAPLAP KÖNYVEI

<i>Nagy Gábor: Tömör gyönyör</i>	256,- Ft
<i>Kis János–Szegei Imre: Új vírusléktan</i>	256,- Ft
<i>Kis János–Szegei Imre: Vírushatározó</i>	256,- Ft
<i>Detrik Péter: Az SQL nyelvről</i>	375,- Ft
Számítástechnikai Alaplexikon I. <i>Jodál Endre: Általános fogalmak</i> (2. kiadás)	356,- Ft
Számítástechnikai Alaplexikon II. <i>Jodál Endre: Adatkommunikáció</i> és számítógép-hálózatok	356,- Ft
<i>Farkas Ernő: PC-szótár</i>	456,- Ft
<i>Kis János: BBS – avagy az elektronikus postaláda</i> (mágneslemez melléklettel)	656,- Ft

Könyveink megrendelhetők postai utánvétellel címünkön:



Cédrus Kiadó Kft

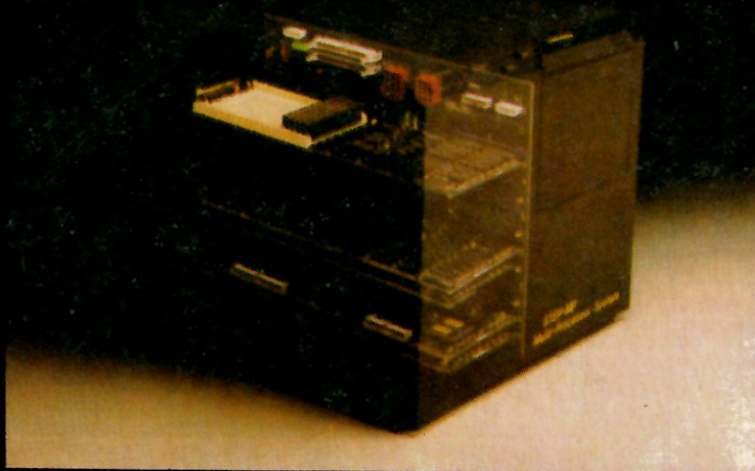
1441 Budapest 70. Pf. 74.

456,- Ft



**INTRAM Szerviz és
Kereskedelmi Kft.**

1072 Budapest VII., Kis Drófa u. 6.
Telefon: 122-0087 Fax: 121-3230



Ilyen még nem volt Magyarországon!
Nálunk a minőség mindig megéri az árát!