

ENTERPRESS

KÉTHAVILAP AZ ENTERPRISE SZÁMÍTÓGÉPEK FELHASZNÁLÓINAK

A TARTALOMBÓL:

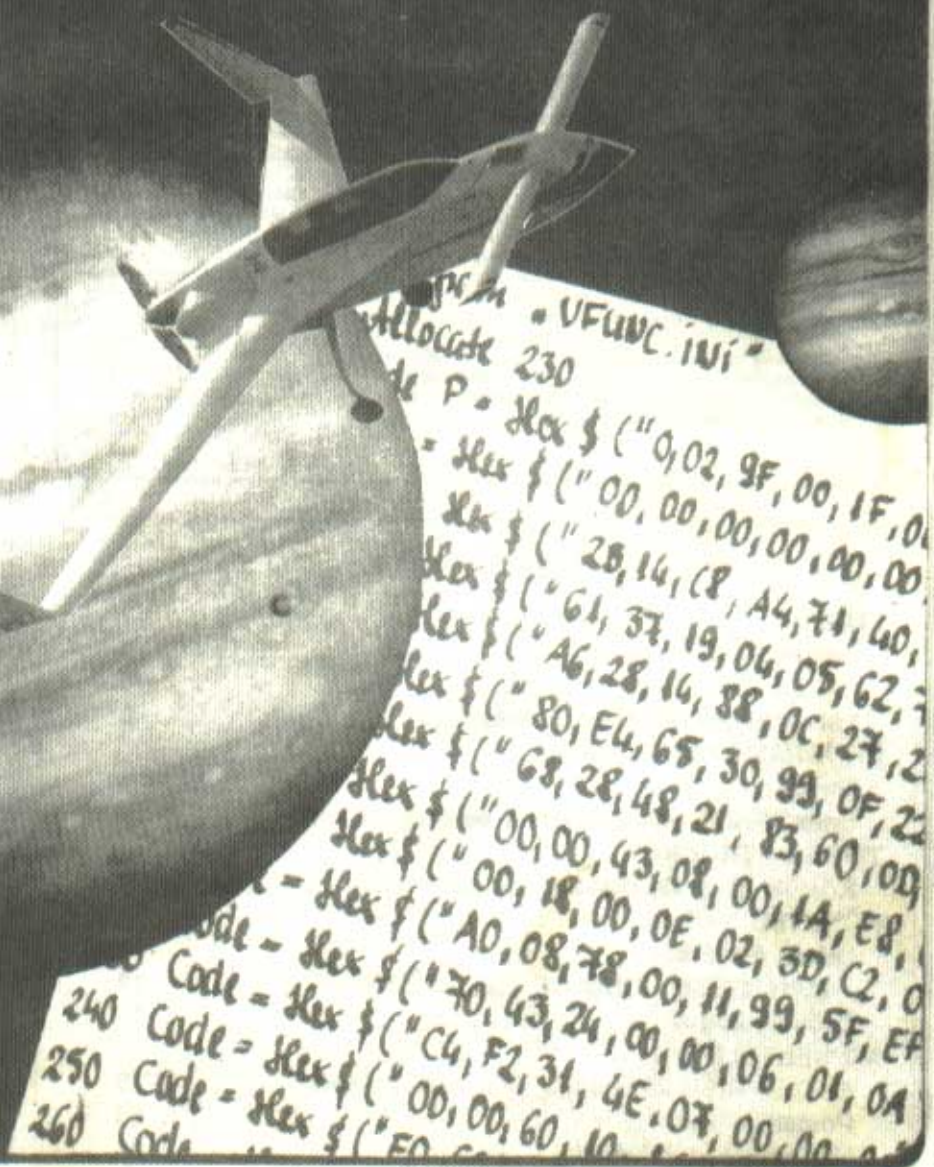
- Assembly
- Hanoi-torony program
- Sprite tervező
- Pascal
- Tippek - trükkök
- Mozart/Vangelis zene demók
- Mindenféle

II. évfolyam

2. szám

1991 március-április

MATRIX Kft.



A helyzet jó, de nem reménytelen

Napról-napra mind több felháborodott olvasói levél érkezik szerkesztőségünk és kiadónk címére. Az olvasók panaszkodnak, hogy hiába keresik égen-földön lapunkat, azt egyetlen újságosstandon sem látják kiéve, az árus pedig még csak nem is hallott az ENTERPRESS-ről.

Bár olvasóinkat ez aligha vigasztalja, sajnos mi is tapasztaljuk ezeket a jelenségeket. A Postával ugyanolyan szerződésünk van, mint más, az ENTERPRESS-nél sokkal régebbi számítástechnikai lapnak. A Posta lapterjesztéssel foglalkozó része állítólag jelentős átszervezés alatt áll, és sajnos ennek levét - meglehetősen drágán - az újságok isszák meg. Sajnálatos módon a Posta hanyag munkája miatt lapunk második száma veszteséges lett, a kiadott példányoknak több, mint fele visszajött kiadónk címére. A Posta illetéke természetesen megmagyarázza a dolgot, hivatkozva a régi lapterjesztési rendszerre. Az indokokat el kell fogadnunk, hiszen a "megújult" Postának igaza is lehet, meg különben sem tudunk mást tenni. A baj csak az, hogy amikor fizetnünk kell a nyomdának, akkor nem hivatkozhatunk a Postára, mely hivatal jóvoltából egy árva fillérünk sincs, mert lassan fizeti ki nehezen szerzett pénzünket.

De hagyjuk abba a panaszkodást! A lényeg: a Postával új szerződést kötöttünk, így várhatóan javul a Posta velünk szemben tanúsított bánásmódja, és egyre türelmetlenebb olvasóink megrendelőink biztosan lapunkhoz jutnak. Mi határozzuk meg, hogy az ország mely részein jelenjen meg az ENTERPRESS, megrendelőinket pedig közvetlenül kiadónk fogja kiszolgálni, (gyök sem lesznek kiszolgáltatók helyzetben).

Visszatérve a remittendára: sem a szerkesztőség, sem a kiadó nem ijed meg az újsághegy láttán, hiszen az összes kiadott példány is csupán a felhasználók 25%-ának lenne elegendő, az igény ennél viszont jóval nagyobb, így a kiadóhoz visszaérkezett példányok néhány hét alatt el fognak kelni. Másrészt a veszteség lehetőség - ahogy mondani szokás - benne volt a pakliban, a Mátix ugyanis az első négy számot akár teljes veszteséggel együtt is kiadja. (Az ötödiket már nem.) A helyzet ennél szerencsére sokkal jobb, az ENTERPRESS létezik, és továbbra is létezni akar.

Amitől inkább tartunk az az, hogy ilyen eladási tempó mellett csak lassan tudjuk lapunk terjedelmét növelni, a megjelenést sűríteni, hiszen csigalassúsággal teremődnek meg a bővített újrateljesítési feltételei.

Pedig az újságra szükség van, ezt bizonyítják a címünkre érkező levelek is. Nem panaszkodhatunk, hogy nincs mit olvasnunk! Folyamatosan érkeznek azok a cikkek, amelyeket olvasóink írtak, következő számunktól kezdve már rendszeresen közlünk ilyeneket, bevezetésképpen már a számunkban is olvashatnak néhányat. A cikkek témái igen változatosak, a géphez kapcsolható aritmetikai processzormodultól kezdve egészen a játékleírásokig nyúlik mondanivalójuk. Megkapjuk az Enteface anyagait is, így tehát van - és reméljük lesz - miből válogatunk.

Bár előző számunkban még csak beharangoztuk a programküldő szolgáltató beindítását, ennek ellenére már jónéhányan jelezték beküldési szándékukat. A jelentkezők leveleit készséggel fogadjuk, de konkrétumokat még máig sem tudunk nekik mondani. A forgalmazás vázlatosan a következőképpen fog kinézni: a program írója elküldi címünkre a programot a dokumentációval együtt, majd mi megvizsgáljuk az anyagot. Ha elnyeri tetszésünket, akkor megkötjük a terjesztési szerződést, és ezen kívül a szerzővel egy hiteltelenségi nyilatkozatot írunk alá. (Ez a nyilatkozat tulajdonképpen bennünket véd meg abban a kényes helyzetben, amikor véletlenül kiderül, hogy a program mégsem a szerző saját kútfőjéből való...) Ezt követően a szoftvert besoroljuk programküldő szolgáltatónk készletébe. A lap hasábjain megjelenő kínálatban feltüntetjük a program címét, egy mondatban leírjuk a lényegét, megjelöljük az árát stb. A beérkező megrendeléseket postázzuk, a nyereség nagyobbik részét pedig rögzített időközönként kifizetjük a szerzőnek. Ez tehát a terv, a megvalósulás már jó úton halad.

Gondolom most sokan azt kérdezik, hogy jó, jó, de miért halad az előkészítés ilyen lassan? A válasz egyszerű: nem akarunk kellő körültekintés és puhatóldozás nélkül belevágni a dologba. Mi történe akkor, ha egy adott programra egyszerre befutna 200 megrendelés? (De szép is lenne...) Egy ilyen nagyszámú tétel másolása nem oldható meg egyetlen magával, ez időigényes, rendkívül "magnófaló" és neveséges lenne, ezért kell valamilyen komolyabb, rugalmasabb másolóberendezést keresnünk (a nepperekaél biztosan találnánk ilyen kapacitást). Másik indokunk pedig az, hogy egyelőre inkább a lap életben tartására, bevezetésére fordítjuk szabad energiáinkat.

Az ENTERPRESS tehát alapvetően jól startolt, de még jócskán van tennivalónk. Sok felhasználó még mindig nem ismeri lapunkat, hozzájuk is el kell jutnunk, ez mindannyiunk érdeke. Szeretnénk propagálni az ENTERPRESS-t más lapokban, de a rendelkezésünkre álló pénzkerekből ez eléggé nehézkesnek tűnik. Azon se lepődjünk meg majd a Tisztelet Olvasó, ha esetleg fogkrémreklámot lát az ENTERPRESS oldalain.

Hajnal Csaba

TARTALOM

KURZUS

Assembly 4.	3-5
A Pascal 3.	6-7
Lehetőségek Páratlan Tárháza (LPT) 3.	8

PROGRAMOZÁSTECHNIKA

Hanoi tornyal	9
EnterSPRITE példaprogram és még valami	10

TIPPEK - TRÜKKÖK

Köztük: Dvorak, Mozart, Vangelis dallamok

KÖNNYED MŰFAJ

Leírás és térkép: MAHJONGG, KING OF THE CASTLE	13
Örökéletkódok	14

MINDENFÉLE

ENTERPRISE képviselő nyílt Budapesten	15
Postafiók 334	16
Hírdetések, felhívások	16

ENTERPRESS

Kéthavilap
az ENTERPRISE számítógépek
felhasználóinak

II. évfolyam 2. szám
1991. március — április

Kiadja a
**MÁTRIX Számítástechnikai,
Kereskedelmi és Szolgáltató Kft.**

Székesfehérvár

Felelős kiadó:

Annus István

ügyvezető

Felelős szerkesztő:

Ujlaki László (UL)

A lap szerkesztői:

Hajnal Csaba (HCs)

DevilSoft (Devil)

Bozai Gábor (BG)

Címlap:

Németh Ferenc

Technikai szerkesztő:

Bartha István

A szerkesztőség és a kiadó címe:

8000 Székesfehérvár

Dózsa György tér 10.

Telefon: (22) 12 - 619

Telefax: (22) 11 - 585

Levélcíme:

ENTERPRESS

1399 Budapest

Pf. 701/334

Lapunk az ENTERPRISE Computers
GmbH kelet-európai képviselőjének, a
VTGe Electronics Ltd.-nek szakmai
támogatását élvezi.

Nyomja a

VIDEOTON Nyomda

Székesfehérvár

Felelős vezető:

Gombaszögi József

ISSN 0866-1820

Terjeszti a

Magyar Posta

Előfizethető a HELIR Bp. 1900 címen

Előfizetési díj

egy évre 294 Ft, fél évre 147 Ft

Ára: 49,- Ft

Következő számunk

május 20-án jelenik meg

Az ENTERPRESS-ben közreadott információk célja az, hogy segítsék, tudnivalókkal lássák el az ENTERPRISE számítógépek felhasználóit. A közölt programokat, kapcsolási rajzokat és leírásokat mindenki szabadon felhasználhatja, de tilos azokat a kiadó írásbeli engedélye nélkül másolni, terjeszteni.

A szerkesztőség kéziratokat nem őriz meg, és nem küld vissza.

Assembly

4. rész

Alapvetően már csak két utasításfajta ismertetése van hátra, a következő részben már az EXOS-t tárgyaljuk. E fejezet végén vázlatosan megismerkedhetünk az ASMON és a DEVPAC programokkal.

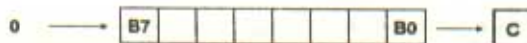
Eltoló utasítások

Ha regiszterben vagy memóriában elhelyezkedő biteket szeretnénk balra-jobbra mozgatni, akkor ezt az eltoló utasításokkal tehetjük meg. Nyilván a léptetés során egy-egy bit mindig kiesik, de nem vész el, hiszen ez a bit bekerül a C bitbe. Ha a C bit tartalma a következő eltoló műveletnél visszalép a túloldalon a bájtbá, akkor ciklikus eltoló művelettel van dolgunk. Az eltoló műveletekkel egyszerűen elvégezhető a kettővel való szorzás és az osztás. Ezt könnyen beláthatjuk, mert ha például a bájtban 00101001b (41) van, és ezt a tartalmat balra egy hellyel eltoljuk, akkor a 01010010b (82) kétszeres érték fog megjelenni. Ilyen utasításokat használunk jobbra-balra scrollozó rutinokban is.

Regiszter vagy tárrekesz tartalmát tudjuk jobbra léptetni az SRL (Shift Right Logical, logikai eltolás jobbra) utasításokkal, a már ismert formákkal:

SRL A
SRL B
SRL C
SRL D
SRL E
SRL H
SRL L
SRL (HL)
SRL (IX+d)
SRL (Y+d)

Az SRL utasítás tehát egy hellyel jobbra lépteti a biteket, a jobb oldalon kipotyogó bit a C bitbe kerül, a bájti 7. bite pedig 0 lesz.

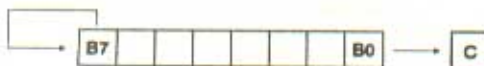


SRL

Másfajta biteltolást végez az SRA (Shift Right Arithmetical, aritmetikai eltolás jobbra) utasítás.

SRA A
SRA B
SRA C
SRA D
SRA E
SRA H
SRA L
SRA (HL)
SRA (IX+d)
SRA (Y+d)

Az SRA utasításnál a jobboldalon kieső bit szintén a C bitbe kerül, a b7 értéke viszont végig az eredeti marad.



SRA

Balra végez eltolást az SLA (Shift Left Arithmetical, aritmetikai eltolás balra) utasítás.

SLA A
SLA B
SLA C
SLA D
SLA E
SLA H
SLA L
SLA (HL)
SLA (IX+d)
SLA (Y+d)

Az utasítás hatására a bájti bitek egy hellyel balra mozdulnak, a baloldalon kieső bit C bitbe kerül, b0-ba pedig 0 kerül.



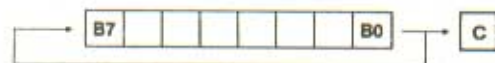
SLA

Eddig bölcsen hallgattam arról, hogy mit is jelent az utasítások elnevezésében a logikai, illetve az aritmetikai eltolás kifejezés. Az aritmetikai eltolásoknál fontos szerepe van az előjelbitnek is. Az SRA utasítás is ezért hagyja változatlanul a 7. bit (előjelbit) értékét. Osszuk el példaképpen a -20-at (11101100b) 2-vel! Az SRA utasítás végrehajtása után a megkapjuk a -10-et (11110110b). Ha balra mozgatunk (azaz kettővel szorzunk), akkor az előjelbit értelemszerűen ki fog esni, de a C bitben rendelkezésünkre áll, ennek kezelése a mi feladatunk. Legyen a szám -30 (11100010b), melynek kétszerese -60 (11000100b) előáll az SLA után. A logikai eltolások - az ábrán látható módon - egyszerűen csak a biteket mozgatják. Mivel az SLA utasítás mindkét céljának tökéletesen megfelel, így nincs külön valamiféle SLL nevezetű utasítás. A C bit (ezek szerint) a kilépő bitet tárolja, a Z bit pedig mindig a kapott eredménynek megfelelően áll be.

A ciklikus eltolásoknál a kipotyogó bitek azonnal, vagy a C biten keresztül visszalépnek a túloldalon. Mindenféle magyarázat helyett lássuk az utasításokat és a működésüket illusztráló ábrákat!

Az RRC (Rotate Right Circular, ciklikus eltolás jobbra):

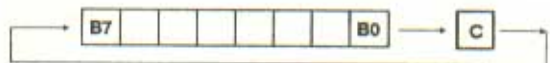
RRC A
RRC B
RRC C
RRC D
RRC E
RRC H
RRC L
RRC (HL)
RRC (IX+d)
RRC (Y+d)



RRC

Az RR (Rotate Right, eltolás jobbra) utasítás:

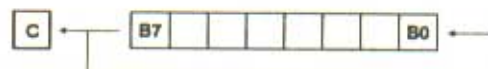
RR A
RR B
RR C
RR D
RR E
RR H
RR L
RR (HL)
RR (IX+d)
RR (Y+d)



RR

Balra mozgatja a biteket az RLC (Rotate Left Circular, ciklikus eltolás balra) utasítás:

RLC A
RLC B
RLC C
RLC D
RLC E
RLC H
RLC L
RLC (HL)
RLC (IX+d)
RLC (Y+d)



RLC

Utolsó a sorban az RL (Rotate Left, eltolás balra) utasítás:

RL A
RL B
RL C
RL D
RL E
RL H

RL L
RL (HL)
RL (IX+d)
RL (IY+d)

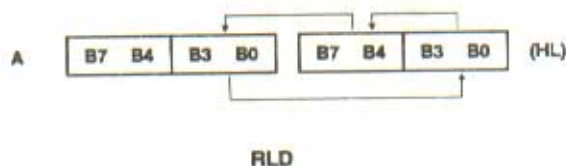
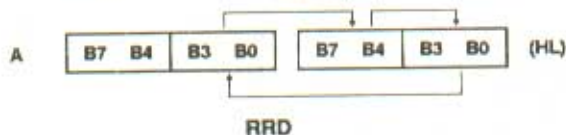


A műveletek az ábrákon látható módon állítják be a C bitet, a Z bit pedig az eltolás után kapott eredménynek megfelelő értéket kap. Ide tartozik még négy utasítás:

RRC A
RRA
RLC A
RLA

Ezek az utasítások az akkumulátoron keresztül végzik az eltolást, és nem állítják a Z bitet. Előnyük viszont, hogy kétszer gyorsabbak RRC A, RRA, RLC A, RLA társaiknál.

Ebből az utasításfajtából már csak az RRD (Rotate Right Decimal, decimális eltolás jobbra), és az RLD (Rotate Left Decimal, decimális eltolás balra) utasítások hiányoznak. Ezek tulajdonképpen az ún. binárisan kódolt decimális (BCD) számokkal végeznek műveletet (a BCD számokról néhány mondatnál később esik majd szó). Működésük az ábrák alapján világossá válik.



Az utasítások a C bitet nem bántják, a Z pedig az akkumulátorban kapott eredménynek megfelelően áll be.

Blokkműveletek

A blokkműveletekkel megadott számú bájt mozgathatunk, összehasonlíthatunk a tárban; beolvashatunk, kiírhatunk a portokra.

Blokkmozgatást az LDI (block Load Increment address, blokkmásolás címnöveléssel), LDIR (block Load Increment address auto Repeat, blokkmásolás címnöveléssel, automatikus ismétléssel), LDD (block Load Decrement address, blokkmásolás címcsoökkentéssel), LDDR (block Load Decrement auto Repeat, blokkmásolással címcsoökkentéssel, automatikus ismétléssel) utasításokkal végezhetjük.

Az

LDI

utasításnál a HL tartalmazza azt a címet ahonnan, a DE pedig azt, ahová akarunk másolni; BC-ben a másolandó bájtok számát adjuk meg. Az utasítás végrehajtása után HL és DE értéke eggyel növekszik, BC eggyel csökken. Lehetőségünk van a helyzet analízisére, ennek megfelelően eldönthetjük, hogy visszaugrunk az LDI-re, vagy sem.

Folyamatosan hajtja végre a blokkmásolást az

LDIR

utasítás. A regisztereket a fentieknek megfelelően kell beállítani. Itt a másolás addig tart, amíg BC nem egyenlő nullával.

Az

LDD

és az

LDDR

utasítások ugyanígy működnek, azzal az óriási különbséggel, hogy ezeknél HL és DE csökken.

Az utasítások nem módosítják a C és a Z biteket.

Blokkösszehasonlítást végezhetünk a CPI (block Compare Increment address, blokkösszehasonlítás címnöveléssel), CPIR (block Compare Increment address, auto-Repeat), CPD (block Compare Decrement address, blokkösszehasonlítás címcsoökkentéssel) és a CPDR (block Compare Decrement address, auto-Repeat, blokkösszehasonlítás címcsoökkentéssel, automatikus ismétléssel) utasításokkal.

A

CPI

utasításnál az A regiszterbe kell töltenünk azt az értéket, amellyel a HL által címzett memóriarekesz tartalmát akarjuk összehasonlítani. Az utasítás a BC regiszter értékét is csökkenti, szükség esetén ezt is felhasználhatjuk.

A

CPIR

utasítással egyszerűen kereshetünk meg bizonyos értékeket a memóriában. A-ba a keresendő értéket, HL-be az indulási címet, BC-be pedig az összehasonlítandó bájtok számát töltjük. A művelet során HL eggyesével növekszik, BC csökken. Ha a CPIR utasítás az A regiszter értékével megegyezőnek találja a HL által címzett rekesz tartalmát, akkor abbahagyja a keresést. Ekkor HL már a következő bájtra mutat! Ha a keresés során a bájtárolóként működő BC tartalma eléri a nullát, akkor szintén befejeződik a végrehajtás.

Ezek után a

CPD

CPDR

utasításokról csak annyit kell elárulni, hogy itt értelemszerűen HL csökken.

A blokkösszehasonlító utasítások a Z bitet állítják, a C bithez nem nyúlnak.

Blokkba tudjuk letárolni egy port tartalmát az INI (block Input Increment address, blokkinput címnöveléssel), INIR (block Input Increment address, auto-Repeat, blokkinput címnöveléssel, automatikus ismétléssel), IND (block Input Decrement address, blokkinput címcsoökkentéssel) és az INDR (block Input Decrement address, auto-Repeat, blokkinput címcsoökkentéssel, automatikus ismétléssel) utasításokkal.

Az

INI

végrehajtása előtt C regiszterbe a port címét, HL-be a letárolás kezdőcímét kell töltenünk, a B regisztert ciklusszámlálóként használhatjuk. A Z bit egybe billen, ha a B regiszter tartalma nulla lesz.

Az

INIR

utasításnál a regiszterek szerepe ugyanaz, mint az INI-nél, de itt a CPU egyhúzásra végrehajtja a portról olvasást, és a letárolást. Az utasítás végrehajtása addig tart, amíg B el nem éri a nullát. A Z bit itt végig 1.

HL csökkentésén kívül ugyanezt teszik az

IND

INDR

utasítások is.

Ez az utasításcsoport a C bitet változatlanul hagyja.

A blokkokkal kapcsolatos kimeneti OUTI (block Output Increment address, blokk kivetel címnöveléssel), OTIR (block Output Increment address, auto Repeat, blokk kivetel címnöveléssel, automatikus ismétléssel), OUTD (block Output Decrement address, blokk kivetel címcsoökkentéssel), OTDR (block Output Decrement address, auto Repeat, blokk kivetel címcsoökkentéssel, automatikus ismétléssel) utasítások már önmagukért beszélnek.

Az

OUTI

utasításnál C tartalmazza a portcímét, HL a forrás helyét, B a kiírandó bájtok számát. Az utasítás hatására tehát B "darab", a HL által címzett blokkban elhelyezkedő bájt tartalma fródiók ki a C regiszterben megadott portra. A Z bit 1 értéket vesz fel, ha B értéke eléri a nullát.

Ugyanezt végzi el egyetlen ciklusban az

OTIR

utasítás, amelynél a Z bit végig 1.

HL tartalmának csökkentésén kívül ugyanígy működnek az

OUTD

OTDR

utasítások.

A blokkos kivetel nem módosítja a C bitet.

És ami kimaradt

Maradt még néhány érdekes, különleges utasítás, melyek nagyrésze nélkülözhetetlen a programozás során. Ezeket most mindenféle külön kategóriába sorolás nélkül nézzük végig. Az

SCF

(Set Carry Flag, az átvitelbit 1-be állítása) és a

CCF

(Complement Carry Flag, az átvitelbit komplementálása) utasítás. A komplementálás azt jelenti, hogy a C bitet ellenkezőjére változtatjuk. A C bitet szokás különféle jelzések (pl. hiba történt) adására használni.

A

DAA

(Decimal Adjust Accumulator, az akkumulátor decimális igazítása) utasítás a BCD formátumban végzett műveletek eredményeinek korrigálásánál tesz jó szolgálatot. A BCD ábrázolásnál a bájt alsó négy bitjén az egyesek, felső négy bitjén a tízesek számát tároljuk. Nézzük meg a 29 BCD reprezentációját!

29 = 0010 1001 (BCD)

Nos, ha két ilyen számot összeadunk, akkor hamis eredmény is adódhat. Legyen az akkumulátorban a BCD-ben ábrázolt 29, a B regiszterben pedig 19 (0001 1001). Ha most a CPU-val végrehajtatjuk az ADD A,B utasítást, akkor 42-t fogunk kapni, amely rossz eredmény. A DAA utasítás azonban korrigálja az eredményt, az akkumulátorban 48 (0100 1000) lesz. A C bit értéke itt több mindentől függ, alkalmas helyen erre visszatérünk. A Z bit természetesen az akkumulátor tartalmának megfelelően áll be.

Ha már az aritmetikánál tartunk, említsük meg a

NEG

(NEGate accumulator) utasítást, amely előállítja az akkumulátorban lévő érték kettes komplementjét. Ha az utasítás végrehajtása előtt az akkumulátorban 0 volt, akkor C bit is nulla lesz, egyébként egy. A Z bit feladatának megfelelően áll be.

A processzornak néha a futó főprogram végrehajtásán kívül, más fontosabb rutinokat kell végrehajtania. Erre különböző perifériák kérhetik a CPU-t. Ilyenkor a processzor megszakítja a program végrehajtását, és kiszolgálja a megszakítást kérő eszközt. A kiszolgálást követően a CPU folytatja félbehagyott munkáját. Néha azonban nem lenne jó, ha programunkat megszakítaná egy ilyen igény, ezért ezt szoftverből letilthatjuk a

DI

(Disable Interrupt, megszakítás tiltás), engedélyezhetjük az

EI

(Enable Interrupt, megszakítás engedélyezés) utasításokkal. (Minden autostartos, 5-ös fejlécű program DI utasítással kezdődik.)

A Z80 háromfajta megszakítási módban képes dolgozni, ezeket az

IM 0

IM 1

IM 2

(Interrupt Mode, megszakítási mód) utasításokkal választhatjuk ki. Számunkra az IM 1 érdekes, mert az EXOS ebben a beállításban fut. Ennek lényege, hogy ha CPU egy megszakítási kérelmet érzékel a megfelelő lábán, akkor egy RST 38h utasítást hajt végre.

A megszakítást kiszolgáló rutinból a

RETI

(RETurn from Interrupt, visszatérés megszakításból) és a

RETN

(RETurn from Non maskable interrupt, visszatérés nem maszkolható megszakításból) tudunk kilépni. A két utasítás pontos jelentése számunkra (egyelőre) teljesen közömbös.

A kettes interrupt móddal (IM 2) kapcsolatos az

LD A,I

LD I,A

utasítások, melyeknél I-nek (Interrupt vektor regiszter) a címkézésben van szerepe.

Az akkumulátorba olvashatjuk a memóriafrissítő regiszter tartalmát az

LD A,R

utasítással. Az R regiszter tartalma olyan gyorsan változik, hogy egyfajta "véletlenszám-generátorként" lehet használni.

Nem csinál semmit a

NOP

(No Operation, nincs működés) utasítás. A processzor a semmit 1 mikroszekundum alatt végzi el.

NOP-okat hajt végre a processzor a

HALT

(HALT, megállás) utasítás hatására. A processzor megszakítás kérésig vagy törlésig várakozik ebben az állapotban.

Ezzel - sokak meglepedésére - kivégeztük az összes utasítást, így ettől a ponttól kezdve már az EXOS-ra koncentrálnunk!

Röviden meg kell ismernünk azt az eszközt, amellyel forrásszintű programjainkat megírjuk, lefordítjuk, belőjük. Az ENTERPRISE-ra eddig két ilyen rendszer készült, az egyik az ASMON (készítője Henrik Bendtsen), a másik a DEVPAC (Hisoft termék). Mindkét programnak vannak előnyei és hátrányai. Az ASMON 1.3 például mindenféle híresztelés ellenére nem tud korrekt relocálható fájlokat készíteni. A DEVPAC tud ilyet, ennek viszont nagyon gyenge az editora.

A sorozat leginkább az ASMON-ra támaszkodik. Ha már az áthelyezhető programokkal foglalkozunk, akkor tesztünk majd egy rövid kicuccanást a DEVPAC-ba.

Egyes felhasználóknak az ASMON SIMON néven van meg a két program lényege ugyanaz. A programokkal kapcsolatos verziószámokra egyébként nem lehet hagyatkozni, mert állítólag Henrik Bendtsen kiadta az ASMON forrásszintű listáját is, így aztán különféle verziók születtek ugyanazon a változatszámra.

Szeretnénk az ASMON-ról egy mellékletet kiadni, amely tartalmazna a program összes lehetőségét. Munkánkhoz addig is kell valami támpont, így most röviden ismerkedjünk meg az ASMON-nal.

Az ASMON tulajdonképpen három főmodulból áll, ezek a következők:

Editor: az editorban tudjuk forrásszintű (assembly-ben) leírni programunkat.

Assembler: az editorban megírt programot az assemblerrel tudjuk lefordítani. A fordítás két menetben történik. Az első menetben történik meg a szintaktikai ellenőrzés, és a szimbólumtábla felépítése. (Hogy mi az a szimbólum, és hogy miért kell hozzá tábla, a későbbiekben majd kiderül.) A második menetben elkészül a kód.

Debugger: a gépi kódú programot tudjuk futtatni, vizsgálni, tesztelni, egyszerűen belőni.

A gépi kódú program készítésekor elsőként be kell gépelnünk azt az editorba. Ezt az assembly szöveget nevezzük „forrásszintűnek”. Ha valamilyen szupertitkos programot írunk, akkor ügyelnünk kell arra, hogy ne lopja el senki a forráslistát. Tulajdonképpen ez a legértékesebb része munkánknak, erre kell a legjobban vigyáznunk.

Az forrásszintű listába ún. assembler direktívákat is kell tennünk. Ezek nem Z80, hanem a fordítást vezérlő utasítások. Sok egyéb mellett megadhatjuk a program fordításának kezdőcímét, feltételes fordítást végezhetünk stb. Munkánkat jelentősen megkönnyítik a címkék, szimbólumok használata, ezekkel sokkal kezelhetőbbé válik a forráslista. A JP utasításokba például nem memóriacímeket kell írunk, hanem címkeneveket.

Az így megírt assembly nyelvű programot lefordítjuk az assemblerrel. A fordítás során hibáüzeneteket kaphatunk, ezeket a hibákat korrigálnunk kell, mert a fordító egy csökönyös állat, de mindig neki van igazusa... A kijavított, hibátlanul fordítható forráslistát gyakran mentjük el!

Sikeres fordítás után előáll a kód, amely remélhetőleg azt csinálja, amit mi akarunk. A tesztelés történhet úgy, hogy egyszerűen ráugrunk a program elejére, és várjuk az eredményt. Az eredmény legtöbbször ugrabugráló képernyő, hörgő hang, érzéketlen billentyűzet stb., majd rendszerindítás formájában jelentkezik. Ettől nem kell megijedni, a gép a hibás programtól soha nem romlik el! Célszerűbb tehát, ha a programot kis rutinokként lépésenként lefuttatjuk, és figyeljük a viselkedését. Ha egy helyen nem azt csinálja, mint amit mi szeretnénk, akkor visszalépünk az editorba. Kijavítjuk, lefordítjuk, elmentjük a forráslistát, ezt követi az újabb nyomkövetés. Így előbb-utóbb elkészül, használhatóvá válik programunk. (folytatjuk)

—HCs—

Fizessen elő a

Hobby Elektronika és a Rádiótechnika

folyóiratokra! Így biztosan mindig hozzájut!

A cím: 1387 Budapest, Pf. 603. Tel.: 117 - 0262

A szerkesztőségben regisztrált HE előfizetőknek ingyenes nyák-film melléklet.

A PASCAL

A logikai típus

A logikai (BOOLEAN) típus mindössze két értékkel rendelkezik: igaz (TRUE) és hamis (FALSE). Az összehasonlítások (relációk), feltételvizsgálatok eredménye logikai érték.

A karakter típus

A karakter (CHAR) típus egyetlen karakter tárolására alkalmas. Gépi megvalósításához többnyire egyetlen bajtot használnak fel.

A felsorolt négy típust - tehát az egész, a valós, a logikai és a karakter típust - összefoglaló néven *skalár* típusként emlegetik, mivel mindegyikük csak egy-egy értéket vehet fel.

A valós típust kivéve, mindegyiküket *sorszámozott* típusnak is nevezik, mivel lehetséges értékeik halmaza megszámlálható.

A string típus

A szabványos Pascal - sajnos - nem ismeri az igen sokoldalú *string* (magyarul *fűzér*) típust, helyette a karaktertömböt lehet használni. A különböző implementációk némileg eltérő módon, de általában mégis megvalósítják a string típust.

A résztartomány típus

Nem minden alkalommal van szűkégünk egy-egy sorszámozott típus teljes értéktartományára. Egy tömbindex például nem vehet fel tetszőleges értékeket, csak azokat, amelyekhez tömbelem tartozik, különben használatával "túlindexelünk" a tömb határain, ami esetleg beláthatatlan következményekkel járhat. A hónap napjait leíró egész érték sohasem lehet nagyobb 31-nél. A csak nagybetűk tárolására szánt karakterváltozónak nem kell tudnia kisbetűket fogadnia. Sokszor kimondottnak hasznos, ha az értéktartomány túllépésére a fordítóprogram figyelmeztet. Erre a célra vezették be a résztartomány típust. Legjobb, ha pár példán át nézzük:

```
CONST
  TOMBMERET = 1000;
TYPE
  HONAPOK = 1 .. 12;      { a hónapok }
  NAPOK   = 1 .. 31;     { a hónap napjai }
  INDEX   = 0 .. TOMBMERET; { a TOMBMERET-et
                             korábban konstansként
                             deklaráltuk }
  NAGYBETU = 'A' .. 'Z';
  KISBETU  = 'a' .. 'z';
  SZAMJEGY = '0' .. '9';
```

Ha most egy változónk, amelyet NAPOK típusúnak deklarálunk, tévedésből 0, 13 vagy -49 értéket kapna, a Pascal rendszer hibát jelezne.

A felsorolási típus

Sok esetben használunk olyan típust, amely néhány diszkrét értéket vehet fel. A BASIC-ben ilyenkor az egész (vagy annak híján a valós) típust kell használni. A Pascal megengedi, hogy felsoroljuk a típus lehetséges értékeit, és később ezeket az értékeket használjuk. Itt is legjobb a példa:

```
TYPE
  HET_NAPJA = ( Hétfő, Kedd, Szerda, Csütörtök,
               Péntek, Szombat, Vasárnap );
  HONAP_NEVE = ( Január, Február, Március,
                Április, Május, Június,
                Július, Augusztus, Szeptember,
                Október, November, December );
  PROCESSZOR_ALLAPOT = ( RUN, WAIT, DMA, IT );
  IT_ALLAPOT = ( ENABLE, DISABLE );
  SZINEK = ( vörös, narancs, sárga, zöld,
            zöldeskék, kék, ibolya );
```

Ha a NAP változót HET_NAPJA típusúnak deklarálunk, később a programban jogos a

```
NAP := Kedd;
```

értékkadás. Ha a PROC_STATE változó PROCESZOR_ALLAPOT típusú, az IT_FLAG pedig IT_ALLAPOT típusú, felírható az alábbi összetett utasítás:

```
IF IT_FLAG = ENABLE THEN
  PROC_STATE := IT;
```

Természetesen a felsorolási típusból is képezhetünk résztartomány típust:

```
TYPE
  HEKOZNAP = Hétfő .. Péntek;
  NYARI_HONAP = Június .. Augusztus;
```

Összetett típusok

A tömb

Az összetett típusok egyik fajta a *tömb* (ARRAY), ez adott mennyiségű azonos típusú elemből áll. A két-dimenziós tömb felfogható úgy is, mint egy egydimenziós tömbökből álló tömb. A tömbindex indulhat 1-től, 0-tól, de indulhat tetszőleges pozitív vagy negatív szám-tól is. Ennek jelentősége később nyilvánvaló lesz.

Figyeljük meg, hogy a tömb mérete meghatározható akár a megfelelő résztartomány típus megadásával, akár pedig az alsó és a felső indexhatár megnevezésével (azaz a példabeli egész és valós tömb azonos méretű lesz). Az is lényeges, hogy a tömbök dimenziójának száma gyakorlatilag tetszőleges lehet, azt csak a memória mérete korlátozza.

```

TYPE
INDEX      = 0 .. TOMBMERET;
EGESZ_TOMB = ARRAY[ INDEX ] OF INTEGER;
VALOS_TOMB = ARRAY[ 0 .. TOMBMERET ] OF REAL;

SOR        = ARRAY[ 1 .. 80 ] OF CHAR;
           { egy képernyősor }
KEPERNYO   = ARRAY[ 1 .. 24 ] OF SOR;
           { a képernyő 24 sora }

PIXEL = 0 .. MAX_SZIN; { a szín-módtól függ }
KEP    = ARRAY[ 0 .. 1279 ] OF ARRAY[ 0 .. 719 ] OF PIXEL;

```

Az utolsó sor felírható egyszerűbben is:

```
KEP = ARRAY[ 0 .. 1279, 0 .. 719 ] OF PIXEL;
```

(bár egy ekkora tömb nem valószínű, hogy elfér a memóriában.

A rekord

Sok objektum nem írható le egyetlen mennyiséggel. A dátum évből, hónapból és nappól áll, egy képpontot a koordinátái és a színe jellemezhet. A Pascal lehetőséget ad arra, hogy az ilyen összetett jellemzőkhöz önálló típust hozzunk létre összetevőiből. Ez a *rekord* típus.

```

TYPE
DATUM = RECORD
    EV : 1900 .. 2100;
    HO : HONAPOK; { Korábban deklaráltuk }
    NAP : NAPOK; { mindkettőt }
END;

NEVTIPUS : ARRAY[ 1 .. 25 ] OF CHAR;

SZEMELYNEV = RECORD
    VEZETEKNEV : NEVTIPUS;
    KERESZTNEV : NEVTIPUS;
END;

CIM = RECORD
    VAROS : ARRAY[ 1 .. 20 ] OF CHAR;
    UTCA : ARRAY[ 1 .. 32 ] OF CHAR;
    HAZSZAM : ARRAY[ 1 .. 16 ] OF CHAR;
    IR_SZAM : 1000 .. 9999;
END;

SZEMELYI_KARTON = RECORD
    NEV : SZEMELYNEV;
    SZULETETT : DATUM;
    LAKCIM : CIM;
    MEGJEGYZES :
        ARRAY[ 1 .. 4, 1 .. 80 ]
        OF CHAR;
END { of RECORD };

```

```

NYILVANTARTAS = ARRAY[ 1 .. 1000 ]
                OF SZEMELYI_KARTON;

```

Láthatjuk, hogy a rekord típusból is létrehozhatunk tömböket, ugyanúgy, mint a skalár típusból. A rekord típussal sok művelet (értékkadás, kiírás fájlba és visszaolvasás stb.) közvetlenül végezhető; más műveletek (pl. összehasonlítás) az egyes elemein egyenként. Ha egy ilyen nyilvántartást Basic-ben akarnánk megvalósítani, vagy külön-külön tömbben kellene kezelni a rekord típus egyes elemeit, ekkor egy elem kiemelése a tömbből esetleg féltucatnyi vagy még sokkal több értékkadást jelentene; vagy az egész adatszerkezetet össze kellene "csomagolni" egy nagy stringbe, akkor állandó gond lenne a ki- és becsomagolás. Nem is beszélve arról, hogy ha később meg kell változtatnunk a rekordszerkezetet (mondjuk, utólag ki akarjuk egészíteni az adatokat a foglalkozással), akkor szinte az egész programot át kellene írunk, míg itt csak egy deklaráción múlik az egész.

A rekord típus további trükköket is megenged. Előfordulhat, hogy a rekordban rögzítendő adatok szerkezete eltérő. Erre az esetre vezették be a változó rekord fogalmát. Ebben egy bizonyos elem határozza meg, hogy az utána következő rekordrész milyen szerkezetű legyen.

```

TYPE
... { itt szerepel a CIM típus
      deklarációja az előbbi példából }

NYILV_SZAM = ARRAY[ NY_NY_SZ_H ] OF CHAR;
           { a nyugdíjnyilvántartó szám
             hosszúságát (NY_NY_SZ_H)
             konstansként kellett megadni }

MUNKAHELY = RECORD
    VALLALAT : ARRAY[ 1 .. 40 ] OF CHAR;
    V_CIM : CIM;
END;

KARTON = RECORD
    NEV : NEVTIPUS;
    CASE FOGLALKOZAS : ARRAY[ 1 .. 32 ] OF CHAR
    OF
        'Nyugdíjas' : ( NYILV_SZAM : ARRAY[ 1 .. 12 ]
                       OF CHAR );
        'Tanuló' : ( ISKOLA : CIM );
        'Egyéb' : ( MH : MUNKAHELY );
    END; { RECORD }

```

Itt a KARTON típusú rekord FOGLALKOZAS eleme mondja meg, hogy milyen legyen a rekord további szerkezete: nyugdíjasoknál a nyugdíjnyilvántartó számot (ez egy adott hosszúságú karaktersorozat), tanuló esetén az iskola címét (ez egy korábban deklarált típusú rekord), egyéb esetben a munkahely nevét és címét tároljuk ugyanabban a rekordelemben. A fordítóprogram ilyen esetben az előforduló leghosszabb változó rekordelemnek foglal le helyet, az összes többi eset így garantáltan elfér.

Lehetőségek Páratlan Tárháza (LPT) 3.

Az előző két részben megismerkedtünk az LPT felépítésével. A színvezérlőbitek használatát nem részleteztem, ezért most ez következik.

Egy LPB-ben négy színvezérlőbit található, ezek az ALTIND0, ALTIND1, LS BALT és MS BALT nevet viselik. Az EXOS leírással ellentétben mindegyik színvezérlőbit mind karakteres, mind grafikus üzemmódban használható. Most nézzük meg részletesen az egyes bitek jelentését!

Grafikus üzemmódban ha az ALTIND0 beállított, a videomemóriában lévő bájtok hetedik bite dönti el, hogy a bájt által meghatározott pontok milyen színek legyenek. Ha a 7. bit a bájtban beállított állapotú, akkor a 0. és 1. palettaszínek helyett a 2. és 3. palettaszíneket használja a gép.

Ha az ALTIND1 be van állítva az LPB-ben, a kijelző bájti 0. bite határozza meg a pontok színét. Ha a bájt 0. bite beállított állapotú, akkor a 0-3 palettaszínek helyett a 4-7 palettaszínek jutnak érvényre. Megjegyzendő, hogy a kijelzésre kerülő bájt 0. illetve 7. bite kijelzéskor olyan, mintha törölt állapotú lenne.

Ha az LS BALT beállított állapotú, a kijelző bájti 7. bite határozza meg, hogy a palettaszínek közül a 0-1 vagy a 2-3 legyen használva.

Ha az MS BALT bit értéke 1, a kijelző bájti 6. bite alapján a pontok színe a 0-3 palettaszínek, vagy a 4-7 palettaszínek közül választódik ki. E két vezérlőbitnél a kijelzéskor a 7. illetve 6. bit beállítottnak számít.

Természetesen 4 színű grafikus üzemmódban az ALTIND1 és MS BALT használata értelmetlen, 16- vagy 256-színűben pedig egyik vezérlőbitet sem szükséges használni.

Karakteres üzemmódokban a vezérlőbitek működése ugyanaz, mint grafikus módban, a különbség csupán annyi, hogy az ALTIND0 és ALTIND1 bitek a karakterkép (ahová a másodlagos video adatcím mutat) bájtjai határozzák meg a színeket, ezért ezek használata nem szerencsés. Az LS BALT és MS BALT bitek pedig a karakterkódok alapján fejtik ki hatásukat.

Az LPT ismertetésének végén még egy hibát szeretnék orvosolni, amely az előző részben követtem el: attribútum módban az LD1 és LD2 pozíciók jelentését felcseréltem. Helyesen tehát az LD1 mutat az attribútumterületre (színterület), az LD2 pedig a bittérképre (képpontterület).

Most pedig - miután megismerkedtünk az LPT felépítésével - elkezdhetjük az ismerkedést a használatával. Elsőként egy gyors scrollozási lehetőséget mutatok be. A scroll rutin assembly listája ASMON-nal fordítható. A kommentek alapján a működése könnyen nyomonkövethető. A megoldás lényege a következő:

A jobbról-balra scrollozást az LPB-k video adatcímeinek növelésével érhetjük el, így az LPB-k által meghatározott pixelsorok egy bájtal balra tolnak. A pixelsorok utolsó bájtjaira a kifrás alatt lévő karakter következő bájtoszlopát kell beírunk, vagy ha új karakter következik, akkor annak első bájtoszlopát. Scrollozás közben minden eltoláskor a videomemória egy bájtal hátrébb kerül a memóriában, ezért ha ezt sokáig így folytatnánk, előbb-utóbb a teljes memóriát felülrílnánk. Ezért egy pixelsor hosszának megfelelő számú scrollozást követően az utolsó pixelsor memóriaterületét átmásoljuk a videomemória legelejére, és a pixelsorhoz tartozó LPB videoadatcímét ide állítjuk. Ez a scrollozási módszer lényegesen gyorsabb annál, mintha mindig az egész videomemóriát mozgatnánk, és bármilyen grafikus videomódú LPT-hez használható akkor, ha az egyes LPB-k csak egy pixelsort definiálnak (tehát minden LPB első bájtja OFFh értékű). A közölt rutin is csak ilyen LPT-vel használható. A rutin hívásakor a megfelelő szegmenseket (LPT és videomemória szegmensét) be kell lapozni. Ahhoz, hogy a kép ne villogjon, a rutint videomegskafálásból érdemes meghívni, ezzel a sebességét is állandósíthatjuk.

(folytatjuk)
-DEVIL-

```

.RADIX 10H
LPTADDR EQU 0 :LPT Z80-AS CIME
LPTLEN EQU 0 :LPB-K SZAMA
VIDMEM EQU 0 :VIDEOMEMORIA NICK-CIME
VIDMEM2 EQU 0 :VIDEOMEMORIA Z80-AS CIME
LINELEN EQU 0 :PIXELSOR HOSSZA BAJTOKBAN
CHRADDR EQU 0 :KARAKTERKESZLET Z80-AS CIME
CHARSX EQU 0 :KARAKTERSZELESSEG
OFFSET EQU VIDMEM2-VIDMEM
ENDMEM EQU VIDMEM2+LPTLEN*LINELEN
SCROLL LD A,(CHRCOUN) :KARAKTERSZELESSEG
DEC A :SZAMLALO CSOKKENTESE
JP NZ,NOENDCH :HA A KARAKTER KESZ
LD HL,(TXTADDR) :AKKOR A KOVETKEZO
INC HL :KARAKTER JON
LD (TXTADDR),HL :SZOVEGMUTATO NOVELESE
LD A,(TXTCOUN) :SZOVEGSZAMLALO
DEC A :CSOKKENTESE
JP NZ,NOENDTX :HA SZOVEG VEGE
LD HL,TEXT :AKKOR ELOLROL
LD (TXTADDR),HL :KELL KEZDENI
LD A,LENTEXT :SZOVEG HOSSZA
NOENDTX LD (TXTCOUN),A
LD A,(HL) :KARAKTERKOD BEOLVASASA
SUB 20 :VEZERLOKODOK ELHAGYASA
LD DE,CHARSX*LPTLEN
LD HL,0
LD B,8
STRTMUL ADD HL,HL :KARAKTERKOD SZORZASA
RLA :EGY KARAKTER
JR NC,NOMUL :MEMORIAIGENYVEL
NOMUL ADD HL,DE
DJNZ,STRTMUL
LD DE,CHRADDR :KARAKTERKESZLET KEZDO-
ADD HL,DE :CIMENEK HOZZAADASA
LD (ACTADDR),HL :ES LETAROLASA
LD A,CHARSX :KARAKTERSZELESSEG
LD (CHRCOUN),A :SZAMLALO FELTOLTESE
LD A,(SCRCOUN) :SCROLL SZAMLALO
DEC A :HA NULLA LETT AKKOR
JP NZ,NOENDME :VEGE A MEMORIANAK
LD HL,ENDMEM :UTOLSO PIXELSOR
LD DE,VIDMEM2 :MASOLASA A
LD BC,LINELEN :VIDEOMEMORIA
LDIR :ELEJERE
LD DE,VIDMEM :A PIXELSORHOZ
LD HL,(LPTCOUN) :TARTOZO LPB-BEN A
LD (HL),E :VIDEOCIM BEALLITASA
INC HL :A VIDEOMEMORIA
LD (HL),D :ELEJERE
LD BC,11 :EGYEL FELJEBB
OR A :LEVO PIXELSORHOZ
SBC HL,BC :TARTOZO LPB-CIM
LD (LPTCOUN),HL :KISZAMITASA
LD A,(LPBCOUN) :LPB-K SZAMLALOJANAK
DEC A :CSOKKENTESE
JP NZ,NOENDLP :HA VEGE AZ LPB-KNEK
LD A,LPTLEN :A LEGUTOLSORA KELL
LD HL,LPTADDR+LPTLEN*10-0CH :VISSZATERNI
LD (LPBCOUN),A
NOENDLP LD A,LINELEN
LD (SCRCOUN),A :AKTUALIS KARAKTERCIM
LD HL,(ACTADDR) :ELTAROLASA
PUSH HL :CIKLUSSZAMLALO
LD A,LPTLEN :EGY LPB HOSSZA
LD BC,10 :ELSO LPB CIME+4
LD HL,LPTADDR+4 :VIDEOCIM KIOLVASASA
LD E,(HL)
NEXTLPB INC HL
LD D,(HL)
INC DE :NOVELESE ES
LD (HL),D :VISSZAIRASA
DEC HL
LD (HL),E
PUSH HL :LPBCIM ELMENTESE
EX DE,HL :NICK-CIM ATSZAMOLASA
LD DE,LINELEN+OFFSET-1
ADD HL,DE :Z80-ASSA ES NOVELESE
EX DE,HL :A PIXELSOR HOSSZAVAL
LD HL,(ACTADDR) :A KARAKTER AKTUALIS
LDI :BAJTJANAK CIME ES A
LD DE,CHARSX-1 :BAJT ATMASOLASA
ADD HL,DE :AKTUALIS BAJT ALATTI
LD (ACTADDR),HL :LESZ AZ AKTUALIS
INC BC :LPB HOSSZANAK
POP HL :VISSZAALLITASA
NOCARRY ADD HL,BC :KOVETKEZO LPB CIME
DEC A :CIKLUSSZAMLALO
JP NZ,NEXTLPB :CSOKKENTESE
POP HL :KARAKTERBAJT MUTATO
INC HL :JOBBRA LEPTETESE A
LD (ACTADDR),HL :KARAKTERBEN
RET :VISSZATERES
ACTADDR DEFW 0
TXTADDR DEFW TEXT-1
SCRCOUN DEFB LINELEN+1
LPTCOUN DEFW LPTADDR+10*LPTLEN-0CH
LPBCOUN DEFB LPTLEN
CHRCOUN DEFB 1
TXTCOUN DEFB 1
TEXT DEFM "SCROLLANDO SZOVEG"
LENTEXT EQU $-TEXT

```


Hanoi tornyai

A buddhista szerzetesek időtlen időkig üldögéltek templomaikban, igyekezve megoldani a Hanoi torony néven közismertté vált feladványt, biztosan tudva, hogy amint ez sikerül valamelyiküknek, igen nagy mennyei boldogság lesz az osztályrésze.

A feladat lényege, hogy az arany rúdra 64 darab lyukas korong van feltéve, ezek mindegyike valamivel kisebb az alatta lévőnél. A 64 korongot át kell tenni az ezüst rúdra, de egyszerre csak egy korong mozgatható, és bármelyik korong mindig csak nála kisebb korongra tehető rá. Segítségképpen (egy mai szakember azt mondaná, pufferként) igénybe vehető a rézrúd.

A feladat igen egyszerűnek látszik. Néhány ügyes mozdulattal sikerülhet is átpakolnunk 3, vagy akár 4 korongot is a szabályoknak megfelelően az ezüstrúdra. Minél tovább megyünk azonban, annál kilátástalanabbnak tűnik a megoldás. Pedig nincs nagy vész, a 3 korongot 7 lépésben tudjuk átrakni, a 4 korongot 15 lépésben, az 5 koronghoz is csak 31 jól megválasztott lépés kell... Megállapíthatjuk, hogy n korong átrakásához $2^n - 1$ lépés szükséges.

Mi már tudjuk, hogy a szegény szerzetesek igen reménytelen feladatba kezdtek. A 64 korong átrakása ugyanis $2^{64} - 1$, azaz mintegy $2,8 \cdot 10^{19}$ lépés szükséges. Ez másként leírva

28 000 000 000 000 000 000, azaz 28 ezer trillió.

Ez olyan nagy szám, hogy elképzelni is lehetetlen. Másodpercenként 1 áthelyezést végezve éjjel-nappal, megállás nélkül, $0,78 \cdot 10^{16}$ óra, vagyis $3,24 \cdot 10^{14}$ nap, vagyis $8,88 \cdot 10^{11}$ év kell a megoldáshoz. Ez csak 10000, azaz egybillió (ezer milliárd) év. Jó lesz hát szietni!

Csak megjegyezzük, hogy véletlenül majdnem pont 28 ezer trillió búzaszemet kapott volna a monda szerint a sakkjáték feltalálója jutalmul (ennél éppen 1-gyel többet), ő a sakktábla minden mezéjére kétszer annyi szemet kért, mint az előzőre. Az ő jutalmát majdnem 3 milliárd, egyenként egymillió tonnás óriás tartályhajó tudná elvinni...

A program bemutatja a Hanoi torony feladat megoldásának egy részletét. A program lényege a TORONY nevű rekurzív (azaz önmagát újra és újra meghívni képes) eljárás, a többi csak díszlet. A program legfeljebb 13 korongot képes átpakolni, de ehhez több mint 2 óra szükséges.

UL & BG

```
100 PROGRAM "Hanoi_tornyai"
110 STRING K$(20)
120 NUMERIC K,L,M(3),POZ(3),COL(3)
130 LET K=13 ! Az oszlopok maximális magassaga
140 LET L=INT(K/2) ! A legnagyobb korong szélessége
150 LET POZ(1)=7-L:LET POZ(2)=20-L:LET POZ(3)=33-L
160 LET COL(0)=15 ! A korongok színe
170 LET COL(1)=11:LET COL(2)=3:LET COL(3)=7
! A pálcák színe
180 CALL GRAPH
190 CLEAR SCREEN
200 PRINT #101,AT 20,1:;
210 SET INK COL(1)
220 PRINT #101:" REZ ";
230 SET INK COL(2)
240 PRINT #101:" ARANY ";
250 SET INK COL(3)
260 PRINT #101:" EZUST ";
270 DO
280 INPUT AT 1,1:PROMPT "Hany korong van a rezoszlopon? ":N
```

```
290 LOOP WHILE N<1 OR N>K
300 CALL IDO(2*N-1)
310 LET M(1)=N:LET M(2),M(3)=0
! oszlopmagassagok (globalis változók)
320 FOR I=N TO 0 STEP-1
330 SET INK COL(1)
340 PRINT #101,AT(19-N+1),POZ(1):K$(0);
350 SET INK COL(2)
360 PRINT #101,AT(19-N+1),POZ(2):K$(0);
370 SET INK COL(3)
380 PRINT #101,AT(19-N+1),POZ(3):K$(0);
390 NEXT I
400 SET INK COL(0)
410 FOR I=N TO 1 STEP-1
420 PRINT #101,AT(19-N+I),POZ(1):K$(1);
430 NEXT I
440 CALL TORONY(N,1,2,3)
450 PRINT AT 3,1:2*N-1;"Lepessel megoldottam!"
";
460 PING
470 END
480 DEF TORONY(P,ELSO,MAS,HAR) ! Rekurzív eljárás
490 IF P>1 THEN CALL TORONY(P-1,ELSO,HAR,MAS)
500 CALL KORONG(P,ELSO,MAS)
510 IF P>1 THEN CALL TORONY(P-1,HAR,MAS,ELSO)
520 END DEF
530 DEF KORONG(Q,HONNAN,HOVA)
540 GET AS
550 IF AS="" THEN WAIT 1
560 ! Ha billentyűt nyomunk, gyorsabban rakosgat
570 SET INK COL(HONNAN)
580 PRINT #101,AT(20-M(HONNAN)),POZ(HONNAN):K$(0);
590 LET M(HONNAN)=M(HONNAN)-1
600 LET M(HOVA)=M(HOVA)+1
610 SET INK COL(0)
620 PRINT #101,AT(20-M(HOVA)),POZ(HOVA):K$(Q);
630 END DEF
640 DEF GRAPH
650 GRAPHICS ATTRIBUTE
660 SET CHARACTER 128,24,24,24,24,24,24,24,24,24
670 SET CHARACTER 129,0,15,15,15,15,15,15,15,15
680 SET CHARACTER 130,0,255,255,255,255,255,255
,255,255,255
690 SET CHARACTER 131,0,240,240,240,240,240
,240,240,240
700 LET K$(0)=" "&CHR$(128)&" "
710 FOR I=1 TO K
720 LET AS=""
730 FOR J=1 TO(K-I)/2
740 LET AS=AS&" "
750 NEXT J
760 IF MOD(I,2)=0 THEN
770 LET AS=AS&CHR$(129)
780 FOR J=1 TO I-1
790 LET AS=AS&CHR$(130)
800 NEXT J
810 LET AS=AS&CHR$(131)
820 ELSE
830 FOR J=1 TO I
840 LET AS=AS&CHR$(130)
850 NEXT J
860 END IF
870 FOR J=1 TO(K-I)/2
880 LET AS=AS&" "
890 NEXT J
900 LET K$(I)=AS
910 NEXT I
920 END DEF
930 DEF IDO(M)
940 NUMERIC E,H,N,O,P
950 LET P=INT(M/60):LET M=MOD(M,60)
960 LET O=INT(P/60):LET P=MOD(P,60)
970 LET N=INT(O/24):LET O=MOD(O,24)
980 LET E=INT(N/365)
990 PRINT AT 3,1:"Ehhez";
1000 IF E>0 THEN PRINT E:"Ev,";
1010 IF N>0 THEN PRINT N:"nap,";
1020 IF O>0 THEN PRINT O:"ora,";
1030 IF P>0 THEN PRINT P:"perc es";
1040 PRINT M;"masodperc szukseges!"
1050 END DEF
1060 END
```

Egy EnterSPRITE példaprogram és még valami

Már ismerjük a sprite tervezés menetét, a spritekezelő lehetőségeit, nincs más hátra, minthogy egy példaprogramot lássunk. A program 110-es és 120-as soraiban történik a sprite-ok betöltése. Ide tehát azokat a sprite neveket kell beírni, amelyeken rögzítettük azokat. Fontos tudnivaló, hogy elég csak egyszer betölteni a sprite-okat, azok a RUN vagy a START, de még a meleg reset hatására sem törlődnek. Tehát ha betöltöttük a sprite-okat, akkor a SLOAD utasításokat tartalmazó sorok elé nyugodtan kitehetjük a felkiáltójelet (REM).

A program semmi különlegesét nem csinál. A betöltést követően beállítja a sprite környezetet, bekapcsolja a HIRES 16 típusú képernyőt. Ekkor egy mozgó és egy időnként a mozgóra ugró sprite-ot látunk. A program egyébként azt akarja demonstrálni, hogy az ugráló sprite miért nem ugrik minden olyan esetben, amikor kellene neki.

Az állandó sprite akkor ugrik (ugrana) a másik sprite-ra, amikor X vagy Y pozíciójuk megegyezik. A program elindítását követően azonban azt tapasztaljuk, hogy jónéhány ilyen esetben nem történik semmi, az álló sprite nem hajlandó az ugrásra.

Mi lehet ennek az oka? A magyarázat egyszerű: a Basic programunk alatt futó spritekezelő sokkal gyorsabb, mint a főprogram. Így hiába egyezik meg a két sprite X vagy Y pozíciója, a Basic program ezt nem veszi észre, mert a pozícióegyezés pillanatában éppen mással foglalkozik.

A program jól példázza két, aszinkron módon futó program nehéz összehangolhatóságát. A spritekezelőt beállítjuk, a sprite-okat elindítjuk, és igazából már el is veszítettük felettük az uralmunkat, az animátor csak fut, és fut... A példában csak két sprite-ot kezelünk, és már egy egyszerű pozícióegyeztetés is komoly problémákat okoz. Ha a sprite-ok ütközését szeretnénk figyelni, akkor még ennél is több bajunk lenne, hiszen ott több irányból kellene pozíciótartományokat figyelni. És mi történne, ha mind a nyolc sprite-ot egyszerre akarunk kezelni..?

Végezetül ismerkedjünk meg a beharangozott új sprite tervezővel, SPRED-del!

```

100 PROGRAM "SPRINTA.BAS"
110 SLOAD 5,"SPR1"
120 SLOAD 6,"SPR2"
130 INIT :RANDOMIZE :SET 26,1
140 STRING EX$(0 TO 3)
150 LET EX$(0)="ne":LET EX$(1)="nw"
160 LET EX$(2)="se":LET EX$(3)="sw"
170 REM 5.sprite
180 ANIMATE 5,1,3,5,7:SPEEDAN 5,9
190 POSITION 5,36,72:SPEEDSPR 5,0
200 REM 6.sprite
210 ANIMATE 6,2,4,6,8:SPEEDAN 6,10
220 POSITION 6,71,0:SPEEDSPR 6,1
230 DIRECTION 6,"ne"
240 REM start
250 GRAPHICS HIRES 16
260 SET #102:SCROLL ON
270 ANIMON 5:ANIMON 6
280 SPRON 5:SPRON 6
290 WHEN EXCEPTION USE BUMMM
300 DO
310 IF XPOS(5)=XPOS(6) THEN
    CAUSE EXCEPTION 100
320 IF YPOS(5)=YPOS(6) THEN
    CAUSE EXCEPTION 100
330 LOOP
340 END WHEN
350 HANDLER BUMMM
360 IF EXTYPE=9229 THEN
370 INIT
380 END
390 END IF
400 IF EXTYPE=100 THEN
410 POSITION 5,XPOS(6),YPOS(6)
420 DIRECTION 6,EX$(RND(4))
430 RETRY
440 END IF
450 END HANDLER

```

A program működéséhez a SPRED és a SPRED.APL fájlra van szükségünk. A teljes egészében gépi kódú program betöltődése után a címlapján megadja a nevét, verziószámát. A szoftvert az Active Record készítette 1989-ben.

A program teljes egészében menüvezérelt, és ENTERPRISE-on eddig ez a legtisztességesebben megírt felhasználói interfész, amit idáig láttunk. Azért ennek is van néhány apró hiányossága, ezekre majd kitérünk.

A jelenleg igen elterjedt megoldásnak megfelelően a képernyő felső sorában láthatók a főmenü parancsai. A kiválasztás úgy történik, hogy a kiemlést a megfelelő menüpontra visszük, majd megnyomjuk... a szökező billentyűt. Jobban szerettük volna, ha a kiválasztás az [Enter]-rel (is) működne, hiszen majdnem minden menüvezérlésnél ez a megszokott. A szerzőnek nyilván az volt a szándéka, hogy külső botkormányról is teljes értékűen vezérelhető megoldást adjon. Ehhez viszont az kellett volna, hogy minden almenüben legyen egy CANCEL pont, amire ráélvén vissza tudunk menni az előző menübe. Ehelyett most mégis a billentyűzethez kell fordulnunk, az [Esc] gomb vezet vissza az almenü láncolatában. Azt viszont el kell mondanunk, hogy a visszalépés a program tetszőleges szintjéről lehetséges, sok más programtól eltérően itt ez korrektül van megcsinálva.

De lássuk a lényegét. A főmenüben a FUNCTIONS almenü választva bejutunk a fő szolgáltatást jelentő tervező képbe. A képernyő baloldalán egy nagy négyzetben tervezhetjük a figurákat. A sprite ugyanakkor jobboldalt fent eredeti nagyságban is látható. Kicsit lejjebb van a nyolc fázisnak megfelelő nyolc ablak, ide kerülnek majd a fázisrajzok. Amint egy fázis a helyére kerül, azonnal részt vesz az animációban, a jobb szélső ablakban "élni kezd".

Jobboldalt lent látható a színválaszték. Természetesen a palettát felülünknek megfelelően állíthatjuk össze, és a felső nyolc szint (BI-AS) is megválaszthatjuk. Ami kicsit nehézkes, az egy levedésből lerakott pont törlése: kilépni a rajztáblából, belépni a tintaszínválasztásba, kiválasztania háttérszint, visszalépni, be a rajztáblába, törölni a pontot, majd hasonló eljárással újra beállítani az eredeti tintaszínt... Könnyíti a helyzetet, hogy a színeket az [1]..[8], [Q]..[I] billentyűkkel közvetlenül elérhetjük. Beleférhetett volna a programba egy kapcsoló, amely a tinta, a radír és a felemelt toll állapot között választana.

A készülő fázisrajzzal gyakorlatilag minden transzfórmáció elvégezhetjük, beleértve a tükrözést, átfordítást, elforgatást, eltolást, színrotálást stb. A rajzot tetszőleges fázishelyre letehetjük, és tetszőleges fázist újra felvehetjük, illetve a fázisokat egymással tetszőleges kombinációban felcserélhetjük, kisebb és nagyobb kapacitású ideiglenes raktárba tehetjük..

Az OPTIONS menüből bármelyik EXOS vagy EXDOS változó értéke megváltoztatható, így például nem kell rettegni, hogy felbontjuk az alvó családokat, ha órákkal korábban elmulasztottuk kikapcsolni a magnóhangot. Ugyanitt állíthatók a program saját opciói is.

A FILE/DEVICE menüben beállíthatjuk, hogy magnóval vagy lemezegységgel kívánunk-e dolgozni, és itt tudjuk a munkánk eredményét elmenteni, vagy egy meglévő munkát betölteni. Itt esetleg felpanaszolhatnánk, hogy a rajzoláshoz "belőt" KEY_DELAY és KEY_RATE túlságosan gyors a fájlnev beírásához, jobb lenne ideiglenesen visszaállni a normál értékekre. A SPRED nagyon biztosan működő perifériakezeléssel rendelkezik.

A HELP_PAGES menü a program összes szolgáltatását felsorol, a képernyőn fel-le görgetve tanulmányozható helpet fr ki, sajnos, angolul. A propos: biztos, hogy a program külföldi sikerét elősegíti az angol feliratok alkalmazása, mi jobban szerettünk volna egy teljesen (!) magyarított változatot kipróbálni.

Nagyon tetszett, hogy a program "kiugrást" enged az operációs rendszerbe a DOS SHELL menüponttal. Ilyenkor tetszőleges EXOS vagy EXDOS parancsot kiadhatunk, például megnézhetjük a lemezkatlogust. Az [ESC] billentyű lenyomásával ezután visszatérhetünk a SPRED-be. Természetesen magnós konfigurációnál ez a funkció nem hívható.

Ha a munkát befejeztük, programból nem kell magunkat "kirobantani" a RESET gombbal, mint sok-sok "lelőhetetlen" program esetében; sőt, az EXIT/RESET menüponttal választhatunk, hogy hiedeg reset-et kérünk, illetve Basic-be vagy WP-be akarunk kilépni.

Ezeknek a "plusz" szolgáltatásoknak minden igényesebb programban meg kellene lenniük. A SPRED itt is magasra állította a mércét. Összefoglalva: a SPRED egy nagyon kezes, jól használható, teljesen EnterSPRITE kompatibilis sprite tervező program, amelyet nem nélkülözhet az, aki a programjait szellem-grafikával akarja színesíteni és életkélni.

(Megjegyzés: a SPRED megrendelhető lesz programküldő szolgáltatunknál.)

Rövid dallamok

Gyakran előfordul, hogy elkészült programunkat szeretnénk feldíszíteni hangokkal, rövid zenékkal. Bár az emberek többségének jó zenei hallása van, mégsem tud mindenki dallamokat DATA sorokba kódolni.

Rajtuk próbálnak segíteni a listákon látható programok, melyek elindítása után kellemes zene szűrődik ki a gépből.

Fóti Marcell

```

90 Set rem 1 OK
100 PROGRAM "MOZART.BAS"
110 ENVELOPE NUMBER 1;0,63,63,2;0,-
    10,-10,4;0,-40,-40,20
120 READ VALT
130 FOR CIKL=1 TO VALT
140 READ A,B
150 SOUND ENVELOPE 1,PITCH A,
    DURATION B
160 NEXT
170 DATA 70
180 DATA 54,6,53,6,53,12
190 DATA 54,6,53,6,53,12
200 DATA 54,6,53,6,53,12
210 DATA 61,24
220 DATA 61,6,60,6,58,12
230 DATA 58,6,56,6,54,12
240 DATA 54,6,53,6,51,36
250 !
260 DATA 53,6,51,6,51,12
270 DATA 53,6,51,6,51,12
280 DATA 53,6,51,6,51,12
290 DATA 60,24
300 DATA 60,6,58,6,57,12
310 DATA 57,6,54,6,53,12
320 DATA 53,6,51,6,49,36
330 !
340 DATA 61,6,60,6,60,12,63,12,57,
    12,60,12,58,12,53,24
350 DATA 61,6,60,6,60,12,63,12,57,
    12,60,12,58,12,53,24
360 DATA 59,6,58,6,58,12
370 DATA 59,6,58,6,58,12
380 DATA 59,6,58,6,58,12
390 DATA 66,18,63,6,62,18,65,6,58,
    18,62,6,63,36

100 PROGRAM "DVORAK.BAS"
110 ENVELOPE NUMBER 1;0,63,63,2;0,-
    10,-10,24;0,-40,-40,40
120 READ VALT
130 FOR CIKL=1 TO VALT
140 READ A,B
150 SOUND ENVELOPE 1,PITCH A,
    DURATION B
160 NEXT
170 DATA 25
180 DATA 20,40,22,20,23,20,22,30,20,10,20,40
190 DATA 20,40,18,20,15,10,18,10,20,60,20,20
200 DATA 20,40,22,20,23,20,22,30,20,10,20,40
210 DATA 20,20,23,7,20,7,23,7,27,20,15,20,20,80

100 PROGRAM "VANGELIS.BAS"
110 ENVELOPE NUMBER 1;0,63,63,2;0,-
    10,-10,24;0,-40,-40,40
120 READ VALT
130 FOR CIKL=1 TO VALT
140 READ A,B
150 SOUND ENVELOPE 1,PITCH A,
    DURATION B
160 NEXT
170 DATA 50
180 DATA 30,20,35,13,37,13,39,13,37,50,34,60
190 DATA 30,20,35,13,37,13,39,13,37,60
200 DATA 30,20,35,13,37,13,39,13,37,50,34,60
210 DATA 34,20,35,13,34,13,30,13,30,60
220 !
230 DATA 42,20,41,13,39,13,37,13
240 DATA 41,30,37,15,39,30,35,15,37,30
250 DATA 42,20,41,13,39,13,37,13,41,60
260 DATA 42,20,41,13,39,13,37,13
270 DATA 41,30,37,15,39,30,35,15,37,30
280 DATA 34,20,35,13,34,13,30,13,30,60

```

Nem hibás a DISKCOPY

Lapunk tavalyi második számában jelent meg a "Hibás a DISKCOPY" című cikk. Az ENTERPRISE képviselőlet hívta fel figyelmünket arra, hogy a CP/M 2.2 változatában a programnak csak 64 Kb-ot kell használnia, így tehát nem hibás e fontos segédprogram, csupán a Digital Research szabványa régies (a CP/M-el együtt).

Bennünk azonban továbbra is kételyek maradtak: alkalmazunk egy szabványos, ám a géphez alig illeszkedő programot, vagy legyen egy szabványba nem illeszkedő, de a központi memóriát teljes egészében kihasználó segédprogramunk?

Ne feledje!

A szív az élet motorja...

SZIV

1991. április 10 - 11.

MEGYEI MŰVELŐDÉSI
KÖZPONT

Székesfehérvár, Liszt F. u. 1.

Székesfehérvári
Informatikai
Vásár

Engedje, hogy bekapcsoljuk a vérkeringésbe!

Várjuk jelentkezését...

MÁTRIX Kft.

8000 Székesfehérvár,

Dózsa Gy. tér 10

Tel.: (22)12 - 619

Fax: (22) 11 - 585

MAHJONGG

Az Amigások, a C64 tulajdonosok és a PC-vel rendelkezők már jól ismerik, és bizonyára szeretik is ezt a logikai játékot. Az akciójátékok kedvelői is felüldülést találhatnak a Devilsoft első, de reméljük nem utolsó játékában. (Ehhez mi is hozzájárulhatunk, csak olvassuk el a legutóbbi ENTERPRESS második oldalán megjelent cikket, mely a nepperek kiszorításáról szól.)

Most pedig térjünk rá a játékra! Betöltés után a menü jelenik meg, ahol a program készítőjének nevén kívül azt is megtudhatjuk, hogy a grafika és a zene kapcsán JOVI-t kell szidni. A zene nem a legjobb, de a grafika első játék révén egész jó. De még van mit tanulni! A választható opciók a következők: [L] - egy már kimentett játékkállás betöltése, [S] - itt választhatjuk ki, hogy melyik szinten szeretnénk játszani. A szintek egyforma nehézségűek, 16384 közül választhatjuk ki a nekünk megfelelőt. A játék a tűzgomb illetve a [SPACE] lenyomására indul.

A játék közben egy újabb muzsika csendül fel. A képernyőn egy cserepekből álló piramist láthatunk felülnezetben. A játék célja a piramis lebontása. A dolgunkat nehezíti az, hogy minden csempét csak a párjával együtt vehetünk le. (Mindégyikből négy darab van összesen, és van négy-négy évszakot illetve virágot ábrázoló csempe is, ezek ugyanúgy pároknak számítanak.) Így még elég egyszerűnek tűnik a feladat, de tudnunk kell még, hogy nem mindegyik darabka vehető le, hanem értelemszerűen csak azok, melyek oldalról hozzáférhetőek. Ha nem az, akkor a program ezt közli velünk (Tile is blocked). A Mahjongg üzenetei még a következők lehetnek: "Tiles remaining: xxx" - itt az xxx helyén álló szám a még levételre váró csempék számát adja meg, "No more step" - ezt akkor közli a gép velünk, ha segítséget kérünk, viszont nincs több levehető csempe, valamint lehet még a "Congratulation" felirat is, de azt hiszem ezt nem kell magyaráznom.

A képernyő sarkaiban találhatóak még csempék. Ezek a "zene" ki- illetve bekapcsolására, segítség kérésre, játékkállás kimentésre, illetve új játék kezdésére szolgálnak. A help funkciót kétféleképpen is használhatjuk: ha csak simán rávisszük a kurzort, és megnyomjuk a tűzgombot, akkor felvilán két levehető csempe; ha viszont az egyik csempét már kiválasztottuk a tűzgomb segítségével, akkor az annak éppen levehető párját mutatja meg a program. A csempék levétele azt hiszem senkinek nem okoz majd problémát, hiszen a logikai játékoknál megszokott kezeléssel tehető meg.

Ha sikerült teljesíteni egy szintet, akkor a "Congratulation" felirat mellett egy kb. egyperces zene szólal meg. Ha megnyomjuk a tűzgombot, akkor a következő szint képe jelenik meg, ha viszont végighallgatjuk a muzsikát, a menübe kerülünk vissza. És még egy érdekesség: Ha meguntuk a játékot, és megnyomjuk a reset gombot, a program illedelmesen elbúcsúzik tőlünk ("Good by my friend!").

KING OF THE CASTLE

Nagy örömmel tölt el, hogy ebben az ENTERPRESS-ben két ENTERPRISE játékról is írhatok. Ez azt jelentené, hogy az utóbbi idők átiratai után a programozók végre új programokkal jelentkeznek, s nem a sok elnyűtt Spectrum programmal kell találkozoznunk játszódozásaink során? Talán. Igaz a KING OF THE CASTLE című játék nem a legfrissebb, hiszen ez a játék az ENTERPRISE magyarországi terjesztésének hajnalán is piacon volt. A játékot az Entersoft adta ki, mely a gépre programokat készítő cégek illetve "magánprogramozók" közül a legnagyobb. Talán nem is sokáig? Reménykedjünk a konkurencia megjelenésében. A játékot David Jones és Ed Hickman kódolta.

Egyszer volt, hol nem volt, élt egyszer egy Arthur nevű angol király, aki az öt körülvevő lovagok bátorságával és ügyességével szerzett magának dicsőséget és hatalmat. A lovagokat egy "Polygon Table" (Sokszög Asztal) nevezetű társaságba gyűjtötte. Aki a Sokszög Asztal tagja szeretne lenni, annak be kell bizonyítania bátorságát, ki kell érdemelnie a tagságot. A "Mágikus Lovagnak" is ez az álma már évek óta, így hát elment a királyhoz, hogy megtudja, mit kell tennie. Arthur őcséni így szólt hozzá: "Ha tagja szeretnél lenni a híres Sokszög Asztalnak, menj el Spriteföld mindenki által rettegett kastélyába, a Rettegés Fejedelmének Kastélyába. Hozz onnan annyi aranyat, amennyit csak tudsz. Ha elhozod nekem az évezredes Varázskardot (Excalibur), akkor ezen karddal foglak lovaggá ütni." A "Mágikus Lovag" tudta jól, hogy akit egyszer ezzel a karddal lovaggá ütnek, azt soha többé senki sem fogja tudni legyőzni. Hát így szólt szeretett királyához: "Uram királyom! Két kezettel vedd el az életem, ha a kardot nem hozom el a

kastélyból." - mondta, és mély hajlongások közepette kivonult a trónteremből. Jól tudta, hogy a "Rettegés Kastélyából" még soha senki nem tért vissza.

Itt kapcsolódunk be a játékba. Először a trónteremben állunk a nagy Arthur király előtt. A kastélyba egy teleport segítségével juthatunk el, mely a trónteremben található. A lovag irányítása a botkormánnyal, illetve a következő billentyűkkel lehetséges: [G] - Get - tárgy felvétele, [T] - Trade - tárgy eladása (csak ha kereskedőnél vagyunk), [D] - Drop - egy tárgy letétele, [E] - Examine - tárgy vizsgálata. A zenét az [F1]-el kapcsolhatjuk ki vagy be.

Ha már a király színe előtt állunk, még ne teleportáljunk, hanem nézzünk meg a kép jobb oldalát. Itt a már látott szobák számát, pénzünk mennyiségét, a nálunk lévő tárgyak értékét, erőnket és életeink számát láthatjuk. Kezdetben négy élet áll a rendelkezésünkre. (Nem hiába mágikus ez a lovag!)

Tehát a játék célját már tudjuk: a pénzgyűjtés után el kell húzni a csíkot a kastélyból. Először is a pénzgyűjtésről: a kastélyban található tárgyaknak megvan a saját értékük, melyet a kép jobb oldalán láthatunk - persze csak ha felvettük már - a legutóbbi valutaárfolyam szerint fontba átszámolva. Az alábbi tárgyakat vehetjük fel: 1.Bicska (Carving Knife) - £1,50 2.Vasjogar (Iron Mace) - £70,00 3.Szörös kutya (Hairy Dog) - £48,67 4.Ezüstserleg (Silver Goblet) - £99,99 5.Gyémántpor (Diamond Dust) - £114,49 6.Ragasztó (Tube of Glue) - £200,00 (Vigyázat! Ragad!) 7.Kovács (A Blacksmith) - £20,00 8.Salétromsav (Salpêtre) (Nem felvehető, mert mar!) 9.Bohóccor (Clown's Nose) - £2,95 10.Tálca (Serving Tray) - £6,30 11.Kristálygömb (Crystal Ball) - £4,87 12.Z80 Chip (A Z80 Chip) - £0,50 13.Aranynyúl (Golden Hare) - £60,21 14.Ónkorsó (A Pewter Mug) - £255,99 15.Sárkánytojás (Dragon's Egg) - £172,42 16.Holdsugár (Moon Beam) - £106,00 17.Mágikus láng (Magic Flame) - £68,01 18.Aranygyűrű (Gold Ring) - £255,99 19.Egy tál tej (Bowl of milk) - £4,44 20.Névnap üdvözlőlap (Birthday Card) - £5,00 21.Kalap (Top Hat) - £26,26 22.Holdkő (Moon Rock) - 75,00 23.Kommunikátor (Communicator) - £10,00 24.Óriásfigyő (Rock Python) - £1,50 25.5 Mbájtos merevlemez (5MB Hard Disk) - £250,00 26.Üres üveg (Empty Bottle) - £1,08 27. Ragadós kalács (Sticky Bun) - £150,00 (Vigyázat!Ragad!) 28.Egy tányér leves (Tin of Soup) - £2,50 29.Sléc (Pair of Skis) - £192,82 30.Gyertya (Unlit Candle) - £14,90 31.Piros ruha (Red Dress) - £14,95 32. Hajómodell (Modell Boat) - £20,00 33.Gyapjú (Rubber Fleece) - £87,49 34.Madárkalitka (Bird Cage) - £51,97 35.Kókuszdió (Coconut) - £0,16 36.Nyerreg (Saddle) - £138,11 37.Selyemköpeny (Silk Cloak) - £33,62 38.Az élet szikrája (Spark of Life) - £30,00 39.Égő fáklya (Burning Torch) - £15,00 40.Az udvari bolond sapkája (Jester's Hat) - £18,62 41.Napóra (Zinc Sundial) - £75,50 42.Üvegcipő (Glass Shoe) - £50,87 43.Sártenger (Pile of Mud) - £0,62 44.Varázskő (Sage's Stone) - £18,35 45.Ólomrúd (Lead Bar) - £27,52 46.Törött kard (Broken Sword) - £5,18

Az eredetileg nálunk lévő tárgyak a következők: 1.Sajt (Some Cheese) - £3,16 2.Olimpiai Medál (Olympic Medal) - £30,00 3.Maeskahús (Catmeat) - £3,00

A hátralevők pedig a kereskedőnél vannak: T1.Gordon lába (Gordon's foot) - £245,95 T2.Kén (Sulphur) - £23,95 T3.Teleport kulcs (Transmat key) - £200,00 T4.Szén (Charcol) - £11,86

A felsorolásból kimaradt, egyetlen tárgy megkeresését az olvasókra bízom, ez a "tárgy" egy egér (Mouse). Ki-ki találja meg maga, legálább nem lesz olyan unalmas a játék. (Itt van az asztalomon, éppen azt kattintgatom. A szerk.)

Néhány felvett tárgyból új tárgyat készíthetünk, így növelhetjük a tőzsdén jegyzett értékét:

Ólomrúd + Varázskő = Aranyrúd (Gold Bar)

Sártenger + Az élet szikrája = Sárszörny (Mud Monster)

Kovács + Törött kard = Varázskard (Excalibur)

Egér (Mouse) + Sajt = Kővér egér (Fat Mouse)

Üres üveg + Hajómodell = Palackozott hajó (Bottled Ship)

Mindent tudunk, ami a pénz gyűjtéséhez kell. Még meg kell szökni a kastélyból. Ehhez először is vigyük fel a ként illetve a szenet, és vigyük a salétromsavhoz, itt tegyük le. Ha most megvizsgáljuk az ott fekvő valamit, akkor láthatjuk, hogy a salétromsav helyén már puskapor (Gunpowder) van. Hozzuk ide a mágikus lángot (Magic Flame) és tegyük azt is le. Hatalmas robbanás után szabad az út kifelé, a macska alakjában megjelent Rettegés Fejedelme elhúzza a csíkot, gondolom a pokol felé. Ha kisétálunk, megérkezünk a "You have escaped" (Mégmenekültél) nevű szobába. Túléltek, itt a vége, fuss el véle! Ha valakinek még nem volt elege a játékból, akkor tovább mászkálhat, pénz gyűjthet. Aki pedig megunta, annak a hátul lévő piros gombot ajánlom a figyelmébe.

You have escaped	Big Pusa 10 8 9 without Boots	11 12 The Needlework room	13 14 The Dr Wombat Bedroom	The 15 16 Ice/Cream Factory
The trader's room 3, 2	The inner chamber 4	5 Teleport Castle teleport room	6 The Astrologer's study 1	Room 7 with a View

Cold Upper Maze

Slimet Lower Maze

Cold
Upper
Maze

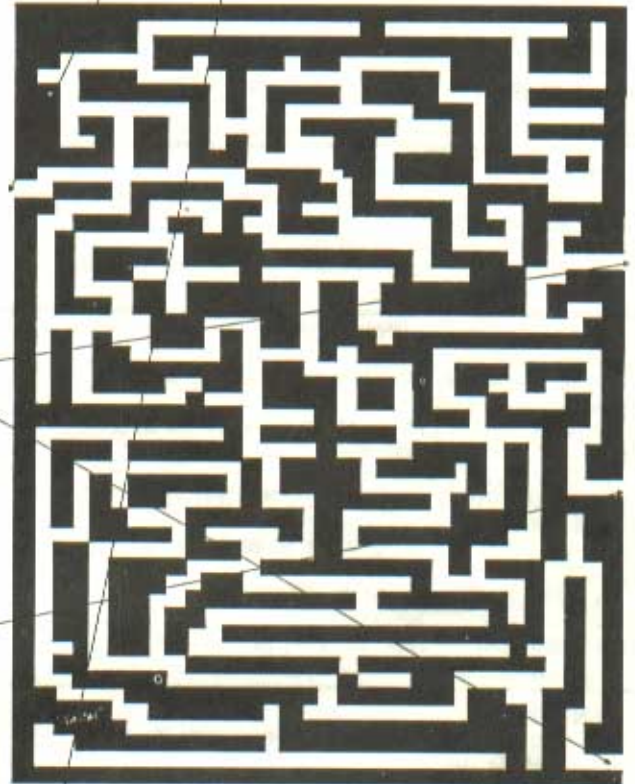
The Treasury 17 18	19 20 The King's art room	Crazy wizard's store 22 21
-----------------------	------------------------------	----------------------------------

Cold
Upper
Maze

24 Dr Wornbat's 23 Laboratory	The Blood 25 26 Bath room	Anna 27 the trader's room
----------------------------------	------------------------------	------------------------------

Cold
Upper
Maze

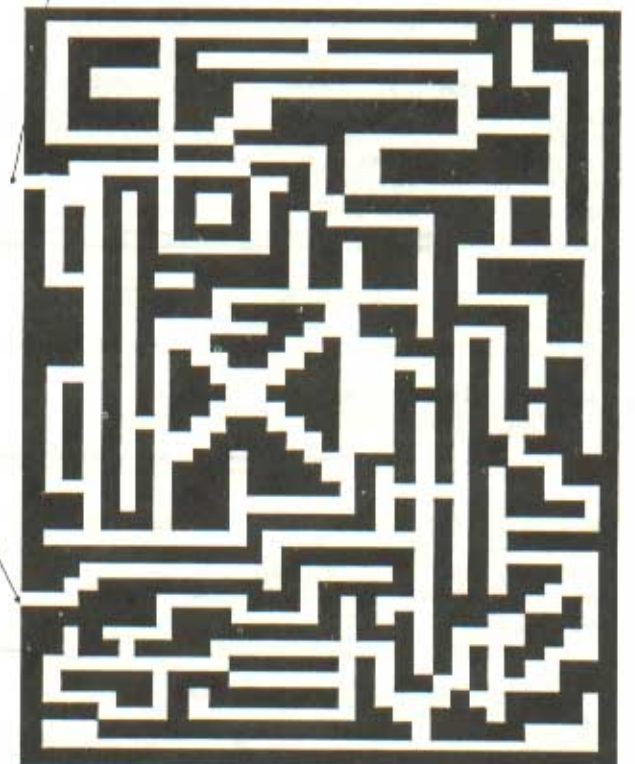
29 The 28 Bear Cellar	30 31 The Coldest room	32 The 33 King's Zun room
--------------------------	---------------------------	------------------------------



Cold Upper Maze

36 The Trans-mat 37 room	Anna Quay's 35 Bedroom	Gort — 34 the trader's room
-----------------------------	---------------------------	--------------------------------

Slimey
Lower
Maze



Slimey Lower Maze

**KING
OF THE
CASTLE**

ÖRÖKÉLETKÓDOK

KING OF THE CASTLE

(A második fájt kell betölteni!)

- [R] 1100 [ENTER] BFFF [ENTER] KING [ENTER]
- "Last address: 94D9"
- [M] 208A [ENTER] B7 [ESC]
- [M] 2B08 [ENTER] B7 [ESC]
- [M] 32EF [ENTER] B7 [ESC]
- [S] 1100 [ENTER] 94D9 [ENTER] KING [ENTER]

BOMB JACK

- [R] 10F0 [ENTER] BFFF [ENTER] BOMB JACK [ENTER]
- "Last address: 17FE"
- [M] 1694 [ENTER] 32 40 C3 [ESC]
- [S] 10F0 [ENTER] 17FE [ENTER] BOMB JACK [ENTER]

BLACK LAMP

- [R] 10F0 [ENTER] BFFF [ENTER] BLACK [ENTER]
- "Last address: 1939"
- [M] 11A7 [ENTER] F3 3E AF 32 6A 80 C3 00 80 [ESC]
- [S] 10F0 [ENTER] 1939 [ENTER] BLACK [ENTER]
- 720 DEGREES
- [R] 10F0 [ENTER] BFFF [ENTER] 720 [ENTER]
- "Last address: 12B0"
- [M] 12B1 [ENTER] AF 32 7E 9B 32 06 97 32 68 95 C3 00 5B [ESC]
- [M] 11E2 [ENTER] C3 B1 02 [ESC]
- [M] 10F2 [ENTER] BE [ESC]
- [S] 10F0 [ENTER] 12BD [ENTER] 720 [ENTER]

SPINDIZZY

- [R] 10F0 [ENTER] BFFF [ENTER] SPINDIZZY [ENTER]
- "Last address: 1BF2"
- [M] 1BF3 [ENTER] 3E C9 32 11 BD C3 00 60 [ESC]
- [M] 1253 [ENTER] C3 F3 0B [ESC]
- [M] 10F2 [ENTER] FB [ESC]
- [S] 10F0 [ENTER] 1BFA [ENTER] SPINDIZZY [ESC]

TRANTOR

- [R] 10F0 [ENTER] BFFF [ENTER] TRANTOR [ENTER]
- "Last address: 1A40"
- [M] 1A41 [ENTER] AF 32 61 C1 32 81 C1 C3 00 5B [ESC]
- [M] 11FA [ENTER] C3 41 0A [ESC]
- [M] 10F2 [ENTER] 4B [ESC]
- [S] 10F0 [ENTER] 1A4A [ENTER] TRANTOR [ENTER]

ROGUE

- [R] 10F0 [ENTER] BFFF [ENTER] ROGUE [ENTER]
- "Last address: 35CF"
- [M] 35CC [ENTER] AF 32 9A 78 32 9B 78 32 13 89 C3 11 02 [ESC]
- [M] 10F2 [ENTER] D9 [ESC]
- [S] 10F0 [ENTER] 35D8 [ENTER] ROGUE [ENTER]

BOUNTY BOB

- [R] 10F0 [ENTER] BFFF [ENTER] BOUNTY [ENTER]
- "Last address: 15CF"
- [M] 1236 [ENTER] C3 D0 05 [ESC]
- [M] 15D0 [ENTER] AF 3D 32 0B C7 C3 4F C2 [ESC]
- [M] 10F2 [ENTER] D8 [ESC]
- [S] 10F0 [ENTER] 15D7 [ENTER] BOUNTY [ENTER]

JOE BLADE

- [R] 10F0 [ENTER] BFFF [ENTER] JOE BLADE [ENTER]
- "Last address: 4165"
- [M] 4166 [ENTER] AF 32 A4 A7 32 68 7C 3E C9 32 A0 93 C3 00 60 [ESC]
- [M] 11BD [ENTER] C3 66 31 [ESC]
- [M] 10F2 [ENTER] 75 [ESC]
- [S] 10F0 [ENTER] 4174 [ENTER] JOE BLADE [ENTER]

AUTOMANIA

- [R] 10F0 [ENTER] BFFF [ENTER] AUTOMANI [ENTER]
- "Last address: 12B2"
- [M] 12B3 [ENTER] 3E B7 32 C8 FD C3 00 82 [ESC]
- [M] 12A4 [ENTER] C3 B3 02 [ESC]
- [M] 10F2 [ENTER] BB [ESC]
- [S] 10F0 [ENTER] 12BA [ENTER] AUTOMANI [ENTER]

EVERYONE'S A WALLY

- [R] 10F0 [ENTER] BFFF [ENTER] WALLY [ENTER]
- "Last address: 16D6"
- [M] 16D7 [ENTER] 21 00 00 22 66 E3 22 68 E3 22 6A E3 C3 84 81 [ESC]
- [M] 1257 [ENTER] C3 D7 06 [ESC]
- [M] 10F2 [ENTER] E6 [ESC]
- [S] 10F0 [ENTER] 16E5 [ENTER] WALLY [ENTER]

JET SET WILLY

- [R] 10F0 [ENTER] BFFF [ENTER] WILLY [ENTER]
- "Last address: 134C"
- [M] 1242 [ENTER] C3 4D 03 [ESC]
- [M] 134D [ENTER] AF 32 3B 8C 3E CA 32 C1 86 3E 87 32 C2 86 C3 CA 87 [ESC]
- [M] 10F2 [ENTER] 5E [ESC]
- [S] 10F0 [ENTER] 135D [ENTER] WILLY [ENTER]

HEAD OVER HEELS

- [R] 10F0 [ENTER] BFFF [ENTER] HEADOVER [ENTER]
- "Last address: 14FF"
- [M] 12FF [ENTER] C3 00 05 [ESC]
- [M] 1500 [ENTER] AF 32 D3 A4 32 FD 89 32 7C A8 3E EB 32 7E A8 C3 36 70 [ESC]
- [M] 10F2 [ENTER] 12 [ESC]
- [S] 10F0 [ENTER] 1511 [ENTER] HEADOVER [ENTER]

CAULDRON II

- [R] 10F0 [ENTER] BFFF [ENTER] CAULDRON.II [ENTER]
- "Last address: 1234"
- [M] 1191 [ENTER] C3 35 02 [ESC]
- [M] 1235 [ENTER] AF 32 EE CE C3 00 FE [ESC]
- [M] 10F2 [ENTER] 3C [ESC]
- [S] 10F0 [ENTER] 123B [ENTER] CAULDRON.II [ENTER]

TRANZAM

- [R] 10F0 [ENTER] BFFF [ENTER] TRANZAM [ENTER]
- "Last address: 12DF"
- [M] 11DA [ENTER] C3 E0 02 [ESC]
- [M] 12E0 [ENTER] AF 32 65 63 3E C3 32 66 63 C3 00 5F [ESC]
- [M] 10F2 [ENTER] EC [ESC]
- [S] 10F0 [ENTER] 12EB [ENTER] TRANZAM [ENTER]

És végül néhány CHEAT lehetőség. Betöltés közben, a játék elindulásáig nyomjuk le a következő billentyűket:

- SUPER ROBIN HOOD [D]
- GUTZ [D]
- STAR RAIDER II [ESC]
- REDHEAT [A]+[S]+[D]
- KARNOV [ESC]+[TAB]+[CTRL]+[LOCK]+[LSHIFT]+[F1]
- GHOULS 'N GHOSTS [A]+[S]+[G] majd játék közben az [INS]-el ki-, vagy bekapcsolhatjuk a sérthetetlenséget.

ENTERPRISE képviselőt nyit Budapest

A képviselőt megletéről szerkesztőségünk már tavaly augusztusban tudomást szerzett, hiszen a képviselőt emberei hallva az ENTERPRESS készülődéséről, azonnal felkerestek bennünket. Olvasóinkat erről azért nem tájékoztattuk eddig, mert az augusztusi találkozónk csak egy futó egyeztetés volt, igazából konkrét információkat még nem tartalmazott.

Sajnos rögtön a cikk legelején el kell árulnunk, hogy még mindig nincs itt az "ENTERPRISE kánaán". A VTGe Elektronikai Kft. - az ENTERPRISE Computers GmbH magyarországi képviselője - nem üzletet, hanem csak egy irodahelyiséget nyitott Budaörsön. Hogy pontosan mivel is foglalkoznak, arra sok egyéb kérdésünk mellett Kopácsy Vilmos ügyvezető igazgató válaszolt.

- A Kft. mostanában leginkább a gépek Szovjetunióbeli elterjesztésével foglalkozik - mondja Kopácsy úr - de az ottani belpolitikai helyzet miatt ez kissé vontatottan halad. A szovjet partner a gép liszenszét szeretné megvásárolni, a gépet pedig ottani alkatrészekből akarja összerakni. Úgy képzeli, hogy a gyártásnál mi csak a háta mögött állnánk. Még a Dave-et és a Nick-et is ő akarja az alaplapra tenni.

Hány gép van már a Szovjetunióban?

- Pontosan nem tudom, de úgy háromezer körül lehet a számuk. A gépeket hálózatba kapcsolva kazahsztáni iskolákban használják, és a gépek összerakása is itt folya. Nagyobb érdeklődés mutatkozik az ENTERPRISE PC-k iránt.

Mitől lesz ENTERPRISE egy PC?

- A PC-k alaplapját az ENTERPRISE cég tervezte, és az alaplapokat Németországban készítik. Kínálatunk az XT termináltól kezdve egészen a 486-osig tart.

Térjünk vissza a gépünkre! Úgy hallottuk, hogy Egyiptomban is van egy nagyobb adag belőle.

- Igen, ott a 64K-s változatból kelt el négyezer darab.

Milyen új szoftvereket várhatnak a hazai felhasználók?

- Az "a" Studio programozói pillanatnyilag 15 játékprogramon dolgoznak, a Centrum megbízásából pedig 10 ilyen készül.

És a játékprogramokon kívül készülnek-e más programok?

- Most dolgozunk egy feliratozó szoftveren, amelyhez egy ún. genlocker kártya is kell. Ez utóbbinak az a feladata, hogy a kamera és a számítógép képeket korrekt módon keverhetők legyenek. A számítógép és egy szintetizátor összekapcsolásához szükséges Midi interfész és szoftver fejlesztése is folyik.

Milyen újdonságok vannak még?

- A 128K-s alapgéphez ma már akár 160MB-os merevlemez is kapcsolható. Elkészült az az illesztőnk is, amellyel a PC-khez kapcsolható kártyák nagyrésze az alapgépbe dugható. Így akár VGA monitort is kapcsolhatunk a konfigurációnkhoz.

Mikor kezdik az új termékek árusítását?

- A szoftver és a hardverfejlesztés egyelőre nem rentábilis. Amíg a régi készleteink is megvannak, semmiképpen sem fogunk bele az árusításba.

Mik ezek a régi készletek?

- Főleg a 4000 darab Spectrum-emulátort és az SzJA'88 programot említhetném.

Mekkora ennek a készletnek az összértéke, és hol van?

- Összértékük ötvenmillió forint, és a Centrum raktáraiban vannak.

Úgy hallottuk, hogy a Novotrade Rt. gondozásában megjelent "EXOS ROM0 visszafertés" című könyvvel kapcsolatban valamilyen galiba volt...

- Igen, a Novotrade kinti megbízottját beperelte az EN-

TERPRISE GmbH. Ez a megbízott a harmadfokú tárgyalás után többszáz ezer márkát kártérítést fizetett az ENTERPRISE-nak. A ROM visszafertésének közléséhez ugyanis az ENTERPRISE cég engedélyre szücségeteltetett volna...

Megbízható forrásból tudjuk, hogy a könyv szerzője már elkészült a 1. szegmens visszafertésével is. Ezek szerint ez nem fog megjelenni... Az ENTERPRESS az olvasók kívánságának megfelelően szeretné a gép és az EXDOS kapcsolási rajzát közzélni. Mit szól ehhez?

- Nem értem, hogy egy amatőrnek miért van szüksége a gép kapcsolási rajzára! Ez a cég belső magánügye.

Talán így még inkább fellendülnének a fejlesztések...

- Aki kapcsolási rajzot akar szerezni a gépről, az már ezt bármelyik klubban megteheti. Egy ilyen nekem is adtak, pedig tudták, hogy ki vagyok. Az ENTERPRESS egyébként azt közöl amit csak akar, de vállalnia kell a következményeit is.

Hmm... Sok olvasónk panaszkol, hogy gondjai vannak az EP-PLUS kártyával, és hiába várnak választ az "a" Studio-tól.

- Az "a" Studio-tól a VTGe Kft. hamarosan átveszi a leveleket, és mindegyikre válaszolni fog.

Készül-e új ENTERPRISE modell?

- Igen, bár itt alapvetően az 128K továbbfejlesztéséről van szó. Az új modell külsőleg leginkább egy lapos PC-hez hasonlítható, külső billentyűzettel. Alaplapján lesz az EXDOS, be lesz építve a Midi interfész, és a hardveres hálózat kezelő. Azt, hogy mekkora központi memóriája lesz, majd az akkori árak fogják eldönteni.

Milyen processzorral és operációs rendszerrel lesz a gép felvértezve?

- Valószínűleg a Hitachi Super Z80-asa lesz a gépben. Ennek a processzornak 21 bites címbusza van, és tartalmazza a matematikai kooprocesszort is. A gép operációs rendszere pedig az EXOS 3.0-ás verziója lesz.

Ez mit tud?

- Mindazt amit az EXOS 2.1-es változat. Ezenkívül jellemzője, hogy felhasználói felülete csaknem teljesen grafikus, és a rendszer ikonvezérelt. A felület az EGI (ENTERPRISE Graphics Interface) névre hallgat, és leginkább az Atari GEM-re hasonlítható.

Szerkesztőségünk nem állhatja meg, hogy megjegyzéseit a cikk végére ne írja. Mint ahogyan azt Kopácsy úr is mondta, a fejlesztések nem hozzák meg a kellő bevételt. Így nem várhatjuk, hogy a VTGe Kft. csupán emberbaráti szeretetből elhalmoz bennünket jobbnál-jobb hardver és szoftver termékekkel. Az ENTERPRISE GmbH természetesen figyeli a magyarországi felhasználók sajátos helyzetét, de igazán döntő lépéseket nem tud tenni.

A beszélgetés során rengeteg szakmai kérdést tettünk fel Kopácsy Vilmosnak, aki - lévén, hogy kisujjában van a gép - készséggel válaszolt ezekre. Ha egyébként olyan olvasói kérdések érkeznek hozzánk, melyeket szerkesztőségünk nem tud megválaszolni, akkor azokat átadjuk a képviselőnek.

Az új, fejlesztés alatt álló ENTERPRISE modellel kapcsolatban van néhány felvetésünk, mindenféle rossz szándék nélkül. Bár nem tudjuk, hogy mikorra készül el a gép, de ha már most sorozatgyártásban lenne, szerintünk már akkor is elkésztetnek számítána. Nem hisszük, hogy a mind olcsóbbá váló PC-knek, Amigának, Atariknak komoly ellenfél lehet egy ilyen adottságú gép. A másik problémát pedig az új gépre nemlétező szoftverek okozzák majd.

Persze mi felhasználók már sokszor tapasztalhattuk, hogy az ENTERPRISE fronton nehéz bármit is előre jósolni. Amíg van időnk és türelmünk, addig várjuk ki ennek is a végét.

Postafiók 334.

Örömmel tapasztaljuk, hogy kezd megjönni Olvasóink programfró és -küldő kedve. Igaz, ez nem megy minden bonyodalom nélkül:

"Elkészíttem egy anyagot ENTERPRISE-om szövegszerkesztőjében, majd azt a KONVERT programjuk segítségével a megfelelő formátumúvá konvertáltam. Ezután pusztán kíváncsiságból az átalakított adatállományt visszatöltöttem a WP-be. Meglepődve tapasztaltam, hogy egy kicsit zűrzavaros lett az egész." írja Kreiner Attila hűséges budapesti olvasónk, és kérdezi, hogy jól van-e ez így.

Kedves Kreiner Attila, és mindenki, aki cikket vagy programleírást kíván küldeni! Az ENTERPRISE-on írt szöveget azért kell konvertálni, mert a WP által előállított szövegfórmátum semmilyen más szövegszerkesztőben sem használható. Bonyolítja a helyzetet, hogy az ékezetes betűket mindenki a saját ízlése szerint jelölheti ki a billentyűzeten, ezt a lehetőséget nem vehetjük el író olvasóinktól. A KONVERT program, amit a vállalkozó szellemű olvasóknak kérésükre megküldünk, az előállított szöveget olyan formátumúra alakítja át, amely már "ehető" az ENTERPRESS előállítását végző Ventura Publisher 2.0 kiadványszerkesztő program számára. Ezért senki ne lepődjön meg, ha az átalakított szöveg az ENTERPRISE-on értelmetlen krikz-kraksznak tűnik, így a jó. Viszont a szerző saját célra mindig tartson meg egy eredeti szöveget is, hiszen csak azt tudja olvasni, módosítani.

Tarl Gábor lórincai olvasónk az ÉTERFACE újság iránt érdeklődik. Sajnos, meg kell erősítenünk értesülését, hogy a lapból csak egy szám jelent meg. A kiadó nem is tervezi több szám megjelentetését. Azt hiszem, hogy ezt az egész ENTERPRISE-os közözség bánja, hiszen egy igazi profi színvonalú újságot kell elfelejtenünk, mielőtt igazán megismerhetjük volna. Feltevezzük, hogy a kiadó elkedvetlenedésének egyik oka a lapterjesztés környékén uralkodó nem igazán paradicsomi állapotok lehetnek, amivel mi, az ENTERPRESS szerkesztői és a kiadó is, nap-mint-nap megküzdünk. A mi lapunkat is keresik az ország számtalan kisebb és nagyobb helységében, míg máshol eladatlan hegyekben áll.

Ugyancsak Tarl Gábor olvasónk panasolja, hogy az ENTERPRESS 2. számában közölt *Hol a kurzor?* című írásban megjelent programban nem működik a WHEREX függvény. Sajnos, a "távgyógyítás" nem tartozik a legkönnyebb feladatok közé, de majdnem bizo-

nyos, hogy olvasónk valamit hibásan írt át a lapbeli listából. Ez annál is inkább bizonyos, mert az előbb meghívott - és a rakoncátlanoktól rutinnal majdnem azonos - WHEREY rutin olvasónknál is működik. Sajnos, mi nem tehetjük meg azt, amit gazdagabb (persze, ennek megfelelően drágább) laptársaink, elsősorban külföldön, megengedhetnek magunkak, azaz nem tehetjük be az újságba kezettán vagy Uram bocsá, lemezen a közölt programokat.

Szentgyörgyi László oroszlányi olvasónk írja: "...Én a gépet nem játéka akarom használni, hanem a gyakorlati életben szeretném alkalmazni. A mai helyzetben ennek csak egy lehetőségét látom: ha magam készítek a célnak megfelelő programokat. Az eddig beszerzett leírások, sajnos, számomra sokszor felfoghatatlanok... Szeretném, ha a cikkeket - a tartalom színvonalának csökkentése nélkül - az abszolút kezdőknek is érthető szintre tudnák hozni, vagy azokat gyakorlati példák segítségével megérteni... A másik nagy problémám, hogy a Centrum felhagyott a gép bármiféle patronálásával, ezért a következő kérdésem lenne: hol, hogyan és mennyiért lehet beszerezni programokat és tartozékokat? Erről is jó volna informálni lap olvasóit."

Olvasónk jól látja a lap feladatát és szerkesztőinek állandó dilemmáját: hogyan lehet a köz-érthetlenség és a színvonalatlanság közötti keskeny (nemlétező?) ösvényen haladni? Reméljük, hogy nekünk sikerül a lehetetlen, de erről csak Olvasóink levelei alapján tudunk véleményt alkotni. A másik felvetésével kapcsolatban az a személyes véleményünk, hogy egy kereskedő vállalatnak nem "patronálnia" kell az általa forgalmazott terméket, hanem állandó, színvonalas ellátást kell megvalósítania. Nem hiszem, hogy a lap terjedelme elég lenne az e téren tapasztalt esetelérésre... (Véletlen-e, hogy az "after sales support" kifejezésnek nincs meghonosodott magyar megfelelője?) Ugyanakkor tudomásunk van arról, hogy nagyértékű megmaradt készlet van ENTERPRISE tartozékokból, ennek forgalomba kerülésére jó esélyek vannak. (I. a VTGe Kft-ről készült riportunkat a 15. oldalon. A szerk.)

Több olvasónk kérésére válaszolva: Az ENTERPRISE gép és kiemelő elemének kapcsolási rajzához, mint már korábban is említettük, szerzői jogok fűződnek, így a rajzok közzélése csak a jogi kérdések tisztázása után lehetséges, addig szíves türelmüket kérjük.

A MIKROVILÁG

márciustól két oldalt szentel

az ENTERPRISE felhasználóknak.

mikrovilág

Az ENTERPRESS előző számai korlátozott példányszámban még megrendelhetők a kiadó címén (MÁTRIX Kft. 8000 Székesfehérvár, Dózsa Gy. tér 10.), vagy megvásárolhatók a Műszaki Könyvruházban (Bp. VI. ker. Liszt F. tér 9.) és a FÓKUSZ Könyvruházban (Bp. VII. ker. Rákóczi út 14).

Apróhirdetések

Eladó memóriabővítő EXDOS lemezillesztő 8500 -, 3,5" -os 720 KB -os új lemezegység 7500 Ft -ért. Bozai Gábor, 8000 Székesfehérvár, Budai út 92. fsz. 5.

ENTERPRISE-hoz különböző bővítőkartrák, EPROM égető, különleges tápegység, lemezegységek eladók! Válaszborítékért részletes ismertetőt küldök! Mészáros Gyula 1029. Bp. Zsírshegy út 110.

A géppel kapcsolatos témákban leveleznének: Bognár Balázs, 9443 Petőháza, Bartók B. u. 11 Budai Attila, 2119 Pécel, Várhegy út 12.

ENTERPRISE klubok,
ENTERPRISE köré
szerveződött baráti társaságok,
ENTERPRISE témákban
levelezni kívánók címeit várjuk!
A címeiket lapunkban rendszeresen közöljük.

hirdetések, felhívások

HIRDETÉSFELVÉTEL

Az apróhirdetések ára: 1 Ft karakterenként. A szöveget és a befizetést igazoló nyugtát (rözsaszín postautalványon) az alábbi címre kérjük elküldeni:

MÁTRIX Kft.

ENTERPRESS
8000 Székesfehérvár
Dózsa György tér 10.

Bankszámlaszám: OTP 679-022096-9
Közületi hirdetőknak kívánságra hirdetési
árajánlatot küldünk.

Megjegyzés: a nem saját fejlesztésű szoftverek másolásával foglalkozó üzletelők hirdeteit nem áll módunkban elfogadni.

TISZTELT LEENDÓ SZERZŐTÁRSOK!

Kérjük Önöket, hogy anyagaik elkészítése, beküldése előtt feltétlenül kérjenek ingyenes tájékoztatót a feltételekről, a szerkesztőség által felállított tartalmi és formai elvárásokról.

Az érdeklődők leveleit levélcímünkre várjuk:
ENTERPRESS
1399 Budapest Pf. 701/334