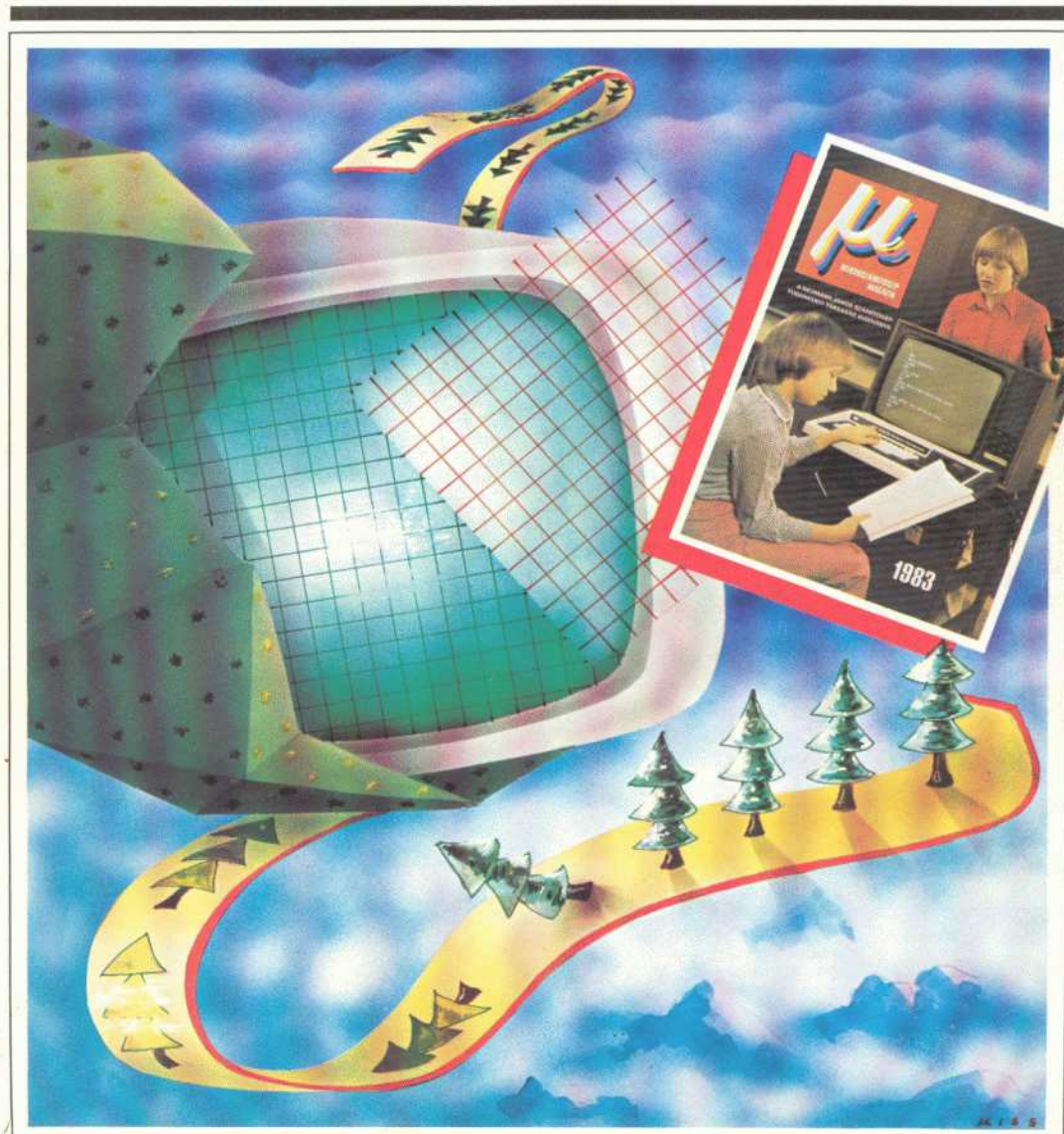


mikro

számítógép

magazin

Ára: 30 Ft



LETTÜNK, ÍRTUNK! GYŐZTÜNK?

1988/12

NINCSEN BASIC



TÖVIS NÉLKÜL . . .



mikro számítógép magazin

6. ÉVFOLYAM
1988/12. SZÁM

A NEUMANN JÁNOS SZÁMÍTÓGÉPTUDOMÁNYI TÁRSASÁG LAPJA

A szerkesztőbizottság vezetője:
Kovács Győző

A szerkesztőség munkatársai:
Bakos Tamás
(programozástechnika)

Broczkó Péter
(hírek)

Ferencz Mária
(tervezőszerkesztő)

Kovács Győző
(levelezés)

Petróczy Judit
(könyvek)

Pinke György
(NJSZT, alkalmazások)

Simonyi Endre

Szebenszki Sándor

Szulyovszky Csaba

Tamásné Lakó Erika

Terebessy Ákosné

Vizesi Mária

Címkepünk:
Kiss Iлона munkája

Felelős szerkesztő:
Könyves Tóth Pál

Szerkesztőség:
1027 Budapest, Fő u. 68.
Telefon: 154-250

Levélcím:
1371 Budapest
Pf. 433.

Kiadja az Ifjúsági Lap-
és Könyvkiadó Vállalat

Felelős kiadó:
dr. Király G. István
igazgató

Kiadóhivatal:
1065 Budapest, Révay u. 16.
Telefon: 116-660

Terjeszti a Magyar Posta
Előfizethető a hírlapkezelés
hivataloknál
és a Posta Hírlap-előfizetési
és Lapellátási Irodáján
(1900 Budapest XIII.,
Lehel u. 10/A)
vagy átutalással a 215-96 162
pénzforgalmi jelzőszámra.

Megjelenik havonta.
Egy szám ára 30,— Ft
Előfizetési díj:
egy évre 360,— Ft
fél évre 180,— Ft
Külföldön terjeszti
a Kultúra.

1389 Budapest, Pf. 149.
és a Magyar Média
1932 Budapest, Pf. 279.
88—1552



Szikra Lapnyomda
Budapest (88—1669)
Felelős vezető:
Csöndes Zoltán vezérigazgató

INDEX: 25 629
ISSN 0236-6088

TARTALOM

- 2 Ót éves a Mikroszámítógép Magazin
- 6 Feladatok — megoldások
- 11 Egy sarokkal olcsóbb!
- 18 Monitorológia
- 20 Mókuserék
- 21 SEIKOSHA SP—1200 VC
- 28 Rendszerfejlesztési eszközök
- 31 Transzisztorországban
- 34 Merre tart a világ?
- 39 μ INFORM
- 40 Olvastunk
- 45 Programtermék
- 46 Adok—veszek—cserélek

ISKOLA-SZÁMÍTÓGÉP

- 3 Statisztikai próbák programozása

CSÍPEGETŐ

- 7 Digitális montázkészítő
- 7 RESET sajátkezüleg
- 8 Korszakváltás, de hogyan...?!
- 8 Magyar nyelvű hibaüzenetek
- 9 Mi miből származik, avagy a nyelvcsaládfáról...
- 10 Mandelbrot

PROGRAMOZÁSTECHNIKA

- 12 BASIC és gépi kód
- 13 Hibajelzések és üzenetek
- 15 A nagyfelbontású képernyő forgatása bajtonként
- 17 Környezetvédő „ZÖLD” rutinok...

ENTERPRISE

- 24 POKE-kal vagy POKE nélkül?
- 25 Megkérdeztük az ENTERPRISE-ről
- 26 Az ellipszis és a többiek
- 27 Mi a manó?

PROGRAMOK

- 36 Segíts magadon!

KLUB

- 38 Klubszervezés és oktatás
- 38 Adom a magyarázatot!

SAKK

- 42 Anyag és mozgékonyság

AZ OLVASÓ ÍRJA

- 43 HÍREK—ÉRDEKESSEGEK, KÖNYVEK

PONTVADÁSZAT

- 46
- 48



Az öregedésnek is — mint minden másnak — megvan a maga kellemes és kellemetlen oldala. Az utóbbit az ember igyekszik elfelejteni, míg az előbbit élvezni. Az előbbihez tartozik például, hogy mostanában meglehetősen sok helyre hívnak meg, csak előadást tartani, de évfordulókat ünnepegni is. Megvallom, minden ilyen alkalomnak nagyon örülök. Így nagy örömet szerzett a CHIP magazin levele is, amelyben meghívtak, hogy vegyek részt a lap alapításának tízedik évfordulója alkalmából rendezett ünnepi összejövetelen.

A CHIP a legnagyobb példányszámú (vagy az egyik legnagyobb példányszámú) európai számítógép-magazin, az első — egy öszeptven kettős — száma 1978 szeptemberében jelent meg, éppen öt évvel előzve meg a Mikroszámítógép Magazin kiadását.

MIKROSZÁMÍTÓGÉP MAGAZIN

Az ünnepség méltó volt egy százhuszszeszre sikerlaphoz. A vendégek lehettek vagy ezren. A hangulat kitünő volt, talán csak az ünnepi szónoklatokkal bántak csínjain a rendezők.

Dr. Kurt Eckenkamp úr, a Vogel Verlag egyik tulajdonosa bensőséges szavakkal méltatta a lap jelentőségét, a német számítástechnikában vállalt szerepét. Elmondta, hogy mit is jelentett és jelent még ma is egy ilyen példányszámú, elsősorban az „amatőröknek” és kevésbé a szakmának szóló újság. Persze annak ellenére, hogy a lap a nagyközönségnek készült — saját tapasztalatom szerint is — a szakma nagyon is szereti és elfogadja a lapot, ami valószínűleg a lap tárgyalagos írásainak, az elfogulatlan hardver-softver teszteknek, az újdonságokról szóló írásoknak és nem utolsósorban annak a ténynek köszönhető, hogy a CHIP nem a cégeknek, hanem a szakmának az elkötelezettje.

Az évforduló ünneplésére Münchenben megjelentek — nyugodtan mondhatom — a szakma legjelesebb európai képviselői — magamat persze nem számítom ide, hiszen minden rendszerben vannak kivételek is —, a legfontosabb számítástechnikai szervezetek, gyártók, szoftverházak, nagyon sok kereskedőház és hirdetésekkel foglalkozó intézmény, valamint más számítástechnikai lapok vezetői és munkatársai is.

Csak rövid tájékoztatóként elmondom, hogy a szerkesztőség a lapon kívül számtalan más — a számítástechnikához kapcsolódó — kiadványt is megjelentet: könyvetek, speciális szakmai tájékoztató anyagokat, szoftvert áru, és még sok minden mással is foglalkozik. — Egyszer például az egyik korábbi látogatásom alkalmával majdnem lehetetlen volt a lap igazgatójának, Richard Kerler úrnak a szobájába bejutnom, mert minden talpalatnyi helyet IBM-klónokkal teli dobozok foglaltak el.

Az történt ugyanis, hogy a gyártóknak és a forgalmazóknak a lap bejelentette, hogy az NSZK-ban árusított PC-k fontosabb jellemzőit egy új vizsgálati módszerrel fogja megvizsgálni és össze fogja hasonlítani a teljesítményüket. Egy ilyen bejelentésre mindenki, aki a piacon gepgyűben érdekelt volt, „összteret magát”, hogy a gépe a tesztelésben részt vegyen. Ha a mi évekkel ez előtti hasonló akciónkra gondolok, az jut az eszembe, hogy akkor a hazai gyártók és forgalmazók vagy elzárkóztak egy objektív tesztben való részvételtől, vagy befolyásolni igyekeztek a lap munkatársait, hogy az eredmények egy kicsit kozmetizálva kerüljenek a lapba. Nem is volt azóta hasonló vállalkozásunk, egy jó kezdeményezés igazságtalan bukása egy időre elengedő tanulságot szolgáltatott számunkra.

Talán ez a példa is érzékelteti, hogy magazinunk más körülmények között dolgozik, mint a CHIP, így feltehetően a Mikroszámítógép Magazin megszületésének ötvenes évfordulója is mel-

lőzzük majd a zajos ünnepségeket, és ha jól sejttem, akkor csak szűkebb — szerkesztőségi körben — fogunk emlékezni az elmúlt évekre. Olvasóink jól tudják, hogy a lap mintegy három hónapos előretartással készül. Most tehát, szeptember elején, amikor ezt a kis eszmefuttatást írom, még nem tudom, valakinek, beleértve volt és jelenlegi gazdáinkat is, eszébe jut-e majd, hogy van és most éppen ötvenes a Mikroszámítógép Magazin. A lapot — valamikor 1983 közepén — meglehetősen szokatlan céllal alapították. Egy olyan fórumot akartunk teremteni, amelynek célja a számítástechnikai kultúra tömeges terjeszté-

ÖTÉVES A MIKROSZÁMÍTÓGÉP MAGAZIN

se, tehát szakmai igényrel elsősorban a nem számítástechnikai szakemberekhez, az amatőrökhöz és a diákokhoz kívánt szólni. Az elmúlt öt év alatt megtanultuk, hogy az olvasók igénye és érdeklődése igen széles körű, amelyet kiszolgálni egyáltalán nem könnyű, és nem vagyok biztos abban, hogy minden igyekezetünk ellenére sikerült elegendőten valamennyi olvasónk várakozásának. Az olvasói levelekből kiderült, hogy nemcsak az amatőrök, de a szakma is olvassa és figyelemmel kíséri lapunkat, és nem csupán a hibáinkra reagálnak gyorsan, de nagyon sok tanácsot is kapunk, hogy hogyan tehetjük színvonalasabbá a magazint.

Annak idején Könyves Tóth Pállal órákat vitatkoztunk kettésben, amíg kidolgoztuk a lap tervét és amíg az első szerkesztőségi ülésen megalkult az első szerkesztőbizottság: Broczkó Péter, Budai György, Garádi János, Jakab Ágnes, Nacsa Sándor, Patáki Ernő, Petróczy Judit, Pogány Csaba, Simonyi Endre, Szabó János, Varga András, Vass Nándor és Votisky Zsuzsa.

Az eredmény: valamikor 1983 karácsonyára utcára került az első szám, amelyet — nem lévén lapengedélyünk — az NJSZT kiadványaként publikáltunk.

Az első, a „történelmi” szám hozzávetőlegesen tízezer példányban jelent meg. Ma az olvasói kör már stabilizálódott, és a lap általában húszezer körüli példányban kerül az olvasókhöz. Nem titok, hogy amióta megkezdődött a gazdasági „szabályozók” drasztikus változása, valamint az áremelések sorozata a nyomdaiparban, azóta állandó gondunk a lap gazdasági stabilitásának a fenntartása.

Visszatérve a lap történetére, az első két évben jelentős támogatást kaptunk a Tudományos-érzési és Informatikai Intézetől — gyakorlatilag az oktatási kormányzattól, hiszen vállaltuk és támogatjuk az iskolai számítástechnika és az iskolán kívüli diákképzés ügyét. Ezzel a témával még ma is jelentős terjedelemben foglalkozunk, annak ellenére, hogy évek óta az oktatásügy — többszöri ajánlkozásunk ellenére — nem érezte magának a Mikroszámítógép Magazin, és valószínűleg ezért, a támogatását is megszüntette. Szerencsére más a helyzet a tanárokkal és a diákokkal, akikről folyamatosan nagyon sok levelet, hozzászólást, cikket és programot, no és ötleteket, javaslatokat is kaptunk, nem beszélve az anyagi támogatásokról, hiszen „veszik a lapot”.

Volt egy mintegy másfél éves „flörtünk” a KISZ KB-val is, amiért nagyon hálaak vagyunk, hiszen ez alatt az idő alatt jelentős támogatást kaptunk az ifjúsági szervezettől. Sajnos új „felgazdánk”-kal 1988 közepén az együttműködés megszűnt, a szerződést felmondták, és ma is-

mét csak egyetlen tulajdonosa van a lapnak, a Neumann János Számítógéptudományi Társaság, támogatásuk nélkül a lapot nem lehetne megjelentetni.

Nagyon sok fórumon javasolták az elmúlt években, hogy menjük a lap árát, legalábbis „bruttósítasuk” a hosszú ideje 30 Ft-os árat. Mindeddig ellenálltunk a nyomásnak, aminek persze a jelenlegi érvényes és sokak által kritizált postai terjesztési szabályozás is az oka. A ma rendelkezéskor szerint ugyanis a posta a lap árának az egyharmadát elviszi terjesztési költségek címén. A saját terjesztés összehasonlíthatatlanul kevesebbe kerülne, mint a jelenlegi 10 Ft-os példányonkénti költség. Ha például a lap árát a háromszorosára emelnék, akkor ugyanezért a mai „szolgáltatásért”, tehát ugyanannyi postai munkáért háromszor annyit kellene fizetnünk, ami jó a postának, és rossz mindenki másnak. Aztán azért sem emeltünk árat, mert akkor — véleményünk szerint — jelentősen csökkent volna a lap iránt az érdeklődés. A mai kereseti és adóviszonyok mellett a családi háztartások minden fillér többletkiadására nagyon érzékenyek, a költségeket még egyáltalán nem fedező, például 60 Ft-os példányonkénti ár negatív következményei — véleményem szerint — beláthatatlanok lennének. Sajnos a jelenlegi ár már sokáig nem tartható, nagyon valószínű, hogy a jövő évben a lap árát — követe a pénz romlását — mérsékelten emelnünk kell.

Tudvalegő, hogy az anyagilag sikeres lapokat a hirdetések tartják el. Sajnos egyéve kevesebb a hal, ugyanakkor a fókák száma növekedik, így sokan tartanak igényt a hirdetői támogatásra. Mi ezen a téren csak néha voltunk ügyesek, az olvasók akkor is elverték rajtuk a port, mondván, hogy „hülye hirdetésekre” pazaroljuk az a dedes írásoknak fenntartott oldalakat. (Idézet a diákok leveleiből.)

Most az írás végén — érzem — egy nagyon optimista, lendületes befejezés kellene, nagyjából az ötvenes évek szocreál újságról-ábrázolásának megfelelően: egyéve kezdem a lap, a másik egy írásra kész golyóstoll, és egy kissé fel-emelt fejfel firkászom a végetlen. Ami a lap tartalmát illeti, az hiszem valóban nagyon optimisták lehetünk. Nem hiszem, hogy csak a barátaink ragadtak volna tollat és írták meg a véleményüket a lapunknak. S talán az olvasó azt sem feltételezi rólam, hogy kizárólag a jó véleményeket szoktam az „Olvasó írja” rovatban közölni, ám olvasóink döntő többsége valóban elégedett a cikkeik színvonalával és a lap szerkesztésével is — az irják, hogy nem vagyok unalmasak. És reventek, hogy fejlődünk, szívesen fogadják, hogy igyekszünk sok minden újat bemutatni. Nagyon jó munkatársaink vannak, dolgozik és igen aktív a diákszerkesztőség (a Csepelrovi rovat), az hiszem, nem élünk távol a való világtól, és — hogy egy közhelyet is mondjak — rajta a kezünk a magyar számítástechnika útjében. Megpróbálunk gazdaságpolitizálni is — egyelőre kevés sikerrel. Szóval, ha rajtuk múlik, akkor a fejlődésünk töretlen lesz, persze csak azt ígérhetjük, hogy továbbra is frissek és érdekesek igyekszünk maradni.

Ami pedig az anyagi háttérrel illeti, arra vonatkozóan is megvannak az elképzeléseink. Az írást a CHIP magazin tízeves évfordulójával kezdtem, nem véletlenül. Most éppen a közepén vagyunk egy tárgyalásorozatnak, hogy a CHIP-pel közös lapot és közös kiadvállalatot hozzunk létre. Ma, szeptember elején még nem tudjuk, mit hoz a jövő, nagyon reméljük, hogy kedvezően zárulnak a megbeszélések, és a Vogel hathatós anyagi és erkölcsi támogatásával, s persze hazai gazdálkodó sikerült újsáztületünk.

En ez kívánom mindnyájunknak a Mikroszámítógép Magazin ötödik születésnapján. Olvasóinknak pedig — a szerkesztőség valamennyi munkatársa nevében — kellemes karácsonyi és sikeres, boldog új évet!
KOVÁCS GYÖZŐ

STATISZTIKAI

PRÓBÁK PROGRAMOZÁSA

A matematika alkalmazási területei közül talán a statisztikai feladatok igénylik a legtöbb numerikus számolást. Legyen szó egyszerűbb leíró statisztikai feladatokról, mint például átlag- és szórásszámításról vagy a nagyobb elméleti apparátust igénylő matematikai statisztikáról. Nem véletlen tehát, hogy már a számítógépek megjelenése előtti időben is törekedtek a nagyszámú adattal végzendő számítások gépésítésére, azok legalább részbeni automatizálására. Ezzel magyarázható az is, hogy a legegyszerűbb házi számítógépektől a mamut-rendszerekig minden géptípushoz a szoftverházak kínálatában a statisztikai feladatok felhasználói programjainak gazdag választékát találjuk.

Az viszont felettebb meglepő, hogy ezek között a matematikai statisztika körébe tartozó programok szinte kivétel nélkül csontkák, hiányosan oldják meg a feladatot. Mind a programcsomagok, mind a szubrutinkönyvtárak a szóban forgó feladatokra olyan komplex programokat vagy programok építésére alkalmas modulokat tartalmaznak, amelyek tökéletesen másolják azt a technológiát, melyet a hagyományos számolóeszközökre támaszkodó statisztikai gyakorlat az „őskorban” alkalmazott. Régóta keresem a változást, az előrelépést a programozási felfogásban, de annak még kísérletével sem találkoztam. Lehet, hogy léteznek teljes megoldások, a hiányszakból viszont annyi van, hogy minden bizonyosan nem lesz hiábavaló az alábbi elemzés.

A kifogásaimat, helyesebben aggályaimat egy igen széles körben alkalmazott statisztikai vizsgálat, a khi-négyzet analízis programozásának részletes tervezésével illesztésem, amivel talán azt is sikerül bemutatnom, hogy gyakran egy „kis program” tervezése is milyen gondos és alapos elemzést igényel.

A khi-négyzet próba

A teljességre törekvés nélkül, röviden tekintsük át a statisztikai próbák elvi alapjait és kivitelezésük módját. A vizsgálatot végző kutató gyakran valamilyen tulajdonság meglétére vagy épensége hiányára gyának. Sokszor, például akkor, ha az adatokat ellenőrizhetetlen és kiküszöbölhetetlen mérési hibák terhelik, a kérdéses nem lehet határozott igen/nem válasz adni. Ilyenkor konstruálni kell egy olyan, az adatokból — mintából — kiszámítható mennyiséget, az ún. próbafüggvényt vagy statisztikát, melyről tudjuk, hogy a mintaelemek meghatározott kezdeti feltételeinél hogyan viselkedik.

Úgy gondolom, nem lesz haszontalan a gondolatmenet követésére egy konkrét példát elemezni: tegyük fel, hogy ellenőrizni akarjuk számítógépünk véletlenszám-generátorát. A kérdés az, hogy a generátor egyenletes eloszlású számsorozatot állít-e

elő. Egyszerű programmal meggyőződhetünk arról, hogy egy kísérleti RND sorozat (minta) elemeinek mennyi az átlaga és a szórása. Ha a generátor jól működik, akkor a (0,1) intervallumból vett minta átlaga 0,5, szórása $1/\sqrt{12} = 0,288675 \dots$ körül lesz. Ha ettől eltér a kísérleti eredmény, akkor a generátor BIZTOSAN rossz. Ha viszont a két érték az elméletileg várható értékek közelében van, akkor csak annyit mondhatunk, hogy LEHET jó a generátor, a kísérlet nem zárja ki a helyes működést. A megbízható diagnózishoz azonban komolyabb próbára van szükség.

Mikor mondhatjuk azt, hogy a véletlen számsorozat egyenletes eloszlású? Akkor, ha a teljes (0,1) intervallum minden részintervallumába eső elemek gyakorisága az intervallumok hosszával arányos. A kísérletet tehát úgy kell megszervezni, hogy a (0,1) intervallumot tetszőlegesen felosztjuk és megszámláljuk, hogy az előállított RND sorozatnak egy-egy részbe, cellába hány eleme kerül.

Mit várhatunk ettől a kísérletől? Egyenletes felosztásnál minden cellába ugyanannyi elem fog esni? Majdnem biztos, hogy nem. Azt viszont elvárhatjuk egy „jó” generátortól, hogy KÖZEL egyenlő gyakoriságokat produkáljon.

A következő kérdés az, hogy az elméletileg várt egyenletes gyakoriságtól való eltérést hogyan mérjük. Jelöljük $f(i)$ -vel az i -edik cellában mért gyakoriságot és legyen $p(i)$ annak a valószínűsége, hogy egy RND-vel generált véletlenszám az i -edik cellába esik. Ha r elemű sorozatot állítunk elő, akkor az i -edik cellában elméletileg várt gyakoriság $r \cdot p(i)$ lesz. A tapasztalati és az elméleti eloszlásnak az egész (0,1) intervallumra vonatkoztatott eltérését jellemezhetjük a következő próbafüggvénnyel:

$$(1) \dots \chi^2 = \sum_{i=1}^m \frac{(f(i) - r \cdot p(i))^2}{r \cdot p(i)}$$

ahol m a cellák száma.

Ha egyenletes felosztást alkalmazunk és a példánkban egyenletes eloszlású generátort vizsgálunk, a khi-négyzet statisztika egyszerűbb alakot vesz fel:

$$(2) \dots \chi^2 = \frac{r}{12} \sum_{i=1}^m (f(i) - r \cdot p(i))^2$$

Felmerülhet az olvasóban az, hogy miért pont ezt a kifejezést használjuk az eltérések mérésére. Látható, hogy az egyes cellákon belüli eltérések négyzetével számolunk, s ennek az a jelentősége, hogy ily módon az ellentétes előjelű eltérések nem közömbösítik egymást. Az igazi ok persze nem ez, hanem az, hogy az így számított próbafüggvénynek ismerjük a viselkedését. Bizonyították, hogy ez a statisztika $n = m - 1$

szabadságfokú khi-négyzet eloszlást követ. Ezeknek a valószínűségi változóknak a súrúség- és eloszlásfüggvényei a normális eloszlásból levezethetően:

$$(3) \dots f_{n, \chi^2}(x) = \frac{1}{2^{n/2} \Gamma(n/2)} x^{n/2-1} e^{-x/2} \quad (x > 0)$$

$$(4) \dots P(n, x) = P(\chi^2_{n, \chi^2} \leq x) = \int_0^x f_{n, \chi^2}(t) dt$$

Térjünk vissza a próba menetére. Mint megállapítottuk, a kísérlet várható eredménye az, hogy a mintából számított (1), illetve (2) próba 0-tól különbözni fog ugyan, de ha igaz, hogy a gép RND függvénye egyenletes eloszlást generál, akkor az eltérés nem lesz túl nagy. De vajon mi az a kritikus érték, amely felett el kell vetnünk az egyenletes eloszlás hipotézisét?

Nyilván akkor kellett így itélnünk, ha a mintából számított statisztika „valószínűtlenül” nagy. Olyan nagy, amit véletlen mintából soha, de legalábbis nagyon kis valószínűséggel kaphatnánk. Ez utóbbi valószínűség a (4)-ben szereplő esemény komplementének valószínűsége: $1 - P(n, x)$.

A statisztikusok $1 - P(n, x)$ -re a 0,001, 0,01, illetve a 0,05 szinteket használják a leggyakrabban.

A próba algoritmus

Az elmondottakból következik a próba végrehajtásának menete:

1. Adatgyűjtés (példánkban az RND sorozat képzése).
2. Az $f(i)$ gyakorisági sor előállítás.
3. A (2) próbafüggvény kiszámítása.
4. A (4) valószínűség kiszámítása.
5. Döntés a kiszámított $1 - P$ alapján.

Ezzel szemben a bírált programok beírják azzal, hogy kiszámítják a próbafüggvényt. (Nemcsak a khi-négyzet próbánál, hanem minden másinnál is.) Az algoritmus 4. lépését egyszerűen elhagyják és a felhasználóra bízzák. Ugyanúgy, mint a hagyományos technológiánál, a felhasználó előveszi a könyvespolcáról a kézikönyvet, függvénytáblát, majd a megfelelő táblázatból kikeresi a döntéshoz szükséges kritikus értéket. (Közben lelkiismeretesen interpolál.)

Talán a statisztikai programok készítői úgy gondolták (gondolják), hogy örüljön a t. felhasználó, ha az adatrendezés és a számolási munka egy részétől megkímélik. Arra nem is merek gyanakodni (?), hogy nem tudják megírni a 4. lépést megvalósító programot minden próbához.

Vizsgáljuk meg a khi-négyzet próbánál ezt a kritikus 4. lépést, melyről látni fogjuk, hogy valóban nem egyszerű, de nem is kivihetetlen.

STATISZTIKAI PRÓBÁK PROGRAMOZÁSA

A sűrűségfüggvény vizsgálata

Először foglalkozunk a (3)-ban szereplő függvények nevezőjében álló

$$(5) \dots c_n = 2^{n/2} \cdot \Gamma(n/2)$$

együtthatóval. Az első tényező, a 2-hatvány kiszámítására minden magas szintű programozási nyelvben találnuk direkt vagy indirekt műveletet. A második tényezőben álló gamma függvénnyel más a helyzet. Ennek helyettesítési értékét a

$$(6) \dots \Gamma(x+1) = \int_0^{\infty} t^x \cdot \exp(-t) dt$$

improprius integrál definiálja, amit közeli formulákkal kívánt pontossággal ki lehetne számítani. Erre azonban nincs szükség, mert az (5) együtthatóban csak speciális argumentumok szerepelnek. Nevezetesen az n szabadsági fok párosságától függetlenül a kiszámítandó helyettesítési érték

$$(7) \dots \Gamma(n/2) = \begin{cases} \Gamma(k) & ; n=2k \\ \Gamma(k+1/2) & ; n=2k+1 \end{cases}$$

Ezeket a kifejezéseket a gamma függvény következő tulajdonságainak a felhasználásával meghatározhatjuk:

$$(8) \dots \begin{cases} \Gamma(x+1) = x \cdot \Gamma(x) & ; x \text{ egész} \\ \Gamma(x+1) = x & ; \theta \in x \text{ egész} \\ \Gamma(1/2) = \sqrt{\pi} \end{cases}$$

Ezzel az (5) együttható egyszerűbb alakban írható fel páros és páratlan n-re egyaránt:

$$(9) \dots \begin{cases} c_{2k} = 2^k \cdot (k-1)! = 2 \cdot (2k-2)!! \\ c_{2k+1} = \sqrt{2} \cdot (2k-1)!! \end{cases}$$

ahol $n!! = n \cdot (n-2) \cdot (n-4) \dots (n-2j)$, minden $j: (n-2j) > 0$.

E két esetet egyetlen formulával írhatjuk fel minden n-re:

$$(10) \dots c_n = c_p \cdot (n-2)!!$$

ahol páros n-re $c_p = 2$, páratlanra pedig $\sqrt{2}$.

A számításához felhasználható a következő rekurzio is:

$$(11) \dots \begin{cases} c_1 = \sqrt{2\pi} & ; c_2^2 = 2 \\ c_{n+2} = n \cdot c_n = c_p \cdot n! \end{cases}$$

Az együttható tehát kiszámítható, csupán az a kérdés, hogy a lebegőpontos aritmetikában nem okoz-e túlsordulást ez a faktoriális típusú kifejezés. Sajnos 8 bites

karakterisztika mellett már $n=58$ -ra sem végezhető el a számolás (hozzávetőlegesen $1.7E+38$), ezért tovább kell folytatnunk a probléma elemzését.

Vizsgáljuk meg előbb magát az integrandust, de figyelmünket fordítsuk az együtthatót nem tartalmazó

$$(12) \dots x_n^2 \Rightarrow c_n \cdot h_n(x) = x^{(n/2)-1} \cdot \exp(-x/2)$$

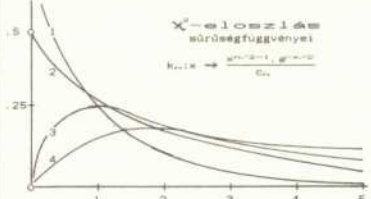
függvényekre. Ezek menetét az ismert analitikus módszerekkel megállapíthatjuk. Mint az a grafikonokon (1. ábra) is látható, a sűrűségfüggvények három típus képviselnek:

— Az $n=1$ esetben a 0 jobb oldali környezetében a függvény nem korlátos, tehát a (4) integrál improprius.

— Az $n=2$ esetben 0-nál a jobb oldali határérték 0,5, a függvény monoton csökken.

— Minden $n > 2$ -re a sűrűségfüggvényeknek $x=n-2$ helyen maximumuk van. Az $x=0$ -nál a határérték 0.

A három típusban közös, hogy $+\infty$ -ben 0 a határérték.



1. ábra

Az eloszlásfüggvények

A szignifikancia szintjének meghatározásához, azaz a (4) határozott integrál kiszámításához ismernünk kell a sűrűségfüggvény primitív függvényeit. Célszerűbb az együtthatót nem tartalmazó (12) függvényekkel foglalkozni. Az $u = \exp(-x/2)$ és $v^2 = x \uparrow (n/2 - 1)$ választással, parciális integrálással azt kapjuk, hogy

$$(13) \dots \int_0^x t^{n/2-1} \cdot \exp(-t/2) dt = \frac{1}{n} \int_0^x t^{n/2} \cdot \exp(-t/2) dt + \frac{1}{n} \int_0^x t^{n/2-1} \cdot \exp(-t/2) dt$$

Ebből rekurzív helyettesítéssel és azonos átalakítással, majd az eredeti sűrűségfüggvényekre visszatérve, azok primitív függvényeit a

$$(14) \dots \int_0^x t^{n/2-1} \cdot \exp(-t/2) dt = \frac{2^{n/2}}{n} \cdot \exp(-x/2) \cdot \left(\frac{x}{n-2} + \frac{x^2}{(n-2)(n-4)} + \dots \right)$$

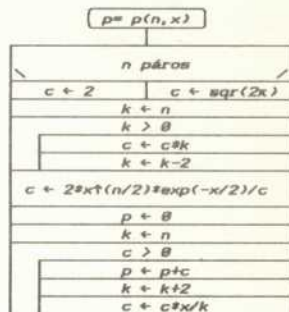
függvénnyel állíthatjuk elő. Mivel ezek helyettesítési értéke az $x=0$ helyen zérus, a keresett (4) valószínűségeket e függvények $x = kh \uparrow 2$ helyen vett helyettesítési érté-

ke adja. Ezt természetesen nem lehet „pontosan” kiszámítani, de látható, hogy a zárójelben álló sor tagjaiban a nevező igen gyorsan növekszik, s így a konvergencia is gyors lesz.

A számítás algoritmus

A $p(n,x) = P(\xi < x)$ valószínűség kiszámítására a (11) rekurzio kezdőértékéről és a (14) függvény sor kiszámításából álló egyszerű programot vezethetünk le.

A program elején az n párosságától függetlenül állítjuk be a (11) rekurzio kezdőértékét ($c := \dots$). A rekurzio végén kiszámítjuk a (14) zárójelben kívül álló tényezőt, melyet a c változóban viszünk tovább. A függvény kiszámítható egyszerű összeházasítással, az összeget a p változóban képezzük. Az összegző ciklust az aktuális tag alácsordulásával állítjuk le. A programban szereplő p változó a ciklusok futó paramétere.



A túlsordulás kivédése

A kiszámítandó $p(n,x)$ valószínűség a definícióból következően mindig a (0,1) intervallumba esik, ha $x > 0$. Arról kell tehát gondoskodnunk, hogy a számítási algoritmus közbenso eredményei ne lépjenek ki a programot futtató számítógép lebegőpontos formátumának korlátai közül. Sok esetben megoldást nyújt a sorozattal számított mennyiségek logaritmus kiszámítása, természetesen csak akkor, ha a logaritmus számításból való visszatérést az algoritmusnak azon a pontján végezzük el, ahol a számított mennyiség már az ábrázolható tartományba esik. A $p(n,x)$ -re írt algoritmus a példánkhoz készített BASIC programban ennek megfelelően úgy módosítottuk, hogy a c iterációjában használtunk logaritmus transzformációt, s a visszatérést akkor végeztük, amikor a (14) zárójelben kívüli helyettesítési értéket kiszámítottuk. Belátható, hogy ennek a számításnak az eredménye mindig kisebb 1-nél, hiszen a másik tényező pozitív x-re mindig 1-nél nagyobb.

Nem állítjuk, hogy a túlsordulás lehetetlen, de csak akkor következik be, ha a közbenso értékek logaritmus ábrázolhatatlan. Ez azonban csak irreálisan nagy, a gyakorlatban elő nem forduló szabadsági foknál következne be.

RND-TEST: COMMODORE- PLUS/4

Megtörténhet azonban, hogy nagy x mellett a (14) elején álló kifejezés alácsondult, és $p=0$ eredményt ad. Ezt az extra esetet van hivatva kivédeni a szubrutin 1180-as sorában szereplő vizsgálat.

Látható, hogy a program kódolásakor a függvény-sor iterációjának ciklusát is $c > 0,0000001$ feltétellel vezéreljük. Ezzel is kielégítő pontosságot érhetünk el.

A szubrutin alkalmazása

A működés bemutatására a szubrutint a példaként szereplő RND-teszthez készített keretprogrammal futtattuk. Ennek listájából a kiírást végző modul elhagytuk. Érdekes megfigyelni, hogy bár mindkét kísérletben (2. ábra) a próba az elfogadási tartományban van ($1-p > 5\%$), a jobb illeszkedést mutató 2000-es mintában az átlag és a szórás jobban eltér az elméleti értéktől.

| i | f(i) | Hisztogram |
|----|------|------------|
| 1 | 194 | ██████████ |
| 2 | 201 | ██████████ |
| 3 | 187 | ██████████ |
| 4 | 190 | ██████████ |
| 5 | 189 | ██████████ |
| 6 | 218 | ██████████ |
| 7 | 212 | ██████████ |
| 8 | 208 | ██████████ |
| 9 | 205 | ██████████ |
| 10 | 196 | ██████████ |

| i | f(i) | Hisztogram |
|----|------|------------|
| 1 | 4945 | ██████████ |
| 2 | 4891 | ██████████ |
| 3 | 5048 | ██████████ |
| 4 | 4985 | ██████████ |
| 5 | 5071 | ██████████ |
| 6 | 4963 | ██████████ |
| 7 | 5059 | ██████████ |
| 8 | 5113 | ██████████ |
| 9 | 5000 | ██████████ |
| 10 | 4925 | ██████████ |

| | |
|---------------------------|----------|
| Kísérletek sz. = | 2000 |
| Szabadsági fok = | 9 |
| Khi ² = | 5.0000 |
| p(n, khi ²) = | 0.1657 |
| 1-p = | 0.8343 |
| Átlag = | 0.5062 |
| Várt átlag = | 0.5000 |
| Szórás = | 0.2073 |
| Várt szórás = | 0.2007 |
| Futási idő = | 00:01:04 |

| | |
|---------------------------|----------|
| Kísérletek sz. = | 50000 |
| Szabadsági fok = | 9 |
| Khi ² = | 9.1448 |
| p(n, khi ²) = | 0.5759 |
| 1-p = | 0.4241 |
| Átlag = | 0.5010 |
| Várt átlag = | 0.5000 |
| Szórás = | 0.2078 |
| Várt szórás = | 0.2007 |
| Futási idő = | 00:26:30 |

2. ábra

```

100 REM *****
110 REM * RND-TEST *
120 REM * BASIC : CBM V3.5 *
130 REM *****
140 PRINT "RND-GENERATOR TESZT"
150 INPUT "KISERLETEK SZAMA=" ; R
160 INPUT "CELLAK SZAMA=" ; M
170 IF R<10*M THEN 150
180 :
190 TIS="000000"
200 DIM F(M)
210 REM --- GYAKORISAGI TABLA ---
220 A=0 : S=0
230 FOR I=1 TO R
240 : F=RND(0)
250 : A=A+F
260 : S=S+F*I
270 : J=INT(M*F)+1
280 : F(J)=F(J)+1
290 NEXT I
300 A=A/R : REM ATLAG
310 S=SQR(S/R-A*A) : REM SZORAS
320 :
330 REM --- KHI2 STATISZTIKA ---
340 X=0
350 FOR J=1 TO M
360 : X=X+(F(J)-R/M)*T2/R*M
370 NEXT J
380 N=M-1
390 IF X=0 THEN P=0 : GOTO 430
400 :
410 : GOSUB 1000 : REM P = N, X :
420 :
430 TMS=TIS : REM TISZTA FUTASIDO
440 :
450 REM --- OUTPUT ---
:
:
999 END
1000 REM ***** P = N, X *****
1010 REM X := KHI2 PROBAFUIGVYENY
1020 REM N := SZABADSAGFOK
1030 IF X=0 THEN P=0 : GOTO 1190
1040 IF M/2=INT(N/2) THEN C=2 : ELSE
:
: C=SQR(2*X)
1050 K=N : C=LOG(C)
1060 DO WHILE K>0
1070 : C=C+LOG(K)
1080 : K=K-2
1090 LOOP
1100 C=2*EXP((N+LOG(K)-X)/2-C)
1110 P=0
1120 K=N
1130 DO WHILE C>0.0000001
1140 : P=P+C
1150 : C=C+K
1160 : C=C*K/K
1170 LOOP
1180 IF P=0 AND X=N THEN P=1 : REM EXTRA
1190 REM P := PIN(X) SZIGNIFIKANCIA
1200 RETURN
    
```

LOTTÓ-TEST: 3. ábra

| | | | | | | | | | |
|----|----|-----|----|-----|-----|----|----|-----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 79 | 68 | 114 | 84 | 66 | 86 | 89 | 85 | 79 | 99 |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| 67 | 98 | 99 | 88 | 88 | 74 | 80 | 99 | 91 | 84 |
| 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 |
| 79 | 85 | 96 | 88 | 89 | 69 | 66 | 75 | 100 | 62 |
| 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 |
| 71 | 80 | 78 | 94 | 87 | 83 | 77 | 82 | 70 | 74 |
| 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 |
| 89 | 94 | 79 | 78 | 85 | 89 | 96 | 78 | 95 | 80 |
| 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 |
| 95 | 76 | 83 | 81 | 77 | 105 | 73 | 69 | 77 | 91 |
| 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 |
| 74 | 75 | 67 | 93 | 84 | 85 | 93 | 72 | 93 | 73 |
| 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 |
| 86 | 83 | 83 | 78 | 100 | 79 | 96 | 82 | 77 | 71 |
| 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 |
| 85 | 75 | 82 | 96 | 78 | 101 | 85 | 67 | 82 | 85 |

Érdeklődésre tarthat számot egy másik próba, mellyel a lottó számsorsolásának egyenletes eloszlását vizsgáltuk (3. ábra) az első 1500 játékévtől (1985. II. 29.) után. Az egyenletes eloszlástól való eltérés gyengén szignifikáns: $1-p = 3\%$. Meg kell jegyezni, hogy ugyanez a próba korábban még jó illeszkedést mutatott.

Dr. Hack Frigyes

| | |
|---------------------------|----------|
| Húzások száma = | 1500 |
| Szabadsági fok = | 89 |
| Khi ² = | 115.6088 |
| p(n, khi ²) = | 0.9695 |
| 1-p = | 0.0305 |
| Elméleti gyak. = | 83.3333 |

Sorozatainkat először minden középiskolások számára, de reméljük, hogy minden olvasónknak tanulási lehetőségét és szórakozását nyújt.

A feladatok a Nemes Tihamér országos számítástechnikai verseny színterének felételek meg. Minden esetben olyat választunk, amely röviden, gyorsan megoldható, de megoldásához ötletre van szükség. A megoldást mindig a következő számban közöljük.

Mivel változatosságra törekünk, különböző programozási nyelveket használunk. Az is előfordul majd, hogy egy feladatra több programnyelven is közlünk megoldást, ezzel is elősegítve az ismeret-szerzést.

A szerkesztőség várja az olvasók, a versenyzők leveleit. A legelőkeltebb program beküldítői könyvtárványul jutalmazuk. Ne feledjenek azonban a programhoz leírást is mellékelni!

6. feladat: Osztlás

Írjon programot, amely egy nyolcvanegyűű előjeles egész számot eloszt egy nyolcvanegyűűvel, majd kiírja a hányadost és a maradékot!

A megoldás

A feladat jellegéből kinalózik, hogy a programot assembly nyelven írjuk meg. Ez különösebb nehézség nélkül megoldható, és gyors, hatékony kódot eredményez. Az előző feladathoz, a szorzás, a mostanéhoz hasonló, amiért is a megoldásunk programjével a 8086 assembly.

De mit tegyen az az olvasó, aki az assembly programozásban kevésbé járatos? Megoldható a feladat más nyelven is? Természetesen igen, akár BASIC-ben is. Az alább ismertetett programmal is ezt igazolom. Kétségtelen, hogy ez nem egy jól használható, hatékony program, ám ezzel a feladattal tökéletesen megoldható.

Azaz, hogy a program megírására magas szintű nyelvet választottunk, még nem mentesítünk a 80 jegyű szám ábrázolásának gondjaitól. Semelyik egyszerű változó nem alkalmas ilyen méretű számok ábrázolására, ezért valamilyen összetett adattípust kell létrehozniuk. Ez cselemben valamilyen tömb lesz.

Valós számokból álló tömb használatakor tudnánk egy elemen a legtöbb jegyet tárolni. Ekkor azonban a valós szám ábrázolását az adott gépen meg kellene ismerni, és még így sem biztos, hogy végeredményben a hatékonyabb programot kapunk, a valós számokkal végzett műveletek lassúsága miatt.

Marad tehát az egész és a karakteres ábrázolás mód. Ha a programot BASIC-ben írjuk, akkor a karakteres ábrázolás a cselemben, mivel így a BASIC sztringműveleteket használhatjuk. Más magas szintű nyelv alkalmazásánál valószínűleg az egészekből álló tömb lenne előnyösebb.

El kell azt is döntönnünk, hogy egy karakteren hány jegyet tárolunk. Értelmezhetjük a tömböt egymást követő bajtként és elhelyezhetjük a számot ezeken bináris alakban. Ekkor a beolvasásnál és a kiírásnál szükséges konverzió bizonyultatós lesz, de cserébe a számítás gyorsabbá válik, és a tárolásra felhasználható terület minimális lesz. (Ehhez hasonló ábrázolást használunk az előző feladat megoldásánál.)

Más számbázolási is elképzelhető: tárolhatunk két jegyet egy karakteren BCD formában, vagy akár egy jegyet karakteres formában is. Nyilván ez utóbbi bány legpazarlóbban a memóriával, de a számok beolvasásánál és kiírásánál minimális átalakítást igényel.

Ez utóbbit használja a mellékelt program is. A program ZX-Spectrumra készült, de semmilyen különleges utasítást nem használ, így könnyen átvihető más gépre.

A program elején, a 90–160-as sorokban a változókat definiáljuk. A 100-as sorban látható értékadás dönti el, hogy hány jegyű számokkal végezzük a műveletet. Az osztó és a maradék 41 jegyű, hogy a számításhál fel-lepő átvitelt is tárolni lehessen. Az es az eredmény első-jelét tárolja. Ez már a beolvasáskor elől.

A program következő részében (190–290-es sorok) a számokat olvassuk be. A beolvasott sztringet a 2000-es Alakít szubrutint a számításokhoz megfelelő 80 jegyűvé alakítja. Az eredmény előjelét már itt meg lehet határozni. Bár ez a szubrutin a program legghosszabb rutinja, megértéséhez különösebb magyarázat nem szükséges, hiszen csupán egy sor feltelevizsgálatból áll.

A számítás teljes egészében 3000-es Osztlás szubrutinra hárul. Az algoritmus majdnem teljesen megegyezik az általános iskolában megismert írásbeli osztással.

Először m5-ba rakja az osztandó első jegyét és ebből annyiszor vonja ki az osztót (oS), amíg m5 < o5 nem lesz. A hányados első jegyét a levonások száma adja, m5 pedig a maradékot tartalmazza. Ezután m5-höz hozzáveszi az osztandó következő jegyét és újra elvégzi a kivonásokat. Ezek száma adja a hányados következő jegyét.

Mindzen így ismétlődik az osztandó utolsó jegyéig. Ek-

FELADATOK — MEGOLDÁSOK

kor m5-ban az osztás maradéka található, h5-ban pedig már megvan a hányados összes jegye.

A 0-val való osztás is tíz küszöbölhető ki. Ez akkor válik nyilvánvalóvá, amikor már 10 kivonás után sem lehet m5 < o5.

A kivonást a 2500-as Kivonás rutin végzi. Az írásbeli kivonásnál is használt algoritmus könnyen bővíthető. Az átvitelt mindig így tartalmazza.

A program befejező része (600-as soról) az eredmény kiírása. A kiírás előtti formálás célja a felesleges vezető nullák törlése.

Az új és a következő feladat az egyik legelső logikai játékhöz, a tizenötös játékhöz kapcsolódik. A végeredmény egy önállóan is jól használható játékprogram lesz, és a gyűjtők igényeit is kielégíti.

A játékot, amelyet bizonyára mindenki ismer, Sam Loyd találta fel 1878-ban. Megjelölése olyasféle szenciót keltett, amelyhez csak a büvös kockák mérhető. Mindmáig a legeljerjedtebb logikai játékok közé tartozik. Ezt bizonyítja a rengeteg műanyagból, fémből, fa-



```
10 REM -----
20 PRINT "Tizenötös Játék"
30 PRINT "1988.07.01."
40 PRINT "Pintér Gábor"
50 REM -----
60 REM -----
70 REM -----
80 REM -----
90 REM -----
100 REM -----
110 REM -----
120 REM -----
130 REM -----
140 REM -----
150 REM -----
160 REM -----
170 REM -----
180 REM -----
190 REM -----
200 REM -----
210 REM -----
220 REM -----
230 REM -----
240 REM -----
250 REM -----
260 REM -----
270 REM -----
280 REM -----
290 REM -----
300 REM -----
310 REM -----
320 REM -----
330 REM -----
340 REM -----
350 REM -----
360 REM -----
370 REM -----
380 REM -----
390 REM -----
400 REM -----
410 REM -----
420 REM -----
430 REM -----
440 REM -----
450 REM -----
460 REM -----
470 REM -----
480 REM -----
490 REM -----
500 REM -----
510 REM -----
520 REM -----
530 REM -----
540 REM -----
550 REM -----
560 REM -----
570 REM -----
580 REM -----
590 REM -----
600 REM -----
610 REM -----
620 REM -----
630 REM -----
640 REM -----
650 REM -----
660 REM -----
670 REM -----
680 REM -----
690 REM -----
700 REM -----
710 REM -----
720 REM -----
730 REM -----
740 REM -----
750 REM -----
760 REM -----
770 REM -----
780 REM -----
790 REM -----
800 REM -----
810 REM -----
820 REM -----
830 REM -----
840 REM -----
850 REM -----
860 REM -----
870 REM -----
880 REM -----
890 REM -----
900 REM -----
910 REM -----
920 REM -----
930 REM -----
940 REM -----
950 REM -----
960 REM -----
970 REM -----
980 REM -----
990 REM -----
1000 REM -----
```

ból készült „manuális” változata. A számítógépes megvalósítás azonban a mai napig kész. Ezt a hiányt ki-
vínyja pótolni a következő két feladat.

A játék 15 számozott, négyzet alakú lemezekből áll amelyek szabadon csúszkálnak egy négyzet alakú dobozban. A dobozban 4x4=16 hely van, ezért egy hely mindig szabadon marad. A játék célja egy összekevert helyzetből az ábrán látható kiindulási állapot visszaállítása. Az elrendezést úgy kell elérni, hogy a lemezek csak csúsztatni szabad, a dobozból kiemelni nem.

A játék sajátossága, hogy nem minden összekevert helyzetből rakható ki.

7. feladat: Tizenötös játék 1.

Írjon programot, amely kirakolja a tizenötös játékot, lehetővé teszi a számok megfelelő mozgását és össze tudja keverni a játékok! Ugyeljen arra, hogy az összekevert állás kirakható legyen!

PINTÉR GÁBOR

```
2010 REM -----
2020 REM -----
2030 REM -----
2040 REM -----
2050 REM -----
2060 REM -----
2070 REM -----
2080 REM -----
2090 REM -----
2100 REM -----
2110 REM -----
2120 REM -----
2130 REM -----
2140 REM -----
2150 REM -----
2160 REM -----
2170 REM -----
2180 REM -----
2190 REM -----
2200 REM -----
2210 REM -----
2220 REM -----
2230 REM -----
2240 REM -----
2250 REM -----
2260 REM -----
2270 REM -----
2280 REM -----
2290 REM -----
2300 REM -----
2310 REM -----
2320 REM -----
2330 REM -----
2340 REM -----
2350 REM -----
2360 REM -----
2370 REM -----
2380 REM -----
2390 REM -----
2400 REM -----
2410 REM -----
2420 REM -----
2430 REM -----
2440 REM -----
2450 REM -----
2460 REM -----
2470 REM -----
2480 REM -----
2490 REM -----
2500 REM -----
2510 REM -----
2520 REM -----
2530 REM -----
2540 REM -----
2550 REM -----
2560 REM -----
2570 REM -----
2580 REM -----
2590 REM -----
2600 REM -----
2610 REM -----
2620 REM -----
2630 REM -----
2640 REM -----
2650 REM -----
2660 REM -----
2670 REM -----
2680 REM -----
2690 REM -----
2700 REM -----
2710 REM -----
2720 REM -----
2730 REM -----
2740 REM -----
2750 REM -----
2760 REM -----
2770 REM -----
2780 REM -----
2790 REM -----
2800 REM -----
2810 REM -----
2820 REM -----
2830 REM -----
2840 REM -----
2850 REM -----
2860 REM -----
2870 REM -----
2880 REM -----
2890 REM -----
2900 REM -----
2910 REM -----
2920 REM -----
2930 REM -----
2940 REM -----
2950 REM -----
2960 REM -----
2970 REM -----
2980 REM -----
2990 REM -----
3000 REM -----
3010 REM -----
3020 REM -----
3030 REM -----
3040 REM -----
3050 REM -----
3060 REM -----
3070 REM -----
3080 REM -----
3090 REM -----
3100 REM -----
3110 REM -----
3120 REM -----
3130 REM -----
3140 REM -----
3150 REM -----
3160 REM -----
3170 REM -----
3180 REM -----
3190 REM -----
3200 REM -----
3210 REM -----
3220 REM -----
3230 REM -----
3240 REM -----
3250 REM -----
3260 REM -----
3270 REM -----
3280 REM -----
3290 REM -----
3300 REM -----
3310 REM -----
3320 REM -----
3330 REM -----
3340 REM -----
3350 REM -----
3360 REM -----
3370 REM -----
3380 REM -----
3390 REM -----
3400 REM -----
3410 REM -----
3420 REM -----
3430 REM -----
3440 REM -----
3450 REM -----
3460 REM -----
3470 REM -----
3480 REM -----
3490 REM -----
3500 REM -----
3510 REM -----
3520 REM -----
3530 REM -----
3540 REM -----
3550 REM -----
3560 REM -----
3570 REM -----
3580 REM -----
3590 REM -----
3600 REM -----
3610 REM -----
3620 REM -----
3630 REM -----
3640 REM -----
3650 REM -----
3660 REM -----
3670 REM -----
3680 REM -----
3690 REM -----
3700 REM -----
3710 REM -----
3720 REM -----
3730 REM -----
3740 REM -----
3750 REM -----
3760 REM -----
3770 REM -----
3780 REM -----
3790 REM -----
3800 REM -----
3810 REM -----
3820 REM -----
3830 REM -----
3840 REM -----
3850 REM -----
3860 REM -----
3870 REM -----
3880 REM -----
3890 REM -----
3900 REM -----
3910 REM -----
3920 REM -----
3930 REM -----
3940 REM -----
3950 REM -----
3960 REM -----
3970 REM -----
3980 REM -----
3990 REM -----
4000 REM -----
```



```

10 REM Digitalis montazskeszito by
Szarka Zoltan
20 Cls
30 Input "Milyen hosszuzagu montaszt
kivan betolteni(max25sec):"A
40 IF AC=0 OR A>25 THEN 20 ELSE A=INT(
A)
50 Print "1 sec kb 1250 byte tart
foglal el.Tehat az On allal kert
";A;" sec ";A*250;" byteot igyenel
"
60 Print "Es megfeleli Onnek (I-N)?"
70 IF Inkey="I" THEN 80 ELSE IF inkey
="N" THEN 20 ELSE 70
80 CH=8880-A*251 : IF CH<18000 THEN
Print "Nem fer el a tarban !" : End
Else X=CH-INT(CH/250)*250:Y=INT(CH
/250):W=A*250-INT(A*250)/250*
250:Z=INT(A*250)/250
90 Poke2000,33,59,64,62,127,119,211,0
,33,x,y,17,w,z,243,6,8,175,219,
0,31,31,31,203,17,197,6,10,0,10
,253,193,16,239,113,35,27,122,1
79,32,230,291,201
100 Print "Figyelem allitasa a magneto-
font lejatszasa es a RETURN le-
nyomasaval egyidoben inditasa is
el!"
110 Print "(RETURN)"
120 IF Inkey<>CHR$(13) THEN 120 ELSE
Beep 33,33
130 G=CALL(20000) : Print "A beolvasas
megtortent!"
140 CH=CH-150 : IF CH=>32768 THEN CH=
CH-65535
145 Print "Az inditasi cim:";Print Tab
(10);CH
150 PokeCH,33,x,y,17,w,z,243,78,6,8,20
3,17,159,197,6,10,0,16,253,193,254
,295,40,15,62,45,211,0,16,236,35,2
7,122,179,32,227,291,24,19,62,63,
211,0,16,221,35,27,122,179,32,212,
251,62
160 Poke CH=53,238,33,59,64,119,211,0,
201
170 Print "Nyomja meg az 'I'-t es a
montaszt megszallit!"
180 IF Inkey<>"I" THEN 180 ELSE G=
CALL(CH)
190 Print "Es megfeleli (I-N)?"
200 IF Inkey="I" THEN 210 ELSE IF
Inkey<>"N" THEN 200 ELSE Print "
Tisztalt felvetellei probalkozt!"
: Goto 30
210 Input "Milyen neven montasem":AS
: H=LEN(AS) : IF H>16 THEN H=16
220 Print "Figyelem kimentes kovetkezik
!" : Print "Allitasa felvetelleire a mag-
netot, ha kess,nyomja meg a RETURN-t
es inditasa el!"
230 Print "(RETURN)"
240 IF Inkey<>CHR$(13) THEN 240 ELSE
Beep 33,33
250 IF CH<>AS THEN CH=CH+65535
295 E=CH-INT(CH/250)*250:W=INT(CH/250)
200 Poke24000,33,92,64,62,131,119,33,2
16,89,205,146,0,33,94,64,62,1,119,
43,43,62,249,119,33,0,230,1,0,0,17
,e,r,205,149,0,33,94,64,126,60,39,
119,43,43,62,177,119,205,150,0,201
270 Poke23000,80,219,89 : FOR I=1 TO H
: Poke23003+I+1,ASC(MID(AS,I,1))
Next I
280 G=CALL(24000)
290 END

```

A magazin 1988/7. számában megjelent Hangdigitalizálás programon belüli alkalmazását szemlélteti ez a rövid példaprogram. Akik kipróbálták a digitalizáló programot, bizonyára észrevették, hogy a hangokat zajosan kapták vissza. Ennek oka a szalag és a felvétel minősége, valamint a kis piezoelektromos „hangszóró”. A montázskészítő programnál ezt figyelembe kell venniük, tehát lehetőség szerint

PRIMÓRA

jó minőségű felvételt juttassunk be. Legtisztábban szintetizátor-, gitár- és más számítógéphegnyakat kaphatunk vissza.

Ennek a programnak a segítségével adott időtartamú hanganyagot tölthetünk számítógépünkbe, majd ezt fájként magnóra vehetjük. Ezáltal lehetőség kínálkozik arra, hogy valamelyik programunkba töltsük és használjuk. A fájl LOAD utasítással olvashatjuk be.

Figyelem! Minden egyes másodperc 4,8 blokknak felel meg! A program 64-es gépre készült a nagy memóriáigény miatt. A másodpercek időtartama csak a gyári frekvencián futó gépeknél érvényes. Működéskor nagybetűket használjunk és ügyeljünk a számok pontos begépelésére is.

Hogyan indíthatjuk el az elkészült és kimentett montaszt? A program futásakor kiírja az indítási címet, ezt jegyezzük fel, mert a később betöltendő montaszt ezen a címen kell meghívunk. A hangfájl használata tehát a következő:

— Elkészítjük a felhasználó-programot, ügyelve arra, hogy a fájl ne jusson a memória területére (ami a képernyő-memória előtt másodpercszer 1250 bajtot jelent).

— A programunk elején LOAD-dal betöltjük a fájlt.

— Szükség esetén a következő utasítással szóalathatjuk meg: ?CAU (feljegyzett cím) vagy A=CAU (feljegyzett cím).

SZARKA ZOLTÁN



RESET SAJÁTKEZÜLEG

Az Egy gép száz bajt csinál? című riportban (1988/11. szám) szó volt a TVC-vel kapcsolatban a RESET gomb hibájáról, amelynek orvoslását — a konstrukció átalakítását — a gyár nem vállalja.

A lényeg az, hogy a fent említett kapcsoló egyszerű lenyomása meleg, kétszeri lenyomása hideg RESET-et eredményez. Ezen házilag elég könnyű segíteni. Miután kéznél van egy pillanatkapcsoló, egy vékonyabb vezeték és egy páka, leszereljük a számítógép hátlapját. Itt a RESET lapka alatt egy egyszerű kis érintkezőt találunk. Ennek két végpontjához hozzáforsztjuk a vezetékeket, majd kivesszük a gép burkolatán kívül elhelyezett pillanatkapcsolóhoz.

Gépeken például a kapcsolót a botkormány-csatlakozó „kupakjához” ragasztottam, és a kéteres ragasztott vezetéket a NYÁK széléhez kis szigetelőszalag-csíkokkal erősítettem fel. Szerencsére a kapcsoló nincs útban, s a kupakot is nehezebb elveszteni.

A megoldás azóta is beválik, és a kapcsoló egyszerű megnyomása után nem kell azon izgulni, hogy vajon elszáll-e a program. Természetesen kétszeri gombnyomásra teljes RESET hajtódik végre, és a régi gomb is ugyanígy működőképes marad.

Nem árt arról sem meggyőződni még beszerelés előtt, hogy a pillanatkapcsoló tényleg csak a gombnyomás idején zárja az áramkört és nem fordítva.

Természetesen az átalakítással érdeemes megvárni, amíg a garanciaidő lejár, és csak utána szabad leszedni a viaszpecsétet.

VAMOS SÁNDOR

Korszakváltás, de hogyan...?!

Bizonyára az olvasók közül is néhányan hallottak az 1988. augusztus 19-i *Korszakváltó* vetélkedőről. Ez olyan történelmi játék volt, amelynek keretében az ország különböző pontjain lévő csapatok versenyeztek egymással. A kapcsolatot rádióval és egy számítógépes rendszerrel tartották, vagyis... Minden nagyon szép és jó lett volna, ha...

A valóság azonban egy kicsit mást hozott, mint amit a rendezők vártak. A rádiós kapcsolat még csak létezett, de a számítógép körül nem jött minden össze. Ezt egy ún. MUPID rendszerrel szeretnék volna megvalósítani, s így telefonvonalakkal összekötni a helyszíneket. Az elgondolás nagyszé- rű volt, de sajnos figyelmen kívül hagyta a mai telefonhelyzet problémáit. Ezek után úgy gondoltam, mindenki sejtje már a fejleményeket is. Nos, mi Siófokon voltunk, s az alábbiak az ott történteknek a leírása. Mellesleg megjegyzendő, hogy az eset egyáltalán nem kirívó, sőt, nagyon is általános.

1988. augusztus 18. Délelőtt érkeztek a helyszínre, s így bőven volt időnk a gépeket a helyükre tennünk. Beállítottuk a programot a gépbe, majd vártuk a 13.00—13.30-ig tervezett főpróbát. A megadott időben megszólalt a telefon. Első kísérletre bejutunk a rendszerbe, örömmünk azonban korai volt. Tíz másodperc elteltével lefagy a rendszer. Újabb hívás — várjunk —, vártunk fél négyig. Majd a telefonközpontra bejelentette, hogy sajnos már nem tud kapcsolni, visszaadja a vonalat. Saját hívásra viszont sikeredett a rendszerbe kerülni. Üzenet mindenkinek, hogy vagytok, milyen utatok volt stb... Majd egy körlevél mindenkinek: ki hisz abban, hogy holnap sikeres lesz a számítógépes kapcsolat? A válaszok többnyire a MI NEM üzenetet tartalmazták. Félóra múlva a központ kitörölte a körlevelet, majd nyomatékossá megkért mindenkit, hogy ne használjuk ezt az információküldési módot, a gondokat holnapra megoldják.

Augusztus 19. Az előkészületek ugyan nem voltak zökkenőmentesek — a magnót hallani lehetett a telefonban, s néha a rádió is foglaltat jelzett —, de reggel kilenckor már várhattuk a központ hívását. Hiába. Már elmúlt tíz óra, mikor végre bejutottunk a rendszerbe. Tíz percre. Az első feladatot ugyan utólag mondták el telefonon, de a megoldást elküldték. Többet azonban nem tudtunk tenni, ugyanis a nap folyamán mint-

egy háromperces órát használhattuk a gépet, 2-3 perces intervallumokban, ami épphogy elég volt a bejelentkezésre, de a válaszküldésre már kevésbé. Így aztán minden létező lehetőséget megragadtunk az eredmény Budapestre küldésére (telefonon, telex stb...). Déltájban a játékvezetőség úgy döntött, hogy a tizenöt helyszínből tiznél a számítógépes vonalat megszünteti, s egyszerű telefonvonalra kapcsolja át. Ezt velünk ugyan közölték, de mivel a Petőfi adó időnként előadásban kapcsolódott be a játékba, a nagyközönség is igényelte volna a teljes informáltságot. Végül az unszólóknak engedve kiderült a probléma technikai oldala. Hasonló volt a helyzet a televízióval is. Néhány helyszínen felvétel készült a másnapra adáshoz, s függetlenül attól, hogy az összeköttetés már rég nem létezett, a műsorban azért bemutatják a számítógépes munkát is. Ez persze a kívülállókknak aztán fel sem tűnt. Mindezek ellenére sikeres lezajlottak a délutáni fordulók is. A játék kinek jobban, kinek kevésbé, de szerencsésen befejeződött. Igaz, az egyik feladat szerint tüzet is kellett rakni, annak ellenére, hogy ezt több helyszínen tiltották, de baj nem lett ebből.

A későbbiek folyamán beszélgetünk olyanokkal is, akiknek minden rendben ment. Ők meg voltak ugyan elégedve a dologgal, de azt nehezményezték, hogy maximum hatvanöt percig tudtak rendszerben maradni, utána a központ automatikusan kiiktatta őket.

Szomorú tapasztalat, hogy ilyen nagy rendezvény megtartásához alkatlan a mai telefonhálózat. Persze ez nem csupán a posta bűne, állítólag a számítógéppont sem volt tökéletes a verseny idején. Volt, aki utánajárt az ügynek a VEIKI-nél. Miért nem tudott kézi hívással belépni a rendszerbe, holott minden feltétel adott volt? A nyomozás kemény próbának bizonyult. Először is nem tudták biztosan megmondani, hogy melyik osztályt kell az ügyben keresni, s ha akadt is valaki, akinek némi köze volt a dologhoz, csak hosszú idő után mutatta jelét, hogy sejtje miről van szó. Mindez azonban csak növelte az utóregzések feszültségét. Mert tudták róla, hogy valamilyen gépek részt vettek ugyan valamilyen vetélkedőn, de semmi másról, ugyanis is az esetleges felelősséget elhárították maguktól. Ezek után felmerül a kérdés: lesz-e jövőre Korszakváltó?

Bártfai Barnabás

```

1 CLEAR S9999
3 GO SUB 300
10 LET A=1090
20 POKE A,108
30 LET A=A+1
40 FOR N=1 TO 30
45 CLS
50 READ A
55 PRINT AT,0,0,0,A
60 FOR B=1 TO 100 LET A=B
70 POKE B,CODE A*(B+A+1)
80 NEXT B
90 POKE B,141
100 LET B=B+1
110 LET A=B
120 NEXT H
130 POKE 5999A,1931 POKE 59997,237
140 POKE 23610,28
145 GO TO 9997A8 TO 1A3091 POKE A+443A8,
200 FOR A=151616 TO 1A3091 POKE A+443A8,
PEEK X NEXT X
210 FOR A=997A8 TO 1A3091 READ B: POKE
B,2 NEXT B
250 LET A=997A4
300 FOR A=1 TO 141 READ X: FOR B=0 TO 7
1 READ Y: POKE A+(X*B)+Y, NEXT B: NEXT
A
400 POKE 2360A,961 POKE 23607,233
500 RETURN
8500 DATA 283,84,4V,3,205,14V,23,205,37A
-2,62,0,205,1,22,205,44,15,205,33,27,25
3,203,0,136,32,18,253,203,48,102,40,64,4
90,92,205,119,2,253,94,0,205,24,221,4
2,89,92,34,93,92,205,251,25,120,177,194,
80,238,223,254,13,40,192,253,203,48,70,1
94,175,12,205
100,253,203,120,13,62,205,241,71,254,10,36,2
0,0,253,114,55,253,114,39,34,11,92,33,1,
0,34,22,92,205,179,22,253,203,95,174,205
110,13,253,203,120,13,62,205,241,71,254,10,36,2
190,2,205,239,21,62,32,215,120,17,162,2
36,705,10,12,237,76,40,205,27,26,62,5
918,253,78,13,6,0,205,27,26,205,181,18
58,58,92,40,40,27,254,9,40,4,254,21,32
3,253,92,13,1,3,0
8510 DATA 17,2,92,93,38,88,92,203,128,40
1,9,237,184,203,94,10,255,203,203,1,198
198,106,237,82,16,1,0,0,198,209,237,237
67,73,92,40,93,92,239,33,72,286,209,42,9
7,92,85,237,82,239,94,105,205,110,25,32
6,205,194,25,205,232,25,193,121,61,176,4
0,40,187,3,3,3,83,27,91,83,93,315,205
85,22,225,34,83,92,73,197,19,42,97,92
43,43,237,184,42,73,92,235,193,112,43,11
3,-43,115,43,114,261,195,94,237
9000 DATA 33,8,0,0,0,0,8,0,2,0
9010 DATA 37,8,0,74,66,126,66,66,0
9020 DATA 38,8,0,74,66,126,66,66,0
9030 DATA 39,20,66,66,66,66,66,66,0
9040 DATA 43,16,16,66,66,120,66,66,0
9050 DATA 64,16,16,0,49,16,16,54,0
9060 DATA 91,20,60,66,66,66,66,66,0
9070 DATA 92,20,60,66,66,66,66,66,0
9080 DATA 93,40,60,66,66,66,66,66,0
9090 DATA 95,20,60,66,66,66,66,66,0
9100 DATA 96,8,126,72,124,64,66,126,0
9110 DATA 123,40,0,56,66,66,66,54,0
9120 DATA 124,16,16,56,66,66,66,54,0
9130 DATA 125,40,60,66,66,66,66,54,0
9140 DATA 126,60,74,66,66,66,66,54,0
9150 DATA 127,40,0,69,48,48,54,0
9999 DATA "0*7",NEXT A:DATA:FOR A=141
TO 1,"Nincs ilyen vektor!"-"Névtelen üze-
n!"-"Nincs hely a memóriában!"-"Nincs hely
a memóriában!"-"Túl nagy szöveg!"-"RETURN ut
asítás GO SUB vektor!"-"File v99!"-"STOP
utasítás!"-"Névtelen argumentum!"-"A to
rlemből nem lehet elszármaztatni A BASIC-ben"
Ilyen nincs!"-"Megszakítás a CONT írással!"
-"Nincs több adat a DATA-ban!"-"Névtelen
argumentum!"-"Túl nagy szöveg!"-"RETURN ut
asítás STOP!"-"Névtelen string!"-"FN u
tasítás DEF nőtlen!"-"Megszakítás hiba!"-"
Hiba a hozottárról betöltéskor!"-"!";
9999 C=90 B=91

```

MAGYAR NYELVŰ HIBÁZJELETEK

A programmal bármilyen hosszú üzenetet definiálhatunk ZX-Spectrumon, csak ügyeljünk arra, hogy a hibazület ne tartalmazzon 128-nál nagyobb ASCII kódú karaktert, mivel ez az üzenet végel jár. Így az üzenet valódi, utolsó kódját úgy kapjuk meg, hogy kivonunk belőle 128-at.

A program tartalmaz még egy ékezetes karakterkészletet is:

| ASCII KÓD | KARAKTER | ASCII KÓD | KARAKTER |
|-----------|----------|-----------|----------|
| 37 | A | 91 | Ü |
| 38 | á | 92 | ú |
| 96 | E | 93 | ő |
| 63 | é | 123 | ü |
| 33 | í | 95 | ű |
| 64 | í | 127 | Ű |
| 126 | Ó | 125 | Ű |
| 124 | ó | 39 | ü |

A program NEV hatáására kikapcsol. Beindítást a következőképpen lehet:
POKE 59996,193:POKE 59997,237
POKE 23606,96:POKE 23607,233

A listában található csillagok helyett egy 'c)' karaktert kell begépelni.

Csoma Roland

Gyakran hallhatunk számítógépes berkeken dúló nagy vitákról, amiket programnyelvek művelői folytatnak, s természetesen ki-ki a saját maga által használt nyelvre esküszik. Az persze nem vitás, hogy minden nyelvnek van előnyös és hátrányos oldala is, de most nem ez a lényeg. *Irássommal nem részletezni akarom a nyelveket, hanem feladatorientáltan áttekinteni.* Tehát ne nyelvismertetést várjon az olvasó!

Minden gép csak a számára értelmezhető bítformációkat érti meg. Gondoljuk el, hogy ha BASIC-ben leírjuk a következő utasítást: FORI=1TO10:PRINTI:NEXTI, az hány rendszerrutin hívását idézi elő fordításnál, illetve ha interpreter van, akkor futásnál. Az interpreter=értelmező; a mikrogepek általában interpreterrel vannak ellátva (lásd BASIC). A compiler=fordító; ilyenkor a tárgyprogram csak egyszer fordítódik le, és ezek után x-szer futtatható.

No de visszatérve a nyelvekhez, alapul a gépi kódot kell venni, ahova az assembler is sorolható, ami nem más, mint a gépi kód utasításainak karakteres szintaktikában való leírása.

A rendszerprogramozók — már akik nyelveket fejlesztettek — több szempont szerint alkottak, pontosan három szerint, amelyek végül matematikai, adatfeldolgozási és általános feladatokra orientálódtak.

Később a matematikai ág egybeolvadt az adatfeldolgozóval. Ilyen matematikai, de az adatfeldolgozó nyelvek csoportjában található az ALGOL60—68 és a Modula-2. Ezekben a nyelvekben olyan utasítások vannak, amelyek nagymértékben építenek a gép aritmetikai műveleteire, és logikai megoldókészségére is, ahol is nagyon fontos a műveleti sebesség.

Ebben az adatfeldolgozó ágban található azokat a nyelveket is, amelyek igen magas szintűek, de mind az adatfeldolgozást, mind a matematikai műveleteket használják, és mégsem általános nyelvek. Ezek a PL/I, a Pascal és az Ada. Persze az igazán adatcentrikus nyelv a COBOL és az OCCAM.

Mint minden nagy, illetve óriási adattömeghez, szükséges a jól szervezett fájlkezelés is, tehát ezek a nyelvek ezt messzemenően támogatják.

A másik nagy csoport, vagyis ág az általános célú nyelveké. Ennél a fejlesztők arra törekedtek, hogy az adott hardverbázis összes lehetőségét elérhesse a felhasználó, ezért viszont fel kellett áldozniuk a részletességet — az assemblerrel ellentétben,

ahol minden elérhető —, de ezért nagyon egyszerűen kezelhető, magas szintű nyelvet kap a felhasználó. Erre tipikus példa a BASIC, amely rendelkezik matematikai függvényekkel, fájlkezelő utasításokkal, adatfeldolgozó parancsokkal, grafikai lehetőségekkel, hanggenerálással stb.-vel.

Igazán azonban egyiket sem lehet használni nagyobb feladatnál! Sok programozó nem szereti a BASIC-nyelvet az említett hibák miatt, főleg, hogy általában PC-n, illetve nagygépes rendszerekben több különböző nyelven írt fájlok összehívezhetőek egy értelmes programfájlra. Egyébként én is ezzel az elvvel értek egyet, annak ellenére, hogy szerintem a BASIC nagyon jó arra, hogy megismerkedve vele, a kezdők látszák, mi

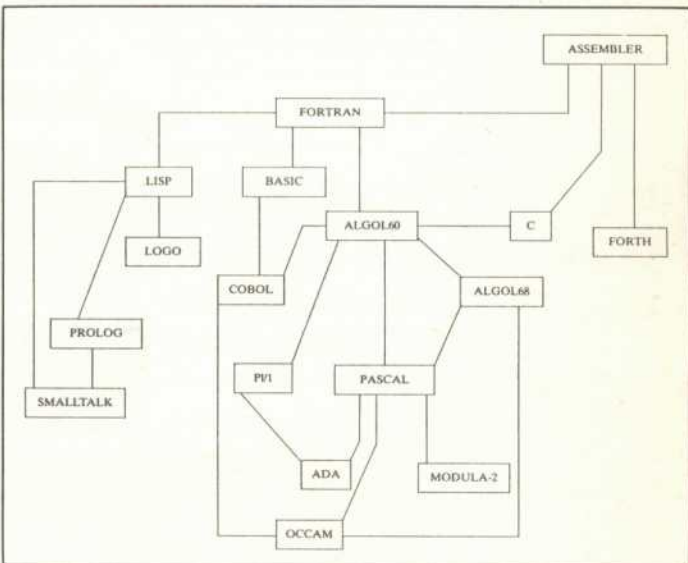
mindent lehet elérni a számítástechnikával. De a BASIC-tudással nem szabad megelégedni!

Az általános nyelvek új irányzatot követnek a nyolcvanas évek eleje óta, amikortól kezdve ugyanis a grafika kapott nagyobb teret, és ezért egyre inkább követelménnyé vált a látvány a tudás mellett, de nem ennek terhére. Ezért a programnyelvek is kezdték kihasználni ezt a lehetőséget is, mint a LOGO és a G-Pascal.

A jövő mindenesetre a grafikaé, tehát az biztos, hogy az ikonvezérlés nem újdonság a PC-k világában, ahol már a DOS is így működik, például az Apple PC-kben.

BARTOS GYULA

(A CHIP Professional ábrája alapján)



Listánkat felhasználói, illetve játékegyesítőkből állítjuk össze. A legjobbakat, legérdekesebbeket a beküldött javaslatok alapján rangsoroljuk. Ehhez kérjük az olvasók közreműködését. C64-re, ZX-Spectrumra. Enterprise-ra, ATARI-ra és IBM-re készült programrangsorokat várunk havonta.

Címünk: Mikroszámítógép Magazin Szerkesztősége 1371 Bp., Pf. 433.

Diákszerkesztőség

TOP - lista

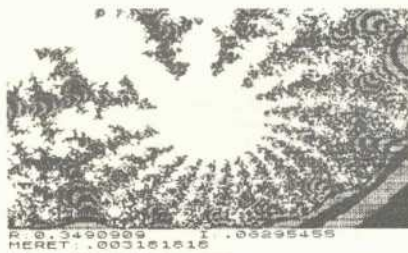
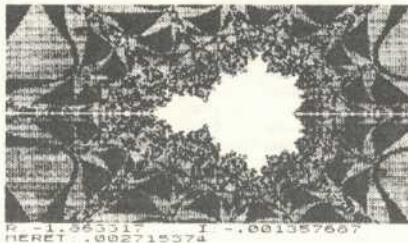
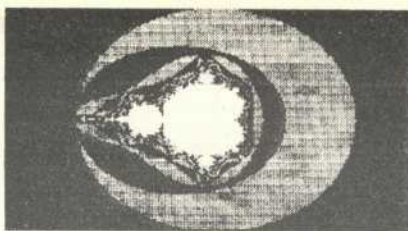
| Felhasználói | | Játék | |
|--------------------|------------|----------------------|------------|
| 1. News room | CG4 | 1. Defender OTC. | CG4 Amiga |
| 2. GEOS 1.32 | CG4 | 2. Pentis | IBM |
| 3. Newsmaster | IBM | 3. Test Drive | CG4 |
| 4. IS DOS | Enterprise | 4. Flight sym. II. | CG4 |
| 5. Printmaster | CG4 IBM | 5. The Last Ninja | CG4 Spect. |
| 6. Windows | IBM Ent. | 6. Skyfox II. | CG4 Spect. |
| 7. 128 Starpainter | CG4 | 7. Out run | CG4 |
| 8. Rockmonitor III | CG4 | 8. Impossible m. II. | CG4 Spect. |
| 9. Printfox | CG4 | 9. The Guild of t. | CG4 |
| 10. Game maker | CG4 | 10. Blood and guts | CG4 |

FIGS

```

10 REM      MANDELBROT
20 REM
30 REM      TURBO-SOFT 1988
40 REM
50 DIM S$(15): LET W=256/176
60 FOR I=0 TO 13: READ A: POKE 23296+I,A
: NEXT I
70 POKE 23658,8: BORDER 7
80 INPUT "LOAD VAGY INPUT "; LINE T$
90 IF T$="L" THEN GO SUB 440: GO TO 230
100 INPUT "R:";A: INPUT "I:";B: INPUT "M:";
:0
110 LET SW=0: LET P=0/176: BORDER 7: CLS
120 PRINT #1;"R:";A;TAB 15;"I:";B: PRINT
#1;"MERET:";0;
130 FOR N=0 TO 175: LET CI=B+N*P
140 FOR M=0 TO 255: LET CR=A+M*P
150 LET X=0: LET R=0: LET I=0
160 LET R1=R*R-I*I+CR: LET I=2*R*I+CI: LET
R=R1
170 LET Z=SQR (R*R+I*I): LET X=X+1
180 IF NOT ((X=100) OR (Z>2)) THEN GO TO
160
190 IF X=100 THEN GO TO 220
200 LET S=((X/2)-INT (X/2))*2
210 IF S OR SW THEN PLOT M,N
220 LET SW=NOT SW: NEXT M: LET SW=NOT SW:
NEXT N
230 LET CA=0: LET X1=128: LET Y1=88: LET
Y2=0
240 GO SUB 500
250 LET A$=INKEY$
260 GO SUB 500
270 IF A$="" THEN GO TO 240
280 IF A$="S" THEN POKE 23307,194: POKE
23308,4: RANDOMIZE USR 23296
290 IF A$="L" THEN GO SUB 440
300 IF A$="N" THEN LET CA=0: BORDER 7
310 IF A$="6" THEN LET X1=X1-1: LET Y1=Y
1-CA/W
320 IF A$="7" THEN LET X1=X1+1: LET Y1=Y
1+CA/W
330 IF A$="9" THEN LET Y1=Y1+1: LET X1=X
1+W*CA
340 IF A$="8" THEN LET Y1=Y1-1: LET X1=X
1-W*CA
350 IF A$="0" AND CA THEN LET A=A1: LET
B=B1: LET O=ABS ((Y1-Y2+1)*P): GO TO 110
360 IF A$="0" AND NOT CA THEN LET CA=1:
BORDER 0: BEEP 1,1: LET A1=A+P*X1: LET B1=
B+P*Y1: LET Y2=Y1: LET X2=X1
370 IF A$=" " THEN GO TO 70
380 IF X1<0 THEN LET X1=0
390 IF X1>255 THEN LET X1=255: IF CA THE
N LET Y1=Y2+(X1-X2)/W
400 IF Y1>175 THEN LET Y1=175: IF CA THE
N LET X1=X2+(Y1-Y2)*W
410 IF Y1<0 THEN LET Y1=0
420 IF CA*(Y1<Y2) THEN LET Y1=Y2: LET X1
=X2
430 GO TO 240
440 LET S$="": POKE 23307,86: POKE 23308,
5: RANDOMIZE USR 23296
450 FOR I=2 TO 14: LET S$(I-1)=SCREEN$( 2
, I): NEXT I: LET A=VAL S$
460 LET S$="": FOR I=17 TO 31: LET S$(I-1
6)=SCREEN$( 22, I): NEXT I: LET B=VAL S$
470 LET S$="": FOR I=6 TO 20: LET S$(I-5)
=SCREEN$( 23, I): NEXT I: LET O=VAL S$
480 LET P=0/176: LET CA=0: BORDER 7
490 RETURN
500 OVER 1: PLOT X1,Y1
510 IF CA THEN DRAW X2-X1,0: DRAW 0,Y2-Y
1: DRAW X1-X2,0: DRAW 0,Y1-Y2
520 OVER 0: RETURN
530 DATA 221,33,0,64,17,0,27,62,255,55,20
5,194,4,201

```



ILYEN VOLT, ILYEN LETT

MANDELBROT

Ezzel a programmal a Mandelbrot-halmaz különböző részeit nagyíthatjuk ki a Spectrum képernyőjére. A halmaz egy területének megadásához három adatra van szükség: R a nagyítandó téglalap bal alsó sarkának X koordinátája, I az Y koordinátája, a MERET pedig a téglalap Y irányú oldalhosszúsága. A terület számításának alapadatai egy-egy ábra alatt láthatók.

Az alaphalmaz az $R = -3, I = -2, MERET = 4$ képpen helyezkedik el. A fehér terület nem része a halmaznak. Ez érdekes képek mindig a halmaz szélén helyezkednek el.

A program indítása után döntenünk kell, hogy egy korábban elkészített képet nagyítunk tovább, vagy koordináták alapján készítünk egyet. Ha kezdetáról töltünk be képet, akkor a program az adatait automatikusan megállapítja. Amikor a gépben egy kész kép van, akkor ezt a kép közepén vibráló

pont jelzi. Ilyenkor kimenthetjük a képet kazettára S-sel, vagy betölthetünk egyet L-lel. Tovább nagyíthatunk is: a vibráló pontot Interface II botkormánnyal mozgassuk a nagyítandó terület bal alsó sarkába (6,7: bal, jobb; 8,9: le, fel) és nyomjuk meg a tűzgombot (0). Ekkor a border feketére vált, és a nagyítandó terület jobb felső sarkát tudjuk mozgatni. Ha a bal alsó sarok nem a megkívánt helyen van, akkor az N billentyű megnyomása után ismét a bal alsó sarok helyét lehet mozgatni. A jobb felső sarok beállítás után a tűzgomb hatására elkezdődik a nagyítás. Ez a művelet — különösen akkor, ha sok a fehér terület — elég időigényes, ezért érdemes a programot lefordíttatni egy olyan fordítóval, amelyik kezeli a törtszámokat (ilyen például a TOBOS FP). Lefordítva egy kép egy-négy óra alatt készül el.

NAGY ZOLTÁN



Egy sarokkal olcsóbb!!

Az ENTERPRISE COMPUTERS GMBH cég felajánlotta szerkesztőségünknek, hogy részt vesz árengedményes akciónban. Aki Münchenben jár — ami a világútlevel korábban már nem csupán vágyalom — és leadja a magazinból kivágott sarokszelvényt, az az alább felsorolt ENTERPRISE gépek, tartozékok, könyvek árából 15% en-

gedményt kap. Egy sarokszelvény — az eddigieknek megfelelően — csak egy-egy tétel engedményére jogosít. Tehát ha valaki egy csomag kazettát és esetleg egy könyvet is vásárol, annak két szelvényt kell leadnia.

A CÉG AJÁNLATA:

| | |
|---|--------|
| ENTERPRISE 128K (német BASIC cartridge) | 298 DM |
| ENTERPRISE 64K (angol BASIC cartridge) | 198 DM |
| EXDOS floppyvezérlő + kézikönyv | 148 DM |
| FLOPPY EGYSÉG (5,25 inch, egyoldalas, 40 sávós, 180 Kbájtra formattál) | 198 DM |
| FLOPPY EGYSÉG (5,25 inch) + vezérlőegység | 298 DM |
| „EP80+” típusú nyomtató (100 kar/s)+ illesztőkábel | 348 DM |
| SPEAK—EASY beszédszintetizátor | 69 DM |
| Mindegyik programkazetta | 15 DM |
| Programkazetta-csomag (3 db) | 40 DM |
| Programkazetta-csomag (10 db) | 100 DM |
| BASIC oktatókönyv | 19 DM |
| EXDOS kézikönyv | 19 DM |
| 15 DOS kézikönyv+ floppy | 29 DM |
| EXOS (műszaki kézikönyv) | 39 DM |
| ENTERPRISE 128K kézikönyv (265 oldalas, német nyelvű) | 10 DM |
| Nyomtatókábel (CENTRONICS, 8 bites, párhuzamos) | 24 DM |
| Botkormány-kábel | 24 DM |
| Monitorkábel (SCART, CYNCH csatlakozóval vagy anélkül) | 24 DM |
| RS232C kábel soros nyomtatókhoz | 24 DM |

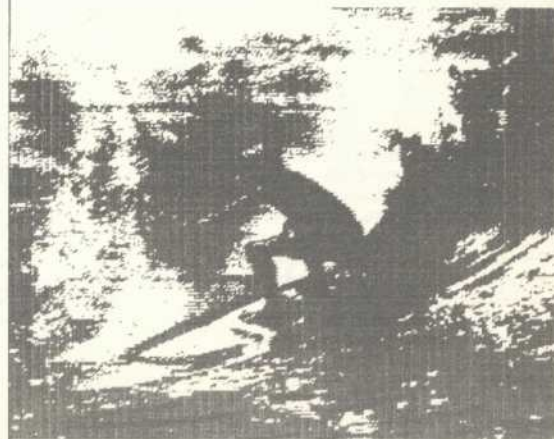
A müncheni cím:
ENTERPRISE COMPUTERS
GMBH
8000 MÜNCHEN 2.
SONNENSTRASSE 3.
NSZK

ENTER-NEWS

1/88 Jan.-März

Die Zeitschrift für den Enterprise-Computer 64K und 128K 10,-DM

- DO-IT-YOURSELF-JOYSTICK-INTERFACE
- HARDWARE-ERWITTERUNGEN
- DIGITALE STOP-UHR
- MEMO-DASGEDACHTNISSPIEL
- TIPS+TRICKS



BASIC

ÉS GÉPI KÓD

Most az augusztusi számban megjelent írásommal kapcsolatos olvasói észrevételekkel és azok tanulságaival foglalkozom.

Három levelet kaptam. Néhányan szóban fejtették ki véleményüket munkahelyemen, a Magiszter Könyvsbolt számítástechnikai részlegén.

Abban mindenki egyetértett, hogy a programlistát és a hozzá tartozó magyarázatot nem szabad elszakítani egymástól. A bemutatott listákkal kapcsolatban megoszlanak a vélemények.

Szabó Péter Pál levelében az augusztusi számban közölt 1. és 2. programlista helyett azok kombinációját ajánlja, hasonlólt a HELP-PLUS, illetve a PROFI-ASS 64 kimeneti listájához. Ifjabb Fekete László a disassemblált alakra szavaz, az assembly és szimulátorlista közlését felesleges helypaparásnak tartja. Scherr Zoltán szerint a szimulátor listája is hasznos lehet néha.

A véleményeket egybevetve, Szabó Péter Pál javaslatát tartom a legcélszerűbbnek, bár a megvalósítással valószínűleg gondjaim lesznek.

Míg Szabó Péter Pál csak a technikai jellegű kérdéseimhez szól hozzá, a másik két levélíró a cikksorozat folytatása mellett foglalt állást.

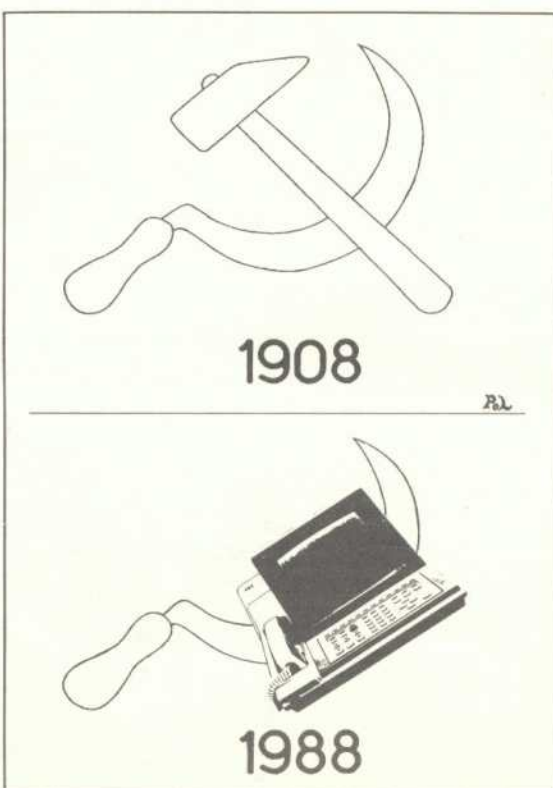
Ifjabb Fekete László hasznos tanácsokat ad, ezenkívül rengeteg témát sorol fel, amelyekről szívesen olvasna. Remélem, ötleteit sikerül megvalósítanom.

Scherr Zoltánnak VC20-as gépe van. ... feltétlenül legyen figyelemmel a VC-20-as sajátosságaira a továbbiakban is, mint eddig" — írja többek nevében is. Levelének más részében a következőket olvasom: „A korábban megjelent információk ismételtetését fe-

Legutóbb a bináris számokat BCD-re alakító rutinnak egy olyan változatát ismertettem, amely a szám elején álló értéktelen nullákat elhagyja, majd néhány, a téma további alkalmazására vonatkozó ötlet felvetésével befejeztem a BCD-aritmetika bemutatását.

leslegesnek tartom. Kivételes esetekben is elég a megjelenés helyére utalni." Könyvsbolti beszélgetőtársaim többsége éppen ellenkező véleményen

volt, és véleményük alapján tudatosult bennem, hogy mit hibáztam el, és miért nem érdemes ezt a cikksorozatot a jelenlegi formájában folytatnom.



A „Gép is ember” c. karikatúrapályázat harmadik helyezett alkotása, LOUIS POSTRUZIN munkája.

Eredeti elképzelésem az volt, hogy az alapoknál kezdve úgy épitem fel cikkeimet, hogy minden új információ a már korábban közöltekre támaszkodjon. Csupán arról feledkeztem meg, hogy ez a módszer egy könyvben jó lehet, de egy folyóiratban megjelenő cikksorozat esetén teljesen elhibázott. Új olvasók kapcsolódnak be, akik, nem ismervén az előzményeket, értetlenül fogadják az újabb ismereteket is. Még annak is nehéz, aki a lap korábbi számainak birtokában próbál újonnan bekapcsolódni a sorozat olvasásába, hát még annak, akinek a korábban megjelent példányok hozzáférhetetlenek.

A másik hiba a munkamód-szeremben rejlett. A sorozatot folytatásosként írtam, legtöbbnyire a szorongató határidő árnyékában. Néha ez volt az oka a programlista és a magyarázat különválásának is. A „határidőre kész kell lenni, mert egy folytatás sem maradhat ki” hangulata olykor kapkodáshoz vezetett. Előfordult, hogy „raktárból” vettem elő egy, a korábbiakkal össze nem függő témát, mert a szükséges program nem akart „összeállni”.

A harmincegyedik folytatással befejezem az összefüggő BASIC és gépi kód című sorozatot. A cím néha megjelenik a lap hasábjain, de ritkábban, és csak akkor, ha valóban van új mondanivalóm. Újabb cikkeket tervezek a BASIC és gépi kódú programozás témaköréből (most már külön-külön), de ezekkel csak akkor jelentkezem, ha „teljesen kész van az anyag”.

Köszönöm az olvasók türelmét, és külön köszönöm a levélíróknak, mert nem sajnálták a fáradságot, hogy segítsenek.

BARNA LÁSZLÓ

Hibajelzések és üzenetek

Ötrészes sorozatunkban mindarról szót ejtünk, amit olvasóinknak érdemes átgondolniuk, ha programozásra adják a fejüket, vagy ha valamely programról véleményét kívánják alkotni. Ezek az írások a szerzőnek lapunkban közölt korábbi fejtegetéseivel együtt (utalunk „Az adattípus fejlődése” és a „Függvények és utasítások” című sorozatokra, lásd 1987/8. és 9., valamint 1987/12., 1988/1. és 2. számainkat), teljeskörösen és közzérthetően elemzik a sorozat címében jelzett (problematicát) témát.

Tegyük fel, hogy van egy teljesen hibátlanul vélt programunk. Vajon száz futásból hány futása lesz hibátlan? Programja váltogatja. Van, amelyiknek kilencven, van, amelyiknek húsz.

Képzeljük el a legideálisabb esetet, egy bemenő adatot nem igénylő, egyszerű kis programcskát. Például azt, amelyik betölti a nyomtatóba a magyar karakterkészletet. Ezt a programot gondosan megírták, belőték, és naponta legalább egyszer gond nélkül használják. Ha elindítjuk, biztosan jól működik, kivéve, ha nincs bekapcsolva a nyomtató, vagy off line-ba van kapcsolva, vagy kilazult a csatlakozó.

Kivételes esetben a következők fordulhatnak elő: a program lefut, mintha mi sem történt volna; a program „abortál” valamilyen nehezen érthető üzenet kíséretében (például „nyugtáztatlan üzenet” vagy „nem jött meg a 'done' bit” vagy „ERROR 2749” stb.); a program szó nélkül leragad és addig vár, amíg rá nem jövünk, mi a hiba; végül jöhet valami értelmes üzenet is, például mindössze ennyi: „nézd meg, mi van a nyomtatóval”.

Vegyünk egy példát a másik végteléből is, mondjuk egy fordítóprogramot. Általános tapasztalat, hogy legalább tízszer el kell indítani a fordítóprogramot, amíg a felhasználó programja először futóképes lesz, és utána minden javításnál kétszer, háromszor.

A fentiekből egyrészt az következik, hogy nincs olyan program, amelyben a hibajelzésnek nincs szerepe, másrészt pedig az, hogy különböző programoknál különböző súlya van a dolognak.

A MINIMÁLIS TENNIVALÓ

A hibakezelésnek szükségszerű feltétele, hogy a hibát észrevegyük. Rengeteg időt és energiát takaríthatunk meg azzal, hogy ak-

kor vesszük észre, amikor keletkeznek, és nem akkor, amikor a következményei mutatkoznak. Egy olyan hiba megkeresése a következmények alapján, amelyen átsiklottunk, rendkívül nehéz és bonyolult feladat még a program írójának is, nem is szólva arról, amikor más programjában kell a hibát megtalálni.

Gondoljunk arra, hogy egy hibásan nyomtatott lap legalább 1 forintba kerül, egy óra gépidő 1000 forintokban mérhető, nem is beszélve saját idegeinkről, vagy — legrosszabb esetben — a téves pályára állt rakétáról.

Jegyezzük meg: most nem a program hibáiról van szó, hanem azokról a hibákról, amelyek azért lépnek fel, mert a program kapcsolatba kerül a külvilággal. Világos, hogy ezt a kapcsolatot ellenőrizni kell. A kapcsolat neha hardver eszközökkel jön létre, de ez a ritkább eset. Aki számítógépes adatgyűjtéssel, folyamattírányítással és hasonló dolgokkal foglalkoznak, nagyon jól tudják, hogy milyen fontos a bejövő adatok hitelességének, illetve hihetőségének ellenőrzése.

A közönséges programok többsége a kapcsolatot a külvilággal a bemenő adatok keresztül érzékeli. A bemenetet is két részre lehet osztani: az előre elkészített adatokra és a felhasználóval való közvetlen kapcsolatra. A bemenő adatokat ellenőrizni kell — bárhonnán is jönnek —, de természetesen nem reagálhatunk egyformán egy előre fájbla készített adatra és arra, amikor egy élő emberrel folytatunk dialógust, és így nyerünk adatokat. Vannak olyan rendszerek és programok, amelyek egyenlőségeket akarnak tenni a két kapcsolat közé, ezáltal valamilyen univerzális látszatát igyekeznek kelteni, de az eredmény igen siralmas.

AZ ELŐRE ELKÉSZÍTETT ADATOK ELLENŐRZÉSE

Az előre elkészített adatok ellenőrzésének legfontosabb eszköze a redundancia. Ez azt jelenti, hogy a programnak nemcsak a működéséhez szükséges adatokat adjuk meg, hanem további adatokat is, amelyek pusztán az előzők ellenőrzésére szolgálnak.

Mindig ideges leszek, ha olyan BASIC programot látok, amely több száz DATA-val megadott adatot tartalmaz, és ezt a program négy-öt különböző helyen, minden ellenőrzés nélkül olvassa be. Sajnos mind a magyar, mind a külföldi folyóiratok, sőt a könyvek is tele vannak ilyen öszszebarakácsolott programokkal. A professzionálisan elkészített programok a következőképpen csinálják ezt. Először is elhatárolják egymástól a program különböző részeihez tartozó adatokat, a beolvasás előtt ráállnak a programrészhöz tartozó első adatra, és a beolvasás végén ellenőrzik, hogy az utolsó adatot olvasták-e be. Ha az adatmennyiség tetemes, közbülső ellenőrző pontokat is beiktathatnak.

A beolvasandó adatok végét, illetve a közbülső ellenőrző pontokat úgy találjuk meg, hogy az adatok közé „mérőldköveket” helyezünk. Például ha a bemenő adatok csupa számból állnak, akkor mérőldköveket az egyes sorozatok végére egy-egy sztringet illesztünk (például „*** 1 *****”, „*** 2 ***” stb.).

Ha valahol kimaradt egy adat vagy egy vessző, a beolvasás tipushiba miatt megakad. Ha viszont kétszer irtunk be egy adatot, akkor az utolsó beolvasás nem az ellenőrző sztringet találja meg, és a program maga fog hibát jelezni. Mindkét esetben világos az utolsó ellenőrző sztring tartalmából, hogy meddig volt jó a bemenő adatok sora és hol kell keresni a hibát.

Ez természetesen igen primitív ellenőrzés, hiszen csak az adatok számát ellenőrzi; szokásos az a sorozat utolsó tagjául egy kontrollszámot adni, mert így meg lehet nézni, hogy az utolsó adat az előzők összeg-e. Ez sem érzékeny két karakter felszerelésére, de ha nagyon fontos a biztonság, valami bonyolultabb összefüggést is képezhetünk. Az ilyen megoldást viszont csak akkor szoktuk használni, amikor az adat lényegében a program szerves része és csak ritkán változik, mert az adatok ilyen bonyolult előkészítése nem kedvelt feladat. Tehát olyankor élünk az efféle megoldással, amikor a programhoz táblázatokat, szótárakat és egyéb ilyesmit olvasunk be, és bizonyosak akarunk lenni afelől, hogy az adatok nem sérültek meg, illetve, hogy jól olvassuk be őket.

A „mérőföldkövek” beiktatása a nagy homogen adatszámokra és emberi léptékek szem előtt tartása miatt is fontos: a kiugró, eltérő formájú adat azoknak is kapaszkodó, akik előkészítik ezt a homogén, jellegtelenné adattömeget.

Ha a bemenő adatoknak eleve van valamilyen struktúrájuk, természetesen nemigen van szükség ilyen mérőföldkövekre: viszonylag hamar kiderül, hogy nem odaiállít adat következik a sorban.

Az adatok sorrendjén és formáján kívül a program ellenőrizheti a bejövő adatok értékét is. Ezt néha fontosnak tartjuk, néha nem. Meglehetősen nagy munka ez, és a program futását is nagyon lassítja. Az Ada nyelv lehetőséget nyújt arra, hogy egy változó típusának alsó és felső határát írjunk elő: amikor ilyen változóba írunk bele, a program automatikusan ellenőrzi, hogy a szám az adott határok közé esik-e. Ezáltal a programozási munkát megspóroltuk, de a futás lelassulása továbbra is fennáll. Az ún. részhalmoz típus segítségével ilyen ellenőrzésre a Pascal és a Modula nyelvben is lehetőség van, de csak egészekenél vagy egész jellegű értékeknél.

AZ EMBER-GÉP KAPCSOLAT ELLENŐRZÉSE

Amikor az adatok nincsenek előkészítve, hanem a program úgy működik, hogy a gép az emberrel tart közvetlen kapcsolatot, akkor a bemenetet egészen másképp kell megszervezni.

Az emberek nem szeretik, ha akár csak egyvel is többször kell leütniük a billentyűt a szükségesnél — tehát a redundanciát kerülni kell. Ezzel szemben az ember viszont elvárja, hogy állandóan tájékozottasság arról, hol tart a munka, milyen bemenő adatot kívánnak tőle. Másrészt általában nem tud gépelni, tehát meg kell adni neki a lehetőséget arra, hogy közben javitson. Majd ellenőrizni kell, mit írt be, és ha hibás, ismét lehetővé kell tenni a javítást.

Viszonylag könnyen összeállítható egy olyan eljárás, ami a következőket csinálja: kiír egy paraméterenként megadott karakterorozatot (amely lehet üres is), engedti, hogy a végére írjunk valami folytatást és a kapott szövegbe beszurjunk, töröljünk, és hogy végül az eredményt elküldjük a számítógépbe. Ez az eljárás igen alkalmas lesz arra, hogy az egész ember-gép kapcsolatot szervezzük vele.

Elvileg egyszer az életben elég lenne egy ilyen eljárást megírni, és akkor mindig, mindenki azt használhatná. Sajnos jelenleg ez nem járható út, mert az apró részletek-ében minden mindent egy kicsit más konvenciókat követ, mint a másik. Az egyik gép az INSERT gomb megnyomására egy karaktert szűr be, a másik oda-vissza változtatja a beszűrő és felülíró üzemmódot.

KONKLÚZIÓK

A futás közbeni hibák két forrásból származnak: az egyik, hogy rosszul gondoltuk át, mit kell a programnak csinálnia, a másik, hogy a környezet nem úgy viselkedik, ahogy vártuk. A két vétek majdnem azonos következményű. Nem tarthatunk egy programot jónak, ha csak ideális, steril körülmények között működik.

Az élet bonyolódik az egészen egyszerű emberi hibáktól is, és néha még a tökéletes berendezések is elromlanak. Manapság a jó program egyik fő jellemzője a robusztusság. Egy ilyen programnak meg se kottyán, ha a bemenő adatok között üres sorok vagy felesleges szövegek vannak. Érthető hibabejelentéseket küld, és az emberi kapcsolaton keresztül lehetőséget nyújt a hiba kijavítására. Amikor a program egy hibán „elszáll”, nem rontja el saját adatfájlját és nem hagyja inkonzisztens állapotban a fájlkezelő rendszert.

Ezek a dolgok természetesen csak megfelelő eszközökkel, operációs szerekkel és programozási nyelvekkel érhetőek el, de a programozónak is hozzá kell tennie a maga részét: a gondos programtervezést és kivitelezést — egyre inkább ez a világszintvonal mércéje.

Más gombot jelöltek ki törlésre az egyik és a másik gépen, másik gombbal jelzik, hogy vége az üzenetnek, és így tovább. Még a magyar ékezetes betűk leírására is legalább tíz különböző megoldást ismerek. Van, ahol először a betűt kell leütni és azután az ékezetet, van ahol fordítva. Ezek után nem csoda, ha a felhasználó, amikor egy új, eredetileg más gépre írt programot először próbál ki, rögtön megállapítja, hogy ezt csak valami agyalágyult követhette el.

Ezért tehát, ha olyan programot írunk, amiről azt hisszük, hogy másnak is szükség lehet rá, a bevettelt végző részt különítsük el a többi résztől amennyire csak lehet, és ezáltal adjuk meg a felhasználónak a lehetőséget arra, hogy olyanra cserélje ki, amelyen az ő ízlésének megfelel.

A HIBAKEZELÉS KÉT MÓDJA

A futás közbeni hibákat két nézőpontból közelíthetjük meg. Az első felfogás szerint a futás közben — a bevettelt hibája miatt — létrejött hiba olyan halálos lövés, ami után a program egyetlen dolga, hogy elküldje a gyászjelentéseket. A másik nézet szerint a hiba támadás a program ellen, amit azonban kis szerencsével ki lehet védeni. A hiba kivédését a modern programozási nyelvek kivételkezelésnek (exception handling) nevezik.

A régebbi nyelvekben is volt lehetőség a kivédésre. A programot végrehajtó függvé-

nyeket úgy kellett megírni, hogy ne csak arra készüljenek fel, hogy a tevékenység jól végbemegy, hanem a hibákra is. Ezért a függvénynek nemcsak a szorosabban vett eredményt kell visszaadnia, hanem egy állapotjelzőt is, ami azt mutatja, hogy sikerült-e elvégezni a feladatot, és ha nem, mi volt az oka. Ezt az állapotjelzőt felhasználva értelmes üzeneteket és korrekciókra való lehetőséget. Ilyesmit leggyakrabban a periféria- és fájlkezelő függvényekben szoktak beépíteni (például kísérlet nem létező fájl megnyitására, a periféria off line-ban van stb.).

A modern nyelvek kivételkezelése abban különbözik ettől a hagyományos megoldástól, hogy úgy tekint a dolgot: egy függvény jó működése a normális állapot, a hibás működés pedig kivételes állapot. A program fő vonalát úgy kell megírni, hogy az csak a normális, jó működéssel foglalkozzon, amikor pedig a kivételes állapot bekövetkezik, a vezérlés automatikusan a program valamilyen speciális helyére, kivételkezelésre kerül.

Ahhoz, hogy a kivételes állapotot kezelni tudjuk, van egy sor kivétel, amelynek a programozási nyelv eleve nevet ad (túlsorolás, rosszul működő berendezés, nem létező fájl stb.), további eseteket a programozó vezet be, és az ezekhez tartozó kivételes állapot a program egy speciális utasításának kiadására következik be.

Ezek után egy függvény működését úgy közelíthetjük el, hogy a függvény vagy jól lefut és a jó eredményt visszaadva ott folytatja a munkát, ahol a függvényhívást kiadta, vagy hiba miatt kivételes állapotba kerül, és a vezérlés a hívó programnak arra a pontjára jut, amelyik a kivételes állapotokat kezeli. A kivételkezelő felsorolása megmondja, melyik kivételnelmi mit kell csinálni, és rendszerint arra is lehetőséget, hogy legutoljára bármilyen addig fel nem sorolt hibát elintézzünk.

A hiba kezelésén különböző dolgokat értenek. A MESA nyelvben arra is lehetőség van, hogy bizonyos módosítások után visszaugorjunk a függvényhívásra vagy a munkát a függvényhívás után folytassuk. Az Ada nyelvben viszont csak annyit tehetünk, hogy az adott programozási egységnek (rendszerint függvénynek vagy eljárásnak) a hiba előfordulása utáni részét egy másik utasítássorozattal helyettesítjük.

A szakemberek között vita folyik arról, hogy mi is tekintendő kivételnek és mit értünk kezelésén. Vannak, akik egészen tágra értelmezik a kivételkezelést: egy fájl beolvasására például irtak olyan függvényt, ahol annak törzse egyetlen végtelen beolvasó ciklus, és mivel természetesen valamilyen beolvasásnál előbb-utóbb „END—OF—FILE” hiba lép fel, ekkor a kivételkezelő bezárja a fájlt és visszatér a hívó programhoz. Mások szerint ez nem becsültes program, hiszen maga idéz elő olyan hibát, amire nincs is szükség.

FARKAS ERNŐ

A NAGYFELBONTÁSÚ KÉPERNYŐ FORGATÁSA BÁJTONKÉNT

COMMODORE 64

II.

Az előző részben a nagyfelbontású képernyő bájtönkénti forgatását ismertettük, melynek hatása azonban olyan volt, mintha csak bitenként forgattuk volna a képernyő memóriatartalmát, mivel a forgatás függőlegesen felfelé irányult, és a forgatott grafikus sorok 1 bit, azaz 1 pixel vastagságúak voltak. Most a képernyő vízszintes, X irányú, bájtönkénti forgatásáról lesz szó, amely már hatásában is mutatni fogja a grafikus kép bájtönkénti, azaz 8 pixellel balra való elmozdulását. Amit korábban mondtunk el a grafikus képernyő szerkezetéről, valamint a BASIC munkaterület 16 kb-ja átalításáról, azok itt is érvényesek.

Az 1. listán a forgató rutin gépi kódú programja látható, amit a SYS(52480)-nal lehet elindítani. A SYS(52604) utasítás segítségével bekapcsolhatjuk a \$2000-nél kezdődő grafikus képernyőt, a SYS(52621) utasítással pedig visszatérhetünk a karakter üzemmódozhoz.

A program sorainak magyarázata

170—180 A két nullás lapon lévő mutató segítségével forgatjuk a képernyőt. Az AKT az éppen aktuális bájt, a KOV pedig a jobb oldalról vele szomszédos (azaz a töle számozásban 8 bájtnyira lévő) bájtja mutat.

200—240 A ROM-ról RAM-ra kapcsolunk át, és eközben az IRQ megszakításokat is letiltjuk, így majd a ROM alatt elhelyezkedő Simon's BASIC képernyők forgatása is lehetségessé válik.

260—390 Kezdőértékek beállítása. Az időleges tároló, a TEMP elejének kezdőcíme: TEMP=KEP (= \$2000); a karaktársorok száma bekerül a SORSZM változóba SORSZM=24; KARBJT=7 az egy karakter-sort felépítő grafikus képpont-sorok számát határozza meg (0-7), míg legvégül az X változóba a képernyőn lévő oszlopok száma kerül (0-39).

410—490 A cikluson belüli értékadás: AKT=TEMP és KOV=TEMP+8.

510—530

A grafikus képsor legelső bájtját a veremtárolóba mentjük ki.

550—650

Itt jön létre a grafikus képsor balra forgatása egy bájtal. Vegyük észre, hogy ez a ciklus negyvenszer (39-től 0-ig) hajtodik végre, miközben az Y értékét 8-cal növeljük. Mivel a $40 \times 8 = 320$ már nem ábrázolható az Y indexregiszter egy bájtján, ezért 256-tal meg kellett növelni az AKT és a KOV mutató értéket.

670—680

A grafikus képsor veremtárolóba kimentett legelső bájtját itt írjuk vissza a veremtárolóból a képsor utolsó bájtjába.

710—750

A TEMP értékét nyolc alkalommal (0-7) megnöveljük 1 bájtal, létrehozva ezzel az új grafikus képsor kezdőértékét. Nyolc ilyen grafikus képsor alkot egy karaktérsort.

780—860

A képernyőn lévő karaktérsorok számának megfelelően 25 alkalommal növeljük meg a TEMP értékét 312-vel.

880—919

A RAM-ról visszkapcsolunk újra ROM-ra, és az IRQ megszakításokat is újra engedélyezzük.

930—990

Átkapcsolás a \$2000-nél (8192) lévő grafikus képernyőre.

1010—1070

Visszkapcsolás a normál kijelzésre.

A 2. listán látható a nagyfelbontású képernyőt forgató rutin BASIC betöltő programja. Ha ennek a programnak az 1050-es sorát átírjuk a következő módon:

1050 W=49152:FOR I=W TO W+157
akkor ezzel a gépi kódú program a SC000 tárterületre tölthető be. Ilyenkor a nagyfelbontású grafika forgatását a SYS(W) utasítással indíthatjuk el. A grafikát a SYS(W+124)-gyel lehet be-, illetve a SYS(W+141) utasítással pedig kikapcsolni. A W-nek más kezdőértéket adva lehet a forgató programot más tárterületre betölteni a BASIC betöltővel. Ha a kizettás magánót nem használjuk, akkor a gépi kódú

program számára jó elhelyezést nyújt a kizettáppuffer is: W=828.

Ha a Simon's BASIC által létrehozott ábrákat kívánjuk forgatni, akkor az 1. lista 1090-es sorában lévő \$20-át kell módosítanunk \$E0-ra, mivel a Simon's BASIC \$E000-tól kezdődően helyezi el a grafikus ábrákat. A 2. listán látható BASIC betöltő programnak az 1020-as sorában található adatát, a 32-t kell módosítanunk 224-re.

Ha nem kívánjuk a teljes képernyőt forgatni, hanem annak csak egy részletét, akkor az 1. lista 1090-es, 1100-as és 1110-es sorában kell végrehajtani módosításokat (illetve a 2. lista 1020-as sorában kell a megfelelő adatokat megváltoztatnunk).

Például az 1. listában módosítva az alábbi sorokat:

1090 KEP .BYTE \$20, \$28
1100 OSZLOP.BYTE 18
1110 SOR .BYTE 10

már csak a képernyő egy kisebb részlete fog rogamajd.

A 2. listában hasonlóan módosítható az 1020-as sor:

1020 DATA 32,40,18,10,24,7,7,0,0,0

ilyenkor természetesen az 1030-as sorban található ellenőrző összeget is módosítani kell 133-ról 138-ra. Közvetlen üzemmódból is végrehajthatjuk ezeket a módosításokat. Így például a

POKE 679,POKE 680,224

utasításokkal a forgatandó képernyő-területet \$E000-tól kezdődően jelölhetjük ki;

a POKE 681, OSZLOP a forgatásban résztvevő karakteroszlopok számát határozza meg; ilyenkor (OSZLOP+1) számú karakteroszlopnak megfelelő memóriaterület vesz részt a forgatásban. A

POKE 682, KARSOR

a forgatásban résztvevő KARSOR+1 számú karaktérsor számát határozza meg. Az OSZLOP, illetve a KARSOR változókra a következő egyenlőtlenségeknek kell teljesülniük:

$0 < OSZLOP <= 39$

$0 <= KARSOR <= 24$

Felhívom a figyelmet arra, hogy ezekkel a módosításokkal óvatosan kell bánnunk, ugyanis tekintettel kell lennünk arra is,

```

: EZ A PROGRAM A NAGYFELBONTASU
: KEPERNYŐT FORGATJA BAJTONKENT
: GRAFIKUS KEPERNYŐ HELYE - #2000
: GRAFIKA BE - SYS(52604)
: GRAFIKA KI - SYS(52621)
: FORGATAS - SYS(52400)
:

```

```

150: CD00          == #CD00
170: CD00          AKT  = #FB
180: CD00          KOV  = #FD
:
200: CD00 A5 01    KEZD   LDA #1
210: CD02 8D 00 02 STA TEMP+2
220: CD05 A9 34    LDA #34
230: CD07 78      SEI          ; IRD TILDS
240: CD08 B5 01    STA #1          ; RAM
260: CD0A AD A7 02 LDA KEP
270: CD0D 8D AE 02 STA TEMP
280: CD10 AD AB 02 LDA KEP+1
290: CD13 8D AF 02 STA TEMP+1
310: CD16 AD AA 02 LDA SOR
320: CD19 8D AB 02 STA SORSZM
330: CD1C AD AC 02 KEPFORG LDA KARAK
340: CD1F 8D AD 02 STA KARBJT
390: CD22 AE A9 02 KARSOR LDX OSZLOP
410: CD25 1B      CLC
420: CD26 AD AE 02 LDA TEMP
430: CD29 85 FB   STA AKT
440: CD2B 69 0B   ADC #B
450: CD2D 85 FD   STA KOV
460: CD2F AD AF 02 LDA TEMP+1
470: CD32 85 FC   STA AKT+1
480: CD34 69 00   ADC #0
490: CD36 85 FE   STA KOV+1
510: CD38 A8 00   LDY #0
520: CD3A B1 FB   LDA (AKT),Y
530: CD3C 4B     PHA
550: CD3D B1 FD   GRAFSOR LDA (KOV),Y
560: CD3F 91 FB   GRAFSOR STA (AKT),Y
570: CD41 18     CLC
580: CD42 98     TYA
590: CD43 69 0B   ADC #B
600: CD45 AB     TAY
610: CD46 90 04   BCC TOVABB
620: CD48 E6 FC   INC AKT+1
630: CD4A E6 FE   INC KOV+1
640: CD4C CA     TOVABB DEX
650: CD4D D0 EE   BNE GRAFSOR
670: CD4F 68     PLA
680: CD50 91 FB   STA (AKT),Y
710: CD52 EE AE 02 STA TEMP
720: CD55 D0 03   BNE TOV2
730: CD57 EE AF 02 INC TEMP+1
740: CD5A CE AD 02 TOV2 DEC KARBJT
750: CD5D 10 C3 02 BPL KARSOR
760: CD5F 18     CLC
790: CD60 AD AE 02 LDA TEMP
800: CD63 69 38   ADC #56
810: CD65 8D AE 02 STA TEMP
830: CD68 AD AF 02 LDA TEMP+1
835: CD6B 69 01   ADC #1
837: CD6D 8D AF 02 STA TEMP+1
850: CD70 CE AB 02 DEC SORSZM
860: CD73 10 A7   BPL KEPFORG
880: CD75 AD B0 02 LDA TEMP+2
890: CD78 85 01   STA #1
900: CD7A 58     CLI
910: CD7B 60     RTS
:
930: CD7C AD 11 D0 GRAFBE LDA #D011
940: CD7F 89 20   DRA #E20
950: CD81 8D 11 D0 STA #D011
960: CD84 AD 18 D0 LDA #D018
970: CD87 89 0B   DRA #0B
980: CD89 8D 18 D0 STA #D018
990: CD8C 60     RTS
:
1010: CD8D AD 11 D0 GRAFKI LDA #D011
1020: CD90 29 DF   AND #5DF
1030: CD92 8D 11 D0 STA #D011
1040: CD95 AD 18 D0 LDA #D018
1050: CD98 29 F7   AND #5F7
1060: CD9A 8D 18 D0 STA #D018
1070: CD9D 60     RTS
:
1085: 02A7          == #02A7
1090: 02A7 00 20   KEP .BYTE#00,#20
1100: 02A9 27     OSZLOP .BYTE39
1110: 02AA 18     SOR .BYTE24
1120: 02AB 18     SORSZM .BYTE24
1140: 02AC 07     KARAK .BYTE7
1145: 02AD 07     KARBJT .BYTE7
1150: 02AE 00 00 00 TEMP .BYTE0,0,0

```

1. lista

```

1000 FOR I=679 TO 688
1010 READ X : POKEI,X : S=S+X: NEXT
1020 DATA 0,32,39,24,24,7,7,0,0,0
1030 IF<>133THENPRINT"HIBAS ADAT !":END
1040 S=0
1050 W=52480: FOR I=W TO W+157
1060 READ X : POKEI,X : S=S+X: NEXT
1070 DATA 165,1,141,176,2,169,52,120
1080 DATA 133,1,173,167,2,141,174,2
1090 DATA 173,168,2,141,175,2,173,170
1100 DATA 2,141,171,2,173,172,2,141
1110 DATA 175,2,174,169,2,24,173,174
1120 DATA 2,133,251,105,8,133,253,173
1130 DATA 175,2,133,252,105,0,133,254
1140 DATA 160,0,177,251,72,177,253,145
1150 DATA 251,24,152,105,8,168,144,4
1160 DATA 230,252,230,254,202,208,238,104
1170 DATA 145,251,238,174,2,208,3,238
1180 DATA 175,2,206,173,2,16,195,24
1190 DATA 173,174,2,105,56,141,174,2
1200 DATA 173,175,2,105,1,141,175,2
1210 DATA 206,171,2,16,167,173,176,2
1220 DATA 133,1,88,96,173,2,208,9
1230 DATA 32,141,17,208,173,24,208,9
1240 DATA 8,141,24,208,96,173,17,208
1250 DATA 41,223,141,17,208,173,24,208
1260 DATA 41,247,141,24,208,96
1270 IF<>19009THENPRINT"ADAT HIBA!":END
1280 PRINT"REND BEN!":END

```

2. lista

```

1 REM *** SPRITE ADATOK BEOLVASASA.
3 FORI=0T062:READA:POKE704+I,A:NEXT
5 V=53248:POKE2040,11:POKEV+21,1
7 POKEV,255:POKEV+1,00
9 POKEV+23,1:POKEV+29,1
11 :
13 SYS52604:REM GRAFIKA BE
15 REM A PONT FEHER, A HATTER FEKETE.
17 FORI=1024T02023:POKEI,16:NEXT
19 FORI=8192T016191:POKEI,..:NEXT
21 BASE=8192:GOSUB45
23 BASE=8392:GOSUB45
25 BASE=11392:GOSUB45
27 BASE=11592:GOSUB45
29 BASE=14592:GOSUB45
31 BASE=14792:GOSUB45
33 REM *** GRAFIKA FORGATASA ***
35 FORI=0T079:SYS(52480):NEXT
37 GOSUB63:REM GRAFIKA KI
39 END
41 :
43 REM SPRITE ATMASOLASA A GRAFIKARA
45 RESTORE
47 FORY=0T020
49 FORI=0T02
51 READ AX:X=I*8
55 BYT=BASE+320*INT(Y/8)+(YAND7)*X
57 POKE BYT,AX
59 NEXT:NEXT:RETURN
61 :
63 GETA#:IFA#=""THEN63
65 POKEV+21,0:PRINT"":REMSPRITE KI
67 SYS52621:REM GRAFIKA KI
69 RETURN
71 :
73 REM * A SZELLEMGRAFIKA ADATAI *
75 DATA 0,0,0,0,15,128,0,48,112,0,64,8
77 DATA0,128,4,8,128,4,1,4,18,127,36,18
79 DATA192,4,17,128,4,17,128,4,16,128
81 DATA4,16,128,4,16,128,196,16,71,130
83 DATA32,63,193,193,24,96,6,0,48,24,0
85 DATA16,96,0,25,128,0,14,0

```

3. lista

hogy a forgatás ne terjedjen ki olyan memóriaterületekre is, melyek már nem tartoznak a nagyfelbontású képhez. Ez az eset áll elő akkor, ha csak a KEP-hez tartozó bájtok értékét növeljük meg \$00. \$20-ról például \$00, \$30-ra. Ebben az esetben hiába toltuk fel korábban a BASIC munkatérület kezdetét 16 kb-jára; mert a forgatás hatása ki fog terjedni a \$4000 feletti memóriaterületre is (!?), tönkretéve így módon az ott elhelyezett BASIC programunkat is!

Környezetvédő „ZÖLD” rutinok, avagy a célzott szemétyűjtés

A Mikroszámítógép Magazin 1988/1. számában megjelent a CHIP újságnak Tóth Ferenc fordításában és kiegészítésében a Szemétyűjtés című cikke. Az abban ismertetett eljárás helyett egy egyszerűbb, mégis hatékonyabb módszert szeretnék ajánlani a programozóknak.

Az említett cikk a sztringmemória-terület gyors nagytakarításáról egy gépi kódú szubrutinnal gondoskodik, amely „sokkal gyorsabb, mint a C64 rendszer-rutinja”.

A szubrutin természetesen működik, és ellátja feladatát. Van viszont néhány kellemetlen vonzata:

- a sztringértékdadásokat tartalmazó programokban minden fűzértékdadás előtt meg kell hívni a rutint,
- maximum 2048 sztringet tud kezelni,

— ha a rutin szemetet talál a sztringterületen, akkor az összes ide mutató sztringtömb és sztringváltozó címét át helyezi az operatív memória \$F000—\$FFFF területére, amit mi esetleg más célra szeretnénk használni,

— adott esetben a szemét nagytakarításának néhány másodperces ideje is túrhetetlenül hosszú,

— maga a rutin is helyet foglal a memóriában, ami BASIC-bővítő programok alkalmazásánál szintén problémákat okoz.

E hátrányokat kiküszöböli a következő, lényegesen egyszerűbb és csak néhány BASIC-utasításból álló szemétyűjtő módszer, amelynek lényege, hogy csak ott gyűjt szemetet, ahol van!

Igen ám, de ehhez tudniuk kell a szemét keletkezéséről. Márpedig szemét akkor keletkezik, ha egy sztring új értéket kap, mert a régi feleslegessé válik. A C64 Garbage Collection rendszerrutinja aktiválódásokor a teljes sztringterületet (a BASIC-terület végétől — S37/S38 — lefele, a S33/S34 által mutatott érték) átfésül szemét után kutatva, akkor is, ha például a sztringterület 90 százaléka egy vagy több változatlan sztringtömb tölti ki a sztringterület elején, amely természetesen szeméttmentes.

E felesleges kutatás a legnagyobb időrabló. Mi hát a megoldás?

— A szemét keletkezése előtt jegyezzük fel a sztringterület végét jelző mutató értékét:

SA = PEEK(51) : SF = PEEK(52)
 — A szemétyűjtás végén, például formázott kiíratást — PRINTUNG-ot — megvalósító alprogram után állítsuk be a szeméttmentes terület végére a BASIC-terület végét:

POKE 55,SA : POKE 56,SF és hívjuk meg a Garbage Collection rendszerrutint: SYS 46374. A rendszerrutin ekként csak azon a viszonylag szűk területen dolgozik, ahol a szemét koncentráldott.

— Állítsuk vissza a BASIC-terület végét jelző mutatót az eredeti értékére: POKE 55,BA : POKE 56,BF. A BA és BF érték a program elején jegyeztük fel:

BA = PEEK(55) : BF = PEEK(56).

A hatást mutatja be az 1. és 2. listán látható két demoprogram. A 2. lista tartalmazza a ZÖLD rutinokat, az 1. nem. Mindkét programban létrehozunk egy-egy nagy sztringtömböt (10-es sor) és 1 darab „szemét” sztringet. A 20-as sorban hívjuk meg a Garbage Collection rendszerrutint. A futási idők jól demonstrálják a módszer előnyét. A 2. ZÖLD rutint természetesen nem fontos közvetlenül a szemét keletkezése után mindig meghívni, megvárhatjuk, míg összegyűlik annyi, amennyinek gyűjtési ideje meg elviselhető számunkra.

Érdemes figyelembe venni azt a gondolatot is, hogy ha nincs szemét, nem kell gyűjteni! Szemét csak sztringből lesz, célszerű ezért a programokban, ahol csak lehet, sztring helyett numerikus adatokkal dolgozni.

Ha ezt tesszük, és a ZÖLD rutinokkal celszerűen védjük programunk környezetét, az sohasem meredek le, mert nem kell „nagytakarítást” csinálni, azaz elérjük célunkat: programunk még másodpercekig tartó szünetek sem zavarják, legfeljebb tizedmásodpercesek, amelyeket szinte észre sem lehet venni.

KÁNTOR JÓZSEF

1. lista

```
T=345 43333s
0 REM----- DEMO 1
5 DIMARR(2000)
10 FOR I=1 TO 2048:ARR(I)=CHR$(65):NEXT I
15 BB=STR$(1):BB=STR$(2)
20 TT=1:GOSUB50:PRINT"TT=";TT;"-TT=";60;"SEC"
30 REM
40 REM
50 SYS46374:RETURN:REM---GARBAGE COLLECTION---
```

2. lista

```
T=0.35s
0 REM----- DEMO 2
2 BA=PEEK(55):BF=PEEK(56):REM=BASICCHARA
5 DIMARR(2000)
10 FOR I=1 TO 2048:ARR(I)=CHR$(65):NEXT I
11 GOSUB40:REM---ZÖLD SZEMÉTYŰJTÉS---
15 BB=STR$(1):BB=STR$(2)
20 TT=1:GOSUB50:PRINT"TT=";TT;"-TT=";60;"SEC"
30 REM
32 REM
40 SA=PEEK(51):SF=PEEK(52):RETURN:REM---ZÖLD---
42 REM
44 REM
50 POKE55,SA:POKE56,SF:SYS46374
55 POKE55,BF:POKE56,BF:RETURN:REM---ZÖLD---
```

Lévéni, hogy a felforgatott memóriaterület ilyenkor \$3000 — \$4F3F-ig, decimálisan: 12288-tól 20287-ig (12 kb-ajától majdnem 20 kb-ajig) terjed. Ha a SYS(W+124) utasítással kapcsoljuk be a grafikus képernyőt, akkor ilyenkor mi nem is látjuk a teljes memóriaterület forgását, mert a SYS(W+124) úgy kapcsolja be a nagyfelbontású képernyőt, hogy az \$2000-tól \$3F3F-ig (8192-től 16191-ig) terjed. Emiatt a nagyfelbontású képernyő felső fele: \$2000 — \$2FFF-ig változatlanul marad; \$3000-tól a képernyő végeig: \$3F3F-ig látjuk a forgást, de az e terület fölötti \$3F40-tól \$4F3F-ig terjedő memóriaterület forgása előttünk is rejte marad. Illetve csak közvetve tapasztalhatjuk a forgatás eredményét úgy, hogy a 16 kb-ajig fölött (\$4000-tól) elhelyezkedő BASIC programunkat a forgató rutin jól kiforgatja eredeti formájából!

A grafikus kép a képernyőről „kiúsztatható”, ha az 1. lista 520-as sorában található LDA (KOV), Y-to megváltoztatjuk LDA # \$00-ra. Ez a

POKE W+58,169 : POKE W+59,0 utasítások segítségével közvetlen üzemmódból is megvalósítható. Ezután meghívja a forgató rutint, az törli a grafikus kép utolsó karakteroszlopának memóriatartalmát. (Próbáljuk ki bekapcsolt grafikai képernyővel a

FOR I=0 TO 39 : SYS(W) : NEXT utasítást!)

Ha elhagyjuk az 1. lista 510—530-as, valamint a 670—680-as sorait, vagy az itteni utasításokat NOP-pal (kódja: \$EA=234) helyettesítjük, akkor akár egy grafikai kép-újságot is megvalósíthatunk. Ilyenkor ugyanis a grafikus képernyő legutolsó karakteroszlopának memóriatartalmát nem írja felül a képernyő legelső karakteroszlopának memóriatartalma, azaz ilyenkor nem jön létre a grafikus kép körbeforgatása, hanem csak a karakteroszlopok memóriatartalma egy oszlopban előlrebb másolódik át. Ilyen módon tehát, ha a legutolsó karakteroszlopba, minden forgatás előtt újabb képi információkat írunk be, akkor egy grafikus kép-újságot valósíthatunk meg.

Végezetül pedig a 3. listán egy BASIC példaprogram található, illusztrálandó a nagyfelbontású képernyő forgatását. (Figyelem! A gépi kódú programot ez előtt a program előtt a C64 memóriájába kell tölteni!) Ez a BASIC program először egy sprite-ot jelenít meg kétszeres nagyságban (1-9-es sor), majd ezt a szellemgrafikát bemásolja a nagy felbontású képernyő különböző helyeire (21-31-es sor). Az így keletkezett grafikus képet 80 alkalommal balfelé lépteti egy karakternyivel, azaz a képet kétszer teljesen körbeforgatja! Figyeljük meg, hogy a kinagyított sprite eközben a helyén marad. Ez a kinagyított sprite azért nem vesz részt a nagyfelbontású képernyő forgatásában, mert nem tartozik hozzá a nagyfelbontású képernyő memóriatartalmához.

A sprite-adatokat természetesen további sprite-adatokkal bővíthetjük ki, és így azok is a nagyfelbontású képbe másolhatók át, például úgy, hogy az átmásoló alprogram nem GOSUB45-tel, hanem GOSUB47-tel hívjuk meg.

A nagyfelbontású képernyő bajtonkénti balra forgatásának ismertetésével be is fejeztük a forgatásokról szóló sorozatunkat. SZABÓ PÉTER PÁL

MONITOROLÓGIA

Mit kell tudnunk a monitorokról?

A különböző fogalmak jobb megértéséért először röviden a fekete-fehér tévékép előállításának elvét tekintjük át. A képet a fluoreszkáló képernyőn egy elektronsugár állítja elő. A kép pontokból áll. Az egyes pontok fényességét az elektronsugár erőssége szabja meg. Az elektronsugár elterítését két egymásra merőleges, változó intenzitású mágneses tér végzi az 1. ábrán látható módon. A színes kép előállítása ettől annyiban tér el, hogy a vörös, zöld, kék alapszíneknek megfelelő három elektronsugár intenzitásainak aránya határozza meg egy pont színét. Ennek megfelelően egyidejűleg három elektronsugár elterítéséről kell gondoskodni...

... DE NEM TERRORRAL

A sugár a képernyő bal felső sarkából indul és jobbra mozog, miközben egy kicsit lefelé is elmozdul, és így leír egy sort. Ezután a sugár visszaugrik az előző kiindulási pont alá, és újabb sort ír le. Ez mindaddig ismétlődik, amíg a sugár a képernyő jobb alsó sarkát el nem éri. Ezután a sugár visszaugrik a bal felső sarokba, és a folyamat az előbb leírt módon ismétlődik. Az elektronsugár visszaugrása alatt az elektronsugarat kikapcsolják, hogy ilyenkor ne lehessen látni. Azt, hogy a képernyőn egyszerre a teljes képet látjuk, a képernyő utánvilágításának és a szem tehetetlenségének köszönhetjük.

Akik gyakran ülnek számítógépük előtt programozás vagy játék céljából, jól tudják, hogy a megjelenítésre használt fekete-fehér vagy színes televízió mennyire fárasztja a szemet. Az igényesebb felhasználók ezért jobb felbontású és élesebb képet szeretnének. A problémát monochrom vagy színes monitor vásárlásával igyekeznek megoldani. Ha azonban nem körültekintően vásárolnak, még több problémával találkoznak. Az alábbiakban a monitor kiválasztásában, a különböző paraméterek és jelek, valamint a csatlakozók sokaságában kívánunk eligazítást nyújtani.

Utánvilágításnak azt a jelenséget nevezzük, hogy az elektronsugár által gerjesztett pont rövid ideig azután is világít, miután az elektronsugár azt elhagyta. A nálunk érvényes televíziós szabványban a sorok száma 625, a sorfrekvencia (a másodpercenként lerajzolt sorok száma) 15 625 Hz. Ezekből 25 Hz-es képfrekvencia (a másodpercenként előállított képek száma) adódik, amely a szem számára a folyamatos változás benyomásának keltésére már elegendő (lásd például a mozgófilmet). Ezzel szemben a televízió technikájában a villódzás elkerülésére trükköt alkalmaznak. A teljes képet két fél képre osztják, és az egyes fél képeket kétszeres sortávolságban egymás után rajzolják fel úgy, hogy a két fél kép egymásba ágyazódik. Ezt nevezik váltott soros letapogatásnak. Mindegyik fél kép 312,5 sorból áll, és a félkép-váltási frekvencia 50 Hz.

Az elektronsugár mozgását végző mágneses tereket a vízszintes és függőleges eltérítőkercsek állítják elő, amelyeket egy 15 625 Hz-es és egy 50 Hz-es szabadon futó oszcillátor vezérel. E két oszcillátor frekvenciája együttesen határozza meg az elektronsugár helyzetét. Ahhoz, hogy egy bizonyos képinformációt egyértelműen a képernyő egy megadott pontjához lehessen rendelni, a két oszcillátort

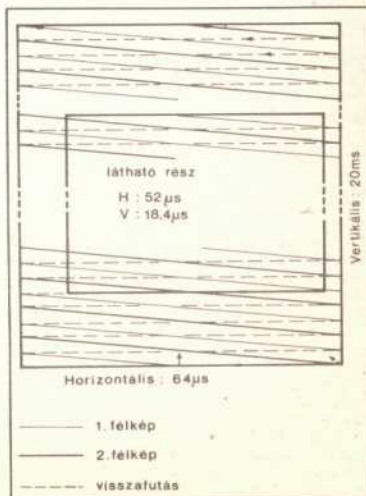
szinkronizálni kell. Ezt akkor hozzuk létre, amikor az elektronsugár a sor végéről a következő sor elejére, vagy a kép jobb alsó sarkából a bal felső sarkába ugrik vissza, mivel ezalatt az elektronsugarat „feketére kapcsoljuk”, vagyis kioltjuk, ezáltal a szinkronizáló jelek nem zavarják a képet.

A képinformáció, a kioltó és a szinkronizáló jelek együttesen alkotják az összetett videojelet, idegen rövidítéssel BASE-jelet, amely a kép egyértelmű előállításához szükséges. Az összetett videojelet felépítését és a jelszinteket a 2. ábra mutatja.

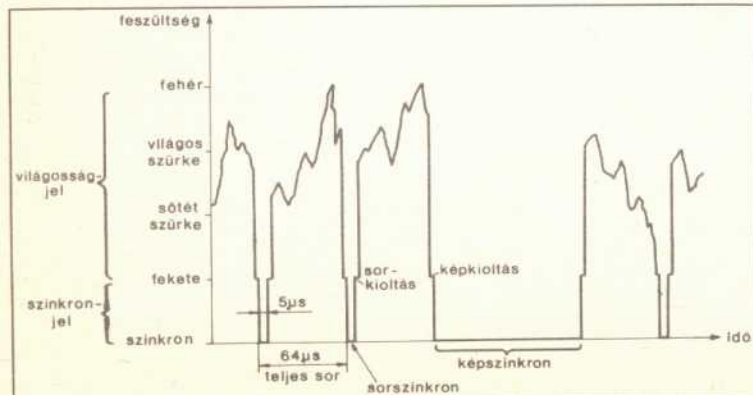
TEMESSÜK A VILLÓDZÁST!

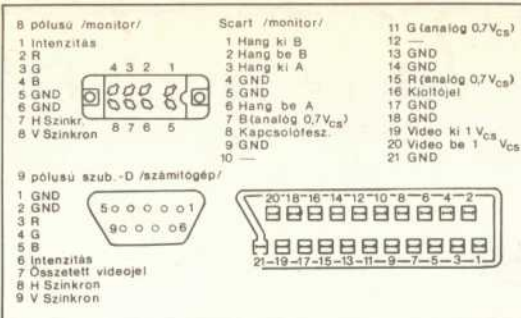
Egy sor lerajzolásának ideje $H=64 \mu\text{s}$, egy fél kép előállításáé pedig $V=20 \text{ ms}$. Monitoroknál nem törekednek arra, hogy a teljes képet lássuk, ezért csak az 1. ábrán bekeretezett kisebb részük látható. Alapvető követelmény viszont, hogy a kontraszt jó legyen, a grafikus megjelenítéskor a legfi-

1. ábra



2. ábra





3. ábra

nomabb vonal se veszítsen a világosságából. Tételizzük fel, hogy vízszintesen 640 pontot jelentünk meg (80 karakter × 8 pont), akkor az időbeli különbség két pont között 78 ns-ra adódik, ha a látható ablak szélességét az 1. ábrán megadott 52 μs-ra vesszük fel. Ez az érték pedig éppen 13 MHz sávzélességet határoz meg, amit már a jobb televíziók videobemenetei sem tudnak teljesíteni. Ha figyelembe vesszük, hogy ez a sávzélesség a nagy felbontású grafikaéhoz igen kevés, továbbá némi tartatléokra is szükség van, valamint a látható ablak gyakran kisebb is, akkor a 18 MHz-es sávzélesség is valójában csak rossz kényezermegoldás. Mivel kizárólag statikus képet ábrázolunk, a váltott soros letapogatásnál a szokásos eltérítési frekvenciáknál a zavaró villódzás keletkezik. Ez háromféle módon küszöbölhető ki:

— A képernyő utánvilágítási idejének növelésével. Az utánvilágítási idő szokásos értéke: 12...22 ms és tetszőlegesen nem növelhető, mert scrollnál zavaró lehet. Bizonyos esetekben 54 ms-ot is alkalmaznak. Ez az érték azonban már az elviselhetőség határán van.

— Egyszerűen elhagyják a váltott soros letapogatást, és mindig csak az egyik fél képet jelenítik meg. A sor- és képfrekvencia azonos marad, a kép nyugodtabb lesz, és kiegyenlítő impulzusokra nincs szükség. Némi manipulációval a 312 sorból 300 láthatóvá tehető. Ha több sorra van szükség, akkor megnövelik a sorfrekvenciát, aminek következtében a szükséges video-sávzélesség azonos módon megnövekszik. Néhány

számítógép lehetővé teszi a váltott soros letapogatás be- és kikapcsolását (interlaced/non-interlaced mode).

— Megnövelik a képfrekvenciát. Ennek következtében a függőleges felbontás csökken, ami a sorfrekvencia növelésével egyensúlyozható, de ehhez az imént említett nagyobb video-sávzélesség kell.

Túlnyomórést szöveges megjelenítésnél a második módszerben leirtakat célszerű alkalmazni, grafikus ábrázolásnál pedig a másik két módszert. Természetesen semmi akadály sincs több módszer egyidejű alkalmazásának.

A fentiek alapján megadhatók a „szokásos” BAS-kimenetű számítógépekhez csatlakoztatható monitorok főbb paraméterei:

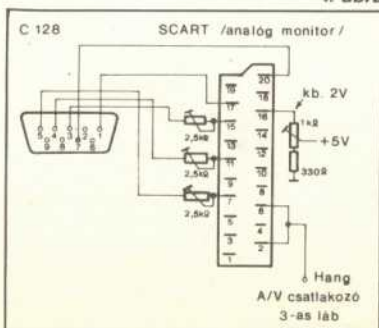
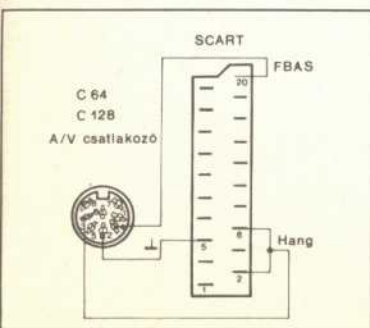
- sávzélesség: nagyobb, mint 18 MHz,
- sorfrekvencia: 15,5...18,5 kHz,
- képfrekvencia: 50...60 Hz,
- utánvilágítási idő: 12...22 ms.

Célszerű olyan monitort választani, amelynél a kép- és sorfrekvencia kívülről forgatógombbal állítható (a monitorok többsége ilyen). Vásárlás előtt mindenképpen ajánlatos a számítógép kimenetét alaposan tanulmányozni, mert az egyes gépek kimenetei a fenti értékektől jelentősen el térhetnek. Jó példa erre az Atari ST; ennek a sorfrekvenciája 36 kHz és a képfrekvenciája 71 Hz.

SZIGORÚ JELLELTÁR

Eddig minden világos lenne, ha a BAS-csatlakozáson kívül nem létezne a TTL-csatlakozások széles skálája. Utóbbiak

4. ábra



nak közös tulajdonsága, hogy a video- és szinkronjeleket külön vezetékeken továbbítják, és legalább ez utóbbiak TTL-szintűek. Ezért meg kell állapítani, hogy a számítógéptől jövő szinkronjelek külön-külön vagy keverve jönnek-e, illetve melyik az aktív szintjük. Gyakran ugyanannak a számítógépnek a különböző videokártyái is más-más jeleket szolgáltatnak.

A szinkronjelekkel kapcsolatban itt két „apróságra” érdemes ügyelni. Van, amikor a horizontális szinkronjelet a vertikális szinkronjellel egyszerűen VAGY kapcsolhatják össze. Ezt a monitor még „megérti”, de a vertikális szinkron alatt a horizontális szinkron kiesik, és a kép kezdetén csak lassan áll vissza. Ebben az esetben az első szövegsorban vagy az első grafikus sorokban „elkenődéseket” láthatunk. A másik változatban a horizontális és vertikális szinkronjeleket KIZÁRÓ VAGY kapcsolatba hozzák, ami a szabványjelhez közelebb áll. Ilyenkor a felső képsáv vilódzhat. Ha a két kisebb galiba valamelyike előfordul, ez nem a mi készülékünk hibája, ezért csupán a kényelmes fejlesztőket kell szidni.

Akinek még ez sem elég, az vásároljon színes monitort, amelynél az egyszerű monitorok „szabvány összevisszaságaihoz” még újabb társulnak. Ezek közül csak a két legfontosabbat említjük meg: a TTL és az analóg RGB rendszer által okozott problémákat. A különbség első ránézésre a három videojel (RGB) szintjeiben van:

- analóg RGB: 0,7...1 V
- TTL RGB: 3...3,5 V

Lényegesebb különbséget jelent, hogy a TTL RGB jel bináris, tehát a fehéret és feketét is beleértve összesen nyolc szín ábrázolható azonos intenzitással. Ezen úgy segítenek, hogy még egy intenzitás-bit fűznek hozzá (16 szín). Ezzel szemben az analóg RGB a vörösből, a zöldből és a kékéből keverhető összes szint tetszőleges intenzitással elő tudja állítani.

A számítógép és a monitor közötti szintproblémák elötét-ellenállásokkal könnyűszerrel áthidalhatók.

A fentieknél is változatosabb képet nyújtanak azonban a számítógépek és a monitor csatlakozói. Ha netán a vásárlásnál az előbbi dolgok alapos tanulmányozása miatt a megfelelő csatlakozókábel vásárlásáról megfeledkeznek, a 3. ábrán megadom néhány gyakori csatlakozó bekötési rajzát. A 4. ábrán két összekapcsolási példát is láthatunk Commodore 64 és 128 típusú számítógépekre. Természetesen ezen a téren teljességre nem törekedhetem a változatok nagy száma miatt, de úgy érzem, hogy a bemutatott példák támpontul szolgálhatnak az egyes egyedi esetek megoldásához. A szakirodalomban egyébként nagyon sok kézenfekvő megoldás található. Az összekapcsoláshoz azonban mindig kérjük ki szakember tanácsát, ha nem akarunk kép helyett „salátát” látni.

Boldogok, akik futják a köroket,
mert fordulatos
az életük

MÓKUSKERÉK

Azt a jelenséget, amikor egy új szubrutint ír valaki, pedig egy azonos célú rutin már ott lapul a programkönyvtárban, a „kerék feltalálásának” nevezik. Az ilyenkor alkalmazott megoldások erős nyelvfüggőséget mutatnak, amit néhány tipikus példával illusztrálunk:

— Az assembly programozók milliöi számára készítenek kereket, de nincs közöttük kettő, amelyik közös tengelyre illene.

— Az Ada-programozók szögletes kereket terveznek, majd pedig olyan utakat specifikálnak, melyeknek gödreibe azok beleillenek.

— Az ALGOL-programozók tartózkodnak a kerék készítésétől, mert azok esetleg go to valahova.

— Az APL-programozók a kereket a levegőben, egy zuhanó repülőgép futóművén lógva építik, és földet érés előtt általában már majdnem készen is vannak.

— A BASIC-programozók két kereket sem találják a tengelyüket.

— A C-programozók kerék helyett propellert készítenek. Ha kereket is tud-

nának, legalább „B minusz” osztályzatuk lenne.

— A COBOL-programozók EZERKEREKŰ-SZÁLLÍTÓ-MODUL-t dolgoznak ki, és megtiltják a gyaloglást.

— A FORTRAN-programozók az örület határára jutnak abbeli igyekeztükben, hogy I-vel kezdődő kerekeket állítsanak elő.

— A FORTH-programozók számlálatlanul gyártják a kerekeket, majd elfelejtik, melyik veremben tárolták őket.

— A LISP-programozók kidolgozzák a kereket (a kerékben (a kerékben (a programozó kerékében)))(újabbban már autójuk is van).

— A LOGO-programozóknak nem kell kerék, nekik teknőcük van.

— A Pascal-programozók az erény szintjére emelik a gyaloglást.

— A PL/1-programozók kijelölnek egy teamet, amelyik megépíti azt a gyárat, ahol majd a mérthibás kerekeket előállítják.

— A PROLOG-programozók által készített dolgok körbe-körbe járnak ugyan, de nem jutnak sehová.

— Az RPG-programozók soha nem készítenek kereket. Egy vezérlésre alkalmas kormánykeréktől eltekintve már minden kerekük megvan.

— A Smalltalk-programozók a kerekeket beszélik rá önmaguk felépítésére.

— A SNOBOL-programozók sztringeken működő kerekeket hoznak létre. Ezeket köznyelven jojonak hívják. (Zeke Hoskin)

(Megjelent az APL Quote-Quad 1988. júniusi számában)



C & ZX

SOFTWARE BÖRZE

folyamatosan bővülő programválasztékkal,
kazettán és lemezen egyaránt.

COMMODORE 64, SINCLAIR ZX SPECTRUM
játékleírások, térképek, ZX POKE-ok,
cheat-ek, keresztrejtvény nyereményekkel,
+4 oldal ENTERPRISE melléklet.

SPECTRUM VILÁG

megjelenik havonta 11000 példányban,
terjeszti a Magyar Posta,
kapható a hírlapárusoknál

SEIKOSHA

SP-1200

VC



Ma még bizonyára kevesen ismerik Magyarországon ezt a Nyugaton már igen elterjedt nyomtatótípust. Azért ajánlom az olvasók figyelmébe, mert előnyös tulajdonságaival messze felülmúlja az MPS 803-ast, amellyel felülről kompatibilis. Leginkább a CITIZEN 120D típusú nyomtatóval lehetne összehasonlítani, hiszen kevés eltéréssel ugyanolyan lehetőségeket kínál.

A gépbe éppúgy befűzhetünk perforált papírt, mint bármilyen más minőségű, A/4-esnél nem nagyobb lapot. Sőt be sem kell fűznünk, csak a vezetőtámlára helyezve fixáljuk. A befűzés és beállítás már automatikus.

A nyomtatást kilenc tű, 120 karakter/másodperc sebességgel végzi. Mi-



2. ábra

0123456789abcdefghijklmnopqrstuvwxyz

0123456789ABCDEFGHIJKLMNPQRSTUVWXYZ

0123456789ABCDEFGHI

0123456789abcdefgh

0123456789ABCDEFGHI

0123456789abcdefgh

0123456789ABCDEFGHI

0123456789ABCDEFGHI

0123456789ABCDEFGHI

0123456789ABCDEFGHI

0123456789ABCDEFGHIJKLMNPQRSTUVWXYZ

0123456789ABCDEFGHIJKLMNPQRSTUVWXYZ

0123456789ABCDEFGHIJKLMNPQRSTUVWXYZ

0123456789ABCDEFGHIJKLMNPQRSTUVWXYZ

1. ábra

3. ábra

0123456789ABCDEFGHI

0123456789ABCDEFGHI

0123456789ABCDEFGHI

0123456789ABCDEFGHI

0123456789ABCDEFGHI

0123456789ABCDEFGHI

0123456789ABCDEFGHI

0123456789ABCDEFGHI

0123456789ABCDEFGHI

0123456789ABCDEFGHI

0123456789ABCDEFGHI

| | |
|-------------------------------|--------------------------|
| CHR\$(14) | DUPLASZELES NYOMTATAS BE |
| CHR\$(15) | DUPLASZELES NYOMTATAS KI |
| CHR\$(27);"-1" | ALAHUZAS BE |
| CHR\$(27);"-0" | ALAHUZAS KI |
| CHR\$(18) | INVERZ NYOMTATAS BE |
| CHR\$(146) | INVERZ NYOMTATAS KI |
| CHR\$(27);CHR\$(108);CHR\$(n) | BAL MARGO BETOLASA |
| CHR\$(27);"Q";CHR\$(n) | JOB MARGO BETOLASA |
| CHR\$(27);"E" | VIZSZINTES DUPLAZAS BE |
| CHR\$(27);"F" | VIZSZINTES DUPLAZAS KI |
| CHR\$(27);"G" | FUGGOLEGES DUPLAZAS BE |
| CHR\$(27);"H" | FUGGOLEGES DUPLAZAS KI |
| CHR\$(16);"n" | KURZOR POZICIO |
| CHR\$(27);"4" | DOLT BETUS NYOMTATAS BE |
| CHR\$(27);"5" | DOLT BETUS NYOMTATAS KI |
| CHR\$(8) | GRAFIKUS UZEMMOD BE |
| CHR\$(15) | GRAFIKUS UZEMMOD KI |
| CHR\$(27);"3" | SORSURUSEG |
| CHR\$(27);"S0" | FELSO NYOMTATAS |
| CHR\$(27);"S1" | ALSO NYOMTATAS |
| CHR\$(27);"T" | NORMAL NYOMTATAS |
| CHR\$(27);"X1" | IROGEP BETUMINOSEG BE |
| CHR\$(27);"X0" | IROGEP BETUMINOSEG KI |
| CHR\$(27);"M" | SURTETT BETU BE |
| CHR\$(27);"@" | SURITETT BETU KI |
| OPEN 13,4,13:PRINT#13 | BETUSURUSEG DUPLAZAS |
| CLOSE 13 | NORMAL BETUSURUSEG |
| CHR\$(12) | LAPDOBAS |

4. ábra

48, 80, 96, 160, 192) (1. ábra). A függőleges sűrűség megválasztásakor a sorok távolságát kell megadni N/216 collban (0-255). Így természetesen a sorok fedhetik is egymást. Mindemellett többféle betűtípust állíthatunk be (2. ábra).

További lehetőségek

A karaktereket megdönthetjük, aláhúzhatjuk, pontduplázást kérhetünk vízszintesen és függőlegesen is (3. ábra).

E műveletek a CHR\$ kódokkal programozhatók (4. ábra). Néhány funkciót a nyomtató kezelőszervéről is kiválaszthatunk: az írássűrűséget, az írógép-betűtípust stb. Ugyanitt kell kérni a lapdobást, a soremelést és az írásengedélyezést is. Mindezek állapotáról négy LED ad felvilágosítást: feszültség = zöld, betűtípus = zöld, papírállás = piros, írásengedélyezés = zöld.

Szükség esetén nemcsak a CBM karakterkészlettel, hanem normál ASCII-ben is dolgozhatunk. Így grafikus üzemmódban akár 1920 pontos víz-

szintes felbontást is elérhetünk. CBM üzemmódban további választási lehetőség a standard és a din karakterkészlet.

A nyomtató hátlapján található kapcsolórendszer különböző, szoftverből nem módosítható funkciók beállítására szolgál (5. ábra) (például a nulla írását kérhetjük áthúzással vagy anélkül, megadhatjuk, hogy a lapvégérzékelő aktív vagy passzív állapotban legyen-e).

A nyomtató egyetlen hátránya, hogy a karakterek függőlegesen nem nyagíthatók, szemben a CITIZEN-nel, amelynél erre is van lehetőség.

BÁRTFAI BARNABÁS

5. ábra

| N | FUNKCIO | BE | KI |
|---|--------------------|----------|------------|
| 1 | Csatornaszam | 5 | 4 |
| 2 | Papirhossz | 12" | 11" |
| 3 | Karakterkészlet 1. | ASCII | CBM |
| 4 | Karakterkészlet 2. | DIN | STANDARD |
| 5 | Irasirany | CSAK ODA | ODA VISSZA |
| 6 | Papirveg erzekelo | NEM MUK. | MUKODIK |
| 7 | Nulla irasa | ATHUZVA | ATH.NEL |
| 8 | CSF | IGEN | NEM |

Minden kedden 17-től 20 óráig
HCC ENTERPRISE klub
a VSZM
Közösségi Házban
(Bp. XI., Fehérvári út 120.)
Klubvezető: Romvári Gábor
Telefon: 810-950/473

Professional

**Olvasóink tájékoztatására közöljük
a Professional
országos számítógépszerviz-hálózatának
címeit.**

1033 Budapest, Kaszásdűlő u. 1-3.

Tel.: 805-587

805-565

805-278

3100 Salgótarján, Rákóczi út 252.

Tel.: 32-13-598

3526 Miskolc, Huba u. 23.

Tel.: 46-89-308

4400 Nyíregyháza, Mártírok tere 9.

Tel.: 42-14-032

4028 Debrecen, Besze J. u. 7.

Tel.: 52-25-687

5601 Békéscsaba,

Tanácsköztársaság u. 75.

Tel.: 66-28-584

6701 Szeged, Retek u. 23.

Tel.: 62-25-448

7690 Pécs, Liceum u. 7.

Tel.: 72-33-955

7400 Kaposvár, Tóth L. u. 12.

Tel.: 82-12-104

8900 Zalaegerszeg, Bíró M. u. 14/A

Tel.: 92-13-789

9700 Szombathely, Rákóczi F. u. 50.

Tel.: 94-13-506

9023 Győr, Buda u. 34.

Tel.: 96-11-440

8000 Székesfehérvár, Móri u. 58.

Tel.: 22-16-814

**ÁTALÁNYDÍJAS
JAVÍTÁSI ÉS = ÖRÖK
KARBANTARTÁSI GARANCIA
SZERZŐDÉS**



**COMMODORE, ATARI, TVC stb. személyi számítógépek
valamint IBM PC/XT/AT professzionális PC számítógépek
és perifériák (floppy, printer)
garanciális és fizető JAVÍTÁSA, KARBANTARTÁSA.**

SZERVIZEINK:

1053 Budapest V., Magyar u. 12-14.

T.: 173-551 Tx: 7621

4034 Debrecen, Holló László u. 14.

5600 Békéscsaba, Bartók Béla u. 37. T.: 66-27-195

1083 Budapest VIII., Szigony u. 9.

T.: 343-153

6726 Szeged, Székelysor 13.

T.: 62-13-377

1191 Budapest XIX., Gábor A. sétány 3.

7400 Kaposvár, Füredi u. 24.

T.: 82-16-307

3100 Salgótarján, Arany János u. 3.

T.: 32-14-007

7624 Pécs, Jurisics M. u. 17.

T.: 72-11-812

3525 Miskolc, Fazekas u. 1-3.

T.: 46-17-011

9700 Szombathely, Szalonok u. 31.

T.: 94-14-519

1053 Budapest, Henszlmann I. u. 9. Telefon: 174-144 Telex: 22-7621

VEVŐSZOLGÁLAT — FOTOELEKTRONIK — NOVOTRADE — ALFA G.T.

VÉTEL — ELADÁS 1077 Budapest, Dohány u. 16. Tel.: 428-936

POKE-kal vagy POKE nélkül?

| | | |
|------------|-----------|-----------|
| DEFAULT | CHANNEL | TIMER |
| KEY | CLICK | RATE |
| DELAY | INTERRUPT | STOP |
| KEY | NET | CODE |
| TAPE | SOUND | PROTECT |
| LEVEL | FAST | SAVE |
| SOUND | STOP | BUFFER |
| SPEAKER | SERIAL | BAUD |
| FORMAT | VIDEO | MODE |
| COLOR | COLOUR | Y |
| X | BORDER | BIAS |
| STATUS | EDITOR | VIDEO |
| KEY | BUFFER | REMI |
| REM2 | NET | CHANNEL |
| NUMBER | MACHINE | PRIORITY |
| ATTRIBUTES | PALETTE | COLOR |
| COLOUR | PAPER | INK |
| CHARACTER | CURSOR | CHARACTER |
| COLOR | COLOUR | SCROLL |
| ON | OFF | UP |
| DOWN | BEAM | ON |
| OFF | LINE | STYLE |
| MODE | | |

1. lista

```

100 'SET ASK további vizsgálata
110 !
120 OUT 177,1:LET S=0
130 FOR I=21213 TO 21543 STEP 5
140 LET S=S+1
150 LET K=256+PEEK(I)+PEEK(I+1)-32768
160 LET H=PEEK(K):LET A$=""
170 FOR R=K+1 TO K+H
180 LET A$=A$CHR$(PEEK(R))
190 NEXT
200 LET A$=A$S" "
210 PRINT A$(0:11)
220 IF S=3 THEN
230 LET S=0:PRINT
240 END IF
250 NEXT
260 PRINT
    
```

2. lista

A C64-tulajdonosok a 2.0 verziójú BASIC szerény szolgáltatásain kívül a POKE utasítást igen gyakran kénytelenek használni. A számítógép-memória megfelelő helyére közvetlenül beírt értékekkel azután csodákat művelnek.

Az egyre magasabb verziószámú BASIC-ek egyre több utasítást tartalmaznak, és így a szín-, a hang- és a grafikus lehetőségek kihasználására új programozási módszerek kínálkoznak.

Az Enterprise BASIC-je hivatovább feleslegessé teszi a POKE utasi-

```

C902 PUSH AF ;állapot mentes
C903 LD HL,BFF2 ;incst csak H kell!
C906 LD A,C ;A-ba EXOS változó száma
C907 CP 01 ;FLAG SOFT IRQ allitau?
C909 JR I,C90E ;akkor címest nem bantunk
C90B ADD A,C5 ;összeadit a cím
C90D LD L,A ;incst a cím BFC5+A
C90E DEC B ;mit csináljak
C90F JP H,C91B ;olvass
C912 JP I,C917 ;irj
C914 LD A,(HL) ;váltó at
C915 CPL ;komplemental
C916 LD D,A ;beírás előkészítése
C917 LD (H),D ;új változó beírás
C918 LD D,(HL) ;értéket visszavias
C919 POP AF ;visszaallit
    
```

3. lista

```

100 ! karakter átalakító
110 ! SET CHARACTER n,BIN.... helyett
120 !
130 LET K=13440
140 PRINT "nyomod le azt a billentyűt"
150 PRINT "amit at akarsz alakítani."
160 PRINT
170 LET A$=INKEY$
180 IF A$="" THEN GOTO 170
190 PRINT A$;" ----> ";
200 LET KOD=ORD(A$):LET KEZD=K+KOD
210 FOR I=0 TO 8
220 READ MIT
230 SPOKE 255,KEZD+I*128,MIT
240 NEXT
250 PRINT CHR$(KOD)
260 DATA 62,65,85,65,73,65,93,65,62
    
```

4. lista

tást — no azért el ne felejtük! —, mert összetett kulcsszó-kombinációival lehetővé teszi szín-, hang-, rajz- és átdéfiníálási feladatok megoldását.

A SET, ASK kulcsszóval kezdődő utasítások természetesen nem csinálnak mást, mint az argumentumként megadott további információk alapján meghatározzák azt a — felülírtá — memóriacímeket, amelyek tartalmát meg kell változtatni a kívánt cél eléréséhez.

Az Enterprise tehát egy adott feladat elvégzésére általában több lehetőséget kínál. A programozó hangulától, stílusától, korábbi beidegződéseitől függ, hogy melyik megoldást választja. Igaz, mindegyik programozási stílushoz meg kell tanulni a megfelelő kulcsszavakat, EXOS változó sorszámot és/vagy a memóriacímeket.

Természetesen nem kell ezeket az értékeket ész nélkül megjegyezni. Használjuk a gépkönyvet vagy a megfelelő segédletet, mert a bőség zavarában bizony nemegyszer nehéz eligazodni.

A SET, ASK, TOGGLE kulcsszavak után az interpreter például a következő kulcsszósorozatára, illetve azok kombinációjára végzi el az elemzést (1. lista). Az ebben szereplő szavakat mi is megkérdezhetjük a géptől a 2. lista programjának segítségével.

Amennyiben a kiegészítő információ szolgáltató szó vagy szavak azonosítása eredménnyel jár, akkor az EXOS 10 H rutin kiszámítja az EXOS változó memóriabeli címét és oda a megfelelő értéket beírja. Az azonosítás utáni memóriacím kiszámításának és beírásának interpreter általi megvalósítása a 3. listán látható programmal lehetséges. A programból kiolvasható, hogy az EXOS változók helye a

BFC5H = 49093D címen kezdődik. Tehát a SET BORDER 110 SET 27,110 POKE 49093+27,110 utasítás azonos hatást eredményez.

A gép mellé adott Felhasználói kézikönyv ismerteti a karakterek átdefiniálására szolgáló programot (93. oldal). Ennek a programnak a hatását is elérhetjük POKE utasítással.

Mielőtt beíránk és futtatnánk a 4. lista programját, egy-két megjegyzést vegyünk figyelembe. A SET CHARACTER CODE (123) utasítás az a betű alakjának átírását teszi lehetővé. Minthogy a betűk alakjait tartalmazó bittérkép a 255. szegmens 13440. helyétől kezdve található meg, csak a megfelelő címekre kell beírni a POKE utasítással a megfelelő értékeket.

A program segít a billentyűk átközlésében.

A 260-as DATA sorba azt a kilenc értéket kell beírni, amit a karakter alakjának megtervezésekor kiszámítottunk. A program, elindítás után, kér egy billentyűlenyomást, és a lenyomott billentyűt a 210-240-es sorok segítségével átalakítja a DATA sor adatainak megfelelő alakra.

Egy szó, mint száz, még az olyan komfortos BASIC mellett sem kell elfeledkezni a POKE utasításról, annál kevésbé, mivel itt is vannak olyan feladatok, amelyek megoldására nincs más lehetőség.

SZ. LUKÁCS JÁNOS

A Professional Szerviz 1985-ben alakult meg az Agroidustria Innovációs Vállalaton belül. Tizenkét vidéki városban az egész országot behálózó, egymással szoros kapcsolatban álló egységekből szervizhálózatot alakított ki. Az Enterprise gépek garanciális javítását kezdetől fogva ellátják. Az eközben gyűjtött tapasztalataikról kérdeztük meg Gottfried Tibor szervizvezetőt.

M. M. Miért a Professional Szervizre esett a választás, amikor szerződést kötött a forgalmazó az Enterprise gépek garanciális javítására?

G. T. A Professional a Novotrade márkaszervize, és ezért minket ajánlott a Centrumnak. Különböző hagyomány van már nálunk a kisgépek javításának, mert a C Plus/4-et és egyéb iskola-számítógépeket is mi szervizeljük.

M. M. Nem vállaltak-e túlzottan nagy kockázatot azzal, hogy egy nálunk teljesen ismeretlen, új géptípussal kezdtek el foglalkozni?

G. T. Számunkra nem volt ismeretlen, ugyanis előzőleg kaptunk három gépet. Ezeket szétszedtük és megvizsgáltuk a szervizelés szempontjából. Azért sem volt annyira új, mert az Enterprise nagyon hasonlít a ZX-Spectrum-ra. Tulajdonképpen egy általános Z80 alapú gép, amelyet két koprocesszorral „okosabbá” tettek. Az üzlet anyagi kockázatát egy általunk felállított menedzsercsoport gazdaságossági mutatók alapján is jónak ítélte.

M. M. Elterjedt az a hír, hogy ezek a cég nyakán maradt, félig összeszerelt gépek voltak, és a megrendelések úgy „hányták össze” azokat.

G. T. Nagyon sok hír terjedt el, jó és rossz egyaránt. Ennek az lehet az oka, hogy soha nem látott mennyiségű gép jött be a magyar piacra, viszonylag alacsony áron. A számítógép olyan, mint a gépkocsi: mindenkinek a sajátja a legjobb. Én attól félek, hogy az Enterprise-ral is elkövetik azt a hibát, amit a C64-vel: nem lesz hozzá hardver- és szoftverkiegészítés. Végül azonban gyorsan reagált a Novotrade, az „A” Stúdió és a Centrum is. Itt jegyzem meg, hogy minden hardverkiegészítőt is mi szervizelünk.

M. M. Van-e az Enterprise-nak tipushibája?

G. T. Ami van, az a fizikai kialakításából adódik, ugyanis a gép nagyon lapos. Az ebből eredő hátrányok persze „kivédhetőek”. Leggyakoribb a fóliataszatúra törése, főleg a botkormány alatt. Nagyon sokszor elmondott, hogy a beépített botkormány nem játéka való. Előfordul, hogy a gép hátulján levő kis magnetofon-csatlakozó kitörik. A csatlakozókra vigyázni kell, mert elég gyengék. Nem tanácsos más magnetofon csatlakoztatni, mint amelyiket a géphez árulnak. Már csak azért sem, mert a magasabb remótárnál elrontja a gépet. Igaz, tönkre nem teszi, de javítást igényel.

M. M. Nem hátrányos-e szervizelés szempontjából, hogy két típus érkezett be?

G. T. Nem! A két típus szervizelése megegyezik.

M. M. Egyik olvasónk megkérdezte, az Enterprise gépre is igaz-e, hogy a színes televízió színeit kiegészíti?

G. T. Igaz, mert tévéműsor vételére is alkalmas! A számítógép a képszo azonos pontjait szintén használja, és ezek a területek idővel

G. T. Ilyen hibát nem tapasztaltunk. Ha a gép három órát hibamentesen üzemel, akkor nem téved. Ha rossz a gép, ez elég korán kiderül. Néhány jellemző hibalehetőség: a „tünet” attól függ, hogy melyik alkatrész melegszi. Ha a Z80 mikroprocesszor, akkor alapvető funkciókat sem kezel. Amennyiben a RAM-ban van a hiba, kismul a kép és nem kezd el a tesztet. A Nick chip hibája nem megengedett vízszintes csíkokat okoz, a Dave pedig eltorzuló katodóg hangokat eredményez, ha nem működik

MEGKÉRDEZTÜK AZ ENTERPRISE -RŐL

bemattulnak. Ezt azonban több száz üzemóra idézi elő. Vigyázni is lehet a képszo. A színelítettséget és a fényerőt le kell venni, és ha nem programozunk, akkor kapcsoljunk át másik csatornára. A játékprogramok nem ártnak, mert azok úgy viselkednek, mint a tévéadás.

M. M. A programok kazettáról általában jól tölthetők. Néhány olvasónk mégis panaszkodott, hogy bizonyos típusú tévék visszahatnak betöltéskor.

G. T. A Bejingnél és a legtöbb Videoton-tévénél mi is észleltük ezt a visszahatást.

M. M. Nekem ez elég elképzelhetetlennek tűnik. Hogyan fordulhat elő?

G. T. Valószínűleg egy földhurok alakul ki, de ennek megállapításához olyan műszerek kellene, amelyek nekünk nincsenek a birtokunkban. Azt javaslom, hogy a két televíziót gyártó cég és a szerviz együtt vizsgálja meg ezt a dolgot. A Beijing akkor is lehetetlenül teszi a betöltést, ha a gép nincs rákötve, de a közelében működik. Sajnos a műszerek hiánya miatt nem tudok pontosabban választ adni. A Videotonnal a kapcsolatot már felvettük.

M. M. Hány Enterprise-t hoztak be javítani az elmúlt időben?

G. T. Az ügyfeleink száma szeptember elsejéig körülbelül 2400 volt. Többségük valamilyen tanácsot kért. A legtöbb kérdés a magnetofon beállításával volt kapcsolatban. Legtöbbször a fejen kellett állítani. A programbetöltés visszatérő gond, igaz, sokszor a hozzá nem értésből adódik.

M. M. Találták-e konstrukciós hibát?

G. T. Tervezésit nem. Egyéb beállítási hibák előfordulnak, mint a szín-, illetve szinkronizációs.

M. M. Hogyan viseli az Enterprise a melegezőt? Téved-e a gép hosszú üzem alatt?

rendesen. Ha a kép „összetörik”, az hangolással helyrehozható.

M. M. Ezek a gépek egyre idősebbek lesznek. Gyarapodnak-e ezáltal a meghibásodások?

G. T. Nem! Legalábbis egyelőre egyre kevesebb a meghibásodás.

M. M. Adna végeztetül néhány tanácsot az Enterprise-tulajdonosoknak?

G. T. A meglévő programokat érdemes egy igen jó minőségű magnetofonnal rögzíteni. A gépet nem szabad zárt helyen, például íróasztalra beépítve üzemeltetni. Sok mérgeződéstől és költségtől megkímélhetjük magunkat, ha a tasztatúrát és a beépített botkormányt finoman kezeljük. Érdemes külön botkormányt vásárolni. Ez azért is fontos, mert a fóliataszatúra-csere körülbelül 1500 forintba kerül, ha a gép már nem garanciális. A gépen lévő nem használt csatlakozók nyílt sebek. Ha ezek statikus elektromossággal feltöltött tárgyhoz érnek, az katasztrófális hibát okozhat. Ha nem használjuk a csatlakozót, ne vegyük le a jobb oldali takarólemez. Végül egy kis kuriozum, amivel újjávarázsolható a Remot Reed relé. Az alábbi két programot futtassuk le:

10 TOGGLE REM1 10 TOGGLE REM2
20 GOTO 10 20 GOTO 10

A program ellenkező állapotba váltja a Reed relé állapotát, majd a 20-as sorban ciklusszerűen újra utasítást kap erre. A stop billentyű megnyomásáig egy halk, percegő hangot hallhatunk. A programot tíz-húsz másodpercig futtatva elvégezzük a relé mágnesésztét.

PINKE GYÖRGY

FEDEZZÜK FEL EGYÜTT! AZ ELLIPSZIS ÉS A TÖBBIK

Nagyon izgalmas feladat és jó szórakozás egy „fekete doboz”, az ismeretlen működését megismerni, leírni. Számítógépünket, az Enterprise-t is felfoghatjuk ilyennek, amiről azt tudhatjuk meg a Felhasználói kézikönyvből, hogy milyen BASIC nyelvű parancsokat, utasításokat fogad el és hatásukra mit csinál a számítógép. Akik a könyv segítségével már megismerték az Enterprise-t, azoknak csupán csak más utat ajánlunk, ami persze sok meglepetéssel szolgálhat.

Felfedező utunkra tehát minden Enterprise-tulajdonost meghívunk, figyelembe véve, hogy kezdők is vannak — és remélhetőleg egyre többen lesznek — közöttünk.

Cikksorozatunkkal csak a felfedezés irányítására vállalkozunk. Azt szeretnénk, ha a felfedezés öröme az olvasó élné át, akkor, amikor kis programjainkat futtatja, módosítja, kiegészíti és önállóan is készít programokat. Az utat olyan programok segítségével járjuk be, amelyek bemutatják a „fekete doboz” működését, alkalmazását, ugyanakkor igénylik az olvasó aktív közreműködését.

A Felhasználói kézikönyvre egyébként többször fogunk hivatkozni, mert abban megtalálható az egyes témakörök részletesebb leírása. A hivatkozás jelölése például: F/7, ami azt jelenti, hogy ezen az oldalon kezdődik a számítógép összeállítását leíró fejezet. Amennyiben más iránk a cikkben, mint ami a könyvben olvasható, úgy hívjuk segítségül a számítógépet a vita eldöntéséhez.

Segítségnkre lehet a tanulásban az alábbi algoritmus, amivel a javasolt tanulási módszert mutatjuk be.

1. Gépeljük be az első programot!
2. Indítsuk el a programot F1 START vagy a megszokott RUN parancsral!
3. Figyeljük meg, hogy mi történik, és fogalmazzuk meg, hogy mit csinálnak a programban szereplő utasítások!
4. Olvassuk el a programhoz tartozó magyarázatot! Nézzük meg a segéd-könyvekben az utasítások leírását!
5. Kísérletezzünk! Változtassuk meg a programban szereplő adatokat, és figyeljük meg, mi történik! Ellenőrizzük az értékhatarokat, az értelmezési tartományokat!
6. Oldjuk meg másféppen a feladatot! Egészítsük ki! Bővítsük a programot!
7. Keressünk feladatokat, amelyeknek a megoldásához felhasználhatjuk a tanultakat!
8. A cikk végére értünk!

— Igen: próbáljuk meg hasznosítani a tanultakat! Jó munkát kívánunk! Vége.

— Nem: összehasonlítjuk a következő programmal a már begépel programot, és szükség szerint módosítjuk vagy töröljük a programot és begépeljük az újat, aztán folytatjuk a munkát a 2. ponttal.

No, de vágunk neki az utunknak! Első lépésként gépeljük be az 1. listán lévő programot. Rögtön egy kis segítség: rossz billentyű lenyomásakor a felesleges karakter(ek)et az ERASE vagy a DEL billentyű lenyomásával törölhetjük. Ha kihagytunk karakter(ek)et, akkor az INS billentyűvel csinálhatunk helyet. A kis botkormánnyal vihetjük a kurzort a javítás helyére. Az utasítássorok végén, illetve a javítás után ne feledkezzünk meg az ENTER billentyű lenyomásáról, mivel az utasítássorok beírása vagy a javítás csak ekkor válik véglegessé. Erről meggyőződhetünk, ha begépeljük a LIST parancsot. A parancsokat is az ENTER billentyű lenyomásával zárjuk le. Akkor is láthatjuk a program listáját, ha a 2-es funkcióbillentyűt nyomjuk le.

```
1 REM --- 1. program ---
10 GRAPHICS HIRES 2
20 PLOT 600,360
30 PLOT ELLIPSE 200,100
40 PLOT 600,360,PAINT
```

1. lista

```
1 REM --- 2. program ---
10 GRAPHICS LORES 256
20 PLOT 600,360
30 PLOT ELLIPSE 200,100
35 INPUT X
40 PLOT X,360,PAINT
50 GOTO 35
```

2. lista

```
1 REM --- 3. program ---
10 GRAPHICS HIRES 2
20 INPUT X,Y
30 PLOT X,Y;
40 GOTO 20
```

3. lista

```
1 REM --- 4. program ---
10 GRAPHICS HIRES 256
20 FOR I=1 TO 100
30 LET X=RND(1280):LET Y=RND(620)
40 SET INK RND(256)
50 PLOT X,Y;
60 NEXT
```

4. lista

```
100 REM --- 5. program ---
110 GRAPHICS HIRES 256
120 SET PAPER 96
130 CLEAR GRAPHICS
140 LET X1=100:LET Y1=200
150 RANDOMIZE
160 FOR I=1 TO 200
170 LET X=RND(1280):LET Y=RND(720)
180 SET INK RND(256)
190 PLOT X1,Y1;X,Y
200 SET INK 96
210 PLOT X1,Y1;X,Y
220 LET X1=X:LET Y1=Y
230 NEXT
```

5. lista

A programot a RUN parancsral indíthatjuk el. Miután megnéztük a képernyőn megjelenő ellipszist, értelmezzük a programban szereplő utasításokat!

10-es sor: kétszínű finom felbontású (HIRES) grafikus képernyőre rajzolunk.

20-as sor: a „ceruzát” a 600,360 koordinátájú képernyőpontra helyezzük. Ezzel jelöltük ki az ellipszis középpontját.

30-as sor: Rajzolunk egy ellipszist, amelyiknek 200, illetve 100 egység hosszúságúak a tengelyei. Egy egység két képernyőpont távolságának felel meg.

40-es sor: a „ceruzát” az ellipszis középpontjába helyezzük, és a PAINT utasítással be akarjuk festeni az ellipszist. A festés csak akkor indul el, ha a kiindulópont nem esik az ellipszis középpontját jelző pontra (F/82).

Egy grafikus pont több képernyőpontból áll. Változtassuk meg a 40-es sorban a koordinátákat és állapítsuk meg, hogy hány képernyőpontból áll egy pont a kétszínű grafikus képernyőn! Ezután a 10-es sor átírásával nézzük meg a többi grafikus képernyőt is. Ellenőrizzük az alábbi táblázatot, ahol

- A: a különböző felbontású és színszámú grafikus képernyőt előállító utasításokat jelöli,
 B: egy pontot előállító képernyőpontok száma vízszintesen,
 C: függőlegesen,
 D: a képpontok száma vízszintesen,
 E: függőlegesen.

| | A | B | C | D | E |
|--------------------|----|---|-----|-----|---|
| GRAPHICS HIRES 2 | 2 | 4 | 640 | 180 | |
| GRAPHICS LORES 2 | 4 | 4 | 320 | 180 | |
| GRAPHICS HIRES 4 | 4 | 4 | 320 | 180 | |
| GRAPHICS LORES 4 | 8 | 4 | 160 | 180 | |
| GRAPHICS HIRES 16 | 8 | 4 | 160 | 180 | |
| GRAPHICS LORES 16 | 16 | 4 | 80 | 180 | |
| GRAPHICS HIRES 256 | 16 | 4 | 80 | 180 | |
| GRAPHICS LORES 256 | 32 | 4 | 40 | 180 | |

Gyorsabban megkereshetjük az értékeket a 2. listán látható programmal. A 35-ös sorban az X változó értékét kérjük, azaz a festés kiindulópontjának vízszintes koordinátáját. Indítsuk el a programot, és a kérdőjel megjelenése után írjuk be a koordináta értékét, és nyomjuk le az ENTER billentyűt. Az 50-es sorról mindig a 35-ös sora urunk vissza a GOTO utasítással, ennél fogva folyamatosan kéri a program a koordinátákat. Ebből a „végtelen ciklus”-ból a STOP billentyű lenyomásával léphetünk ki. Írjuk a 35-ös sorba az INPUT "PONT":X utasítást. Ekkor a PONT szó jelenik meg a képernyőn, és ez után kell begépelnünk a pont első koordinátáját. Így írathatjuk ki azt, hogy mit kérünk. Figyeljük meg, hogy csak a négy alsó sorban látszik a kiírás. Ennyi a szöveges képernyő, a képernyő többi része grafikus. Ide rajzolhatunk.

A 3. listának a programjával szakaszokból ábrát rajzolhatunk a képernyőre. A szakaszok végpontjainak koordinátáit úgy kell megadnunk, hogy a két adat közé vesszőt teszünk és a második adat után lenyomjuk az ENTER billentyűt.

A program begépelése előtt adjuk ki a NEW parancsot, amivel töröljük az előző programot.

A 30-as sorban az PLOT utasításban a koordináták után a pontosvesszővel azt jelezzük, hogy letesszük a „ceruzát” a „papírra”. Ez azt jelenti, hogy amikor a következő pontba mozdítjuk a „ceruzát”, akkor rajzolni fog egy szakaszt.

Keretezzük be a képernyőt, azaz keressük meg a legkisebb és legnagyobb koordinátájú pontjait a képernyőnek!

Most a számítógép adja meg véletlenszerűen a szakaszok végpontjait (4. lista). A begépeléshez segítség: az egyenlőségjel és a zárójelk begépelésekor a SHIFT billentyűt lenyomva kell tartani. A képernyőpontok vízszintes koordinátáit 1280-nál kisebb, függőleges koordinátáit 720-nál kisebb nem negatív egész számmal adjuk meg. Az RND-függvények ilyen számokat állítanak elő a programban ($0 \leq X < 720, 0 \leq Y < 1280$). A 256 szín közül az RND-függvénnyel választjuk ki a „ceruza” színét (30-as sor).

A FOR és a NEXT utasítás együtt hoz létre egy ciklust, azaz ismétlődést. I értéke 1-től kezdődően egyesével növekszik 100-ig, tehát százszor ismétlődnek a FOR és NEXT utasítások között megadott utasítások, vagyis száz képernyőpont kerül a képernyőre (F/59).

- Írjuk a 15-ös sorba a RANDOMIZE utasítást!
- Töröljük az 50-es sorból a pontosvesszőt! Mi történik?
- Írjuk át az 50-es sort: 50 PLOT 640,360:X,Y.
- Adjuk ki a RENUMBER (SHIFT és a 3-as funkcióbillentyű), majd a LIST (2-es funkcióbillentyű) parancsot (F/46).

Most törölni fogjuk a megrajzolt szakaszt, csak egy szakaszt látunk majd a képernyőn (5. lista). A kirajzolt szakasz végpontja lesz a következő szakasz kezdőpontja, ezért az X és Y változók értékeit átesszük, és így megörözzük az X1 és Y1 változóknak (F/28).

A SET PAPER utasítással a papír színét állítjuk be, és szükség van a CLEAR GRAPHICS utasításra is, hogy a képernyő beszíneződjön. A vonalat úgy töröljük, hogy a „ceruza” színét a „papír” színére állítjuk (180-as sor) (F/84).

DUSZA ÁRPÁD

Mi a manó?

Fórum az Enterprise-ről

A Centrum Áruházak Vállalat és a Mikroszámítógép Magazin szervezésében fórum lesz az Enterprise géppel kapcsolatban. A feltett kérdésekre a Centrum, az „A” Stúdió, a Novotrade, a Professional Szerviz és az Enterprise cég NSZK-beli képviselői válaszolnak. A kérdéseket írásban a szerkesztőségünkbe várjuk, de a helyszínen is feltehetik. A fórum helyéről és időpontjáról a napilapokban jelenik meg tájékoztatás.

Enterprise-összeszerelés Magyarországon?

H. M. Windisch, az Enterprise Computer GmbH igazgatója elmondta, hogy magyarországi panelgyártást, illetve összeszerelést terveznek. Jelenleg tárgyalásokat folytatnak, de igen pozitívan ítélik meg a lehetőséget. Tipusváltást is terveznek az új Enterprise-gép standard DIN-szabvány-csatlakozókkal is. A konstrukciót is modernizálták, a gép gyorsabb lesz, tárhelykapacitását bővítik, és önálló külső tasztatúra is csatlakoztatható hozzá. Így például az IBM PC-billentyűzet is. A jelenlegi elképzelések szerint az új gép olcsóbb lesz.

Enterprise a Szovjetunióban

Az Enterprise GmbH tárgyalásokat folytatott a Szovjetunióval. A várható igény 100 000 db/év. A Szovjetunióba szánt gépek nyomtatási áramkörű kártyáit egyelőre Magyarország kívánják gyártatni. Ebben az ügyben is biztató tárgyalásokat folytatnak.

Enterprise PC

Az Enterprise cég PC-kegy gyártatni, ezek kompatibilisek lesznek a jelenleg kapható gépekkel. Az új Enterprise PC a jövő év első negyedében jelenik meg az NSZK-piacon. Az Enterprise céggel megállapodunk abban, hogy magazinunk részére folyamatosan kis híreket és anyagokat küld. Az átlaluk kiadott Enternews című kiadványból bármelyik cikket átvehetjük. A hozzánk érkezett olvasói kérdéseket is szívesen megválaszolják. Lapunk olvasói várhatóan kedvezményt kapnak az Enterprise cég müncheni irodájában.



Gaetsch Günterné rajza

A sorozat alap gondolata — azon a régi felismerésen túl, hogy az elektronika és a számítástechnika elválaszthatatlan egymástól — a következő tapasztalatot summázza. A szoftver — a programok — jelentősége egyre nő, de az is tény, hogy az igazán jó (az adott számítógép nyújtotta lehetőségeket maximálisan kihasználó) programok megírásához a programozónak rendelkeznie kell alapfokú áramkört hardverismerettel is. Megerősíti ezt, hogy szaporodik az olyan berendezések, mikroprocesszort alkalmazó rendszerek száma, amelyek programvezérelten működnek. Az ilyen rendszerek tervezőinek és fejlesztőinek is szükségük van integrált hardver- és szoftverismeretekre.

Hardver

Be- és kimeneti áramkörök

A mikroszámítógépes rendszerekben a külvilággal, tehát a környezettel a kapcsolatot a processzorhoz kapcsolódó be- és kimeneti egységek teszik lehetővé. Ezek olyan elemek, amelyekben keresztül a környezet felől bináris információkat juttathatunk a rendszerbe — ezek a bemenetek —, vagy pedig rajtuk keresztül bináris adatokat vihetünk át a környezetbe — ezek pedig a kimenetek. Szókat az angol elnevezésük alapján inputoknak (bemeneteknek) és outputoknak (kimeneteknek) és röviden I/O egységeknek is nevezni.

A mikroszámítógépben az adatokat a mikroprocesszor, a tár és a bemeneti áramkörök között párhuzamosan, az adatbuszon keresztül kezeljük. A külső adatokat azonban akár párhuzamosan — hasonlóan az adatbuszhoz való adatáramláshoz —, akár sorosan, vagyis bitenként, sorban egymás után kezelheti a mikrogep.

A mikrogepes rendszerek környezethez csatlakoztatásának leg egyszerűbb eszközei az ún portok (I/O port). Alapvető sajátosságuk, hogy a hozzájuk érkező jeleket vezérlő parancsra mintavételezik, és mindaddig tárolják, amíg újabb mintavétel nincs. Ez a megoldás az időztési problémákat jelentősen csökkenti. A belső adatbuszra kötött kimeneti port az adatviteli utasításkor mintát vesz a busz jeléből, és megőrzi akkor is, amikor az adatbusz már egészen más információ van. A bemeneti portok egy része is képes a bemenetére kerülő információt megőrizni, és azt adott parancsra a belső adatbuszra kapuzni. Az egyszerűbb bemeneti portoknak nincs ilyen tárolási képességük, és a kapuzó jel hatására a bemenetükön lévő, éppen aktuális tartalmat juttatják a belső adatbuszra.

A következőkben először a párhuzamos adatvitelt, majd a soros adatvitelt tekintjük át.

Párhuzamos adatátvitel

A mikroszámítógépes rendszerek leggyakoribb adatviteli for-

mája. A mikroprocesszor oldaláról nincs lényeges különbség, hogy az adatforgalom a memóriával vagy a külvilággal kapcsolatos. Mivel a kibocsátással címezhetünk, az adatátvitelre pedig az adatbuszt használjuk, a helyet csupán a vezérlőjelek segítségével különböztethetjük meg. A címbuszra adott címkombinációval választhatjuk ki az adatátvitel célját (angolul: destination, ejtsd desztinésön) vagy forrását (angolul: source, ejtsd szorsz). Az egység kiválasztását a címdekóder áramkör végzi, aminek a kimenete akkor aktív, ha a címsímen lévő címkombináció a dekóderen beállítottal megegyezik.

Az átviteli irányát az RD (olvasás), illetve a WR (írás) vezérlőjel határozza meg. Azt, hogy memóriával vagy a külvilággal kapcsolatos az adatátvitel, a MEMRQ (memory request, ejtsd memóri kérés) vezérlőjelek döntik el.

Ez a változó megoldás a Z80 processzorra jellemző. A 8080-as mikroprocesszornál a MEMRQ és IORQ jeleket egyetlen jellé vonták össze, ez az IO/M jel. A jel magas szintje be/kimeneti műveletet, alacsony szintje memóriaműveletet jelez. Bár a 16 bites címbuszon 65536 cím jelölhető ki, a 8 bites mikroprocesszorok tervezői csak a címbusz alsó felét használták perifériacímzésre. Így 256 perifériacím alakítható ki.

Konkrét példaként nézzünk egy nagyon egyszerű, Z80-as mikroprocesszorhoz kapcsolódó, 16 be- és 16 kimeneti vonalas perifériacímalkitást, amely az 1. ábrán látható.

Bemenetként olyan 8 bites meghajtó áramköröket (74LS244) használunk, amelyeknek háromállapotú kimenetük van. Az áramkör a bemenetere kapcsolt bináris információt akkor tudja a mikroprocesszor adatbuszára és így a mikroprocesszorba küldeni, ha az engedélyező bemeneti (1G, 2G) aktív, alacsony szintűek. Ilyenkor az áramkörnek az adatbuszra kap-

csolódó háromállapotú kimenetei az eddigi nagy ellenállású (lebegő) állapotból aktív állapotba kerülnek, és a bemeneten lévő értékeket az adatbuszra kapuzzák.

A be-, kimeneti kapuk címeinek dekódolását az U6A jelű, kettőről négyre dekódoló áramkör végzi. Ha az A0=L és A1=1 szintű és IORQ=L, akkor U6A 4. lába lesz alacsony szintű, ami lehetővé teszi U1 és U3 kiválasztását. A cím felső hat bitjének értéke közömbös, azaz nem vesz részt a periféria címeinek kialakításában. Ezt a szokásos módon XXXXX00 alakban jelezzük. Ez a kimeneti jel az RD és WR vezetékek segítségével határozza meg azt, hogy az adott perifériacímről olvasni vagy a címre írni akarunk. Mivel tehát a 74LS244 áramkörnek két egymással E4 kapcsolatban lévő engedélyező bemenete van, ezért egyszerűen a jelvezetékek rákötésével oldható meg a kapuk aktivizálása (a másik bemenetre az RD jelet kötjük). Hasonló módon az U2 és U4 periféria címe: XXXXX01.

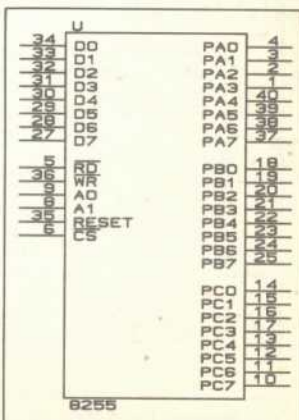
Kimenetként 8 bites 74LS273 típusú tároló áramköröket használunk. Ezek bemenetét az adatbuszra kapcsolódunk. A bemeneteken lévő tartalom akkor íródik be a tárolóba — és így a kimeneten is megjelenik —, ha a beíró bemenetén alacsony szint van. Ezt a jelet a címkiválasztó jel és a WR jel együtteséből állítjuk elő az U5 jelű VAGY kapukkal. Ezek kimenete akkor lesz a tárolók aktivizálásához szükséges L szintű, ha mindkét bemenetük L szintű. A RESET jel a tárolók alaphelyzetbe állítására (törölésre) szolgál.

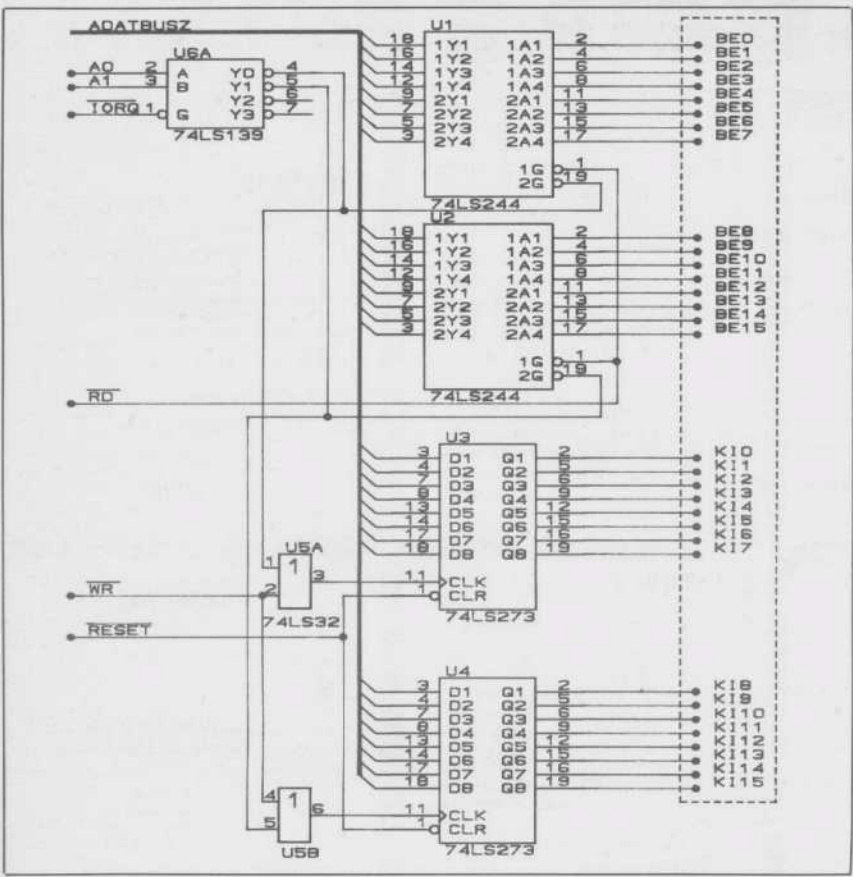
Az előbbieket jól illusztrálják, hogy mennyire hasonló a mikroprocesszorok memóriá- és be/kimeneti egységének a kezelése. Ezek szerint lehetőség van arra, hogy egy memóriarész helyére be/kimeneti áramköröket tervezzünk, a kiválasztásukat az IORQ jel helyett az MREQ jellel végezzük el. Egy tárolási ciklus során az adat-sín állapota beolvasódik a mikro-

processzorba, függetlenül attól, hogy honnan származik az adatsín tartalma. A memóriaként kezelt be/kimeneteket szókat memóriába ágyazott (memory mapped, ejtsd memóri meppid) be/kimeneteknek nevezni. Ezek alapján nem meglepő, hogy van olyan mikroprocesszor, például a Motorola cég 6800-as sorozata — a család egyik eleme a C64 központi egysége —, amelynek egyáltalán nincs be/kimeneti utasítása! A működéshez szükséges be/kimeneteket memóriába ágyazva alakítják ki.

Be/kimeneti egységeknek — a szokásos TTL áramkörökön kívül — kifejlesztettek sokféleképpen felhasználható, a mikroprocesszor buszára közvetlenül kapcsolódó áramköröket is. Ezek tartalmazzák egyrészt azokat az áramköröket, amelyek a busszal teremtene kapcsolatot, másrészt a külvilághoz kapcsolódó be/kimeneti áramköröket.

2. ÁBRA





1. ÁBRA

3. ÁBRA

| A1 | A0 | RD | WR | CS | MŰKÖDÉS |
|----|----|----|----|----|-----------------------|
| L | L | L | H | L | A kapu → D0...D7 |
| L | H | L | H | L | B kapu → D0...D7 |
| H | L | L | H | L | C kapu → D0...D7 |
| L | L | H | L | L | D0...D7 → A kapu |
| L | H | H | L | L | D0...D7 → B kapu |
| H | L | H | L | L | D0...D7 → C kapu |
| H | H | H | L | L | D0...D7 → par.reg. |
| X | X | X | X | H | ez IC nincs kijelölve |
| H | H | L | H | L | ÉRVÉNYTELEN |

Ezeknek az áramköröknek az üzemmódját — például azt, hogy egy adott vonal bemenetként vagy kimenetként működjön — az IC-ben lévő parancsregiszter(ek)be való írással határozhatjuk meg, amit úgy mondunk, hogy az áramkört programozzuk. Igen sokféle ilyen áramkör van. Jellegzetes példaként egyet, az Intel 8255 típusú, programozható perifériavezérlő egységet mutatjuk be. A tok bekötési rajza a 2. ábrán látható.

Az áramkör 24 programozható be/kimeneti vonalat tartalmaz. A be- és kimeneti vezetékeket két, AC, valamint BC jelű csoportban kezeli az IC. Az AC csoporthoz tartozik a 8 bites A jelű adatkapu és a C kapu négy magasabb helyi értékű bite, a többi a BC csoport.

Az áramkör jelcsatlakozásainak szerepe a következő:

A0, A1 Az IC belső egységeinek kiválasztására szolgáló bemenet

- CS Az IC kiválasztójele, aktiv alacsony szintű
- D0—D7 Az adatbuszra kapcsolódó adatvonalak
- RD Aktiv alacsony szintű jel, azt jelzi, hogy a CPU adatot olvas a 8255-ből
- WR Aktiv alacsony szintű jel, azt jelzi, hogy a CPU adatot ír a 8255-be
- RESET Törlőjel, törli a belső regisztereket, a kapukat bemenet üzemmódba állítja, aktív magas szintű
- PA0—PA7 Az A adatkapu vonalai
- PB0—PB7 A B adatkapu vonalai
- PC0—PC7 A C adatkapu vonalai

A RESET jel megszűnésekor mindhárom kapu bemenetként működik. Ha így akarjuk használni, a tokot már programozni sem kell, azonnal használható. Ha más adatránnyt vagy üzemmódot kívánunk elérni, akkor azt a használat előtt a parancsregiszterbe írt vezérlőszavakkal állíthatjuk be. A vezérlőszót jelentő bájtot az A0=H és az A1=H cím bitek mellett írjuk a parancsregiszterbe.

A 8255 három üzemmódban használja kapuit:

MODE 0 üzemmód. Ilyenkor beállítható, hogy a négy csoport (A port, B port, C port alsó négy bite, C port felső négy bite) bemenetként vagy kimenetként működjön. A C portnak minden vonala a vezérlőszóval egyenként is be- vagy kimenetnek programozható. Az adatsín és a külvilág közötti kapcsolatot a vezérlőjelek alakítják ki, amit a 3. ábrán foglaltunk össze.

MODE 1 üzemmód. Ebben az üzemmódban a 8 bites adatátviteli vonalon kívül három vezérlőjel is közreműködik a megfelelő adatátvitelben. Az adatátvitel lehetséges az A vagy a B kapun keresztül, bemenetként vagy kimenetként működtetve. A kapuk vezérlése megszakításkérő logikával is kiegészül.

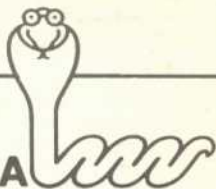
MODE 2 üzemmód. Csak az A kapu üzemelhet ilyen módon. Öt vezérlőjellel és megszakításkérő logikával segített, kétirányú adatátvitelt tesz lehetővé ez az üzemmód.

A tok részletes programozásáról bőveges magyar nyelvű irodalom található.

Egy feladatnál mindig el kell dönteni, hogy milyen típusú periféria-áramkörrel készítjük el az illesztést, ami függ a feladat jellegétől, a mikroprocesszor típusától, valamint az ár- és beszerzési viszonyoktól.

Az egyszerű felépítésű be/kimeneti portok és a nagy bonyolultságú periféria-áramkörök bő választéka lehetővé teszi csaknem minden illesztési feladat egyszerű és hatékony megvalósítását.

DR. KÓNYA LÁSZLÓ



COBRA

ELEKTRONIKAI ÉS SZOLGÁLTATÓ KISSZÖVETKEZET
1097 Budapest, IX. Illyus ut 7. Telefon 476-160/388

KISSZÖVETKEZETEK!

| | |
|-------------------------------------|--------------|
| Egyedülálló kínálatunk: | |
| Számlakészítő program | 19 900,— Ft |
| Számlanyilvántartó program | 24 900,— Ft |
| Bér- és jövedelem-számfejtő program | 24 900,— Ft |
| Főkönyvi könyvelőprogram | 114 800,— Ft |
| COBRA—CONTO programcsomag | 99 000,— Ft |
| IBM—XT kompatibilis számítógép | 169 000,— Ft |
| STAR LC—10 nyomtató | 49 000,— Ft |
| | 317 000,— Ft |

helyett mindezt már
299 000,— Ft-ért is megvásárolhatja!

TUTTI

ELECTROCOOP
KISSZÖVETKEZET

- IBM PC kompatibilis gépek
- HARDVERTELEPÍTÉS SZERVIZ ÉS GARANCIA
- SZOFTVERES TÁMOGATÁS
- RÖVID HATÁRIDŐ

Cím: Bp., Üllői út 81. 1091
Tel.: 334-354



ECOSOFT
Számítástechnikai
Szolgáltató
Kisszövetkezet

IBM PC/AT
kompatibilis számítógép
ár: **229 000 Ft-tól**
TURBO/32
32 bit, 24 MHz
ár: **479 000 Ft-tól.**
Az általunk forgalmazott
eszközök lizingelhetők is.
Tel.: **863-677**



IRODA:
VI., Nagymező u. 51.
TEL.: 325-768

KARÁCSONYI MEGLEPETÉS:

- adatátviteli rendszerek
- MIKROMOD, MODÉM-CSALÁD
- CAD-rendszerek
- nagy teljesítményű perifériák
- magyar VERSACAD
- szuper mini számítógépek

ASY

Telefon:
415-166

Kereskedelmi és szoftveriroda
1061 Bp., Liszt F. tér 10.
Telex: 22-4378

- **ASY—16 szupermikro számítógép** — 12 terminál
- VME busz
- UNIX
- **CRT TERMINÁLOK** VT—52, QUT—102, Siemens 8160
- **BILLENTYŰZETEK**
- **MONITOROK**
- **INTEGRÁLT VÁLLALATI INFORMÁCIÓS RENDSZER**
- UNIX környezetben üzemeltethető.



PERIFÉRIA

Elektronikai
Fejlesztő
és Szolgáltató
Kisszövetkezet
Bp. VII., Peterdy u. 30.
Telefon: 213-588

ajánlata:

- **P—XT:** 140 E Ft-tól + áfa
- **P—AT:** 200 E Ft-tól + áfa
- igény szerinti konfigurációk
- **FX—1000 PRINTER** 90 E Ft + áfa



ez a

védjegy a

megbízható

procontrol

termékeket jelöli

- Profi XT, AT, 386 gépek
- PLATON folyamatirányítók
- PORTAPRINT blokkolóórák
- T80 biztonsági rendszerek
- BCR vonalkód eszközök
- PROCONTROL KISSZÖVETKEZET**
6725 SZEGED,
VERESÁCS U. 28/B
TEL.: 62/21-165, 28-985,
TELEX: 82-726

ERIKA
a legújabb
hordozható irodai
elektronikus írógép.
CSÖNDES,
PONTOS,
MEGBÍZHATÓ,
ára: 25 200 Ft,
áfa-val együtt.

VÁSÁROLJA MEG nálunk:
Bp. VI., Népköztársaság
útja 2.



1146 Bp., AJTÓSI DÜRER
SOR 10.
Levélcíme:
1393 Pf.: 319.
Telefon: 421-974
Telex: 22-6544

december havi kínálata:

a 80 386-os
mikroproceszorra
alapozott, multiuser
üzemű

ACER SYS 32

rendszer, amely
széleskörűen,
változtatható kiépítésben,
alap és alkalmazói
szoftverekkel együtt
kapható.

TRANZISZTORORSZÁGBAN

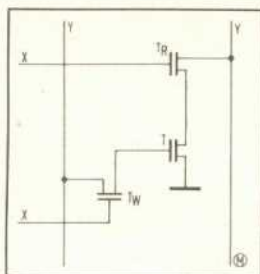
A tárc felépítésének elemzését a sztatikus MOS RAM-ok bemutatásával folytatjuk. A RAM betűszo arra utal, hogy a tárc írható-olvasható, bár ha teljesen korrektek akarunk lenni, meg kell említenünk, hogy a RAM egyébként más fogalomnak, a véletlen hozzáférésű tárcnak a rövidítése. A sztatikus tárolási tulajdonság azt jelenti, hogy a tárc a benne lévő információt annak átírásáig vagy a tápfeszültség kikapcsolásáig megőrzi.

A MOS tranzisztorok előnyei a bipoláris tranzisztorokkal szemben (kisebb helyigény, kisebb tápáramfelvétel, egyszerűbb gyártástechnológia) különösen jól kihasználhatók a nagy kapacitású tárc gyártásánál. Például a jelenleg már általánosan alkalmazott 64 k x 1 bites tárc előállítása bipoláris technológiával elfogadható áron lehetetlen lenne.

A sztatikus MOS RAM-ok működésének megértéséhez vizsgáljuk meg az 5. ábrán látható, 6 tranzisztorral felépített cellát! A tároló kétállapotú (bistabil), két invertezőből ($T_1 - T_2$, valamint $T_3 - T_4$) áll. A T_5 és T_6 tranzisztoros kapcsolók teremtnek kapcsolatot a B és \bar{B} vezetékkel; a kapcsolókat a W szövezetékre adott jellel lehet zárni és megszakitani.

Állandósult állapotban, amikor írás-olvasás nincs, a W vezeték feszültsége a T_5 és a T_6 tranzisztorokat lezárja. Ilyenkor a kétállapotú tárc megtartja eredeti állapotát.

Olvashoz a T_5 és a T_6 tranzisztorokat a W vezetékre kapcsolt jellel nyitni kell. Mivel a T_2 és a T_4 nyelőfeszültsége különböző,



6. ábra

a B és a \bar{B} vezetékeken különböző áramok alakulnak ki. Ezeket a „vezetékek végén” lévő egy-egy olvasóerősítő érzékeli, így az erősítők kimenetén a tárolt információra jellemző jelek keletkeznek.

Íráskor a T_5 és a T_6 tranzisztorokat szintén nyitni kell. A tárolandó információknak megfelelő jeleket a kis impedanciájú kimenettel rendelkező íróerősítők a B és a \bar{B} vezetékeken a bistabilhoz juttatják, rákényszerítve az új állapotot. Természetesen az olvasóerősítők ilyenkor áramot detektálnak, de — mivel írni és olvasni egyszerre értelmetlen — az olvasóerősítő kimeneti jeleit ilyenkor nem vesszük figyelembe.

A bemutatott cellából a már ismert módon mátrixot szervezhetünk. Egy cella kiválasztása a hozzá tartozó W és B — \bar{B} vezeték vezérlésével érhető el: az X vezetéknek a W, az Y vezetéknek a B — \bar{B} felletethető meg, vagy fordítva. A két író- és

két olvasócsatlakozás megfelelő kapcsolástechnikával egyetlen ki-, illetve bemenetű egyesíthető.

A sztatikus tárolási módnak kétségtelenek az előnyei, de két hátránya is van. A tápáramfelvétel viszonylag nagy, ami egyébként CMOS technológiával csökkenthető; a bistabilt alkotó inverterek komplexementer tranzisztorokból épülnek fel, de mivel a kettő közül az egyik zárva van, az inverter fogyasztása igen kicsi. A cella továbbá nagy helyet foglal a lapkán, ami nem teszi lehetővé egy tokban nagy tárcapacitás megvalósítását.

Állandósult állapotban a tápáramfelvétel úgy csökkenthető, hogy a terhelőtranzisztorok (munkaellenállás-tranzisztorok) kapuelektrodáit külön vezetékkel táplálva, kisebb nyelőáramot állítunk be. Az eredeti áramokat csupán az írás és olvasás közben, tehát a működési idő kis százalékában kell biztosítani.

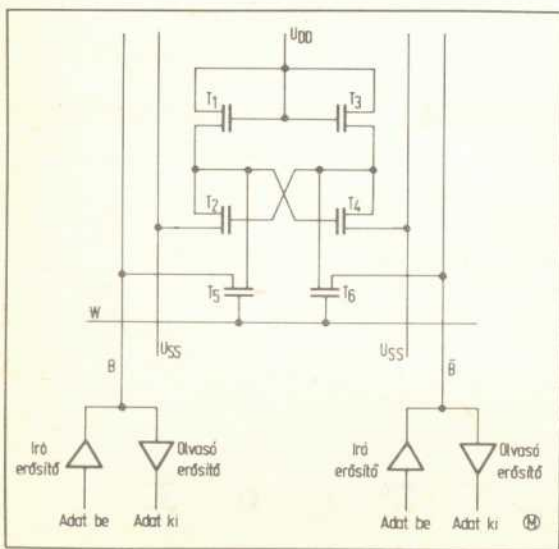
Állítsuk be a kapufeszültséget úgy, hogy az állandósult állapotban a terhelőtranzisztorok teljesen lezárta legyenek. Ekkor a tranzisztorok el is hagyhatók, és ezzel a megoldással tulajdonképpen már el is jutottunk az ún. dinamikus tárc szerkezeti felépítéséhez.

Dinamikus MOS RAM-ok

Dinamikus tárcnak nevezzük azokat az eszközöket, amelyek a tárolt információt külső beavatkozás nélkül csak meghatározott időtartamig, néhány milliszekundumig képesek megőrizni. Hogy a tárc tartalma el ne vesszen, az említett idő eltelte előtt a tárc tartalmát ki kell olvasni és újra vissza kell írni. Ennek a folyamatnak a közismert elnevezése a frissítés. A teljesség kedvéért megjegyezzük, hogy ma már sok olyan típus van forgalomban, amely a frissítő áramköröket is tartalmazza, tehát a frissítésről nem a felhasználónak kell gondoskodnia. Neki az eszköz sztatikus tárcként látható, erre utal elnevezése is: kvázisztatikus RAM.

Dinamikus tárcellaként általában kondenzátorokat alkalmaznak. A megoldás a MOS tárcnál kézenfekvő: a kondenzátorok az áramkörben megtalálhatók (a C_{GS} , esetleg a C_{GD} tranzisztor-elektroda kapacitások), és a kisütési időállandók a MOS tranzisztorok nagy ellenállásértékei miatt — a kis kapacitások ellenére — éppen eléggé nagyok.

A dinamikus tárcella létrehozásához és működésének vizsgálatához induljunk ki az 5. ábrán látható sztatikus tárcellából! Ha nem kívánjuk meg a sztatikus tárolási tulajdonságot, a T_1 és a T_3 terhelőtranzisztorok elhagyhatók. Mivel a T_2 és a T_4 — a bistabil multivibrátor elvéből következően — ellenkező állapotban vannak — az egyik nyitott, a másik zárt —, az egyik az infor-



5. ábra

áció elvészésének veszélye nélkül el is hagyható. Ekképpen háromtranzistoros áramkörhöz jutunk, amelyben egy-egy tranzisztor látja el a tárolást, a tárolótranzisztorba való beírás és a tárolótranzisztoroknak az összekapcsolását az olvasóvezetékkel (ezeknek a jelölése a 6. ábrán rendelt: T , T_W , T_A). A létrehozott áramkörök külféleképpen szervezhetők mátrixba a cellakijelölés és az írás-olvasás (frissítés) szempontjából.

A cellát a mátrixba 2 darab X és 2 darab Y vezeték köti be. A vezetékek szerepének megértését a 7. ábra segíti.

Írások az „Írásengedélyezés” vezetékre adott impulzus kinyitja a T_W tranzisztor, így a C_{G5} tárolókapacitás az „Íróvezeték”-en keresztül a tárolandó információnak megfelelően feltöltődhet.

Olvasás előtt az „Olvasóvezeték”-et rövid időre pozitív feszültségre kell kapcsolni, aminek hatására a C_R vezetékcapacitás feltöltődik. Ezután az „Olvasásengedélyezés” vezetékre kapcsolt impulzussal a T_R tranzisztor vezetőt tesszük; ekkor a T tárolótranzisztor és az olvasóvezeték között létrejön a kapcsolat. Ha a C_{G5} kapacitás töltése révén a kapu pozitív feszültségű, a T tárolótranzisztor vezetni fog, és az olvasóvezeték C_R kapacitása rajta keresztül küld. Ha a kapufeszültség 0 V körüli, T zárva van, és az olvasóvezeték kapacitása eredeti állapotában (pozitívrá töltve) marad. Az olvasóvezeték „végén” lévő olvasóerősítő — mintegy „megmérve” a vezeték feszültségét — állítja elő a tárolt információra jellemző kimeneti jelet.

A három tranzisztorból felépített tárolócella tovább egyszerűsíthető, ha a T_W és a T_R tranzisztor helyett egyetlen tranzisztor (T_G) alkalmazunk az írás és az olvasás engedélyezésére, továbbá elhagyjuk a T tárolótranzisztor is, és a tárolást egy kondenzátorral oldjuk meg. Az írás és olvasás egyetlen tranzisztorral való vezérlése egyben a vezetékek számának csökkentését is eredményezi; egy cella kiválasztására egy-egy X és Y vezeték lesz elegendő. A cella felépítését és kapcsolatát a vezetékekkel a 8. ábra szemlélteti.

Beíráshoz a „Szókielölés” (X) vezeték kell vezérelni. Ekkor a T_G tranzisztor kinyit, és a C_1 tárolókapacitás a „Bitkielölés” vezetéken lévő feszültségnek — ezt az íróerősítő szolgáltatja — megfelelően töltődik fel.

Az olvasás előtt a C_R bitvezeték-kapacitást pozitív feszültségre kell feltölteni. Ekkor a K_1 zárt és a K_2 nyitott állapotban van. A pozitív feszültség akkora legyen, hogy az I_2 inverter munkapontját az aktív szakasz közepére, az $U_{ic} - U_{i1}$ jelleggörbe állapotváltozást leíró „meredek” részére állítsa be.

Ezután K_1 -et nyitjuk és K_2 -t zárjuk, azaz a két invertert flip-floppá kapcsoljuk össze. Ha most a T_G tranzisztor a szókielölő vezetékre adott jellel kinyitjuk, a C_T tárolókondenzátor — töltöttségi állapotának megfelelően — megváltoztatja a C_R bitvezeték-kapacitás feszültségét, és így a flip-floppot valamelyik határozott állapotába bilitenti. A cella a háromtranzisztorosnál lényegesen egyszerűbb felépítésű, de amíg annál a kiolvasás a tárolt információt nem

törli, az egytranzisztoros elemnél a kiolvasás törlődik.

Az ismertetett tárolócella n -csatornás szilíciumkapu technológiájú kivételben a jelenleg már közhasználatúnak tekinthető közepes-nagy kapacitású dinamikus RAM-ok alapeleme ($16 \text{ k} \times 1 \text{ bit}$, $16 \text{ k} \times 4 \text{ bit}$, $64 \text{ k} \times 1 \text{ bit}$). Egyszerű felépítése miatt bonyolultabbak a kiszolgáló áramkörei — például egy $16 \text{ k} \times 1$ bites RAM-ban 128×128 -as mátrixot és így 128 szókielölő áramkört, továbbá 128-128 író- és olvasóerősítőt kell alkalmazni —, de a ma már elérhető nagy alkatrészűsűség lehetővé tette, hogy ezeket a tárolóchipen helyezték el. A jelenleg forgalomban lévő áramkörök binárisan címezhetők, a hozzáférési idő 100-200 ns körül van, a frissítéshez szükséges egységeket részben vagy egészben tartalmazzák és TTL kompatibilisek.

A fixtárakról általában

A fixtárak olyan eszközök, amelyeknek tartalma csak olvasható, illetve a típusok egy részénél megváltoztatható ugyan, de nem üzemszerűen és csak hosszabb idő alatt. (Az angol Read Only Memory — csak olvasható tár — kifejezésből kialakult ROM betűszó a szakmában általánosan használt megnevezésük.) Kétféle típusuk az újra nem programozható és az újraprogramozható ROM.

Az újra nem programozható táruk információtartalma kétféle módon hozható létre. Az egyik megoldásnál a tartalmat a gyártáskor helyezik a táriba. Ez azt igényli, hogy a felhasználók a szükséges információt

ót a gyártó cégnek átadják, aminek alapján a „készfegyártás” elvégezhető. Mivel a felhasználók az információt általában nem szívesen adják ki, továbbá az említett eljárás hosszadalmas és csak nagy sorozatú gyártáskor kifizetődő, többnyire a beírás másik formáját használják, amikor is a felhasználó írja be az információt az „üres” táriba.

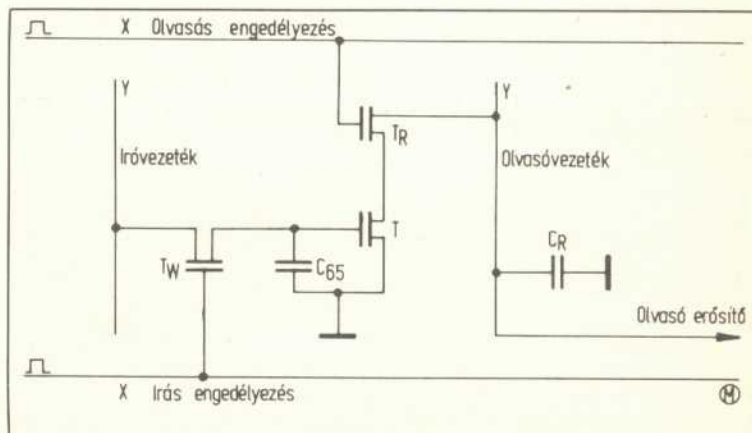
Az újraprogramozható táruk programozását általában a felhasználók készítik el saját igényeik szerint. A tár csak olvasható ugyan, de megfelelő — elég hosszadalmas — eljárással tartalma megváltoztatható. A régi tartalmat ultraibolya vagy röntgensugárzással 10-30 perc alatt lehet törölni, az új tartalmat beírni a módszertől és a tár kapacitásától függően néhány perc alatt.

A hardvertechnikában jártasabbaknak említésre méltó, hogy a ROM-okat nem csak tárolásra (például egy BASIC interpreter „beépítése” egy HC-be), hanem annál sokkal szélesebb körben lehet alkalmazni. Ha arra gondolunk, hogy egy 8 kibit bemenetes és 8 tartalombit kimenetes ROM egyes rekeszeinek címzésénél — bármi is végezzük ezt — mindig ugyanaz az információ jelenik meg a kimeneteken, világossá válik, hogy ez az áramkör nem más, mint egy olyan kombinációs logikai hálózat, amelynek 8 bemenete és 8 kimenete van, azaz nyolc 8 változós függvényt valósít meg.

MOS ROM-ok

Az újra nem programozható MOS ROM-ok cellája igen egyszerű: egy MOS tranzisz-

7. ábra



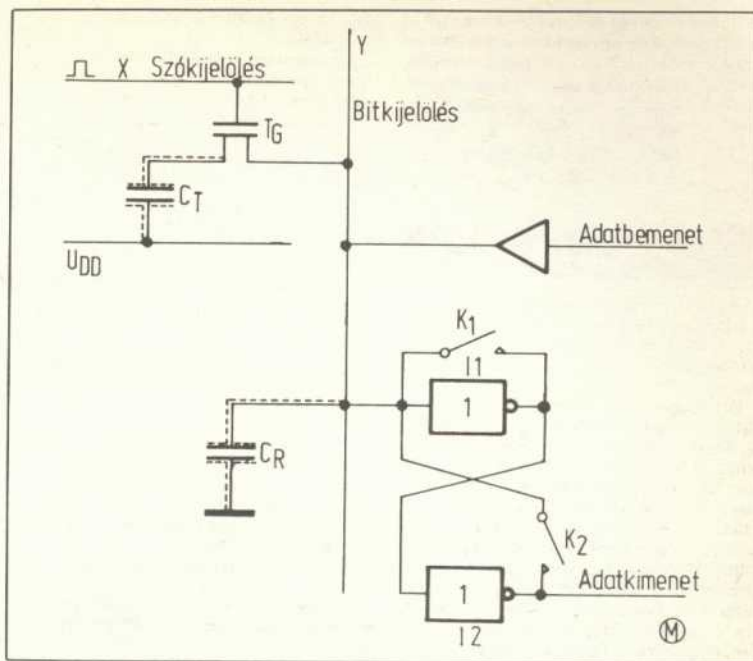
T R A N Z I S Z

tor, amely a információbit értékétől függően vagy létrehoz vagy nem hoz létre kapcsolatot a szó- és olvasóvezeték között. Egy négy cellát tartalmazó mátrixrészlet a 9. ábrán látható.

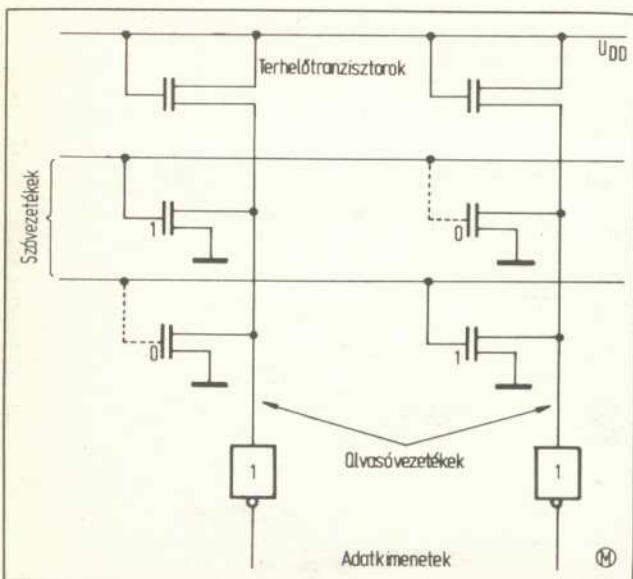
Azokon a helyeken, ahol a „Szóvezeték”-hez a kapu csatlakozik, a cella 1-et tárol, a többi helyen (az ábrán a szaggatott kapubekötés) a tárolt bit értéke 0. A „Szóvezeték” vezérlésekor a 1-et tároló tranzisztorok kinyitnak, és az „Olvasóvezeték” potenciálját — szakzsargonnal kifejezve — lehúzzák. Az adatkimeneteken az invertálás révén H szint jelenik meg, az 1-es kiolvasást jelezve. A cellák létrehozása technológiailag egyszerű: a 0-át tároló cellák tranzisztorainak kapu-oxidrétegét lényegesen vastagabbra készítik, így a kapufeszültség a nyelőforrás-csatornát nem képes vezetévé tenni, nyelőáram tehát ezeken a tranzisztorokon nem folyik. A jelenség egyszerűen kifejezve olyan, mintha a tranzisztor ott sem lenne.

Az újraprogramozható tárcák megvalósításához általában a FAMOS tranzisztor (Floating gate Avalanche injection MOS — lebegő vezérlőelektródájú lavinafeltöltésű MOS) alkalmazzák. A FAMOS kapuelektrodájának kivezetése nincs, körös-körül nagy tisztaságú, tehát nagyellenállású szilíciumoxid veszi körül, amelynek vastagsága az n-típusú csatorna felé mintegy 100 nm. A kapu tehát meghatározatlan feszültségen van, „lebeg”.

Ha nagy negatív (–40...–50 V) feszültséget kapcsolunk a nyelő és a kapu közé, akkor e feszültség nagy része a lezárt



8. ábra



9. ábra

kapu-nyelő p-n átmenetre esik. Az átmenetnél lavinaeffektus jön létre, és az elektronok a vékony szilíciumoxid rétegen keresztül néhány ms alatt feltöltik a kaput. Ez a töltés elegendő ahhoz, hogy a kapuelektrodán –10...–15 V küszöbfeszültséget hozzon létre; ugyanakkor azokon a helyeken, ahol nem lépett fel a lavinaeffektus, a küszöbfeszültség max. –1...–1,2 V lesz. Az így „programozott” tranzisztorok különbözőképpen fognak viselkedni az áramkörben: az egyik nyitott (vezető), a másik zárt lesz. A minden oldalról szigetelt kaputöltése igen hosszú ideig megmarad: a kisütési áram 10^{-40} A nagyságrend körül! A küszöbfeszültség 10 százalékos megváltozásához szükséges időt közelítő számításkal 80-100 évre becsülik.

Az információ törléséhez sugárzást kell alkalmazni. Ha a tokon van kvarcüveg ablak, ultrabolya fény használható, ha nincs, a törlés röntgensugárzással végezhető el. A szükséges sugárzási intenzitást és időt a gyártó cégek megadják.

Az egy tokban megvalósítható MOS ROM-ok kapacitása jelenleg 64–256 kbit, a hozzáférési idő 100-200 ns.

NAGY IMRE

Sorozatunkban szokatlan az új hardver- és szoftvertermékeket ismertetjük, amelyek várhatóan általánosan elterjednek, és meghatározó szerepük lesz a fejlődés irányainak kialakításában.

Merre tart a világ?

MONITOROK

SONY

A monitorgyártásban ugrásszerű változás nem következett be. A képcsövek laposodnak, felbontásuk nő, és egyre gazdagabb a színválasztékuk is.

A GDM-1953 19" képméretű, nagy felbontású, CAD/CAM célra tervezett monitor. Clem Shemanski, a cég eladási igazgatója szerint az általuk gyártott és ebben a monitorban is használt Trinitron[®] cső miatt a pontok élessége és fényessége jobb, mint a hasonló felbontású más termékeké. Előnye még a készüléknek, hogy lapos a képcsöve, a képszélek torzításmentesek, ugyanúgy torzításmentes a rajzvízszázadási képessége, valamint a fényvisszaverődést is csökkentették. Az 1. képen látható monitor adatait az 1. táblázat tartalmazza.

A CPD-1303 Multiscan monitor (2. kép, 2. táblázat) a legnagyobb felbontású a hasonló méretűek között. Ebben szintén Trinitron[®] csövet építettek. Az IBM PC/XT és PS/2 sorozatához, valamint az Apple Macintosh II típushoz ajánlják. A különlegesen nagy felbontóképessége miatt, a cég igazgatójának véleménye szerint, nemcsak a mai színes monitorokkal szemben támasztott követelményeknek felel meg, hanem a jövőben is kielégíti ezeket. Eleget tesz a CGA módtól a VGA módig az összes elterjedt grafikus felbontási előírásnak. Ráadásul szinte tökéletesen, ami nem mondható el a többi gyártmányról, mert van, amelyik az egyik, van,

amelyik a másik szabványhoz idomul jól, míg másokhoz kevésbé. A bemenőjel egyaránt lehet analóg RGB vagy digitális jel. Az előző monitornál felsorolt előnyök, az azonos típusú képcső használata miatt, a CPD-1303 Multiscan is jellemzik.

HITACHI

A cég termékeivel a számítástechnikai hardvergyártás majdnem teljes választékát kínálja, és ezek egyaránt ismertek, akár integrált áramkőről — például a 68 számkódú mikroprocesszorokról, statikus és dinamikus RAM-okról, a különféle ROM-okról stb. —, akár részegységről — különféle háttértárolókról — vagy önálló berendezésekről — plotterekről, rajzdigitalizálókról — legyen szó. Szinte természetes, hogy a monitorgyártás is „erőssége” a Hitachinak.

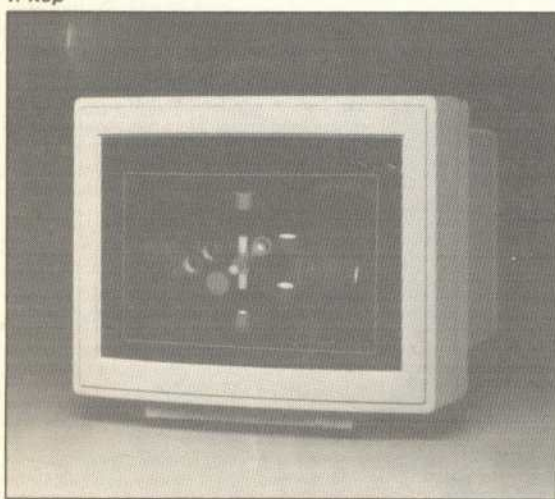
A HM- (High Tech Monitor) sorozatnál, ahogy az előző cégnél is, néhány lényeges és azonos megoldást alkalmaztak.

Ilyen a DDC (Dynamic Digital Convergence modul) és az Inline-tube. A DDC egy beépített célszámítógéppel teszi lehetővé a konvergencia be szabályozását az összes előforduló frekvencián. A számítógépes irányítás gondoskodik egyrészt arról, hogy a beállított értékhez rendelje a többi paramétert, másrészt ezeket az értékeket tartja. Ezzel a megoldással mintegy megkétszerezte a felbontóképességet. Az Inline-tube nagyobb fényerőt ad, ezáltal lehetővé válik az energiatakarékosabb megoldás. A kábelezése különleges, emiatt a monitort dönteni, forgatni lehet anélkül, hogy az ilyen használat a kábelnél problémát okozna.

A sorozat rendkívül nagy képméretű tagja a 4625 (3. kép), amelynek mérete 25". Az 5219 sávszélessége egészen nagy, a 4219 pontmérete pedig igen kicsi. A 4619 típus nagyon hasonlít az előzőre, csak a pontmérete nagyobb, viszont a bemenőjelének szintje alacsonyabb. A négy típus adatait a 3. táblázat foglalja össze.

— SE —

1. kép



SONY GDM-1953 Monitor

2. kép



SONY CPD-1303 Multiscan Monitor



Hitachi monitor

3. kép

1. táblázat

| Megnevezés | Érték |
|-----------------------|------------------------------------|
| Képterület | 343 × 274 mm |
| Pontméret | 0,31 mm |
| Felbontás | 1280(H) × 1024(V) pont |
| Horizontális szinkron | 64 |
| Sávszélesség | 100 MHz |
| Vertikális szinkron | 60 Hz |
| Befoglaló méret | 480 × 446 × 535 mm |
| Tápadatok | 100–120 V, 50/60 Hz, 2,6 A max. |

2. táblázat

| Megnevezés | Érték |
|-------------------------------|--------------------|
| Képterület | 274 × 207 mm |
| Pontméret | 0,26 mm |
| Felbontás | 1024(H) × 768(V) |
| Horizontális frekvenciasáv | 15,75–36 kHz |
| Vertikális szinkron | 50–100 Hz |
| Befoglaló méret | 370 × 330 × 415 mm |

3. táblázat

| Megnevezés | Érték | | | |
|----------------------------------|------------------|-------------------|-------------|-------------|
| | 4219 | 4619 | 4625 | 5219 |
| Képterület | 340 × 270 | 340 × 270 | 450 × 355 | 341 × 273 |
| Pontméret | 0,31/0,26 | 0,31 | 0,37 | 0,26 |
| Konvergencia | 0,1/0,3 | 0,3/0,5 | 0,5 | 0,3/0,5 |
| Felbontás | 1280 × 1024 | 1280 × 1024 | 1280 × 1024 | 1600 × 1280 |
| Horizontális frekvenciasáv | 61–65 | 61–65 | 62–65 | 78–90 |
| Vertikális frekvenciasáv | 55–65 | 55–65 | 55–65 | 60–75 |
| Horizontális visszafutási idő | 3,5 μs | 3,5 | 4,0 | 3,2 |
| Vertikális visszafutási idő | 550 μs | 550 | 600 | 530 |
| Sávszélesség | 15–100 | 15–100 | 15–100 | 15–100 |
| Névleges esési idő | 5 ns | 5 | 5 | 3 |
| Bemenőjel | analóg | analóg | analóg | analóg |
| Bemenőjelszint | 0,7–1,4 V | 0,5–1,5 | 0,7–1,4 | 0,7–1,2 |
| Külső szinkronsint | TTL | 4,0 | 4,0 | 2,5 |
| Külső szinkronsint szabvány | RS343A | RS343A | RS343A | RS343A |
| Tápadatok | 87–132 V, 2 A | 175–264 V, 3,7 | 47–63 Hz | 5 2,4 |

Segíts magadon!

Ötletek a képernyőtartalom kinyomtatására

Elég régóta használom a Masterfile programot. Igen praktikusnak találom, hogy az adatokat bármely szempontból képes rendezni. Am sokszor nehézségem támadt amiatt, hogy az ilyen módon rendezett adatokat nem láthattam, elemezhetem nyomtatásban, mert nincs Spectrum nyomtató. Nagyon megörültem, amikor lehetőségem nyílt egy Brother M-1109 típusú nyomtató időnkénti használatára. Sajnos azonban a nyomtató „nem érti” a COPY parancsot. Először egy BASIC program segítségével nyomtattam, de ennek a megoldásnak két hibája volt. Az egyik, hogy egy képernyő kinyomtatása hozzávetőlegesen tíz percig tartott, a másik ennél sokkal alapvetőbb. De előbb nézzük a képernyőt!

Mindazoknak, akiknek valamilyen RS232 csatlakozású nyomtatójuk van a Spectrumhoz, gondot jelent a képernyőtartalom kinyomtatása. Ennek oka a számítógép képernyőjének a szervezése. A gépkönyv szerint a képernyő két részből áll:

- képtartalomból (Display File)
kezdőcím: 16384 (4000H)
- szintartalomból (Attributes)
kezdőcím: 22528 (5B00H)

Ez azonban a mi szempontunkból igen kevés információ. Nézzük a részleteket! Mindenki, aki már betöltött képernyőtartalmat a képernyő-memóriába, észrevette, hogy a képernyő három (nyolc betűsoros) részre van osztva.

| | Kezdőcím | | Utolsó cím | | Képernyősor |
|-----------|----------|------|------------|------|-------------|
| | DEC | HEX | DEC | HEX | |
| 1. harmad | 16384 | 4000 | 18431 | 47FF | 0— 7 |
| 2. harmad | 18432 | 4800 | 20479 | 4FFF | 8—15 |
| 3. harmad | 20480 | 5000 | 22527 | 57FF | 16—23 |

Milyen az első karakter a memóriában? Mely memóriacímek és milyen formában tartalmazzák?

| | 128 | 64 | 32 | 16 | 4 | 2 | 1 |
|-------|-----|----|----|----|---|---|---|
| 16384 | . | . | . | . | . | . | . |
| 16640 | . | . | . | . | . | . | . |
| 16896 | . | . | . | . | . | . | . |
| 17152 | . | . | . | . | . | . | . |
| 17408 | . | . | . | . | . | . | . |
| 17664 | . | . | . | . | . | . | . |
| 17920 | . | . | . | . | . | . | . |
| 18176 | . | . | . | . | . | . | . |

Ez azt jelenti, hogy a rajzolat vízszintes vonalakkból alakul ki. Ha a képernyő koordinátahálózatát nézzük, a 16384-es memóriarekesz bal szélső bitje a (0,175) pontnak felel meg.

Mélyedjünk bele a kettes számrendszer és a Spectrum képernyőszervezésének rejtelmeibe. A Spectrum egy memóriarekesze alkalmas egy 0—255 közötti szám tárolására. Hogy miért? A memóriarekesz nyolc bitből áll, és egy bit ugyebár vagy 0 vagy 1 lehet. Tehát a 00000000=0 és az 11111111=255. Egy betűhely, amit a program írásánál egy egységnek vettem — majd kiderül, miért —, nyolc memóriarekesz tartalma, tehát 8 x 8 bit. Visszatérve a kettes számrendszerre: a 0 és 1 számok a 2 hatványainak megfelelően helyiértékekkel rendelkeznek, mint ahogy az alábbi ábrán látható:

$$2^7 \quad 2^6 \quad 2^5 \quad 2^4 \quad 2^3 \quad 2^2 \quad 2^1 \quad 2^0$$

$$128 \quad 64 \quad 32 \quad 16 \quad 8 \quad 4 \quad 2 \quad 1$$

BASIC utasítások segítségével a képernyő pontjai — azaz a képernyőrekeszek bitjei — egyenként címezhetők. A PLOT utasítással a biteket 1-be állíthatjuk, vagyis rajzolhatunk, a POINT uta-

sítás segítségével pedig kiolvashatjuk a képernyőpontnak, tehát a memóriarekesz egy bitjének az állapotát. A képernyő első két sora ilyen módon nem címezhető, mert a képernyő (0,0) pontja alulról a harmadik sor bal alsó sarka, és a POINT, valamint a PLOT utasítás nem ismeri a negatív számokat. Ezért BASIC-ből a 22. és 23. „hibaüzenet” sorok pontjai nem címezhetők. Vagyis ez a BASIC másik hibája.

A nyomtatók típusától függően nyolc vagy kilenc függőlegesen elhelyezett tüvel írunk. Az a nyomtató, amivel dolgozom, bitnyomtató üzemmódban nyolc tüvel nyomtat, ezért a program erre készült. Természetesen nehezíti a dolgot a képernyő harmadokra való osztása is. Nagyobb gond azonban a bitek „átfordítása”, vagyis ahogy az előbb leírtam: a vízszintes vonalakkól felépített képet függőleges vonalakra „átírni”. Hogyan oldottam meg végül a problémát?

Míg a Spectrum memóriájában a jobb szélső pont értéke 1 és a bal szélső ponté 128, a nyomtatóban a legalsó pont az 1 és a legfelső a 128. Tehát nem elég a memóriarekeszek egyszerű PEEK-elése, hanem a kapott számot át kell számolni a kettes számrendszerbe, ami elképzelhetetlenül lassú lenne. Ezért a pontokat grafikusnak kell egyenként kiolvasni a POINT utasítással. Emiatt lassú a BASIC program.

Térjünk vissza a képernyőhöz! Egy karakter elrendezése feljebb látható. A 18176-os memóriarekeszt a karakter báziscímének neveztem el. A képernyő három harmadra van felosztva. Egy harmad nyolc sor, egy sorban 32 karakter van. Azaz egy harmad 256 betűhely. Ezeknek a báziscímei sorban egymást követő számok. Tehát

- az első harmad:
18176, 18178, 18179, ..., 18431
- a második harmad:
20224, 20225, ..., 20479
- a harmadik harmad:
22272, 22273, ..., 22527

Az első képernyőcímnél (ami az első betű teteje) a báziscím $7 \times 256 = 1792$ -vel nagyobb. A harmad első báziscíménél az utolsó 255-tel nagyobb (ennyi betű van a harmadban).

Nézzük tehát a program lényegét!

A program változói:

- IY > CIM a betűhelyen levő nyolc memóriarekeszt számláló rekesz címe
- IY+1 > egy memóriarekesz 8 bitjét számolja
- IY+2 > a sor 32 karakterét számolja
- IY+3 > a képernyőharmad 256 karakterét számolja
- IX > PUFF kinyomtatandó memóriarekesz címe
- HL > a betűhely báziscíme

A fontosabb szubrutinok:

- ESCBIT a nyomtató beállítása minden képernyősor előtt
- START a program kezdete
- SOROK az (IY+2) és az (IY+3) beállítása
- HARM egy harmad kiolvasása és kiírása
- VALT a képernyőharmad váltáskor aktiválódik
- VISSZ a BIT rutinból való visszatérés a főprogramba
- PRINT egy betűhely kiírása
- BIT egy betűhely „átfordítása”
- VEGE kilépés a programból

A szubrutinok címét lásd a táblázatban!

A program csak relatív címeket használ, ezért bárhová betölthető. Hossza 240 bájt. A program elkészítésekor az vezérelt, hogy 256

| | | | |
|-------|------|--------|------|
| BET | 5B80 | BETU | 5B84 |
| BIT | 5BB2 | CIKL | 5B98 |
| CIM | 5BF0 | ESCBIT | 5B1D |
| FOLYT | 5B61 | HAR | 5B54 |
| HARM | 5B4E | LEP | 5BC0 |
| LEPES | 5BC8 | OSZL | 5BBC |
| PLUSZ | 5B7A | PRINT | 5B92 |
| PUFF | 5BE8 | SOROK | 5B46 |
| START | 5B3E | TOV | 5BB0 |
| TOVA | 5BCF | TOVAB | 5BCC |
| VALT | 5B6D | VEGE | 5BE6 |
| VISSZ | 5B89 | | |

bájt nál rövidebb legyen, hogy ha a memóriába máshová nem helyezhető el, akkor a PRINTER PUFFER-ben elférjen. Ezért is betűnkénti és nem soronkénti a kiírás.

TABULÁCIÓ: a POKE (betöltési cím + 12), N segítségével. A KIÍRATÁS MÓDJA: POKE (betöltési cím + 41), M ahol $0 < M < 7$.

PROGRAMBÓL VALÓ KILÉPÉS: SYMBOL SHIFT + bármelyik billentyű lenyomására (kivéve a CAPS SHIFT).

A program leírása:

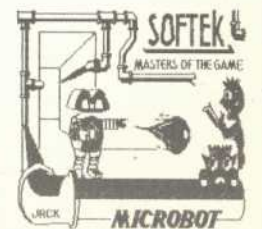
- 60-70 csatornamegnyitás
80-150 TAB-beállítás
120 TAB mértéke
160-190 ESC 1: sorköz nincs
200 ESCBIT rutin végén H-től függő ugrás van, a program során H-ban nem 0 van
210-240 kocsni vissza, soremelés
250-260 TAB-aktiválás
270-360 ESC*: bitnyomató üzem mód
310 nyomtatásimód-beállítás $0 < \text{mód} < 7$
330, 350 a nyomtatandó bitek száma (egy ciklusban 256)
370-400 első végrehajtáskor nem ugrik (így egyszerűbb a bitnyomtatást soronként beállítani)
410 a képernyő első betűhelyének „báziscíme”
420 a címet elteszem a verembe
440-450 a harmad képernyő 8 sor \times 32 betűhely = 256 ciklus kell $> \text{IY} + 3$, 32 betűhely $> \text{IY} + 2$ készen van a 32 betűs sor?
490-520 a HL növelése és elmentése
530-550 készen van a 256 ciklus?
560-570 ha igen, jön a következő harmad
580 ha nem, újra
590 képernyő utolsó cím? Ha igen, VÉGE
600-640 ha nem, megnöveljük a címet 8×256 -tal
650-700 a betűn belüli bitpozíció kijelölése, illetve növelése
720-730 a puffer növelése
750 a betű bitsorának utolsó bitje? Ha nem, újra
760-780 a pufferrekeszek tartalmának kinyomtatása
790-840 a regiszterek tartalmának elmentése, billentyűfigyelő rutin hívása, SYS+billentyű megnyomására kilépés
850-970 BASIC-be
990 IY számláló beállítása a 8 függőleges bit kiolvasásához
1010-1020 a HL betöltése a veremből
1030 az IY+1-be már előzőleg betöltöttük azt, hogy a regiszter melyik bitje van soron. Legyen ez a C bit
1050-1060 a C bit 0. pozícióba állítása
1070 a 0. kivételével a bitek törlése
1080-1120 a bit visszaléptetése
1130-1140 hozzáadjuk a pufferrekeszhez
1150-1170 kész a nyolc függőleges bit?
1190-1200 ha nem, vonjunk ki HL-ből 256-ot, mert alulról meggyünk felfelé
1240-1320 a nyolc memóriarekesz a nyolc bitpozíciónak
1320-1350 a négy számlálórekesz
A programnak van egy még kiküszöbölhetlen, de már felfedezett hibája: csak egy képernyőtartalom nyomtatható ki, ezután a nyomtató ki kell kapcsolni, mert különben a következő ábra eleje mindenféle „szemét” lesz. A nyomtató ki-, majd bekapcsolása után a következő ábra baj nélkül nyomtatható.
Remélem, ha valaki más típusú nyomtatója miatt a programot egy az egyben nem is tudja használni, azért ötletet kap saját programjának megírásához.

DR. GRESZ MIKLÓS

| | | | |
|-----|---------------------------|--------|--|
| 10 | HARDCOPY SPECTRUMRA | | |
| 20 | BROTHER M-1109 nyomtatóra | | |
| 30 | Gresz Miklós 1987 | | |
| 40 | | | |
| 50 | ORG 23300 | | |
| 60 | LD A, 3 | | |
| 70 | CALL #1601 | | |
| 80 | LD A, 27 | | |
| 90 | RST 16 | | |
| 100 | LD A, 68 | | |
| 110 | RST 16 | | |
| 120 | LD A, 1 | | |
| 130 | RST 16 | | |
| 140 | LD A, 0 | | |
| 150 | RST 16 | | |
| 160 | LD A, 27 | | |
| 170 | RST 16 | | |
| 180 | LD A, 49 | | |
| 190 | RST 16 | | |
| 200 | LD H, 0 | | |
| 210 | LD A, 13 | ESCBIT | |
| 220 | RST 16 | | |
| 230 | LD A, 10 | | |
| 240 | RST 16 | | |
| 250 | LD A, 9 | | |
| 260 | RST 16 | | |
| 270 | LD A, 27 | | |
| 280 | RST 16 | | |
| 290 | LD A, 42 | | |
| 300 | RST 16 | | |
| 310 | LD A, 0 | | |
| 320 | RST 16 | | |
| 330 | LD A, 0 | | |
| 340 | RST 16 | | |
| 350 | LD A, 1 | | |
| 360 | RST 16 | | |
| 370 | LD (IY+2), 32 | | |
| 380 | LD A, 0 | | |
| 390 | CP H | | |
| 400 | JR NZ, FOLYT | | |

| | | | |
|-----|-------|------|------------|
| 410 | START | LD | HL, 18176 |
| 420 | | PUSH | HL |
| 430 | | LD | IY, CIM |
| 440 | SOROK | LD | (IY+3), 0 |
| 450 | | LD | (IY+2), 32 |
| 460 | HARM | LD | IX, PUFF |
| 470 | | JR | BET |
| 480 | HAR | DEC | (IY+3) |
| 490 | | DEC | (IY+2) |
| 500 | | LD | A, (IY+2) |
| 510 | | CP | 0 |
| 520 | | JR | Z, ESCBIT |
| 530 | FOLYT | POP | HL |
| 540 | | INC | HL |
| 550 | | PUSH | HL |
| 560 | | LD | A, (IY+3) |
| 570 | | CP | 0 |
| 580 | | JR | Z, VALT |
| 590 | | JR | HARM |
| 600 | VALT | POP | HL |
| 610 | | PUSH | HL |
| 620 | | LD | A, #58 |
| 630 | | CP | H |
| 640 | | JR | Z, VEGE |
| 650 | | POP | HL |
| 660 | | LD | DE, 256 |
| 670 | | LD | B, 7 |
| 680 | PLUSZ | ADD | HL, DE |
| 690 | | DJNZ | PLUSZ |
| 700 | | PUSH | HL |
| 710 | | JR | SOROK |
| 720 | BET | LD | (IY+1), 0 |
| 730 | BETU | INC | (IY+1) |
| 740 | | JR | BIT |
| 750 | VISSZ | INC | IX |
| 760 | | LD | A, 7 |
| 770 | | CP | (IY+1) |
| 780 | | JR | NC, BETU |
| 790 | PRINT | LD | IX, PUFF |
| 800 | | LD | B, 8 |
| 810 | CIKL | LD | A, (IX) |
| 820 | | RST | 16 |
| 830 | | INC | IX |
| 840 | | DJNZ | CIKL |
| 850 | | PUSH | HL |
| 860 | | PUSH | BC |
| 870 | | PUSH | DE |
| 880 | | CALL | 654 |
| 890 | | LD | A, D |
| 900 | | CP | 24 |
| 910 | | POP | DE |
| 920 | | POP | BC |
| 930 | | POP | HL |
| 940 | | JR | NZ, TOV |
| 950 | | POP | HL |
| 960 | | RET | |
| 970 | TOV | | |
| 980 | | JR | HAR |

| | | | |
|------|-------|------|-----------|
| 990 | BIT | LD | (IY), 9 |
| 1000 | | LD | (IX), 0 |
| 1010 | | POP | HL |
| 1020 | | PUSH | HL |
| 1030 | OSZL | LD | B, (IY+1) |
| 1040 | | LD | A, (HL) |
| 1050 | LEP | RLCA | |
| 1060 | | DJNZ | LEP |
| 1070 | | AND | 01 |
| 1080 | | LD | B, (IY) |
| 1090 | LEPES | DJNZ | TOVAB |
| 1100 | | JR | TOVA |
| 1110 | TOVAB | RRCA | |
| 1120 | | JR | LEPES |
| 1130 | | ADD | A, (IX) |
| 1140 | TOVA | LD | (IX), A |
| 1150 | | DEC | (IY) |
| 1160 | | LD | A, (IY) |
| 1170 | | CP | 2 |
| 1180 | | JR | C, VISSZ |
| 1190 | | LD | DE, 256 |
| 1200 | | SBC | HL, DE |
| 1210 | | JR | OSZL |
| 1220 | VEGE | POP | HL |
| 1230 | | RET | |
| 1240 | PUFF | NOP | |
| 1250 | | NOP | |
| 1260 | | NOP | |
| 1270 | | NOP | |
| 1280 | | NOP | |
| 1290 | | NOP | |
| 1300 | | NOP | |
| 1310 | | NOP | |
| 1320 | CIM | NOP | |
| 1330 | | NOP | |
| 1340 | | NOP | |
| 1350 | | NOP | |
| 1360 | | END | |



KLUBSZERVEZÉS ÉS OKTATÁS

Cikkem csak az átalakítás néhány jelével foglalkozik, melyeket a két országban tapasztaltam.

A szocialista országokban jó néhány év késéssel vetők tudomásul, hogy a számítógépek használatának tömeges elterjesztése a mikroszámítógépek tömeges elterjesztésével jár együtt. Mi ennek felismerésében élen járunk, és ez elsősorban a magánipornak nevezett, alulról jövő kezdeményezés volt köszönhető. Ebben az úttörő szerepet, ahogy a világ számos más országában is, a klubok — nálunk az első és sokáig egyetlen HCC — töltötték be.

A címben említett két országban is az első lépést a klubok tették meg, miután tanulmányozták a miénket, de ott más úton jártak. A klubokat felülről szervezték, vagyis az állam hozta létre azokat, adta a felszerelést és a működését is rá hárul.

Bulgária az előtte álló úton már jóval előbb tart, mint a Szovjetunió. Burgaszban létesült talán a világon az első olyan klubépület, amelyet eleve erre a célra építettek, és amelynek egyetlen feladata is ez. A háromszintes épületben összesen 31 különböző típusú számítógépet — bolgár gyártású Apple II és XT kompatibilis gépeket — helyeztek el. Ezek nagy többsége díjmentesen használható bárki számára. Minden gépnek van háttértárolója (hajlékonylemez-egység), néhánynak kis kapacitású merevlemezegysége), nyomtatójuk viszont jelenleg még nincs.

A klubok az üzemeltetési és továbbfejlesztési költségeket (a személyzet bérét, az energiaköltségeket kivéve) szoftvergyártással teremtik elő. Így például a Szófia Városi Komputerklub animációs filmet készít a helybeli televízió részére. Vállalatok felkérésére is készítenek szoftvereket. Ilyen megrendelésre ugyanez a klub készített szállodai nyilvántartó rendszert a szófia szállodáknak.

A gépek elterjesztése érdekében Szófiában PC-szalont nyitottak. A miénkhez hasonlóan a hazai gyártmányokat mutatják be és árúsítják. Ami érdekes, magánszemély is vásárolhatnak a szalontban. (A forgalom valószínűleg nem lehet nagy, mert háromszor is jártam arra, de a boltban az eladókön kívül mást nem láttam.)

A gépváltásnál az Apple II kompatibilis Pravez 8 és az XT kompatibilis Pravez 16. Erdekességként, azért, hogy a kettőt összehasonlítsam, megtudtam három konfigurációt árárt. A Pravez 16 256 k tárral, két hajlékonylemez-egységgel, egyszínű monitorral és az ezeket kiszolgáló illesztőkártyákkal 7800 leva (1 leva = 16 forint), ugyanez 10 Mбайтos merevlemezegységgel 12 500 leva, 20 Mбайтossal pedig 14 500 leva. Kapható még az Epson cég RX80 típusával kompatibilis nyomtató és számos szoftvertermek. Ez utóbbiak teljesen kezdőknek BASIC-anyagokból és professzionális felhasználóknak készítettékből (például PROLOG-ból) állnak. Áraikat nem tudom.

A Szovjetunióban nem országos szervek központi szervezésében jöttek létre az első klubok, hanem kutatóintézetek kísérleti vállalkozásaiéik. Az első két, általában megismert klub a szibériai Novoszibirszkben, illetve a hozzá tartozó Akagyemgorodokban jött létre. Mindkét klub maga fejleszt és épít gépeket. Az elkészített gépeket, amelyekhez alkatrészt, eszközöket,

helyiségeket a klub ad, jelképes összegért a tagok használatába adják. A gépeket építő szakember klubtagok az építési részben munkaköri kötelezőségeket végzik. A klubokat a később alakítandó nagyszámú klub modelleként vezélik.

Velünk egyetemben, náluk a klubok igyekeznek együttműködni, ezért létrehozhat egy szervezet az első mintegy ötven klub részvételével. (Nem tudom megérteni, hogy nálunk sokszori felhívás, az együttműködés előnyeinek ismertetése ellenére miért nem sikerült elérni az együttműködést a HCC és más klubok között. Ez annál is érthetősebb, mert a HCC és több külföldi — most már szovjet is — klub között élő kapcsolat áll fenn, ami elsősorban nekünk előnyös, és ezt a velünk együttműködni szándékozó hazai klubok is hasznosítanák.) Törekednek a nemzetközi együttműködés kialakítására is. Csatlakoztak felhívásunkhoz, ami a klubok nemzetközi szervezetének felállítására irányul.

A rendelkezésre álló aktrészválaszték alapján az egyik gépük 8080 típusú mikroprocesszorral épülő CP/M gép. A 64 k RAM táru, aránylag jó grafikai (320 × 256 pont, fekete-fehér) gépet hajlékonylemez-egységgel, monitorral együtt 1000 rubelért (1 rubel = 18 forint) kívánják forgalomba hozni. A másik két géptípus Spectrum 48 k kompatibilis. Mindkét gép közös tulajdonsága, hogy az ULA helyett TTL IC-eket alkalmaztak. Így az egyetlen import alkatrész a mikroprocesszor. A két változat között a legfontosabb eltérés a billentyűzet, ami az egyiknél a megszokott gumimembrán, a másiknál pedig normál elektromechanikus billentyűt használunk. Mindkét változat érdekes tulajdonsága a képernyőmemória lapozhatósága.

A két klubbal létrejött megállapodásban vállaltuk a kölcsönös segítségnyújtást, ami a mi részünkről — emlékezve az amerikai kluboktól kapott hasonló segítségre — elsősorban alkatrészek küldéséből áll.

Megtudtuk, hogy más gépekre, például C64-re alapozott klubok is léteznek. Ezekkel is igyekszünk kapcsolatba lépni.

A bolgár iskolahelyzetről már több cikkben is beszámoltunk. Változás elsősorban az XT kompatibilis gépek terjedése. A Szovjetunióban a Novoszibirszk-Akagyemgorodokban székelő Jersov akadémikus által vezetett kutatóintézetet bízták meg mind a módszerek kidolgozásával, mind azok gyakorlati bevezetésével. Ez igen nagy feladat, hiszen 140 000(!) iskoláról van szó.

A gépek szerint két szintet hoznak létre: az alsót a tömegesen alkalmazzák, a magasabbat a terminálként is kapcsolható gépek alkotják. Tudomás szerint az alsóra két típus a jelölt. Az egyik a japán Yamaha cég MSX2 szabványú gépe, lemezegységgel, monokrom monitorral, a másik pedig a francia Thomson cég MOSE gépe, 6809 típusú mikroprocesszorral. Az előzőhöz (és a Tandy CoCo, illetve a Dragon géphez) hasonló BASIC-vel, sörös és párhuzamos csatornával, két analóg bemenettel. A magasabb szintet vagy a saját gyártmányú Apple II kompatibilis Agat típusal, vagy valamilyen XT kompatibilis géppel fogják betölteni.

Az oktatási célról, az oktatókandó anyagról közelebbi információk nincs.

SIMONYI

ADOM A MAGYARÁZATOT!

Lapunk 1988/8. számában kérdeztük, hogy miért ad hibás eredményt az ott ismertetett egyenletrendszer-megoldó program.

A program írja feljelen kívül hagyta azt a tényt, hogy a matematika korlátlan számtartománnyal dolgozik, a számítógépek azonban nem. Ebből két hiba adódik.

1. Minden BASIC-változtnál létezik egy, a változattól függő legkisebb pozitív szám (M1), amelynél kisebb abszolút értékű már zérusnak tart a fordító. Ez az oka annak, hogy $0 < ABS(D) < M1$ esetén az egyenletrendszerre a program — tévesen — nem ad megoldást.

Bővítsük a Cramer-szabály egyenleteit $1/M1$ értékkel. Például: $y = \frac{DY}{D} - \frac{DY/M1}{D/M1}$

Ha egyenletünk nevezőjére igaz, hogy $M1 < ABS(D/M1) < 1$, akkor adódnak téves eredmények. Nem kapunk azonban ilyenkor sem téves eredményt, ha a bővített egyenletkel dolgozunk.

2. Létezik egy legnagyobb szám (M2). Ennél a számmal bővítés nélkül is túl nagy lehet. Például $ABS(DY) > M2$. Ilyenkor, ha a nevező egy-nél kisebb abszolút értékű, a keresett változó — ami nálunk y — is nagyobb lesz M2-nél. Ezen az esetben az eredeti program a BASIC „túlcsordulás” hibáüzenetével leáll!

Bekövetkezhet ez akkor is, ha megengedhető nagyságú számok adunk be? Igen, például másodrendű egyenletrendszerrel $DY > M2$, ha történetesen $M2 > a(1) > 1 + SQR(M2)$

$M2 > b(2) > 1 + SQR(M2)$
 $0 < a(3) < 1$
 $0 < b(1) < 1$ mert ekkor
 $DY = a(1)b(2) - [a(3)b(1)] > [1 + SQR(M2)]^2 -$
 $- 1^2 = [2 + SQR(M2)] SQR(M2) > M2$

És elozokban ismertetett két probléma együtt is jelentkezhet. Nézzük meg a lehetséges eseteket:

1. Egyik *sincs*. Erre az esetre jó az eredeti program.

2. Csak az első van. Erre jó, ha az $1/M1$ érték bővítés alkalmazzuk, kivéve, ha a bővítőstől a számláló lesz túl nagy, tehát ha $ABS(DY/M1) > M2$. Ilyenkor mint a harmadik eset.

3. Csak a második van. Erre az esetre jó, ha $1/M2$ értékkel bővítsük. Kivéve, ha a bővítőstől a nevező lesz túl kicsi. Ilyenkor mint a második eset. Ez alól is van azonban kivétel. Amennyiben először volt a második eset, abból lett a harmadik, és aztán újra a második lenne. Ekkor olyan, mintha a negyedik eset lenne.

4. Mindkettő van. Ez is bekövetkezhet, például másodrendű esetben, ha y-t keressük és

$a(1) = 1$
 $a(2) = 2 \cdot 1 \cdot M1$

$a(3) = 1$
 $a(4) = 3 \cdot M1$
 $b(1) = -M2/1.8$
 $b(2) = M2/2$ akkor
 $DY = 19 \cdot M2/18 > M2$
 $D = 0.9 \cdot M1 < M1$
 Ha itt bővítsük, akkor
 $DY = 19/(18 \cdot M1)$
 $D = 0.9/M2$
 A legtöbb Microsoft BASIC-nél
 $M2 = 1/(2 \cdot M1)$ így
 $D = 1.8 \cdot M1 > M1$ de
 $DY = 19 \cdot M2/9 > M2$

Tehát a probléma általában bővítéssel oldható meg. Kérdés, hogy szükséges-e az egyáltalán? Ennél az esetenél $DY > M2$, $D < M1$, így $Y > M2/M1$, ami tehát a BASIC-fordító számára „túlcsordulás”. Így ennél az esetenél számítás nélkül azt kell kiírni, hogy „NAGYOBB MINT”: $M2/M1$

A cikk elején említett probléma megoldására szolgáló programnak a négy eset mindegyikére kiterjedő megoldást kell tartalmaznia. Ez azt jelenti, hogy az eredeti programot úgy kell módosítani, hogy az eseteknél említett megoldásokat is tartalmazza. Egy ilyen megoldást ismertetek a következő számban.

S. E.

INFORM

Rovatunkban az Apple, Atari, Commodore és Sinclair mikro tulajdonosait feltehetően érdekli, angol és német nyelvű cikkekéről informáljuk olvasóinkat egy tartalomeleőr szólókn segítségével.

A forráshely karakterosorozatot nyílvonattal követi, ezt a / jellegű folyóirat kódja követi (lásd táblázat). A két / jel között a megjelenési adatokat (év, hó), illetve a cikk kezdő oldalszámát szerepeltetjük. A második / jel után pedig - az esetleges másolatkerést megkönnyítő módon - a cikk teljes oldalterjedelmét közöljük.

A folyóiratok megtekinthetők a SZÁMALK (Bp. XI. Szakasits Á. u. 68.), illetve - a x-gal jelzetek - az OMIKK

(Bp. VIII. Múzeum u. 17.) szakkönyvtárban. (A másolás díja oldalanként 8 Ft.)

A Magazin helyhiány miatt mindössze egyetlen címszó, a programlista közöttérelre vállalkozhatott a folyamatosan bővített adatbázisból. A kedvező visszhang alapján az OMIKK háromhavi bontásban kiadja a tartalomeleőr szőlőncok permutálásával és alfabetikus rendezésével szerkesztett teljes anyagot. Az "APACS Mikroindex" első füzeté már megjelent, ára 50,- Ft.
Több példány vásárlása/rendelése esetén 10 darabonként 2 tiszteletpéldányt térítésmentesen ajándékoz a terjesztőknek az OMIKK elővizsgálata (Bp. Pf.: 12. 1428).

A folyóirat neve Kódja

| | |
|------------------------|------|
| x 64'er Magazin | 64er |
| x Antics | anti |
| x Chip Magazin | chip |
| x Computer! | cute |
| x Dr. Dobb's Journal | dobb |
| x Elektor Electronics | etor |
| x Happy Computer | happ |
| x mc - Zeitschrift | mc |
| x Run (USA) | run |
| x Run (NZK) | run2 |
| x Your Computer | your |
| x ZX Computing Monthly | ZXCM |

```
PROGRAMLISTA
  wdfbnz13 c64 filekezesel 30000 kara
  kérés szövegváltás állományok használ
  ktechnikai adatok ->run2/87.09-102/5
PROGRAMLISTA
  amigal kinyitnyok ezeles menuparancs b
  ovitov utasítások (supernarus)
  ->cute/87.09-86/4
PROGRAMLISTA
  anivac00 atari xl/vel 4. részlemez
  ->anti/87.09-20/6
PROGRAMLISTA
  atari at0 programirasi basic forrasfil
  e beillesztés geometriai
  ->happ/87.09-105/2
PROGRAMLISTA
  atari xl/vel f-billentyűk programozása
  at (reagard 2000) ->happ/87.09-66/2
PROGRAMLISTA
  atari xl/vel jatek (attack on the doo
  mstar) ->anti/87.09-15/4
PROGRAMLISTA
  atari xl/vel jatek (be the eggsan)
  ->anti/87.09-35/5
PROGRAMLISTA
  atari xl/vel jatek (oktatás) kviz-file
  készítés analíziseles
  ->anti/87.09-33/4
PROGRAMLISTA
  atari xl/vel kamasztizmitasi (annual 2
  rate) ->anti/87.09-40/7
PROGRAMLISTA
  atari xl/vel levelezési címlista össze
  állítás nyomatás ->anti/87.09-30/6
PROGRAMLISTA
  atari xl/vel oktatás (matematika) 1. sz
  woloqep -modul fi/1 (telegopontos arit
  metika) ->anti/87.09-26/6
PROGRAMLISTA
  atari xl/vel programirasi basic karakter
  sorsorozat-kereso rutin
  ->cute/87.09-84/2
PROGRAMLISTA
  c128 boot-funkciók kiegészítés (boo
  t-saktor-manipulator)
  ->run2/87.09-49/3
PROGRAMLISTA
  c128 filekonverzió grafikai c64-forma
  tum illesztés c128-modushoz
  ->run2/87.09-39/3
PROGRAMLISTA
  c128 lemezkezesel (c128 notepad)
  ->run/87.09-48/4
PROGRAMLISTA
  c128 programirasi f-billentyűk atprog
  ramozasa ->run2/87.09-43/2
PROGRAMLISTA
  c64 BIOS felismerés-grafikon készítés
  ->64er/87.09-109/5
PROGRAMLISTA
  c64 grafikai mandelbrot 1. rész alapfo
  galmas példák ->64er/87.09-61/7
PROGRAMLISTA
  c64 grafikai programirasi color/screen
  ram címek megjelenítése kurzorral
  screen genis ->run/87.09-62/2
PROGRAMLISTA
  c64 jatek (duell) ->64er/87.09-38/3
PROGRAMLISTA
  c64 jatek (falling ball)
  ->run2/87.09-95/6
PROGRAMLISTA
  c64 jatek (mini breakout)
  ->run2/87.09-116/3
```

```
PROGRAMLISTA
  c64 jatek (phaser phire)
  ->run/87.09-38/4
PROGRAMLISTA
  c64 jatek (stepy-burvy)
  ->happ/87.09-51/7
PROGRAMLISTA
  c64 lemezkezesel f-billentyűk progra
  mozasa ->run/87.09-63/2
PROGRAMLISTA
  c64 mail box sysop paraméterek haszna
  lat elmozdítások példák
  ->happ/87.09-139/1
PROGRAMLISTA
  c64 matematikai grafikai készítés (h
  sepper) ->64er/87.09-57/4
PROGRAMLISTA
  c64 testregezés intelligenciadobát
  iv gondolkodás szimuláció (csart a
  lec) ->cute/87.09-77/5
PROGRAMLISTA
  c64 nyomatás 23. 9mmx23.5mm-es híres
  grafikai készítés
  ->happ/87.09-57/2
PROGRAMLISTA
  c64 programirasi assemblenl segitet
  t 16. csillag szöveg pull-down menük
  ->64er/87.09-68/8
PROGRAMLISTA
  c64 programirasi gépi kód segédfunkci
  ók bespites az operációs rendszerbe
  (c64) ->cute/87.09-97/3
PROGRAMLISTA
  c64 programirasi linker a címgörgeto
  szinvarialo eloprogramhoz lead (87.0
  7) ->happ/87.09-60/2
PROGRAMLISTA
  c64 programirasi üzemszoveg (letter
  maker) ->run2/87.09-101/1
PROGRAMLISTA
  jatek plus (boulder-dash) adaptaci
  o ->run2/87.09-126/4
PROGRAMLISTA
  ram atari xl/vel bővítés programirasi
  (hidden ram) ->cute/87.09-91/2
PROGRAMLISTA
  c128 lemezkezesel (c128 notepad)
  ->run/87.09-48/4
PROGRAMLISTA
  c128 programirasi f-billentyűk atprog
  ramozasa ->run2/87.09-43/2
PROGRAMLISTA
  c64 BIOS felismerés-grafikon készítés
  ->64er/87.09-109/5
PROGRAMLISTA
  c64 grafikai programirasi mandelbrot 1. rész alapfo
  galmas példák ->64er/87.09-61/7
PROGRAMLISTA
  c64 grafikai programirasi color/screen
  ram címek megjelenítése kurzorral
  screen genis ->run/87.09-62/2
PROGRAMLISTA
  c64 jatek (duell) ->64er/87.09-38/3
PROGRAMLISTA
  c64 jatek (falling ball)
  ->run2/87.09-95/6
PROGRAMLISTA
  c64 jatek (mini breakout)
  ->run2/87.09-116/3
```

```
PROGRAMLISTA
  c64 jatek (phaser phire)
  ->run/87.09-38/4
PROGRAMLISTA
  c64 jatek (stepy-burvy)
  ->happ/87.09-51/7
PROGRAMLISTA
  c64 lemezkezesel f-billentyűk progra
  mozasa ->run/87.09-63/2
PROGRAMLISTA
  c64 mail box sysop paraméterek haszna
  lat elmozdítások példák
  ->happ/87.09-139/1
PROGRAMLISTA
  c64 matematikai grafikai készítés (h
  sepper) ->64er/87.09-57/4
PROGRAMLISTA
  c64 testregezés intelligenciadobát
  iv gondolkodás szimuláció (csart a
  lec) ->cute/87.09-77/5
PROGRAMLISTA
  c64 nyomatás 23. 9mmx23.5mm-es híres
  grafikai készítés
  ->happ/87.09-57/2
PROGRAMLISTA
  c64 programirasi assemblenl segitet
  t 16. csillag szöveg pull-down menük
  ->64er/87.09-68/8
PROGRAMLISTA
  c64 programirasi gépi kód segédfunkci
  ók bespites az operációs rendszerbe
  (c64) ->cute/87.09-97/3
PROGRAMLISTA
  c64 programirasi linker a címgörgeto
  szinvarialo eloprogramhoz lead (87.0
  7) ->happ/87.09-60/2
PROGRAMLISTA
  c64 programirasi üzemszoveg (letter
  maker) ->run2/87.09-101/1
PROGRAMLISTA
  jatek plus (boulder-dash) adaptaci
  o ->run2/87.09-126/4
PROGRAMLISTA
  ram atari xl/vel bővítés programirasi
  (hidden ram) ->cute/87.09-91/2
PROGRAMLISTA
  ram-floppy c128 segédprogramok az '1
  780/1750 expansion modul-okhoz
  ->run2/87.09-51/5
PROGRAMLISTA
  ram-floppy c128 vdc-0563 atrogramok
  asak vdc-ram floppy
  ->run2/87.09-46/3
PROGRAMLISTA
  suctrun programirasi basic compiler
  (cuchi) 3. rész lemezias utbetato
  ->happ/87.09-68/4
PROGRAMLISTA
  szovegfeloldozas plus (script/plus
  modul modositasi) kiegészítés special
  is karferekkel ->run2/87.09-123/3
PROGRAMLISTA
  zenel apple 11 kompakt lemez rögzítés
  a kazettaponton keresztülmegzeirt
  adatok playback -je
  ->cute/87.09-86/2
PROGRAMLISTA
  zenel c64 grafikai programirasi 48 u b
  asic utbasitas poke-helyettesites
  ->run2/87.09-109/9
PROGRAMLISTA
  zenel c64 hang ismeret digitalizálás
  a udok azzetrol a 'databases' fel
  ->run2/87.09-107/2
```

Egyesek úgy vélik, hogy a számítógép-tudomány (a számítástudomány) alig több, mint programozás — nem érdemes önálló diszciplinaként tanítani. A szóban forgó cikk írója 1981-ben nem hivatalosan megkérdezte a számítástudomány több előadóját: saját tizenéves gyermekének fő tantárgyáknak ajánlaná-e saját szakterületét. Egyikük sem ajánlotta. Inkább a matematikát, a gazdaságtant, a fizikát stb. javasolták volna. Csak utána jöhetett volna a számítástudomány. A megkérdészetek egyébként a számítástechnikát mint pályát, mint hivatást, foglalkozást természetesen mind jók tartották, de úgy gondolták, amit ma tanítanak ezen a címen, az hamar, gyorsan elavul. Gyakran hivatkoztak arra, hogy a számítástechnikai (tudományi) végzettségük a végzés után néhány évvel nem produkálnak ezen a területen többet, mint a más szakon végzettek.

Az idézett cikk szerzője azt állítja, hogy kialakítható olyan filozófia, összeállítható olyan tanterv, mely alapján önálló diszciplinaként tanítható a számítástudomány. Ez a „tudás—információ—adat” hármas fogalomcsoportha köré szervezhető. De mi az adat? Mi az információ? Azon kívül, hogy azt mondjuk: ezek a valóság valamilyen reprezentációi, leterképezései, a válasz a problémát elemző ember szemléletétől, illetőleg attól az absztrakciós szinttől függ, amelyen az illető dolgozik. Például egy biológiai rendszer meghatározó tulajdonságai nem feltétlenül jelennek meg a kémia alapvető szintjén, bár elvileg megmagyarázhatók lennének a kémia és a fizika fogalmaival. Azt lehetne mondani, hogy történetesen a szociológia, a pszichológia, a biológia, a biokémia, a kémia és a fizika valamilyen hierarchiát alkot. Nem kell mindenkinek feltétlenül egyetértenie egy ilyen hierarchiával, de ez most nem is fontos. Az idézett cikk szerzője szerint az információk rendszerek tervezésénél valamilyen általános elfogadott megközelítésből kiindulva el lehet jutni valamilyen megfelelő hierarchiához. Példát is hoz fel erre:

információs szint; koncepcionális szint; fizikai szint vagy a teljes rendszer szintje; információs szint, az adatok szintje; a fizikai reprezentáció szintje, avagy tudás, információ, nyelv; adatok; fájlok, rekordok; jelek; számítógépes ábrázolás, karakterek, szavak, fájlok, elektronika (és manapság száloptika).

Bár rovatunkban magyar nyelvű publikációkkal nem szoktunk foglalkozni, most mégis kivételt teszünk, mert a témához szorosan kapcsolódó alapvető publikáció jelent meg a Magyar Tudomány 88/7—8. számában: „A számítástechnika és a számítógép-tudomány értérendi kérdései” címmel. Mivel ez az írás Magyarországon viszonylag könnyen hozzáférhető, ezért izelítőlünk sem különk belőle részleteket, csak minden téma iránt érdeklődő olvasónk szíves figyelmébe ajánljuk.

Augusztus óta rovatunk minden egyes cikkében teljes egészében közöltünk egy-egy Uptdike-írást is. Következzék most is egy — fogadják szeretettel.

J. Uptdike: Vízvezetékeink

Az öreg szerelő gyöngéden előrehajol, nemrég vett házam pincéjének fémhálóban megmutat egy mives, antik csökötet.

— Már vagy harminc éve nem csinálják így — mondja. Hangja vékony szivárog, akár a rozsdán átpréselődő víz. — Harminc-negyven éve. Mikor elkezdtem dolgozni apám mellett, még csak így csináltak. Régi ólomkötés. Kenni kellett: egyik kezemben forró öntőkanál, másikban vizes rongy. Tizenhat mozdulatot kellett megtenni, mielőtt az ólom kihűlt. Tizenhat különböző mozdulatot. Másképp vége, a kötés tönkrement, le kellett az egészest vésni, kezdhetne az ember elől. Amikor nekivágtam a szakmának, így kellett csinálni. Lehettem vagy tizenöt-tizenhat éves. Ez a kötés, itt ni, lehet vagy ötvenéves.

Ő ismeri csöveimet, én csupán birtoklom őket. Tulajdonosok jönnek-mennek, ő marad. Azt hisszük, tudjuk, látjuk, kik vagyunk, aztán rájövünk, valójában csupán egy rakás szerencsétlenség vagyunk. Úgy véljük, élettelet, szép kilátást vásároltunk, mikor valójában csupán labirintust, történelmet, csöleleteket, csatlakozásokat, bifurkációkat, szelepeket. A szerelő egy vastag, sötét csövet mutat, mely átlovas az alapha fut.

— Látja ezt a vonalat a cső alján? — Fehér vonal, lehetetleni lerakódás, fakó oxid.

— Ne nyúljon hozzá, csepegni fog. Látja, ezt a csövet két részből öntötték, úgy kellene beépíteni, hogy a varratok oldalra nézzenek. De néha

varratall lefelé építik be őket. — Két begömbölyített tenyerével mutatja, hogyan. A két tenyér lassan szétrnyilik — tágul a rés. Markai között kileves magam is szinte kiutat kereső vízzel válok ügyes bemutatásában.

— Látja, a végét elkezd csöpögni. — Zseblámpájával visszafelé követi az áruklódó fehér vonalat.

— Négy-öt új szakasz megtenné. — Felsőhagyi, zihál. Pupillája a félmályban töltött élet miatt tágabb, mint az átájtáemberé. Költő: ahol én csupán hibát, bosszantó kiadást előlezőző tökéletlenséget látok, ott ő gyöngéd pillogat, elmereng a korrózió és az áramlások örökkévalóságában. Ironikus számlarekműveket küld nekem, benne helyes kis alkatrészek jegyzékével:

1 db 1 1/4 × 1" galv. karmantyú 58 cent
1 db 3/8" sárgaréz visszacsapó szelep 90 cent
3 db 1/2" csökötés 13 cent.

Szinte örülnek tünik e kínos precizitás, melyet a végén nyel le a győmindent elsőprő „munkadíjnak” nevezett szép, kerek összeg: munkadíj 550 dollár.

Feltételezem, mostani elzérékenyült meditációt — a hosszú, pillogásokkal kitöltött szünetekkel együtt — szintén munkának számítja.

Régi házunk, melyből elköltöztünk, egy mér földnyire innen, szinte megkönyebbült, hogy megszabadult tőlünk. A szobák — valamikori étkezéseink, ceremóniáink, magánrandmaink színterei, hol gyözelek közöltünk felsérdültek, ahol a tér, a lépcsők telítődve mindennapi mozdulatainkkal és szabálytalanságaikkal úgy csontjainkba ivódtak, hogy éjszaka is baj nélkül tudunk közeledni bennük — várakozás ellenére nem látszanak gyászolni minket. A ház ujjong hirtelen visszanyert méretein, üres sarkai tágaságában. A sokáig szönyeggel takart padlódeszkák ragyognak, mintha frissen lakoztak volna őket. A nap háborítatlanul süt be a függönyöket vesztett ablakokon. Megfátalodott a ház. Amíg az új tulajdonosok meg nem érkeznek, ismét saját maga lehet, ismét saját életét élheti, melyet egy időre elaltart a miénk. A padlódeszkákat most csak a holdvígát recseggeti. Ha néhreggelente visszatérek egy-két ottefejtett apróságért — kandallóvaséri, képkeretért —, szüzs szemtelenséggel fogad. Ha aajtáját kitárom, mintha csak a macskát engedném be reggeli tejéért: miakol egyet éjszakai álmuunk melegét meg öröz ágyunk fele haladtabban. Szokásai — e nyavogás és a megosztott közös fedél erejéig — hajszálfinomnan illeszkednek a mieinkhez. A természet nem annyira érzélgős, mint ahogy azt az ekologusok feltételezik. A ház mindössze egy nap alatt elfelejtett bennünket.

Büntudatot érzek, hogy oly kevéssé vittük birtokunkba: három szállítómunkás meg egy fuvalat ki tudta törölni még az emlékünket is. Jó tíz éve, amikor beköltöztünk, nem volt kísértéltába, bár gerendái és a kandelok lehetek vagy háromszáz évesek. Azt hittem, kora miatt az lesz, de egy amatőr boszorkány, akit feleségem meg az egyetemről ismert, megkopogtatta a hálósobák falait, körülszaglászott a padlószobában és biztosított bennünket, hogy a hely kísértetlenek. Szemei (nem furcsa?) ugyanolyan természetellenesen tágba nyíltak, mint a szerelő. A ház puritán farmerrel építették. A XIX. században vendégfogadóként szolgálhatott: a newbury porti bekötőt itt vezettelt el nem messze. S a harmincas években berházta alakították. A ma oly tágas szobákat gipszkarton falakkal megosztották, a falakon lyukakat vágtak, ezen a lakók cukrot, lisztet cserélhetek. Falusias idők, szégyény idők. Egy időben még csirkéket is tarthattak a padláson. Kezdetben a gyerekek azt mondogatták, esős időben érzik a nedves toll szágot, de ezt csupán a fantázia és a mítosz hatalmának tulajdonítottam. A kertet ásva, egy letűnt kor technikájának emlékeikt, néha ónkanalak és furcsa törött flaskák kerülnek elő a földből. Utánunk mindössze néhány gollfabla maradt az iriszek között, és néhány poros teniszlabda a fűtőtestek alatt. Hátrahagytok szellemeknek csak mi magunk látjuk.

Egy férfít látok szmokingban és egy asszonyt hosszú fehér ruhában, amint ide-oda lépegetnek a kertben. A hideg szél nevetésre ingerli őket. Hajnali kettő van, hűsvét hajnalra. Főliába burkolt csokoládétojásokat rejtenek el mindenfelé, be vannak csipve. Reggel selymesen bízsergető fejfájással a csokoládémasztó gyerekek kiabálására és veszekedésére ébrednek, amint azok masztos szájjal és émielyítően édes lehellettel bemásznak szüzeik ágyába. De a hajnali jelenés az, amit a könyhában álva a kijózanodott tudat perspektívájából látok, a fételen kedvű párt látom, amint ott ügyetlenkednek a kert sarkában, az aranyesőbokr és a-hinta körül — két hűsvét nyuzi.

Férfi hajjal egy gyerekgáj fölé, együtt imádkozik a gyerekekkel. Bajban vannak az imádság szövegével — más-más vasárnap iskolába jártak.

egy cikket a számítástudomány és a informatika tanításának filozófiájáról (L. Capper: „A Philosophy for the Teaching of Computer Science and Information Technology,” the Computer Journal, Vol. 29. No. 1., 1986.) és erről eszünkbe jutottak azok a fiatalok, akik már megismerkedtek a számítógéppel, a BASIC nyelvvel, megszerették a számítástechnikát és azt fontolgatják, hogy válasszák-e a számítástechnikát egy életre hivatásul. A cikk központi kérdése: tudomány-e, tudományosan megalapozható-e a számítástechnika — vagy másképpen: kelően megalapozható-e a számítástechnika oktatása.

A férfi elgyötört és (talán a csirkeotlak szelleme miatt?) kissé asztmásan átmege a szomszéd szobába, miközben már az jár eszébe, hogy le kellene menni elolvasni egy könyvet, meginni egy pohár italt. Ott a gyerek — egy nagyobb — halik felkiált, amikor felajánlja neki, hajtsák együtt imára fejüket. „Apa, ne!” Kerek fehér arca — csak homályosan kivethető az esti félhomályban — szinte átizzik a feszültségtől, a zavartól, az esdekléstől. A férfi maga is zavarban van — túl könnyen zavarba lehet hozni —, megcsókolja a lányát, kihátrál, becsukja a hálószoba ajtaját, otthagyja őt a sötét szobában.

A legnagyobb szoba falai — eltekintve azoktól a kísérletszerű négy-szögletes faltoktól, ahol valaha a könyvszekrények álltak, képek lógtak —, most csupaszok. Itt vitatkoznak, szenvedélyesen gesztikulálnak. Az asztson, a feleség, éppen elhajt valamit — eredeti szándéka szerint egy hamutartót, de a rendszeret még ilyen rózsavörös arcú, feldúlt állapotban is győzedelmeskedik benne, és végül is egy könyv mellett dönt. Könnyekben tör ki, talán puritán gyökerű képtelensége miatt, hogy egyszer jól odavágyon egy hamutartót. Átrohan egy másik szobába, és nem felejt el átugrani a küszöböt sem, melyben idegenek gyakran megbotlanak. A gyerekek közben csendesen fel-le lopakodnak a lépcsőn. Sápadtak, tele vannak büntudattal, árthatlan kis szívük rejteken magukat hibáztatják e zűrzavarért. Még a kutyja is összekuporodik, farkát maga alá húzza. A férfi a heverőbe sápad, mely már nincs ott a helyén. Bokái összeeszer, feje leszegezve, mintha bilincsbé lenne verve. Túlajtsza helyzetét, úgy tesz, mintha börtönbe lenne zárva. Valószínűleg nyár van, mert egy kis képzőstalepke oda nem illően fehérlik a szalagáterem, melyhez kívülről mályvarozásák dörszölnönek-ütödnék gyengédn. Az asszony visszatér, arca vörösből már rózsaszínre váltott, most már szabatosan, megfontoltan érvel. A férfi feláll és kiabál. Az asszony megüti, az félreléki a másik kezét és oldalba vágja. Meglepődik, milyen kellemesen szivacsos anyagot ér: egy zsák zsiger. Ott ugrádoznak az útjukba első bútorok között, amikből apró porfelhők szállnak fel. A gyerekek egy lépcsőfokkal feljebb húzódnak. A kutyja begörbíti a hátát, mintha vernék, az ajtóhoz megy, kikéredkedik. A férfi átöleli az asszonyt és suttog neki valamit. Az rózsaszín, felhevült és folyvak a könnyei. A férfi észreveszi, hogy ő maga is sír, milyen jó érzés, mint amikor végre hányni tud vagy jól kiizzad az ember. Mit mond egymásnak, miről beszél egymással az a két érozásos, ijedt ember? Változásról, természetes folyamatról, az idő múlásáról, a halálról.

Gyarló szellemek. Eltűnnek, mint a lehelet nyoma az ablaküvegről. El-lenpontok emeketeztem ideződnék gyermekekorr hatalmas, súlyos, szafos kökuszreszelékkel töltött hűsüti tormaik. Nehezek, mint egy öntvény, vagy inkább tágasak, mint egy színház, telis-teli papírkivágással — bennük egész kis sziluettvilág, mely saját napfényt áraszt. Ezek a tojások lila forgaszészükben felhözök a rejtelmek csillagok között, szivárgásmentesen soha le nem zárható kútjából azokat az időket, amikor azok a réges-régi, születésem előtti fénýképek készültek, amikor Isten még odafigyelt ránk. Vasárnap reggel van, szinte még éjszaka. Imádkozom. Egy fekszem egy sötét, sarkaiban szellemlelta házban, mint egy szalmaszál a szakadék peremén. Walt Disney-ihletésű, begörbített karmú szörnyek vesznek körül. Egy városban vagyok, melynek legfőbb bűkesége a központjában elterpeszködő ravatalozó, melyet kőrökörül boszorkányjelkekkel megjelölt pajták fognak körül. A szalon szőnyegén egy kontinens alakú föld étkelenkedik, picit koromban odahánytam. Mitosz mitoszom: már három- vagy négyéves vagyok, éhes, mindenféle piszkot eszem furcsa, tollbokrátszerű, kődös, trópusi páfrányokat tartó virágcserepekből. Egyik nagyanyám babonás hite szerint a gyerekek évente legalább egy font piszkot kell megennie, attól lesz erős. Később — kilenc-tíz éves koromban — a hasamon fekszem ugyanott, az újságot olvasom fel nek gyagapámnak. Először a halálórási rovatot, aztán a helyi híreket, végül az első oldalról a főcímeke a

janárokrol és Rooseveltről. A papírnak jellegzetes szaga van, nem áporodott, mint a képregenyéké. Frissebb, kevésbé édeskes, mint a fánkos zacskóé, de fűszeresebb, izgatóbb, benne van a jövő, a ropogás, frissen, a kis-még melegebb csomagolt dolgok illata, az *újdomságoké*. Észreveszem, ez a szag naponta megérkezik és el is enyészik. Aztán tizenhárom éves leszek, és elbűcsűzünk a szalontól. Költözünk. A kontinens formájú föl mellett ott virítanak a páfrányok cserepeinek kerek lenyomatai is. A függöny nélküli albakokon zavartalanul besütő nap szinte felfedezi őket. Olyan mélyek, mint a dinoszauruszok lábnyomai.

Gyerekeink érzékelték-e vajon a mi hajnali hűsüti nyusziskodásaink frivolságát? A fiatalabbik újját szópva a kisebb szobában feküdt, és mellette a sötéte bámult valamire. Számára házunk úgy át lehetett itatva rettenettel, hogy az emlékezetének minden rétegere rányomhatba bélyegét. Neki minden apró karcolás a falon megfújhatetlen jelentéssel telítődhetett. Ő volt az egyetlen gyerek, aki beszélni szokott a halálról. Holnap volt a születésnapja.

- Nem akarok születésnapot. Nem akarok kilencéves lenni.
- De nőnökkel. Mindenki felnő. A fák is.
- Nem akarok.
- Nem szeretnél olyan nagy lány lenni, mint Judit?
- Nem.

— Kifethetnél magad, melltartót viselhetnél, biciklizhetnél a fűtácán.

- Nem akarok biciklizni a fűtácán.
- De miért nem?
- Mert a végén megöregszem és meg fogok halni.

És patakaznak a könnyei, és a férfinak mellette erre már nincs szava, ahogy egyetlen férfinak sem lenne ebben a kis szobában, ahol semmi sem marad utánunk, csupán néhány karcolás és az ablakkereten egy félig levakart, kifakult Maci-embléma. Ha nem költöztünk volna el, ideje lett volna már feltelni a téli védőablakot az ablakokra.

Nyílik az ősi kikerics régi házunknak, az újnál virágoznak a nárciszok. A gyerekek, akik előtünk laktak itt, teniszlabdákat rejtettek el számunkra a radiátorok alatt — meglepetésül. Amikor akudtunk az új házra, észre-érezteztük őket, amint ott bujkáltak házuk körül, és a bokrok, a kerítés mögé rejtőzve bámultak minket, jövendőjük bitorilói. Miután ők már kiköltöztek, és mi még nem költöztünk be, boldong, féktelen játékokat játszottunk az üres szobákban, a falaknak pattintottuk a labdákat, össze-vissza uggrabugráltunk. A labdák hamarosan újra elvesztek. A szobák megteltek bútorral.

A szerelő gyengédn elmereng, és egy lefűrészelt csöcsönkot mutat, a kútból a nyomótartályba vezető szakasz egy darabját. A cső belső átmérője az ásványi lerakódásoktól nem nagyobb, mint az ujjja. A koncentrikus rétegek olyan vékonyak, akár a papír. Az egész egy végétől kezdve olvasandó könyv képzetét kelti, olyanét, melyet nem szokás kinyitni, amit papok őriznek bölcsen, lakat alatt.

- Figyelje csak — mondja —, mindez negyven-ötven év alatt rakódott le. Emlékszem, mikor apámmal beszereltük ide egy szivattyút, ez a cső már itt volt. Nem lehet ez ellen semmit se tenni, az oldott ásványi anyagok miatt van. Nincs mit tenni, ki kell ásní és ki kell cserélni egy új ötnegyedesre vagy hatnegyedesre.

Magam elé képelem, amint a pázsímatot felszaggatják, amint a nagy, aranyszínű markolóéket letapossa a nárciszaimat, amint dollárjaim szépen elfolydognak. Tiltakozásom hatástalan.

A szerelő felsőhajt, ahogyan a költők szoktak, fél szemmel a hallgató-ságra sandítva.

— Nézze, ha így hagyja, egy szép napon leég az új szivattyúja. Túl nagy munkát kell kifejtene, hogy felszívja a vizet. Cserélje ki most, soha többé nem lesz gondja vele. Túléli magát.

Engem, meg őt. Szemei tágira nyílnak a korrozó és az áramlás néma jelenlétében. Keresztülpréseljük magunkat a pincefajlárón; az égbolt egy vakító darabja foglalja el helyét felettünk ideiglenes-ídotlen felhőkkel tarkítva. Korás-körül — minden túlélt minket.

(Fordította: K. E.)

From the book *Museums and other stories* by John Updike.

© 1971, 1972, 1975, 1983 by John Updike.

„Plumbing” originally appeared in *The New Yorker*.

ANYAG ÉS MOZGÉKONYSÁG

A sakkhadállások értékelésének két legfontosabb tényezője az anyag és a mozgékonyosság. Ezeknek az értékelési módjaival régebbi számainkban már megismerkedhett az olvasó. Ezúttal a velük kapcsolatos statisztikai adatokat ismertetem, amelyeket Dap Hartman dolgozott fel a legutóbbi alkalommal már említett, 832 nagymester-játszma számítógépes elemzésével. A játszmák feldolgozásánál az anyagi érték számítását Shannon 1950-es definíciója alapján végezte el. Ezek szerint:

$F_{anyag} = 0,9 \cdot V + 0,5 \cdot B + 0,3 \cdot F + 0,3 \cdot H + 0,1 \cdot GY$, ahol V a vezérek, B a bástyák, F a Futók, H a huszárok, GY a gyalogok száma. Ezek szerint az alapállásban világosnak és sötétnek egyaránt 3,9 anyagi összértékű figurája van.

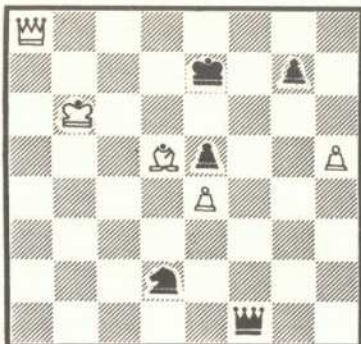
Az 1. ábrán azt láthatjuk, hogyan alakul az anyagi érték a féllépésszám függvényében. Szembetűnő, hogy a tizedik féllépésig szinte stagnál, majd rohamosan csökken a 129-edik féllépésig, amelytől viszont kissé emelkedik, de emelkedik.

Az anyagi érték változásának ez a három szakasza jól szemlélteti a sakkjátszma három fázisát. Az első szakaszban, a megnyitásban, mindkét fél kifejlődik tisztjeivel, ezért kevés a cserelhetőség. A második szakaszban, a középjátékban viszont a kombinatív játék kerül előtérbe, ami a figurák lecserélését is magával hozza. A leegyszerűsítések után a középjáték fokozatosan átmeny a végjátékba. A 129-edik féllépés után pedig a gyalogátalakulások miatt ismét nő világos és sötét anyagi értéke.

A 2. ábra világos és sötét anyagi értékek a különbségét mutatja a féllépésszám függvényében. Látható, hogy a két fél anyagi értékének a különbsége általában akkor a legnagyobb, amikor anyagi értékük összege a legkisebb. Ez ellentmondásnak tűnik, de ha szemügyre vesszük a 2. ábrát, akkor láthatjuk, hogy a két fél anyagi értéke közötti legnagyobb különbség sem nagyobb kétgyalognyinál.

Ez azért van, mert a nagymesterek kiegyensúlyozottan játszanak, és a legkisebb pozíciós előnyért is nagy harcot vívnak egymással. Ezért egy vagy két gyalog előny, illetve hátrány legtöbbször már a győzelem, illetve a vereséget jelenti.

A mozgékonyosság szintén ismert fogalom. Az adatok pontos ismertetése és azok összehasonlítása kedvéért háromféle definíció szerint mutatjuk be a függvényeket. Az első meghatározás szerint a mozgékonyosság adott helyzetben a lehetséges legális, a második meghatározás szerint a pszeudolegális lépések száma. Ez utóbbiak azokat a lépéseket is tartalmazzák, amelyek sakkbán hagyják a királyt. A sakkprogramok általában ezt a megoldást alkalmazzák. Ilyenkor a sakkdást csak a lépés megtétele után ellenőrzik, ami szemben áll a mozgékony



3. ábra

értékkel, hiszen ez a lépés megtétele előtt számítják ki a lépéslista alapján. A harmadik lehetőségre De Groot professzor muta-

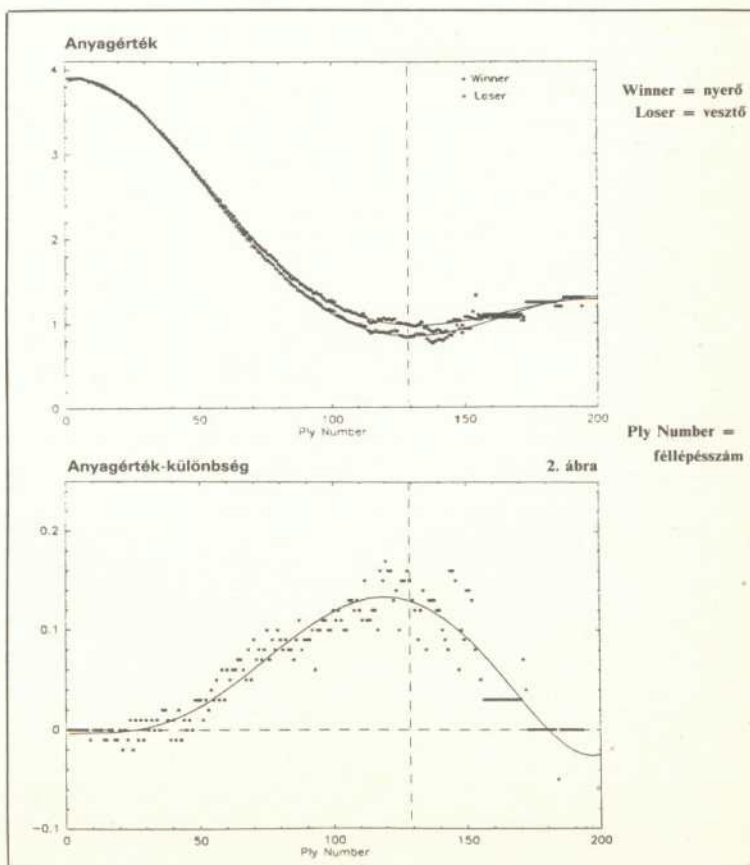
tott rá, aki a pszeudolegális lépések számát tekinti a mozgékonyági értéknek, de az előző definícióval ellentétben nem értelmezi a függvényt abban az esetben, ha az egyik fél királya sakkbán áll.

Nézzük meg a 3. ábrát, amelyen a Kaszparov—Beljavszki-féle 1981-es játszma 116. féllépése után létrejött hadállást mutatjuk be. Világos és sötét legális lépéseinek a száma egyaránt 31, a pszeudolegális lépések száma világgal 33, sötéttel 36.

A feldolgozott játszmák azt mutatják, hogy a mozgékonyági érték a 40. féllépésben éri el a maximumot, majd csökken a 129. féllépésig. Innen az anyagi érték növekedése miatt a mozgékonyági érték is fokozatosan növekszik. Az e játszmákban előállott pozíciók közül a minimális legális lépések száma 68 volt, a minimálisé 2.

(Folytatjuk)

KOVÁCS P. ATTILA



Winner = nyerő
Loser = vesztes

Ply Number =
féllépésszám

2. ábra

Az Olvasó írja

A rovataban ezúttal is két levelet olvashatnak, mégpedig azért, mert mindkét levelehez nagyon sok hozzáfűzni valóm van.

Pauló János, Salsóvártúrján

Már régóta halasztagos, hogy írok önöknek. Azért csak most fogok tölteni, mert fellelkesült, hogy egyre több foglalkozóknak az ENTERPRISE számítógépe. Nagyon szeretnék a gépi kódolásával megismerkedni, de nem tudok a megfelelő szakiradalomhoz hozzájutni. Jó lenne, ha többet írnának róla.

Sokat foglalkozom zenei programokkal, tanulok orgonálni is. Nagyon szeretem Johann Sebastian Bachot. Főleg az ő műveit írom gépre. A szeptemberi számban Dömösi Attila írt egy rövid zenei programot. Számítógéppel azonban, akárcsak a többi, mint hangszert nem nagyon lehet kiharánálni. Viszont mint zenésdi számítógéppel fantasztikus hatást lehet elérni: Bach: Nolyk kis preludium és fuga' orgonára írt műveiből már hatot ráírtam a számítógépre. A programok egyenként átlag 28 kilobájtosak. Hármashangzatok szólamok és sztereóban. A program, annak ellenére, hogy BASIC-ben van, gyors. Éléri a művekre előírt gyors tempót. A legjobban fejhalgatható lehet élevení a zené. Ha valakét érdekelnek ezek a programok, szívesen küldök nekik. Közösítőm segítségével.

Az Enterprise-ről még egyszer. Ha olvasónk újra áttöngészeli a korábbi Mikroszámítógép Magazin-számokat, akkor számos olvasói levelet talál, amelyek dicsérik az Enterprise-ot és a gép programíróitól, vagy szidják a forgalmazókat, hogy harcolt árulnak és korábbi szoftverjeiket nem teszik. Sokan engem is elmarasztalnak, hogy az utóbbiak partján állok, pedig ez valóban nem így van. Mi is a Magazin álláspontra? Nálunk — talán észrevették, ez már hazai szokás — az olvasók érdeke és nem a cégnek kívánása vagy politikája a legfontosabb, hiszen minden kérdésben szeretnénk független és objektív álláspontot kialakítani. Az tény, hogy egyre több Enterprise-számítógépet kerül forgalomba, és ezért kötelesebbnek, hogy az Enterprise-ok amióta tulajdonosait a lehető legtöbb információval lássuk el. Szerencsére néhány jó tollú professzionális alkalmazó is vásárolt ilyen gépet, így várható, hogy a jövőben még több, az Enterprise otthoni alkalmazásával, illetve szoftver- és hardverlehetőségeivel kapcsolatos írás jelenik meg a Mikroszámítógép Magazinban.

Zene. — Előjáróban: én nem vagyok muzsikusi, de az utóbbi időben az NYSZT Számítógépes Zenei Bizottság tagjaival, Patachich Ivánnal és más muzikusokkal is sokszor dolgoztam együtt. Ezért azt is mondhatom, hogy meggyőzettem én így lassan értegetem is a számítógépes zenét. Amióta leveleiben írt, hogy Bach műveit átírja számítógépre, azt hiszem, ez csak a kezdet, hiszen a számítógépet mint így új és sokat tudó hangszert használja. Megérttem lelkesedését, hiszen mi is átéltük ugyanazt valamikor az ötvenes évek végén, amikor az M-3 hőszer előszította Bach egy szerzeményét. (Ki tudja, miért — ha jól emlékszem, mi is ezzel kezeltük.) A számítógép

ma már nagyon sokféle feladatot lát el, például zenei kutatásban is. Nem is olyan régen alkalom volt többször is meglátogatni a két leghíresebb párizsi zenei kutató műhelyt, az IR-CM-ot (Institut de Recherche et Coordination Acoustique Musicale) és a Naxos alapított CEMAMU-t (Centre Mathématique Musicale), ahol a zeneszerzés és a zenei kutatás soha nem képzelt lehetőségeit átlánk a kutatók és a zeneszerzők rendelkezésére.

Vicetréve leveleire, az a javaslatom, hogy ne álljon meg Bach műveinek számítógépes interpretálásánál. Ha a számítógép és a zene kapcsolata valóban érdekel, akkor legjobb a hangok mikrovilágába merülni, ahíhez igen magas esztétikai érzék is szükséges. Azt hiszem, hogy ennélkül ma már nincs esély arra, hogy valaki komoly számítógépet alkalmazó zenei szakember legyen.

Varga Zaigmond, F., Mészárszka

Az első szám megjelenése óta veszem magazinjukt, és kísérlem figyelemmel cikkeiket. Nagyon sokat tanultam belőlük. Így jutottam arra az észra, hogy a gyermekeimnek számítógépet vegyek, ami egy C16-osban öltött testet. Műtán előmúvelési ipari technológiá üzemérnök vagyok, természetesen érdekel és érdekel az alkalmazotti számítástechnika. Sajnos, szakértőletem olyan nagy a lemaradás, hogy arra sző sem érdemes fordítani, talán az adátfeldolgozásban (administrációban) van mozgás. Ez indított arra, hogy gyermekeim játékra szánt gépet először programokkal "növeltem", majd hardvermodosítással C64-re változtattam át. Most öt tartok, hogy a SUMMATECH Ipari és Szolgáltató Kiszerveket SRD 256 típusú memóriával RAM-DISK-jet próbálom a C64-esz interfészrel a mi gépünkre alkalmazni tenni (az expansions port csatlakozás különbözőség miatt). Természetesen ez már nem gyermekek leheltségét szolgálja, hanem az onyomat. En ugyanis időközben annyira kedvet kaptam a számítástechnikához, hogy elkezdtem tanulni a rendszer-szervező szakon. S bár a kis "játékok" esetleges tanulóhátrá még alkalmazható, már nem jó, de bővíthető bizony dolgokra még használható. Foglalkoztam az a megoldás is, hogy van már illesztési megoldás az IBM — COMMODORE között, tehát érdekelne lenne ezt a lehetőséget megpróbálni, hogy a meglévő expansions kapu hogyan lehetne felhasználni olyan cartridge csatlakozására, amelyekben valamelyik DOS adva van (bár a RAM-DISKZ és a lemez meghajtók is saját operációs rendszerrel dolgoznak).

Az a véleményem, hogy az ország olyan területén, mint ahol én is élek, még sokáig léteznek a C64-es gépek, vannak olyanok is, ahol az adátfeldolgozás most gépesítik, és erre vetek C64-esek felől kíváncsi, sokkal jobban tudatosítani kellene, hogy a korábban említett kizsárvételek RAM-DISK-ZE nagyobb lehetőségeket kínálna azoknak, akik a régi-új C64-süket szeretnék

tovább használni. Talán valamely fórumon ezt meg lehetett tenni, valamint azt is, hogy a C64-es IBM-illesztése és halozható állítása is megoldott mar.

Egy éve, körülbelül, bevezető jelleggel, a város 5. Sz. Alulnaks Iskolájának 2. osztályában megpróbáltam számítástechnikát tanítani. Azt hiszem, sikerült felkelteni a gyerekek érdeklődését. Az iskolának külön számítástechnikai kabineje van. C Plus-és és Videoton-számítógépekkel. A pedagógusokkal folytatott beszélgetések során kiderült, hogy keret hiányában semmilyen bővítést nem tudnak csinálni, így pedig alacsony szinten, gyors fejlődés nélkül mehet csak az oktatás. Sokat gondolkodtam azon, hogy a drábaközvetlen további növelése lehetett egy komolyabb gépnek (gyelven IBM PC/XT-kompatibilis gépek) a lenne-e lenne olyan oktatási intézményekben, ahol komolyan kívánják oktatni a számítástechnikát. Elénezt, hogy így elszaladtam...
A lapról általában e s a k jót írhatok. Önöknek köszönetem, hogy végül is részintem magam ismét a tanulásra. Közösítőm.

Amikor öt évvel ezelőtt Kőnyves Tóth Pállat, a Mikromagazin elindított, akkor arra gondoltam, hogy a lapnak egyik legfontosabb feladata az lesz, hogy a számítástechnikát és a számítógépet megszerettesse a nem számítógépes szakemberekkel is, és annyira közel vigye hozzájuk, hogy na csak otthon, de a munkahelyükön saját napi feladataik megoldására is felhasználják.

Varga Zaigmond levele nekem — de azt hiszem, a szerkesztőség valamennyi munkatársának — igaz ünnepi ajándék. Ilyen szót szorít — amelyet öt éve még elképzelni sem mertünk — csak az ártit produkálhat.

Egy élelmiszer-ipari technológus üzemérnök olvassa a Mikrot, mert rendszeresen megveszi a gyerekeknek. A lap cikkének hatására ő is vész — persze csak a gyerekeknek — egy C16-ot. Ha már a gép megvan, akkor ekkor használni, majd ne találna kielégítőnek a hardver- és szoftverkörnyezetet, továbbfejlesztés a gépet. A levelek ez a része olyan szakzszerű és pontos, hogy csak a vak nem látja, hiszen az élelmiszeripari szakembere számítástechnikai specializációval rendelkezők, akiknek a számítástechnika a hardver- és szoftverkörnyezeti igénye, kiárandulás a PC-világba. Ami azt követi, már egyáltalán nem szokványos, ti. azt akarja adni tudását azoknak, akiknek leginkább szükségük van rá, a diákoknak. Tanítani kezd és hamarosan világosul válik számára az, amit az ezért felelősök sokszor még mindig nem látnak, az iskolában a mennyiségű szemeleletet már régen fel kellett volna váltani a minőségigé. Nem arra kellene törekedni, hogy minél több alkalmazott számítógépet legyen az oktatásban, hanem arra, hogy inkább kevés gép legyen, de az sokat tudjon és el legyen látva szoftverrel és technológiával. Ami persze nem csak a BASIC tanításra, de a mai számítástechnikában nélkülözhetetlen alkalmazotti környezetet (táblakezelés, szövegszerkesztés, adatbázisok, hálózati VTX-ek) megismertetésére is alkalmas.

Végül, még egyszer emlékezve és emlékeztetve a Mikroszámítógép Magazin ötödik születésnapjára, ezúton köszönetet mondok a rovat szerkesztéséért, az Olvasók megválasztásáért, azt a sok levelet, amelyet az elmúlt öt évben kaptam. Egyben elnevezést is kérek azoktól a levelezőktől, akiknek esetleg nem válasszunk, illetve leveleiket nem közöljük. En így szem mindig úgy válogattam, hogy a rovataban a kézzelírt levelekre szántartó leveleket jelöltessük meg. A következő öt évben várja leveleiket:

KOVÁCS GYÖZÖ

Az Ön feje nem adattárolásra való, hanem fontos döntések hozatalára. Joggal várhatja el, hogy kezében legyen az eszköz, ami munkáját minőségivé teszi, döntését megalapozza.

A számítástechnikában viszont a széles választékból nem könnyű a legjobb mellett dönteni.

az "elérhető" munkakörünk,

amikor létrehoztuk az első, Közép-Európában egyedülálló számítástechnikai szalonot. Meghívtuk a legfontosabb gyártókat és forgalmazókat, hogy a választék együtt legyen áttekinthető, kipróbálható, tanulmányozható, összehasonlítható. Felkészült szakembereink várják az érdeklődőket, a leendő vásárlókat. Reméljük, döntésünk új korszakot nyit az Ön mindennapi munkájában.



PC szalon

Budapest XIII., Sallai Imre u. 6.

☎ 315-136, 310-776

Lépjön új korszakba velünk.

NOVOTRADE



FIGYELEM!

Régi s bevált módszer, hogy a diákok a tanév elején beszerzik a könyvek és munkafüzetek mellett a feladatlapokat is, hogy előre felkészülhessenek a dolgozatírásokra.

A taneszközök fejlődése, a számítógépek iskolai alkalmazásának elterjedése ma már új feladatok elé állítja a tanárokat és a diákokat is. Az új tanévben használható oktatóprogramok már kaphatók a Tudományszervezési és Informatikai Intézetnél.

Intézetünk HT 1080Z, Videoton TV-Computer, Commodore Plus/4, illetve Commodore 64 gépen futtatható programokat kínál kedvezményesen számotokra.

Aki a felsorolt programok közül vásárol, programonként 10%, tiznél több program esetén 20% kedvezményt kap.

Commodore Plus/4-en futtatható oktatóprogramok:

| PROGRAMNÉV | | ÁR | IDEGEN NYELV | |
|-------------------|----------------------------------|----------|------------------------|---|
| MATEMATIKA | | | IN/20 | RJAK 313,0 Ft |
| MA/10 | Számelmélet | 313,0 Ft | IN/21 | Angol erős igék 313,0 Ft |
| MA/11 | Egyenlet | 313,0 Ft | IN/22 | Keyword 313,0 Ft |
| MA/12 | Kombinatorika | 313,0 Ft | IN/24 | Úrcsata 313,0 Ft |
| MA/13 | Függvény és transzformáció | 313,0 Ft | IN/25 | Játsszunk a szavakkal 313,0 Ft |
| MA/23 | Gyökkereső | 313,0 Ft | FÖLDRAJZ | |
| MA/24 | Galton | 313,0 Ft | FŐ/03—04 | SZU—USA 625,0 Ft |
| MA/26 | Prim | 313,0 Ft | TÖRTÉNELEM | |
| MA/28 | Rajzolóprogram | 375,0 Ft | TŐ/13 | Magyarország felszabadítása 1—2. 438,0 Ft |
| MA/29 | Egyváltozós függvény | 375,0 Ft | MAGYAR IRODALOM | |
| MA/30 | Vektor skalár függvény | 375,0 Ft | Mi/02 | Vajh, ki ő és merre van hazája 313,0 Ft |
| MA/30 | Primitivék | 375,0 Ft | MAGYAR NYELVTAN | |
| MA/48 | Négyszögek | 313,0 Ft | MN/35 | Helyesírás 375,0 Ft |
| MA/50 | Háromszög koordináta geometriája | 313,0 Ft | SZAKKÖZÉPISKOLA | |
| MA/52 | Függvények | 313,0 Ft | SK/36 | Környezeti tényezők szerepe 313,0 Ft |
| MA/58 | Parabola | 313,0 Ft | SK/37 | Zöldsefélek csoportosítása 313,0 Ft |
| MA/100 | Korrelációk | 313,0 Ft | SK/38 | Zöldsefélek származása 313,0 Ft |
| MA/101 | Számrendszerek | 375,0 Ft | SK/39 | Összefoglaló-barkochba 313,0 Ft |
| MA/102 | Függvény | 375,0 Ft | EGYÉB | |
| FIZIKA | | | EG/3 | Atlétikai teszt 500,0 Ft |
| FI/03 | Soros RC | 313,0 Ft | EG/04 | Ki mit tud? 313,0 Ft |
| FI/73 | Eredő 1 | 313,0 Ft | Eg/10 | UNIFEL 313,0 Ft |
| FI/74 | Eredő 3 | 313,0 Ft | JÁTÉK | |
| FI/92 | Mágneses mező | 313,0 Ft | JA/10 | Memóriateszt 313,0 Ft |
| FI/96 | Súlypont | 313,0 Ft | | |
| KÉMIA | | | | |
| KÉ/18 | Sűrűség | 313,0 Ft | | |
| KÉ/19 | Tömeg | 313,0 Ft | | |
| KÉ/20 | Mólók | 313,0 Ft | | |

Áraink a 25% áfa-t is tartalmazzák.

Címünk: Tudományszervezési és Informatikai Intézet

1111 Budapest XI., Egy József u. 1—9.

BME „E” épület XI. emelet.

Postai cím: Budapest, Pf.: 454. 1372

A TUDOMÁNYSZERVEZÉSI ÉS INFORMATIKAI INTÉZET
előzetes megbeszélés szerint díjmentes programbemutatót tart
(vidéken is) az általa forgalmazott oktatóprogramokból.
Horváth Zsuzsa 665-011/2663 mellék vagy 813-197
Budapest, Pf. 454. 1372

PROGRAMTERMÉK

ÚJ ASSEMBLER A TVC-RE

Ha új, akkor bizonyára régebbi is volt. Most, hogy éppen nem egy oktató-programot elemzek — aminek az az oka, hogy a szerkesztésben a kezembe nyomtak egy programot, nézzem meg —, töprengök, hogyan is fogjak hozzá. Kérdősködés után kiderült, hogy VIDAS néven már a Videoton is kínálja assemblerét a gépéhez. Kézlenkvő tehát, hogy összehasonlítsuk az újat azzal, ami van.

A program készítője kisvállalkozó, aki termékek a sok mikrogépes vállalkozást igen dinamikusan támogató Magiszter Könyvesbolt útján terjeszti.

A program leírása bizony elég vékonyka, talán azért is, hogy elférjen a kezeltartatóban. A VIDAS leírása sem több. Megtaláltam viszont egy olyan Z80 assembler leírását, amit korábban Spectrum-on használtam. Ez a SPECTRE—MAC—MON, mely fullscreen editorral integrált mikroassembler. A leírása majdnem harminc oldal. A szövegek körülbelül arányosak a mondanivalóval. A VIDAS leírása csak azért olyan hosszú, mert tartalmaz egy mnemonik táblázatot, egyébként az összehasonlítottak közül a legkevesebb funkció található benne. A programot lényegében a végleges címre helyezi, ami a lehetőségeit korlátozza. Az „új” termék ennél többet tud. A LOAD funkcióval utasítjuk, hogy a végleges cím hol legyen.

A VIDAS ténylegesen csak assembler, igaz, annak elég gyors. Az „új” termék viszont, a SPECTRE—MAC—MON-hoz hasonlóan, komplett szoftverfejlesztő rendszer. Ezzel a tulajdonságával hasonlít a Borland cég turbo szoftvereinek filozófiájához, ezért elég korszerű programnak ítélem. Ha a tárgyeket hasonlítjuk össze, az eltérés szintén szembeötlő. A VIDAS 32 k-s gépen is működik, míg az „új” termék csak 64 k-s gépen tud futni. A szolgáltatásaihoz tehát igen sok tárat elrabol. Ezt nyugodtan felróhatjuk neki, hiszen a több funkciót ismerő SPECTRE—MAC—MON mindössze 16 k-t igé-

nyel, pedig nem a Spectrum BASIC editorára hivatkozik, mint akár a VIDAS, akár a TV-Computer Assembler. A nyelvi elemek tekintetében a VIDAS-ban a gépi kód mnemonikáin kívül csak elemi utasításokat használhatunk (ORG, EQU, DB, END). Az „új” termék a már említett LOAD áthelyező utasításon kívül csak néhány újabb adatmezőtípust definiáló utasítást ismer. Ezzel szemben a SPECTRE—MAC—MON ismeri a feltételes fordítás lehetőségét — beleértve a ciklusokat is —, és a szokásos assembler direktívákon túl, mint neve is jelzi, mikroprocesszor-lehetőségeket is tartalmaz.

A SPECTRE—MAC—MON-nal akár újradefiniálhatjuk a kódgenerátor-funkciót is. Amikor használtam, éppen alkalmam nyílt ennek a lehetőségnek a kipróbálására. A programbelőveknél azt vettem észre, hogy a programomban az egyik ADC utasítás helyett ADD van. Nem így a forráskódban. Nos, meglepődtem egy kicsit, mert hosszú pályafutásom alatt assemblert még nem csíptem rajta hibás működésen. A leírás és a programhoz mellékelte forrásnyelvi makrokönyvtár elemzésével sikerült kiderítenem, hogy az ADC

neve nem volt felsorolva az egyik tulajdonságlistán. Ennek kijavításával a programot újrageneráltam, és lássunk csodát, a programhibám megszűnt. A „szerszámom” alkalmas volt arra, hogy még önmagából is kiköszörülhető legyen a csorba.

Mellesleg a forrás- és lefordított program mentő és ellenőrző-betöltő kódreszei egy BASIC modulban voltak. Ez lehetővé tette, hogy a SPECTRE—MAC—MON-t meghajtó helyett át lehessen helyezni 3,5"-os hajlékonylemezre. Ebben a konfigurációban egy Spectrum 128 k-s gépen a meghajtóval elképzelhetetlen fejlesztési sebességet lehetett elérni.

Az „új” termék végül is tartalmaz assembler, disassembler, programlistázási, nyomtatási, monitor (adatbeírás, tárlistázási, tármásolási, tármozgatási), programbelővesi (regiszterállapot, töréspontkezelés, lépéskénti végrehajtás, elindítás) és egyéb funkciókat (munkaesztét határai, számrendszeralapszám-váltás, kimentés-betöltés a forrás- és tárgykódra).

A használat közben gyorsan visszatérhetünk a BASIC editorba javítás és programbevitel céljából. A program igen jól használható fejlesztő eszköz, még középiskolai iskolagép-környezetben is többet nyújt annál, ami igazán kihasználható. Kevésbé használható profi fejlesztéseknél. Ilyen célra a SPECTRE—MAC—MON-féle megoldás lehet csak igazán jó, aminél színtaktikus vagy címhivatkozási hibánál az assembly leállításakor a kurzor a hibás ponton villog. Ezenkívül legalább a feltételes fordítás az, ami egy profi fejlesztésnél nélkülözhetetlen. Hogy ne is elemezzük a keresztfejlesztést lehetővé tevő mikroprocesszor szükségességét. A SPECTRE—MAC—MON talán működne a TVC testvérgépén, az Enterprise-on, Spectrum emulátorral. Ki kellene próbálni, ha még nem tette meg valaki.

Az „új” termék tehát elég korszerű elveken alapuló, jó program, amely iskolagép-környezetben feltétlenül, profi célokra kevésbé, de még így is javasolható.

ZSADÁNYI PÁL

ÖSSZEFOGLALÓ ADATOK

| | |
|---------------|---|
| Forgalmazó: | Magiszter Könyvesbolt Bp. V., Városház u. 1. |
| Terméknév: | TV-Computer Assembler |
| Szerző: | Kós Géza |
| Géptípus: | TVC |
| Hordozó: | kazetta |
| Dokumentáció: | 12 oldalas, a kazettatartóban |
| Ár: | 472 Ft (ÁFA-val) |

MINŐSÍTŐ ADATOK

| | |
|------------------|--------|
| Kezelhetőség: | jó |
| Teljesseg: | jó |
| Dokumentáltság: | jó |
| Ár/teljesítmény: | kiváló |
| Összbenyomás: | jó |

Yoneji Masuda:
Az információs társadalom
 (Budapest, 1988.
 OMIKK,
 156 oldal.
 Fordította: Hámori Ferenc.)

A japán professzor könyvvé formált látomásában megjelenő „információs társadalom” nem más, mint az ún. posztindusztriális társadalom általa elképzelt változata. Legsajátosabb jellemzője az, hogy tagjai magas szintű intellektuális kreativitás birtokában vannak, hiszen a legmagasabb szinten rendelkezésükre áll minden, a számítástechnika által kínált innovációs lehetőség.

A látomás lenyűgöző! Egészen utópikus fantaszitumnak tűnik ugyan, bár bizonyos részletei a világban már itt-ott kirajzolódni látszanak. Masuda professzor maga is ismert néhány olyan projektet, amely Japánban, Kanadában, ill. Svédországban a 70-es évek óta működik a közigazgatásban, az oktatásban, a közlekedésben és az egészségügyben. A körvonalakhoz sorolhatók a különböző információs szolgáltató rendszerek is — a kábeltele-

víziótól a teletexen át az oktatási és egészségügyi programokig. De vajon mikor jöhet létre teljességében ez a társadalom? A számítások szerint az információs forradalom beteljesüléséhez 64 év elegendő lesz. S 2010 már nincs is olyan messze!

Félő azonban, hogy a további fejlődés nem lesz olyan kiegyensúlyozott, mint amilyennek Masuda feltételezi. Annál is inkább, mert mintha egyáltalán nem számolna — egyebek között — a világ jelenlegi politikai megosztottságával. De azzal sem érthetünk egyet, hogy ő az emberi kiteljesedés kizárólagos színterének a szabadidőt tekinti; a munka világában csak „nyűgöt” lát. Hiszen dolgozni a legutópszintikusabb jövőben is kell majd — hogyan is mondhatna le az ember éppen a munka további humanizálásáról?

Garai Géza:
TV-Computer operációs rendszer
 (Budapest, 1988.
 Novotrade, 135 oldal.
 Ára: 190,— Ft.)

A kötet a Videoton TV-Computer gépi kódú programozásához szükséges ismereteket tartal-

mazza és segítséget nyújt a BASIC lehetőségeinek jobb kihasználásához is. Foglalkozik a memóriakiosztással, ismerteti a rendszerváltozókat, az operációs rendszer eszközmeghajtóit és azok funkcióit. Az olvasó a programozói munka segítésére hardverismereteket is talál a könyvben.

Külön fejezet dolgozza fel a mikroprocesszorokat és a memóriát, a megszakításkezelést, a be/kiviteli rendszert, a rendszerváltozókat, a kernelhívásokat, a képmegjelenítést, a billentyűzetet, a képernyőszerkesztőt, a nyomtatókezelést és a kazettakezelést.

Purgathofer, Werner:
Grafikai adatok számítógépes feldolgozása.
 (Budapest, 1988.
 Műszaki Könyvkiadó,
 155 oldal. Ára: 98,— Ft.)

A kötet szerzője nem átfogó kézikönyvet kívánt megjelentetni a grafikus adatfeldolgozásról, hanem alapos áttekintést szeretett volna adni a témáról és fontosabb alapfogalmairól. Arra törekedett,

ADOK—VESZEK

Ebben a rovatban rövid, szöveges, a mikroszámítógépekkel kapcsolatos hirdetéseket közlünk. A díjszabás: közületeknek gépelt soronként (60 karakter) 100,- Ft, magánszemélyeknek az első sor 50,- Ft, minden további sor 20,- Ft. Az NJSZT tagjainak az első három sor ingyenes. Hirdetéseiket a szerkesztőség címére várjuk.

ADOK

Atari program idegen nyelvek tanulásának megkönnyítéséhez. Válaszboríték ellenében ingyenes tájékoztató! Dr. Gerő László, Szeged, Budapesti krt. 4/b. VIII/31. 6723

Commodore 128D eladó. Árajánlatokat a következő címre kérek: Donga György, Budapest, Balzac u. 39. 1136

C16, C Plus/4, C64-es programok eladók, 10 Ft/db utánvétellel. Tel.: 429-168

C16, Plus/4, C64 programok eladók kazettánként 40-50 program. Egy kazetta 500 Ft. A programok külön is megvehető 10 Ft/db. Listát előre küldök. Válaszokat válaszborítékkal a következő címre kérek: Szilágyi Imre, Kisvárdány, Rákóczi u. 53. 4811

Commodore 64-hez GYORS-HÁTTÉRTÁR cartridge. Kapacitása 2-31 kb-át. A GYORS-HÁTTÉRTÁR-ba maximum 7 db, célszerűen gyakran használt program vihető be. A gép bekapcsolása után menüvel jelentkezik be, és gombnyomásra a kiválasztott program azonnal fut. Javasolt programcsomagok: Turbo tape, Assembler, Monitor, Supergrafik, Help plus, Turbo másoló. File másoló 1699 Ft és Turbo tape 699 Ft.

Hozott programok elhelyezése a GYORS-HÁTTÉRTÁR-ban. Trompler László, Budapest, Attila u. 22. 1201. Tel.: 287-493 este.

Commodore 64 (új típusú) számítógép magnóval, joystick-kal, német nyelvű kézikönyvekkel és sok minőségi programmal eladó. Ár: 20 000,- Ft. Oláh István, Szolnok, Kassai u. 56. I/4. 5000 Tel.: 56/15-496

C64 + VC 1541 + 1 db joy. + programok lemezen eladók. Árajánlatokat levélben várok. Irimi János, Hajdunánás, Ady E. krt. 33. I/6. 4080

C64-es programkazettáimat és lemezeimet eladnám. Magyar Attila, Kapuvár, Lenin u. 10. 9330

Commodore 64 (új típusú), magnó és 2 db joystick eladó. 653-742 este 6-9-ig.

10 db C64-es programkazettámat 510 db programmal eladom. Egyben 4500,- Ft, esetleg 5-5 db 2500,- Ft-ért. Esetleg videokazettára, illetve floppy lemezekre cserélem. Nagy Márta, Mezőkövesd, Szihalmi u. 1. 3400

C64, 1541-es floppy, cartridge, 50 diszk a legújabb játék- és felhasználói programokkal, 2 joystick sürgősen, kedvező áron eladó. Beregi, Budapest, II. Törökvesz u. 128. vagy 150-149 este 6 után

C64-es új számítógépet eladom tartozékokkal + TV-Computer (64 k) eladó 200 programmal. Hata József, Tiszaörs, Lenin u. 62. 5262

C64-es programokat adok-cserélek. Új 88-as játékok nagy választékban. Várnai Miklós, Pécs, Egri Gyula u. 84. 7632

C64 (15 000), VC 1541(18 000), 2 db holland mikrokapcsolós joystick (4000), 26 lemez játék+szoftver (4000), szakönyvek (3000), összes Mikro Magazin (1500), együtt 40 000,- Ft-ért eladó. Sárosi Sándor, Környe, Zsíros u. 1/a. 2851, Tel.: 34/72-044 este

Computer Persönlich Magazin 87/88 összes száma eladó: 661-677

Figyelem! Programsegítő modulok, FAST-LOAD, SPEEDTAPE cartridge-k kaphatók C64 számítógéphez. Vállalom továbbá bővített karakterkészlet, Reset gomb, IRQ (Pause) kapcsoló beépítését, Seikosha SP1200VC nyomtató magyarosítását, bioritmus és horoszkóp készítését. Szívesen küldök tájékoztatót! Bártfai Barnabás, Budapest, Móricz Zs. u. 22. 1193

Primo számítógéphez készült új, gépi kódú játékprogramok eladók 170-370 Ft-os áron. A programokat postán, utánvétellel küldöm, esetleg személyesen az esti órákban. Válaszboríték ellenében ingyenes tájékoztatót küldök. Paller Gábor, Budapest, Varga Gy. park 8/a. 1149

Primo A64 tápegységgel, kazettákkal, könyvekkel 6000 Ft-ért eladó. ZX-Spectrum programcsere (keresem a THE

LAST NINJA-t). Szántó Péter, Gyömrő, Bercsényi u. 20. 2230

PRINTFOX, CHARAKTERFOX, LUPEFOX és GEOS V1.3 magyar nyelvű dokumentáció eladó. Honti József, Csákvár, Május 1. u. 11. 8083

Teljesen új mikrokapcsolós QuickShot IX jogball eladó. Állítható jobb és bal kézre ill. autofire. Irányár: 3000 Ft. Egri Imre, Békésszentandrás, Pacsirta u. 6. 5561

TV-Computer programokat adok, veszek cserélek. Érdeklődni levélben vagy személyesen vasárnap 9-11 óra között. Csatlós Béla, Mezőtúr, Ifjúsági ltp. XIX. ép. II. lh. 1/3. 5400

Spectrum+ számítógép többszáz programmal, fényceruzával és Enterprise számítógép programokkal sürgősen, olcsón eladó. Kristóf Attila, Teskánd, Petőfi u. 49. 8991

Spectrum (48 k) 10 000 Ft-ért, Commodore 64/128 1541 Mouse 5000 Ft-ért eladó. Magyar Elek, Kecskemét, Dohnányi Ernő u. 54. 6000

Sinclair (128 k) ZX-Spectrum + 2 típusú komplett számítógép eladó. Dr. Hajas hétköznap este 895-960, napközben 632-070

Spectrumosok! A Zotyocopy forrása a készítőtl! 200 Ft utánvétellel. Kazetta szükséges. Graff Zoltán, Budapest, Mester u. 61. II/14. 1095

Spectrum (48/128 k) programkazetták a legújabb angol slágerlistákról eladók. Tájékoztatót válaszborítékban küldök. Horváth Péter, Siklós, Pf.: 129. 7800

Spectrumosok figyelme! Eladó néhány 5 1/4-es, 1x 35 track-es MOM floppy drive eredeti gyári csomagolásban, eredeti áron, 3500 Ft-ért. Illesztő építésben is tudunk segíteni! Érdeklődni: Bécsi út - Vörösvári út sarki Virágüzletben (VOLÁN buszvégállomásnál).

ZX-Spectrum (TV hanggal) + 2-es interfész + 300 program eladó. Irányár: 13 500 Ft. Molnár László, Martfű, Lenin u. 18. 5435

ZX-Spectrum szuper (80 k), floppy + Specy Dos interfész, magnó, irodalom együtt vagy külön eladó. Érdeklődni lehet: Varga József, Kerepestarcsa, Szabadság u. 162. 2144

CATS

Nemcsak hát, hanem számítógéppel támogatott tesztelési rendszert (Computer Aided Testing System — CATS) fejlesztett ki a Logos nyelviskola. A lényege, hogy egy „üres” tesztvizsgájműtemény hoztak létre, melyben az egyes tesztek meghatározott számú tesztdeszköz tartozik, továbbá előre rögzítettek, hogy az egyes kérdésekre hány válasz közül választhat a tanuló, s ezen belül melyik válasz a helyes. Az egyes tesztszámoknak azonosított jelűk van, azaz befejezésükkel felhívhatók.

A tesztvizsgá viszonylag egyszerű munkával gyorsan feltölthető bármilyen témájú anyaggal. Például az International House Language School angol, német, olasz nyelvismeret faggató anyaggal töltötte fel a teszteket, és előkészületben vannak a francia, orosz tesztvizsgák is. Ezeket a fűzeteket bárki egyszerűen, akár önállóan is használhatja. Természetesen semmi akadály nincs annak sem, hogy a nyelvtanulástól távol, akár Kesznyeggyal töltsük fel a vizsgázókat.

A program PTA zsebszámológépre, Commodore 64-re és IBM PC-re készült el. Meghökkenítő az alacsony ára is: a Commodore 64-es változat tesztenként 600, az IBM PC-s változat pedig 900 forintba kerül.

Két gép az NDK-ból

Megjelentek az első, európai szocialista országokból származó IBM PC-vel kompatibilis mikroszámítógépek a hazai piacon. A Migért által kínált A 7150 és ESZ 1834 közös jellemzője, hogy az NDK-beli Robotron kombinát készítette, mikroprocesszoruk az Intel 8086 szöveg megfelelője, 40 Mbjáts winchester tárat tartalmaznak s a hazai árákhoz képest igen olcsók: a nyári induló áruk 180 ezer forint volt.



A Robotron A 7150 típusú mikroszámítógép

Karácsonyi ajánlat

Commodore és Enterprise számítógépek tulajdonosai részére:

| | | |
|--|--|--------------------------|
| Fényceruza C64-hez. | C128-hoz | 1000 Ft |
| Fényceruza C16-hoz. | Plus/4-hez | 1500 Ft |
| Fényceruza EPROM kártyás kezelő-programmal | C16-hoz és C Plus/4-hez. | alkalmazói programokkal, |
| kazettán | | 2000 Ft |
| Joystick (C64, C Plus/4, C16 és Enterprise típusokhoz) | | 625 Ft |
| Monitor-adapter JUNOSZTY tv-hoz | (hang és minőségi kép, hátlagos szerelhetőség) | 400 Ft |

Szállítás postai utánvétellel.

Cím: COMPUTEAM GM., Kaposvár, Berzsenyi u. 32. 7400

hogyan ezt a szerteágazó területet, egyszerűen, jól érthetően mutassa be. A szerző feltételezi az általános adatfeldolgozás alapfogalmainak ismeretét, viszont részletesen tárgyalja a grafikus adatfeldolgozás alapfogalmait. Igézkedett egyszerűen bemutatni a nagyobb összefüggéseket, ami remélhetően segíti egy-egy ítmakör jobb megértését. A szövegben nem hivatkozik a szakirodalomra, de a függelékben megtalálhatók a kapcsolódó könyvek és folyóiratok címei. A kötet rövid idő alatt két kiadást ért meg.

ZSEBKÖNYVTÁR

Napjainkban egyre gyorsabban terjednek az adattárolás új eszközei, az optikai lemezek. Közülük is leggyorsabban a CD-ROM (Compact Disc — Read Only Memory). Ez voltalképpen a kompakt hanglemezeknek a számítógépes adatok rögzítő „testvére”. Sokszorosítása is a kompakt hanglemezekhez hasonló: felhasználója

téhat írni nem tud rá, hanem csak leolvashatja a gyárilag ráírt információit. Eppen ezért, mert nem változtatható, ideális eszköz arra, hogy kiadóvállalatok számítógéppel olvasható adatbázisokat, kezékönyveket, szótárakat, lexikonokat, programcsomagokat, oktatási csomagokat forgalmazzanak rajta. Tárolási kapacitása óriási: egy-egy 12 centiméter átmérőjű lemezen mintegy felmilleió megabájt — azaz mintegy negyedmilleió oldal — hasznos információ fér el. A CD-ROM lemezek használatához be kell szerezni egy CD-ROM olvasóberendezést, ami mikroszámítógépekhez csatlakoztatható. Idei amerikai újdonság egy hordozható változat megjelenése: a mikroszámítógép és a CD-ROM-olvasó egy nagyobb akatáskájában is elfér és persze szállítható. Ez a méretcsökkenés már jelentősen kiterjeszti a CD-ROM lemezes kiadványok használhatóságát. Hazánkban is már folyókn az előkészületek a CD-ROM széles körű alkalmazására. A CD-lejátszó és egy digitális optikai tárolással kifejlesztéséhez és gyártásának előkészítéséhez a Videoton licenct vásárolt a Thomson cégtől.

— CSERÉLEK

ZX-Spectrumba játé- és felhasználói programok előadók. Válaszborítékért katalógust küldök. Bogóv Frigyes, Pécs, Bihar J. u. 3/B. f.s.z. 1. 7633

Személyi számítógépek illeszthető lemez meghajtó egység eladó. Nagy Jenő, Győr, Tákó u. 2/A. 9025

VC 1541 eladó. Bölöni Tibor, Várpalota, Hősök tere 2. I/3. 8100

CSERÉLEK

C16, Plus/4 felhasználói- és játéprogramokat cserélek. Listát kérek! Csókas Csaba, Tamási, Rákóczi u. 47. 7090

C64-es programokat adok-veszek-cserélek kazettán, játé- és felhasználói programok egyaránt érdekelnek. Rácz Balázs, Budapest, Kulpa u. 12. 1025

C64-es programcsere kazettán. Bányai Attila, Iatabánya II. ker. Banyai János u. 3. 2800

C64-es programokat cserélek kazettán. Listát kérek! Hábel Péter, Szolnok, Verseghy u. 38. 5000

C64-es programokat cserélek kazettán. Listát kérek! Bernáth Csaba, Eger, Meder u. 3. 3300

Enterprise programcsere. Azok is írhatnak, akiknek kevés programjuk van. Listát kérek! Tóth Zoltán, Cegléd, Sas u. 9. 2700

ZX-Spectrum (48 k) programokat cserélek. Válaszokat kizárólag levélben a következő címre várom: Vágner Gyula, Budapest, Hidegkúti u. 80/a. 1028. Kérem, hogy programlistával válaszoljanak!

ZX-Spectrumba színvonalas programokat cserélek. Válaszokat listával kérem! Batkó Péter, Jászberény, Sárkány u. 1. 5100

Színvonalas Spectrum programokat cserélek. Válaszokat listával kérem! Berecz Gábor, Miskolc, Engels u. 38. 3529

Spectrum (48 k) játéprogramokat cserélek. Listát kérek! Dávid Robert, Bányaterenyé, Bányász u. 13. 1/10. 3070

TVC (64 k) tulajdonosokkal levelezni, programokat cserélnék. Listát kérek! Molnár János, Szolnok, Jászi F. u. 10. 1/25. 5000

TVC-n kb. 450 programmal rendelkezem, hasonló cserepartnereket keresek, listát kérek. Dubrovics Zsolt, Sopron, Frankenburgi u. 48. 9400

Videoton TV-Computerhez játéprogramokat cserélnék! Majer László, Zirc, Akácfa u. 23. 8420

Ez a rovatunk KODEX 2000 szövegszerkesztővel készült.

VESEK

Color Geine, Ge 2000-es számítógéphez gépkönyvet (programozási segédletet) vennék, vagy hosszabb időre kölcsönkérnék. Bokányi József, Zalaegerszeg, Hegyalja u. 58. II/7. 8900

C64 játé- és felhasználói programokhoz keresek leírásokat, programozási ötleteket. Meggyezésem levélben. Dukán Zoltán, Sopron, Laktanya u. 20. 9400

1985 előtt gyártott Commodore 64-est veszek. Ormos Zoltán, Budapest 4, 1364 postán maradó

1541 C diszket vennék. Keresem a Microcompiler valamint a Petspeed használati utasítását. Tel.: 202-155 18 óra után

Enterprise gépi kódú programokat vennék. Patek Alajos, Budapest, Faludi u. 28. 1131, Tel.: 291-483

Enterprise programokat veszek, cserélek, keresem a Magic Ball programot. Daróczi Tibor, Tát, Árpád u. 57. 2534

Keresem a Mikroszámítógép Magazin 1983-1986 évfolyamok számait. TURBOPASCAL 4.0 (estleg 3.0) leírást (angol vagy magyar) vásárolnék, vagy másolnék. Csákvári Zoltán, Gúzfő Gyártelep, Gagarin u. 22. 8184

Primára írt, kazettán leforgatható játéprogramokat veszek. Szele István, Mohács, Jókai u. 1. I/3. 7700

Primához használható sornymatatót vásárolnék. (DATACODP, MOM, TMT, FT, KX típusokat). Horváth László 583-544 délelőtt

Sinclair ZX-Spectrum+ (128 k) géphez játé- és felhasználói programokat vennék. Hencz Attila, Győrsövényház, Gárdonyi u. 41. 9161

Rejtvényorozatunk minden fordulójában két feladatot közlünk: az első logikai, matematikai tudást, a második számítástechnikai alapismereteket is igényel.

A feladatok után közöljük az elérhető maximális pontszámot. A részmegoldásokat is pontozzuk.

Ezzel a feladatokkal a Pontvadászat-sorozatot lezárjuk. A legjobb tíz versenyző nevét magazinunkban közzé is tesszük, ők lesznek azok, akik könyvtulványt is kapnak.

A helyes megoldások a feladatok közreadása után egy lap-számmal később jelennek meg. Ezek után kíváncsian várhatják a nyertesek névsorát.

Beküldési határidő: 1988. december 31.

Címünk: Mikroszámítógép Magazin Szerkesztősége

1371 Budapest, Pf. 433.

Az 1988/6. szám 1. és 2. feladatára 12-12 megfejtés érkezett be. Ezek közül az 1. feladat megfejtési százaléká, a maximális 4 pontot véve 100 százalékának, 91,7 százalék, a 2. feladaté, a maximális 16 pontot véve 100 százalékának, 91,1 százalék volt.

Jó vadászatot kíván a feladatok összeállítója:

dr. Hoffmann Tibor

1. feladat

Egy képzeletbeli tengerben van egy A típusú és egy B típusú hal. Az A típusú hal más táplálékon kívül előszerrel fogyasztja a B típusú halat, mégpedig úgy, hogy ahhoz, hogy életben maradjon, négyhetente 5 darab B típusú halat kell fel-falnia. Ha életben van, akkor egyhetes érési idővel minden második héten minden A hal további 1 hallal szaporodik.

A B típusú halak egyéb kis halakkal táplálkoznak, melyek gyakorlatilag kimeríthetetlen mennyiségben találhatóak a tengerben. A B típusú halak érési idő nélkül, hetenként 2 új hallal szaporodnak.

a) *Kérdés: hogyan változik az A és B típusú halak aránya, ha a folyamat kezdetén ugyanannyi A és B típusú hal volt a tengerben?* (8 pont)

b) *Hogyan változik a fenti folyamat, ha a B típusú halak szaporodása csak hetenként 1 új hal?* (4 pont)

2. feladat

NO egy adott nagyon nagy egész és X egy 1-nél nagyobb szám. Tekintsük a következő FORTRAN szubrutint:

FUNCTION F(X)

F=0

X=(X-1)/(X+1)

DO 1 I=1, NO

J=2I-1

G=Y**J/J

F=F+G

IF(G-10**(-9)) 2, 1, 1

1 CONTINUE

2 F=2 F

RETURN

END

Milyen függvényt ad meg F(X)? (5 pont)

Az 1988/10. szám feladatainak megoldása

1. feladat

Célszerű a számítást úgy végezni, hogy egy alapszázalékhoz sávonként megadjuk, hogy mennyi additív százalékot kell hozzátennünk. Az adósávok alsó határát a_i -vel jelölve, és p_i -vel jelölve a sávban az adó százalékos növekedését, ez a következőképpen írható fel:

$$\left[1 - \frac{a_i}{a} + \left|1 - \frac{a_i}{a}\right| \frac{p_i}{2}\right]$$

ugyanis a szögletes zárójelen belüli kifejezés θ , ha $a < a_i$, míg minden más esetben a mellette lévő faktorról szorozva a százalékos effektív növekedése.

Látható, hogy a szögletes zárójel növekvő a -val alulról közelíti meg a 2-t, és így a százalék növekedése alulról közelíti meg a p_i -t, de azt soha el nem éri. A most bevezetett értékekkel ez a százalékos adókulcsot bármely a -ra a következő képlettel adhatjuk meg (itt a lehetséges összevonásokat elvégeztük):

$$p = 30 - \frac{5930000}{a} + 10 \left[1 - \frac{48000}{a}\right] + 2,5 \left(\left[1 - \frac{70000}{a}\right] + \left[1 - \frac{90000}{a}\right] + \left[1 - \frac{120000}{a}\right] + \left[1 - \frac{180000}{a}\right]\right) + 2 \left(\left[1 - \frac{150000}{a}\right] + \left[1 - \frac{240000}{a}\right] + \left[1 - \frac{360000}{a}\right] + \left[1 - \frac{600000}{a}\right] + \left[1 - \frac{800000}{a}\right]\right)$$

Látható, hogy a progresszivitásban kis inkonzekvencia van a 180 000 és 240 000 Ft közötti évi adóalap esetében (egyenletesen elosztva havi 15 000 és 20 000 Ft között), ahol jobban emelkedik az adókulcs a 150 000 Ft és 180 000 Ft közöttihez (havi 12 500 Ft és 15 000 Ft között) képest, noha ez a következő sávban is ellenkezőleg van. Ennek nem világos az oka. (5 pont)

2. feladat

A bitek számának a megduplázásával, vagyis $2(n+1)$ bittel tulajdonképpen kétszer adjuk le ugyanazt a kódot. Ha csak egyetlen hibát veszünk tekintetbe, akkor a feladat megadásában vázolt hibajelző kóddal az első vagy a második kódfélben jelez hibát a rendszer és akkor a második vagy az első kódfél korrekt. (8 pont)

A gyakorlatban vannak ennél kevesebb bit továbbítását igénylő, és ennek megfelelően komplikáltabb korrekciós módszerek is.

A SKÁLA METRÓ ÁRUHÁZ
METRO-COMP üzlete
*kiváló minőségű számítástechnikai és híradástechnikai cikkek
gazdag választékát kínálja.*



Ajánlatunkból:

*Parrot floppy lemezek angol importból.
Kiváló minőségben.*

mérsékelt áron, széles választékban kaphatók:

| | |
|-----------------------------|---------------------|
| 5.25" – DS.DD 48 tpi 180 Ft | 3.5" – DS.DD 400 Ft |
| – SS.DD 48 tpi 160 Ft | 8" – SS.DD 220 Ft |
| – DS.DD 96 tpi 220 Ft | – SS.DD 240 Ft |
| High Capacity 400 Ft | – DS.DD 280 Ft |

*Egyéni és közületi vásárlóinkat egyaránt várjuk a
Budapest VI., Jókai u. 40. szám alatti METRO-COMP üzletünkbe.*

Telefon: 127-442 • Telex: 22 7562

VIDEOTON

VT110, VT160, VT180

professzionális személyi számítógépek

A család tagjainak közös jellemzői:

- Az információk max. 4 db háttértárolón rögzíthetők. Ezek lehetnek: floppy diszk, Winchester diszk vagy streamer tárolók.
- A megjelenítés monochrom vagy színes grafikus monitoron történhet.
- Billentyűzetek: 84 nyomógombos PC kompatibilis vagy 113 nyomógombos magyar ékezetes taszt.
- Ajánlott felhasználási területek: adatheldolgozás ügyvitel szövegszerkesztés és feldolgozás műszaki-tudományos számítások grafikus alkalmazások oktatás
- Lokális hálózati alkalmazások: Ethernet, Arcnet
- Software: MS-DOS 3.2 operációs rendszer GWBASIC felhasználói programcsomagok nagy választéka

