

računari

Specijalno izdanje časopisa „Galaksija“
izdaje BIGZ – OOUR „Duga“

11

cena
250
dinara
izlazi
jedanput
mesečno
januar 1986.

- *hard diskovi*
- *video digitajzer*
- *hakerski marketing*
- *adresar profi klubova*



operativni sistemi „amstradov“ kernal

11

Izlazi jednom mesečno / Izdaje BIGZ OOOR „Duga“
računari
 Specijalno izdanje časopisa „Galaksija“

Cena 250 dinara / januar 1986.

Izdaje
 Beogradski izdavačko-grafički zavod
 OOOR Novinska delatnost „Duga“
 11000 Beograd
 Bulevar vojvode Mišica 17

Telefoni
 650-161 (redakcija)
 650-529 (prodaja)
 651-733 (propaganda)

Generalni direktor
 Dobrosav Petrović

Direktor OOOR „Duga“
 Bratoljub Babić
Glavni i odgovorni urednik
 Gavril Vučković
Urednik izdanje
 Jova Regasek

Tehnički urednik
 Mirko Popov

Redakcija časopisa „Galaksija“
 Tanasije Gavranović, pomoćnik
 glavnog i odgovornog urednika
 Esad Jakupović, zamenik glavnog
 i odgovornog urednika
 Aleksandar Milinković, urednik
 Jova Regasek, urednik
 Zorka Simović, sekretar redakcije
 Srdan Stojančević, novinar
 Gavril Vučković, glavni i odgovorni
 urednik

Stručna saradnja
 Dejan Ristanović
 Dušan Slavić
 Nevenka Spalević
 Anđelko Zgorelec
 Mihajlo Karapandžić

Autori tekstova:

Nada Aleksić
 Branko Baković
 Donat Greber
 Vladimir Kostić
 Vladimir Krstonošić
 Dejan Muhamedagić
 Blažimir Miše
 Ivan Nador
 Radomir Nikolajev
 Đorđe Janković
 Dejan Ristanović
 Zoran Obradović
 Miodrag Potkonjak
 Jelena Rupnik
 Dušan Slavić
 Nevenka Spalević
 Jovan Skuljan
 Srdan Stakić
 Mihajlo Karapandžić

Naslovna strana
 Vladimir Simović
Fotografije
 Vladimir Simović

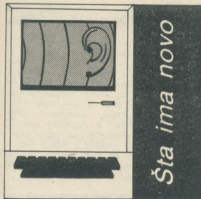
Ilustracije
 Miodrag Marković

Prevodioci
 Esad Jakupović
 Ksenija Pješčić-Lebedinski
Izdavački savet „Galaksije“
 Dr Rudi Debijadi, prof. dr Branislav Dimitrijević (predsednik), Radovan Drašković, Tanasije Gavranović, Živorad Glišić, Esad Jakupović, Velizar Masić, Nikola Pajić, Željka Perunović, prof. dr Momčilo Ristić, Vlada Ristić, dr inž. Milorad Teofilović, Vidoklo Velicković, Velimir Vesović, Milojke Vuković

Štampa
 Beogradsko izdavačko-grafički zavod
 11000 Beograd, Bulevar vojvode Mišica 17
 Žiro-račun kod SDK 60802-833-2463
 Devizni račun kod Beobanke
 60811-620-6-82701-999-01066
 Za inostranstvo cena dvostruka (400 D,
 2,50 US \$, 6,50 DM, 45 Sch, 5,50 Sfr,
 20 Firs)

Na osnovu mišljenja Republičkog sekretarijata za kulturu broj 413-77/72-03 i „Službenog glasnika“ broj 26/72, ovo izdanje oslobođeno je poreza na promet.

- 3/šta ima novo
- 6/razglednica iz Londona
- 8/kako to radimo mi
- 10/periferijska oprema
hard diskovi
- 13/peek & poke show
- 14/dejanove pitalice
spiskovi sreće
- 15/kompjuterske mozgalice
neupisivo upisivanje
- 16/naš test
video digitajzer
- 17/superkompjuteri '86
džinovi u bocli
- 20/kako postati milioner (2)
hakerski marketing
- 23/računari iz mog ugla
računari kao političko pitanje
- 24/kako to rade drugi
piratska je tuga pregolema
- 25/biblioteka knjiga
početnički za početnike
- 26/operativni sistemi
„amstradov“ kernal
- 30/naš test
podmladeni BBC
- 32/na drugi način
poker bez karata
- 33/biblioteka programa
- 37/umetnost programiranja
najkraći put
- 38/majstorije na računaru/komodor 64
srpskohrvatski bejzik
- 41/interfejsi
„komodor“ u mreži
- 44/programiranje u bejziku
matrice i nizovi
- 46/majstorije na računaru/spektrum
novi rom
- 52/radionica logičnih igara
- 55/matematički softver
hiperboličke funkcije
- 50/numerički metodi
lagranžovi polinomi
- 58/računari u poslovnoj prameni
zalihe na disketi
- 60/u svetu komponenti
- 61/udruženi programeri
profi adresar
- 62/mali oglasi
- 65/računari i igre
- 66/load „dragi računari“



Šta ima novo

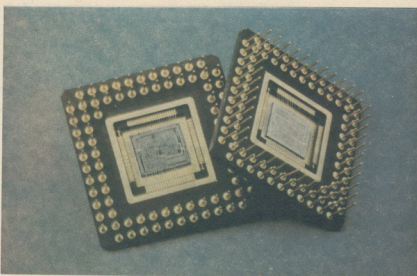
Dvoboj divova

Nismo se još ni privikli na postojanje jednog moćnog, a pristupačnog procesora kao što je MK68000, a proizvođač NATIONAL SEMICONDUCTOR se postarao da nas iznenadi novim, po ceni već pristupačnim, tridesetdvo-bitnim procesorom.

NS32332 je nastao kao logičan proizvod trke za prestiž između NATIONAL SEMICONDUCTOR-a i INTEL-a. Intel je u početku ostvario premoć svojim brzim procesorom 80386. Ova procesor je sa svojih 3-4 miliona instrukcija po sekundi (MIPS) bio oko 4 puta brži od serije 32000 National Semiconductor-a. Pojavom poboljšane verzije NS32332, ova trka za prestiž, sasvim sigurno, dobija potpuno nove okvire.

Ovoga puta pred nama je nešto sasvim izuzetno. Slično parče kristala, obradeno VLSI tehnologijom, u 84 pinskom pin-grid arandu pakovanju, može da izvrši 2,5 do 3 miliona operacija u sekundi. Za adresiranje, pristupaću mu je prostor od fantastičnih 4 G b (GIGA bajta) centralne memorije po zadatku. NS32332 poseduje potpunu tridesetdvo-bitnu unutrašnju strukturu, potpuno simetričan set instrukcija, HLL (high level languages) instrukcije sa 4 različita HLL načina adresiranja, prave registre opšte namene i podršku aritmetike pokretnog zarez. Proizvođač se, za razliku od dosadašnje prakse, postarao da zadrži potpunu kompatibilnost sa ranijom serijom, kako na gore tako i na dole. To znači da će svi programi pisani za seriju 32000 raditi i na novoj seriji 32300 i obratno. Na novom procesoru može se koristiti sav softver razvijen za seriju 32000. Ovo podrazumeva mogućnost programiranja u fortranu, pakalu, C-u i, donekle, u ADI, čija se implementacija za seriju upravo dovršava. Od operativnih sistema dostupni su AT&T UNIX SYSTEM V, Microsoft Corporation XENIX i Nacionalova verzija UNIX-a 4.1 GENIX. Svi ovi operativni sistemi obezbeđuju „pejzozvanje virtualne memorije“. Pošto MMU hardverski rešava neke od ključnih funkcija rada sa virtualnom memorijom, objektni kod, generisan kompajlerom, je za oko 20 posto kompaktniji nego kod računara VAX 11/750.

Upotrebna vrednost novog mikroprocesora, velikih performansi, je vrlo široka. Već danas, NS32000 i NS32332, sa svojim pratećim komponentama, koriste se u CAD/CAE računarskim stanicama, za konstrukciju po-



slovnih računara, u robotici, pri izgradnji multiprocesorskih sistema i tako dalje. Obzirom na cenu, koja sada iznosi oko 200 US dolara, verovatno da nije daleko dan kada će mnoštvo „komodora“, „spektruma“, „atarija“, ... biti zamenjeno mašinama sa mogućnosti bliskim računaru tipa VAX 11/750.

Radomir Nikolajev

Kućni IBM

Da li se IBM nadnosi nad tržište kućnih računara? Možda i ne, ali je zato dobroano preplašilo to tržište svojim IBM JX-om. To je računar koji je pre postojao samo u japanskoj verziji, ali je sada prepravljen na englesku varijantu i, za probu, gurnut na australsko tržište. Neki tvrde da je to samo nova varijanta „PC juniora“, dok drugi misle da se radi o dugo najavljivanom PC 2. Biće da ni jedni ni drugi nisu u pravu, iako bi bilo interesantno videti šta zaista jeste taj računar. Ništa kod IBM-a nije naivno.

Robot operator

Radio-klub Nikola Tesla iz Beograda (Timočka 18) je poznat po prvoj kompjuterskoj sekciji u Jugoslaviji: dok je 1979. godine slabo ko znao šta kućni računar uopšte može, nezamislivo broj učenika beogradskih škola (računajući i autora ovoga teksta) imao je prilike da proba svoje (programerske) sposobnosti na „pravom pravcom“ kućnom računaru — TSR 80 i.

Radio-klub Nikola Tesla je nedavno ostvario još jedan uspeh — primenio je računar u obučavanju mladih radio-amatera i tako je, u saradnji sa Poslovnom zajednicom radio-difuznih organizacija Srbije, nastao prvi robot-operator na amaterskim radio-stanicama u Jugoslaviji.

Radio-amateri dobro znaju da je za sva takmičenja bitan broj održanih veza i daljinu na koju su te veze održavane. Da bi oni podaci registrovani, na početku svakog razgovora treba razmeniti pozivne znakove, dati izveštaj o kvalitetu prijema, izračunati rastojanje i uraditi nekoliko drugih sitnica koje mladi telegrafisti redovno zaboravljaju. Ukoliko se, međutim, zadesite na frekvenci-

ji 144.075 MHz ili 3560 KHz, dočekaće vas pozivni znak YU1R — prvi amaterski robot operator. Robot će od vas zahtevati da se identifikujete i date druge protokolom predviđene podatke, a onda će vam uzvratiti ravnom merom. Zatim će podaci o vezi biti upisani u računarevu evidenciju i dopunjeni rastojanjem između vas i komputera koje će biti jednostavno izračunato prema vašem pozivnom znaku. Posle nekoliko dana ćete dobiti potvrdu o održanoj vezi koja će važiti za bilo koje takmičenje radio-amatera.

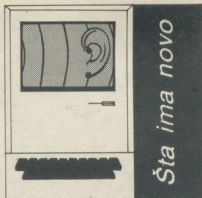
Iako je sasvim jasno da robot-operator ne može da razgovara sa vama kao Daneel Olivava, tvorcu ovog projekta možemo da uputimo samo komplimente — ing Zvonimir Makovec je našao izvanredan način za primenu kućnog računara. Ukoliko želite da porazgovarate o njegovom delu, pozivni znak ing Makoveca je YU1ZM dok je pozivni znak Radio-kluba Nikola Tesla YU1AH. Pozivni znak robota je, da ponovimo još jednom, YU1R.

Dejan Ristanović

Univerzalna pisaraća mašina

Ako ste od onih koji znaju puno jezika, ili bar jedan sa malo egzotičnijim pismom, ili, na kraju krajeva, bar cirilicu i latinicu, pa hoćete da koristite oba pisma u prepiscu, onda je vaše probleme rešio jedan od brojnih univerziteta u Bostonu. Brandeis univerzitet je realizovao prvu fazu softvera koji će omogućavati pisanje na 192 svetska jezika. Ova faza obuhvata 16 jezika, uključujući engleski, ruski, arapski i hebrejski. Program je izuzetno koristan za lingviste kojima, na primer, omogućava istovremeno prevodjenje i ispisivanje teksta na terminalu ili štampaću sa srednjovekovnog ruskog na savremeni ruski i engleski, tako što se na ekranu terminala istovremeno prikazuju sva tri teksta. Slova na tastaturi se ne menjaju pri promeni pisma, već se na terminalu pokazuje koje slovo novog pisma odgovara tipki standardne tastature. Pošto je ekran od tečnog kristala, nema ni neugodnog treperenja pri dužem radu. Sistem će se uskoro naći i u komercijalnoj prodaji i prvenstveno je namenjen političkim i komercijalnim organizacijama koje imaju intenzivne veze sa inostranstvom.

N. Aleksić



Šta ima novo

Amstrad nameravao da koristi više čipova sa manjom memorijom, ali je u toku završavanja računara došlo do naglog pojeftinjenja kapacitetnijih čipova, pa su iskorišćeni oni i zato je ostalo mesta za još nekoliko čipova. Tako, ako budete pažljivi u rukovanju iznutricama računara, možete proširiti memoriju „PCW 8256“ za 256K. Nije malo.

Na niske grane

Čirtech je izbacio pločicu sa Z-80 koja računaru „epi lic“ omogućava da se na njemu koristi CP/M Plus. Izuzetno značajan napredak! Da li to znači da možemo uskoro očekivati da Eplove računari počnu da imitiraju čak i „komodor 64“ i „spektrum“.

Proširenja RAM-a

Engleska firma DkTronics je ponudila tržištu RAM proširenja za Amstradove modele 464,664 i 6128 — preklapajuća RAM pakovanja i RAM disk.

RAM pakovanja se isporučuju u dve varijante, od 64 K i 256 K, sa cenama od 49,95 i 99,95 funti. Kao i kod CPC-a 6128, nije vam omogućeno pisanje dužih bežik programa, ali zato dodatnu memoriju možete iskoristiti za smeštanje podataka, ostavljajući time više mesta za program u glavnom RAM-u. Prateći softver uvodi desetak novih bežik naredbi za manipulisanje i pristup ovom proširenom memorije, iako je preklapanje ovih RAM blokova moguće i bez dodatnog softvera. Ovo proširenje omogućava — vlasnicima CPC-a 464 i 664 da rade sa programima pisanim i za CP/M plus.

Druga novost je silicijumski RAM disk od 256 K koji se može koristiti kao drugi ili treći disk drajv. Najveća prednost RAM diska je njegova brzina, jer će se program daleko brže izvršavati nego prilikom rada sa normalnim diskom. Upravljanje je moguće iz bežika dodatnim RSX naredbama DISCLOAD i DISCSAVE, kao i pod CP/M-om uz pomoć SETUP.COM programa. Sa držaj RAM diska, kao i memorijskih proširenja, ostaje sačuvan i nakon resetovanja računara, ali ne i nakon isključenja. Cena od 99,95 funti je dosta umerena, ali ne treba zaboraviti da se Amstradova druga normalna disk jedinica može nabaviti po istoj ceni.

D. Greber

Programi za pise

Pojavljuju se i prvi programi za Amstradov „PCW 8256“, iako su neki zlobnici tvrdili da je to, možda, zgodan aparat za obradu teksta, ali da nikad neće biti neki naročiti računari. Radi se o tekst procesorima i komunikacijskim paketima.

Više memorije za PCW 8256

Još malo o „PCW 8256“. Ako se razume u elektronicu, možete da otvorite kućište računara i tamo ćete otkriti prazna podnožja za memorijske čipove. Izgleda da je

Čipovi preko satelita

Intel je naručio izgradnju instalacije u San Huanu u Portoriku koja će omogućiti naručivanje bilo kog Intelovog čipa, koji je proizveden u bilo kom delu sveta, preko satelita za kupce širom sveta. Instalacije izgrađuju britanska firma Litton po ceni od 10 miliona dolara. Pored informacija koje interesuju kupce, Intel će i sve podatke koji omogućavaju efikasnu i ekonomičnu proizvodnju, na primer o zalihama sirovina potrebnim za proizvodnju, prenositi preko satelita do svog sedišta u Santa Klari u Kaliforniji. Tako će kompanija imati sve potrebne podatke o bilo kom od svojih 16200 proizvođača.

N. Aleksić

Poklon za igru

Cena „komodora 16“ tako naglo pada da se može očekivati da će ga uskoro davati kao poklon uz svaku igru za njega koju kupite.

Piratski čale

Izgleda da se Amstrad izvukao iz susedskog procesa koji se protiv njega vodi zato što je svoja dupli kasetas reklamirao kao idealnu stvaricu za pirate. Sud je odlučio da nije nezakonito prodavati takvu opremu, ali da ne sme u reklami da se potiče na piratstvo.

Za mentalno zaostale

Na univerzitetu Uil (Keele University) POD imenom Mr Kgh je napravljena prva video igra za mentalno zaostale. Zlobnici bi rekli da su sve takve.

Ovde daleko

Izgleda da Britanska, Francuska i Italijanska vlada povoljno gledaju na pokušaj Ejkorna, Tompsona i Olivetija da razviju evropski standard za obrazovni računari. Kad se zna kako je Ejkor uspeo sa BBC-om, to nije tako iluzorno. Osim možda za nas, jer mi smo daleko, daleko.

B. Dak.

Kvalitet i jeftinoća

Saga Systems (Saga Systems) polako dokazuje da je jedan od najzobljivijih proizvođača za „spektrum“. Najsvježiji primer je veoma dobar štampač pod oznakom LTR-1. Štampač pruža otisak sasvim nalik pisaočji mašini, iako je izuzetno jeftin — košta samo 149 funti (možda bi ovo „izuzetno jeftin“ za naše prilike trebalo pretvoriti u „relativno jeftin“). Printer prima papir formata A4 u odvojenim komadima. Tih nije — ali nije nešto naročito ni bučan. Sa bidirekcionalnim štampanjem dostiže brzinu od 10 znakova u sekundi. Za kraj još treba reći da ima standardan „centronics“ interfejs.

B. Dak

Tajni greh

Lotus je javno priznao da je njihov polulegendarni program Simfoni 1.1 (Symphony 1.1) ima bubicu (bag) koja može u trenu da pojede veliku količinu informacija. Do trenutka njihovog priznanja, bilo je prodato preko 25 000 komada programa. I veliki greše. A tek mali...



Unix za slepe

Britanski Royal National Institute počeo je obimni program čiji je cilj omogućavanje slepim osobama da u što većem obimu koriste mogućnosti računara. Prvi projekat je organizacija petodnevnog Unix kursa za slepe. Takođe je prevedena na Brajovu azbuku i knjiga Banahana i Ruttera „Unix the book“. Udžbenik, koji je takođe štampan, izveden je u mašinski čitljivom obliku tako da se iako može pomoću specijalnih mašina prebaciti u Brajovu azbuku, umesto muktotropnog, skupog i sporog posla prebacivanja „karakter po karakter“ iz klasičnog formata.

N. Aleksić

Crtanje na QL-u

Srtanje kompanija PSION, za koju se u poslednje vreme ne čuje mnogo, a koja se proslavila odličnim programima za QL, pustila je u prodaju jedan novi program za QL računari. Program se zove „Q DRAW“ (gde je ostalo slovo L?), a služi za crtanje. Taj isti program su upotrebljavali programeri ove firme za kreiranje grafike za odlično ocenjene programe „QL Chess“ (šahovski program) i „Match Point. Sada je dostupan svakom po ceni od 15 funti.



Razglednica
iz Londona

Andelko Zgorelec

Žir iz pepela

Protekla je još jedna računarska godina — godina u kojoj slavili desetogodišnjicu mikroracunarstva, ali i godina koja je bila i veoma teška za ovu mladu industriju. Jedni proizvođači su propali, drugi su se jedva održali. No, izgleda da je ono najgoro prošlo i mikroracunari su opet počeli da se prodaju. Proizvođači i trgovačka mreža nisu baš najsrećniji, jer su morali drastično da snize cene, za neke modele čak i napola za godinu dana, ali glavno da je krenulo. Ljubitelji mikroracunarstva se sigurno ne bune — uštedeće bar malo novaca ako kupe nešto starije modele, a i novim računarima, koji imaju odlične specifikacije, cena će biti znatno manja. Interes u Britaniji za mikroracunare je ponovo toliko porastao da su neke radnje sigurno zažalile što su uklonile računare sa svojih policia. Iz najveće trgovačke mreže za prodaju elektronskih produkata, Dixon, javljaju da prodaja ide vrlo dobro. Isto predviđa i vodeći poslovni dnevnik „Fajnalšal Tajms“ (Financial Times), koji smatra da će prodaja računara opet krenuti nabolje.

Racunari u paketu

Predstavnik najvećeg nezavisnog distributera za Amstradove i Komodorove računare, firme Lajtning (Lightning), Loretta Cohen, smatra da je tokom prošle godine u Britaniji prodato preko milion malih računara, što je veoma ohrabrujuća brojka. Za razliku od prošle godine, trgovačka mreža više ne govori o prodatim računarima, kao jedinici, već o takozvanim „paketima“ — računar sa nekim perifernim dodacima, poput kasetofona, palice, pa čak i disk jedinice, prodaje se u „paketu“ znatno jeftinije.

Nema sumnje da su potencijalni kupci zainteresirani za „pakete“, jer sada im je potreban samo televizor da računar bude u punoj spremi. Na taj se način, takođe, smanjuje i mogućnost greške da se kupi nešto što nije kompatibilno sa samim računarom.

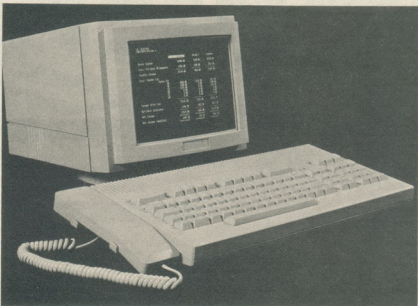
Mnogi modeli računara se sada prodaju po „damping“ cenama, pa neki trgovci tvrde da računari neće biti nikad više tako jeftini. Ima modela računara čija je cena

sada samo 29 funti („spectravideo“), a do nedavno su se prodavali i za preko 200 funti. Ne bismo, međutim, nikome savetovali da kupi neki od takvih računara, jer za njih gotovo nema programske podrške, niti im izgleda da se, pošto se više ne proizvode, za njih nabave rezervni delovi. Dakle, ova „povoljna kupovina“, ili, kako se to ovde kaže, „bargain“, je samo za — najhambrije. Nešto bolja situacija je da se kupe neki od Komodorovih računara (modeli Plus/4 ili C16), jer ima nešto više programa, ali ni taj izbor ne bismo preporučili, jer i to su već „otpisani“ računari, koje Komodor više neće proizvoditi, pa se ni jednoj softverskoj kompaniji ne isplati da za njih piše

svoje računare najviše za obradu teksta i datoteke (navodno, čak 70% vlasnika računara ih koristi u iste svrhe), za njega je PCW8256 računar godine.

Ejkor „komunikator“

Sredinom novembra se već tradicionalno održava najveći kompjuterski sajam u Britaniji zvani Compec. Na njemu je prethodnih godina bilo najviše velikih i mikroracunara, dok je ove godine bilo izloženo oko 100 različitih modela mikroracunara, među kojima su neki pobudili veliki interes. Zvezda sajma, međutim, nije bio mikroracunar neke od profesionalnih firmi, već dolazi pribora. Pošto vaš dopisnik upotrebljava



softver. (Komodor sada koncentriše sve napore na računare 128 i „amigu“).

U modi je „amstrad“

Šta onda, dakle, kupiti? Vaš dopisnik preporučuje svojim prijateljima i poznanicima, koji dolaze u London, (jasno, ako im je potreban savet), da nabave neki od računara koji ma veliku korisničku bazu u našoj zemlji — „komodor 64“, „epi“ ili neki od „amstrada“. To su provereni računari, sa dosta softvera kod nas i velikim krugom korisnika.

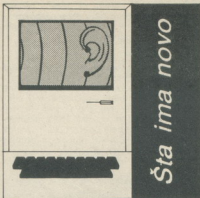
Najnoviji Amstradov proizvod PCW8256 dobio je neobično velik publicitet (pored ostalog, i naslovna strana časopisa „Personal Computer World“), jer je to kompletan računarski sistem — monitor, štampač i jedna disk jedinica — koji se prodaje po ceni od samo 399 funti bez poreza. Računar se, izgleda, definitivno može preporučiti za one koji mnogo pišu, kao i za malu poslovnu primenu. Računar se, za sada, odlično prodaje i gotovo ga je nemoguće nabaviti i pored mesečne produkcije od 40.000 primeraka iz nekoliko fabrika u Južnoj Koreji. Naravno, što je popularan u poslovnim krugovima većih firmi, jer, kako tvrdi jedan list, šefovima poslovnih jedinica nije potrebno odobrenje odozgo za nabavke do 500 funti, pa se kupovina ovog računara jednostavno zavodi u poslovne knjige kao nabavka sitnog kancelarijskog

od firme Ejkor (Acorn—žir), koja je jedva preživela ovu godinu. (Ejkor nikako da se reši problema, najnoviji je došao od poreskih organa, koji tvrde da im ova kompanija duguje zaostali porez u iznosu od 2,6 miliona funti za 1983. godinu).

Računar je nazvan Acron Communicator i to je prvi računar ove firme sa 16-bitnim procesorom WD65SCB16, koji je potpuno kompatibilan sa dobro poznatim 6502, koji pored Acorna, za svoje računare koriste i Apple, Atari i Commodore. Ali, ovaj procesor ima 24-bitni adresni bas, pa korisnička memorija nije više ograničena na 64 K, već se poput Motorolinog procesora 68000, može kretati do gigantских 16 megabajta.

Communicator je, zapravo, mali poslovni kancelarijski računar, sa ugrađenim modemom i telefonom, ali se na njega mogu priključiti disk jedinice i ostali dodaci od toliko popularnog BBC računara, kao i ROM čipovi sa programima. U ovaj računar je ugrađen i Econet, pa je moguće povezati čak 254 računara u mrežu, što je velika prednost za poslovnu upotrebu. Cena se za sada još ne zna, ali će biti oko 1000 funti.

Takozvani dobro obavešteni krugovi smatraju da će ovaj računar biti baza za najnoviji model BBC-jevog 16-bitnog računara, koji bi trebalo da se pojavi u prodaji u prvoj polovini ove godine. Izgleda da se Ejkor nada da bi takav računar mogao da bude usvojen kao standard za evropski obrazovni računar.



Šta ima novo

CP/M na „amstradu“

Za vlasnike „amstrada“ pojavio se velik broj prebačenih i prilagođenih CP/M programa na formatu od 3 inča.

Najinteresantniji od njih su, svakako, programi poznate firme Digital Research — PASCAL/MT+i CBASIC COMPILER, kao i grafički paketi DR DRAW i DR GRAPH. Za svaki od njih treba izdvojiti 49.95 funti.

Oxford Computer Systems je takođe izbacila na tržište svoju verziju paskal kompajlera (OXFORD PASCAL) za modele 6128 i 8256 po ceni od 49.95 funti.

Firma MicroPro je prilagodila najpoznatiji tekst-program, sada pod imenom POKET WORD STAR, ali ne i džepnom cenom od 119 funti. Program zahteva najmanje 56 K memorije, tako da je kompatibilan već sa CP/M 2.0. Uskoro će biti predstavljeni POKET CALCSTAR (spreadsheet program) i POKET DATASTAR (baza podataka).

Za one koji se bave vođenjem računovodstva i obradom platnih spiskova, programska kuća Sage im je obezbedila vrhunske programe (po vrhunskim cenama) i to POPULAR ACCOUNTS (100 funti) i POPULAR PAYROLL (70 funti).

Ovo, svakako, nije konačan spisak noviteta, jer se pretpostavlja da je „amstradova“ biblioteka CP/M programa već dostigla brojku od dve stotine.

D. Greber

Prstom po ekranu

Da li ste ikada poželeli da prstom pišete po ekranu računara? Ako jeste, ovo vam je idealna prilika da to i probate. Britanska firma Barnard Elektroniks isporučuje paket pod imenom Goust Skrin (Ghost Screen) koji vam omogućava da brljate direktno prstom po ekranu. Paket sadrži monitor, demonstracioni softver i korisnički paket softvera za ubacivanje u vaše sopstvene programe. Postoje verzije za „spektrum“ i „komodor 64“. Kada spojite aparaturu sa vašim računarem, pažljivo pročitate uputstva i učitajte program. Dodirujući prstom ekran, možete na njemu da ostavljate trag u obliku linija. Zar ne izgleda mnogo bolje od svetlosne olovke? Cena takvog paketa je oko 150 funti.

6/šta ima novo



Jagma za disketama

Na računarskom tržištu oseća se veliki nedostatak disketa od 3 inča. Izgleda da su i proizvođači i distributeri i prodavci popuno zakazali u proceni prodaje Amstradovih računara. Ništa, naučiće se za sledeći put.

Na udaru kritike

Izgleda da je Atarijev „ST 520“ najzad završen. Bio je on i pre skoro gotov, osim što je malo kuburio u vezi sa operativnim sistemom. Samo to. Sad je gotov i već stižu prve zamerke. Zamerke se tiču super igre za njega, Bratakasa, za koju oni koji su je probali masovno tvrde da predstavlja direktnu kopiju Megaigre za „komodor“. Ko zna, možda su i u pravu. Čučemo šta Atari ima da kaže na to.

Jeftinija „galaksija“

Dok pripremamo ovaj broj, saznajemo da će se računar „Galaksija“ do 31. decembra prodavati po izuzetnoj ceni od 55 555 dinara. Mi ćemo izaci iz štampe tek kad taj popust prođe, ali nismo mogli da propustimo a da ne pomenemo tu neverovatnu vest. Da li je neko pomenuo krizu?

Taksi za PC

Epsonov programski paket „Taxi“, napisan za Epson PC, koji je omogućavao sve blagodeti koje ima „mekintos“ i to u kolo-ru, sada se može dobiti i u varijanti za original PC. To je veliki plus, jer je TAXI zaista izvrstan program koji omogućava da čak i druge programe koje koristite na PCju obogatite njegovim mogućnostima, što nije odlika, recimo, GEMA.

Listing na majici

U prodaji su specijalne printerske trake koje omogućuju takvu vrstu štampe kakva se posle pomoću vruće pegle može preneti na majicu koja ima najmanje 35% poliestera. Da li ste razmišljali da malo švrljate po vašoj majici pomoću računara?

„Epson“ kao „epi“

Finger print je set od tri dopunska čipa za Epsonovu FX-80 seriju koji čine da „epson“ štampa nalik IBM-ovom i eplovom (Apple) printeru. To nije sve. Moguć je i NLQ mod koji je jako sličan printeru sa lepezom.

Nožni miš

Jedno veliko iznenađenje u hardverskim dodacima za računare! Radi se o mišu, ali ne o običnom, nego o nožnom! To je uređaj za IBM P6 koji se drži ispod stola i pomera nogom. Funkcija mu je ista kao i normalnom „stonom“ mišu, a izgleda kao metalna kutijica sa gumenom papučicom. Sad se, dakle, računariima može raditi i pomoću nogu. Šta je sledeće?

Grafički paketi

Posle tekst procesora i spređitova, evo još jedne udarne teme za softversku industriju kućnih računara. U prodaji je sve više grafičkih programskih paketa za crtanje i bojadisanje. Najpopularniji su Art studio (IBM PC), Fluent Fonts (Mac), Mac the Knife (Mac), Dotwriter 4.0 (Apple II, Mac, C 64, Atari 400—800), Blazing Paddles (Apple II, C 64) i PC Palette (IBM PC). Sigurno će ih biti još, sve boljih i sve jeftinijih.

Muzej kompjutera

Ako ste od onih koji imaju sve osim čokoladnog kompjuterskog čipa ili relea koji se koristio za prvi računar na Harvardu, onda nema problema — pišite kompjuterskom muzeju na adresu: The Computer Museum Store, 300 Congress Street, Boston, MA02210, USA i oni će vam poslati besplatan katalog. Kompjuterski muzej prati razvoj procesa obrade informacije od računalske do siccijumskih i galijum-arsenid čipova putem izložbi, kataloga, knjiga i programa. Ovo je jedini muzej ove vrste u svetu, bar za sad.

N. Aleksić



Drugarska pretnja

Ako u roku od mesec dana ne objavite bar jedno od šest pisama koje sam vam poslao, prestaju da vam ih pišem! Najboljinji!

Velibor Todorović, Zemun

Dragi Velibore, javi mi se telefonom. Želim bismo da se upoznamo sa tobom.

Više prakse

U nekim brojevima Vaše revije pisali ste o raznim primenama računara. Mišljenja sam da bi trebalo odvojiti više prostora za pisanje o upotrebi računara u oblastima prirodnih nauka, arhitekture, građevinarstva itd. Piše malo više o aplikacijama u hemijskoj, fizičkoj, električkoj i drugim laboratorijama. Bilo bi lepo kad bi sve to bilo proučeno i prigodnim programima iz konkretnih oblasti. Možda ne bi bilo loše kad bi se povremeno pisalo i o specijalizovanim računarima.

O. Spirovski, Kumanovo

Kome da verujem

Veole bih da mi razjasnite jednu stvar koja me čudi. Kada sam prvi put čuo za Amstradov „PCW 8256“, bio sam veoma zainteresovan, jer je svoj računar uglavnom koristim baš za obradu teksta. Zato je zanimljivo što sam se puno obradovao kada sam video da u „Računarima 10“ imate pregled te mašine. U stvari, nije bio jedan pregled već dva jer su i Dejan Ristanović i Branko Đaković pisali o računaru „PCW 8256“. Ono što me čudi je to što su njihove ocene nepomirljivo različite. Veole bih da znam da li se to dogodilo slučajno ili namerno i kako da tumačim takve ocene. Veri- ni čitalac

Miloš Jęftović, Beograd

Da si čitao pisma u „Računarima 9“, primetio bi da je već rečeno da ćemo pokušati da za svaku novu mašinu bude bar dva pregleda, ali pogleda, radi objektivnosti i sadržajnosti. U ovom slučaju, Dejan je gledao sa aspekta računarskog stručnjaka, a Branko samo kao korisnik. Kao i svaki drugi računar i „PCW 8256“ ima dva lica. Trudićemo se da i dalje ka takav način predstavljamo nove mašine.

7/load „dragl računari“

Knjiga od „Računara“

Imam još od pre jednu ideju koja mi se ponovo javila kad sam video vaš registar tema o kojima ste pisali u dosadašnjim brojevima „Računara“. Znam vaš običaj da u kraju godine ukoričite izvestan broj „Galskija“ i da ih tako prodajete. Mislim da bi slična ideja sa „Računarima“ bila izvanredna. Takva „knjiga“ bi predstavljala zaista dragocenu pomoć za bilo kog pasioniranog programera. Možda bi jedini problem mogla da predstavlja činjenica da je prva dva broja „Računara“ rasprodala, ali vi biste to mogli da dostapmate, zar ne? Kad se takva „knjiga“ pojavi, ako se pojavi, ja ću biti siguran kupac, iako imam sve dosadašnje „Računare“.

Goran Janković
Banja Luka

Ispunjenje snova

Bilo mi je veoma drago kada sam u poslednjem broju „Računara“ pročitao članak o vrednoj takozvanog kertricitu za C-64. Građma bih voleo da učestvujem u realizaciji ove super-ideje, a za početak dajem nekoliko predloga za programe koji bi bili na kertricitu: 1. Simon's Basic, 2. Hasb, 3. Monitor 49152 ili slično. Ideja o kertricitu u koji bi bile upisane rutine za korišćenje grafike (i muzike) promeonom editora je dobra. Međutim, to mi više liči na promenu ROMa, a o tome treba posebno pričati. Žbog lošeg vida bio sam prinuđen da napustim i elektroniku i računare (po savetu doktora), ali sam u ovom članku video ostvarenje svojih snova.

Radeta Nebojša
Jovana Cvijica 7/6
15000 Šabac

Zanemarena klasa

Nepripravno ste zanemarili cijelu klasu džepnih računara. Uprkos očiglednim manama, to nisu samo igračke. Ne sećam se da sam ista pročitao o modelu SHARP 5100 kojim se služiš već pet godina i koji je potpunosti zadovoljava moje koje su svakodnevene potrebe. Iako se na njemu ne mogu ubiti mrski osvajači iz svemira. Nedavno sam u reviji „Naš Dom“ čitao o vrlo interesantnim modelima džepnih računara. Nadam se da ni „Računarima“ neće biti ispod časti da objave nešto o takvim mašinama, o njihovoj primjeni i pripadajućoj opremi. Puno pozdrava.

B.Š. Zagreb

Ne sećamo se da poznajemo ikoga ko koristi sharp 1500. Zašto ne pripremite nešto za svoje istomišljenike?

Muke sa šiframa

Na rešavanje vašeg nagradnog pitanja broj 2 utrošio sam oko dvanaest časova i nisam uopšte koristio računara. Karaktera sam prebrojavao „ručno“, jer bi mi za taj posao računarno bilo potrebno gotovo isto toliko vremena.

Kao dokaz da sam teško dešifrovao ove prve dve rečenice teksta:

„Ja se zovem Robin Brodhead i najboljagija sam osoba u čitavom sunčevom sistemu ...“

Posle dešifrovanja teksta htelo sam da uradim isto ono što je i Dejan uradio. Za to mi je trebalo broj N u RANDOMIZE N. Da bih ga dobio, modifikovao sam Dejanov program za šifrovanje (zato što mi „spektrum“ nema funkciju MID, ubacio ga u FOR-NEXT petlju i zapeljao računaru u posao s kojim se ovaj borio čitavo popodne, a stigao je da ispiše samo prvih 3000 slučajeva (od 65535). Zato sam odustao.

Želim redakciji da nastavi putem kojim je krenula!

Tihomir Kućinar

Ni luk jeo, ni luk mislao

„Računare“ smatram izvrsnim časopisom, ali su me u poslednje vreme malo razočarali. Čini mi se da „Računari“ kao, uostalom, i ostali računarski časopisi zaboravljaju činjenicu da je tipičan YU haker srednjoskolar ili student. Žbog toga mi je usliko neke kod nas u smeće, pogotovo ako dinar nastavi ovako padati. Mislim da vas je i vaša traktetka uverila u to, jer bolje je vrabac u ruci ...

Druga stvar koja mi smeta, a smatram je i puno opasnijom, jeste neobjektivnost! Ova zamerka se najviše odnosi na tekstove Dejana Ristanovića.

Čitajući: ... Spuštanje cene računara na 200 funti ... ovaj tekst je oblična potvrda više puta izrečene tvrdnje da je QL promašena mašina ... (broj 8, strana 4, naslov „Nikada ne veruj Sinkleru“ ...

„Rad sa mikrodajrom je toliko mukotrpan da se mnogi vlašnici „spektruma“ bolje osećaju bez njega nego sa njim.“ (broj 4, strana 26)

„Očekuj uskoro jedan dobar članak o QL. Neki zlobnici, doduše, pokušavaju da u domaći folklor provuku izreku „Veruj keru ne veruj Sinkleru“, ali mi ne spadamo među njih.“ (broj 10, strana 19)

Ovo zapravo, doduše, nije potpuno, ali je slično, naslov prvog citata, a u stilu liči na Dejana. Ovakvi, nazgled sitni propusti, opravdano izazivaju revolt kod čitalaca.

Umeci su vam pun pogodak.

Ne bih vam ovo pisao da ne volim „Računare“ P.S. Dejana izgradište, Vlastu, pozdravite a Jelenu poljubite u moje ime!

Mihailička Damir
Gundulićeva 148
54000 Osijek

Vlastu smo pozdravili, Jelenu smo poljubili a Dejana, naravno,

specijalno izdanje rubrike load „dragl računari“ objavljujemo na str. 66

nismo izgrдили. On nema nikakve veze sa onim što nam zameras. Neka se sam javi ko je kriv! P.S. Zamoli nas još jednom ono poslednje pod P.S.

Boris o sebi

Evo, ponovo vam se javljam da bih dopunio svoje pismo koje ste objavili u „Računarima 10“. Zamerili ste mi što vam nisam napisao potpunu adresu i prezime. Sada vam ih šaljem kao poklon za Novu 1986. godinu. Zovem se Vuklović Boris, stanujem u ulici D. Vukasovića 78, blok 61a, Novi Beograd. Boris iz Devete beogradske P.S. Najbolji ste. ZIVEO QL!

Drugi o Borisu

U „Računarima 10“ sam pročitao pismo „Boris iz Devete beogradske“ (idem u isto odeljenje sa njim). Prvo bih želeo da ispravim njegovu grešku u objavljenoj adresi: Vuklović Boris, Dušana Vukasovića 78/14, 11070 Novi Beograd.

Mislim da je Boris u svom pismu preterao sa tvavljenjem QLa. Sigurno ste primetili da Boris nije naveo ni jednu njegovu manu. Piše da tastatura zadovoljava. Ako pritisnete tipku lačno u sredinu, nema problema, ali ako pritisnete bliže ivici, tipka koči i otežava kucanje. „Softver“ najkvalljetiniji! Uostalom, kad je pokazao memo, zašto čitocima „Računara“ nije napisao i cene programa za QL? Zato što bi nekoliko programa zajedno prevazišlo cenu QLa!

Da je QLov editor, koji je ostao u njegovoj spekmatici, dobar? Naravno da nije! Zašto su napravili QL Toolkit koji pokušava da od QLovog bežikta napravi dobar bežik?

Boris je želeo da kupi QL. Tada je isticao samo njegove dobre osobine, a za mane se pravio glup. Onda se navino odlučio da kupi „atari 520 ST“, i naravno, isticao samo dobre osobine, a QL-u je isticao samo mane (mikrodajr je najveća glupost na svetu, tastatura nije profesionalna, editor je glup). Kad je otkrio da „atari“ ne odgovara njegovim potrebama, vratio se QL-u. Zaboravivši da je nekada pričao o njegovim manama, Počeo je ponovo da ga hvati i kaže da nema njegov procesor 68008. Jedan njegov drug misli da je oboleo od nove bolesti! QLTTISA. Simptomi su: oboleli više od 100 puta u toku dana izgovori reč QL.

Ja nemam ništa protiv QLa, ali sam protiv preveličavanja njegovih dobrih karakteristika.

Členković Slobodan (uz pomoć Trhulji Zlatka) iz Devete beogradske

Drage računardžije iz Devete beogradske, tačno je da ste lepo prikazali i zagrznene pobornike QLa i njegove kritizere, ali zar ne smatrate da se ovaj novinski razgovor mogao nastaviti u miru prelepe Devete beogradske? Što se Borisa tiče, svako ima pravo i da nešto odluči i da se — predomisli.

Priprema: Branko Đaković

računar po glavi

Kako to radimo mi

Posle čitanja ovakvog uvoda, prosečnom čitaocu, neupučenom u filozofiju našeg računarskog biznisa, sasvim sigurno zastaje dah. Niko od nas nije zamišljao da put u vode zdrave ekonomije može biti tako jednostavan. No, već ovde, sasvim sigurno, postavljaju se i neka pitanja, na koja nije lako dati odgovor. Nismo li mi negde na začelju po broju stručnjaka, obučenih da se uhvate ukostac sa problemima nove tehnologije? Nije li naša zemlja negde na začelju evropske lestvice tehnologije? Samo u dve naše najveće fabrike moguće je dobiti kvalitetnu dvostronu štampanu ploču. O višestolnoj zasadi nema ni pomena. Da li je, osim toga, neko čuo kako je neka naša fabrika uspeła da proizvede makar i najbednij mikroprocesor, neophodan u računarskoj industriji? I kako se, uopšte, naša industrija bori sa ovim gotovo nerešivim problemima?

Veliki mali pirat

Kao i svuda u svetu, i kod nas se računarska industrija deli na softversku i hardversku, ali sve između ova dva vida jače nego bilo gde drugde. Kod nas se prodajom softvera, u pravom smislu reči, bave jedino industrija hardvera i, naravno, bezoblični pirati. Ako se odlučite za nabavku računara nekog od naših proizvođača, on će vam u cenu proizvoda zaračunati i vrednost softvera, koji se uz njega isporučuje. Za sve ostale softverske dodatke koji će vam u budućnosti trebati, moraćete dobro da platalite. Drugo rešenje je da se za ovakve softverske dodatke obratite piratima. Oni ih sasvim sigurno imaju i to po nižoj ceni. Manji problemi nastaju ako ste nabavili neki od većih računara (VAX, PDP, IBM, HONEYWELL). Za ove računare još uvek nećete naći oglaš u novinama koji nudi prodaju softvera, ali ne odajavajte. Potražite nekoga ko ima takav računaru duže od vas. On se gotovo sigurno već pozabava sveopštom razmenom pameti.

Ako ne računamo minimalne zarade, koje na ovakav način ostvaruju industrija hardvera i pirati, onda se ovaj vid sticanja računarskog hebla može gotovo sasvim zanemariti. Marginalnost ovakvog prilaza računarskom biznisu posledica je psihologije našeg društva. Mere za vrednovanje softverskih proizvoda kod nas još uvek nisu sasvim usaglašene. Još uvek se vode maršantske debate da li to treba da bude kilobajb čistog mašina, jezik u kojem je program pisan ili, pak, neka sasvim nova mera, koja će nepogrešivo utvrditi vrednost svakog napisanog programa. Direktna posledica ovakvog pristupa je da kod nas, u stvari, čisto softverske kuće i ne postoje.

Drugi i daleko efektniji način prodaje softvera je posredni. Pri tom je potrebno da

nabavite što veći računaru. Ako uspete da uvezete neki CRAY ili nešto slično, bićete verovatno najjači u posrednoj prodaji softvera kod nas. Drugi korak u ovom poslu je da dovedete nekoliko stručnjaka (ne moraju biti pravi), sa što je moguće zvučnijim imenima. Na kraju, zbog vrednosti preme i stručnosti kadra, proglašavate se računarskim centrom od opšteg značaja. Posle ovoga možete sasvim biti sigurni da vam je budućnost obezbeđena. Sve firme iz vaše mesne zajednice biće vam verni potrošači. Obradivaćete im podatke dugo godina. Vaš računaru će postati neizbežna banka podataka od koje zavisi čitav vaš kraj, i, naravno, eto neiscrpnog izvora zarade! Vi možete da diktirate cene, bez ikakvog ograničenja. Ako se neko pobuni, nema obrade, nema ni podataka. Naravno, ovde se kao jedini validni savetnik pri kupovini računara morate neprestano truditi da suzbijete konkurenciju. Nekim od firmi za koje radite sasvim sigurno će pasti na pamet da i same osnuju slične centre. Tada vi nastupate kao savetnik i u njihovo ime kupujete neko bezvrednu mašinu. Najbolje da ta mašina bude da obezbeđi uvoznih podataka, ali nikako i obradu. U taj način, rešavate se dosadnog kucanja i viška skupere radne snage. To za vaš računaru i smešno nisku cenu rade svi ostali. Vi prihvatate ove podatke i obradujete ih na vašem moćnom računaru. Naravno, u krajnju cenu obrade uračunavate sve, i vrednost amortizacije sistema i softvera, i unos koji ionako ne radite i ko zna šta još ne! Ljudska mašta je neiscrpna. Nemojte samo zaboraviti da na kraju kalkulisanja ovakve cene stavite prst u uvo i odoak pomnožite dobijene rezultate bar sa pet. Ostali koji se bave ovim poslom neće vam nikad oprostiti.

Uz svaku diaku

Nedostaci ovakvog načina bavljenja računarskom industrijom su u visokim početnim ulaganjima. Cena mašina, prostora u koji će biti smeštene, papira neophodnih za otvaranje ovakve firme je od prilike nekoliko stotina milijardi (starih naravno). Osim toga, ne treba zanemariti ni cenu stručnjaka neophodnih u ovoj jrgi. Oni danas savršeno znaju svoju vrednost i polako postaju ono što su nekad bili fudbaleri. A ako ih slučajno uvredite nekim vašim smešnim predlogom, odoše u Ameriku i — eto kraja vašeg poslu!

Drugi i daleko primamljiviji vid računarskog buma kod nas je proizvodnja hardvera. Istorijski gledano, u ovom našem većito burnom podneblju materijalne stvari bile su uvek cenjenije od duhovnih vrednosti. Tako se i danas opaža jedan na prvi pogled stari paradoks. Dok u celom svetu cena hardvera drastično opada, a cena softvera nezadrživo raste, kod nas se zbiva sasvim obratan proces. Cena hardveru nezadrživo raste. Mnoge se komplikacije oko nabavke. Admi-

nistracija uporno uzima svoj ne tako mali dinar pri svakoj kupovini mašina. Sve to polako počinje da liči na priču o duvanu. Čoveku je najlakše uzeti tamo gde je primoran da potroši. U isto vreme, ta ista administracija još uvek je nezainterosovana za softver, pa njegova cena diktirana piratskim tržištem naglo opada. Ovakvi i na prvi pogled sasvim čudni uslovi na našem tržištu dali su snažan podstajac savremenoj računarskoj tehnologiji kod nas. Sve je više promućurnih ljudi koji zaključuju da se igra u ovakvo mutnim vodama isplati. Adekvatan je i broj novonastalih firmi koje se bave proizvodnjom računara.

Ako živite u maloj, nerazvijenju opštini, možda je vreme da počnete da razmišljate o svemu što ste ovde pročitali. Možda je sve ovo pravi put da postanete spastitelj vašeg kraja. Prosvetitelj u novoj tehnološkoj revoluciji. Za sve to vam, u stvari, ne treba toliko puno kao što vam se možda čini. Za početak, vašoj firmi je dovoljno nekoliko hiljada maraka da biste obezbedili računsku osnovu za uvoz. Neophodna su vam i dva-tri političara uticajna u vašem kraju. Razgovarajte sa njima. Navedite im sve aspekte vezane za novu industriju i oni se neće oglušiti o vaš poziv. Oni veoma brzo računaju i biće im vrlo brzo sve ovo jasnije nego vama. Uskoro ćete imati sve saglasnosti neophodne za stvaranje nove fabrike računara. Ođavde, naravno, sve zavisi samo od vas. Kada proizvodnja na kraju počme, tržište vam je sasvim sigurno obezbeđeno. Svi koji u vašem kraju budu želeli da nabave neku od inteligentnih mašina moraću da se obrate vama. To je u interesu regiona! Zato požurite. Drugi imaju slične ideje, a što pre počnete, veće tržište je pred vama.

Pljuni pa zalepi

Ako vas još uvek brine tehnologija proizvodnje, iznenadite se koliko je to, u stvari, jednostavno. Vlada Bulatović će mi, nadam se, oprostiti što ću se ovde poslužiti jednom njegovom pričom. Verujem da će mu se dopasti otkriće da je on bio jedan od prvih vizionara naše računarske strategije.

Kod nas postoje gotovo isključivo dva metoda za proizvodnju računara. U oba slučaja, postupak počinje na isti način. Prvo uvezete kompletan računaru, najbolje iz Amerike. Zatim uvezete nalepnice, najbolje iz Italije. Pri tome morate smisliti što zvučnije ime novog proizvoda. Ako ste malo štedljiviji, nalepnice možete naručiti i na našem tržištu, ali tada rizikujete da svi zaključke kako ste neobizbilni proizvođač i napuste vas. Čak je i naša tehnologija nalepnica zaostala.

Nadalje, možemo razlikovati dva proizvodna postupka. U prvom od njih, pljunete na nalepnicu, zalepite je na računaru i prodate ga. Po drugoj tehnologiji, prvo pljunete na računaru, pa zatim zalepite na-

Računarska industrija je jedna od retkih privrednih grana kod nas koja nepogrešivo beleži zavidne rezultate. Danas sa ponosom možemo reći da smo po broju fabrika za proizvodnju računara ispred do juče neprikosnovene sile kakva je Amerika. Uplašeni enormnim profitima naše industrije vezane za računare, divovi iz silicijumske doline sve češće bacaju zabrinute poglede na ovo podneblje. Nama, međutim ni to nije dosta i ništa nas ne može zaustaviti na putu ka našem idealu — računar po glavi stanovnika, u svakoj mesnoj zajednici fabrike računara!



lepicu na njega i na kraju ga opet prodate.

Pri svemu ovome, morate voditi računa jedino o krajnjoj ceni vašeg proizvoda. Ona se određuje otprilike ovako: preračunate po dolarskom kursu nabavnu cenu računara u dinare. Cenu nalepnice možete galantno izostaviti. Kod ovog računa, naravno, uzimate najrniji mogući kurs dolara, od kojeg i samom dolaru dode tolika muka da još više pozeleni. Zatim, da biste bili sasvim fer, na tako sračunatu cenu dodate oko 48 posto za carinu i poreske dažbine (one su vama naplaćene po bančinom kursu). Eventualno, *ako vam hitno treba novac, dodate još deset posto za neakve svoje troškove, zatim stavite prst u uvo i cenu pomnožite odoka najmanje sa pet. Ovaj deo kalkulacije nikako ne smete propustiti — ostali koji se bave ovim vidom proizvodnje neće vam oprostiti i isključuje vas iz tima povlašćenih.

Kao što vidite, posao je izuzetno jednostavan. U isto vreme, to je verovatno jedini posao koji na uloženu marku donosi pet. Toliku produktivnost nema ni jedna industrija sveta. Naravno, pri svemu ovome treba voditi računa o obrtu sredstva, ali tu već sigurno znate sve. Održavajte redovne veze sa carinom. Jedino od nje zavisi koliko brzo ćete nabavljati reproma-

terijal, a od toga vam opet, direktno, zavisi obrt. Dobro ćete proći ako uspete da pronađete način za kupovinu deviza na crno. Na ovom tržištu kurs je samo za oko deset posto viši od bančinog, a vi zarađujete do 500 posto! Jeste li ikad čuli za bolji posao?

Mona Liza na računaru

Ako se, na kraju, odlučite za ovaj posao, povežite se što pre sa onima koji ga već rade. Na taj način obezbedujete ulaznicu u povlašćeno društvo tehnološki obradovanih, a u isto vreme obezbedujete se od većine konflikata, koji mogu nastati između vas i onih koji ovim tržištem već vladaju. Možda vam ova cela priča liči na onaj vic da je Jugoslovenu bilo najteže dok nije izmislio Amerikanca, a sada taj isti Amerikanac misli na njega. No, neka vas to ne brine. Dok god se novac siliva u vašu kase, ov vam ne mora biti važno. To što ste dali devize za računaru upotšte nije bitno. Dinar-ska protivrednost, obilato umnožena, silila se u vaše fondove. Vi ste u stanju da, kupujući devize, nabavljate sve savršenije i skupije modele računara i, naravno, pri tome ostvarujete sve zavidniju materijalnu dobit. To što je pokvareni Zapad, u stvari, pronašao novi put za izvlačenje novca vaš naravno ne brine, jer pri tome i vi zarađujete, a svojim krajnjim finansijskim efektima doprinosite stabilizaciji. Možda vam izgleda da se ostatak novca samo prelio iz jedne

kase u drugu, ali utisci su površni i varaju. Ništa nije istina dok se ne dokaže. Ugla-vnom, vama je dobro, a ostali su zahvalni, jer ste im omogućili da dobiju kartu za voz povlašćenih u novo doba konkurencije u proizvodnji.

Možda vas još pomalo brine pitanje stručnjaka neophodnih u ovoj igri. Samo mimof! Sećam se vremena od pre nekih desetak godina. Otišao čovek na dvonedeljni kurs programiranja u Rovinj. Ostao dve nedelje, vratio se sa gomilom papira ispisano-g na računaru. Pohvalio se kako ume da natera glupavu mašinu da ispiše nule po dijagonalni papira. Poklonio važnim ljudima Mona Lizu nacrtnu računaru i postao važan čovek. On se najbolje razume s računaru. Danas je cenjeni direktor jednog računskog centra. Ako nemate neke probleme, upoznacu vas sa njim. Samo se javite. On zna sva rešenja.

Sve što ste pročitali bila bi čista idila, da u celoj stvari ne postoji i jedna na izgled sitna opasnost. Ceo ovaj posao izaziva do sada nepoznatu bolest (nešto po epidemiošnosti slično SIDI). U stvari, od predučkog držanja prsta u uvu, pri razmišljanju o novim cenama proizvoda, postoji opasnost da vas uvo trajno zaboli za sve. Ne dajte se, međutim, ometiti! Samo hrabro, u nove radne pobeđe!

Radmir Nikolajev

Periferijska
oprema

hard diskovi

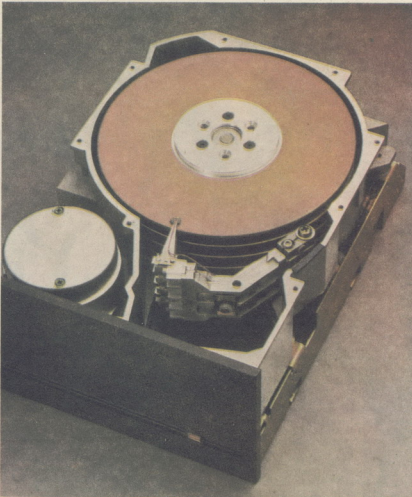
Iako je količina podataka koje se smeštaju na disketu u poslednjih nekoliko godina gotovo udesetostručena ugrađivanjem dodatne glave, povećanjem broja traka i uvođenjem dvostruke gustine upisa, flopi disk jedinice su 1984. i 1985. godine došle do svog vrhunca: pojavile su se fizičke prepreke koje praktično onemogućavaju povećanje njihovog kapaciteta. Kakve su to fizičke prepreke? Pre svega, flopi disk se okreće „svega“ 300 puta u minutu, što znači da se podaci sa osamdeset traka moraju preneti u memoriju najmanje 80/300=0,26 minuta ili oko 15 sekundi. U primenama u kojima se radi sa bazama podataka problemi se značajno uvećavaju: upisno-čitajuća glava treba intenzivno da se pomera između traka, što znači da će intervali u kojima čitav sistem čeka da se sektor koji treba pročitati nađe ispod glave postajati mnogo duži od vremena potrebnog za efektivno učitavanje podataka. Rezultat: neefikasnost.

I to je sporo

Zašto disk ne bi mogao da se okreće brže? Možda vam je poznato da je upisno-čitajuća glava spuštена do površine diske, što znači da između dve po prirodi različite površine postoji intenzivan fizički kontakt a samim tim i nezanemarljivo trenje koje se strahovito povećava ubrzavanjem diska. Veće trenje bi, jasno, dovelo do bržeg trošenja glave i same diske; posle neke granične brzine magnetni sloj na disketi (a samim tim i podaci na njoj) bi bio jednostavno ostrugan.

Disketa je, kao što smo pisali, podeljena na četrdeset ili osamdeset traka, pri čemu ovaj broj ne može bitno da se poveća. Problem nije u tome što je teško preciznije pozicionirati glavu diska, već u tome što se disketa ne nalazi uvek na istom mestu: ako dva puta umećemo disketu u drajv, nećemo je jednako snažno gurnuti, niti ćemo jednako energično pritisnuti klapnu. Osim toga, ne može se očekivati da će azimutni glavica raznih drajvova biti apsolutno jednaki. Zato između traka mora da postoji dovoljan prostor da se glava može pozicionirati na svaku od njih uz potrebnu toleranciju. Ograničen broj traka povlači, jasno, i ograničen kapacitet diske od oko 1 Mb.

Poslednje od ozbiljnih ograničenja flopi diskova je broj (kilo) bajtova koji mogu da se upišu na jednu traku: podaci se, kao što vam je verovatno poznato, upisuju promenom magnetnog fluksa kroz materijal. Diskete su, bez obzira na zaštićenost kartonskom kutijom i omotačem, izložene vazduhu koji je prepun čestica prašine; mora se nekako obezbediti da pad nekog takvog zrnca na disketu ne bude poguban po podatke na njoj, a takvo se obezbeđenje



Deset puta bolji: Hard diskovi imaju desetak puta veći kapacitet i deset puta kraće vreme pristupa podacima ali im cena nije deset puta veća od cene slopi diskova

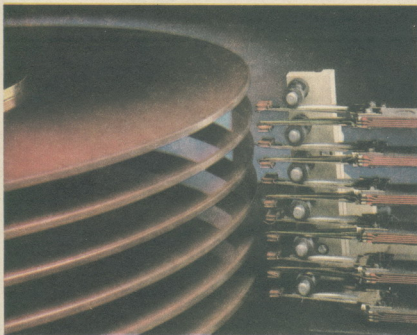
nikako ne slaže sa željom da se podaci na trakama pregusto pakuju!

Rešenje sva tri problema može da se dobije jedino suštinskom promenom koja se naziva masivni disk. Ukoliko disketu zamenimo čvrstim magnetnim diskom koji se nikada neće vaditi iz drajva, problem preciznog pozicioniranja se rešava već u fabrici kada se uređaj fino šteluje. Ukoliko je disk, zajedno sa upisno-čitajućom glavom, hermetički zatvoren, nikakva spoljna prašina ne može da ugrozi podatke na njemu, pa je kapacitet jedne trake ograničen jedino karakteristikama materijala koji industrija može da proizvede.

Ostao je još problem brzine okretanja diska: njeno povećanje, svakako, skraćuje

radni vek uređaja. Trebalo bi, dakle, držati glavu na deliću milimetra iznad površine diska (pri tom bi, jasno, trebalo pojačati sva magnetna polja), što bi moglo da se izvede pomoću veoma precizne mehanike. Kako, međutim, da držite glavu na deliću milimetra iznad diska i da je pri tom brzo šetate od trake do trake? Za rešenje ovoga problema koje ne zahteva basnoslovne troškove je zaista trebalo dosta inspiracije: ako se disk okreće dovoljno brzo, vazduh koji se nalazi uz njegovu površinu će se kretati skoro jednako brzo. Ako je upisno-čitajuća glava dovoljno laka i aerodinamički oblikovana, ona će plivati na tom sloju vazduha kao na nekom vazdušnom jastuku! Rastojanje između glave i površine diska danas varira

Do pre jedva godinu dana flopi disk jedinice su bile san miliona vlasnika kućnih računara koji su se svakodnevno patili sa sporim i nepouzdanim kasetofonima. Mnogi od ovakvih snova su danas ostvareni ali, kako filozofi kažu, apetit dolazi sa jelom. Kapacitet od samo jednog megabajta po disketi je premali, okretanje flopija presporo, podacima se pristupa uz mnogobuke... sasvim dovoljno da počnemo da sanjamo o masivnom (hard ili vinčester) disku koji, po sve umerenijim cenama, nudi neverovatno povoljnije karakteristike!



Ni samo jedan: U kutiji jednog hard diska hermetički je zatvoreno nekoliko diskova, ali sistem funkcioniše kao jedinstvena celina

između 2 mikrometra i 300 nanometara; poredenja radi, svaka dlaka na vašoj glavi je debela oko 70 mikrometara!

Kombinacija masivnog nepomičnog diska u hermetički zatvorenoj kutiji iznad koga pliva upisno-čitajuća glava je poznata pod imenom vinčester (Winchester) disk. Druga vodeća tehnologija u ovoj oblasti, *whitney*, daleko je manje zastupljena kod personalnih računara, pa joj nećemo posvećivati posebnu pažnju.

Usporeni početak...

Winchester diskovi starijih generacija su mogli da se prepoznaju po priličnoj buci: zvuk koji izaziva disk u pogonu je unekoliko prigušen zvukom ventilatora koji se obavezno ugrađuje u kutiju. Ova je buka konstantna: dok flopi disk počinje da se okreće tek kada je računar pozeleo da mu pristupa, hard disk se okreće neprekidno, jer mu je potrebno nekoliko minuta da postigne radnu brzinu od 3000 obrtaja u minuti i još nekoliko minuta da se iz te pune brzine zaustavi.

Zbog ovog ubrzavanja i usporavanja, procedura uključivanja računara koji su

opremljeni hard diskovima ume da bude prilično komplikovana. Najpre se uključuje glavni prekidač, a onda hard disk jedinica. Sledi poduže nestrpljivo posmatranje indikatora *Ready* posle čijeg paljenja smemo da uključimo i sam kompjuter. Neki proizvođači računara ugrađuju samo jedan prekidač na čitavu opremu (prekidač se nalazi na monitoru, ne pitajte zašto), a zatim uvode potrebna kola za kašnjenje tako da se računar automatski pali tek kada je disk operativan. IBM je taj problem rešio softverski: model XT „satina“ proverava svojevrsnu ispravnost po svakom uključivanju!

Činjenica da glava pliva na vazdušnom jastuku je zaslužna za ogromne kapacitete vinčester diskova, ali i predstavlja izvor problema: ukoliko u bilo kom trenutku i iz bilo kog razloga glava padne na magnetnu površinu, podaci na njoj će biti izgubljeni, a moguće je i trajno-oštećenje diska i/ili glave. Gotovo isključivi razlog za ovakve katastrofe je prestanak napajanja uređaja. Disk se, jasno, ponekad mora i isključiti, a to je tada situacija drugačija: kada pritisnemo taster *Ready*, sijalica će se ugasiti i disk će pomeriti glavu nad posebno određeno „parking“ na koji će, posle dvadesetak sekundi kočenja, lagano i bezbedno ateri-rati. Moderni hard diskovi su unekoliko obezbedeni od nestanka struje, pošto im energija u kondenzatorima omogućava da

čak i bez napajanja dovedu glavu do „parkinga“, ali se ipak treba odrediti rada sa računarom u trenutima kada se mogu očekivati problemi sa napajanjem. A takvi trenuci nisu kod nas baš tako retki!

... i prebrzi rad

Pošto ste nekako povezali hard disk sa računarom (potreban vam je i poseban interfejs jer hard diskovi ne mogu da koriste kontrolere za flopije), počnite da uživate u njegovim karakteristikama. Danas se standardni vinčester diskovi od 5.25 inča okreću 3536 puta u minutu (uporedite ovo sa brzinom flopija od 300 obrtaja u minutu), „pamte“ 10, 20, 40 ili 80 megabajta podataka (kod flopija ovaj broj retko prelazi 1 Mb), što zahvaljuju gustini pakovanja od 690 traka po inču (najviše 96 tpi kod flopija) i omogućavaju pristup svakom podatku za samo 8 ms (kod dobrih flopija vreme pristupa obično nije kraće od 100 ms). Brzina prenosa podataka je, ponekad, prevelika za neke osmootbitne mikroprocesore, pa se u hard diskove ugrađuju veliki baferi (ponekad se čak primenjuje i dvostruko, pa i višestruko baferisanje) što značajno povećava njihovu cenu. Alternativno rešenje je zamena programiranog ulaza—izlaza (koji se zasniva na mehanizmima takozvanog *hand-shaking-a* ili, daleko češće, nemaskiranih prekida) mehanizmima direktnog memorijskog pristupa (DMA), kod koga mikroprocesor nema udela u prenošenju podataka između periferije i memorije. Principe direktnog memorijskog pristupa smo objasnili u prikazima „atarija 520 ST“ i „amige“.

Jasno je da hard disk nema mnogo smisla koristiti kao brzi kasetofon: treba imati dobar operativni sistem i još bolje aplikativne programe da bi se iskoristile blagodeti ogromne spoljne memorije i ultra brzog pristupa podacima. Što se operativnog sistema tiče, neophodna je mogućnost kreiranja stabla kataloga. Dok se na jednu disketu obično smeštaju samo datoteke jednog tipa (na primer bajzik programi, tekst koji se obrađuje, poslovni softver i slične stvari), na hard disk smeštamo ogroman broj po prirodi različitih datoteka, što znači da je njihovo ostavljanje u jednom katalogu nepregledno i veoma rizično (možda nekada snimimo program PROBA i time obrišemo tekst PROBA). Zato je prirodno u osnovnom katalogu formirati samo nekoliko podkataloga (na primer potkataloge TEKST, BASIC, IGRE, PODACI i SYSTEM) a zatim u njima grupisati odgovarajuće datoteke. Datoteke u bilo kojem potkatalogu se mogu potpuno nezavisno imenovati tako da nema nikakvih prepreka da koristimo ime PROBA u svakom od njih. Kod boljih operativnih sistema možemo da pravimo potkataloge u potkatalogima i tako do neke ograničene dubine.

Pri radu sa hard diskom se postavlja i problem zaštite podataka. Često se dešava da kompjutere sa masivnim diskovima kori-

sti više ljudi (na primer u kancelariji) i da svaki od njih ne želi da mu se ostali, namerno ili slučajno, mešaju u posao. Tada je prirodno kreirati po jedan direktorijum za svakog od korisnika i zaštititi taj direktorijum lozinkom: samo onaj ko zna lozinku će moći da čita, modifikuje ili briše programe iz odgovarajućeg direktorijuma (ponekad se čak uvode nezavisne lozinke za čitanje, upis i brisanje). Prilikom formatiranja hard diska (da, i hard disk se formatira, ali uglavnom jednom u životu) uvodi se i posebna lozinka koju će znati samo vlasnik računara i pomoću koje će moći da poništi ostale lozinke i tako u potpunosti kontroliše svoju opremu.

Ako uz hard disk ne nabavite i neki poznati operativni sistem kao što je Unix ili MS DOS, nećete lako pronaći dobar aplikacioni softver! Ukoliko je, naime, osnovni operativni sistem vašeg računara dobro koncipiran, trebalo bi da programi koji rade sa flopijama rade i sa hard diskom, ali to još ne znači da ti programi mogu da iskoriste njegove potencijale. Ako je, na primer, autor nekog *data base* sistema prevideo najviše 1024 slova veličine od po jednog kilobajta (1024 K obično prevazilazi kapacitete flopija), program jednostavno neće moći da iskoristi 20 megabajta hard diska! Osim toga, autori programa koji koriste flopije se trude da minimiziraju broj pristupa disku, što znači da ne koriste sve potencijale vinčestera. Treba, dakle, postati Unix kompatibilan, ali tada možete da očekujete enormne cene softvera. O cenama ćemo, međutim, govoriti tek kad razrešimo probleme vezane za sigurnost podataka.

Kopija od 800 funti

Vinčester diskovi su, u celini gledano, veoma pouzdani, pa se često zanemaruje mogućnost gubitka podataka na njima. Čak i ako zanemarimo kvarove, sasvim je moguće da ćete u nekom trenutku greškom nepovratno izgubiti sve datoteke iz nekog direktorijuma. Zato je veoma važno da u vremena na vreme preprišete čitav sadržaj vinčester diska na neki drugi medij, obično na diskete. Za ovo prepisivanje će vam, na žalost, biti potrebno 10, 20 40 ili možda 60 disketa (zavisno od kapaciteta vinčester diska i flopija koji posedujete!) Umetati dvadesetak disketa u drajv i njihovo snimanje je posao koji može da potraje više od sata, što znači da se na njega nije lako odlučiti — možda ćete se radije odlučivati da se pozudate u svoju srećnu zvezdu. Ukoliko tako rezonezujete, poslušajte bar jedan savet: **obavezno** prebacite sav softver koji je proizvođač isporučio uz hard disk na flopije (ovaj je softver obično veoma skup i teško ga je samostalno nabaviti), a docnije, naredbom COPY, prepisujete sva svoja programerska remek dela, tekstoive i podatke na diskete; ako nekada budete morali da formatirate hard disk, utrošićete dosta vremena na pretraživanje disketa, ali ćete bar sačuvati rezultate svog rada.

Na tržištu mogu da se nađu i specijalizovani uređaji za prepisivanje kompletnog sadržaja hard diska na neku vrstu mikro-drajva. Iako veoma brzi (čitav se backup završi za dvadesetak minuta i to bez vašeg učešća), ovi uređaji imaju jednu veliku manu — koštaju čak i po 800 funti, ponekad više od samog hard diska! Moguća je, naravno, i kupovina dva hard diska, od kojih će jedan

PROIZVOĐAČI I MODELI

Ukoliko vas je naš napis zainteresovao toliko da ste počeli da razmišljate o hard disku, zanimae vas podaci o mogućnostima povezivanja popularnih personalnih modela sa njima. Podimo (abecednim) redom:

Apricot F1 je najprirodnije povezati sa „oficijelnim“ modelom vinčester diska MSD 3.5. Za 1100 funti ćete dobiti dva praktično nezavisna uređaja, od kojih je jedan običan izvor za napajanje, a drugi disk od 10 M! Operativni sistem se odlično uklapa u MS DOS, tako da hard disk biva automatski konfigurisan kao drajv A. Vaš stari flopi postaje drajv B.

Atari 520 ST, sa svojim izvanrednim karakteristikama, prosto plače za hard diskom. Na žalost, iako je na nedavnom sajmu časopisa PCW prikazan masivni disk povezan sa modelom 520 ST, Atari za sada ne pominje izlazak na tržište, specifikacije niti eventualne cene.

BBC mikrokomputer nije teško povezati sa hard diskom — za 1300 funti Acorn prodaje vinčester disk od 10, a za 1900 funti disk od 30 megabajta. U cenu je uračunat i interfejs (File Server Level III) kao i ROM sa ADFS-om, operativnim sistemom koji omogućava kreiranje hijerarhijskog stabla kataloga, zašti-

tu direktorijuma i mnoge druge stvari. Za daleko manje novca (460 funti) se od solidiska (0702 354674) može nabaviti hard disk od 10 megabajta sa pripadajućim softverom koji je u mnogim aspektima bolji od Acornovog. Na žalost, kompatibilnost ova dva sistema je prilično sumnjiva. Može se, najzad, kupiti i torch-ov Unicorn koji, zajedno sa mikroprocesorima MC68000 i Z80, nudi 512 K RAM-a, 20 M hard disk i čuvene operativne sisteme Unix i CP/M. Cena je „samo“ 2450 funti.

Sinclair QL se, za razliku od prethodnih računara, može opremiti jedino hard diskovima nezavisnih proizvođača. Firma CST (zovite ih na 0223 323302) prodaje hard disk od 10 M zajedno sa potrebnim interfejsom za 1200 funti. Za dodatnih 400 funti možete da dobijete i jedan flopi disk pošto ćete se teško odlučivati da radite backup na preko 80 kaseti za mikrodrajvove. Karakteristike uređaja su veoma povoljne, posebno vreme prenosa od 1.6 mikrosekundi po bajtu.

Sanyo 550 i 555 možete da snabdate masivnim diskovima firme Logitek (0257 426644). Cene su, na žalost, previsoke: 2000 funti za 10 M i oko 1200 za svakih sledećih deset. Mogu se, za uzvrat, nabaviti i „džinovski“ diskovi od 80 M.



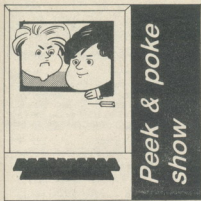
prevashodno biti korišćen za rezervne kopije.

Pitanje cene do koga smo u ovom prikazu konačno došli je i dalje bolna tačka masivnih diskova. Radi se o veoma preciznim i kvalitetnim uređajima koje nije lako izraditi i koje zato treba dobro naplatiti. Smatra se, osim toga, da su masivni diskovi profesionalni uređaji koje kupuju oni koji uz pomoć računara zaraduju novac. A takvi uređaji su dobra prilika za nabijanje cena!

Sam hard disk sa pripadajućim drajvom obično košta između 300 i 400 funti za svakih 10 megabajta. Potrebno je, naravno, kupiti i disk kontroler i odgovarajući operativni sistem, što će vas koštati daljih 200 funti. Ako vas suma od oko 1000 funti za

disk od 20 megabajta ne plaši, uzmite u obzir i cene softvera: kada jednog dana stupite u profesionalni svet, cene engleskih programa koje su vam izgledale astronomске u poređenju sa piratskom ponudom postaju prava pesma. Vrlo je verovatno da ćete za dobar tekst procesor, bazu podataka i spreadsheet platiti najmanje 1500 funti; fortran kompajler i slični sistemski programi mogu da budu i daleko skuplji! Zbog toga sa žaljenjem možemo da završimo ovaj napis konstatacijom da će, iako cene hard diskova neprekidno padaju, korišćenje njihovih potencijala zadugo izmicati mogućnostima jugoslovenskog hakera.

Dejan Ristanović



Reklama svuda oko nas

Reklamiranje priloga o računarima počelo je i pre pojave računarskih listova, što je i logično kada se ima u vidu već poslovična nesposobnost naših struktura, što neformalnih, što formalnih, da izađu u susret narodnim željama u svakom, pa i računarskom pogledu. Dakle, prvo su u drugim listovima nastali prilogi o računarima, pa specijalizovani listovi o računarima, a između te dve faze u našoj računarskoj revoluciji došlo je do podfaze reklamiranja pomenutih listova, a koju takođe možemo podeliti na tri oblika, i to reklamu u novinama, na radiju i na televiziji.

Štampa

Pošto su novine najstarije, najkласičnije i najnepromenljivije sredstvo javnog informisanja, jasno je da i reklama u novinama ne može da bude mnogo drugačija nego reklama za druge, neračunarske listove, pa se stoga dešava da se na isti način reklamiraju, na primer, "Računari", "NIN", "Danas", ali i "Praktična žena" i (o, zar se i to može?!), "Tempo" i "Zvezdina revija". Međutim, u novinama se malo može uraditi, pa da pogledamo šta je s

Radijlo

gde je situacija nešto bolja, jer se računarski listovi ne reklamiraju onako kako se reklamira keks Detić ili, pak nova ploča Hamida Ragipovića-Beska, nego se čak koriste i programički pomoću kojih se može očitati kompletan sadržaj lista. To što je taj zvuk prilično neprijatan, čak razdražujući za slušaocе koji vole već pomenuto Beska ili "Poslednju igru leptira", nije za nas bitno, jer je to vrsta ljudi koji, poput mamuta i australopitheksa, preti izumiranju. A mi, kojima pripada budućnost, krecemo korak dalje, tj. u

Televiziju

koju ovog puta treba pohvaliti, jer je predstavljanje računarskih listova i časopisa urađeno na prilično moderan način korišćenjem mogućnosti medija, mada ne u potpunosti, jer, recimo, nikad nismam video da se neki list ("Računari", na primer) oglašava pomo-

ću neke šaljive, duhovite, da ne kažem lepršave kompjuterske igre, recimo popularnog Pećinara (vlasnici "komodora" koji vole da se igraju znaju na šta mislim). No, reklamama (na televiziji i uopšte) nećemo biti zadovoljni svi dok se budu radile u paketi-ma sa drugim robama i dok ih budu radili ljudi kojima je do računara stalo toliko koliko im reklama za iste donosi bodove varijabla, što može poslužiti i kao Zarključak.

PEEK & POKE SHOW Raspisuje javni nagradni konkurs za

„NAJGORI MIKRORAČUNAR U 1985.“

Ako želite da učestvujete u ovom konkursu, potrebno je samo da nam pošaljete razglednicu, dopisnicu ili pismo (može i telegram) sa vašim mišljenjem o tome koji računar je bio najgori u upravo protekloj godini. Ako imate komentar ili objašnjenje zašto je upravo vaš izbor taj za koji mislite da jeste, napišite ga, to će nam puno pomoći.

Nagrade

Kada izbrojimo sve negativne glasove i kada ustanovimo koji je računar zvanično izabran za najgori u 1985, onda ćemo iz gomile onih koji su glasali za njega uz pomoć našeg specijalnog gosta (neka njegov identitet za sada ostane tajna) izvući tri srećnika koji će dobiti nagrade koje uzbuđuju.

1. nagrada — kasetna na kojoj kompletna redakcija „Računara“ peva pesmu „Program tvog kompjutera“;

2. nagrada — kasetna na kojoj urednici „Računara“ pevaju pesmu „Soba 23“;

3. nagrada — kasetna na kojoj priredivači „Peek & poke show-a“ pevaju pesmu „Ja sam izljudiv“.

Rezultati

Rezultati konkursa i imena nagrađenih će biti objavljeni u „Računarima 13“.

Ako odziv na konkurs bude veliki, nabavićemo još nekoliko specijalnih nagrada i deluho iznenađenja.

Pišite na adresu „Peek & poke show“, Rač. — Galaksija, BIGZ Bulevar vojvode Mišića 17, Beograd.

Vašar oplemenjuje

Izgleda da su računari stekli pravo građanstva (ili seljanstva) koje im kod nas tako dugo falj. Da, oni su postojali i pre, ali nisu baš pripadali ovom podneblju, nešto kao banane ili papagaji. Srećni događaji ponarodnjavanja računara dogodilo se na kermesu nazvanom Novogodišnji vašar. Pažljivi posmatrač i posetilac je mogao da primeti da pored čevapčića, original poliesterske svile, predratne mode, ogledalčadi, i sličnog probranog kiča, na Vašaru postoji i promocija Komodorovog računara C-128, koji je organizovao Sin-Com klub iz Beograda. Tako su, najzad, računari pušteni sa svog ezoteričnog filmskog mesta u narod, i šire. To su, verovatno, prvi udari računarske revolucije koja samo što nije počela. Sad će, samo što nije. Zato bi trebalo pažljivo zapisati datum kada se sa računari pojavili na Velikom Vašaru. Možda ćemo to slaviti jednog dana.

Uticaj kurseva na svetsko neznanje

Danas više ne možete da otvorite novine a da ne naletite na pravu gomilu reklama i oglasa za male škole bejzika, ubrzane kurseve mašinskog jezika i slične rasprodaje



znanja. Ideja nije naročito originalna, ali je — za organizatore — izuzetno korisna. To je evoluirana šema razvijena još pre mnogo godina na kursevima fotografisanja, dopisnim školama šivenja, slepim kursevima slepe daktilografije i sličnim pametnim tezgama. Ovog puta tema je: kako živeti od hakera? Izgleda da nije teško. „Peek & poke show“ je veoma diskretan pa zato neće pominjati imena, ali trenutno u našem glavnom gradu postoji desetak (najmanje) „kurseva“ koji pružaju onoliko računarskog znanja koliko i Mikijev Almanah. To i nije tako malo kad se uzme u obzir kakvi su neki predavači. Da bi obrazovanje (i zarada) bili kontinuirani, organizatori su se potrudili da razbiju kurseve u male, srećne celine, pa tako imamo početne, srednje, više, usavršene, specijalne, ubrzane i kondenzovane kurseve. Pravo bogatstvo za nas! Dobro gledajte gde ćete da naučite vaš bejzik. Nećete biti se malo! Grad je pun ljudi koji su svoj kurs blistavo završili i sad se blistavo kaju. Kursevi suzama ne veruju.

Pripremaju

Branko Đaković i Saša D. Kovačević

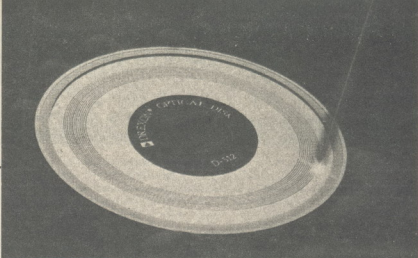
Kompjuterske mozgalice

Neupisivo upisivanje

Digitalni optički diskovi predstavljaju novi memorijski medij koji pruža nove perspektive. Disk prečnika 12 inča (1 inč=2,54 cm), sa cenom manjom od 100\$, može da memorise više od 100 milijardi bita pri današnjoj tehnologiji, a u neposrednoj budućnosti i znatno veće količine podataka pri još boljem odnosu veličina memorije/cena. Ovolika količina podataka zahteva, na primer, oko 40 kolovata magnetne trake, ali je vreme pristupa podacima na trakama daleko veće od 1/10 sekundi, koliko je potrebno kod video diskova.

Digitalni optički diskovi, međutim, imaju i jednu neugodnu osobinu za većinu korisnika: informacije se na njih upisuju samo jednom! U računarstvu, međutim, ništa nije nemoguće.

Tanak sloj telurijuma, na kojem se pomoću lasera stvara submikronska rupica, čime se označava prelaz sa inicijalnog stanja „0“ u stanje „1“ ako se želi upisati jedinice, ne može ničim da se regenise. Pri čitanju laser



jednostavnim primeru. Pretpostavimo da smo željenu informaciju zapamtili kodovanu brojevima 00,01,10 i 11 tako što smo koristili inicijalnu kolonu tablice 1. zapisa. Ako u sledećem upisivanju želimo da ostane upisani isti broj, onda ništa ne menjamo u zapisu, a ako želimo da upišemo neki drugi broj, upisujemo sa kodovanog kao u novoj koloni tablice 1.

Dekodovanje zapisa je jednostavno: reč abc predstavlja 2-bitnu vrednost (b'c), (a'c), bez obzira u kojoj se koloni nalazi zapis, tako da pri novom upisivanju u memoriju moramo prvo da iščitavamo me-

minisati potrebe za magnetnim diskovima. Profesor David Klarner (koji predaje matematiku na Univerzitetu države Njujork u Bingamponu), generisao je elegantan cikličan kod. Inspirisan klasičnom teorijom kodovanja, koji omogućava trostruko upisivanje. Kod je dat u tablici 3.

Koristeći Klarnerovu tehniku, student Nebraska univerziteta u Linkolnu David Livit je generisao kod koji omogućava četiri upisivanja koristeći 7 bita. Među ostalim autorima kodova ističe se Džems Sakse sa Karnegi Melon univerziteta u Pitsburgu, koji je pri tom koristio rekurziju i svojstva

tablica 1.

%	inicijalno stanje	novo stanje
00	000	111
01	100	011
10	010	101
11	001	011

tablica 2.

broj upisivanja	stanje pri jednom upisivanju
1	1.000
2	1.294
3	1.549
4	1.783
5	2.003
10	2.983
20	4.668
50	8.960
100	15.191
200	26.346

tablica 3.

%	prva generacija	druga generacija	treća generacija
0	10000	01001 00010	01111 10110 11001
1	01000	10100 00011	10111 01011 11100
2	00100	01010 10001	11011 10101 01110
3	00010	00101 11000	11101 11010 00111
4	00001	10010 01100	11110 01101 10011

osvetljava svaku poziciju i smanjena refleksija rupice lako se registruje. Veliki potencijal ovog memorijskog medijuma navele su Ronalda Rajvesta (Ronald Rivest) i Adi Samira (Adi Shamira), a zatim i veći broj drugih istraživača, da preispitaju suštinu jednodupisivosti optičkih diskova. Oni su uveli i posebno ime za ovu vrstu memorija „woms“ (write-once memories).

Pored digitalnih optičkih diskova u jednodupisive memorije, spadaju i PROMS (programmable read-only memories), kod kojih se topi spoj koji smo selektivno izabrali a omogućavao je protok struje, papirna traka i papirne kartice. Ako imate neki od ovih memorijskih medijuma, pokušajte da ih više puta iskoristite.

Ideja o višedupisivosti jednodupisivih memorija zasniava se na rezultatima teorije kodovanja i može da se objasni na sledećem

memoriju, ali pri čitanju memorije ne vodimo računa o tome da li je memorija upisivana jednom ili dvaput. Znakom * je označena operacija „ekskluzivno ili“ koja je realizovana na čipu. Rajvest i Samir su dali odgovor i na pitanje u vezi sa graničnim problemom: za koliko je manji broj bita koje možemo da upišemo pri N-tostrukom upisivanju nego pri samo jednom upisivanju. Deo njihovog odgovora dat je u tabeli 2. U prvoj koloni je dat broj upisivanja, a u drugoj veličina smanjenja memorije pri takvom upisivanju.

Pre nego što pogledamo pregled nekih kodova za višekratno upisivanje na jednodupisivu memoriju, primetimo da su digitalni optički diskovi posebno pogodni za čuvanje podataka koji su manje-više statični, kao na primer programi, dokumenti, slike ili baze podataka kod kojih se ažuriranje podataka na vrši odveć često, a takav deo baza podataka je velik. Posebnim tehnikama organizacije podataka na digitalnim optičkim diskovima mogu se potpuno eli-

simetrije. Ako vam se svideo princip višedupisivosti na jednodupisive memorije, a želite da konstruisete nov kod požurite — u literaturi, a polako i u praksi, stalno ih je sve više. Ipak, pored konstrukcije novih kodova, ostavljeno je još mnogo drugih pitanja koja traže odgovor (na primer, da li je svaki novoupisan kod veći numerički od prethodnog ako se računa zapis kao broj zapisa u binarnom broju):

- kolika je kompleksnost dekodovanja i preupisivanja;
- kako prevazići eventualne greške pri upisu;
- da li se svojstva višedupisivosti mogu primeniti i u drugim oblastima računarstva i nauke;
- kako upisivati u memorije koje imaju tri ili više mogućih stanja tako da se ova optimalno koriste; i
- koji su kodovi praktično korisni.

Miodrag Potkonjak

video digitajzer

U prošlom broju „Računara“ pisali smo o mogućnostima digitalizacije zvuka na kućnom računaru. Ovoga puta pripremili smo test uređaja za digitalizaciju slike — video digitajzera firme Print-Tehnik iz Minhena. Uređaj koji smo mi testirali radi na „komodoru 64“, ali se iste ili slične sprave mogu nabaviti i za ostale računare. Iako se još uvek nalaze u primitivnoj fazi, ovakvi uređaji značajno proširuju upotrebnost vrednost računara.

Video digitajzer je uređaj koji digitalizuje bilo koju sliku i prebacuje je na ekran i u memoriju računara za dalje korišćenje. Do pre nekoliko meseci ovakvi uređaji su izgledali kao pravo čudo tehnike. Bila je to velika, zastrašujuća ploča sa ogromnom količinom delova. Napretkom tehnologije, uređaj je sveden na dimenzije plastične kutije dugačke 8, široke 6 i debele 2 centimetra, koja se odlično uklapa u pozadinu računara. Na samom uređaju se nalaze priključak za video kameru i dva mala potencijometra za podešavanje osvetljenja i kontrasta. Sam uređaj se priključuje na paralelni izlaz računara i radi vrlo jednostavno. Uz uređaj se dobija i jedna disketa, na kojoj se nalaze programi koji mu omogućavaju da radi, i nekoliko demonstracionih programa.

Kada učitate program, na ekranu dobijate lepo urađeni meni. Pored digitalizacije, autori softvera su najviše pažnje posvetili štampanju i snimanju slike. Kursor treba dovesti do prozora na kojem piše DIGITIZE i pritisnuti RETURN. Ekran će se zamračiti i proces digitalizacije će početi. Ceo proces digitalizacije jedne slike traje oko četiri sekunde i posle njega na ekranu se dobija deo slike koju ste digitalizovali. Digitajzer nije ograničen samo na rezoluciju samog računara. Kod „komodora“ uređaj digitalizuje sliku veličine 480 puta 400 tačaka, pri čemu se samo deo vidi na ekranu. Pomoću tastera za vođenje kursora ekran se može šetati po slici sve dok se ne odabere najbolji položaj. Ovo je mnogo zgodnije, jer kameru možete fiksirati, a sliku pomerati u svim pravcima.

Obrada slike

Kada dobijete prvu sliku, nemojte se odmah razočarati (kao mi) ako slika bude loša. Potrebno je malo znanja i prakse da bi se dobila najbolja slika. Pri tom je vrlo važno podesiti osvetljenje i kontrast na digitajzeru, jer to bitno utiče na kvalitet slike.

Kada dobijemo onakvu sliku kakvu smo želeli, vraćamo se u meni i biramo opciju koja nam najviše odgovara. Kao prvo, samu sliku treba snimiti i sačuvati za buduću upotrebu. To možemo uraditi na više načina. Moguće je, pre svega, snimiti kompletnu sliku koju smo dobili (480×400), ali ćemo takvu sliku moći da koristimo samo uz softver samog digitajzera. Na sreću, na raspolaganju imamo još nekoliko mogućih



Korak dalje: Video digitajzeri značajno proširuju upotrebnost računara

načina snimanja. Najzgodnije je sliku snimiti tako da se posle može ponovo iskoristiti u nekom programu za crtanje ili grafiku. Na raspolaganju su vam KOALA-SAVE i PAINT-SAVE komande, pomoću kojih se dobijaju fajlovi na disk koji se mogu učitati u KOALA-PAINTER ili PAINT-MAGIC. Nakon toga na slici se mogu vršiti sve manipulacije koje dozvoljavaju ovi programi. Ovi fajlovi se lako mogu preraditi i za ostale programe za crtanje (DOODLE, BLAZING PADDLE...), a korisno je i slike prebaciti u oblik kakav će moći da se učita u SIMON'S Bejzik ili neki drugi prošireni bejzik (G, GR). Tako možete slike koje ste dobili i vrlo praktično da примените.

Pošto smo sliku sačuvali za dalju upotrebu, mogli bismo i da je odštampamo. Nijedan vlasnik „komodora“ ne treba da se plaši da program neće raditi sa njegovim printerom. Na raspolaganju su vam opcije za sve printere koji se mogu priključiti na „komodor“. Možete, čak, koristiti i paralelni printer — i on je predviđen programom. Treba samo da kažete koji printer imate i kako je priključen i štampanje će početi. Program štampa kompletnu sliku, znači 480×400, pa je moguće dobiti dosta veliku sliku na papiru. Na disketi je dat i poseban program koji na Epsonovim i još nekim kvalitetnijim printerima štampa slike u 16 crno-belih nijansi, što izgleda zaista impresivno. Ukoliko želite sliku manjeg ili još većeg formata, možete je odštampati i z

nekom drugom programu koji podržava štampač.

Savršeni alarm

Pored programa za šesnaest boja, na disketi se nalazi još nekoliko interesantnih i korisnih programa. Jedan od njih je i alarm. Fiksirate kameru u jednom pravcu, digitalizujete sliku i uključite alarm. Računar će svakih pet sekundi ponovo digitalizovati sliku i u slučaju da se pojavi neka razlika između dve slike, začuće se grozan zvuk sirene za uzbunu koja će raditi sve dok ne pritisnete neki taster na računaru, a tada sve ide iz početka. Pre startovanja alarma možete odrediti računaru na koliku promenu da reaguje. Program, u stvari, radi tako što sabira binarne vrednosti cele slike i ako razlika zbroja dve slike pređe određenu vrednost, uključuje se zvuk sirene. Osim ovog programa, na disketi se nalazi još nekoliko korisnih programa i nekoliko demonstracija. Posebno je interesantna digitalizovana slika prodavca iz radnje koja uređaj prodaje, jer je slika zaista urađena na zavidnom nivou.

Sve u svemu, jedan vrlo interesantan dodatak za vaš računaru, koji bi mogao da ima vrlo široku primenu. Može se koristiti za izradu slika u finoj grafici, kao čuvar (alarm) ili za nešto sasvim treće. Ako imate veliku količinu raznih slika ili crteža koje treba čuvati, možete da ih sve digitalizujete i spremite na nekoliko disketa.

Vladimir Krstošić

Računari
u izlogu

SUPERKOMPJUTERI '86.

džinovni u bocici

Rešili ste da kupite novi računar? U redu, kakav bi on trebalo da bude? Pa, naravno, što brži, sa što većom memorijom i... Ako imate desetak miliona dolara i strpljenja da sačekate kraj (ili bar sredinu) ove godine, a zadovoljavaju vas brzine od nekoliko hiljada megafloпова (flop je jedna operacija u pomoćnom zarezu u sekundi, a megaflop je milion floпова — IBM 3081 K operiše sa oko 2,3 megafloпова, VAX 11/780 sa oko 0,3, a VAX 11/750 oko 0,2 megafloпова) i memorije od nekoliko milijardi bajtova, onda pažljivo pročitajte tekst, proučite tabele i odaberite nešto dobro za sebe ili svojoj devojci za rođendan.

Što treba gledati kad kupujete superkompjuter? Isto što i kod bilo kog drugog računara: hardver, sistemski i aplikativni softver.

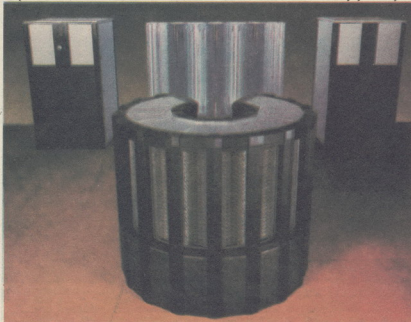
Egzotične tehnologije

Hardver je osnovni preduslov za brži računar. Veće brzine postižu se, zahvaljujući, pre svega, povećanoj brzini komponenti, smanjenju vremena prenošenja informacija smanjenjem dužine veza između komponenti i povećanjem suštine pakovanja, ali i promenama u arhitekturi, koja omogućava paralelno izvršavanje operacija. Povećanje brzine komponenti je ograničeno, u najoptimističkim slučajevima, brzinom svetlosti ili nekom drugom brzinom koju daje teorijski fizika. Tri tehnologije koje najviše obećavaju u ovoj i nekoliko sledećih godina su brza CMOS kola koja se hlade tečnim nitrogenom sa temperaturom od 77 Kelvinih stepeni, kola bipolarna integralna kola i GaAs kola koja će, najverovatnije, dominirati i početkom sledeće dekade.

Krajem veka i početkom sledećeg doći će do izražaja i naprednije i egzotičnije tehnologije koje su još u fazi ispitivanja: molekularna episkijalna, komplementirani galijum-arsenid, galijum-aluminijum arsenijski tranzistori sa velikom mobilnošću elektrona i Dtozefsonov (Josephson) spoj. Najviše nade se polaže u GaAs-A16As tranzistor koji može da menja stanja za vreme koje se meri pikosekundama (milionitni delovima milionitog dela sekunde), a istovremeno zrače izuzetno malo toplote. Sve ostale tehnologije su bar za red veličine slabije od ove. Ovaj tranzistor se naziva i HEMT (high-electron-mobility transistor) ili MOD-FET (modulation doped field-effect transistor) ili TEGFET (two-dimensional electron-gas FET) ili SDHT (selectively doped heterojunction transistor) ili...

U znaku Unixa

Sistemski softver doprinosi povećanju brzine generisanjem jezika koji omogućavaju efikasnost u kodovanju onih algoritma koji je u skladu sa arhitekturom. Do ove godine operativni sistemi su bili najboljnija tačka superkompjutera. Primera radi, Sajber 205 nema mogućnost istovremenog rada više korisnika, što znači da nije zgodno ulaziti u kupovinu sa prijateljima. Nedav-



Najbrži na svetu: Novi „krej“, „krej-2“, sastoji se od 240.000 čipova koji troše između 300 i 500W, ali mu to ne smeta da radi sa klokom od svega 4,1 nanosekunde (procesor Z80A je sporiji 60 puta)

no je Convex Computer Corp. iz Ričardsona u Teksasu implementirala na svom 64-bitnom superračunaru Convex C-1 operativni sistem Unix 4.2. Po rečima Bartona Smita, šefa Deneclor Inc. iz Aurore u Koloradu, za njihov HEPe HEPI Krejove kompjutere Unix je rešenje za problem oko operativnih sistema, iako ima i drugačijih mišljenja.

Glavni programski jezik većine superkompjutera je fortran. Fortran kompajler imaju STAR-100, svi Krejovi (Cray-1 dva: CFT i CIVIC, Cray X-MP ima takođe dva: CFT 110 i CFT 113+), 100 megaflopni ASC firme Texac Instruments i Burroughs-ov BSP. Sajber 205 ima kompajler za fortran 4, a i CDC-ov metaassembler. Na HEP-ovima je realizovan fortran koji ima posebne pozive (Create) koji omogućavaju odlično korišćenje arhitekture. Britanski tridesetdvo-bitni matricni procesor DAP (Distributed Array Processor) firme ICL, koji može da izvrši 125 miliona sabiranja i 11 miliona množenja u sekundi, ima, pored fortran kompajlera, i

algot kompajler, a i poseban jezik DAP. NASA je kupila za obradu slike Zemljine površine sa satelita kod američke firme Goodyear Aerospace Corp za 4,7 miliona dolara izuzetno moćan matricni procesor MPP (Massively Parallel Processor) koji je izrađen integralnim kolima sa silicijumskim strukturama na safiru, a ima 6 milijardi sabiranja u sekundi i 2 milijarde množenja kada je dužina reči 8 bita, a dva, odnosno devet puta manju brzinu kada je reč dužine 12 odnosno 32 bita. Ovaj procesor se, međutim, ne može programirati ni na jednom jeziku visokog nivoa. Najjači matricni procesor, Burroughs-ov Phoenix, koji može da uradi više od 10 milijardi operacija nad 64 bitnim rečima, a koji preporučujemo ljubiteljima dobrog zvuka jer se koristi za rešavanje akustičkih zadataka, programira se, takođe, na fortranu.

Bezjik i paskal ne postoje ni na jednom superkompjuteru, ali zato za one kojima je bliska ada, a imaju veze u američkoj armiji, preporučujemo procesor Mark IIa koji je proizveden u Lourensovoj laboratoriji u

Livermoru. On radi na adi brzinoj od 400 megaflopa, dužinu reči od 18,36 i 72 bita, a košta svega 300.000 dolara. Koga ova brzina ne zadovoljava, a ne može bez programiranja na adi, neka za 6 miliona dolara kupi, naravno, opet ako ima jake veze, S-1 Mark IIA brzine 1000 megaflopa. Inače, ovaj procesor koristi američka mornarica za brzu obradu podataka i upravljanje gadanjem, pa je izuzetno pouzdan i ima poseban procesor za dijagnostiku i remont. Niska cena je postignuta korišćenjem automatizovanog sistem projektovnja SCALD (Structural Computer-Aided Logic Design), koja sadrži standardne programe za funkcionalno, logičko pa čak i vremensko modeliranje čvorova i elemenata na svim nivoima hijerarhije projektovanja. Korišćenjem SCALDa vreme projektovanja Marka I je sa 100 godina-ljudi smanjeno na 2 godine-čoveka.

Nije sve u algoritmu

Aplikativni softver donosi sve efikasnije algoritme i tako zahteva sve manji i manji broj koraka da bi se došlo do željenog rezultata. Kada kupujete superkompjuter, pazite da ima vama potreban aplikativni softver ili barem da vi znate kako ga možete generisati na hardveru i sistemskom softveru koji kupujete.

Koliki je udeo algoritma i softvera u brzini, a koliki hardvera? Odgovor na ovo pitanje je, možda, pomalo iznenađujući, a može se dobiti gledanjem tabele 1. u kojoj je dat razvoj brzine algoritma za rešavanje trodimenzionalne eliptičke parcijalne jednačine (to je tipičan zadatak za jedan superkompjuter) i istorijski razvoj i brzina najbržih kompjutera u prošlosti. Uporedite i današnje računare sa nekadašnjim superkompjuterima. Biće vam jasno zašto je Najl Linkoin iz CDC-a definisao superkompjuter kao sistem koji je za samo jednu generaciju ispred računarskih potreba nauke i tehnike formulom

SUPERKOMPJUTER + KOLOSALNA IZRAČUNAVANJA + ENORMNA OČEKIVANJA + FANTASTIČAN RIZIK

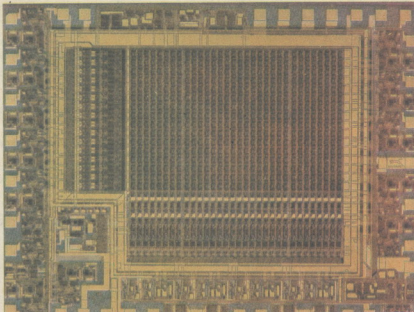
i zašto naslovi njegovih članaka glase „Nije baš zabavno napraviti superkompjuter, kao što je jednostavno projektovati“ ili „Razvoj superkompjutera — zadovoljstvo i bol“.

Sam algoritam nije dovoljan za brzu realizaciju na računaru: ono što je brzo na jednom računaru, čak i na vrlo sličnom može biti sporo. Upečatljiv primer navodi Piter Benion: program koji se izvršavao brzinom od 2,4 megaflopa na CDC 7600 imao je, direktno prenet, brzinu od 2,1 megaflopa na Sajberu 205, što je, preuredivanjem programa, povećano na 110 megaflopa! (Sajber 205 ima 9 puta veću brzinu od CDC 7600 i proistekao je iz njega preko STARa 100 i Sajbera 203.)

Oni koji su bili

Šta će nam biti dostupno u 1986. godini? Pogledajmo tabelu 2. Za ljubitelje starijih dajemo podatke o superkompjuterima iz 1983. pa čak i iz 1982. godine u tabeli 3.

U tabeli 2 ste sigurno uočili poveću količinu znakova pitanja, jer sem šturih



Sumrak silicijuma: Nanoram na bazi galijum-arsenida kapaciteta 1 kilobit (256x4 bajta) ima vreme pristupa od svega 1 nanosekunde

Klejevskih (čitaj Krejevskih) izjava malo šta se pouzdanije zna o ovim kompjuterima, sem njihove hardverske osnove i činjenice da su sigurno proizašli, kao i uvek do sada, iz prethodnih modela. Pogledajmo, zato, istoriju iz koje se ekstrapolacijom može zaključiti o onome što ćemo videti u ovoj godini.

Sajber 205 je počeo da se realizuje početkom 1979. godine, a usavršavanje u

godine. CDC FORTRAN na Sajberu ima posebne programe za pozivanje vektorskih operacija. Cena je do 7,9 do 16,5 miliona dolara.

Cray-1 ima maksimalnu brzinu od 160 Mflopova, vreme sata od 12,5 nanosekundi i košta oko 4,5 miliona dolara. Proistekao je iz CDC6600/7600 mašina koje je Krej projektovao dok je u Mineapolisu radio za CDC. Cray X-MP je poboljšan tako da je

Tabela 1.

algoritmi	godina	broj operacija	broj operacija za n=1000
Gausova eliminacija	1945	n^{**7}	10^{**21}
SOR iteracija (suboptimalna)	1954	$8n^{**5}$	8^{**15}
SOR iteracija (optimalna)	1960	$(8n^{**4}) \log(n)$	8^{**13}
ciklična redukcija	1970	$(8n^{**3}) \log(n)$	8^{**10}
multigrad metode	1978	$60n^{**3}$	6^{**10}
kompjuter	godina	megaflopa (prosečno)	
Manchester Mark 1	1947	0,0002	
IBM 701	1954	0,003	
IBM Stretch	1960	0,3	
CDC 6600	1964	2	
CDC 7600	1969	5	
Cray-1	1976	50	
cOpCray-2	1985	125	

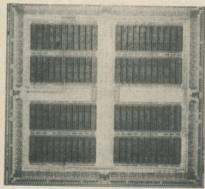
odnosu na ranije modele se sastojalo u dodavanju bloka za skalarnu obradu podataka i VLSI memorije koja ima korekciju eventualnih jediničnih grešaka i detekciju dvostrukih grešaka pri prenosu u ili iz nje. Skalarni procesor ima 5 funkcionalnih blokova sa pokretnim zarezom, upravljački i asocijativni blok. Vektorski procesor ima dva aritmetička bloka i blok za komunikaciju sa memorijom. U skalarni procesor podaci mogu da ulaze svakih 20 nanosekundi, a u vektorski svakih 40. Sajber ima 12 kanala za rad sa periferijama i mogućnost izdavanja i primanja podataka brzinom od 50 megabita u sekundi. Prva mašina je postavljena i puštena u rad novembra 1980.

dostigao performanse iz tabele 1, u stvari, koristi dve Cray-1 mašine. U prodaji je i verzija X-MP sa četiri Cray-1 računara. Cena od 5 do 14 miliona dolara.

Hitachi je lansirao S810/20 koji koristi prošireni M280H (računar kompatibilan sa IBM-om) i radi sa jednom do pet vektorskih jedinica. O kvalitetima Fujitsu-ovog VP-200 dovoljno je reći da je u svim testovima američkih vrhunskih računarskih institucija pokazao bolje performanse od Cray X-PM-a.

Krej se odvojio od CDC (Control Data Corp.), a Amdal (Amdahl) od IBM. Kod Amdala se mogu kupiti dva vektorska procesora: modeli 1400 i 500 koji imaju brzinu

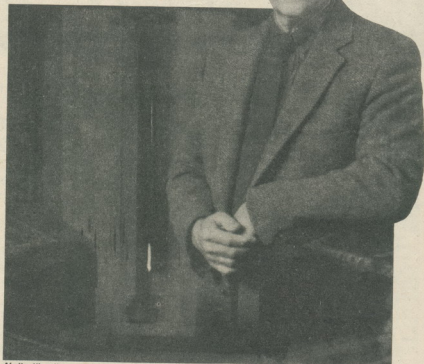
od 1140 odnosno 133 megaflopova i memorije je od 256 odnosno 128 megabajta. Dok je model 500 moguće nabaviti po ceni od 4,9 do 9,7 miliona dolara, za model 1400 treba sačekati sredinu godine, spremili između 12,8 i 21,8 miliona dolara i naručiti ga na adresu: Amdahl Corp., P.O. Box 470 Sunnyvale, CA 94086, USA. O svim uslovima kupovine, ako vam se žuri, možete se raspitati na telefon (408) 746-6000.



Superbrzi logični niz: Sprezna kola na bazi ECL tehnologije unose kašnjenje od samo 150 pirosekundi

... i oni koji dolaze

Možda najlepši i najinteresantniji projekti u vezi sa kompjuterima visokih performansi vode se na američkim univerzitetima (na Kalifornijskom tehnološkom institutu Cosmic Cube, Data Flow na MIT-u, Pyramid



Najbolji mikro-arhitekta sveta: Tvorac „kreja-1“ i „kreja-2“ Sejmur Krej

Tabela 2.

	CRAY 2	CRAY 3	ETA GF10	DENELCOR HEP-2	NEC SX-2
PERFORMANSE					
MAKSIMALNI BROJ MFLOPOVA	1000	?	10000	4000	1300
vreme sata (ns)	4	1	?	12.5	6
MAKSIMALNA MEMORIJA (miliona bajta)	?	?	256	4000	256
TEHNOLOGIJA					
logička kola	ECL	ECL(GaAs)	CMOS	ECL	?
glavna memorija	MOS	MOS	MOS	MOS	64 K MOS statička
HLADENJE	tačno	?	***	vazduh	voda

Tabela 3.

	FUJITSU VP-200 nov. 1983	HITACHI S-810/20 nov. 1983	CRAY 1M sep. 1983	CRAY K-MP jun. 1983	CDC 205 feb. 1982.
PRVA INSTALACIJA					
PERFORMANSE					
maksimalni broj mflopova	500	630	250	630	400
vreme sata (ns)	15/7,5	15	12,5	9,5	20
VELIČINA GLAVNE MEMORIJE (milioni bajta)	256	256	32	32	64
TEHNOLOGIJA					
logičkih kola	ECL	ECL	ECL	ECL	ECL
glavna memorija	MOS	MOS	MOS	4K ECL	MOS
HLADENJE	vazduh	voda	freon	freon	freon

putera, na tržištu će se uskoro pojaviti i rezultati britanskog Alvey projekta, Esprit projekta Evropske komisije za komunikacije i nemačkog nacionalnog projekta. U Kini je montiran Cray-1, a realizovan je i kompjuter sa preko 30 megaflopova.

U okviru evropskog projekta Eureka, francuska Matra i norveška mini kompanija Norsk data dobile su ugovor sa francuskom vladom za izgradnju superkompjutera brzine 100 Mflopova koji treba da bude završen do 1988. godine. Francuski Bull i nemački Siemens treba do 1990. godine da završe znatno moćniji računar.

Ako posle svega ovoga niste mogli da se odlučite šta da kupite, a treba vam stvarno stručan savet, preporučujemo vam dva vrhunska stručnjaka za superkompjutere: Miloša Ercegovca i Danijela Gajskog. Ercegovac je šef Computer Science Departmenta na UCLA univerzitetu u Los Angelesu, a završio je elektrotehniku u Beogradu. Gajski je profesor na Ilionis univerzitetu u Urbani Sampaň, a elektrotehniku je završio u Zagrebu. Obojica su doktorirali u SAD-u. Ercegovac je, primera radi, predavač na predavanjima o superkompjuterima koje IEEE organizuje širom Amerike.

na Ilionis univerzitetu u Urbani-Sampaň, WISPAC na Viskonsin univerzitetu, RISC na

Stenfordu i Berkliju, na Perdjuu kompjuter za obradu slika i mnogi drugi). U Evropi je posebno uspeo projekat Data Flow kompjutera na Mančesterskom univerzitetu u Engleskoj.

Pored američkih i japanskih superkomp-

Miodrag Potkonjak

hakerski marketing

Šta ne programirati

Da vidimo, pre svega, šta nije lakše uraditi pomoću programa za računar.

(1) Ne treba pisati program koji će se izvršavati jedanput ili dvaput godišnje. Za pisanje programa potrebno je ručno napraviti jedan ili više test-primera, i napisati program (tj. u opštim terminima izraziti problem i njegovo rešenje), a to znači da čemo tokom stvaranja programa problem ručno rešiti najmanje dvaput. Kada se, tome, dodaju neizbežne greške u toku procesa planiranja, kodiranja i testiranja programa — onda postaje jasno da nema smisla pisati program koji se neće izvršavati bar tri-četiri puta godišnje.

Sledeće pravilo je, međutim, važan izuzetak.

(2) Treba napisati program koji omogućava da se uradi neki težak „pešački“ posao, pa makar i jednom godišnje. Ovo je jedan od najvažnijih razloga za široku prihvaćenost računara. Mnogi naši inženjeri kupili su računar da bi svoje standardne proračune vršili lakše i brže, a studenti su prigrlili programabilne ručne kalkulatora kao izvanrednu pomoć na pismenim ispitima.

(3) Ne treba pisati program ako se dati posao može obaviti ručno za manje od pet minuta. Računari i njihovo programiranje mogu ponekad da iskomplikuju život. Nema baš mnogo smisla pisati program koji odgovara na pitanje „Koliko će mi biti plata ako dobijem povećanje 12.5%?“, jer se to može izračunati napamet, ili pomoću kalkulatora, pa čak i pomoću računara u direktnom režimu rada. Sve nas to navodi na sledeće pravilo:

(4) Ne treba pisati program ako se cilj može dostići pomoću ručnog kalkulatora, kutije podsetničkih listića, ili pomoću običnog kalendara i olovke. Ovu vrstu „knjižovodstvenih“ poslova kućni računari komplikuju učitavanjem programa, pa morate da imate računar pri sebi itd. Pre nekoliko godina baš ovakve primene su navedene kao idealne za kućne računare, ali to uglavnom nije tačno. Neki fizički medijumi za čuvanje podataka, jednostavno, nisu praktični. I najbolji program za tekući račun ostaje nekorisćen na računaru koji ima samo kasetofon kao spoljnu memoriju.

(5) Ne treba pisati program ako tačnost nije bitna. Čemu podatak sa ekrana da je za tortu potrebno meriti 328 grama šećera, ako se količina meri suprenom kašikom a ne apotekarskom vagom?

(6) Ne treba ponovo izmišljati točak! Treba se raspitati da li se neki postojeći program može iskoristiti? Ako problem zahteva čist numerički proračun nad nizom

matrica, onda je bolje nabaviti program za unakrsna izračunavanja (spreadsheet), nego mučiti se na bezjuki. Neracionalno je praviti adresar kao zaseban program, spisak knjiga u kućnoj biblioteci kao zaseban program, sopstvenu bibliografiju kao zaseban program, jer sve to treba da radi jedna dobra baza podataka.

Poznavanje bibliografije gotovih potprograma je od posebnog značaja. Na primer, za jezik C nude se posebne biblioteke za rad sa bazama podataka, za grafiku, za rad sa ekranskim prozorima, za konverziju bezjuki programa u C-programe, i tako dalje. Na fortanu postoji ogromna biblioteka gotovih numeričkih potprograma, svaka knjiga iz paskala sadrži korisne procedure i funkcije, a računarski časopisi takođe mogu biti značajna riznica tuđih iskustava.

Gornji kriterijumi pomažu da u želji za programiranjem ne zalutamo u pisanje trivijalnih programa. Razmotrimo, slobodno, šta treba obratiti pažnju prilikom izbora teme za „sledeći program“.

Vera u srečnu zvezdu

Osnova za pisanje uspešnog programa je pravilo koje se često navodi u knjigama na temu „1001 način za brzo bogacenje“, a ono glasi: treba otkriti jednu ljudsku potrebu, a zatim joj udovoljiti. U skladu s tim, treba prvo ispitati tržište, pa proceniti svoje mogućnosti.

Istraživanje tržišta je bitan deo pripreme za pisanje novog programa. Na prvi pogled izgleda neobično, čak i nepotrebno, ispitivati tržište mikro-računarskog softvera. Najveći broj programa pišu ljudi koji računare koriste za sopstvene potrebe, a ne kao medijum za razmenu i prodaju softvera. U programu za ličnu upotrebu kozmetika u vidu menija, HELP-naredbi, legot štampačnja i sličnog, jednostavno nije ni potrebna. Ipak, program koji napišete za sebe u najvećem broju slučajeva ne biva uništen; naprotiv, često ostaje u svakodnevnoj upotrebi. Takav program primete drugi programeri i korisnici, zainteresuju se, zamole za kopiju, i time sve počinje iznova: još jedan nepažljivo sročeni program ulazi u svakodnevnu upotrebu. . . Srećom, postoje i pozitivni primeri, poput programa Sidekick, napisanog za IBM PC i klonove. To je neka vrsta telefonske sekretarice za profesionalne programere: pamti sastanke, prekida rad računara u određeno vreme, čuva pribeležke, telefonske brojeve, radi i kao ručni kalkulator i slično. Program je nastao iz čisto privatnih potreba programera u Borslandovoj kompaniji (proizvođač Turbo Paskala), koje su bili često prekidani telefoničnim, sastancima i sličnim. Tek posle nekoliko meseci intenzivne upotrebe „unutar kuće“, firma je primetila da u rukama ima pravo komercijalno blago. . .

Ako ste zakleti tvrdica, i svoje programe nikom nikada nećete davati ni prodavati, čak i tada ima smisla praviti „ljubazne“ programe. Možda ćete se datom programu vratiti kroz nekoliko godina, a tada sleduje neminovno čuđenje u stilu, „ček, ček“, šta sad treba da radim, samo da se setim. . .“ i u slučaju da ste vi svoje sopstveno „tržište“, treba se potruditi oko početnog dizajna programa.

Računarska industrija još uvek boluje svoje dečije bolesti. Mnogi profesionalni programi započeli su svoj život bez skoro ikakvog istraživanja tržišta. Međutim, tržište softvera se izuzetno brzo menja, pa će uvek biti mesta za razvoj programa bez sigurne vizije o plasmanu na tržištu, samo sa verom u sopstvenu srečnu zvezdu.

Programiranje za budućnost

Osnovno pravilo istraživanja tržišta je do krajnosti jednostavno. Treba samo naći odgovor na pitanje „Šta ljudi žele?“. Ovdje razmatramo stvaranje uspešnih računarskih programa, pa to opšte pitanje pretvaramo u: „Koje oblasti primene personalnih računara znam dovoljno dobro da mogu da ocenim šta im ljudima potrebno i šta žele da im se napravi?“. U slučaju programa za slobodnu prodaju, pitanje se mora postaviti nekoliko drugačije: „Šta to ljudi hoće i mogu da plate?“. Sva tri pitanja potencijalno da stvaralač programa pre pisanja programa mora da poznae svoje buduće kupce. Akcenat je na reči „budući“. Za stvaranje dobrog programa potrebno je nekoliko meseci ljudskog rada. Ako danas započnete svoje remek-delo, ono će biti spremno kroz godinu dana, dakle, morate oceniti šta će tada biti aktuelno na tržištu. Ne treba procenjivati sadašnje tržište — tu nemate šta da tražite, već tržište koje će tek postojati! Ovo je lakše reći nego učiniti, ali je baš to ključ uspeha.

Ima mnogo načina da saznate nešto više o svojim budućim kupcima. U slučaju da ste svoj sopstveni „mušterija“ i da pravite program iz oblasti svoje specijalnosti, možda ste vi upravo tipičan predstavnik cele jedne kategorije stručnjaka! Najbolji specijalizovani programi iz npr. inženjerskih oblasti ili štatističke, stvoreni su upravo na taj način. Štolski primer je nastanak programa Visicalc, odnosno programa za unakrsna izračunavanja tabela uopšte. Na tu ideju je došao jedan student ekonomije (!) da bi se olakšao predviđanje uspeha nove tržišne strategije kompanije Coca-Cola.

Život se komplikuje ako nameravate da pišete programe za komercijalnu upotrebu, odnosno prodaju. Možda za nameravani program već postoji tržište, a vaš program treba da bude u jednom ili više detalja bolji od postojećih? Tada je bitno da saznate koliko je sličnih programa već prodato, koje su prednosti na strani konkurencije,

Posle izvesnog vremena provedenog uz računar, bezjick i ostali programski jezici postaju sredstvo za dostizanje nekog cilja. U prvo vreme uspeh je napisati program koji radi. Kasnije, kodiranje, tj. ispisivanje naredbi na programskom jeziku, postaje stvar rutine, a u prvi plan izbija svrha programa. Postoje dva osnovna pravca produktivnog iskorišćenja te razudene intelektualne snage: ne treba pisati trivijalne programe, odnosno treba dobro proceniti kakve su potrebe za neovim programom kojeg nameravate da napišete. Bogatstvo se teško može steći pisanjem programa kakve biste sami voleli da imate, već samo pisanjem programa koji su potrebni velikoj armiji korisnika.



kakav je trend u razvoju softvera, kakve su bile kritike po časopisima itd. Osim primarnog, da li postoje i sekundarna tržišta, a zatim da li se tržište u Evropi razlikuje od tržišta u SAD, i slično; koliki obim tržišta nameravate da obuhvatite da biste bili zadovoljni, a, recimo, koji deo tržišta pokriva vaše troškove, i tako dalje.

Mogućnosti procene su znatno sužene ako vaš program predstavlja stvarnu novost na tržištu. Bez pravih informacija, možete podeliti potencijalne mušterije u dve klase. U prvj su svi oni koji bi kupili kompletan računarski sistem samo zbog vašeg programa ili zajedno sa njim. Najpoznatiji primer te vrste je Visicalc, za koji kažu da je prodao nekoliko stotina hiljada računara Apple II. Danas je slična situacija sa progra-

mom Lotus 1-2-3 i računarima tipa IBM PC, a na kućnim računarima postoji primer avanture Elite zbog koje je prodan izvestan broj računara BBC model B, dok bi se slično moglo reći za Hu-BASIC u odnosu na računar Sharp MZ-700/800.

U drugoj klasi su svi kupci koji već imaju neki računar, pa bi za njega hteli da kupe vaš program. Zatim možete da uporedite cenu vašeg proizvoda sa prihodima potencijalnih kupaca i da procenite koliko vaš produkt može da im uštedi ili donese para (direktno ili indirektno). Na osnovu svega toga možete da utvrdite kojoj grupi želite da ponudite vaš program i po kojoj cen.

Kako prodati program

Nije dovoljno imati kvalitetan produkt — treba ga na pravi način i reklamirati. Marketing je tradicionalno slaba tačka računarske industrije, mada je otprve jasno da niko ne može da kupi vaš program ako i ne zna za

njegovo postojanje. Osnovni princip marketinga je zavodljivo jednostavan: treba privući pažnju na svoj proizvod, a zatim je tu i održati. Dva su standardna načina za to: reklama preko novina i specijalizovanih revija, odnosno, reklama kroz prodavnice koje će vaši potencijalni kupci posetiti. Trenutno u Jugoslaviji ne postoji široka mreža prodavnica koje bi bile isključivo posvećene računarima. Ako se u prodavnici muzičkija prodaju i računarske kasete, onda bi bilo dobro oformiti specijalne štandove tako da računarske kasete budu odvojene od muzičkih, napraviti prospekte sa kratkim opisom programa koje bi radoznali kupac mogao da ponese kući i da se na miru odluči, zatim atraktivno pakovanje sa npr; pravilima igre, ili osnovama dokumentacije za program, a i da ne spominjem demonstracije programa na računaru uživo. Uzgred, među velikim softverskim „proizvođačima“ sve prisutniji je trend prodaje demo-verzija programa, tako da kupac za

znatno manje para može da se iz prve ruke uveri šta program radi.

Uspesha programa bitno zavisi od kvaliteta prateće dokumentacije. Rad programa se obično prikazuje kroz slike ekrana. Nije lako napisati dobar priručnik. Problem je u tome što je krajnjem korisniku svejedno kako program radi „Iznutra“ — ako dobro radi. Dokumentaciju prečesto pišu sami pisci programa, a oni na svoje čedo gledaju pristrasno i kroz priručnik demonstriraju briljantno poznavanje računara, mašinskog jezika i slično. Moglo bi se pomisliti da je odsustvo bilo kakve dokumentacije najgora vrsta dokumentacije, o čemu bi kupci piratskih programa sigurno imali šta da kažu. Međutim, od nepostojanja dokumentacije još je gora netačna dokumentacija, koja se često sreće kod novih mašina na tržištu.

Dokumentacija treba da se sastoji iz najmanje dve vrste priručnika. Vašem kupcu je prvo potrebna početnica za korišćenje programa, dakle uputstvo kako da odmah uz pomoć programa počne da radi nešto konkretno. Korisniku u toj fazi nisu potrebne dugačke liste naredbi niti šta sve program može, već jednostavno upoznavanje sa mogućnostima programa. Druga vrsta priručnika je podsetnik, u kojem su sve moguće opcije logično povezane. Detaljan sadržaj i indeksi su od velike važnosti za ovu vrstu priručnika.

Čarobni trougao

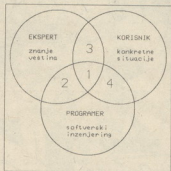
Računare često koristimo kao zamenu za nedostupnog stručnjaka. Program za tekući račun bi trebalo da radi poput knjigovode — vrhunskog poznavaoca bankarskog poslovanja. Svaki korisnički program u sebi mora da sadrži neko znanje, koje se ispoljava u vidu dijaloga programa sa krajnjim korisnikom. Još uvek ne postoje programi koji pišu druge programe bez pomoći čoveka, a krajnji korisnici računarskih programa i dalje su ljudi. Zato stvaranje dobrog programa u sebi mora objediniti znanja i veštine triju profila: eksperta, programera i korisnika. Odnos ovih „saucešnika“ u pisanju programa je shematski prikazan na slici 1. Brojevi od 1 do 4 označavaju različite interakcije između ova tri osnovna tipa.

Ekspert, na primer lekar ili advokat, poseduje duboko poznavanje principa svoje struke, što je većim delom irelevantno za strukturu programa. Programer ne mora da poznaje te principe, ali je u stanju da kodira veštine eksperta na način dostupan računaru, a krajnjem korisniku potreban je samo deo te veštine — upravo ono što program i treba da radi. Slično, programer poseduje opšta znanja iz softverskog inženjeringa (tj. zna da napravi program na bilo koju zadatu temu), a korisnik zna specifične uslove u kojima će se program izvršavati, ali su te „konkretne situacije“ po pravilu nepoznate i programeru i ekspertu.

Interakcija između pojedinih partnera je takode trojaka. Ekspert mora programeru da prenese svoja znanja, i to se obično naziva sistemskom analizom (na slici 1, označeno brojem 2). Taj korak uvek prethodi samom programiranju, i prilika je da se demonstrira komunikacioni jaz između domernih profesija. Cela kibernetika kao nauka je i nastala iz želje Norberta Vinera da se uspostavi efikasna razmena informacija između stručnjaka različitih profesija! Pro-

blemi komunikacije se obično rešavaju tako što programer nauči delić neke druge nauke, a ekspert sazna ponešto o programiranju... što sve čini programerski posao tako interesantnim. Moderniji način da se preskoči ovaj jaz je korišćenje tzv. ekspertnih sistema, no o njima nekom drugom prilikom.

Preklapanje između korisnika i programera spada u probleme „računarske konverzije“ (označene brojem 4). To je zapravo način na koji će se budući program obraćati korisniku, a već sada je jasno da se upravo na ovom polju bude odsudna bitka za definitivno prihvatanje računara u svakodnevnom životu. Ne moramo da znamo kako radi gramofon niti kako je snimljena neka ploča, i nadajmo se da jednog dana nećemo morati da znamo ništa o principima rada operativnih sistema, a da ipak budemo u stanju da koristimo računar poput gramofona, telefona, televizora i svih ostalih čuda popularne elektronike.



Slika 1. - Odnos između eksperta, programera i korisnika

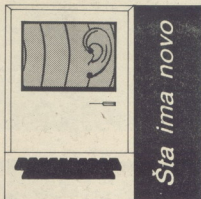
LEGENDA:

- 1 - program
- 2 - kodirano znanje
- 3 - zajedničko znanje
- 4 - računarska konverzija

Preklapanje između eksperta i korisnika (broj 3 na slici 1), obično se previda u knjigama o računarskom programiranju, jer se one uvek koncentrišu na radnje bliske računaru. Daleki, nedostižni cilj računarske industrije je da sa dva znanja postanu dostupna kroz konverziciju sa računaru.

Konačno, zajednički deo za sva tri profila je računarski program (broj 1 na slici 1). Uloga programera bi trebalo da bude pasivna: on mora da nauči kako bi želeli da saraduju ekspert i korisnik, i nije njegov zadatak da nešto menja u interakciji između njih. U pisanju korisničkih programa programer je najslabija karika u lancu!

Sve u svemu, poznavanje programiranja samo za sebe nije dovoljno da bi se došlo do dobrog korisničkog programa. U nedostatku ekspertnih znanja, hakeru preostaje da piše igre za uništavanje džojstika, ili avanture — ili da se bavi programiranjem radi sopstvenog intelektualnog napretka. Osećaj neumerenosti koji se javlja kod uznapredovanih početnika treba prevazići ličnim uzdizanjem i produbljivanjem znanja u izabranu oblast. Time dolazimo na već spomenuto, ali na sreću relativno čest i plodotvoran slučaj, kada su sva tri profila — ekspert, programer i korisnik — spojeni u jednu ličnost.



Bibliografija računarskih knjiga

Kupujete sve knjige iz računarstva, zar ne? Da biste bili sigurni da vam neka, bar od američkih izdavača, nije promakla, nabavite Annual Bibliography of Computer Oriented Books, koji izdaje Kolorado univerzitet (to je onaj što se, tu i tamo, vidi u Dinastiji, a je verovatno niste baš naj-sigurniji koja je to zgrada). Knjige su podeljene u 86 kategorija prema tipu (priručnik, udžbenik, monografija) i stilu izlaganja (programirani kurs, naracija, esej), a najviše ih je iz oblasti personalnih računara (kao, ustalom, i kod nas). Cena kataloga je 4 dolara, a adresa na koju možete da uplatite ovu svotu je Computing Newsletter, Box 7345, Colorado Springs, CO 80933, USA. Požurite, jer katalog se stalno osvežuje, tako da je knjige starije od 3 godine brišu sa spiska.

N. Aleksić

Za male Francuze

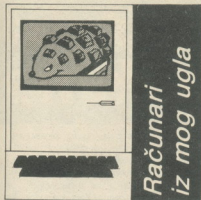
Francuska vlada je razdelila oko 120 000 računara osnovnim školama u Francuskoj. Paralelno sa tom velikom deobom, 110 000 učitelja je pohađalo bliz kurseve obučavanja za rad na računaru. Mi ćemo još pisati o ovom zanimljivom (to je najmanje što se može reći) pokušaju, a vi razmiselite: koliko sve ovo liči na situaciju kod nas?

Vremena su konfuzna

Sud je zabranio da se Somašild (Somaschild), specijalna folija koja se lepi preko ekrana monitora, reklamira kao sredstvo koje smanjuje štetno dejstvo zračenja ekrana zato što „ne postoji pouzdani dokaz da su ekrani zaista štetni“. Zaista?

Enciklopedija na disku

Američki firma Groljier (Grolier) je stavila svih devet miliona reči Američke akademske enciklopedije na jedan jedini kompaktni-disk prečnika pet palca. Disk se može koristiti kao ROM, ali samo za IBM PC i „atari 520 ST“. Na disku se, pored enciklopedije sa 300 000 tema (koja, inače, stajuje u dvadeset jednu knjigu) nalazi i potpuni indeks reči i softver koji omogućuje baratanje. Sasvim je za verovati da bi jedan ovakav disk bio dovoljan i za osam naših enciklopedija. Ko zna, možda nam je stvarno toliko i potrebno.



Računari
iz mog ugla

RAČUNARI KAO POLITIČKO PITANJE

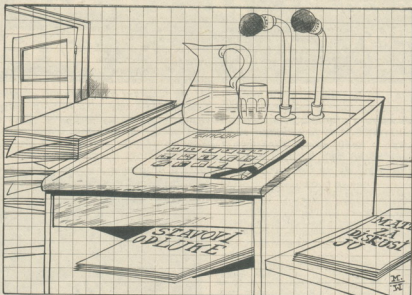
Ako imate bar malo igračkog staža u arkadnim igrama, malo ste da primetite da često viši nivoi nisu teži od početnih. Jedino što treba nekako dospeti do tih niva.

Na prvi pogled, izgleda da ni bavljenje računarskom politikom na visokom nivou nije osobito teško. Osim što nema pravo saveznika da nastupi na tom nivou.

Kompiuterski strateg pojavljuje se na TV u odelu tamnijih boja. Važno je da odabere dostojanstvenu pozu, da mu glas bude siguran (po mogućstvu dobok) i, uopšte, da ostavlja autoritativni utisak. I naravno, važno je da pravilno odabere dezen na svojoj kravati, koji će dovoljno dobro održavati njegovu ličnost. Ako se upravo vratio iz Amerike, može otkopčati dugmad na kariranom sakou koji je druge boje od besprekorno opeglanih pantalona. Povremeno će ležerno staviti ruke u džepove.

Dok to gledate vi lećite svu kompiuterski kompleks loveći autoritete u greškama. Kad ustanovite kako oni baluzne, vama će svakako biti mnogo lakše. Nemojte davati paušalne ocene da je jednostavno biti računarski političar i da je dovoljno biti ozbiljan klov. Nije lako biti u nekom inicijativnom odboru, za standardizaciju računara, ako prema malim računarima osećate averziju do te mere da ne možete ni da ih pogledate. Među njima ima vrlo dragocenih ljudi koji nisu čuli da kod nas postoje računarski časopisi. Samo oni mogu bez predrasuda i predubeđenja objektivno da kreiraju računarsku politiku.

O računarskoj strategiji ne treba suditi na osnovu površnog posmatranja računarskih stratega. To je odgovorna, složena i kompleksna problematika koja se ne može svesti na prvu aproksimaciju. Nikako nije cilj da računari budu što praktičniji, sa izobiljem softvera, i što jeftiniji. Obični ljudi



Ilustracija: Misa Marković

svojom prostonarodnom logikom nisu u stanju da dokuče svu suptilnost računarske politike.

To što su kod nas školske vlasti jedne republike dale preporuku da tasteri RETURN i RESET treba da budu jedan ispod drugog, a oni za pomeranje kurzora sa leve i desne strane tastature, nije glupost. Sve to ima dublji didaktički smisao. Recimo, ako ste gadali RETURN, pa malo promašili i resetovali računar, to će vas naučiti da budete pažljivi i da olako ne pritisakate dugmiće.

Pomeranje kurzora čak prevazilazi didaktičke okvire i ima širi društveni značaj. Ako su tasteri raspoređeni prema preporuci, nećete moći da pomerate kurzor samo jednom rukom. Ovakvim uravnoteženim simetrično-skladnim rasporedom primoravate učenike da obe ruke drže na računaru, a ne da tamo nešto mute ispod klupe. Trebalo bi tu koncepciju razviti i dalje, da za korišćenje računara bude potrebno tri, četiri, pa i više ruku. Tako će vesti učenika da radi na istoj mašini, što će razvijati njihov kolektivni duh, a onemogućavati uvek opasne individualiste koji bi nasamo mogli da zloupotrebe računar. U vezi s tim, predlažem tastaturu sa osam tastera od kojih bi jedan bio univerzalan (za sve znakove), i sedam (7) raznih SHIFT-ova. Trebalo bi razmotriti i predloge naših čitalaca, ukoliko ih ima.

Obilje jeftinog softvera nije prednost, nego velika opasnost. Ne samo da time prepuštamo omladinu pogubnim uticajima, nego se i moralno izopućuju presnimavajući programe. Pod hitno bi morale biti uvedene komisije koje bi pregledale i cenzurisale softver, i institucije koje bi potpuno preuzele na sebe obaveze i odgovornosti distribucije. Cene nikako ne smeju da budu niske. Kome treba, taj će i da plati.

Nije cilj da računari budu što jeftiniji. Učinjena je greška i dozvoljeno je da narod stihijski nabavlja računare i da za to troši dragocene devize koje bi tako dobrodošlo

domaćim proizvođačima. Ljubitelji inostrane robe morali bi imati na umu englesku poslovicu: „Nisam dovoljno bogat da bih kupovao jeftine stvari“. Mnogo je važnije da se naše opštinske ekonomije celovito i kompletno razvijaju.

U osnovi zdravu ideju „u svakoj opštini omladinska fabrika“ koja bi proizvodila računare potpuno nekompatibilne sa drugim, trebalo bi proširiti, i po mesnim zajednicama otvarati penzionerske fabrike. Tako bi se i penzioneri držali na oku, jer su nešto počeli u poslednje vreme mnogo da gundaju. Ujedno bi se zadovoljio princip da neko ne treba da radi ono što mu se ne sviđa.

Ovakvo krupne strateške greške ne bi bile počinjene da smo na vreme mislili i školovali odgovarajući kadar. U škole svakako treba uvesti usmerenje kompiuter-političar. Budući kreatori naše svetle računarske budućnosti treba da kanališu računarstvo da posluži očuvanju stanja kakvo jeste. To će biti vrlo teško, s obzirom na nizak nivo društvene svesti privatnih lica, koja odbijaju da kupe bar neki od domaćih računara. Mogućno je da ubedevanje neće uroditi plodom, jer postoji veliki broj zlonamernih, potpuno nekompletnih pojedinaca (ne razumeju se ni u računare ni u strategiju), koji vrše poredenja domaćih i stranih računara, kao da se to može porediti.

Ako drugačije ne ide, trebalo bi iz osnovne promeniti globalnu strategiju. Trebalo bi se opredeliti za jedinstveni računar za celu zemlju. Zgodno bi bilo da se usvoji neki model koji niko više nigde ne proizvodi i koji može biti kompatibilan samo sa običnom peglom. Kada bi to bilo usvojeno kao standard, naša bi industrija bila zaštićena od neloyalne konkurencije. Naravno da bi tada i računarske novine morale da prestanu da pišu o stranim računarima, da se javnost ne bi bez razloga uznemiravala i da bismo bezbrizno zakoračili u 21. vek.

Jelena Rupnik

Kako
to rade
drugi

piratska je tuga pregolema

Pirati su postali omiljena tema u svim računarski orijentisanim glasilima. Malo ko je propustio šansu da piše o toj temi u kojoj se ukrštaju interesi svih učesnika računarskog tržišta. Kako vreme prolazi, rasprave o piratima se polako zaoštravaju, a velike firme prelaze sa reči na dela. Odnosi su sad toliko konfrontirani da je direktan sukob neminovan, a kako taj sukob može da otpočne pokazuje upravo ovaj tekst. U našoj zemlji, praktično, i ne postoji tržište softvera, pa domaći pirati za razliku od američkih, još uvek mirno dišu. Filmski distributeri su uspeali da izadu veoma brzo na kraj sa piratima video kasete. Da li to znači da se i piratima kompjuterskih programa sprema slična sudbina?

Pirati su imali sreće dok su operisali u „dobrim, starim vremenima“ kada su ih jedni tapšali a drugi im pretili prstom, a oni samo uzimali — lovu. Sad je sve sasvim drugačije. Velike kompanije su shvatile neke stvari i, naravno, „poduzele neke korake“. Koje?

U najvećim računarskim časopisima širom SAD objavljen je oglas koji izgleda ovako: Na slici je mlad, lepo obučeni čovek (odelo i kravata). On je sasvim nalik uspešnom poslovnom čoveku — u ruci mu je telefon, a lice mu je nasmevano. Pored njega piše:

„Običan kriminalac

On ne izgleda kao običan kriminalac, ali on je upravo protivzakonito prekopirao jedan program. To nije običan zločin. To je zločin saveznog nivoa.

Kopiranje softvera bez dozvole je kršenje zakona o autorskim pravima i kažnjava se po zakonu, znači lišavanjem slobode i novčano.

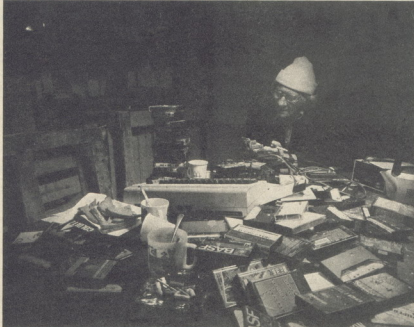
Ovom kriminalcu ni slučajno ne bi palo na pamet da ukrade novac iz nečijeg novčanika. On je čovek od principa. No, on ne shvata da je pravljenje protivzakonite kopije programa identično sa džeparenjem autora programa. On takođe ne shvata da svi plaćaju zbog njegovog čina, jer će oni koji program kupuju platiti višu cenu.

Ako poznajete osobe koje protivzakonito kopiraju programe recite im da krše zakon. Pomozite u zaustavljanju ovog kriminalca pre nego što bude kasno.

NEOVLAŠČENO KOPIRANJE SOFTVERA JE KRIMINALI!

Dileme nema, proizvođači softvera preko svog udruženja objavljuju krstaški pohod na pirate.

Izgleda da je uдела u ovom potezu imala i velika anketa na temu piratstva. Anketa je sprovedena na uzorku od 45000 ljudi i pokazala je da se na svaki legalno prodati program proda i jedan piratski. Suočeni sa takvim podatkom, proizvođači su morali da „uzvrate udarac“. Čak ni sa pretpostavkom da svi oni koji su program kupili od pirata ne bi kupili taj isti program legalno situacija ne izgleda mnogo bolja. Procene za 1986. godinu su da će na celokupnom tržištu softvera 25 do 30% prometa ostvariti pirati. Gubitak u zaradi za softverske firme je reda veličine 800 miliona dolara.



Uvežbavanje otpora

Softverski lobi u Kongresu SAD je „boksovao“ zakone koji bi trebalo da štite proizvođače softvera, a prva primena federalnog zakona upravo se isprobava na tajvanskom biznismenu Teh Yi Huangu koji se bavio piratstvom velikog obima, a uхваćen je na delu. Njega sad očekuje kazna zatvora do 14 godina trajanja (!!!!), plus troškovi suđenja koji idu i do 75000 dolara.

Osim iznuđivanja ovakvih zakonskih mera, udruženje organizacija koje se bave obradom podataka postavile su i određena pravila koja treba poštovati pri poslovanju:

— Svaki zaplašeni mora da potpiše izjavu da će zvati softver od pirata, kao i da je upoznat sa posledicama učestvovanja u aktu piratstva (posledice se kreću od otkaza od krivičnog gonjenja).

— Potrebno je razviti metode redovne kontrole cirkulacije, distribucije i kopiranja softvera, kao i sastavljanje spiska zaposlenih koji imaju pristup softveru.

— Treba formulirati pravne metode koje bi kompanije primenjivale da zaštite sebe u slučaju piratstva. One treba da su tako formulisane da krivica i sve posledice pad-

nu na onoga ko je i kopirao programe.

— Neophodno je postaviti kontrolne detektore na izlazu iz firmi — sličnih onima na aerodromima — da bi se smanjila mogućnost iznošenja raznih medijuma sa podacima.

— Pored ekipa za fizičko obezbeđenje zgrada od provala, požara i sličnog, poželjno je formirati posebne timove čiji je jedini zadatak da štite softver kako od pirata, tako od oštećenja, zlonamernog menjanja i sličnog.

— Predviđeno je osnivanje posebnih fondova koji treba da pokrivaju štetu koju naprave pirati (pre svega, za nabavku novih programa umesto ukradenih). Pošto se iz razloga bezbednosti ne prave kopije, jedan isti program se kupuje od originalnog proizvođača softvera više puta.

— Personal treba obučiti da se štiti od pirata (putem kurseva, seminara, plakata, i brošura, pri čemu je glavni slogan „Ne budite naivni!“).

— Treba organizovano prići propagandi i izazvati diferencijaciju među piratima. Treba ih izdellati na one koji to rade za novac, one koji rade iz radoznalosti, one koji to

rade zabave radi i tako dalje, sa posebnim akcentom na kriminalnom aktu koji podleže zakonskim sankcijama.

Nije ostalo sve samo na zakonskom gonjenju i ovim preporukama. Neke firme su probale i sa zaista originalnim rešenjima. Kompanija Mikropro Internešnel (Micropro International) je pokušala da daje besplatno originalne programe za programe koji su kupljeni od pirata. Zauzvrat je tražena informacija o tome kod koga je program kupljen. Akcija je trajala šest meseci, na njoj su radila četiri službenika sa punim radnim vremenom, ali veliki bosovi i oni koji su kralji programe na izvoru — nisu uhvaćeni. Na kraju je Mikropro Internešnel digao ruke od jalovog posla i prestao sa akcijom. „Osećao sam se kao direktor Dženeral Motorsa (General Motors) koji daje nove automobile za ukradene“, rezignirano je izjavio posle akcije direktor Mikroproa. Ipak, bio je to dobar pokušaj.

Crvi u jabuci

U toku ogorčene borbe na videlo su isplivale i neke činjenice koje sugerišu da inicijalni krivci sede možda baš unutar softverskih i hardverskih firmi. Predstavljajući se kao spoljni saradnici firme Epl (Apple), dva privatna detektiva koje je unajmio jedan računarski časopis (ne, nisu Računari) su od jednog službenika firme za samo 1100 dolara kupili desetak vrlo skupih programa za računar „mekintoš“ među programima je bio i čuveni „Jazz“). Jedan od kupljenih programa je bio i program koji je u Eplu bio samo na doradi i proveri, tako da još nije bio pušten u prodaju (redovna je praksa da softverske firme šalju proizvođačima hardvera programe na probu bez izbacivanja na tržište).

Pritešnjen dokazima, službenik koji je program prodao je izjavio da u Eplu vlada hakerska atmosfera još od osnivanja firme, da je kopiranje programa redovna praksa, da to „rade svi“ i da je on do programa došao jednostavno tako što su mu drugi pustili kopije na sto. U firmi je već jednom bio uhvaćen i upozoren, pa je zato preneo kopije programa kući i nastavio celu rabotu tamo, ali „samo za prijatelje Epla“. Naravno, uz „skromnu“ nadoknadu.

Nezvanično, jedan od direktora proizvodnje u Eplu kaže da je „Epl kao prodavnica bombona. Teško je odoleti iskušenju“, te da se dobar program prenosi kroz firmu „kao požar“ i „dok se okrenete“, svi ga imaju. Po njemu, ipak treba praviti razliku između onih koji to rade da bi ostvarili nezakonitu dobit i onih koji to rade radi analize programa ili stručnog usavršavanja.

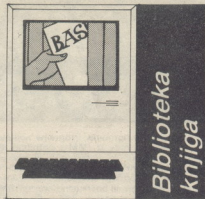
Zvanično, u Eplu su „iznenađeni i uvređeni“. Tvrdi da će se primeniti drakonske mere nad onima koji budu uhvaćeni u piratstvu, ali i da se „to dešavalo i dešavaće se“.

Izgleda da intenzivirana borba protiv pirata, sa jedne strane, i otkrivanje legala pirata u najvećim računarskim firmama tek ukvirkuju ceo piratski problem i da rešenje nije ni malo blizu. Ono što je sigurno je da protiv pirata započeta bespoštedna borba, sa ciljem da se oni u potpunosti istrebe.

Odgovor pirata još nije poznat. Da li će osnovati udruženje za zaštitu ugroženih pirata ili će, jednostavno, samo — povećati svoje cenovnike?

Ivan Nador

25/piratska je tuga
pregolema



Početnički za početnike

T. F. Fraj: Računari za početnike

Izdavač: Nolit, tiraž: 7.000 primeraka,
cena: 1.200 d.

U Nolitovoj biblioteci „Zanimljiva nauka“ pojavio se prvi naslov posevećen računarima — prevod knjige T. F. Fraja „Računari za početnike“. U knjizi se kroz poglavlja Osnovni pojmovi o računarima, Brojčani sistemi i računar, Računarska logika, Računarski ulaz i izlaz, Centralni procesor, Uskladištavanje i iskladištavanje informacija, Kako funkcionišu računarski programi, Veliki i mali računarski sistemi i Računar i društvo, pokušava da približi širokom krugu čitalaca način funkcionisanja računarskih sistema.

Kao što iz sadržaja možete videti, obrađene su teme iz nastavnih programa svih naših srednjoškolskih usmerenja posevećen računarstvu, kao i početnih kurseva programiranja na višim školama i fakultetima. Dakle, u ovoj knjizi govori se o stvarima o kojima već desetak godina postoji više knjiga na srpskohrvatskom jeziku naših autora. Stoga bi logično bilo očekivati da knjiga, čija je svrha popularizacija nauke bude čitljivija, bolja i jeftinija od drugih koje tretiraju iste teme.

Mada autor ovog prikaza, koji već dosta godina predaje upravo te osnovne pojmove računarstva početnicima, nema puno suštinskih primedbi originalu, engleskoj verziji ove knjige, i mada smatra da ni jedna knjiga ne može biti štetna niti suvišna, ne može, a da ne skrene pažnju potencijalnim čitaocima ove knjige na sledeće činjenice.

Po svemu sudeći, u Nolit u nisu smatrali da je potrebno da pri donošenju odluke koju knjigu treba izabrati za početnike u računarstvu konsultuju nekog ko se računarima i bavi. Ova greška ponovljena je i time što prevod izabrane knjige najverovatnije nije dat na stručnu recenziju. Kako inače objasniti da se u knjizi, koja bi trebalo

da bude uzor kako uvoditi nove pojmove, demonstrira elementarno nepoznavanje stručne terminologije. Tako se u mestu odo-maćenog „spoljašnja memorija“ koristi termin „skladište“, prihvatna memorija (bafer) ovde je „tampon-spremište“, a uvode se i „potporna skladišta“. Oni koji su odlučili da svoje vidike prošire pomoću ove knjige mogu naučiti da se „bajst sastoji najčešće od šest bitova“ (ne pominje se ni jedan drugi eksterni kod sem šestobitnog), kao i da „U decimalnom računu kompletan jednog broja je onaj broj koji mu se mora dodati da bi se kao rezultat dobila nula“. Nije razjašnjeno šta je potpuni, a šta nepotpuni komplet, a koliko navedena definicija ima smisla, prosudite sami.

Ovo je, na žalost, tek deo primedbi koje se mogu staviti na Nolitove „Računare za početnike“. Istina, i u knjigama koje smo ranije prikazivali moglo se naći nedostataka, kao što i ova ima svojih dobrih strana. Pre svega, u njoj se može naći analiza koja sa nacima rešavanja problema iz svakodnevnog života koje na slikovit način opisuju kako računari funkcionišu. Uz mnoštvo drugih informacija dato je dosta primera primene računara. Sve u svemu, ova knjiga može biti korisna kao dopunska literatura kako onima koji tek upoznaju računare, tako i nastavnicima programiranja, ali nebi-smo je mogli preporučiti kao jedini izvor saznanja o ovoj oblasti.

Razlog što smo sada znatno strožiji je što se prevod ove knjige, koja metodološki nije bolja od odgovarajućih knjiga naših autora, našao u renomiranju biblioteci renomiranog izdavača, dok su prethodne knjige bile prve koje na našem jeziku pokrivaaju određenu oblast, prevenci svojih mladih autora, štampane u malim tiražima, te se njihovim izdavačima moglo i moralo opustiti za sitnije nedostatke. Ne možemo se otići utisku da je zarada koja se može ostvariti od prodaje 7.000 primeraka knjige „Računari za početnike“ za izdavača bila važniji motiv od popularizacije nauke. Žao nam je što se u izlozima naših knjizara uz ovo izdanje ne može videti i udbenik sa sličnim, ali korektno obrađenim sadržajem — „Osnovi računarskih sistema“, T. Katanić i M. Markovića koji je Zavod za obrazovanje administrativnih kadrova SR Srbije štampao u tiražu od svega 200 primeraka. Tada bismo, bez obzira na cenu, našim računarskim početnicima imali šta da preporučimo.

No nije to jedina knjiga koja svrha na pristupačan način može uvesti u svet računara. Ako ste odlučili da saznate nešto o računarima i mogućnostima njihove primene, prelistajte pažljivo sve knjige koje se mogu naći u našim knjizarama tehničke literature, pa tek onda donesite odluku da li ćete kupiti baš onu koja je najnovija i koja se najviše reklamira.

N. Spalević

"amstradov" kernal

Postoje tri različite vrste programskih celina koje se mogu naći u „amstradovli“ memoriji. To su programi prednjeg plana (foreground), programi zadnjeg plana (background) i RSX (resident system extension).

Princip upotrebe bočnih ROM-ova kod „amstrada“ gotovo je isti kao i kod BBC-jevih računara. Jedina razlika je u tome što „amstrad“ pejdžuje ROM-ove preko adresnog prostora koji zauzima i RAM, i to od #C000 do #FFFF (tu se obično nalazi vidna memorija), dok je kod BBC-ja određeno 16K koji se koriste samo za ROM memorije. U praksi to izgleda ovako: jedan od ulazno-izlaznih portova procesora (#DFxx — bitan je samo bajt veće vrednosti) se koristi za određivanje da li će u poslednjoj četvrtini adresnog prostora biti „uključen“ neki od ROM-ova ili RAM. Tako svaki od ROM-ova koji su priključeni ima svoju adresu od jednog bajta. Pri tome su četiri vrednosti rezervisane: 252 — „uključiti“ donji i gornji ROM, 253 — „isključiti“ donji i „uključiti“ gornji ROM, 254 — „isključiti“ gornji i „uključiti“ donji ROM i 255 — „isključiti“ donji i gornji ROM, i one se koriste za stanje ROM-ova, dok se ostale vrednosti koriste za biranje gornjih ROM-ova (tom prilikom stanje se ne menja). Jedini 16K ROM memorije je rezervisano samo za operativni sistem. Napomenimo da stanje ima uticaj jedino na „čitanje“ sa određene memorijske lokacije koju zahvata preklapanje RAM-a i ROM-a, dok se upis uvek vrši u RAM. Ovo važi samo za procesor — video kontroler 6845 uvek ima pristup do RAM-a gde se smešta slika.

Na spoljni port opšte namene mogu se priključiti do 252 bočna ROM-a, od kojih je 7 sa programima zadnjeg plana i to na adresama #01—#07. Posle inicijalizacije, operativni sistem ispituje da li se na adresi =00 nalazi neki ROM i, ako se nalazi, kontrolu predaje prvom ulazu koji pronađe u tabeli naredbi ROM-a 0 (obično se i nalazi samo jedan ulaz), u suprotnom se proizvodi osnovni ROM (bežik). Ostali programi prednjeg plana se mogu iz bežika jednostavno pozvati jednom naredbom kao WORD (pokušajte jednom BASIC).

U prvaj liniji

Računar se nalazi pod kontrolom programa prednjeg plana, tj. njima je prepuštena komunikacija sa korisnicima i odluke o tome kakav odgovor dati na određeni zahtev. Takav je, recimo, „amsword“, na kome sada nastaje ovaj tekst, ili tolkohlavjeni železničarski (Locomotive) bežik koji je ugrađen u računar, kao i mnogobrojne igre. Ovi programi se mogu podeliti na dve podvrste: jedni su u ROM-u, a drugi u RAM-u. Iako sam operativni sistem među njima ne pravi razliku, sa stanovišta kori-

snik razlika je ogromna. Programe koji su predviđeni da rade iz RAM-a potrebno je stalno učitati sa kasetofona ili diska; pored toga, oni obično oduzimaju priličnu količinu slobodnog prostora. ROM foreground program, ili bočni (sideways) ROM kako se najčešće naziva, uvek je prisutan i zauzima mali deo RAM-a koji se koristi samo za promeniive i radni prostor. Karakteristično je da se u ROM upisuju programi koji se jako često koriste. To su, uglavnom, sistemski (programski jezici, assembler) i uslužni programi (tekst—procesori, programi za razočarano mnogobrojne ljubitelje igrara — svoju omiljenu igru neće moći da pronađu u ROM-u i jedino im preostaje da angažuju lokalnog hakera da je priredi u novom aranžmanu.

Programi prednjeg plana koji se nalaze u ROM-u mogu biti dugački do 64K. Pošto već naziremo pravu bujicu pitanja, hitamo da namo odgovor. Moguće je (softverski) povezati četiri ROM-a tako da, u stvari, čini jedan program. Prelazak iz jednog u drugi vrši se na vrlo jednostavan način. Jedan od restarta je žrtvovan da bi se omogućio poziv u drugi ROM sa samo tri bajta, dakle istom dužinom kao i normalna CALL instrukcija. To je RST #10 iza koga sledi 16-bitna reč, pri čemu gornja dva bita određuju u kom od ROM-ova se nalazi potrebna rutina, a donjih 14 bita određuje adresu relativno od #C000. Na taj način moguće je smestiti kompilovane i visoko profesionalne programe u ROM i opet imati na raspolaganju onih 40 i nešto kilobajta slobodnog prostora.

Programi iz pozadine

Programi zadnjeg plana su (verovatno već naslućujete) kao stvoreni za podršku periferijskim jedinicama. To su programi koji većinu vremena provode odmarajući se, sve dok ih ne prozove program prednjeg plana da reši neki problem vezan, recimo, za štampač ili disk. Oni se obavezno nalaze u ROM-u i sastavljeni su od više potprograma koji uglavnom deluju nezavisno jedan od drugog. Pre nego što počne da se koristi, pozadinski ROM se mora inicijalizovati. Tu brigu na sebe preuzima takođe program prednjeg plana i ta inicijalizacija se obično vrši na samom početku rada. Prva od rutina u pozadinskom ROM-u se koristi u prethodno navedenu svrhu. Njoj se prosleđuju adrese prvog i poslednjeg slobodnog bajta u memoriji da bi se izjasnilo o potrebnom radnom prostoru za valjanu podršku periferijskom uređaju koji će obavljati taj background program. Ovo se obavlja posredstvom operativnog sistema koji obaveštava foreground program o tome koliko memorije je pozadinski program rezervisao za sebe. Za svaki program zadnjeg plana koji nam je u radu potreban biće ponovljena ista procedura, a za ovo su predviđene dve rutine operativnog sistema

— prva inicijalizuje sve pozadinske ROM-ove, a druga, koja to čini samo za određeni ROM, postoji samo za slučaj kada nam nisu potrebni svi programi ove vrste.

Izgleda da je nastala neopisiva gužva u memoriji našeg jadnog „amstrada“, sa velikom verovatnoćom da nijedan program ne izvuče živu glavu. Ovo i nije tako daleko od istine. Prilično je teško snaći se sa svom tom količinom programa od kojih svaki zahteva malo radnog prostora. Zbog toga je morala da se primeni dinamička raspodela memorije. Pri tome se i pisci softvera za programe zadnjeg plana moraju pridržavati takve raspodele. To, konkretno, znači da se nijedan, doslovno nijedan podatak u radnoj memoriji (RAM-u) ne sme adresirati apsolutno. Operativni sistem to podržava tako što se prilikom svakog poziva u neki od pozadinskih programa u indeksnom registru IY nalazi adresa početka radnog prostora tog programa. To je jedini način da se izbegne zbrka i nepredviđena interakcija između raznih programa zadnjeg plana. Programi prednjeg plana nisu opterećeni ovim zahtevima, ali je zato prelazak iz jednog u drugi praćen reinicijalizacijom računara.

Zagonezni RSX

O RSX-u je dosta pisano u svim našim računarskim revijama — verovatno zbog nedostavne upotrebe, pre svega za dodavanje novih bežik naredbi. Programi RSX-a su, inače veoma slični programima zadnjeg plana. Razlika je samo u tome što se nalaze u RAM-u, pa ih je potrebno posle svakog resetovanja računara učitati u memoriju. Pri tom, dužnost foreground programa je da ih predstavi operativnom sistemu, koji ih pridružuje listi spoljašnjih naredbi na kojoj su i naredbe pozadinskog programa. Pri pisanju RSX-a naredbe moraju biti relokativne zbog dinamičke raspodele memorije.

Kad god program prednjeg plana ustanovi da je pronašao naredbu koja bi mogla da bude u RSX-u, background ili foreground programu, trebalo bi da pozove rutinu operativnog sistema (KL FIND COMMAND), dodajući joj adresu imena naredbe. Operativni sistem se služi listom adresa tabela svih opslužbica spoljašnjih naredbi u koje spadaju samo inicijalizovani RSX i pozadinski programi. Pretraživanje se vrši obrnutim redosledom od redosleda inicijalizacije (poslednja inicijalizovana programska celina se „pregleda“ prva). Ukoliko naredba nije pronađena, pokušava se sa ROM-ovima programa prednjeg plana počevši od adrese #00. Pri tom je moguća jedna od sledećih akcija:

- 1) ako je naredba u nekom RSX-u, vraća se „bliška“ adresa rutine,
- 2) ako je naredba u nekom od pozadinskih ROM-ova, vraća se „daleka“ adresa koja se sastoji od adrese ROM-a i adrese rutine,
- 3) ako je naredba u prednjem ROM-u,

zaključeno da je u pitanju prekid koji je generisala periferna jedinica, poziva se potprogram na adresi #3B. Na to mesto se prilikom ubacivanja rutine postavlja 5 bajtova, toga će, verovatno, uključivati i neku JP instrukciju, pri čemu se prethodni sadržaj tog dela kopira na neko drugo mesto. Ova procedura je potrebna iz prostog razloga što je moguće da je prethodno bila instalirana neka rutina koja je intervenisala pri prekida ma sa drugog periferijskog uređaja. Ako rutina za servisiranje zaključi da nije kompetentna za obradu prekida koji je nastao, trebalo bi da skoči na kopiju tih 5 bajtova. Jasno, zbog mogućeg prebacivanja, taj deo dužine 5 mora biti nezavisan od adrese na kojoj se nalazi, tj. lokobaliban. Poslednja kopija, kojih može biti neograničeno mnogo, jeste ona koju sistem postavlja pri naredbama RET. Rutina za servisiranje ima zadatak da bar resetuje izvor interapta, da ne bi došlo do beskonačne petlje.

... i događaji

Sve što je do sada izneto u ovom tekstu već je manje-više viđeno. Ovo što sada dolazi predstavlja zaista novost u projektovanju operativnih sistema za mikračunare.

Događaji (events) su uvedeni prvenstveno zbog potrebe da se osetljivo procesiranje spoljašnjih prekida obavi van rutine za obradu interapta. Zbog specifičnog načina obrade interapta, rutine za procesovanje spoljašnjih prekida su opterećene nizom zahteva: moraju biti kratke, ne smeju da omoguće prekid i moraju se nalaziti u RAM-u, koji je već opterećen radnim prostorom OS-a i programa prednjeg i zadnjeg plana. Programu za obradu prekida je omogućeno da hardversku pojavu — interapt — dalje transformiše u softversku pojavu — događaj.

Tehnika je sledeća: kada rutina za obradu spoljašnjih prekida ustanovi da može da obradi nastali prekid, ne obrađuje ga, već samo resetuje izvor prekida i zahteva od operativnog sistema da priključi na jedan od redova čekanja (pending queues) događaj. Od prirode prekida zavisi kada će se on obraditi i o tome odlučuje rutina koja je poslala zahtev za inicijalizovanje događaja.

Pronaći ćete na mnogim mestima (pa čak i u „oficijelnom“ Firmware manual-u) da je događaj softverski ekvivalent interapta. To je, donekle, tačno, jer je svaki interapt moguće prevesti u događaj, ali nisu svi događaji dobijeni na taj način. To znači da postoje događaji koji su nezavisni od interapta, ako ga posmatramo kao postavljanje INT nožice mikroprocesora u aktivno stanje. U principu, moguće je definisati bilo kakve uslove pod kojima će se javljati zahtevi za obradu događaja.

Događaj se sistemu predstavlja kao blok podataka. Blok se sastoji od 7 bajtova:

0:1: Chain — ovaj podatak postavlja i koristi samo operativni sistem da bi mogao da ulanča listu blokova

2: Count — ovaj bajt se koristi kao brojač razlike između broja zahteva za obradom događaja i broja obrađenih zahteva (posebno značajno za sinhronne događaje)

3: Class — klasa događaja

Bit 0: 1 — „bliska“ adresa rutine
0 — „daleka“ adresa rutine
Bit 1—4: prioritet sinhronih događaja
Bit 5: 0
Bit 6: 1 — ekspresni događaj
0 — normalni događaj
Bit 7: 1 — asinhroni (neusklađeni) događaj
0 — sinhroni (usklađeni) događaj

4,5: Address — adresa rutine za obradu događaja

6: ROM — adresa ROM-a u kom se nalazi rutina za obradu događaja (ukoliko je u klasi naznačeno da se rutina nalazi na „bliskoj“ adresi — bit 0 setovan — ovaj podatak se ne uzima u obzir)

Brojač (Count) se prilikom inicijalizacije bloka postavlja na 0. Kasnije se prilikom svakog zahteva povećava i posle svake obrade događaja smanjuje za 1. Da nema ovog podatka. Kernal bi „ispustio“ zahteve ako ih se više javi pre obrade događaja. Brojač se može upotrebiti i za onemogućavanje događaja ako ga postavimo na negativnu vrednost, ili za odbacivanje neželjenih zahteva, postavljanjem na 1.

Klasu (Class) događaja postavlja rutina koja ga inicijalizuje. Ona određuje da li će događaj biti sinhroni ili asinhroni, ekspresni ili normalni, i, okvirno, položaj rutine za obradu događaja. „Bliska“ adresa označava da se rutina nalazi u donjem ROM-u ili centralnom delu RAM-a, a „daleka“ da se nalazi u nekom od bočnih ROM-ova. Pri tome ekspresni asinhroni događaji moraju biti na „bliskoj“ adresi. Prioritet sinhronih događaja može biti od 0 do 15 (što je veća vrednost, veći je i prioritet).

Iza svakog bloka događaja mogu se nalaziti korisnička polja (user fields) koje će koristiti rutina za obradu ili (i) neka druga programska celina. Prilikom svakog poziva rutine za obradu događaja, kernal proleđuje u nekom od registara adresa preko koje se može izračunati adresa polja (BLK je adresa početka bloka):

— ako je rutina na „bliskoj“ adresi, registar DE će sadržati BLK+5 (u knjizi „Firmware manual“ ovdje je data netačna informacija da je u HL BLK+6)

— ako je rutina na „dalekjoj“ adresi, registar HL će sadržati BLK+5

Tako je pružena mogućnost da se samo jedna rutina za obradu događaja koristi za obradu svih događaja, ukoliko se pridržava pravila relativnog adresovanja podataka (setite se dinamičke raspodele memorije).

Zahtev i obrada

Moguće je napraviti ledno zanimljivo, iako grubo poređenje između raznih vrsta događaja i ponašanja čoveka u nekim situacijama. Zamislite da ste uzeli da sruđujete svoj album sa značkicama. Iznenirali ste se što vam je album u lošem stanju, koncentracija je popustila i uboli ste se. Odmah ste trgli ruku uz nekoliko prigodnih opaski. Naravno, to bi bio jedan ekspresni asinhroni događaj, dođuše proprančeni nekim uzgrednim manifestacijama u čiju se prirodu sada nećemo upuštati, jer se nikad ne dešavaju u računaru (ukoliko neko, ipak, zna jedan takav primer, bićemo vrlo srećni da ga objavimo). Zamislite samo kako te razne periferije „bockaju“ računaru. Došla je zima, uskoro će početi i već tradicionalne restrikcije struje. Ustanovili ste da je ostalo još pola časa do neminovnog isključenja

računara i počnete da tražite kasetu na koju će biti snimljeno novo remek-delo programiranja. Snimili ste program i najzad krenuli na spavanje. To je, otprilike, jedan normalni asinhroni događaj, osim ako vas je nestanak struje zatekao nespremnog, ali tada će doći samo do već gore spomenutih uzgrednih manifestacija (rekli smo da to ipak nije predmet ovog izučavanja). A šta ćete uraditi ako se na vratima vaše sobe pojavi neko od brižnih ukučana i opomene vaše da bi možda bilo krajnje vreme da već jednom sreditte svoju sobu? Naravno, stavićete tu primedbu na listu čekanja, u stvari pošto je već na listi čekanja, samo ćete povećati brojca za jedan i nastaviti sa svojom igricom. Ukoliko imate posebno pedantne roditelje, verovatno će vam biti potreban brojač od 16, a ne od 8 bita. To je jedan sinhroni događaj i to vrlo niskog prioriteta.

Autur ovog teksta se čuva greške koju je lako načiniti ako se mehanizam za procesiranje događaja ne shvati dobro. Na većini mesta se pomnije reč zahtev, a ne obrada. Programer vodi računa o zahtevima za obradu događaja, a kernal o samoj obradi. Granica između dva pomenuta pojma je strogo postavljena i oni se razmatraju sasvim odvojeno.

Dakle, postoje dve osnovne klase događaja:

a) sinhroni događaji — upotrebljavaju se za prekide kod kojih je potrebna gotovo trenutna akcija. Mogu biti:

1) ekspresni — obrada događaja se vrši odmah — iz rutine za obradu prekida.

2) normalni — blok koji opisuje događaj smesta se u red čekanja i obavlja neposredno pre izlaska iz rutine za obradu prekida.

b) sinhroni događaji — koriste se za interakciju među raznim delovima glavnog programa. Takođe mogu biti ekspresni i normalni. O tome kada će biti procesovani, odlučuje glavni program.

Shodno prethodnoj podeli, postoje i dva reda čekanja, koja se sasvim odvojeno procesuju: asinhroni i sinhroni red čekanja. Blokovi koji opisuje događaje ulančani su u liste pomoću sistemskog podatka (Chain) koji se nalazi na početku svakog bloka.

Asinhroni događaji

Ekspresni asinhroni događaji se koriste samo kad je potrebna zaista hitna akcija. Rutina za njihovu obradu je opterećena istim zahtevima kao i rutine za obradu samih prekida.

Normalni asinhroni događaji su najlakši za upotrebu. Birzina kojom se pozivaju uglavnom zadovoljava sve zahteve koje postavljaju periferne jedinice. Ako se zahtev javi van interapta (od strane glavnog programa), događaj se procesira odmah. Rutine za obradu ovih događaja koriste interni stek dužine 128 bajta. Mogu trajati koliko je potrebno, a svi zahtevi nastali u međuvremenu biće uzeti u obzir i opušteni po završetku obrade zahteva koji je prispeo.

Asinhroni događaji mogu se onemogućiti slično maskiranjem interapta. To može da se učini postavljanjem brojača na negativnu vrednost ili pozivanjem za to predviđene rutine (KL DISARM EVENT) koja radi istu stvar. Kasnije se mogu omogućiti inicijalizovanjem bloka (KL INIT EVENT) ili postavljanjem brojača na nulu. Onemogućavanje događaja ne sprečava kernal da po svakom zahtevu za obradu pogleda da li je

dogadjaj omogućiti ili ne. Ako bi se zahtevi slali često, došlo bi do priličnog usporjenja čitavog sistema. Zato je bolje svaki onemogućeni asinhroni događaj skinuti sa njegove liste čekanja.

Sinhroni događaji

Sinhroni događaji se nalaze pod kompletnom kontrolom glavnog programa. Kada se javi zahtev za obradom ove vrste događaja, blok događaja se postavlja na jedan od redova čekanja, ali se ne obrađuje sve dok foreground program to ne naredi. Zato bi trebalo da program prednjeg plana, kada god „uodi“ da sinhroni red čekanja nije prazan, pozove kernalove rutine za obradu sinhronih događaja, mada, u krajnjoj liniji, nije obavezan da obavi tu proceduru. Takođe, postoje normalni i ekspresni sinhroni događaji; ekspresni događaji imaju viši prioritet od normalnih.

Sinhroni događaje korisnik može onemogućiti korišćenjem rutine KL DEL SYNCHRONOUS koja, istovremeno, skida i blok koji opisuje događaj sa liste. Postoji i rutina (KL DISARM EVENT) koja onemogućava sve normalne sinhronne događaje, dok se ekspresni normalno obrađuju.

Ima nekoliko primera upotrebe sinhronih događaja u neproširenoj mašini. Tu su, naravno, naredbe bejzika za „rad sa interaptom“ (razume se, ne radi se sa spojašnjim, već sa unutrašnjim vremenskim interaptom), koje su jedan od glavnih „amstradovih“ aduta. Atraktivne naredbe, nema šta, ali za njih je ponajmanje zaslužan bejzik. Čitav posao obavlja kernal, a radi se o normalnim sinhronim događajima. Naime, kada interpreter naiđe na naredbu EVERY ili AFTER, inicijalizuje jedan normalni sinhroni događaj, pri čemu će rutina koja opslužuje događaj biti sam interpreter. Pri svakoj obradi zahteva izvršice se potprogrami koji počinju od programske linije navedene iza svake od dve pomenute naredbe. Posle svake naredbe koja se interpretira, bejzik proverava da li je sinhroni red čekanja prazan, i ako nije, obrađuje se svi zahtevi koji su pristigli.

Drugi primer je događaj koji generiše operativni sistem prilikom čitanja tastature. Kad god uoči da je taster ESCAPE pritisnut, šalje zahtev da se obradi jedan ekspresni sinhroni događaj koji obrađuje zahtev korisnika da se prekine program. U većini slučajeva rutina za obradu ovog događaja samo smešta marker u karakter-bafer koji obavestava program koji se izvršava (recimo, bejzik Interpreter) da je korisnik uputio zahtev za obradu „brejka“.

Svaki od ova dva primera ima neke veze sa interaptom — prvi je direktna posledica normalnog vremenskog prekida, dok drugi nastaje sa tastature koja se skanira pri istom prekidu. Navlašće sada jedan primer gde se obrada događaja odvija sasvim nezavisno od bilo kakve hardverske pojave. Radi se o naredbi čiju upotrebu omogućava deo operativnog sistema koji se bavi kontrolom zvučnog čipa i generisanjem zvuka. Zvučni čip AY-3-8912 je, da dozvolimo sebi malo literarne slobode, tročlani orkestar. To znači da ima tri kanala i svaki od njih može da ispušta nekakav zvuk, pa ako ste dobar dirigent, to može da liči na muziku. Svakom od kanala je dodeljen po jedan red čekanja u koje se može smestiti najmanje po tri „zvuka“, tj. zvučne naredbe koje sasvim određuju kako će izgledati ono što

neki od kanala ispušta. U bejziku postoji naredba koja ima sledeći izgled:
ON SQ(kanal) GOSUB nnnn

Ona se interpretira na sledeći način: definiše se i inicijalizuje blok koji opisuje jedan događaj (tj. potprogram koji počinje od linije nnnn) i šalje obaveštenje kontroleru zvuka gde se nalazi blok i za koji kanal je događaj zadužen. Zahtev za obradu događaja će biti poslat kada se nađe prazno mesto u redu čekanja koji je pridružen tom kanalu. Najverovatnije je da bejzik interpreter definiše sinhroni događaj (pogodite zašto!), dok bi mašinski program radije definisao neki normalan asinhroni događaj. Ova naredba uveliko pojednostavljuje pravljenje muzike. Kada god postoji mogućnost da se nekom od redova doda „zvuk“ biće pozvan potprogram, čiji je zadatak da pogleda koje je nota na redu i da je doda redu. Tako se sve odvija „automatski“ i transparentno sa stavovišta glavnog programa koji za to vreme može nesmetano da obavlja svoje poslove.

4.5:Recharge Count — vrednost na koju se postavlja prethodni podatak posle svakog zahteva za obradom događaja
6+ : blok događaja

Zahtev za servisiranjem ovih događaja se ne javlja posle svakog vremenskog prekida, već se samo broj otkućaja smanjuje za jedan. Kada dostigne nulu, šalje se zahtev, a brojčak se postavlja na vrednost koju definiše Recharge Count.



Tri izvora

Pored već pomenutih mogućnosti, kernal ima definisana tri izvora generisanja događaja i prema tome tri grupe:

- 1) Fast Ticker Events — 300 puta u sekundi, tj. kada se javi brzi vremenski interapt
- 2) Frame Flyback Events — na kraju svakog generisanja slike (50 puta u sekundi)
- 3) Ticker Events — pri normalnom vremenskom prekidu (50 puta u sekundi) generiše se zahtevi za obradu ovih događaja — koriste se od strane bejzika za interpretaciju EVERY i AFTER naredbi

Blokovi ovih grupa su nešto prošireni, ali na kraju svakog od njih nalazi se blok događaja koji smo već opisali (dužine 7 bajta).

Prve dve grupe imaju opisne blokove dužine 9 bajta, pri čemu su prva dva bajta adresa (Frame Chain ili Fast Chain) koju koristi kernal za ulančavanje odojovnih lista svake od grupe.

Treća grupa događaja je izuzetno fleksibilna. Blok koji ih opisuje je dugačak 13 bajta i ima sledeći izgled:

- 0.1:Tick Chain — adresa koja služi za ulančavanje blokova
- 2.3:Tick Count — brojač otkućaja

Na kraju, evo i jednog primera za sve one kojima se ne sviđa „amstradov“ kursor. Defini program menija linkova za oranje (ND: TXT DRAW CURSOR) i brisanje (ND: TXT UNDRAW CURSOR) kursora. Jedan im sastavni deo postavlja skicovne neke slike. Novi kursor će biti ostavljen na ekranu. Kovanje je zadužen jedan od prvih sinhronih događaja koji koristi mogućnosti normalnih vremenskih prekida. Rutina može da kontrolishe samo jedan kursor. Za drugi (COPY) kursor potrebno je izmeniti odgovarajuće ulaze u glavnom džamp-bloku (TXT PLACE CURSOR i TXT REMOVE CURSOR), kao i glavne rutine. Rutina koja servira događaj može ostati ista, jer se pridržava relativnog adresiranja. Program je relativno dugačak, jer svega zbog loše projektovanje linka. Nova rutina koja zamenjuje osnovnu ne dobija nikakve informacije od strane sistema; mora sama da pronađe kredovata smestiti kursor, da li je potrebno skrolovati prozor, pa čak i da ustanovi da li je kursor omogućen ili ne, za šta ne postoji način da se utvrdi posredstvom džamp-bloka. Zbog ovoga, autor nije siguran da li će program raditi na CPC664 i CPC6128 modelima. Ipak, učinjeno je sve da ne dođe do kraha i verovatno je ipak ostvarena kompatibilnost.

Dejan Muhamedagić

podmlaćeni BBC

Naš test
Drugi 6502

Sa donje strane BBC-ja su smešteni portovi za priključivanje raznih dodataka — disk jedinice, štampača, modema kao i neizbežni port opšte namene. Sa krajnje desne strane računara je smešten priključak nazvan *Tube*, pri čemu se pri svakom pominjanju ovog porta kaže da je *Tube* registrovani trademark Acon-a. BBC, naime, hardverski i softverski podržava povezivanje sa spoljnim mikrokompjuterima koji će preuzeti sve njegove vitalne funkcije, svodeći bazični računar na inteligentni grafički terminal. Acon trenutno prodaje osmootne koprocesore 65C02 i Z80B, kao i šesnaestobitni koprocesor 32016; cene ova tri uređaja iznose respektivno 200, 400 i 1400 funti.

S kraja ka početku

Podimo, suprotno uobičajenom poretku stvari, od kraja i pogledajmo fascinantne karakteristike koprocesora 32016. National Semiconductor 32016 je trideset dvo-bitni mikrop procesor kod nas prilično zasenjen Motorolaom 68000; 32016 je, međutim, procesor u svakom pogledu superioran Motorolaom čipu i pripada klasi mikroprocesora 68020, od koga je noviji i prema tome bolji. Za sada se uglavnom primenjuje na velikim kompjuterima firme Digital Research i na nekim IBM-ovim modelima, dok mu je BBC prvi izlazak u svet mikroracunara. Acon ga je u startu opremio jednim megabajtom RAM-a (adresni prostor od 64 megabajta verovatno neće biti popunjen u skoroj budućnosti), operativnim sistemom PANOS (verzija unix-a; pravi Unix može da se primeni tek kupovinom hard diska) i kompjalerima za Fortran 77, pascal, C kao i BBC bezik interpretatorom. Mogu se navesti i kompjajleri i interpretatori drugih programskih jezika kao što su Lisp, Logo i prolog; Acon planira da 32016 (i njihov trenutno vodeći računar Cambridge Workstation) bude kompatibilan (i po brzini rada superioran) sa u svetu i kod nas sve popularnijim mini računarima VAX koji, premda obično rade pod operativnim sistemom VMS, mogu da rade i pod Unix-om. Tome pomaže i aritmetički koprocesor NS32081 koji obavlja množenje, deljenje i druge operacije sa brojevima u pokretnom zarezu mnogo puta brže nego što stari Z80 i 6502 sabiraju intidžer! Čini nam se da se nekeko ko planira da nabavi Cambridge Workstation trenutno više isplati da kupi BBC, hard disk i 32016; jeste da će dati stotinak funti više, ali će bar imati i BBC koji, bez svih ovih skupih dodataka, može da izvršava razne „smrtnre“ programe koji uključuju i dobre igre.

Koprocesor Z80B se isporučuje sa 64 K RAM-a (bedno zvuci prema onim megabajtima ali...). Z80B je, verovatno, pripremljen da bi učinio BBC kompatibilnim sa pozna-



tim operativnim sistemom CP/M, pa ćete tako, uz sam uređaj, dobiti i četiri diskete pune CP/M softvera: tekst procesor, baza podataka, unakrsna izračunavanja, makro bezik... Acon je verovatno u pravu kada kaže da sav ovaj softver vredi na slobodnom tržištu više od 400 funti koliko Z80B košta. Sa druge strane, verovatno nećete biti zadovoljni svim ovim programima, pa ćete postepeno kupovati skuplje (ako mislite da ćete ih lako i besplatno kupovati iz neke firme koja koristi 'Partnera', varate se — različiti formati diskova će vas dobro namučiti). Najneprijatnija osobina Z80 koprocesora je što ni jedan ROM kojim ste, uz dosta izdataka, opremili vaš BBC neće moći da koristi njegove potencijale — program pisan za 6502 *nikako* ne može da radi na Z80! Ukoliko ste se, dakle, navikli na tekst procesor View ili na bazu podataka View Store, moraćete da se od njih odviknete! Zbog svega toga Z80B predstavlja prilično promašen koprocesor koga slabo koji pojedinac nabavlja — odgovara jedino firmama koje žele da unificiraju operativne sisteme na raznim računarima. Vlasnici BBC-ja, pa tako i autor ovog teksta, se obično odlučuju za nabavku koprocesora 65C02 koji je, na neki način, najbolje primeren BBC-ju. Ni parametar da se radi o najjeftinijem nabavljenom koprocesoru („svega“ 200 funti) svakako nije bez značaja...

U visokom društvu

Procesor 65C02 je upakovan u kutiju veliku koliko i pola BBC-ja; postavlja se uz sam računar sa kojim je spojen neprijatno kratkim kablom. Sa zadnje strane uređaja je, uz neizbežni osigurač, smešten prekidač pomoću koga ga uključujete ili isključujete; kada isključite 65C02 i pritisnete BREAK, ridićete sa „sasvim običnim“ BBC-jem i tako rešiti sve probleme kompatibilnosti o kojima ćemo govoriti tek na kraju ovog prikaza.

Nije, međutim, dovoljno samo da priključite 65C02 i počnete sa radom: najpre je potrebno otvoriti BBC-jevu kutiju i zameniti DFS ROM (ROM u koji je upisan operativni sistem za rad sa diskovima) novim DFS, NET ROM-om. U ovaj je čip, osim naprednih rutina za komunikaciju sa diskom i računarskom mrežom („Econet“), smešten i softver za komunikaciju sa koprocesorom (osnovni operativni sistem poziva ove module preko takozvanih „proširenih vektora“; o njima nekom drugom prilikom). Ako ste se, međutim, navikli na korišćenje nekog drugog DFS-a i njegove karakteristike kao što su dvostruka gustina upisa ili više od 32 datoteke na disketi, povratka na Aconov DFS će vam biti prava beda. Rešenje, na sreću, postoji: ostavite stari DFS u računaru, a zatim postavite DFS, NET tako da bude u podnožju nižeg prioriteta (levo od DFS-a koji koristite). Nevojta kod ovoga rešenja je što će, ako isključite koprocesor, PAGE biti na & 1D00, što znači da ćete u osnovni BBC moći da učitate samo najkraće programe. Watford Electronics DFS 1.43 je uspešno rešio ovaj problem; ovaj ROM na neki vošaban način „ubija“ deo DFS, NET-a tako da možete da koristite kako osnovni BBC (PAGE na & 1900) tako i dodatni procesor.

Kada nazad završite sa povezivanjem, uključite aka računara i pritisnite CTRL BREAK; na ekranu će se, umesto standardnog 'BBC Computer 32 K' pojaviti napis 'Acon Tube 64K' — u kutiji sa dodatnim procesorom je, naime, ugrađeno 64 K dinamičkog RAM-a koji bi praktično u celini trebao da bude slobodan za programe i podatke. Pokušajmo da se uverimo u to otkučavši PRINT (HIMEN-PAGE) 1024; na ekranu će se pojaviti broj 30 što znači da je za bezik slobodno svega 30 kilobajta. U čemu je problem?

Na slici 1 je prikazan adresni prostor 65C02 koprocesora. Početak i kraj adrese

BBC B je poznat kao odličan i nadasve brz računar sa brojnim proširenjima i dobrom softverskom podrškom, ali i kao mašina sa malom memorijom — svega 32 K. Trideset dva kilobajta i ne bi bila toliko nedovoljna kada BBC ne bi imao odličnu grafiku — u najvišem grafičkom modu ekran zauzima čak 20 K RAM-a što znači da će za bezik programe preostati 8.5 kilobajta (svega 5.75 K ako koristite disk) — premda da bi se bilo šta ozbiljno preduzelo. BBC-jev RAM se, naravno, može proširivati bilo kupovinom raznih RAM tabli bilo zamenom za model B+ kod koga je video memorija odvojena od osnovnog adresnog prostora. Pa ipak, pravo proširenje zamišljeno u doba koncipiranja ovoga računara je dodatni procesor.

slika 1:

Tube FIFO	FFFF
OS dodatnog 65C02	
Slobodno	F800
BASIC ili neki drugi jezik	C000
BASIC stek i promenljive	8000 HIMEN LOMEN
BASIC program	0800
Radni prostor jezičkog ROM-a	PAGE 0400
Sistemske promenljive i stek	0000

mape su rezervisani za stek, sistemske potrebe i FIFO stek preko koga se na kloku od 2 MHz komunicira sa osnovnim procesorom. BASIC ROM zauzima adrese od &8000 — &C000 — jednostavno je prekopiran u adresni prostor koprocesora. Prostor između &C000 i &F800 je potpuno slobodan — u osnovnom modelu računara ovaj prostor zauzima operativni sistem koji dodatnom procesoru nije potreban pošto se u potpunosti oslanja na I/O procesor. Tih 14 K možete da koristite za podatke tako što ćete, na početku bezik programa, napisati LOMEN=&C000. Acorn se, međutim, setio i načina da se ovaj prostor bolje iskoristi uz dodatni procesor čete primetiti i ROM sa takozvanom HI-BASIC-om.

Ukoliko planirate da koristite HI-BASIC, moraćete još jednom da raskopitate BBC i ugradite ovaj čip u jedno od podnožja (ako nemate više slobodnih podnožja, krajnje je vreme da kupite neku ROM tablu) tako da se nalazi levo od osnovnog BASIC ROM-a. Kada dočinite uključite čitav sistem, otkućaćete *FX 142, n gde smo sa n označili broj podnožja u koje ste ugradili HI-BASIC. PRINT (HIMEN-PAGE) 1024 će vas uveriti

slika 2:

Tube FIFO	FFFF
OS dodatnog 65C02	
HI-BASIC ili drugi HI-ROM	F800
BASIC program i podaci	B800 HIMEN
Radni prostor jezičkog ROM-a	0800 PAGE
Sistemske promenljive i stek	0400 0000

da vam je za bajzik programe i podatke sada raspoloživo maksimalnih 44 kilobajta.

Slika 2 ilustruje rad HI-BASIC-a: radi se o bezik interpretatoru koji je asemlbiran tako da se upisuje u memoriju počevši od &B800 (a ne od &8000) što znači da ga sa *SAVE HIBASIC 8000 +4000 možete snimiti na disketu, odstraniti ROM sa njim iz računara i ubuduće ga startovati sa *HIBASIC. Acorn je najavio da će prodavati 'Hi' verzije ostalih uslužnih ROM-ova koje je izbacio na tržište — tek će tako 6502 koprocesor biti u potpunosti iskorišćen. Za sada je u prodaji HI-VIEW (tekst procesor koji oslobađa 44 kilobajta za tekst) i, u trenutku kada ovo budete čitali, HI-COMAL.

Kompatibilno ili ne

Mnoge nezavisne softverske firme nisu, na žalost, spremne da proizvode Hi verzije svojih programa. Da bi program radio u memoriji koprocesora, on mora da bude pisan tako da koristi *jedino* dokumentovane rutine operativnog sistema. Pogledajmo jednostavan primer: program koji se izvršava i je smešten u adresni prostor koprocesora i, dok je video memorija smeštena u adresni prostor I/O procesora. Ukoliko program koji se izvršava crta po ekranu pokučujući direktno u video memoriju, na ekranu se ništa neće videti!

Praćkično ni jedna komercijalna igra ne radi dok je dodatni procesor uključen, što je prilično razumljivo i nije zabrinjavajuće. Kažemo da je nekompatibilnost igra razumljiva pošto se kod njih, zbog zahteva za brzinom, obično direktno pristupa video memoriji. Kažemo da njihova nekompatibilnost nije zabrinjavajuća jer igre rade sasvim lepo kada je koprocesor isključen i verovatno bi, kada bi ih on izvršavao, bile prebrze i neupotrebljive. U poslednje vreme se pojavljuju igre pisane specijalno za BBC sa dodatnim 65C02; programe kao što je elite 2 bi svakako trebalo nabaviti!

Ako o (ne)kompatibilnosti igara ne vredi mnogo razmišljati, nekompatibilnost svakog poslovnog i korisničkog programa predstavlja pravu malu tragediju, pogotovo kada se radi o dobrom ROM-u koji bi itekako mogao da iskoristi veću memoriju i brzi rad dodatnog 65C02. Svi ROM-ovi koji sadrže samo servisne rutine (PRINTMASTER, DUMP/OUT, Graphic ROM, Disc Doctor...) su 100% kompatibilni sa dodatnim 65C02, jer ovaj nema nikakve veze sa njihovim izvršavanjem. Programski jezici (paskal, kornal, fort) i razni asemlberi (npr. ade) su takođe kompatibilni sa dodatnim procesorom, dok se najveći problemi javljaju sa pojedinim poslovnim programima: Wordwise je, na primer, nekompatibilan, a Wordwise Plus samo delimično kompatibilan (editovanje i dalje mora da se obavlja u modu 7, tj. bez nashi slova) sa koprocesorom, Inter Chart i Inter Sheet rade savršeno, Starbase ne radi zbog zaštite (verovali li

je nesreća što Beebugsoftovi zaista izvršavajući ROM-ovi Toolkit (skup komandi koje pomažu bezik programerima), Exm2n (monitor / disassembler) i Sleuth (ROM koji omogućava izvršavanje bezik programa naredbu po naredbu) nisu kompatibilni sa koprocesorom, što za čoveka koji se navikao na njihove usluge predstavlja veliku neprijatnost. Exm2n 2 se mora zameniti daleko slabijim Gremlin-om (100% kompatibilan), dok je Toolkit jednostavno nezamenljiv. Treba sačekati da iz Engleske pristigne već naručeni Toolkit Plus i videti da li će on raditi sa dodatnim 65C02.

Najinteresantniji su ROM-ovi sa programima koji se samostalno prilagođavaju osnovnom ili dodatnom procesoru. Ovi ROM-ovi će neke svoje opcije primenjivati samo u radu sa dodatnim 65C02, dok će vlasnici osnovnog BBC-ja moći da ih koriste u zadovoljavajućoj meri. Jedina dva programa ovoga tipa koja su nam došla do ruku su Technomatic-ov novacac (Computer Aided Design) i View 3.0.

I/O procesor

Verovatno ste se već zapitali „a šta radi stari BBC kada je dodatni procesor preuzeo sve poslove?“. Stari BBC je sveden na takozvani I/O procesor: brine se o tastaturi, ekranu, štampaču, disku, računarskoj mreži, prima i obrađuje interapeote od drugih uređaja i, uopšte, brine o periferiji. Dodatni procesor će tako postati koordinate tačka koje treba spojiti linijom, a zatim nastaviti da izvršava program dok će I/O procesor raditi na crtanju te linije. Ukoliko su, dakle, programi dobro pisani (malo računanja pa malo crtanja i tako dalje), bile ostvarene paralelizam u radu, što znači da će se programi u kombinaciji BBC/dodatni 6502 izvršavati čak dva puta brže od programa na osnovnom modelu računara. Razliku u brzini koja je posledica kako paralelizma u radu tako i brzog clock-a pokazuju čak i jednostavni Benchmark testovi koje smo objavili u „Računara 9“.

Memorija I/O procesora se koristi za pamćenje ekrana, definicija tastera i preddefinisanih karaktera (možete da predefinišete sve znakove, a da vas to ne košta ni jednog bajta memorije u koju se upisuje bezik program), kao radni prostor aktivnih i neaktivnih ROM-ova i za mnoge druge sistemske potrebe. Ukoliko dobro poznajete asemlber, možete čak i da pišete programe koji će paralelno izvršavati na dva procesora!

Sve u svemu, dodatni 65C02 predstavlja proširenje primereno BBC-ju koje obezbeđuje komforan i brz rad i kompatibilnost sa većinom već nabavljenih ROM-ova. Cena mu je, naravno, previska: za 200 funti bi se mogao kupiti QL. Amstrad ili neki mašina iz silne klase. Ipak, za ljude koji planiraju da još izvesno vreme ostanu na BBC-ju dodatni 65C02 predstavlja odlično rešenje.

Dejan Ristanović

Na drugi način

Poker bez karata

„Svako ko je ikada igrao karte zna da se događa da poneko, tu i tamo, pokušava (pa i uspeva) da vara. A kakva je tek prilika da se vara kad se poker igra bez karata, recimo telefonom. Da li biste se zaista usudili da igrate poker, a da karte deli vaš protivnik i o tome vas obaveštava telefonom? „Da li je moguće igrati poker bez karata, a da svi učesnici igre budu u ravnopravnom položaju, da niko od njih ne može da vara, a da ostali to uvek, kad neko pokuša, odmah primete?“ — upitaše se jednom Robert Floyd (Robert Floyd) sa Stanford univerziteta u Kaliforniji, inače dobitnik Turingove nagrade. Odgovor na ovo pitanje dali su generisanjem potrebnog algoritma Adi Samir (Adi Shamir), Ronald Rajvest (Ronald Rivest) i Leonard Adelman sa MITa u Bostonu.

Igra počinje poštomom deobom karate, što podrazumeva da je svaki igrač dobio karte na potpuno slučajan način — da niko sem njega ne zna karte koje je dobio i da ne postoji karta koju su u toku partije dobila dva igrača. Naravno, svaki igrač mora da zna koje je karte dobio, a o protivničkim ništa više sem očigledne informacije da protivnik nema ni jednu od karata koju ovaj ima. U toku igre može doći do situacije da neki od igrača uzme kartu sa vrha špila i algoritam mora da obezbedi da igrač može da uzme potrebnu kartu, a da pri tom ni o njoj ni o nekoj drugoj karti koje je kod njega protivnicima ne da nikakvu informaciju. (U nekim drugim igrama sa kartama pojavljuje se situacija da jedan igrač treba da budi drugom neku kartu tako da je niko od ostalih učesnika u igri ne vidi, pa algoritam mora i ovakvu mogućnost da realizuje.) Na kraju svake partije svi igrači moraju imati mogućnost da provere da li je igra bila poštena ili je neko pokušavao da vara. Recimo, ako je neko u toku partije izjavio da ima četiri kraja, svi igrači mogu da se na kraju partije uvere da ih je on stvarno i imao. (Ili u nekim drugim igrama da nije imao određenu kartu.) Na prvi pogled nije baš jednostavno realizovati traženi algoritam, ali ako svaki od igrača ima pri ruci i kompjuter i odgovarajuću količinu domišljatosti ili znanja, a najbolje i jedno i drugo, algoritam se može realizovati. Pre nego što pročitate članak do kraja, pokušajte da rešite problem — javite nam ako uspete na neki drugi način.

Pretpostavimo da poker igraju dvoje, Ana i Mirko, i da se pre početka igre dogovore o dve funkcije K (kodirajuća funkcija) i D (dekodirajuća funkcija) sa sledećim osobinama:

1. $Kc(X)$ kodirana varijanta signala X u kodu c; (signalom X igrači javlja partneru da je dobio kartu X)

2. $Dc(Kc(X))=X$ pri bilo kom X ili pri bilo kom kodu c;

3. $Kc(Kd(X))=Kd(Kc(X))$ za bilo koju kartu X i kodove c i d;

4. ako su X i Kc(X) zadati ni najbolji

razbijač šifri ne može otkriti kod c za bilo koje X i c.

5. za bilo koja dva signala X i Y nemoguće je izabrati kodove c i d tako da je $Kc(X)=Kd(Y)$.

Osobina 3. je malo neobična ali je ostvariva, osobina 4. označava kriptografsku sigurnost, a osobina 5. služi za onemogućavanje varanja kada se na kraju partije vrši provera da li su svi igrači igrajući korektno. O tome kako se generišu ovakve funkcije možete pročitati u sledećem broju „Računara“ u članku o kriptografskim protokolima, ako u međuvremenu ne smislite takve funkcije. Dogovorivši se o funkcijama K i D, Mirko i Ana tajno jedno od drugog biraju kodove M i A. Oni ove kodove čuvaju do kraja igre, kada ih pokazuju jedno drugom da bi se uverili u poštnu igru. Mirko generiše 52 signala (pravi poker se igra sa 52 karte), recimo brojeve od 1 do 52, i kodira ih kodom M i zatim ih u slučajnom poretku šalje Ani. Slučajni poredak ostvaruje tako što promeša karte, pa ih, po poretku špila, pošalje Ani, ili još bolje i brže, pomoću generatora slučajnih brojeva u svom računaru. Ana bira 5 signala (karti) i šalje ih Mirku, koji ih dekoduje. Pošto je kod M poznat samo Mirku, Ana ništa ne zna o Mirkovim kartama do kraja igre. Zatim Ana šalje još 5 signala Mirku ali ih prethodno koduje svojim kodom A. Svaki od signala je dvostruko kodovan K (K (n)), što je po osobini 3, jednako K(K (n)) za bilo koje n. Mirko dekoduje signale i dobija K (n) i šalje ih nazad Ani, koja dekoduje signal i dobija svoje karte, koje Mirko nije mogao da vidi pošto ne zna kod A.

Majkl Rain sa Harvarda u Bostonu predložio je zanimljivu fizičku realizaciju algoritma. Kodovanje predstavlja kao stavljanje katanca na kutiju sa kartom. Mirko prvo stavlja svaku kartu u posebnu kutiju i na svaku kutiju stavlja katanac za koje ima isti ključ M. Ana bira 5 kutija i šalje ih Mirku, koji ih otključava i vidi svoje karte. Zatim Ana stavlja na 5 slučajno izabranih kutija i svoj katanac, kutije šalje Mirku koji otključava svoj katanac, šalje kutije Ani i pošto ih ona otključava vidi svoje karte koje Mirko nije mogao da vidi. Analogija sa izmenjenim redom otključavanja katanaca je potpuna kao u algoritmu sa kodovanjem. Ako u toku igre nekom od igrača trebaju karte, algoritam je isti kao kod podele karata. Na kraju igre igrači otkrivaju svoje kodove i uveravaju se da partner nije varao. Ova procedura se, očigledno, lako realizuje i kad je neka druga igra kartama u pitanju, ili kada igraju više igrača.

Samir (koji se u međuvremenu vratio u Izrael), Rajvest i Adelman (koji se preselio na Južnokaliifornijski univerzitet u Los Angelesu) uradili su i mnogo više od generisanja algoritama za igranje karata pomoću računara (koji koduje i dekoduje signale) i telefona. Najlepši njihov rezultat u vezi sa „mental pokerom“ je dokaz (strogi, matematički, koji možete da dobijete ako pišete redakciji „Računara“) da je nemoguće generisati algoritam koji bi obezbedio nepovredivost protivničkog koda. Rešenje paradoksa je u sledećem: kada se generišu kvalitetni kodovi, oni se mogu razbiti samo tako velikim računarima kakvi danas ne postoje, pri čemu je im je za to potrebno vreme veće od nastanka vasiona, a igrači uglavnom nemaju toliko vremena, barem ne za — kartel!



Tos 1.0 — turbo tape

Gotovo da nema vlasnika „komodora 64“ koji u radu sa kasetofonom ne koristi TURBO TAPE program za ubrzano učitavanje sa trake. Po našem mišljenju, osim same činjenice da ubrzava rad, program ne poseduje nikakve druge kvalitete, pa smo, zadržavajući osnovni format zapisa, napravili poboljšanu verziju. Svi programi snimljeni starom verzijom mogu se učitati sa novom i obratno.

Naredbe SAVE, LOAD i VERIFY se pišu kao u običnom CBM64 bejziku (puna reč ili skraćeni oblik) uz prefiks j (desna uglasta zagrada). Zatim sledi naziv na uobičajen način i, zatim, opcionalno, dve adrese. Ovo poslednje je i najznačajnija razlika u odnosu na standardni Turbo Tape, jer omogućuje snimanje ne samo običnih bejzik programa već i bilo kog memorijskog segmenta. Na primer, za snimanje mašinskog programa može se upotrebiti oblik: $SAVE „naziv“$, početna adresa, krajnja adresa

Pri kasnijem učitavanju sa: $LOAD „naziv“$ program će biti upisan na izvornu adresu, ali se može upisati i relacijano naredbom: $LOAD „naziv“$, nova početna adresa.

Pri učitavanju, kada naiđe na neki program, TOS će ispisati ne samo njegov naziv već i bilo koju i krajnju adresu koju zauzima, tako da se lako dobiju informacije o tipu programa, dužini itd.

U ostalim ponašnjama nove naredbe su potpuno kompatibilne sa standardnim bejzik naredbama, pa se mogu koristiti i u okviru programa, na primer pri pisanju „loadera“ koje će učitati delove nekog programa i zatim ga startovati bez intervencije korisnika.

Ugrađene su još dve nove osobine: za vreme SAVE i LOAD na ekranu se vidi eho dešavanja na ulazno/izlaznim linijama (kao kod spectuma), pa se lako razlikuju programi snimljeni običnom naredbom SAVE (pruge na ekranu su široke) od onih snimljenih TOS-om. Uz to, proces snimanja i upisivanja se može u svakom trenutku prekinuti tasterom STOP bez potrebe da se računar resetuje STOP i RESTORE tasterima.

Kada unesete program u računar, stajtuje ga sa RUN i, zatim, ispisane sume kontrolišite prema priloženoj listi. Ako neka


```

1 REM----- TURBO TAPE - TOS -----
2 REM
3 REM          C64
4 REM
5 REM  LOAD="naziv",start add)
6 REM  JUREVIF="naziv",start add)
7 REM  JSAVE="naziv",start,end add)
8 REM
9 REM  AUTOR: ZDRAN ZIVOTIC (85)
10 REM-----
20 REM --- PROVERA DATA LINIJA ---
21 RESTORE: FOR I=100 TO 828 STEP 4:S=0
22 FOR J=1 TO 32 READ R,S:9=S:NEXT J
23 PRINT "S/I=";S;"*";I;
24 AS="D":INPUT "OK";AS:IF AS="D" GOTO 27
25 PRINT CHR$(147)"LIST" I;"-";I;"-";POKEB31,19
26 POKEB32,13;POKEB18,2;SYS$2115
27 NEXT I:PRINT "1. OK"
28 RESTORE,S=0:FOR I=1 TO 105:READ B
29 S=S+B:NEXT I
30 IF S=128025 THEN PRINT "2. OK":END
31 PRINT "2. GRESKA!" : END
40 REM ----- KREIRANJE PROGRAMA -----
41 RESTORE:FOR I=209B TO 3104
42 READ B,I:POKE I,B:NEXT I
43 POKE13,1;POKE4,8;POKE5,31;POKE16,12:LIST
100 DATA 0, 48, 8, 0, 0, 158, 50, 45
101 DATA 48, 48, 58,162, 58, 34,141,145
102 DATA 18, 32, 32, 32, 84, 79, 83, 32
103 DATA 32, 32, 90, 46, 90, 73, 86, 79
104 DATA 84, 73, 67, 32, 32, 32, 32, 32
105 DATA 32, 32, 56, 53,160,167, 17, 17
106 DATA 0, 0, 0, 0, 169,129,162, 8
107 DATA 133,251,134,252,169, 0,162,204
108 DATA 133,253,134,254,160, 0,177,251
109 DATA 145,253,200,208,249,230,254,230
110 DATA 252,165,254,201,208,208,239, 32
111 DATA 0,204,169, 90,160, 8, 32, 30
112 DATA 171, 95,147, 17, 17, 32, 32, 32
113 DATA 32, 32, 32, 32, 32, 32, 32, 32
114 DATA 32, 18, 32, 32, 84, 79, 83, 32
115 DATA 32, 90, 47, 90, 32,146, 13, 13
116 DATA 0,169, 11,141, 8, 3,169,204
117 DATA 191, 9, 3,166, 32,150, 0, 32
118 DATA 80,274, 78,177,167,801, 93,240
119 DATA 8,104,104,32,121, 0, 76,231
120 DATA 167, 32,115, 0, 801,147,144, 4
121 DATA 201,150,144, 3, 75,115,207,133
122 DATA 2,169, 0,133,163,133,185, 32
123 DATA 115, 0,240, 36,201, 58,240, 32

```

```

124 DATA 32, 87,226, 32,121, 0,240, 24
125 DATA 201, 58,240, 20,169, 0, 32,211
126 DATA 206,230,185,165, 2,201,148,208
127 DATA 36,169, 2, 32,211,148,176, 15
128 DATA 165, 5,201,148,208, 12,168, 4
129 DATA 181, 2,149,171,207,208,249, 76
130 DATA 176,205,165, 43,133,195,165, 34
131 DATA 133,196,24,84, 8,165,172,133
132 DATA 32, 54,198,165,171,207, 2,240, 4
133 DATA 2,201,147,201, 1,232,230,185
134 DATA 134,10,134,147, 32, 5,207, 32
135 DATA 12,807,201, 0,240,249,133,171
136 DATA 32, 54,198,165,171,207, 2,240, 4
137 DATA 208,249,165,171,207, 1,240, 4
138 DATA 201, 2,208,227, 32, 46,206,162
139 DATA 4,189, 59, 3,149,171,202,208
140 DATA 248,134,145,160, 99, 32, 47,241
141 DATA 160,165,177,174, 32,210,255,208
142 DATA 192, 21,208,246, 32,238,206,169
143 DATA 6,170,168,133,252,173, 1,220
144 DATA 201,255,208, 10,202,208,246,136
145 DATA 208,249,165,171,207, 2,240, 4
146 DATA 240, 6,136,177,187,217, 69, 3
147 DATA 208,194,152,208,245,132,144, 32
148 DATA 210,245,173, 62, 3, 56,27,32
149 DATA 3,133,174,173, 63, 3,24,21,61
150 DATA 165,165,177,168, 6,197,185,240
151 DATA 14,197,171,208, 10,173, 60, 3
152 DATA 133,195,173, 61, 1,333,195,165
153 DATA 195,133,172, 24,101,174,133,174
154 DATA 165,165,133,273, 101,105,132,175
155 DATA 32,236,206, 32, 5,207,165, 1
156 DATA 41,254,133, 1, 32, 12,207, 32
157 DATA 54,207,196,147,208, 2,145,195
158 DATA 2,134,144, 2,134,144, 2,215
159 DATA 133,215,230,196,208, 203,196
160 DATA 165,195,197,174,160,196,229,175
161 DATA 144,221, 32, 94,207, 32, 46,206
162 DATA 165,189, 89,215, 5,144,133,144
163 DATA 208, 45,186, 10,240, 3, 76,141
164 DATA 225,165,171, 24,101,185,201, 2
165 DATA 208, 11, 32, 51,165,165,195,133
166 DATA 45,165,196,133, 46,165,123,201
167 DATA 2,206, 5, 32, 89,166, 76,116
168 DATA 164, 32,142,166, 175,116,165
169 DATA 10,240, 3, 76,118,207, 76,121
170 DATA 207,162, 7,134,171, 32, 56,248
171 DATA 32,233,206, 32, 77,206,165
172 DATA 165, 0,165, 4,177,187,136,163
173 DATA 185, 24,105, 1,202, 32,146,206
174 DATA 182, 8,185,172, 0, 32,146,206
175 DATA 162, 6,200,192, 5,234,208,242
176 DATA 162, 0,162, 4,177,187,136,163
177 DATA 144, 3,169, 32,202, 32,146,206

```

```

178 DATA 162, 5,200,192,187,208,237,169
179 DATA 3,133,171, 32,114,206,152, 32
180 DATA 146,206,132,215,162, 7,234,174
181 DATA 17, 32,146,206,152, 32,208,172
182 DATA 208, 4,230,173,202,202,165,172
183 DATA 197,147,165,173,229,175,144,231
184 DATA 234,165,215, 32,146,206,162, 7
185 DATA 136,208,248, 32, 56,206, 96,185
186 DATA 1, 9, 4,133, 1,165,251,141
187 DATA 32,208,173,17,208, 9, 16,141
188 DATA 17,208,165, 1,160, 1, 1
189 DATA 133, 1,132,192, 88, 96,160,173
190 DATA 17,208,14,238,141, 37,208,172
191 DATA 32,208,133,251, 69, 6,141, 32
192 DATA 208,169,127,141, 0,260,165, 1
193 DATA 41, 31,160, 0,133, 1,132,192
194 DATA 162, 0, 96,160, 0,169, 2, 32
195 DATA 146,206,162, 7,136,192, 9,208
196 DATA 244,162, 5,198,171,208,238,152
197 DATA 32,146,206,162, 7,136,208,247
198 DATA 202,202, 96,133,189, 89,215,133
199 DATA 215,169, 8,133,163, 5,165,165
200 DATA 1, 41,247, 32,198,206,162, 12
201 DATA 234, 77, 44, 1,220, 48, 3, 76
202 DATA 116,207,173, 32,208, 73, 1,141
203 DATA 32,208,104, 9, 32,195,165,206
204 DATA 162, 14,198,163,205,165, 96,202
205 DATA 208,253,144, 5,162, 11,202,208
206 DATA 253,133, 1, 96, 76, 32,253,174
207 DATA 2, 138,173, 32,217,183,194,207
208 DATA 165,20,149,172,165, 21,149,173
209 DATA 56, 96, 32,143,246,169, 32, 32
210 DATA 10,255,166,172,165,173, 32,205
211 DATA 189,169, 45, 32,210,255,166,174
212 DATA 165,175, 32,205,189, 96, 32, 23
213 DATA 248, 32, 77,206, 96,160, 0,132
214 DATA 215,169, 7,140, 1,6,221,162, 1
215 DATA 32, 73,207, 38,189,165,189,201
216 DATA 32, 208,248, 136,208, 73,194,207
217 DATA 801, 2,240,249,196,189,208,232
218 DATA 32, 54,207,136,208,246, 96,169
219 DATA 8,133,163, 32, 73,207, 38,189
220 DATA 234,224,234,198,163,208,244,185
221 DATA 189, 88,173, 38,208, 73, 1,141
222 DATA 32,208,169, 16, 44, 1,220, 48
223 DATA 3, 76,112,207, 44, 13,220,240
224 DATA 243,173, 13,221,146, 7,221, 7
225 DATA 165,25,141, 15,221,104, 74, 4
226 DATA 95,162, 30, 44,162, 11,16,206
227 DATA 28, 44,162, 29, 32, 46,206, 76
228 DATA 55,164, 80, 58, 32, 50, 76, 55
229 DATA 164, 0, 0, 0, 0, 0, 0, 0
230 DATA 0, 0, 0, 0, 0, 0, 0, 0
231 DATA 0, 0, 0, 0, 0, 0, 0, 0

```

ne odgovara, odgovorite sa 'N' i na ekranu ce biti izlistane 4 linije u okviru kojih postoji greška. Kada sve sumnje budu u redu, program ce sam proveriti ukupan zbir kao dodatnu kontrolu. Ako nakon toga računar ispiše 2. OK, možete preći na konačno kreiranje programa. Potrebno je prvo da ovaj izborni program snimate. Zatim otkucajte sledeće:

```

NEW
POKE44,100:POKE25600,0
NEW

```

Sada učitate program i zadajte RUN 40. Nakon toga u memoriji će se naći konačni oblik programa koji se snima i upisuje na standard način.

Treba napomenuti da TOS, nakon upisi-

```

S 0 = 1928
S 1 = 2684
S 2 = 5391
S 3 = 1655
S 4 = 2984
S 5 = 3598
S 6 = 3885
S 7 = 4281
S 8 = 4341
S 9 = 4505
S 10 = 4756
S 11 = 3633
S 12 = 4438
S 13 = 4300
S 14 = 4483
S 15 = 4390
S 16 = 3803

```

```

S 17 = 3850
S 18 = 4007
S 19 = 4206
S 20 = 5246
S 21 = 3808
S 22 = 3544
S 23 = 3408
S 24 = 4809
S 25 = 3341
S 26 = 4544
S 27 = 4438
S 28 = 3884
S 29 = 4425
S 30 = 4246
S 31 = 3249
S 32 = 734

```

vania i RUN, zadajte adrese od 52224 tako da i dalje zauzima slobodan deo za mašinske programe od adrese 49152. Ako se računar resetuje u toku rada, a TOS i 1 i dalje u

memoriji, može se inicijalizovati naredbom SYS 52224.

Zoran Životić

Najpre uzмите jednu praznu kasetu i na samom početku snimate "turbotape" (ili neki kompatibilan program). Ostavite je sa strane, a zatim kucajte program 2 -- "MERAČ TRAKE". On će vam omogućiti da glavni program prilagodite svom kasetofonu. Startujete ga sa RUN i u kasetofon stavite kasetu C60 (od 60 minuta) i prometajte je na početak. Pritisnite "RETURN" i unesite "1" za dužinu trake. Pritisnite taster "FFWD" na kasetofonu i strpljivo sačekajte da se traka premota do kraja. Čim se to prometavanje završi, odmah pritisnite taster "RETURN". Nakon toga unesite broj koji pokazuje brojčak brojača i zapišite negde programsku liniju koja se pojavila. Sada ponovo startujte program ali u kasetofon stavite kasetu C12 (od 12 minuta), prometajte je na početak, pritisnite "RETURN" i unesite "2" za dužinu kasete. Nastavak postupka je isti kao u prethodnom slučaju.

Po završetku ovog posla, treba da imate dve programske linije na parčetu papira. Izbršite program sa "NEW", učitate i startuje "turbo-

Tragač programa

Svaki korisnik računara koji ne poseduje disk jedinicu za koliko je teško naći program na kaseti, pogotovo ako nemate njihov spisak. Ovaj program obezbeđuje pretraživanje trake sa brzom prometaivanjem napred i uz to, jednostavan vid u sadržaj svake trake. Pretraživanje, na žalost, nije automatski da bi program mogao da funkcioniše u njega se, najpre, moraju uneti određeni podaci, ali se oni daju samo jednom -- za svagda. Rad sa ovim programom zahteva sledeće pripreme:

- programe snimljene „turboepnom“ razdvojite po kasetama od programa snimljenih pri normalnoj brzini;
- sam „turbotape“ snimite na posebnu traku kako bi vam ubrsk bio pri ruci;
- numerisite kasete, počevši od broja jedan;
- napravite spisak svih programa, po kasetama, za svaku od njih, kao i brojeve na kojima se traka odmah na početku kaseta, a ako je program odmah na početku kasete, upišite broj 1. Ako brojčak stoji između dva broja, vi uzimate ubrsk najpri kako se ne bi dogodilo da prilikom prometaivanja budete početak programa.

Kada predete unositi podatke u program 1, računac će, od vas tražiti da za svaki program, nazalaz, napišete na kojem kasetu i na kojoj se traci nalazi, broj na kojem počinje i tip programa. Ime ne može biti duže od 24 slova i ne sme počinjati brojem. Tip programa je neka proizvoljna karakteristika za koju je najbolje koristiti skraćenice kao AK (akcije igre), AV (igre avanture), UL (uslužni programi) itd.

```

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

```

tejp" pa započeti ukucavanje glavnog programa (program I) Unesite liniju 5000 i 5010, unesite one dve sa papira. Kada se ceo program bude nalazio u memoriji, startuje ga sa „RUN“, ali samo ovog puta, jer ćete inače izgubiti sve podatke o programu. Počnete da unosite imena i u ostale pojednosti o programu. Ako pritom pogrešite, mesto imena programa ubacite slovo „M“ kako biste došli do glavnog menija i pritisnite „F3“, a kod drugog menija „F6“ za dobijanje liste svih programa. Kada naštete broj programa kod kojeg ste pogrešili, pritisnite taster „M“ za glavni meni. Kod glavnog menija pritisnite „F5“ i kod drugog takode. Unesite broj programa i sve njegove podatke. Ako vas računar pita za ime sledećeg programa, možete i njega izmeniti, ili se vratiti na glavni meni unošenjem slova „M“ umesto imena. Ako ste se odlučili za ovo poslednje, na meniju pritisnite „F5“ pa „F1“ i navesti sa unošenjem novih podataka.

Program vam omogućuje da po želji dobijete listu svih programa, listu po kasetama i tipu, da unosite nove podatke, brišete i menjate stare i, konačno, da pronalazi programe brzim prematavanjem unapred od početka trake. Po završetku unošenja podataka, vratite se na glavni meni i pritisnite taster „RUN/STOP“. Stavite ranije pripremljenu kasetu u kasetofon i posle „turbotajpa“ saučajite glavni program naredbom „S“. Kada budete hteli da nađete neki program, učitaite i startuje „turbo tejp“ i pomoću njega unesite program 1, koji će pokrenuti sa „GO TO 100“.

Kod glavnog menija pritisnite „F1“ i unesite ime ili broj programa koji vam je potreban. Računar će vas informisati o tome gde se ovaj nalazi i da treba da prematate kasetu na početak.

```

1570 IF (A=0) THEN GOTO 1580
1580 PRINT "*****"
1590 NEXT I
1600 GOTO 1610
1610 PRINT "*****"
1620 PRINT "*****"
1630 PRINT "*****"
1640 PRINT "*****"
1650 PRINT "*****"
1660 PRINT "*****"
1670 PRINT "*****"
1680 PRINT "*****"
1690 PRINT "*****"
1700 PRINT "*****"
1710 PRINT "*****"
1720 PRINT "*****"
1730 PRINT "*****"
1740 PRINT "*****"
1750 PRINT "*****"
1760 PRINT "*****"
1770 PRINT "*****"
1780 PRINT "*****"
1790 PRINT "*****"
1800 PRINT "*****"
1810 PRINT "*****"
1820 PRINT "*****"
1830 PRINT "*****"
1840 PRINT "*****"
1850 PRINT "*****"
1860 PRINT "*****"
1870 PRINT "*****"
1880 PRINT "*****"
1890 PRINT "*****"
1900 PRINT "*****"
1910 PRINT "*****"
1920 PRINT "*****"
1930 PRINT "*****"
1940 PRINT "*****"
1950 PRINT "*****"
1960 PRINT "*****"
1970 PRINT "*****"
1980 PRINT "*****"
1990 PRINT "*****"
2000 PRINT "*****"
2010 PRINT "*****"
2020 PRINT "*****"
2030 PRINT "*****"
2040 PRINT "*****"
2050 PRINT "*****"
2060 PRINT "*****"
2070 PRINT "*****"
2080 PRINT "*****"
2090 PRINT "*****"
2100 PRINT "*****"
2110 PRINT "*****"
2120 PRINT "*****"
2130 PRINT "*****"
2140 PRINT "*****"
2150 PRINT "*****"
2160 PRINT "*****"
2170 PRINT "*****"
2180 PRINT "*****"
2190 PRINT "*****"
2200 PRINT "*****"
2210 PRINT "*****"
2220 PRINT "*****"
2230 PRINT "*****"
2240 PRINT "*****"
2250 PRINT "*****"
2260 PRINT "*****"
2270 PRINT "*****"
2280 PRINT "*****"
2290 PRINT "*****"
2300 PRINT "*****"
2310 PRINT "*****"
2320 PRINT "*****"
2330 PRINT "*****"
2340 PRINT "*****"
2350 PRINT "*****"
2360 PRINT "*****"
2370 PRINT "*****"
2380 PRINT "*****"
2390 PRINT "*****"
2400 PRINT "*****"
2410 PRINT "*****"
2420 PRINT "*****"
2430 PRINT "*****"
2440 PRINT "*****"
2450 PRINT "*****"
2460 PRINT "*****"
2470 PRINT "*****"
2480 PRINT "*****"
2490 PRINT "*****"
2500 PRINT "*****"

```

**** PROGRAM II ****

```

10 REM *** MERIC TRAKE ***
20 PRINT "*****"
30 PRINT "*****"
40 PRINT "*****"
45 GETA$: IFA$=""
50 INPUT "UNESITE BROJ SA BROJACA KASETOFONA"; BR
60 PRINT "*****"
70 PRINT "*****"
80 PRINT "*****"
90 IF (PEEK (1) AND 16) = 16 THEN 90
100 I$=""
110 GETA$: IFA$=""
120 V=T
130 INPUT "UNESITE BROJ SA BROJACA KASETOFONA"; BR
140 PRINT "*****"
150 PRINT "*****"
160 PRINT "*****"

```

READY.

Zatim će vas upitati za dužinu kasete i obavestiti da pritisnete na kasetofonu taster za brzo prematavanje. Kasetofon će se sam zaustaviti pre početka programa. Tada ćete pritisnuti „STOP“ na kasetofonu, „RETURN“ na tastaturi i „PLAY“ na kasetofonu i traženi program će se učitati. Ako računar ne nađe program, uvećate za jedan

```

2510 PRINT "*****"
2520 PRINT "*****"
2530 PRINT "*****"
2540 PRINT "*****"
2550 PRINT "*****"
2560 PRINT "*****"
2570 PRINT "*****"
2580 PRINT "*****"
2590 PRINT "*****"
2600 PRINT "*****"
2610 PRINT "*****"
2620 PRINT "*****"
2630 PRINT "*****"
2640 PRINT "*****"
2650 PRINT "*****"
2660 PRINT "*****"
2670 PRINT "*****"
2680 PRINT "*****"
2690 PRINT "*****"
2700 PRINT "*****"
2710 PRINT "*****"
2720 PRINT "*****"
2730 PRINT "*****"
2740 PRINT "*****"
2750 PRINT "*****"
2760 PRINT "*****"
2770 PRINT "*****"
2780 PRINT "*****"
2790 PRINT "*****"
2800 PRINT "*****"
2810 PRINT "*****"
2820 PRINT "*****"
2830 PRINT "*****"
2840 PRINT "*****"
2850 PRINT "*****"
2860 PRINT "*****"
2870 PRINT "*****"
2880 PRINT "*****"
2890 PRINT "*****"
2900 PRINT "*****"
2910 PRINT "*****"
2920 PRINT "*****"
2930 PRINT "*****"
2940 PRINT "*****"
2950 PRINT "*****"
2960 PRINT "*****"
2970 PRINT "*****"
2980 PRINT "*****"
2990 PRINT "*****"
3000 PRINT "*****"
3010 PRINT "*****"
3020 PRINT "*****"
3030 PRINT "*****"
3040 PRINT "*****"
3050 PRINT "*****"
3060 PRINT "*****"
3070 PRINT "*****"
3080 PRINT "*****"
3090 PRINT "*****"
3100 PRINT "*****"
3110 PRINT "*****"
3120 PRINT "*****"
3130 PRINT "*****"
3140 PRINT "*****"
3150 PRINT "*****"
3160 PRINT "*****"
3170 PRINT "*****"
3180 PRINT "*****"
3190 PRINT "*****"
3200 PRINT "*****"
3210 PRINT "*****"
3220 PRINT "*****"
3230 PRINT "*****"
3240 PRINT "*****"
3250 PRINT "*****"
3260 PRINT "*****"
3270 PRINT "*****"
3280 PRINT "*****"
3290 PRINT "*****"
3300 PRINT "*****"
3310 PRINT "*****"
3320 PRINT "*****"
3330 PRINT "*****"
3340 PRINT "*****"
3350 PRINT "*****"
3360 PRINT "*****"
3370 PRINT "*****"
3380 PRINT "*****"
3390 PRINT "*****"
3400 PRINT "*****"
3410 PRINT "*****"
3420 PRINT "*****"
3430 PRINT "*****"
3440 PRINT "*****"
3450 PRINT "*****"
3460 PRINT "*****"
3470 PRINT "*****"
3480 PRINT "*****"
3490 PRINT "*****"
3500 PRINT "*****"

```

ili više broj koji se oduzima od promeni je „CT“ u liniji 320 glavnog programa. Bilo kakva izmena u programu 1 doveće do dubljenja podataka i zato ga prvo testirajte za samo dva-tri programa sa jedne kasete.

Dorde Janković

Spektrum
YU TASWORD II

Jedna od najvećih nedostataka kvalitetnih inostranih tekst procesora je, svakako, odsustvo specifičnih „YU“ slova (č, ž, š i ć). Ni printeri, na žalost, nisu bolji – ni oni u svome ROM-u najčešće ne sadrže naše latinično pismo. Da li rešili ove probleme, vlasnici kućnih računara obično menjaju ROM u svome printeru, a sam tekst procesor

sor prevajajući tako da neke retko korišćene znake (kao što su uglaste zagrade, „majmunkso a“ itd.) koriste da bi označili naša slova.

Ovo komplikovano, a i skupo rešenje, navelo je autora ovog teksta da napiše program koji će ispraviti ove nedostatke i bez mnogo novca (samo cena ovog broja „Računara“) rešiti problem YU slova. Program je pisan za računar „spektrum“ i služi da se njime izvrši programska modifikacija poznatog tekst procesora tasword II. Ovakvo modifikovana verzija tasworda prepoznaje velika i mala slova č, ž, š i ć bez potrebe za promenom ROM-a štampa ili na printerima EPSON RX 80 – 80 F/T. Za ostale štampače potrebne su male izmene u programu. Osnovna ideja programa za štampanje

YU slova sastoji se u tome da se YU znacima dodele specifični kodovi (160 do 167) koje će program tumaćiti kao „normalne“ i na ekranu ih prikazivati konsultujući deo memorije u kome se nalaze definicije izgleda znakova tasword prikazuje 64 karakteru u redu). Pošto je prepoznat kod određenog YU slova, ono biva poslato na ekran u svom normalnom izgledu. Prilikom štampanja teksta na printeru, svi se YU karakteri štampaju u BIT IMAGE modu i to imajući u vidu tip slova koje se upravo štampa kao i mod u kome se printer nalazi. Samo ukucavanje YU slova sa tastature je veoma jednostavno – potrebno je samo ukucati osnovno slovo i apostrof. Oni kojima se ne dopadaju ovakva dodela mogu lako, jednostavnim izmenama u programu, da uvedu

svoju.

Na primer:

č = c' (c i posle njega dva puta apostrof)

ć = c'

ž = z'

š = s'

Na koji način treba izvršiti programsku modifikaciju tasvarda? Prvo učitate taj svod II i iz njega se vratite u bejzik, a potom i, bez ikakvog brisanja prethodnog programa, asembler GENS i to direktno naredbom: LOAD „GENS“ CODE 35000: RANDOMIZE USR 35000

Kada učitate GENS, upišite u njega program koji je dat u ovaj članak i asemblerajte ga. Kada je asembleriranje završeno, vratite se u bejzik i editujte liniju 250 ili 290 i iz njih izbacite naredbe RANDOMIZE USR 59806. Posle ovoga tasvard možete startovati sa RUN i snimiti ovu izmenjenu verziju na kasetu. Ovom programskom modifikaci-

TEST YU KARAKTERA

Pica size..... CČŠšZžZćć QWERTY qwerty
Enlarged..... CČŠšZžZćć QWERTY qwer
Ty..... CČŠšZžZćć QWERTY qwerty
Emfazis..... CČŠšZžZćć QWERTY qwerty
Emfazis Enlarged..... CČŠšZžZćć QWERTY qwer
Ty..... CČŠšZžZćć QWERTY qwerty
Kondenzed..... CČŠšZžZćć QWERTY qwerty
Kondenzed Enlarged..... CČŠšZžZćć QWERTY qwerty
Elite size..... CČŠšZžZćć QWERTY qwerty
Enlarged..... CČŠšZžZćć QWERTY qwerty

HISOFT GENSMZ ASSEMBLER

ZK SPECTRUM

Copyright (C) HISOFT 1983,4
All rights reserved

Pass 1 errors: 00

F70A 180 ORG #F70A IPROGRAMSKE MODIFIKACIJE
F70A C3E5E2 20 JP HARGIN I;ODNOVNO PROGRAMA TASVARD II
E76D C02CE3 40 CALL PROBA1
E77A 0800 40 ORG #E77A
E77A C033E3 40 CALL PROBA2
FCAB 70 ORG #FCAB
FCAB CDC4E2 60 CAG START
ED9E 90 ORG #ED9E
ED9E C349E3 180 JP PRINT
F400 110 ORG #F400
F400 31E4 120 DEFM INIC
E000 130 ORG #E000
E000 AC180181 140 DEFIB 180,27,129,129 IDEFINISANJE PIKA SIZE
E000 AC2084C2 160 CCI DEFIB 120,66,0,66 ISETA YU KARAKTERA
E000 80420844 160 DEFIB 120,66,0,66
E000 A0000000 170 DEFIB 160,0,0,0
E110 0000 160 DEFIB 0,0,0,0
E110 8C100220 190 cc1 DEFIB 12,16,2,32
E116 82E00240 200 DEFIB 2,224,2,160
E116 00000000 210 DEFIB 0,0,0,0
E116 0000 220 DEFIB 0,0
E120 82044A00 230 Z1 DEFIB 2,4,74,0
E124 1208E200 240 DEFIB 0,120,226,0
E12C A0000000 250 ZC1 DEFIB 64,0,0,0
E2C0 0000 260 DEFIB 0,0
E2C0 82082A00 270 Z1 DEFIB 2,0,30,0
E3C2 2AC85200 280 DEFIB 42,192,50,120
E3C4 20000000 290 DEFIB 32,0,0,0
E3C4 0000 300 DEFIB 0,0
E3C0 84224052 310 01 DEFIB 4,34,64,82
E400 8400204C 320 DEFIB 120,210,120,76
E44 00000000 330 DEFIB 64,0,0,0
E400 0000 340 DEFIB 0,0
E44 82100220 350 61 DEFIB 2,16,2,40
E44 82E00440 360 DEFIB 0,232,4,160
E5C0 00000000 370 DEFIB 0,0,0,0
E5C4 0000 380 DEFIB 0,0
E5C0 8C120842 390 C3 DEFIB 12,16,32,66
E5C4 82084400 400 DEFIB 0,66,120,60
E6A0 20000000 410 DEFIB 32,0,0,0
E6A4 0000 420 DEFIB 0,0
E6A4 8C100220 430 c1 DEFIB 12,16,2,32
E6A4 82A00240 440 DEFIB 2,96,2,160
E6A4 00000000 450 DEFIB 0,0,0,0
E72 0000 460 DEFIB 0,0
E200 470 ORG 57856
E200 80020504 480 TCC DEFIB 5,2,5,4 IDEFINISANJE YU KARAKTERA
E200 84050200 490 DEFIB 4,5,2,0 IZA PRIKAZ NA EKRANU
E200 80050205 500 Tcc DEFIB 0,5,2,5
E20C 84040300 510 DEFIB 4,4,3,0
E210 82070182 520 Tz DEFIB 2,7,1,2
E214 82040700 530 DEFIB 2,7,7,0
E218 80020701 540 Tz DEFIB 0,2,7,1
E21C 82040700 550 DEFIB 2,4,7,0
E220 82038402 560 T8 DEFIB 1,1,6,2
E224 81010A00 570 DEFIB 1,1,6,0
E228 80020304 580 T8 DEFIB 0,2,3,4
E22C 82010A00 590 DEFIB 2,1,6,0
E230 82020504 600 TC DEFIB 2,2,7,0
E234 84050200 610 DEFIB 4,5,2,0
E238 80020304 620 Tc DEFIB 0,2,3,4
E23C 84040300 630 DEFIB 4,4,3,0
E240 18 640 DEFIB 27
E241 2A040C00 650 B1STAF DEFIB ** ,4,12,0
E245 3CC20842 660 STARTA DEFIB 60,194,120,66 IDEFINISANJE ITALIC SIZE
E246 804208C2 670 DEFIB 0,66,120,194 ISETA YU KARAKTERA
E240 24000000 680 DEFIB 36,0,0,0
E251 0000 690 DEFIB 0,0
E251 8C220842 700 c'' DEFIB 18,34,0,162
E257 A00A0022 710 DEFIB 64,162,0,34
E250 00000000 720 DEFIB 0,0,0,0
E25F 0000 730 DEFIB 0,0
E261 804244CA 740 ZZ DEFIB 0,66,60,202
E265 18C24842 750 DEFIB 16,226,64,66
E269 00000000 760 DEFIB 0,0,0,0
E260 0000 770 DEFIB 0,0
E26F 23040240 780 Z' DEFIB 34,0,162,72
E275 A2102200 790 DEFIB 162,16,34,0
E277 00000000 000 DEFIB 0,0,0,0
E278 0000 010 DEFIB 0,0
E27D 245200D2 820 8' DEFIB 36,82,0,210
E281 08020052 830 DEFIB 0,210,0,82
E285 AC000000 040 DEFIB 76,0,0,82
E287 0000 050 DEFIB 0,0
E28B 182A00AA 860 8' DEFIB 16,42,0,170
E28F 40A0002A 870 DEFIB 64,170,0,42
E293 84000000 080 DEFIB 4,0,0,0
E297 0000 890 DEFIB 0,0
E299 3C420042 900 CC DEFIB 60,66,0,66
E29D 08C20042 910 DEFIB 120,194,0,66
E2A1 24000000 920 DEFIB 36,0,0,0
E2A5 0000 930 DEFIB 0,0
E2A7 1C220022 940 cc DEFIB 20,34,0,34
E2AB 00000022 950 DEFIB 64,162,0,34
E2AF 00000000 960 DEFIB 0,0,0,0
E2B3 0000 970 DEFIB 0,0
E2B5 00000000 980 MARGIN PUSH 0; IPOSTAVLJANJE LEVE MARGINE
E2B6 2101EE 990 LD H,3E0D1
E2B9 0A03 1000 LD B,3
E2BB 7E 1010 LD0PL LD A,0E
E2BC 00FC04 1020 CALL BYTD
E2BF 2B 1030 L -H
E2C0 10F9 1040 D0NZ LD0PL
E2C2 C1 1050 POP AF
E2C3 C9 1060 RET
E2C4 FE27 1070 START CP 39 ISTART PODPROGRAMA
E2C6 2086 1080 DEFIB 2,YUCAR IZA YU KARAKTERA
E2C8 35CFE2 1090 LD YUCAR,I1,A IPROGRAMSKE MODIFIKACIJA NA
1110
E2CB C37FFF 1100 JP #FF37
E2CE 3E20 1110 YUCAR LD A,32 I KRAKTERA KOJI TREBA PREDEFINISATI
E2D0 FE43 1120 CP 67 I1 DODELITI NOVI KOD KARAKTERU
E2D2 2084 1130 JP NZ,NEXT1 I KOJI DEFINISE YU KARAKTER
E2D4 3E46 1140 LD A,166
E2D6 1856 1150 JR OUT
E2D8 FE43 1160 NEXT1 CP 99
E2DA 2084 1170 JR NZ,NEXT2
E2DC 3E47 1180 LD A,167
E2DE 182E 1190 JR OUT
E2E0 FE5A 1200 NEXT2 CP 98
E2E2 2084 1210 JR NZ,NEXT3
E2E4 3E42 1220 LD A,162
E2E6 1826 1230 JR OUT
E2E8 FE7A 1240 NEXT3 CP 122
E2EA 2084 1250 JR NZ,NEXT4
E2EC 3E43 1260 LD A,163
E2EE 181E 1270 JR OUT
E2F0 FE83 1280 NEXT4 CP 93
E2F2 2084 1290 JR NZ,NEXT5
E2F4 3E44 1300 LD A,164
E2F6 1816 1310 JR OUT
E2F8 FE73 1320 NEXT5 CP 97
E2FA 2084 1330 JR NZ,NEXT6
E2FC 3E45 1340 LD A,165
E2FE 180E 1350 JR OUT
E300 FE6A 1360 NEXT6 CP 166
E302 2084 1370 JR NZ,NEXT7
E304 3E40 1380 LD A,160
E306 1806 1390 JR OUT
E308 FE77 1400 NEXT7 CP 167
E30A 2019 1410 JR NZ,NEXT8
E30C 3E41 1420 LD A,161
E30E FC 1430 OUT PUSH AF I KRAKTER SPREMAN ZA PRINTOVANJE
E310 3A00F3 1440 LD A,(#F300)
E312 CB0F 1450 BIT 3
E314 2085 1460 JR 2,NOEND
E316 3E80 1470 LD A,0
E318 0040FC 1480 CALL #FC04
E31C 1490 NOEND LD A,0
E31D C004FC 1500 CALL #FC04
E320 F1 1510 POP AF

jom tasvoda otklonjen je i bag koji je uzrokovao da ako u toku štampanja pritisnemo BREAK dolazi do uništenja teksta. Urađena je, takođe, i nova rutina za postavljanje leve margine teksta, koja direktno programira printer da postavi levu marginu teksta.

Program radi sa KEMPSTON interfejsom, ali se štampanje može izvršiti i na ZX printeru, na kome će takođe biti ispravno odštampana sva YU slova. Ukoliko se u tasvodu izabere mod koji prikazuje 32 znaka u redu, tada na ekranu nećemo imati YU slova, ali će oni biti na printeru ispravno odštampani.

Italic Character Set..... C0C5220C QWERTY qwerty
Enlarged..... C0C5220C QWERTY qwer
 t y
Emfazis..... C0C5220C QWERTY qwerty
Emfazis Enlarged..... C0C5220C QWERTY qwer
 t y
Kondenzed..... C0C5220C QWERTY qwerty
Kondenzed Enlarged..... C0C5220C QWERTY qwerty
Elite size..... C0C5220C QWERTY qwerty
Enlarged..... C0C5220C QWERTY qwer

Dragoslav Jovanović

E321 32CFE2	1520	LD	(YUCAR+1),A	E3C9 FE80	2320	CF 0	
E324 C337FF	1530	JP	#FF37 ;POVRATAK U TASMOR II	E3CB 288C	2330	JR	NZ,HEM1
E327 3E27	1540	NEXTB	LD A,39	E3CD 30	2340	DEC	A
E329 C337FF	1550	JP	#FF37	E3CE 77	2350	LD	(HL),A
E32C C03AE3	1560	PROBA1	CALL TESTYU ;TESTIRANJE DALI JE	E3CF 2B	2360	DEC	HL
E32F 3225E7	1570	LD	(#E775),A ;YU KARAKTER	E3D0 47	2370	LD	B,A
E332 C9	1580	RET		E3D1 3D	2380	DEC	A
E333 C03AE3	1590	PROBA2	CALL TESTYU ;TESTIRANJE DALI JE	E3D2 77	2390	LD	(HL),A
E336 320E27	1600	LD	(#E782),A ;YU KARAKTER	E3D3 3E23	2400	LD	A,35
E339 C9	1610	RET		E3D5 32F3E3	2410	LD	(ELIT),A
E33A 2B00	1620	TESTYU	LD H,0	E3D8 25	2420	INC	HL
E33C AF	1630	LD	L,A	E3D9 3E81	2430	MEM1	LD
E33D FE9F	1640	CF	159	E3DB FE8E	2440	CF	14
E33F 3EEE	1650	LD	A,8EE	E3DD 2002	2450	JR	NZ,NORMAL
E341 0B	1660	RET		E3DF CB26	2460	SXA	(HL)
E342 3E0D	1670	LD	A,8DD	E3E1 EB	2470	NORMAL	EX DE,HL
E344 C9	1680	RET		E3E2 C5	2480	PUSH	BC
E345 D680	1690	PRINT	SUB #00 ;ODREĐIVAJE PRINT MODA	E3E3 06A5	2490	LD	B,5
E347 FE0A	1700	CF	10 ;I TIPIA KARAKTERA	E3E5 1140E2	2500	LD	DE,BISTAD-1
E349 2003	1710	JR	NZ,NOEMFA	E3EB 1A	2510	LOOPB1	LD A,(DE)
E34B 329FE3	1720	LD	(MEM5+1),A	E3EF C0DFE4	2520	CALL	BYTOUT
E34E FE80	1730	NOEMFA	CF 5	E3EC 13	2530	INC	DE
E350 2003	1740	JR	NZ,KONDEZ	E3ED 10F9	2540	DJNZ	LDOOP1
E352 329FE3	1750	LD	(MEM5+1),A	E3EF C1	2550	POP	BC
E355 FE0D	1760	KONDEZ	CF 13	E3F0 AF	2560	XOR	A
E357 2003	1770	JR	NZ,NOKOND	E3F1 B6	2570	LOOP1	OR (HL)
E35A 32A7E3	1780	LD	(MEM2+1),A	E3F2 57	2580	LD	B,A
E35C FE82	1790	NOKOND	CF 2	E3F3 00	2590	ELIT	NOF
E35E 2003	1800	JR	NZ,ENLAR	E3FA C0DFE4	2600	CALL	BYTOUT
E360 32A7E3	1810	LD	(MEM2+1),A	E3F7 3ADA33	2610	LD	A,(MEM1+1)
E363 FE0E	1820	ENLAR	CF 14	E3FA CB3F	2620	SRL	A
E365 2003	1830	JR	NZ,NOENLA	E3FC 5885	2630	JR	C,NDCALL
E367 32DAE3	1840	LD	(MEM1+1),A	E3FE 7A	2640	LD	A
E36A FE01	1850	NOENLA	CF 1	E3FF B6	2650	OR	(HL)
E36C 2003	1860	JR	NZ,ENLAR	E400 C0DFE4	2660	CALL	BYTOUT
E36E 32DAE3	1870	LD	(MEM1+1),A	E403 7E	2670	NDCALL	LD A,(HL)
E371 FE00	1880	ENLAR	B	E404 23	2680	INC	HL
E373 2003	1890	JR	NZ,ELITND	E405 18EA	2690	DJNZ	LOOP1
E375 32CBE3	1900	LD	(MEM7+1),A	E407 AF	2700	XOR	A
E378 FE07	1910	ELITND	CF 7	E408 32F3E3	2710	LD	(ELIT),A
E37A 2003	1920	JR	NZ,ELITOK	E40B 01	2720	POP	DE
E37C 32CBE3	1930	LD	(MEM7+1),A	E40C 3B9ED	2730	JP	HEB9 ;POVRATAK U TASMOR II
E37F FE0B	1940	ELITOK	CF 11	E40F C5	2740	BYTOUT	PUSH BC ;ISLANJE SADRZAJA AKUMULATORA
E381 2003	1950	JR	NZ,INDITAL	E410 E5	2750	PUSH	HL ;PREKO PARALALANDG INTERFEISA
E383 32B7E3	1960	LD	(MEM4+1),A	E411 F5	2760	PUSH	AF ;INA PRINTER
E386 FE04	1970	NDITAL	CF 4	E412 01BF2E	2770	LD	BC,50047
E388 2003	1980	JR	NZ,ITAL	E415 C0D41F	2780	01	CALL 8020 ;TEST BREAK TASTERA
E38B 32B7E3	1990	LD	(MEM4+1),A	E418 D200D	2790	INC	BC,3528
E38D 87	2000	ITAL	ADD A,A	E41B ED78	2800	IN	A,(C)
E38F FE21	2010	CF	33	E41D 1F	2810	RRA	
E390 DAA1ED	2020	JP	C,#EDA1	E41E 38F5	2820	JR	C,01
E393 D5	2030	PUSH	DE	E420 F1	2830	POP	AF
E394 AF	2040	LD	H,A	E421 05	2840	DEC	B
E395 2680	2050	LD	H,0	E422 05	2850	DEC	B
E397 E5	2060	PUSH	HL	E423 ED79	2860	OUT	(C),A
E398 01	2070	POP	DE	E425 0AE3	2870	LD	B,227
E399 0606	2080	LD	B,6	E427 3E0E	2880	LD	A,14
E39B 19	2090	LOOP7	ADD HL,DE	E429 ED79	2890	OUT	(C),A
E39C 18FD	2100	DJNZ	DOBB7	E42B 3C	2900	INC	A
E39E 3E85	2110	MEM5	LD A,5	E42C ED79	2910	OUT	(C),A
E3A0 FE0A	2120	CF	10	E42E E1	2920	POP	HL
E3A2 3E01	2130	LD	A,1	E42F C1	2930	POP	BC
E3A4 280A	2140	JR	2,80	E430 C9	2940	RET	
E3A6 3E02	2150	HEM2	LD A,2	E431 01BF33	2950	INC	LD
E3A8 FE0D	2160	CF	13	E434 3E81	2960	LD	BC,58303 ;INICIJALIZACIJA PIO 8255
E3AA 3E03	2170	LD	A,3	E436 ED79	2970	OUT	(C),A
E3AC 2002	2180	JR	2,80	E438 3E0F	2980	LD	A,15
E3AE 3E02	2190	JR	A,2	E43A ED79	2990	OUT	(C),A
E3B0 3240E2	2200	GO	LD (BISTAD+1),A	E43C 24F5C	3000	INC	HL,(23631) ;IZMENA PODATAKA KANILA
E3B6 3E04	2210	LD	BC,STARTA-44B	E43F 010F00	3010	LD	BC,15 ;ZA ISLANJE NA PRINTER
E3B8 FE0B	2220	HEM4	LD A,4	E442 09	3020	ADD	HL,BC
E3BA 3E08	2230	CF	11	E443 110FE4	3030	LD	DE,BYTOUT
E3BC 2003	2240	JR	NZ,ITALIC	E446 73	3040	LD	(HL),E
E3BE 0144EC	2250	LD	BC,6404A	E447 25	3050	INC	HL
E3BF 09	2260	ITALIC	ADD BC,BC	E448 73	3060	LD	(HL),D
E3C0 EB	2270	EX	DE,HL	E449 C9	3070	RET	
E3C1 2143E2	2280	LD	HL,BISTAD+2				
E3C4 060C	2290	LD	B,12				
E3C6 78	2300	LD	(HL),B				
E3C7 3E87	2310	HEM7	LD A,7				

Pass 2 errors: 00

Table used: 775 from 800

Majstorije na računaru komodor 64

srpskohrvatski bejzik

Sve reči koje jedan računar ume da prepoznaje LOAD, PRINT, PRINT# i tako dalje nazivaju se službene reči. Spisak svih ovih reči nalazi se u ROM-u računara, u obliku tabele. Kod "komodora 64" ova tabela počinje na adresi \$A09E i dugačka je tačno 256 bajtova. Poslednji bajt u tabeli ima vrednost 0 i po njemu računar raspoznaje kraj tabele. Reči su

Službene reči...
Službene reči, naravno, nije moguće menjati direktno u ROM memoriji, već je potrebno oformiti, negde na pogodnom mestu u RAM-u, potpuno novu tabelu, napisati nove rutine za rad sa tabelom i promeniti vektore na trećoj strani koji ukazuju na ove rutine. Tek posle ovoga računara će biti u stanju da razume nove reči kao što su

kod za službenu reč FOR, a u ostala tri teksti MAT. Ovakav način pamćenja i interpretacije bejzik programa u računaru C-64 obezbeđuje duž rutine.

Prva rutina se nalazi u okviru programa LIST i služi za prevođenje "tokena" u čitljive reči. Ova rutina nalazi se na adresi \$A71A i u nju se ulazi preko vektora \$0306. Rutina BTOK u asemblerskom programa

Druga rutina radi u okviru interpreterske petlje i služi za pronalaženje službenih reči u tekstu i njihovo pretvaranje u "tokene". Nalazi se na adresi \$A57C i poziva se preko vektora \$030A. Njoj u asemblerskom listingu odgovara rutina TOKB. Sve što je rečeno za prethodnu rutinu važi i za ovu. Po uočeњу i inicijalizaciji ove rutine, sve naredbe se moraju kucati

```
asemblerki zapis rutina  
bejzika.VU  
1800 sys74096  
1810 not 0  
1820 == #C000  
1830  
1840  
1850  
1860  
1870  
1880  
1890  
1900  
1910  
1920  
1930  
1940  
1950  
1960  
1970  
1980  
1990  
2000  
2010  
2020  
2030  
2040  
2050  
2060  
2070  
2080  
2090  
2100  
2110  
2120  
2130  
2140  
2150  
2160  
2170  
2180  
2190  
2200  
2210  
2220  
2230  
2240  
2250  
2260  
2270  
2280  
2290  
2300  
2310  
2320  
2330  
2340  
2350  
2360  
2370  
2380  
2390  
2400  
2410  
2420  
2430  
2440  
2450  
2460  
2470  
2480  
2490  
2500  
2510  
2520  
2530  
2540  
2550  
2560  
2570  
2580  
2590  
2600  
2610  
2620  
2630  
2640  
2650  
2660  
2670  
2680  
2690  
2700  
2710  
2720  
2730  
2740  
2750  
2760  
2770  
2780  
2790  
2800  
2810  
2820  
2830  
2840  
2850  
2860  
2870  
2880  
2890  
2900  
2910  
2920  
2930  
2940  
2950  
2960  
2970  
2980  
2990  
3000  
3010  
3020  
3030  
3040  
3050  
3060  
3070  
3080  
3090  
3100  
3110  
3120  
3130  
3140  
3150  
3160  
3170  
3180  
3190  
3200  
3210  
3220  
3230  
3240  
3250  
3260  
3270  
3280  
3290  
3300  
3310  
3320  
3330  
3340  
3350  
3360  
3370  
3380  
3390  
3400  
3410  
3420  
3430  
3440  
3450  
3460  
3470  
3480  
3490  
3500  
3510  
3520  
3530  
3540  
3550  
3560  
3570  
3580  
3590  
3600  
3610  
3620  
3630  
3640  
3650  
3660  
3670  
3680  
3690  
3700  
3710  
3720  
3730  
3740  
3750  
3760  
3770  
3780  
3790  
3800  
3810  
3820  
3830  
3840  
3850  
3860  
3870  
3880  
3890  
3900  
3910  
3920  
3930  
3940  
3950  
3960  
3970  
3980  
3990  
4000  
4010  
4020  
4030  
4040  
4050  
4060  
4070  
4080  
4090  
4100  
4110  
4120  
4130  
4140  
4150  
4160  
4170  
4180  
4190  
4200  
4210  
4220  
4230  
4240  
4250  
4260  
4270  
4280  
4290  
4300  
4310  
4320  
4330  
4340  
4350  
4360  
4370  
4380  
4390  
4400  
4410  
4420  
4430  
4440  
4450  
4460  
4470  
4480  
4490  
4500  
4510  
4520  
4530  
4540  
4550  
4560  
4570  
4580  
4590  
4600  
4610  
4620  
4630  
4640  
4650  
4660  
4670  
4680  
4690  
4700  
4710  
4720  
4730  
4740  
4750  
4760  
4770  
4780  
4790  
4800  
4810  
4820  
4830  
4840  
4850  
4860  
4870  
4880  
4890  
4900  
4910  
4920  
4930  
4940  
4950  
4960  
4970  
4980  
4990  
5000  
5010  
5020  
5030  
5040  
5050  
5060  
5070  
5080  
5090  
5100  
5110  
5120  
5130  
5140  
5150  
5160  
5170  
5180  
5190  
5200  
5210  
5220  
5230  
5240  
5250  
5260  
5270  
5280  
5290  
5300  
5310  
5320  
5330  
5340  
5350  
5360  
5370  
5380  
5390  
5400  
5410  
5420  
5430  
5440  
5450  
5460  
5470  
5480  
5490  
5500  
5510  
5520  
5530  
5540  
5550  
5560  
5570  
5580  
5590  
5600  
5610  
5620  
5630  
5640  
5650  
5660  
5670  
5680  
5690  
5700  
5710  
5720  
5730  
5740  
5750  
5760  
5770  
5780  
5790  
5800  
5810  
5820  
5830  
5840  
5850  
5860  
5870  
5880  
5890  
5900  
5910  
5920  
5930  
5940  
5950  
5960  
5970  
5980  
5990  
6000  
6010  
6020  
6030  
6040  
6050  
6060  
6070  
6080  
6090  
6100  
6110  
6120  
6130  
6140  
6150  
6160  
6170  
6180  
6190  
6200  
6210  
6220  
6230  
6240  
6250  
6260  
6270  
6280  
6290  
6300  
6310  
6320  
6330  
6340  
6350  
6360  
6370  
6380  
6390  
6400  
6410  
6420  
6430  
6440  
6450  
6460  
6470  
6480  
6490  
6500  
6510  
6520  
6530  
6540  
6550  
6560  
6570  
6580  
6590  
6600  
6610  
6620  
6630  
6640  
6650  
6660  
6670  
6680  
6690  
6700  
6710  
6720  
6730  
6740  
6750  
6760  
6770  
6780  
6790  
6800  
6810  
6820  
6830  
6840  
6850  
6860  
6870  
6880  
6890  
6900  
6910  
6920  
6930  
6940  
6950  
6960  
6970  
6980  
6990  
7000  
7010  
7020  
7030  
7040  
7050  
7060  
7070  
7080  
7090  
7100  
7110  
7120  
7130  
7140  
7150  
7160  
7170  
7180  
7190  
7200  
7210  
7220  
7230  
7240  
7250  
7260  
7270  
7280  
7290  
7300  
7310  
7320  
7330  
7340  
7350  
7360  
7370  
7380  
7390  
7400  
7410  
7420  
7430  
7440  
7450  
7460  
7470  
7480  
7490  
7500  
7510  
7520  
7530  
7540  
7550  
7560  
7570  
7580  
7590  
7600  
7610  
7620  
7630  
7640  
7650  
7660  
7670  
7680  
7690  
7700  
7710  
7720  
7730  
7740  
7750  
7760  
7770  
7780  
7790  
7800  
7810  
7820  
7830  
7840  
7850  
7860  
7870  
7880  
7890  
7900  
7910  
7920  
7930  
7940  
7950  
7960  
7970  
7980  
7990  
8000  
8010  
8020  
8030  
8040  
8050  
8060  
8070  
8080  
8090  
8100  
8110  
8120  
8130  
8140  
8150  
8160  
8170  
8180  
8190  
8200  
8210  
8220  
8230  
8240  
8250  
8260  
8270  
8280  
8290  
8300  
8310  
8320  
8330  
8340  
8350  
8360  
8370  
8380  
8390  
8400  
8410  
8420  
8430  
8440  
8450  
8460  
8470  
8480  
8490  
8500  
8510  
8520  
8530  
8540  
8550  
8560  
8570  
8580  
8590  
8600  
8610  
8620  
8630  
8640  
8650  
8660  
8670  
8680  
8690  
8700  
8710  
8720  
8730  
8740  
8750  
8760  
8770  
8780  
8790  
8800  
8810  
8820  
8830  
8840  
8850  
8860  
8870  
8880  
8890  
8900  
8910  
8920  
8930  
8940  
8950  
8960  
8970  
8980  
8990  
9000  
9010  
9020  
9030  
9040  
9050  
9060  
9070  
9080  
9090  
9100  
9110  
9120  
9130  
9140  
9150  
9160  
9170  
9180  
9190  
9200  
9210  
9220  
9230  
9240  
9250  
9260  
9270  
9280  
9290  
9300  
9310  
9320  
9330  
9340  
9350  
9360  
9370  
9380  
9390  
9400  
9410  
9420  
9430  
9440  
9450  
9460  
9470  
9480  
9490  
9500  
9510  
9520  
9530  
9540  
9550  
9560  
9570  
9580  
9590  
9600  
9610  
9620  
9630  
9640  
9650  
9660  
9670  
9680  
9690  
9700  
9710  
9720  
9730  
9740  
9750  
9760  
9770  
9780  
9790  
9800  
9810  
9820  
9830  
9840  
9850  
9860  
9870  
9880  
9890  
9900  
9910  
9920  
9930  
9940  
9950  
9960  
9970  
9980  
9990  
10000  
10010  
10020  
10030  
10040  
10050  
10060  
10070  
10080  
10090  
10100  
10110  
10120  
10130  
10140  
10150  
10160  
10170  
10180  
10190  
10200  
10210  
10220  
10230  
10240  
10250  
10260  
10270  
10280  
10290  
10300  
10310  
10320  
10330  
10340  
10350  
10360  
10370  
10380  
10390  
10400  
10410  
10420  
10430  
10440  
10450  
10460  
10470  
10480  
10490  
10500  
10510  
10520  
10530  
10540  
10550  
10560  
10570  
10580  
10590  
10600  
10610  
10620  
10630  
10640  
10650  
10660  
10670  
10680  
10690  
10700  
10710  
10720  
10730  
10740  
10750  
10760  
10770  
10780  
10790  
10800  
10810  
10820  
10830  
10840  
10850  
10860  
10870  
10880  
10890  
10900  
10910  
10920  
10930  
10940  
10950  
10960  
10970  
10980  
10990  
11000  
11010  
11020  
11030  
11040  
11050  
11060  
11070  
11080  
11090  
11100  
11110  
11120  
11130  
11140  
11150  
11160  
11170  
11180  
11190  
11200  
11210  
11220  
11230  
11240  
11250  
11260  
11270  
11280  
11290  
11300  
11310  
11320  
11330  
11340  
11350  
11360  
11370  
11380  
11390  
11400  
11410  
11420  
11430  
11440  
11450  
11460  
11470  
11480  
11490  
11500  
11510  
11520  
11530  
11540  
11550  
11560  
11570  
11580  
11590  
11600  
11610  
11620  
11630  
11640  
11650  
11660  
11670  
11680  
11690  
11700  
11710  
11720  
11730  
11740  
11750  
11760  
11770  
11780  
11790  
11800  
11810  
11820  
11830  
11840  
11850  
11860  
11870  
11880  
11890  
11900  
11910  
11920  
11930  
11940  
11950  
11960  
11970  
11980  
11990  
12000  
12010  
12020  
12030  
12040  
12050  
12060  
12070  
12080  
12090  
12100  
12110  
12120  
12130  
12140  
12150  
12160  
12170  
12180  
12190  
12200  
12210  
12220  
12230  
12240  
12250  
12260  
12270  
12280  
12290  
12300  
12310  
12320  
12330  
12340  
12350  
12360  
12370  
12380  
12390  
12400  
12410  
12420  
12430  
12440  
12450  
12460  
12470  
12480  
12490  
12500  
12510  
12520  
12530  
12540  
12550  
12560  
12570  
12580  
12590  
12600  
12610  
12620  
12630  
12640  
12650  
12660  
12670  
12680  
12690  
12700  
12710  
12720  
12730  
12740  
12750  
12760  
12770  
12780  
12790  
12800  
12810  
12820  
12830  
12840  
12850  
12860  
12870  
12880  
12890  
12900  
12910  
12920  
12930  
12940  
12950  
12960  
12970  
12980  
12990  
13000  
13010  
13020  
13030  
13040  
13050  
13060  
13070  
13080  
13090  
13100  
13110  
13120  
13130  
13140  
13150  
13160  
13170  
13180  
13190  
13200  
13210  
13220  
13230  
13240  
13250  
13260  
13270  
13280  
13290  
13300  
13310  
13320  
13330  
13340  
13350  
13360  
13370  
13380  
13390  
13400  
13410  
13420  
13430  
13440  
13450  
13460  
13470  
13480  
13490  
13500  
13510  
13520  
13530  
13540  
13550  
13560  
13570  
13580  
13590  
13600  
13610  
13620  
13630  
13640  
13650  
13660  
13670  
13680  
13690  
13700  
13710  
13720  
13730  
13740  
13750  
13760  
13770  
13780  
13790  
13800  
13810  
13820  
13830  
13840  
13850  
13860  
13870  
13880  
13890  
13900  
13910  
13920  
13930  
13940  
13950  
13960  
13970  
13980  
13990  
14000  
14010  
14020  
14030  
14040  
14050  
14060  
14070  
14080  
14090  
14100  
14110  
14120  
14130  
14140  
14150  
14160  
14170  
14180  
14190  
14200  
14210  
14220  
14230  
14240  
14250  
14260  
14270  
14280  
14290  
14300  
14310  
14320  
14330  
14340  
14350  
14360  
14370  
14380  
14390  
14400  
14410  
14420  
14430  
14440  
14450  
14460  
14470  
14480  
14490  
14500  
14510  
14520  
14530  
14540  
14550  
14560  
14570  
14580  
14590  
14600  
14610  
14620  
14630  
14640  
14650  
14660  
14670  
14680  
14690  
14700  
14710  
14720  
14730  
14740  
14750  
14760  
14770  
14780  
14790  
14800  
14810  
14820  
14830  
14840  
14850  
14860  
14870  
14880  
14890  
14900  
14910  
14920  
14930  
14940  
14950  
14960  
14970  
14980  
14990  
15000  
15010  
15020  
15030  
15040  
15050  
15060  
15070  
15080  
15090  
15100  
15110  
15120  
15130  
15140  
15150  
15160  
15170  
15180  
15190  
15200  
15210  
15220  
15230  
15240  
15250  
15260  
15270  
15280  
15290  
15300  
15310  
15320  
15330  
15340  
15350  
15360  
15370  
15380  
15390  
15400  
15410  
15420  
15430  
15440  
15450  
15460  
15470  
15480  
15490  
15500  
15510  
15520  
15530  
15540  
15550  
15560  
15570  
15580  
15590  
15600  
15610  
15620  
15630  
15640  
15650  
15660  
15670  
15680  
15690  
15700  
15710  
15720  
15730  
15740  
15750  
15760  
15770  
15780  
15790  
15800  
15810  
15820  
15830  
15840  
15850  
15860  
15870  
15880  
15890  
15900  
15910  
15920  
15930  
15940  
15950  
15960  
15970  
15980  
15990  
16000  
16010  
16020  
16030  
16040  
16050  
16060  
16070  
16080  
16090  
16100  
16110  
16120  
16130  
16140  
16150  
16160  
16170  
16180  
16190  
16200  
16210  
16220  
16230  
16240  
16250  
16260  
16270  
16280  
16290  
16300  
16310  
16320  
16330  
16340  
16350  
16360  
16370  
16380  
16390  
16400  
16410  
16420  
16430  
16440  
16450  
16460  
16470  
16480  
16490  
16500  
16510  
16520  
16530  
16540  
16550  
16560  
16570  
16580  
16590  
16600  
16610  
16620  
16630  
16640  
16650  
16660  
16670  
16680  
16690  
16700  
16710  
16720  
16730  
16740  
16750  
16760  
16770  
16780  
16790  
16800  
16810  
16820  
16830  
16840  
16850  
16860  
16870  
16880  
16890  
16900  
16910  
16920  
16930  
16940  
16950  
16960  
16970  
16980  
16990  
17000  
17010  
17020  
17030  
17040  
17050  
17060  
17070  
17080  
17090  
17100  
17110  
17120  
17130  
17140  
17150  
17160  
17170  
17180  
17190  
17200  
17210  
17220  
17230  
17240  
17250  
17260  
17270  
17280  
17290  
17300  
17310  
17320  
17330  
17340  
17350  
17360  
17370  
17380  
17390  
17400  
17410  
17420  
17430  
17440  
17450  
17460  
17470  
17480  
17490  
17500  
17510  
17520  
17530  
17540  
17550  
17560  
17570  
17580  
17590  
17600  
17610  
17620  
17630  
17640  
17650  
17660  
17670  
17680  
17690  
17700  
17710  
17720  
17730  
17740  
17750  
17760  
17770  
17780  
17790  
17800  
17810  
17820  
17830  
17840  
17850  
17860  
17870  
17880  
17890  
17900  
17910  
17920  
17930  
17940  
17950  
17960  
17970  
17980  
17990  
18000  
18010  
18020  
18030  
18040  
18050  
18060  
18070  
18080  
18090  
18100  
18110  
18120  
18130  
18140  
18150  
18160  
18170  
18180  
18190  
18200  
18210  
18220  
18230  
18240  
18250  
18260  
18270  
18280  
18290  
18300  
18310  
18320  
18330  
18340  
18350  
18360  
18370  
18380  
18390  
18400  
18410  
18420  
18430  
18440  
18450  
18460  
18470  
18480  
18490  
18500  
18510  
18520  
18530  
18540  
18550  
18560  
18570  
18580  
18590  
18600  
18610  
18620  
18630  
18640  
18650  
18660  
18670  
18680  
18690  
18700  
18710  
18720  
18730  
18740  
18750  
18760  
18770  
18780  
18790  
18800  
18810  
18820  
18830  
18840  
18850  
18860  
18870  
18880  
18890  
18900  
18910  
18920  
18930  
18940  
18950  
18960  
18970  
18980  
18990  
19000  
19010  
19020  
19030  
19040  
19050  
19060  
19070  
19080  
19090  
19100  
19110  
19120  
19130  
19140  
19150  
19160  
19170  
19180  
19190  
19200  
19210  
19220  
19230  
19240  
19250  
19260  
19270  
19280  
19290  
19300  
19310  
19320  
19330  
19340  
19350  
19360  
19370  
19380  
19390  
19400  
19410  
19420  
19430  
19440  
19450  
19460  
19470  
19480  
19490  
19500  
19510  

```

Bejkica računara „komodor 64“ sigurno nije najbolja stvar koja se može videti — nezgrapna, nedovoljno instruktivna, ponekad sa žudnom sintaksom i mnogo „pešačkih“ rešenja preko POKE instrukcije, on pre pripada istoriji računara. Međutim, kada se taj isti bejkik pogleda iznutra, ovaj utisak se unekoliko menja. Proizvođač bejkika i operativnog sistema Microsoft ostavio je veliki broj mogućnosti za izmenu i bejkika i operativnog sistema. Gotovo sve važnije rutine iz ROM-a pozivaju se preko vektora na trećoj strani RAM-a instrukcijama JMP (\$03xx). Dovoljno je da korisnik napiše svoj program, koji će raditi umesto postojeće u ROM-u, i da postavi odgovarajući vektor tako da ukazuje na početak nove rutine i računare već radi drukčije. Naravno, ovdje treba biti veoma pažljiv, jer gotovo svaka greška dovodi do pada sistema i tada je jedini lek reset taster ili isključenje računara.

tabeli grešaka, koja je organizovana isto kao i tabela službenih reči. Odmah iza ove tabele u ROM-u je smještena lista početnih adresa za svaku moguću poruku. Pre ulaska u rutinu za ispisivanje greške u X registru mora se nalaziti redni broj greške. Rutina ERRO je kopija kraćeg WARM START rutine i početne rutine za prikaz grešaka. U njoj se jedino za razliku od

3851:	ipov moguce adrese	3532:	iporuka u greškama
3852:	ilispisuje	3533:	
3848:	bne dal i meina tevit	3540:	
3849:	it test stakle na	3541:	
3851:	ispoviz od rutine za	3542:	irutina za nai
3852:	ilispis poruka operativnog sistema	3543:	
3848:	ida #181x:	3590:	nni pha
3870:	cap #837	3510:	pha
3869:	bne sign ino_nezad u	3511:	tya
3871:	isoutput	3512:	tya
3872:	ida #104x:	3513:	pha
3180:	cap #841 ihigh bajt	3544:	ida #87f
3181:	bne sign ino_in_output	3545:	sta #0000
3182:	tya	3546:	idy #0000
3180:	id #0	3547:	bpi #e50
3148 govj:	cap obuf u pronalazanje	3548:	jap #f072
3141:	ilispisuje poruku na	3549:	fe50
3142:	ipomovu rastojanja	3178:	bne fe50
3143:	ipristigli od u rutinu	3179:	byte #6c,#82,#0B,jap
3144:	ida #01a:	(#0000)	
3178:	baq ppaw loh_pronadano	3178:	fe50
3179:	im	3179:	jar #f6bc
3180:	bne govj ino_pomovno tevit	3178:	fe50
3180:	ppaw ida nbuf u ilispis nove	3179:	jar #f672
3181:	isprouke	3179:	jar #f615
3190:	tya	3178:	jar #f62a
3200:	prad nva id ngory	3179:	ida #fichosta #32a
3201:	imovno	3179:	imovno
3202:	and #87f	3179:	ipostavi bejkik vektore
3203:	jar #f412	3179:	ida ukazuju na nove
3204:	iny	3179:	irutine
3205:	slp	3888:	ida #fichosta #327
3206:	slp pradi	3810:	ida #fichosta #310
3270:	pla i ispredi stak	3828:	ida #fichosta #319
3270:	lan	3829:	byte #6c,#82,#0B,jap
3290:	pla	(#0002)	
3300:	pla		
3310:	pla		
3320:	pla		
3330:	gic		
3340:	rtu		
3341:	iz rutine ispisuje		
3350:	sign pla inezad u output		
3350:	lan ispredi stak		
3380:	jap #f1ca ina output		
3390:	tya		
3400:			
3410:	inova rutina za prikaz poruka		
3420:	u greškama		
3430:			
3440:	pravo tva		
3450:	bpi fyaw igreska postovi		
3470:	cap #870 ino_in_ready		
3470:	fyaw tva		
3480:	axl a		
3490:	lan		
3500:	ida rtt_2, inzapuni adresu		
3501:	inove poruka		
3502:	sta #22		
3520:	ida rtt_1, ishig bajt		
3530:	jap #e4d3 inezad u staru		
3531:	irutinu za prikaz		

originalne rutine uz pomoć rednog broja greške biraju adrese novih poruka. Uvođenjem ove rutine, računare će sve poruke prikazivati na našem jeziku.

Računar je sada na najboljem putu da „progovori“ u potpunosti naš jezik. Sada se na engleskom prikazuju samo po-

39/srpskohrvatski
bejkik

ruke bejkik interpretera, kao što su READY, ERROR itd. i poruke operativnog sistema, na primer PRESS PLAY ON TAPE.

Zamena ovih poruka nije tako jednostavna kao što su bile prethodne, pošto rutine za njihov prikaz nisu rešene vektorskim pozivima. Jedna od ideja za rešavanje ovog problema bila je „prestreti“ neku vektorizovanu rutinu, pa testirati da li je

hiska dapp službenih reči i poruka u greškama	3850:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff
3850:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3851:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3852:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3853:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3854:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3855:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3856:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3857:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3858:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3859:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3860:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3861:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3862:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3863:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3864:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3865:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3866:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3867:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3868:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3869:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3870:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3871:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3872:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff	
3873:	cap #850 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 a0 a1	

„komodor“ u mreži

Interfejsi/ komodor 64

Zbog načina na koji je izveden, RS232C interfejs na „komodoru 64“ bi se mogao smatrati čistim dobrotkom za vlasnike koji firmu nije koštao ni dinara. Većina „komodorista“ smatra da je ovaj interfejs namenjen prvenstveno za povezivanje štampača. On, međutim, može dobiti i uzbudljiviju ulogu — njime se mogu povezati dva ili više računara u mrežu. Ova oblast je posebno interesantna zbog mogućnosti korišćenja jeftinog „komodora“ kao periferne jedinice na kojoj se može obaviti mnogo „ozbiljnog“ posla, a ono što prevazišla njegove potencijale poslati na obradu većem računaru.

Interfejs RS 232 je izveden na portu B čipa CIA #2. Operativni sistem, koristeći tajmer A istog čipa, generiše potreban „tajming“ izlaznog signala, dok se tajmerom B određuje tajming za ulazni signal i na osnovu toga formira primljeni bajt. Ovakvo rešenje je jeftinije od upotrebe specijalizovanog čipa, ali i postavlja određena ograničenja. Da bi „komodor“ detektovalao nailazeći signal bez potrebe da se stalno vrti u petlji, mora se koristiti interapt koji će biti generisan pri svakom prelasku signala sa „1“ na „0“ na ulaznoj liniji za podatke. Metod je jednostavan, ali je nevolja u tome što isti, ili sličan koncept koriste i tastatura, kasetofon, disk jedinica i printer... Kako bi bilo koji zastoj zbog nekog drugog interapta ili sprečavanja mašinskom instrukcijom SEI poremotio tačnost očitavanja ulaznog signala i praktično onemogućio rad interfejsa, morao je biti izabran nemaskirani interapt.

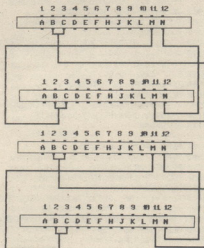
U apsolutnom prioritetu

Tako je RS232C dobio potpuni prioritet u odnosu na ostale periferije, pa se nepažljivim korišćenjem računara lako može zavrteti u mrtvoj petlji. Ako je otvoren kanal za RS232C, disk jedinica nikako neće uspeti da završi započeti posao učitavanja nekog programa! Isti interapt generiše i istovremeni pritisak na STOP i RESTORE tastere, ali ovo ne može da izazove zabunu: ako već želite da resetujete računara, onda i nije važno šta će se desiti sa komunikacijom.

Drugi problem koji proističe iz ovakvog rešenja je nemogućnost ostvarivanja svih brzina prenosa koji su definisani RS232C standardom. Pri otvaranju veze zadaje se željena brzina prenosa i na osnovu nje, operativni sistem iz tablice u ROM-u uzima potrebne podatke za tajmer A. Za brzina od 300 bauda sadržaj tajmera će biti 310 (treba za svaki bit da odobri 310 taktova sistemskog časovnika od 1 MHz), za 600 bauda je već samo 174, dok je za 1200 potrebno svega 105 otkucanja, tako da veće brzine nije moguće ostvariti. Zato je „komodor“ na ovoj brzini stao. Iako je u „Reference Guide“-u dat način otvaranja veze za sve standardne brzine, one preko 1200 bauda nije moguće koristiti. Po našem iskustvu i 1200 je već dovoljno kritična brzina pa je njeno korišćenje moguće samo u pažljivo napisanom mašinskom programu. Za modem se uglavnom koristi 300 bauda dok je

za „razgovor“ dva komodora 600 bauda sasvim prihvatljiva brzina pa ovo ne predstavlja posebno ograničenje.

Da bi isprobali komunikaciju preko RS232C interfejsa, najjednostavnije će biti da povežete dva „komodora“. Dovoljno su dva konektora za korisnički port i trožilni provodnik. Vežu izvedite na sledeći način:



SLIKA (1)

Ovakvim povezivanjem se ostvaruje najjednostavnija veza, bez hendšejkinga, pri kojoj su moguće greške u komunikaciji, ali može odlično da posluži za demonstraciju rada i razvoj softvera.

Otvoravanje veze

Veza za RS232C se otvara standardnom naredbom OPEN u sledećem obliku: OPEN logički broj veze,2,0,CHR\$(kontrolni bajt)+CHR\$(komandni bajt)

Polje za naziv pri otvaranju veze sa ostalim periferijama ovde je iskorišćeno za definisanje parametara komunikacije. Ako ste „komodor“ povezali na pokazan način, u jedan od njih ukucajte sledeći kratak program:

```
10 OPEN 30.2,0,CHR$(6)
20 GET #30,AS:IFAS<> THENPRINTAS:
30 GETBS:IFBS<> THEN PRINT#30,B-
$:PRINTBS
40 GOTO20
```

Ovim programom može se voditi razgovor dva korisnika. Izabrana je brzina od 300 bauda, i stop bit, dužina reči od 8 bitova, bez provere parnosti. Da bi se poveo razgo-

vor, treba startovati program, ali se isti mora nalaziti i u drugom „komodoru“. Primer je kratak pa ga je lako ukucati, ali ćemo uraditi nešto bolje — zašto ga ne bismo poslati preko veze i tako u startu demonstrirali jednu od pogodnosti koju veza pruža. „Komodor“ nije predvideo SAVE i LOAD naredbe za RS232C, pa će biti neophodno da u prijemnom računaru ipak napišemo dve programske linije:

```
10 OPEN 7.2,0,CHR$(6)
20 GET#7,AS:PRINTAS::GOTO20
Sada u prvom, u direktnom modu, otkucajte sledeće:
OPEN 30,2,0,CHR$(6)
CMD30:LIST
PRINT#30:CLOSE30
```

Na ekranu drugog računara će se pojaviti program iz prvog, ali samo na ekranu. Zato pritisnite STOP i kursorom pređite preko linija pritisajući RETURN na svakoj. Oba „komodora“ će, sada, imati isti program, pa možete početi sa razgovorom — sve što otkucate pojavice se na vašem ekranu, ali i na ekranu drugog „komodora“. Ako istovremeno budete kucali, slova će se mešati ali isto važi i pri običnom ljudskom razgovoru. Ako je vaš sagovornik nepristojan pa vam ne da da dođete do reči, već stalno kuca nešto, možete mu lepo odgovoriti pritiskom na CLR taster. Ovo će obrisati i vaš i njegov ekran, ali i ilustrovati jednu lepu osobinu interfejsa: ako je izabrana dužina reči od 8 bita, mogu biti slati svi karakteri od 0 do 255 (CLR ima kod 147). Da ste izabrali dužinu reči od 7 bita, osmi bit bi uvek bio 0, pa bi CLR taster samo pomerio kursor na početak ekrana. Izbor dužine reči ne mora da vas brine — izaberite uvek 8 kada koristite vezu dva „komodora“. Dužina od 7 bita predstavlja standard uglavnom pri komunikaciji sa većim računarima.

Razvoj programa

Datim primeru nedostaje dosta toga da bi postao pravi program za komunikaciju. Pošto je veza bila namenjena isključivo za razgovor, nije bilo potrebno obezbeđenje od gubitka znakova pri prijemu, jer je malo verovatno da možete da kucate brže od 300 bauda (približno 30 znakova u sekundi!). Za tu brzinu prenosa i bežik je dovoljno brz, ali se pri većim brzinama ili manjim razmacima očitavanja ulaznih karaktera (sa GET) može desiti da računara, jednostavno, ne stigne sve da obradi. Pošto se prijem, kao i slanje, odvija pod nemaskiranim interaptom, dakle nezavisno od trenutnog dešava-

nja u bejzik (ili mašinskom) programu, potrebno je da se za znakovi po prijemu (ili pre slanja) smestaju u neki red, tako da uvek budu na raspolaganju interapt rutini. Ovo je izvedeno tako što se u momentu otvaranja veze u vrhu memorijiskog-prostora dostupnog bejziku odvajaju dva puta po 256 bajtova (ulazni i izlazni bafer). Kada sa PRINT# pošaljete karakter na RS232, operativni sistem će ga samo proslediti u ovaj prostor i uvećati brojca znakova koji čekaju na slanje. Pri tome interapt rutina neprekidno uzima redom znakove iz ovog prostora i kontinualno obavlja slanje i smanjuje sadržaj brojača. Isto se odnosi na prijem — u interaptu se znakovi primaju sa RS232 i redom smestaju u bafer, povećavaju brojca, a GET# samo iz memoriji uzima prvi primljeni i smanjuje sadržaj brojača. Ako se desi da interapt rutina prepuni bafer jer ne stizete dovoljnom brzinom da ih odatle pokupite, ostatak će biti nepovratno izgubljen dok ne uzmete nekoliko znakova i tako otvorite prostor za nove. Još drastičniji stvar se dešava pri zatvaranju veze. Bez obzira da li je poslao sve znakove iz bafera ili ne, „komodor“ momentalno prekida komunikaciju i ostobada bafere. Očigledno je da po poznavanje stanja bafera pri ione većim brzinama postaje posebno važno. Način na koji operativni sistem odvajaju bafere za ovu namenu detaljno je opisan u dodatku „Sve komodorove rutine“ („Računari 8“).

Informacija o stanju komunikacije dobija se iz standardne bejzik promenljive STATUS. Iako je ova bejzik varijabla ista i za sve ostale periferije, njen sadržaj se za RS232 formira na drugačiji način. Osnovna razlika je u tome što pri svakom testiranju njene vrednosti dobija sadržaj 0, pa nije moguće, na primer:

```
IF (ST=0) OR (ST=8) THEN . . . .
```

jer nakon prvog logičnog izraza ST gubi originalni sadržaj, pa drugi logički izraz nema smisao. Zbog toga je u ovakvim prilikama potrebno uvođenje nove varijable kojoj će biti dodeljen sadržaj STATUS-a, a zatim njena vrednost testirana.

Svaki bit STATUS varijable ukazuje na neki tip greške u prenosu. Jedino bit 3 nema ovu namenu, već označava da je prijemni bafer prazan i postavlja se nakon naredbe GET. Ostale greške i nije potrebno posebno testirati. Dovoljno je da utvrdite da ST nije jednako 0 ili 8, što znači da je bajt pogrešno primljen i da ga, u najvećem broju slučajeva jednostavno ignorisite.

Kako to kod „komodora“ obično biva, ovakav način nije dovoljan da pokrije sve moguće situacije. To se, prvenstveno, odnosi na slučaj kada treba da završimo razgovor i zatvorimo vezu, a nemamo informaciju da li je interapt rutina uspeła da pošalje čitavu poruku. Testiranjem sistem-ske varijable ENABL (673), čiji je bit 0 uvek resetovan kada je prenos završen, može se rešiti i ovaj problem.

Poznavanje STATUS varijable i stanja bafera je dovoljno, što se „komodora“ tiče, da se napiše korektan program za komunikaciju. Ostatak zavisi od toga koji će računarski biti na drugoj strani linije.

Povezivanje sa većim računarima obično naravno samo jedan problem: konverziju ASCII kodova „komodora 64“ na pravi ASCII kod. Uobičajeno je da se „komodor kod“, kao što smo i mi u prethodnoj rečenici učinili, naziva ASCII, iako s tim

```

10 REM -----
20 REM                                KOMUNIKACIJA PREKO RS232 INTERFEJSA
30 REM -----
90 CLR:POKE56,32:CLR                                :REM REZERVISANJE MEMORIJE
100 POKES3208,0:POKES3201,0:PRINTCHR$(147):REM IZBOR OPERACIJE
110 PRINT"1. RAZGOVOR"
120 PRINT"2. PRENOS DATOTEKE"
130 PRINT"3. KRAJ RADA"
140 GETA$:IFA$="GOT100"
150 ONVAL(A$)GOTO1000,2000,170
160 GOT1000
170 END
1800 REM
1801 REM                                RAZGOVOR
1802 REM
1810 OPEN2,2,0,CHR$(6)+CHR$(0):GET#2,A$           :REM INICIJALIZACIJA VEZE
1811 PRINTCHR$(147)"RAZGOVOR":POKE204,0         :REM KURSOR VIDJIV
1820 GETA$:IFA$="GOT1100"                          :REM UCITAVANJE TASTATURE
1830 PRINT#2,A$:PRINTCHR$(5)A$                   :REM SLANJE KARAKTERA
1840 IFA$=CHR$(95)GOTO6000                        :REM PREKID VEZE (STRELICA LEVO)
1100 GET#2,B$:IFST<>0GOTO1020                    :REM PRIJEM, DA LI JE BAFER PRAZAN
1110 IFA$=CHR$(95)GOTO6000                        :REM SAGOVORNIK PREKIDA RAZGOVOR
1120 IFA$=CHR$(128)GOTO3000                      :REM SAGOVORNIK SALJE DATOTEKU
1130 PRINTCHR$(28)B$:GOTO1020                    :REM ISPIS PRIHLJENOS KARAKTERA
2000 REM
2001 REM SLANJE DATOTEKE
2002 REM
2010 GOSUB5000
2020 OPEN4,8,3,N$:AD=8192
2030 GET#4,A$:IFST<>0GOTO2060
2040 POKEAD,ASC(A$+CHR$(0)):AD=AD+1
2050 GOTO2030
2060 POKE253,AD/256:POKE252,AD-256:PEEK(253):CLOSE4
2070 OPEN2,2,0,CHR$(6)+CHR$(0):GET#2,A$
2080 PRINT#2,CHR$(128)
2090 FORF=1TO1000:GET#2,A$
2100 IFA$=CHR$(128)GOTO2120
2110 NEXTF:GOTO6010
2120 FORF=8192TOPEEK(252)+256:PEEK(253)-1
2130 PRINT#2,CHR$(PEEK(F)):NEXTF
2140 GOTO6000
3000 REM
3001 REM PRIJEM DATOTEKE
3002 REM
3010 PRINT#2,CHR$(128)
3020 PRINT"PRIJEM DATOTEKE:"
3030 AD=8192
3040 FORF=1TO1000:GET#2,A$:IFST=0GOTO3130
3050 NEXTF
3060 POKE253,AD/256:POKE252,AD-256:PEEK(253)
3070 CLOSE2:GOSUB5000
3080 OPEN4,8,3,N$+"W"
3100 FORF=8192TOPEEK(252)+256:PEEK(253)-1
3110 PRINT#4,CHR$(PEEK(F)):NEXTF
3120 CLOSE4:GOTO100
3130 POKEAD,ASC(A$+CHR$(0)):AD=AD+1
3140 PRINTA$:GOTO3040
5000 REM
5001 REM UPIS NAZIVA
5002 REM
5010 PRINTCHR$(147)"NAZIV DATOTEKE:"
5020 OPEN1,0:INPUT#1,N$:CLOSE1
5030 RETURN
6000 REM
6001 REM KRAJ KOMUNIKACIJE
6002 REM
6010 IF(PEEK(673)AND1)GOTO6010
6020 CLOSE2
6030 GOTO100
READY.

```

standardom ima malo veze. U drugom situ (velika i mala slova), ono što su u ASCII velika slova kod „komodora“ su mala i obratno. Uz to, kodovi od 96 do 127 zapravo i ne postoje, već su pomeneni na 192 do 223 — iako će PRINT CHR\$(97) dati na ekranu veliko slovo A, ako sa GET ispitujemo tastaturu pa pritisnemo A+S-

HIFT, dobićemo kod 193. Kod kontrolnih kodova je situacija komplikovanija jer postoji malo slaganja. Dovoljno je, ipak, znati da „komodorov“ DEL kod (brisanje prethodnog znaka) u pravom ASCII-u ima vrednost 8.

Objavljujemo i jednu malo bolju verziju programa za komunikaciju kojim se može

voditi razgovor korisnika dva „komodora 64“ i, ujedno, razmenjivati datoteke sa diska. Izborom opcije 1. možete voditi razgovor kao i sa prethodnim programom, ali će sada pritisak na taster „strelca levo“ bilo koga od dva korisnika prekinuti razgovor i oba vratiti na glavni meni. Prenos datoteke se odvija izborom opcije 2. onog koji šalje i 1. onog koji prima. Ostatak se, pored zadavanja naziva, odvija automatski. Obratite pažnju kako je izveden protokol za ovu vrstu prenosa i kako je iskorišćen bajt 128 (proizvoljno izabrana vrednost). Ovo nije najsigurniji metod, ali je dovoljno ilustrativan, pa ga možete proširiti i time poboljšati program. Zbog prostora i bolje preglednosti, izostavili smo standardne provere postojanja datoteke na disku itd.

Komunikacija iz mašinske

Do sada je u „Računarima“ bilo dosta reči o komunikaciji sa periferijama iz mašinskog jezika. Za RS232 važi potpuno isto što i za ostale uređaje. Standardne rutine operativnog sistema SETLFS, SETNAM, OPEN, CHKIN, CHKOUT, CHRIN, CHROUT, GETIN, CLRCHN i CLOSE imaju iste namene, pa praktično program koji je napisan za ispis, na primer na printer, sada može — uz izmenu naziva datoteke koji nije bilo potrebno definisati i promenu vrve adrese (2 umesto 4) — potpuno korektno da radi.

Za uzimanje znakova sa veze, ako su „komodori“ spojeni na način koji smo ovde pokazali, obavezna je upotreba rutine GETIN a ne CHRIN. CHRIN rutina je namenjena za komunikaciju sa hendsejkingom i očekuje postojanje signala DSR, koji se u ovoj varijanti ne koristi, pa je operativni sistem neće napuštati. Tada jedino reset računara može pomoći.

Komunikacija putem modema se odvija na potpuno identičan način kao i kod dva direktno povezana „komodora“. Uz mode-m se najčešće dobija i program koji je predviđen za ostvarivanje veze sa bilo kojim računarem, pa je u njega ugrađena sva potrebna konverzija kodova itd. Zbog ovoga ih najčešće nije moguće koristiti za razmenu programa (mada ima i takvih), ali nema razloga da u tu svrhu ne koristite program koji smo dali. Moguće je napraviti i sigurniju (mada sporiju) verziju, koja bi koristila neku od metoda za proveru ispravnosti primljenog koda. Na primer, predajni računar bi mogao pri slanju računati i kontrolni zbir i na kraju poslati i njega da bi ga prijemni računar uporedio sa onim što je dobio računajući kontrolni zbir primljenih podataka. Najsigurnija varijanta je provera bajt po bajt, ali to znači dvostruko duže vreme za prenos programa, što pri brzini od 300 bauda može preći tolerantnu granicu.

Zoran Životić

Akcije

„Galaksijina“ trakoteka

Da biste nabavili naslove iz naše Biblioteke, jednostavno popunite narudžbenicu ili je, ako ne želite da oštete svoje „Računare“, prepišite na dopisnicu i pošaljite na adresu „Galaksija“ (za Biblioteku programa), Bulevar vojvode Mišića 17, Beograd. Verujemo da će naručeni programi biti isporučeni do kraja godine.

Obzrom da će programi 1—5 moći da se nabave isključivo posredstvom našeg časopisa, svi poručioći će biti upisani u kompjutersku bazu podataka, što znači da će biti lično obavешteni o unapređenjima i novim verzijama programa (dokumentacije) i da će imati priliku da te verzije dobijaju ili besplatno ili uz minimalnu uplatu.

Ekranski editor

1. Ekranski editor je namenjen svima koji pišu bezik programe. Omogućava efikasno ispravljanje softvera uz rad sa 51 slovom u redu, dva kurzora, funkcijske tastere, prenumeraciju, zamrzavanje dela ekrana, različita skrolovanja i mnogo drugih stvari. Ovaj 100% mašinski program je potpuno kompatibilan sa raznim proširenjima za „spektrum“. Pridodata je vrlo opsežna dokumentacija. Autor Vladimir Kostić.

Hiperbežik

2. Hiperbežik je unapređena verzija pobeđnika prošlogodišnjeg „Galaksijinog“ konkursa. Učitaljiv ga, bezikju vašeg „spektruma“ dodaje tridesetak novih naredbi i, da stvar bude posebno lepa, i sami dobijate mogućnost da taj bezik dalje proširujete bez ikakve potrebe da se bavite assemblerom. Pridodata je opsežna dokumentacija radena prema standardima našeg konkursa. Autor Fischer.

Velika akcija

3. Velika akcija je prva kompleksna igra-avantura pisana potpuno na našem jeziku. Kao diverzant treba da izvedete akciju u okupiranom gradu i prebegnete na slobodnu teritoriju. Program je vrlo dopadljivo realizovan: tehnike kompresije ekrana su omogućile da u memoriju stane veliki broj slika u boji, koje bivaju prikazane kad god dođete na neko mesto. Pridodata je kratko uputstvo. Autor Aca Radovanović.

Eatinglish 1

4. Eatinglish 1 program za učenje engleskog jezika kroz igru i zabavu; raden je po principu koji su autori nazvali „All you wish“ čime je program postao „nešto kao korpa sa igračkama iz koje se vadi šta se želi i kada se želi“. Eatinglish ima preko 90 slika, rečnik od 360 reči kao i lekcije koje nastavnik engleskog mogu da modifikuju. Pridodata je uputstvo radeno prema standardima našeg konkursa. Autori S. Milekić i D. Tanaskovski.

Trodimenzionalna grafika

5. Trodimenzionalna grafika Jovana Skuljana — četrnaest kilobajta čistog mašinska — omogućuje grafičko predstavljanje elementarnih funkcija u tri dimenzije u aksionometriji i perspektivi. Program je opremljen namenskim ekran-skim editorom koji pojednostavljuje unošenje parametara i, da bi se dobilo na brzini prikaza, sopstvenim kalkulatorom.

6. Modifikacija „spektrumovog“ rom-a koja podržava rad ne samo sa eprom-diskom nego i sa rom-ovima u senci, detektuje tastaturu (naredbe se kucaju slovo po slovo) i uvodi nekoliko novih naredbi, kao što je EDIT, i modifikacija GENS-a da može da radi iz rom-a.

Servis za eprome

o Programiranje eproma sa modifikacijom „spektrumovog“ rom-a 400
o Programiranje eproma sa modifikacijom GENS-a da radi iz rom-a 400
o Programiranje oba eproma (rom+gens) 600
o Programiranje eproma sa modifikacijom assemblera PROF1 64 i disassemblera da radi iz rom-a („komodor 64“) 600
U eprom-servisu redakcija obavlja samo usluge programiranja. To znači da je zajedno sa potvrdom o uplati na adresu redakcije potrebno poslati i odgovarajuće eprome (jedan ili dva eproma 27128 „spektrum“ i jedan eprom 2764 ili 2732 za „komodor“). Programiranje će biti obavljeno najkasnije za deset dana. Ako redakcija prekoraci ovaj rok, programirane usluge programiranja, poštanskih troškova i neophodne modifikacije, a ne i samih programa. Programi su pod kopirajtom i redakcija nema ni pravo ni nameru da ih distribuira. Podrazumeva se da oni koji naručuju modifikacije već imaju osnovne verzije određenih programa.

NARUĐZBENICA

Ovim neopozivo naručujem pouzecom sledeće programe iz „Galaksijine“ biblioteke	
1. Ekranski editor	(spektrum) 800 din
2. Hiper bezik	(spektrum) 750 din
3. Velika akcija	(spektrum) 400 din
4. Eatinglish	(spektrum) 500 din
5. Trodimenzionalna graf.	(spektrum) 800 din
6. Modifikacija	spektrumovog rom-a 800 din

Takođe naručujem sledeće knjige:

5. Sve „spektrumovog“ rutine	300 din
6. Sve „komodorovog“ rutine	300 din

Ime i prezime _____

Adresa _____

Mesto _____

Potpis _____

Odgovarajući iznos cu uplatiti poštom prilikom preuzimanja programa.

matrice i nizovi

Nizovi

Numerički, ili brojni niz je skup promenljivih sa zajedničkim imenom, kao recimo: **A(1)**, **A(2)**, **A(3)**, i td. Da bi se preciziralo koliko članova neki niz sadrži, mora se obaviti njegovo dimensionisanje, korišćenjem bejzik naredbe **DIM**. Na primer, ako izvršimo **DIM A(10)**, obavestićemo računara da uvodimo niz **A** od deset članova. U memoriji će se, jednostavno, rezervirati potreban prostor za smeštanje deset promenljivih sa imenima **A(1)**... **A(10)**, pri čemu je svaka od njih potpuno ravnopravna sa bilo kojom drugom bejzik promenljivom. Uostalom, bejzik i ne poseduje naredbe za neposredan rad sa nizovima. Bilo bi divno kada bismo mogli da štamparamo sve članove niza jednom jedinom naredbom **PRINT A**. Neki jezici to mogu, a mi ćemo morati da tretiramo niz doslovno kao skup nezavisnih varijabli. Sa **PRINT A(1)**, recimo, štampani bismo samo prvi član.

Na prvi pogled, čini se da ništa posebno ne dobijamo uvođenjem niza. Jednostavnije bi bilo koristiti obične promenljive **A1**, **A2**, **A3**, itd, ako ni zbog čega drugog, onda bar zbog kraćeg zapisa. Zato uzmimo jedan konkretan primer: zamislimo da vršimo neko merenje jačine struje u **N** vremenskih trenutaka. Podatke treba ubaciti u računar i zatim naći, recimo, srednju aritmetičku vrednost struje. Najprirodnije je dimensionisati nizove **T(N)** i **I(N)**, gde je **N** ukupan broj merenja, koji moraju biti poznat unapred. Unošenje i ispisivanje podataka je, u tom slučaju, vrlo jednostavno:

```
1000 REM "INTENZITET VEKTORA"
1010 LET A=0
1020 FOR K=1 TO 3
1030 LET A=A+(K)*A(K)
1040 NEXT K
1050 LET A=A/3
1060 PRINT A
1070 STOP
```

Srednja vrednost struje takođe ne predstavlja problem:

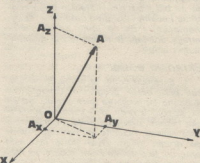
```
1100 REM "SREDNJA VREDNOST"
1110 LET I=0
1120 FOR E=1 TO N
1130 LET I=I+I(E)
1140 NEXT E
1150 LET I=I/N
1160 STOP
```

Dakle, korišćenjem nizova, programi postaju kratki i pregledni, što se nikako ne bi moglo izvesti sa varijablama **T1**, **T2**, **T3**... **T1**, **I2**, **I3**,... Nizove treba uvoditi uvek u situacijama kada radimo s većim brojem podataka iste prirode.

Vektor

Vektor je matematički objekat, koji bi se mogao definisati kao uređeni skup brojeva, npr. (3, 2, 5). Svaki od brojeva iz tog skupa predstavlja jednu komponentu u vektora. Komponenti može biti jedna, dve, tri ili više, ali mi ćemo se prvo ograničiti na

vektore sa manjim brojem komponenti, jer smo tada u stanju da ih i geometrijski predstavimo. Smatraćemo, zapravo, da je vektor neka orijentisana duž u prostoru, a da su komponente prosto projekcije te duži na ose koordinatnog sistema. Ili, ako početak vektora dovedemo u početak koordinatnog sistema, onda će krajnja tačka imati koordinate jednake komponentama tog vektora:



Sl. 1 Vektor

Treba samo imati u vidu da će se pri promeni koordinatnog sistema (recimo pri rotaciji) promeniti i komponente vektora **A**, mađ njegov položaj i veličina ostaju nepromenjeni. Ali, u fiksiranom koordinatnom sistemu, vektor **A** je potpuno i jednoznačno određen sa svoje tri komponente **A_x**, **A_y** i **A_z**. U programu ćemo, razume se, koristiti niz **A(1)**, **A(2)**, **A(3)**, dok bismo neki drugi vektor **B** prikazali sa **B(1)**, **B(2)**, **B(3)**.

Pokazaćemo sada kako se sve karakteristike jednog vektora mogu izračunati samo znajući njegove komponente. Prvo ćemo odrediti dužinu vektora, ili njegov intenzitet. Rešenje dajemo u obliku programa, što znači da sa **GO SUB 1000** dobijamo dužinu vektora **A**, ako smo unapred definisali niz **A(1)**, **A(2)**, **A(3)**.

```
3050 NEXT I
3060 GO SUB 2000: LET D=DET
3070 FOR P=1 TO N
3080 FOR Q=1 TO N
3090 FOR K=1 TO N
3100 LET G(K,P)=F(Q,R)+E(Q,R)+G(Q) AND R=P
3110 NEXT K
3120 NEXT Q
3130 GO SUB 2000: LET I(F)=DET/D
3140 NEXT P
3150 RETURN
```

Uglovi koje vektor **A** zaklapa sa osama koordinatnog sistema lako se nalaze kad je poznata dužina vektora. Sa **U(1)** označili smo ugao sa **X**-osom, sa **U(2)** ugao sa **Y**-osom, a sa **U(3)** ugao sa **Z**-osom.

```
5000 REM "SISTEM 313"
5010 DATA +3,-5,+1, + 3
5020 DATA +2,+4,+6, +38
5030 DATA +4,+7,-8, - 6
5040 LET M=3
5050 DIM F(N,N): DIM C(N,N): DIM X(N)
5060 RESTORE 5010
5070 FOR I=1 TO N
5080 FOR J=1 TO N
5090 READ F(I,J)
5100 NEXT J
5110 READ C(I)
5120 NEXT I
5130 GO SUB 3000
5140 PRINT "I=";I(1)"E=";I(2)"Z=";I(3)
5150 STOP
```

Zbir vektora **A** i **B** geometrijski se dobija njihovim nadovezivanjem, a algebarski sabiranjem odgovarajućih komponenti. Ako rezultujući vektor označimo sa **C**, onda možemo pisati program:

```
10 REM "ULAZ/IZLAZ"
20 FOR E=1 TO N
30 INPUT T(E),I(E)
40 PRINT T(E),I(E)
50 NEXT E
60 STOP
```

Vektori se mogu i množiti, i to na dva načina. Mi ćemo ovdje samo pomenuti tzv. skalarni proizvod, 'kada' se kao rezultat dobija običan broj (skalari). Po definiciji, skalarni množenje vektora vrši se tako što se pomoću dužine tih vektora i kosinus ugla među njima. Može se dokazati da se isti rezultat dobija ako se pomoću parovi odgovarajućih komponenti, a onda svi proizvodi saberu. Skalarni proizvod vektora **A** i **B** označili smo sa **W**.

```
1100 REM "UGLOVI SA OSAMA"
1110 DIM U(3)
1120 FOR K=1 TO 3
1130 LET U(K)=ACS(A(K)/A)
1140 NEXT K
1150 RETURN
```

Ugao između bilo koja dva vektora **A** i **B** lako se nalazi ako su poznate dužine vektora i odgovarajući skalarni proizvod:

$$U = ACS(W/(A*B))$$

Treba jedino obratiti pažnju na činjenicu da funkcija **ACS** daje isključivo rezultate u intervalu između nule i 180 stepena, a to ne mora uvek odgovarati našim potrebama. Međutim, problem nije tako veliki. Ako nas umesto oštrog ugla **U**, koji nam daje funkcija **ACS**, interesuje onaj drugi, tup, dobićemo ga sa **2*PI-U**.

Dimenzije

Sve što smo govorili u vezi sa vektorima i tri dimenzije, važi i za vektore u dve dimenzije — orijentisane duži u ravni. Dvodimenzioni vektori imaju samo dve komponente (projekcije na **X** i **Y** osu), a svi programi koje smo dali na vezi i u tom slučaju. Samo treba u **FOR-NEXT** petljama pisati 2 umesto 3, ili tretirati dvodimenzione vektore kao trodimenzione, pri čemu je treća komponenta nula (nema projekcije na **Z**-osu).

Broj dimenzija može, jednostavno, biti bilo koji prirodan broj: deset, sto, milion... Zamislite, recimo, skalarni proizvod dva vektora od po hiljadu dimenzija. Zamislite, uostalom, skalarni proizvod dva vektora u četiri dimenzije. Šta je to četvorodimenzioni vektor? Svakako, skup od četiri komponente, npr. (3, 7, 5, 2). Ali, šta bi to bilo geometrijski? Potrebne su nam, očigledno, četiri međusobno normalne koordinatne ose, a to je u našem trodimenzionom svetu nemoguće izvesti.

Standardni bejzik se, svakako, ne može ubrojati u jezike koji omogućuju korisniku strukturirano programiranje. Ipak, ako se nešto više pažnje posveti organizaciji podataka, može se i pomoću običnih FOR-NEXT petlji postići dobra struktura, preglednost i kratkoća programa. Pri tom mislimo pre svega, na nizove i matrice, a iz iskustva znamo da početnici uglavnom izbegavaju ovaj tip podataka, verovatno smatrajući da je to sve isuviše komplikovano. Pokušaćemo, zato, da približimo čitaocima pojam dimenzionisanih promenljivih, uz nekoliko primera njihove upotrebe u programima.

Računar se, međutim, ne bavi filozofijom, i sa podjednako malo zanimanja za problem viših dimenzija računae uspešno sve operacije sa vektorima u hiljadu dimenzija, baš kao da ih ima samo tri. Tako će, recimo, potprogram za nalaženje dužine vektora utvrditi da vektor (3, 7, 5, 2) ima dužinu 9.32738, ali ostaje van granica našeg poimanja kako i gde je postavljena ta duž.

Matrice

Kada smo rekli da se vektor može shvatiti kao uredni skup brojeva, nismo se dovoljno precizno izrazili, mada smo podrazumevali urednost u smislu niza: brojevi slede jedan iz drugog. Međutim, sasvim se lepo brojevi mogu poredati i u dve dimenzije:

$$\begin{pmatrix} 4 & -3 & 7 & 2 \\ 6 & 4 & 4 & -1 \\ 8 & 0 & -4 & 3 \end{pmatrix}$$

Sl. 2. Matrica

Ovakva šema brojeva se u matematici zove matrica. Njen geometrijski smisao, naravno, ne možemo imati, jer se pojavljuje veliki broj komponenti (u našoj šemi čak 12). Međutim, nekako se intuitivno nameće ideja da bi matrica trebalo da predstavlja uopštenje pojma vektora. Svaku vrstu i svaku kolonu matrice možemo shvatiti kao jedan vektor. Tada bi se moglo reći da se vektori sastoje od brojeva, a matrice od vektora. Ali tu je otkrili kraj naše mašte. Matrice ćemo sabirati, množiti, transponovati, invertovati, a da pri tome praktično nećemo imati predstavu o tome šta radimo. A što se tiče računara, njemu će i dalje biti svejedno, što nam savršeno odgovara, jer nas ionako sada interesuje praktična primena matrica, a ne filozofiranje na tu temu.

Kao najinteresantiji primer primene matrica, uzećemo problem rešavanja sistema linearnih algebarskih jednačina. Sa time smo se svi sreli, ako ne na drugom mestu, onda bar u srednjoj školi, mada tamo niko verovatno nije radio sa matricama.

Sistemska jednačina

Poćemo sa jednostavnim sistemom od dve jednačine sa dve nepoznate:

$$\begin{aligned} 4x + 5y &= 30 \\ 2x - 3y &= 4 \end{aligned}$$

Zadatak je, dakle, pronaći takve vrednosti za x i y , da date jednačine budu identički zadovoljene. Ima dosta načine da se pronađu te vrednosti za x i y . Ako imate iskustva

sa matematikom, onda rešenje možete naći za nekoliko sekundi, samo gledajući u jednačine: drugu pomnožite sa dva, oduzmite od prve i tako eliminišete $x \dots$. Bilo kako bilo, rešenje sistema je $x=5$ i $y=2$.

Dogovorivši se sada da sistem linearnih jednačina zapisujemo u malo neuobičajenom obliku:

$$\begin{pmatrix} 4 & 5 \\ 2 & -3 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 30 \\ 4 \end{pmatrix}$$

Slika 3. Sistem jednačina u matricnom obliku

Iskazano rečima, to znači da proizvod matrice koeficijenta sistema i vektora nepoznatih daje vektor slobodnih članova: matrica \times vektor = vektor. Istina, vektori su sada zapisani kao kolone (komponente su jedna iznad druge), ali to ne treba da nas buni. Naš cilj je da nađemo vektor (x, y) .

Prvo ćemo matrici sistema dodeliti nešto što se zove determinanta:

$$\begin{vmatrix} 4 & 5 \\ 2 & -3 \end{vmatrix}$$

Slika 4. Determinanta

Determinanta, za razliku od matrice, nije nikakva šema sa nedokučivim smislom. Jednostavno, one dve vertikalne crte sa strane označavaju operaciju koja se mora izvršiti nad brojevima u šemi, a rezultat te operacije će biti *njaobičajniji broj*. Pravilo je sledeće: pomnožiti brojeve na glavnoj dijagonali i od toga oduzeti proizvod brojeva na sporednoj dijagonali. Pri tome, glavnom dijagonalom se naziva ona koja ide iz levog gornjeg ugla ka desnom donjem uglu matrice.

$$D = \begin{vmatrix} 4 & 5 \\ 2 & -3 \end{vmatrix} = 4 \cdot (-3) - 5 \cdot 2 = -22$$

Slika 5. Rešavanje determinante

Determinanta našeg sistema jednačina iznosi $D = -22$.

Treba izračunati ovs dve determinante D_x i D_y , koje se dobijaju kada se vektor slobodnih članova stavi u determinantu sistema umesto vektora koeficijenta uz x (prva kolona), a zatim uz y (druga kolona).

$$D_x = \begin{vmatrix} 30 & 5 \\ 4 & -3 \end{vmatrix} = -110 \quad ; \quad D_y = \begin{vmatrix} 4 & 30 \\ 2 & 4 \end{vmatrix} = -44$$

Slika 6. Determinante promeljivih

Rešenje sistema se dobija po formuli:

$$\begin{aligned} x &= D_x/D = 5 \\ y &= D_y/D = 2 \end{aligned}$$

Savetujemo vam da ponovite ceo postupak za neki drugi sistem jednačina, recimo:

$$\begin{aligned} 8x - 2y &= 10 \\ 4x + 3y &= 33 \end{aligned}$$

Rešenje treba da bude $x=3$, $y=7$. Potpuno isti postupak važi i za sistem od N jednačina, sa N nepoznatih $x_1, x_2, x_3, \dots, x_N$. Opet treba izračunati determinantu sistema D i još ukupno N determinanti za nepoznate: $D_1, D_2, D_3, \dots, D_N$, koje se dobijaju tako što se vektor slobodnih članova stavlja na mesto odgovarajućih kolona u determinanti sistema. Rešenja su opet: $x_1 = D_1/D$, $x_2 = D_2/D$, itd.

```
1200 REM "ZBIR VEKTORA"
1210 DIM C(3)
1220 FOR K=1 TO 3
1230 LET C(K)=A(K)+B(K)
1240 NEXT K
1250 RETURN
```

Nismo jedino objasnili kako se izračunava determinanta sa većim brojem vrsta i kolona. Tu takođe ima više načina, a jedan je da se nekim transformacijama svi elementi determinante sa jedne strane glavne dijagonale (iznad ili ispod) dovedu na nulu. U tom slučaju je determinanta prosto jednaka proizvodu elemenata na glavnoj dijagonali. Prilikom transformiranja determinante, bilo koja vrsta se sme zameniti zbirom te iste vrste i neke druge, eventualno pomnožene proizvoljnim brojem (sabiranje vrsta se vrši kao sabiranje vektora). Isto, naravno važi i za kolone.

```
1300 REM "SKALARNI PROIZVOD"
1310 LET W=0
1320 FOR K=1 TO 3
1330 LET W=W+A(K)*B(K)
1340 NEXT K
```

Kao prilog ovom tekstu napišete potprogram za računanje determinante $D(N,N)$ metodom koju smo ukratko i opisali. Rezultat će na povratku biti smešten u varijablu DET .

Takođe dajemo i kompletan potprogram za rešavanje sistema N jednačina sa N nepoznatih. Koeficijenti sistema se na ulazu moraju naći u matrici $F(N,N)$ a slobodni članovi u nizu $G(N)$. Rešenja će biti data nizom $X(N)$. Kao konkretan primer smo rešili sistem tri jednačine:

$$\begin{aligned} 3x - 5y + z &= 3 \\ 2x + 4y + 6z &= 38 \\ 4x + 7y - 8z &= -8 \end{aligned}$$

Rešenje je: $x=3$, $y=2$, $z=4$. Napominjemo, na kraju, da rešavanje sistema linearnih jednačina nije baš tako

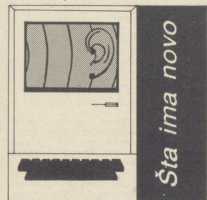
jednostavan posao, kako bi to možda moglo da izgleda na osnovu ovoga što smo govorili. Prvo, *ne mora* svaki sistem da ima rešenje. Ako se desi da determinanta sistema *D* bude nula, program koji smo dali će prijaviti grešku u aritmetičkom prekoracanju, jer se vrši deljenje nulom pri izračunavanju samih nepoznatih. Dakle, program je

```
3000 REM          SISTEM: "F(N,N)*X(N)=G(N)"
3010 FOR I=1 TO N
3020 FOR J=1 TO N
3030 LET D(I,J)=F(I,J)
3040 NEXT J
```

predviđen samo za slučaj kada *D* nije nula. Drugi problem je tačnost sa kojom računara radi. Dovoljno je da su koeficijenti sistema nejednakačnu po svojoj veličini, pa da ovim putem nikada ne dođete do rešenja.

Problema dakle ima, ali čemo ih ostaviti onima koji se bave numeričkim metodama. Nama su sasvim dovoljni, bar za sada, neki od osnovnih principa i postupaka u programiranju.

Jovan Skuljan



Šta ima novo

Hitovi u senci

Protekla godina je bila veoma uspešna za programere. U prošlom broju „Računara“ ukratko su prikazani najveći hitovi za „spektrum“, „amstrad“ i „komodor“. Medutim, mnogo dobrih programa je ostalo u senci. Evo nekoliko takvih za „spektrum“:

Mnoge starije igre su bile i dalje popularne, među njima i *Sabre Wulf*, *Knicht Lore*, *Alien 8*, *Match point*, *Match day*, *Beach Head*, *Pyamarama* i mnoge druge. Početkom i sredinom 1985. pojavio se talas igara napravljenih krajem 1984. Pomenimo samo neke: *E. C. Bill*, *Bruce Lee*, *Dark Star*, *Everyone's a Wally*, *Jet Set Willy 2*, *Sam Stoot*, *Spy vs Spy*, *Spy Hunter* i još mnoge druge.

Od septembra pa na ovamo pojavilo se na desetine izuzetno vrednih igara: *Night Shade*, *Hypersports*, *On the Run*, *The way of the Exploding fist*, *Frank B.B.*, *Herbert's Dummy Run*, *Glass...* iako je izgledalo da je to već previse, u poslednjih par meseci 1985. pojavilo se nekoliko igara koje imaju šansu da budu među najboljima u 1986. To su *Marsport*, *Impossible mission*, *International basketball*, *Popay*, *Dynapile Dan*, *Scool Daze 2*, *Beach Head 2*, *Kung fu 2*, *Scobby Doo* i još nekoliko njih.

Uslužnih programa je nešto manje, ali su kvalitetniji. *Megabasic*, *Arist*, *Beta basic 3.0*, *Designer's pencil*, *Dungeon Builder*, *Supercode 3.5* i drugi će se, zasigurno koristiti još dugo.

Sve u svemu, uspešna godina za programere na „spektrumu“.

Ivan Nožinić

46/šta ima novo

Majstorije na računaru zx spektrum

GENS iz ROM-a

Na štampanoj ploči za proširenje ROM-a ima mesta za četiri eproma, koje čemo označiti sa ROM_0, ROM_1, ROM_2 i ROM_3. Pri tome je ROM_0 rezervisan za operativni sistem sa bezik interpreterom, pošto se, pri uključenju računara, kontrola automatski prenosi na eprom u toj poziciji.

ROM_2 je predviđen za smeštanje asemblera, ali nije nikakav problem, uz malu prepravku programa koje objavljujemo, da istu ulogu dobiju ROM_1 ili ROM_3.

Preporučujemo, osim toga, da jedna pozicija (recimo ROM_3) ostane rezervisana za originalni „spektrumov“ ROM — prosto zato što se može desiti da neki komercijalni program ne podnosi novi hardver, pa je rešenje jedino u tome da se uključi originalni ROM. U tu svrhu, iz bezika samo treba izvršiti:

OUT 239,3

Pri tome će sadržaj čitave memorije ostati neizmenjen. A povratak u ROM_0 može se izvršiti sa:

OUT 239,0

Sadržaj originalnog ROM-a je najjednostavnije prekopirati u jedan prazan eprom, koji onda treba ubaciti u poziciju ROM_3. Medutim, na to mesto se sasvim lepo može staviti i sam ROM, pod uslovom da mu se na nožicu 27 direktno dovede signal MREQ, a prekine se veza koja spaja tu nožicu sa ostalim epromima.

ROM_2: kompletan DEVPAC

Kao što smo ranije već govorili, GENS3M u toku rada koristi nekoliko potprograma iz osnovnog „spektrumovog“ ROM-a. To malo komplikuje stvar oko organizacije memorijskog prostora, sve dok želimo da se potprogrami nalaze u istom ROM-u u kome se, inače, nalazi i GENS.

Istina, mesta ima dovoljno, ali se potprogrami moraju relocirati i skupiti u zajednički blok. Medutim, ako hoćemo da u isti ROM smestimo i disasemler *MONS3M*, pokazaće se da prostor od 16 kilobajta više nije dovoljan.

Daleko elegantnije rešenje bilo bi da se GENS-u omogući neposredno pozivanje potprograma iz osnovnog ROM-a. Time se ROM_2 oslobađa suvišnog tereta i u stanju je da prihvati čitav *DEVAPAC*. Doduše, sad disasemler nije prilagođen radu iz ROM-a, ali on po svojoj prirodi treba da ostane relokabilan, kako bi bilo moguće analizirati pomoću njega proizvoljne segmente memorije. Kada korisnik to bude zahtevao, *MONS* će biti iskopiran u RAM i tamo startovan.

Pitanje je, sada, kako osposobiti GENS da poziva potprograme direktno iz osnovnog ROM-a? Svakako, ne dolazi u obzir da se ispred *CALL* naredbe prosto doda *OUT* instrukcija za pozivanje osnovnog ROM-a, jer bi, odmah iza te *OUT* instrukcije, procesor počeo da komunicira sa ROM-om koji je pozvan. Ukoliko tamo, na istim adresa-

HISOFT GENS3M2 ASSEMBLER ZX SPECTRUM

Copyright (C) HISOFT 1983,4
All rights reserved

Pass 1 errors: 00

	1	*C-	
5C3D	10	ERR_SP EQU	23613
5C49	20	E_PPC EQU	23625
5C59	30	E_LINE EQU	23641
5C5D	40	CH_ADD EQU	23645
5C61	50	WORKSP EQU	23649
5C65	60	STKEND EQU	23653
5C92	70	MEMBOT EQU	23698
5C82	80	RAMTOP EQU	23730
	90		
0095	100	TOXTAB EQU	#0095
0D6B	110	CLS EQU	#0D6B
0F2C	120	ED_ROM EQU	#0F2C
0F38	130	EDLOOP EQU	#0F38
0FA9	140	EDEDIT EQU	#0FA9
107F	150	ED_ERR EQU	#107F
12A2	160	MAIN EQU	#12A2
12B4	170	ROM_M2 EQU	#12B4
1601	180	CHAN_O EQU	#1601
16B0	190	SETMIN EQU	#16B0
196E	200	L_ADDR EQU	#196E
198B	210	NXTONE EQU	#198B
19E8	220	RCLM_2 EQU	#19E8
1B17	230	LSCAN EQU	#1B17
1BEE	240	CHCK_E EQU	#1BEE
1C82	250	EPT_1 EQU	#1C82
1C8A	260	REP_C EQU	#1C8A
1E99	270	F_INT2 EQU	#1E99
1E9F	280	REP_B EQU	#1E9F
1EDA	290	REP_M EQU	#1EDA
21B9	300	IN_ASS EQU	#21B9
24FB	310	SCAN EQU	#24FB
2530	320	SYNT_2 EQU	#2530
	330		
013D	340	STR_G EQU	#013D
0181	350	WARM_G EQU	#0181
2000	360	MONS_A EQU	#2000
17C4	370	MONS_L EQU	#17C4
3AAA		ORG	#3AAA
	400		
3AAA	410	TEST_Z CP	#5C
3AAC	420	JR NZ, BACK	
3AAE	430	POP HL	
3AAF	440	POP HL	
3AB0	450	JP REP_C	
3AB3	460	BACK LD (CH_ADD), DE	
3AB7	470	RET	
3AB8	480	SETPPC CALL GETNUM	
3ABB	490	LD (E_PPC), BC	
3ABF	500	RET	
3AC0	510	TOKENS DEC HL	
3AC1	520	LD (CH_ADD), HL	
3AC4	530	JR TOKEN	
3AC6	540	FIND_Q RST #20	
3AC7	550	CP #0D	
3AC9	560	RET Z	
3ACA	570	CP #22	
3ACC	580	JR NZ, FIND_Q	
3ACE	590	TOKEN RST #20	
3ACF	600	CP #0D	
3AD1	610	RET Z	
3AD2	620	CP #22	
3AD4	630	JR Z, FIND_Q	

U tekstu „GENS iz ROM-a“, objavljenom u „Računarima 9“, pružili smo čitaocima ideju o tome kako da preprave svoj assembler i ubace ga u ROM računara. Međutim, za prepravljnje komercijalnih programa neophodno je izuzetno dobro poznavanje mašinskog jezika, a svesni smo činjenice da je za većinu korisnika daleko prihvatljivije rešenje kupovina gotovog programa, umesto disasembliranja nekoliko kilobajta čistog koda. Stoga, ove naše tekstove treba više shvatiti kao informaciju šta se sa GENs-om može uraditi, a manje kao uputstvo za samostalan rad. Uostalom, naš servis za eprone stoji na usluzi svim čitaocima. Projekat je, u protekla dva meseca, dosta usavršen. Razvijeni je softver za rad sa novim assemblerom, i sada smo u prilici da konačno objavimo šta konkretno dobijaju čitaoci koji pošalju svoje eprone u programiranje.

3AD6	640	CP	">"	3B53	1360	LD	(MEMBOT),BC	3BE9	2080	OUT	(239),A		
3AD8	650	JR	Z,CONT	3B57	1370	RET		3BEB	2090	RET			
3ADA	660	CP	"<"	3B58	1380	RCLM	BIT	0,(IY+90)	3BEC	2100	COPY_M	LD	E,C
3ADC	670	JR	Z,CONT	3B5C	1390	JP	NZ,RCLM_2	3BED	2110	LD	D,B		
3ADE	680	CALL	ALPHA	3B5F	1400	LD	B,C	3BEE	2120	LD	HL,MONS_A		
3AE1	690	JR	NC,TOKEN	3B60	1410	SPCS	LD	(HL),#20	3BF1	2130	LD	BC,MONS_L	
3AE3	700	DEC	HL	3B62	1420	INC	HL	3BF4	2140	PUSH	DE		
3AE4	710	LD	A,(HL)	3B63	1430	DJNZ	SPCS	3BF5	2150	LD	A,#02		
3AE5	720	CALL	ALPHA	3B65	1440	DEC	HL	3BF7	2160	OUT	(239),A		
3AE8	730	JR	C,TOKEN	3B66	1450	RET		3BF9	2170	LDIR			
3AEA	740	CONT	LD	DE,TOKTAB	3B67	1460	PRGM_C	RST	#10	POP	BC		
3AED	750	LD	BC,#A00	3B68	1470	LD	A,(HL)	3BFC	2190	LD	HL,16		
3AF0	760	LD	(MEMBOT),BC	3B69	1480	CP	#20	3BFF	2200	ADD	HL,BC		
3AF4	770	SEARCH	CALL	FIND	3B6B	1490	RET	NZ	3C00	2210	LD	C,L	
3AF7	780	JR	C,EXIT	3B6C	1500	INC	HL	3C01	2220	LD	B,H		
3AF9	790	CALL	COMP	3B6D	1510	RET		3C02	2230	RET_G	XOR	A	
3AFC	800	JR	C,SEARCH	3B6E	1520	L_SCAN	LD	HL,(E_LINE)	3C03	2240	OUT	(239),A	
3AFE	810	CALL	CORR	3B71	1530	SET	0,(IY+90)	3C05	2250	RET			
3B01	820	JR	SEARCH	3B75	1540	CALL	TOKENS	3C06	2260	MONS	CP	#0D	
3B03	830	EXIT	LD	BC,(MEMBOT)	3B78	1550	JP	LSCAN	3C08	2270	JR	Z,WARM_M	
3B07	840	LD	A,C	3B7B	1560	INPUT	LD	HL,(WORKSP)	3C0A	2280	CP	#3A	
3B08	850	AND	A	3B7E	1570	SET	0,(IY+90)	3C0C	2290	JR	Z,WARM_M		
3B09	860	JR	Z,TOKEN	3B82	1580	CALL	TOKENS	3C0E	2300	CALL	EXPT_1		
3B0B	870	LD	HL,(CH_ADD)	3B85	1590	JP	IN_ASS	3C11	2310	CALL	CHKC_E		
3B0E	880	DEC	HL	3B88	1600	VAL	LD	HL,(CH_ADD)	3C14	2320	CALL	F_INT2	
3B0F	890	LD	A,(HL)	3B8B	1610	PUSH	HL	3C17	2330	COLD_M	CALL	COPY_M	
3B10	900	INC	HL	3B8C	1620	RES	0,(IY+90)	3C1A	2340	PUSH	BC		
3B11	910	CP	"*"	3B90	1630	CALL	TOKENS	3C1B	2350	RET			
3B13	920	JR	NZ,PUT_TK	3B93	1640	POP	HL	3C1C	2360	WARM_M	CALL	SYNT_Z	
3B15	930	LD	A,B	3B94	1650	LD	(CH_ADD),HL	3C1F	2370	RET	Z		
3B16	940	CP	#CE	3B97	1660	JP	SCAN	3C20	2380	LD	HL,(STKEND)		
3B18	950	JR	NC,TOKEN	3B9A	1670	GENS	CP	"#"	3C23	2390	LP_M1	LD	A,H
3B1A	960	PUT_TK	LD	(HL),B	3B9C	1680	JR	Z,COLD_G	3C24	2400	OR	L	
3B1B	970	LD	B,#00	3B9E	1690	CALL	CHKC_E	3C25	2410	JR	Z,T_COLD		
3B1D	980	INC	HL	3BA1	1700	LD	HL,27001	3C27	2420	LD	A,(HL)		
3B1E	990	DEC	BC	3BA4	1710	LD	A,#ED	3C28	2430	INC	HL		
3B1F	1000	CALL	RCLM	3BA6	1720	LD	B,#10	3C29	2440	CP	#EF		
3B22	1010	JR	TOKEN	3BA8	1730	LP_G1	CP	(HL)	3C2B	2450	JR	NZ,LP_M1	
3B24	1020	FIND	LD	HL,(CH_ADD)	3BA9	1740	JR	NZ,COLDG1	3C2D	2460	LD	B,#0F	
3B27	1030	MOVE	LD	A,(DE)	3BAB	1750	INC	HL	3C2F	2470	LP_M2	LD	A,(HL)
3B28	1040	INC	DE	3BAC	1760	DJNZ	LP_G1	3C30	2480	CP	#EF		
3B29	1050	BIT	7,A	3BAE	1770	LD	BC,WARM_G	3C32	2490	JR	NZ,LP_M1		
3B2B	1060	JR	Z,MOVE	3BB1	1780	JR	RUN_G	3C34	2500	INC	HL		
3B2D	1070	INC	B	3BB3	1790	COLD_G	RST	#20	3C35	2510	DJNZ	LP_M2	
3B2E	1080	SCF		3BB4	1800	CALL	CHKC_E	3C37	2520	LD	BC,29		
3B2F	1090	RET	Z	3BB7	1810	COLDG1	LD	DE,27000	3C3A	2530	ADD	HL,BC	
3B30	1100	LD	A,(DE)	3BBA	1820	LD	HL,(STKEND)	3C3B	2540	LD	C,L		
3B31	1110	CP	(HL)	3BBD	1830	LD	BC,50	3C3C	2550	LD	B,H		
3B32	1120	RET	Z	3BC0	1840	ADD	HL,BC	3C3D	2560	PUSH	BC		
3B33	1130	JR	MOVE	3BC1	1850	LD	HL,DE	3C3E	2570	RET			
3B35	1140	COMP	LD	C,#01	3BC3	1860	JP	NC,REP_M	3C3F	2580	T_COLD	LD	BC,58000
3B37	1150	MATCH	INC	DE	3BC6	1870	EX	DE,HL	3C42	2590	JR	COLD_M	
3B38	1160	INC	HL	3BC7	1880	LD	(RAMTOP),HL	3C44	2600	GETNUM	CALL	F_INT2	
3B39	1170	INC	C	3BCA	1890	POP	DE	3C47	2610	LD	HL,9999		
3B3A	1180	LD	A,(DE)	3BCB	1900	POP	BC	3C4A	2620	XOR	A		
3B3B	1190	RES	7,A	3BCC	1910	LD	(HL),#3E	3C4B	2630	SBC	HL,BC		
3B3D	1200	CP	(HL)	3BCE	1920	DEC	HL	3C4D	2640	JP	C,REP_B		
3B3E	1210	SCF		3BCF	1930	LD	SP,HL	3C50	2650	RET			
3B3F	1220	RET	NZ	3BD0	1940	PUSH	BC	3C51	2660	EDIT	CALL	GETNUM	
3B40	1230	LD	A,(DE)	3BD1	1950	LD	(ERR_SP),SP	3C54	2670	LD	(E_PPC),BC		
3B41	1240	AND	#80	3BD5	1960	PUSH	DE	3C58	2680	POP	HL		
3B43	1250	JR	Z,MATCH	3BD6	1970	INC	HL	3C59	2690	POP	HL		
3B45	1260	INC	HL	3BD7	1980	INC	HL	3C5A	2700	LD	(IY+49),#02		
3B46	1270	LD	A,(HL)	3BD8	1990	LD	B,#10	3C5E	2710	CALL	SETMIN		
3B47	1280	ALPHA	CP	#41	3BDA	2000	LP_G2	LD	(HL),#ED	3C61	2720	XOR	A
3B49	1290	CCF		3BDC	2010	INC	HL	3C62	2730	CALL	CHAN_0		
3B4A	1300	RET	NC	3BDD	2020	DJNZ	LP_G2	3C65	2740	LD	HL,ROM_M2		
3B4B	1310	CP	#5B	3BDF	2030	LD	BC,STRT_G	3C68	2750	PUSH	HL		
3B4D	1320	RET		3BE2	2040	RUN_G	LD	HL,RET_G	3C69	2760	LD	HL,(ERR_SP)	
3B4E	1330	CORR	LD	A,C	3BE5	2050	PUSH	HL	3C6C	2770	LD	HL	
3B4F	1340	CP	(IY+88)	3BE6	2060	PUSH	BC	3C6D	2780	LD	HL,ED_ERR		
3B52	1350	RET	C	3BE7	2070	LD	A,#02	3C70	2790	PUSH	HL		

```

3C71 2800 LD (ERR_SP),SP
3C75 2810 LD HL,EDLOOP
3C78 2820 PUSH HL
3C79 2830 JP EDEDIT
3C7C 2840 DELETE CALL GETNUM
3C7F 2850 PUSH BC
3C80 2860 CALL SETPPC
3C83 2870 POP DE
3C84 2880 LD L,E
3C85 2890 LD H,D
3C86 2900 AND A
3C87 2910 SBC HL,BC
3C89 2920 JP C,REP_B
3C8C 2930 PUSH DE
3C8D 2940 LD L,C
3C8E 2950 LD H,B
3C8F 2960 CALL L_ADDR
3C92 2970 EX (SP),HL
3C93 2980 CALL L_ADDR
3C96 2990 LD E,L
3C97 3000 LD D,H
3C98 3010 CALL Z,NXTONE
3C9B 3020 POP HL
3C9C 3030 AND A
3C9D 3040 EX DE,HL
3C9E 3050 SBC HL,DE
3CA0 3060 LD C,L
3CA1 3070 LD B,H
3CA2 3080 EX DE,HL
3CA3 3090 CALL NZ,RCLM_2
3CA6 3100 POP HL
3CA7 3110 POP HL
3CA8 3120 JP MAIN
3CA9 3130 CAPS SET 3,(IY+48)
3CAF 3140 JP CLS
3CB2 3150 SET_DE LD DE,(CH_ADD)
3CB6 3160 LD C,A
3CB7 3170 RST #20
3CB8 3180 LD A,C
3CB9 3190 RET
3CBA 3200 LET CALL TEST_Z
3CBD 3210 NOP
3CBE 3220 LD A,#23
3CC0 3230 RET
3CC1 3240 SMALL RST #28
3CC2 3250 DEFB #02
3CC3 3260 DEFB #E2
3CC4 3270 DEFB #38
3CC5 3280 RET
3CC6 3290 AD_END INC HL
3CC7 3300 LD (HL),A
3CC8 3310 INC HL
3CC9 3320 LD (HL),E
3CCA 3330 INC HL
3CCB 3340 LD (HL),D
3CCC 3350 INC HL
3CCD 3360 LD (HL),0
3CCE 3370 DEC HL
3CDF 3380 DEC HL
3CD0 3390 DEC HL
3CD1 3390 DEC HL
3CD2 3400 SUB #FF
3CD4 3410 OR E
3CD5 3420 OR D
3CD6 3430 POP DE
3CD7 3440 RET NZ
3CD8 3450 LD (HL),#91
3CDA 3460 INC HL
3CDB 3470 LD (HL),#80
3CDD 3480 DEC HL
3CDE 3490 RET
3CDF 3500 MSGADD DEFB #80
3CE0 3510 DEFM "ZX "
3CE3 3520 DEFM "Spectrum "
3CE6 3530 DEFM "Extended "
3CF5 3540 DEFM "BASIC Read"
3CF7 3550 DEFB "y"+#80

```

Pass 2 errors: 00

Table used: 998 from 2000

48/majstorije na računaru

ma, ne postoji nastavak započetog programa, doći će, logično, do kraha.

Najbezbednije je, zato, obavljati promenu ROM-ova pomoću programa koji radi iz RAM-a. Ako, na primer, *GENS* treba da pozove potprogram za očitavanje tastature, on neće izvršiti *CALL&02BF*, već, recimo, *CALL KB_SUB*, gde je *KB_SUB* neki potprogram u RAM-u, čija je uloga samo da prozove *ROM_0*, obavi *CALL&02BF*, ponovo prebaci *ROM_2* i vrati se nazad u *GENS*:

```

KB_SUB SUB A
      OUT (239),A
      CALL&02BF
      LD A,2
      OUT (239),A
      RET

```

Isti princip se može primeniti na bilo koji program. Jedino treba voditi računa o tome da neki potprogrami imaju kao ulazni ili izlazni podatak sadržaj akumulatora, koji, u tom slučaju, treba da ostane sačuvan dok se izvršava prebacivanje ROM-ova. Na primer, *ROM_0* bi trebalo prozivati sa:

```

      PUSH AF
      SUB A
      OUT (239),A
      POP AF

```

GENS3M dosta često koristi naredbu *RST&10* za štampanje ASCII znakova. To ne treba menjati, ali se zato na adresi &0010 u istom ROM-u mora naći naredba *JP PR_SUB*, gde je *PR_SUB* odgovarajuća rutina u RAM-u, koja poziva *PRINT_A_2* iz osnovnog ROM-a, na već opisan način.

Kompletan blok za pozivanje potprograma iz osnovnog ROM-a izgledao bi ovako:

```

ROM_0 PUSH AF
      SUB A
      OUT (239),A
      POP AF
      RET
ROM_2 LD A,2
      OUT (239),A
      POP AF
      RET
CH_SUB CALL ROM_0
      CALL&1601
      JR ROM_2
PR_SUB CALL ROM_0
      CALL&15F2
      JR ROM_2
KB_SUB CALL ROM_0
      CALL&02BF
      JR ROM_2
BP_SUB CALL ROM_0
      CALL&03B5
      JR ROM_2
SA_SUB CALL ROM_0
      CALL&04C2
      JR ROM_2
LD_SUB CALI ROM_0
      CALL&0556
      JR ROM_2

```

Smatramo da posebna objašnjenja ovde nisu potrebna. Potprogrami *ROM_0* i *ROM_2* prozivaju odgovarajuće ROM-ove, a *CH_SUB*, *PR_SUB*, *KB_SUB*, *BP_SUB*, *SA_SUB*, i *LD_SUB* su adrese koje *GENS* treba da poziva umesto *CHAN_OPEN*, *PRINT_A_2*, *KEYBOARD*, *BEEPER*, *SA_BYTES* i *LD_BYTES*. Na samom početku svog rada, *GENS* će prvo formirati ovaj blok potprograma u RAM-u, i tek onda nastaviti sa uobičajenim poslovima.

Da je, nekim slučajem, broj potprograma koje treba zvati iz osnovnog ROM-a bio mnogo veći, rešenje koje smo upravo opisali ne bi se baš moglo svrstati u najracionalnija. Svaki novi potprogram tu zahteva dodatnih osam bajtova i tabela polako raste kao pantiflora...

Bilo bi sasvim moguće, umesto toga, napraviti jedan jedini program koji bi mogao da poziva bilo koju rutinu iz osnovnog ROM-a. Jedino se adresa rutine mora dostaviti tom našem univerzalnom programu u vidu ulaznog parametra. Na primer:

```

CALL univerzalni program
DEFW adresa rutine

```

To, međutim, znači da sada za svaki potprogram treba trošiti pet bajtova, umesto tri, koliko zahteva uobičajen poziv naredbom *CALL*.

Idealno rešenje sastoji se u upotrebi neke od *RST* instrukcija, recimo *RST&30*, koja se retko koristi (*GENS* je ne koristi doslovno nikad). Time se poziv bilo kojeg potprograma može izvršiti sa samo tri bajta:

```

RST&30
DEFW adresa rutine

```

Podrazumeva se, naravno, da na adresi &0030 stoji naredba *JP univerzalni program*. Ostaje još da se vidi kako će „univerzalni program“ prikupiti adresu rutine koja se poziva, ali o tome ćemo, možda, nekom drugom prilikom.

ROM_0: „novi spektrum“

Osnovni ROM je pretrpeo veći broj izmena, kako bi se prilagodio novom hardveru i radu sa novim asemblerom. Osim toga, smatrali smo da, o istom trošku, treba ispraviti i sve poznate bagove originalnog „spektrumovog“ ROM-a. Uvedene su, dalje, četiri nove naredbe (umesto postojećih za mikrodrajv), a sve službene reči bezikva ukucavaju se sa tastature slovo po slovo (ne postoji više kursor „K“).

Podrazumevaju se i sve izmene objavljene u tekstu „ROM od sedam milja“, ali ćemo ovde navesti i njih, radi potpunosti.

Izmena 1.

Pripremanje prostora u RAM-u za *ROM_OS*, &1235 LD HL,&5E53

Izmena 2

Rutina za inicijalizaciju, &04AA CALL&0DEF
LD A, &C3
LD (&5CB6),A
LD HL,&5CC0
LD (&5CB7),HL
LD A,&C9
LD (&5CB9),A
CALL&386E
RET

Izmena 3.

Poziv rutine za inicijalizaciju, &128B CALL&04AA

Izmena 4.

Obrade grešaka.
a) &0008 CALL&5C86
LD HL(&5C5D)
JR &005F
b) &005F LD (&5C5F),HL
JR &0053

Izmena 5.
Postavljanje velikih slova (CAPS LOCK).
&1292 CALL&3CAB

Izmena 6.
Aritmetička naredba bez LET.
a) &1B41 CALL&3CB2
b) &1B46 CALL C,&3CBA

Izmena 7.
Štampanje programskog kursora.
&187A CALL NZ,&3B67

Izmena 8.
Štampanje služenih reči („tokena“).
a) &0C17 JR &0C22
b) &0C38 RET

Izmena 9.
Kursor „K“ se nikad ne štampa.
&18FC NOP
NOP

Izmena 10.
Kursor „K“ se ignoriše pri očitavanju tasta-
tura.
&0358 NOP
NOP

Izmena 11.
Tokenizacija programske linije.
&12B4 CALL&3B6E

Izmena 12.
Tokenizacija INPUT linije.
&2159 CALL&3B7B

Izmena 13.
Tokenizacija argumenta funkcije VAL.
&35FD CALL&3B8B

Izmena 14.
Nove službene reči.
&011F GENS
MONS
EDIT
DELETE

Izmena 15.
Parametri novih naredbi.
a) &1B06 DEF&05
DEF&3C06; MONS
b) &1B0A DEF&06
DEF&00
DEF&3C51; EDIT
c) &1B10 DEF&08
DEF&00
DEF&3C7C; DELETE
d) &1B14 DEF&05
DEF&3B9A; GENS

Izmena 16.
Operacija poredjenja i vrednovanja.
a) &353B RRCA
b) &35E2 RRCA

Izmena 17.
Štampanje realnih brojeva.
&2E24 CALL&3CC1

Izmena 18.
Funkcija INT.
&3223 JR &323F

Izmena 19.
Sabiranje celih brojeva.
&3032 JP &3CC6

Izmena 20.
Operacija deljenja.
&31FF JR Z,&31DB

Izmena 21.
Nova sistemska poruka.
&1296 LD DE,&3CDF

Izmena 22.
Operativni sistem ROM_OS, počev od adre-
se &386E.

Izmena 23.
Nove naredbe i pomoćni potprogrami, po-
čev od adrese &3AAA.

Nova poruka

Kada se računar uključio, na ekranu se pojavljuje sistemska poruka „ZX Spectrum Extended BASIC Ready“. Kursor je automatski postavljen na „C“ i kucanje programa se mora vršiti slovo po slovo. Tek po pritisku tastera ENTER, programska linija će biti prevedena u standardni format, dok svaka slušena reč ima svoj jedinstveni kod.

Prilikom listanja programa, linije se pojavljuju isključivo onako kako smo ih ukucali, tj. bez umetnutih praznih polja ispred ili iza služenih reči, a programski kursor će se štampati ili između linijskog broja i prve naredbe, ili, ako je prvi znak u liniji prazno polje, štampanje kursora će se obaviti umesto tog praznog polja. Da bi sve ovo bilo jasnije, najbolje je ukucati neki bezik program pod kontrolom originalnog „spektrumnog“ ROM-a, a onda preći u ROM-0 i pogledati listanj.

Osim ovih izmena vezanih za sam editor, ispravljaju je niz bagova, o kojima je već toliko puta bilo reči i na stranicama „Računara“. Funkcije VAL, VALS i operacije poredjenja su sada ravnopravne sa ostalim operacijama kalkulatora, tako da se mogu slobodno pozvati iz iliste koja prati instrukciju RST&28 (Izmena 16.).

Izbegnuti su svi nesporazumi oko broja -65536 (Izmena 18 i 19).

Vrlo grub bag (zaboravljena nula na računskom steku) u okviru rutine za štampanje malih brojeva sada je otklonjen (Izmena 17) i „spektrum“ konačno ume da izračuna izraz tipa „1/2“ + STR&05.

Poseban „specijalitet“ novog ROM-a predstavlja mogućnost kucanja aritmetičkih naredbi bez obavezne slušene reči LET. Recimo, naredba A=2*PI će se izvršiti apsolutno identično naredbi LET A=2*PI, što je, inače, uobičajeno na svim računarima koji koriste bezik.

Najzad, korigovana je i omaška u programu za deljenje (Izmena 20), a umesto naredbi za mikrodrajv (CAT, FORMAT, MOVE i ERASE) uvedene su četiri nove: GENS, MONS, EDIT i DELETE, sa kodovima: 207, 208, 209 i 210. Opisaćemo dejstvo svake od njih:

GENS

Naredba GENS poziva assembler GENS3M. RAMTOP će pri tome biti postavljen na 27000, a iza te adrese će se formirati radni prostor za potrebe assemblera.

Ukoliko je bezik blok prevelik, tako da RAMTOP nije moguće spustiti tako nisko, imaćemo grešku „M RAMTOP no good“. Moraćemo tada da sami napravimo mesta primenom CLEAR, DELETE ili NEW naredbe.

Sve ovo važi samo u slučaju kada se assembler poziva prvi put posle uključanja računara. Međutim, ako je program već bio jednom startovan, naredba GENS će obaviti običan vruć start.

Kako to naredba GENS može znati da li je assembler bio ili nije bio startovan pre toga? Vrlo jednostavno: prilikom inicijalizacije programa, odmah iznad RAMTOP-a se

postavlja jedan blok od šesnaest bajtova, koji su svi jednaki &ED. Uloga tog bloka je samo da služi naredbi GENS kao znak raspoznavanja: ako je blok prisutan, treba obaviti vruć start, a ako ga nema, obavice se hladan start sa inicijalizacijom.

Ukoliko, bez obzira na bilo šta, želimo izričito hladan start, dodaćemo naredbi GENS znak „*“ (CHRS 35): GENS *

Povratak iz assemblera u bezik (naredbom B u okviru GENS editora) vrši se na uobičajenu adresu za povratak iz svih bezik naredbi (&1B76, STMT_RET), što znači da je naredba GENS potpuno ravnopravna sa bilo kojom drugom.

MONS

Naredba MONS poziva disassembler MONS3M. Ako se program već nalazi u memoriji iznad bezik bloka (prepoznaje se po bloku od šesnaest bajtova jednakih &EF), biće obavljen vruć start. U protivnom, MONS3M se kopira na adresu 58000 i tamo startuje.

Ukoliko želimo da startujemo disassembler na proizvoljnoj adresi n, izvršićemo MONS n Parametar n biće zaokružen na najbliži ceo broj, ali ako je van opsega (0 — 65535), imaćemo grešku „B Integer out of range“, bez ikakvih promena u sadržaju memorije.

Povratak iz disassemblera vrši se neposredno u glavnu izvršnu petlju bezik editora, što znači da ni jedna naredba iz MONS neće biti otkrivena.

EDIT

Naredba EDIT n postavlja programski kursor na programsku liniju n i zatim preslikava tu liniju u prostor za editovanje, na potpuno isti način kao da je bio pritisnut komandni taster EDIT (SHIFT I). Parametar n se zaokružuje na najbliži ceo broj. Ako je van opsega (1 — 9999) prijavljuje se greška „B Integer out of range“.

Slično naredbi MONS, povratak iz naredbe EDIT vrši se takođe u glavnu izvršnu petlju bezik editora.

DELETE

Naredba DELETE m, n izbacuje iz listin-
ga blok programskih linija počev od linije m, pa zaključno sa linijom n. Ako linija m ne postoji, podrazumeva se prva sledeca. Ako linija n ne postoji, podrazumeva se prethodna.

Parametri m i n se zaokružuju na najbliže cele brojeve, a ako je bilo koji od njih van opsega (1 — 9999), ili je m veće od n, imaćemo grešku „B Integer out of range“.

Povratak se i ovde vrši direktno u bezik editor.

Jovan Skuljan

lagranžovi polinomi

Numerički metodi (2)

U prethodnom nastavku „Numerički metoda“ uveden je zadatak interpolacije u kome na osnovu poznatih

$$x_0, x_1, x_2, \dots, x_n \quad (1)$$

$$f(x_0) = y_0, f(x_1) = y_1, \dots, f(x_n) = y_n \quad (2)$$

najpre treba odrediti neku funkciju $F(x)$ koja zadovoljava uslov

$$f(x_0) = F(x_0), \dots, f(x_n) = F(x_n) \quad (3)$$

a zatim traženu nepoznatu vrednost $f(x)$, za zadato x , izračunati kao $F(x)$. Opisan je metod i dat program lineare interpolacije, a uveden opšti pristup polinomijalnoj interpolaciji.

U ovom nastavku opisaćemo Lagranžov (Lagrange) metod i dati program koji ga realizuje.

Interpolacioni polinom

Određivanje interpolacionog polinoma, koji će se koristiti za interpolaciju nepoznatih vrednosti $f(x)$, može se pojednostaviti ako se on najpre zapiše u posebnoj, pogodnoj formi. Cilj je neposredno dobiti vrednosti koeficijenta polinoma. Umesto da se polinom traži u standardnom obliku:

$$F(x) = a_0 + a_1 x + a_2 x^2 + \dots + a_n x^n \quad (4)$$

zapisanimo ga u (Lagranžovom) obliku: $(x-x_1)(x-x_2)\dots(x-x_n)$

$$F(x) = c_0 \frac{(x-x_1)(x-x_2)\dots(x-x_n)}{(x-x_0)(x-x_1)\dots(x-x_{n-1})} + c_1 \frac{(x-x_0)(x-x_2)\dots(x-x_n)}{(x-x_0)(x-x_1)(x-x_2)\dots(x-x_{n-1})} + \dots + c_n \frac{(x-x_0)(x-x_1)\dots(x-x_{n-1})}{(x-x_0)(x-x_1)\dots(x-x_{n-1})} \quad (5)$$

Lako je pokazati da je ovo zaita polinom stepena n . Broilac svake od sabiraka je polinom stepena n , imeniti su konstante, pa ceo izraz $F(x)$ kao zbir predstavlja polinom n -og stepena.

Postavimo sada uslov da polinom (5) prolazi kroz date tačke, tj. uslov (3). Ovaj uslov se svodi na: $f(x_0) = c_0$

$$f(x_0) = c_0$$

$$f(x_1) = c_1$$

$$\vdots$$

$$f(x_n) = c_n$$

$$(6)$$

pa su time koeficijenti Lagranžovog polinoma direktno određeni. Kada nađene vrednosti zamenimo u (5) dobija se definitivan oblik Lagranžovog interpolacionog polinoma:

$$F(x) = f(x_0) \frac{(x-x_1)(x-x_2)\dots(x-x_n)}{(x-x_0)(x-x_1)\dots(x-x_{n-1})} + \dots + f(x_n) \frac{(x-x_0)(x-x_1)\dots(x-x_{n-1})}{(x-x_0)(x-x_1)\dots(x-x_{n-1})} \quad (7)$$

Radi ilustracije, na osnovu izvedene formule (7), formirajmo Lagranžov interpolacioni polinom koristeći sledeće vrednosti nezavisno promenljive x i odgovarajuće vrednosti funkcije $f(x)$:

$$x_0 = -1, f(x_0) = 1 \\ x_1 = 0, f(x_1) = 1 \\ x_2 = 1, f(x_2) = 1 \\ x_3 = 2, f(x_3) = -5 \\ \dots + f(x_4) = \frac{(x-0)(x-1)(x-2)}{(x-1)(-1)(-1)(-2)} + \frac{(x+1)(x-1)(x-2)}{(x+1)(0)(-1)(-2)} + \frac{(x+1)(x-0)(x-2)}{(x+1)(1)(-1)(-2)} + \frac{(x+1)(x-0)(x-1)}{(x+1)(1)(-1)(-2)} = -x^2 + x + 1 \\ R(x) = f(x) + F(x) \quad (8)$$

Procena greške

Izneoćemo sada formulu na osnovu koje se može proceniti greška koja se čini kada se tražena vrednost $f(x)$ izračunava kao $F(x)$. Ova greška se može nazvati greška interpolacione metode, radi toga

što je sadržana u samoj metodi, i ne treba je mešati sa greškom koja nastaje u računaru (ako se interpolirana vrednost određuje pomoću programa) radi ograničenja sa brojevima koji imaju ograničenu tačnost. Obeležimo sa $R_n(x)$ razliku $R_n(x) = f(x) - F(x)$ razliku. (8) Ova razlika je jednaka nuli u svim tačkama

$$x_0, x_1, \dots, x_n \quad (9)$$

a treba proceniti njenu vrednost u ostalim tačkama.

Ne ulazeci u detalje izvođenja (koje se bazira na konstrukciji specijalne funkcije i uzastopnoj promeni Rolove teoreme) navodimo formulu za grešku koju se dobija u obliku:

$$R_n(x) = \frac{(x-x_0)(x-x_1)\dots(x-x_n)}{(n+1)!} f^{(n+1)}(\xi) \quad (10)$$

gde je ξ neki tačka iz intervala

$$[x_0, x_n] \quad (11)$$

sve pod pretpostavkom da su tačke (9) uređene po rastućem redosledu, a interpolaciona tačka x leži u intervalu datom sa (11).

Prokomentarišimo formulu za grešku.

Odmah se vidi da greška zavisi od x , tj. od toga gde se u intervalu (11) nalazi tačka za koju treba interpolirati.

Pored ovoga, kao u formuli za grešku figurise izvod reda $n+1$ funkcije f , to je jasno da greška zavisi od „tipa“ funkcije f , ili bolje rečeno od toga koliko je f pogodna za aproksimaciju polinomom. Kako polinom ne može da raste ili opada veoma brzo, to se ne može očekivati velika tačnost interpolacije za funkcije koje rastu ili opadaju mnogo brže, kao što je recimo slučaj u blizini tačaka prekida. Nasuprot tome, kod „sporih“ funkcija, ili funkcija sa visokom diferencijabilnošću, mogu se očekivati izvrsni rezultati.

Kada se ne poznaju izvod funkcije f , onda se formula za grešku ne može primeniti. Tada se mogu izvršiti neke procene greške na osnovu tzv. konačnih razlika, ali o tome će biti reči u narednom nastavku. Dodajmo još i to da se o položaju tačke ξ ne može ništa reći osim da pripada intervalu (11). Radi toga se formula za grešku često zamenjuje sa

$$|R(x)| \leq \frac{(x-x_0)(x-x_1)\dots(x-x_n)}{(n+1)!} M_{n+1} \quad (12)$$

gde je M_{n+1} označen maksimum $n+1$ -og izvoda na intervalu (11). Ovaj izraz za gornju granicu greške podrazumeva ne samo postojanje

$n+1$ -og izvoda funkcije f , već i njegovu ograničenost.

U prethodnom nastavku je izostavljena formula za grešku u slučaju lineare interpolacije. Nju je sada moguće neposredno dobiti kao specijalan slučaj formule (10). Dovoljno je, u tom cilju, primetiti da se linearna interpolacija može posmatrati kao polinomijalna interpolacija sa polinomom oblika:

$$F(x) = a_0 + a_1 x \quad (13)$$

koji zadovoljava uslove

$$F(x_0) = f(x_0), F(x_1) = f(x_1)$$

bilo koji drugi podinterval $[x_0, x_1]$. U tom slučaju formula (10) postaje:

$$R_1(x) = \frac{(x-x_0)(x-x_1)}{2} f''(\xi) \quad (14)$$

Navodimo bezik program kojim se realizuje Lagranžova interpolacija. Na početku se u dijalogu primaju ulazni podaci (vrednosti (1), (2) i interpolacioni argument), omogućava njihova eventualna korekcija nakon unošenja, a zatim izdaje interpolirane vrednosti. Na kraju se odgovorom na pitanje „Nova interpolacija ili kraj?“ program može iskoristiti za računanje nove interpolirane vrednosti na osnovu istih podataka koji se ne moraju ponovo unositi.

Ilustrirajmo jednim primerom rezultate programa i upotrebu formule za grešku. Neka treba interpolirati funkciju $\ln(x)$ za $x=1$ na osnovu sledećih datih vrednosti:

$$x_0 = 0.8 \quad \ln(x_0) = -0.223143 \\ x_1 = 0.9 \quad \ln(x_1) = -0.105360 \\ x_2 = 1.1 \quad \ln(x_2) = 0.095310 \\ x_3 = 1.2 \quad \ln(x_3) = 0.182321$$

Za $x=1$, na osnovu ovih ulaznih podataka, program daje interpoliranu vrednost $y = \ln(1) = 0.000103$. Kako je tačna vrednost $\ln(1) = 0$, to je greška učinjena Lagranžovom interpolacijom: greška = $0 - 0.000103 = -0.000103$. Procenimo sada grešku koristeći formulu (10). U ovom slučaju $n=3$, a potrebni četvrti izvod funkcije f se određuje kao:

$$f'(x) = \frac{1}{x} \\ \dots \\ f^{(4)}(x) = -\frac{1}{x^2}$$

Mnogi konkretni problemi koji se rešavaju na računarima imaju u svojoj osnovi zadatke numeričke matematike. Numerički metodi poseduju veliku opštost i primenljivost, ali kako njihovim razumevanje i programiranje nije uvek sasvim jednostavno, to im upotreba, odnosno popularnost među vlasnicima mikračunara, nije na nivou koji zaslužuju.

Ova serija napisa ima za cilj da jednostavno, i samo u najvažnijim elementima, prikaže nekoliko metoda kojima se rešavaju neki osnovni zadaci numeričke matematike, kao i da ponudi programe koji su bazirani na tim metodama. Programi su radeni na bezik jeziku za „komodor 64“, ali se jednostavno mogu preneti i na druge mikračunare.

```

100 REN *****
105 REN *
110 REN * LAGRANĐEOVA INTERPOLACIJA *
115 REN *
120 REN *****
125 REN * I-ULAZNA VREDNOST ARGUMENTA * 345 INPUT "UNESITE ARGUMENT INTERPOLACIJE";X
130 REN * INTERPOLACIJE * 350 PRINT
135 REN * A-NIZ VREDNOSTI NEZAVISNO * 355 PRINT
140 REN * PROMENLJIVE (ULAZ) * 360 PRINT
145 REN * V-ULAZNI NIZ VREDNOSTI FUNKCIJE * 365 PRINT"***** UNOS PODATAKA ZAVRŠEN *****"
150 REN * D-BROJ DATIH TACA (ULAZ) * 370 REN
155 REN * Y-INTERPOLIRANA VREDNOST FUNKCIJE * 375 REN *****
160 REN * (IZLAZ) * 380 REN * VREDNOSTI LAGRANĐEOVOPOLINOMA *
165 REN *
170 REN *****
175 PRINT CHR$(147) 390 Y=0
180 PRINT"***** 400 P=(I):REM POCETAK CIKLUSA ZA RACUNANJE ZBRA
185 PRINT" * PROGRAM ZA INTERPOLACIJU FUNKCIJE * 405 J=1
190 PRINT" * JEDNE NEZAVISNO PROMENLJIVE * 410 IF J=1 THEN 420:REM CIKLUS ZA PROIZVOD
195 PRINT" * 415 P=P*(I-A(J))/(A(I)-A(J))
200 PRINT" * LAGRANĐEOVA INTERPOLACIJA * 420 J=J+1
205 PRINT" ***** 425 IF J<=D THEN 410
210 NI=5 430 Y=Y+P
215 SOBUSD30:REN IZAVANJE PRAZNIH REDOVA 435 I=I+1
220 INPUT" ( PRITISNITE RETURN DA NASTAVITE I";PR 440 IF I<=D THEN 400
225 PRINT CHR$(147) 470 SOBUD 565:REN IZAVANJE VREDNOSTI POLINOMA
230 DIN A(100),V(100) 475 REN *****
235 REN ***** 480 REN * KRAJ ILI NOVA INTERPOLACIJA *
240 REN * ULAZNI PODACI * 485 REN *****
245 REN ***** 490 NI=5
250 INPUT "UNESITE BROJ TACA";D 495 SOBUD 530
255 PRINT 500 INPUT"(1)-NOVA INTERPOLACIJA, (2)-KRAJ";M
260 PRINT"UNESITE JEDAN ISPOD DRUGOPAROVE" 505 IF M<1) THEN END
265 PRINT"TACA X(I),Y(I)" 510 BOTO 340
270 PRINT 515 REN *****
275 FOR L=1 TO D 520 REN * STAMPANJE NI PRAZNIH REDOVA *
280 PRINT" X(1);L1),Y(1);L1)"=A1 525 REN *****
285 INPUT A(L),V(L) 530 FOR M=1 TO NI
290 NEXT L 535 PRINT
295 PRINT 540 NEXT N
300 PRINT"AKO JE I-TI RED POGRESAN, UNESITE I" 545 RETURN
305 INPUT"AKO NEMA GREŠKE, UNESITE 0";M 550 REN *****
310 PRINT 555 REN * IZAVANJE INTERPOLIRANE VREDNOSTI *
315 IF M=0 THEN 345:REN TACNO UNETO, DALJE 560 REN *****
320 IF M>0 THEN 295:REN VEĆE OD DIMENZIJE, PONOVO 565 PRINT
325 PRINT"UNESITE X(1);N1),Y(1);N1)"=A1 570 PRINT"INTERPOLIRANA VREDNOST FUNKCIJE"
330 INPUT A(N),V(N):REN UNOŠENJE ISPRAVKE 575 PRINT"Y =";Y
335 BOTO 295 580 PRINT
340 PRINT 585 RETURN

```

$$f^{(3)}(x) = \frac{2}{x^3}$$

$$f^{(4)}(x) = \frac{-6}{x^4}$$

Dakle na osnovu (10) sledi:

$$R_3(1) = \frac{(1-0,8)(1-0,9)(1-1,1)(1-1,2)}{4!}$$

$$= \frac{0,0004 \cdot \frac{-6}{3^4}}{24} = \frac{-0,0001}{3^4}$$

gde za ksi znamo da leži negde u intervalu [0,8,1,2] mada tačnu vrednost za ksi ne možemo odrediti. Možemo, međutim, odrediti gornju i donju granicu za grešku. Kako je poslednji izraz za grešku monotona funkcija promenljive kxi, to će njegove ekstremne vrednosti biti dostignute na granicama intervala [0,8,1,2]. Te vrednosti su respektivno -0,000244 i -0,000048. Dakle i greška za x=1 leži u intervalu [-0,000244, -0,000048]. Ovo se slaže sa gore određenom greškom od -0,000103, jer ova vrednost leži između upravo nađene gornje i donje granice za grešku.

U vezi sa greškom, može se postaviti niz važnih i interesantnih pitanja. Kako redukovati grešku? Šta učiniti kada nam nisu poznati izvodi funkcije f? Da li je moguće interpolacijom dostići proizvoljnu unapred zadatu tačnost?

O ovim pitanjima, kao i o Newtonovom (Njutnovom) interpolacionom polinomu, biće reči u narednom nastavku.

kamenčići na kompjuteru

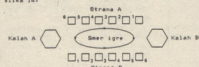
Minimaks postupak, obzirom na mane koje smo izložili u prošlom broju, možete da koristite samo kod sasvim jednostavnih igara; za složenije igre nam je potrebno bolje oružje, što ne znači da će nam upravo stečena znanja biti nekorisna. Ukoliko ste pre čitanje ove Škole u slobodnom vremenu razmišljali o programiranju neke logičke igre, verovatno ste formulisali skup pravila koje neki igrač treba da sledi da bi pobeđio. Zar ta pravila ne bi bilo moguće ugraditi u program? Zašto da ne! Posmatrajmo, na primer, „krsčić — kružić“ igru čija pravila znaju sva deca. Program koji bi igrao koristeći sledeća pravila bi bio sasvim dobar: jeste da ne bi mnogo pobeđivao ali ne bi ni gubio:

1. Ako postoji pobeđnički potez, odigraj ga.
2. Ako protivnik ima pobeđnički potez, blokiraj ga.
3. Ako je središnje polje prazno, zauzmi ga.
4. Odigraj bilo koji (RND) potez.

Rupice i kamenčići

Nije nam, međutim, cilj da pišemo programe koristeći neke hibride ljudskog razmišljanja i kompjuterske primene. Lepše je (ali ne uvek i efektivnije) razvijati makar rudimentarne oblike veštačke inteligencije pa ćemo se pozabaviti igrom „Kalah“ (poznata i pod imenom „Awari“) koja je, da vas utešimo, komplikovanija nego „krsčić—kružić“!

slika 14:



Na slici 14 je prikazana tabla za igru „Kalah“. Učestvuju, jasno, igrači A i B. Ispred svakog od njih se nalazi šest rupa koje su obeležene brojevima 1—6. Sa desne strane svakoga od njih je specijalna rupa nazvana „Kalah“. U početku se u svakoj od 12 malih rupa nalazi po n kamenčića. Igrač koji je u potezu uzima sve kamenčiće iz jedne od malih rupa koje se nalaze ispred njega i raspoređuje po jedan od njih u svaku od sledećih rupa u smeru obrnutom od kretanja kazaljke na satu. Ukoliko je potrebno, neki od kamenčića ulaze u igračev „kalah“, a neki mogu da preteku čak i za protivničku stranu table, ali ne i za njegov „kalah“. Postoje i dva dodatna pravila:

Igrači razmenično igraju po jedan potez osim kada zadnji od kamenčića koje neko raspoređuje završi u njegovom „kalahu“. Taj igrač tad igra još jednom.

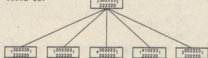
Ukoliko zadnji od kamenčića koje neko raspoređuje završi u praznoj rupi sa njegove strane, taj kamenčić se zajedno sa svim eventualnim kamenčićima u protivničkoj rupi sa istim brojem seli u igračev „kalah“.

Igra se završava kada neki od igrača ne može da odigra potez. Tada se svi kamenčići iz malih rupa protivnika (ako ih ima) sele u njegov „kalah“, pa pobeđuje onaj ko je u „kalahu“ sakupio više kamenčića. Pravila čete, naravno, najbolje uvežbati ako pokušate da igrate „kalah“ sa nekim prijateljima ili, u nedostatku njega, sami sa sobom. Čini se da je razumno odlučiti da se u svakoj od rupa u početku nađu 2 ili 3 kamenčića; igra sa 6 kamenčića u svakoj rupi je više nego teška!

Da je neko izložio pravila ove igre i rekao da čete vi sastaviti program koji je dobro igra, mislili biste da se šal! Videćete, međutim, da programirati „kalah“ i slične igre nije naročito teško; treba samo ispravno pristupiti problemu i rešavati ga „u koracima preciziranja“. U prvoj iteraciji ćemo sastaviti program koji ispituje svaki moguć potez i bira onaj posle koga se dolazi u najbolju moguću poziciju; donjice ćemo uzeti u obzir i ispitivanje odgovora protivnika kao i dalja povećanja „dubine razmišljanja“.

Da bi program mogao da izabere najbolji potez, treba mu pružiti mogućnost da isпита vrednosti pojedinih pozicija. Kako minimaks postupak ne dolazi u obzir (stabilno „kalah“ je nezamislivo uče do stable igre „poslednji pobeđuje?“), vešćemo takozvanu statičku funkciju pozicije. Jasno je da statičku funkciju pozicije možemo da izaberemo na mnogo raznih načina; odlučili smo da za početak izbrojimo sve kamenčiće u rupama sa naše strane table i da im dodamo broj kamenčića u našem „kalahu“, a da zatim od zbira oduzemo broj kamenčića u otvorima koji pripadaju protivniku i u njegovom kalahu. Na slici 15 je prikazana jedna moguća pozicija i svih pet poteza koje A može da odigra; uz svaki od njih je upisana vrednost statičke funkcije pozicije što znači da će A odigrati središnji potez koji mu daje 6 kamenčića prednosti.

slika 15:

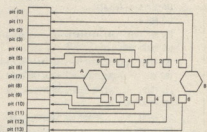


Može i sasvim jednostavno

Pre nego što se prihvatimo pisanja programa, moramo da se odlučimo za način na koji ćemo „pamtiti“ neku poziciju. Prvo što pada na pamet je koristiti dvodimenzionalnu matricu pit (7,2) u čije je elemente upisan broj kamenčića u svakoj od rupa. Ukoliko hoćemo da pravimo nekakvu tabelu odluka, mogli bismo hesadekadno pa-

kovati podatke pa ih „poukovati“ u memoriju koju bismo na taj način štedeli. Nama se učinilo najjednostavnijim da dimenzionismo niz pit (13) u čijim elementima smeštamo broj kamenčića u svakoj od rupa. Iako ovaj pristup ne daje dovoljno uverljiv utisak o dvojednoj tabli sa slike 14, vrlo je pogodan za manipulisanje potezima: obzirom da je tabla za „kalah“ kružna, raspoređivanje kamenčića u rupu može da se realizuje vrlo jednostavno koristeći funkciju MOD: broj := (broj+1) MOD 14 će nam uvek dati broj sledeće rupe u koju treba da stavimo kamenčić. Ovaj način „pamćenja“ table je prikazan na slici 16.

slika 16:



Što se pisanja programa tiče, potrebna nam je najpre procedura koja će ispisivati na ekranu trenutnu poziciju. Obzirom da ne želimo mnogo da se bavimo grafikom koja i onako znatno varira od računara do računara, jednostavno ćemo ispisati broj kamenčića u svakoj od rupa tako da se na ekranu dobije utisak o dvodimenzionalnoj tabli; ukoliko ste ljubitelj grafike, pokušajte da uradite nešto bolje. Potrebna nam je, osim toga, procedura koja će inicijalizovati igru (tj. staviti po nekoliko kamenčića u svaku rupu i isprazniti „kalah“), kao i procedure koje će testirati kraj igre i proglasiti pobeđnike. Ovi postupci su prikazani na slici 17 i nije im potreban nikakav naročit komentar.

slika 17:

```

10 = procedure displayboard;
11 = var p; integer;
12 = begin
13 = write(10);
14 = write(' ');
15 = write(' ');
16 = for p:=0 to 5 do write ('0');
17 = write(' ');
18 = write(' ');
19 = write(' ');
20 = for p:=0 to 5 do write ('0');
21 = write(' ');
22 = end;
23 = procedure startboard;
24 = var p, n; integer;
25 = integer;
26 = integer;
27 = begin
28 = write('Kalah kamenčića po gomili:');
29 = readln(n);
30 = for p:=0 to 5 do pit [(p+1) mod 14];
31 = for n:=0 to 12 do pit [(n+1) mod 14];
32 = pit [(13) mod 14] := 0;
33 = write('Tablu iz ovog potkret:');
34 = readln(p);
35 = if (p mod 14) = 0 then (p := p + 14);
36 = if (p mod 14) = 13 then (p := p - 14);
37 = else turn:=1;
38 = end;
39 = end;
40 = procedure testboard;
41 = var move, start, p; integer;
42 = integer;
43 = if turn=1 then start:=1 else start:=0;

```



```

205 @ =      if turn="A"
206 @ =      then begin
207 @ =          displayboard: playerd;
208 @ =          testgameover;
209 @ =      end;
210 @ =      if turn="B"
211 @ =      then begin
212 @ =          displayboard: playerb;
213 @ =          testgameover;
214 @ =      end;
215 @ =      until turn="A";
216 @ =      displayboard;
217 @ =      end;
218 @ =      end;
219 @ =      until turn="A";
220 @ =      displayboard;
221 @ =      until turn="B";
222 @ =      displayboard;
223 @ =      until turn="A";
224 @ =      displayboard;
225 @ =      until turn="B";
226 @ =      displayboard;
227 @ =      until turn="A";
228 @ =      displayboard;
229 @ =      until turn="B";
230 @ =      displayboard;
231 @ =      until turn="A";
232 @ =      displayboard;
233 @ =      until turn="B";
234 @ =      displayboard;
235 @ =      until turn="A";
236 @ =      displayboard;
237 @ =      until turn="B";
238 @ =      displayboard;
239 @ =      until turn="A";
240 @ =      displayboard;
241 @ =      until turn="B";
242 @ =      displayboard;
243 @ =      until turn="A";
244 @ =      displayboard;
245 @ =      until turn="B";
246 @ =      displayboard;
247 @ =      until turn="A";
248 @ =      displayboard;
249 @ =      until turn="B";
250 @ =      displayboard;
251 @ =      until turn="A";
252 @ =      displayboard;
253 @ =      until turn="B";
254 @ =      displayboard;
255 @ =      until turn="A";
256 @ =      displayboard;
257 @ =      until turn="B";
258 @ =      displayboard;
259 @ =      until turn="A";
260 @ =      displayboard;
261 @ =      until turn="B";
262 @ =      displayboard;
263 @ =      until turn="A";
264 @ =      displayboard;
265 @ =      until turn="B";
266 @ =      displayboard;
267 @ =      until turn="A";
268 @ =      displayboard;
269 @ =      until turn="B";
270 @ =      displayboard;
271 @ =      until turn="A";
272 @ =      displayboard;
273 @ =      until turn="B";
274 @ =      displayboard;
275 @ =      until turn="A";
276 @ =      displayboard;
277 @ =      until turn="B";
278 @ =      displayboard;
279 @ =      until turn="A";
280 @ =      displayboard;
281 @ =      until turn="B";
282 @ =      displayboard;
283 @ =      until turn="A";
284 @ =      displayboard;
285 @ =      until turn="B";
286 @ =      displayboard;
287 @ =      until turn="A";
288 @ =      displayboard;
289 @ =      until turn="B";
290 @ =      displayboard;
291 @ =      until turn="A";
292 @ =      displayboard;
293 @ =      until turn="B";
294 @ =      displayboard;
295 @ =      until turn="A";
296 @ =      displayboard;
297 @ =      until turn="B";
298 @ =      displayboard;
299 @ =      until turn="A";
300 @ =      displayboard;
301 @ =      until turn="B";
302 @ =      displayboard;
303 @ =      until turn="A";
304 @ =      displayboard;
305 @ =      until turn="B";
306 @ =      displayboard;
307 @ =      until turn="A";
308 @ =      displayboard;
309 @ =      until turn="B";
310 @ =      displayboard;
311 @ =      until turn="A";
312 @ =      displayboard;
313 @ =      until turn="B";
314 @ =      displayboard;
315 @ =      until turn="A";
316 @ =      displayboard;
317 @ =      until turn="B";
318 @ =      displayboard;
319 @ =      until turn="A";
320 @ =      displayboard;
321 @ =      until turn="B";
322 @ =      displayboard;
323 @ =      until turn="A";
324 @ =      displayboard;
325 @ =      until turn="B";
326 @ =      displayboard;
327 @ =      until turn="A";
328 @ =      displayboard;
329 @ =      until turn="B";
330 @ =      displayboard;
331 @ =      until turn="A";
332 @ =      displayboard;
333 @ =      until turn="B";
334 @ =      displayboard;
335 @ =      until turn="A";
336 @ =      displayboard;
337 @ =      until turn="B";
338 @ =      displayboard;
339 @ =      until turn="A";
340 @ =      displayboard;
341 @ =      until turn="B";
342 @ =      displayboard;
343 @ =      until turn="A";
344 @ =      displayboard;
345 @ =      until turn="B";
346 @ =      displayboard;
347 @ =      until turn="A";
348 @ =      displayboard;
349 @ =      until turn="B";
350 @ =      displayboard;
351 @ =      until turn="A";
352 @ =      displayboard;
353 @ =      until turn="B";
354 @ =      displayboard;
355 @ =      until turn="A";
356 @ =      displayboard;
357 @ =      until turn="B";
358 @ =      displayboard;
359 @ =      until turn="A";
360 @ =      displayboard;
361 @ =      until turn="B";
362 @ =      displayboard;
363 @ =      until turn="A";
364 @ =      displayboard;
365 @ =      until turn="B";
366 @ =      displayboard;
367 @ =      until turn="A";
368 @ =      displayboard;
369 @ =      until turn="B";
370 @ =      displayboard;
371 @ =      until turn="A";
372 @ =      displayboard;
373 @ =      until turn="B";
374 @ =      displayboard;
375 @ =      until turn="A";
376 @ =      displayboard;
377 @ =      until turn="B";
378 @ =      displayboard;
379 @ =      until turn="A";
380 @ =      displayboard;
381 @ =      until turn="B";
382 @ =      displayboard;
383 @ =      until turn="A";
384 @ =      displayboard;
385 @ =      until turn="B";
386 @ =      displayboard;
387 @ =      until turn="A";
388 @ =      displayboard;
389 @ =      until turn="B";
390 @ =      displayboard;
391 @ =      until turn="A";
392 @ =      displayboard;
393 @ =      until turn="B";
394 @ =      displayboard;
395 @ =      until turn="A";
396 @ =      displayboard;
397 @ =      until turn="B";
398 @ =      displayboard;
399 @ =      until turn="A";
400 @ =      displayboard;
401 @ =      until turn="B";
402 @ =      displayboard;
403 @ =      until turn="A";
404 @ =      displayboard;
405 @ =      until turn="B";
406 @ =      displayboard;
407 @ =      until turn="A";
408 @ =      displayboard;
409 @ =      until turn="B";
410 @ =      displayboard;
411 @ =      until turn="A";
412 @ =      displayboard;
413 @ =      until turn="B";
414 @ =      displayboard;
415 @ =      until turn="A";
416 @ =      displayboard;
417 @ =      until turn="B";
418 @ =      displayboard;
419 @ =      until turn="A";
420 @ =      displayboard;
421 @ =      until turn="B";
422 @ =      displayboard;
423 @ =      until turn="A";
424 @ =      displayboard;
425 @ =      until turn="B";
426 @ =      displayboard;
427 @ =      until turn="A";
428 @ =      displayboard;
429 @ =      until turn="B";
430 @ =      displayboard;
431 @ =      until turn="A";
432 @ =      displayboard;
433 @ =      until turn="B";
434 @ =      displayboard;
435 @ =      until turn="A";
436 @ =      displayboard;
437 @ =      until turn="B";
438 @ =      displayboard;
439 @ =      until turn="A";
440 @ =      displayboard;
441 @ =      until turn="B";
442 @ =      displayboard;
443 @ =      until turn="A";
444 @ =      displayboard;
445 @ =      until turn="B";
446 @ =      displayboard;
447 @ =      until turn="A";
448 @ =      displayboard;
449 @ =      until turn="B";
450 @ =      displayboard;
451 @ =      until turn="A";
452 @ =      displayboard;
453 @ =      until turn="B";
454 @ =      displayboard;
455 @ =      until turn="A";
456 @ =      displayboard;
457 @ =      until turn="B";
458 @ =      displayboard;
459 @ =      until turn="A";
460 @ =      displayboard;
461 @ =      until turn="B";
462 @ =      displayboard;
463 @ =      until turn="A";
464 @ =      displayboard;
465 @ =      until turn="B";
466 @ =      displayboard;
467 @ =      until turn="A";
468 @ =      displayboard;
469 @ =      until turn="B";
470 @ =      displayboard;
471 @ =      until turn="A";
472 @ =      displayboard;
473 @ =      until turn="B";
474 @ =      displayboard;
475 @ =      until turn="A";
476 @ =      displayboard;
477 @ =      until turn="B";
478 @ =      displayboard;
479 @ =      until turn="A";
480 @ =      displayboard;
481 @ =      until turn="B";
482 @ =      displayboard;
483 @ =      until turn="A";
484 @ =      displayboard;
485 @ =      until turn="B";
486 @ =      displayboard;
487 @ =      until turn="A";
488 @ =      displayboard;
489 @ =      until turn="B";
490 @ =      displayboard;
491 @ =      until turn="A";
492 @ =      displayboard;
493 @ =      until turn="B";
494 @ =      displayboard;
495 @ =      until turn="A";
496 @ =      displayboard;
497 @ =      until turn="B";
498 @ =      displayboard;
499 @ =      until turn="A";
500 @ =      displayboard;
501 @ =      until turn="B";
502 @ =      displayboard;
503 @ =      until turn="A";
504 @ =      displayboard;
505 @ =      until turn="B";
506 @ =      displayboard;
507 @ =      until turn="A";
508 @ =      displayboard;
509 @ =      until turn="B";
510 @ =      displayboard;
511 @ =      until turn="A";
512 @ =      displayboard;
513 @ =      until turn="B";
514 @ =      displayboard;
515 @ =      until turn="A";
516 @ =      displayboard;
517 @ =      until turn="B";
518 @ =      displayboard;
519 @ =      until turn="A";
520 @ =      displayboard;
521 @ =      until turn="B";
522 @ =      displayboard;
523 @ =      until turn="A";
524 @ =      displayboard;
525 @ =      until turn="B";
526 @ =      displayboard;
527 @ =      until turn="A";
528 @ =      displayboard;
529 @ =      until turn="B";
530 @ =      displayboard;
531 @ =      until turn="A";
532 @ =      displayboard;
533 @ =      until turn="B";
534 @ =      displayboard;
535 @ =      until turn="A";
536 @ =      displayboard;
537 @ =      until turn="B";
538 @ =      displayboard;
539 @ =      until turn="A";
540 @ =      displayboard;
541 @ =      until turn="B";
542 @ =      displayboard;
543 @ =      until turn="A";
544 @ =      displayboard;
545 @ =      until turn="B";
546 @ =      displayboard;
547 @ =      until turn="A";
548 @ =      displayboard;
549 @ =      until turn="B";
550 @ =      displayboard;
551 @ =      until turn="A";
552 @ =      displayboard;
553 @ =      until turn="B";
554 @ =      displayboard;
555 @ =      until turn="A";
556 @ =      displayboard;
557 @ =      until turn="B";
558 @ =      displayboard;
559 @ =      until turn="A";
560 @ =      displayboard;
561 @ =      until turn="B";
562 @ =      displayboard;
563 @ =      until turn="A";
564 @ =      displayboard;
565 @ =      until turn="B";
566 @ =      displayboard;
567 @ =      until turn="A";
568 @ =      displayboard;
569 @ =      until turn="B";
570 @ =      displayboard;
571 @ =      until turn="A";
572 @ =      displayboard;
573 @ =      until turn="B";
574 @ =      displayboard;
575 @ =      until turn="A";
576 @ =      displayboard;
577 @ =      until turn="B";
578 @ =      displayboard;
579 @ =      until turn="A";
580 @ =      displayboard;
581 @ =      until turn="B";
582 @ =      displayboard;
583 @ =      until turn="A";
584 @ =      displayboard;
585 @ =      until turn="B";
586 @ =      displayboard;
587 @ =      until turn="A";
588 @ =      displayboard;
589 @ =      until turn="B";
590 @ =      displayboard;
591 @ =      until turn="A";
592 @ =      displayboard;
593 @ =      until turn="B";
594 @ =      displayboard;
595 @ =      until turn="A";
596 @ =      displayboard;
597 @ =      until turn="B";
598 @ =      displayboard;
599 @ =      until turn="A";
600 @ =      displayboard;
601 @ =      until turn="B";
602 @ =      displayboard;
603 @ =      until turn="A";
604 @ =      displayboard;
605 @ =      until turn="B";
606 @ =      displayboard;
607 @ =      until turn="A";
608 @ =      displayboard;
609 @ =      until turn="B";
610 @ =      displayboard;
611 @ =      until turn="A";
612 @ =      displayboard;
613 @ =      until turn="B";
614 @ =      displayboard;
615 @ =      until turn="A";
616 @ =      displayboard;
617 @ =      until turn="B";
618 @ =      displayboard;
619 @ =      until turn="A";
620 @ =      displayboard;
621 @ =      until turn="B";
622 @ =      displayboard;
623 @ =      until turn="A";
624 @ =      displayboard;
625 @ =      until turn="B";
626 @ =      displayboard;
627 @ =      until turn="A";
628 @ =      displayboard;
629 @ =      until turn="B";
630 @ =      displayboard;
631 @ =      until turn="A";
632 @ =      displayboard;
633 @ =      until turn="B";
634 @ =      displayboard;
635 @ =      until turn="A";
636 @ =      displayboard;
637 @ =      until turn="B";
638 @ =      displayboard;
639 @ =      until turn="A";
640 @ =      displayboard;
641 @ =      until turn="B";
642 @ =      displayboard;
643 @ =      until turn="A";
644 @ =      displayboard;
645 @ =      until turn="B";
646 @ =      displayboard;
647 @ =      until turn="A";
648 @ =      displayboard;
649 @ =      until turn="B";
650 @ =      displayboard;
651 @ =      until turn="A";
652 @ =      displayboard;
653 @ =      until turn="B";
654 @ =      displayboard;
655 @ =      until turn="A";
656 @ =      displayboard;
657 @ =      until turn="B";
658 @ =      displayboard;
659 @ =      until turn="A";
660 @ =      displayboard;
661 @ =      until turn="B";
662 @ =      displayboard;
663 @ =      until turn="A";
664 @ =      displayboard;
665 @ =      until turn="B";
666 @ =      displayboard;
667 @ =      until turn="A";
668 @ =      displayboard;
669 @ =      until turn="B";
670 @ =      displayboard;
671 @ =      until turn="A";
672 @ =      displayboard;
673 @ =      until turn="B";
674 @ =      displayboard;
675 @ =      until turn="A";
676 @ =      displayboard;
677 @ =      until turn="B";
678 @ =      displayboard;
679 @ =      until turn="A";
680 @ =      displayboard;
681 @ =      until turn="B";
682 @ =      displayboard;
683 @ =      until turn="A";
684 @ =      displayboard;
685 @ =      until turn="B";
686 @ =      displayboard;
687 @ =      until turn="A";
688 @ =      displayboard;
689 @ =      until turn="B";
690 @ =      displayboard;
691 @ =      until turn="A";
692 @ =      displayboard;
693 @ =      until turn="B";
694 @ =      displayboard;
695 @ =      until turn="A";
696 @ =      displayboard;
697 @ =      until turn="B";
698 @ =      displayboard;
699 @ =      until turn="A";
700 @ =      displayboard;
701 @ =      until turn="B";
702 @ =      displayboard;
703 @ =      until turn="A";
704 @ =      displayboard;
705 @ =      until turn="B";
706 @ =      displayboard;
707 @ =      until turn="A";
708 @ =      displayboard;
709 @ =      until turn="B";
710 @ =      displayboard;
711 @ =      until turn="A";
712 @ =      displayboard;
713 @ =      until turn="B";
714 @ =      displayboard;
715 @ =      until turn="A";
716 @ =      displayboard;
717 @ =      until turn="B";
718 @ =      displayboard;
719 @ =      until turn="A";
720 @ =      displayboard;
721 @ =      until turn="B";
722 @ =      displayboard;
723 @ =      until turn="A";
724 @ =      displayboard;
725 @ =      until turn="B";
726 @ =      displayboard;
727 @ =      until turn="A";
728 @ =      displayboard;
729 @ =      until turn="B";
730 @ =      displayboard;
731 @ =      until turn="A";
732 @ =      displayboard;
733 @ =      until turn="B";
734 @ =      displayboard;
735 @ =      until turn="A";
736 @ =      displayboard;
737 @ =      until turn="B";
738 @ =      displayboard;
739 @ =      until turn="A";
740 @ =      displayboard;
741 @ =      until turn="B";
742 @ =      displayboard;
743 @ =      until turn="A";
744 @ =      displayboard;
745 @ =      until turn="B";
746 @ =      displayboard;
747 @ =      until turn="A";
748 @ =      displayboard;
749 @ =      until turn="B";
750 @ =      displayboard;
751 @ =      until turn="A";
752 @ =      displayboard;
753 @ =      until turn="B";
754 @ =      displayboard;
755 @ =      until turn="A";
756 @ =      displayboard;
757 @ =      until turn="B";
758 @ =      displayboard;
759 @ =      until turn="A";
760 @ =      displayboard;
761 @ =      until turn="B";
762 @ =      displayboard;
763 @ =      until turn="A";
764 @ =      displayboard;
765 @ =      until turn="B";
766 @ =      displayboard;
767 @ =      until turn="A";
768 @ =      displayboard;
769 @ =      until turn="B";
770 @ =      displayboard;
771 @ =      until turn="A";
772 @ =      displayboard;
773 @ =      until turn="B";
774 @ =      displayboard;
775 @ =      until turn="A";
776 @ =      displayboard;
777 @ =      until turn="B";
778 @ =      displayboard;
779 @ =      until turn="A";
780 @ =      displayboard;
781 @ =      until turn="B";
782 @ =      displayboard;
783 @ =      until turn="A";
784 @ =      displayboard;
785 @ =      until turn="B";
786 @ =      displayboard;
787 @ =      until turn="A";
788 @ =      displayboard;
789 @ =      until turn="B";
790 @ =      displayboard;
791 @ =      until turn="A";
792 @ =      displayboard;
793 @ =      until turn="B";
794 @ =      displayboard;
795 @ =      until turn="A";
796 @ =      displayboard;
797 @ =      until turn="B";
798 @ =      displayboard;
799 @ =      until turn="A";
800 @ =      displayboard;
801 @ =      until turn="B";
802 @ =      displayboard;
803 @ =      until turn="A";
804 @ =      displayboard;
805 @ =      until turn="B";
806 @ =      displayboard;
807 @ =      until turn="A";
808 @ =      displayboard;
809 @ =      until turn="B";
810 @ =      displayboard;
811 @ =      until turn="A";
812 @ =      displayboard;
813 @ =      until turn="B";
814 @ =      displayboard;
815 @ =      until turn="A";
816 @ =      displayboard;
817 @ =      until turn="B";
818 @ =      displayboard;
819 @ =      until turn="A";
820 @ =      displayboard;
821 @ =      until turn="B";
822 @ =      displayboard;
823 @ =      until turn="A";
824 @ =      displayboard;
825 @ =      until turn="B";
826 @ =      displayboard;
827 @ =      until turn="A";
828 @ =      displayboard;
829 @ =      until turn="B";
830 @ =      displayboard;
831 @ =      until turn="A";
832 @ =      displayboard;
833 @ =      until turn="B";
834 @ =      displayboard;
835 @ =      until turn="A";
836 @ =      displayboard;
837 @ =      until turn="B";
838 @ =      displayboard;
839 @ =      until turn="A";
840 @ =      displayboard;
841 @ =      until turn="B";
842 @ =      displayboard;
843 @ =      until turn="A";
844 @ =      displayboard;
845 @ =      until turn="B";
846 @ =      displayboard;
847 @ =      until turn="A";
848 @ =      displayboard;
849 @ =      until turn="B";
850 @ =      displayboard;
851 @ =      until turn="A";
852 @ =      displayboard;
853 @ =      until turn="B";
854 @ =      displayboard;
855 @ =      until turn="A";
856 @ =      displayboard;
857 @ =      until turn="B";
858 @ =      displayboard;
859 @ =      until turn="A";
860 @ =      displayboard;
861 @ =      until turn="B";
862 @ =      displayboard;
863 @ =      until turn="A";
864 @ =      displayboard;
865 @ =      until turn="B";
866 @ =      displayboard;
867 @ =      until turn="A";
868 @ =      displayboard;
869 @ =      until turn="B";
870 @ =      displayboard;
871 @ =      until turn="A";
872 @ =      displayboard;
873 @ =      until turn="B";
874 @ =      displayboard;
875 @ =      until turn="A";
876 @ =      displayboard;
877 @ =      until turn="B";
878 @ =      displayboard;
879 @ =      until turn="A";
880 @ =      displayboard;
881 @ =      until turn="B";
882 @ =      displayboard;
883 @ =      until turn="A";
884 @ =      displayboard;
885 @ =      until turn="B";
886 @ =      displayboard;
887 @ =      until turn="A";
888 @ =      displayboard;
889 @ =      until turn="B";
890 @ =      displayboard;
891 @ =      until turn="A";
892 @ =      displayboard;
893 @ =      until turn="B";
894 @ =      displayboard;
895 @ =      until turn="A";
896 @ =      displayboard;
897 @ =      until turn="B";
898 @ =      displayboard;
899 @ =      until turn="A";
900 @ =      displayboard;
901 @ =      until turn="B";
902 @ =      displayboard;
903 @ =      until turn="A";
904 @ =      displayboard;
905 @ =      until turn="B";
906 @ =      displayboard;
907 @ =      until turn="A";
908 @ =      displayboard;
909 @ =      until turn="B";
910 @ =      displayboard;
911 @ =      until turn="A";
912 @ =      displayboard;
913 @ =      until turn="B";
914 @ =      displayboard;
915 @ =      until turn="A";
916 @ =      displayboard;
917 @ =      until turn="B";
918 @ =      displayboard;
919 @ =      until turn="A";
920 @ =      displayboard;
921 @ =      until turn="B";
922 @ =      displayboard;
923 @ =      until turn="A";
924 @ =      displayboard;
925 @ =      until turn="B";
926 @ =      displayboard;
927 @ =      until turn="A";
928 @ =      displayboard;
929 @ =      until turn="B";
930 @ =      displayboard;
931 @ =      until turn="A";
932 @ =      displayboard;
933 @ =      until turn="B";
934 @ =      displayboard;
935 @ =      until turn="A";
936 @ =      displayboard;
937 @ =      until turn="B";
938 @ =      displayboard;
939 @ =      until turn="A";
940 @ =      displayboard;
941 @ =      until turn="B";
942 @ =      displayboard;
943 @ =      until turn="A";
944 @ =      displayboard;
945 @ =      until turn="B";
946 @ =      displayboard;
947 @ =      until turn="A";
948 @ =      displayboard;
949 @ =      until turn="B";
950 @ =      displayboard;
951 @ =      until turn="A";
952 @ =      displayboard;
953 @ =      until turn="B";
954 @ =      displayboard;
955 @ =      until turn="A";
956 @ =      displayboard;
957 @ =      until turn="B";
958 @ =      displayboard;
959 @ =      until turn="A";
960 @ =      displayboard;
961 @ =      until turn="B";
962 @ =      displayboard;
963 @ =      until turn="A";
964 @ =      displayboard;
965 @ =      until turn="B";
966 @ =      displayboard;
967 @ =      until turn="A";
968 @ =      displayboard;
969 @ =      until turn="B";
970 @ =      displayboard;
971 @ =      until turn="A";
972 @ =      displayboard;
973 @ =      until turn="B";
974 @ =      displayboard;
975 @ =      until turn="A";
976 @ =      displayboard;
977 @ =      until turn="B";
978 @ =      displayboard;
979 @ =      until turn="A";
980 @ =      displayboard;
981 @ =      until turn="B";
982 @ =      displayboard;
983 @ =      until turn="A";
984 @ =      displayboard;
985 @ =      until turn="B";
986 @ =      displayboard;
987 @ =      until turn="A";
988 @ =      displayboard;
989 @ =      until turn="B";
990 @ =      displayboard;
991 @ =      until turn="A";
992 @ =      displayboard;
993 @ =      until turn="B";
994 @ =      displayboard;
995 @ =      until turn="A";
996 @ =      displayboard;
997 @ =      until turn="B";
998 @ =      displayboard;
999 @ =      until turn="A";
1000 @ =      displayboard;

```

Isprobali smo ovako dobijeni program na BBC B koji je poznat kao prilično brz kompjuter i ustanovili da postavljanje maxdepth:=3 zahteva desetak sekundi „razmišljanja“ za svaki potez, dok maxdepth:4 produžava ovo bar po pet puta. Pokazalo se, međutim, da i maxdepth:=2 daje sasvim pristojnu igru računara koga ćemo pobeđivati tek kada steknemo malo više iskustva (ako poželite da povećate brzinu rada programa prevodeći ga na assembler, poslušajte prijateljski savet i odustanite od togal).

Kalah bez paskala

Pošto smo napisali kostur programa, nije ga teško poboljšavati i modifikovati. Najperspektivnija mogućnost je selektivnost pri odluci o prekidu dalje analize pozicije: umesto da se zaustavimo kada dostignemo maxdepth, možemo da predvidimo da se ispitivanje prekida kada je dostignuta maksimalna dubina i kada je sekvenca uzimanja završena (često se dešava da naša verzija programa ne primeti da će posle spektakularnog skoka skoro doći još spektakularniji pad jer će protivnik u sledećem potezu uzeti ogroman broj kamenčića). Modifikacija je više nego jednostavna: treba promeniti minval i maxval tako da konačni uslovi glase: IF (depth>=maxdepth) and (caprute <=0) then min:=staticval



Iako smo na ovaj način stigli do kraja drugog dela naše škole logičkih igara, nismo ni iz daleka rekli sve što bi moglo da se kaže na nivou za koji smo se opredelili. Želeli smo, naime, da pokažemo da se logičke igre mogu napisati bez mnogo logičkog razmišljanja: naš program za Kalah bi uz vrlo male prepravke pobeđivao u bilo kojoj drugoj logičkoj igri što znači da nije teško simulirati inteligenciju korišćenjem islučivo procesorskih moći računara. Dok čekate naš sledeći nastavak u kome ćemo se pozabaviti tehnikama poznatim kao tree pruning (već ćemo se setiti kako to da prevedemo), zavarajte se sledećim problemom — nagradimo najbolja rešenja.

Možete li da programirate „Kalah“ na bejziku? A na „galaksijinom“ bejziku? Naš Katalog programa je nekada imao program za igru „Awar“ pisan za TRS 80 koji ne omogućava rekurziju; program je, čak, učio u toku igre!

Dejan Ristanović

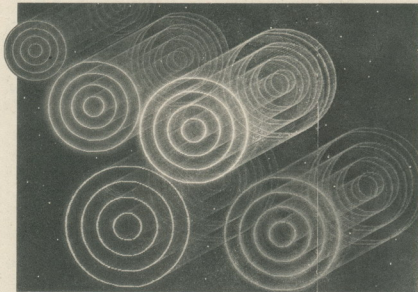
55/kamenčići na kompjuteru

hiperboličke funkcije

Matematički softver

To može i bolje (3)

Korisnici računara veruju da se vrednosti hiperboličkih funkcija jednostavno računaju svodenjem na pouzdanu firmu eksponencijalnu funkciju EXP. Programeri znaju da je to verovanje dvostruko pogrešno: niti se računanje vrednosti hiperboličkih funkcija može svesti na eksponencijalnu funkciju, niti je firmi program EXP pouzdan. Prof. dr Dušan Slavić nastavlja seriju „To može i bolje“, koja sadrži detalje njegovog dve decenije dugog istraživanja u oblasti računarskih algoritama.



Program za izračunavanje vrednosti računarske funkcije je valjan ako za svaku vrednost argumenta daje vrednost funkcije tačnu do poslednjeg bita mantise. Ovdje pretpostavljamo da je program za izračunavanje vrednosti eksponencijalne funkcije valjan. U Računarima 10 (str. 54–56) dat je takav program. Uz pretpostavku da je EXP valjan program, istraživanje pokazuje da se vrednosti hiperboličkih funkcija

- (1) $sh(x) = (exp(x) - exp(-x))/2$,
- (2) $ch(x) = (exp(x) + exp(-x))/2$,
- (3) $th(x) = (exp(2x) - 1)/(exp(2x) + 1)$

ne mogu dobiti direktnom primenom formula (1), (2) i (3). To je veliko iznenađenje za korisnike računara, koji misle da za programiranje nije potrebno poznavati numeričku matematiku. Od formula (1), (2), (3) do valjanih programa za izračunavanje vrednosti hiperboličkih funkcija dug je put. Taj put vodi preko vrednovanja numeričke upotrebljivosti, kako tih formula, tako i brojnih drugih formula koje sadrže: beskonačni proizvod, potencijalni razvoj, racionalni razvoj ili verižni razvoj — i to bez ekonomizacije ili sa ekonomizacijom. Kako nijedna od ovih aproksimacija nije dovoljna za sve vrednosti argumenta, neophodno je pribеći kombinacijama nekoliko aproksimacija za svaku računarsku funkciju.

Naravno, uvek je moguće napisati (u smislu matematičke) tačne formule (1), (2), (3) — pa šta bude! Ne samo hartija, nego i računari trpe sve... Nevolja je jedino što se za većinu mogućih vrednosti argumenta tada ne dobija nijedna tačna cifra rezultata. U korišćenju tako „programiranih“ funkcija „programer“ ubrzo uvida (na svoju bruku) da je pitanje hiperboličkih funkcija za njega postalo hiperbolno. Lakše je naučiti numeričku matematiku nego raditi bez nje.

Hiperbolički sinus

Slika 1 pokazuje da se formula (1) ne može upotrebiti za računanje vrednosti hiperboličkog sinusa sh(x) za male vrednosti modula argumenta. Ako je abs(x) manji od (1/2) log(2), vrednosti exp(x) i exp(-x) su bliske, pa je razlika tih vrednosti veoma neprecizna. Relativna greška raste sa smanjenjem abs(x), tako da se za abs(x) < 2⁻ⁿ nema nijedna tačna cifra rezultata. Greška predstavljenom slikom 1 ne može se smanjiti — jedino je rešenje da se u pomenutoj oblasti argumenta upotrebi neka druga formula. Sa J je označen broj bita mantise broja. Na primer, za „ZX spektrum“, „komodore C64“ ili Sharp MZ-700 taj broj J je 32. Relativna greška predstavljenja brojeva u računaru je 2^{-J}, u pomenutom računaru brojevi su predstavljeni sa X=0 ili

$$X = 2^m \text{ m}$$

gde je k karakteristika (ceo broj od -127 do 127) i m mantisa (od 1/2 do $1-2^{-J}$ u koraku po 2^{-j}). Najveći broj koji se može predstaviti je

$$Z = 2^{2J}(1-2^{-2J}) = 1.701411834E38$$

Za argumente x modula

$$\text{abs}(x) \geq \log(Z) + \log(2) = 128 \log(2)$$

vrednost modula hiperboličkog sinusa je veća od najvećeg broja Z, pa treba javiti prekoračenje opsega brojeva (overflow). To se može i posredno učiniti iz programa EXP.

Veliki broj autora predviđa da je u intervalima za

$$\log(Z) < \text{abs}(x) < \log(2Z)$$

moguće izračunati sh(x). Radi se o neopravdanom smanjenju opsega vrednosti funkcije na upola manji interval.

Pojedini autori za velike vrednosti abs(x) koriste aproksimaciju

$$\text{sh}(x) = \exp(\text{abs}(x) - \log(2)) \cdot \text{sgn}(x)$$

koja sadrži vrlo osetljivu približnu promenu argumenta eksponencijalne funkcije.

Formula (1) ne dozvoljava izračunavanje modula hiperboličkog sinusa u intervalu $[Z/2, Z]$. To su vrednosti koje bi se dobile u intervalima argumenta za koje je

$$127 \log(2) < \text{abs}(x) < 128 \log(2)$$

Kako se (naravno) želi sprečiti sužavanje oblasti definisanosti računarske funkcije, dovoljno je u tim intervalima usvojiti formulu

$$\text{sh}(x) = \exp(\text{abs}(x) - 88) \cdot E \cdot \text{sgn}(x),$$

gde je $E = \exp(88/2)$. Preostali član formule (1) je zanemarljiv. On je bio zanemarljiv i ranije. Naime, ako je $J = 32$, onda je $\exp(-\text{abs}(x))$ zanemarljivo za

$$\text{abs}(x) \geq 16 \log(2).$$

Dakle, u intervalima za koje važi

$$16 \log(2) < \text{abs}(x) < 127 \log(2)$$

može se sh(x) računati sa

$$\text{sh}(x) = \exp(\text{abs}(x))/2 \cdot \text{sgn}(x).$$

Za x iz intervala za koje važi

$$(1/2) \log(2) < \text{abs}(x) < 16 \log(2)$$

može se hiperbolički sinus računati pomoću

$$s = \exp(\text{abs}(x))/2, \text{ sh}(x) = -0.25/s.$$

Za $\text{abs}(x) < (1/2) \log(2)$ potrebno je naći bolji metod. U matematičkim priručnicima nudi se formula

$$\text{sh}(x) = x \prod_{k=1}^{+\infty} (1 + (x/\pi k)^2).$$

Ako se u toj formuli umesto $+\infty$ uzme broj n dobije se približna formula

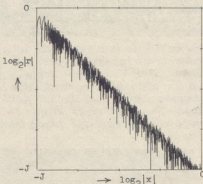
$$\text{sh}(x) = x \prod_{k=1}^n (1 + (x/\pi k)^2)$$

koja ima veoma veliku grešku, videti sliku 2. Kako bi potreban broj množenja bio stotine hiljada i kako želimo da dobijemo rezultat za života programera, ovaj postupak je neupotre-

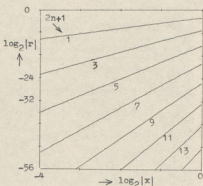
bljiv. Neko bi rekao da ova formula „ima veliki teorijski značaj“, što treba shvatiti ovako: praktičan značaj te formule je nikakav.

Slika 3 pokazuje da je neuporedivo korisniji potencijalni razvoj

$$\text{sh}(x) = \sum_{k=1}^{+\infty} x^{2k-1} / (2k-1)!$$

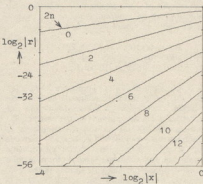


Slika 1. Relativna greška formule $\text{sh}(x) = (\exp(x) - \exp(-x)) / 2$ ako se funkcija EXP računa tačno. J je broj binarnih cifara mantise.



Slika 3. Relativna greška formule

$$\text{sh}(x) \approx x \sum_{k=0}^n x^{2k+1} / (2k+1)!$$

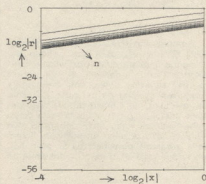


Slika 5. Relativna greška razvoja

$$\text{ch}(x) \approx \sum_{k=0}^n x^{2k} / (2k)!$$

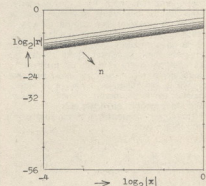
Umesto ovog potencijalnog rada dovoljno je uzeti nekoliko (n) sabiraka. Ako je abs(x) manje od (1/2) log(2), onda je veza između stepena polinoma $2n-1$ i broja tačnih bitova B data sledećom tabelom

2n-1	1	3	5	7	9	11	13
B	6	12	21	30	40	50	60



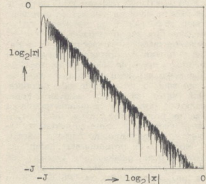
Slika 2. Relativna greška formule

$$\text{sh}(x) \approx x \cdot \prod_{k=1}^n \left(1 + \left(\frac{x}{\pi k} \right)^2 \right).$$



Slika 4. Relativna greška formule

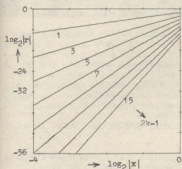
$$\text{ch}(x) \approx \sum_{k=1}^n \left(1 + \left(\frac{x}{\pi(k-1/2)} \right)^2 \right)^2.$$



Slika 6. Relativna greška formule

$\text{th}(x) = \text{sgn}(x) \cdot (t-1)/(t+1)$ ako se funkcija $t = \exp(2 \text{abs}(x))$ računa tačno.

Ekonomizacijom potencijalnog razvoja broj tačnih bitova na istom intervalu argumenta uz isti stepen polinoma (ali druge koeficijente) može se još povećati. Ako se usvoji stepen polinoma (npr. $2n-1=7$) i maksimalna relativna greška (npr. 2^{-3}), onda se ekonomizacijom može proširiti interval primene polinoma na $\text{abs}(x)$ manje od 0.39. To je učinjeno u aproksimaciji



Slika 7. Relativna greška razvoja

$$\text{th}(x) = \sum_{k=1}^n 2^{2k} (2^{2k-1}) B_{2k} x^{2k-1} / (2k)! \cdot$$

$$\text{sh}(x) = x + 0.166666667 x^3 + 0.00833333 x^5 + 0.0001988 x^7$$

Za male vrednosti modula argumenta dovoljno je usvojiti

$$\text{sh}(x) = x.$$

Ako je $J=32$ (relativna greška $2^{-32}=2.3E-10$), onda se to može učiniti u intervalu

$$\text{abs}(x) < (6 \cdot 2^{-4})^{1/3} = 0.00004401.$$

Hiperbolički kosinus

Računarsku funkciju hiperbolički kosinus $\text{ch}(x)$ lakše je programirati od hiperboličkog sinus $\text{sh}(x)$. Ako je x veće od $(J/2) \log(2)$, onda važi $\text{ch}(x) = \text{sh}(x)$, pa se deo zaključaka o izračunavanju vrednosti $\text{sh}(x)$ može koristiti i za $\text{ch}(x)$.

slika 4 predstavlja dokaz da je beskonačni proizvod

$$\text{ch}(x) = \prod_{k=1}^{+\infty} (1 + x^2/(k-1/2)^2)$$

neupotrebljiv u numeričkoj matematici, jer zahteva stotine hiljada operacija množenja. Korisnici programa ne bi bili toliko strpljivi.

Slika 5 pokazuje da se za male vrednosti argumenta potencijalni razvoj

$$\text{ch}(x) = \sum_{k=0}^{+\infty} x^{2k} / (2k)!$$

može uspešno primeniti. Granica primene ove formule prema formuli (2) nije vezana za tačnost, već za trajanje izračunavanja.

Ako je moduo argumenta mali, onda se za $\text{ch}(x)$ može usvojiti

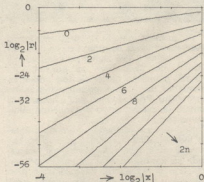
$$\text{ch}(x) = 1.$$

Interval primene te formule je

$$\text{abs}(x) < 2^{1-J/2} = 0.000021579.$$

Hiperbolički tangens

Računarska funkcija hiperbolički tangens $\text{th}(x)$ ne može se računati samo pomoću formule (3).



Slika 8. Relativna greška razvoja

$$\text{th}(x) \approx x / \left(\sum_{k=0}^n 2^{2k} B_{2k} x^{2k} / (2k)! \right).$$

```

11 LABEL "SINH"
12 A=ABS(X)
13 IF A<.00004401396394 S=X:RETURN
14 IF A<.35 A=AKA:S=(A+.0001988+.008333
333 )A+.166666667 )AAX+X:RETURN
15 ON ERROR GOTO 18: S=EXP(A)*.5XSGN(X)
16 IF A<11.89835489 S=S-.25/5
17 RETURN
18 S=EXP(A-BB)*.8.25B181275E+37XSGN(X):
RETURN

```

```

21 LABEL "COSH"
22 A=ABS(X)
23 IF A<.00002157918644 C=1:RETURN
24 ON ERROR GOTO 27: C=EXP(A)*.5
25 IF A<11.89835489 C=C+.25/5
26 RETURN
27 C=EXP(A-BB)*.8.25B181275E+37:RETURN

```

```

31 LABEL "TANH"
32 T=X:RETURN
33 IF T<.0000272705983 T=X:RETURN
34 IF T<11.89835489 T=SGN(X):RETURN
35 IF T<.3465735983 T=T:T
36 T=(1+T)/(3+T/5+T*.142633): RETURN
37 T=EXP(T+T): T=(T-1)/(T+1)XSGN(X):
RETURN

```

Slika 6 pokazuje da se formula

$$\text{th}(x) = \text{sgn}(x) \cdot (t-1) / (t+1),$$

ge je $t = \exp(2 \text{abs}(x))$, ne može koristiti za $\text{abs}(x)$ manje od $(1/2) \log(2)$. Za $\text{abs}(x)$ manje od 2^4 ne dobija se nijedna tačna cifra rezultata. Ta formula nije upotrebljiva ni za

$$\text{abs}(x) > (1/2) \log(2)$$

zbog prekoračenja opsega brojeva (overflow). To je razlog što se računanje $\text{th}(x)$ ne svodi samo na $\text{sh}(x) / \text{ch}(x)$.

Ako je J broj bitova mantise, onda je za

$$\text{abs}(x) > (J/2) \log(2) = 16 \log(2)$$

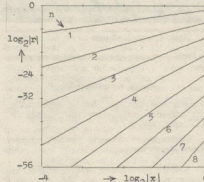
dovoljna formula

$$\text{th}(x) = \text{sgn}(x).$$

Za $\text{abs}(x) < (1/2) \log(2)$ treba naći pogodnu formulu. Za $\text{abs}(x) < \pi/2$ važi potencijalni razvoj

$$\text{th}(x) = \sum_{k=1}^{+\infty} 2^{2k} (2^{2k-1}) B_{2k} x^{2k-1} / (2k)!.$$

gde su B_k Bernoullijevi brojevi: $B_1=1$, $B_2=-1/2$, $B_3=1/6$, $B_4=1/30$, $B_5=-1/42$, $B_6=1/30$, $B_7=-5/66$, ..., $B_{32}=0$ za prirodan broj p . Slika 7 pokazuje da je za $\text{abs}(x) < (1/2) \log(2)$ i $J=32$ potrebno uzeti prvih 7 članova razvoja.



Slika 9. Relativna greška razvoja

$$\text{th}(x) \approx \frac{x}{1+} \frac{x^2}{3+} \frac{x^2}{5+} \dots \frac{x^2}{2n-1}$$

Za $\text{abs}(x) < \pi$ važi razvoj

$$\text{th}(x) = x / \left(\sum_{k=1}^{+\infty} 2^{2k} B_{2k} x^{2k} / (2k)! \right).$$

Slika 8 pokazuje da je za $\text{abs}(x) < (1/2) \log(2)$ i $J=32$ potrebno uzeti prvih 5 članova razvoja. Broj članova razvoja može biti i 4 ako se izvrši ekonomizacija.

Za proizvoljno realno x važi verizni razvoj

$$\text{th}(x) = \frac{x}{1+} \frac{x^2}{3+} \frac{x^2}{5+} \dots \frac{x^2}{2n-1}$$

Slika 9 pokazuje da je za $\text{abs}(x) < (1/2) \log(2)$ i $J=32$ dovoljno uzeti verizni razlomak reda $n=5$.

Ako se primeni ekonomizacija, ovi razvoji za isti opseg argumenta i isti broj bitova mantise rezultata J mogu imati i niži stepen (naravno uz izmenu vrednosti koeficijenta). Tako je dobijena formula

$$\text{th}(x) = x / (1 + (3+1)/(5+1 - 0.1426)),$$

gde je $t = x^2$.

Ako je $J=32$ za

$$\text{abs}(x) < (3 \cdot 2^{47})^{1/3} = 0.0000277$$

dovoljna je formula

$$\text{th}(x) = x.$$

Priloženi su programi za izračunavanje vrednosti hiperboličkih funkcija za računar Sharp MZ-700 (HuBasic). Ako čitalac ima „ZX spektum“ ili „komodore“ lako će ubaciti reči LET ili THEN tamo gde su potrebne.

Na osnovu svega izloženog jasno je da je dug put od poznavanja programskog jezika do valjanog programa. Taj put vodi kroz teško prohodni teren koji se zove numerička matematika. Već za tako jednostavne funkcije kao što su hiperboličke funkcije bilo je neophodno kombinovati više formula kako bi se za svaku vrednost argumenta obezbedila tačnost rezultata do poslednjeg bita mantise.

Dušan Slavić

Računari u poslovnoj primeni (3)

zalihe na disketi

Kako postaviti sistem za obradu podataka o zalihama koji omogućuju kako kontrolu nivoa stanja pojedinih artikala na zalihama tako i upravljanje zalihama? Ovakvo koncipiran zahtev nam omogućuje da spojimo u rešenje tri vrste zaliha: sirovine, potrošni materijal i rezervne delove. Ove tri vrste zaliha imaju skoro identične sadržaje, tako da ih je moguće rešiti praktično na identičan način. Jasno je, pri tome, da na personalnom računaru, zbog njegovih ograničenih mogućnosti, ove tri vrste zaliha treba razdvojiti u tri nezavisne obrade.

Kao i u prethodnim slučajevima, obradu treba izvesti na računaru koji ima 64 K centralne memorije, serijski štampač i dve disketne jedinice.

ANALIZA ZAHTEVA

Da bi se dao odgovor na postavljena pitanja, kada i koliko naručiti, potrebno je izvršiti analizu pojedinih uticaja na same zalihe. Tako, na primer pri analizi za odgovor koliko naručiti potrebno je razmotriti nekoliko međusobno protivrečnih činioca:

- troškove održavanja zaliha,
- troškove naručivanja,
- troškove izazvane kupovinom manjih količina ako se daje niža cena za nabavku kod istog dobavljača (bilo po broju kupljenih komada bilo za iznos ukupne nabavke), i
- troškove zbog gubitaka izazvanih zastarevanjem, kvarenjem ili zbog promene cene.

Za analizu ekonomične količine za nabavku na raspolaganju je više metoda, pri čemu one same nisu predmet našeg članka. Međutim, za rad na sistemu je važno da se ta količina unese u sistem kako bi se mogao pojednostaviti proces same nabavke.

S prethodnom analizom je povezano i pitanje kada izvršiti nabavku. Pri tome se mora imati na umu da takozvano vreme ispunjavanja zaliha teče od trenutka kada se da informacija da treba naručiti pa sve do trenutka kada artikal stigne u magacin. Postoji više različitih metoda za određivanje trenutka naručivanja:

- unapred utvrđeni nivo ponovnog naručivanja,
- utvrđeno vreme pregleda,
- grupe narudžbe,
- prave potrebe,
- rezerve,

ili kombinacija dve ili više metoda. U našim uslovima se, uglavnom, primenjuju prve dve metode, pa ćemo ukratko objasniti i parametre koji su kod njih korišćeni.

Pri primeni metode unapred utvrđenog nivoa naručivanja parametri su nivo maksimalnih zaliha i nivo zalihe pri kojoj se vrši naručivanje. Ovaj nivo se određuje na osnovu planirane potrošnje tog artikla za vreme dok traje proces popunjavanja zalihe u skladištu. Pri tome nivo zaliha nikad ne bi trebalo da padne ispod dozvoljenog minimuma (rezervne zalihe).

Kod metode utvrđenog vremena pregleda stanja zaliha parametri su nivo maksimalnih zaliha i utvrđeni vremenski interval kada se kontrolise nivo zaliha (na primer, nedeljno ili mesečno). Za razliku od prve metode, u kojoj je naručena količina uvek ista (razlika između maksimalne količine i nivoa za ponovno naručivanje), a vremenski interval naručivanja različit, kod ove metode je naručena količina uvek različit, ali se naručivanje vrši u jednakim vremenskim intervalima.

Kada je izabrana metoda kontrole zaliha, i definisana dva parametra koji je karakterišu, potrebno je odgovoriti i na pitanje kako voditi cene kod artikla koji je nabavljen sa različitim vrednostima. Moguća su dva rešenja: svodenje cene na poslednju nabavnu cenu ili vođenje višestrukih cena za isti artikal. Teorijski, postoji i metoda prosečne cene, ali se ona kod nas praktično ne primenjuje. Od izbora načina iskazivanja cene artikla zavisi i način obrade podataka. Naime, u slučaju jedinstvenih cena potrebno je imati program koji će pri svakoj novoj nabavci da izvrši izmenu cene kod artikla koji je nabavljen, te da se odgovarajući kumulativni zbir razlika u ceni usmeri na odgovarajući konto. U slučaju višestrukih cena, ovaj program nam nije potreban, ali je zato potrebno šifru artikla na zalihama povećati za jednu cifru na kraju (ako se želi pratiti do 10 različitih cena po artiklu). Na primer, osnovna šifra artikla je 05674325, pri čemu šifra 05674325 označava petu cenu, a 05674328 osmu cenu za isti artikal. Ovakav izbor rešenja zahteva nešto složenije programe za obradu, ali je u potpunosti ostvarljiv na zadanoj konfiguraciji. Izbor ove metode, međutim, povlači sa sobom znatno povećanje prostora na disketama. O ovome treba posebno voditi računa u slučaju kada se nabavne cene često menjaju. Rešenje se može naći u periodičnom „čišćenju“ datoteke artikala od slogova sa stanjem nula i odgovarajućim pomeranjem šifara.

Konačno, u zahtevu nije ništa rečeno da li se želi praćenje samo nivoa zaliha ili je potrebno pratiti i analitiku promena po svakom pojednom artiklu. Ako se želi praćenje samo nivoa zaliha u skladištu, posao je u mnogome olakšan — kako sa stanovišta obrade tako i sa stanovišta potrebnog prostora na disketama.

Datoteke artikala

U našoj obradi postoje dve osnovne datoteke, i to datoteka artikala i datoteka dobavljača. U slučaju da je potrebno preče-

nje i analitika promena po pojedinim artiklima, poželjno je, radi uštede prostora u osnovnoj datoteci artikala, formirati posebnu datoteku naziva artikala. Ovo će, sa svoje strane, da povećava vreme same obrade, ali će zbog uštede u nazivu artikla (koji je izbačen iz osnovne datoteke) omogućiti praćenje većeg broja promena unutar jednog sloga.

Datoteka artikala bi trebalo da sadrži bar:

- šifru artikla (uključujući i jedan znak za varijantu cene i eventualno, ako se prati i analitika promena, i jedan znak za redni broj sloga),
- naziv artikla,
- šifru dobavljača,
- maksimalno dozvoljeni nivo zalihe,
- minimalno dozvoljenu količinu,
- nivo zalihe pri kome se vrši naručivanje, odnosno datum (na primer svaki 10-ti dan ili 30-dan u mesecu) zavisi od izbora metode za naručivanje,
- trenutni nivo zalihe,
- jediničnu cenu,
- jedinicu mere (šifrirano),
- naručenu količinu,
- datum poslednjeg izlaza artikla iz skladišta, i
- rezervisanu količinu.

Ovim pregledom bi se obuhvatio minimalan broj podataka u slogu za zalihe, ako nam nije potrebna analitika promena. Ukoliko se, pak, želi i analitika promena, poželjno je formirati novu datoteku koja bi sadržala šifru artikla, naziv artikla i šifru dobavljača, kako bi se u slogu oslobodio prostor za podatke o analitici. Takođe je moguće i formirati novu datoteku, koja bi sadržala same promene, a osnovna datoteka ostala kao što je već navedeno. Izbor između ove dve mogućnosti je, u suštini, izbor između brzine vremena obrade i raspoloživog prostora na disketama. Verovatno, u većini slučajeva čemo žrtvovati vreme obrade, jer smo ipak sa prostorom na disketi jako ograničeni.

Analitika promena

Podaci koje bi bilo poželjno pratiti u analitici promena su:

- datum promene,
- iznos promene i
- šifru promene

Kako se ovi podaci javljaju u jednakim razmacima unutar sloga, to je potrebno uvesti i indikator koji ukazuje na prvi slobodan element za promenu koja se očekuje.

Ako uz prva tri podatka za svaku promenu dodamo i podatak

- mesto troška (ili broj radnog naloga po kome je roba izdata),

Dva do sada prikazana primera primene personalnih računara u radnoj organizaciji mogu čitaoca navesti na pomisao da se ovakvi računari primenjuju samo na nekim „sporednim“ poslovima, koji nisu vezani neposredno za proces proizvodnje. U primeru koji opisujemo u ovom broju, kontrola i upravljanje zalihama, pokušaćemo da prikazemo mogućnost primene u oblasti koja ima veoma veliki uticaj na sam proces proizvodnje. Pri tome se necemo obazirati na izvesne razlike, koje svakako postoje, uslovljene različitim vrstama zaliha — sirovina, polufabrikati, gotovi proizvodi, ambalaza, potrošni materijal, rezervni delovi, itd — već ćemo se zadržati samo na opštem opisu kontrole i upravljanja zalihama, a zainteresovani čitalac će svakako moći da, uz malo imaginacije, postavi i svoj problem u okvir prikazanog rešenja.

moгуće je pratiti i odgovarajuće troškove na željenim mestima u proizvodnji.

Dodajući još i šifru artikla, dobićemo i konačan sadržaj datoteke promena.

Jasno je da ove datoteke moraju biti organizovane kao indeksne datoteke, kako bi nam bilo omogućeno da pristupimo svakom artiklu po želji. Osim toga, ovako organizovane datoteke nam omogućuju da bez većih problema olačavamo manje slogeve u jedinstvene celine.

Osnovni problem koji se ovde javlja je veličina samih datoteka, odnosno ograničen kapacitet koji imamo na disketama. U slučajevima kada radimo sa oko 2000 artikala, verovatno da ne bi trebalo očekivati nedostatak prostora na jedinicama sa direktnim pristupom, ali ukoliko imamo veći broj artikala i, uz to, želimo da radimo i analitiku promena, verovatno da ćemo datoteku artikala morati da proširimo na više diska. Ovo, sa svoje strane, uglavnom utiče na komfornost rada, ali ne i na mogućnosti koje su nam na raspolaganju. Pri tome, svakako, ne mislimo na količine koje se mere desetina hiljada artikala. Ukoliko to i jeste slučaj, moguće je artikle podeliti ili po računopolažacima ili po vrstama roba (na primer, metalna, elektro, proizvodi od gume i slično), pa ih tako razvrstane postaviti na različite diskete ili, pak, na više malih računara. Ulaganja uglavnom nisu velika, ali su koristi u odnosu na ručni rad nemerljive.

Upravljanje zalihama

Na ovaj način formirani sadržaji datoteka nam omogućuju dobijanje većeg broja izveštaja koji se mogu koristiti u procesu kontrole i upravljanja zalihama. Bez ulaznja u opise pojedinih rezultata obrade, navešćemo samo neke izveštaje koji se mogu dobiti:

- ABC klasifikacija robe na zalihama,
- popisne liste,
- izveštaj o artiklima koji stoje određeno vreme u magacinu a nisu trebovani,
- izveštaji o zalihama koje su ispod minimuma,
- pregled vrednosti zaliha,
- izveštaji o potražnji pojedinih artikala,
- izveštaj o koeficijentima obrta pojedinih artikala,
- izdavanje naloga za nabavku,
- praćenje i prijem naručene robe,
- praćenje troškova po mestu troška ili radnom nalogu,
- dobijanje analitičke kartice promena za svaki artikal,
- materijalni bilansi za buduću proizvodnju (u posebnom procesu obrade).



— Bili su u pravu! Kompjuteri zaista olakšavaju vođenje inventara!

Većina ovih izveštaja se dobija sekvencijalnim čitanjem datoteke artikala, tako da je procena vremena obrade za izradu jednog izveštaja teško ostvarljiva bez prethodnog poznavanja veličine same datoteke. Medutim, uzimajući u obzir da je štampač usko grlo našeg sistema, realno je očekivati da bi vreme za izradu pojedinih izveštaja, uključujući i vreme potrebno za manipulisanje sa disketama, bilo reda veličine desetak minuta. Pri tome se izuzimaju izveštaji kao što su popisne liste ili, pak, izveštaj o koeficijentima obrta za sve artikle u skladištu. Poseban problem je vreme potrebno za izlistavanje kompletne analitike promena svakog artikla. Ovo vreme može biti tako dugo da realno, za jako velike datoteke i veliki broj promena, može biti neprihvatljivo. U svakom slučaju, moguće je računati da je vreme potrebno da se odštampa jedan red, na prosečnom štampaču, oko 1 sekun-

de. S druge strane, izvestan broj izveštaja ili informacija se može dobiti praktično trenutno (na primer, stanje zalihe određenog artikla) ili u toku same obrade ulaznih podataka (prijem naručene robe).

Personalni računari mogu, očigledno, imati svoje mesto i u poslovima vezanim za kontrolu i upravljanje zalihama. Pri tome treba biti obazriv pri postavljanju zahteva ukoliko se radi o velikom broju artikala koje želimo da pratimo i analitički. U tom slučaju je verovatno najpogodnije izvršiti prethodno ABC klasifikaciju, a zatim na sistem postaviti ili samo klasu A ili eventualno i klasu A i B, dok bi klasa C ostala da se i dalje vodi ručno, odnosno na način kao i do sada.



Era CMOS-a

U skladu sa predviđanjima eksperata, komplementarna MOS (CMOS) tehnologija sve više postaje vodeća tehnologija u elektronici 80-tih godina, sa trendom da to oстане i sledeće dekadde. Praktično nema poznatog tipa mikroprocesora ili vrste memorije koji se ne proizvode, kako u izvornoj NMOS, tako i u CMOS verziji.

Z80 u CMOS-u

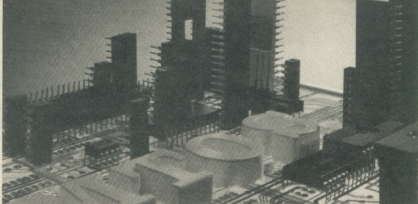
Zanimljivost za korisnike popularnog Z80 mikroprocesora: japanski proizvođači (inače vodeći u CMOS tehnologiji), a pre svih firme TOSHIBA i SHARP, proizvode CMOS verzije Z80 CPU, kao i perifernih kola iz familije (PIO, CTC, SIO a uskoro i DMAC), dok TOSHIBA ima i specijalni CLOCK generator. CMOS Z80 (TOSHIBA oznaka TMPZ84COOP) izrađen C²MOS Silicon Gate tehnologijom je potpuno hardverski (pin-for-pin) i softverski kompatibilan sa izvornim ZILOG Z80 CPU, ali ima prednosti koje inherentno sadrži CMOS: potpuno statički rad i ekstremno niska potrošnja (15 mA na 5 MHz i +5V aktivno, odnosno samo nekoliko mikroampera u HALT-u; idealno za portabl uređaj).

Kompletirana familija

INTEL je kompletirao svoju seriju 16-bitnih mikroprocesora tipom IAPX 286/10 (oznaka 80286) koji, kao i njegovi prethodnici IAPX 86/10 (8086) i IAPX 186 (80186), ima svoj numerički koprocessor (80287), poboljšani kontroler za direktan pristup memoriji (82258 ADMA), LAN (Local Area Network) coprocessor (82586/8), itd; 80286 je mašina sa virtuelnom memorijom (integrirani memory Management Unit — MMU adresira do 1 GByte), pristup fizičkoj memoriji je do 16 MByte, radi sa CLOCK-om od 8 MHz, izvršava supersek 8086 instrukcija i pogodan je za egzekuciju operativnih sistema ili viših programskih jezika. Takođe ga proizvode (tzv. Second Source) i AMD i SIEMENS, tako da zajednički utiru put punom 32-bit CPU koji se uskoro očekuje (80386).

Tridesetdvobitne laste

MOTOROLA je jedna od prvih firmi koja je iznela na tržište puni 32-bit CPU (MC 68020) iz već čuvene 68000 familije, koji uz



standardni 16-bit 68000, zatim MC 68010 (virtuelna mašina) i MC 68008 (8-bit DATA-BUS) čine snažan tim za svaku aplikaciju. U tome mu pomažu periferna kola kao MC 68120/1 IPC, MC68230 Timer, MC68450 DMAC, MC68451 MMU, MC68488 GPIB adapter itd., a sada i MC68681 Dual UART i MC68881 Floating Point Coprocessor (FPU koji odgovara IEEE P754 predlogu). MOTOROLA 680XX seriju po licenci proizvode i SIGNETICS/PHILIPS, MOSTEK, THOMSON, HITACHI i ROCKWELL... Pa ipak, najviše periferija za 68000 ne proizvodi MOTOROLA, koja je originalno napravila 68K CPU (a zajedno sa MOSTEK-om i SIGNETICS-om osmislila tzv. VMEbus), već SIGNETICS koji na programu ima petnaestak LSI kola, od 68154 IGOR do 68920 MAC...

Procesori na bejziku

Onima koji mikroprocesore koriste za kontrolerske aplikacije, a ne žele da se upuštaju u mašinsko programiranje, već izvesno vreme stoje na raspolaganju CPU sa inkorporiranim bejzik interpreterom na čipu (upročeni, tzv. TINY BASIC): ZILOG ima Z8671 (2K ROM, preprogramirana verzija Z8 CPU) a NATIONAL SEMICONDUCTOR INS8073 (4K ROM, preprog. INS8070 CPU), dok NEC pravi UPD7901. Ovi „bejzik interpretari na čipu“ imaju standardne bejzik instrukcije, ali rade sa INTEGER brojevima, imaju sada konkurenciju u INTEL-u (preprogramirana verzija 8052 CPU sa 8K ROM-a ima Floating Point BASIC)...

Brza CMOS kola

Dugo smo koristili standardna TTL kola (74 seriju), brza TTL (Schottky, 74S) i brza sa malom potrošnjom (Low Power Schottky, 74ALS); danas se sve više upotrebljavaju poboljšana brza TTL kola (74AS, sa 40% većom brzinom rada) ili 74ALS serija sa 60% manjom potrošnjom. Međutim, izgleda da će ih u nekoliko sledećih godina potpuno zameniti brza CMOS kola (74HC sa CVOS logičkim nivoom, odn. 74HCT sa TTL nivoom) zbog svojih dobrih karakteristika: brzine rada, koja je otprilike jednaka brzini TTL kola (ranije je to bila glavna mana CMOS tehnologije), cene koštanja (jedna nešto skuplja od ekvivalentnih TTL kola), ekstremno male potrošnje (za nekoliko redova veličine), znatno veće pouzdanosti (posledica niske potrošnje, statističko vreme između otkaže reda stotina, pa i hiljada godina!), širokog opsega napona napajanja, velike ulazne impedanse, simetričnog raspona izlaznog napona (jednak radnom naponu), itd.

Brzi...

ZILOG je znatno snizio cenu brze verzije Z80 CPU (oznaka Z80H) koja radi sa CLOCK-om od 8 MHz, sa nadom da će ovo doprineti boljoj prodaji i široj upotrebi (protivnik je bila visoka cena).

... i modifikovani Z80

AMI proizvodi S83 OSP (Operating System Processor), koji se u suštini sastoji od Z80 CPU plus kontroler za dinamičke RAM-ove, plus 8K ROM (masked) koji ne smeta spoljnoj memoriji (adresibilnost i dalje 64K bajta); interni ROM služi za smeštanje rutina spovnogostog REAL-TIME operativnog sistema.

Eprom sa lozinkom

Iako se današnji ultravioletni EPROMi prave sa kapacitetom od pola megabita (organizacija 64Kx8, i ovde je, naravno, prvi bio INTEL) sa planom za megabit i više, posle invazije CMOS verzija (naročito japanskih firmi, sa maksimalnim kapacitetom od četvrt megabita kod 27C256), INTEL je pripremio i nešto drugačiji odgovor: na tržištu se pojavio prvi 128K KEPPROM (oznaka 27916, KPE EPROM) kojem se može, pristupiti samo ako poznajete šifru zapisa- na u njemu. Ovaj tip memorije sadrži pored EPROM-a, LOCK/KEY deo, generator slučajnih brojeva i deo za šifrovanje.

68000 na japanski način

Firma HITACHI je smestila popularni HD68000 CPU u znatno manje i ekonomičnije kućište tipa PLCC sa 68 pinova (Plastic Leadless Chip Carrier, novi JEDEC standard).

Od jednog do dvesto

Raspon cena modernih mikroprocesora pokazuje zavidnu dinamiku: dok neke verzije mikrokontrolera koštaju ispod jednog dolara (npr. 68XX plastična verzija, OEM količine), INTEL-ov 80286 košta oko 200 US dolara (naravno vojne verzije mikroprocesora visokih performansi — 16/32 bit — prema MIL-STD-883 sa dodatnim testovima koštaju i znatno više).

Mostek pred bankrotstvom

Sveža vest: firma MOSTEK (poznata po dinamičkim memorijama i mikroprocesorima) je bankrotirala, ali se francuski gigant THOMSON—CSF sprema da kupi.

Pripremio: Blažimir P. Miše, dipl. ing.



**UDRUŽENI
PROGRAMERI**

Profi adresar

American Federation of Information Processing Societies, Inc (AFIPS)
Suite 800
1815 N. Lynn Street
Arlington, VA 22209
USA
(702) 558-3600

Association of Computer Users (ACU)
1690 38th Street
P.O. Box 9003
Boulder, CO 80301
USA
Association for Computing Machinery (ACM)
11 W 42nd Street
New York, NY 10036
USA
(212) 869-7440

Association for Educational Data Systems (AEDS)
1201 16th Street NW
Washington, DC 20036
USA
(202) 833-4100

American Institute of Aeronautics and Astronautics (AIAA)
1290 Avenue of the Americas
New York, NY 10104
USA
(212) 581-4300

American Mathematical Society (AMS)
P.O. Box 6248
Providence, RI 02940
USA
(401) 272-9500

American Society for Information Science (ASIS)
1010 Sixteenth Street, NW
Washington, DC 20036
USA
(202) 659-3644

American Statistical Association (ASA)
806 15th Street NW
Washington, DC 20005
USA
(202) 393-3253

Association for Systems Management (ASM)
24587 Bagley Road
Cleveland, OH 44138
USA
(216) 243-6900

Association of Computer Programmers and Analysts (ACPA)
c/o Cuts Corp.
Suite 808
11800 Sunrise Valley Drive
Reston, VA 22091
USA
Association of Data Processing Service Organization (ADAPSO)
Suite 300
1300 N 17th Street
Arlington, VA 22209
USA
(703) 522-5055

U „Računarima 10“ pisali smo o društva programera koja okupljaju profesionalce. Od njih se, razume se, ne mogu dobiti kataloge najnovijih igara za „spektrum“ niti novonik novih dodataka za „komodor“. Oni, odista, svakom programeru koji drži do sebe pružaju mnogo dragocnije — informacije i priliku da unapredi svoje tehnike programiranja. Možete, tako, da posećujete razne konferencije, simpozijume i izložbe koje to društvo organizuje po izuzetno povoljnim uslovima, ili da se pretplatite na računarske časopise po izuzetno povoljnim cenama, da kupujete knjige sa popustima i od 70%, da dobijete stipendiju

od društva za usavršavanje ili da položite određene ispite koji se visoko cene pri zapošljavanju. Za ostale pogodnosti raspitajte se kod samog društva.

Ako se odlučite da nekom od navedenih društava pišete, možete da pripremite pismo u kome drug Sofronije piše Association for Computing Machinery. Naveli smo samo obavezni deo teksta, a na mesto tri zvezdice (***) možete da napišete nešto što će doprineti društvu da stekne bolji uvid u vaše želje i kvalifikacije. Društva su vrlo ekspeditivna i odgovaraju na sva pisma i hakera i akademca.

CAUSE: The Progeessional Association for the Development, Use, and Management of Information Systems in Higher Education
737 29th Street
Boulder, CO 80303
USA
(303) 449-4430

Computer and Automated Systems Association, Society of Manufacturing Engineers (CASA/SME)
One SME Drive
P.O. Box 930
Dearborn, MI 48128
USA

Computer and Business Equipment Manufacturers Association (CBEMA)
1828 L Street NW
Washington, DC 20036
(202) 466-2288

Computer and Communications Industry Association (CCIA)
Suite 512
15500 Wilson Boulevard
Arlington, VA 22209
USA
(703) 524-1360

Association for Computing Machinery
11 W 42nd Street
New York, NY 10036
USA

Dear Sir,
My name is Sofronije Hadži-Jančićević. *** I am very interested in joining the (ime društva). Please send me all information needed to become a member at the above address by air mail.

Computer Law Association, Inc. (CLA)
6106 Lorcom Court
Springfield, VA 22152
USA
(202) 272-2906

Computer Security Institute (CSI)
43 Boston Post Road
Northboro, MA 01532
USA
(617) 393-3663

Conference Board of the Mathematical Sciences (CBMS)
1500 Massachusetts Avenue NW
Washington, DC 20005
USA
(202) 293-1170

Data Processing Management Association (DPMA)
505 Busse Highway
Park Ridge, IL 60068
USA
(312) 825-8124

EDUCOM, The Interuniversity Communications Council, Inc.
P.O. Box 364
Princeton, NJ 08540
USA
(609) 734-1915

Independent Computer Consultants Association (ICCA)
P.O. Box 27412
St. Louis, MO 63141
USA
(314) 567-9708

Institute for Certification of Computer Professionals (ICCP)
35 East Wacker Drive
Chicago, IL 60601
USA
(312) 782-9437

Institute of Electrical and Electronic Engineers (IEEE) Computer Society
Suite 201
1109 Spring Street
Silver Spring, MD 20910
USA
(301) 589-3386

Institute of Internal Auditors (IIA)
249 Maitland Avenue
Altamont Springs, FL 32701
USA
(305) 830-7600

Instrument Society of America (ISA)
P.O. Box 12277
76 Alexander Drive
Research Triangle Park, NC 27709
USA
(919) 549-8411

Sofronije Hadži-Jančićević
Medvedgradska 20
21000 Novi Sad
Yugoslavia
January 15, 1985

Sincerely,
(potpis)
Sofronije Hadži-Jančićević

International Federation for Information Processing (IFIP)
IFIP Secretariat
3 Rue du Marche
CH-1204 Geneva
Switzerland

International Information/Word Processing Association (IIWPA)
1015 North York Road
Willow Grove, PA 19090
USA
(215) 657-6300

Mathematical Association of America (MAA)
1529 Eighteenth Street, NW
Washington, DC 20036
USA
(202) 387-5200

National Computer Graphics Association, Inc (NCGA)
Suite 801
8401 Arlington Blvd.
Fairfax, VA 22031
USA

National Science Foundation (NSF)
Division of Mathematical and Computer Sciences
Room 339
1800 G Street NW
Washington, DC 20550
USA
(202) 632-5960

mali oglasi

OBRADUJTE SVOJ RAČUNARI! OBJAVITE MU MALI OGLAS U „RAČUNARIMA“!

Spektrum

● **SPEKTRUM** — Rečnik, englesko-srpskohrvatski (oko 1400 najčešće upotrebljivanih reči) + kasetna + poštarina. (1300 din.) Tel. (011) 497-862 od 17-19 h. Dragan Marjanović, 11000 Beograd, B. Jerković 123/36

● **Spektrumovci!** Professional software vam nudi sve što vam je potrebno. Posedujemo najbolje strane i sve nove programe (Impossible Mission, Hacker, WS Basketball...), Tražite besplatan katalog. Uverite se da su nam cene među najjeftinijim u Jugoslaviji. Novi programi stalno pristižu. Professional software. Nučičeva 1, 15000 Sabac, tel. 015/24-734

● **P. N. P. electronic HARDVER ZA SPEKTRUM POVOLJNO! PRODAJEMO KEMPTON JEDNOSTRUKI I DVOSTRUKI INTERFEJS ZA PALICU, PALICE, I/O INTERFEJS, A/D, D/A KONVERTORE, SVJETLOSNU OLOVKU, EPROM PROGRAMATOR, MEGAROM, AUDIO POJAČALO, PREPRAVLJENI ROM ZA SPEKTRUM, PROGRAMIRAMO EPROME, POPRAVLJAMO SPEKTRUM I KOMODOR, JOŠ MNOGO TOGA. BESPLATAN KATALOG. PNP electronic Jeretova 12 Split**

● **ZA SPEKTRUM** prodajem programe za radio-amatere: ERTY, SSTV (rad bez interfejsa) ORB, QTH, CV, KEY BOARD i mnogo toga korisno radio-amaterima. Prodajem najnovije programe za igru (dečembarski hitovi u Engleskoj). Branković Rade, P. Box 37, 12240 Kučevci

● **SPEKTRUM** — OBAVEZAN PRILUČNIK ZA POČETNIKE I NAPREDNE: BASIC PROGRAMIRANJE I brošura „UVOD“. Do sada nemađnađe kvalitete! DUŠKO BJELOTOMIĆ, CENTAR 1, 5455 VALPOVO, tel. 054/82-865 ili 041/683-141 3 cm

● **Prodajem programe za Spektrum.** Komplet od 14 programa 700 din. + vaša kasetna. Sa mojom kasetom 1000 din. Milan Vujanak, Mrkovička 50/8 11000 Beograd, tel. 535-532

● **Najbolja ponuda za Spektrum:** povoljne cene novih programa, najizjece beskrpti i uputstva. Popusti, besplatan spisak. Sinadinović Dragan, Gundulićeva 12, 34300 Arandelovac, tel. 034/714-948 3 cm

● **Besmrtni poke katalog za Spektrum.** Preko 100 pokova za igre — 200 d. Sada možete igrati do kraja. Toti Tibor, B. Dejanović 1, 21205 Sremski Karlovci, tel. 021/881-650

● **SPEKTRUM — SERVIS** Servisiranjem svih karovera na Spektrum za samo 3300 d. Cena je fiksna, a u nju su uručani i delovi. Spektrum možete poslati poštom. Rok seviriranja je 24 časa. Bečarovski Aco, Grijid k. 12/40, 91000 Skopje, tel. 091/239-551

● **Super Turbo** li ubrzava učitavanje — snimanje, omogućava preinamiranje programa zaštićenih ubrzanim učitavanjem. (Program, kasetna, uputstva — 1000 d.) Komplet najnovijih programa 800 d. Tražite katalog. Krtić Aleksandar, Bažakova 30, 21000 Novi Sad

● **SPEKTRUM: UBEDLJIVO NAJJEFTINIJU PAKETI NA YU TRŽIŠTU — ZAHTEVAJTE NOV BESPLATAN KATALOG SA PREKO 1000 PROGRAMA.** Radulović Rade, Vozački put 10, 61000 Ljubljana, tel. 061/225-588

● **SPEKTRUM: Profesionalni predovi: MASINAC ZA POČETNIKE (1000), DISASEMBLIRANI ROM (1200), NAPREDNI MASINAC (1300), KOMPLET (3100).** Prevedena uputstva za prevedene programe: DEVPAC, MEGA BASIC, BETA BASIC 1.4, ARTIST, MELBOURNE DRAW, QUILL, MONITOR DISASSEMBLER, EDITOR ASEMBLER, FIFT, LEONARDO. Pojedinačna uputstva 500, snimljeni programi (200). Bilo kojih pet uputstava + snimljeni programi (2500). Isporuka za 24 časa. KOMPLITER BILJKA, Filipa Filipovića 41, 32000 ČAČAK, tel. 032/31-20

● **VRHUNSKI GRADEVINSKI PROGRAMI ZA ZPEKTRUM:** OKVIRI, REŠETKE, ROŠTILJI, PILOTI, DIMENZIONIRANJE, VODOVOD I mnogi drugi. Za radne organizacije i pojedince. Besplatan katalog. Gino Gracin, Kosala 17, 51000 RIJEKA, tel. 051/517-291

● **Svakog meseca najnoviji ZX Spektrum programi u polusnatim kompletima (po šest programa) za samo 160 dinara.** Tražite katalog za 800 programa. Marko Marković, Dž. Bijićeva 27A/XI, 71000 Sarajevo, tel. 525-512

● **Prodajem programe za Spektrum.** Jedan (bilo koji) program košta 50 dinara. Na svakih pet kupljenih programa jedan dajem kao poklon. Pišite za besplatan katalog Janković Silard, Petefi Sandora 84, 25222 Telečka

● **KEMPTON ITERFACE (KEMPY) REDOSTIK PALICA (REDY) CIJENA KOMPLETA — 12.000 din.**

● **SPEKTRUMOVCI!** I u Novoj godini — najnoviji i najkvalitetniji kompleti sa dvadeset programa po 1000 dinara plus kasetna, uz velike popuste za veće narudžbe. Komplet 21: Popeye, W.S. Basketball, Superipeline 2... Komplet 22: Winter sports, Commando, Robin of the Wood... komplet 23: Saboteur, Yie Ar Kung fu, Impossible Mission... Tražite katalog s opisima (50 d.) sa naznakom „ZA Spektrum“. Snimke profesionalne, isporuka ekspresna! FUTURE ORION, 41000 ZAGREB, Rubetićeva 7, tel. 041/417-052. **POSEBAN PROGRAM ZA ODRASLE:** Sex Mission Zajedno sa kasetom i troškovima otplate 950 d.

● **SPEKTRUMOVCI,** novi hit-programi (48k). Komplet 7: Skool Daze 2, Beach Head 2, Boulder Dash 2, International karate 1, International karate 2, impossible Mission, Sorcerons Shadow, Dynamite Dan, Monty on the Run, Exploding Fist, Dunderach, Frenkie. Komplet + kasetna + poštarina, 1400 d. Garantovan kvalitet i brza isporuka. Narudžbine 800 d. Sve najnovije Srdar. Oslobođenja II deo 6, 11194 Beograd — Rušanji. ili na tel. 011/888-222

DATA

● **SPEKTRUM — AMSTRAD — SCHNEIDER** Programski paketi za stručnjake: STATIST LINSET GODEET NEW DATA, D. Brašovana 8/10, 21000 Novi Sad **SCHNEIDER — AMSTRAD — SPEKTRUM**

● **SPEKTRUM RAINBOW SOFTWARE** vam nudi izbor od preko 1500 programa. Kompleti od 25 programa 800 d. Sve najnovije programe koje drugi nude kod nas možete dobiti za samo 100 d. Svake nedelje novi komplet. Rainbow software snima direktno iz računara. Katalog je besplatan. Mihajlović Kirčo, Moše Pijade 128, 91300 Kumanovo, tel. 091/23-800

● **ZPS SOFTWARE** — još prošle godine dobili smo ove hitove za Spektruma: COMMANDO, ROLLER COASTER, SABOTEUR, NEVER ENDING STORY, BACK TO SKOOL, BEACH HEAD 2, WINTER SPORTS, ROBIN ON THE WOOD, YIE AR KUNG FU i još mnogo drugih. Informacije o najnovijim programima na tel. 041/670-071, Berisav Vahčić, Viktora Kabarića 36, 41020 Zagreb

● **EPROM PROGRAMATOR, MEGAROM, LAJT PAM, PRINTER INTERFEJSI I JOŠ MNOGO TOGA** P.N.P. electronic Jeretova 12 Split

Ako ne možete da podnesete da drugi nemaju ono što vi nemate, javite se na neki od malih oglasa u „Računarima“.

Ako ne volite da se dopisujete sa „Računarima“ svoj mali oglas možete nam izdiktirati preko telefona 011/650-161 svakog radnog dana od 10—14 sati.

Prva stvar koju treba da uradite je da se odlučite da li želite običan ili uokviren mali oglas. **CENA OBIČNOG MALOG OGLASA** do dvadeset reči je 600 dinara. Svaka naredna reč košta 40 dinara, s tim što oglas ne sme da ima više od 50 reči. Adresa oglašivača ne računa se u cenu.

CENA UOKVIRENOG MALOG OGLASA je 600 dinara po visinskom centimetru, s tim što se mogu zakupiti najmanje po 32 slova znaka. Ako se ne iskoristi čitav prostor u jednom redu, računa se broj redova a ne broj znakova. Za uokvirene oglase preko 5 cm cena je 900 dinara po centimetru.

Poželjno je da vaš mali oglas počinje sa Prodajem, Kupujem, Držim časove, Mupljenj... ili nečim sličnim što ukazuje na sadržaj oglasa. Svi mali oglasi koji stignu u našu redakciju do 30. januara do 12 sati biće objavljeni u „Računarima“ broj 12 koji izlazi iz štampa polovinom februara.

Da ne bi bilo zabune, obavezno naznačite da li želite običan ili uokviren mali oglas, i zajedno sa tekstom vašeg malog oglasa pošaljite i priznanicu o uplati na adresu redakcije: „GALAKSIJA“, BULEVAR VOJVODE MIŠIĆA 17, „ZA MALE OGLASE“.

Vaš mali oglas možete nam izdiktirati preko telefona 011/650-161, a mi ćemo vam onda naknadno poslati ispunjenu uplatnicu.

„RAČUNARI“ OBJAVLJUJU NAJJEFTINJE RAČUNARSKE MALE OGLASE!!!

- Spectrum news:
 1. COMMANDO (Elite), arkanid rat.
 2. ROLLER COASTER (Elite), juna park...
 3. YIE AR KUNG FU (Imagine), kung fu II...
 4. RASPUTIN (Firebird), bolji od knight lora.
 5. B.C. QUEST (Software project), konačno na Spectrumu.
- Na miravino I: BEACH HEAD II, BOULDER DASH II, SABOTHEUR, INTERNATIONAL KARATE. Kao što vidite MasterClub je izabrao za Vas samo najbolje. Ako vas zanimaju apsolutni noviteti, pogledajte što smo pripremili za Vas u januarском broju MOG MIKRA. Novosti za Vas SPECTRUM i COMMODORE 64. Obavezno 50 nd. za katalog.

MASTERCLUB, ZAGREBAČKA 21, RIJEKA, 051/37-545

SPECTRUM SOFTWARE STUDIO

- izbor od preko 1000 programa
- svaki program sa uputstvom
- veliki izbor literature — knjiga i originalnih programskih uputstava na engleskom i srpskohrvatskom jeziku
- spisak programa je besplatna, za katalog sa opisom posila 1200 d.
- brza i kvalitetna usluga — proverite!

PAJNIC MIRKO
STRAHINIĆ BANA 56, 11000 BEOGRAD
 Tel. 011/188-190 posle 15 časova

• Spectrumovci! Najmoćnije logaritamske tablice od sada i za vaš kompjuter. Kasete sa programom + kompletna uputstva za samo 700 din. Logos Soft — Vujčić Zoran, Ratka Vujović 2/XII, 11090 Beograd, tel. 011/532-308

• Spectrumovci! Povoljno prodajemo najnovije programe pojedinačno i u kompletu. Najnoviji super komplet sa 12 hit programa: INTERNATIONAL KARATE, NEVER ENDING STORY, WRIGGLER, SUPER BRAT (Match Point 2). Nerlović Srdan, Braće Santini 54, 58000 Split, tel. 058/512-640

• Spectrumovci! Mega prilika! Sve one igre za koje kažu da su najbolje u 1985. (Računari 10) osim Gyrona, D. Busters, DT Dechation, ali sa SEX MISSION, IMPOSSIBLE MISSION, FIGHTING WARRIOR i INTERNATIONAL BASKETBALL prodajemo za samo 800 din. + kasete. Križevci, Rađe Vena 5, 43260 Mrkopalje, tel. 043/841-782

• Spectrumovci! Nudimo širok izbor najnoviji superhitova u polučasovnim kompletima po izboru. WINTER GAMES, INTERNATIONAL BASKETBALL, SABOTER... čekaju na vas. Tražite besplatne kataloge. Tel. 345-844

Comet
SOFTWARE

Sve za ZX SPECTRUM — video igre — namenski programi — programska uputstva — literatura — SPECIJALNO — BETA BASIC 3.0 sa kompletnim originalnim uputstvom, kasetom i poštarinom za samo 1000.-d.

Za sve usluge garancijal MILOVANOVIC LJUBIŠA Petra Lekovića 57, 11030 BEOGRAD
 tel: 011/558-007 posle 17 časova

• CHALLENGER SOFTWARE — za Spectrum prodajemo IMPOSSIBLE MISSION (program godine), BEACH HEAD 2 (US Gold), BOULDER DASH 2 (nezastavljivo padanje stene), MACADAM BUMPER (filper po meri), SKOOL DAZE 2 (izvršno), WARRIOR (exploding fist 2) + još šest super hitova. Samo 600 din. Čeda Nedeljković, XI kraljevske divizije 61 11000 Beograd, tel. 011/538-812

• EMP SOFTWARE vam predstavlja najnovije igre engleskih top lista za Spectrum (izvršni BASKETBALL, RED ARROWS). Za svaki komplet programa dobijate objašnjenje u obliku kataloga (besplatno) + besplatni katalog. Koser Igor, Augusta Cesarica 4, 54400 Čakovo, tel. 054/841-812

• Izaberite si najnovije programe za ZX Spectrum. Za više od 20 programa dobijate popust od 10%. Programi se prodaju pojedinačno i u kompletima. Tražite besplatni katalog. Tel. 041/686-747 i 041/672-051

• SPECTRUMOVCI! Nervira vas sporo učitavanje TURBO TAPE za SPECTRUM u dvije verzije. Prva ubrzava učitavanje i animiranje na 3000 bauda i jednostavnija je za rukovanje, dok druga verzija radi na 3600 i 7200 bauda. Cijena pojedinačne verzije je 1000 din. Ako kupite obe, cijena tada iznosi 1700 din. Tel. 041/686-747

• Ako još ne posedujete generator zvuka za vaš ZX Spectrum, ovo je idealna prilika da ga nabavite! Cena 15000 dinara. Dražen Abramović, Bjelovarskog odreda 8, 41020 Zagreb.

• Spectrumovci, komplet 17 — komplet snova: Popeye, W. S. Basketball, Mout on the run, Tiranaog 3, Fahrenheit, S. Pipeline 2, Dynamite Dan, A. S. Profanation, Red Arrows, Drive in, T. T. Spirit, Day in life, Confusion... Sve to za 700 din. + kasete. Posedujem i moćno stariji komplet Perić Nenad, Braće Mladinovi 12, 37000 Kruševac, tel. 037/33-510

Spectrumovce najvećeg neprijatelja — prepiravanje možete savladati ako naručite precizne upute za samogradnju stabilizatora koji će produžiti život vašem ljubimcu. Cena prava sitnica. Branko Vukasović, Rave Janjčević, 6, 71000 Sarajevo

• Svakog meseca najnoviji ZX Spectrum programi u polusatnim kompletima (po šest programa) za samo 160 dinara. Tražite katalog sa 700 programima. Marko Marković, Dž. Blijedića 27A/XI, 71000 Sarajevo, tel. 525-212

• Prodajem ZX SPECTRUM 16 K sa kompletnim priborom i priručnikom na njemačkom. Cijena 50.000,00 dinara. Zoran Kahrić, Virovitica, tel. 046/721-523

• Spectrum — multisoft uvek sa vama: Exploding fist, D. T. Super-test, Pentagram ltd. Dragan Jovanović, Dragoslava Jovanovića 12, 37000 Kruševac, tel. 037/30-568

• Spectrum Rainbow Software vam nudi super najnovije programe po najpovoljnijoj ceni. Izbor od preko 1500 programa. Kompleti od 25 programa 2300 din. Katalog besplatno. Mihajlović Kišo, Moše Pijade 128, 91300 Kumanovo, tel. 0901/23-800

• NAJLEPŠIJE!!! BEZ KONKURENCIJE!!! SPECTRUM — Komplet od 170 programa — 1,700 din. na vašim, ili 3,000 din. na moje 4 kasete. Savinovski Saša, Gajeva 4, 43400 Virovitica

• SPECTRUM — Najbolji i najnoviji programi u kompletima od 14—20 programa — 800 d. na vašoj ili 1000 d. na mojoj kaseti. Savinovski Saša, Gajeva 4, 43400 Virovitica

• Sretnu Novu godinu želi vam El Toro Software! Udite u ženi godinu sa najboljim programima za ZX Spectrum: Winter Games, Yie are kung fu, Back to skool, Beta basic 3.0, International karate. Pišite za katalog sa slikama i opisima programa na adresu: El Toro, ul. Korrada Babnika 24, 61210 Ljubljana

• Prodajem nov ZX-Spectrum 48K sa svim priborom, kasetom i uputstvima, te nov ULA-čip. Zvati od 16—19h. Tel. 024/35-523

• Specijalni novogodišnji popust! Devpac 3M, Beta bejzik, Mega bejzik za svega 500 dinara. Vučić Vladimir, Buč. oktobarske revolucije 7—11, 34000 Kragujevac

• SPECTRUMOVCI!!! KOMPLET 10: Fourth protocol (3 programa), Hacker, Marsport, Red Arrows, M. A. Bumper, Bored of Rings (4 programa) Shadowfire. KOMPLET 11: Bounty Bob Strikes Back, Match Fishing, Beach Head 2, Super Brat (Match Point 2), Abu Simbel, Neve-

ending Story (3 programa), International Karate 1, 2, Rats, Freeman Cena jednog kompleta: 800 din. Radović Branislav, Sonje Marinković 14/4 21000 Novi Sad, tel. 021/28-682 ili 022/424-824 (vikendom)

• SPECTRUMOVCI... KOMPLET 12: Fairlight, Cyclon Attack, Clusdo, J.S. Willy 3, Mountains of Kat, Thunder Hawks, Chambers of Horrors, Grid Bomb, Lords of Time, Back to Skool, Highrise Harry, Am. Football. KOMPLET 13: Chaos, Covenant, G.G. Cricket, Talos, 3D Vortex, Soho Sex Quest, S.H. Quiz, Aquirmy Wormy, Dragon Fire, Arhon, S. Bells, Dodge City, Radović Branislav, Sonje Marinković 14/4, 21000 Novi Sad, tel. 021/28-682 ili 022/424-824 (vikendom)

• Komodor 64 — hit programi za kasete: HACKER, S.V.S. 3, ROAD, RACE, KNIGHT LORE, TOUR THE FRANCE. Specijalni popusti: Moguća razmena, Perkićević Ojani, Milana Račića 28, 11000 Beograd, tel. 011/424-744

• Prodajem paket programe za Komodor 64. Cena po dogovoru. Javi se na adresu: Mikišević Dimitrije, Petra Drapšina 55, 21000 Novi Sad, tel. 021/21-758

• Prodajemo najnovije svetske hitove za Komodor 64. Cena programa od 50 din. na više. Na komplete dobijate popust i do 50%. Sve informacije i narudžbe na tel. 041/417-819

• COMMODORE 64 — Novi (i „stariji“) hit programi za vaš Komodor. Besplatni katalog, nagradni programi, brza i kvalitetna usluga. Kmet, Flavio, G. Martinuzzi 11, 51470 Umag

• COMMODOREOVCI I u Novoj godini — najnoviji i najelitniji programi u kompletima koji sadrže do 50 igara. Komplet I: Summer Games, Pitstop 2, Hockey... Komplet J: Winter Games, Summer Games 2, Exploding Flat... Komplet K: Flight Simulation 2, Speed Kung, Activation Box... Komplet L: Rambo, Zorro... Komplet stoji 1000 do 1500 dinara plus kasete. Tražite katalog (50 d.) za naznaku „za CBM 64“ FUTURE ORION, 41000 ZAGREB, Rubčićeva 7, tel. 041/417-052

• Prodajem Commodore VIC 20. Hitno. Tel. 034/215-912. Zvati svaki danom posle podne sem ponedjeljka.

• Komodorci, prvi put na jednom mestu možete naći najbolje i najnovije disk i kasetne programe za Komodor 64 vrlo povoljno. Izbor od preko dve hiljade programa, brza i kvalitetna usluga. Tel. (posle 17h) 011/131-641. Adresa: Andričić Zdenko, il bulevar 34/52, 11070, Novi Beograd

• Novi! Po prvi put u Jugoslaviji trostruki razdnežnik, što znači da sa jednog kasetofona učitavate, a čak na dva animete. To znači da u isto vreme pravite dve kopije i samim time olakšavate rad oko preinamivanja. Cena razdnežnika sa poštarinom je 3200 din. Vujović Đurica, Ustašićka 168, Beograd, tel. 488-5242

KOMODOR

● **COMMODORE 64** — Neverovatna izbor kasnetnih programa: PIT-STOP 3, Kung Fu Master, Skool Daze, SVS 1 i 2, JSW 1 i 2, Pjyamaama 2 i 3, Elite, Tir-na-nog, Kokošnik Wilf, Stellar 7, Sorcery, Boxing, Spiffire 40, Tour the France, Rugby, Eureka, Flight sim. 2, Exploding fist i mnogi drugi. Uputstva za početnike. Besplatan katalog Žuljivic Kemal, Braće Kosorić 13, 72220 Zavidovići, tel. 072/874-441

● **COMMODORE**: Profesionalni prevodi: PRIRUČNIK (650), PROGRAMERS REFERENCE GUIDE (1300), GRAFIKA I ZVUK (900), DISK 1541 (700), KAKO DA PROGRAMIRATE C-64 (1000), 200 MAPA ZA SPRAVITELJE (850), MAŠINSKI PROGRAMIRANJE ZA POČETNIKE — prevod 1985. g. (1300). Profesionalni prevodi uputstva za uslužne programe: SIMONS BASIC (700), VIZAVRITE (600), EASY SCRIPT (400), PRAKTIKALK (750), MAE (500), HELP 64+ (500), PASCAL (400). U kompletu (3000 d.) Zbog velikog interesovanja za kompletnu literaturu, cena svih 16 knjiga (10.000 din). Rok isporuke 24 časa. KOMPJUTER BIBLIOTEKA, Filipa Filipovića 41, 32000 Čačak, tel. 032/31-20

● **COMMODORE 64** — Veliki izbor vrhunskih programa po super niskoj ceni od 30 din. po programu. Besplatan katalog, razmena, saradnja, sigurna usluga. Stanojević Nerada F, Filipa Filipovića 6, 17500 Vranje

● **Commodore 64** — Kako u ovoj gomili oglasa naći onaj pravi??? Odgovor je "kopija-software", 50000 Dubrovnik, Put bratstva i jedinstva b. b. Preko 400 najnovijih i HIT programa. Besplatan katalog.

● **Commodore** — multisoft je obezbedio najnovije hitove za vašeg ljubimca. Besplatan katalog, isporuka u roku od 24h. Dragan Jovanović, Dragoslava Jovanovića 12, 37000 Kruševac, tel. 037/30-568

● **COMMODORE 64** — U Novu godinu sa novim hitovima. Komplet programa JSV 2, CHIMERA, BOXING, LORDS OF MIDNIGHT, RED MOON, Za 100 din. Tražite katalog. Moguća razmena. Dejan Rajić, Sretna Miladenovića 7, 11165 Beograd, tel. 530-637 ili 534-839

● **Prodajem Commodore 64**, kazetofon, džojstik, 800 programa, reset, literaturu. Cena: 110 000. Komodor 64 programi. Najniže cene do 20din! Štefić Ronald, Maksima Gorkog 9, 42000 Vraždina

● **Epson RX80** printer i interfejs EC-64 za Commodore 64, sve novo povoljno prodajem. Dubravko Lisac, D. Bjelidica 6B, 47000 Karlovac, tel. 047/37-467

● **COMMODORE 20**, 16+ 4, 64 programi. Tražite besplatan katalog! Đerman Sandor, Rade Končara 23, 23000 Zrenjanin

64/mali oglasi

● Izrađujući različite hardverske dodatke za Commodore 64: CP/M kartica sa sistemskom disketom i mnogo softvera, 8 bitni A/D konverter, univerzalni I/O port sa 4 slotova za proširenje, 80 kolonska kartica itd. Tražite besplatan katalog. Dražen Šćuric, Klokočevac 105, 43211 Predavac

● **P.N.P. elektronic HARDVER ZA KOMODOR NOVO IZ P.N.P. electrica. DODACI SADA I ZA KOMODOR 64. MEĐU PRVIJM SIMONS BASIC NA KARTICI KOJA SE SPAJA NA USER PORT. NEMA VIŠE DUGOTRAJNOG ČEKANJA DA SE UPIŠE SA TRAKE. RADIMO JOŠ I LIGHT PEN, A USKORO I I/O PORT, TE CENTRONICS PRINTER INTERFEJS. JAVITE SE SA VAŠIM ŽELJAMA. BESPLATAN KATALOG NA ADRESU: P.N.P. electriconic Jeretova 12 58000 Split**

● **Commodore 64** — Izbor od preko 2000 program! Najnovije video igre, uslužni programi, novi programski jezici, baze podataka, muzika, grafika, obrada teksta i još mnogo toga a uskoro i još više. Sve je svakako i to na jednom mestu! Veliki izbor, niske cene, razmena, popusti, saveti za početnike! Tražite besplatan katalog na adresu: Željko Zagar, Vukosavićeva 82, 11090 Beograd.

● **Commodore 64** — postanite ekspert za paskal uz najnoviji interpret paskala OXFORD PASCAL. Program + kasetna + poštarina = 950 d. Capan, Zdenačić-gaj 71, 43293 Veliki Zdenci, tel. 046/87-025

● **Najtraženiji programi za C-64: Jet Set Willy II, Super Zaxxon II, Flight Simulator II, Staff of Karnath II, Boulder Dash II, Hyper Circuit, Elite, Sorcery, A View to the Kill, Sabre Wolf, Jump Jet, Grubbiys Day Out, Stop the Express, Spiffire 40 + još tri programa + poklon hardverski dodatka. Sve 10000 dinara. Sačekaj Rajko, Novi Sor 86 21466 Kucura**

● **COMMODORE 64** — NAJKVALITETNIJI, NEOPHODNI I PROFESIONALNI PREVODI: PROGRAMMER'S REFERENCE GUIDE — 1300 d., MAŠINSKI JEZIK ZA POČETNIKE — 1450 d., GRAFIKA I ZVUK NA C-64 — 780 d., UMJETNOST GRAFIKE NA C-64 — 900 d., BASIC PRIRUČNIK — 660 d., SIMON'S BASIC — 660 d., PASCAL 450 din. Izvanredna kvaliteta, niska isporuka, na višestruke narudžbe popust 10%! DUŠKO BJELOTOMIĆ, CENTAR 1, 54550 VALPOVO, tel. 054/82-665 ili 041/683-141.

● **COMMODORE 64** Najpopularnije igre januar 86: — Rambo 2 — School daze — Commando — Wizard's lair — Who dare wins 2 — Superman — Cyru Programi + kasetna = 1600 dinara Despotović Milan, Milana Zečara 6, 11120 Beograd, tel. 011/712-442

● **COMMODORE** najnovije hitove prodajem i to po vrlo niskim cenama i uz program prodajem i literaturu i reset dirke. **POPUSTI NA PAKETNI KATALOG BESPLATAN** — Roberto Kristović, Radničko šata lište 44, 58000 Split, tel. 058/46-832

NAJNOVIJI HIT PROGRAMI ZA COMMODORE 64

Ovo je prilika da nabavite samo najbolje programe po povoljnim cenama, snimljene na uvornim BASF kasetama.

KOMPLET 14. Break Fever, Minotax, Swag, Seaworld, Tir-na-nog, Stop Express, Asterix, Etker.

KOMPLET 15. Blagger goes Hollywood, Spiffire 40, Digital Dross, Quango, Bear Tread, Tiger Shark, Wher's My Bones.

KOMPLET 16. Dummi Run, Baseball II, Dark Tower, Arhipealog, Black Knight, Slash 3D, Eldon.

KOMPLET 17. Winter Games (sastoji se od 6 igara), Exploding Fist, KOMPLET 19.

Summer Games I (sastoji se od 8 igara), KOMPLET 20.

A view To a Kill (4 igre sa Džems Bondom), Jet Set Willy II, Barnu Bledita, Paradroid.

KOMPLET 21. Summer Games II (8 sportskih igara), KOMPLET 22.

For Hacker, Skyjet, Who Dares Wins, International Tennis, Cyru, Mario's Brother, Staff of Karnath I.

KOMPLET 23. Beach Buggy II, (4 ratne igre), Hyper Sports, Scrolls Abandon, Spy Hunter.

KOMPLET 24. Super Hueye (4 igre, najbolja simulacija helikoptera), Staff Of Karnath II, Chimera, Hi Bouncer.

Jedan komplet sa kasetom 1000 d. Dva kompleta 1800 d. Tri 2600 d. Četiri 3400 d. Pet 4200 d. Šest 5000 d. Svaki sledeći komplet još je za 200 dinara manje. Spisak isporučnih 13 kompleta izlazio je u oktobarskom broju Mo Mikra, PRODUKCIJA U ROKU OD 24 ČASA. Programe možete naručiti na adresu KRSTIĆ DRAGISA S, J. VUKOTIĆA 32, 11090 BEGRAD III na telefon 011/533-611 (svakom ko narudi programe šaljem besplatan katalog sa preko 1500 programa.)

● **Prodajem i menjam programe za Komodor 64.** Cena povoljna. Milan Radovanović, Bul. Lenjina 185, 11070 Novi Beograd

● **COMMODORE 64** Kompleti (sa kasetom 1500,- dinara) komplet I who dares win II staff of karnath II wizards lair cyru (knight lore) dynamic dan chimera (alien 8) komplet 2 skool daze rambo 2 superman kung fu master commando willow pattern final synth smp i još preko 2100 kasnetih i disketnih programa. Čurčić Dušan, Osmana Đikića 16a, 11000 Beograd, 011/762-022

● **Komodori!** Najnoviji programi na kaseti: SORCERY, HAKER, TOUR THE FRANCE, EXPLODING FIST, uz najniže cene. Tražite besplatan katalog. Stanić Branko, II bulevar 128, II ulaz, 11070 Novi Beograd, tel. 122-545 ili 152-083

● **Prodajem floppy disk 1541 i 1000 programa za Komodor 64.** Gitić Saša, Cerska 9, 19320 Kladovo, tel. 019/87-569 ili 88-849

● **Prodajem najnovije programe za Komodor 64.** Isporka odmah. Tel. 603-321, Maksimović Bane, Goece Delčeva 34/26, 11070 Novi Beograd

● **Prodajem moduli SIMON'S BASIC za 15.000 dinara.** Tel. 024/29-760

● **Komodori: SUMMER GAMES 2, EXPLODING FIST, PITSTOP 2, DO-MALD DUCK, SPIFFIRE 40, ELITE, THEATRE DUROCVE, STOP THE EXPRESS + 5 poklon programa za 1000 din.** Adrović Miroslav, Milutina Uskovičića 43, 24000 Subotica, tel. 024/29-760.

● **Jeftino programe za Commodore 64 prodajem za disk i kasetu: MAG-BET, SORCERY, SUMMER GAMES, WINTER GAMES.** Kresoja Dragiša, Vojvode Stepe 421B, 11000 Beograd, tel. 475-419, radnim danom od 16—20 časova.

● **C-64** — Brian Bloodaxe, Spy vs Spy II, Elite, Boulderdash II, Pink Panther, Everyone's Wally, Jump Jet, Coultron, Knocout II, sa kazetom i poštarinom 1000 dinara. Pljačanje pozuceem. Miroslav Gakić, Poljska 31, Strahinec 42300 Čakovec

● **Prodajem Commodore 64 + kazetofon + 2 džojstika + 200 fantastičnih programa (SUMMER GAMES I i II, EXPLODING FIST...).** Staro dva meseca. Posnajek Dražen, Gajeva 179, 43405 Pitomaca, tel. 046/782-019

BBC B
● **BBC/electron** — multisoft. Veliki izbor + male cene = veliki razlog da naručite besplatan katalog. Dragan Jovanović, Dragoslava Jovanovića 12, 37000 Kruševac, tel. 037/30-568

AMSTRAD
CPC 464 DO 6128
Nudimo vam veliki izbor od preko 200 najboljih fabričkih programa na kasetama i disketama po najpovoljnijim cenama. Od poslednjih programa svih vrsta i naziva izdavačima: Word Star, Pascal, Fortran, Micro Script, Master File, Mini Office. Pružamo vam izvanrednu priliku da nabavite najnovije igre za vaš računari: F. B. Boxing, Beach Head, Street Hawks, Everyone's a Wally, Exploding Fist, Chidler, Locomotion, Android One, Dark Star, te još mnogo novina za vaše kućne ljubimce. Uverite se u našu profesionalnost i odlične usluge. Za katalog i detaljne informacije obratite se na tel. 021/20-742, Veselinov Dordje, M. Gorkog 6, 21000 Novi Sad

• **AMSTRAD — multitsoft**
Najnoviji, najbolji programi po najnižim cenama: A VIEW to a Kill, Super Pipeline 2, Everyone's Wally, Trypods, Rocky Horror Show, The Way of Exploding Fist, Crystal Theft, Dragan Jovanović, Dragoslava Jovanovića 12, 37000 Kruševac, tel. 037/30-568

• **AMSTRADOVCI** Svi programi: igra, uslužni, copy... profesionalno presnimljeni direktno iz kompjutera. Samo po 50 dinara. Najjeftinije u Jugoslaviji! Tražite besplatan katalog, Trajkovski Pande, Krume Spirkoski 7, 97500 Prilep, tel. 098/26238

• **Prodajem SCHNEIDER CPC 464** sa zelenim monitorom i sa preko 1200 programa za 15 starijih miliona. Jovanović Dragan, Dragoslava Jovanovića 12, 37000 Kruševac, tel. 037/30-568

AMSTRAD CPC 464 (SCHNEIDER) — NAJKVALITETNIJI, ODABRANI PROFESSIONALNI PREVODI: UPUTSTVO ZA RAD NA AMSTRADU — 1100 d, LOCOMOTIVE BASIC — 1200 d, MAŠINSKO PROGRAMIRANJE ZA POČETNIKE — 1300 d, UPUŠTVA ZA DEVPAC I TASWORD PO 950 d, GRAFIKA I ZVUK NA CPC 464 — 850 din. Izvanredna kvaliteta, nitna isporuka, na višestruke narudbe popust 10%! **DUŠKO BJELOMČIĆ, CENTAR 1, 54550 VALPOVO, tel. 054/82-665 ili 041/683-141.**

• **Bogata ponuda programa za Amstrad CPC 464.** Jedan program 60 s. Saša Volarić, Juličić 127, 51213 Jurdani

• **AMSTRAD: Profesionalni prevodi: PRIRUČNIK CPC 464 (1100), LOCOMOTIV BASIC (1200), MAŠINSKO PROGRAMIRANJE (1300), ZAJEDNO (3400).** Prevodna uputstva za uslužne programe: DEVPAC, TASWORD, PASCAL, QUILL, MASTERFILE, HOME BUDGET. Pojedinačno (700), KOMPLET (3500). Snimljeni programi (200). **AMSTRAD FUTURE, Bate Jankovića 79, 32000 Čačak, tel. 032/30-34**

• **OMEGA SOFT** vam i dalje nudi veliki broj programa za Amstrad, pojedinačno i u kompletu. Specijalno za ovaj mesec: "Winter Sports", osam različitih programa sa skijanjem, hokejom, klizanjem, bobom i dr. Adresa: Đurić Staviša, Vladimira Gačinovića 19, 11000 Beograd

• **Prodajem ŠNAJDER CPC 6128** sa zelenim monitorom (CP/M, 2, CP/M + GSB, DR LOGO), ocarinirano. Tel. 024/30-341

• **HARRIÉR SOFT** vam nudi programe za Amstrad po najnižim cijenama a najpovoljnijim uslovima. Komplet programa za presnimavanje (TAPE TO DISK, DISK TO TAPE, DISK TO DISK I TAPE TO TAPE). Cijena kompleta + poštarina + kazeta iznosi 1500 din. Naručite besplatan katalog ostalih kompleta i programa. Adresa: Leo Lugović, Viktora Kovačića 26, 41000 Zagreb, tel. 041/679-689

• **AMSTRADOVCI, BINGSOFT** predstavlja: PLATINENKIT, izrada štampanih pločica, Voice Chess, Donald Duck, Jammin, Hockey, Nibbler, Super protoni, Millioner, Wild Bunch, Tripode, Chomper, Artist, 50 din. katalog, Trumbičeva 14, 41000 Zagreb, tel. 670-679

QL

• **QL SOFT** ima skoro sve postojeće programe za Sinclair QL. Puna garancija za programe. Trenutna isporuka. Besplatan katalog. Novitet: Super bežični kompajler, Ukoro: Fortran 77 kompajler, Petković Dejan, Dušana Dugotića 6, Beograd, tel. 404-690 ili Boško Gavović, tel. 633-501

QL

Prijelodi uputa: **QL ARCHIVE (1500 d)** — inteligentna baza podataka. **QL TOOLKIT (1200 d)** — proširenje SuperBASIC-a. Korisne programe za QL: jezike, asemblere, monitore, nove verzije programa, šah itd. i literaturu: SuperBASIC, mašinsko programiranje, MC8000, tehničke priručnice itd. razmjernoj ili prodajem (program s mikrokazetom 2400 d). Leon Kuna, Mihanovićeve 18/3, 43500 Daruvar, tel. (046) 31-893

• **QL: PROFESSIONALNI PREVODI: QL-ARCHIVE (1500), QL-TOOLKIT (1200), QL-PASCAL (1500), QL-FORTH (1500), U KOMPLETU (5000).** Zarić Slobodan, Bate Jankovića 79, 32000 Čačak, tel. 032/30-34

• **QL SOFTWARE** — razmena i prodaja najnovijih program! Najjeftinije cijene i veliki izbor. Katalog! Tel. 068/44-772, Tomaž Gorec, Sentlovcenc 21, 68212 Velika Loka

RAZNO

• **Prodajem Sensorstick** — Senzorski dožojast osam pravaca, automatska vatra. RAT-JOY — senzorski dožojast sa daljinskim upravljanjem. Nevenko Zmo, poste restanthe, 78000 Banjaluka

• „Electron news“ opet sa vam! Sheme: predajnici, light show, lečenje magnetskim poljem, elektonska akupunktura, pojačala, višestruki ispravljači alarma. Katalog 20 din. Electron news, 54000 Osijek, Kalnička 29A

• **Hitno prodajem palicu** za igru sa dtekom za Komodor, Atari ili Amstrad. Četiri pravca, pravo, četiri ukoso i pucanje sa vrha palice. Cena 3.200 din. pouzeteč. Milošević Ivan, Kovaškova 107, 37000 Kruševac, tel. 037/24-659

• **KAKTUS JOYSTICK** — Novo na našem tržištu. Originalan dizajn i prilagodljivost ruči. Tražite podrobne informacije HARD SOFT, P.P. 163, 51000 Rijeka

• **Prodajem nov. ocarinjen Atari 800 XL** sa kasetofonom, palicom Quickshot II i 30 programa. Romić Zoran, Aleksandra Tajkova 13/2 55300 Slavonska Požega, tel. 055/75-837 od 8 do 13 časova.

• **ŠPICA HARDWARE**
Spektrum: Profesionalna tastatura ŠPICA, GenSons EPROM kerizid, EPROM programer Komodor 64: CP/M modul sa uslužnim CP/M programima, RS 232. ŠKD Forum, Mikroradionica ŠPICA, Kersnikova 4, 61000 Ljubljana

Literatura



NAJVEĆI IZBOR LITERATURE KOD NAS

Preko 70 knjiga iz svih oblasti računarske tehnike
— o programskim jezicima
— programiranju
— operativnim sistemima
— mikroprocesorima
— hardvaru itd.

Velik izbor knjiga u originalu. Sve što želite da pročitate o vašim ljubimcima naćete kod nas!
— SPEKTRUM
— QL
— BBC
— COMMODORE
— APPLE II

MILOVANOVIĆ LJUBIŠA
Petra Lekovića 57, 11030 BEOGRAD
tel: 011/558-007 posle 17 časova.

RAZMENA

Ženska osoba prijatne spoljašnjosti 64K/100-320 tačaka, traži poznavstvo odgovarajućeg kompatibilnog partnera, dobro hardverski opremljenog, radi skidanja zaštite sa njenih programa i zajedničkih prijatnih softverskih trenutaka. U obzir dolaze samo ozbiljne ponude. Romantične sam prirode i volim sve što vole mladi. Javi se, čeka nas zajednička sreća.

šifra „VIRGINSOFT“

ATARI ST520

SOFTWARE Poslovni programi

— Jezet

— Ige

LITERATURA

HARDWARE Centronics paralelni interfejs za povezivanje računara sa štampačem (Epson, Star, Panasonic itd.)

— RGB28 kabl

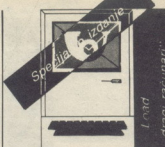
— Proširenje memorije na 1 Mbyt (š Mbyt u razvoju)

SERVIS — Servis manjih opravki

IZRADA PROGRAMERSKE OPREME sa organizacijom udruženog reda i sanjatiže

informacije **HARDWARE SERVIS** Veče 31A, 01815

Metkova, tel. (061) 618-568, u sredu i nedelju



Jedan glas za ZX81

Vašu reviju pratim od prvog broja i veoma mi se sviđala. Najviše mi se sviđa rubrika „Periferna oprema“ u kojoj opisuju razne dodatke. Čini mi se da se previše stranica posvećujete velikim kompjuterima koji su dostupni retko kojom Jugoslovenu. Volio bih kada biste objavljivali više tekstova o ZX81, koga sam nabavio nedavno, ali o njemu ima vrlo malo tekstova, kao i praktičnih uputstava za rad sa njim. Takođe bih želio da se u jednom od narednih brojeva pojavi umetak „sve rutine za ZX81“: isto tako mi se čini da bi trebalo da objavljujete veći broj malih nekloga. Ako znate adresu neke osobe ko prodaje prevod uputstva za ZX81, molim vas objavite je.

DrugarSKI vas pozdravljam

Dimitrije Šijan
Brigadara Ristića 46
23272 Novi Bečej

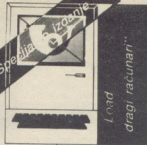
Dimitrije, sve što možemo da uradimo za stari, dobar ZX81 jeste da objavimo tvoje pismo. Nadamo se da će ti ovaj besplatni „mail oglas“ pomoći da ti se jave istomišljenici i ljudi koji se bave računarnom ZX81.

Kako s novim ROMom?

Interesuje me GENS koji se može nabaviti kod vas (u epromu) i novi ROM za „spektrum“. U vašem časopisu sam pročitao da je novi ROM kompatibilan sa svim programima koji ne koriste interfejs 1. Molim vas da mi to malo objasnite. Sam GENS upotrebljava interfejs 1 jer radi sa mikrodravom, pa je sve to malo čudno. Može li se sa novim ROM-om raditi sa mikrodravom i sa RS 232, a ako ne može, demu onda novuROM?

Urdić Ignac
Striz trg 247
66280 Slavenji Gradec
(Slovenija)

Priča je prilično dugačka. Ukratko: kompletan projekat namenjen je onim spektrumcima koje ne interesuju interfejs 1 i mikrodrav.



VRUĆE, HLADNO, VRUĆE

Poštovana redakcijo,

Povodom napisa druga Dejana Ristanovića „Dan po jutru, a jutro tmurom“ u broju 10 Računara i recenzije programa „VRUĆE LETOVANJE“, molimo Vas da u sledećem broju časopisa Računari objavite sledeće: Smatramo da je osnovni red u prikazu jednog programa navesti sve one koji su učestvovali i izradi tog programa i čija imena stoje u programu i na uputstvima uz program.

Autor grafike i dizajna programa „VRUĆE LETOVANJE“ je Igor Kordić, evropski poznat crtač stripova, što je u pomenutom prikazu ispušteno.

Zahvaljujemo na pogrešci kojom nas drug Ristanović imenuje dobričnim „prve nagrade na prvom Galakšijinom konkursu za najboljeg programera“. No, treba reći da smo na tom konkursu dobili drugu nagradu.

Pitam se da li je to bilo tako davno da su uspomene druga Ristanovića i redakcije časopisa Računari toliko izbledele da se ne sećaju redosleda koji su sami odredili.

Tvrdnje da su „igre avanture jedan od glupljih načina za utucavanje slobodnog vremena“ je odličan primer totalne odsutnosti objektivnosti. Samo nekoliko paragrafa iznad ove tvrdnje, drug Ristanović izjavljuje da ima rejting „deadly“ u igri „Elite“. Pošto je to igra avantura (mada ne tekstualna jer pripada tipu „arcade adventure“), čitaoći mogu sami izdućevati zaključak na koji način drug Ristanović „utucava“ svoje slobodno vreme.

Ili možda drug Ristanović tvrdi da su igre arkadnog tipa jedan od pametnijih načina za „utucavanje slobodnog vremena“?

„Uvek sam bio protiv ovakvih avantura jer slike nemaju nikakav funkcionalan značaj!“ iznosi ispravan stav druga Ristanovića. Slike su tu da smanje jaz između računara i korisnika-igrača i da daju nov kvalitet programu.

Ovo je jedan od prvih domaćih programa ovog tipa. Setite se koliko je avantura u Engleskoj napravljena dok na red nisu došle one kod kojih slike zavise od situacije. Da je ovo stoga domaća avantura, podržali bismo mišljenje druga Ristanovića, ovako smatramo da je deplasirano.

„Da biste Srećku, junaku VRUĆEG LETOVANJA, naložili da uzme nešto, morate biti do kraja precizni: UZMI GVOZDENI KLJUČ se ne može zameniti ni sa UZMI, ni sa UZMI KLJUČ pa čak ni sa UZMI GVOZDENI KLJUČ (slovo 'd' dobijate kao „SHIFT“ 'X' i ne možete ga, na nesreću, zameniti sa 'c')“ je još jedna neobjektivna tvrdnja.

Sve što je drug Ristanović ovom recenicom rekao je istina. Evo šta nam smeta: u srpsko-hrvatskom jeziku se reč „ključ“ piše sa 'č' pa ne vidimo razloga da neko očekuje da se piše sa 'c'. Mi implementaciju slova specifičnih našem jeziku smatramo prednošću a drug Ristanović je upravo tome našao zamerku. Bez veze!

Da bi naglasili neobjektivnost druga Ristanovića dovoljno je da kažemo da nije ni jednom rečju napomenuo da program igrača potpitanijima vodi ako unese nepotpunu recenicu, na primer:

igrač: OTVORI 'CR'
program: ŠTA?
igrač: VRATA 'CR'
program: KUJA?
igrač: PREDNJA 'CR'
program: OTVORIO SAM PREDNJA.

Ne znamo ni za jedan program koji se ovako inteligentno ponaša!

Činjenica je da „VRUĆE LETOVANJE“ nije jednostavna igra i da bi se komandama koje istovremeno utiču na više objekata igrač vrlo brzo našao u situaciji koju ne može da kontroliše, pa je stoga i obračuna pažnja da komande ne budu dvosmislene.

„Posebno je neprijatno što program ne prepoznaje naredbe koje su u kvi svih igrača avantura: N, S, E, W, U, D, INV, HELP moraju da se zamene sa IDI NAPRED, IDI NAZAD, ŠTA NOSIŠ (???) i tome sličnima“ je još jedna besmislica nepromerena drugu Ristanoviću.

Da možda drug Ristanović nije očekivao da domaći program namenjen domaćem tržištu prepoznaje engleske skraćice? Ne vidimo šta korisni-

ku može da smeta dosledna upotreba srpsko-hrvatskog jezika. Mi to samo smatramo prednošću! Nismo toliko Englezi koliko se prvi drug Ristanović.

„Naredba HELP, dobra prilika da autor programa izgrava Kirku, čak nije ni implementirana“ je već traženje dlake u jajetu. Tvrdimo da je mnogo važnije obezbediti drugu kvalitet programa nego igraču u trenutku kada mu u određenoj situaciji ništa ne pada na pamet, reći „Baš ti dobro ide.“

S druge strane, izgleda da drug Ristanović mnogo voli da ga program iritira dok igra. U tom slučaju preporučujemo mu da program IGRA.

„Spisak glagola koje računaru poznaje je umereno siromašan, a broj poruka o greškama minimalan“ ne može da prođe kao tvrdnja jer drug Ristanović izgleda nije ni pročitao da program razume sledeće glagole:

ČITAJ, ČEKAJ, DAJ, IDI, IZADI, JEDI, KUPI, OTVORI, OTKLUČAJ, OSTAVI, POPLI, POLOMI, PREGLEDAJ, POPNI SE, PUSTI, STAVI, SPAVAJ, SIDI, TRČI, UZMI, UDI, VOZI, ZATVORI, ZAKLJUČAJ, BROJ NOVAC, GLEDAJ, ŠTA NOSIŠ.

Brojem, to je 27 glagola. Ne delimo mišljenje da je to „umereno siromašan“ broj.

Posebno je iritirajuća tvrdnja da je „broj poruka o greškama minimalan“ jer program u svakoj situaciji na svaku akciju igrača daje odgovarajuću poruku o grešci, ako je akcija igrača pogrešna. Evo samo nekih od situacija do kojih može doći u igri:

igrač: IDI U SAMOPOSUGU 'CR'
program: SAMOPOSUGA NE RADI.

ili
igrač: POLOMI ORMAN 'CR'
program: NE MOGU. SUVIŠE JE TVRDO.

„Isprobali smo raznorazne psovke, i spectrum je odgovarao jedino NE RAZUMEM ŠTA ZNAČI...“

Cenimo vrlo bogat rečnik druga Ristanovića ali je podsticanje vulgarnog izražavanja zadržje što smo ovim programom želeli da izazovemo. Da je program na jednu psovku iz repertoara druga Ristanovića odgovorio na bilo koji način, to bi ga sigurno podstaklo da detaljno istraži ugrađeni rečnik psovki što bi tek bilo maltretiranje programa.

„S obzirom da nismo imali vremena da se iole ozbiljnije pozabavimo samom avanturom, teško možemo da sudimo o njenoj inventivnosti“ je izjava kojom se drug Ristanović ograđuje od recenica koje slede u daljem tekstu.

Smatramo da, budući da se nije iole ozbiljnije pozabavio samim programom, nije ni trebalo o njemu da piše. Umesto toga, on nalazi gomilu zamerk, izgleda da je drug Ristanović zaboravio da je „VRUĆE LETOVANJE“ samo igra i to, kako je dobro primetio, iz svakodnevnog, a ne realnog života.

„Ma koliko je Kontrabant siromašno urađen i neatraktivan, čini mi se da se u njemu kriju mnogo više originalnih i duhovitih ideja. Puno pravo na takvu tvrdnju ju, jasno, imati tek kada dovedem Srećka na mora, tj. nikada.“ je recenica od koje se čoveku jedino kosa može dići na glavi! Potpuno subjektivna izjava iza koje njen autor niti ne stoji što nam odmah i stavlja do znanja (???)

Ne shvatamo kako drug Ristanović može ocenjivati program „VRUĆE LETOVANJE“ kada je očigledno, prema njegovoj sopstvenoj izjavi, da nije video ni jednu dvadesetinu onoga što je u njemu sadržano?

Mi bismo prema njegovom receptu mogli napisati:

„Čini nam se da je drug Ristanović pozeleno od zavisti videvši naš program, ali pravo na takvu tvrdnju imaćemo, jasno, tek kada ga vidimo kako ga igra, tj. nikada.“

Zbog svega nabrojanog, molimo druga Ristanovića da razmisli o prikazu programa „VRUĆE LETOVANJE“ u kome bi na objektivni način citacima časopisa Računari prikazao kako loše tako i dobre strane programa. Molimo ga, ipak, da pre toga igru IGRA!

Autori programa „VRUĆE LETOVANJE“:

Ivan Gerendić
Milan Pavićević

Me šta neki ljudi rekli, uveren sam da rubrika „Load „Dragi računari“ nije uluko bačen prostor i da se iz nje može mnogo naučiti. Ja, na primer, već godinama štim u uverenju da je program bolji ako ima nekog opciju; ako se ta opcija nekome ne sviđa, neka je ne koristi. Iz ovoga sam pisma, međutim, naučio da su dobre strane programa „Vruće letovanje“ što:

* Ne omogućava zmenu slova č, ć, ž i š slovima c, c, ž i s koje se lakše kucaju;

* Ne omogućava skraćivanje naredbi i izostavljanje njihovih delova;

* Nema default vrednosti skoro ni za šta;

* Ne prepoznaje skraćice N, S, E, W, U, D;

* Nema naredbu HELP;

* Ne razume psovke;

* Saznaje da je program koji nema razne stvari veoma vredan je poboljšalo i moje mišljenje o njegovim autorima. Jer, ni oni nemaju prvu nagradu na „Galakšijinom“ Konkursu.

Dejan Ristanović

Računar se 1946. godine koristilo jedino za balističke proračune, prognozu vremena i neka naučno-tehnička izračunavanja, a dvadeset godina kasnije primenjivao se u više od 600 oblasti ljudske delatnosti. Pre desetak godina govorilo se o 2500 računarskih profesija, a danas gotovo da nema posla čiji bi bar deo preuzeo računar. Da bismo mogli da govorimo šta nam valja sutra činiti, osvrnimo se na razvoj računara kroz priču o ličnostima koje su najviše doprinele njihovom razvoju i utrljali put informatičkoj revoluciji.

Engleski inženjeri ne samo da su išli u korak sa svojim američkim kolegama, već su ih, u mnogo čemu, i preticali. Tako je CRT (cathode ray tube) — memorijski uređaj na bazi katodne cevi koji omogućava da se informacija pamti u obliku elektrostatičkog naboja na površini ekrana — izum naučnika sa mančesterskog univerziteta Vilijamsa (F.C. Williams). U jesen 1947. Vilijams je uspeo da obezbedi pamćenje 2048 (2K) bitova za nekoliko časova. Korišćenjem CRT mančesterski kompjuter „MARK I“ uspešno je počeo sa radom 21. juna 1948. Osim toga, u mančesterskom računaru MADAM“ prvi put je bio primenjen INDEKS REGISTAR koji olakšava transformaciju naredbi u procesu rešavanja zadataka i koji je postao obavezan deo svih narednih računara.

Jedan od rukovodilaca projekta „MADAM“ bio je poznati engleski matematičar Alan Tjuring koji je ranije aktivno učestvovao u izgradnji mašine ACE (Automatic Computing Engine — „engine“ je ime dato u čast Bebidžu).

Tjuringovski metod

Alan Tjuring (Alan Turing) rođen je 1912. godine u Londonu. Sa šest godina samostalno je naučio da čita, piše i računa, sa jedanaest je vršio složene hemijske opite, a u petnaestoj godini sa olovkom u ruci proučavao je Ajnštajnovu teoriju relativiteta. „Ja mislim da će on biti matematičar“ — pisao je njegov učitelj. „Takvi učenici kao što je on sreću se jednom u dvesta godina“. Alan je zaista upisao matematiku na jednom od koleđa Univerziteta u Kembridžu i ubrzo po diplomiranju, 1937. godine, publikovao je delo „O izračunljivim brojevima“ („On computable Numbers“) koje mu je donelo svetsku slavu.

Dr Robin Gandi se sećao: „U vreme prvih godina svog naučnog rada Tjuring se interesovao za niz pitanja, uključujući teoriju brojeva i kvantnu mehaniku, i počeo je da pravi mašine za izračunavanje Rimanove zeta-funkcije. ... Interes za računanje doveo ga je na misao o svršishodnosti razmatranja pitanja — kakve vrste procesa može da izvrši mašina i opisao je „univerzalnu“ mašinu koja je, zahvaljujući odgovarajućim pravilima ponašanja, mogla da imitira ponašanje bilo koje druge mašine. Na taj način Tjuring je dao preciznu definiciju IZRACUNLJIVOSTI i pokazao da postoje matematički zadaci čije rešenje NIJE IZRACUNLJIVO u ovom smislu. Rad koji sadrži ove rezultate predstavlja tipičan tjuringovski metod: polazeći od osnovnih principa i

koristeći konkretne primere, on razvija teoriju koja ima karakter opštosti“.

U toku poslediplomskih studija na Prinstonskom univerzitetu, Tjuring se bavio matematičkom logikom. Po odbrani doktorske disertacije 1938. godine odbija da ostane u Americi kao asistent Džona von Nojmana i vraća se u Englesku, gde postaje predavač na kraljevskom koledžu u Kembridžu.



HRONOLOGIJA

- 1912. Roden u Londonu 23. juna.
- 1937. Publikuje delo „O izračunljivim brojevima“.
- 1938. Doktorira na Prinston univerzitetu i vraća se u Englesku. Postaje profesor Kembridžskog King koledža.
- 1946. Počinje da se bavi pitanjima izgradnje praktičnih računskih mašina.
- 1948. Dolazi na Mančesterski univerzitet i uključuje se u razradu prototipa „Mark I“.
- 1950. U maju počinje da radi računara ACE koji je osmislio. Objavljuje Prvi Mančesterski programerski priručnik i znamenitu studiju koja izaziva burne diskusije — „Može li mašina da misli“.
- 1951. Počinje primena prvog komercijalnog računara na svetu — Ferranti Mark I čiji je programski sistem projektovao Tjuring.
- 1952. Postaje jedan od rukovodilaca projekta računara „MADAM“ u kojem je prvi put korišćen INDEKS REGISTAR.
- 1954. 18. juna pronađen je mrtav u svojoj postelji.

Prvi programski sistem

Za vreme rata radio je u Britanskom departmanu veza vršeći niz računskih i teoretskih poslova za vojsku. Između ostalog, bio je u ekipi koja je uz pomoć elektromehaničkog računara Colossus uspeła da „provali“ šifrovane poruke koje su Nemci slali pomoću svog remera-dela — Enigme. Počevši od 1946. godine, Tjuring se bavi pitanjima izgradnje konkretnih ra-

čunara, ali uporedo radi i na teorijskim naučnim istraživanjima. Njegov dolazak u Mančester stimulatino je delovao na programsku aktivnost. Tjuring je napisao prvi „Mančesterski programerski priručnik“ (1951. godina), kada je objavio i knjigu „Računarske mašine i razum“ („Computing Machinery and Intelligence“) preštampanu pod nazivom „Može li mašina da misli“ („Can a machine think“), u kojoj izlodi paralelu između rada mozga i dejstva računara. Ova studija, koja je izazvala burne diskusije među pionirima računarstva, ni danas nije izgubila na aktualnosti.

Tjuring je, takode, tvorac prvog programskog sistema koji je implementiran u prvi komercijalni računar na svetu — Ferranti Mark I. Zahvaljujući programskom sistemu, ovaj računar se mogao programirati na simboličkom (asemblerskom) jeziku umesto na mašinskom, kako su se jedino mogli programirati prethodni računari sa unutrašnjim programom. Sem toga, u Tjuringovu sugestiju, u ovaj računar ugrađen je i generator slučajnih brojeva. (Ferranti Mark I je prvi komercijalni računar na svetu, jer je bio na tržištu u februaru 1951. godine, naše, pet meseci pre UNIVAC-a kome se nezaluzno pripisuje prvenstvo.) Ko zna za koliko bi još izuma čovečanstvo danas bilo bogatije da Tjuring nije tragično završio život 1954. godine.

Čudak iz Londona

Alan Tjuring privatno ni izbliza nije tako poštovan kao Alan Turing — naučnik. Po rečima njegovih savremenika, bio je čudak koji je predstavljao neskladnu kombinaciju genijalnog dečaka i senilnog profesora. Bio je odbojan i zatvoren individualac koji ni najmanje pažnje nije posvećivao svom izgledu. Nemarno odeven, neočešljan i zapušten, stalno je izgledao kao da je tek ustao iz kreveta. U njegovom životu majka je bila jednina žena, a kako je ignorisao sve ljude koji mu nisu bili intelektualno ravnj, imao je vrlo malo prijatelja. Kada je 8. juna 1954. godine naden mrtav u svojoj postelji, njegova majka i malobrojni prijatelji nisu mogli da poveruju nalazima lekara koji su ustanovili samoubistvo trovanjem kalcijum-cijanamidom, smatrajući da je po sredi nesretn slučaj. Međutim, činjenice o njegovom privatnom životu i neprikladnosti opetpretrpčavanim normama ponašanja ukazuju da je to ipak bilo samoubistvo. Smrt je uspeła da zaustavi i takav energičan i kreativan um kakav je imao Alan Tjuring, ali i nije izbrisala njegove ideje i dela koja i danas osvetljavaju put svima koji kreću izazovnim stazama računarstva.

Nevenka Spalević

NOLIT T. F. Fraj RAČUNARI ZA POČETNIKE

T. F. Fraj

RAČUNARI ZA POČETNIKE



Cena 1.200 dinara

NOLIT

IRO „NOLIT“ OUR I

Beograd, Terazije 13/IV

NARUDŽBENICA Računari br. 11

Ovim neopozivo naručujem knjigu RAČUNARI ZA POČETNIKE po ceni od 1.080 dinara (sa popustom 10%)

Porudzbine telefonom: 011/338-150