

**CALCULATOARE
PERSONALE**

8 / 91

(11)

125 Lei

ISSN 1220-1529

Revistă lunară editată de Micro ATCI S.R.L. Tîrgu Mureş

Cartele fax

Fonturi

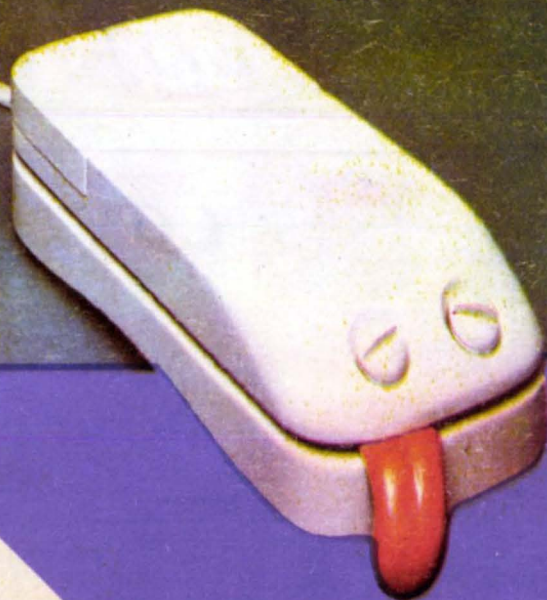
Turbo Vision

Cursuri

- Turbo Prolog
- OOP - C++

Memento:

Comenzile Paradox 3.5



- **Doriți** mijloace standard de conectare la baze de date din Europa?
- **Doriți** interconectarea calculatoarelor dumneavoastră?
- **Vreți** transferuri de date între diverse tipuri de calculatoare?
- **Aveți** echipamente cărora doriți să le atribuiți valențe noi?
- **Doriți** o linie tehnologică completă pentru editare cu caractere în alfabet românesc?
- **Vă este necesar** un sistem de introducere și verificare date multiacces pe PC, adaptabil la aplicațiile dumneavoastră?

Experiența și realizările laboratorului reprezintă cartea noastră de vizită:

- emulatoare transfer fișiere în comunicație sincronă compatibile IBM 2780/3780 pe Felix, mini, micro 8 biți și Robotron 1834 (sau orice PC care are comunicație sincronă);
- concentrator de terminale IBM 3271(mini,micro);
- linia ARIEL (ARIEL, EMMA, emma);
- acces și transfer fișiere multiacces mini-micro în comunicație sincronă (EMMI, emmi);
- editor de fonte VENTURA, fonte românești pentru VENTURA și WORDSTAR, procesoare de texte;
- software pentru conectarea calculatoarelor la o rețea X25;
- PAD-X25 - asamblor/dezasamblor de pachete într-o rețea de calculatoare X25;
- IDA - sistem data-entry multiuser pentru PC cu instrumente de dezvoltare aplicații specifice Data-entry;
- instalări de rețele locale de tip Novell.

NU EZITAȚI!

CONTACTAȚI:

ICI - Laborator **SYSNET**
 B-dul G-ral Averescu, Nr.8-10
 tel. 65.60.60/123, 118, cam. 105,
 112

if

revistă de Informatică
editată de firma Micro ATCI

Director: ing. Dumitru Dunca

Redacția:

ing. Iosif Fettich,
ing. Ingrid Maier,
ing. Romulus Maier,

Colaboratori externi:

ing. fiz. George Anescu,
stud. Daniel Buleu,
ing. Alin Flaidăr,
ing. Stelian Florea,
ing. Răzvan Grigorescu,
ing. Cristian Malide,
ing. Voicu Manole,
ing. Francisc Publik,
ing. Stan Sava
stud. Tiberiu Socaciu,
cercetător Pandele Stănescu

Tiparul: tipografia Tîrgu Mureș

Revista apare lunar.

Preț: 69 lei

Manuscrise originale sau listin-guri de programe sînt primite cu plăcere de redacție, cu condiția să nu fi fost publicate și în altă parte. Prin expedierea unui manuscris pe adresa redacției, autorul consimte implicit la publicarea materialului său în cadrul revistei. Onorariul se negociază cu redacția. Materialele nepublicate nu se înapoiază și nu se rețin.

Revista noastră vă oferă spațiu pentru reclamă și publicitate. Pentru amănunte vă rugăm să luați legătura cu redacția.

Cei care doresc să anexeze revistei pliante publicitare tipărite în regie proprie, sînt rugați și ei să se adreseze redacției.

Adresa și telefonul redacției:

Micro ATCI, redacția "if",
RO-4300 Tîrgu Mureș,
C.P. 172, O.P. 1,
tel./fax 954/31660 (direct),
33612, 24158, 20057, 33511,
int. 134 sau 189
41417 sau 46885 (după ora 19)
telex 65354 (Intertour)

Happy Anniversary !

Firile sentimentale au sărbătorit în august împlinirea a 10 ani de la nașterea calculatoarelor personale. În august 1981 IBM prezenta primul calculator personal. Acești 10 ani au fost caracterizați de o dezvoltare în ritm susținut a hardware-ului, ritm care cel mai adesea nu a putut fi urmărit și de dezvoltarea software-ului.

În următorii 10 ani situația se va schimba. În centrul atenției se află trei domenii: comunicațiile, grafica și software-ul orientat obiect.

Comunicații: Se estimează că în următorii 3(!) ani fiecare PC va fi cuplat într-o rețea. Problema legăturilor fizice este rezolvată (în alte țări de soare pline), mai rămîne de rezolvat problema soft-ului. Se mai estimează că în curînd poșta electronică (E-Mail) va juca un rol mai important decît transmisia prin fax. Pînă atunci puteți folosi însă calculatorul Dvs. și ca aparat fax. Cum? Veți afla din articolul prezentat în acest număr.

Grafică: Și în acest domeniu se vor realiza progrese semnificative. Cel mai important domeniu de utilizare a graficii pe calculator va fi în cadrul aplicațiilor multimedia. Încă la sfîrșitul acestui an va fi prezentat Windows 3.1 care va avea o extensie multimedia. Un "Multimedia-Postscript" va rezolva problema diferitelor formate ale datelor. Despre multimedia am mai vorbit și vom mai reveni. În acest număr vă prezentăm, în articolul despre fonturi, problemele tehnice care pot să apară la tehnoredactarea pe calculator. Facilitățile grafice sînt esențiale și în acest caz.

Software orientat obiect: Aceasta nu înseamnă altceva decît că utilizatorii vor fi în măsură să-și ajusteze singuri aplicațiile. Aplicațiile vor deveni scalabile și vor fi livrate sub forma unor biblioteci de obiecte urmînd ca utilizatorii să-și definească un mediu de lucru "personal". Sperăm să deschidem o fereastră spre astfel de aplicații prezentîndu-vă produsul Turbo Vision al firmei Borland și antamînd un curs de C++.

În acest număr facem și cîteva dezvoltări senzaționale - citiți articolul "Cine ești dumneata Bill Gates, și ce-ai făcut în ultimii 5 ani?", intitulat de fapt: "Factorul 20%".

Lectură plăcută!

Romulus Maier



Cuprins

Știri

- Noutăți pe piața produselor software pag.4
- Instant Logic pag.4
- NTT cu display 3D! pag.4
- Un nou harddisk Seagate pag.4
- Powersave 500 pag.5
- A doua generație
de programe antivirus pag.5
- Dublați capacitatea
discului Winchester ! pag.6
- Expoziție pag.6

Profil

- Factorul 20% pag.7

O legendă vie: Bill Gates

- Strategia Microsoft pag.8

Windows here, Windows there,
Windows everywhere

Puncte de vedere

- Protecția programelor pag.10

Comunicații

- PC sau telefax? - Cartele fax pag. 13

Ca alternativă la telecopiator ca echipament separat, PC-ul - dotat cu hard și soft corespunzător - câștigă din ce în ce mai mult teren ca emițător și receptor de telecopii. "if" vă prezintă avantajele și dezavantajele cartelelor fax.

Fonturi

- Scrierile și calculatorul pag.20
- Generatoarele de fonturi pag.23
- Emulatoarele Postscript pag.25
- 6 din cele mai importante soluții hard pag.28

Software

- Turbo Vision pag.30

În primăvara acestui an profetul limbajelor de programare orientate obiect, Borland, a împărtășit credincioșilor săi ultima sa viziune. Turbo Vision nu este doar o colecție de obiecte ce trasează cadrul general al unei aplicații, ci este și suportul unui concept nou în programarea PC-urilor. Deși timpul de inițiere a crescut, a scăzut dramatic durata finisării unei aplicații iar cizelarea ei poate fi realizată impecabil.

Cursuri

- Turbo Prolog (partea a 2- a) pag.36

După introducerea în principiile de bază ale programării logice din episodul anterior, de data aceasta vom examina tehnica de programare în Turbo-Prolog, prin intermediul unui exemplu.

- OOP: C++ - (partea întâi) pag.40

Nici o filozofie de programare nu a stîrnit atîta interes în ultimul timp ca programarea orientată obiect. După cursul introductiv pentru limbajul Pascal, vă prezentăm acum continuarea pentru cei interesați de C++.

Rețele Novell

- Novell Lite v1.0 pag.46

Practică

- Parolă variabilă pag.50
- 1024 pag.51
- Cîte ceva despre tastatura AT pag.51
- O glumă pag.52
- Recondiționarea riboanelor pentru imprimantă pag.53

Rubrici

- Caseta redacției pag. 1
- Editorial pag. 1
- Poșta redacției pag.54
- Mica publicitate pag.54

Noutăți pe piața produselor software

LOGIC, în calitate de distribuitor în România, a fost anunțat că la începutul lunii iulie 1991, **Borland** a încheiat un acord pentru achiziționarea **Corporației Ashton Tate**.

Borland va continua să actualizeze și să dezvolte produsele **Paradox Borland** și **dBase Ashton Tate** pentru sistemul de operare DOS. Mai mult, Borland va continua să ofere **dBase IV** pentru mai multe sisteme de operare, incluzând **UNIX 386**, **SUN OS**, **DEC VAX VMS** și **Macintosh**.

Borland este în curs de elaborare a unei tehnologii pentru interpretor și compilator **dBase** pentru utilizatori **DOS**, **Windows** și pentru alte sisteme de operare.

Noile produse **Windows**, numite **"Paradox for Windows"** și **"Object dBASE"** vor fi compatibile cu omo-

loagele lor **DOS**. Produsele vor fi realizate pe baza noii arhitecturi **Borland engine**, permițând interschimbarea transparentă a datelor între **Paradox**, produsele **dBASE**, **Btrieve**, **spreadsheet** și **SQL**. Data de punere în vânzare nu a fost anunțată.

Produsele **Windows** vor fi bazate pe tehnologia **Borland** evoluată, orientată pe obiecte. Acestea vor fi capabile să gestioneze obiecte cum ar fi forme, rapoarte sau cereri. Aceste noi versiuni ale produselor **dBASE** și **Paradox** vor fi complet interoperabile cu produsele **Borland** de vîrf de pe piață, incluzînd **Object Vision** și **Borland C++**.

Borland va continua să susțină și să dezvolte produsele **Ashton Tate Interbase** pentru sistemele existente, incluzînd **UNIX** și **VMS**.

De asemenea **Borland** plănuiește să asigure susținerea în continuare și a altor produse **Ashton Tate** și va dezvălui detalii despre acele produse după terminarea integrării.

Definitivarea unirii complete între **Borland** și **Ashton Tate** este dependentă, pe lângă alte lucruri, de acordul acționarilor fiecărei companii și de obținerea tuturor aprobărilor guvernamentale necesare. Tranzacția se așteaptă să fie încheiată cel mai târziu la sfîrșitul acestui an.

O altă noutate, oferită de firma **LOGIC**, este oferta de a achiziționa produsul **Borland Turbo Pascal Professional 5.5** la un preț de numai **6495 lei**.

*Stelian Florea
director - LOGIC Sibiu*

Instant Logic

Inginerii și designerii circuitelor integrate își pot face acum sintezele logice și pe un **PC**, (în curînd pe orice stație de lucru), cu ajutorul noului pachet intitulat **"Instant Logic"** al firmei **Autodesign Inc.** Pot fi generate logici pentru execuții logice, tabele de adevăr și pot fi definite automate finite.

Versiunea 2.0 a produsului **"Instant Logic"** poate fi rulată pe orice calculator compatibil **IBM**, de tip **PC AT/386**, cu cel puțin **640 KB** memorie. Un circuit poate să aibă pînă la **300 - 400** de porți. Versiunile viitoare vor avea incorporate extensii **DOS**, ușurînd munca de design considerabil.

"Instant Logic" este o colecție de unelte (tools); incluzînd un sintetizator, cu ajutorul căruia se pot utiliza listele **Orcad**, execuții logice, descrieri de automate finite sau tabele de adevăr și toate acestea pot fi sintetizate pentru implementarea porților; un utilizator - analizor al costului și timpului necesar pentru design; un simulator funcțional; un viewer logic; un program generator de biblioteci pentru construcția simbolurilor grafice necesare schemelor lui **"Instant Logic"**.

Elementele primitive (porțile și bistabilii) pot fi descrise cu ajutorul descriptorilor **ASCII** care specifică ridicările și coborîrile de întîrziere, perimetrul porților și pinii exteriori ai circuitului anticipat. Oricum sistemul are o bibliotecă cu seturi generice.

Utilizatorii pot rula **"Instant Logic"** pe orice **PC AT** standard cu **286** sau **386**. Cu un **386** la **20 MHz** se compilează cu o rată de **100** porți pe minut. Soft-ul suportă monitoare color atît **EGA**, cît și **VGA**. **"Instant Logic"** este disponibil de un an deja la un preț de **495\$**.

(S.T.)

NTT cu display 3D!

Nippon Telegraph & Telephon Corp. (NTT) a construit un nou display color pentru imagini tridimensionale fără a utiliza nici o sticlă specială. Ecranul color are o dimensiune de **15 inch** și pe el se proiectează două imagini - una pentru fiecare ochi -, care ajung separat și direct la fiecare ochi în parte. Rezultatul (din punct de vedere optic) este o singură imagine la care nu mai trebuie să te uiți cu familiarii ochelari de corecție. Se pare că singura dificultate este producerea lentilelor (după spusele unui cercetător de la **NTT Human Interface Laboratories**). **NTT** combină cele două lentile plate cu un adeziv ultraviolet. Noile display-uri sînt echipate cu un "infra red sensor" care este capabil să "ducă" capul utilizatorului în timp real.

(S.T.)

Un nou harddisk Seagate

În cursul trimestrului trei/91 firma **Seagate** va lansa pe piață **STY2400N**, un harddisk de **1 1/4"** pentru interfața **SCSI-2**.

Capacitatea sa de **2,4 Go** va constitui cel mai înalt nivel de stocare în acest format. Prețul de vânzare va fi de aproximativ **3500\$**.

Trebuie menționat că **Seagate** dispune de un harddisk de **426 Mo SCSI-2** în format **3 1/2"**.

(V.M.)

Powersave 500

Această realizare a firmei Dakota Microsystems împiedică pierderea de date în cazul întreruperii curentului refăcând starea memoriei la revenirea tensiunii.

Esențial pentru o sursă neîntreruptibilă este să furnizeze calculatorului curent continuu un anumit timp. Soluția externă nu este cea mai judicioasă deoarece este necesară trecerea prin două conversii: CA/CC și CC/CA (curent alternativ/curent continuu și invers) cu un randament foarte bun, ceea ce se realizează greu din punct de vedere tehnic. Firma Dakota și-a imaginat plasarea sursei direct în calculator și a creat cartela Powersave 500 ce poate fi considerată ca o baterie de 12V/1,2A cu reincărcător încorporat. Un regulator transformă această tensiune în ieșiri +5V/20A, +12V/8A, -5V/5A și -12V/5A. Sistemul este comutat în cazul în care detectorul de tensiune constată o scădere de la +5V la 4,75V.

Volumul redus ocupat de cartelă nu permite utilizarea unei baterii voluminoase astfel că Powersave 500 poate furniza 200W numai 90 de secunde, timp suficient însă. Pentru întreruperile: pînă la 2 secunde ea înlocuiește doar sursa de alimentare; între 2 și 20 de

secunde salvează memoria calculatorului. Dacă curentul revine în mai puțin de 30 de secunde atunci va restaura contextul existent anterior, utilizatorul puțin să-și continue munca. În caz contrar va opri calculatorul rămînînd în așteptare și va putea reface vechea stare la revenirea tensiunii.

Punerea în funcțiune a cartelei Powersave 500 este ceva mai complicată decît în cazul unei surse neîntreruptibile externe. În primul rînd este necesar un conector liber situat dacă este posibil cît mai departe de interfața controllerului de harddisk din cauza radiațiilor parazite. Grosimea cartelei face imposibilă utilizarea conectorului alăturat. Montarea cablurilor nu este prea elegantă deoarece fișele P8 și P9 ale alimentării trebuie mutate la un capăt al cartelei, aceasta aflîndu-se în cele mai neașteptate locuri în calculator. (Este vorba despre compatibile). Cel mai supărător lucru este însă instalarea soft-ului. Aceasta înseamnă octeți în minus în memoria de lucru, o linie de transmisie ocupată permanent (COM1, COM2 sau LPT2) și 32 Ko ocupați în high-memory pentru BIOS-ul cartelei Powersave 500.

Refacerea contextului la sfîrșitul unei întreruperi de curent este din fericire parametrizabilă (manual sau automat). Salvarea memoriilor EMS și video este, de asemenea, prevăzută. Restaurarea corectă în cazul unor programe care lucrează cu interfața video într-un mod de emulare face necesară recunoașterea acestora, complicîndu-se apoi cu apelul unei proceduri DOS specifice acestui mod.

Un lucru pe care trebuie să încercăm să nu-l uităm este prezența cartelei în calculator. Înainte de oprirea lui este necesară apăsarea tastelor ALT-D. Cei prea grăbiți sau cei obișnuiți cu mijloace de deconectare mai puțin ortodoxe vor avea surpriza, la viitoarea punere în tensiune, de a-și regăsi ultima aplicație la care au lucrat.

Powersave 500 nu este o alimentare suplimentară, de rezervă, ci un dispozitiv care permite utilizatorului recuperarea datelor în cazul întreruperii curentului. De fapt interfața face același lucru ca oricare altă sursă neîntreruptibilă externă oferind în plus și modul automat ceea ce îi oferă un raport bun calitate/preț (1050 DM).

(V.M.)

A doua generație de programe antivirus

Firma californiană "Xtree" a conceput o a doua generație de programe de protecție împotriva programelor de sabotaj. Programul de protecție "Virusafe" nu recunoaște doar virușii cunoscuți ai calculatoarelor, ci este capabil să controleze și potențialul distructiv al unor programe suspecte (posibili viruși).

Experții au din nou păreri diferite. În SUA au fost găsite, pînă în prezent, 500 de variante de programe de sabotare, în schimb profesorul Klaus Brunnstein a numărat de curînd chiar 1000. Zăpăceala intervenită în timpul numărării virușilor nu trebuie să ne mire. În funcție de faptul că variantele unor programe de sabotaj cunoscute se încadrează ca viruși "independenți" sau se consideră ca "descendenți" ai virușilor deja apăruți, numărul acestora variază între limite foarte largi.

Programul de devirusare al firmei californiene "Xtree" promite o curățire meticuloasă. Noua versiune a programului lor de devirusare "Virusafe" ar fi în stare să recunoască și să elimine și viruși încă necunoscuți.

Cea mai mare problemă a luptei contra virușilor, pînă în prezent, a fost cursa contra timpului. Puhoiul de programe antivirus a șchiopătat mereu în urma inventivității criminale a programatorilor de viruși și în urma "softurilor degenerate" apărute întîmplător.

Noua versiune a programului "Virusafe" nu recunoaște doar toți virușii cunoscuți pînă în prezent ci și noi "variante" ale lor. El nu elimină doar virușii existenți, ci deține și un sistem de verificare pentru software suspect, care funcționează în felul următor: atunci cînd "Virusafe" categorisește un program, sau părți ale lui, ca fiind "suspecte de viruși", el generează un director temporar cu "material de joacă". Ulterior software-ul suspect va fi implementat în acest director și "Virusafe" urmărește ceea ce se întîmplă.

Dacă datele memorate în directorul de test vor fi alterate sau modificate, programul va șterge directorul de test împreună cu fișierul original din care provine programul suspect. Apoi virusul recunoscut va fi introdus automat în lista "programele de sabotaj cunoscute". La următoarea încercare de infiltrare virusul va fi șters imediat.

Programul antivirus al celei de-a doua generații este imun la infectarea proprie cu vre-un virus. El constă din 6 segmente, fiecare segment conținînd secțiuni de program reacoperibile. Dacă la autotestare unul dintre cele 6 segmente este depistat ca fiind "virusat", atunci această secțiune va fi ștearsă. Programul este în stare să se regenereze din elementele de program redundante existente în celelalte secțiuni.

(I.M.)

Dublați capacitatea discului Winchester !

În luna februarie 1991 firma Stac Electronics lansează pe piață un produs care se va bucura de un deosebit succes în lunile următoare: **STACKER 1.10**.

Ce este de fapt **STACKER** ? Nimic altceva decât un produs software de compactare a datelor în timp real.

În esență, **STACKER** este capabil să dubleze capacitatea discului Winchester fără probleme deosebite. Este remarcabil faptul că poate fi instalat sub diferite variante DOS ca: MS-DOS și PC-DOS versiuni mai mari decât 3.0, DR-DOS 5.0, Compaq și Tandon DOS.

Recunoaște partițiile mai mari de 32 MB, se instalează ușor și nu este necesară reformatarea discului Winchester la instalare.

Există două modalități distincte de instalare și anume:

- instalare incrementală, în acest caz toate datele de pe discul Winchester sînt compactate, dar înainte de aceasta se elimină toate "gap"-urile de pe disc, de aceea durata instalării este mai lungă.

- instalare numai pe zona liberă a discului, durata acestui tip de instalare fiind mai scurtă.

Viteza de prelucrare a datelor nu este diminuată de operațiile de compactare/restaurare din RAM, dimpotrivă ea poate fi chiar accele-

rată luînd în considerare faptul că acum capetele de citire/scriere pentru aceeași cantitate de informație vor efectua numai jumătate din deplasarea anterior necesară.

La instalare **STACKER** creează o nouă unitate logică, aceasta se comportă ca și cînd utilizatorul ar avea la dispoziție un disc Winchester de capacitate dublă.

Odată instalat **STACKER** oferă patru noi comenzi specifice și anume:

- - **SDIR**, furnizează informații asupra raportului de compactare pentru fiecare fișier dintr-un subdirector.
- - **SCHECK**, este similar cu comanda CHKDSK din DOS.
- - **SCREATE**, permite crearea de noi unități logice de capacitate dublă.
- - **SREMOVE**, anulează o unitate logică creată anterior.

Lansarea în execuție este asigurată de un "device driver", înscris automat la instalare în fișierul CONFIG.SYS, care ocupă între 30 KB și 50 KB din memoria operativă. Cu ajutorul unor programe de gestiune a memoriei precum QRAM, ABOVE DISK, QEMM 386 etc., se poate încărca acest "device driver" în memoria extinsă, eliminînd astfel și acest mic neajuns referitor la memoria operativă.

Trebuie avut în vedere și faptul că datele compactate cu **Stacker** nu mai pot fi citite decât cu ajutorul driver-ului specific. Dacă dintr-un motiv oarecare vă decideți să nu mai utilizați produsul - pentru că, de exemplu, nu aveți memorie extinsă și doriți să utilizați un program care necesită întreaga memorie, nepermițîndu-vă alocarea a nici măcar 30 KB în alte scopuri - atunci fișierele compactate nu mai sînt vizibile decât după o decompactare cu driver-ul specific **Stacker** instalat.

În afara variantei software, firma Stac Electronics oferă și o variantă hardware. Aceasta constă dintr-o plachetă care conține un coprocesor specializat și 32 kB memorie RAM, ceea ce mărește și mai mult viteza de lucru, cît și raportul de compactare a datelor.

Făcînd un mic calcul, vom constata că achiziționarea acestui produs este o investiție rentabilă, chiar dacă vom converti numai un disc Winchester de 40 MB la 80 MB. Prețul unui nou disc Winchester împreună cu controller-ul aferent depășește prețul **STACKER**-ului. Produsul poate fi achiziționat de la firma AbMod, tel. 95-156350

(Publik Francisc,
AbMod, Cluj-Napoca)

Expoziție

Expoziția organizată de firma OCTOGON COMPUTERS SRL din Tîrgu Mureș, în perioada 10-12 septembrie, în sala de expoziții a casei de cultură a tineretului Mihai Eminescu, a stîrnit interesul general al celor interesați de tehnica de calcul. Vizitatorii au putut aprecia calitatea echipamentelor prezentate și a aplicațiilor realizate cu aceste echipamente. Cei interesați au putut vedea o stație grafică de lucru pentru proiectare asistată de calculator compusă dintr-un sistem 386 cu coprocesor matematic Cyrix 83D87 de 33 MHz și echipată cu un plotter Hewlett-Packard 7575A. Un alt punct de atracție l-a constituit demonstrarea posibilităților de realizare de sisteme de achiziții de date cu calculatoare DTK. Un stand a fost destinat prezentării de aplicații economice și financiar-contabile realizate de colegi de la alte firme. Posibilitățile de lucru într-o rețea Novell ale unui program de evidență a furnizorilor și comenzilor dintr-o întreprindere au fost demonstrate pe un stand constituit din opt calculatoare, legate într-o rețea de tip Arcnet. Prezentările au fost condimentate cu un set de programe de demonstrații grafice, menite să pună în evidență calitatea excepțională a calculatoarelor DTK, calculatoare foarte apreciate și în SUA, așa cum a rezultat și dintr-o

anchetă realizată de reputata revistă PC Magazine, în luna iunie '90, în rîndul a 18.000 de utilizatori americani. Calculatoarele DTK au obținut punctaje ridicate la toate întrebările referitoare la fiabilitate și performanțe. Acesta este unul din motivele pentru care firma OCTOGON COMPUTERS SRL a hotărît importul acestor echipamente. Reputatele imprimante EPSON, dintre care au fost prezentate cele de uz general FX-1050, una color LQ-2550 și una rapidă DFX-8000, s-au integrat perfect în configurațiile prezentate și au ilustrat în mod convingător dorința firmei OCTOGON COMPUTERS SRL de a oferi partenerilor echipamente de calitate, în configurații specifice aplicațiilor beneficiarului. Pentru echipamentele livrate firma asigură garanție și service postgaranție. OCTOGON COMPUTERS SRL asigură partenerilor și software autorizat, fiind capabilă să satisfacă necesitățile beneficiarilor, conform dorințelor acestora, începînd cu determinarea necesităților partenerilor, procurarea de echipamente, instalare, service, proiectare de aplicații și școlarizare.

Telefonul firmei este 954/38095.

Factorul 20%

Microsoft. Making it all make sense. Deviza unuia dintre cei mai mari producători de software nu este o vorbă goală. Aproape că nu este domeniu al soft-ului destinat compatibilelor IBM PC în care să nu existe cel puțin un produs purtând prestigioasa semnătură. O succintă (și desigur, incompletă) enumerare a acestora va fi, sperăm, edificatoare: MS-DOS (sistem de operare), LAN-Manager (sistem de operare în rețea), Windows (suprafață utilizator grafică), Multiplan, Excel (programe de calcul tabelar), Word, Word for Windows (editoare de texte), Works (sistem integrat), Project, Project for Windows (programe de administrare proiecte), Powerpoint (program de prezentare grafice), Windows Paintbrush (program de pictură și desen), MS-Basic, MS-Fortran, MS-Cobol, MS-C (compilatoare), Windows SDK (programare Windows), SQL Server (server de baze de date). Completăm această enumerare cu sublinierea că cele mai multe dintre aceste produse se plasează pe primele locuri în topul preferințelor utilizatorilor de compatibile IBM PC (v. nr. 4/91 al revistei "if", pag. 4) și cu mențiunea că pentru Word, Excel, Powerpoint și Works există versiuni Apple Macintosh care însă nu epuizează gama de produse Microsoft destinate acestor microcalculatoare.

Însă orice prezentare a firmei Microsoft este incompletă în lipsa conturării personalității aceluia care îi este fondator, președinte, director executiv, cel mai important acționar și, probabil, angajatul care muncește cel mai mult: **William H. Gates**. Cel care este supranumit "Billion Dollar Gates" este considerat în momentul de față persoana cea mai influentă în industria calculatoarelor. În acest context, personalitatea sa a dat naștere unei game foarte largi de atitudini, de la admirație și respect



Bill Gates

la suspiciune și invidie (fenomenul este explicabil, dacă nu chiar normal). În rândurile care urmează sînt cuprinse cîteva dintre cauzele și dovezile acestei capacități de a influența direcțiile de dezvoltare ale industriei calculatoarelor.

Din cei 60-70 de miliardari din S.U.A. 5 aparțin industriei calculatoarelor iar 2 dintre aceștia din urmă, Bill Gates și asociatul său, Paul Allen sînt de la Microsoft (cei alți trei sînt William Hewlett, David Packard și H. Ross Perot de la EDS). Gates a devenit cel mai tînr miliardar "self-made" din istorie în 1986, cînd Microsoft a devenit companie publică. Avea 30 de ani. Astăzi, averea sa este estimată la 2-3 miliarde dolari, aproape în întregime plasată în acțiuni Microsoft. Salariul și celelalte drepturi convenite funcției de director executiv sînt, prin comparație (cu cele 2-3 miliarde, nu cu salariul dvs, evident), modeste: 207.000\$ în 1989. Puternica personalitate a lui Gates a fost conturată în aproape 50 de editoriale de revistă în 1990. Publicații americane cum sînt: Computer Retail News, Electronic Business, Personal Computing și Upside îl plasează cu regularitate printre cei mai influenți factori executivi, deseori considerîndu-l cel mai influent. PC Magazine i-a decernat în 1986, deci la numai 30 de ani, al său Lifetime Achievement

Award, lucru ce poate părea surprinzător dacă ne gîndim că premiile de acest gen semnifică aprecierea și recunoașterea unei activități de viață. Dacă însă avem în vedere remarcă unui colaborator al lui Gates din perioada de început a industriei calculatoarelor personale, potrivit căreia atunci "un an era o viață" faptul apare explicabil. După cum explicabilă este (mai mult pentru ei decît pentru mulți dintre noi) apariția, prevăzută pentru 1992, a unei biografii cuprinzătoare (500 pagini) a celui care în anul apariției va împlini 36 de ani.

Și dacă tot am ajuns la biografie, să punctăm rapid ascensiunea lui William Gates: absolvent al Universității Harvard, elaborează împreună cu Paul Allen și Marty Davidoff un Basic pentru unul dintre primum-type tipuri de calculatoare personale, MITS Altair, după care se mută împreună cu Allen la Albuquerque pentru a continua dezvoltarea de software pentru MITS. Revenit acasă la Seattle reușește să obțină pentru mica sa firmă, MicroSoft (așa se scria atunci), contractul pentru elaborarea sistemului de operare pentru PC-urile gigantului IBM, devansînd firma Digital Research. Cumpără apoi de la Tim Paterson sistemul de operare QDOS (Quick-and-Dirty (?) Operating System) pentru 50.000\$, pe care îl va folosi ca bază pentru MS-DOS.

Acesta a fost începutul. Astăzi Microsoft are 5-6 mii de angajați și vânzări al căror volum depășește un miliard de dolari anual. Este una dintre cele mai profitabile companii din industria de calculatoare, dacă nu cea mai profitabilă. O rată de creștere de 31% și un profit de 22,3% pentru acționarii Microsoft în ultimii 5 ani sînt numai două dintre argumentele care vin în sprijinul acestei afirmații.

(continuare în pag. 9)

Strategia Microsoft

Strategia de viitor a firmei Microsoft, la capitolul sisteme de operare (S.O.), ar putea fi rezumată astfel: Windows pe toate platformele hard. Acesta este răspunsul celor de la Microsoft la campania de promovare a sistemului de operare OS/2, declanșată de IBM.

Încă din 1985, IBM și Microsoft își făcuseră planuri comune de promovare a S.O. OS/2, ca urmaș al S.O. MS-/PC-DOS. După mulți ani de speculații, de teste soft și de discuții se pare că drumurile celor doi uriași se vor despărți. În ultimul timp ei vorbesc mai des unul despre altul, decât unul cu altul. La o expoziție de PC-uri organizată la New York, IBM a arătat, mai mult pe sub mână, software care trebuia să demonstreze cât de ușor poate provoca Windows căderea sistemului. Numele programului, după cum afirmă apropiații casei, este "Bad App". Pe de altă parte, la o întrunire cu analiști ai pieței, la sfârșitul lunii iulie, Microsoft a prezentat programul "Terminator", care demonstrează cât de instabil este OS/2. Atât Bad App cât și Terminator au fost realizate pentru necesități proprii și nu pentru a fi comercializate. A vorbi despre un divorț între cei doi parteneri este, totuși, prea devreme. Atât doar că drumurile lor se despart. Momentan pot fi reținute, ca fapte, și respectiv declarații de intenție, trei aspecte: mai întâi că nu va exista un sistem de operare cu numele OS/2 3.0 produs de Microsoft. Locul acestuia va fi luat de Windows NT (New Technology). În al doilea rând Windows va fi astfel dezvoltat încât să poată fi utilizat pe o paletă hard cit mai largă. În acest scop vor exista două linii Windows: Windows X.X și Windows NT (New Technology). În al treilea rând, imediat după terminarea lui OS/2 2.0, IBM și Microsoft îl vor oferi spre vânzare pe căile obișnuite, Microsoft prin intermediul OEM, deci al producătorilor hard care au cumpărat licența, și IBM prin inter-

mediul canalelor proprii de comercializare. Strategia Microsoft, revăzută, va funcționa deci abia pentru urmașii lui OS/2 2.0.

Microsoft se vede obligat să-și revizuiască strategia, în privința sistemelor de operare, ca urmare a "impresionantului succes" pe care l-a avut Windows 3.0. Momentan, în lume, există cca. 4 milioane de instalări Windows 3.0 la cca. 70 de milioane de PC-uri DOS.

Windows va fi acceptat ca standard. După datele furnizate de institutul de prospectare a pieții IDC, în ultimul an, în Germania, au fost vândute 1,5 milioane PC-uri cu procesoare Intel 80286/80386 și 80486 și cca. 300.000 de copii Windows. În 1991 se va ajunge la 600.000 de copii Windows. După cum afirmă specialiștii, pînă la sfârșitul acestui an pe o cincime din PC-uri se va lucra sub Windows. IDC apreciază că pînă în 1995 în Germania vor exista 3 milioane de instalări Windows pe PC-uri și că Windows va fi utilizat în proporție de 40-50% pe piața profesională și neprofesională. Încă din 1993 80% din sistemele noi vor fi livrate cu Windows.

Windows va deveni un standard, deoarece suprafața utilizator grafică va fi acceptată de mulți producători, proiectanți și utilizatori. Cu toate că la sfârșitul lui 1990 Windows 3.0 era pe piață de abia 8 luni, suprafața utilizator grafică a lui utilizată în aplicații a adus o cifră de afaceri de 400 de milioane dolari. Aplicațiile Macintosh au atins o cifră de afaceri de 463 milioane dolari. Diferența între cifrele de afaceri realizate cu aplicații orientate caracter și aplicațiile grafice este încă semnificativă. În cazul livrărilor de produse noi balanța se va înclina, însă, în favoarea aplicațiilor grafice. Această tendință este sprijinită de uneltele din domeniul Enduser-Programming (uneltele pentru programatorii de aplicații),

cum ar fi de exemplu, Object Vision, Oracle Card, Toolbook, Object Script sau Visual Basic.

În timp ce Windows și aplicațiile realizate cu acesta pot da impulsuri semnificative pieței, după Microsoft, OS/2 nu a avut nici pînă astăzi ecoul așteptat. La sfârșitu! lui 1990 existau cca. 600.000 de instalări OS/2. Această situație s-ar putea schimba odată cu livrarea, la sfârșitul acestui an, a lui OS/2 2.0. Succesul lui OS/2 va depinde de calitatea sa și de respectarea promisiunii de a se putea rula sub el fără probleme, și fără modificări, aplicațiile DOS și Windows. În acest caz n-ar mai fi necesar ca Microsoft să aducă pe piață aplicații cu PM (Presentation Manager). Teama că aplicațiile Windows ar putea într-adevăr să ruleze mai bine sub OS/2, a fost un alt motiv care a determinat Microsoft să-și revadă strategia de viitor la capitolul sisteme de operare. Dacă ceea ce promite IBM se va întâmpla, va fi rău pentru piața Windows, dar va fi bine pentru Windows - API (Application Program Interface).

Important este, de asemenea, dacă IBM-ului îi va reuși să cucerască proiectanții de soft pentru OS/2, pentru a scrie aplicații specifice OS/2. Deoarece pentru ca OS/2 să se poziționeze mai bine decât Windows, vor trebui să existe suficiente aplicații atractive pe piață. În caz contrar doar avantajele care vor decurge din faptul că aplicațiile Windows vor putea fi rulate mai rapid sub OS/2 și că va fi posibil un multitasking real, nu vor fi suficiente pentru a determina o "masă critică" de utilizatori să lucreze cu OS/2. Din aceste motive este de înțeles interesul pe care-l are Microsoft ca Windows să poată concura OS/2.

IBM promovează S.O. OS/2 pe piață cu toată puterea. În SUA, cine schimbă vechiul PC-DOS cu OS/2 1.3 nu trebuie să plătească decât

99\$. Atunci cînd va fi disponibil OS/2 2.0, actualizarea va fi făcută gratuit. IBM duce o agresivă, dar acceptabilă din punctul de vedere al utilizatorilor, campanie de prețuri pentru a acapara pentru OS/2 o parte din sectorul de piață deținut de Windows.

Conflictele de interese dintre IBM și Microsoft, în ceea ce privește Windows și OS/2 nu pot fi negate. Dacă situația se va schimba după anunțarea noii strategii Microsoft, rămîne de văzut, deoarece abia dezvoltarea în continuare a S.O. OS/2 2.0 este afectată. Cu Windows, sau cu OS/2 2.0 pot fi utilizate cca. 90% din aplicațiile de pe PC-uri. Numai pentru restul de 10% va fi necesar OS/2 sau Windows NT ca sistem de operare server, pentru High-End-Multitasking, pentru comunicații PC-Host etc.

Atît IBM cît și Microsoft sînt interesate în crearea unui S.O. portabil care să fie privit ca o dezvoltare tehnologică a lui OS/2 2.0. Acest proiect ar putea fi susținut și de IBM cu toate că pînă acum nu s-au făcut confirmări oficiale. În această direcție se îndreaptă acordurile încheiate între IBM și Borland. crearea unui sistem de operare portabil, inclusiv pentru procesoarele RISC. Cît de mult vor fi afectate relațiile de cooperare de viitor, între IBM și Microsoft, nu se poate ști încă.

Controversat este modul de introducere pe piață a S.O. OS/2. Microsoft dorește ca migrația să se

realizeze via Windows, în timp ce IBM ar dori să înlocuiască imediat S.O.DOS cu OS/2. O premisă importantă, pentru Microsoft, este ca noile produse să rămînă compatibile cu cele deja existente pe piață, pentru a putea fi utilizate în continuare produsele instalate. Din acest motiv vor exista două linii Windows: Windows X.X. și Windows NT. Windows X.X. este gîndit pentru marea majoritate a aplicațiilor PC. Suplimentar va fi disponibil Windows cu "extensii.Pen" pentru utilizare de Notepads. Se are în vedere și includerea HDTV (High Definition TV), controlul și comanda copiatoarelor, fax-urilor, telefoanelor și a altor mijloace de muncă și organizare din birouri după cum a afirmat Bill Gates, la o conferință de presă în Redmond, USA, la sfîrșitul lunii iulie. Este de imaginat și o deschidere spre jocurile video, pentru a putea satisface astfel întreaga piață de bunuri de consum tradiționale. Deviza Microsoft în prezent: Windows everywhere, ca sistem scalabil, inclusiv multimedia (integrare de imagini, vorbire și date), Pen-Technology și orientare obiect, într-o perspectivă de timp medie. Baza o vor constitui în continuare Windows-APIs (Application Program Interface).

Windows NT, dimpotrivă, mizează pe un nou sistem de operare, de aici și denumirea New Technology și va fi un sistem de operare High-End (de vîrf) pentru servere și aplicații complexe pe stații de lucru. După datele furnizate de Microsoft, Windows NT va fi

scris în C și va fi astfel portabil și pentru alte platforme hard. Încă din acest an vor fi disponibile SDK-uri (Software Development Kits) pentru Windows NT. Pe un produs finit se poate conta abia pentru anul viitor.

Windows NT corespunde astfel cu ceea ce Microsoft desemnase a fi OS/2 3.0 și care urma să fie unul din cele două sisteme de operare de promovat în cadrul inițiativei ACE (Advanced Computing Environment). La conferința de presă de bilanț a firmei Microsoft, de la sfîrșitul lunii iulie, din Redmond, SUA, Bill Gates nu a respins categoric posibilitatea portabilității și pe alte procesoare RISC, de exemplu pe procesoarele SPARC.

Din cauza conflictului de interese apărut între IBM și Microsoft se pune întrebarea: într-o întreprindere, în viitorul apropiat, trebuie să se mizeze pe Windows sau pe OS/2? Interesant va fi pentru utilizatorii care folosesc în prezent OS/2 1.X într-o versiune OEM provenită de la Microsoft și care vor dori să treacă apoi la Windows NT. Pentru aceștia va trebui să existe un program de conversie (Migration Layer) care să le dea posibilitatea de a converti aplicațiile create cu Presentation Manager pentru Windows NT. Sarcina lui Migration Layer: conversia PM-API în Windows-API.

(R.M.)

Factorul 20%

(continuare din pag. 7)

În ceea ce privește influența firmei, cele peste 60 milioane de calculatoare pe care rulează software Microsoft reprezintă numai partea vizibilă a acesteia. Cercetările firmei în domeniile multimedia, comunicarea inter-aplicații, interfața utilizator pentru "creionul electronic" (pen based

computing), standarde CD-ROM, dovedesc că Microsoft, ca orice firmă care are de apărut o poziție conducătoare, privește în perspectivă.

Revenind la personalitatea lui William Gates, ceea ce ni se pare a fi lucrul cel mai important este ponderea acestuia în cadrul firmei, apreciată la 20%. Cînd Gates este considerat "factorul 20%" nu se face referire la volumul acțiuni-

lor sale, acesta ridicîndu-se, de fapt, la 38% ci reprezintă o evaluare a importanței acestuia pentru firmă. Înseamnă că din orice se face sau se poate face la Microsoft, o cincime este sau va fi realizată de un singur om. Ceea ce, veți recunoaște, este extraordinar!

(Răzvan Grigorescu, Octogon, Tîrgu Mureș)

Protecția programelor

Avem actualmente lipsuri de tot felul, nu ne putem plînge! Dar știm bine că resursele sînt nu numai materiale, umane și financiare, ci și informaționale (neglijate pînă acum la noi). Sub acest aspect, există la noi un imens vid de legislație, o mare lacună în cultura juridică. Între domeniile aflate în umbră, este și protecția intelectuală. Ceva interesant s-a discutat pe această temă la REP '90 (Ediția a XI-a), 7- 9 sept. la Snagov. Fiind vorba de consemnări de discuții, textul este în mod inerent incomplet.

S-a vorbit desigur despre etapa actuală: Cel mai mare pericol este provizoratul. Se știe că provizoratul tinde să devină definitiv (starea de provizorat este cea mai sigură). Riscul cel mai mare este să prelungim prea mult această tranziție... Despre **tranziția plan-piață**: Fenomenul este greșit înțeles, dacă primul pas se referă la micul întreprinzător. Dacă se începe cu această atomizare, risipire, dispersare a forțelor, necuplată cu o demonopolizare a proprietății de stat, înseamnă că ne reîntorcem la perioada de dinainte de '40. Fără o piață liberă, nu poate fi vorba de o libertate reală, nici în economie și nici în general...

În condițiile trecerii de la plan la piața liberă, crește rolul mijloacelor de protecție intelectuală și, concret, de protecție industrială. Schimbarea de concepție la nivel general implică și o schimbare în legislație. Există preocupări, îngrijorări. Legislația este în curs de revizuire, s-a făcut un proiect de lege a invențiilor. A fost publicat, dar ecurile au fost foarte slabe.

În 1985, pe plan mondial, s-a încercat introducerea unui sistem de protecție pentru plachete și circuite integrate. Cei mari, SUA, Japonia au realizat legislația, cei mici au vrut să scape, au subminat, au boicotat proiectul. Americanii le-au dat ultimatum: altfel nu le mai

vînd cipuri. În '87- 88 toți din CEE au trebuit să accepte protecția, să adopte legi naționale, dar fiecare cum vroia. În 1988 s-a simțit nevoia unui sistem unificat de protecție. În 1989 (26 mai, la Washington) s-a semnat Tratatul internațional asupra protecției industriale asupra CI. În consecință, a fost trimis spre consultare și la noi în țară un proiect de lege. De la OSIM s-a ocupat de această problemă V. Erhan, dar se pare că cei interesați n-au luat la cunoștință.

- - Din 100 de invenții, 5- 6 sînt foarte valoroase, ar putea să ne scoată din substanța aia în care eram... Dar inventatorii s-au lovit de..., așa cum s-a spus, invidia aceea perversă, nici măcar conducerea proprie nu i-a sprijinit. Din cauza necinstei.
- - Ai un produs brevetat - cine te împiedică să-l faci și să-l dai la export? În momentul de față, cine verifică dacă nu folosește altul brevetul tău? Legea actuală este făcută pentru economia socialistă, deci cedezi titlul întreprinderii, poate fi folosit de toți. Nu-i avantajos și nici stimulat: întreprinderea titulară plătește taxe, n-are nici un avantaj. Dacă titlul ar fi personal, respectivul ar urmări aplicarea...

Protecția programelor de calculator

Și aici, americanii au impus protecția. Nu există încă un tratat internațional, dar avem exemple concrete. Cehoslovacia, pentru a obține clauza națiunii celei mai favorizate, era condiționată (între multe altele) și de rezolvarea protecției programelor de calculator prin legea drepturilor de autor.

La OSIM, de problema aceasta se ocupa Dan Șova și Doina Manicată.

- Cum se poate face protecția produselor software?

- În nici un fel! Legislația actuală din țară **NU** poate asigura protecția programelor. Nu se poate aplica legislația privind protecția invențiilor. Pe plan mondial, cei mari încearcă să forțeze protecția produselor de informatică.

Literatura privitoare la brevetabilitate afirmă: *NU se poate. Softul nu constituie o soluție tehnică în accepțiunea brevetabilității*. O soluție tehnică brevetabilă se referă la aspecte concrete ale tehnicii, iar în programare se lucrează cu abstractizări. Sînt brevetabile soluțiile tehnice concrete, palpabile fizic. În general nu sînt brevetabile **noile aplicații**, ci revendicarea lor.

De altfel, invențiile pionier sînt foarte rare, mult mai frecvenți sînt pașii tehnici de perfecționare. Remediile, ameliorarea unei soluții cunoscute. De aceea, în revendicare se descriu părțile cunoscute ("prior art") + elementele noi; de exemplu perfecționarea unei scheme bloc.

O nouă aplicație pentru o schemă bloc cunoscută **NU** se brevetează: să presupunem că avînd un calculator pentru jucat șah, iau Epromul, îl reprogream și îl folosesc pentru ghidat rachete... nu pot să protejez noua aplicație!

La soft, nu există o nouă soluție tehnică concretă, există doar o nouă utilizare, care s-ar putea breveta dacă duce la efecte deosebite pentru societate, dacă se fac cheltuieli foarte mari pentru programare. Dar utilizatorii nu vor, obstrucționează.

Există posibilitatea protecției prin drepturi de autor. Primele decizii ale tribunalelor **NU** s-au referit la programele propriu-zise, ci la calculatoare și programe. De exemplu: avem un desen tehnic, îl putem considera o "operă de artă", la fel o melodie, un film de desene animate...

Au fost programe brevetate, adică li se conferea recunoașterea priorității internaționale.

În 1990, țara noastră a fost (?) la Geneva pentru protecția pe baza drepturilor de autor. În CAER au fost discuții, orientate spre *cum să NU se adopte*.

Sînt protejate programele care reflectă un proces tehnologic, deci în care se regăsesc pașii de proces respectivi - se poate considera că se descrie o metodă. Din tot programul, elementul de protecție este revendicarea, dacă nu descrii programul în întregime, cu toți cei 10.000 de pași pe care-i are, înseamnă că nu este protejat.

Această situație nu-i acceptată nicăieri în lume. Din '85 s-au făcut presupuneri (?), modificări în legislația privind drepturile de autor: Japonia, RFG. Concret, nu s-a realizat mare lucru, cu excepția legii japoneze, care cere precizări amănunțite privind programul. Solicitantul trebuie să depună o cerere și programul este înscris într-un registru în care sînt înregistrate drepturile de autor.

În privința drepturilor de autor, exista două concepții principale:

- a) cea continentală: acordul semnat la Berna în 1890 prevede protecția operelor literare, artistice și științifice fără înregistrare. Durata = viața autorului + 50 de ani pentru moștenitori. Este limitată în timp și spațiu. Timp: 15 ani la noi, în alte țări 8, 12 sau 20 de ani. Este în studiu reglementarea armonizării legislației - să se standardizeze la 20 de ani (sîrbii aveau 8 ani). După această perioadă, operele intră în patrimoniul public, se pot consulta și folosi gratuit. Sînt în studiu următoarele aspecte:

- » - cum se face înscrierea;
- » - ce aplicații folosește(?);
- » - cum se justifică.

Bush a acordat clauza Cehoslovaciei, deci au acceptat protecția programelor!

În consecință, concepția continentală prevede nu înregistrarea, ci recunoașterea drepturilor de autor pentru o operă, din momentul în care a fost realizată.

- b) cea engleză, cu Copyright. Este o noțiune puțin deosebită: nu înseamnă recunoașterea auto-

mată a oricăror realizări, dacă o realizare nu a fost înregistrată, nu este recunoscută. Trebuie să se facă înregistrarea în toate țările în care se dorește ca opera să fie protejată. Deci, dacă un program n-a fost înregistrat în țară la noi, nu este protejat.

- Noi vrem să facem programe și să le protejăm, dar le facem utilizînd un soft de bază. De unde-l avem? Nu este și el protejat?

În '85 s-a început lucrul la protecția circuitelor integrate, în '87, sub presiunea americanilor și japonezilor, au fost adoptate 7 legi naționale în Europa de West. În '89 a fost semnat Tratatul de protecție pentru circuitele integrate. A fost semnat și de țara noastră, astfel că de acum înainte utilizarea unui circuit integrat copiat într-un aparat (de larg consum, tehnică de calcul, etc) poate crea grave probleme dacă aparatul, chiar brevetat, ajunge în West.

Clauze privind înregistrarea circuitelor integrate: nu mai tîrziu de 2 ani de la utilizarea lui comercială. Se brevetează arhitectura, structura, măștile. N-are nimeni dreptul să zică: "N-am știut!"

Situația este similară și pentru soft. Trebuie stabilit clar UNDE, cine păstrează, care-s condițiile de păstrare.

Se fac excepții:

- » - în scop de cercetare se poate analiza o lucrare, pentru ca să nu se cadă din întîmplare peste aceeași idee;
- » - pentru învățămînt.

Trebuie să cunoaștem toate porțițele!

Trebuie să existe o societate pentru protecția drepturilor respective, așa cum sînt Uniunea Scriitorilor sau Uniunea Compozitorilor, să aibă un birou care să se ocupe de verificarea respectării drepturilor de autor.

Deci, după adoptarea legislației respective, va trebui stabilit un cadru executiv corespunzător, o **Poliție de Soft!**

De exemplu, pentru muzică: compozitorii au un salariat care ascultă toate posturile de radio, notează că "bucata X a lui Cacoveanu a fost cîntată ieri 25" pe postul București III", le percepe taxele corespunzătoare, iar la sfîrșitul anului îi spune lui Cacoveanu că are de luat drepturi de autor de 28.000 lei, din care se scade taxa de urmărire de 10%.

Așadar, trebuie ceva similar și pentru programe. Depozitar: ceva gen BNP. A depozita legal înseamnă să ții sigilat, răspunderea este foarte mare, este o problemă de încălcare a unor drepturi. Cum se aduce la cunoștință unui potențial beneficiar? Trebuie spus că programul respectiv a fost înregistrat, la ce poate fi folosit... trebuie făcută publicitate, reclamă (să se sublinieze avantajele: viteză, precizie, economie de memorie) pentru ca beneficiarul să cumpere sau să închirieze programul.

Sînt tot felul de probleme

- » - biblioteca are drept de copie-re?
- » - folosim un limbaj de programare. În Japonia, acestea sînt excluse de la protecție!
- » - cum se poate verifica folosirea ilegală a unui program?
- » - intră și mediile de programare în această categorie? Dacă fac o aplicație în Data Base și vreau s-o vînd. Trebuie să vînd și dBase-ul beneficiarului? Trebuie să-i spun cumpărătorului sub ce mediu a fost elaborată aplicația?
- » - dacă vreau să export în West un program făcut cu DataBase IV, care-i furat, intru în pușcărie? Nu, utilizatorul trebuie să-l cumpere?
- Nu sîntem noi cei mai mari hoți. Măcar să fi fost, am fi acum precum Hong-Kong-ul sau Coreea de Sud! Mai este în vigoare decretul din '63, pe care nu-l respectă nimeni, nu ține cont de legislația internațională...

(Cristian Malide, ICCE)

L40 SX - una dintre cele mai frumoase surprize ale firmei IBM

După numeroase tentative nereușite, Big Blue a apărut cu un nou model, un laptop autonom care confirmă revenirea numărului unu pe piața explozivă a calculatoarelor portabile.

În 1990 s-au vândut doar 10.000 de laptop-uri autonome având la bază microprocesorul 80386SX, astfel că apariția lui L40SX, cu funcționalități standard, pe care concurenții le oferă opțional, va declanșa o luptă acerbă între producători.

L40 dispune de un microprocesor 80386SX cu frecvența de ceas de 20 MHz. În versiunea de bază el conține un modem/fax al firmei Com 1 livrat într-un Kit complet hard-soft. El permite transferul de fișiere la viteza de 2400 bds (V23, V21, V22 și V22 bis).

Protocoloalele MNY și S de corectare a erorilor și de compresie a datelor sînt integrate, de asemenea un modul V27 ter/29 care permite accesul la Fax Groupe III. Light Manager al lui Com 1 funcționează sub MS DOS și poate deveni în acest caz rezident. Suportă de

asemenea "en tache de fond" emisia în timpul recepției fax. Telecopiile pot fi transmise imediat după prelucrarea cu procesorul de texte Evolution (IBM). Memoria internă este de 2 Mo (80 ns) extensibilă prin două slot-uri pînă la 18 Mo.

Afișarea este făcută pe un ecran LCD STN de 10", spre deosebire de celelalte portabile care au doar 8" sau 9". Ea permite modul VGA cu 32 niveluri de gri. Ca la toate ecranele LCD pasive este imposibilă evitarea efectului "fantomă" în timpul deplasării cursorului. IBM a găsit însă un subterfugiu: imaginea pointerului se estompează în timpul mișcării, această funcțiune fiind parametrizabilă.

Harddisk-ul de 2 1/2" al firmei Conner Peripherals are o capacitate de 60 Mo cu timpul de acces de 19 ms. Lectorul de dischete este de 3 1/2" 1,44 Mo. Probabil lectorul de 2,88 Mo care este așteptat și care va funcționa sub DOS 5.0, va fi opțional în viitoarea configurație.

Referitor la tastatură, IBM nu a acceptat nici un compromis privind talia

tastelor și amplasarea lor. Doar tastele de deplasare a cursorului au fost mutate puțin mai la stînga. Tastatura numerică separată este furnizată în standard ceea ce constituie o premieră în domeniul portabilelor. Ea se poate conecta la portul rezervat mouse-ului. Aceasta nu înseamnă că utilizatorii sub Windows nu îl pot folosi, deoarece acesta se poate conecta într-o priză existentă în spatele modulului ce conține tastatura numerică. De asemenea există: o ieșire paralelă, una serială și una pentru un ecran VGA extern. Utilizatorii pot alege două carcase pentru plăci externe conform standardelor ISA și MCA.

Originală este existența pe panoul frontal a unor diode LCD care pot indica permanent: starea de încărcare a bateriei, accesul la harddisk și la lectorul de dischete, lucrul cu modemul, funcționarea difuzorului încorporat, starea de deflare a ecranului, modul pauză și modul de lucru cu tastele numerice și majuscule. Există de asemenea o diodă de alarmă în cazul depășirii temperaturii sau umidității. La nivelul acestor diode este un comutator care permite gestionarea energiei. Viteza variază, în mod automat, în funcție de încărcarea sistemului (5, 10 sau 20 MHz). Acest mod permite reducerea cantității de energie utilizată prelungind astfel viața bateriei.

L40SX dispune de o baterie NiCd care în mod normal (la 20 MHz) poate funcționa trei ore; cu ajutorul sistemului de gestionare a energiei durata de funcționare poate ajunge la opt ore. În timpul înlocuirii acumulatorului o baterie tampon furnizează energia necesară calculatorului.

În sfîrșit, trebuie să menționăm că portabilul este livrat fără nici un sistem de operare, astfel că utilizatorii pot face o alegere între DOS și OS/2. Ce se mai poate aștepta în anul viitor de la IBM? Probabil un frate sau un văr sau de ce nu un notebook?

(Voicu Manole)

Tabel comparativ (Laptop 386SX - 20 MHz)

	AST	COMPAQ	DELL	GRID	IBM	TOSHIBA
Model	Premium Exec	STL 386 S/20	320N	1550SX	L40SX	T3200SX
Memorie standard	2 Mo	2 Mo	1 Mo	2 Mo	2 Mo	1 Mo
Memorie maximă	8 Mo	10 Mo	5 Mo	8 Mo	18 Mo	13 Mo
Harddisk	60 Mo	60 Mo	40/60 Mo	60/120 Mo	60 Mo	120 Mo
Lector de dischete 3.5" (1.44 Mo)	da	da	da	da	da	da
Ecran	LCD 32n*	LCD 16n	LCD 16n	LCD 32n	LCD 32n	plasmă
Port video VGA	da	da	da	da	da	da
Port serial	1	1	1	2	1 (2 opț.)	2
Port paralel	1	1	1	1	1	1
Port mouse	da	da	da	nu	da	nu
Modem intern	nu	nu	nu	nu	da	nu
Tastatură numerică separată	opțional	opțional	opțional	opțional	da	opțional
Carcasă de extensie	nu	da (opț.)	nu	da (opț.)	da (opț.)	nu
Autonomie	3 h	3 h	4 h	3 h	3 h	3 h 30'
Greutate	3 kg	3.4 kg	2.9 kg	5.4 kg	3.5 kg	1.9 kg
Preț	4050\$	7000\$	3250\$	8100\$	6000\$	6015\$

(* nuanțe)

PC sau telefax?

- Cartele fax -

Ca alternativă la telecopiator ca echipament separat, PC-ul - dotat cu hard și soft corespunzător - câștigă din ce în ce mai mult teren ca emițător și receptor de telecopii. "if" vă prezintă avantajele și dezavantajele cartelelor fax.

Avantajele faxului față de corespondența obișnuită sînt evidente. Telefaxul nu este numai una din cele mai rapide variante de a trimite texte, grafice, fotografii, scrisori, etc. oriunde în lume (teoretic), ci, deseori, o variantă chiar mai ieftină

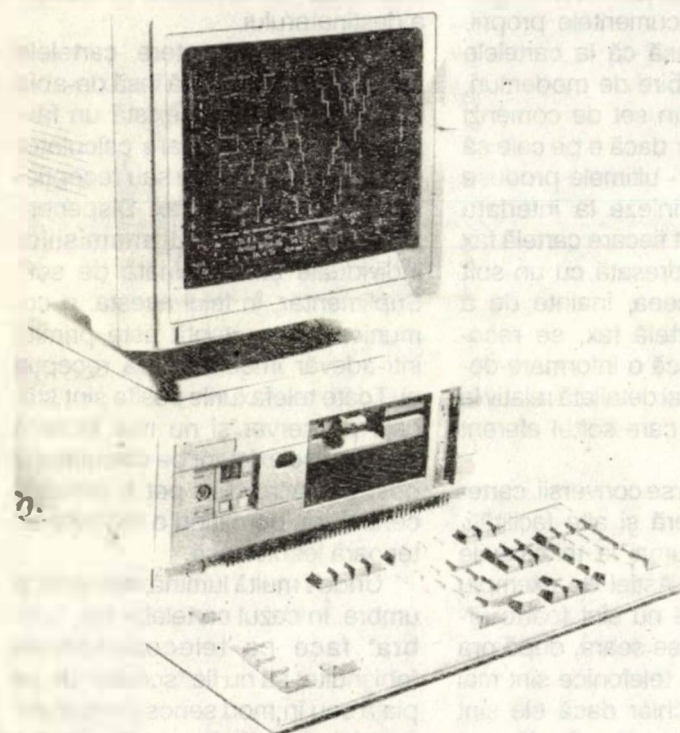
celălalt capăt al liniei, echipamentului receptor transformă informațiile sosite într-o copie a paginii originale, pe care apoi o tipărește la o imprimantă. Astfel, fiecare participant primește o copie fidelă a originalului. De aici și numele: "fax" provine

de la "facsimil", termen folosit mai ales în industria tipografică (cf. DEX: "reproducere exactă a unui text, a unei semnături, a unui desen sau a unei picturi cu ajutorul fotografierii, al fototipiei, prin copiere manuală etc.")

Pentru ca telecopiatoarele din toate țările să se poată înțelege între ele, în anii '70s-a făcut normarea pe plan mondial a serviciilor telefax. Echipamentele au fost împărțite în 4 grupe (vezi cartea).

Pe cît de convingătoare sînt avantajele telefaxului - pentru utilizatorul unui PC, transmisia prin telefax înseamnă un procedeu mai degrabă anevoios. Înainte de a putea trimite o comunicare prin fax, ea trebuie să existe pe hîrtie, deci trebuie întii tipărită (imprimată). Apoi hîrtia trebuie dusă la telecopiator - de regulă, în alt birou sau chiar în altă clădire. De-abia după ce imprimatul a fost introdus în telecopiator și s-a format numărul corect al destinatarului, începe transmisia propriu-zisă.

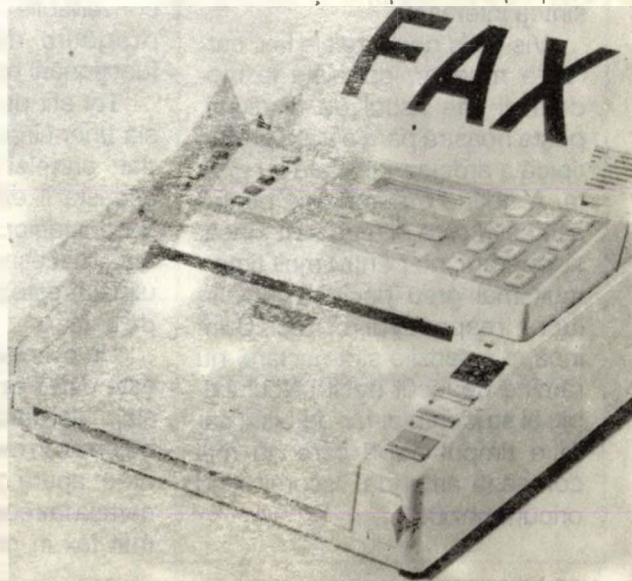
Aceste dificultăți pot fi evitate (aproape) fără probleme, folosind telecopierea prin intermediul PC-ului. Pentru a transfera un PC într-un telefax, este nevoie doar de o cartelă fax corespunzătoare, softul aferent și - desigur - o linie telefonică la care să se facă conexiunea. Desigur, este nevoie - teoretic și de aprobarea poștei.



În prezent, în Germania sînt instalate peste 700.000 de telecopiatoare; se așteaptă ca la sfîrșitul acestui an să se depășească numărul de 1.000.000. În lume, la începutul anului 1992 vor fi instalate peste 12.000.000 telefaxuri. La noi, extrapolînd numărul de faxuri instalate în județul Mureș - 100 - se poate aproxima la 6.000 numărul de telefaxuri instalate în țară.

deci corespondența obișnuită. O scrisoare către un partener de afaceri din același oraș ajunge prin fax în numai cîteva minute și costă, cu aproximație, 1 leu. Cu poșta normală, ar ajunge (cel devreme) a doua zi și ar costa 4 lei.

Pentru a fi transmis ca fax, originalul este "citit" (scannat) de telecopiator, transpus într-un format propriu și apoi trimis pe linia telefonică. La



Ce spune poșta

La noi, cartelele fax încă nu au ajuns să fie cunoscute de ministerul sau forul de resort. Sau, cel puțin, nu și-a făcut în nici un fel publică știința.

În prezent, sînt autorizate telecopiatoare standard (lista am publicat-o pe coperta 3 a numărului 6 din "if") și modemuri standard (lista poate fi consultată în "if" nr. 7). Desigur, aceste liste se actualizează în permanență; amănunte la zi ar trebui să poată dea oricare poștă județeană (filiala RomPres Telecom) sau, respectiv, Ministerul Comunicațiilor, Direcția Generală de Reglementări, București (această adresă pe un plic este suficientă ca să ajungă la destinație).

Taxa de instalare - dacă postul telefonic există - este de 1000 lei pentru un telecopiator și de 1800 lei pentru un modem (2200 dacă e vorba de o instituție). Abonamentul lunar este de 200 lei/lună pentru un telecopiator, respectiv de 800 lei/lună pentru un modem.

În afară de abonament, se mai plătesc, desigur, și impulsurile consumate - la tariful obișnuit, 60 bani/impuls (toate valorile amintite sînt orientative - ele au fost adevărate pe data de 11 septembrie 1991, ora 12. Pentru valorile momentului, vă rugăm să vă adresați poștei locale, dacă sînteți interesat).

Vis-a-vis de cartelele fax, cartelele modem, cartelele faxmodem și alte astfel de bîzdîcuri, poșta noastră pare a avea poziția tipică a ardeleanului față cu girafa. A obține o autorizație pentru așa ceva este în mod cert - încă, și, sperăm, nu pentru mult timp - mult mai greu decît a pune în lucru respectivul bîzdîc. Cum încă și amenziile sînt neclare, nu rămîne de făcut decît calculul tipic al studentului-făr-de-bilet: care e timpul după care nu mai contează amenda, "scorul" fiind oricum pozitiv ?...

Odată cu instalarea cartelei fax în PC, problemele mai sus amintite dispar: texte sau grafice pot fi concepute și transmise direct din PC, fără să fie necesară imprimarea prealabilă a documentului ce urmează a fi transmis. Importante și interesante sînt posibilitățile noi pe care le deschide un PC cu cartelă fax. Cînd un PC recepționează un fax, atunci acesta e disponibil într-un format grafic propriu. Cu un soft adecvat - cel puțin teoretic - un fax ce conține numai text ar putea fi convertit în fișier ASCII și folosit în procesorul de texte cu care se lucrează în mod curent. Deasemenea, graficele ar putea fi convertite în diverse formate pentru a fi înglobate apoi în documentele proprii. Trebuie spus însă că la cartelele fax, spre deosebire de modemuri, nu există încă un set de comenzi "standard" (chiar dacă e pe cale să "se nască" unul - ultimele produse încep să se alinieze la interfața CAS). Astfel încît fiecare cartelă fax trebuie să fie adresată cu un soft specific. De aceea, înainte de a cumpăra o cartelă fax, se recomandă să se facă o informare-documentare cît mai detaliată relativ la posibilitățile pe care softul aferent le oferă.

Pe lingă diverse conversii, cartelele fax mai oferă și alte facilități, întîlnite altfel numai la telefaxurile foarte scumpe. Astfel de exemplu comunicările ce nu sînt foarte urgente pot fi trimise seara, după ora 18, cînd tarifele telefonice sînt mai convenabile - chiar dacă ele sînt pregătite dimineața, în timpul funcționării normale a birourilor.

Tot atît de simplă este transmiterea unor circulare: textul este "pre-dat" cartelei fax o singură dată; aceasta îl expediază apoi tuturor destinatarilor. Selecția fiecărui abonat - necesară la telecopiatoarele uzuale - este superfluă, fiind preluat de soft.

Un avantaj clar al PC-faxurilor este calitatea mai bună a transmisiei. Căci dacă la un echipament fax obișnuit, în mod necesar, prin scannare, apare o pierdere de calitate, cartela fax convertește datele în format fax în mod optim. Dispare și

necesitatea imprimării prealabile a documentului - altă etapă care are ca efect o diminuare a calității imaginii de transmis. Astfel, pierderile de informație prin teletransmisie sînt reduse la un minim.

Și faptul că folosind cartelele fax se poate lucra fără hîrtie - sau, cel puțin, fără hîrtia specială, scumpă și poluantă, cu care lucrează telecopiatoarele obișnuite - constituie un avantaj important al cartelelor fax.

Suplimentar, multe cartele fax oferă prin softul aferent o carte de telefon a tuturor abonaților la serviciile fax. Astfel că se evită și căutarea și formarea manuală - uneori, incomodă - a numărului de telefon a destinatarului.

Adevărata lor putere, cartelele fax și-o demonstrează însă de-abia într-o rețea, în care există un fax-server. În rețea, fiecare calculator conectat poate emite sau recepționa comunicări prin fax. Dispecerizarea în timp a transmisiilor individuale este preluată de soft. Suplimentar, în felul acesta, o comunicare importantă este primită într-adevăr imediat după recepția ei. Toate telefaxurile sosite sînt stocate pe server și nu mai încarcă spațiul disc existent pe calculatorul post de lucru. Ele pot fi arhivate centralizat, permițînd o regăsire ulterioară lesnicioasă.

Unde-i multă lumină, mai apar și umbre. În cazul cartelelor fax, "umbra" face ca telecopiatoarele "obișnuite" să nu fie "scoase" de pe piață sau în mod serios concurate: cartelele fax rămîn mai degrabă o completare. Căci, în rutina zilnică a oricărui birou apar o multitudine de documente care vor trebui să fie transmise ca faxuri obișnuite și pe viitor - să amintim numai banalele semnături. Pentru a putea fi transmise prin intermediul PC-ului, aceste documente ar trebui scannate în prealabil. În acest caz, un telecopiator normal este, evident, mai comod. Mai ales că nici combinația cartelă fax + scanner nu este o alternativă pentru telecopiatoarele oferite la prețuri din ce în ce mai mici. Mai interesante - și din ce în ce mai răspîndite - sînt soluțiile care

folosesc combinația cartelă fax + telecopiator obișnuit, acesta din urmă preluând și funcția de scanner pentru PC, când este nevoie.

Cartelele fax mai au și alte slăbiciuni. Cine vrea să poată fi găsit la orice oră, trebuie să aibă PC-ul pornit și pregătit 24 de ore din 24. În plus, desigur, softul pentru controlul cartelei fax ocupă loc în memoria principală. Mulți utilizatori vor fi semnificativ incomodați de lipsa celor 100-200 coțeți pe care îi revendică, tipic, softul aferent unei cartele fax.

Cartela fax poate fi considerată ca alternativă pentru un telecopiator obișnuit numai în anumite cazuri, bine determinate și destul de rare. Ca o suplimentare judicioasă însă, ea și-a câștigat de pe acum un loc în PC - nu înlocuind telecopiatorul "standard", ci oferind o sumă de servicii și comodități suplimentare în avantajul utilizatorului de PC.

Listă de verificare

Dacă vă tentează achiziționarea unei cartele fax, poate că e util să treceți în revistă, punct cu punct, relația între ceea ce se oferă și necesarul dumneavoastră, relativ la:

- autorizarea de funcționare din partea poștei
- suprafața utilizator
- viteza de transmisie
- posibilitățile de conversie (PC Fax)
- imprimantele cu care poate lucra (tip, set de caractere, fonturi)
- posibilitatea de lucru în rețea (sisteme de operare, număr utilizatori)
- carte de telefoane și adrese
- repetarea automată a apelului
- mixajul de texte, grafice și imagini
- utilitare de scannare

- posibilitățile de editare (text, grafică)
- posibilitățile de înglobare în/interfațare cu aplicații deja existente
- inteligență proprie (procesor, RAM, etc.)
- transferul de fișiere
- posibilitățile de a lucra cu mai multe linii telefonice deodată
- necesarul de memorie principală
- raportul preț/performanță
- posibilitățile de utilizare a unui telecopiator autonom ca scanner și/sau imprimantă
- posibilitățile de emisie la anumite ore, programate
- posibilitățile de emisie a circulațiilor
- funcționarea în background
- jurnal de emisie/recepție

ing. *Fettich Iosif*

Ce facilități trebuie să ofere softul aferent unei cartele fax?

Cu cât e mai simplă suprafața utilizator, cu atât mai bine. Transmisia prin telecopiere fiind în genere foarte simplă - introdu hîrtia, apasă butonul, formează numărul, gata - nici soluția PC nu are voie să fie complicată - altfel, utilizatorul obișnuit va prefera în continuare varianta "clasică", cu utilizarea unui telefax obișnuit.

Un punct câștigat în ce privește acceptarea de către utilizatorii finali îl reprezintă posibilitatea de a putea transmite (sau recepționa) telefaxuri direct din cadrul aplicației cu care lucrează de obicei - ceea ce implică posibilitatea de a include în aplicații module care să lucreze cu cartela fax. Tendința accentuată spre suprafețe utilizator grafice face ca posibilitatea de a utiliza softul de fax de sub Windows să fie recomandabilă.

La transmisia de texte, ar trebui să existe nu numai varianta ASCII, ci și emulări ale unor imprimante matriceale sau laser (Epson, HP Laserjet). Numai astfel se atinge calitatea textelor obținute cu procesoarele de texte uzuale, făcînd posibilă utilizarea cartelelor fax pentru corespondența de afaceri.

Ca formate grafice, ar trebui să fie disponibile măcar formatele PCX și TIFF. Posibilități de conversie în/din programe de grafică precum GEM, Dr. Halo, PC Paintbrush nu sînt neapărat necesare, dar ușurează editarea faxurilor. În acest context, este

recomandabil și suportul soft pentru lucrul cu mouse.

Pentru realizarea unor scrisori de afaceri care să corespundă calitativ, posibilitatea de-a amesteca fără restricții grafice și texte este o necesitate. În plus, trebuie să existe posibilitatea combinării mai multor documente (prospecte, oferte etc.) într-unul singur.

Posibilitatea stabilirii orei de transmisie - pentru a economisi din taxe - este o necesitate, la fel cum nu ar trebui să lipsească o carte de telefon. Dacă aceasta în plus poate fi împărțită în grupe, devine mai simplă transmisia unor circulare la anumiți destinatari. Nu are voie să lipsească posibilitatea de a repeta în mod automat apelul, dacă acesta nu a reușit din prima încercare.

Un jurnal detaliat al transmisiilor permite o administrare rațională a documentelor.

Mai ales pentru lucrul în rețea, funcționarea în background a softului este indispensabilă.

Iar pentru ca aplicațiile uzuale să poată fi folosite normal, necesarul de memorie rezidentă nu ar trebui să depășească 100 ko.

Și, nu în ultimul rînd, manualul de utilizare ar trebui să fie conceput didactic, constituindu-se într-o documentație ușor inteligibilă a funcțiilor oferite.

Comunicații

	FSE Discovery 2448HT	OAFax V3	FSE AMS ZX 1896	ARC Dynamlink	JT Fax 4800	P+W Speedyfax
Preț (DM)	318	575	598	690	900	1497
Cerințe sistem						
Memoria principală (koct)	640	640	640	640	640	256
Slot	8 biți	8 biți	8 biți	8	8 biți	8 biți
Versiune MS-DOS	3.0	3.3	3.0	3.0		
Date tehnice						
Utilizabil ca modem	da	da	da	da	da	
Cablu telefonic	2,1 m / RJ11	2m / RJ11	2m / RJ11	2m / RJ11		3 m / TAE-8
Funcții soft						
Aplicație DOS		da			da	
Aplicație Windows		nu			nu	
Memorie ocupată ca TSR		72 ko			200 ko	
Preimprimare	nu	da	nu	nu		da
Transmisie de ecrane	nu	nu	nu	nu		da
Snapshot	da	da	da	da	da	nu
Intercepția imprimării	da	da	da	da	da	da
Preconversie	da	da	da	da		da
Conversie automată	da	da	da	da		nu
Conversie în background	nu	nu	nu	nu		nu
Format -> faxformat	PCX, TIF, BFX, DCX	ASCII, PCX, GEM, TIF	PCX, TIF, BFX, DCX	PCX, DCX	ASCII, PCX	ASCII, PCX, TIF, IMG
faxformat -> Format			PCX	PCX	PCX	PCX, TIF, IMG
Viteza de transmisie	2400, 4800 bps	2400,4800, 7200,9600 bps	2400, 4800, 7200, 9600 bps	2400, 4800, 7200, 9600 bps	2400,4800 bps	2400, 4800, 9600 bps
Moduri de transmisie	normal	normal/fin	normal	normal		normal / fin
Nivele de gri	nu	da	nu	nu		nu
Comprimare pagină	da (deselectabil)	da (deselectabil)	da (deselectabil)	da (deselectabil)		da (deselectabil)
Antet	da	da	da	da	da	da
Pagină-copertă	da (deselectabil)	nu	da (deselectabil)	da (deselectabil)		da (deselectabil)
Recunoaștere fax	nu	da	nu	nu		nu
Program de grafică	nu	da	nu	nu	da	da
Editor de texte		da			nu	da
Carte de telefon integrată		da	da		da	da
Transm. e-dată în timp	da	da	da	da	da	da
Circulare	da	da	da	da	da	da
Jurnal	Em	Em/Rec	Em/Rec	Em/Rec		Em/Rec
Polling	nu	da	nu	nu	nu	nu
Recepție în background	da	da	da	da		da
Pr. princ. în background	da	da	da	da		da
Mouse	nu	nu	nu	nu		nu
Suport hard-scanner	nu	Microtek, Umax OADC	nu	nu	da	Canon I-12, HP, Logitech, Panaso
Suport hard - imprimante	HP-Laserjet, ESC-P compatibile	HP-Laserjet II Brother M1724, NEC 220, Ink-Jet BJ80 Epson 8/9/24 Pin	HP-Laserjet, ESC/P compatibile	HP-Laserjet, ESC/P compatibile	ESC-P compatibile	HP-Laserjet, ESC-P compatibile
Informații suplimentare		A&M Computertechnik GmbH, Ötztaler Str. 18, 8000 München 70 tel. (089)743170, fx (089)74317100	FSE Computerhandels GmbH, Schmiedstr. 11, 6750 Kaiserslautern tel. (0631)67096 fx(0631)60697	ARC Computer- vertrieb GmbH, Freisinger Str. 3, 8057 Eching, tel. (08165)62899 fx(08165)64188	Interquad, 6078 Neulsenburg	P+W Electronic GmbH, Lortzingstr. 52, 7967 Bad Waldsee, tel. (07524)4508, fx(07524)4419

	Ferrari Fax	Dr. Neuhaus Faxy Master	ARC PCFax 9600 II	AEG Twinfax	PC Fax 2000	DNS Gammfax CP
Preț (DM)	1704	1708	1980	3021	3400	3751
Cerințe sistem						
Memoria principală (koct)	640	450		384		512
Slot	8/16 biți	8 biți	8	8 biți		8 biți
Versiune MS-DOS	3.1			3.0		3.3
Date tehnice						
Utilizabil ca modem	da	da	da	da		
Cablu telefonic	3m / TAE-8	5,15m / TAE-8		6,2m / TAE-8		2,15m / RJ11
Funcții soft						
Aplicație DOS	limitat	da		da		
Aplicație Windows	da	nu		nu		
Memorie ocupată ca TSR	29 ko	103 ko		60	40 / 80 ko	
Preimprimare	da	da	da	da		nu
Transmisie de ecrane	nu	nu	nu	da		nu
Snapshot	nu	nu	nu	da	da	nu
Intercepția imprimării	da	da	nu	da	da	da
Preconversie	nu	da	da	da		da
Conversie automată	nu	da (opțional)	nu	da		da (ASCII)
Conversie în background	nu	da	nu	nu		da (ASCII)
Format -> faxformat	ASCII, PCX, DCX	ASCII, CUT, PCX, IMG	ASCII, PCX, IMG, MSP	ASCII, PRN	PCX	ASCII, PCX, TIF, IMG, RST
faxformat -> Format	PCX, DCX	CUT, IMG, PCX	MSP, PCX	TIF, PCX, PCL, IMG	PCX	ASCII, PCX, TIF, IMG, RST
Viteza de transmisie	2400, 4800, 7200, 9600 bps	2400, 4800, 7200, 9600,	2400, 4800, 7200, 9600 bps	2400, 4800, 7200, 9600 bps	2400, 4800, 7200, 9600 bps	2400, 4800 bps
Moduri de transmisie	normal / fin	normal/fin	normal	normal/fin/mic		normal
Nivele de gri	nu	nu	nu	nu		nu
Comprimare pagină	da (deselectabil)	da (deselectabil)	da (deselectabil)	da (deselectabil)		da (deselectabil)
Antet	da	da	da	da		da
Pagină-copertă	da (deselectabil)	nu	da (deselectabil)	da (deselectabil)		nu
Recunoaștere fax	nu	nu	nu	nu		nu
Program de grafică	nu	nu	da	da		nu
Editor de texte	da	da		da		
Carte de telefon integrată	da	da	da	da		
Transm. c-dată în timp	da	da	da	da	da	da
Circulare	da	da	da	da	da	da
Jurnal	Em/Rec	Em/Rec	Em/Rec	Em/Rec		Em/Rec
Polling	da	nu	da	da	nu	da
Recepție în background	nu	da	da	da	da	da
Pr. princ. în background	nu	nu	da	da	da	da
Mouse	opțional	nu	opțional	opțional		nu
Suport hard-scanner	via Windows 3.0	Faxy SP, HP, Microtek, Panasonic	Canon, Chinon MS, Tele Scan	telefax autonom	da	Canon IX-12, HP
Suport hard - imprimante	via Windows 3.0	Faxy SP, Fuji, Olivetti, Laserstar, HP-Laserjet IIP + III, Epson FX, LQ, Juki	HP-Laserjet II, IBM Graphics Pr., Canon BJ-80,130, Epson 9/24 pin, Brother M-1724	ESC/P compatibil, IBM Graphics Printer	Epson	HP-Laserjet+, IBM Graphics Pr., IBM Proprinter, Xerox 4045, Canon, Epson, NEC

Precizări relative la tabelul comparativ

- Preț** - prețurile sînt prețuri de vînzare recomandate; cu reduceri, este posibil să puteți obține echipamentele și mai ieftin. În preț este inclusă obligatoria taxă de valoare adăugată, fixată prin lege (în Germania) la 14%. (Cumpărătorilor ce duc peste graniță bunurile cumpărate, li se returnează contravaloarea acestei taxe, la prezentarea dovezii corespunzătoare de la vamă).
- Memoria principală** - dimensiunea minimă a memoriei de bază necesară softului anexă pentru o funcționare ireproșabilă. Datele sînt cele declarate de producători.
- Slot** - se referă la tipul slot-ului liber în care urmează să se plaseze cartela fax la instalarea în PC
- Versiune MS-DOS** - cea mai mică versiune DOS cu care softul aferent cartelei DOS poate lucra. Mai ales laptop-urile și notebook-PC-urile se livrează cu versiuni MS-DOS foarte vechi (dar foarte economice în ce privește consumul de memorie). Atenție, marea majoritate a programelor fax necesită versiuni ≥ 3.0 .
- Utilizabil ca modem** - uneori, cartela fax este de fapt o cartelă fax-modem, putînd fi folosită și ca modem (de regulă, cu restricții, mai mult sau mai puțin importante).
- Cablu telefonic** - dimensiunea cablului anexat la livrare, împreună cu tipul ștecherului (TAE-8 - standard german, RT11 - standard american; la noi, încă nu există nici un fel de standardizare în această direcție; la instalare se fixează legăturile necesare cu șuruburi ...)
- Aplicație DOS/Windows** - precizează dacă softul aferent cartelei funcționează ca aplicație DOS, sub Windows, sau și-și.
- Memorie ocupată ca TSR** - în cazurile cînd programul poate lucra ca program rezident, în background, specifică memoria "furată" de către acesta.
- Preimprimare** - o funcție utilă a unor programe fax este inserarea opțională a unor antete de scrisoare sau logo-uri de firmă, astfel încît faxul ajuns la destinatar să arate ca o scrisoare "obișnuită".
- Transmisie de ecrane (screenshot)** - în softurile fax rezidente, există uneori posibilitatea ca o simplă combinație de taste să aibă ca efect transmisia ca fax a ecranului curent.
- Snapshot** - este vorba de posibilitatea de a stoca, într-un fișier, ecranul curent, pentru eventuale prelucrări ulterioare.
- Intercepția imprimării (printcapture)** - cu această funcție, orice program de prelucrare texte poate fi folosit pentru pregătirea de faxuri. Un mic utilitar rezident deturneză datele trimise spre imprimantă, orientîndu-le către cartela fax.
- Preconversie** - este vorba de conversia prealabilă a fișierelor ce se vor transmite, înainte de stabilirea legăturii - dacă un program nu are această funcție, va trebui să acceptați eventuali timpi de așteptare inutili, pe timpul transmisiei.
- Conversie automată** - la unele programe, înainte de transmisie, datele trebuie convertite "manual". Este desigur mai comod dacă această conversie se face automat, fără intervenția utilizatorului.
- Conversie în background** - unele programe convertesc fișiere ASCII în mod automat înaintea transmisiei, fără a bloca calculatorul pe durata acestei conversii.
- Format** - > **faxformat** - precizează care sînt formatele grafice recunoscute prin soft. Dacă vreți să transmiteți și grafică, nu numai texte, atunci softul aferent cartelei fax trebuie să poată prelucra fără probleme formatul grafic în care i se prezintă datele. (Altfel, este necesară imprimarea prealabilă și scannarea imaginii ce urmează a fi transmisă). Softuri performante de fax ar trebui să "știe" cît mai multe formate grafice.
- faxformat** -> **Format** - precizează care sînt formatele grafice în care poate fi convertit un fax recepționat
- Viteza de transmisie** - 9600 bps sînt aproape standard în domeniul faxurilor actuale. Pentru legături telefonice foarte proaste, ar trebui însă să existe și posibilitatea trecerii la viteze de transmisie mai mici. De reținut diferența între bps (biți pe secundă) și Baud. Ultima unitate de măsură determină numai numărul de schimbări de frecvență (pe secundă) pe linia telefonică. Începînd cu 600bps, la o singură schimbare de frecvență se transmit mai mulți biți.
- Moduri de transmisie** - cu cît e mai mare rezoluția unei pagini transmise prin fax, cu atît calitatea ei este mai bună. Dar rezoluții mărite necesită și timpi de transmisie corespunzător mai mari. Pentru texte obișnuite, rezoluția normală e suficientă.
- Nivele de gri** - cartelele fax cele mai noi prelucrează graficele nu numai ca și combinații de puncte albe și negre, ci pot transmite și diferite nuanțe de gri.
- Comprimare pagină** - cartelele fax din grupa III oferă un procedeu de compresie care permite reducerea timpilor de transmisie.
- Antet** - începînd cu norma fax pentru grupa III, este obligatoriu pe documentul transmis un rînd în care se specifică data, numărul de apel și numele expeditorului.
- Pagină - copertă** - la transmisia mai multor pagini, prima pagină ar trebui să conțină date relative la numărul total de pagini precum și date exacte privind expeditorul și destinatarul. Desigur, este avantajos ca această funcție să fie integrată în programul fax aferent. Totodată, la cerere, ea ar trebui să poată fi invalidată.
- Recunoaștere fax** - unele cartele fax "ridică receptorul" la sosirea unui apel telefonic, așteptînd un semnal de la chemător. Dacă acest semnal nu vine într-un interval scurt, echipamentul anunță apelul

utilizatorului, eliberînd linia telefonică pentru o convorbire normală.

Program de grafică (integrat) - cu aceste editoare grafice minimale integrate, utilizatorului i se dă posibilitatea, de ex., să adnoteze paginile fax recepționate cu scurte notițe.

Editor de texte (integrat) - permite editarea minimală a textului ce urmează a fi transmis, direct, fără intermediul unui procesor de texte "standard".

Carte de telefon integrată - multe programe fax "vin" împreună cu cartea de telefon a abonaților serviciului telefax, permițînd astfel găsirea rapidă și comodă a numărului dorit.

Transmisie comandată în timp - poate fi avantajos uneori ca transmisia unui document să nu se facă imediat, ci la o anumită oră (sau dată) - pentru a beneficia de exemplu, de tarifele reduse din timpul nopții.

Circulare - cînd un fax trebuie să ajungă la mai mulți participanți, în programele fax bune este suficientă elaborarea listei destinatarilor - restul acțiunilor le rezolvă softul singur, fără intervenția utilizatorului.

Jurnal - mai ales la cartelele ce se utilizează în domeniul comercial, jurnalul (evidența) protocolului de utilizare (emisii-recepții) este un auxiliar indispensabil pentru facturare.

Polling - această funcție, utilizată rareori, permite preluarea unor faxuri disponibile de la un alt abonat. Echivalent (oarecum) al serviciilor gen "post-restant" sau "cutie poștală".

Recepție în background - cele mai multe programe fax lucrează și în background, dar sînt puține cele ce permit continuarea lucrului cu calculatorul și pe durata recepției (cel mai adesea, utilizatorul e pus în așteptare - nu i se solicită însă nici o intervenție, recepția se desfășoară "pe nevăzute").

Programul principal lucrează și în background - dacă cartela fax trebuie să fie gata de recepție 24 de ore din 24, atunci programul fax trebuie să poată lucra rezident, în background.

Mouse - precizează dacă programul fax poate sau nu să fie utilizat folosind un mouse.

Support hard/scanner - pentru transmisia de pagini scrise de mînă sau a unor desene, este necesară scannarea anterioară a acestora, folosind un scanner. Uneori, ca scanner poate fi utilizat chiar un telecopiator "adevărat" (autonom).

Support hard/imprimantă - bineînțeles, trebuie să puteți "scoate la imprimantă" paginile recepționate prin fax. Dacă documentul nu este sec, ci are și informații de formatare (sublinieri, îngroșări, diferite tipuri de caractere, etc.) atunci acestea trebuie transmise imprimantei în așa fel încît să imprime textul într-o formă corectă - evident, cel care vă trimite faxul nu prea poate ști ce fel de imprimantă aveți în sistem. Din păcate, datele furnizate de producători la acest punct sînt, uneori, foarte superficiale.

Grupele fax

Telecopiatoarele (telefaxurile - prin extensie, prin telefax sau fax se înțelege atît echipamentul care realizează transmisia, cît și documentul transmis) se categorisesc în 4 grupe. Aceste grupe se deosebesc prin caracteristici de bază; pentru utilizatori, diferențele "se văd" în viteza și calitatea transmisiei.

Telecopiatoare din grupa I

Această grupă este caracterizată de o tehnică depășită de transmisie și este neinteresantă pentru noi - în țară, se pare, nu sînt instalate astfel de telefaxuri.

Telecopiatoare din grupa II

Și în acest caz este vorba de o tehnică foarte veche. Informațiile sînt prelucrate analogic, ca și în cazul grupei I, ceea ce face ca echipamentele fax din grupa II să mai existe foarte puține. Viteza lor de lucru redusă (cca. 3 minute pentru o pagină A4) precum și insuficiența corecției a erorilor le fac neinteresante pentru aplicațiile actuale.

Telecopiatoare din grupa III

sînt cele mai răspîndite în prezent și lucrează cu tehnici numerice. Deoarece se face o compresie a datelor, timpul de transmisie variază funcție de conținutul documentului. Compresia datelor se face fie unidimensional (orizontal, pentru fiecare rînd scannat) fie bidimensional (orizontal și vertical) și are rolul de a minimiza timpul necesar transmisiei. Astfel, o pagină A4 cu un conținut obișnuit poate fi transmisă pe linia telefonică în circa 1 minut. Viteza fizică pentru faxurile din grupa III este stabilită de regulă de norme CCITT V.27 - deci 2400 și 4800 biți pe secundă. Telecopiatoarele mai noi se conformează și normelor CCITT V.29, pentru 7200 și 9600 bps. Rezoluția normală pentru echipamentele din această grupă este de 203 x 98 dpi (dots per inch - puncte pe inch). Deseori, rezoluția pe verticală poate fi mărită pînă la 196 dpi, iar pe alocuri există deja echipamente ce lucrează cu 400 x 400 dpi.

Telecopiatoare din grupa IV

Aceste echipamente sînt construite pentru legături digi ale, lipsite de erori de transmisie, cum sînt asigurate de exemplu în ISDN. Datele de transmisie sînt comprimate la maxim și transmise apoi receptorului. Echipamentele din această categorie vor deveni interesante de-abia cînd ISDN va fi disponibil pe scară largă, în toată lumea - lucru improbabil în viitorul apropiat.

Fonturi

Luați un borcan de marmeladă, de pe un raft cu bunuri de consum, dintr-un magazin ecologic occidental, și examinați-i eticheta: un scris de copil ușor nesigur - ca de petabla unei școli. Eticheta n-a fost în nici un caz scrisă de mână, ci a fost tipărită. De ce s-a folosit atunci atunci acest scris de tablă de școală? Este foarte simplu: acest caracter de literă semnalizează: "Pur natural - în întreprinderea noastră totul este lucrat de mână, deci și scris de mână". Deoarece cel mai adesea produsele nu pot fi gustate înainte de a fi cumpărate, inscripționarea de pe ambalaj trebuie nu numai să-i lase clientului "gura apă" ci și să-i câștige încrederea.

Același lucru se întâmplă și în prospectul unui aparat CD (Compact Disk) produs de o anumită întreprindere High-Tech - caracterul de literă (fontul) utilizat în acest caz: Helvetica, fără serife (deci fără linii horizontale sau cîrlige la sfîrșitul caracterelor). Eleganța simplă, lipsită de podoabe semnalizează precizia cultivată și competența tehnică.

Două exemple din care se poate observa că se folosește un caracter de literă pentru a transmite un mesaj; caracteristicile estetice ale unui caracter de literă nu reprezintă neapărat un scop artistic în sine, ci sînt folosite în anumite scopuri. Corespunzător cu aceasta există un număr aproape de necuprins de caractere de literă și de familii de caractere de literă diferite. Toate, însă, nu doresc decît un singur lucru: să-l ia pe cititor de mînă și să-l conducă prin text (sau

să-l rătăcească, în funcție de morală).

Doriți să realizați o invitație elegantă, o listă de bucate împoșonată sau o broșură cu calculatorul? pentru ca ea să arate într-adevăr bine, aveți nevoie de o colecție corespunzătoare de familii de litere și de diferite dimensiuni de literă. Procesorul D-vs. de texte vă oferă suficiente alternative? Atunci desigur că n-ați avut nefericita experiență, ca de exemplu textul unui antet să nu fie tipărit cu același caracter de literă ca și textul cursiv - chiar dacă imprimanta folosită ar fi putut să facă acest lucru.

Astfel, de exemplu, procesorul de texte "Word 5.0", chiar lucrînd cu imprimanta laser HP Laserjet, nu curățește pentru caracterul de literă Courier decît dimensiunile 10 și 12 puncte (punct: înălțimea unui caracter. 1 punct = 0,335 mm). Dacă pentru text se utilizează dimensiunea 12 puncte, și se dorește o scriere mai marcantă lipsesc dimensiunile de scriere mai mari (de exemplu de 18 sau 24 de puncte).

Problemele nu apar numai la titluri: dacă doriți să imprimați o notă de subsol, cu caracterul de literă Courier, atunci dimensiunea de 10 puncte nu este suficient de mică.

Dar nu trebuie să renunțați numai decît, dacă trebuie să scrieți un referat sau o disertație cu Word: există mijloace și căi de a utiliza și alte caractere de literă și dimensiuni, și aceasta nu numai pentru Word, ci pentru majoritatea aplicațiilor. Cum puteți ajunge la noi

caractere de literă pentru calculatorul D-vs., cît costă și cum le puteți utiliza optimal, veți putea afla din paginile care urmează. În plus vor fi reprezentate fundamentele temei familii de caractere de literă și multe sfaturi folositoare.

Utilizarea corectă a scrierilor

Cine dorește să utilizeze efectul optic al diferitelor familii de caractere, trebuie să știe să le folosească. Utilizarea unei provizii bogate de familii de caractere, așa cum oferă calculatoarele cu ajutorul programelor DTP (DTP = Desk Top Publishing - programe pentru tehnoredactare), în lipsa unei know-how corespunzător, conduce cel mai adesea doar la rezultate de diletanți.

Una din regulile de bază spune: "Să nu utilizezi niciodată prea multe caractere de literă pe aceeași pagină". Hotărîtoare este alegerea unui caracter de literă și-abia apoi - variațiile ei (mare/mică, îngroșată, cursiv, etc.). Problema este să se constituie mai întîi o colecție cît mai mare de caractere de literă din care să se poată opera selecția. Dar să vedem mai întîi ce probleme apar la prelucrarea scrierilor pe calculator.

Scrierile și calculatorul

Pentru ca utilizatorul unui calculator să poată utiliza un anumit tip de caracter de literă, procesorul de texte respectiv trebuie să cunoască scrierea respectivă. Mai

ABCDE ABCDEF ABCDEF ABCDEFGH
 abcdefgh abcdefg abcdefgh abcdefghijkl abcdefghijkl

Times

Text de mașină de scris

Helvetica

Scris de mînă

Fraktur

întîi trebuie pregătit, aşadar, caracterul de literă dorit pentru a putea fi utilizat din procesorul de texte dorit şi pentru a putea fi tipărit la imprimantă.

Pentru a putea reprezenta pe ecran un caracter, sau pentru a-l putea trimite la imprimantă, din interiorul unui procesor de texte, caracterele trebuie transformate după o schemă "înţeleasă" de calculator. Un caracter devine inteligibil pentru un PC, respectiv pentru un soft, de îndată ce a fost transformat într-un mănunchi de date, deci după ce a fost codificat în biţi şi bytes. În acest mănunchi de date este cuprinsă alcătuirea caracterului dintr-o anumită ordonare a unor puncte individuale. Aceste puncte sînt plasate într-un domeniu bine definit (de exemplu un bloc de 28 de puncte pe orizontală şi 36 de puncte pe verticală), acesta constituind așa-numita "matrice caracter".

Pentru fiecare caracter tastat, la tastatura PC-ului, programul determină calculatorul să reprezinte respectivul caracter într-o anumită poziţie de pe ecran într-o astfel de matrice-bloc. Mai multe astfel de matrice-bloc una după alta compun un şir de caractere, un cuvînt, sau şi mai mult un întreg text.

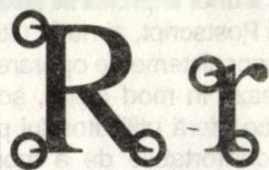
Un procesor de texte trebuie să poată accede la un set complet de caractere, pentru a putea construi pe baza acestor "metode" caracterul tastat, pentru a crea în acest mod un text şi al stoca într-o formă reutilizabilă pe un harddisk sau pe o dischetă.

Care sînt scrierile utilizabile pe calculator?

Un caracter de literă care poate fi reprezentat foarte uşor într-o matrice pentru calculator este Courier, creat de IBM. Acest caracter de literă a fost conceput în mod special pentru reprezentarea pe monitoare, la sfîrşitul anilor 50. Rezoluţia disponibilă în acel moment

era de 320x200 puncte - corespundea deci standardului CGA. Din cauza acestor limitări, în cazul caracterului de literă Courier, s-a renunţat la multe caracteristici, cum ar fi grosimi diferite de linii şi rotunjimi la colţuri, care sînt tipice pentru multe alte familii de caractere.

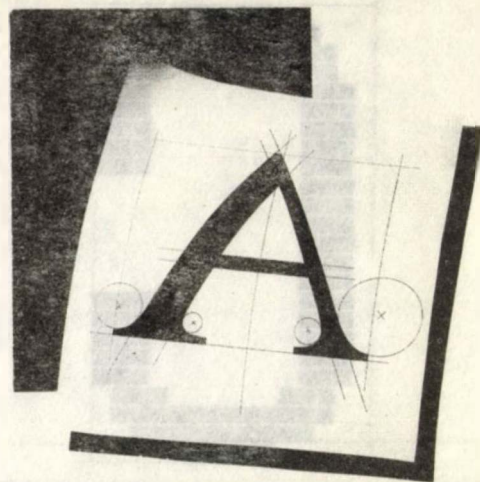
Majoritatea celorlalte scrieri au fost elaborate manual înainte de apariţia calculatoarelor, fără să ţină cont de limitările impuse de acestea. Cîteva dintre acestea sînt scrieri decorative şi ornate. Ele dispun - ca şi toate celelalte familii de caractere create de-a lungul timpului cu creionul şi pana - de o mulţime de caracteristici de prezentare. Ele dispun de rapoarte de mărime diferite în partea superioară, de mijloc şi de jos; multe caractere avînd şi anumite serife (linii orizontale sau cîrlige la părţile terminale ale literei). La acestea se adaugă diferite grosimi de scriere în interiorul unui caracter. Toate acestea îngreunează transcrierea în biţi şi bytes - şi doar așa pot fi prelucrate de calculator.



Serife

Scanarea unei scrieri nu este practicabilă

Cu toate că o scriere poate fi "citită" cu ajutorul unui scanner, codificată sub forma unui rastru de puncte, transformată în biţi şi prezentată pe monitor, de exemplu cu ajutorul unui program de pictură (cu ajutorul căroră s-ar putea elabora chiar caractere de literă proprii), aceste date nu pot fi prelucrate în continuare ca nişte caractere obișnuite de către un procesor de texte. Pentru aceasta



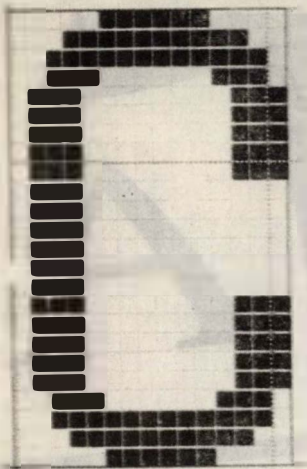
ar trebui soft-uri suplimentare - ceea ce este prea puţin practică.

Programele de recunoaştere optică a caracterelor create pentru scanner-e pot citi caractere, atît timp cînd sînt elaborate relativ simplu (altfel pot apare erori de identificare), programele moderne putînd fi chiar "învăţate" să recunoască şi scrieri decorative. Dar scrierile recunoscute nu sînt redactate în aceeaşi formă în care au fost citite, ci este utilizată o formă standard pentru reprezentarea lor pe monitor sau pentru tipărirea lor la imprimantă. Această soluţie poate fi utilizată la introducerea unor volume mari de texte, dar nu şi pentru îmbogăţirea repertoriului de scrieri.

Două căi pentru producerea de scrieri utilizabile pe calculator:

Fonturi - bitmap

Între timp s-au stabilit două tehnici de bază. Una funcţionează în mod asemănător, ca în primul nostru exemplu, caracterul de literă Courier: fiecare caracter este construit dintr-o succesiune de puncte (pixeli) poziţionate pe un rastru predefinit. Caracterele sînt reprezentate ca "hărţi de biţi" (bitmaps) orientate pe pixeli - cea mai simplă metodă de a crea noi scrieri (numite şi fonturi) pentru calculator - de aici şi denumirea de fonturi bitmaps.



Font bitmap: construcție din puncte

Această metodă este recomandabilă pentru caractere "pline" (caractere care nu sînt alcătuite doar din linii). Trebuie luată în considerare, însă, și existența cîtorva dezavantaje, cum ar fi apariția așa-numitului efect treaptă, sau dinți de fierăstrău, pe conturul unor astfel de caractere. Acest dezavantaj poate fi contracarat prin utilizarea unui rastru mai fin, dar fiecare pixel care intră în plus în componența unui caracter, necesită spațiu suplimentar de memorie - ceea ce implică utilizarea unui hard mai pretențios, fapt care duce la creșterea costurilor și se lovește de limitele tehnicii.

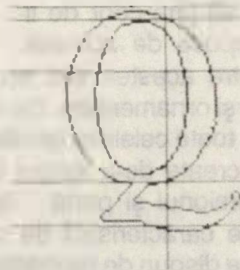
Fonturi outline

Efectul dinților de fierăstrău, obținut la mărirea unui caracter bitmap, poate fi înlăturat prin utilizarea tehnicilor de prezentare vectorizată a caracterelor.

Această tehnică presupune un efort suplimentar de programare și de calcul în comparație cu tehnica bitmap, și presupune adesea și existența unui hard mai puternic. În cazul caracterelor de literă vectoriale nu mai trebuie definite toate punctele imaginii, ci doar punctele de început și de sfîrșit ale unei linii drepte. Pentru o formă tridimensională sînt necesare trei coordonate, pentru un patrulater patru, un cerc fiind aproximat printr-un număr cît

mai mare de segmente de dreaptă. Cu cît sînt utilizate mai multe puncte pentru descrierea unei serife sau a unei rotunjiri, cu atît mai exact va putea fi reprezentat caracterul pe ecran sau pe hîrtie.

Un exemplu îl constituie așa-numitele fonturi outline. Aceasta înseamnă că aceste caractere sînt descrise numai prin liniile exterioare care urmăresc contururile.



Font outline: Caracterele sînt descrise prin contur

Abia cu dezvoltarea unui hard puternic și cu apariția unor calculatoare corespunzătoare (Apple Macintosh sau calculatoare personale cu microprocesoare Intel 80386), a unor sisteme grafice puternice, a unor imprimante laser cu facilități Postscript, și nu în ultimul rînd a unor sisteme de operare care lucrează în mod grafic, softuri puternice oferă utilizatorului posibilități confortabile de a modela scrieri.

Pentru ca imprimantele să poată tipări fonturile este necesar un software de comandă special: driverul de imprimantă (situație valabilă și pentru fonturile standard).

Fonturi la pachet

Pentru a dispune de mai multe caractere de literă în programele utilizate se pot cumpăra pachete de fonturi. Astfel de pachete oferă, de exemplu, firmele Adobe și Bitstream. Pachetele lor de fonturi au fost elaborate în mod special pentru a fi utilizate în programele DTP, cum ar fi: "Ventura Publisher",

"Pagemaker" sau "Quark Express". Avantajul: utilizatorul poate încărca noile fonturi direct în programul dorit, fără alt efort de instalare. Dezavantaj: un astfel de pachet este limitat la cîteva caractere de literă și dimensiuni caracter. Aceste produse presupun doar existența programelor DTP, și nu necesită nici o suprafață utilizator specială - este suficient sistemul de operare MS-DOS.

Suprafețe utilizator grafice

În învechitele moduri caracter (mod text, nu grafic) ale primelor generații de PC-uri, rezoluția monitorelor și a imprimantelor nu era suficientă pentru a putea oferi utilizatorului posibilitatea de a modela fonturi.

Datorită suprafețelor utilizator grafice (Macintosh, pentru PC "Windows" sau "Geoworks Ensemble") utilizatorul vede caracterele pe ecran la fel cum vor arăta la imprimantă. Aceasta înseamnă că noile suprafețe utilizator permit așa - numita reprezentare WYSIWYG (What You See Is What You Get): ceea ce se vede pe ecran corespunde cu ceea ce se va vedea la imprimantă. Abia aceste medii de programare oferă utilizatorului posibilitatea folosirii unui număr mai mare de fonturi și nu numai ca fonturi bitmap ci și ca fonturi vectoriale.

O soluție convenabilă: programele Shareware

În multe cazuri, o soluție convenabilă o oferă programele din sectoarele Public Domain (PD) și Shareware. Paleta de funcții a acestor programe se limitează, adesea, la funcțiile de bază. De exemplu există programe utilitare care permit utilizatorilor procesorul de texte Word, utilizarea alfabetului chirilic, care nu apare ca font în repertoriul programului, și textele create pot fi listate chiar și pe imprimante matriciale Epson.

Mai multe caractere de literă pentru textele D-voastră:

Generatoarele de fonturi

Cu ajutorul generatoarelor de fonturi, utilizatorul își poate crea caractere de literă și dimensiuni de literă în limite largi.

Bitstream - Fontware

Bitstream este o firmă cu o tradiție îndelungată în proiectarea și realizarea de caractere de literă. "Bitstream Fontware" este un pachet combinat, alcătuit dintr-un pachet de scrieri și un generator de fonturi. Programul de instalare furnizat copiază toate fișierele necesare pe harddisk. Din scrierile disponibile pe dischetă pot fi elaborate caracterele de literă la dimensiunile dorite specifice imprimantei și modului grafic în care lucrează calculatorul. Generarea propriu-zisă, atunci când selecția de scrieri și de dimensiuni este mare, poate să dureze câteva ore.

Programul permite o operare foarte ușoară. Pe monitor sînt afișate două ferestre de meniuri, în fereastra din stînga sînt afișate fonturile, iar în cea din dreapta dimensiunile de literă posibile.

Toți pașii de lucru (selecție și generare) pot fi parcurși prin intermediul opțiunilor situate sub cele două ferestre. Programul este livrat ca utilitar și în furnitura programului DTP "Pagemaker 3.0"

Bitstream - Fontware
Ofertant: EDTZ,
8012 Ottobrunn bei München
Preț: cca. 700 DM
Sistem de operare: MS-DOS

Typografica

Firma Schneider Data din Freising oferă generatorul de fonturi numit "Typografica". Typografica este vîndut ca produs individual,

sau, în unele situații, împreună cu alte programe de aplicație (de exemplu "Timeworks Publisher").

După instalare, generatorul de fonturi este adaptat imprimantei și modului grafic utilizat. Abia după definirea mediului de lucru hard și soft, pot fi selectate, din ferestrele de lucru, fonturile dorite. Spre deosebire de Bitstream - Fontware, Typografica nu poate prelucra la un moment dat decît un singur caracter de literă. După selectarea acestuia urmează specificarea stilului de literă și a dimensiunii. Informațiile și opțiunile suplimentare au fost astfel concepute încît aproape că se poate lucra fără manual de utilizare.

Typografica
Ofertant: Schneider Data
8050 Freising
Preț: cca. 130 DM
Sistem de operare: MS-DOS

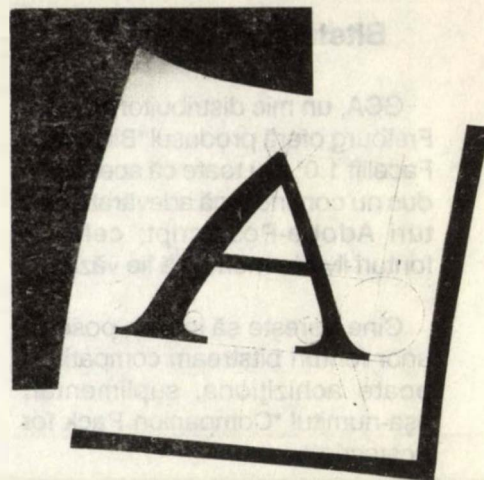
Fontware

Cu Windows 3.0 s-a scris un nou capitol din romanul foileton al suprafețelor utilizator grafice pentru PC-urile MS-DOS. Windows oferă soluția cvasi-ideală pentru astfel de programe "fontware".

Cu ajutorul programelor fontware, fiecărei aplicații Windows îi sînt puse la dispoziție fonturi liber scalabile, care vor arăta la fel și în reprezentarea de pe ecran și la imprimantă.

Adobe Type Manager 1.0

"Adobe Type Manager" (ATM) a fost croit special pentru facilitățile suprafeței utilizator grafice Windows 3.0. Furnitura de livrare conține 13 fonturi diferite din fami-



liile Courier, Helvetica, Times și Symbol. Alte 22 de fonturi sînt oferite în pachetul suplimentar "Adobe Plus Pack". În urma instalării de pe cele 3 dischete cu cele 35 de fonturi și soft-ul de comandă corespunzător (constînd din așa-numitele Control Panel și un driver Windows) se ocupă pe harddisk, 1,5 MByte.

În cursul procesului de instalare driverul este adăugat la fișierul "SYSTEM.INI". Lucrul cu ATM este foarte simplu: la nevoie se poate deschide un mic cîmp de control de sub Windows, pentru a șterge, de exemplu, fonturi. Altfel ATM lucrează neobservat în fundal.

Toate aplicațiile Windows profită de ATM, și mai ales aplicațiile DTP cum sînt "Page Maker" sau "Ventura Publisher 3.0". Caracterele de literă sînt reprezentate corect pe ecran și se caracterizează printr-o calitate ridicată la tipărire. Mai ales lucrul cu imprimante laser compatibile HP funcționează excelent, rezultatul apropiindu-se de calitatea Postscript. Suplimentar fonturilor oferite de ATM pot fi încărcate toate fonturile Adobe-Type-1.

Fontware
Ofertant: Letraset, 6000 Frankfurt
Preț: pachetul de bază cca. 280 DM, cu pachetul suplimentar optimal: 599 DM
Sistem de operare: MS-DOS

Bitstream Facelift 1.0

GCA, un mic distribuitor soft din Freiburg oferă produsul "Bitstream Facelift 1.0". Cu toate că acest produs nu conține încă adevărate fonturi Adobe-Postscript, cele 13 fonturi livrate merită să fie văzute.

Cine dorește să intre în posesia unor fonturi bitstream compatibile poate achiziționa, suplimentar, așa-numitul "Companion Pack for Postscript".

Programul este livrat pe 4 dischete și după instalare ocupă 2,7 Mbyte pe harddisk. Cele două drivere ale programului sînt adăugate la "WIN.INI" și "SYSTEM.INI" ca așa-numite PS-Outline (PS semnifică Postscript) pentru 13 fonturi ne-Postscript.

De sub Windows, Facelift este controlat prin intermediul unei ferestre proprii. Cu ajutorul diferitelor icon-uri din fereastra activată se stabilesc fonturile care urmează a fi disponibile.

Dacă sub Windows sînt instalate mai multe imprimante, se poate stabili care dintre ele să lucreze cu Facelift și care nu. Programul lucrează în fundal. O imprimare cu Facelift durează ceva mai mult decît în cazul calitatății fonturilor este deosebită.

Bitstream Facelift

Ofertant: GCA, Freiburg

Preț: cca. 300 DM (pachetul de bază), cca. 400 DM cu Companion Pack.

Sistem de operare: MS-DOS

Programe de fonturi la nivel DOS

Outline

"Outline" este un program care lucrează la nivel DOS. Programul oferă cîteva particularități: el este asemănător ca principiu cu un cartridge, deci cu o extensie hard

pentru o imprimantă laser. Outline nu este totuși decît un produs pur soft care utilizează conceptul fonturilor soft al imprimantelor laser compatibile HP, Kyocera, Brother sau Sharp. Extensiile pot fi adăugate pentru o serie întregă de texte cunoscute (Wordperfect, Word, PC-Text, etc.).

La instalare, utilizatorul stabilește fonturile și dimensiunile care vor fi utilizate în diferitele procesoare de texte. Dimensiunile de literă nu mai pot fi modificate ulterior. Fonturile și dimensiunile selectate ca imprimanta să poată interpreta corect instrucțiunile de reprezentare, programul generează simultan și un driver de imprimantă care este încărcat automat înainte de imprimare.



Outline: fontware la nivel DOS

Outline oferă pentru diferitele fonturi o serie întregă de efecte speciale: scrieri cu umbre, litere umbrite și chiar scrieri 3D. Datorită faptului că se utilizează o metodă de compresie, fișierele de fonturi Outline necesită cu cca. 60% mai puțin spațiu pe disc decît fonturile compatibile HP.

Fonturile sînt liber scalabile între 4 și 144 de puncte și stau la dispoziție ad-hoc, în formatul dorit, pe imprimantele Laserjet III.

Outline

Ofertant: S.A.X., 7500 Karlsruhe

Preț: cca. 1.100 DM

Sistem de operare: MS-DOS.

Laser Star

"Laser Star" produs al firmei Star Division din Luneburg face parte de regulă din furnitura de livrare a imprimantelor din seria Brother HL-8 dar este oferit și separat.

Programul poate încărcă în imprimanta laser pînă la 50 fonturi soft diferite. Metoda este uluitoare de simplă, în loc să genereze mai întîi fonturile soft, programul generează un driver imprimantă care recunoaște cele 50 de fonturi. Utilizatorul trebuie să specifice mai întîi doar aplicația în care dorește să utilizeze aceste fonturi, programul generînd driverul să utilizeze aceste fonturi, programul generînd driverul necesar și încărcîndu-l în aplicația dorită.

Cele 50 de fonturi soft sînt copiate pe harddisk și sînt încărcate la nevoie. Cu așa-numita operare "download" aceste fonturi sînt încărcate cît se poate de rapid și transmise imprimantei. Cu toate că Laser Jet se limitează astfel la un singur program de aplicație, pentru acesta sînt disponibile toate variațiile posibile cu cele 50 de fonturi.

Laser Star

Ofertant: Star Division, 2120 Lüneburg

Preț: cca. 500 DM

Sistem de operare: MS-DOS

Mai multe caractere de literă pentru textele D-voastră:

Emulatoarele Postscript

Cine nu dorește să-și cumpere o imprimantă Postscript scumpă, își poate comanda imprimanta cu ajutorul unor soft-uri speciale.

Postscript este un limbaj de programare elaborat pentru comanda aparatelor electronice de ieșire (v. "if" 3/90). Acest limbaj de programare a fost botezat Postscript la înființarea firmei Adobe System Inc. în 1982 și este o dezvoltare a limbajului "Impress".

Cu ajutorul unui interpretor Postscript, textele și graficele pot fi imprimate în orice formă imaginabilă. Modul de lucru: Postscript "descrie" imprimantei modul în care trebuie să arate o pagină întreagă și nu doar o linie de text. O pagină prelucrată cu Postscript poate conține o multitudine de elemente: mai întâi trebuie să fie specificat un sistem de coordonate, în raport cu care sînt posibile tot felul de rotiri, modificări de proporții (translații) și scalări, ale diferitelor fonturi. Toate caracterele obținute vor fi tratate de Postscript ca și grafice individuale. Ulterior, cu ajutorul acestui interpretor, toate obiectele pot fi reproduse la o rezoluție corespunzătoare, aceasta putînd depăși 2400 dpi (dot per inch). Aceste rezoluții ridicate nu pot fi obținute decît pe exponatoare speciale, cum ar fi de exemplu Linotype. Imprimantele laser moderne ating astăzi o rezoluție maximală de 600 pîna la 300 dpi. Și totuși aceste imprimante pot utiliza Postscript. Limbajul poate fi implementat direct în hard-ul imprimantei, caz în care imprimanta convertește comenzile primite de la calculator, înainte de imprimare.

O soluție interesantă este oferită de emulatoarele Postscript - programe care mai întâi convertesc fișierele de tipărit obținute cu procesoare de texte sau cu programe

DTP în fișiere Postscript și abia apoi trimite aceste fișiere la imprimantă. Dezavantajul acestor interpretoare: pentru a putea imprima un fișier trebuie părăsit mai întîi programul de aplicație, lansat în execuție emulatorul și abia apoi se așteaptă pentru imprimare. Dacă listarea trebuie repetată dintr-un motiv oarecare, întreaga procedură trebuie repetată ceea ce implică un consum ridicat de timp. Un alt dezavantaj: aceste interpretoare sînt destul de lente.

Avantajul hotărîtor: aceste emulatoare vă scutesc de o cheltuială semnificativă, în loc să aveți nevoie de o imprimantă Postscript puteți utiliza o imprimantă matriceală. Pentru calculatoarele MS-DOS sînt oferite trei emulatoare interesante (v. și "if" nr. 3/90).

Freedom of Press

Translatorul Postscript al firmei Macrotron se caracterizează printr-o operare simplă. Freedom of Press calculează și imprimă un fișier într-un ritm viu. Se poate întîmpla însă că anumite elemente ale unei pagini să fie jertfite vitezei ridicate. Emulatorul Postscript recunoaște toate seturile de caractere de literă Adobe-Type-1 și Adobe-Type-3. În total programul conține 35 de fonturi diferite. Freedom of Press poate fi utilizat pe orice calculator compatibil IBM, XT sau AT cu o memorie minimă de 640 KByte și cu un harddisk pe care să fie liberi cel puțin 3 MByte.

Ofertant: Macrotron, 8000 München.
Preț: cca. 800 DM.

Recomandat pentru: toate seturile de fonturi Adobe-Type-1 și Adobe-Type-3.
Goscript 3.0 Plus



Din SUA provine "Goscript Plus" ajuns la versiunea 3.0 are nevoie de cel puțin 1 MByte de memorie și de cel puțin 3 MByte pe harddisk.

Acest program este ușor de operat. Din păcate Goscript Plus nu poate trata decît un document la un moment dat. Imprimarea mai multor pagini cere multă răbdare din partea utilizatorului, deoarece interpretorul nu poate lista din interiorul programului de aplicație. Sînt recunoscute toate fonturile din setul Adobe-Type-3.

Ofertant: Michael Binkert Software,
7892 Albruck

Preț: cca. 550 DM.

Recomandat pentru: toate seturile de fonturi Adobe-Type-3.

Ultrascript PC 2.1

Acest translator Postscript foarte interesant este oferit de către producătorul american QMS. Programul dispune de cele mai puternice caracteristici în acest domeniu. Merită amintite mai ales posibilitatea de a imprima direct de sub un program de aplicație și faptul că Ultrascript PC poate prelucra simultan mai multe pagini. Ultrascript PC poate fi încărcat ca program rezident în memorie la pornirea calculatorului. Programul poate verifica, în timpul calculului, dacă are suficientă memorie la dispoziție. Dacă nu fișierul este redirecționat pentru stocare temporară pe harddisk, (spooling). Datele sînt păstrate acolo într-o coadă de așteptare pînă cînd me-

АБВГДЕЁ
абвгдеёжз

س س ص ص
ك ك ك ل ل ل

Scrierea ehirică poate fi obținută și cu programe shareware,
scrierea arabă nu poate fi obținută decât cu procesoare de texte speciale

moria de lucru este din nou liberă și este pregătită pentru a prelua date. Ultrascript PC recunoaște toate fonturile Adobe-Type-3. Programul conține 44 de fonturi diferite.

Cerințe sistem: PC compatibil IBM XT/AT, cu minim 640 KByte memorie de lucru (pentru imprimante laser 1 MByte), sistem de operare MS-DOS 3.1 sau mai recent, 4 MByte liberi pe harddisk.

Ofertant: Michael Binkert Software,
7892 Albruck

Preț: cca. 650 DM

Recomandat pentru: toate fonturile Adobe-Type-3.

Vastprint

Vastprint este un spooler de imprimantă extrem de flexibil. El utilizează multitudinea posibilităților de memorare pe un PC ca și clipboard (depozit intermediar de date). Pe lângă memoria principală a PC-ului poate fi utilizată, la dorință și memoria extinsă sau harddisk-ul. Vastprint poate capta fișierele trimise pentru imprimare la un anumit port de ieșire (de ex. COM1 sau PAR1) și le poate redirecta spre o altă ieșire. Cu ajutorul unei tabele de conversie fiecare caracter dintr-o "provizie ASCII" poate fi convertit în memorie și poate fi apelat la nevoie cu ajutorul unei combinații de taste.

Ofertant: Dawicontrol, 3400 Gottingen
Preț: 140 DM

Sistem de operare: MS-DOS.

Fonturi ecran din categoria Public-Domain

Din domeniul Public-Domain este de remarcat programul share-

ware EGA-Font. Cu toate că acest program permite afișarea caracterelor numai pe ecranul monitorului, și nu la imprimantă, cu ajutorul lui pot fi obținute variante de caractere amuzante. Toate cele 255 de caractere pot fi încărcate într-o mască de prelucrare și pot fi modificate după dorință. Ca urmare se poate obține, de exemplu, un Norton Commander cu caractere baroce sau o suprafață DOS futuristă.

Fonturi pentru Mac

Programe auxiliare: Adobe Type Manager (Apple Macintosh)

Multe programe se împăunează cu prezentarea WYSIWYG (What You See Is You Get), dar puține își respectă promisiunile făcute.

Rareori o scriere este identică ca reprezentare și pe monitor și la imprimantă. Mai ales la caracterele de dimensiuni mari, la rotunjiri pot apare trepte urite.

Adobe Type Manager este un program care lucrează în fundal și care se îngrijește de atenuarea efectului de treaptă la reprezentarea caracterelor pe ecran.

Type Manager este livrat cu 13 fonturi Adobe, de la Times și Helvetica și pînă la Courier și Symbol. Toate aceste fonturi sînt disponibile ca și caractere îngroșate, cursive sau ca și combinații de îngroșat și cursiv.

Adobe Type Manager poate fi utilizat cu toate programele de aplicație uzuale, de la procesoare de texte și pînă la DTP. De asemenea

cu ajutorul acestui pachet se obțin imprimări de foarte bună calitate atît pe Hewlett-Packards Deskwriter, cît și pe Imagewriter și pe Laserwriter.

Adobe Type Manager este un program indicat tuturor utilizatorilor de calculatoare Mac care sînt pretențioși în ceea ce privește tehnoredactarea unor documente.

Ofertant: Adobe, Amsterdam

Preț: 300 DM

Sistem de operare: Apple/Macintosh

Recomandat pentru: toate programele utilizator în care poate fi editat text (procesare de texte, DTP, calcul tabelar, etc.).

TypeStyler și LetraStudio

sînt două programe puternice pentru producerea de titluri mari, antete, titluri și logo-uri. Ele permit lucrul cu culori, rastre și efecte speciale. Fonturile pot fi preluate din bibliotecile uzuale de fonturi, cum ar fi: The Font Company, Casady & Greeve, Bitstream, Double Click sau Adobe.

Scrierile pot fi scalate într-un domeniu larg. Extrasele de text pot fi rotite, întinse, tasate, deformate, oglindite și remodelate. Lățimea caracterelor ca și distanța dintre caractere pot fi modificate de utilizator.

Toate fonturile pot fi variate în continuare, adăugîndu-li-se o culoare, rastru, contururi etc. Scrierile pot fi așezate flexibil pe toate formele posibile. Este posibil, de exemplu, să se îndoie un text în jurul unei cutii sau să fie moldelat pentru gîtul unei sticle. Paleta posibilităților fiind atît de bogată este

destul de greu să ai o privire de ansamblu asupra întregului.

Rezultatul prelucrărilor poate fi memorat în format "EPSF" și "PICT". În timp ce Type Styler este strâns legat de QuarkXPress, Pagemaker și Drawing Table, LetraStudio este dedicat unei întregi serii de produse de tehnoredactare, cum ar fi: FontStudio și DesignStudio pentru prelucrarea la nivel de pagină, ImageStudio pentru prelucrarea de imagini halftone și ColorStudio pentru manipularea de imagini color.

Letraset Studio

Ofertant: Letraset 6000 Frankfurt

Preț: cca. 2200 DM

Sistem de operare: Apple/Macintosh

Recomandat pentru: DesignStudio, ImagineStudio, ColorStudio și alte programe de tehnoredactare.

Type Styler

Ofertant: Prisma, 2000 Hamburg

Preț: cca. 800 DM

Sistem de operare: Apple/Macintosh

Recomandat pentru: QuarkXPress, Pagemaker, Drawing Table

Image Club Typeface Library

Acest program conține o bibliotecă bogată de fonturi pentru Apple/Macintosh. Paleta se întinde de la Helvetica și Times Roman și pînă la caracterele de literă utilizate în germana veche, și anume Fraktur.

"Image Club Typeface Library" este una din multele biblioteci de fonturi care pot fi utilizate de către utilizatorii de Mac. Concurența este constituită din Adobe, Bitstream și alți producători. Typeface Library oferă o selecție uriașă de fonturi, care sînt reproduse într-un catalog de aproape 80 de pagini. Selecția unui font din cele mai bine de 200 de fonturi prezentate este astfel ușurată.

Fonturile oferite au nume sonore și uneori comice, cum ar fi: "Cen-

tury Schoolbook", "ITC Cheltenham" sau "Harry" și "Rock-A-Billy".

Bibliotecile de fonturi ca Typeface Library sînt ideale pentru toți cei care caută caractere de literă speciale pentru anumite scopuri.

Image Club Typeface Library

Ofertant: Kagema, 7000 Stuttgart

Preț: cca. 2000 DM

Sistem de operare: Apple Macintosh

Recomandat pentru: toate procesoarele de texte.

Fontstudio 2.0

Dacă nu se dorește alegerea niciunuia dintre fonturile existente în bibliotecile de fonturi disponibile, atunci se poate utiliza "Fontstudio". Cu ajutorul acestui program utilizatorul își poate crea fonturi proprii, sau le poate remodela pe cele existente. Caracterele pot fi prelucrate fie ca bitmap fie ca outline. Astfel un caracter poate fi creat punct cu punct (bitmap), sau cu ajutorul așa-numitelor curbe Bezier (outlines). Fontstudio permite și cuplarea celor două moduri de lucru. Bucățile de literă care ar putea fi reutilizate ulterior pot fi stocate într-o bibliotecă. Astfel pot fi arhivate serifele uzuale și crochiurile de linii rezultînd un set de "scule" personal. Ca unelte de lucru sînt disponibile creionul colorat, radiera, unghiul drept, ovale și pîlnia de umplere (desigur soft). Literale, sau anumite bucăți din ele, pot fi rotite, scalate, oglindite sau înclinate. Caracterele individuale pot fi umbrite cu pînă la 256 de nivele de gri, putînd fi utilizate pînă la 256 de culori. Construcția poate fi pornită de la un desen în format "PICT", "Paint" sau "TIFF".

Fontul obținut este în format "Illustrator" și este disponibil astfel pentru multe programe.

Fontstudio 2.0

Ofertant: Letraset, 6000 Frankfurt



Preț: cca. 2200 DM

Sistem de operare: Apple/Macintosh

Recomandat pentru: DTP și procesoare de texte

TScript

TScript este un interpretor Postscript. Majoritatea programelor Mac, cum ar fi: Illustrator, Freehand, Pagemaker, pot scrie comenzile Postscript și într-un fișier și nu trebuie trimise direct la imprimantă.

Listarea este mijlocită ulterior de programe ca TScript care știu interpreta comenzile Postscript. Calitatea Postscript este posibilă astfel și cu imprimante ieftine. TScript poate converti comenzile pentru HP Laserjet și Deskjet, respectiv Deskwriter, ca și pentru Apple Laser și Image Writer. Aceste imprimante devin astfel compatibile Adobe Postscript.

TScript, care anterior a fost comercializat sub numele "MacRIP", permite și editarea fișierelor Postscript. Întru-cît Postscript este un limbaj de programare, utilizatorii versați pot prelucra documentele și manual. Rezultatul prelucrărilor poate fi privit înainte de imprimare într-o fureastră de previzualizare.

TScript

Ofertant: Zero One, 8600 Bamberg

Preț: cca. 450 DM

Sistem de operare: Apple/Macintosh

Recomandat pentru: DTP și procesoare de texte

Mai multe caractere de literă pentru textele D-voastră:**6 din cele mai importante soluții hard**

Din 1990 încoace au fost vândute, în întreaga lume, mai multe imprimante laser compatibile HP decât imprimante matriciale. Față de imprimantele matriciale, imprimantele laser au nu numai avantajul unei imprimări de calitate mai bună ci și avantajul utilizării mai multor fonturi, cu ajutorul unor casete de fonturi speciale, așa-numite "cartridges". Casetele extind nu numai paleta de fonturi ci oferă și caractere speciale, simboluri grafice și logo-uri. Unele casete transformă chiar o imprimantă convențională într-o imprimantă Postscript. Pentru a putea utiliza o casetă Postscript trebuie extinsă mai întâi memoria imprimantei la cel puțin 2 MByte.

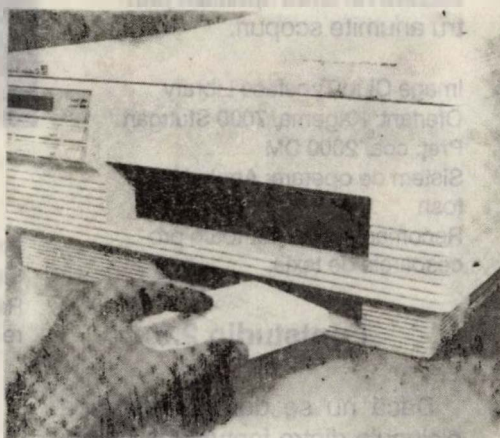
Casetele se diferențiază între ele și prin viteza de lucru și prin paleta de fonturi oferite. La testarea unui produs sînt folosite ca referință casetele firmelor americane Pacific Page sau IQ-Engineering.

Puteți cumpăra următoarele casete de fonturi:

Hewlett-Packard-International-Collection

Caseta este recomandată pentru toate modelele HP Laserjet II și Laserjet IID/IIP. Caseta conține trei fonturi (Letter Gothic, Prestige Elite și Courier; fiecare în forma normală, îngroșată și cursivă), cu distanțe fixe între caractere. În plus mai sînt conținute CG Times și Univers Normal (două scrieri proporționale). Cu Roman-8, PC-8, PC-D/N și PC 850 sînt puse la dispoziție patru seturi de caractere. Caseta poate fi utilizată și de modelele mai vechi 500+ și Laserjet 2000.

Ofertant: Macrotron, 8000 München
Preț: cca. 780 DM

**Hewlett-Packard-Cartridge**

Caseta este destinată modelelor HP Laserjet IID/P și Laserjet III și conține 35 de fonturi standard Postscript ale specialistului în fonturi Adobe. Pentru utilizarea casei este necesară o memorie de cel puțin 2 MByte.

Ofertant: Macrotron, 8000 München
Preț: cca. 1700 DM

Pacific-Page / Personal-Edition (PE)

Casetă de fonturi pentru HP Laserjet IIP/D, Laserjet III și Siemens PT-7500/PT-7800. Caseta conține 35 de fonturi Adobe Postscript. Pentru a putea fi utilizată este necesară o memorie de cel puțin 1,5 MByte. Dacă imprimanta dispune de o memorie de cel puțin 3 MByte atunci se lansează așa-numita Double-Buffering, ceea ce determină o sporire a vitezei de lucru.

Ofertant: AMS Computech,
8000 München
Preț: cca. 1700 DM

Pacific-Page /25 in One

Casetă de fonturi pentru Laserjet II, Laserjet IIP/D, Laserjet III, Laserjet IIID, Canon LBP8-II, Siemens PT-7500 și Siemens PT-7800. Caseta conține întreaga paletă de fonturi a 25 de casete HP diferite, în total 172 de fonturi. Dimensiunile de caractere posibile se găsesc în plaja de valori de la 4 pînă la 18 puncte. Nu este necesară o extensie de memorie.

Ofertant: AMS Computech,
8000 München
Preț: cca. 750 DM

Super-Cartridges 2

Casetă de fonturi pentru HP Laserjet II, IIP/D, III. Caseta conține 17 scrieri diferite și 8 seturi de caractere cu dimensiuni între 4 și 30 de puncte. Nu este necesară o extensie de memorie. Furnitura de livrare conține și drivere pentru diferite procesoare de texte și Windows 3.0

Ofertant: Computer Connections,
4030 Ratingen-Lintorf
Preț: cca. 1130 DM

Super-Cartridges 3

Casetă de fonturi pentru HP Laserjet III. Caseta conține 44 de fonturi Postscript. Toate fonturile pot fi imprimate la dimensiuni cuprinse între 3,5 și 30 de puncte. Nu necesită memorie suplimentară.

Ofertant: Computer Connection,
4030 Ratingen-Lintorf
Preț: cca. 1470 DM

(R.M.)

Memento: Comenzi Paradox 3.5

=	atribuie o valoare unei variabile, tablou sau câmp NumeVariabilă = expresie NumeTablou [Element] = expresie SpecCâmp = expresie	COEDIT	permite coeditarea unei tabelă COEDIT NumeTabelă
?	scrie în ecranul utilizator (canvas) PAL pe linia care urmează după cursor ? [ListăExpresii]	COEDITKEY	<Alt/F9> stabilește modul de lucru coedit COEDITKEY
??	scrie în ecranul utilizator PAL pe linia pe care se află cursorul ?? [ListăExpresii]	COPY	copiază o tabelă și familia ei COPY NumeTabelă1 NumeTabelă2
@	poziționează cursorul în ecranul utilizator @ Linie, Coloană	COPYFORM	copiază o formă COPYFORM NumeTabelă1 NumeFormă1 NumeTabelă2 NumeFormă2
ACCEPT	acceptă o valoare de la tastatură ACCEPT TipDată [PICTURE format] [MIN valoare] [MAX valoare] [LOOKUP NumeTabelă] [DEFAULT valoare] [REQUIRED] TO NumeVariabilă	COPYFROM-ARRAY	copiază elementele dintr-un tablou într-o înregistrare a unei tabelă COPYFROMARRAY NumeTablou
ADD	adaugă articolele dintr-o tabelă la cele din altă tabelă ADD NumeTabelă1 NumeTabelă2	COPYREPORT	copiază un raport COPYREPORT NumeTabelă1 NumeRaport1 NumeTabelă2 NumeRaport2
ARRAY	definește un tablou ARRAY NumeTablou [Număr]	COPYTO-ARRAY	copiază o înregistrare într-un tablou COPYTOARRAY NumeTablou
BACKSPACE	șterge caracterul aflat la stânga cursorului BACKSPACE	CREATE	crează o nouă tabelă CREATE NumeTabelă ListăDefCâmpuri CREATE NumeTabelă1 LIKE NumeTabelă2
BEEP	emite un semnal sonor (beep) BEEP	CREATELIB	crează o nouă bibliotecă de proceduri CREATELIB NumeBibliotecă [SIZE Număr]
CALCDEBUG	comută afișarea mesajelor pentru câmpurile calculate on sau off; setează șirul de atribuire pentru date invalide CALCDEBUG { ON OFF } Șir	CROSS-TABKEY	<Alt/X> stabilește referințele încrucișate pentru tabela curentă CROSSTABKEY
CANCELEDIT	întrerupe o operație CANCELEDIT	CTRL-BACKSPACE	<Ctrl/Backspace> șterge câmpul curent CTRLBACKSPACE
CANVAS	controlează afișarea pe ecranul utilizator PAL (canvas) CANVAS [ON OFF]	CTRLBREAK	<Ctrl/Break> oprește exec. prog. curent CTRLBREAK
CHECK	<F6> plasează sau șterge un marcaj check într-o cerere de interogare CHECK	CTRLEND	aceiași efect ca și <Ctrl/End> CTRLEND
CHECK-DESCENDING	<Ctrl/F6> plasează sau șterge un marcaj check descending într-o cerere de interogare CHECKDESCENDING	CTRLHOME	aceiași efect ca și <Ctrl/Home> CTRLHOME
CHECKPLUS	<Alt/F6> plasează sau șterge un marcaj checkplus într-o cerere de interogare CHECKPLUS	CTRLLEFT	aceiași efect ca și <Ctrl/←> CTRLLEFT
CLEAR	șterge ecranul utilizator PAL CLEAR [{ EOL EOS }]	CTRLPGDN	aceiași efect ca și <Ctrl/PgDn> CTRLPGDN
CLEARALL	<Alt/F8> șterge toate imaginile din spațiul de lucru CLEARALL	CTRLPGUP	aceiași efect ca și <Ctrl/PgUp> CTRLPGUP
CLEARIMAGE	<F8> șterge imaginea curentă din spațiul de lucru CLEARIMAGE	CTRLRIGHT	aceiași efect ca și <Ctrl/→> CTRLRIGHT
CLOSE PRINTER	închide accesul la o imprimantă CLOSE PRINTER	CURSOR	comută afișarea cursorului pe ecranul utilizator on / off, sau selectează forma cursorului CURSOR { BAR BOX NORMAL OFF }
		DEBUG	oprește execuția unui program (script) și pornește depanatorul DEBUG
		DEL	aceiași efect ca și DEL
		DELETE	șterge o tabelă și familia sa DELETE NumeTabelă
		DELETELIN	<Ctrl/Y> șterge începând de la cursor pînă la sfîrșitul liniei DELETELIN
		DITTO	<Ctrl/D> repetă valoarea câmpului din înregistrarea precedentă DITTO

CONSECO S.R.L.

cu sediul în Baia Mare , C.P.517, str. Siugariu 23/38, tel./fax: 40-94-36433

în colaborare cu

Societatea Comercială Româno - Canadiană

JM & CO. S.R.L.

din București, str. Miletin nr. 1, sectorul 3, tel.:20.33.24, tel./fax : 20.27.05

vă oferă :

A) REȚELE LOCALE ARTISOFT " LANtastic "

- LANtastic AE-2 Ethernet Starter Kit (2 adaptoare de rețea, soft pentru maxim 300 utilizatori, cablu coaxial subțire mufat de 8 m, 2 conectori T, 2 terminatori, documentații pentru hard și soft) - magistrala IDE. 875 \$
- LANtastic 2MBps Starter Kit (2 adaptoare de rețea, soft pentru maxim 300 utilizatori, cablu torsadat mufat de 5 m, 2 conectori T, 2 terminatori, documentații pentru hard și soft) - magistrala IDE. 625 \$
- LANtastic Voice Adapter Kit (inclusiv receptor telefonic). 124 \$
- LANtastic NETWORK EYE (program pentru supravegherea activității rețelei și modul ei de funcționare) 225 \$/rețea

De asemenea, vă putem oferi toate componentele hard și soft necesare pentru orice configurație dorită.

Pentru celelalte prețuri vă rugăm să ne contactați.

GARANȚIA OFERITĂ PENTRU TOATE PRODUSELE ARTISOFT ESTE DE 5 ANI.

B) Dispozitive de tușat benzi pentru imprimante de calculator.

Dispozitivul electric semiautomatizat de tușat, inclusiv o sticlă de tuș, costă 49000 lei, iar rezervele suplimentare 4700 lei/buc. Cu o rezervă de tuș se pot tușa cca. 30 de cartușe, de orice lățime - și orice tip de cartuș, între 6 - 30 mm fără a necesita scoaterea benzii din cartuș.

C) GOALKEEPER - o alternativă superioară la U.P.S.

- la o cădere de tensiune între 0 - 3 sec. GOALKEEPER suplinește instantaneu sursa de curent continuu a calculatorului.
- la o cădere de tensiune între 3 - 90 sec.- GOALKEEPER face back-up în 10 - 90 sec., depinzând de mărimea memoriei RAM. Dacă între timp revine tensiunea, calculatorul nu trebuie oprit, putându-se continua lucrul.
- la o cădere de tensiune mai mare de 90 sec. - GOALKEEPER salvează toate datele pe harddisk și oprește calculatorul. Când revine tensiunea, se pot restaura toate datele.

Cerințe hardware și software: un calculator IBM PC, XT sau AT sau compatibil, harddisk, DOS 2.1 sau versiuni mai noi, cel puțin 256 Kb RAM.

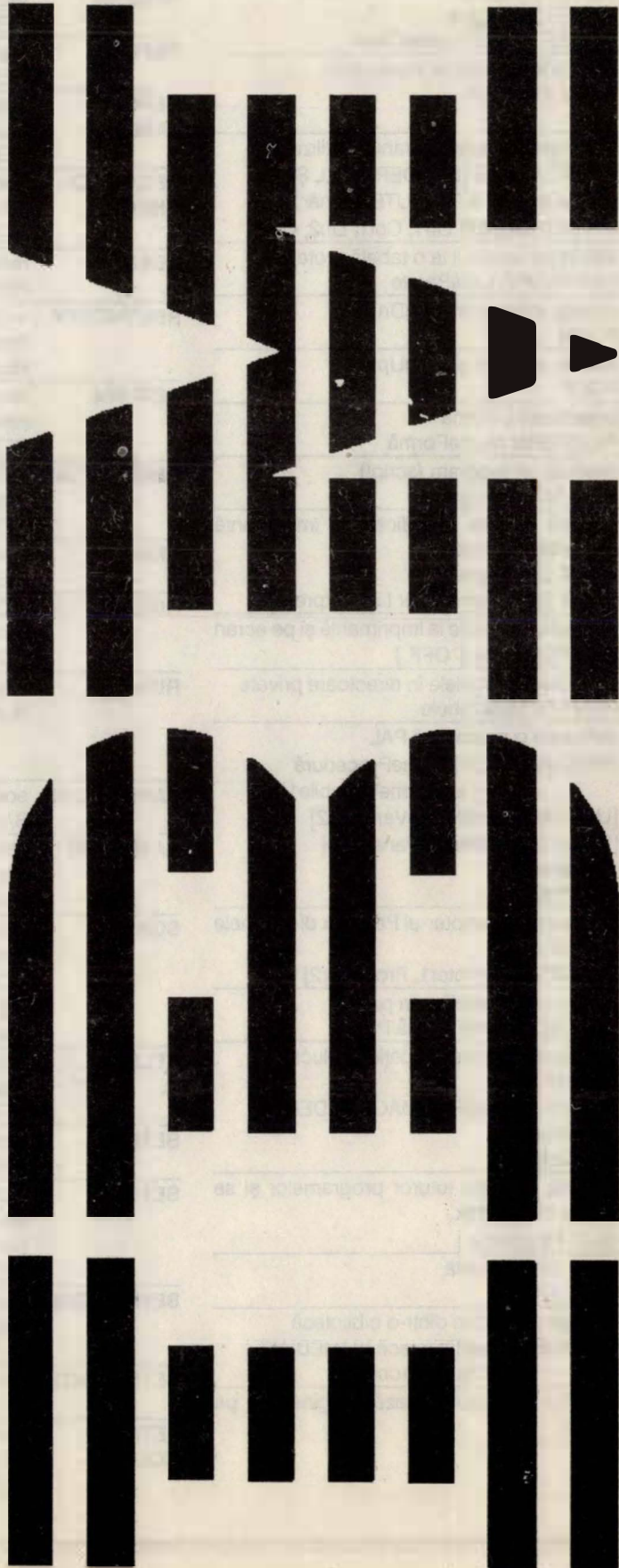
Dimensiuni : 21.5 x 14.5 x 4.7 cm

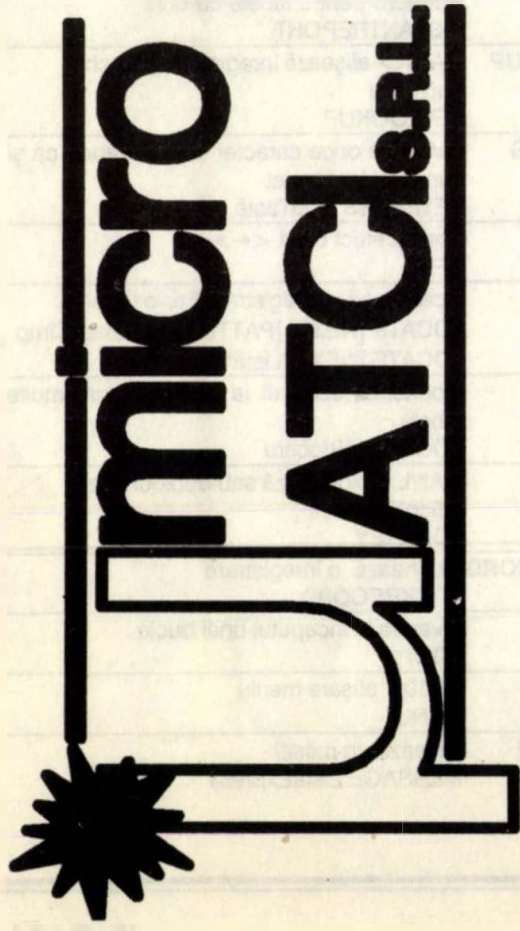
Greutate : 3 Kg

Prețul : 69500 LEI

MOVETO	mută cursorul la un câmp, înregistrare sau imagine specificată MOVETO SpecificatorCâmp MOVETO RECORD Număr MOVETO Număr MOVETO Imagine MOVETO FIELD NumeCâmp
OPEN PRINTER	deschide accesul la imprimantă OPEN PRINTER
PAINTCANVAS	stabilește atributele ecranului utilizator PAINTCANVAS [BORDER] [FILL Șir] {ListăOpțiuni ATTRIBUTE Număr BACKGROUND} Lin1, Col1, Lin2, Col2
PASSWORD	stabilește accesul la o tabelă protejată PASSWORD ListăParole
PGDN	același efect ca și <PgDn> PGDN
PGUP	același efect ca și <PgUp> PGUP
PICKFORM	selectează o formă PICKFORM NumeFormă
PLAY	execută un program (script) PLAY NumeProgram
PRINT	listează valorile specificate la imprimantă sau la într-un fișier PRINT ListăExpresii PRINT FILE NumeFișier ListăExpresii
PRINTER	trimite toate ieșirile la imprimantă și pe ecran PRINTER { ON OFF }
PRIVTABLES	memorează tabelele în directoare private PRIVATE ListăTabele
PROC	definește o procedură PAL PROC [CLOSED] NumeProcedură ((ListăNumeVariabile1)) [USEVARS ListăNumeVariabile2] [PRIVATE ListăNumeVariabile3] Comenzi ENDPROC
PROMPT	înlocuiește prompter-ul Paradox din primele două linii PROMPT [Prompter1, Prompter2]
PROTECT	protejează o tabelă prin parolă PROTECT NumeTabelă Parolă
QUERY	plasează o cerere în spațiul de lucru QUERY [{ TABLEORDER IMAGEORDER }] FormeCerere ENDQUERY
QUIT	termină execuția tuturor programelor și se revine în Paradox QUIT [Expresie]
QUITLOOP	ieșire dintr-o buclă QUITLOOP
READLIB	extrage proceduri dintr-o bibliotecă READLIB NumeBibliotecă [IMMEDIATE] ListăNumeProceduri
REFRESH	<Alt/R> reîmprospătează imaginea de pe ecran

RELEASE	elimină proceduri, variabile, sau tablouri din memorie RELEASE PROCS { ALL ListăNumeProc } RELEASE VARS { ALL ListăNumeVar }
RENAME	redenumește o tabelă și familia acesteia RENAME NumeTabelă1 NumeTabelă2
REPORT	imprimă un raport REPORT NumeTabelă NumeRaport
REPORT-TABLES	listează tabelele înlănțuite într-un raport multitabelar REPORTTABLES
REQUIRED-CHECK	comută on/off procedura de validare pentru câmpurile interogate REQUIREDCHECK { ON OFF }
RESET	revenire la meniul principal Paradox RESET
RESYNCKEY	<Ctrl/L> resincronizează înregistrările într-o formă multitabelară RESYNCKEY
RETURN	revenire dintr-un program (script) sau procedură RETURN [Expresie]
REVERSE TAB	<Shift/Tab> mută cursorul la câmpul anterior REVERSE TAB
RIGHT	același efect ca și <-> RIGHT
ROTATE	<Ctrl/R> rotește câmpurile aflate la dreapta cursorului ROTATE
RUN	execută o comandă sau un program DOS RUN [BIG] [SLEEP Număr] [NOREFRESH] [NORESTORE] [NOSHELL] ComandăDOS
SAVETABLES	scrie buffer-ele tabelelor pe disc SAVETABLES
SAVEVARS	salvează valorile variabilelor și tablourilor într-un fișier SAVEVARS { ALL ListăNumeVariabile }
SCAN	parcurge o tabelă înregistrare cu înregistrare SCAN [FOR Condiție] [Comenzi] ENDSCAN
SELECT	selectează o comandă sau tastează un răspuns la prompter-ul unui meniu SELECT Expresie
SETDIR	schimbă unitatea sau directorul de lucru SETDIR CaleDOS
SETKEY	redefinește operația executată de o combinație de taste (definește un macro pentru tastatură) SETKEY CodTastă [Comenzi]
SETMARGIN	stabilește marginea din stînga pentru textul afișat SETMARGIN { OFF Număr }
SETMAXSIZE	stabilește dimensiunea maximă a unei tabele SETMAXSIZE Număr
SETNEG-COLOR	stabilește culoarea care va fi folosită pentru evidențierea valorilor negative dintr-o tabelă SETNEG-COLOR { CURRENCY NUMERIC BOTH } { ON OFF }





str. Gheorghe Doja nr. 36
C.P. 64 O.P. 1
4300 Tîrgu Mureş
Telefon/Telefax: 954/31660

**vă oferă întreaga gamă
de produse originale**

DO_ITI	<F2> termină sau execută o operație DO_ITI	FORMTABLES	listează tabelele îmbricate într-o formă multitabelară FORMTABLES NumeTabelă NumeFormă NumeTablou
DOS	<Ctrl/O> suspendă Paradox și accede sis- temul de operare DOS DOS	GRAPHKEY	<Ctrl/F7> reprezintă grafic tabela curentă GRAPHKEY
DOSBIG	<Alt/O> suspendă Paradox și accede S.O. DOS cu max. de memorie RAM disponibilă DOSBIG	GROUPBY	<Shift/F6> grupează articolele pentru cererea curentă fără să le afișeze GROPUPBY
DOWN	același efect ca și <↓> DOWN	HELP	<F1> afișează ecrane de ajutor HELP
DOWNIMAGE	<F4> mută în jos o imagine în spațiul de lucru sau într-o formă multitabelară DOWNIMAGE	HOME	același efect ca și tasta <Home> HOME
ECHO	afișează modul de lucru intern al unui program (script) ECHO {FAST SLOW OFF NORMAL }	IF	execută anumite comenzi în funcție de îndeplinirea unor condiții IF Condiții THEN Comenzi1 [ELSE Comenzi2] ENDIF
EDIT	permite editarea unei tabele EDIT NumeTabelă	IMAGERIGHTS	setează sau șterge drepturile de acces la o tabelă IMAGERIGHTS { {UPDATE READONLY} }
EDITKEY	<F9> stabilește modul de lucru Edit EDITKEY	INDEX	crează un index secundar pentru o tabelă INDEX [MAINTAINED] NumeTabelă ON NumeCîmp
EDITLOG	acceptă sau reface modificările făcute într-o tabelă așa cum au fost înregistrate în jurnalul tranzacțiilor EDITLOG { MARK REVERT INDEX PERMANENT }	INFOLIB	afișează numele de proceduri dintr-o bibliotecă INFOLIB NumeBibliotecă
EMPTY	șterge toate înregistrările dintr-o tabelă EMPTY NumeTabelă	INS	același efect ca și <Ins> INS
END	același efect ca și <End> END	INSTANTPLAY	<Alt/F4> execută programul Instant INSTANTPLAY
ENTER	același efect ca și <Enter> ENTER	INSTANT- REPORT	<Alt/F7> listează la imprimantă raportul standard pentru tabela curentă INSTANTREPORT
ESC	<Esc> revenire la meniul anterior, prompter sau spațiul de lucru ESC	KEYLOOKUP	<Alt/K> afișează înregistrările cu chei duplicate KEYLOOKUP
EXAMPLE	<F5> introduce un exemplu pentru o cerere EXAMPLE	KEYPRESS	transmite orice caracter spre Paradox ca și cum ar fi fost tastat KEYPRESS CodTastă
EXECPROC	execută o procedură fără argumente EXECPROC NumeProcedură	LEFT	același efect ca și <←> LEFT
EXECUTE	execută o secvență de comenzi PAL EXECUTE Comenzi	LOCATE	localizează o înregistrare într-o tabelă LOCATE [NEXT] [PATTERN] ValcareCîmp LOCATE [NEXT] ListăValoriCîmp
EXIT	părăsește Paradox EXIT	LOCK	blochează accesul la una sau mai multe tabele LOCK ListăBlocare
FIELDVIEW	<Alt/F5> sau <Ctrl/F> stabilește modul de lucru editare cîmp (field view) FIELDVIEW	LOCKKEY	<Alt/L> blochează sau deblochează articolul curent LOCKKEY
FIRSTSHOW	face ca imaginea curentă să fie prima imagine din spațiul de lucru FIRSTSHOW	LOCKRECORD	blochează o înregistrare LOCKRECORD
FOR	controlează de cîte ori va fi executată o secvență de instrucțiuni FOR NumeVariabilă [FROM Număr1] [TO Număr2] [STEP Număr3] Comenzi ENDFOR	LOOP	revenire la începutul unei bucle LOOP
FORMKEY	<F7> comută între modurile de afișare tabelar și formă FORMKEY	MENU	<F10> afișare meniu MENU
		MESSAGE	afișează un mesaj MESSAGE ListăExpresii

Cine ridică mânușa?

În numărul din septembrie 91 al revistei Dr. Dobbs' Journal Ray Valdés, într-un articol intitulat "Little Languages - Big Questions", lansează o provocare.

Din punct de vedere "calculatoristic" (computationally speaking), unele limbaje mici sînt foarte "puternice", în ciuda aparenței lor insignifiante, iar altele sînt niște "piticoți" (travestiți în giganti). Deși există teste care să permită evaluarea codului generat de un compilator sau a vitezei de compilare, nu există încă teste (benchmarks) care să permită evaluarea "puterii" unui limbaj de programare - a posibilităților de a exprima în el anumite lucruri. Într-un sens mai larg, toate limbajele - pînă la cele

mai rudimentare - sînt echivalente, pentru că toate sînt capabile să acționeze ca mașini Turing. Privite în detaliu, totuși, limbajele sînt suficient de diferite - o comparație fiind posibilă la fel ca o comparație între mere și portocale.

În ciuda acestei observații, expresivitatea unui limbaj poate fi măsurată - prin ușurința cu care pot fi exprimate probleme fundamentale.

Un test, mai elocvent, este exercițiul de programare cu care Ken Thompson (coautor al Unix-ului) își trecea timpul în studenție. Conform unei declarații a lui (1983, la primirea premiului ACM Turing) - exercițiul ce urmează este popular "din același motiv pentru care sînt populare speciile de animale cu 3 picioare". Scopul este de a scrie cel mai scurt program autoreproducător: "mai exact, de a scrie un program sursă care, după compilare și execuție, va produce ca ieșire o copie exactă a sursei sale".

Thompson adaugă: "dacă nu ați făcut niciodată acest lucru, vă sugerez cu insistență să încercați singur. Descoperirea soluției este o revelație care depășește cu mult orice beneficiu obținut prin faptul că cineva vă dă soluția de-a gata".

Programul lui Thompson, în C, este prezentat în Exemplul 1. Deși el îl consideră ca punct de plecare pentru "cel mai isteț program pe care l-am scris vreodată", codul constă din 223 linii de C oarecum banal.

Prin comparație, echivalentul acestui program poate fi exprimat, foarte elegant, în Lisp pur, în numai 3 linii de cod (ex. 2). (Lisp-ul pur este un subset al Lisp-ului constînd din funcții care nu au efecte colaterale, deci funcții care din punct de vedere matematic sînt "pure". Codul din Ex. 2 este scris de John Mc Carthy și Ca-

Care credeți că este cel mai folosit limbaj de programare folosit în prezent? Deși multora dintre noi le-ar place, nici C-ul și nici Pascal-ul nu stăpînesc lumea. E prea tîrziu pentru ADA sau ASM, și prea devreme pentru C++ . Fortran sau Cobol. Atunci? Răspunsul corect: nici unul din toate acestea.

De fapt, limbajul de programare cu cea mai largă răspîndire este limbajul de macroui din Lotus 1-2-3! Sigur, programatorii "adevărați" nu folosesc macrolimbajul Lotus, dar un număr foarte mare de oameni o fac - și obțin rezultate utile, în fiecare zi.

Mumps este un limbaj de programare ce folosește o sintaxă asemănătoare Basic-ului. Mumps leagă un amestec bogat de funcționalități sofisticate tip bază de date de o sintaxă - schelet, constînd din 26 de comenzi alcătuite din cîte o literă a alfabetului, incluzînd "G" pentru "goto". Această ciudățenie de limbaj este baza pentru unul din cele mai mari proiecte de dezvoltare software lansate vreodată, contractul guvernamental de 1,2 miliarde \$, ce începe acum, pentru computerizarea spitalelor americane.

rolyn Talcott).

În exemplul 3, este prezentat un program echivalent în Mumps - un limbaj asemănător Basic-ului, care accede o sofisticată mașină - bază - de date - asociativă (associative database engine). Deși listingul are numai cîteva linii, codul este extrem de enigmatic și criptic.

Cu toate că fiecare dintre aceste programe are la bază aceeași logică, pașii necesari ajungerii la țintă, în fiecare limbaj, sînt atît de diferiți încît s-ar putea paria că fiecare program implementează un algoritm distinct.

Cum se descurcă limbajul Dvs. preferat (sau cel mai nesuferit)? Soluțiile Dvs. pentru acest exercițiu sînt așteptate (atît de noi, cît și) de Dr. Dobb's Journal, M&T Publishing Inc., 501 Galveston Dr., Redwood City, CA 94063, for Ray Valdés. Listingurile cele mai interesante, elegante, și/sau bizare vor fi publicate. Un premiu special va fi acordat primei soluții scrise în limbajul de macroui propriu Lotus-ului.

(I.F.)

Ex. 1. C

```
char s[] = {
    '\t',
    '0',
    '\n',
    '}',
    ',',
    '\n',
    '}',
    '\n',
    '\n',
    '\n',
    '\n',
    '\n',
    '\n',
    '\n',
    ... 213 lines deleted ...
};
/*
 * Sirul s este o reprezentare a acestui program,
 * de la '0' pina la end.
 */
main ()
{
    int i;

    printf ("char\ts[] = {\n");
    for (i = 0; s[i]; i++)
        printf ("\t%d, \n", s[i]);
    printf ("%s", s);
}
```

Ex. 2 Lisp pur

```
((lambda (x)(list x (list (quote quote) x)))
 (quote
 (lambda (x)(list x (list (quote quote) x )))))
```

Ex. 3 Mumps

```
S N = $(C13) $(C10) $(C9) S Q = $(C34) S S = "W
N "" S N = $(C13) $(C10) $(C9) S Q = $(C34) S
S = "" Q; F I = 1:1:$(L(S) S C = $(E(S,I) W
$$ (C = $(C34):Q_Q,1:C); W Q_N; F I = 1:1:$(L(S) S
C = $(E(S,I) W $$ (C = $(C59):N, 1:C); Q;"
W N "" S N = $(C13) $(C10) $(C9) S Q = $(C34) S
S = " Q
F I = 1:1:$(L(S) S C = $(E(S,I) W $$ (C = $(C34):Q_Q,1:C)
W Q_N
F I = 1:1:$(L(S) S C = $(E(S,I) W $$ (C = $(C59):N, 1:C)
Q;
```

Turbo Vision

sau cum să moștenim roata în viziunea Borland

În primăvara acestui an profetul limbajelor de programare orientate obiect, Borland, a împărțit credincioșilor săi ultima sa viziune. Turbo Vision nu este doar o colecție de obiecte ce trasează cadrul general al unei aplicații, ci este și suportul unui concept nou în programarea PC-urilor. Deși timpul de inițiere a crescut, a scăzut dramatic durata finisării unei aplicații iar cizelarea ei poate fi realizată impecabil.

Ceea ce era previzibil încă din toamna lui 1988 a devenit acum evident: unificarea suprafețelor utilizator, a interfețelor program și a protocoalelor de control în oferta de programe aplicative. Windows 3.0 vine în forță iar producătorii se grăbesc să se alinieze noului standard.

Dintre producătorii de compilatoare primul care a reacționat cu promptitudine este Borland care înaintează cu încă un pas cu ale sale Turbo Pascal 6.0 și Turbo Pascal for Windows (TPW). Succesul nu este asigurat atât de calitatea remarcabilă a compilatoarelor în sine cât de cele două biblioteci livrate împreună cu acestea. Dacă Object Windows Library (OWL) -oferit cu TPW- furnizează un cadru general pentru o aplicație Windows, Turbo Vision (TV) - livrat cu TP 6.0 - oferă aceeași structură pentru o aplicație în mod text rulând sub MS-DOS.

TV și OWL prezintă o similitudine remarcabilă deși implementează idei identice pentru două platforme foarte diferite (frapantă nu este atât diferența mod text - mod grafic, cât diferența mod real de adresare - mod protejat / multitasking). Amândouă oferă o arhitectură de

program bazată pe tratarea evenimentelor, împreună cu componentele soft pentru interfețe cu utilizatorul și respectiv cu sistemul.

Ambele sînt în mod inevitabil orientate obiect. Ideea de bază este că realizarea unei aplicații concrete se face prin moștenirea unui cadru general de aplicație, gol, la care se adaugă cod pentru realizarea sarcinilor specifice aplicației. Cadrul conține cuiele necesare atîrnării pieselor componente și este dispus pe axele carteziane OOP ale generalității și specificității. Preluînd comparația dragă autorilor lui Turbo Vision, scheletul unei aplicații se constituie din meniuri, ferestre, cîmpuri de editare, rutine de tratare a erorilor, de control al mouse-ului, în timp ce carnea rezolvă probleme specifice aplicației. Cele mai multe aplicații se conformează unei asemenea reprezentări, iar programatorii scriu adesea mai multe linii de cod pentru descrierea scheletului decît pentru materializarea cărnii, nereușind să facă o distincție clară între ele. Acum Borland ne propune ca în loc să reinventăm roata s-o moștenim. Și nu o roată oarecare ci mii de linii de program testat și optimizat ce ar putea reduce propria aplicație la surprinzător de puțin cod.

Aici este momentul să ne separăm de OWL. Să lăsăm bufnița să zboare, căci toate considerațiile care urmează rămîn perfect valabile și pentru ea, cel puțin din punct de vedere conceptual. Portabilitatea codului dinspre TV înspre OWL este remarcabilă, diferențele ținînd de încorporarea în Windows a unei părți din arhitectura de control a evenimentelor, de modul specific

de creare a resurselor grafice. Majoritatea obiectelor TV se regăsesc în OWL.

Abstracțiuni

Turbo Vision se prezintă cu trei categorii de obiecte, grupate pe criterii funcționale: obiecte primordiale, obiecte vizibile pe ecran (le vom referi deseori ca *views*), precum și obiecte invizibile.

În ultimele două categorii există numeroase tipuri de obiecte cu caracter abstract. Ele au practic potențialul de a face orice dar rămîn prea generale pentru a face ceva concret. Utilizatorul trebuie să definească tipuri de obiecte descendenți care să întreprindă acțiuni specifice folosind cît mai mult din codul cu caracter general al obiectului părinte. Paradoxal, tocmai existența acestor obiecte abstracte conferă soliditate conceptuală bibliotecii și servește din plin dezideratul reducerii efortului de programare.

Să considerăm pentru exemplu trei tipuri de obiecte: *TWindow*, *TDesktop* și *TProgram*. Acestea au în comun o caracteristică esențială: fiecare poate conține, (în sensul reprezentării pe ecran), mai multe obiecte vizibile (*views*) care se află pe același nivel al aplicației și "concertează" împreună. *TProgram* grupează un meniu-linie (*TMenuBar*), o linie status (*TStatusLine*) și un suport - *TDesktop* - care la rîndul lui poate conține un număr practic nelimitat de *view*-uri: boxe de dialog, cîmpuri de introducere a datelor, ferestre, etc. O fereastră (*TWindow*) conține un cadru, bare de desfășurare (*scrolling bars*), diverse alte obiecte: butoane, liste, ...

. Interacțiunea dintre obiectele conținute, selectarea lor cu mouse-ul sau tastatura se face prin intermediul obiectului posesor, *TProgram*, *TDesktop* sau *TWindow*. Această importanță trăsătură, de a deține și coordona *view*-uri, a impus necesitatea definirii tipului abstract de obiect grup - *TGroup* - din care derivă cele trei obiecte. În discuție. Pentru programator devine banală acum derivarea din *TGroup* a unui tip original de obiect-cadru de aplicație, cu un comportament similar lui *TProgram* dar cu un meniu standard vertical și o linie status situată în partea de sus a ecranului, de exemplu.

Utilitatea instanțierii unui tip de obiect depinde de circumstanțe. Numeroase tipuri au metode abstracte ce impun definirea lor în descendenți. Există și categoria metodelor pseudo-abstracte care oferă un minimum de acțiuni implicite; redefinirea lor nu este necesară decât dacă nu corespund scopurilor urmărite de programator. Ca o regulă generală, pe măsură ce se coboară în ierarhia TV, de la tipurile primordiale de obiecte la descendenții lor cei mai îndepărtați, obiectele devin tot mai specializate și mai puțin abstracte, putând fi instanțiate direct, fără a mai deriva descendenți.

Obiecte și poze

Revenind la categoriile de obiecte Turbo Vision remarcăm trei tipuri primitive, extrem de simple. *TPoint* și *TRect* reprezintă un punct și respectiv o zonă dreptunghiulară pe ecran. Fără descendenți și având puține metode, ele sînt utilizate intens de toate tipurile de obiecte vizibile. De altfel *view*-urile sînt fără excepție dreptunghiulare. "Tatăl tuturor", *TObject*, este un tip abstract, fără cîmpuri, cu trei metode: *Init*, baza tuturor constructorilor TV, alocă memorie și inițializează cu zero toate cîmpurile descendenților; *Free* dealcă memoria iar *Done* este un destructor

abstract ce trebuie redefinit în descendenți. *TObject* este tipul de bază din care derivă absolut toate tipurile de obiecte TV cu excepția lui *TPoint* și *TRect*.

Cărămida de bază a unei aplicații TV este obiectul vizibil - *view* - descendent al *TView*. În fapt tot ceea ce apare pe ecran trebuie să fie sau să aparțină unui obiect vizibil, iar proprietatea sa cea mai importantă este aceea de a ști să se autoreprezinte ori de cîte ori condițiile o cer. *View*-urile pot și trebuie să coexiste; ca urmare se pot suprapune, redimensiona, pot comunica între ele, se pot deplasa pe ecran.

Acelor cititori familiarizați cu Windows nu le va fi greu să sesizeze particularitățile tipurilor de obiecte vizibile ce urmează a fi trecute în revistă. De la ferestre la butoane, Turbo Vision reușește o remarcabilă emulare a graficii foarte personalizate de sub Windows cu umila paletă a celor 256 de caractere IBM. Chiar dacă în ceea ce privește aspectul "text" ar mai fi cîte ceva de cîrțit, comportamentul *view*-urilor este impecabil iar timpii de afișare a imaginii pe ecran sînt insesizabili - cu toții ne amintim cu neplăcere lenea cu care Windows pictează ecranele AT-urilor cu plăci grafice lente.

Portrete de familie

Numeroase sînt tipurile "grup" de obiecte vizibile. A programa fără ele în TV este de neconceput: aplicația însăși este un grup care acoperă întregul ecran, dar aceasta nu se observă întrucît dialogul utilizatorului decurge cu obiecte specifice, administrate de aplicație.

Deseori, ceea ce pare a fi un singur obiect este în fapt un grup de obiecte aflate în interacțiune. Astfel se prezintă boxele de dialog - descendenți ai lui *TDialog* - ce conțin "*subview*"-uri precum butoanele (*TButton*), căsuțele de bifare (*TCheckBoxes*), butoanele radio (*TRadioButtons*), cîmpuri de edita-

re (*TInputLine*). *TDialog* este un caz mai special de fereastră, de dimensiune fixă, dar care poate fi "plimbată" pe ecran. Uneori existența ei se limitează strict la interacțiunea cu utilizatorul: atunci dialogul este imperativ, utilizatorul nu are posibilitatea s-o lase să aștepte și să dialogheze cu alte obiecte. Se spune că dialogul este în stare "modală" de execuție. Obiectele modale se diferențiază de cele "inseparate" în grup, ce pot fi selectate și abandonate, iar distrugerea lor se face în mod explicit și nu la abandonarea lor. De remarcat că există întotdeauna cel puțin un obiect modal într-o aplicație, iar acesta este aplicația însăși. Indiferent că derivă din *TApplication* sau este un tip mai special derivat de programator din *TGroup*, ea este executată în stare modală în cadrul unei secvențe "clasice":

```
prog.Init;
prog.Run;
prog.Done.
```

Vă puteți imagina un main mai simplu?

Fereastră este probabil cel mai versatil *view*. Descendenții lui *TWindow* se bucură de calități speciale: pot fi "apucați" de un colț și deplasați pe ecran sau prinși de colțul opus și redimensionați. Ferestrele pot fi închise sau numai ascunse, cele numărate (*n*) pot fi selectate cu combinația de taste *Alt-n* și, evident, aruncă "umbre" pe suport. Desigur, o fereastră rămîne goală dacă nu este "umplută" de programator cu obiecte iar calitățile enumerate sînt de asemenea la discreția lui, putîndu-și personaliza ferestrele.

Amintit anterior, suportul - *TDesktop* - se bucură de două metode speciale, de mare efect, ce permit organizarea obiectelor subordonate redimensionabile în cascadă (suprapuse și decalate precum un pachet de cărți de joc) sau în domino (spațiul suportului se împarte în părți egale alocate fiecare cîte unui obiect). Windows a mai pierdut astfel două privilegii. Reprezentarea instantanee pe ecran și absența pîlpîirilor determi-

nate de desenări succesive ne determină să săpăm mai adînc și descoperim că TV întreține cîte un buffer de memorie video-cache pentru fiecare grup, în care finalizează reprezentarea și abia apoi "toarnă" imaginea pe ecran. Două metode, *Lock* și *Unlock* sînt responsabile de ascunderea și prezentarea cache-ului pe ecran. Să remarcăm de asemenea că memoria cache nu este numai dinamic alocată ci și deplasată și reorganizată pentru a evita fragmentarea memoriei libere rămase la dispoziția programatorului.

Portrete

Dacă obiectele-grup se manifestă în background, consistența TV este conferită în cele din urmă de obiectele terminale - *view-uri* conținute de grupuri și avînd funcții și aspect specific. Un obiect vizibil terminal nu poate fi grup. El nu poate conține alte obiecte vizibile și încheie de aceea lanțurile de *view-uri* subordonate.

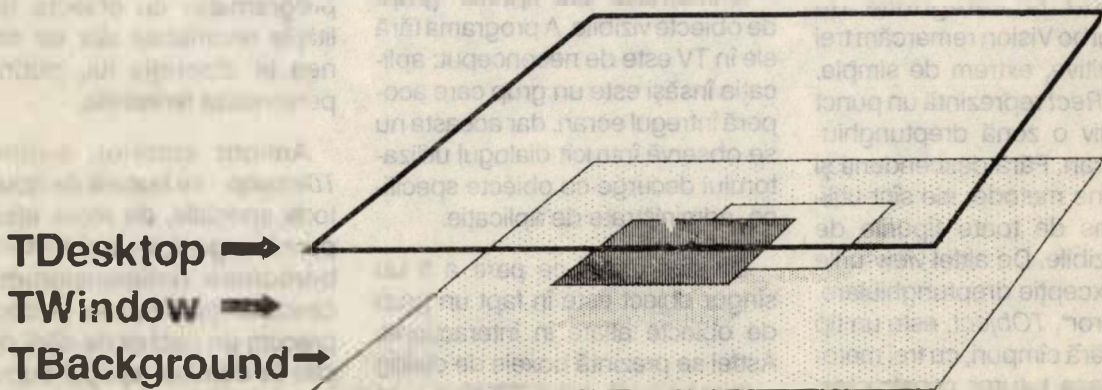
Butoanele au sarcina să asocieze evenimente de tastatură sau mouse cu o comandă. Ele generează o comandă specifică cînd sînt "apăsate" cu mouse-ul sau cu tasta activă corespunzătoare. Estetica lor este indiscutabilă: umbra dispare în momentul "apăsării".

Evidența standardizării unei interfețe utilizator este dată de meniuri. Cu tipul *TApplication*, TV s-a aliniat la standardul Windows de meniuri pull-down; totuși meniul linie - *TMenuBar* - și meniul pop-up - *TMenuBar* - se pot înlănțui ușor pe oricîte nivele pentru a defini orice specie imaginabilă de meniu. Instanțierea unui meniu este elegantă: într-o singură instrucțiune se specifică lista înlănțuită de meniuri și elemente de meniu, fiecare element indicîndu-l pe următorul. Evident meniurile sînt sensibile la mouse, dar implementează și taste active (shortcuts); fiecărui element i se poate asocia un indice de context de help, care permite apariția unei explicații pe linia status sau invocarea unei ferestre de help.

Între obiectele terminale, discrete dar foarte importante sînt barele de desfășurare (*TScrollBar*) precum și tipul de desfășurător (*TScroller*). În filozofia TV în care prezentarea informației pe ecran este optimizată prin posibilitatea de suprapunere a obiectelor, informația trebuie să poată fi desfășurată sus-jos sau stînga-dreapta într-o fereastră de dimensiuni mici. Barele de desfășurare pot furniza control pe verticală sau orizontală și răspund atît la mouse cît și la tastatură. Modul în care interacționează prin mesaje barele de desfășurare și obiectele ce sînt desfășurate vom avea ocazia să-

prezentăm mai jos. Funcția foarte comună de inspectare și selectare din liste este implementată prin *TListViewer*. Acest tip este asociat cu una sau două bare de desfășurare iar selectarea unui articol se poate face atît cu mouse-ul cît și cu săgețile tastaturii. Utilizatorul poate redefini metodele *SelectItem* și *FocusItem* pentru a îndeplini sarcini specifice la selectarea sau simpla focusare a unui articol; de asemenea poate interoga lista pentru a afla articolul curent. Tipul listei este la dispoziția programatorului: listă înlănțuită de pointeri, vector de înregistrări, etc. *TListViewer* deține o metodă specifică - *GetText* - prin care se definește modul de prezentare al articolului pe ecran. Un descendent - *TListBox* - implementează o boxă de vizualizare a listelor de string-uri, utilizată de foarte sofisticatele tipuri de obiecte de vizualizare și selectare a numelor de fișiere și directoare - *TFileDialog* și *TChDirDialog*.

Liniile de introducere a datelor (*TInputLine*) sînt componente indispensabile pentru editarea unei linii de text sau în construcția unei machete de culegere a datelor; suportă toate funcțiile standard de editare, inclusiv marcarea de blocuri cu ajutorul mouse-ului. Editarea poate avea loc într-o fereastră de dimensiuni mai mici decît lungimea liniei. Deseori le sînt asociate



Un grup poate fi conceput ca un sandwich de obiecte în care felia de deasupra este transparentă.

mecanisme de selecție a unei valori anterior introduse - așa numitele liste istorice (*THistory*, *THistoryWindow*, *THistoryViewer*).

Să privim un ultim "portret". Linia status - *TStatusLine* - este un obiect mai important decât s-ar putea crede: permite o afișare dinamică pe ecran, sensibilă la contextul aplicației și implementează comenzi foarte importante (cum ar putea fi comanda de părăsire a aplicației) ce trebuie să fie permanent accesibile. Cîmpul *Items* indică o listă înlănțuită de înregistrări conținînd "etichete" asociate cu comenzi și taste active (etichetele sînt, evident, sensibile la mouse). Un alt cîmp, *Defs* indică o listă de string-uri help ce pot fi asociate cu opțiuni de meniu prin intermediul indicilor de context help. TV trișează un pic cu liniile status: ele sînt active chiar și în timpul unei execuții modale, evenimentele și comenzile generate de linia status fiind tratate ca și cînd ar proveni de la obiectul modal.

Rîuri subterane de cod

Și totuși cum arată schema logică a unui program Turbo Vision? Unde începe execuția și unde se termină? În programarea clasică sîntem obișnuiți să urmărim execuția liniilor de program una după alta precum meandrele unui rîu de la izvor la vărsare.

În fapt avem de-a face cu o rețea de rîuri subterane de cod ce apar temporar la lumina zilei pe fundul unui aven pentru a dispărea și curge apoi pe căi neștiute. Programatorul nu are acces decât la cursul paralel, de suprafață, pe care el însuși îl crează. Dar rîul propriu-zis curge în adînc.

De ce o asemenea structură? Pentru că o aplicație TV este prin excelență un program interactiv și este greu de imaginat o altă arhitectură decât cea bazată pe tratarea evenimentelor. TV controlează în totalitate input-ul utilizatorului și-l împachetează în înregistrări Pas-

cal numite "evenimente" pe care le dispecerizează spre view-urile corespunzătoare. Programatorul nu-i mai rămîne decât să scrie cod care să știe să trateze evenimentele interesante, fără a mai culege evenimentele, fără a mai trata diferențiat sursele de input, de la mouse sau tastatură.

Evenimentele pot fi concepute ca mici pachete de informație ce descriu semnale discrete la care aplicația trebuie să răspundă. Fiecare tastă apăsată, fiecare mișcare de mouse, o condiție generată de altă componentă de program constituie un eveniment distinct. TV recunoaște patru tipuri de evenimente: de mouse, tastatură, mesaje și evenimente nule.

Mouse-ul poate genera următoarele evenimente: buton apăsător, buton eliberat, mouse mișcat, mouse "auto" (apăsarea continuă a unui buton), fiecare eveniment conținînd informație relativă la poziția mouse-ului pe ecran și butonul în cauză. Evenimentele de tastatură sînt banale: apăsarea unei taste generează un eveniment ce conține informație referitoare la poziția tastei pe tastatură și codul ASCII returnat.

Mesajele pot fi comenzi, emisiuni (*broadcasts*) și mesaje utilizator; diferențierea mesajelor ține de modul în care sînt tratate. Cele mai multe evenimente sfîrșesc prin a fi transformate în comenzi. Selectarea unei opțiuni dintr-un meniu cu ajutorul unei taste active determină obiectul meniu să transforme evenimentul de tastatură într-un eveniment comandă. O comandă specifică este "interceptată" de obiectul "instruit" să o facă.

Comportamentul unui *view* este în totalitate determinat de o metodă numită *HandleEvent* ce primește ca argument evenimentul transmis obiectului. În această metodă programatorul definește modul în care obiectul trebuie să reacționeze la primirea unor anumite comenzi sau evenimente de orice tip. Întrucît obiectul trebuie să aibă comportamentul specific al

clasei căreia îi aparține, este obligatorie moștenirea metodei *HandleEvent* a tipului de obiect părinte. Codul care trebuie să modifice comportamentul părintelui se inserează înainte de apelul *HandleEvent*, în timp ce codul care adaugă sarcini specifice va figura în program după apelul metodei. Există un set predefinit de comenzi TV care impun acțiuni standard: închide fereastră, selectează *view*, părăsește program, etc. Programatorului îi rămîne însă un domeniu larg de definire a propriilor comenzi asociate unor acțiuni proprii aplicației.

Pentru a înțelege și celelalte tipuri de evenimente să urmărim cum sînt trimise evenimentele într-o aplicație. "Trimiterea" unui eveniment către un obiect înseamnă apelul metodei *HandleEvent* a acestuia, avînd ca argument evenimentul respectiv. Dacă obiectul receptor este la rîndul lui un grup el poate transmite evenimentul mai departe, invocînd metodele *HandleEvent* ale subobiectelor deținute. Modul de dispecerizare diferențiază evenimentele "poziționale" de cele "focusate" și de "emișiuni".

Evenimente poziționale sînt (dacă programatorul nu decide altfel) numai evenimentele de mouse. Un eveniment pozițional este transmis obiectului modal (întotdeauna cel puțin unul într-o aplicație). Obiectul modal caută între *subview*-urile sale pe cel vizibil care conține poziția respectivă și îi transmite evenimentul. Dacă *subview*-ul este la rîndul său un grup procesul continuă pînă cînd va fi găsit un obiect terminal sau poziția respectivă nu mai este conținută într-un *subview*. Ultimul obiect găsit va trata evenimentul.

Evenimentele "focusate" sînt în general evenimente de tastatură sau comenzi ce sînt transmise în jos, de-a lungul lanțului de obiecte selectate pînă la un obiect terminal. Să considerăm cazul unui editor de texte scris cu TV care este compus dintr-un obiect aplicație

care conține pe lângă meniu și o linie status și un suport, în care se pot deschide și închide una sau mai multe ferestre de editare, fiecare conținând două bare de desfășurare, un cadru și un interior - editorul propriu-zis. Apăsarea unei taste generează un eveniment de tastatură care circulă de la obiectul modal curent - aplicația - în jos pe lanțul obiectelor selectate: suport (*TDesktop* este totdeauna obiectul selectat în *TApplication*), fereastra activă, editorul interior ferestrei (singurul obiect selectabil în fereastră). Evenimentul va sfârși prin a fi tratat de ultimul obiect selectat - editorul interior ferestrei.

Și totuși cum reacționează meniul aplicației la taste active de meniu, sau cum sesizează bara de desfășurare din fereastră tastele de desfășurare (*PgUp*, *PgDn*), ambele fiind obiecte neselectabile în grupurile de care aparțin? Există două excepții de la regula procesării evenimentelor focusate, bazate pe un flag de pre/postprocesare al *view*-ului în cauză, care determină trei faze ale parcursului:

- - evenimentul este transmis tuturor *subview*-urilor din grup care au *flag*-ul de preprocesare setat

(în ordinea inversă inserării lor); este cazul meniurilor;

- - dacă evenimentul nu a fost anulat în timpul preprocesării el parvine obiectului selectat din grup;
- - dacă evenimentul este în continuare valid el va fi transmis de obiectul grup tuturor *subview*-urilor având *flag*-ul de postprocesare setat (în ordinea inversă inserării lor în grup); astfel parvin evenimentele de tastatură barelor de desfășurare.

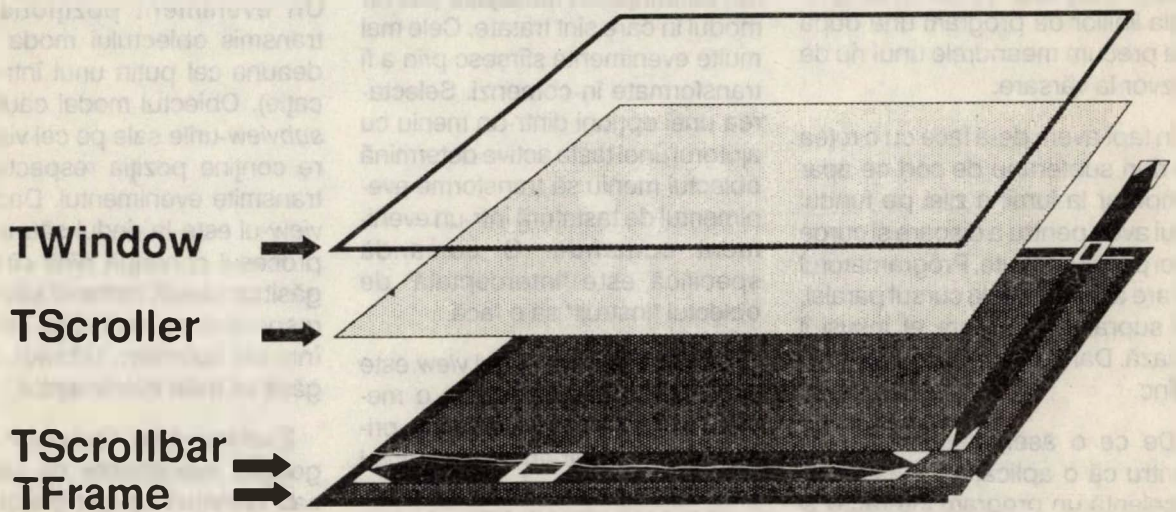
S-ar părea că tehnica aceasta rezolvă problema concurenței pentru un eveniment între obiectele conținute pe același nivel al unui grup. Iată însă cazul unei boxe de dialog care conține o linie de introducere date (*TInputLine*) și un buton etichetat "OK". Pentru ca butonul să poată fi selectat cu combinația de taste **Alt-O** chiar și atunci când linia de editare este obiectul curent selectat, el trebuie să primească primul evenimentele focusate, în faza de preprocesare. Întrucât butonul poate fi "apăsător" și cu tasta **O**, acest eveniment trebuie tratat în faza de postprocesare, pentru a preveni imposibilitatea de a mai introduce caractere **O** în linia de editare. Cum recunoaște

butonul faza în care se află? Fiecare grup conține un câmp numit *Phase* având trei valori posibile, corespunzătoare fazei de procesare curente. Prin verificarea valorii câmpului *Phase* al obiectului posesor, *TButton* poate afla în ce fază i s-a "pasat" evenimentul.

Un eveniment emisiune (*broadcast event*) nu-și cunoaște destinația și este ca urmare trimis tuturor *subview*-urilor din obiectul modal curent. Emisiunea se propagă în jos pe arborele de *subview*-uri pînă la atingerea obiectelor terminale. Emisiunile au trebuit inventate pentru a facilita comunicația între obiecte, dezinventat greu de realizat prin mijloace clasice în programarea OOP.

Un prim exemplu: barele de desfășurare comunică listei noua poziție a elementului focusat prin intermediul unui mesaj semnalizînd schimbarea, lansat spre grupul posesor, care-l va emite spre toate obiectele conținute, dintre care numai lista va "ști" să-l interpreteze.

Cum se poate afla că un anumit obiect există și nu mai este necesară instanțierea lui? Lansarea unui mesaj se face cu funcția



Fereastra unui editor de texte grupează patru obiecte: editorul propriu-zis, totdeauna activ (selectat) în grupul ferestrei, cadrul exterior al ferestrei și două bare de desfășurare care se suprapun parțial cu cadrul.

Message după care controlul revine rutinei "emițătoare" împreună cu adresa obiectului ce a interceptat mesajul sau un pointer nil în cazul în care obiectul căutat nu există. Semnalul universal al interceptării și tratării unui eveniment este transformarea lui într-un eveniment nul. Dacă se dorește ca un mesaj de genul "Stingerea!" să ajungă la toate obiectele de pe suport cerându-le "dispariția", este suficient ca fiecare obiect să evite anularea evenimentului după tratare, permițându-i să călătorească mai departe.

Programatorul poate defini cu ușurință noi tipuri de evenimente, cum ar fi cele asociate apariției unor caractere la portul serial. Implicit, TV le va trata ca emisiuni dar pot fi redefinite ca evenimente poziționale sau focus etc.

Puțini programatori începători în Turbo Vision rezistă tentației de a specula *TProgram.Idle*, o spectaculoasă găselniță a arhitecturii de tratare a evenimentelor. Această metodă primește controlul ori de câte ori nu sînt evenimente disponibile și permite înălțarea unui proces paralel cu cel din *view*-ul curent. TV ne oferă două asemenea gadget-uri, un ceas și un contor de memorie liberă. Deși *Idle* nu oferă un multitasking veritabil, procesul din *view*-ul curent avînd prioritate absolută, totuși în general calculatorul lenevește în așteptarea evenimentelor iar procesul secundar poate primi mai mult de 90% din timpul de execuție

Ceea ce nu se vede

Dincolo de obiectele vizibile și de traiectoriile evenimentelor, Turbo Vision păstrează un set de scule profesionale; stăpînirea lor nu este obligatorie pentru crearea unei aplicații TV, dar este o etapă inevitabilă pentru cei decizi să optimizeze codul pe linia OOP. Curente, colecțiile și resursele sînt toate tipuri derivate din *TObject*.

Colecțiile reprezintă alternativa TV la structurile de date clasice, de tipul vectorilor. *TCollection* este o listă înlănțuită de pointeri, realocabilă dinamic și avînd un set de metode foarte rapide și flexibile pentru operațiuni standard de iterare, inserare, căutare. Un descendent - *TSortedCollection* - întreține colecții sortate după o cheie de index programabilă. Remarcabil este însă polimorfismul colecțiilor: elementele pot fi obiecte diferite și de dimensiuni variabile; un strămoș abstract comun le conferă una sau mai multe metode virtuale ce permit exploatarea lor iterativă.

Încapsularea datelor cu codul în OOP a dus imediat la apariția noțiunii de curent (*stream*) ca răspuns la întrebarea: de ce nu s-ar salva și obiectele pe disc? În tehnicile tradiționale trebuie scrise seturi distincte de funcții pentru tratarea operațiunilor I/O pentru tipuri diferite de date. Un curent TV permite salvarea și citirea de obiecte derivate din *TObject*. Cheia recunoașterii diverselor tipuri de obiecte dintr-un curent este înregistrarea prealabilă a tipului de obiect (tipurile standard TV sînt preînregistrate), utilizînd procedura *RegisterType*. *TStream* este un tip abstract de curent furnizînd un set de metode standard (*Get*, *Put*, *Reset*, *Error*, *CopyFrom*, *ReadStr*, *WriteStr*) și șapte metode virtuale (*GetPos*, *GetSize*, *Seek*, *Truncate*, *Read*, *Write* și *Flush*) definite în descendenți specializați: *TDosStream* - curent de date stocate în fișiere DOS, *TBufStream* - versiune de *TDosStream* controlînd o memorie cache și *TEMSStream* - curent de date stocate în memoria expandată.

O remarcabilă noutate în programare, preluată și de Turbo Vision, o constituie noțiunea de resursă, un nume comun pentru elementele de program modificabile prin înlocuirea valorilor cîmpurilor din obiecte, incluzînd meniuri, boxe de dialog, liste de string-uri, etc. Resursele sînt în mod deliberat definite în afara codului într-o manieră relativ independentă de cod,

astfel încît aplicațiile pot deveni independente de limbă și chiar de alfabet, cu minimum de modificări în cod. Resursele pot fi stocate în fișierul EXE, dar în afara codului executabil.

Programarea fără creare de resurse este de neimaginat în Turbo Pascal for Windows, dublat de un kit de editare interactivă a resurselor grafice. În absența unui asemenea kit pentru TV, se pot scrie deocamdată programe auxiliare de generare și întreținere a resurselor. Spre deosebire de alte porțiuni din Turbo Vision, nu este necesară redefinirea mecanismului resurselor, foarte robust și flexibil.

Ușa din față

Turbo Vision se dovedește a fi destul de revoluționar pentru a pune serioase probleme de adaptare programatorului clasic în Turbo Pascal. Neplăcuta surpriză de a nu putea depana pas cu pas o aplicație demonstrativă din kit-ul *TVDemos* poate descuraja un programator nerăbdător.

Cu regretul tipic românului sărac trebuie să spunem începătorilor în OOP că referința originală Turbo Pascal 6.0 și documentația Turbo Vision sînt cărți absolut necesare. Nu este altă cale decît prin ușa din față. Ghidul TV este excelent scris și rapid asimilabil, mai ales dacă urmați pașii propuși de autori. De parte de a fi exhaustiv în ceea ce privește posibilitățile bibliotecii, el lasă loc la plăcute surprize ulterioare.

Cei experimentați în OOP pot găsi și intrarea din spate, *help*-ul integrat compilatorului și *demo*-urile fiind excelente. O referință a bibliotecii TV rămîne însă indispensabilă, așa că, în limita spațiului disponibil, vom încerca să vă oferim și o astfel de referință.

(ing. Flaidăr Alin)

Pe urmele soluției

Programarea logică cu Turbo-Prolog, Partea a doua

După introducerea în principiile de bază ale programării logice din episodul anterior, de data aceasta vom examina tehnica de programare în Turbo-Prolog, prin intermediul unui exemplu.

O semnificație deosebită pentru Prolog o are operatorul relațional **"not"**. Predicatul standard **"not"** este satisfăcut atunci când predicatul conținut eșuează. De exemplu **"not (5 > 7)"** va furniza rezultatul **"True"**, deoarece 5 nu este mai mare decât 7.

Totuși **"not"** nu poate fi privit exact ca o negație matematică care inversează valoarea de adevăr. Prolog nu atribuie direct un **"not"** unui **"goal"** ci încearcă mai întâi să verifice predicatul conținut. Dacă verificarea eșuează atunci predicatul **"not"** este satisfăcut.

Strategia decizională se bazează pe faptul că există ca dată o bază de cunoștințe închisă, adică pe faptul că sistemului Prolog îi sînt puse la dispoziție toate informațiile necesare. Dacă nu se întîmplă așa se poate întîmpla ca un goal să nu poată fi demonstrat și deci să fie realizat non goal-ul corespunzător.

Exemplu:

```
Domains
  persoana = symbol
predicates
  masculin (persoana)
  feminin (persoana)
clauses
  masculin (ion)
  feminin (X):- not (masculin (X)).
```

Dacă sistemului de mai sus i se pune întrebarea:

Goal: feminin (petru)

sistemul Prolog va răspunde cu YES, cu toate că **"petru"** nu este conținut în baza de cunoștințe. Tre-

buie remarcat și faptul că în cazul lui **"not"** nu au voie să fie utilizate variabile care nu au fost legate încă de o valoare.

```
domains
  persoana = symbol
predicates
  sportiv (persoana)
  face_sport (persoana)
  fumeaza (persoana)
clauses
  face_sport (cristi).
  face_sport (gelu).
  fumeaza (gelu).
  sportiv (X):-not (fumeaza (X)),
  face_sport (X).
```

Dacă după acest exemplu se pune întrebarea **"sportiv (X)"** sistemul va răspunde pentru **"not"** sau **"retractal"** cu mesajul de eroare **"704 Free variables are not allowed"**. Pentru a înlătura acest neajuns vor trebui operate modificările:

```
sportiv (X):- face_sport (X),
  not (fumeaza (X)).
```

Programul modificat devine executabil deoarece variabila X este legată de o valoare înainte de apelul predicatului **"not"**.

Pentru controlul execuției unui program Turbo-Prolog pune la dispoziție predicatele standard **"cut"** și **"fail"**.

Cu ajutorul lui **cut** (reprezentat printr-un semn de exclamare **"!"**) în anumite condiții poate fi limitată procedura de backtracking. Acest predicat este satisfăcut întotdeauna. Pentru aceasta trebuie luate în considerație două reguli:

- 1) atunci când operatorul **cut** este poziționat la sfîrșitul unei clauze, atunci se va obține o singură soluție. Backtracking-ul, și cu aceasta căutarea și a unor alte soluții, este întrerupt(ă).
- 2) dacă operatorul **cut** apare în interiorul unui predicat, atunci va fi executat backtracking-ul numai în spatele acestuia.

Următorul program exemplifică modul de utilizare al operatorului **cut**.

```
predicates
  numar (integer)
  combinatie (integer, integer, integer,
  integer)
clauses
  numar (1).
  numar (2).
  numar (3).
  numar (4).
  numar (5).
  combinatie (A,B,C,D):- numar (A),
  numar (B),
  A<B,
  !,
  numar (C),
  numar (D),
  <C, A<D,
  B<C, B<D,
  C<D.
```

Dacă pentru acest exemplu se interoghează:

Goal: "combinatie (A,B,C,D)"

se obțin următoarele soluții:

```
A=1, B=2, C=3, D=4
A=1, B=2, C=3, D=5
A=1, B=2, C=4, D=5
3 Solutions
```

După o primă instanțiere a variabilelor A și B, compilatorul mai caută combinații doar pentru variabilele C și D. Dacă se înlătură operatorul **cut**, se obțin soluțiile:

```
A=1, B=2, C=3, D=4
A=1, B=2, C=3, D=5
A=1, B=2, C=4, D=5
A=1, B=3, C=4, D=5
A=2, B=3, C=4, D=5
5 Solutions
```

În acest caz sînt găsite două soluții suplimentare deoarece backtracking-ul nu mai este limitat de operatorul **cut**.

Predicatul predefinit **"fail"** nu poate fi satisfăcut niciodată. Cu **fail** se forțează deci un backtracking. Acest lucru are însemnătate în cazul unui goal intern, când se doresc toate soluțiile posibile. Exemplu:

```
predicates
  numar (integer)
  combinatie (integer, integer, integer)
goal
  combinatie (X,Y,Z).
```

clauses

```

numar (1).
numar (2).
numar (3).
comblnatie (X,Y,Z):-
  numar (X), numar (Y), numar (Z),
  X<>Y, Y<>Z, X<>Z,
  write (X,Y,Z), nl,
  fail.

```

Prin utilizarea lui fail se vor obține toate combinațiile posibile formate din cifre de la 1 la 3.

Puterea limbajului Prolog stă în prelucrarea nenumerică a datelor. Totuși pot fi efectuate și calcule matematice, asemănător modului în care se poate programa acest lucru cu limbajele convenționale Pascal sau C. Pe lângă operatorii aritmetici de bază (+, -, *, /, mod, div) și operatorii aritmetici relaționali (=, <, <=, >, >= și <>) mai există o serie întreagă de funcții și predicate predefinite care pot fi utilizate pentru soluționarea problemelor numerice.

Funcții aritmetice:

```

sin (X), cos (X), tan (X), arctan (X),
log (X), ln (X), exp (X),
abs (X), sqrt (X), random (X), round (X),
bitand (X,Y,Z), bitor (X,Y,Z), bitxor (X,Y,Z),
bitnot (X, Y), bitleft (X,N,Z), bitright (X,N,Z)

```

Expresiile aritmetice, în Turbo-Prolog, se scriu cu ajutorul semnelor de egalitate, de exemplu: "X = 25 + 7" va da soluția "true" pentru X = 32.

Trebuie avută grijă deoarece, spre deosebire de Pascal sau C, în Prolog expresia poate fi interpretată ca o alocare de valoare. Dacă X are deja o valoare diferită de 32 predicatul care conține această expresie eșuează. Dacă în acel moment variabila X este încă liberă, atunci va fi instanțiată cu valoarea 32. Această alocare de valori poate determina în anumite situații ca procesul de backtracking să mai facă un pas înapoi. Pentru a face, de exemplu, alocarea "X = Y - Z", variabila Y trebuie să fi fost deja instanțiată, altfel sistemul Prolog va afișa un mesaj de eroare specific.

În cazul unei expresii comparative, ca de exemplu "X <= Y - 2", ambele variabile trebuie să fi fost instanțiate.

Exemplul următor prezintă un scurt program aritmetic de soluționare a unor ecuații de gradul doi:

```

predicates
  solutii (real, real, real)
clauses
  solutii (A,B,C):-
    B*B-4*A*C < 0,
    write ("Nu exista solutii"), nl.
  solutii (A,B,C):-
    B*B-4*A*C = 0,
    X = -B/(2*A),
    write ("Solutia: ",X), nl.
  solutii (A,B,C):-
    B*B-4*A*C > 0,
    X1 = (-B + sqrt (B*B-4*A*C))/2/A,
    X2 = (-B - sqrt (B*B-4*A*C))/2/A,
    write ("Solutii: ", X1, ", ", X2) nl.

```

Programul rezolvă ecuații de gradul doi de forma:

$$A * X * X + B * X + C = 0.$$

Pentru aceasta el trebuie apelat cu goal-ul: "solutii (A,B,C)", unde A,B,C vor fi coeficienții ecuației. Cu "solutii (1,-2,1)" se vor obține soluțiile ecuației "X * X - 2 * X + 1 = 0". De predicatele de ieșire "write" și "nl" ne vom ocupa mai târziu.

O listă este o înșiruire de obiecte ce constituie o unitate. Spre deosebire de cîmpurile din alte limbaje de programare, în Prolog listele sînt structuri dinamice. Aceasta înseamnă că în momentul inițializării încă nu trebuie cunoscut numărul de elemente din care va fi formată lista.

Cu ajutorul listelor pot fi obținute și tipuri de date mai complexe cum sînt arborii și grafurile. În Prolog listele sînt delimitate de paranteze drepte, elementele unei liste fiind separate prin virgulă. Exemplu: [0,1,2,3,4]. În Turbo-Prolog, elementele unei liste trebuie precedate de o instrucțiune DOMAINS care să le specifice tipul. Exemplu: "lista_de_intregi = integer".

O listă constă din antet și din corpul listei. Antetul este primul element din listă, iar corpul este format din restul listei. Exemplu:

Lista	Antet	Corp
[1,3,5,7]	1	[3,5,7]
[1]	1	[]
[[1,3,5],[7]]	[1,3,5]	[[7]]

O listă vidă este scrisă ca []. Antetul și corpul ei nu sînt definite.

În Prolog antetul și corpul unei liste pot fi separate cu ajutorul simbolului "|". Exemplu:

```

Daca: [Antet | Corp] = [0,1,2,3,4,5]
atunci Antet = 0
      Corp = [1,2,3,4,5]

```

Prin această separare a antetului de corpul unei liste pot fi accesate toate elementele listei. Următorul program exemplifică modul în care pot fi obținute elementele unei liste:

```

domains
  lista_de_numere = integer *
predicates
  afisare_lista (lista_de_numere).
clauses
  afisare_lista ([]):- nl.
  afisare_lista ([Antet | Corp]):-
    write (Antet, " "),
    afisare_lista (corp).

```

Pentru execuție programul trebuie apelat, de exemplu, cu goal-ul "afisare_lista ([6,4,78,36,23,9])."

Se vorbește despre recursivitate dacă o procedură se apelează pe ea însăși. Acest lucru se întâmplă în Prolog atunci cînd antetul unei reguli apare și în corpul regulii. Ca exemplu de apel recursiv vom da sursa unui numărător simplu:

```

predicates
  fara_sfirsit (integer)
goal
  fara_sfirsit (0).
clauses
  fara_sfirsit (X):- write (X, " "),
    X1 = X+1,!,
    fara_sfirsit (X1).

```

Prin introducerea unei instrucțiuni de întrerupere, din bucla fără sfîrșit se obține un numărător pînă la o valoare dată:

```

predicates
  fara_sfirsit (integer)
goal
  fara_sfirsit (0).
clauses
  fara_sfirsit (X):- write (X, " "),
    X1 = X+1,
    X1 < 50,!,
    fara_sfirsit (X1).

```

Apelul va fi repetat pînă cînd X1 va lua valoare 50 și condiția "X1 < 50" nu va mai fi satisfăcută ceea ce va duce la întreruperea procesului recursiv.

Recursivitatea este folosită și la definirea listelor. Calculul factorialilor este un exemplu standard. Pro-

gramul următor calculează valoarea unui factorial cu ajutorul recursivității.

```
predicates
  factorial (integer, real)
  calculează_factorial
goal
  calculează_factorial.
clauses
  factorial (0,1).
  factorial (1,1).
  factorial (N, Rezultat):-
    N > 1,
    N1 = N-1,
    factorial (N1, RezultatN1),
    Rezultat = N * RezultatN1.
  calculeaza_factorial:-
    write ("Factorial de "),
    readint (X),
    factorial (X,Y),
    write ("Rezultat: ",Y), nl.
```

Turbo-Prolog dispune de o serie de predicate pentru citirea de date de la tastatură și respectiv pentru afișarea datelor. În sistemul Prolog pot fi utilizate pentru intrări următoarele predicate:

- **readln (Rind)**, Citirea unei variabile de tipul șir sau de tipul simbol.
- **readint (Integer)**, Citirea unei variabile întregi.
- **readreal (Real)**, Citirea unei variabile de tipul real.
- **readchar (Char)**, Citirea unui singur caracter.
- **inkey (Char)**, Citește un caracter, dacă în fișierul de intrare există un caracter, altfel predicatul eșuează.
- **keypressed**, Cu acest predicat se poate stabili dacă o tastă a fost apăsată, fără să se citească caracterul tastat. keypressed reușește atunci când o tastă oarecare este apăsată.

Pentru afișarea unor informații poate fi folosit predicatul "write" cu următoarea sintaxă de apel:

- write (Term 1, Term 2, Term 3, ...)

Numărul argumentelor predicatului "write" este oarecare. Cu el pot fi afișate constante sau variabile care au fost legate de o valoare.

Predicatul "nl" determină un avans de linie. În interiorul predicatului "write" pot fi utilizate următoarele comenzi cu backslash pentru controlul ieșirii:

- /n avans de linie
- /t tabulator
- /z afișează caracterul cu valoarea ASCII z
- // afișează un backslash

Cu ajutorul predicatului "writef" se pot afișa ieșiri formate. Sintaxa predicatului "writef" este următoarea:

- writef (Format, Term 1, Term 2, ...).

Formatul trebuie specificat în modul următor: "%-m.pfeg", unde parametrii individuali au semnificațiile:

- ieșire aliniată la stînga, dacă liniuța lipsește mesajul va fi aliniat la dreapta
- m lățime minimă cîmp, nu este obligatoriu
- p precizia un număr în virgulă fixă sau lungimea maximă a unui șir
- f reprezentarea numerelor reale cu punct (standard)
- e reprezentarea numerelor reale cu virgulă
- g utilizează cel mai scurt format

Modul de utilizare al predicatelor de intrare/ieșire este exemplificat în programul următor:

```
predicates
  introducere_date (integer, real, string)
  afisare_date (integer, real, string)
  repeatuntilheypressed
  integerread (integer)
  realread (real)
  stringread (string)
goal
  introducere_date (X,Y,Z).
  afisare_date (X,Y,Z).
clauses
  repeatuntilheypressed:-
    write ("Pentru continuare apasa orice tasta..."),
    readchar (Ch), Ch=Ch, nl.
  integerread (X):-
    write ("Numar intreg: "), readint (X).
  realread (X):-
    write ("Numar real: "), readreal (X).
  stringread (X):-
    write ("Sir: "), readln (X).
  introducere_date (X,Y,Z):-
    integerread (X),
    realread (Y),
    stringread (Z),
    repeatuntilkeypressed.
  afisare_date (X,Y,Z):-
    writef ("aliniat stînga: %-10./n", X),
    writef ("aliniat dreapta: % 10./n", X),
    writef ("reprezentare cu punct: %-10.5f/n", Y),
    writef ("reprezentare cu virgula: %-19.5e/n", Y),
    writef ("aliniat dreapta: %-20.10/n", Z),
```

writef ("aliniat stînga: %-20.10/n", Z).

Pe lîngă operatorii logici "și" și "sau"

- A & B (conjunție)
- A V B (disjunție)

pot fi folosiți și alți operatori ca de exemplu:

- A nor B (sau exclusiv)
- A <=> B (echivalență)
- A ==> B (implicație)

Valorile de adevărat sau fals (t=true, f=false) se dau cel mai adesea ca valori de adevăr.

Vă vom prezenta în continuare tabela de adevăr pentru operatorii logici și combinațiile lor:

A	t	t	f
B	t	f	t
A & B	t	f	f
A V B	t	t	t
A ==> B	t	t	t
A <== B	t	t	t
A <=> B	t	f	f
A	f	f	t
B	f	t	t
A nand B	f	t	t
A nor B	f	f	f
(A ==> B)	f	t	f
(A <== B)	f	f	f
A xor B	f	t	f
Contradiția	f	t	f

Operatorii anteriori pot fi reprezentanți și prin combinații de "?" și "v". Implicația "(A ==> B)" poate fi scrisă și ca "(A v B)". Operațiile anterioare pot fi date în Prolog și ca fapte dacă se iau valorile corespunzătoare din tabela de adevăr:

```
xor (0,0,0). /* sau exclusiv */
xor (0,1,1).
xor (1,0,1).
xor (1,1,0).
and (0,0,0). /* și */
and (0,1,0).
and (1,0,0).
and (1,1,1).
```

Există o mulțime de legi ale algebrei booleene cu ajutorul cărora expresiile booleene pot fi reordonate pentru stabilirea valorii de adevăr a termenilor.

- A & B <==> B & A Legile comutativității
- A V B <==> B V A
- A & (B V C) <==> (A & B) V (A & C) Legile distributivității
- A V (B & C) <==> (A V B) & (A V C)
- (A & B) <=> (A) V (B) Legile lui de Morgan
- (A V B) <=> (A) & (B)
- A V A <==> A Legile de idempotență

Clase, obiecte și moșteniri

Nici o filozofie de programare nu a stîrnit atîta interes în ultimul timp ca programarea orientată obiect. După cursul introductiv pentru limbajul Pascal, vă prezentăm acum continuarea pentru cei interesați de C++.

Transpunerea acestei noi filozofii în soft nu se oprește în fața nici unui limbaj de programare. Ce se află îndărătul concepției orientate obiect? Doar și pînă acum, oricînd, datele de prelucrat puteau fi abstractizate într-atît încît să poată fi gestionate eficient! Și totuși, cine descoperă noile extensii de limbaj, e nevoit să-și schimbe părerile.

Cine vorbește de C++, se referă la un C orientat pe clase, deoarece, în acest limbaj, obiectele sînt denumite clase. Analogia cu biologia vine de la sine: și aici se vorbește de clase, subclase, specii, etc. Această analogie vine în ajutorul începătorilor, permițînd o acomodare mai rapidă cu noțiunile neobișnuite de care, inevitabil, se lovesc.

Clasele servesc perfecționării modelelor. Modelele biologice existente sînt supuse - chiar și după ani și ani de cercetare - unor modificări sau unor extensii.

În limbajele de programare uzuale, pentru date cu caracteristici identice, dar de tipuri diferite, trebuiesc create funcții diferite. Adunarea numerelor întregi, reale sau complexe se deosebește prin modul de predare al tipului de date. Conform ANSI C, parametrii de ape! trebuie să corespundă celor din funcția apelată. Funcțiile primesc nume diferite, chiar dacă principial fac același lucru, doar cu date de alt tip. În C nu există altă posibilitate, decît a produce trei funcții, cu nume diferite.

Cauza pentru întărirea controlului asupra tipului datelor este posibila transmitere - generatoare de erori - a unor date de alt tip decît cele așteptate către o funcție. Este exact ce se evită prin încapsularea datelor și a codului aferent lor. Funcții-element care fac același lu-

cru (de exemplu, adună) pot avea același nume. În C++ este chiar indicat să se folosească această extensie - denumită, uzual "supraîncărcare" (overload). Este, în principiu, la latitudinea programatorului, în ce măsură folosește avantajele programării orientate obiect. Este important să se înțeleagă și să se folosească aceste avantaje. Efectul va fi un control mai bun asupra structurii programelor și modularitate. În C++ pot fi generate tipuri de date noi, cu operatori speciali, care se potrivește numai lor.

Caracteristicile programării orientate obiect sînt sintetizate în noțiunile de încapsulare, ereditate și polimorfie. Ce înseamnă acestea, mai în amănunt, se poate vedea din fig. 2. Unii experți sînt de părere că, pentru a învăța C++, trebuie uitat tot ce s-a învățat mai înainte (ceea ce ei, cu siguranță, nu au făcut...). Parțial au dreptate, parțial nu. Apropierea de C++ se face cel mai bine adăugînd la cunoștințele de bază despre C extensiile de limbaj orientate obiect, la fel cum s-ar face acest lucru în Pascal.

După aceste considerații preliminare, prima încercare OOP. Este vorba de adunarea descrisă, în cazul unor date de tipuri diferite.

Programul din listingul ADD.CPP produce următoarele mesaje pe ecran:

Prima sumă este 5.000

A doua sumă este 7.264

A treia sumă este 2.001.

La prima adunare, funcției ADD îi sînt date două valori întregi. Înaintea celei de-a doua adunări, sînt definite două numere în virgulă mobilă. Această adunare poate fi executată numai cu cea de-a doua funcție ADD. La versiuni mai vechi de C++, funcția supraîncărcată trebuia declarată cu cuvîntul cheie prefixat "overload" Acesta a fost declarat depășit între timp.

Cele două funcții ADD per-

mit revelaarea unei alte caracteristici importante a C++. Parametrii funcției li se pot atașa valori inițiale. Dacă înaintea celei de-a treia adunări al doilea termen al adunării n-ar fi fost inițializat cu zero, rezultatul ar fi fost greșit. Interesant este că funcții supraîncărcate pot fi folosite și independent de clase speciale. Dar să nu anticipăm.

Cum se definesc clasele? C++ dispune pentru aceasta de cuvîntul cheie "class". Dar și vechile cuvinte cheie "struct" respectiv "union" pot fi folosite în acest scop. După unul din aceste trei cuvinte cheie urmează denumirea clasei ce se definește. Exemplu:

```
class student {
    char facultate [30];
    int semestru;
};
```

Pentru clasa "student" au fost stabilite atributele (data members) "facultate" și "semestru". Majoritatea claselor sînt derivate pornind de la clase de bază. Să presupu-

```
1 | // exemple de functii supraincarcate
2 | // adunare intregi sau . in virgula mobilă
3 |
4 | #include <stdio.h>
5 |
6 | int add(int add1 = 0, int add2 = 0)
7 | {
8 |     return (add1 + add2);
9 | }
10 |
11 | //overload
12 | float add (float add1 = 0, float add2 = 0)
13 | {
14 |     return (add1 + add2);
15 | }
16 |
17 | int main()
18 | {
19 |     float x, y, z;
20 |
21 |     z = add (2, 3);
22 |     printf ("\n Prima suma este: %6.3f",z);
23 |     x = 2.001; y = 5.263; z = add (x,y);
24 |     printf ("\n A doua suma este: %6.3f",z);
25 |     z = add(x);
26 |     printf ("\n A treia suma este: %6.3f",z);
27 | }
```

ADD.CPP

```

1 // Listing exemplu nr 1
2 // un obiect cu functii-element
3
4 #include <conio.h>
5 #include <stdio.h>
6
7 class Semn //definitia clasei Semn
8
9 int X, Y; //coordonate
10 char valoare; //val. alfanumerica
11 char attr; //atribut ecran
12
13 public:
14 Semn (int InitX, int InitY)
15 {
16     X = InitX;
17     Y = InitY;
18     valoare = ' ';
19
20 //Functii-element publice
21
22 int GetX () { return X; }
23 int GetY () { return Y; }
24 void Position (int PosX, int PosY)
25 {
26     X = PosX; Y = PosY;
27 }
28 void Citire ()
29 {
30     getch(X, Y, X, Y, &valoare);
31 }
32 void Scriere (char Sign)
33 {
34     valoare = Sign;
35     puttext (X, Y, X, Y, &valoare);
36 }
37
38 int main ()
39 {
40     int X1, Y1;
41
42     clrscr ();
43     X1 = 4; Y1 = 10;
44     Semn ASemn (X1, Y1);
45     ASemn.Citire ();
46     ASemn.Scriere ('A');
47     printf ("\n Apasati o tasta ... ");
48     getch ();
49     return 0;
50 }

```

EX1.CPP

```

1 // Listing exemplu nr. 2
2 // alte cteva functii-element
3
4 #include <conio.h>
5 #include <stdio.h>
6
7 class Semn //definitia clasei Semn
8
9 int X, Y; //coordonate
10 char valoare; //val. alfanumerica
11 char attr; //atribut ecran
12

```

```

13 public:
14 Semn (int InitX, int InitY)
15 {
16     X = InitX;
17     Y = InitY;
18     valoare = ' ';
19 }
20
21 //Functii-element publice
22
23 int GetX () { return X; }
24 int GetY () { return Y; }
25 void Position (int PosX, int PosY)
26 {
27     X = PosX; Y = PosY;
28 }
29 void Citire ()
30 {
31     getch (X, Y, X, Y, &valoare);
32 }
33 void Scriere (char Sign)
34 {
35     valoare = Sign;
36     puttext (X, Y, X, Y, &valoare);
37 }
38 void Indicare ()
39 {
40     puttext (X, Y, X, Y, &valoare);
41 }
42 void Stergere ()
43 {
44     char memo;
45
46     memo = valoare; valoare = ' ';
47     Indicare ();
48     valoare = memo;
49 }
50 void Miscare (int NewX, int NewY)
51 {
52     Stergere();
53     Position (NewX, NewY);
54     Indicare ();
55 }
56 }
57
58 int main ()
59 {
60     int X1, Y1;
61
62     clrscr ();
63     X1 = 4; Y1 = 10;
64     Semn ASemn (X1, Y1);
65     ASemn.Citire ();
66     ASemn.Scriere ('A');
67
68     printf ("\n Apasati o tasta ... ");
69     getch ();
70     ASemn.Miscare (X1 + 4, Y1 - 4);
71     getch ();
72     ASemn.Scriere ('B');
73     getch ();
74     ASemn.Stergere ();
75     getch (); return 0;
76 }

```

EX2.CPP

nem că există o clasă de bază "persoană" cu următoarele atribute:

```

class persoana {
    char nume [30];
    char prenume [20];
    int virsta;
}

```

Atunci ar fi posibilă derivarea clasei "student" din clasa "persoana". Pentru aceasta, ar trebui modificată definiția clasei "student":

```

class student:
    persoana {
        char facultate [30];
        int semestru;
    };

```

Clasa "student" moștenește toate caracteristicile (atributele) clasei "persoana", reprezintă însă o clasă derivată.

Să precizăm că modul de acces la caracteristicile individuale poate fi definit altfel în cele două clase.

După aceste considerații a devenit limpede cum se definește o clasă simplă. Forma generală este:

```

<classkey>
<classname>
< : baselist > {
    <member list>
}

```

Dar ce este o clasă fără metode care să facă posibilă preluarea și modificarea atributelor sale? Eficiența OOP se obține, în principal, tocmai prin moștenirea de către clasele derivate a metodelor proprii clasei "mamă".

Exemplul conține definiția clasei "Semn". Clasa dispune de 4 caracteristici: coordonatele x și y pe ecran, un caracter și un atribut al acestuia. Valorile sînt păstrate în memoria de împrăștiere a imaginii.

Urmează declararea funcțiilor-element. Atunci cînd descrierea funcției -element urmează imediat definiției ei, se vorbește de funcții incorporate (inline). "GetX ()" este descrisă complet în interiorul clasei "Semn", între delimitatorii } și {, prin intermediul lui return X; funcții de acest fel, pot fi precedate de cuvîntul cheie "inline":

```

inline int GetX ()
{ return X;
}

```

Dacă funcția este definită în afara clasei, atunci în descrierea defi-

niției trebuie să se facă referire la clasa căreia funcția îi aparține:

```
int Semn:: GetX ()
{ return X;
};
```

În listingul reprodus există metodele necesare preluării coordonatelor precum și pentru citirea și scrierea caracterelor în mod text. La citire, se preia și atributul ecran existent.

Se poate urmări și încapsularea datelor împreună cu codul ce permite manipularea lor.

În funcțiile "Citire ()" și "Scriere ()" se folosește operatorul "&" pentru referințe. Acesta se folosește asemănător cu un pointer. La transmiterea argumentelor nu se transmite valoarea lor, ci adresa de referință la valoare.

Referințele sînt pointeri spre obiecte, putînd fi utilizate (oarecum) în locul acestora.

Definiția unei clase reprezintă declararea unui tip de date specific utilizatorului, iar obiectul este o variabilă de acest tip. În exemplu, se declară variabila "ASemn" de tip "Semn". Acest obiect este static, deoarece spațiul de care are nevoie i se alocă încă din timpul compilării. Obiectelor dinamice li se alocă memoria necesară pe stivă, în momentul execuției - cînd își termină existența, memoria de care au avut nevoie este eliberată. Observații mai în detaliu referitoare la obiecte dinamice vor fi prezentate într-o parte separată a cursului.

În program, se declară obiectul "ASemn" avînd coordonatele $x=4$ și $y=10$. Apoi, se citește caracterul aflat în acel loc, împreună cu atributul lui. Apoi, caracteristicii "valoare" a obiectului i se dă valoarea "A", afișarea făcîndu-se păstrînd vechiul atribut. Pentru afișare, se folosește metoda "Scriere" aplicată obiectului "ASemn".

În exemplul 2, clasei "Semn" i-au fost adăugate alte cîteva funcții-element. Scopul a fost mișcarea obiectului în alt loc pe ecran. Pentru aceasta a trebuit să fie posibilă ștergerea din poziția veche, transmiterea noilor coordonate și afișarea obiectului în noua poziție. Metoda "Indicare ()" se deosebește de "Scriere (char Sign)" prin faptul că nu se transmite o valoare nouă. În acest exemplu, "Ștergere

()" semnifică afișarea unui spațiu în punctul de coordonate x, y .

Obiectului "ASemn" i se predă mesajul "Mișcare ($x1+4, y1-4$)". Urmează și alte mesaje, simplu de înțeles. Mesajele denumite în acest fel trebuie să ajute utilizatorul să înțeleagă sensul programului. Prin OOP, "transparența" aplicației poate fi îmbunătățită.

Mai interesant devine Exemplul 3. Nici un obiect nu există numai pentru sine. Exemplul vrea să ilustreze interacțiunea dintre obiecte pe un caz simplu.

Obiectul "BSemn" este copiat pe tot ecranul, astfel încît fondul apare ca o plasă. Obiectul "ASemn" ur-

```
1 // Listing exemplu nr. 3
2 // extind. caract. obiectului
3 // ptr. prez. interact. intre obiecte
4
5 #include <conio.h>
6 #include <stdio.h>
7
8 class Semn // definitia clasei Semn
9 {
10     int X, Y; // coordonate
11     char valoare; // val. alfanumerica
12     char attr; // atribut ecran
13     char valoare_veche;
14     char attr_vechi;
15
16     public:
17     Semn (int InitX, int InitY)
18     {
19         X = InitX;
20         Y = InitY;
21     }
22
23     // Functii-element publice
24
25     int GetX () { return X; }
26     int GetY () { return Y; }
27     void Position (int PosX, int PosY)
28     {
29         X = PosX; Y = PosY;
30     }
31     void Citire ()
32     {
33         getch (X, Y, X, Y, &valoare_veche);
34     }
35     void Scriere (char Sign)
36     {
37         Citire ();
38         valoare = Sign;
39         puttext (X, Y, X, Y, &valoare);
40     }
41     void Modificare (char Sign)
42     {
43         valoare = Sign;
44         puttext (X, Y, X, Y, &valoare);
```

```
45     }
46     void Indicare ()
47     {
48         puttext (X, Y, X, Y, &valoare);
49     }
50     void Stergere ()
51     {
52         valoare = valoare_veche;
53         attr = attr_vechi;
54         Indicare ();
55     }
56     void Miscare (int NewX, int NewY)
57     {
58         char memo_valoare;
59         char memo_attr;
60
61         memo_valoare = valoare;
62         valoare = valoare_veche;
63         memo_attr = attr;
64         attr = attr_vechi;
65         Indicare ();
66         valoare = memo_valoare;
67         attr = memo_attr;
68         Position (NewX, NewY);
69         Citire ();
70         Indicare ();
71     }
72     void Copiere (int NewX, int NewY)
73     {
74         Position (NewX, NewY);
75         Indicare ();
76     }
77     void Seteaza_attr (char attr_nou)
78     {
79         attr = attr_nou;
80         Indicare ();
81     }
82 };
83
84 int main ()
85 {
86     int X1, Y1, s, z;
87
88     X1 = 1; Y1 = 1;
89     Semn BSemn (X1, Y1);
90     BSemn.Scriere (' ');
91     BSemn.Seteaza_attr (BLUE);
92     for (s = 0; s < 25; s++)
93         for (z = 0; z < 80; z++)
94             BSemn.Copiere (X1 + z, Y1 + s);
95     X1 = 4; Y1 = 10;
96     Semn ASemn (X1, Y1);
97     ASemn.Scriere ('A');
98     ASemn.Seteaza_attr (WHITE);
99     printf ("\n Apasati o tasta ... ");
100     getch ();
101     ASemn.Miscare (X1 + 4, Y1 - 4);
102     getch ();
103     ASemn.Modificare ('B');
104     getch ();
105     ASemn.Stergere ();
106     getch (); clrscr ();
107     return 0;
108 }
```

EX3.CPP

```

1 // Listing exemplu nr. 4
2 // def. a doua obiecte diferite
3 // unul fiind derivat din celalalt
4 // Semn --> Semn2
5
6 #include <conio.h>
7 #include <stdio.h>
8
9 class Semn //definitia clasei Semn
10 {
11 public:
12     int X, Y; // coordonate
13     char valoare; // val. alfanumerica
14     char attr; // atribut ecran
15
16     Semn (int InitX, int InitY)
17     {
18         X = InitX;
19         Y = InitY;
20     }
21
22 // Functii-element publice
23
24     int GetX () { return X; }
25     int GetY () { return Y; }
26     void Position (int PosX, int PosY)
27     {
28         X = PosX; Y = PosY;
29     }
30     void Citire ()
31     {
32         getch (X, Y, X, Y, &valoare);
33     }
34     void Scriere (char Sign)
35     {
36         Citire ();
37         valoare = Sign;
38         puttext (X, Y, X, Y, &valoare);
39     }
40     void Modificare (char Sign)
41     {
42         valoare = Sign;
43         puttext (X, Y, X, Y, &valoare);
44     }
45     void Indicare ()
46     {
47         puttext (X, Y, X, Y, &valoare);
48     }
49     void Stergere ()
50     {
51         valoare = ' ';
52         Indicare ();
53     }
54     void Miscare (int NewX, int NewY)
55     {
56         char memo_valoare;
57
58         memo_valoare = valoare;
59         Stergere ();
60         valoare = memo_valoare;
61         Position (NewX, NewY);
62         Indicare ();
63     }
64     void Copiere (int NewX, int NewY)
65     {
66         Position (NewX, NewY);

```

```

67     Indicare ();
68     }
69     void Seteaza_attr (char attr_nou)
70     {
71         attr = attr_nou;
72         Indicare ();
73     }
74 };
75
76 // definitia clasei Semn2
77
78 class Semn2 : public Semn
79 {
80     char valoare_veche;
81     char attr_vechi;
82
83 public:
84     Semn2 (int InitX, int InitY) :
85         Semn (InitX, InitY)
86     {
87         valoare_veche = 0x0;
88         attr_vechi = 0x0;
89     }
90 // Functii-element publice
91
92     void Citire ()
93     {
94         getch (X, Y, X, Y, &valoare_veche);
95     }
96     void Scriere (char Sign)
97     {
98         Citire ();
99         valoare = Sign;
100        puttext (X, Y, X, Y, &valoare);
101    }
102    void Modificare (char Sign)
103    {
104        valoare = Sign;
105        puttext (X, Y, X, Y, &valoare);
106    }
107    void Indicare ()
108    {
109        puttext (X, Y, X, Y, &valoare);
110    }
111    void Stergere ()
112    {
113        valoare = valoare_veche;
114        attr = attr_vechi;
115        Indicare ();
116    }
117    void Miscare (int NewX, int NewY)
118    {
119        char memo_valoare;
120        char memo_attr;
121
122        memo_valoare = valoare;
123        valoare = valoare_veche;
124        memo_attr = attr;
125        attr = attr_vechi;
126        Indicare ();
127        valoare = memo_valoare;
128        attr = memo_attr;
129        Position (NewX, NewY);
130        Citire ();
131        Indicare ();

```

```

130     }
131     void Copiere (int NewX, int NewY)
132     {
133         Position (NewX, NewY);
134         Indicare ();
135     }
136     void Seteaza_attr (char attr_nou)
137     {
138         attr = attr_nou;
139         Indicare ();
140     }
141 };
142
143 int main ()
144 {
145     int X1, Y1, s, z;
146
147     X1 = 1; Y1 = 1;
148     Semn BSemn (X1, Y1);
149     BSemn.Scriere ('+');
150     BSemn.Seteaza_attr (BLUE);
151     for (s = 0; s < 25; s++)
152         for (z = 0; z < 80; z++)
153             BSemn.Copiere (X1+z, Y1+s);
154     X1 = 4; Y1 = 10;
155     Semn2 ASemn (X1, Y1);
156     ASemn.Scriere ('A');
157     ASemn.Seteaza_attr (WHITE);
158     printf ("\n Apasati o tasta ... ");
159     getch ();
160     ASemn.Miscare (X1+4, Y1-4);
161     getch ();
162     ASemn.Modificare ('B');
163     getch ();
164     ASemn.Stergere ();
165     getch (); clrscr ();
166     return 0;
167 }

```

EX4.CPP

mează să citească și să memoreze informația ecran anterioară, pentru a șterge sau mișca să se vadă din nou fundalul (plasa să nu se găurească). Pentru aceasta, sînt introduse pur și simplu două noi atribute pentru obiectul "Semn". Mai trebuie în plus modificate funcțiile-element "Scriere ()" și "Ștergere ()". Înainte de scriere, trebuie reținute valoarea veche și atributul existent. Ștergerea în cadrul acestei clase modificate înseamnă acum afișarea valorii originale. Dacă obiectul "ASemn" urmează să primească o altă valoare în aceeași poziție, acest lucru nu are voie să se întîmple prin metoda "Scriere ()", deoarece în acest fel s-ar strica valoarea memorată inițial în "valoarea veche". De aceea s-a introdus o altă metodă, care permite modificarea valorii existente în aceeași

```

1 // Listing exemplu nr. 5
2 // functii-elemente mostenite
3
4 #include < conio.h >
5 #include < stdio.h >
6
7 class Semn // definitia clasei Semn
8 {
9     int X, Y; // coordonate
10 public:
11     char valoare; // val. alfanumerica
12     char attr; // atribut ecran
13
14     Semn (int InitX, int InitY)
15     {
16         X = InitX;
17         Y = InitY;
18     }
19
20 // Functii-elemente publice
21
22     int Semn::GetX () { return X; }
23     int Semn::GetY () { return Y; }
24     void Semn::
25         Position (int PosX, int PosY)
26     {
27         X = PosX; Y = PosY;
28     }
29     void Semn::Citire ()
30     {
31         gettext (X, Y, X, Y, &valoare);
32     }
33     void Semn::Scriere (char Sign)
34     {
35         Citire ();
36         valoare = Sign;
37         puttext (X, Y, X, Y, &valoare);
38     }
39     void Semn::Modificare (char Sign)
40     {
41         valoare = Sign;
42         puttext (X, Y, X, Y, &valoare);
43     }
44     void Semn::Indicare ()
45     {
46         puttext (X, Y, X, Y, &valoare);
47     }
48     void Semn::Stergere ()
49     {
50         valoare = ' ';
51         Indicare ();
52     }
53     void Semn::
54         Miscare (int NewX, int NewY)
55     {
56         char memo_valoare;
57
58         memo_valoare = valoare;
59         Stergere ();
60         valoare = memo_valoare;
61         Position (NewX, NewY);
62         Indicare ();
63     }
64     void Semn::
65         Copiere :(int NewX, int NewY)

```

```

63     {
64         Position (NewX, NewY);
65         Indicare ();
66     }
67     void Semn::
68         Seteaza_attr (char attr_nou)
69     {
70         attr = attr_nou;
71         Indicare ();
72     }
73
74 // definitia clasei Semn2
75
76 class Semn2 : public Semn
77 {
78     protected:
79     char valoare_veche;
80     char attr_vechi;
81
82     public:
83     Semn2 (int InitX, int InitY) :
84         Semn (InitX, InitY)
85     {
86         valoare_veche = 0x0;
87         attr_vechi = 0x0;
88     }
89
90 // Functii-elemente publice
91
92     void Semn2::Citire ()
93     {
94         gettext (GetX(), GetY(), GetX (),
95             GetY(),&valoare_veche);
96     }
97     void Semn2::Scriere (char Sign)
98     {
99         Semn2::Citire ();
100        Semn::Scriere (Sign);
101    }
102    void Semn2::Stergere()
103    {
104        valoare = valoare_veche;
105        attr = attr_vechi;
106        Indicare ();
107    }
108    void Semn2::
109        Miscare (int NewX, int NewY)
110    {
111        char memo_valoare;
112        char memo_attr;
113
114        memo_valoare = valoare;
115        valoare = valoare_veche;
116        memo_attr = attr; attr = attr_vechi;
117        Indicare ();
118        valoare = memo_valoare;
119        attr = memo_attr;
120        Position (NewX, NewY);
121        Citire ();
122        Indicare ();
123    }
124
125 int main ()

```

```

123 {
124     int X1, Y1, s, z;
125
126     X1 = 1; Y1 = 1;
127     BSemn.BSemn (X1, Y1);
128     BSemn.Scriere ('+');
129     BSemn.Seteaza_attr (BLUE);
130     for (s = 0; s < 25; s++)
131         for (z = 0; z < 80; z++)
132             BSemn.Copiere (X1 + z, Y1 + s);
133     X1 = 4; Y1 = 10;
134     Semn2 ASemn (X1, Y1);
135     ASemn.Scriere ('A');
136     ASemn.Seteaza_attr (WHITE);
137     printf ("\n Apasati o tasta ... ");
138     getch ();
139     ASemn.Miscare (X1 + 4, Y1 - 4);
140     getch ();
141     ASemn.Modificare ('B');
142     getch ();
143     ASemn.Stergere ();
144     getch (); clrscr ();
145     return 0;
146 }

```

EX5.CPP

poziție. Această metodă este "Modificare ()". Totodată, cele două obiecte urmează să se deosebească și prin atributele ecran. Acest lucru se întâmplă prin intermediul metodei "Setează_attr ()".

De fapt, stricto sensu, în acest exemplu sînt două obiecte. Unul din ele acoperă fundalul, fără a necesita informație despre ce a fost înainte. Celălalt obiect este o clasă derivată, extinsă cu caracteristici de "istorie". Acest lucru determină modificarea necesară a metodelor.

Următorul listing arată cum ar fi dotate două clase diferite cu toate metodele necesare într-o manieră uzuală de programare. De remarcat că toate atributele lui "Semn" au fost declarate publice, astfel încît ele pot fi accesate și din clasele derivate. Acest exemplu vrea numai să evidențieze diferența față de limbajele de programare condiționale.

Procedeeul uzual însemna că la tipuri de date noi trebuia adaptat întotdeauna codul vechi, chiar dacă modificarea tipului datelor nu se răsfrîngea direct asupra anumitor părți ale programului. Ceea a generat, deseori, surprinzător de multe erori. Scopul ar trebui să fie deci testarea serioasă a codului necesar, urmînd ca acest cod să fie dat mai departe claselor derivate, care moștenesc caracteristicile.

```

1 // o incercare in modul grafic
2 // def.: in fisier separat: first.h
3 // fctiile-element sint in first.cpp
4
5 #include <graphics.h>
6 #include <stdio.h>
7 #include <conio.h>
8
9 #include "first.h"
10
11 // Fctiile-element ale cl "Position"
12
13 Position::
14 Position (int InitX, int InitY)
15 {
16     X = InitX;
17     Y = InitY;
18 };
19
20 int Position:: GetX(void)
21 {
22     return X;
23 };
24
25 int Position:: GetY(void)
26 {
27     return Y;
28 };
29
30 // Fctiile-element ale clasei "Semn"
31 // Sistemul grafic a fost initializat
32 // de catre programul principal
33
34 Semn:: Semn(int InitX,
35 int InitY): Position (InitX, InitY) {
36     moveto (InitX, InitY);
37 }
38
39 void Semn::Scrie(char *valoare) {
40     moveto(getx(), gety());
41     outtext (valoare);
42 };
43
44 int main()
45 {
46     int midx, midy;
47     int gdriver = DETECT, gmode;
48     char *text;
49
50     initgraph(&gdriver, &gmode,
51 "\\BORLANDC\\BGI\\");
52
53     midx = getmaxx() / 2;
54     midy = getmaxy() / 2;
55
56     Semn APos (midx, midy);
57     text = "Hello, world\0";
58     APos.Scrie(text);
59
60     getch();
61     closegraph();
62 }

```

FIRST.CPP

```

1 // first.h contine doua clase:
2 // "Position" da pozitia, ( X si Y)
3 // "Semn" = caracter in Position
4
5 // Functiile element sint in first.cpp
6
7 class Position {
8     protected:
9     // cl. deriv. au acces la date "private"
10
11     int X;
12     int Y;
13
14     public:
15     // Functii ce pot fi apelate din ext.
16
17     Position (int InitX, int InitY);
18     int GetX();
19     int GetY();
20 };
21
22 class Semn : public Position {
23     // derivata din clasa "Position"
24     // in int. lui "Semn", X, Y protejate
25
26     protected:
27     // Pentru a permite accesul
28     // functiilor derivate din "Semn"
29     char *value;
30
31     public:
32     Semn (int InitX, int InitY);
33     void Scrie (char *value);
34 };

```

FIRST.H

numai aceasta trebuie ajustată pe măsura tocmai acestei clase.

Clasa "Semn2" posedă funcțiile element modificate "Citire ()", "Scriere ()", "Ștergere ()" și "Mișcare ()". De remarcat în acest exemplu că în metoda Scriere a clasei Semn2 au fost combinate "Citire ()" din clasa Semn2 cu "Scriere" din clasa Semn. În partea a doua a cursului, acest exemplu va fi mai reluat odată, reducând sursa programului.

Toate exemplele prezentate au în comun faptul că funcțiile lor element sînt funcții inline. Urmează un exemplu în care în header sînt prezentate numai definițiile de clase cu metodele aferente, în timp ce descrierea metodelor urmează să se facă în altă parte. Ultimul exemplu, Hello.CPP, afișează un șir de caractere în mod grafic. Important este nu conținutul exemplului, ci forma: faptul că în fișierul header sînt prezentate numai definiții.

Tip obiect: Un tip obiect este un tip de date definit de utilizator. În el sînt definite structurile de date și metodele de prelucrare corespunzătoare.

Obiect: Un obiect este declarat și manipulat ca și o variabilă. Compilatorul rezervă deci pentru un obiect spațiul de memorie necesar, care este accesibil prin intermediul numelui obiectului.

Metodă: Pentru a putea lucra cu datele declarate în interiorul unui obiect, definiția lui trebuie să conțină și declarațiile funcțiilor și procedurilor corespunzătoare lor. Tocmai acestea sînt metodele.

Sarcină: După stabilirea unui obiect, numele unei metode este utilizat pentru a-i transmite o sarcină acestuia.

Protocol: Totalitatea sarcinilor accesibile unui tip obiect este desemnată ca protocol.

Încapsulare: Cuprinderea împreună a structurilor de date și a metodelor de prelucrare se numește încapsulare. Definirea obiectelor în cadrul unui Unit întărește încapsularea.

Moștenire: Tipurile obiect își pot transmite toate însușirile unor noi obiecte. Această modalitate de moștenire elimină necesitatea de a redefini metodele deja definite. Trebuie declarat doar ceea ce s-a schimbat față de obiectul "mamă" sau care sînt datele și metodele care trebuie adăugate.

Ierarhie de obiecte: Deoarece tipurile obiect își pot moșteni însușirile, se poate construi o ierarhie arborescentă a tuturor urmașilor. Toate tipurile obiect ale acestei ierarhii sînt legate prin "legături de rudenie". De datele și metodele obiectului rădăcină pot dispune toți urmașii.

Dacă o metodă nu este satisfăcătoare pentru o clasă, atunci

(I.F.)

Novell anunță:

Netware Lite v1.0

Firma Novell, gigantul rețelor de calculatoare (în primul rând personale) lansează pe piață versiunea sa de sistem de operare de rețea de tipul "peer to peer".

Aceasta oferă unor utilizatori lipsiți de pretenții și mai puțin dispuși să facă investiții substanțiale, posibilitatea de a partaja resursele (aplicații, fișiere de date, imprimante, mijloace de comunicație etc.) de care dispun. Respectivele resurse se află distribuite pe calculatoare pe care rulează sistemul de operare DOS.

Fiecare din calculatoarele cuplate în rețea poate fi utilizat ca server, client sau ambele. Calculatorul server va partaja resursele sale cu calculatoarele client din rețea; un server poate partaja maxim 25 de resurse cu celelalte calculatoare.

Produsul oferă utilizatorului posibilitatea de a executa rapid operațiuni specifice rețelei cum ar fi accesul la un disc partajat sau la serviciile unei imprimante. Interfața cu utilizatorul se prezintă sub forma unui ecran care cuprinde un sistem de meniuri.

Accesul la resursele partajate ale rețelei este controlat prin intermediul unei scheme de protecție

simple.

Pentru fiecare utilizator se creează un singur cont la nivelul întregii rețele; în felul acesta accesul nu se face la nivelul unei resurse anume ci la totalitatea resurselor din rețea. Gestiunea resurselor este ușurată de faptul că ea se realizează de la oricare din calculatoarele din rețea.

Accesul utilizatorului se efectuează la resursă și nu la calculatorul care o oferă la un moment dat, apărând posibilitatea deplasării resursei de pe un calculator pe altul fără ca prin aceasta drepturile de acces ale utilizatorului la resursa respectivă să fie redefinite.

Conceptual, acest aspect, combinat cu cel al accesului la rețea și nu la calculatoarele din rețea, reprezintă o apropiere de noțiunea de prelucrare distribuită.

Toate programele aferente se găsesc pe o singură dischetă de 1,2M. Procesul de instalare este foarte rapid și nu necesită cunoștințe aprofundate.

Partea de hardware pentru rețea este furnizată de o companie legată de Novell încă din 1984, anume Eagle Technology, o diviziune a lui Anthem Electronics (aceasta din urmă cotate la bursa

din New York și creditată cu un venit de \$408 milioane în anul 1990).

Produsul se comercializează sub două forme:

un kit inițial: NetWare Lite Ethernet Starter Kit, ce cuprinde două plăci interfață rețea Ethernet, cablurile și conectorii aferente, sistemul de operare NetWare Lite pentru două PC-uri (deci două dischete) precum și documentația aferentă;

un kit pentru un nod cuprinzând sistemul de operare, placa de rețea cu cablul și conectorii aferente, precum și documentația respectivă.

Prin achiziția starter kit-ului se realizează o rețea de două calculatoare (noduri) după care acestea i se adaugă câte un nod, prin achiziția celui alt kit.

Plăcile de rețea oferite sînt NE2000 (pentru arhitecturi ISA) și NE/2 (pentru arhitecturi Micro-Channel). Aspectul este extrem de important deoarece dacă peste un timp utilizatorul respectiv dorește să treacă la o rețea mai sofisticată (NetWare v2.2 sau NetWare v3.11) plăcile de rețea ca și cablurile vor putea fi folosite fără probleme.

De altfel, NetWare Lite coexistă cu NetWare v2.2 și v3.11. Produsul (hard + soft) poate fi comandat de la firma SCOP S.R.L., str. Polonă 86, 71154 București, tel. 11.74.21, fax 11.73.74.

Firma SCOP S.R.L. este distribuitor Novell pentru România. Sînt binevenite și solicitări din partea firmelor care doresc să devină remarketeri pentru orice alte produse Novell.

Prețul sistemului de operare este de \$99/nod, la care se adaugă costul plăcii de rețea, al cablului și al conectorilor aferente.

Marius Sturzoiu, SCOP S.R.L.





NOVELL INTERNATIONAL AUTHORISED REMARKETER

Vă propune integrarea în sistemul autorizat de distribuție în România. Veți beneficia astfel în mod oficial de:

- programul de upgrade al firmei NOVELL
- autorizarea firmei dvs. ca NOVELL AUTHORISED RESELLER
- suportul tehnic și de marketing necesar.



 **Summagraphics** / HOUSTON INSTRUMENT



AUTHORISED REMARKETER IN ROMANIA

Telefon: 11.74.21 / 11.92.48.

Fax: 11.73.74.

Noi sîntem soluția !

Expoziție

În perioada 8-10 octombrie '91, Facultatea de Electrotehnică din Cluj Napoca a fost gazda expoziției de tehnică de calcul și a "Zilelor profesionale SzKI" organizate de societatea comercială COMPREX import-export S.R.L. din Cluj Napoca și de SzKI (Institutul de Cercetări și Centrul de Inovații pentru Tehnică de Calcul Budapesta). Au fost prezentate mai multe configurații de calculatoare personale (Laptop-uri, AT-uri, 386), la prețuri rezonabile (un AT standard 189.000), soluții de interconectare, demonstrații software.

Firma mixtă româno-germano-maghiară COMPREX, distribuitor autorizat ARL, va comercializa, printre altele, renumitele calculatoare produse de această firmă și pachetele software Recognita (nr. 3 mondial la categoria software OCR, produs comercializat în 16 state), Ventura Publisher (versiune românească), Pro-C (generator de programe sursă C).

Firma a anunțat și un program de amnistie - în schimbul copiilor ilegale de Ventura Publisher, în următoarele 6 luni, pot fi obținute copii legale ale acestui produs la cca. 100.000 lei.

Firma COMPREX poate fi contactată la unul din telefoanele: 95-117623, 95-115145, sau prin fax: 95-112474.

Tib '91

Multicolore impresiile de la acest târg internațional, organizat parcă sub așteptări. Dacă este sau nu exclusiv meritul mineriadei, este discutabil; cert este că o susținută activitate de instalare/amenajare - încă în a treia zi de târg! - este o notă proastă. Tot lipsuri organizatorice sînt cauza faptului că o listă a firmelor ce expun este atît de greu de obținut încît e preferabil să te lași păgubaș. De liste tematice - produse chimice, industria lemnului sau calculatoare - să nu mai vorbim.

Indiferente la ce povestește televiziunea pe la telejurnal, destule standuri s-au umplut abia pe parcurs - și nu numai datorită faptului că unele firme au fost anunțate prin telex despre standul "obținut", trecîndu-se peste "amănuntul" că nu au telex...

Totuși, faptul că vă putem prezenta alăturat lista celor circa 80 de firme pe care le-am reperat la târg denotă o ofensivă continuă a tehnicii de calcul. Am încercat să ne limităm destul de strict la specificul revistei, trecînd (prea repede și cu o undă de regret) pe lîngă destul de multe firme ce foloseau PC-uri înglobate în aplicații industriale sau în aparatură de laborator.

Multe firme comercializează PC-uri, de la XT-uri ieftine-ieftine (ODE-

ROM) pînă la AT-uri "supergrele" cu 486, SVGA, harddisk de 180 Mbyte sau chiar mai mult. De la cei care oferă produse de un anumit tip (IBM, Tandon, KT, Laser) și pînă la angroșiști de genul KSL, ce caută dealeri pentru produsele înșirate într-un catalog-listă de prețuri de 116 pagini tipărite mărunț, mai toate felurile posibile de comerț sînt reprezentate. Dacă System Plus ține la "nume" (Dell, Iversson, Summagraphics), SCP se mîndrește cu cele mai accesibile prețuri. Se profilează cîteva "Computerland"-uri, în care să se poată cumpăra totul dintr-o dată, de la dischete și mouse-uri pînă la sisteme complete, cu pachete de soft și documentații: MBL Computerland, SCP, ABTech, Circle Systems International, ADISAN, Tokyo Electronics etc. Surprinde plăcut Infocomp, care oferă și elemente componente: cipuri de memorie, coprocesoare, integrate diverse.

În ce privește softul, produsele Borland, vîndute sub amnistie de Logic la prețuri incredibil de avantajoase, stau alături de produsele pentru a căror vînzare este nevoie de autorizații speciale, cum ar fi AutoCad 11.0 (ce poate fi cumpărat cu reducere de 70% de la A&C cu condiția să fie folosit numai pentru învățămînt și nu în proiectare!). Rețelele Novell, prezente ca un laitmotiv, sînt "atacate" de Digital, care oferă condiții avantajoase tuturor posesorilor de I100

sau Coral care, conștienți de limitele PC-urilor, optează pentru mașini DEC și VMS sau Unix. Unix-ul vine încet, dar sigur (Teleinvest, Bull), iar concepte revoluționare precum Open System Architecture pot fi urmărite "pe viu" la MDS GmbH, unde - ca într-un joc cu cuburi - plecînd de la o configurație minimă, se pot construi configurații extrem de complexe adăugînd module hard și soft noi, de o manieră cuceritor de simplă.

Periferia se diversifică: pot fi cumpărate imprimante rapide (RCD, FW-20, 2000 linii/minut) sau color (AREXIM), discuri de pînă la 300 Mb (RCD), CD-ROM-uri (Alfa-Bit S.R.L. 600 Mbytes, 150.000 lei), cartele fax sau modem, plotter (HP în primul rînd) etc. Poate cea mai tentantă ofertă: Top9+ vinde benzi magnetice Qualstar, împreună cu softul care permite transferul de date de pe benzi Felix sau mini.

Și în materie de comunicații au apărut noutăți: Motorola oferă aparatură profesională pentru telecomunicații, iar Cedru S.A. anunță o legătură ISDN cu Germania, via satelit, în colaborare cu ER&P, reprezentanți unici Eurodata pentru Europa de Est.

Devine grea viața cumpărătorului de tehnică de calcul: în orice direcție se răstește la Sesam, ușa se deschide...

Iosif Fettich

Firma	Observații	Fax	Telefon
A&C International	Autodesk - Authorised Dealer	535328	535315
ABTech			104273
ADISAN			158165
AREXIM	imprimante		
ARIS Group		595270	595270
ARC Brașov S.R.L.		951/50022	951/50022

ARTIMEX S.R.L.			
ATC - A-Tronic Computer		102428	106196
ATI S.R.L.		959/12425	959/16899
Alfabit S.R.L.	CD-ROM com. radio	114973	144639
Axaware Publishing	DTP		
Baneasa S.A.	fost IPRS	11203	334050
Baza de instruire informatic	cursuri la Vatra Dornei		988/73534
Blue Ridge International			
BroDeCo	se retrage din calculatoare	111895	112263
Bull	Unix, Zenith		(331)45029090
C.P. Info S.A.		133840	816125
Cedru S.A.	ISDN, Eurodata	433102	435954
Ciel Roumanie	financiar - contabil		
Circle Systems International	soc. romano-americana	143681	143001
DATATIM S.A.		961/33418	961/30078
Delta Design		136040	154580/2404
Digital		69874275	69875700
EDCG		212644	143594
East Europ Trading Ltd.			794194
Electron S.R.L.			
Electronum		120438	120439
Eurotrade			
Eurotrade Company			
FEPER			876714
Finro	în colaborare cu ICI	127556	336999
Goldstar			
HAMEG			
Hewlett Packard		120294	697080
IBM			143929
ICE + HITEC	Unix	450253	450253
ICE Felix Computers S.A.		887820	886030
IIRUC Enter	firmă romano-elena		
IIRUC S.A. MDS GmbH	OSA, SQL, Posix, C.TOS, Unix		882070/249
IMC Data Systems	soc. romano-olandeza		
IMSAT		376101	376550
INFA S.A.	contabilitate, INFA NET	127501	127501
ITC + ATC			
ITC + AMT	imprimante AMT	797770	797770
Infcon S.R.L.			
Info C.P.	fost C.C. al M.I.Et.		
InfoStar			921/14143
Infocomp	si componente diverse	157795	132697
Innovator			334097
Intercom Leu S.R.L.	tehnica de birou	120317	136209
K.S.L. Europe	en gros; cauta dealer		
KT Technology Romania Ltd.	RCD + Singapore	141022	152316
LECOM	reprez. NEC		506420
Laser + Realsoft			
Logic		757135	754900
M&M Computers			
MBL Computers SRL	Computerland Romania	127568	333227
MDF Europe Ltd.	Pilot Computer Systems	(0483)574217	(0483)65951
Memorii Timisoara			
Mercator			111675
MICROROM	cauta dealer	146114	144309
Motorola			
ODEROM	XT-uri ieftine	142240	151507
RCD			
Robor Chisinau	soc. sovieto-americana	(0422)229321	(0422)222259
RomSoft	fost BNP		652540
Romtrust SRL	imprimante Citizen	334528	336616
Romwest Euro S.R.L.			
SCOP		117374	117421
SCP		753155	757005
SIS			115326
SIX Ltd.	DTP		776595
Shark			396274
SiaSoft S.R.L.	soft spec. ind. alim.	150439	148797
Sistema S.R.L.		795970	795970
Software ITC SA		127612	127611
Sogetec	inf. industrială		
Sysgraph	DTP, Recall Redac		
System PLUS S.R.L.	reprez. IBM, Dell, Iversson...		791391
Tandon Romania S.R.L.		171039	170960
Tecom Multicom	tim real - seră dozoatoare...	752052	387814
Teleinvest	Unix, Progress, 3COM	565678	565678
Tokyo Electronics		654510	657030
Top 9+	labcards; benzi magnetice	107036	102787
Trend S.R.L.	dealer HP	333389	127634

```

1 o| #include <string.h> |o
2 o| |o 61 o| mov cx, 2 |o
3 o| #define L_PASS 10 |o 62 o| mov ah, 3fh |o
4 o| |o 63 o| int 21h |o
5 o| void change_password(char *new_pass); |o 64 o| mov ax, word ptr password //manevra |o
6 o| |o 65 o| // mov ah, byte ptr password+1 |o
7 o| /* password0 - parola initiala |o 66 o| mov l_header, ax |o
8 o| password - parola de lucru |o 67 o| } |o
9 o| newpass - noua parola |o 68 o| |o
10 o| exe_name - numele executabilului */ |o 69 o| /* det. dist. relativa a parolei fata de incep. fis. |o
11 o| char password0[L_PASS+1]="CUCUCUCUCU", |o 70 o| executabil pe disc */ |o
12 o| newpass[L_PASS+1]="testteste", |o 71 o| exe_pos +=16*l_header; |o
13 o| password[L_PASS+1], |o 72 o| |o
14 o| exe_name[50]; |o 73 o| /* modifica parola */ |o
15 o| /*-----*/ |o 74 o| strcpy(password, new_pass); |o
16 o| void main(int argc, char *argv[]) |o 75 o| asm { |o
17 o| { |o 76 o| mov al, 0 //pozitionare in fisier |o
18 o| strcpy(password, password0); |o 77 o| mov bx, file_handle |o
19 o| strcpy(exe_name, argv[0]); |o 78 o| mov cx, word ptr exe_pos+2 //cx:dx poz. in fisier |o
20 o| strcpy(newpass, argv[1]); |o 79 o| mov dx, word ptr exe_pos |o
21 o| printf("Parola curenta este: %s\n", password); |o 80 o| mov ah, 42h |o
22 o| printf("Parola noua va fi: %s\n", newpass); |o 81 o| int 21h |o
23 o| change_password(newpass); |o 82 o| mov ax, SEG password //scriere parola |o
24 o| } |o 83 o| mov ds, ax |o
25 o| /*-----*/ |o 84 o| mov dx, OFFSET password |o
26 o| void change_password(char *new_pass) |o 85 o| mov bx, file_handle |o
27 o| { static unsigned |o 86 o| mov cx, L_PASS |o
28 o| file_handle, psp_seg, pass_seg, pass_off, l_header; |o 87 o| mov ah, 40h |o
29 o| static unsigned long exe_pos; |o 88 o| int 21h |o
30 o| |o 89 o| mov bx, file_handle //Inchidere fisier |o
31 o| /* det. dist. rel. a parolei fata de inceputul programului |o 90 o| mov ah, 3eh |o
32 o| incarcat in memorie. Se tine seama de lung. PSP */ |o 91 o| int 21h |o
33 o| asm { |o 92 o| } |o
34 o| mov ah, 62h |o 93 o| } |o
35 o| int 21h |o 94 o| /*-----*/ |o
36 o| mov word ptr psp_seg, bx //segment PSP |o
37 o| mov pass_seg, SEG password0 |o
38 o| mov pass_off, OFFSET password0 |o
39 o| } |o
40 o| exe_pos=16*(pass_seg-bsp_seg)+pass_off-0x100; |o
41 o| |o
42 o| /* determina marime header fisier pe disc */ |o
43 o| asm { |o
44 o| mov ax, SEG exe_name //deschidere fisier |o
45 o| mov ds, ax |o
46 o| mov dx, OFFSET exe_name |o
47 o| mov al, 2 |o
48 o| mov ah, 3dh |o
49 o| int 21h |o
50 o| mov file_handle, ax |o
51 o| mov al, 0 //pozitionare in fisier |o
52 o| mov hx, file_handle |o
53 o| mov cx, 0 //cx:dx ofs. in fis. a dim. header |o
54 o| mov dx, 8h |o
55 o| mov ah, 42h |o
56 o| int 21h |o
57 o| mov ax, SEG password //cit. marime header fisier |o
58 o| mov ds, ax |o
59 o| mov dx, OFFSET password |o
60 o| mov bx, file_handle |o

```

PAROLVAR.C

Parolă variabilă

Implementarea unei parole în cadrul unui program este un lucru destul de simplu. Este suficientă crearea unei constante și cu o valoare cunoscută numai de anumiți utilizatori și limitarea accesului la execuția programului pe baza comparării unui string introdus de utilizator cu stringul predefinit. Lucrurile se complică atunci când se dorește modificarea parolei într-un mod interactiv, adică evitându-se modificarea fișierului sursă și recompilarea programului. Pentru rezolvarea acestei probleme propunem folosirea funcției `change_password()` prezentată în acest mic articol. Funcția a fost testată în Turbo C, dar autorii vă asigură de portabilitatea sa pentru orice implementare de limbaj ANSI C pe calculatoare compatibile IBM-PC sub sistemul de operare MS-DOS. Prezentăm și cadrul general în care funcția a fost testată.

*George Anescu, Pandeles Stănescu,
I.C.N. Pitești*

1024

Conform promisiunii din numărul 7 al revistei, vom premia - în fiecare număr al revistei - cel mai interesant program mai mic de 1024 octeți (în sursă) pe care îl primim spre publicare. Primul câștigător: dl. Buleu Daniel, str. Simeon Florea Marian nr. 3, 5875 Rădăuți, pentru programul DYNAMIC ASM. Felicitări!



Cite ceva despre tastatura AT

Tastatura AT (și microcontrollerul încorporat) are facilități de programare și este prin aceasta mult mai interesantă decât vechile tastaturi model PC. Informații relativ detaliate relativ la aceste facilități pot fi găsite de ex. în (binecunoscutul) TechHelp (The Electronic Technical Reference Manual, Copyright (c) 1985, 1990 by Flambeaux Software, Inc. Cele două programe de mai jos sînt două exemple de utilizare. Ele trebuisc introduse cu un editor ASCII, asamblate cu un macroasambler și link-editate.

DYNAMIC.EXE se va juca cu LED-urile corespunzătoare tastelor "Lock", transformîndu-le în lumini dinamice.

PAROLA.EXE poate fi utilizat pentru protecția accesului în sistemul Dvs. de calcul. Introducînd o linie cu numele fișierului în AUTOEXEC.BAT veți fi puși în situația de a tasta o parolă pe care nici Dvs. nu o știți! Vă vin în ajutor LED-urile amintite care, în caz că sînt în ordinea NUM, CAPS, SCROLL (de la stînga la dreapta) vă vor da codul binar al următoarei taste de apăsare (*on*=1, *off*=0). Dacă ordinea nu este aceasta, trebuie doar să modificați tabela de conversie "tabc".

La ieșirea din program se reface starea LED-urilor în concordanță cu octetul de stare al tastaturii de la adresa 0000:0417.

*Buleu Daniel,
str. Simeon Florea Marian nr.3,
5875 Rădăuți, jud. Suceava*

```

1  o | data segment          |o
2  c | etc db 0ah, 0dh, '  Press any key to stop ! $' |o
3  o | data ends             |o
4  o |                       |o
5  o | code segment         |o
6  o | assume cs:code, ds:data |o
7  o |                       |o
8  o | DELAL proc           |o
9  o |     mov cx, 2000h    |o
10 o | del2: loop del2     |o
11 o |     ret              |o
12 o | DELAL endp          |o
13 o |                       |o
14 o | DELAY proc           |o
15 o |     mov dl, 40h     |o
16 o | del: call DELAL     |o
    
```

```

17 o |     dec dl          |o
18 o |     jnz del         |o
19 o |     ret              |o
20 o | DELAY endp          |o
21 o |                       |o
22 o | PROG proc           |o
23 o |     mov al, 0edh    |o
24 o |     out 60h, al     |o
25 o |     call DELAL      |o
26 o |     ret              |o
27 o | PROG endp           |o
28 o |                       |o
29 o | start:  mov ax, data |o
30 o |         mov ds, ax  |o
31 o |         mov ax, 0    |o
32 o |         mov cx, ax   |o
33 o |         mov ah, 9    |o
34 o |         mov dx, offset etc |o
35 o |         int 21h      |o
36 o |                       |o
37 o | begin:  call PROG   |o
38 o |         mov al, 2    |o
39 o |         out 60h, al  |o
40 o |         call DELAY  |o
41 o |         call PROG   |o
42 o |         mov al, 1    |o
43 o |         out 60h, al  |o
44 o |         call DELAY  |o
45 o |         call PROG   |o
46 o |         mov al, 4    |o
47 o |         out 60h, al  |o
48 o |         call DELAY  |o
49 o |         mov ah, 0bh  |o
50 o |         int 21h      |o
51 o |         cmp al, 0    |o
52 o |         jz begin     |o
53 o |                       |o
54 o | quit:  mov al, 0edh  |o
55 o |         out 60h, al  |o
56 o |         mov si, 417h |o
57 o |         mov al, es:[si] |o
58 o |         mov cl, 4    |o
59 o |         shr al, cl   |o
60 o |         and al, 7    |o
61 o |         call DELAL  |o
62 o |         out 60h, al  |o
63 o |         mov ah, 04ch |o
64 o |         int 21h      |o
65 o |                       |o
66 o | code ends           |o
67 o | end start           |o
    
```

DYNAMIC.ASM

```

1  o | data segment | o
2  o | tabc db 0, 1, 4, 5, 2, 3, 6, 7 | o
3  o | ecod db 0ah, 0ah, ' Introduceți codul de acces : $' | o
4  o | egr db 0ah, 0ah, 0dh, ' Cod incorect !', 0ah, 0dh | o
5  o | db ' Deconectati sistemul ! $' | o
6  o | eok db 10 dup (0ah), 0dh, ' Ok!', 7, 0ah, 0dh, '$' | o
7  o | loc db 0, 0, 0, 0 | o
8  o | data ends | o
9  o | | o
10 o | code segment | o
11 o | assume cs:code, ds:data | o
12 o | | o
13 o | DELAY proc | o
14 o |     mov cx, 3000h | o
15 o | del2: loop del2 | o
16 o |     ret | o
17 o | DELAY endp | o
18 o | | o
19 o | GENER proc | o
20 o |     mov ah, 0 | o
21 o |     mov bl, 16 | o
22 o |     div bl | o
23 o |     and ah, 7 | o
24 o |     mov loc[si], ah | o
25 o |     inc si | o
26 o |     and al, 7 | o
27 o |     mov loc[si], al | o
28 o |     inc si | o
29 o |     ret | o
30 o | GENER endp | o
31 o | | o
32 o | VERIF proc | o
33 o |     mov al, 0edh | o
34 o |     out 60h, al | o
35 o |     call DELAY | o
36 o |     mov al, loc[si] | o
37 o |     out 60h, al | o
38 o |     mov ah, 0 | o
39 o |     mov dl, ax | o
40 o | et1: mov ah, 7 | o
41 o |     int 21h | o
42 o |     cmp al, 30h | o
43 o |     jb et1 | o
44 o |     cmp al, 39h | o
45 o |     ja et1 | o
46 o |     mov dl, tabc[dl] | o
47 o |     sub al, 30h | o
48 o |     cmp dl, al | o
49 o |     je et2 | o
50 o |     mov ch, 1 | o
51 o | et2: inc si | o
52 o |     mov ah, 2 | o
53 o |     mov dl, ',' | o
54 o |     int 21h | o
55 o |     ret | o
56 o | VERIF endp | o
57 o | | o
58 o | start: mov ax, data | o
59 o |     mov ds, ax | o
60 o |     mov ax, 0 | o
61 o |     mov es, ax | o
62 o | | o
63 o |     mov ah, 2ch | o
64 o |     int 21h | o
65 o |     mov al, dh | o

```

```

66 o |     mov si, 0 | o
67 o |     call GENER | o
68 o |     mov al, dl | o
69 o |     call GENER | o
70 o | | o
71 o |     mov ah, 9 | o
72 o |     mov dx, offset ecod | o
73 o |     int 21h | o
74 o | | o
75 o |     mov ch, 0 | o
76 o |     mov si, 0 | o
77 o | et3: call VERIF | o
78 o |     cmp si, 4 | o
79 o |     jnz et3 | o
80 o |     cmp ch, 0 | o
81 o |     jz et5 | o
82 o |     mov ah, 9 | o
83 o |     mov dx, offset egr | o
84 o |     int 21h | o
85 o | et4: mov ah, 2 | o
86 o |     mov dl, 7 | o
87 o |     int 21h | o
88 o |     call DELAY | o
89 o |     jmp et4 | o
90 o | et5: mov ah, 9 | o
91 o |     mov dx, offset eok | o
92 o |     int 21h | o
93 o | | o
94 o | quit: mov al, 0edh | o
95 o | | out 60h, al | o
96 o |     mov si, 417h | o
97 o |     mov al, es:[si] | o
98 o |     mov cl, 4 | o
99 o |     shr al, cl | o
100 o |     and al, 7 | o
101 o |     call DELAY | o
102 o | | out 60h, al | o
103 o |     mov ah, 04ch | o
104 o |     int 21h | o
105 o | | o
106 o | code ends | o
107 o | end start | o

```

O glumă

Dacă sînteți în situația de a folosi calculatorul cu care lucrați împreună cu alți colegi, este aproape sigur că fiecare are cerințele lui proprii vizavi de acesta. Poate lucrează cu alt driver de tastatură, poate preferă alt mesaj de prezentare la lansare, poate are nevoie de niște TSR-uri care pe Dvs. vă deranjează, ocupînd prea mult loc, etc. Și care este locul ideal de a-și "croi" un sistem după gust, dacă nu AUTOEXEC.BAT? Iar dacă gusturile diferă substanțial, scandalul tinde să se permanentizeze. De aici ideea de a crea un AUTOEXEC.BAT "nemodificabil". Trucul este surprinzător de simplu, dacă dispuneți de un program care să permită editarea unui fișier COM (PCTools, NU, etc.).

Mai întîi, confecționați-vă o dischetă sistem care să nu conțină numai fișierele sistem, ci și alte programe utilitare, cum ar fi FORMAT, RECOVER, CHKDSK, KEYB etc. După care creați un nou fișier AUTOEXEC, căruiu însă e bine pentru început să-i dați alt nume - GLUMA.BAT de ex.

Selectați apoi de pe discheta unul din programele utilitare pe care în mod normal nu îl utilizați - RECOVER.COM de ex. Vă notați ora și data de pe acesta, așa cum le afișează DIR-ul. Acum ștergeți RECOVER.COM, după care redenumiți GLUMA.BAT în RECOVER.COM. Dați acum noului fișier ora și data celui original (cu TOUCH, de ex.). La prima vedere, nimeni nu va putea descoperi că acest fișier este de fapt un fișier de comenzi. Aceste pregătiri sînt premisele pentru trucul propriu

-zis. Intrați acum cu un editor de discheta în COMMND.COM. În MS-DOS v3.3 găsiți în sectorul relativ 11 înregistrarea "AUTOEXEC.BAT". Schimbați acum pur și simplu această valoare în RECOVER.COM. Puteți mări confuzia: ștergeți AUTOEXEC-ul original, după care dați comanda

copy recover.com autoexec.bat

Aveți acum un nou AUTOEXEC.BAT și, dacă mai corecți și data și ora, totul arată aproape ca la început. Diferența față de discheta originală este aproape imperceptibilă.

Colegului Dvs. care își face de lucru la PC, mesajul de eroare îi va produce o sperietură serioasă. Dacă i se pare că va putea rezolva problema modificând, ca de atâtea ori, AUTOEXEC.BAT-ul, va avea o surpriză. Căci orice modificări va face - mesajul de eroare se va încapățâna să apară. Mai ales "specialiștii" de circumstanță vor fi îmbrobodiți.

Încă o observație: procedeul funcționează numai în MS-DOS. DR-DOS 5.0 nu se lasă dus: sistemul se blochează pur și simplu.

(I.F.)

Recondiționarea riboanelor pentru imprimantă

Imprimantele PC-urilor au banda tușată montată într-o cutie (cartuș) mobilă sau fixă. După uzare, dacă avem la dispoziție un cartuș nou, îl schimbăm. Dacă nu, îi schimbăm numai banda tușată. Majoritatea cartușelor au banda tușată continuă, cu capetele răsucite la 180 de grade înainte de lipire (banda Mobius). Așadar există trei categorii de benzi tușate, respectiv cartușe în care se montează:

- nerăsucite
- răsucite pe dreapta
- răsucite pe stînga

Prima categorie utilizează, în general, neeficient banda întrucît se imprimă pe o singură pistă (centrală) în timp ce celelalte două folosesc două piste. În multe cazuri banda tușată scoasă dintr-un cartuș care nu folosește banda răsucită poate fi folosită în alt cartuș (pentru altă imprimantă) în care este nevoie de răsucire, și viceversa, bineînțeles cu relipirea corespunzătoare a capetelor.

Chiar banda tușată uzată mai poate fi încă folosită, retușînd-o. Acest lucru presupune pregătirea unui amestec de 2/3 tuș ștampilă și 1/3 glicerină tehnică. Amestecul se pune într-un vas de plastic (butelie de șampon de exemplu) căruia i se atașează la dop un tub umplut cu împîslitură (din interiorul unui creion carioca). Tușarea presupune parcurgerea întregii benzi și "vopsirea" ei cu "pensula" astfel fabricată. O metodă mai rapidă constă în impregnarea unei bucăți de pîslă cu amestec pregătit (tuș + glicerină) și tragerea întregii benzi prin această pîslă (eventual printr-o tăietură practică). În ambele cazuri nu este nevoie ca banda tușată să fie scoasă afară din cartuș. Pentru a evita neplăcerile legate de contactul cu soluția, murdăria de mîinilor, se poate folosi o pereche de mînuși de bucătărie sau chirurgicale. După terminarea operației se introduce banda din nou în cartuș învîrtind de axul antrenor.

Dacă dispunem de bandă nouă, nelipită, o vom introduce în cartuș, fără să-l desfacem, astfel:

- - se trage puțin banda veche afară (cca 20 cm) și se taie la mijloc.
- - se lipește capătul dinspre axul de antrenare al benzii vechi de un capăt al benzii noi (cum se lipește se va vedea mai jos).
- - se trage capătul nelipit al benzii uzate pînă se termină.
- - se învîrtește de axul de antrenare pînă cînd rămîne afară o bucată de cca 20 cm din banda nouă.
- - se trage de banda uzată pînă cînd apare lipitura continuînd tragerea apoi pînă cînd se scoate afară cca 20 cm din banda nouă.
- - se rupe la lipitură și se îndepărtează banda veche.
- - se lipește capetele benzii noi.

Trebuie avut în vedere ca banda nouă să nu fie mai lată decît cea veche întrucît se produc înțepeniri. Înțepeniri se pot produce și dacă banda nouă se înghesuie (nu încape) în cartuș. Acest lucru se poate datora fie faptului că este mai lungă fie faptului că este mai grosă. În ambele cazuri se mai scurtează.

Chiar dacă dispunem de banda lipită și răsucită corespunzător recomandăm dezlipirea ei și utilizarea metodei de mai sus care are avantajul că prelungeste durata de viață a cartușului (desfacerea repetată duce la deteriorări). Este însă nevoie de o anume perfecționare în lipire.

Introducerea în cartuș a unei benzi gata lipite se poate face și prin desfacerea cartușului. Cartușul se desface și dacă se constată că în interiorul lui s-au produs nereguli. Montarea noii benzi într-un cartuș desfăcut se face fără a ne pune problema "plierii" acesteia. Banda se trece printre cele două roți dințate antrenore (dacă roata presoare are arc se împinge puțin în ea), se merge cu ea prin locurile pe unde trebuie să treacă (eventual se răsucește cu 180 de grade în sens corespunzător dacă configurația cartușului o cere) și se scoate afară. Se pune capacul cartușului la loc

```

1  @@echo off
2  rem GLUMA.BAT
3  cls
4  echo Supervizorul sistemului anunta:
5  echo.
6  echo CATASTROFA ! Harddisk distrus !
7  echo.
8  echo Harddiskul se formateaza ACUM
9  echo.
10 echo Apasati o tasta oarecare si cautati-
11 echo URGENT pe dl. Director Popescu !
12 echo E singurul autorizat sa actioneze!
13 echo in aceste circumstante ! URGENT !
14 echo.
15 pause nul

```

GLUMA.BAT

și apoi se trage banda înăuntru prin învîrtirea axului antrenor cu mîna.

În final vă vom prezenta un procedeu eficient de lipire a capetelor tușate, cu minimum de instrumente :

- » 1) un pistol de lipit electric
 - » 2) o lamă uzată
 - » 3) o bucată de geam (10x10cm este suficient) așezată pe o masă. Poate lipsi dacă masa este acoperită cu geam sau melamină.
 - » 4) o riglă metalică sau din lemn (nu neapărat gradată).
- Procesul se desfășoară astfel :
- - se suprapun fețele interioare (cele dinspre cartuș) ale capetelor benzii astfel încît cele două capete să le ținem cu mîna dreaptă iar cartușul cu mîna stîngă:
 - - se așează capetele suprapuse pe geamul de pe masă.
 - - se rotește capătul de deasupra al benzii tușate, în plan orizontal, în sens trigonometric cu cca 120 de grade (nu este esențial unghiul, lipitura va ieși mai mult sau mai puțin înclinată).
 - - se așează rigla peste bandă astfel încît marginea ei să treacă prin punctele de intersecție exterioare ale benzii (A, B):
 - - între punctele A și B se trece vîrfurile încălzite ale pistolului de lipit pînă cînd se taie complet banda, capetele din dreapta îndepărtîndu-se.
 - - va rezulta o lipitură care mai are un anumit "grad" de bavură. Se așează banda desfăcută, cu bavura în sus iar peste bavură se așează lama. Cu pistolul de lipit "se calcă" banda peste lamă (ca și cu fierul de călcat), în lungul bavurii. Va rezulta o lipitură netedă.

(ing. Sava Stan, Buzău)

IMPORTANT !

Începînd cu numărul viitor al revistei, oferim tuturor celor interesați posibilitatea de a face **reclamă gratuită** produselor proprii - soft sau hard. Vom disponibiliza în acest scop una sau două pagini, în care vom publica gratuit textele publicitare care ne vor fi trimise. Singura condiție este ca produsele să fie **elaborate** de cei care-și fac publicitate. Ne gîndim în primul rînd la diversele aplicații, programe, biblioteci etc. care ar merita să fie cunoscute, dar căroră în general autorii lor nu-și pot permite să le facă publicitatea pe care, poate ar merita-o. Nu putem promite că vom respecta toate posibilele dorințe relativ la modul de apariție al acestor reclame, (reclama plătită este cea la care clientul decide cum anume apare) dar putem promite că vom încerca să le publicăm pe toate (chiar dacă va fi "inghesuală").

REDUCERE !!

Oferim tuturor celor interesați posibilitatea de a se abona la revista noastră la un preț mai convenabil: oricine solicită un număr de cel puțin 5 exemplare, va avea o reducere de 10%; expedierea se va face prin ramburs, iar plata expediției o vom suporta noi. Procentul de reducere poate crește pînă la 20%, la cantități ce depășesc 200 de exempl.

Pentru cei care doresc eventual numere mai vechi ale revistei: mai sînt disponibile o parte din revistele editate anul acesta. Pentru numerele 1-5/91, oferim o reducere de 20%, pentru cei care pot ridica revistele de la redacție; pentru cei care le solicită prin ramburs, din bonificație vom scade taxa de expediție

Poșta redacției

"...Aș cumpăra cu plăcere revista dumneavoastră, dacă s-ar găsi și prin chioșcurile din Iași, și dacă ar costa, să zicem, 12 - 15 lei. (Pentru a reduce prețul revistei vă sugerez să folosiți caractere mai mărunte. Pasionații de informatică sînt în majoritate tineri și au vedere bună. Oricum, o lupă costă mai puțin decît două numere de if)."

Cristian Trelea, Iași

Aveți, desigur, dreptate: revista este scumpă. Știm acest lucru, dar, cu toate eforturile noastre, mai ieftin nu reușim să o scoatem. Ne pare foarte rău că am fost nevoiți să mai ajustăm odată prețul revistei, și din păcate în sus. Anunțăm însă pe această cale că **pentru toți abonații, prețul revistei rămîne cel vechi, pe durata pentru care a fost achitat abonamentul** (pentru abonații care au făcut abonament pînă în data de 30.09.91. Cei mai noi sînt rugați să corecteze sumele achitate, dacă sînt în situația de a fi făcut abonamentul în perioada în care a intervenit noua scumpire). Este, aceasta, nu ațit o modalitate de a încerca să ciștigăm abonați noi, cît mai mult o tentativă de a le mulțumi celor vechi. Mai încercăm, desigur, diverse variante, și sîntem recunoscători oricui ne dă idei viabile în acest sens. Relativ la un caracter de literă mai mic: așteptăm eventuale păreri în acest sens și de la alți cititori. Oricum, nu credem că cititul cu lupa poate face plăcere. Și am vrea ca revista să fie citită și cu plăcere.

Mica publicitate

Firma "LEXINA" - S.R.L. vă asigură **service calificat** pentru orice unitate de discuri - MD50, MD40 și în special pentru unitățile MX200. Tel. 90/43.32.97 - București.

Cumpăr documentație pentru Turbo Assembler. Biscoveanu Costel, str. Baba Novac nr. 117, 8700 Constanța.

Cumpăr en gros dischete 3,5" sau 5,25" HD, firme cunoscute: TDK, Sony, 3M etc. Rețin doar oferte cu preț. Bucur Daniel, 3125 Mediaș, O.P.1, C.P. 47, jud. Sibiu.

Vînd două microcalculatoare Junior 80B cu unități floppy de 5,25" absolut noi. Tel. 952/15768, int. 262

"...Ziceam că revista place. Nu chiar într-atît încît s-o cumpere, totuși... De vină este și prețul ridicat, dar și lipsa de publicitate. Evident, și publicitatea costă, dar fără, nu faci nimic.

Iar tirajul este insuficient. Cu un tiraj mai mare s-ar putea reduce prețul, care, așa cum am spus, este unul din obstacolele majore în calea sporirii vînzărilor. Ca și apariția neregulată. Știu, nu e ușor, dar cercul vicios trebuie rupt.

Ar fi bine să fie și articole proprii, adică despre soft românesc, sau realizări (periferice, sisteme, rețele) de la noi. Nu-s suficiente doar traducerile sau prelucrările...

Ar mai trebui și niște comparații cu tabele sinoptice... Comparațiile ar trebui să trateze atît elemente de hard și sisteme (precum "Oglinda prețurilor" din Germania din mai 1991, "Noutăți pe piața românească" din nr. 4, etc., cît și soft și programe de aplicații. Astfel ca cine vrea să-și extindă sistemul să-și cumpere un PC sau să-și instaleze o rețea să fie foarte interesat să apeleze dacă nu la Micro ATCI măcar la colecția revistei "if" pentru ca, pornind de la condițiile specifice și necesități, sau, mai adesea la noi, de la limitele de buget, să știe ce să-și ia.

Un exemplu concret. Un colectiv din institut are un Felix XT, cu hard-disk de 20 Mo și 2 floppy de 360 ko, afișaj grafic CGA. Pentru a mai scoate un ban, vor să se apuce de DTP. Ce program ar fi mai eficient, Ventura, Page Maker sau partea de DTP de la Works? ...

M-ar interesa ceva despre WAN (și-n if, și-n documentație am mult mai puțin ca LAN) și despre sistemul de operare al lui Apple (Mac). Și-au luat unii și nu știu să-l folosească.

Cristian Malde, București

Mulțumim, atît pentru aprecieri, cît mai ales pentru sugestii și observații. Ne vom strădui să ținem seama de ele. Cît despre tiraj și preț... ca și cum i-ai spune celui ce se-neacă că ar fi mai bine să înoate...

Iosif Fettich

Și Dumneavoastră puteți contribui la îmbunătățirea conținutului revistei !
Ajutați-ne, completând chestionarul alăturat, și revista se va apropia mai
mult de ceea ce așteptați de la o revistă destinată utilizatorilor de PC-uri !
Vă mulțumim !

Chestionar

1. Cel mai mult mi-au plăcut următoarele articole:
1. _____
2. _____
3. _____
2. Cel mai puțin mi-au plăcut următoarele articole:
1. _____
2. _____
3. _____
3. În numerele următoare aș dori să fie incluse articole care să trateze:
1. _____
2. _____
3. _____
4. Utilizez un calculator:
|_| acasă |_| la birou
5. Tipul calculatorului utilizat:

6. Lucrez mai frecvent cu următoarele pachete software:
1. _____
2. _____
3. _____
7. P.C-urile mă interesează ca:
|_| profesionist |_| hobby
|_| student |_| elev
|_| utilizator |_| posibil utilizator
8. Citesc revista "if":
|_| regulat |_| ocazional
9. Ce-ar mai fi de spus ? _____

Data: _____ Semnătura: _____ 18 / 91

**Informații la zi din
lumea calculatoarelor
personale puteți
obține numai citind
regulat revista
"if" !**

Micro ATCI
C.P. 64, O.P. 1
RO - 4300 Tîrgu Mureș

**Aveți ceva de vîndut?
Doriți să
cumpărați ceva ?
Vreți să vă oferiți
serviciile sau aveți
nevoie de ajutor într-o
problemă ?
Folosiți mica publicitate
de specialitate din
"if" !**

Micro ATCI
C.P. 172, O.P. 1
RO - 4300 Tîrgu Mureș

OFERTA DE TOAMNA LA AbMod

Dupa marele succes inregistrat in aceasta vara

ACTIUNEA EGA continua :
SISTEME COMPATIBILE IBM PC/AT la numai

239.000 lei

AT-286 : 12 MHz, 1MB RAM, HD 40 MB, 1 FDD 1,2 MB 525", 1 FDD 144 MB 35",
Monitor color EGA, 1 Mouse, 10 dischete 525", 10 dischete 35",
Tastatura 101 taste (USA), Interfata seriala, Interfata paralela .

GARANTIE 1 AN *de la livrare !*

*Pentru noul dvs. calculator firma AbMod va ofera
posibilitatea achizitionarii celor mai noi versiuni de
produse software (originale, legal inregistrabile) :*

MS-DOS v. 5.0	27.000 lei
WINDOWS v. 3.0	24.600 lei
MS-DOS v. 5.0 + WINDOWS v. 3.0	47.600 lei
NORTON UTILITIES v. 6.0	63.000 lei
NORTON COMMANDER v. 3.0	43.200 lei
STACKER v. 1.10	55.800 lei

(dubleaza capacitatea discului Winchester)

ATENTIE !

*Pentru comenzi en-gros firma
acorda reduceri de pret .*



ASTEPTAM COMENZI FERME LA UNA DIN ADRESELE :

3400 CLUJ-NAPOCA B-DUL 22 DECEMBRIE NR.135 TELEFON 95/156350

3700 ORADEA STR DACIA NR. 40 TELEFON 991/60278; FAX 991/56881



Delta Design S.A., Hotel București, 63-81 Victoria Street,
Wing D - Agencies, Stair E 2, 4th Floor, Sect. 1, Bucharest,
Phones: 15.42.26 / 40 015.45.80 Ext.2404 / Fax: 40 13.60.40.

99.94% of the users fully satisfied of the delivered products

Authorised distributors for the following companies:
Tulip Computers, Apple Computer Inc., Microsoft, Autodesk,
Ashton Tate, Claris, Corel, Delta Design International etc.