

Cartea nr. 7

EDITAT DE MILOAN

LEI 3,50



JOCURI
PE
CALCULATOR
LOGO

UNIVERSITATEA DIN CRAIOVA
BIBLIOTECA

Cota

INVENTAR 109.767

UNIVERSITATEA DIN CRAIOVA
BIBLIOTECA
CENTRALA
DEPOZIT LEGAL

JOCURI PE CALCULATOR —
o nouă posibilitate de instruire
în timpul liber

După părerea unanim acceptată, calculatorul trebuie privit ca un prieten, o unealtă a omului, o sursă de potențare a puterii de creație cu largi orizonturi.

„Calculatorul — spunea Jean Jaques Servan Schreiber — este o unealtă, așa cum a fost cartea după Guttenberg. Ca și cartea, e o trambulină pentru creatori. În fața revoluției electronice nu există decît o alternativă: ori înveți să controlezi tehnologia, ori te lași controlat de ea”.

Jocurile pe calculator nu mai sînt o noutate, dar ceea ce dă importanță domeniului, avînd calculatorul ca partener „inteligent” de întrecere, este și faptul că se vor putea trece aproape toate jocurile logice — așa-zise vechi — pe calculator și că există mari posibilități ca o multitudine de jocuri logice noi să fie lansate pe piață mai întii (sau numai) pe calculator.

Jocul și calculatorul se completează, deci, atît de bine, încît par făcuți unul pentru celălalt. Totuși, ceva intervine și aici. Astfel, în jocul clasic, regulile trebuie cunoscute de toți participanții. Calculatorul are regulile lui ce nu pot fi modificate sau ignorate. Totodată, jucătorul nu mai este un creator, ci un simplu participant, iar calculatorul nu este niciodată un partener

egal, ci un arbitru și un instrument de simulare. Noua dimensiune pe care o introduce calculatorul, bogăția sa, o constituie multitudinea de situații și variante pe care le poate simula.

Dar calculatorul nu se limitează numai la jocuri. El are o gamă largă de posibilități. Dintre acestea, o mare utilizare o are în domeniul instruirii personale în diverse discipline de învățământ, respectiv de la matematică, fizică, chimie până la limbile străine.

RECOOP este interesat să primească orice sugestie și propunere de programe pe calculator.

Publicul, din ce în ce mai pasionat, caracteristicile deosebite ale calculatorului și, mai ales, extraordinara inventivitate în acest domeniu, garantează viitorul unor noi programe pe calculator.

Dr. GH. FEȚEANU

Limbaajul LOGO

Limbaajul LOGO și-a demonstrat calitățile de instruire pentru copii, utilizarea lui permițind familiarizarea cu calculatorul, dezvoltând în același timp capacitatea intelectuală, vorbirea și scrișul, imaginația, puterea de a anticipa acțiuni, orientarea, etc.

LOGO este un limbaaj de programare de nivel înalt, care îmbină simplitatea — modul de adresare se apropie de cel natural — cu un larg spectru de facilități, care permit lucrul interactiv, repetarea unor acțiuni, deținerea de noi proceduri de lucru, pe baza setului de comenzi existente și utilizarea lor recursivă. Astfel, sub forma unui joc, se pot însuși noțiuni de bază de geometrie, aritmetică, proiectare și desen, trigonometrie, muzică, dar și tehnici avansate de programare a calculatoarelor. Jucându-se cu LOGO, copilul învață, descoperind singur legi și fenomene, într-un proces de instruire liberă și autocondusă.

Pentru utilizarea limbaajului LOGO este necesară următoarea configurație:

— un calculator personal compatibil cu tipul Sinclair Spectrum (CIP, HC, TIM-S, COBRA) cu o memorie internă de 48 Ko;

— un televizor sau monitor alb/negru sau color (acesta din urmă fiind de preferat deoarece LOGO are comenzi speciale pentru lucrul cu culori);

— un casetofon uzual;

— caseta magnetică LOGO, editată de RECOOP în colaborare cu ITCI.

Fața 1

LOGO

Pentru a se putea lucra cu limbajul de programare **LOGO**, acesta va trebui încărcat de pe caseta magnetică. Deci, puneți caseta de la început, introduceți comanda **LOAD** " " și porniți casetofonul. În momentul în care apare pe ecran mesajul „**BINE AȚI VENIT ÎN LOGO**” și un semn de întrebare, înseamnă că puteți să opriți casetofonul și să începeți să introduceți comenzi **LOGO**.

Dar atenție! **LOGO** face distincție între literele mici și cele mari, iar comenzile **LOGO** trebuie scrise cu litere mari, literă cu literă. În consecință, primul lucru pe care îl veți face este să treceți sistemul în modul de scriere cu litere mari, prin acționarea simultană a tastelor **CS** și **2**.

ROMÂNĂ

Dacă doriți să folosiți și comenzi **LOGO** în limba română, va trebui să le încărcați, în prealabil, în memoria calculatorului, ele constituind următorul fișier de pe caseta magnetică. Observați că în **LOGO** vom folosi în locul termenului de program (ca ansamblu de instrucțiuni memorate de calculator) pe cel de fișier (ca ansamblu de proceduri-secvențe de comenzi **LOGO**, memorate de calculator sub un nume). **Încărcarea** fișierului de comenzi în limba română se face cu comanda **LOAD „ROMANA”**, după care veți porni casetofonul. În timpul încărcării numele comenzilor în limba română vor fi afișate pe ecran. După încărcare, va apărea din nou pe ecran semnul întrebării, care semnifică faptul că se așteaptă o comandă **LOGO**. Veți putea introduce acum atât comenzile originale, cit și pe cele în limba română al căror titlu l-ați observat pe ecran.

DESENE

Următoarele fișiere de pe caseta magnetică vor evidenția câteva din posibilitățile și facilitățile oferite de limbajul **LOGO**. Astfel, printre cele mai importante, se numără cele grafice și muzicale, sugestiv demonstrate prin următorul fișier de pe casetă, intitulat **DESENE**. **Încărcarea** lui în memoria calculatorului se va face prin comanda **LOAD „DESENE”**, în timp ce **lansarea în execuție** (după ce apare semnul întrebării pe ecran) prin simpla tastare a cuvintului **DESENE**.

Veți observa desenarea pe ecran a câtorva spirale colorate interesant, realizate prin intermediul tehnicilor de recursivitate pe care limbajul **LOGO** le pune la dispoziția utilizatorilor.

JOC

Următorul fișier este un joc ce se poate folosi în vederea acomodării cu distanțele de pe ecran care, în **LOGO**, se măsoară în „pași”. Este recomandat, deci, ca jocul să se practice înainte de învățarea utilizării comenzilor **FORWARD** (**ÎNAINTE**) și **BACK** (**ÎNAPOI**). **Încărcarea** jocului se face prin comanda **LOAD „JOC”**, iar **lansarea în execuție** (pornirea jocului) prin tastarea cuvintului **JOC**. Instrucțiunile de utilizare se vor afișa pe ecran.

„Broasca” (un triunghi în care virful ascuțit indică direcția) poate fi rotită prin intermediul **tastelor S** pentru **stinga** sau **D** pentru **dreapta** pentru a fi orientată spre o țintă afișată pe ecran sub forma unui pătrat. După ce broasca a fost orientată, jucătorul va trebui să acționeze tasta **T** și, apoi, să indice numărul de „pași” pe care trebuie să-i efectueze „broasca” pentru a lovi ținta. Distanța parcursă de broasca va fi echivalentă cu cea obținută pe baza comenzii **LOGO FORWARD** (sau **ÎNAINTE**), urmată de numărul de pași respectivi. Dacă broasca intra cu virful în pătrat, jucătorul va fi felicitat. Jocul poate fi continuat până când jucătorul va acționa tastele **CS** și **SPACE** pentru întreruperea jocului.

ŞAH

Ultimul fişier de pe prima faţă a casetei nu se referă la jocul de şah, fiind numai o frumoasă imagine grafică a unei table de şah cu piese în perspectivă. **Încărcarea** acestei imagini se va face cu comanda **LOADSCR „ŞAH“**, imaginea apărind pe ecran în timpul încărcării. După încărcare, imaginea se poate modifica sau şterge cu comenzi **LOGO** sau, pur şi simplu, se poate indica o altă comandă **LOGO**.

Faţa 2

LOGO

Primul fişier de pe faţa a 2-a a casetei este identic cu cel de pe faţa 1, fiind repetat ca o măsură de siguranţă. Fişierul poate fi utilizat în cazul deteriorării celui de pe faţa 1 sau în cazul în care cel de pe faţa 1 nu se poate încărca din cauza unei erori de citire (apare mesajul „Tape loading error“).

Încărcarea se face (la fel) cu comanda **LOAD “**.

Nu uitaţi, să treceţi în modul de scriere cu litere mari (**CS + 2**) după încărcare.

TITATO

Este vorba de jocul cunoscut şi sub numele de **X** şi **O** pe care acum le puteţi să îl practicaţi jucând împotriva calculatorului. **Încărcarea** jocului se face cu comanda **LOAD „TITATO“**, în timp ce **lansarea în execuţie** se face (după apariţia semnului de întrebare) prin tastarea **TTT**. Atenţie! Încărcarea jocului **TITATO** se va face fără să existe în memorie şi setul de comenzi informaţii în limba română, deoarece în acest caz procedurile pentru joc nu vor mai avea loc în memorie. Dacă, totuşi, există mai multe proceduri în

memorie, atunci, înainte de încărcarea lui **TITATO**, se va proceda la ştergerea lor cu ajutorul comenzii **ERASE** (sau **UITA** dacă este încărcat setul de comenzi în limba română).

După începerea jocului se trasează pe ecran o grilă de 3×3 poziţii şi jucătorul are posibilitatea să aleagă cine va efectua prima mutare: jucătorul (tasta **X**) sau calculatorul (tasta **O**).

Pentru efectuarea mutărilor se vor folosi **tastele numerice**: tasta **1** va indica o mutare pe poziţia din colţul stînga sus, în timp ce tasta **9** va indica o mutare pe poziţia din colţul din dreapta jos. Mutările jucătorului se efectuează prin punerea semnului ales (**X** sau **O**) pe poziţia indicată, iar cele ale calculatorului prin punerea celui/lalt semn pe o anumită poziţie aleasă de calculator.

Dacă pentru efectuarea unei mutari se indică alta tasta decît una numerică sau se indică o poziţie deja ocupată, apare mesajul „mutare ilegală“, iar mutarea se poate repeta. Mutările vor alterna. Scopul jocului este de a realiza 3 semne pe aceeaşi linie (orizontală, verticală sau diagonală).

Dacă nici calculatorul şi nici jucătorul nu reuşesc acest lucru după umplerea tuturor poziţiilor grilei, atunci se da mesajul „**REMIZA**“ şi jocul se poate relua după apăsarea oricărei taste cu comanda **TTT**. Dacă jucătorul ori calculatorul va realiza punerea a 3 semne pe aceeaşi linie, se va da mesajul respectiv („Ai câştigat!“ sau „Am câştigat!“) şi jocul se va putea relua după acţionarea oricărei taste cu comanda **TTT**.

FOTO

Ca şi pe prima faţă a casetei, apare o frumoasă imagine grafică, de data aceasta însă, a unui aparat de fotografie. **Încărcarea** imaginii se va face cu comanda: **LOADSCR „FOTO“**. Imaginea va apărea pe ecran în timpul încărcării. După apariţia imaginii, se poate introduce orice comandă **LOGO**.

Caseta magnetică **LOGO** a fost realizată de Ion Diamandi şi Cristian Constantinescu.

UN SUMAR AL LIMBAJULUI LOGO

Notă: acest sumar prezintă câteva reguli de bază ale versiunii limbajului **LOGO** implementat pe calculatoarele **Sinclair Spectrum** și compatibile, precum și o foarte succintă descriere a comenzilor și operațiilor pe care limbajul le pune la dispoziție. Din acest punct de vedere materialul se prezintă ca un memorator, adresându-se mai degrabă unor persoane avizate decît unor începători. Pentru această ultimă categorie, în scopul învățării limbajului **LOGO**, recomandăm folosirea, împreună cu materialul de față a **GHI-DULUI LOGO**.

În cadrul unui capitol, comenzile și operațiile sînt descrise în ordine alfabetică.

Se folosesc următoarele notații: **CS** pentru **CAPS SHIFT**; **SS** pentru **SYMBOL SHIFT**; **CS + tastă** pentru acționarea simultană a celor două taste.

Manevrarea cursorului

Ștergerea caracterului dinaintea cursorului: **CS + 0**

Deplasare la dreapta: **CS + 8**

Deplasare la stînga: **CS + 5**

Deplasare în jos: **CS + 6**

Deplasare în sus: **CS + 7**

Schimbarea regimului tastaturii și al ecranului

Regim „I” regim „C”: **CS + 2**

(litere mici) (litere mari)

Regim „I” sau „C” regim „E”: **CS + SS**

Regim grafic al ecranului regim textual: comanda **TEXT-SCREEN (TS)**.

Proceduri

Definiție: se numește **procedură** orice secvență de instrucțiuni memorate de calculator sub un **nume**, în vederea executării ei la întîlnirea numelui respectiv.

Structura unei proceduri:

TO nume
comenzi

•

•

•

END

Apelarea procedurii: prin numele ei

Întreruperea executării unei proceduri: **CS + SPACE**

Obiecte LOGO

Obiectele LOGO sînt cuvinte sau liste utilizate ca intrări sau ieșiri ale procedurilor.

Un **cuvînt** este o serie de caractere alfabetice sau numerice; pentru a fi deosebite de numele de proceduri, înaintea cuvintelor se pun ghilimele; un cuvînt nu poate conține caracterul „spațiu”, deoarece acesta indică terminarea cuvîntului. Un cuvînt este, de obicei, delimitat prin spații. Mai există și alți **delimitatori**, ca de exemplu: $[]$ $()$ $=$ $>$ $<$ $+$ $-$ \times $/$ care separă un cuvînt de restul liniei. Între un cuvînt și oricare din acești delimitatori nu este necesară inserarea spațiului, deși, uneori, se obișnuiește acest lucru pentru claritate.

Numeralele sînt, de asemenea, cuvinte dar ele se pot scrie fără ghilimele ca și cuvintele care exprimă valori logice (**TRUE** — adevărat și **FALSE** — fals).

O **listă LOGO** constă dintr-o serie de obiecte **LOGO**, adică cuvinte sau liste; o listă este cuprinsă între paranteze drepte, iar elementele unei liste sînt separate prin spații; **lista vidă** (fără nici un element) este $[]$.

Cuvintele pot fi utilizate și ca **nume de variabile**.

Elementele unui cuvînt sînt caractere, iar elementele unei liste sînt cuvinte sau liste.

Cuvintele sau listele pot fi „depuse” în variabile printr-o instrucțiune de atribuire (MAKE).

Proceduri cu subiect

Pentru a putea funcționa, unele proceduri necesită subiecte sau parametrii de intrare. Acestea sînt obiecte **LOGO** (cuvinte sau liste). De exemplu:
? PRINT [BUNA DIMINEAȚA]

Lista [BUNA DIMINEAȚA] este un subiect (sau o intrare) pentru procedura primitivă PRINT. Semnul întrebării este **promptul** specific lui **LOGO** care semnifică faptul că se așteaptă introducerea unei comenzi **LOGO**.

Comenzi și operații

O procedură poate fi ori o comandă ori o operație (funcție). O operație este o procedură care furnizează o valoare, deci, poate fi utilizată ca parametru de intrare pentru altă comandă sau operație.

Ghilimele, două puncte, paranteze pătrate și drepte

LOGO interpretează orice cuvânt ca o procedură. Pentru ca un cuvânt să nu fie interpretat ca o procedură va trebui precedat de unul din următoarele semne:

- " ghilimelele indică lui **LOGO** că șirul de caractere care urmează, terminându-se cu un spațiu, este un cuvânt; două puncte indică lui **LOGO** că șirul de caractere care urmează, terminându-se cu un spațiu, este numele unui obiect **LOGO**.
- [] parantezele pătrate indică lui **LOGO** că elementele dintre aceste paranteze formează o listă.
- () parantezele rotunde sînt utilizate pentru a permite mai mult de două intrări pentru unele primitive. De asemenea, parantezele rotunde mai sînt utilizate pentru realizarea unor operații aritmetice. De exemplu:
? PRINT 2 X (3 + 5)
16

Variabile

În **LOGO** variabilele au un nume. În mod normal, variabilele sînt create cu comanda **MAKE (PUNE)**. De exemplu: ?MAKE "A 1

Valoarea variabilei poate fi obținută prin punerea caracterului înainte de numele variabilei. De exemplu: ?PRINT:A

O variabilă în **LOGO** poate avea ca valoare o varietate de tipuri de date: o valoare numerică, o listă de valori numerice, un cuvînt, o listă de cuvînte, etc.

O variabilă creată în **LOGO** prin comanda **MAKE** va fi o variabilă **globală**, adică va exista atît în timpul executării procedurii, cît și după aceea.

O variabilă declarată ca parametru de intrare într-o procedură (și nu creată anterior) va fi o variabilă **locală** a acestei proceduri și a oricărei alte proceduri, adică nu va avea atribuită o valoare decît în timpul executării procedurii.

Editorul LOGO

Pentru modificarea procedurilor definite sau chiar pentru definirea noilor proceduri, se folosește **editorul LOGO**.

Apelare: **EDIT (ED)** numele procedurii

Utilizare editor:

- CS + 5 deplasează cursorul un caracter la stînga
- CS + 6 deplasează cursorul o linie mai jos
- CS + 7 deplasează cursorul o linie mai sus
- CS + 8 deplasează cursorul un caracter la dreapta
- CS + 0 șterge caracterul dinaintea cursorului
- CS + 6 (în mod E) deplasează cursorul la începutul liniei
- CS + 6 (în mod E) deplasează cursorul la sfîrșitul ecranului
- CS + 7 (în mod E) deplasează cursorul la începutul ecranului
- CS + 8 (în mod E) deplasează cursorul la sfîrșitul liniei
- B (în mod E) deplasează cursorul la începutul textului
- E (în mod E) deplasează cursorul la sfîrșitul textului
- N (în mod E) deplasează cursorul la pagina următoare
- P (în mod E) deplasează cursorul la pagina precedentă
- Y (în mod E) șterge de pe ecran linia pe care se află cursorul
- R (în mod E) reinserează linia care a fost ștersă cu comanda Y (în mod E)

C (în mod E)

CS + SPACE

părăsirea editorului cu încorporarea modificărilor efectuate și intrarea în LOGO
părăsirea editorului cu ignorarea modificărilor efectuate și intrarea în LOGO

COMENZI ȘI OPERAȚII

1. Broasca țestoasă

BACK n deplasare înapoi cu n pași (pixeli).

BK n

ÎNAPOI n

IP n

BACK GROUND furnizează un număr (operație) care reprezintă culoarea fondului. Numărul furnizat poate fi: 0 — dacă fondul are culoarea neagră, 1 — albastră, 2 — roșie, 3 — violetă, 4 — verde, 5 — bleu, 6 — galbenă, 7 — albă.

Notă: broasca țestoasă este numele dat micului triunghi care apare pe ecran când se utilizează ecranul grafic. Mișcându-se dintr-un punct în altul, broasca va lăsa urme, putându-se trasa în acest mod linii pe ecran. Ecranul grafic, numit și cîmpul broaștei țestoase, apare oricînd se introduce o comandă referitoare la mișcarea broaștei. În mod normal, ecranul grafic are 256 de pași (puncte) pe orizontală și 175 de pași (puncte) pe verticală, iar originea este în centrul ecranului.

CLEAN

ștergerea ecranului fără aducerea broaștei acasă.

CLEARSCREEN

ștergerea ecranului cu aducerea broaștei acasă.

CS

ȘTERGE

DOT [x y]

PUNCT [x y]

desenează un punct la poziția specificată prin coordonatele x, y. Broasca nu se mișcă de la locul ei și nu se trasează nici o linie.

FENCE GARD

limitează mișcările broaștei în cadrul ecranului grafic.

FORWARD n

FD n

ÎNAINTE n

IN n

deplasare înainte cu n pași.

HEADING DIRECȚIE

furnizează un număr (operație) între 0 și 359 care reprezintă orientarea actuală a broaștei în ecran. 0 = direcție N, 90 = Est, 180 = Sud, 270 = Vest.

HIDETURTLE HT FĂRĂ BROASCA FB

ascunde broasca (face broasca invizibilă).

HOME ACASA

readucerea broaștei acasă.

LEFT n

LT n

ȘTINGĂ n

SA n

rotire spre stînga cu n° (grade) fără ca broasca să-și schimbe poziția.

PENCOLOUR PC CULOARE

furnizează un număr (operație) care reprezintă culoarea cu care va desena broasca. La început culoarea va fi 0 (neagră).

PENDOWN PD CREION CR

penița jos: cînd broasca se va mișca ea va trasa o linie.

PENERASE PE GUMA GU

ștergerea liniiei: broasca va șterge orice linie peste care va trece. Comenzile PENDOWN, PENUP sau PENREVERSE vor anula efectul comenzii PENERASE.

PENREVERSE
PX
CI
peniță inversă: broasca în mișcare va trasa linii, dar va șterge liniile peste care va trece (desenează cu culoarea inversă).

PENUP
PU
FARA CREION
FC
penița sus: broasca în mișcare nu va trasa linii.

POSITION
POS
POZIȚIE
furnizează o listă de două numere (operație) care reprezintă poziția actuală a broaștei.

RIGHT n
RT n
DREAPTA n
DR n
rotire spre dreapta cu n° (grade) fără ca broasca să-și schimbe poziția.

SCRUNCH
RELXY
furnizează o listă de două numere (operație), care reprezintă proporția dintre mărimea unui pas pe verticală și a unuia pe orizontală.

SETBG n
FIXFOND n
fixează culoarea fondului (0 — negru, 1 — albastru etc).

SETBORDER n
SETBR n
FIXCHENAR n
fixează culoarea chenarului (borderului) (0 — negru, 1 — albastru, etc).

SETHEADING n
SETH n
FIXDIR n
rotește broasca, astfel încât orientarea ei să fie n. La început orientarea broaștei este 0.

SETPC n
FIXCULOARE n
fixează culoarea cu care se va desena (n = 0 culoarea neagră, n = 1 culoarea albastră, etc.)

SETPOS [x y]
FIXPOZ [x y]
deplasează broasca în punctul de coordonate x și y, fără a-i schimba orientarea (efectuează o translație pînă în punctul respectiv).

SETSCRUNCH [x y]
SETSCR [x y]
FIXSCARA [x y]
fixează proporția dintre mărimea pasului pe verticală și mărimea pasului pe orizontală. În mod normal (la început) această proporție este [100 100].

SETX n
FIXX n
deplasează broasca într-un punct a cărui coordonată pe orizontală (X) este n.

SETY n
FIXY n
deplasează broasca într-un punct a cărui coordonată pe verticală (Y) este n.

SHOWNP
operație care furnizează cuvîntul TRUE dacă broasca este vizibilă (este în modul SHOWTURTLE) sau FALSE dacă nu este.

SHOWTURTLE
ST
BROASCA
arată broasca (face broasca vizibilă).

TOWARDS [x y]
CAP [x y]
furnizează orientarea pe care trebuie să o capete broasca pentru a fi îndreptată spre punctul de coordonate x și y (operație); nu modifică orientarea broaștei.

WINDOW
FEREASTRA
permite broaștei să se deplaseze în afara ecranului grafic. Cînd broasca este în modul WINDOW, va continua să execute comenzile de deplasare chiar fără a fi văzută.

WRAP
IESEREVINE
permite broaștei ca atunci cînd iese din ecran să apară din partea opusă a ecranului.

XCOR
furnizează coordonata pe orizontală (abscisa) actuală a broaștei (operație).

YCOR

furnizează coordonata pe verticala (ordonata) actuală a broaștei (operație).

2. Cuvinte și liste

ASCII caracter

operație care furnizează codul ASCII al unui caracter.

BUTFIRST obiect

operație care furnizează obiectul fără primul element al obiectului.

BF obiect

FARAPRIMUL obiect

FP obiect

BUTLAST obiect

operație care furnizează obiectul fără ultimul element al obiectului.

BL obiect

FARALTIMUL obiect

FU obiect

CHAR n

CARACTER n

operația care furnizează caracterul al cărui cod ASCII este n. Codul n trebuie să fie un întreg cuprins între 32 și 165.

COUNT obiect

NREL obiect

operație care furnizează numărul de elemente al obiectului (cuvânt sau listă).

EMPTY obiect

GOL obiect

operație care furnizează valoarea logică TRUE dacă obiectul LOGO este gol și valoarea logică FALSE în caz contrar.

EQUALP obiect 1,

obiect 2

EGAL obiect 1,

obiect 2

operație care furnizează valoarea logică TRUE dacă obiect 1 și obiect 2 sînt două numere egale, două cuvinte sau liste identice.

FIRST obiect

PRIMUL obiect

operație care furnizează primul element al obiectului. Acesta va fi un caracter dacă obiectul este un cuvînt ori un cuvînt sau o listă dacă obiectul este o listă.

FPUT obiect listă

PUNEPRIMUL

obiect listă

Operație care furnizează o nouă listă, formată prin punerea (alipirea) obiectului la începutul listei (FirstPUT).

ITEM n obiect

Operație care furnizează al n-lea element al unui obiect

LAST obiect

ULTIMUL obiect

Operație care furnizează ultimul element al obiectului. Acesta va fi un caracter dacă obiectul este un cuvînt sau o listă dacă obiectul este o listă.

LIST obiect 1, obiect 2

(**LIST** obiect 1,

obiect 2 ... obiect n)

LISTA obiect 1, obiect 2

(**LISTA** obiect 1,

obiect 2 ... obiect n)

Operație care furnizează o listă ale cărei elemente sînt obiect 1, obiect 2, etc.

LISTP obiect

Operație care furnizează valoarea logică TRUE dacă obiectul este o listă și FALSE în caz contrar. O listă vidă este considerată un cuvînt gol.

LPUT obiect listă

PUNEULTIMUL

obiect listă

Operație care furnizează o nouă listă, formată prin punerea (alipirea) obiectului la sfîrșitul listei (LastPUT).

MEMBERP obiect listă

Operație care furnizează valoarea logică TRUE dacă obiectul este un element al listei și FALSE în caz contrar.

NUMBERP obiect

Operație care furnizează valoarea logică TRUE dacă obiectul este un număr și FALSE în caz contrar.

SENTENCE obiect 1, Obiație cu care se formează o listă compusă din obiectele date.
obiect 2
(**SE** obiect 1
obiect 2 ... obiect n)

FRAZA obiect 1, obiect 2
(**FRAZA** obiect 1,
obiect 2 ... obiect n)

WORD cuvânt 1, cuvânt 2 Obiație cu care se formează un cuvânt din obiectele date.
(**WORD** cuvânt 1,
cuvânt 2 ... cuvânt n)

WORDP obiect Obiație care furnizează valoarea logică TRUE dacă obiectul este un cuvânt și FALSE în caz contrar.

3. Variabile

MAKE nume obiect atribuie valoarea „obiect” variabilei „nume” (pune în locația de memorie numită „nume” valoarea „obiect”).
PUNE nume obiect

NAMEP obiect obiație care furnizează valoarea logică TRUE dacă obiectul are o valoare și FALSE în caz contrar.

THING nume obiație care furnizează conținutul unui nume.

4. Operații aritmetice

ARCCOS n furnizează valoarea, în grade, a arccosinusului de n.

ARCCOT n furnizează valoarea, în grade, a arccotangentei de n.

ARCSIN n furnizează valoarea, în grade, a arcsinusului de n.

ARCTAN n furnizează valoarea, în grade, a arctangentei de n.

COSINE n furnizează valoarea, în grade, a cosinusului de n.
COS n

COTANGENT n furnizează valoarea, în grade, a cotangentei de n.
COT n

DIV a b furnizează citul obținut prin împărțirea lui a prin b.

INT n furnizează partea ÎNTreagă a lui n.

PRODUCT a b furnizează produsul subiectelor (intrărilor). Este echivalentă cu operația de înmulțire simbolizată prin semnul \times .
(**PRODUCT** a b ... n)
PROD a b
(**PROD** a b ... n)

RANDOM n furnizează un număr aleator întreg, cuprins între 0 și (n-1).

Notă: adunarea, scăderea, înmulțirea și împărțirea pot fi utilizate în forma infix, în care operatorii (+ - \times /) se pun între parametri de intrare.

Adunarea, înmulțirea și împărțirea pot fi utilizate și în forma prefix, dar în acest caz, SUM, PRODUCT, respectiv DIV, vor fi urmate de doi parametri de intrare. În LOGO se pot efectua operații aritmetice atât cu numere întregi, cit și zecimale.

REMAINDER a b furnizează restul împărțirii lui a prin b

REST a b

ROUND n furnizează n rotunjit la cel mai apropiat întreg.

SINE n furnizează valoarea, în grade, a sinusului de n.
SIN n

SQRT n furnizează rădăcina pătrată din n; n trebuie să fie pozitiv.

SUM a b
SUM a b ... n

furnizează suma subiectelor (intrărilor). Este echivalentă cu operația de adunare simbolizată prin semnul +.

TANGENT n
TAN n

furnizează valoarea, în grade, a tangentei de n.

a + b

furnizează suma lui a cu b.

Notă: în operațiile aritmetice (cu semne) se recomandă ca semnul să fie încadrat de spații, altfel LOGO poate interpreta că unul dintre subiecte este un număr negativ, de exemplu în cazul scăderii.

a - b

furnizează diferența dintre a și b.

a × b

furnizează produsul dintre a și b.

a/b

furnizează cîțul dintre a și b (a împărțit la b).

a < b

furnizează valoarea logică TRUE dacă a este mai mic decît b și FALSE în caz contrar.

a > b

furnizează valoarea logică TRUE dacă a este mai mare decît b și FALSE în caz contrar.

a = b

furnizează TRUE dacă a este egal cu b, de oricare tip ar fi aceste subiecte (cuvinte, liste sau numere) și FALSE în caz contrar.

5. Definiri și editări de proceduri

EDIT procedură
ED procedură

intrarea în modul de lucru de editare a procedurii.

EDNS nume

permite editarea numelor și a va-

EDNS [nume lista]

END

TO nume

lorilor lor.

Fără subiect se vor lista toate variabilele cu valorile lor; cu subiect se vor lista numai variabilele numite, cu valorile lor.

indica sfîrșitul definirii unei proceduri.

permite începerea definirii unei proceduri.

6. Condiții de control

BYE

ieșire din LOGO cu trecerea controlului în interpretorul BASIC. Se poate intra din nou în LOGO prin comanda RUN.

IF condiție

[lista 1 de comenzi]

[lista 2 de comenzi]

DACA condiție

[lista 1 de comenzi]

[lista 2 de comenzi]

dacă este îndeplinită condiția, se execută lista 1 de instrucțiuni, dacă nu, se execută lista 2.

Formă particulară:

IF condiție [lista de
instrucțiuni]

lista este executată doar dacă este îndeplinită condiția; în caz contrar lista nu se execută și se trece la linia LOGO care urmează.

OUTPUT obiect

OP obiect

ieșire dintr-o procedură cu furnizarea unui rezultat, care se atribuie ca valoare numelui procedurii.

Notă: o procedură se transformă în „funcție” dacă se termină cu instrucțiunea de „ieșire” OUTPUT.

REPEAT n

[listă de comenzi]

REPETA n

[listă de comenzi]

repetă de n ori comenzile din listă.

Notă: n trebuie să fie pozitiv.

RUN [listă de comenzi] se execută lista ca o linie LOGO.
STOP oprește execuția procedurii curente și redă controlul procedurii care a apelat-o.
TOPLEVEL oprește execuția procedurii curente și redă controlul lui LOGO (apare promptul ?). Este echivalentă cu acționarea tastelor CS și **SPACE**.

7. Operații logice

Notă: Parametri de intrare pentru primitivele care reprezintă operații logice pot fi TRUE sau FALSE sau condiții. Termenul de predicat (pred) este folosit pentru a descrie o funcție ai cărei parametri de intrare pot fi TRUE sau FALSE sau condiții și care furnizează ca rezultat TRUE sau FALSE, cum este cazul operațiilor logice.

AND pred1 pred2 furnizează TRUE dacă toți parametrii de intrare sînt TRUE, și FALSE în caz contrar.
(**AND** pred1 pred2 ... predn)

SI pred1 pred2
SI pred1 pred2 ... predn)

NOT pred furnizează TRUE dacă predicatul (parametrul de intrare) este FALSE, și FALSE dacă predicatul este TRUE.
Nu pred

OR pred1 pred2 furnizează TRUE dacă cel puțin unul din predicate este TRUE, și FALSE dacă nici unul din predicate nu este TRUE.
(**OR** pred1 pred2 ... predn)
SAU pred1 pred2
(**SAU** pred1 pred2 ... predn)

8. Extragere rezultate

KEYP furnizează TRUE dacă este acționată o anumită tastă ori o combinație de anumite taste valide, sau FALSE în caz contrar.

PRINT obiect
PR obiect
SCRIE obiect

READCHAR
RC
CITCAR

READLIST
RL
CITLIST

SHOW obiect

SOUND [durată înălțime]
SUNET [durată înălțime]

TYPE obiect
(**TYPE** obiect1 obiect2 ... obiectn)

WAIT n
AȘTEAPTA n

afișează conținutul obiectului (cuvînt, listă sau variabilă). Cuvîntul se va afișa fără ghilimele, iar lista fără paranteze pătrate.

comandă calculatorului să aștepte apăsarea unei taste și, apoi, furnizează caracterul introdus, care va fi preluat de o procedură, ca orice rezultat al unei operații.

furnizează lista dată de utilizator la tastatură. Întreaga linie, introdusă înainte de acționarea tastei CR, este considerată o listă. În urma acționării tastelor, caracterele corespunzătoare sînt afișate pe ecran.

afișează cuvîntul, lista sau numerele date ca parametri de intrare. Listele sînt afișate cu paranteze pătrate.

produce un sunet de o anumită durată și de o anumită înălțime. Durata sunetului se dă în secunde (de la 0 la 255), iar înălțimea în semitonuri. Do-ul central este reprezentat prin valoarea 0, apoi sunetele cu înălțimea mai mare prin numere întregi pozitive iar sunetele cu înălțimea mai mică prin numere întregi negative. Al doilea parametru trebuie cuprins între -62 și 75.

afișează obiectele. Spre deosebire de PRINT și SHOW, nu provoacă linefeed după afișare.

așteaptă n/50 secunde.

9. Ecranul

Notă: după încărcarea lui LOGO, ecranul se află în modul text, putându-se afișa 22 linii de text. În modul grafic (în care se intră automat după orice comandă grafică), cele 22 de linii pot fi folosite pentru grafică, iar alte două linii, din partea de jos a ecranului, pentru mesaje.

Pe fiecare linie se pot afișa 32 de caractere. Pe prima coloană în modul comandă și în modul de definire a unei proceduri este un prompt (care indică așteptarea introducerii unei comenzi), iar ultima coloană este rezervată pentru semnul exclamării, care indică o linie LOGO neterminată, mai lungă de 32 de caractere.

BRIGHT n	stabilește dacă afișarea va fi cu luminositate ($n = 1$) sau nu ($n = 0$).
CLEARTEXT CT ȘTERGETEXT	șterge tot textul de pe ecran. În modul grafic, această comandă va șterge textul de pe ultimele două linii ale ecranului.
COPYSCREEN	realizează o copie a ecranului la imprimantă (hard copy).
CURSOR	furnizează o listă formată din două numere: primul va reprezenta coloana, iar al doilea linia pe care se găsește cursorul.
FLASH	afișarea se va face pe un fond „clipitor“ pînă la întîlnirea comenzii NORMAL.
INVERSE	afișarea se va face inversîndu-se culoarea pentru text și desene cu cea a fondului pînă la întîlnirea comenzii NORMAL.
NORMAL	afișarea se va face fără „clipire“ și fără inversarea culorilor.
OVER n	afișarea se va face peste orice de-

sen sau text (supratipărire) în cazul lui OVER 1. Cu OVER 0 se revine în modul obișnuit.

SETCURSOR [a b]

SETCUR [a b]

FIXCURSOR [a b]

SETTC [a b]

TEXTCOLOUR

TC

TEXTSCREEN

TS

11. Spațiul de lucru

Notă: o parte din memoria, inițial disponibilă, fiind ocupată chiar de interpretorul LOGO, partea de memorie care rămîne la dispoziția utilizatorului se numește **spațiul de lucru**.

ERALL

șterge tot ce a fost creat (proceduri și variabile) în spațiul de lucru (ERases ALL).

Notă: folosirea acestei comenzi va avea ca efect și ștergerea tuturor procedurilor în limba română, acestea fiind și ele memorate în spațiul de lucru. Comanda ERALL nu afectează conținutul curent al editorului. Pentru a șterge și editorul se folosește ERALL și EDIT [].

- ERASE** nume șterge procedura sau procedurile (din listă) din spațiul de lucru.
- ER** nume
- UITA** nume
- ERN** nume șterge variabila (variabilele) numite din spațiul de lucru (ERases Named).
- ERNS** șterge numele și valorile atribuite tuturor variabilelor din spațiul de lucru (ERases NameS).
- ERPS** șterge toate procedurile din spațiul de lucru (ERases all the ProcedureS).
- PO** nume **Notă:** folosirea acestei comenzi va avea ca efect și ștergerea tuturor procedurilor în limba română afișează definiția procedurii sau procedurilor din listă (Prints Out).
- POALL** afișează numele și definițiile tuturor procedurilor, precum și valorile tuturor variabilelor din spațiul de lucru (Print Out ALL).
- PONS** afișează numele și valorile tuturor variabilelor din spațiul de lucru (Prints Out the NameS).
- POPS** afișează definițiile tuturor procedurilor curente din spațiul de lucru (Prints Out ProcedureS).
- POTS** afișează numele tuturor procedurilor definite (Prints Out the TitleS).

12. Salvări și încărcări de fișiere

Notă: toate procedurile definite în timpul unei sesiuni de lucru sînt memorate de către LOGO în spațiul de lucru. Toate realizările (proceduri, desene, etc) pot fi salvate oricînd pe caseta magnetică și refolosite, la nevoie, prin încărcarea lor în memoria calculatorului de pe același suport magnetic. Realizările pot fi aranjate în fișiere pe trei tipuri:

- fișierul de proceduri LOGO (tipul LOG) este un fișier care conține proceduri LOGO. De exemplu, ROMANA sau DESENE de pe caseta LOGO sînt fișiere de proceduri LOGO.
- fișierul editor (tipul TXT) este reprezentat prin conținutul curent al editorului LOGO.
- fișierul ecran (tipul SCR) este reprezentat prin imaginea ecran curentă.

- LOAD " nume fișier "** încarcă fișierul **nume fișier** de pe casetă în spațiul de lucru. Dacă extensia care precizează tipul fișierului lipsește, atunci se consideră implicit că tipul este LOG. În timpul încărcării sînt afișate pe ecran numele procedurilor din fișier, iar după încărcare, promptul LOGO (?) va reapărea pe ecran.
- LOAD " nume fișier "** încarcă fișierul editor salvat cu **SAVED " nume fișier și il face conținutul curent al editorului.**
- LOADSCR " nume fișier "** încarcă fișierul ecran salvat cu **SAVESCRC " nume fișier și afișează conținutul său (imaginea) pe ecran.**
- PRINTON** tipărește la imprimantă tot ce se va afișa pe ecran.
- PRINTOFF** oprirea tipăririi la imprimantă.

SAVE " nume fișier "
[nume]

salvează procedurile din lista de proceduri sub forma unui fișier cu numele nume fișier. Numele dat fișierului poate avea pină la 7 caractere.

SAVEALL " nume fișier " salvează tot ce există în spațiul de lucru (proceduri și variabile) sub numele nume fișier.

SAVED " nume fișier " salvează tot ce există în Editorul LOGO sub numele nume fișier.

SAVESCR " nume fișier " salvează imaginea ecran sub numele nume fișier.

13. Funcții și primitive avansate

Notă: unele primitive permit ca procedurile să fie definite și minuite din interiorul altor proceduri.

Unele primitive afectează însuși sistemul LOGO. Acestea se pot utiliza în vederea citirii conținutului memoriei. Se recomandă însă salvarea în prealabil a spațiului de memorie, acesta putându-se altfel distruge (în mod accidental). În general, numele acestor primitive începe cu un punct.

NODES

furnizează numărul de noduri libere sau, cu alte cuvinte, spațiul de memorie disponibil în spațiul de lucru pentru proceduri, variabile și rulări de proceduri. Un nod ocupă 5 octeți. Prin utilizarea lui **NODES**, imediat după **RECYCLE**, se furnizează numărul de noduri care sînt încă libere.

RECYCLE

eliberează maximum posibil de noduri prin procedul „colectarea gunoierului”.

COPYDEF "nume nou"
nume

copiază definiția unui nume de procedură existentă sub un alt nume (nou). Procedura existentă nu este ștearsă.

DEFINE " nume "
[param ... paramn]
[liniei ... liniieni]

Comanda are doi parametri de intrare: primul este numele unei proceduri, iar al doilea o listă. Elementele listei sînt: o listă de parametri de intrare pentru noua procedură și, al doilea, o listă ale cărei elemente sînt liniile procedurii. Cu această comandă se pot scrie proceduri care definesc alte proceduri.

● **CONTENTS**

furnizează o listă care include toate procedurile, variabile, etc, din sistemul LOGO.

● **PRIMITIVES**

listează toate primitivele LOGO.

● **RESERVE** n

rezervă n octeți în spațiul de lucru pentru un program în cod mașină. Poate fi utilizată numai la începutul unei sesiuni de lucru LOGO.

● **RESERVED**

furnizează adresa de început și sfîrșit a zonei de memorie rezervată prin **RESERVE**.

● **BLOAD** " nume fișie "
adresă

încarcă în memorie un fișier de pe suport extern, începînd de la adresa specificată.

- **BSAVE** "nume fișier" [adresă n] salvează n octeți de la adresa specificată, sub un fișier al cărui nume este indicat.
- **SETSERIAL** n stabilește viteza de transmisie. n poate fi 50, 110, 300, 600, 1200, 2400, 4800, 9600, 19200. Inițial, viteza de transmisie este 9600 bauds.
- **SERIALIN** citește un octet de la portul serial (Interfața RS232).
- **SERIALOUT** n trimite un octet la portul serial (interfața RS232).
- **DEPOSIT** adresă n plasează valoarea n în locația de memorie specificată prin adresă. Poate fi utilizată pentru a defini un caracter grafic utilizator (UDG).
- **EXAMINE** adresă regăsește o valoare memorată la adresa specificată.
- **CALL** adresă apelează un program în cod mașină, instalat în prealabil printr-o comandă „BLOAD“.

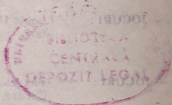
Jocuri realizate de Cristian Constantinescu și Ion Diamandi

RECOOP a realizat următoarele programe pentru calculator:

1. — **JOCURI** : Șah, Jump (salt), Rubic (Jocul pătratelor), GO (Gomoku), Mastermind, Grafice (joc matematic), Lab (culesul cireșelor), Reversi, MLS, Rebec, Dipo (Vulpi și iepuri) **(Caseta nr. 1)**
2. — **JOCURI** : GO (Gomoku), Mastermind, Șah, Reversi, Lab (Labirint) **(Caseta nr. 2)**
3. — **JOCURI** : Fotbal logic, Superevol, Raliu, Simultan **(Caseta nr. 3)**
4. — **JOCURI** : Logic IM-2, Biliard, Iepure, Simultan **(Caseta nr. 4)**
5. — **JOCURI** : Tictactoe, Broșcuțe, Deplasare, Nim, Cărare, Drum, Pătrate, Impas, Triplet, A (Animale), Obstacole, Jungla, Traversare, Vrăjitorul, Ariadna, Vinătoare, Robac, Domino, Hang, Hanoi, Loyd, Predau, Peștera, Labirint, Dipo, Șeptică, Dist. **Caseta nr. 5)**
6. — **CHIMIE** : Acizi, Formula, Mendeleev, Valență, Ecuatii **(Caseta nr. 6)**

7. — **JOCURI**: Logo, Română, Desene, Joc, Șah, Titato, Foto (Caseta nr. 7)
8. — **LIMBA STRAINA**: Limba engleză, Hang, Cuvintele vrăjite, Rebus (Caseta nr. 8)
9. — **FIZICA**: În curs de apariție (Caseta nr. 9)
10. — **MATEMATICA**: Izometrii, TLS (Tangenta, limită a secantei), Triunghi (Caseta nr. 10)

NOTĂ: Cei care pot elabora programe pentru jocuri, diferite discipline de învățămînt, grafică etc. și doresc să colaboreze cu RECOOP, se pot adresa la tel. 13 81 75, 15 04 10, 15 72 93 / interior 112 sau 115.



00363 DEP. LEGAL 1991

L 28 / 125 / 1990

Tiparul executat la I. P. Sibiu

Lee: (3,50)