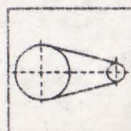
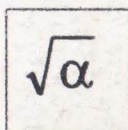




SERIA INFORMATICA

M. M. POPOVICI



BASIC

pentru
calculatoarele
**ZX SPECTRUM, HC,
TIM-S, COBRA,**

COLECTIE DE PROGRAME

EDITURA APH



Bucuresti 1993



SERIA INFORMATICA

M. M. Popovici

BASIC

pentru calculatoarele
ZX SPECTRUM,
HC 90, TIM-S, COBRA,
CIP, JET ...

COLECTIE DE PROGRAME

EDITURA APH



BUCURESTI 1993

În momentul de față nu există domeniu de activitate umană pentru care să nu se fi realizat programe pe calculator. În știință, tehnică, tehnologie, gestiune economică, medicină, învățămînt, proiectare asistată, cercetare științifică, servicii, etc., o serie de programe performante ușurează munca și îi conferă certitudine. Din aceste motive este dificil de selecționat asemenea programe care sînt realizate de profesioniști și instituții, au o lungime apreciabilă și se adresează cu precădere specialiștilor dintr-un anumit domeniu. Într-o lucrare de introducere în limbajul BASIC apare deci utilă oferirea unor programe aplicative care să prezinte - în principal - următoarele caracteristici:

- să rezolve o gamă de probleme de interes general;
- să fie clare și ușor de integrat în alte programe;
- să valorifice fondul de informații acumulat în urma lecturii capitolelor anterioare.

Avînd în vedere aceste considerente s-a apreciat că grupul de aplicații trebuie să fie divizat în trei categorii:

- aplicații în tehnică (rezistența materialelor, mecanisme și organe de mașini)
- aplicații în domeniul matematicii, avînd în vedere impactul matematicii asupra multor profesii;
- aplicații de uz general, folosibile oricărui posesor de calculator personal.

Copyright © 1993 EDITURA APH - SRL
 str. Cap. Preda nr. 12 sect. 5, 76437 - București 69
 tel. 780.93.97; 780.74.77

Tehnoredactare, coperta: R. M. HRISTEV

Bun de tipar: 2.2.1993

Format 84x59/16. Coli de tipar: 14.5

Tipografia I.I.R.U.C

13. APLICAȚII ÎN TEHNICĂ

În acest capitol sînt selecționate cîteva teme din domeniul științelor ingineresti : rezistența materialelor, mecanisme și organe de mașini.

O analiză succintă clasifică științele în două mari categorii:

- **științe analitice**, care investighează în laborator fenomene ale naturii cărora le stabilesc proprietățile și legile care le guvernează stările și transformările ; aceste fenomene sînt modelate matematic folosind *sisteme de ecuații determinate* cu soluții unice (ex.: mecanica punctului, solidului și fluidelor ; termodinamica ; electricitatea ; chimia ; mecanismele, ș.a.) ;

- **științe sintetice**, care aplică legile științelor analitice la condițiile mediului, cu scopul de a crea instrumente noi în serviciul omului și al societății ; modelele matematice ale acestor științe sînt *sistemele de inecuații* cu un număr infinit de soluții cuprinse într-un **domeniu al soluțiilor posibile**, ceea ce permite investigarea *soluțiilor optime* (ex.: rezistența materialelor; organe de mașini ; mașini-aparate-instalații ; organizarea și conducerea întreprinderilor, ș.a.).

Această clasificare deosebește net cele două categorii de științe nu numai prin preocupări (cunoaștere și sinteză), ci și prin metodele de rezolvare a modelelor lor matematice. Astfel, domeniul infinit al soluțiilor științelor sintetice oferă posibilitatea alegerii subiective a uneia din soluții, creind impresia competenței și a rutinei, deoarece o soluție dintr-o infinitate posibilă poate fi mai convenabilă decît alta.

În ultimele decenii matematica a inițiat *cercetarea operațională*

care asociază sistemelor de inecuații o funcție de scop extremală numită **criteriu optimal**, prin care se asigură unicitatea soluției din punctul de vedere al acestei funcții de scop.

În domeniul organelor de mașini prof. **Octavian Rădulescu** a elaborat *metoda submodelelor plane* pentru calculul optimal al acestora [6].

Se vor aplica modelele matematice ale cercetării operaționale pentru temele din domeniile rezistenței materialelor și organelor de mașini.

13.1. CALCULUL OPTIMAL AL BAREI DREPTÉ SOLICITATE LA ÎNTINDERE CENTRICĂ.

Acest calcul este exemplificat prin următoarea temă : "să se dimensioneze o bară dreaptă de lungime l [cm] solicitată la întindere axială de o forță F [daN], astfel încît să asigure o alungire $\Delta l \geq f$ la preț minim, știind că prețul este proporțional cu aria secțiunii transversale A și cu calitatea materialului σ_a conform relației $(\min)P = \alpha A + \beta \sigma_a$ [lei] și că cel mai bun material are tensiunea admisibilă la întindere statică σ_M [daN/cm²]."

a) **Metoda clasică.** Se folosesc relațiile:

$$A \geq F/\sigma_a \quad (\text{restrictia tensiunii în secțiune})$$

$$\Delta l = Fl/EA \geq f \quad (\text{restrictia rigidității})$$

1) *Soluția 1* : Se alege materialul (σ_{a1}) și se calculează aria secțiunii $A_1 = F/\sigma_{a1}$, după care se verifică dacă $\Delta l = Fl/EA_1 \geq f$. În cazul cînd condiția menționată nu este verificată se pot folosi două căi de rezolvare :

- se păstrează materialul (σ_{a1}) și se mărește secțiunea la valoarea $A_2 > A_1$, după care se verifică dacă $\Delta l = Fl/EA_2 \geq f$, procesul repetîndu-se pînă la verificarea condiției ;

- se alege alt material superior ($\sigma_{a2} > \sigma_{a1}$), se calculează secțiunea $A_2 = F/\sigma_{a2}$ și apoi se verifică condiția $\Delta l = Fl/EA_2 \geq f$, procesul repetîndu-se pînă la verificarea condiției.

Presupunind că nici o soluție nu se verifică, inclusiv cea care folosește cel mai bun material, problema se declară incompatibilă.

2) *Soluția 2:* din relația $\Delta l = Fl/EA = f$, în care se înlocuiește

$A = F/\sigma_a$, rezultă materialul $\sigma_a = Ef/l$; se calculează apoi secțiunea

$A = F/\sigma_a$ și prețul de cost: $P = \alpha A + \beta \sigma_a$

b) **Metoda operațională.** Se formează modelul operațional:

$$\begin{cases} u_1 = A \sigma_a - F \geq 0 \\ u_2 = A - \frac{Fl}{Ef} \leq 0 \\ u_3 = \sigma_a - \sigma_M \leq 0 \\ (\min) P = \alpha A + \beta \sigma_a \end{cases}$$

după care se reprezintă grafic ecuațiile atașate inecuațiilor și funcția de scop fig.13.1) după cum urmează: $u_1 = 0$ ($\sigma_a = F/A$ hiperbolă, limită inferioară a domeniului soluțiilor posibile D). $u_2 = 0$ ($A = Fl/Ef = \text{const.}$, dreaptă verticală - limită dreaptă a domeniului), $u_3 = 0$ ($\sigma_a = \sigma_M = \text{const.}$, dreaptă orizontală - limită superioară a domeniului) și funcția de scop explicitată $\sigma_a = (-\alpha A/\beta) + (P/\beta)$ - familie de drepte înclinate care se apropie de origine când scade prețul.

Soluția optimă aparține punctului P de tangentă dintre dreapta de preț minim și hiperbola u_1 . Coordonatele punctului P se deduc din egalarea coeficienților unghiulari:

$-\alpha/\beta = -F/A^2$ (unde $m_1 = -\alpha/\beta$; $m_2 = d\sigma_a/dA = -F/A$)
obținându-se:

$$P \left[A_P = \sqrt{F\beta/\alpha} ; \sigma_{aP} = F/A_P = \sqrt{F\alpha/\beta} \right]$$

Coordonatele celorlalte puncte sînt următoarele:

$L = u_1 \cap u_3 \rightarrow \sigma_{aL} = \sigma_M ; A = F/\sigma_M$ (soluția de volum minim)

$N = u_1 \cap u_2 \rightarrow A_N = Fl/Ef ; \sigma_{aN} = F/A_N = Ef/l$ (soluția materialului cel mai slab)

Soluția optimă se deduce astfel:

- cînd $\sigma_{aP} > \sigma_{aL}$, prețul minim îl asigură punctul L;

- cînd $\sigma_{aP} < \sigma_{aN}$, prețul minim îl asigură punctul N ;
- cînd $\sigma_P \in [\sigma_{aL}; \sigma_{aN}]$, prețul minim îl asigură punctul P.

Compatibilitatea problemei implică : $\sigma_{aN} \leq \sigma_{aL}$ sau $Ef/l \leq \sigma_M$.

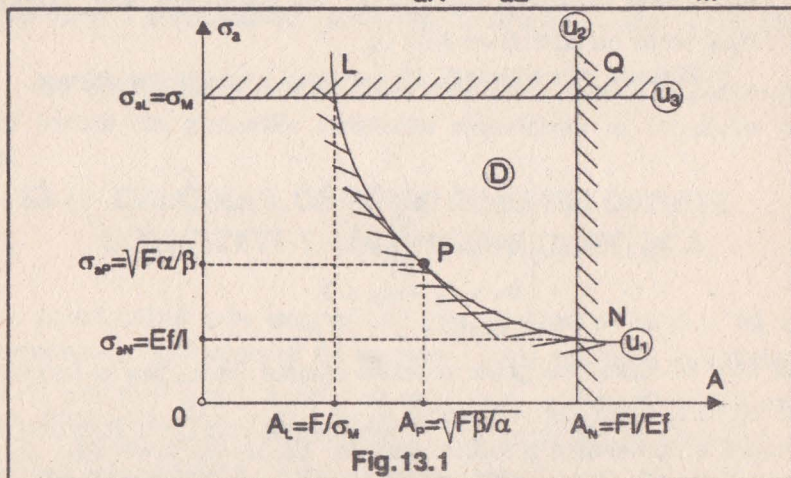


Fig.13.1

Pe baza acestor date a fost elaborat programul de calcul următor în care majusculele subliniate se tastează în modul *grafic* iar (sp) înseamnă *blank*.

```

8 BORDER 0: PAPER 0 : INK 7 : CLS
9 LET ch1=9000: LET apasa=9300 : LET cursor=9400:
  LET udg=9500: DEF FN r(x)=INT x+(x<>INT x)
10 GOSUB udg
15 CLS : PLOT 19,132 : DRAW 216,0 : DRAW 0,- 88 :
  DRAW -216,0 : DRAW 0,88 : PLOT 20,131 : DRAW
  214,0 : DRAW 0,-86 : DRAW -214,0 : DRAW 0,86
20 PRINT PAPER 7, INK 1, AT 1,0 ; "SINTEZE OPTIMALE
  IN CONSTRUCTIA" : AT 3,11 ; "DE MASINI": AT 5,6
  :(29sp) : AT 16,5 ; "M.M.POPOVICI SOFTWARE"
25 PRINT INK 6 ; AT 9,4 : FLASH 1; "REZISTENTA
  MATERIALELOR", FLASH 0 : FOR j =7 TO 0 STEP -1
  : POKE 23606,j : PAUSE 3 : PRINT AT 11,5;
  "CALCULUL BAREI DREPTE": NEXT J : PRINT AT 12,5;
  "LA INTINDERE CENTRICĂ": AT 13,8;"(CU COST
  'MINIM')"
30 GOSUB apasa

```



```

40 BORDER 5: PAPER 5: INK 1: CLS : PRINT AT 2,8 :
   LET a$="DATELE PROBLEMEI":GOSUB cursor
50 INPUT "Introduceti forta F [daN]",F : PRINT AT
   4,1;"Forta axiala F = ";F;" daN"
60 INPUT "Introduceti lungimea barei l [cm]",l:
   PRINT AT 6,1:"Lungimea barei l = ";l;" cm "
70 INPUT "Introduceti deformatia f[cm]",s: PRINT
   AT 8,1,"Deformatia f = " ; s ; " cm "
80 INPUT "Introduceti coef.A,B",a,b:PRINT AT 10,1;"
   Coeficientii pretului : " ; AT 11,4;"A= ";a; AT
   11,16:"B=";b
90 INPUT "Introduceti modulul E [daN/cmC]",E: PRINT
   AT 13,1;"Modulul de elasticitate "; AT 14,4;"E
   = "; " daN/cmC "
100 INPUT "Introduceti ED [daN/cmC]",sigm : PRINT
   AT 16,1;"Tensiunea admisibila maxima";AT 17,4;"
   ED= ";sigm;" daN/cmC"
110 PRINT AT 20,9; INVERSE 1;"CORECT(d/n) ? ":
   PAUSE 0
120 GO TO 40*(INKEY$= "n" ) + 130*(INKEY$ ="d")
130 LET comp=s*E/l: IF comp <= sigm THEN GOSUB chl:
   PRINT AT 1,1; PAPER 7; INK 2;" METODA OPERATION
   ALA ": GO TO 140
135 IF comp > sigm THEN GOSUB chl: PRINT AT 11,15;:
   LET a$= "PROBLEMA INCOMPATIBILA": GOSUB cursor:
   BEEP .02,40: PAUSE 4: BEEP .02,- 40: FOR w=1 TO
   30: BORDER 3: BORDER 5: BORDER 6: BORDER 4:
   BORDER 2: NEXT w
137 PRINT AT 20,6; PAPER 6; INK 2;" Reluati cu alte
   date ": PAUSE 0: GO TO 40
140 LET rad=SQR (F*a/b)
150 IF rad > sigm THEN LET sigo=sigm: LET Ao=F/sigo
160 IF rad >= comp AND rad <= sigm THEN LET sigo=SQR
   (F*a/b): LET Ao=SQR(F*b/a)
170 IF rad < comp THEN LET sigo=comp: LET Ao=s*F/l
180 PRINT OVER 0; AT 3,1: LET a$="Materialul EA =":
   GOSUB cursor: PRINT AT 3,15; PAPER 6; INK 2;
   FN r(sigo);" daN/cmC": PRINT AT 4,1;: LET a$=
   "Aria sectiunii transversale": GOSUB cursor:

```

```

PRINT AT 5,13;"A="; PAPER 6; INK 2;Ao;" cmC"
190 LET Po=a*Ao+b*sigo: PRINT AT 6,1;: LET a$=
"Pretul de cost minim": GOSUB cursor: PRINT AT
7,13;"P="; PAPER 6; INK 2;Po;" lei"
191 LET Ac=F/comp: LET Pc=a*Ac+b*comp: PRINT AT 9,1;
PAPER 7; INK 1;"METODA CLASICA": PRINT AT 10,1;
PAPER 1; INK 7;"Materialul EA= "; FN r(comp);"
daN/cmC"; AT 11,1; PAPER 1; INK 7;"Aria sectiun
ii transversale"; AT 12,10; PAPER 1; INK
7;"Ac=";Ac;" cmC";AT 13,1:"Pretul(3sp)P = ";
Pc;" lei "
192 PRINT AT 15,1; INVERSE 1;"COMPARATII"; INVERSE 0
; AT 16,2;"- raportul volumelor :"; AT 17,4;
"Vc/Vo="; Ac/Ao; AT 18,2;"- raportul
preturilor:"; AT 19,4;"Pc/Po=";Pc/Po
200 PRINT # 0; AT 0,8;"Alte date (d/n) ?": PAUSE 0
210 GO TO 40*(INKEY$ ="d")+220*(INKEY$="n")
220 CLS : PLOT 19,132: DRAW 216,0: DRAW 0,-88: PLOT
20,131: DRAW 214,0: DRAW 0,-86: DRAW -214,0:
DRAW 0,86
230 PRINT INK 6; AT 9,4; FLASH 1;"REZISTENTA MATERI
ALELOR"; FLASH 0: FOR j=7 TO 0 STEP -1:
POKE 23606,j: PAUSE 3: PRINT AT 11,5;"CALCULUL
BAREI DREPTE": NEXT j:PRINT AT 12,5;" LA INTINDE
RE CENTRICA "; AT 13,8;"(CU' COST MINIM) "
250 PRINT PAPER 7; INK 1; AT 1,0;"SINTEZE OPTIMALE
IN CONSTRUCTIA"; AT 3,11; "DE MASINI"; AT 5,6;
"(23sp)"; AT 16,5;" M.M. POPOVICI SOFTWARE "
260 PRINT PAPER 4; INK 6; AT 20,9; "S F I R S I T !"
8999 STOP
9000 BORDER 2: PAPER 1: INK 6: CLS : PLOT 0,0: DRAW
255,0: DRAW 0,175: DRAW -255,0 : DRAW 0,-175:
PLOT 2,2: DRAW 251,0: DRAW 0,171: DRAW -251,0:
DRAW 0,-171: RETURN
9300 LET b$="": IF b$="" THEN FOR i=0 TO 31: LET b$=
b$+CHR$ 17+CHR$ (i-8*INT (i/8))+ " ": NEXT i
9301 LET u=1: POKE 23569,u: PRINT # 0;"(4sp) APASATI
O TASTA OARECARE (4sp)"
9302 POKE 23659,2

```

```

9303 PRINT # 0; AT u,0; OVER u; INK 9;b$: IF IN 254=
191 THEN LET b$=b$(4 TO)+b$(TO 3): GO TO 9303
9304 OVER 0: PRINT # 0; AT u,0; TAB 31;" ": RETURN
9400 INK 9: FOR i= TO LEN A$
9401 IF a$(i)=CHR$ 13 THEN PRINT a$(i);: NEXT i
9402 FOR k=0 TO 4: PRINT PAPER k;a$(i); CHR$ 8;: NEXT
k: PRINT a$(i);: NEXT i: RETURN
9500 RESTORE 9501: FOR q=0 TO 39: READ a: POKE
65368+q,A: NEXT q
9501 DATA 0,0,0,56,72,72,60,0,0,28,34,60,34,34,76,0
9502 DATA 96,16,32,120,0,0,0,0,0,0,34,54,42,34,34,34,
0,62,64,64,68,68,56,0
9503 RETURN

```

Pentru $F=10^6$ daN, $l=150$ cm. $f=0,1$ cm, $\alpha=0,5$, $\beta=0,02$, $E=2,1 \cdot 10^6$ daN/cm² și $\sigma_M=6500$ daN/cm² se obțin următoarele rezultate :

- metoda clasică: materialul $\sigma_a=1400$ daN/cm²; $A_c=714,28$ cm²; $P_c=385,14$ lei ;

- metoda operatională: materialul $\sigma_a=5000$ daN/cm² ; $A_o=200$ cm² ; $P_o=200$ lei

Raportul volumelor de material $V_c/V_o=3,57$ si raportul preturilor
 $P_c/P_o=1,92$.

13.2. CINEMATICA ANALITICĂ A MECANISMULUI MANIVELĂ-PISTON

În construcția de mașini mecanismul plan manivelă-piston este larg folosit deoarece transformă mișcarea de rotație a manivelei în mișcare de translație oscilantă a pistonului și invers. El este întâlnit la compresoare, pompe, mașini de mortezat și debitat, precum și în construcția motoarelor cu ardere internă. Schema cinematică a mecanismului și diagramele sale cinematice sînt prezentate în fig.13.2, a, b, iar formulele cu care se determină deplasarea pistonului, viteza și accelerația acestuia sînt următoarele :

$$s_B = r \left[(1 - \cos \varphi_1) + \frac{\lambda}{4} (1 - \cos 2\varphi_1) \right] ;$$

$$v_B = r \omega_1 \left(\sin \varphi_1 + \frac{\lambda}{2} \sin 2\varphi_1 \right) ;$$

$$a_B = r \omega_1^2 (\cos \varphi_1 + \cos 2\varphi_1)$$

Viteza și accelerația butonului de manivelă (punctul A) sînt:

$$v_A = r \omega_1 ; \quad a_A = r \omega_1^2$$

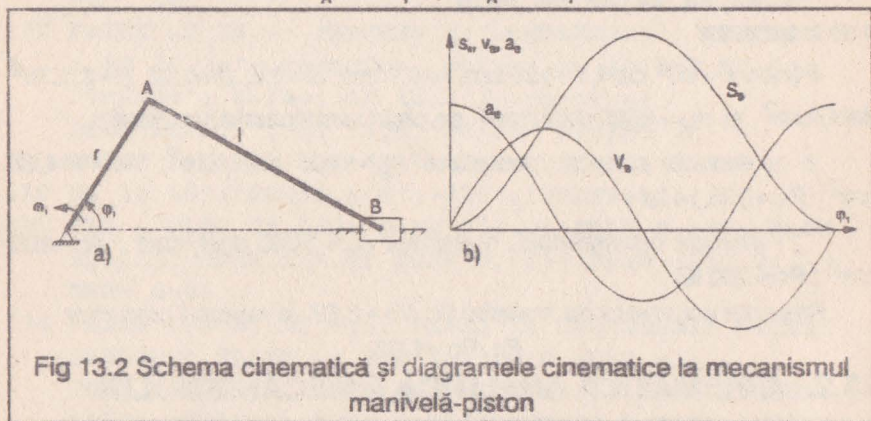


Fig 13.2 Schema cinematică și diagramele cinematice la mecanismul manivelă-piston

În aceste relații s-au notat : r - lungimea manivelei, l - lungimea bielei, φ_1 - unghiul manivelei, ω_1 - viteza unghiulară a manivelei ; $\lambda = r/l$.

```

9 LET ch1=9000: LET ch2=9100: LET apasa=9200: LET
  udg=9300: GOSUB udg
10 BORDER 1: PAPER 0: INK 0: CLS
20 FOR n=7 TO 38 STEP 8: FOR m=7 TO 0 STEP -1:
  BORDER m: PAPER m: PRINT AT 0, (n-m); " "; AT 21,
  31-(n-m); " "; AT 1, (n-m); " "; AT 20, 31-(n-m);
  " "; AT 2, (n-m); " "; AT 19, 31-(n-m); " "; NEXT m:
  NEXT n
30 BRIGHT 1: PAPER 7: INK 7: PRINT AT 3, 0;
  "(32 sp)"; AT 18, 0; "(32sp)": REM sp=blanc
40 PLOT 0, 175: DRAW 255, 0: DRAW 0, -38: DRAW 255, 0:

```

```

DRAW 0,38
50 PLOT 2,173: DRAW 251,0: DRAW 0,-34: DRAW -251,0:
DRAW 0,34
60 PRINT AT 2,1; FLASH 1; PAPER 0; BRIGHT 1;
INK 4 ;"(30sp)"; AT 2,3:"MECANISMUL MANIVELA-
PISTON"
90 PRINT AT 7,10; PAPER 0; INK 6;"M.M.POPOVICI"; AT
10,10; INK 7;"(2sp)SOFTWARE(2sp)"
95 PLOT 0,0: DRAW 255,0: DRAW 0,175: DRAW -255,0:
DRAW 0,-175
100 FOR i=7 TO 0 STEP -1: POKE 23606,i : PAUSE 3:
PRINT AT 14,6; PAPER 6; INK 1;"CINEMATICA ANALI
TICA": NEXT i: GOSUB apasa
110 BORDER 2: PAPER 0: INK 7: CLS
120 PRINT AT 1,8: INVERSE 1;"DATELE PROBLEMEI":
INPUT " Introduceti r [m]",r: PRINT AT 3,1;
"Lungimea manivelei r=";r;" m"
125 INPUT "Introduceti l[m]",l: PRINT AT 5,1;"Lungi
mea bielei l=";l;" m"
140 INPUT "INTRODUCETI AB[gr]",fig: PRINT AT 7,1;
"Unghiul manivelei AB=";fig;" gr": REM
majusculele subliniate se tasteaza in modul
grafic
150 INPUT "Introduceti CB[l/s]",ol: PRINT AT 9,1;
"Viteza unghiulara CB=";ol;" l/s"
160 PRINT AT 18,6; PAPER 6; INK 1:" DATE CORECTE
(d/n)?" : PAUSE 0
170 GO TO 110*(INKEYS="n")+180*(INKEYS="d")
180 LET lb=r/l: LET fir=fig*PI/180
190 LET vA=r*ol: LET aA=r*ol*ol
200 LET a=1-COS fir: LET b=.25*((1-COS(2*fir))*lb^
205 LET sB=r*(a+b)
210 LET vB=r*ol*(SIN fir+.5*lb*SIN(2*fir))
220 LET aB=r*ol*ol*(COS fir+lb*COS(2*fir))
260 GOSUB chl: PRINT AT 2,6;"MARIMILE CINEMATICE"
270 PRINT AT 4,2; PAPER 3; INK 6;"a) MANIVELA";
AT 6,3; PAPER 6; INK 0; "vf=";vA;" m/s"; AT 7,3;
"af=";aA;" m/sD"
280 PRINT AT 9,2; PAPER 3; INK 6;"b)PISTONUL";

```

```

AT 13,3; PAPER 6; INK 0;"sE=";sB; AT 13,15;
" m"; AT 14,3;"vE=";vB; AT 14,15;" m/s"; AT
15,3;"aE=";aB; AT 15,13;" m/sD"
290 PRINT # 1; AT 1,7; PAPER 1; INK 7; "ALTE VALORI
(d/n)?: PAUSE 0
300 GO TO 110*(INKEY$="d")+ 310*(INKEY$="n")
310 PRINT # 1; AT 1,0;"(32sp)": PRINT # 1; AT 1,3 ;
PAPER 3; INK 7:"UN CICLU CINEMATIC (d/n) ?":
PAUSE 0
320 GO TO 330*(INKEY$="d")+500*(INKEY$="n")
330 CLS : INPUT "Introduceti pasul unghiular [gr]",
pas
335 GOSUB ch1: PRINT AT 2,6:" MARIMILE CINEMATICE"
340 LET fig=0: LET w=fig: FOR w=fig TO 360 STEP pas
380 LET fir=fig*PI/180
400 LET a=1-COS fir: LET b=.25*((1-COS(2*fir))*lb)
405 LET sB=r*(a+b)
410 LET vB=r*ol*SIN fir+.5*lb*SIN (2*fir))
420 LET aB=r*ol*ol*(COS fir+lb*COS(2*fir))
460 PRINT AT 11,11;"aB="; AT 11,14;w;" gr"
465 IF w=360 THEN PRINT AT 13,6;"0(9sp)"; AT 14,6;"0
(9sp)": GO SUB apasa: GO TO 500
480 PRINT AT 9,2; PAPER 3; INK 6;"PISTONUL"; AT 13,3
; PAPER 6; INK 0;"sE=";sB; AT 13,15:" m"; AT
14,3;"vE=";vB; AT 14,15;" m/s"; AT 15,3;"aE=";
aB; AT 15,15;" m/sD"
490 LET fig=fig+pas
495 GOSUB apasa
496 NEXT w
500 GOSUB ch2: PRINT AT 11,4; PAPER 2; INK 4;"DORITI
DIAGrameLE (d/n)?: PAUSE 0
510 GO TO 600*(INKEY$="d")+520*(INKEY$="n")
520 PRINT AT 11,5; PAPER 7; INK 1;"M.M. POPOVICI
SOFTWARE"
530 FOR i=7 TO 0 STEP -1: POKE 23606,i: PAUSE 3:
PRINT AT 14,6: PAPER 6; INK 1; "CINEMATICA
ANALITICA": NEXT i: STOP
600 BORDER 2: PAPER 0: INK 7: CLS : PRINT AT 2,1:
INVERSE 1;"DIAGrameLE MISCarii PISTONULUI";

```

```

INVERSE 0:PRINT # 1; AT 1,3;"Pasul unghiular
pas=10 [gr]": PAUSE 0: LET j=0: LET pas=10: PLOT
0,0: DRAW 255,0: DRAW 0,175: DRAW -255,0: DRAW
0,-175: PRINT AT 6,1;"sE"; AT 6,10;"vE"; AT
6,20;"aE"; AT 11,10;"AB(8sp)AB(7sp)AB"
601 FOR f=25 TO 175 STEP 75: PLOT f,136: DRAW 0,-50:
DRAW 50,0: PLOT f-2,132: DRAW 2,2: DRAW 2,-2:
PLOT f+52,84: DRAW 2,2: DRAW -2,2: NEXT f
602 LET fig=0: LET w=fig: FOR w=fig TO 360 STEP pas:
LET fir=fig*PI/180
604 LET a=1-COS fir: LET b=.25*((1-COS(2*fir))*1b)
605 LET sB=r*(a+b)
610 LET vB=r*o1*(SIN fir+.5*1b*SIN(2*fir))
620 LET aB=r*o1*o1*(COS fir+1b*COS(2*fir))
630 IF w=360 THEN PRINT AT 18,4;"360"; AT 17,13;
"0(10sp)"; AT 18,13;"0(8sp)": GOSUB apasa:
GO TO 710
635 PLOT 75,10: DRAW 0,30
640 PLOT 25+j*1.5,90+sB*10: PRINT AT 18,1;"AB=";
j*10; AT 17,10;"sE=";sB; AT 17,25;" m"; AT
18,10;" vE=";vB; AT 18,25;" m/s"; AT 19,10;
"aE=";aB; AT 19,25;" m/sD": PLOT 100+j*1.5,
90+vB: PLOT 175+j*1.5,85+aB/100: LET j=j+1
690 LET fig=fig+pas
700 NEXT w
710 GOSUB ch2: PRINT AT 11,9;"RELUATI (d/n)?" :
PAUSE 0
720 GO TO 110*(INKEYS$="d")+730*(INKEYS$="n")
730 GOSUB ch2: GO TO 520
8999 STOP
9000 BORDER 5: PAPER 6: INK 2: CLS
9001 PLOT 15,0: DRAW 255,0: DRAW 15,15: DRAW 0,145:
DRAW -15,15: DRAW -225,0: DRAW -15,-15:
DRAW 0,-145: DRAW 15,-15: DRAW 0,15: DRAW -15,0:
PLOT 255,15: DRAW -15,0: DRAW 0,-15: PLOT
255,160: DRAW -15,0: DRAW 0,15:PLOT 0,160:DRAW
15,0: DRAW 0,15
9002 PRINT AT 0,0: OVER 1; PAPER 5;"(2sp)": AT 0,30 :
"(2sp)"; AT 1,0;"(2sp)"; AT 1,30:"(2sp)":

```

```

AT 20,0;"(2sp)"; AT 20,30:"(2sp)"; AT 21,0 ;
"(2sp); AT 21,30;"(2sp)": RETURN
9100 LET t$="(15sp)": PAPER 6: BRIGHT 1: INK 0:
BORDER 7: CLS : PLOT 0,0: DRAW 255, 0:
DRAW 0,175: DRAW -255,0: DRAW 0,-175:
PRINT AT 1,2; PAPER 5;t$;t$(2 TO):
FOR i=2 TO 20: PRINT AT i,1; PAPER 7;
t$(2 TO); AT i-1,30; PAPER 5;" ": NEXT i
9200 LET a$="(4sp)APASATI O TASTA OARECARE(4sp)"
9201 PRINT # 1; AT 1,0; a$
9202 LET a$=a$(32)+a$(TO 31)
9203 PAUSE 5: IF INKEY$="" THEN GO TO 9201
9204 RETURN
9300 RESTORE 9301: FOR i=1 TO 47: READ a:
POKE 65368 +i,a: NEXT i
9301 DATA 0,12,82,82,60,16,16,16,0,0,0,0,192,64,
64,224
9302 DATA 0,68,130,130,146,146,108,0,96,16,32,120,
0,0,0,0
9303 DATA 0,0,0,224,144,224,144,224,0,0,0,240,
144,240,144,144
9304 RETURN

```

Acest program definește noi udg-uri, calculează mărimile cinematice pentru o singură poziție cerută sau pentru un cliclu cinematic și trasează diagramele mișcării, afișând rezultatele numerice. De exemplu pentru $r=0,5$ m, $l=2,5$ m și $\omega_1=42$ 1/s, rezultă datele din tabelul 13.1 (pentru un pas unghiular de 15°).

Tabelul 13.1

Unghiul	s_B [m]	v_B [m/s]	a_B [m/s ²]
0	0	0	1058,4
15	0,020386451	6,4851999	1004,7135
30	0,07948729	12,318653	852,03441
45	0,1714466	16,949242	623,66818
60	0,2875466	20,005187	352,86818
75	0,4171411	21,334442	75,511516
90	0,5572411	21,33442	-176,4516
105	0,6750601	19,234442	-381,0452

120	0,7875601	16,367882	-529,2452
135	0,8785533	12,749242	-623,6681
150	0,9455127	8,6812467	-675,6344
165	0,9863122	4,3852467	-699,1797
180	1,9863122	0,3852467	-705,6797
195	0,9863122	-4,385299	-699,1797
210	0,9455127	-8,681346	-675,6344
225	0,8785533	-12,74924	-623,6681
240	0,7875533	-16,36788	-529,2681
255	0,6760601	-19,23444	-381,0452
270	0,5560601	-21,33444	-176,4452
285	0,41722411	-21,33444	75,5115
300	0,2875411	-20,00518	352,81516
315	0,1714466	-16,94924	623,66818
330	0,7948729	-12,31865	852,03441
345	0,2038645	-6,485265	1004,7135
360	0	0	1058,4135

Se menționează că programul realizează o trasare corectă a diagramelor numai pentru $\lambda = r/l = 1/4$ și $\omega_1 = 40 \dots 70$ 1/s.

13.3. ECHILIBRAREA STATICĂ A MECANISMULUI MANIVELĂ-PISTON

În timpul funcționării mașinii, datorită mișcării elementelor cu viteze variabile se produc forțe de inerție care reprezintă reacția cinetică a masei elementului la accelerația ce i se imprimă odată cu mișcarea sa. Forțele de inerție produc solicitări suplimentare în elemente și cuple care, în cazul mașinilor rapide, sînt superioare solicitărilor date de forțele specifice procesului de lucru al mașinilor. Pe de altă parte, deoarece forțele de inerție variază periodic în timp, alături de solicitările la oboseală pe care le cauzează, ele provoacă și vibrații ale sistemului asupra căruia acționează. Vibrațiile pot deveni deosebit de periculoase în situația cînd frecvența lor este egală cu frecvența de oscilație proprie a unui element al mașinii, întrucît este posibilă apariția fenomenului de *rezonanță mecanică* concretizat printr-o creștere considerabilă a solicitărilor, depășindu-se

limitele admisibile ale tensiunilor pentru ce au fost dimensionate.

Înlăturarea forțelor de inerție este imposibilă, deoarece nu se pot construi elemente fără masă, dar anihilarea lor prin alte forțe de inerție este realizabilă. În acest scop se pot introduce mase adiționale în sistem numite **contragreutăți**, operație cunoscută în tehnică sub denumirea de *echilibrarea maselor mecanismului*. Un mecanism este echilibrat dacă

torsorul forțelor de inerție este nul : $\tau_1(\bar{F}_1, \bar{M}_1) = \bar{0}$; în situația când numai

forța de inerție rezultantă este nulă ($\bar{F}_1 = \bar{0}$) se realizează **echilibrarea statică**.

Se consideră un mecanism manivelă-piston (fig.13.3, a) ale cărui elemente cinematice au lungimile r, l [m] și masele m_1, m_2, m_3 [kg] presupuse concentrate în centrele lor de greutate $G_1, G_2, G_3 = B$, poziționate față de articulația anterioară prin distanțele s_1 și s_2 [m]. La aceste date de intrare în program se

mai adaugă distanțele ρ_I și ρ_{II} de amplasare a maselor de echilibrare m_1 și m_2 (prima în prelungirea manivelei, iar a doua în prelungirea bieiei fig.13.3,b).

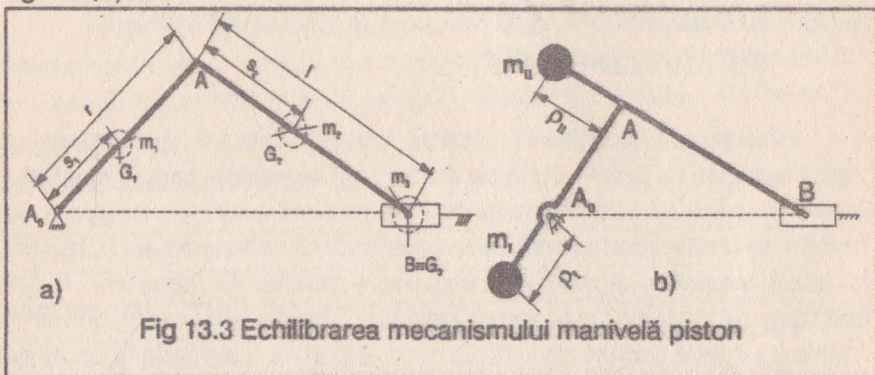


Fig 13.3 Echilibrarea mecanismului manivelă piston

Relațiile de calcul care se aplică sînt următoarele :

- repartizarea masei m_1 în articulațiile A_0 și A : $m_{1A_0} = m_1(r-s_1)/r$;
 $m_{1A} = m_1 s_1 / r$
- repartizarea masei m_2 în articulațiile A și B : $m_{2A} = m_2(l-s_2)/l$;
 $m_{2B} = m_2 s_2 / l$

Masa m_3 se repartizează integral în articulația B : $m_{3B} = m_3$. În acest fel în articulațiile A_0 , A și B sînt amplasate masele

$$m_{A0} = m_{1A0} ; m_A = m_{1A} + m_{2A} \text{ (masă rotativă)} ; m_B = m_{2B} + m_3 \text{ (masă translantă)}$$

Dintre aceste mase urmează a fi echilibrate numai masele din articulațiile mobile A și B, deoarece masa din A_0 fiind fixă nu influențează mișcarea mecanismului. Rezultă masele de echilibrare :

$$m_{II} = m_B l / \rho_{II} \text{ [kg]} \text{ și } m_I = (m_A + m_B + m_{II}) r / \rho_I \text{ [kg]}.$$

În acest fel masa întregului sistem - inclusiv masele adiționale - este concentrată în articulația fixă A_0 care reprezintă centrul de greutate al mecanismului. Masa totală a mecanismului echilibrat este:

$$m_e = m_{1A0} + m_A + m_B + m_{II} + m_I \text{ [kg]}$$

Cu aceste date rezultă următorul program pe calculator :

```

8 LET apasa=9000: LET udg=9100: LET c=9400: LET
bord=9500: LET box=9600: LET x$+"32sp)": DEF FN
w(a,b,c,d)=100: REM (sp)=blanc
9 GOSUB udg: GOSUB bord
10 BORDER 2: PAPER 0: INK 7: CLS : FOR i=0 TO 10
STEP 2: PLOT 32-i,60-i : DRAW 186+2*i,0: DRAW
0,75+2*i: DRAW -186-2*i,0: DRAW 0,-75-2*i:
NEXT i
13 PRINT AT 6,8;"M.M. POPOVICI "; AT 7,8;
"SOFTWARE(3sp)1992"; AT 10,5;"ECHILIBRAREA(2sp)
STATICA"; AT 11,5;"TOTALA A MECANISMULUI";AT
12,7 ;"MANIVELA(3sp)PISTON"
15 PLOT 0,0: DRAW 255,0: DRAW 0,175: DRAW -255,0:
DRAW 0,-175
20 GOSUB apasa: RANDOMIZE USR 60000: GOSUB c
30 BORDER 7: PAPER 7: INK 0: CLS : FOR y=175 TO
170 STEP -1: PLOT 0,y: INK 2: DRAW 255,0: NEXT
y: FOR y =166 TO 163 STEP -1: INK 4: PLOT 0,y:
DRAW 255,0: NEXT y: PLOT 0,159: INK 0: DRAW
255,0
40 PRINT AT 3,8; PAPER 1; INK 6;"DATELE PROBLEMEI"
45 FOR y=20 TO 15 STEP -1: PLOT 0,y: DRAW INK 4;
255,0: NEXT y: FOR y=13 TO 8 STEP -1: PLOT 0,y:
DRAW INK 2;255,0: NEXT y

```

```

50 INPUT "Introduceti r [m]",r: PRINT AT 5,1;
   "1)Lungimea manivelei r=";r;" m"
60 INPUT "Introduceti l [m]",l: PRINT AT 6,1;
   "2)Lungimea bielei(4sp)l=";l;" m"
70 INPUT "Introduceti sA [m]",s1: PRINT AT 7,1;
   "3)Pozitia c.d.g.(5sp)sA=";s1;" m":REM
   literele majuscule se tasteaza in modul grafic
80 INPUT "Introduceti sB[m]";s2: PRINT AT 8,1;
   "4)Pozitia c.d.g.(5sp)sB=" ;s2;" m"
90 INPUT "Introduceti mA [kg]";m1: PRINT AT 9,1;
   "5)Masa manivelei(5sp)mA =";m1;" kg"
100 INPUT "Introduceti mB[kg]",m2: PRINT AT 10,1;
   "6)Masa bielei(8sp)mB=";m2;" kg"
110 INPUT "Introduceti mC [kg] ",m3: PRINT AT 11,1;
   "7)Masa pistonului (4sp)mC=";m3;" kg"
120 INPUT "Introduceti DH [m]",r1: PRINT AT 12,1;
   "8)Distanta(11sp)DH=";r;" m"
130 INPUT "Introduceti DI [m]",r2: PRINT AT 13,1;
   "9)Distanta(11sp)DI=";r2;" m"
200 PRINT # 1; AT 1,9;"CORECT (d/n) ?": PAUSE 0
210 GO TO 30*(INKEY$="n")+215*(INKEY$="d")
215 GOSUB c
220 BORDER 5: PAPER 6: INK 1: CLS : LET m1A0=
   m1*(r-s1)/r: LET m1A=m1*s1/r: LET m2A=
   m2*(1-s2)/l: LET m2B=m2*s2/l: LET m3B=m3: LET
   mA0=m1A0: LET mA=m1A+m2A: LET mB=m2B+m3: LET
   mII=mB*l/r2: LET mI=(mA+mB+mII)*r/r1: LET
   mPA=mA+mB+mII: LET me=m1A0+mPA+mI
230 BORDER 5: PAPER 1: INK 0: CLS:LET
   i$="0102301761":GOSUB box: PRINT AT 3,11; PAPER
   3; INK 7;"REZULTATE": PAPER 6: INK 1: PRINT AT
   7,5 ;"mAEG=";m1A0; AT 7,21;" kg"; AT 8,5 ;
   "mAE=";m1A; AT 8,21;" kg"; AT 9,5;"mBE=";m2A; AT
   9,21;" kg"; AT 10,5;"mBF=";m2B; AT 10,21;" kg";
   AT 11,5 ;"mCF=";m3B; AT 11,21;" kg"; AT 12,5;
   "mEG=";mA0; AT 12,2;" kg"; AT 13,5;"mE=";mA; AT
   13,21;" kg"; AT 14,5;"mF=";mB; AT 14,21 " kg";
   AT 15,5;"mH=";mI; AT 15,21;" kg"; AT 16,5;
   "mI=";mII; AT 16,21;" kg"; AT 17,5; "mJ=";me; AT

```

```

17,21;" kg"
240 PRINT AT 20,9; PAPER 4 ; INK 6;"ALTE DATE
(d/n)?" : PAUSE 0
250 GO TO 30*(INKEY$="d")+260*(INKEY$="n")
260 RANDOMIZE USR 60000
270 BORDER 5: PAPER : INK 7: CLS : FOR i= 0 TO 10
STEP 2: PLOT 32-i,60-i: DRAW 186+2*i,0: DRAW
0,75+2*i: DRAW -186-2*i,0: DRAW 0,-75-2*i: NEXT i
280 PRINT AT 6,8;"M.M. POPOVICI"; AT 7,8;
"SOFTWARE(3sp)1992"; AT 10,5;"ECHILIBRAREA
(2sp)STATICA"; AT 11,5;"TOTALA A MECANISMULUI";
AT 12,7;"MANIVELA(3sp)PISTON"
8999 STOP
9000 FOR k=19 TO 20: PRINT PAPER 2; AT k,3;"(25▣)" :
NEXT k: PRINT AT 18,3;"APASATI ORICE TASTA
DORITI": PAUSE 0: RETURN : REM ▣ semnul se
obține cu tasta 3 in modul grafic
9100 RESTORE 9101: FOR f=USR "a" TO USR "j" a: POKE
f,a: NEXT f
9101 DATA 0,0,0,0,64,192,64,224,0,0,0,0,96,16,31,120
9102 DATA 0,0,0,42,72,16,72,48,0,0,0,2,255,2,0,0
9103 DATA 0,0,0,240,144,240,144,144,0,0,0,224,144,
224,144,224
9104 DATA 0,0,0,0,96,144,144,96,0,0,0,0,224,64,64,224
9105 DATA 0,0,0,0,238,68,68,238,0,0,112,136,136,240,
128,248
9106 RETURN
9400 LET z=0: LET u=1: OVER u: INVERSE u: FOR j=1 TO
7 STEP 2: BORDER j: INPUT ; :INK 9: FOR i=0 TO
87 STEP 8: PLOT i,i: DRAW PAPER j;255-2*i,z:
DRAW PAPER j;z,175-2*i: DRAW PAPER j;2*i-255,z:
DRAW PAPER j;2*i-175: NEXT i: NEXT j: PAPER 5:
INK 9: BORDER 6: INVERSE z: CLS: RETURN
9500 RESTORE 9501: FOR n=1 TO 17: READ a: POKE
60000+n,a: NEXT n
9501 DATA 17,232,3,62,255,71,211,254,175,16,251,27,
122,179,32,243,201
9502 RETURN
9600 LET xo=VAL i$(1 TO 2): LET yo=VAL i$(3 TO 4):

```

```

LET x1=VAL i$(5 TO 6): LET y1=VAL i$(7 TO 8)
9601 PAPER VAL i$(9): INK VAL i$(10)
9602 LET i=0
9603 IF i THEN LET x=FN w(xo,yo,x1,y1)
9604 LET i$=x$(TO x1): FOR i=yo TO yo+y1-1: PRINT AT
i,xo;i$: NEXT i
9605 PLOT 8*xo+1,174-8*yo: DRAW 8*x1-3,0: DRAW
0,3-y1*8: DRAW 3-x1*8,0: DRAW 0,y1*8-3
9606 RETURN

```

Folosind următoarele date de intrare: $r=0,12$ m; $l=0,6$ m; $s_1=0,1$ m; $s_2=0,2$ m; $m_1=2$ kg; $m_2=8$ kg; $m_3=10$ kg; $\rho_I=0,08$ m și $\rho_{II}=0,16$ m, se obțin rezultatele:

$m_{1A0}=1$ kg; $m_{1A}=1$ kg; $m_{2A}=5,37(3)$ kg; $m_{2B}=2,66(6)$ kg;
 $m_{3B}=10$ kg;

$m_{A0}=1$ kg; $m_A=6,33(3)$ kg; $m_B=12,6(6)$ kg; $m_I=166,25$ kg și
 $m_{II}=47,5$ kg

Se constată că a rezultat o masă echilibrată $m_e=233,75$ kg cu valoare mare, ceea ce îngreunează construcția. De asemenea, adăugarea masei adiționale m_{II} are ca efect mărirea masei bielei și implicit a momentului ei de inerție masic, ceea ce conduce la creșterea dezechilibrului datorită momentului rezultat al forțelor de inerție pe acest element. Drept consecință se va renunța la echilibrarea statică totală a forțelor de inerție și se aplică varianta de echilibrare parțială, prin utilizarea unei singure mase adiționale amplasată în prelungirea manivelei A_0 A. O asemenea soluție este folosită la motoarele cu ardere internă și la compresoare.

13.4. SINTEZA OPTIMALĂ A ARCULUI BARĂ DE TORSIUNE

Bara de torsiune (fig.13.4) este o bară dreaptă, de regulă de secțiune circulară, utilizată ca arc amortizor de șocuri pe vehicule rutiere și de cale ferată. De cele mai multe ori bara este încastrată la unul din capete într-un dispozitiv de prindere, iar la capătul liber este asamblat prin presare cu un levier asupra căruia acționează forța variabilă $F \in (F_m -$

forța minimă: F_M - forța maximă) ce produce momentul de răsucire $M_t = Fa \in [M_{tm} - \text{momentul minim}; M_{tM} - \text{momentul maxim}]$.

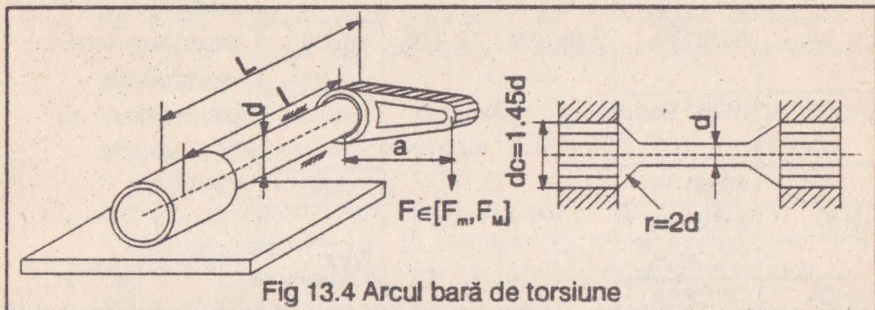


Fig 13.4 Arcul bară de torsiune

Aplicînd metoda operațională rezultă următorul algoritm de calcul:

Date: M_{tm} , M_{tM} [daNcm] ; materialul (G, σ_{+1} - tensiunea la rupere statică [daN/cm²]; unghiul de răsucire specific admisibil θ_a [rad/cm]; L [cm] - lungimea maximă permisă de gabarit.

Rezolvare: 1) Coeficientul de asimetrie al ciclului la oboseală : $R = M_{tm}/M_{tM}$

2) Tensiunea tangențială admisibilă la oboseală: $\tau_{aR} = 0,9\sigma_{+1}/(3-R)$

3) Diametrul barei de torsiune: $d = \sqrt[3]{16 M_{tM}/\pi \tau_{aR}}$ [cm]

4) Lungimea barei de torsiune : $l = (\text{min.}) \{ \pi \theta_a Gd^4/32M_{tM}; L \}$ [cm]

5) Raza de racordare: $r = 2d$ [cm]

6) Diametrul de calare $d_c = 1,45d$ [cm].

Caracteristicile principalelor oțeluri de arc sînt prezentate în tabelul 13.1 :

Tabelul 13.1

Simbol	Denumire	E [daN/cm ²]	G [daN/cm ²]	σ_{+1} [daN/cm ²]	Utilizări
-	62Si2W			18640	Arcuri puternic solicitate

Arc 2A	51Vcr11A	2.06.10 ⁶	7.8.10 ⁵	13200	Arcuri cu foi, elicoidale, bară de torsione
Arc 3A	60Si15A			14700	Arcuri lamelare, cu foi,elicoidale
Arc 4A	56Si17A				Arcuri cu foi,elicoidale
	40Si17A			11800	(CFR)
Arc 5	51Si17A				
	OLC65A			9800	Arcuri lamelare, cu foi,elicoidale
Arc 6A	OLC55A				
Arc 10A	OLC75A			10800	Arcuri lamelare, cu foi, elicoidale, spirale
Arc 7	OLC85A			11300	idem
RS				7000	
RM	OLC	8000			
RR		10000			

Programul de calcul este următorul :

```

9 LET ch1=9000: LET ch2=9100: LET apasa=9200: LET
cortina=9300: LET udg=9400: LET box=9600: LET
x$="32sp": DEF FN w(a,b,c,d)=100: DEF FN r(x)=
INT x+(x<>INT x)
10 BORDER 7: PAPER 7: INK 0: CLS
15 GOSUB udg
20 FOR f=g TO 16: PRINT PAPER 5; INK 2; AT f,6;
"(22sp)": NEXT f: REM (sp)=blanc
30 FOR f=7 TO 17: PRINT PAPER 1; INK 1; AT f,5;" ":
NEXT f: PRINT PAPER 1; INK 1; AT 17,6;"(19sp)"
40 PRINT PAPER 4; INK 6; AT 7,6;"M.M.POPOVICI
SOFTWARE"; AT 9,15;"1993"; AT 13,6; FLASH 1;
INVERSE 1"ARCUL BARA DE TORSIUNE": GOSUB apasa
50 BORDER 5: PAPER 5: INK 1: CLS
60 PLOT 0,8: DRAW 255,0: DRAW 0,167: DRAW -255,0:
DRAW 0,-167: LET i$="0101251800": GOSUB box:
LET i$="0202251626":GOSUB box

```



```

70 PRINT AT 3,5;"DATELE PROBLEMEI"
80 INPUT "Introduceti CD[daN/cm]",mtM: PRINT
  AT 5,3;"1)Momentul maxim"; AT 6,4"CD=";mtM;
  " daNcm": REM majusculele subliniate se tasteaza
  in modul grafic
90 INPUT "Introduceti AB[daNcm]",mtmin: PRINT
  AT 7,3;"2)Momentul minim"; AT 8,4:"AB=" mtmin;
  " daNcm"
100 INPUT "Introduceti G [daN/cmI]",g: PRINT AT 9,3;
  "3)Materialul";AT 10,4;"G=";g ;" daN/cmI"
110 INPUT "Introduceti EF [daN/cmI]",sig1: PRINT
  AT 11,4;"EF=";sig1;" daN/cmI"
120 INPUT "Introduceti GH [rad/cm]",teta: PRINT AT
  12,3;"4)Deformatia admisibila"; AT 13,4;
  "GH=";teta;" rad/cm"
130 INPUT "Introduceti L[cm]",l: PRINT AT 14,3;
  "5)Lungimea maxima"; AT 15,4;"L=";l;" cm"
140 LET i$="0717150362": GOSUB box: PRINT AT 18,8;
  "CORECT (d/n)?: PAUSE 0
150 GO TO 50*(INKEY$="n")+160*(INKEY$="d")
160 LET k=6: GOSUB cortina
170 GOSUB chl: LET r=mtmin/mtM:LET tau=
  .9*sig1/(3-r): LET d=(16*mtM/(PI*tau))^(1/3):
  LET l1=(PI*g*teta*FN r(d)^4/(32*mtM)): LET l2=l
180 IF l1 <= l2 THEN LET lung=l1
190 IF l2 <= l1 THEN LET lung=l2
200 LET i$="1002110317": GOSUB box: PRINT AT 3,11;
  "REZULTATE"
210 PAPER 6: INK 0: PRINT AT 5,2;"Diametrul barei";
  AT 7,4;"d="; FN r(d); AT 7,18;" cm"; AT 9,2;
  "Lungimea barei"; AT 11,4;"l=";FN r(lung);
  AT 11,18;" cm"; AT 13,2;"Raza de racordare";
  AT 15,4;"r=";2*FN r(d); AT 15,18;" cm";
  AT 17,2;"Diametrul de calare"; AT 19,4;"dc=";
  1.45*FN r(d); AT 19,18;" cm"
220 LET i$="2208081346 : GOSUB box: PRINT AT 9,2;"A"
  AT 10,25;"L"; AT 11,25;"T" AT 12,25;"E"; AT
  14,25;"D";AT 15,25 ;"A" ; AT 16,25;"T"; AT

```

```
17,25;"E"; AT 19,23;"(d/n)?" : BEEP .02.40: PAUSE
4: BEEP .02.-40: PAUSE 0
230 GO TO 50*(INKEY$="d")+240*(INKEY$="n")
240 GOSUB ch2
300 FOR f=6 TO 16: PRINT PAPER 5; INK 2: AT f,6;
"(22sp)": NEXT f
310 FOR f=7 TO 17: PRINT PAPER 1; INK 1; AT f,5;
" ": NEXT f: PRINT PAPER 1; INK 1; AT 17,6;
"(19sp)"
320 PRINT PAPER 4; INK 6; AT 7,6;"M.M.POPOVICI
SOFTWARE"; AT 9,15;"1993"; AT 13,6;
FLASH 1; INVERSE 1;"ARCUL BARA DE TORSIUNE"
330 PAUSE 0
8999 STOP
9000 BORDER 5: PAPER 5: INK 2: CLS
9001 PLOT 15,0: DRAW 255,0: DRAW 15,15: DRAW 0,145:
DRAW -15,15: DRAW -255,0: DRAW -15,-15:
DRAW 0,-145: DRAW 15,-15: DRAW 0,15: DRAW -15,0:
PLOT 255,15: DRAW -15,0: DRAW 0,-15: PLOT
255,160: DRAW -15,0: DRAW 0,15: PLOT 0.160: DRAW
15,0: DRAW 0,15
9002 PRINT AT 0,0: OVER 1; PAPER 5;"(2sp)"; AT 0,30;
"(2sp)"; AT 1,0;"(2sp)"; AT 1,30;"(2sp)"; AT
20,0;"(2sp)"; AT 20,30;"(2sp)"; AT 21,0;"(2sp)";
AT 21,30;"(2sp)": RETURN
9100 LET t$="(15sp)": PAPER 6: BRIGHT 1: INK 0:
BORDER 7: CLS : PLOT 0,0: DRAW 255,0: DRAW
0,175: DRAW -255,0: DRAW 0,-175
9101 PRINT AT 1,2; PAPER 5;t$;t$(2 TO): FOR i=2 TO
20: PRINT AT i,1; PAPER 7;t$;t$(2 TO); AT i-
1,30; PAPER 5;" ": NEXT i
9102 RETURN
9200 LET a$="(4sp)APASATI O TASTA OARECARE (4sp)"
9201 PRINT # 1; AT 1,0;a$
9202 LET a$=a$(32)+a$(TO 31)
9203 PAUSE 5: IF INKEY$="" THEN GO TO 9201
9204 RETURN
9300 FOR i=0 TO 127 STEP 8: PLOT i,0: DRAW INK k
PAPER k;0,175: PLOT 255-i,0: DRAW INK k: PAPER
```

```

k:0,175:NEXT i: RETURN
9400 RESTORE 9401: FOR q=0 TO 71: READ a: POKE
    65368+q,a: NEXT q
9401 DATA 130,198,170,146,130,131,0,0,0,0,0,128,159,
    213,149,85,130,198,170,146,130,131,0,0,0,0,
    128, 209,155,149,81
9204 DATA 0,62,64,64,68,68,56,0,0,0,0,4,12,68,228,78
9403 DATA 0,48,74,124,72,72,48,0,0,0,0,0,96,144,
    144,120
9404 DATA 96,16,32,120,0,0,0,0
9405 RETURN
9600 LET xo=VAL i$(1 TO 2): LET yo=VAL i$(3 TO 4):
    LET xl=VAL i$(5 TO 6): LET yl=VAL i$(7 TO 8)
9601 PAPER VAL i$(9): INK VAL i$(10)
9602 LET i=0
9603 IF i THEN LET x=FN w(xo,yo,xl,yl)
9604 LET i$=I$(TO xl): FOR i=yo TO yo+yl-1: PRINT AT
    i,xo;i$: NEXT i
9605 PLOT 8*xo+1,174-8*yo: DRAW 8*xl-3,0: DRAW
    0,3-yl*8: DRAW 3-xl*8,0: DRAW 0,yl*8-3
9606 RETURN

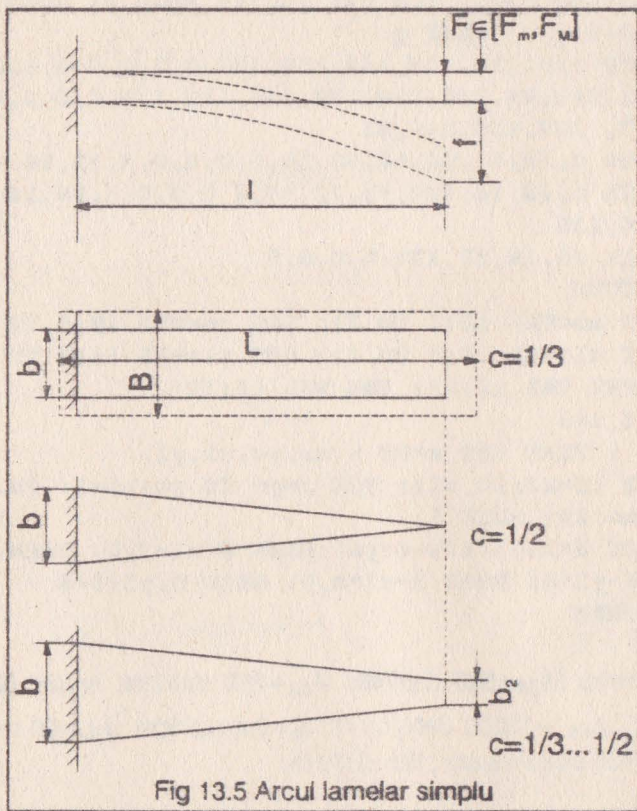
```

Pentru $M_{tm}=600$ daNcm; $M_{tm}=200$ daNcm; Arc2A ($G=7,8 \cdot 10^5$ daN/cm²; $\sigma_{+1}=13200$ daN/cm²); $\theta_a=0,2$ rad/cm și $L=50$ cm rezultă: $d=1$ cm; $l=26$ cm; $r=2$ cm; $dc=2,9$ cm.

13.5. SINTEZA OPTIMALĂ A ARCULUI LAMELAR SIMPLU

Arcul lamelar simplu (fig.13.5) este o foaie metalică încastrată la un capăt și liberă la celălalt capăt unde acționează forța exterioară variabilă $F[\text{daN}] \in \{F_m - \text{forța minimă}; F_M - \text{forța maximă}\}$ care solicită arcul la încovoiere.

Forma arcului poate fi dreptunghiulară, triunghiulară, trapezoidală, etc. și se execută din oțel pentru arcuri. Această formă se ia în considerare la determinarea săgeții (deformației) arcului prin intermediul unui coeficient cu următoarele valori:



{	1/3	forma dreptunghiulara
	1/2	forma triunghiulara
	1/3 .. 1/2	forma trapezoidala

Se utilizează la mecanismele cu clichet, releu, comutatoare

electrice, perii colectoare, site, vibratoare, etc.

Deformația pe care o capătă arcul în urma solicitării sale poate fi cel mult egală cu o deformație admisibilă f_a [cm], caz în care se numește *arc cu săgeată maximă*, sau poate fi cel puțin egală cu deformația admisibilă când se numește

arc cu săgeată minimă.

Aplicînd metoda operațională se deduce următorul algoritm de calcul :

Date: forța maximă F_M [daN] ; forța minimă F_m [daN]; săgeata admisibilă f_a [cm]; dimensiunile permise de gabarit B [cm], L [cm]; materialul arcului

(σ_{+1} [daN/cm²]; E [daN/cm²]); forma foii arcului ($c=1/3...1/2$); tipul arcului (cu săgeată maximă sau minimă).

Rezolvare : 1) Coeficientul de asimetrie al ciclului la oboseală

$$R = F_m / F_M$$

2) Tensiunea admisibilă la oboseală $\sigma_{aR} = 1.3\sigma_{+1}/(3-R)$.

a) *Arcul cu săgeată maximă* ($f \leq f_a$)

- lungimea foii $l = L$ [cm]

- lățimea foii $b = (\min)\{1.5 F_M E^2 f_a^2 / c^2 \sigma_{aR}^3 L^3 ; B\}$

- grosimea foii $h = \sqrt{6 F_M L / b \sigma_{aR}}$

b) *Arcul cu săgeată minimă* ($f \geq f_a$)

- compatibilitatea problemei $\sigma_{aR} \geq \sqrt[3]{1.5 F_M E^2 f_a^2 / c^2 B L^3}$

- dimensiunile arcului $l = L ; b = B ; h = \sqrt{6 F_M L / B \sigma_{aR}}$

Acest algoritm este transpus în programul de calcul care urmează, realizat pentru arcul lamelar simplu cu foaie dreptunghiulară ($c=1/3$).

```

1 CLS : OVER 0: LET Y$="(32sp)": REM sp=blanc
2 LET ch1=9000: LET ch2=9100: LET bord=9200: LET
  apasa=9300: LET udg=9400: LET scris=9500
30 GOSUB udg
40 BORDER 1: PAPER 1: INK 6: CLS

```

```

45 LET b=1: GOSUB bord
50 PRINT AT 4,5; PAPER 0; INK 7;"M.M.POPOVICI
SOFTWARE"; AT 12,6; PAPER 4; INK 6;"ARCUL
LAMELAR SIMPLU"; AT 14,4;"DE FORMA DREPTUN
GHIULARA"
60 LET col=24: LET a$="(Sinteza optimală)": LET lin
=16: GOSUB scrie
70 GOSUB apasa
5000 BORDER 6: PAPER 6: INK 1: CLS
5003 PLOT 0,24: DRAW 255,0: DRAW 0,144: DRAW -255,0:
DRAW 0,-144
5010 LET i$="04022403071": GOSUB 9600: PAPER 5: INK
1: LET A$="(2sp)DATELE PROBLEMEI(2sp)": LET
lin=3: LET col=25: GOSUB scrie
5020 PAPER 6: INK 0: INPUT "Introduceti forta maxima
" [daN]",FMAX: PRINT AT 6,2;"-forta maxima
EF=";FMAX;" daN": REM majusculele subliniate se
tasteaza in modul grafic
5030 INPUT "Introduceti forta minimaCD[daN]",Fm:
PRINT AT 8,2;"-forta minima CD=";Fm;" daN"
5040 INPUT "Introduceti sageata admisibila fG [cm]",
fa: PRINT AT 10,2;"-sageata admisibila fG=";fa;
" cm"
5050 INPUT "Introduceti dimensiunile maxi- me permise
de gabarit B si L[cm]",B,L
5060 PRINT AT 12,2;"-dimensiunile gabaritului :";
AT 14,5;"B=";B; " cm";"(2sp)L=" ; " cm"
5070 LET c=1/3
5080 PRINT # 0: PAPER 5; INK 2; FLASH 1;"(4sp)APASATI
O TASTA OARECARE(4sp)": BEEP .005,35: PAUSE 3:
BEEP .005,45: PAUSE 0
5090 FOR i=6 TO 17: FOR j=2 TO 30: PRINT AT i,j;" ":
NEXT j: NEXT i
5100 INPUT "Introduceti HI[daN/cmJ]",sig: PRINT AT
8,2;"-tensiunea la rupere statica"; AT 9,3:"HI="
sig;" daN/cmJ"
5110 INPUT "Introduceti modulul de elasti-(2sp)citate
lon-(3sp)gitudinal E[daN/cmJ]",E: PRINT AT 11,2;
OVER 1;" -modulul de elasticitate

```

```

longitudinal E=";E;" daN/cmJ"
5115 INPUT "Indicati tipul arcului :(10sp)cu sageata
maxima (1) sau cu(4sp)sageata minima (2)",s$:
PRINT AT 13,2;-arc cu sageata "
5116 IF s$="1" THEN PRINT AT 13,18;"maxima"
5117 IF s$="2" THEN PRINT AT 13,18;"minima"
5120 LET i$="02152803511": GOSUB 9600: LET AS="DATELE
SINT CORECTE (d/n)?": LET lin=16: LET col=28:
GOSUB scrie
5125 PAUSE 0
5130 GO TO 5140*(INKEY$ ="d")+5000*(INKEY$ ="n")
5140 CLS: LET R=Fm/FMAX: LET sigmar=1.3*sig/(3-R)
5146 IF s$="2" THEN GO TO 5200
5150 LET l=L: LET v1=1.5*FMAX*E*E*fa*fa/(c*c*sigmar*
sigmar*sigmar*L*L*1): LET v2=B
5155 IF v1 <= v2 THEN LET b=v1
5160 IF v2 <= v1 THEN LET b=v2
5170 LET h=SQR(6*FMAX*L/(b*sigmar))
5170 GOSUB chl: LET i$="0402240316": BORDER 5: PAPER
6: GOSUB 9600: LET a$="Dimensiunile arcului :":
LET lin=3: LET col=26: GOSUB scrie: PRINT AT
7,6;"-lungimea l=";L;" cm"; AT 9,6;"latimea
b=";b;" cm"; AT 11,6;"-grosimea h=";h;" cm"
5185 LET b=5: GOSUB apasa
5190 GO TO 6000
5200 LET k=(1.1*FMAX*E*E*fa*fa/(c*x*B*L*L*))1/3
5210 IF sigmar < k THEN GOSUB chl LET i$=
"0511240317": GOSUB 9600: BORDER 5: PAPER 6: LET
a$="PROBLEMA INCOMPATIBILA": LET lin=12: LET
col=27: GOSUB scrie: PRINT AT 17,6; PAPER 4; INK
6; "RELUATI CU ALTE DATE !": LET b=5: GOSUB
apasa: GO TO 5000
5220 LET h=SQR(6*FMAX*L/B*sigmar))
5230 GOSUB chl: LET i$="0402240316": GOSUB 9600: LET
a$="Dimensiunile arcului:": BORDER 5: PAPER 6:
LET lin=3: LET col=26: GOSUB scrie: PRINT AT
7,6;"-lungimea l=";L;" cm"; AT 9,6;"-latimea
b=";b;" cm"; AT 11,6;"grosimea h=";h;" cm"
5240 LET b=5: GOSUB apasa

```

```

6000 GOSUB ch1: PRINT AT 12,4;"RELUATI APLICATIA
      (d/n)?" : PAUSE 0
6010 GO TO 5000*(INKEY$ ="d")+6020*(INKEY$ ="n")
6020 BORDER 1: PAPER 1: INK 6: CLS: GOSUB bord: PRINT
      AT 12,4:"PROGRAMUL S-A TERMINAT !": STOP
9000 BORDER 5: PAPER 6: INK 2: CLS: PLOT 15,0: DRAW
      225,0: DRAW 15,15: DRAW 0,145: DRAW -15,15: DRAW
      -225,0: DRAW -15, -15: DRAW 0,-145: DRAW 15,-15:
      DRAW 0,15: DRAW -15,0: DRAW 0,-15 PLOT 255,160:
      DRAW -15,0: DRAW 0,15: PLOT 0,160: DRAW 15,0:
      DRAW 0,15
9001 PRINT AT 0,0; OVER 1; PAPER 5;"(2sp)"; AT 0,30;
      "(2sp)"; AT 1,0;"(2sp)"; AT 1,30;"(2sp)"; AT
      20,0;"(2sp)"; AT 20,30;"(2sp)"; AT 21,30;
      "(2sp)": RETURN
9100 PLOT 0,0: DRAW 255,0: DRAW 0,175: DRAW -255,0:
      DRAW 0,-175 RETURN
9200 FOR f=2 TO 19: PRINT AT f,31;"A"; AT f,30;"B":
      BEEP .01,20: NEXT f
9202 FOR f=31 TO 0 STEP -1: PRINT AT 21,f;"B"; AT 20,
      f;"A": BEEP .01,20: NEXT f
9203 FOR f=19 TO 2 STEP -1: PRINT AT f,0;"A";AT f,1;
      "B": BEEP .01,20: NEXT f
9204 RETURN
9300 LET d$="APASATI O TASTA OARECARE"
9301 FOR a=LEN d$ TO 1 STEP -1: PRINT # 1; AT 1,4;d$
      (a TO): BEEP .002,50+INT(RND*10): OUT 254,4:
      BORDER b: NEXT a: PAUSE 0: CLS : RETURN
9400 RESTORE 9401: FOR q=0 TO 79: READ a: POKE
      65368+q.a: NEXT q
9401 DATA BIN 100001, BIN 110000, BIN 1110000, BIN
      11111000, BIN 11001100, BIN 10000110, BIN 11,
      BIN 11, BIN 11, BIN 11, BIN 10000110, BIN
      11001100, BIN 11111000, BIN 11100000, BIN
      110000, BIN 10000
9402 DATA 248,128,128,224,135,133,133,5,0,0,0,0,192,
      64,64,64,248,128,128,224,132,134,133,4,0,0,0,0,
      64,192,64,64,0,0,0,56,72,72,60,0,0,62,64,64,68,
      68,56,0,0,0,0,4,12,68,228,78,96,16,32,120,0,

```



```

0,0,0
9404 RETURN
9500 FOR x=LEN a$ TO 1 STEP -1: PRINT AT lin,col;
a$(x); :LET col=col-1:NEXT x: RETURN
9600 LET xo=VAL i$(1 TO 2): LET yo=VAL i$(3 TO 4):
LET xl=VAL i$(5 TO 6): LET yl=VAL i$(7 TO 8)
9610 PAPER VAL i$(9): INK VAL i$(10)
9620 LET i=0
9630 LET Y$=Y$(TO xl): FOR i=yo TO yo+yl-1: PRINT AT
i,xo;i$: NEXT i
9640 PLOT 8*xo+1,174-8*yo: DRAW 8*xl-3,0: DRAW
0,3-yl*8: DRAW 3-xl*8,0: DRAW 0,yl*8-3
9650 BORDER 6: PAPER 6: INK 9: RETURN

```

Aplicatie numerică: $F_M = 100 \text{ daN}$; $F_m = 50 \text{ daN}$; $f_a = 0,4 \text{ cm}$;
 $B = 2 \text{ cm}$; $L = 25 \text{ cm}$: materialul ARC 5 cu $E = 2,06 \cdot 10^6 \text{ daN/cm}^2$ și
 $\sigma_{+1} = 9800 \text{ daN/cm}^2$.

Rezultă: $R = 0,5$; $\sigma_{aR} = 5096 \text{ daN/cm}^2$

a) Arc cu săgeată maximă ($f \leq f_a$)

$l = 25 \text{ cm}$; $b = 0,4432 \text{ cm}$; $d = 2,58 \text{ cm}$

b) Arc cu săgeată minimă ($f \geq f_a$)

$5096 > 2353, 5236 \text{ daN/cm}^2$ (problemă compatibilă)

$l = 25 \text{ cm}$; $b = 2 \text{ cm}$; $h = 1,21 \text{ cm}$.

13.6. SINTEZA OPTIMALĂ A ARCULUI ELICOIDAL CILINDRIC GENERATOR DE FORȚE

Arcul elicoidal cilindric din sîrmă rotundă este realizat dintr-o bară (sîrmă) de secțiune circulară de diametru d (fig.13.6) roluită după o elice cilindrică de diametru D , astfel încît să formeze n spire sub sarcina variabilă $F[\text{daN}] \in \{F_m - \text{forța minimă}; F_M - \text{forța maximă}\}$, dispuse pe o înălțime de lucru H [cm]. Arcul se montează într-o carcasă sau un spațiu aferent de diametru D_c [cm] și poate avea în interiorul său un ax sau un amortizor de diametru D_a [cm].

Valorile standardizate ale diametrului sîrmei arcului sînt

următoarele:

d [cm] = 0,1; 0,2; 0,3; 0,4; 0,5; 0,6; 0,7; 0,8; 1; 1,2; 1,5; 1,8; 2,2; 2,5; 2,8; 3; 3,2; 3,5; 3,8; 4; 4,2; 4,5; 4,8; 5.

Acest arc poate lucra în domeniul său interior (cînd se numește *arc de comprimare*) sau în domeniul său exterior (cînd se numește *arc de extindere*; el se utilizează la vehiculele rutiere și de cale ferată, supape, tampoane, etc.,

În calcul optimal al arcului se folosesc următorii coeficienți adimensionali care se referă la dimensiunile arcului:

- coeficientul de formă al spirei: $i = D/d$

- coeficientul de gabarit: $\gamma = D_c/D_a$

- coeficientul sîrmei: $\rho = d/D_a$

Metoda operațională se concretizează în algoritmul de calcul care urmează, realizat pentru arc de comprimare.

Datele problemei: Forța minimă F_m [daN]; forța maximă F_M [daN]; diametrul carcasei D_a [cm]; diametrul axului D_a [cm]; înălțimea de lucru H [cm]; modulul de elasticitate transversal G [daN/cm²]; materialul (neimpus: $\sigma_{+1M} = 18640$ daN/cm² -conf. tab.13.1; impus σ_{+1} [daN/cm²]).

Rezolvare: 1) Valori de calcul: $R = F_m/F_M$; $\sigma_{aR} = 1,3\sigma_{+1}/(3-R)$ la materialul neimpus respectiv $\sigma_{aR} = 1,3\sigma_{+1}/(3-R)$ la materialul impus;

$$\lambda = D_a^2 \sigma_{aR} / 5 F_M; \gamma = D_c/D_a$$

2) Compatibilitatea problemei:

$$\sqrt[3]{\lambda} \geq \left\{ 1,6/\sqrt[3]{\gamma^2} - 1; 6/\sqrt[3]{\gamma^2} \right\}$$

3) Soluția optimă teoretică: $i_A = \sqrt[3]{\lambda \gamma^2} - 1$; $\rho_A = \gamma/(i_A + 1)$

$$d_A = \rho_A D_a$$

4) Soluția tehnică:

a) Cînd $d = d_{STAS}$ atunci $i = i_A$; $\rho = \rho_A$ și se folosește cel mai bun material de arc.

b) *Material impus*: diametrul barei arcului $d = (\text{proxim})d_{STAS} < d_A$; $\rho = d/D_a$; $i = \lambda \rho^2 - 0,665$.

c) Material

neimpus : diametrul barei
arcului: $d = (\text{proxim})d_{STAS}$

$> d_A$; $\rho = d/D_a$; $i = (\rho/\gamma) - 1$;

$\sigma'_{aR} = 5F_M(i + 0,665)/D_a^2 \rho^2$;

materialul trebuie să aibă
tensiunea la rupere statică

$\sigma'_{+1} \geq (3 - R) \sigma'_{aR} / 1,3$

(rezultă materialul optim din
tab.13.1)

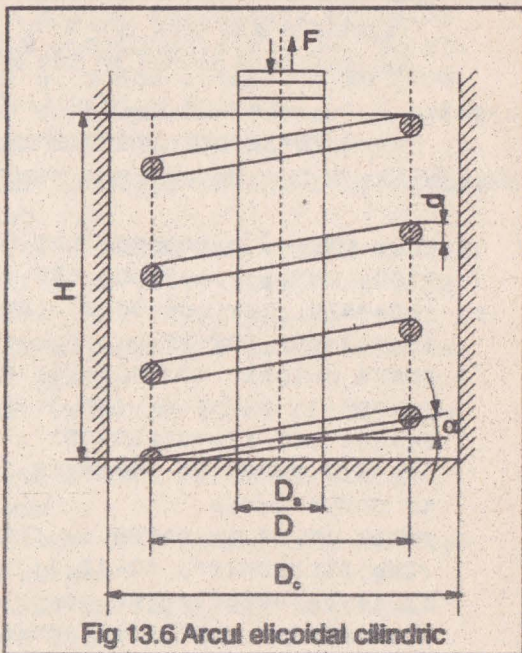


Fig 13.6 Arcul elicoidal cilindric

5) Structura arcului :

- diametrul eliciei $D = id$;
- numărul spirelor active $n = GHd/Gd^2 + 8(F_M - F_m) \beta^3$
- înălțimea arcului în stare liberă $H_0 = H + (8F_m \beta^3 n/Gd)$

6) Rezemarea arcului :

$$\beta = 2,62D/H_0 \begin{cases} \leq 0,5 & \text{ghidare pe lungime} \\ = 0,5 \dots 1 & \text{ghidare în reazem} \\ \in (1 \dots 2) & \text{ghidare într - un reazem} \\ > 2 & \text{neghidat} \end{cases}$$

7) Detalii constructive :

- numărul spirelor de rezemare

$$n_r = \begin{cases} 1,5 & \text{pentru } n \leq 7 \\ 2,5 & \text{pentru } n > 7 \end{cases}$$

- numărul total de spire $n_t = n + n_r$
- lungimea barei (sîrmei) din care se confecționează arcu

$$\alpha = \arctg (H_0 / \pi D n) : l = \pi D n_t / \cos \alpha$$

Programul pe calculator pentru sinteza optimală a arcului elicoidal cilindric de comprimare este prezentat în cele ce urmează.

```

1 CLS: LET culori=9000: LET bord=9100: LET scrie=
  9300: LET apasa 9400: LET chl=9500: LET
  ch2=9550: LET udg=9560: LET box=9600: LET
  X$="32sp": DEF FN w(a,b,c,d)=100: REM (sp)=blanc
2 GOSUB culori: GOSUB bord: GOSUB udg
3 BORDER 3: PAPER 6: INK 0: CLS : RANDOMIZE USR
  64000: LET i$="010302051": GOSUB box: LET
  a$="M.M.POPOVICI SOFTWARE ": LET lin=3: LET col=
  4: GOSUB scrie
4 PRINT AT 10,3: PAPER 6; INK 0;"ARCUL ELICOIDAL
  (2sp)CILINDRIC"; AT 12,7;"GENERATOR DE FORTE";
  AT 14,6;"-arc(2sp)de comprimare-";AT 16,7;
  INVERSE 1;1;"(Sinteza optimala)"
5 GOSUB apasa
6 GO TO 3000
99 REM diametrele barei de arc
100 DIM d(25): RESTORE 100: FOR i=1 TO 25: READ
  d(i): NEXT i: DATA.1,.2,.3,.4,.5,.6,.7,.8,1,1.2,
  1.5,1.8 ,2,2.2,2.5,2.8,3,3.2,3.5,3.8,4,4.2,
  4.5,4.8,5
101 RETURN
199 REM materialele pentru bara de arc
200 DIM u$(13,8): DIM v(13)
201 RESTORE 201: FOR k=1 TO 13: READ u$(k): NEXT k:
  DATA "RS","RM","51Si17A","OLC65","RR","OLC55A",
  "OLC75A","OLC85C","56Si17A","40Si17A",
  "40Si17A","51VCr11A","60Si15A","62Si2W"
202 RESTORE 202: FOR k=1 TO 13: READ v(k): NEXT k:
  DATA 7000,8000,9800,9800,10000,10800,10800,
  11300,11800,11800,13200,14700,18640
203 RETURN
299 REM Problema incompatibila

```

```

300 PRINT AT 11,7;"NU EXISTA MATERIAL"; AT 18,3;
    "MODIFICATI DATELE PROBLEMEI": FOR w=1 TO 15:
    BEEP .01,40: NEXT w: GOSUB apasa: GO TO 3000
499 REM cazul arcului de comprimare-material neimpus
500 LET de=i1*d1: LET n=G*H*d1/(G*d1*d1+8*
    (f2-f1)*i1^3): LET h0=h+((8*f1*i1^3*n)/(G*d1):
    LET beta=2.6*de/h0
510 LET i$="04132409701": GOSUB box: PRINT AT 1,4;
    "STRUCTURA ARCULUI"; AT 16,7;"D[cm]=";de; AT
    17,7;"n=";n;" spire"; AT 18,7;"HN[cm]=";h0; AT
    19,7;"O";beta: REM majusculele subliniate se
    tasteaza in modul grafic
520 IF beta <= .5 THEN PRINT AT 20,5;"ghidare pe
    lungime"
525 IF beta>.5 AND beta<1 THEN PRINT AT 20,5;
    "ghidare in reazeme"
530 IF beta >= 1 AND beta <= 2 THEN PRINT AT 20,5;
    "ghidare intr-un reazem"
535 IF beta>2 THEN PRINT AT 20,5;"arc neghidat"
540 GOSUB apasa: GOSUB chl
545 PRINT AT 6,6;"DETALII CONSTRUCTIVE"
550 IF n <=7 THEN LET nr=1.5
555 IF n>7 THEN LET nr=2.5
560 PRINT AT 8,3;"nr=";nr; AT 8,16;" spire de
    reazem" AT 10,3;"nt=";nr+n; AT 10,16;" nr.total
    de spire"
565 LET abc=h0(PI*de*n): LET alfa=ATN(abc): PRINT
    AT 12,3;"D=";alfa*180/PI;"(2sp)grade"
570 LET l=PI*de*(nr+n)/COS(alfa): PRINT AT 14,3;
    "l=[cm]="; l; AT 15,3;"lungimea barei de arc"
580 GOSUB apasa
640 GOSUB ch2: PRINT AT 12,2; PAPER 7; INK 0;
    "RELUATI CU ALTE DATE (d/n)?: PAUSE 0
650 GO TO 3000*(INKEY$="d")+5000*(INKEY$="n")
2999 REM intrarea datelor
3000 BORDER 3: PAPER 6: INK 0: CLS : RANDOMIZE USR
    64000: LET i$="0101302060": GOSUB box
3015 LET a$="DATELE DE PROIECTARE": LET lin=2: LET
    col=3: GOSUB scrie

```

```

3020 INPUT "Introduceti forta minima FA[daN]", f1:
OVER 1: PRINT AT 4,2;"1)Forta minima"; AT 5,3;
"FA [daN]=";f1
3030 INPUT "Introduceti forta maxima FB[daN]=", f2:
PRINT AT 6,2;"2)Forta maxima"; AT 7,3;"FB
[daN]=";f2
3060 INPUT "Introduceti diametrul carcasei(2sp)DC
[cm]",dc: PRINT AT 8,2;"3)Diametrul carcasei":
AT 9,3;"DC[cm]=";dc
3070 INPUT "Introduceti diametrul axului DD[cm]",da:
PRINT AT 10,2;"4)Diametrul axului"; AT 11,3;
"DD[cm]=";da
3080 INPUT "Introduceti inaltimea de lucru H[cm]", h:
PRINT AT 12,5;"5)Inaltimea de lucru" AT 13,3;
"H[cm]=";h
3085 LET i$="05152203611": GOSUB box: PRINT AT 16,7;
"DATE CORECTE (d/n)?: BEEP .02,50: PAUSE 5:
BEEP .02,10: PAUSE 0
3086 IF INKEY$="n" THEN GO TO 3000
3087 CLS : RANDOMIZE USR 64000: LET i$="0101301860":
GOSUB box
3088 LET a$="DATELE DE PROIECTARE": LET lin=2: LET
col=3: GOSUB scrie
3090 INPUT "Introduceti modulul de elasti-(2sp)citate
transversal G[daN/cmP]",g: PRINT AT 4,2;"6)Modul
ul de elasticitate"; AT 5,3;"G[daN/cmP]";g
3091 INPUT "Introduceti : 1)material impus(33sp) 2)
material neimpus", m$
3092 IF m$="1" THEN PRINT AT 6,2 ;"7)Material impus
": INPUT "Introduceti simbolul materialu- lui
si tensiunea la rupere sta- tica EF[daN/cmP]"
w$,s1: PRINT AT 7,2;w$: AT 8,3;"EF [daN/cmP=";s1
3093 IF m$="2" THEN PRINT AT 6,2;"7)Material
neimpus": INPUT "Introduceti tensiunea la rupere
statica EFB[daN/cmP] a celui(3sp)mai bun
material",sm: PRINT AT 7,2;"EFB[daN/cmP]=";sm
3095 PRINT AT 9,2;"8)Tipul arcului : "; AT 10,3 ;"de
comprimare"
3100 LET i$="05152203611": GOSUB box: PRINT AT 16,7;

```

```

"DATE CORECTE (d/n)?": BEEP .02, 50: PAUSE 5:
BEEP .02,10: PAUSE 0
3105 IF IKEY$="d" THEN RANDOMIZE USR 64100: GO TO
3115
3110 IF INKEY$="n" THEN RANDOMIZE USR 64100: GO TO
3000
3115 IF m$="2" THEN GO TO 3130
3120 IF m$="1" THEN GO TO 4000
3130 LET R=f1/f2: LET sar=1.3*sm/(3-R): LET lbd=
da*da*sar/(5*f2): LET gama=dc/da: CLS : GOSUB
ch1: PRINT AT 1,8; PAPER 4; INK 7;"VALORI DE
CALCUL": PAPER 6: INK 2: PRINT AT 3,3;"R";R; AT
4,3"EGH[daN/cmP]=";sar;AT 5,3;"I=";lbd; AT 6,3;
"J=";gama; AT 9,4;"(I)↑(1/3)="
3140 LET z1=lbd↑(1/3): LET z2=1.6/
((gama*gama)↑(1/3))-1: LET
z3=6/((gama*gama)↑(1/3))
3141 IF z1<z2 AND z1<z3 THEN GO TO 6000
3150 IF z1 >= z2 AND z1 >= z3 THEN PRINT AT 8,14;
" >"; AT 10,4;"[ ";z2;" ";z3;"]"; AT 12,6 ;
INVERSE 1;"PROBLEMA COMPATIBILA"
3060 PRINT AT 14,2;" SOLUTIA OPTIMA TEORETICA"
3170 LET ia=((gama*gama*lbd)↑(1/3))-1: LET roa=
gama/(ia+1): LET d0=roa*da: PRINT AT 16,3;
"iK=";ia; AT 17,3;"LK=";roa; AT 18,3;
"dK[cm]=";d0
3180 GOSUB apasa: GOSUB ch1: GOSUB 100
3190 FOR k=1 TO 25
3200 IF d0=d(k) THEN GO TO 3240
3210 IF d(k)>d0 THEN LET d1=d(k): GO TO 3230
3220 NEXT k
3240 PRINT AT 1,15; INVERSE 1;" SOLUTIA TEHNICA";
INVERSE 0: LET roal=d(k)/da: LET iil=(gama
/roal)-1: LET spar=(5*f2)*(iil+0.665)/
(da*da*roal*roal): PRINT AT 1,3; "d[cm]=";d(k);
AT 2,3;"L=";roal; AT 3,3; "i=";iil; AT 4,3;
"EGHM[daN/cmP]=";spar
3250 LET sprim=(3-R)*spar/1.3: PRINT AT 5,3;"EFM

```

```

[daN/cmP] >";sprim
3260 GOSUB 200
3270 FOR k=1 TO 13
3280 IF sprim>v(13) THEN GO TO 300
3290 IF sprim=v(k) THEN GO TO 3320
3300 IF sprim <= v(k+1) THEN LET k=k+1: GO TO 3320
3310 NEXT k
3320 LET i$=03072605171": GOSUB box: PRINT AT 8,4;
OVER 1;"Materialul optim :"; AT 8,21; u$(k);
AT 9,4 "tens.la rupere statica "; AT 10,7;
v(k);" daN/cmP"
3350 GO TO 500
4000 LET R=f1/f2; LET sar =1.3*s1/(3-R): LET lbd=
da*da*sar/(5*f2): LET gama=dc/da: GOSUB chl:
PRINT AT 1,8; PAPER 4; INK 7;"VALORI DE
CALCUL": PAPER 6: INK 2: PRINT AT 3,3;"R=";R; AT
4,3 ;"EGH[daN/cmP]=;sar; AT 5,3;"I=";lbd; AT
6,3;"J=" ;gama: AT 8,4 ;"(I)^(1/3)="
4010 LET z1=lbd^(1/3): LET z2=1.6/(gama*gama)^(
(1/3)-1): LET z3/((gama*gama)^(1/3))
4015 IF z1<z2 AND z1<z3 THEN GO TO 6000
4016 IF z1 >= z2 AND z1>=z3 THEN PRINT AT 8,1;" >";
AT 10,4;"[" ;z2;" ";z3;"]"; AT 12,6; INVERSE
1;"PROBLEMA COMPATIBILA"
4017 PRINT AT 14,2;"SOLUTIA OPTIMA TEORETICA
4020 LET ia=((gama*gama*lbd)^(1/3))-1: LET roa=
gama/(ia+1): LET d0=roa*da: PRINT AT 16,3;"iK=";
ia; AT 17,3;"LK=";roa; AT 18,3" dK[cm]=";d0
4025 GOSUB apasa: GOSUB chl: GOSUB 100
4030 FOR k=25 TO 1 STEP -1
4035 IF d0=d(k) THEN GO TO 4050
4040 IF d0>d(k) THEN LET k=k-1: LET d1=d(k-1): GO TO
4050
4045 NEXT k
4050 PRINT AT 1,15; INVERSE 1;"SOLUTIA TEHNICA";
INVERSE 0: LET roal=d(k)/da: LET iil=
(gama/roal)-1: PRINT AT 1,3;"d[cm]=";d(k); AT
2,3;"L=";roal; AT 3,3;"i="^(iil

```



```

4060 LET n=g*h*d(k)/(g*d(k)+8*(f2-f1)*iil↑3): LET
Ho=h+(8*f1*iil 3*n)/(g*d(k)): LET beta=2.62
*d(k)/Ho
4090 LET i$="03072605171": GOSUB box: PRINT AT 8,4;
"Materialul impus:"; AT 8,21;w$; AT 9,4;"tens.
la rupere statica"; AT 10,7;s1;" daN/cmP"
4092 LET i$="04132409701": GOSUB box: PRINT AT 14,9;
"STRUCTURA ARCULUI"; AT 16,7;"D[cm] =";iil*d(k);
AT 17,7;"n=";n;" spire"; AT 18,7;"HN[cm] =";
Ho; AT 19,7;"Q=";beta
4095 IF beta <= .5 THEN PRINT AT 20,5;"ghidare pe
lungime"
4110 IF beta>.5 AND beta <= 1 THEN PRINT AT 20,5;
"ghidare in reazeme"
4115 IF beta >= 1 AND beta >= 2 THEN PRINT AT 20,5;
"ghidare intr-un reazem"
4120 IF beta>2 THEN PRINT AT 20,5;"arc neghidat"
4125 GOSUB apasa: GOSUB chl
4130 PRINT AT 6,6;"DETALII CONSTRUCTIVE"
4140 IF n <= 7 THEN LET nr=1.5
4150 IF n>7 THEN LET nr=2.5
4155 PRINT AT 8,3;"nr=";nr; AT 8,17;" spire de
reazem" AT 10,3;"nt=";nr+n; AT 10,17;"nr.total
spire"
4160 LET abc=Ho/(PI*iil*d(k)*n): LET alfa=ATN(abc):
PRINT AT 12,3;"D=";alfa*180/PI;"(3sp)grade"
4165 LET l=PI*iil*d(k)*(nr+n)/COS(alfa): PRINT AT
14,3 ;"l[cm]=";l
4170 PRINT AT 15,3;"(lungimea barei de arc)"
4175 GOSUB apasa
4200 GOSUB ch2: PRINT AT 12,2; PAPER 7; INK 0;
"RELUATI CU ALTE DATE(2sp)(d/n)?: PAUSE 0
4210 GO TO 3000*(INKEY$="d")+5000*(INKEY$="n")
5000 RANDOMIZE USR 64100: BORDER 3: PAPER 6: INK 0:
CLS : RANDOMIZE USR 64000: LET i$="0101302051":
GOSUB box: LET a$="M.M.POPOVICI SOFTWARE": LET
lin=3: LET col=4: GOSUB scrie
5004 PRINT AT 10,3; PAPER 6; INK 0;"ARCUL ELICOIDAL
(2sp)CILINDRIC"; AT 12,7; "GENERATOR DE FORTE";

```

```

AT 14,6;"-arc((2sp)de comprimare -"; AT 16,7;
INVERSE 1;"(Sinteză optimală)"
5005 PAUSE 0: STOP
6000 PRINT AT 11,5 INVERSE1;"PROBLEMA INCOMPATIBILĂ";
INVERSE 0: PRINT AT 18,3;" MODIFICATI DATELE
PROBLEMEI": FOR w=1 TO 15: BEEP .01,40: NEXT w:
GOSUB apasa: GO TO 3000
9000 LET adr1=64000: RESTORE 9001: FOR i=1 TO 29:
READ a: POKE adr1+i,a: NEXT
9001 DATA 1,64,1,255,23,117,237,176,33,0,88,1,224,2,
125,230,28,7,238,56,119,35,11,120,177,32,243,201
9002 RETURN
9100 LET adr2=64100: RESTORE 9101: FOR i=1 TO 38:
READ a: POKE adr2+i,a: NEXT i
9101 DATA 17,232,3,6,255,58,80,175,211,254,61,16,251,
27,122,179,200,24,240,0,0,17,232,3,62,182,
71,211,254,175,16,251,27,122,179,32,243,201
9102 RETURN
9300 FOR w=0 TO LEN a$-1: LET col=col+1
9301 IF col=32 THEN LET lin=lin+1: LET col=1
9302 PRINT AT lin,col;"■"; CHR$;a$(w+1): BEEP
.00004,69: NEXT w: RETURN
9400 LET q$="(4sp)APASATI O TASTA OARECARE(4sp)"
9401 FOR i=1 TO LEN q$: PRINT # 1; AT 1,i-1;q$(i):
PAUSE 2: NEXT i: PAUSE 10
9402 FOR i=LEN q$ TO 1 STEP -1: PRINT # 1; AT 1,i-1;
OVER 0;" ": PAUSE 2
9403 IF INKEY$ <>" " THEN CLS : RETURN
9404 NEXT i= GO TO 9400
9500 BORDER 5: PAPER 6: INK 2: CLS
9501 PLOT 15,0: DRAW 255,0: DRAW 15,15: DRAW 0,145:
DRAW -15,1: DRAW -225,0: DRAW -15,15: DRAW
0,-145: DRAW 15,-15: DRAW 0,15: DRAW -15,0: PLOT
255,15: DRAW -15,0: DRAW 0,-15: PLOT 255,160:
DRAW -15,0: DRAW 0,15: PLOT 0,160: DRAW 15,0:
DRAW 0,15
9502 PRINT AT 0,0; OVER 1; PAPER 5;"(2sp)"; AT 0,30;
"(2sp)"; AT 1,0;"(2sp)"; AT 1,30;"(2sp)"; AT
20,0;"(2sp)"; AT 20,30;"(2sp)" AT 21,0;"(2sp)";

```

```

AT 21,30;"(2sp)"; RETURN
9550 LET t$="(15sp)": PAPER 6: INK 0: BORDER 7: CLS
9551 PLOT 0,0: DRAW 255,0: DRAW 0,175: DRAW -255,0:
DRAW 0,-175: PRINT AT 1,2; PAPER 5;t$(2 TO): FOR
i=2 TO 20: PRINT AT i,1; PAPER 7;t$;t$(2 TO); AT
i-1,30: PAPER 5;" ": NEXT i
9552 RETURN
9560 RESTORE 9561: FOR q=0 TO 8*16-1: READ a: POKE
65368+q,a: NEXT q
9561 DATA 0,0,0,0,248,168,168,168,0,0,0,0,136,
216,168,136
6562 DATA 0,0,0,0,96,128,128,96,0,0,0,56,72,72,60,0
9563 DATA 0,62,64,64,68,68,56,0,0,0,0,4,12,68,228,78
9564 DATA 0,0,0,28,2,30,34,30,0,0,0,240,136,136,240,
144
9565 DATA 0,56,4,4,28,36,34,0,0,0,72,48,32,32,32,0
9566 DATA 0,0,0,60,66,126,66,66,0,0,12,18,18,28,16,32
9567 DATA 32,64,128,0,0,0,0,0,0,0,0,96,144,144,96
9568 DATA 0,28,34,60,34,34,76,0,96,16,32,120,0,0,0,0
9570 RETURN
9600 LET xo=VAL i$(1 TO 2): LET yo=VAL i$(3 TO 4):
LET xl=VAL i$(5 TO 6): LET yl=VAL i$(7 TO 8)
9601 PAPER VAL i$(9): INK VAL i$(10)
9602 LET i=0
9603 LET X$=X$(TO xl): FOR i=yo TO yo+yl-1: PRINT AT
i,xo;i$: NEXT i
9604 PLOT 8*xo+1,174-8*yo: DRAW 8*xl-3,0: DRAW
0,3-yl*8: DRAW 0,yl:*8-3
9605 BORDER 6: PAPER 6: INK 9: RETURN

```

Aplicație numerică : $F_m = 400 \text{ daN}$; $F_M = 600 \text{ daN}$; $D_c = 24 \text{ cm}$;
 $D_a = 6 \text{ cm}$; $H = 30 \text{ cm}$; $G = 78000 \text{ daN/cm}^2$; material neimpus
($\sigma_{+1M} = 18640 \text{ daN/cm}^2$)

Valori de calcul: $R = 0,66666667$; $\sigma_{aR} = 10385,143 \text{ daN/cm}^2$;

$\lambda = 124,62171$; $\gamma = 4$; $\sqrt[3]{\gamma} = 49949511 > (1,0527409; 2,381106)$ - problemă
compatibilă

Soluția optimă teoretică: $i_A = 11,586488$; $\rho_A = 0,31780112$; $d_A = 1,9069067$

cm.

Soluția tehnică: $d = 2$ cm; $\rho = 0,33333333$; $\sigma'_{\text{aR}} = 8748,75$ daN/cm²;

$\sigma'_{+1} = 15703$ daN/cm²; rezultă materialul optim 62Si2W avînd $\sigma_{+1} = 18640$ daN/cm²

Structura arcului: $D = 22$ cm; $n = 8,9149649 = 9$ spire; $H_0 = 54,34014$ cm (înălțimea arcului în stare liberă; $\beta = 1,060726$ (ghidare într-un reazem)

Detalii constructive: $n_r = 2,5$ spire de rezemare; $n_t = 11,414965 = 12$ spire

$\alpha = 5,0399824 = 5$ grade; $l = 792$ cm (lungimea barei din care se va confecționa arcu)

13.7. SINTEZA OPTIMALĂ A TRANSMISIILOR CU CURELE TRAPEZOIDALE ÎNGUSTE

Transmisia prin curele trapezoidale înguste reprezintă un mecanism patruleter cu element intermediar flexibil (cureaua), prin care se transmite mișcarea de rotație de la roata conducătoare (1) la roata condusă (2) - fig.13.7.

Curelele înguste sînt realizate în 5 tipuri de dimensiuni *SPZ*, *SPA*, *SPB*, *16x15* și *SPC* și se livrează sub formă de curele fără fine cu lungimea primitivă $L = [400; 450; 500; 560; 630; 710; 800; 900; 1000; 1120; 1250; 1600; 1700; 1800; 2000; 1340; 2500; 2800; 3150; 3350; 3750; 4000; 5000; 5600; 6300; 7100; 8000; 9000; 10000; 11200]$ mm.

Notînd cu n_1 [rot/min] - turația roții motoare; P_0 [CP] - puterea unei curele ; P [CP] - puterea de transmis; $z \geq P/P_0 \geq 2$ - numărul curelelor și D_m - diametrul roții motoare, din diagrama reprezentată în fig.13.8 [7] se obțin diverse soluții pe *orizontala turației impuse* n_1 prin tema de proiectare și *verticala unui diametru* D_m [mm] standardizat,, pentru un anumit tip de curea.

Punctul de intersecție dintre *orizontala* n_1 și *verticala* D_m determină puterea P_0 [CP] care revine unei curele. Întrucît se oferă mai multe soluții se va alege soluția corespunzătoare unei *funcții de scop* și anume:

- pentru transmisii cu **număr minim de curele** $z = P/P_0 = 2$ se reține soluția curelei mari cu antrax a mare:

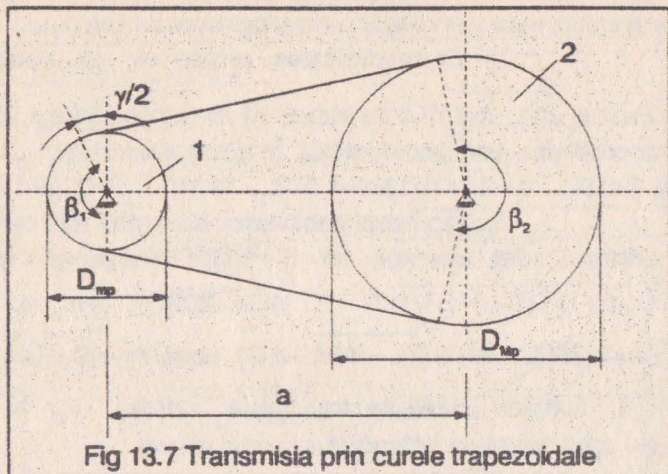


Fig 13.7 Transmisia prin curele trapezoidale

- pentru transmisii cu gabarit redus se reține soluția curelelor mici în număr mare;
- pentru transmisii ponderate se reține o soluție intermediară preferind

$$z \in (2..5).$$

De exemplu pentru o putere $P=20$ CP și o turație $n_1=2800$ rot/min, din fig.13.8 se deduc soluțiile din tabelul următor :

Tip	P_o [CP]	D_m [mm]	$z=P/P_o$
SPZ	3	70	7
	4	80	5
	5	90	4
	6	100	3
	3	125	2
	10	150	2
SPA	8	120	2
	10	125	2

Metoda operațională a condus la următorul algoritm de calcul :

Datele problemei: puterea transmisă P [CP]; coeficientul dinamic $k_d = 1...2$; turația n_1 [rot/min]; raportul de transmitere i .

Rezolvare : 1) Din Diagrama ARO-9 reprezentată în fig.13.8

se obțin câteva soluții care se vor testa prin programul de calcul.

2) Compatibilitatea problemei: se determină

coeficientul $f = (n + \sqrt{n^2 - mp})/m$ care trebuie să fie cuprins între 0,75 și 2; în caz contrar problema este incompatibilă. În relația anterioară:

$$n = 8D_m (i+1)^2 ; n = 2(i+1)[1600 - 1,57D_m (i+1)] ; p = D_m(i-1)^2$$

3) Geometria transmisiei: diametrul roții conduse

$D_M = iD_m$; antraxul de referință $a^* = f(i+1)D_m$; lungimea curelei $L \geq 2a^* + 1,57D_m(i+1) + [D_m(i-1)^2/4a^*] \in [400 \dots 2000]$ mm; antraxul

transmisiei $a = 0,25[(L - W) + \sqrt{(L - W)^2 - 8C}]$, unde $W = \pi D_m(i+1)/2$ și

$C = D_m^2(i+1)^2/4$; numărul curelelor transmisiei $z \geq Pdk_d / P_o$ în care $d = 1 \dots 1,5$ este coeficientul de înfășurare a curelei pe roți.

4) Forțele din curea :

$$\text{- în ramura motoare } S_1 = \frac{2M_{t1}}{D_m} \cdot \frac{e^{\mu\beta}}{e^{\mu\beta} - 1} \text{ [daN]}$$

$$\text{- în ramura condusă } S_2 = \frac{2M_{t1}}{D_m} \cdot \frac{1}{e^{\mu\beta} - 1} \text{ [daN]}$$

în care $M_{t1} = 7 \cdot 10^4 P / n_1$ [daN.cm] - momentul de răsucire transmis de roata motoare; $\mu = 0,35 + 0,012v$ - coeficientul de frecare dintre curea și roți unde $v = \pi D_m n_1 / 6 \cdot 10^4$ [m/s]; $\beta = \pi - \gamma$ iar $\gamma = (D_M - D_m) / a$.

Rezultanta forțelor din curea care solicită arborii și înclinarea suportului ei față de axa centrelor roților $0_1, 0_2$ sînt date de relațiile:

$$Q = \sqrt{S_1^2 + S_2^2 + 2S_1 S_2 \cos \gamma} ; \angle (\bar{Q}; 0_1 0_2) = \arctg \frac{S_1 - S_2}{S_1 + S_2} \operatorname{tg} \frac{\gamma}{2}$$

$$f = (n + \sqrt{n^2 - mp}) / m = (0,75 \dots 2)$$

$$\begin{cases} m = 80_m (i+1)^2 \\ n = 2(i+1)(1600 - 1,57 D_m (i+1)) \\ p = D_m (i-1)^2 \end{cases}$$

$$a' = 10_m (i+1)$$

$$L_{\text{stoc}} \geq 2a' + \frac{\pi}{2} D_m (i+1) + \frac{D_m^2 (i-1)^2}{4a'}$$

$$a = 0,25 [(L - W) + \sqrt{(L - W)^2 - 8C}]$$

$$W = \pi D_m (i-1) / 2$$

$$C = D_m^2 (i-1)^2 / 4$$

$$z \geq \frac{PK_f d}{P_o}$$

$$\begin{cases} K_o = 1..1,8 \\ d = 1..1,5; (\beta = 180^\circ \dots 90^\circ) \end{cases}$$

ARO - 9

CURELE TRAPEZOIDALE
ABACA PENTRU CURELE ÎNGUSTE
($v_s = 40$ m/s)

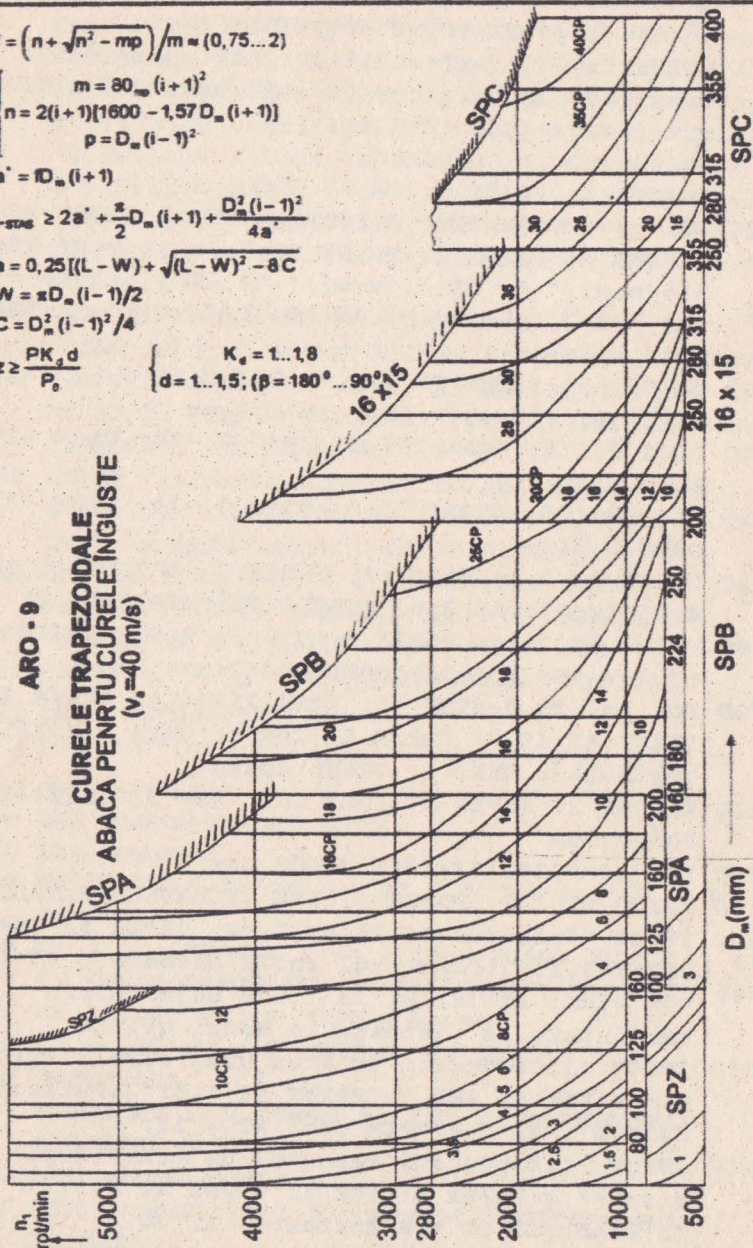

 D_m (mm)

Fig 13.8 Diagrama pentru calculul curelelor trapezoidale înguste

Programul de calcul aferent este următorul:

```

8 LET ch1=9000: LET ch2=9100: LET apasa=9200: LET
  box=9300: LET efect=9400: LET udg=9500: DEF FN
  r(x)=INT x+(x <> INT x)
9 GOSUB efect: GOSUB udg
10 BORDER 1: PAPER 0: INK 0: CLS
20 FOR n=7 TO 38 STEP 8: FOR m=7 TO 0 STEP -1:
  BORDER m: PAPER m: PRINT AT 0,(n-m);" "; AT 21,
  31-(n-m);" "; AT 1,(n-m);" "; AT 20,31-(n-m):
  " "; AT 2,(n-m);" ": AT 19,31-(n-m);" ": NEXT m:
  NEXT n
30 BRIGHT 1: PAPER 7: INK 7: PRINT AT 3,0:"(32sp)";
  AT 18,0;"(32sp)": REM (sp)=blanc
40 PLOT 0,175: DRAW 255,0: DRAW 0,-38: DRAW -255,0:
  DRAW 0,38
50 PLOT 2,173: DRAW 251,0: DRAW 0,-34: DRAW -251,0:
  DRAW 0,34
60 PRINT AT 2,1; FLASH 1; PAPER 0; BRIGHT 1; INK
  4;"(32sp)"; AT 2,6"CURILE TRAPEPEZOIDALE"
90 PRINT AT 7,10; PAPER 0; INK 6;"M.M.POPOVICI"; AT
  10,10: INK 7;"(2sp)SOFTWARE(2sp)"
100 FOR i=7 TO 0 STEP -1: POKE 23606,i: PAUSE 3:
  PRINT AT 14,8; PAPER 6; INK 1; OVER 0;"SINTEZA
  OPTIMALA": NEXT i: GOSUB apasa
110 BORDER 2: PAPER 2: CLS: LET a$="0101512817":
  GOSUB box
120 LET a$="0202152817": GOSUB box
130 PRINT AT 3,7; PAPER 5; INK 0;"DATELE PROBLEMEI"
140 INPUT "Introduceti DA[mm]",dm: PAPER 5: INK 1:
  PRINT 5,3;"1)Diametrul rotii motoare"; AT 6,8;
  "DA"=;dm; AT 16,15;"mm": REM majusculele
  subliniate se tasteaza in modul grafic
150 INPUT "Introduceti puterea unei curele PC[CP]",
  po: PAPER 5: INK 1: PRINT AT 7,3;"2)Puterea unei
  curele"; AT 8,8;"PC"=;po; AT 8,15;"CP"
160 INPUT "Introduceti raportul de trans(2sp)
  mitere",g:PAPER 5: INK 1: PRINT AT 9,3;
  "3)Raportul de transmitere"; AT 10,8;"i"=;g
170 INPUT"Introduceti puterea transmisa(3sp)P[CP]",

```



```

put: PAPER 5: INK 1: PRINT AT 11,3;"4)Puterea
transmisa"; AT 12,8;"P=";put; AT 12,15;"CP"
175 INPUT "Introduceti turatia rotii(7sp)motoare
n[rot/min]",tur; PAPER 5: INK 1:PRINT AT 13,5;
"5)Turatia motoare"; AT 14,8;"n=";tur; AT
14,15;"rot/min"
180 GOSUB apasa: RANDOMIZE USR 60000
190 BORDER 2: PAPER 2: CLS : LET a$="0101512819":
GOSUB box
200 LET a$="0202152819": GOSUB box
210 PRINT AT 3,7; PAPER 5; INK 0;"DATELE PROBLEMEI"
220 INPUT"Introduceti coef.dinamic kD=1...2", k:
PAPER 5: INK 1: PRINT AT 9,3;" 6)Coeficientul
dinamic"; AT 11,8;"kD=";k
230 INPUT "Introduceti coef.de infasurare (2sp)
d=1...1,5",d: PAPER 5: INK 1: PRINT AT 13,3;" 7)
Coef.de infasurare"; AT 15,8;"d=";d
240 LET a$="0617471503": GOSUB box: PRINT AT 18,8;
PAPER 7;INK 2;"CORECT (d/n)?" : PAUSE 0
250 RANDOMIZE USR 60000: CLS: GOSUB chl
260 PRINT AT 3,3;"COMPATIBILITATEA PROBLEMEI"
270 LET m=8*dm*((g+1)2)
280 LET n=2*(g+1)*(1600-1.57*dm*(g+1)): LET p=
dm*((g-1)2)
290 LET num=n+SQR(n*n-m*p)
300 LET s=num/m
302 PRINT AT 5,3;"m=";m; AT 7,3;"n=";n; AT 9,3;
"p=";p
310 IF s >= .75 AND s <= 2 THEN PRINT AT 12,6; PAPER
4; INK 6;"PROBLEMA COMPATIBILA"; PAPER 7; INK 0;
AT 16,10;"f=";s: GOSUB apasa
320 IF s<.75 OR s>2 THEN GO TO 7000
340 GOSUB chl
350 PRINT AT 3,1; PAPER 5; INK 1;"REZULTATELE
SINTEZEI OPTIMALE"
370 LET ast=s*(g+1)*dm
380 LET lung=2*ast+1.57*dm*(g+1)+(dm*dm+(g-1))/
(4*ast)

```

```

390 DIM b(31): REM lungimile standardizate ale
    curelelor
400 RESTORE 410: FOR i=1 TO 31: READ b(i): NEXT i
410 DATA 400,450,500,560,630,710,800,900,1000,1120,
    1250,1400,1600,1700,1800,2000,2340,2500,2800,
    3150,3350,3750,4000,5000,5600,6300,7100,8000,
    9000,10000,11200
420 FOR i=1 TO 31: IF lung<b(1) THEN LET lp=b(1): GO
    TO 500
430 IF lung=b(i) THEN LET lp=b(i): GO TO 500
440 IF lung>b(i+1) THEN LET lp=b(i+1): GO TO 500
450 IF lung>b(31) THEN PRINT AT 12,1;"NU EXISTA
    CUREA STANDARDIZATA"; AT 16,8:"Reluati calculul"
    AT 18,0;"cu alta solutie din abaca ARO-9": BEEP
    .02,40: PAUSE 0: GO TO 110
460 NEXT i
500 PRINT AT 9,2;"LUNGIMEA PRIMITIVA A CURELEI"; AT
    10,4;"L=";lp;" mm"
510 LET dmare=g*dm: PRINT AT 6,2;"DIAMETRUL ROTII
    CONDUSE"; AT 7,4;"DB="; FN r(dmare);" mm"
520 LET w=1.57*dm*(g+1)
530 LET h=dm*dm*(g-1)*(g-1)/4
540 LET a=(lp-w)+SQR((lp-w)2-8*h): LET antrax=a/4
550 PRINT AT 12,2;"ANTRAXUL"; AT 13,4;"a=" ; FN
    r(antrax);" mm"
560 LET z=put*k*d/po
570 PRINT AT 15,2;"NUMARUL DE CURELE"; AT 16,6;"z=";
    FN r(z)
580 GOSUB apasa
581 GOSUB chl: LET mt=70000*put/tur: LET gama=
    (dmare-dm)/antrax: LET v=PI*tur*dm/60000: LET
    niu=0.35+0,012*v: LET beta=PI-gama
582 LET x=(2*mt/(.1*dm)): LET y=2.72↑(niu*beta): LET
    r=(y-1): LET s1=x*y/r
583 LET s2=x/r
584 LET forta=SQR(s1*s1+s2*s2+2*s1*s2*COS(gama))
585 LET xx=s1-s2: LET yy=s1+s2: LET t=TAN gama/2:
    LET unghi=ATN(xx/yy)*t*(180/PI)

```

```

586 PRINT AT 1,2;"FORTELE DIN CUREA"; AT 4,2;"Ramura
motoare"; AT 5,4;"S1="; FN r(s1);" daN"; AT 7,2;
"Ramura condusa"; AT 8,4;"S2="; FN r(s2);" daN";
AT 10,2;"Forta rezultanta"; AT 11,4;"Q="; FN
r(forta);" daN"; AT 13,2;"Inclinarea rezultantei
fata de"; AT 14,2;"axa rotilor :"; AT 14,16; FN
r(unghi); " grade"
587 GOSUB apasa
590 GOSUB ch2: PRINT AT 11,2; PAPER 7; INK 2; FLASH
1;"INTRODUCETI ALTE DATE (d/n)?: PAUSE 0
600 GO TO 110*(INKEY$="d")+610*(INKEY$="n")
610 RANDOMIZE USR 60000: BORDER 7: PAPER 7: INK 0:
CLS: GOSUB ch2
620 PRINT AT 3,5;"M.M.POPOVICI SOFTWARE"; AT 9,8;
PAPER 7; INK 2;"TRANSMISII PRIN"; AT 11,6; PAPER
7; INK 2;"CURELE TRAPEZOIDALE"
630 FOR i=7 TO 0 STEP -1: POKE 23606,i: BEEP
.005,10: PRINT AT 15,7; PAPER 6; INK 4; OVER
0;"(SINIBIA OPTIMALA)": NEXT i: PAUSE 0: STOP
700 PRINT AT 11,3;"PROBLEMA INCOMPATIBILA !!!"; AT
15,5; OVER 1;"Modificati solutia cu(16sp)abaca
ARG-9": FOR y=10 TO 0 STEP -1: BEEP .005 y:
PAUSE 1: BEEP .005,14: PAUSE 1; BEEP.005,14:
NEXT y: GOSUB apasa: GO TO 110
9000 BORDER 5: PAPER 6: INK 2 CLS
9001 PLOT 15,0: DRAW 225,0: DRAW 15,15: DRAW 0,145:
DRAW -15,15: DRAW -225,0: DRAW -15,-15: DRAW
0,-145: DRAW 15,-15: DRAW 0,15: DRAW -15,0: PLOT
255,15: DRAW -15,0: DRAW 0 -15: PLOT 255,160;
DRAW -15,0: DRAW 0,15: PLOT 0,160: DRAW 15,0:
DRAW 0,15
9002 PRINT AT 0,0; OVER 1; PAPER 5;"(2sp)";AT 0,30;
"(2sp)"; AT 1,0;"(2sp)"; AT 1,30;"(2sp)"; AT
20,0; "(2sp)"; AT 20,30;"(2sp)"; AT 21,0;
"(2sp)"; AT 21,30;"(2sp)":RETURN
9100 LET t$="(15sp)": PAPER 6: INK 0: BORDER 7: CLS:
PLOT 0,0: DRAW 255,0: DRAW 0,175: DRAW -255,0:
DRAW 0,-175
9101 PRINT AT 1,2; PAPER 5;t$;t$(2 TO): FOR i=2 TO

```

```

20: PRINT AT i,1: PAPER 7;t$;t$(2 TO): AT
i-1,30: PAPER 5;" ": NEXT i
9102 RETURN
9200 LET a$="(4sp)APASATI O TASTA OARECARE(4sp)"
9201 PRINT # 1; AT 1,0;a$
9202 LET a$=a$(32)+a$(TO 31)
9203 PAUSE 5: IF INKEY$="" THEN GO TO 9201
9204 CLS : RETURN
9300 OVER 1: PRINT AT VAL a$(3 TO 4),0;
9301 FOR I=1 TO VAL a$ (9 TO)
9302 PRINT PAPER 8; INK 8; FLASH 8; BRIGHT 8; TAB VAL
a$(1 TO 2); INK VAL A$(5); PAPER VAL a$(6); TAB
VAL a$(TO 2)+VAL a$(7 TO 8)-1;" "
9303 NEXT i: RETURN
9400 RESTORE 9401: FOR n=1 TO 17: READ a: POKE
60000+n,a: NEXT n
9401 DATA 6,127,33,0,88,237,95,119,35,62,91,188,32,
247,16,242,201
9402 RETURN
9500 RESTORE 9501: FOR q=0 TO 31: READ a:POKE
65368+q,a: NEXT q
9501 DATA 0,0,0,0,248,168,168,168,0,0,0,0,136,216,
168,136
9502 DATA 0,0,0,0,96,144,144,96,0,0,0,16,16,112,144,
112
9503 RETURN

```

Aplicatie numerică : curea SPA cu $D_m = 125$ mm. $P_o = 10$ CP.
 $n_1 = 2800$ rot/min, $i = 2$, $P = 20$ CP, $k_d = 1,5$ și $d = 1,1$ rezultă: $m = 9000$:
 $n = 6067,5$; $p = 125$; $f = 1,3379526$ (problema compatibilă); $D_M = 250$ mm:
 $L = 1'00$ mm; $a = 502$ mm; $z = 4$ curele; $S_1 = 100$ daN ; $S_2 = 20$ daN;
 $Q = 110$ daN; $\angle ((\overline{Q}; 0_1 0_2)) = 5$ grade.

13.8. SINTEZA OPTIMALĂ A ANGRENAJELOR CILINDRICE EXTERIOARE

Angrenajele reprezintă organe de mașini prezente în structura

oricărei mașini. Din acest motiv există un interes aparte pentru studiul și proiectarea lor optimă. În același timp trebuie menționat că proiectarea angrenajelor este complicată și implică stăpînirea unui număr important de noțiuni teoretice. Pentru ca cititorul să poată înțelege și aplica programul de calcul este necesar să se facă o prezentare prealabilă a noțiunilor ce stau la baza sintezei optimale.

Angrenajele cilindrice reprezintă mecanisme trilaterale alcătuite din:

- trei *elemente* dintre care două mobile (*pinionul 1* și *roata condusă 2*) și unul fix (hașurat și notat cu 0);
- trei *cuple cinematice* dintre care două inferioare de rotație (*lagărele* cu alunecare sau cu rostogolire 0_1 și 0_2) și o cuplă superioară M în care se produce fenomenul de angrenare a roților (fig.13.9.a).

Calculul angrenajelor constă în calculul cuplei superioare M .

Dintele roților se realizează cu un profit compus din două *evolvente* antisimetrice, limitate la exterior de către *cercul exterior* (de cap) C_a și la interior de către *cercul interior* (de picior) C_f - fig.13.9.b. Folosirea evolventei (curbă trasată de punct invariabil aparținând unei drepte care se rostogolește fără alunecare pe un cerc) pentru profilarea dinților se datorează avantajului de a *menține constant raportul de transmitere a mișcării (i) indiferent de precizia de montaj a roților și uzura lagărelor*.

Angrenajele cilindrice utilizate pe scară largă au *dinții dreپți* sau *inclinăți* cu unghiul β (fig.13.9.c).

Raportul de transmitere al angrenajului este:

$$i = n_1 / n_2 = z_2 / z_1 = R_{w2} / R_{w1}$$

unde n_1, n_2 [rot/min] - turațiile roților, z_1, z_2 - numerele de dinți, iar R_{w1}, R_{w2} - razele cercurilor de *rostogolire*.

Studiul valorilor optime pentru raportul de transmitere la angrenajele cu o treaptă a condus la concluzia $i \in [1...5.8]$. Valorile standardizate ale raportului de transmitere sînt :

$$i = [1; 1,12; 1,25; 1,4; 1,6; 1,8; 2; 2,25; 2,5; 2,8; 3,15; 3,55; 4; 4,5; 5; 5,6; 6,3; 7,1; 8; 9]$$

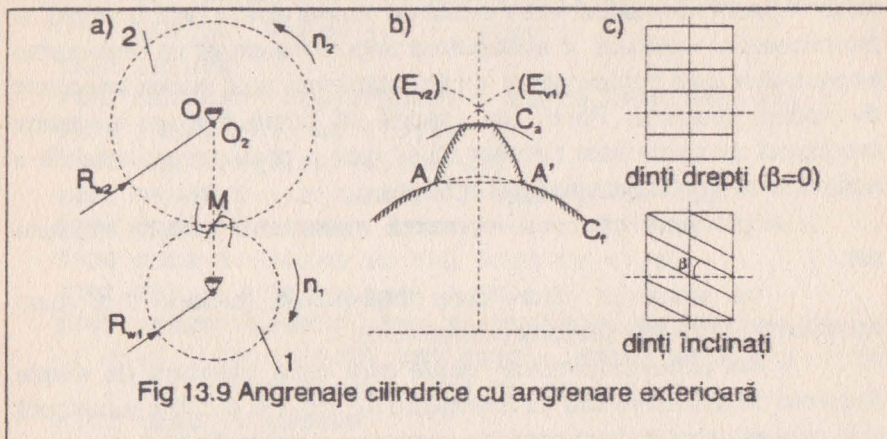


Fig 13.9 Angrenaje cilindrice cu angrenare exterioară

Materialele pentru angrenajele din construcția de mașini sînt indicate în diagrama din fig. 13.10,a, în care fiecare material este înscris într-un dreptunghi în care se precizează (fig.13.10,b):

-simbolul materialului (ex.:28TMC12);

-tensiunea admisibilă la oboseală (σ_a) și presiunea admisibilă la contact (p_a), exprimate în daN/mm² (ex.: pentru 28TMC12 cele două valori sînt $\sigma_a = 28,5 \text{ daN/mm}^2 = 2850 \text{ daN/cm}^2 = 285 \text{ MPa}$; $p_a = 180 \text{ daN/mm}^2 = 18000 \text{ daN/cm}^2 = 1800 \text{ MPa}$);

-caracteristica materialului $\delta_2 = E\sigma_a/p_a^2$ (ex.:pentru 28TMC12 această valoare este $\delta_2 = 18$).

Materialele se împart în două categorii și anume :

-materiale grupa A (materiale tenace) cu performanțe mecanice medii sau reduse avînd costuri mici dar care realizează volume mari de material;

-materiale grupa B (materiale dure) cu performanțe mecanice superioare avînd costuri ridicate dar care realizează volume mici de material.

În timpul funcționării angrenajelor aceste materiale trebuie să asigure rezistența corespunzătoare solicitării centrifugale a roții și danturii sau, altfel spus, materialul să fie compatibil. Condiția de compatibilitate constă în a calcula coeficientul de compatibilitate ρ care trebuie să

satisfacă relația: $\rho \leq \rho_1, \rho_2$

unde ρ_1 - restricția vitezei periferice,

ρ_2 - restricția solicitării centrifugale a discului roților (cu valorile indicate în fig. 13.10 în funcție de tipul materialului).

• În lucrarea "Sinteze optimale în construcția de mașini" [6], prof. Octavian Rădulescu stabilește următoarele criterii optimale:

- volum minim-minimorum, cînd se obține roata limită cu cel mai mic număr de dinți și implicit cel mai mic volum, dar care este mai puțin rezistentă deoarece are concentratori de tensiuni la baza dintelui ; numărul de dinți necesar pinionului se stabilește cu relația :

$$z_1 = (\max) \{ 9(i+1)/i ; 2/0,06(i+1) \}$$

- ponderat (volum minim și rezistență mare), cînd se obține roata de bază cu număr mic de dinți și robustețe apreciabilă ; în acest caz numărul de dinți necesar pinionului este dat de relația:

$$z_1 = (\max) \{ 13 ; 2/0,03(i+1) \}$$

-maxidurabilitate, cînd se obține angrenajul extrapolar ce asigură cea mai mare fiabilitate dar cu numere mari de dinți și implicit cu volume mari de material; pinionul angrenajului extrapolar are numărul de dinți dat de relația :

$$z_1 = (\max) \{ 25 ; 66/i \}$$

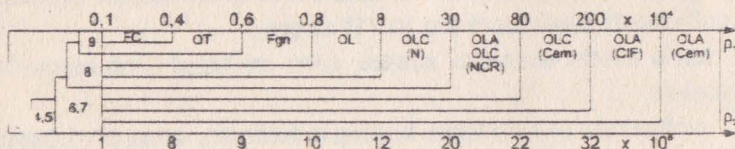
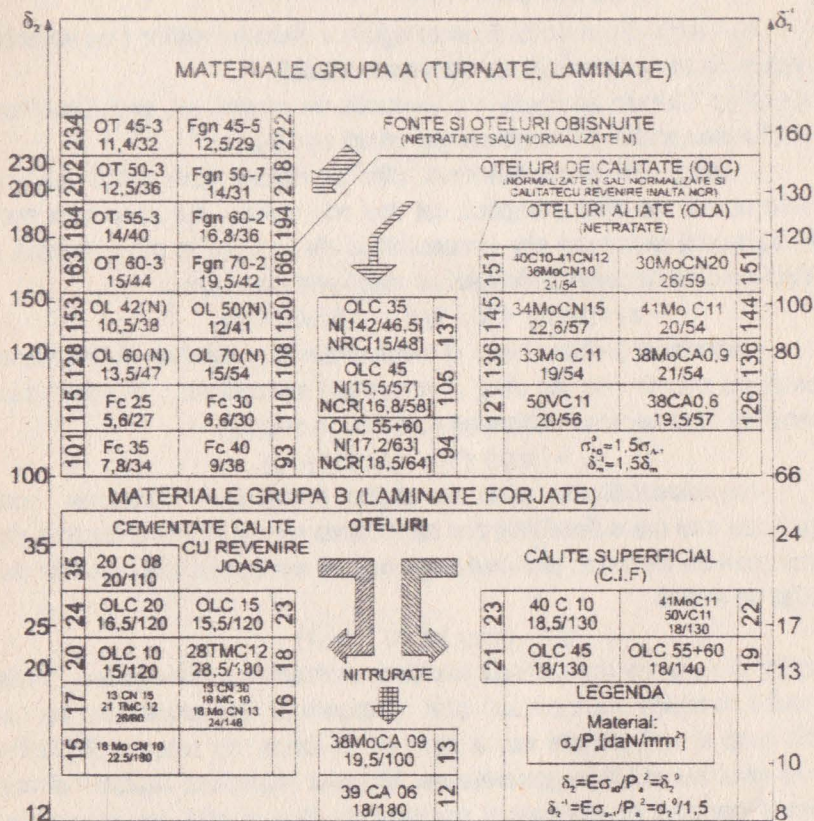
• Roțile dințate se danturează folosind cremaliera generatoare care realizează modulul standardizat prin rostogolirea fără alunecare pe un anumit cerc al roții dințate numit cerc de divizare, cu oricare din liniile sale de nivel numită linie generatoare. Modulul reprezintă raportul dintre diametrul cercului de divizare și numărul de dinți al roții, iar cremaliera este o roată dințată particulară (fig. 13.11) ce are :

- toate razele cercurilor infinite, ceea ce transformă cercurile în drepte paralele;

- evolventele transformate în drepte înclinate, ceea ce determină dinți cu formă trapezoidală și în număr infinit;

-modulul constant la orice linie de nivel, ceea ce permite standardizarea modulului cu următoarele valori (pentru mecanică generală și grea): $m[\text{mm}] = \{1; 1,125; 1,25; 1,375; 1,5; 1,75; 2; 2,25; 2,5; 2,75; 3; 3,5; 4; 4,5; 5; 5,5; 6; 7; 8; 9; 10; 11; 12; 14; 16; 18; 20; 22; 25; 28; 32; 40; 45; 50; 55; 60; 70; 80; 90; 100\}$

MATERIALE PENTRU ANGRENAJE



a)

b)

Simbolul materialului
 σ/P_d

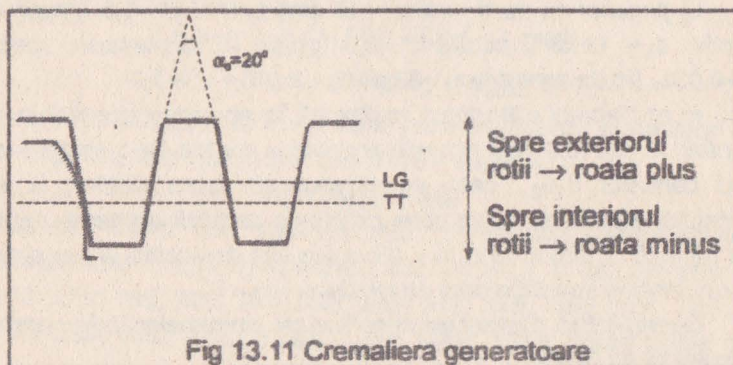
28 TMC 12
28,5/180

18

Fig 13.10 Materiale pentru angrenaje

Două roți dințate pot angrena numai dacă au același modul.

Orice dreaptă orizontală a cremalierii (LG) constituie o linie generatoare; un caz particular al acestor linii îl constituie *linia de referință* TT pe care arcul dintelui și arcul golului sînt egale.



- Profilul dintelui se obține deplasînd linia generatoare LG a cremalierii:

- fie c tre *exteriorul roții dințate*, c nd rezult  roata plus av nd unghiul de angrenare $\alpha > 20^\circ$; aceast  deplasare este limitat  de *ascuțirea excesiv  a dinților* ceea ce a impus ca arcul dintelui pe cercul de cap s  fie cel puțin egal cu valoarea k^*m , unde coeficientul de ascuțire $k^* = 0,3 \dots 0,5$ iar m este modulul roții (roata plus are dinții rezistenți la solicitarea compus  de încovoiere și compresiune de la baza dintelui, ca și la presiunea de contact hertzian dintre dinți);

- fie c tre *interiorul roții dințate*, c nd rezult  roata minus av nd unghiul de antrenare $\alpha < 20^\circ$; aceast  deplasare este limitat  de *subțaiere* (interferența dintre cremaliera generatoare și dintele roții) care se evit  translat nd linia generatoare LG deasupra liniei AA format  prin unirea punctelor de început ale evolventelor profilului dintelui (fig.13.9,b); roata minus are propriet ți similare roții plus dar  n sens invers;

Atunci c nd generarea roții dințate se face folosind linia de referință TT tangențială la cercul de divizare, se obține roata zero cu unghiul de angrenare $\alpha = 20^\circ$

Folosind aceste date  n cazul angrenajelor optime se deduc urm toarele:

a) *pinionul limită* are $z_1 = 11$ dinți pentru $k^* = 0,3$ (ascuțire normală la materialele din grupa A) și respectiv $z_1 = 13$ pentru $k^* = 0,5$ (ascuțire normală la materialele din grupa B), deplasarea specifică fiind dată de relația $x_1 = 1 - 0,06 z_1$ (roata zero are $z_1 = 17$ dinți);

b) *pinionul de bază* are $z_1 = 13$ dinți pentru $k^* = 0,3$ (grupa A) și respectiv $z_1 = 18$ dinți pentru $k^* = 0,5$ (grupa B), deplasarea specifică $x_1 = 1 - 0,03z_1$ (roata zero are $z_1 = 33$ dinți);

c) *angrenajul extrapolar* realizează începerea angrenării în polul angrenării M (fig.13.9,a) și această angrenare are loc de o singură parte a liniei centrelor O_1O_2 ceea ce diminuează uzura dinților; la roțile angrenajului extrapolar nu se pune problema ascuțirii excesive, numărul minim de dinți la pinion fiind $z_1 = 25$; roțile sînt deplasate unitar simetric avînd deplasările specifice date de relația $x_1 = x_2 = 1$.

Generalizînd, deplasările specifice ale pinioanelor roților optime se calculează cu relația:

$$x_1 = 1 - qz_1 \quad \text{unde} \quad q = \begin{cases} 0,06 & (\text{pinion limita}) \\ 0,03 & (\text{pinion de baza}) \\ 0 & (\text{pinion extrapolar}) \end{cases}$$

- Geometria angrenajului cilindric se stabilește cu relațiile din tabelul următor:

Tabelul 13.2

Dimensiunile \ Roțile	Pinion	Roata condusă
Diametrul de divizare	$D_1 = mz_1 / \cos \beta$	$D_2 = mz_2 / \cos \beta$
Diametrul cercului de cap	$D_{a1} = m[(z_1 / \cos \beta) + 2 + 2x_1]$	$D_{a2} = m[(z_2 / \cos \beta) + 2 + 2x_2]$
Diametrul cercului de picior	$D_{f1} = m[(z_1 / \cos \beta) - 2,5 + 2x_1]$	$D_{f2} = m[(z_2 / \cos \beta) - 2,5 + 2x_2]$
Lățimea roților	$B = \Psi_D D_1 \cos \beta$	

în care : m - modulul standardizat ; z_1, z_2 - numerele de dinți; x_1, x_2 - deplasările specifice, $\beta = \arcsin((3/\Psi_D D_1))$ (cu $\beta = 0$ la dinții drepți), Ψ_D - coeficientul diametral.

Studiul restricțiilor cinematice ale angrenajelor a condus la concluziile ce urmează:

- a) *Restricția jocului de flanc nul*. Angrenarea este asigurată cînd

jocul de flanc, măsurat la nivelul cercurilor de rostogolire, este nul; acest joc este nul atunci cînd sînt egale arcele dintelui cu cele ale gurilor dintre dinți. Studiul analitic al acestei condiții a condus la expresia deplasării specifice sumă

$$x = x_1 + x_2 = (z_1/2)(1+i)x_0(\alpha) \text{ unde}$$

$$x_0(\alpha) = [(\operatorname{tg} \alpha - \alpha - (\operatorname{tg} \alpha_0 - \alpha_0)) / \operatorname{tg} \alpha_0]$$

în care $\alpha_0 = 20^\circ$; relația lui x servește pentru *alinierea antraxelor* adică pentru rotunjirea sau egalarea antraxului de referință

$$a_0 = mz_1(1+i)/2$$

la o valoare dorită sau standardizată ; antraxele standardizate au următoarele valori: $a[\text{mm}] = [40; 45; 50; 56; 63; 71; 80; 90; 100; 112; 125; 140; 160; 180; 200; 225; 250; 280; 315; 355; 400; 450; 500; 560; 630; 710; 800; 900; 1000; 1120; 1250; 1400; 1600; 1800; 2000; 2250; 2500]$

Se pot ivi două situații:

1) cînd a_0 [mm] este un *număr întreg* și eventual din gama valorilor standardizate atunci :

- antraxul angrenajului este $a = a_0$ [mm]

- unghiul de angrenare $\alpha = \alpha_0 = 20^\circ$ (angrenaj zero) ;

- deplasarea specifică sumă $x = x_1 + x_2 = 0$ de unde rezultă deplasarea specifică a roții conduse $x_2 = -x_1$.

2) cînd a_0 [mm] este un *număr real* și se rotunjește (aliniază) la o valoare dorită $a_r = a_0 \cos \alpha_0 / \cos \alpha$, se deduc succesiv :

$$\alpha = \arccos(0,94a_0/a_r); x_0(\alpha) = [(\operatorname{tg} \alpha - \alpha - (\operatorname{tg} \alpha_0 - \alpha_0)) / \operatorname{tg} \alpha_0;$$

$$x = [z_1(1+i)/2] x_0(\alpha); x_2 = x - x_1 .$$

Se menționează că pentru *angrenajele optime* unghiul de angrenare

$$\alpha = 16^\circ \dots 24^\circ . . .$$

iar angrenajele extrapolare nu se pot alinia fiind angrenaje zero.

La angrenajul cilindric cu *dinții înclinați* formulele sînt identice cu mențiunea că $\alpha = \alpha_n$ considerînd secțiunea normală:

$$\alpha_n = \arccos(0,94a_0/a_r); x(\alpha) = [(\operatorname{tg} \alpha_n - \alpha_n - (\operatorname{tg} \alpha_0 - \alpha_0)) / \operatorname{tg} \alpha_0 ;$$

$$x = [z_1(1+i)/2] x_0(\alpha); x_2 = x - x_1 .$$

b) *Restricția interferenței.* Interferența este fenomenul de pătrundere a vîrfului dintelui roții conduse la baza dintelui pinionului, avînd ca urmări uzura dintelui pinionului, șocuri în funcționare și uneori chiar blocarea angrenajului. Studiul analitic al interferenței a condus la concluzia că angrenajele optime (limită, de bază, extrapolare) *nu interferează* cu excepția angrenajului limită minus care se va evita.

c) *Restricția gradului de acoperire.* Continuitatea angrenării impune să existe permanent în angrenare cel puțin o pereche de dinți și se apreciază prin *gradul de acoperire* ε ; pentru o angrenare silențioasă și fără șocuri $\varepsilon > 1,1$. Studiul interferenței efectuat pentru angrenajele optime a condus la următoarele date :

- la angrenajul limită $\varepsilon_{\alpha} = 1,12 \dots 1,70$;
- la angrenajul de bază $\varepsilon_{\alpha} = 1,4 \dots 1,85$;
- la angrenajul extrapolar $\varepsilon_{\alpha} = 1,44 \dots 1,64$.

pentru $\alpha = 16^{\circ} \dots 24^{\circ}$. La angrenajul cu dinți înclinați $\varepsilon \geq 2$ indiferent de valoarea unghiului α .

Prin urmare angrenajele optime asigură grade de acoperire cu valori ridicate pentru orice unghi de angrenare.

- Studiul *restricțiilor dinamice* s-a concretizat în următoarele concluzii; a) Soluția cinematică pentru stabilirea materialului

$$\delta_1 = 2iz_1 / (1 + i)$$

Materialul se stabilește cu ajutorul fig.13.10,a, în care sînt trecute proprietățile de bază: $p_a, \sigma_a, \delta_2 = E\sigma_a / p_a$, aceasta din urmă trecută în ordonata din stînga pentru ciclul pulsator și în dreapta pentru ciclul alternant simetric (ciclurile de solicitare la oboseală).

Materialul de referință este cel care asigură *echiportanța* $\delta_1 = \delta_2$.

Pentru $\delta_2 > \delta_1$ se impune materialul cu $(\max)p_a$, iar pentru $\delta_2 \leq \delta_1$ se va prefera materialul cu $(\max)\sigma_a$.

Cînd se impune materialul (δ_2) se stabilește soluția cu condiția $\delta_1 \leq \delta_2$.

b) Functia de viteză determinată pe baza recomandărilor privind lățimea roților

$$v = cPN_1^2 \frac{\delta}{i\sigma_a} \text{ unde } c = \begin{cases} 70 \text{ (dinti drepti)} \\ 60 \text{ (dinti inclinati)} \end{cases}; \delta = (\max)\{\delta_1; \delta_2\}$$

c) Coefficientii de calcul

- *Coefficientul de repartizare a sarcinii pe dinte* (k_r) stabilit experimental în funcție de rigiditatea arborelui și starea flancurilor dinților

$$k_r = \begin{cases} 1,1 & \text{(materiale grupa A)} \\ 1,25 & \text{(materiale grupa B).} \end{cases}$$

Coefficientul dinamic (k_d) stabilit experimental în funcție de clasa de precizie, viteza periferică și duritatea flancurilor. Se menționează că roțile dințate se uzinează în raport cu o anumită clasă de precizie după cum urmează :

- clasele 9 și 8 pentru mașini grele cu turații reduse (mașini agricole sau pentru construcții) ;
- clasele 7,6,5 pentru mașinile uzuale ;
- clasele 4,3 pentru roți etalon sau pentru turbine.

Stabilirea valorii coeficientului dinamic se face cu relațiile :

$$k_d = \begin{cases} (400 + 1,5 v) / (600 + v) & \text{- clasa 5} \\ (150 + 1,5 v) / (300 + v) & \text{- clasa 6} \\ (30 + 1,5 v) / (100 + v) & \text{- clasa 7} \\ (200 + 1,5 v) / (200 + v) & \text{- clasa 8} \end{cases}$$

Cu cât numărul clasei este mai mic cu atât prelucrarea mecanică este mai bună.

- *Coefficientul de majorare a momentului de răsucire* $k = k_r k_d$
- *Coefficientul axial* (Ψ_A) stabilit experimental în funcție de rigiditatea arborelui, viteza periferică a roților și felul ungerii

$$\Psi_A = \begin{cases} (1500 + 0,15v) / (1800 + v) & \text{- pentru reductoare} \\ 0,015 & \text{- pentru transmisii deschise} \\ & \text{si cutii de viteze cu roți} \\ & \text{baladoare (deplasabile)} \end{cases}$$

- *Coefficientul diametral* (Ψ_D) stabilit în funcție de rigiditatea arborelui:

$$\Psi_D = (\min)\{\Psi_D', \Psi_D''\} \quad \text{unde} \quad \Psi_D' = \begin{cases} 0,6 & (\text{pinion în consolă}) \\ 1,2 & (\text{pinion nesimetric}) \\ 1,6 & (\text{pinion simetric}) \end{cases}; \quad \Psi_D'' = \frac{(1+i)\Psi_A}{2}$$

După cum se observă, montajul pinionului în raport cu lagărele arborelui care îl susține (în consolă, montat nesimetric sau simetric) are importanța sa. În relația lui Ψ_D'' intervine și raportul de transmitere i .

d) Compatibilitatea materialului. Ținând seama de recomandările privind limitarea vitezei periferice a roților în funcție de clasa de precizie precum și de solicitarea centrifugală a discului roții, se determină coeficientul de compatibilitate

$$\rho = u P N_1^2 \frac{k}{\Psi_D} \frac{(1+i)}{i} \delta \leq \rho_1; \rho_2 \quad \text{unde} \quad u = \begin{cases} 1 & (\text{dinti drepti}) \\ 0,9 & (\text{dinti inclinați}) \end{cases}$$

e) Modulul calculat

$$m_c [\text{cm}] = \frac{1,31}{z_1} \sqrt[3]{\frac{M_t k (1+i) \delta}{\Psi_D i \sigma_a}} \quad \text{unde} \quad M_t = 70P / N_1 \cos^3 \beta [\text{daNcm}]$$

Se alege modulul standardizat $m_{STAS} [\text{mm}] \geq m_c [\text{mm}]$.

Dacă $m_c \neq m_{STAS}$ se recalculează

$$\Psi_D = \frac{1,31^3 M_t k (1+i) \delta}{(m z_1 \cos \beta)^3 i \sigma_a}; \quad \sin 2\beta = 3 / \Psi_D z_1 \text{ și unghiul } \beta$$

(pentru dinții drepti $\beta = 0$).

În cazul angrenajelor cu dinți înclinați se stabilește în plus numărul de dinți pentru pinion

$$z_1 = z_1 \cos^3 \beta \quad (\text{în raport cu } z_1 \text{ de la soluția cinematică și unghiul } \beta).$$

Cu aceste date se determină în continuare numărul de dinți pentru roata condusă $z_2 = iz_1$ și geometria angrenajului cu relațiile din tabelul 13.2.

f) Fortele din angrenaj (fig.13.12):

- pentru angrenaj cu **dinți drepti** (fig.13.12,a)

$$F_t [\text{daN}] = \{2M_t \cos \alpha / 0,94D_1 [\text{cm}]\} \quad \text{- forța tangențială}$$

$$F_n [\text{daN}] = \{2M_t \sin \alpha / 0,94D_1 [\text{cm}]\} \quad \text{- forța normală (radială)}$$

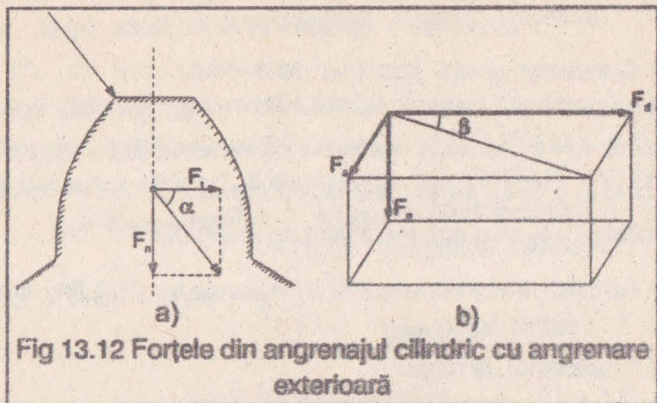


Fig 13.12 Forțele din angrenajul cilindric cu angrenare exterioară

- pentru angrenajul cu dinți înclinați (fig.13.12,b)

$$F_{tf}[\text{daN}] = \{2M_t \cos \alpha_f / D_1 \cos \alpha_{of} \quad - \text{forța tangențială normală}$$

$$F_r[\text{daN}] = F_{tf} \operatorname{tg} \alpha_f \quad - \text{forța normală (radială)}$$

$$F_a[\text{daN}] = F_{tf} \operatorname{tg} \beta \quad - \text{forța axială}$$

$$\text{unde } \alpha_f = \begin{cases} \alpha_{of} & - \text{antrax nealiniat} \\ \operatorname{arc} \left(\cos \frac{a_0}{a_r} \cos \alpha_{of} \right) & - \text{antrax aliniat} \end{cases};$$

$$\alpha_{of} = \operatorname{arc}[\operatorname{tg} 20^\circ / \cos \beta]$$

• Cu aceste precizări teoretice algoritmul sintezei optimele a angrenajelor cilindrice este următorul :

Date : *puterea transmisă* $P[\text{CP}]$; *turația pinionului* n_1 [rot/min]; *clasa de precizie* (5; 6; 7; 8); *poziția pinionului* (în consolă, nesimetrică, simetrică); *tipul de angrenaj optimal* [angrenaj limită cu volum minim-minimorum-folosit la vehicule spațiale, navale și terestre de sport;- angrenaj de bază(ponderat) cu volum mic și rezistență mare - folosit la mașinile uzuale;- angrenaj extrapolar (cu maxdurabilitate) - folosit la mașinile de laborator și de uz casnic].

Rezolvare : 1) *Numărul de dinți (inițial)*

$$- \text{angrenajul limită} \quad : z_1 = (\max) \{ 9(1+l) / l; 2/0,06(1+l) \}$$

$$- \text{angrenajul de bază} \quad : z_1 = (\max) \{ 13; 2/0,03(1+l) \}$$

$$- \text{angrenajul extrapolar} \quad : z_1 = (\max) \{ 25; 66/l \}$$

unde $z_1 = (\text{proxim}) z_1 \in N$.

2) Constanta pentru stabilirea materialului $\delta_1 = 2iz_1 / (1+i)$; din fig.13.10,a se stabilește materialul pe baza următoarei analize :

- cînd $\delta_1 = \delta_2$ se alege materialul cu valoarea δ_2 (echiportanța) ;
- cînd $\delta_1 < \delta_2$ se alege materialul cu $(\max) p_a$;
- cînd $\delta_1 > \delta_2$ se alege materialul cu $(\max) \sigma_a$.

3) Funcția de viteză : $v = cPN_1^2 \delta / i \sigma_a$ unde $c = [70$ (dinți dreپți); 60 (dinți înclinați)]

4) Coeficienții de calcul

$$k_r = \begin{cases} 1,1 & \text{- materiale grupa A} \\ 1,25 & \text{- materiale grupa B} \end{cases} \text{ (coeficientul de repartizare)}$$

$$k_d = \begin{cases} (400 + 1,5 v) / (600 + v) & \text{- clasa 5} \\ (150 + 1,5 v) / (300 + v) & \text{- clasa 6} \\ (30 + 1,5 v) / (100 + v) & \text{- clasa 7} \\ (200 + 1,5 v) / (200 + v) & \text{- clasa 8} \end{cases} \text{ (coeficientul dinamic)}$$

$$k = k_r k_d \quad \text{(coeficientul de majorare)}$$

$$\Psi'_A = \begin{cases} (1500 + 0,15 v) / 1800 + v & \text{- reductoare} \\ 0,015 & \text{- transmisii deschise si cutii de viteza cu roti baladoare} \end{cases}$$

(coeficientul axial)

$$\Psi_D = \begin{cases} 0,6 & \text{- montaj pinion în consola} \\ 1,2 & \text{- montaj pinion nesimetric între lagare} \\ 1,6 & \text{- montaj pinion simetric între lagare} \end{cases}$$

$$\Psi_D^0 = 0,5(1+i)\Psi_A$$

$$\Psi_D = (\min) \{ \Psi_D^0; \Psi_D^0 \}$$

5) Unghiul de înclinare $\sin 2\beta = 3 / \Psi_D z_1 \rightarrow \beta$

6) Modulul calculat : $m_c [\text{cm}] = \frac{1,31}{z_1} \sqrt[3]{\frac{M_t k(1+i)\delta}{\Psi_D i \sigma_a}}$ unde

$$M_t = 70P / N_1 \cos \beta ; N_1 = n_1 / 1000$$

Se alege modulul standardizat $m[\text{mm}] = (\text{proxim}) \quad m_{\text{STAS}} \geq m_0[\text{mm}]$

Dacă $m_c \neq m_{\text{STAS}}$ se recalculează

$$\Psi_D = 1,31^3 M_t k (1+i) \delta / (m z_1 \cos \beta)^3 \mid \sigma_a \text{ și } \sin 2\beta = 3 / \Psi_D z_1 \rightarrow \beta$$

7) Numerele de dinți și deplasarea specifică a pinionului

$$z_1 = z_{1\text{inițial}} \cos^3 \beta \quad (\text{numai la dinții înclinați})$$

$$x_1 = 1 - q z_1 \quad \text{unde} \quad q = \begin{cases} 0,06 & (\text{angrenaj extrapolar}) \\ 0,03 & (\text{angrenaj de baza}) \\ 0 & (\text{angrenaj limita}) \end{cases}$$

$$z_2 = i z_1 \in \mathbb{N}; \quad i = z_2 / z_1$$

8) Antraxul de referință

$$a_0 = m_{\text{STAS}} z_1 (1+i) / \cos \beta$$

Dacă nu se aliniază rezultă angrenaj zero deplasat simetric (roata condusă avînd deplasarea specifică $x_2 = -x_1$) și se trece la geometria angrenajului.

Dacă antraxul se aliniază la o valoare a_r [mm], atunci se calculează succesiv:

$$z_1 = 2a_r \cos \beta / m_{\text{STAS}}(1+i) \in \mathbb{N}; \quad z_2 = i z_1 \in \mathbb{N}$$

$$\cos \alpha = 0,94 a_0 / a_r \quad (\text{la dinții dreپti}); \quad \cos \alpha_n = 0,94 a_0 / a_r \quad (\text{la dinții înclinați})$$

$$x_0(\alpha) = [\text{tg } \alpha - \alpha - (\text{tg } \alpha_0 - \alpha_0)] / \text{tg } \alpha_0; \quad \alpha_0 = 20^\circ \quad (\text{la dinții dreپti})$$

$$x_0(\alpha) = [\text{tg } \alpha_n - \alpha_n - (\text{tg } \alpha_0 - \alpha_0)] / \text{tg } \alpha_0 \quad (\text{dinții înclinați})$$

$$x = m z_1 (1+i) / 2 \cos^3 \beta x_0(\alpha); \quad x_2 = x - x_1$$

9) Geometria angrenajului

$$\text{Pinion} \begin{cases} \text{Diametrul de divizare} & D_1 = m z_1 / \cos \beta \\ \text{Diametrul de cap} & D_{a1} = m [(z_1 / \cos \beta) + 2 + 2x_1] \\ \text{Diametrul de picior} & D_{f1} = m [(z_1 / \cos \beta) - 2,5 + 2x_1] \end{cases}$$

$$\text{Roata condusa} \begin{cases} \text{Diametrul de divizare} & D_2 = m z_2 / \cos \beta \\ \text{Diametrul de cap} & D_{a2} = m [(z_2 / \cos \beta) + 2 + 2x_2] \\ \text{Diametrul de picior} & D_{f2} = m [(z_2 / \cos \beta) - 2,5 + 2x_2] \end{cases}$$

$$\text{Lățimea roților: } B = \Psi_D D_1 \cos \beta$$

10) Forțele din angrenaj

- angrenajul cilindric cu dinți drepti

$$\begin{cases} F_t = 2M_t \cos \alpha / 0,94D_1 \text{ [cm]} \\ \quad \text{(forța tangentială)} \\ F_n = 2M_t \sin \alpha / 0,94D_1 \text{ [cm]} \\ \quad \text{(forța normală)} \end{cases}$$

(la angrenajul cu antraxul nealinat $\alpha = \alpha_0 = 20^\circ$)

- angrenajul cilindric cu dinți înclinați:

$$\begin{cases} F_{tf} = 2M_t \cos \alpha_f / D_1 \text{ [cm]} \cos \alpha_{of} \text{ (forța tangentială)} \\ F_r = F_{tf} \operatorname{tg} \alpha_f \text{ (forța normală)} \\ F_a = F_{tf} \operatorname{tg} \beta \text{ (forța axială)} \end{cases}$$

$$\text{unde } \alpha_{of} = \begin{cases} \alpha_{of} & \text{- antrax nealinat} \\ \operatorname{arc} \left(\cos \frac{a_0}{a_r} \cos \alpha_{of} \right) & \text{- antrax aliniat} \end{cases};$$

$$\alpha_{of} = \operatorname{arc} [\operatorname{tg} 20^\circ / \cos \beta]$$

• Programul de calcul aferent este prezentat în cele ce urmează.

Ca și la programele anterioare mai întâi se definesc noile caractere grafice, iar în interiorul programului majusculele subliniate se tastează în modul grafic. De asemenea blancurile sînt indicate prin notația (sp).

```
8 LET ch1=9000; LET ch2=9100: LET apasa=9200: LET
box=9300: LET udg=9400: LET cortina=9500: LET
sctie=9600: LET x$="(32sp)": DEF FN w(a,b,c,d)=
100: DEF FN r(x)=INT x+(x <> INT x)
```

```
9 GOSUB udg
```

```
10 BORDER 0: PAPER 1: INK 7 CLS
```

```
20 FOR n=7 TO 37 STEP 8: FOR m=7 TO 0 STEP -1:
BORDER m: PAPER m: PRINT AT 0,(n-m);" "; AT 21,
31-(n-m);" "; AT 1,(n-m);" "; AT 20,31-(n-m);
" "; AT 2,(n-m);" "; AT 19,31-(n-m);" ": NEXT
m: NEXT n
```

```
30 BRIGHT 1: PAPER 7: INK 7: PRINT AT 3,0;"(32sp)";
AT 18,0;"(32sp)"
```

```
40 PLOT 0,175: DRAW 255,0: DRAW 0,-38: DRAW -255,0:
DRAW 0,38
```

```
40 PLOT 2,173: DRAW 251,0: DRAW 0,-34: DRAW -251,0:
   DRAW 0,34
60 PRINT AT 2,1; FLASH 1; PAPER 0; BRIGHT 1; INK 4;
   "(32sp)"; AT 1,6;"ANGRENAJE CILINDRICE"; AT 2,1;
   "CU DINTII DREPTI SAU INCLINATI"
110 PLOT 0,0: DRAW 255,0: DRAW 0,34: DRAW -255,0:
   DRAW 0,-34: PLOT 2,2: DRAW 251,0: DRAW 0,32:
   DRAW -251,0: DRAW 0,-32
120 LET i$="0106301062": GOSUB box
130 PRINT AT 7,10; INK 0;"M.M.POPOVICI"; AT 10,10;
   INK 7;"(2sp)SOFTWARE(2sp)"
140 OVER 0: FOR i=7 TO 0 STEP -1: POKE 23606,i:
   PAUSE 3: PRINT AT 14,8; INK 1;"SINTEZA
   OPTIMALA": NEXT i
150 GOSUB apasa: GOSUB cortina
155 BORDER 6: PAPER 6: INK 6: CLS
160 LET i$="02022281661": GOSUB box: PRINT AT 3,8:
   PAPER 2: INK 7;"DATELE PROBLEMEI"
170 INPUT "Introduceti puterea transmisa(3sp)P[CP]",
   put: PRINT AT 5,3;"1)Puterea P=";put;" CP"
180 INPUT "Introduceti turatia pinionului(2sp)n $\underline{a}$ 
   [rot/min]",tur: PRINT AT 6,3;"2)Turatia n $\underline{a}$ =";
   tur;" rot/min": LET N1=tur/1000
190 INPUT "Introduceti clasa de precizie",clasa :
   PRINT AT 7,3;"3)Clasa de precizie : ";clasa
200 INPUT "Introduceti pozitia pinionului (3sp)1)in
   consola(20sp)2)nesimetrice(19sp)3)simetrice",
   b): PRINT AT 8,3;" 4) Pozitia pinionului : "
210 IF b=1 THEN PRINT AT 9,5;"in consola  $\underline{M}$ =0,6"
220 IF b=2 THEN PRINT AT 9,5;"nesimetrice  $\underline{M}$ =1,2"
230 IF b=3 THEN PRINT AT 9,5;"simetrice  $\underline{M}$ =1,6"
240 INPUT "Introduceti criterial optimal(4sp)1)volum
   minim-minimorum(9sp)2)ponderat(22sp)
   3)maxidurabilitate", criteriu: PRINT AT 10,3;"5)
   Criterial optimal:"
250 IF criteriu=1 THEN PRINT AT 11,5;"volum mninim-
   minimorum"
260 IF criteriu=2 THEN PRINT AT 11,5;"ponderat"
270 IF criteriu=3 THEN PRINT AT 11,5;
```

```

"maxidurabilitate"
280 INPUT "Introduceti tipul transmisiei(31sp)
      1)reductor(22sp) 2)transmisie deschisa (11sp)
      3)roti baladoare",tip
290 IF tip=1 THEN PRINT AT 12,3;"6)Tipul
      transmisiei"; AT 13,5;"reductor de turatie"
300 IF tip=2 THEN PRINT AT 12,3;"6) Tipul
      transmisiei"; AT 13,5; "Transmisie deschisa"
310 IF tip=3 THEN PRINT AT 12,3;"6)Tipul
      transmisiei"; AT 13,5;"cu roti baladoare"
315 INPUT "Introduceti raportul de transmi-tere i",
      rat: PRINT AT 14,3;"7)Rap.de transmitere i=";
      rat
320 LET i$="0917160346": GOSUB box: PRINT AT 18,10;
      "CORECT (d/n) ?": PAUSE 0
330 GO TO 5000*(INKEY$ ="d")+150*(INKEY$="n")
500 GOSUB cortina: GOSUB ch2: PRINT AT 3,12; INVERSE
      1;"CONTINUT";INVERSE 0
510 PRINT AT 6,3; PAPER 7; INK 0;"1)ANGRENAJ
      CILINDRIC"; AT 7,5;"CU DINTI DREPTI"
520 PRINT AT 9,3; PAPER 7; INK 0;"2)ANGRENAJ
      CILINDRIC"; AT 10,5;"CU DINTI INCLINATI"
530 LET a$="Tastati 1 sau 2 !": LET i$="0314220399":
      GOSUB box: LET lin=15: LET col=4: GOSUB scrie:
      INPUT c$
540 IF c$="1" THEN LET fel=1
545 IF c$="2" THEN LET fel=2
800 REM *** REZOLVAREA PROBLEMEI ***
805 GOSUB chl
810 PRINT AT 2,2; INVERSE 1;"***REZOLVAREA
      PROBLEMEI***"
820 PRINT AT 5,2;"1 ": CIRCLE 19,131,7: PRINT "NR.DE
      DINTI PT.PINION"
825 IF criteriu=1 THEN GO TO 830
826 IF criteriu=2 THEN GO TO 840
827 IF criteriu=3 THEN GO TO 850
830 IF criteriu=1 THEN LET f1=9*(rat+1)/rat: LET
      f2=2/(0,06*(rat+1))
831 IF f1 >= f2 THEN LET z1=f1: GO TO 835

```

```

832 IF f2 >= f1 THEN LET z1=f2: GO TO 835
835 IF z1 <> INT z1 AND z1< INT z1+.5 THEN LET
    z1=INT z1
836 IF z1 <> INT z1 AND z1> INT z1+.5 THEN LET
    z1=INT z1+.1
837 LET a$="K=(max)[9(1+i)/i ;2/(0,06(1+i)]]": LET
    i$="0007320346";GOSUB box: LET lin=8: LET col=1:
    GOSUB scrie: PRINT AT 11,12; PAPER 7; INK 1;
    "K="; FN r(z1): LET z1=FN r(z1): LET x1=
    1-0.06*z1: GO TO 860
840 IF criteriu=2 THEN LET f1=13: LET f2=
    2/(0,03*(rat+1))
841 IF f2 >= f1 THEN LET z1=f2: GO TO 845
842 IF f1 >= f2 THEN LET z1=f1: GO TO 845
845 IF z1 <> INT z1 AND z1< INT z1+.5 THEN LET
    z1=INT z1
846 IF z1 <> INT z1 AND z1> INT z1+.5 THEN LET
    z1=INT z1+.1
847 LET a$="K=(max)[13;2/(0,03(1+i)]]": LET
    i$="0007320346": GOSUB box: LET lin=8: LET
    col=1: GOSUB scrie: PRINT AT 11,12; PAPER 7; INK
    1;"K="; FN r(z1): LET z1=FN r(z1): LET
    x1=1-0.03*z1: GO TO 860
850 IF criteriu=3 THEN LET f1=25: LET f2=66/rat
851 IF f1 >= f2 THEN LET z1=f1: GO TO 855
852 IF f2 >= f1 THEN LET z1=f2: GO TO 855
855 IF z1 <> INT z1 AND z1< INT z1+.5 THEN LET
    z1=INT z1
856 IF z1 <> INT z1 AND z1> INT z1+.5 THEN LET
    z1=INT z1+.1
857 LET a$="K=(max)[25 ;66/i]]": LET i$="0307270346":
    GOSUB box: LET lin=8: LET col=4: GOSUB scrie:
    PRINT AT 11,12; PAPER 7; INK 1;"K=";FN r(z1):
    LET z1=FN r(z1): LET x1=1
860 LET delta1=2*rat*z1/(1+rat): PRINT AT 13,2;
    PAPER 8;2" ": CIRCLE 19,68,7: PRINT AT 13,4; INK
    9; "CARACTERISTICA MATERIALULUI"
870 LET i$="0215150359": LET a$="BA=2iK/(1+i)":
    GOSUB box: LET lin=16: LET col=3: GOSUB scrie:

```

```

PRINT AT 16,18; PAPER 7; INK 1;"BA=";delta1
871 RESTORE 872: FOR q=0 TO 7: READ y: POKE USR
    "N"+q,y: NEXT q
872 DATA 96,16,32,120,0,0,0,0
880 GOSUB apasa: GOSUB cortina: GOSUB chl
882 PRINT AT 1,6; INVERSE 1;"MATERIALE DISPONIBILE"
883 PRINT AT 3,1; OVER 0;"Se va afisa lista
materialelor"; AT 5,1;"-cind BA=BJ
(echiportanta)"; AT 7,3;"se va alege acest
material "; AT 9,1;"-cind BJ>BA se alege
materia-"; AT 11,3;"lul cu (max)pH"; AT 13,1;"-
cind BJ<BA se alege materia-"; AT 15,3;"lul cu
(max)DH"
884 OVER 1: PRINT AT 18,1;"NOTATI NR.DE ORDINE AL
MATE-";AT 19,1;"RIALULUI si apasati tasta `s`":
886 GOSUB apasa: GOSUB cortina: BORDER 2: PAPER 0:
INK 7: CLS : GOSUB 8000
887 PRINT AT 21,4; INVERSE 1;"APASATI O TASTA
OARECARE"
888 FOR e=1 TO 38
895 OVER 0: PRINT AT 3,3;" nr,ordine=";n(e); AT
3,18;"BJ=";d(e); AT 5,1;o$(e); AT 7,1;"DH=";
s(e);" daN/cmN"; AT 9,1;"pH=";p(e);" daN/cmN";
AT 11,1;"IA=";r(e); AT 13,1;"IJ=";u(e)
896 PLOT 18,141: DRAW 14*8+2,0: DRAW 0,12: DRAW
(-14*8-2),0: DRAW 0,-12
897 PAUSE 0: IF INKEY$ ="s" OR INKEY$ ="S" THEN GO
TO 899
898 NEXT e
899 GOSUB cortina
900 CLS : INPUT "Introduceti nr.de ordine(2sp)pt.
materialul ales",,e
903 LET delta2=d(e): LET rov=r(e): LET rod=u(e):
LET sigma =s(e)
905 PRINT AT 12,1;o$(e); AT 14,0;"clasa=";clasa; AT
16,0;"BJ=";delta 2; AT 16,15;"DH=";s(e);
" daN/cmN"; AT 18,0;"IA=";r(e); AT 18,15;"IJ=";
u(e);AT 20,9;"CORECT (d/n)?" : PAUSE 0
906 IF INKEY$ ="n" OR INKEY$ ="N" THEN GO TO 899

```

```

907 IF delta1 >= delta2 THEN LET delta=delta1
910 IF delta2 >= delta1 THEN LET delta=delta2
911 IF fel=1 THEN const=70
912 IF fel=2 THEN const=60
920 LET niu=const*put*N1*N1*delta/(sigma*rat)
930 GOSUB chl: PRINT AT 2,2;"3 ": CIRCLE 19,155,7:
PRINT AT 2,4;"FUNCTIA DE VITEZA": LET a$=
"E=70cosFGP(NA)NB/iDH": LET i$="0204240361":
GOSUB box: LET lin= 5: LET col=3: GOSUB scrie
940 PRINT AT 8,7; PAPER 7; INK 0;"E=";niu
942 IF clasa=5 THEN GO TO 950
943 IF clasa=6 THEN GO TO 960
944 IF clasa=7 THEN GO TO 970
945 IF clasa=8 THEN GO TO 980
950 IF clasa=5 THEN LET kd=(400+1.5*niu)/(600+niu):
PRINT AT 10,2; PAPER 7;"4 ": CIRCLE 19,92,7:
PRINT AT 10,4;"COEF.DINAMIC kd": LET i$=
"0212260361": GOSUB box: LET a$=
"kd=(400+1,5E)/(600+E)": LET lin=13: LET col=3:
GOSUB scrie: PRINT AT 16,7; PAPER 7; INK 0;
"kd=";kd
960 IF clasa=6 THEN LET kd=(150+1.5*niu)/(300+niu):
PRINT AT 10,2; PAPER 7;"4 ": CIRCLE 19,92,7:
PRINT AT 10,4;" COEF.DINAMIC kd": LET i$=
"0212260361": GOSUB box: LET a$=
"kd=(150+1,5E)/(300+E)": LET lin=13: LET col=3:
GOSUB scrie: PRINT AT 16,7; PAPER 7; INK 0;
"kd=";kd
970 IF clasa=7 THEN LET kd=(30+1.5*niu)/(100+niu):
PRINT AT 10,2; PAPER 7;"4": CIRCLE 19,92,7:
PRINT AT 10,4;"COEF.DINAMIC kd": LET i$=
"0212260361": GOSUB box:LET a$=
"kd=(300+1,5E)/(100+E)": LET lin=13: LET col=3:
GOSUB scrie: PRINT AT 16,7; PAPER 7; INK 0;
"kd=";kd
980 IF clasa=8 THEN LET kd=(200+1.5*niu)/(200+niu):
PRINT AT 10,2; PAPER 7;"4 ": CIRCLE 19,92,7:
PRINT AT 10,4; "COEF.DINAMIC kd": LET i$=
"0212260361": GOSUB box: LET a$=

```

```

"kd=(200+1,5E)/(200+E)": LET lin=13: LET col=3:
GOSUB scrie: PRINT AT 16,7; PAPER 7; INK 0;
"kd=";kd
990 GOSUB apasa: GOSUB cortina
995 INPUT "Introduceti grupa materialului(2sp)(A
sau B)",g$: GOSUB chl
996 IF g$="B" OR g$="b" THEN LET g$="B"
997 IF g$="A" OR g$="a" THEN LET g$="A"
1000 PRINT AT 2,2;"5 ": CIRCLE 19,155,7: PRINT AT 2,5
;"COEF.DE REPARTIZARE"
1010 LET i$="0004320337": GOSUB box: LET a$=
"KQ=[1,1 (gr.A) ;1,25 (gr.B)]: LET lin=5: LET
col=1: GOSUB scrie
1020 IF g$="A" THEN PRINT AT 8,8; PAPER 7; INK 0;
"KQ=1,1": LET kr=1.1
1030 IF g$="B" THEN PRINT AT 8,8; PAPER 7; INK 0;
"KQ=1,25": LET kr=1.25
1035 PAPER 7: INK 0: PRINT AT 10,2;"6 ": CIRCLE
19,92,7: PRINT AT 10,5;"COEF. DE MAJORARE"
1040 LET kapa=kd*kr: LET i$="0212100346": GOSUB box:
LET a$="k=kQkd": LET lin =13: LET col=3:GOSUB
scrie: PRINT AT 16,8; PAPER 7; INK 0;"k=";kapa
1050 GOSUB apasa : GOSUB cortina
1060 GOSUB chl
1070 PRINT AT 2,2;"7 ": CIRCLE 19,155,7: PRINT AT 2,5
;"COEFICIENTUL AXIAL"
1071 RESTORE 1072: FOR q=0 TO 7: READ a: POKE USR
"C"+ q,a: NEXT q
1072 DATA 32,168,112,32,47,41,47,9
1080 LET i$="010400817": GOSUB box: LET a$=
"C=(1500+0,15E)/(1800+E)": LET lin=5: LET col=1:
GOSUB scrie: LET a$="(reductoare)": LET lin=6:
LET col=1: GOSUB scrie: LET a$="C=0,015": LET
lin=8: LET col=1: GOSUB scrie: LET a$=
"(tr.deschise, roti baladoare)": LET lin=9: LET
col=1: GOSUB scrie
1090 IF tip=2 AND tip=3 THEN PRINT AT 15,8; PAPER 7;
INK 0;"C=0,015": LET psia=0.015
1100 IF tip=1 THEN LET

```



```

      psia=(1500+0,15*niu)/(1800+niu)
      : PRINT AT 15,8; PAPER 7; INK 0;"C=";psia
1110 GOSUB apasa: GOSUB cortina
1120 GOSUB chl: PAPER 7: INK 0: PRINT AT 2,2;"8 ":
      CIRCLE 19,155,7: PRINT AT 2,5;"COEFICIENTUL
      DIAMETRAL"
1125 LET i$="0004320351": GOSUB box: LET a$="
      "(min/[0,5(1+i)C;0,6/1,2/1/6]": LET lin=5: LET
      col=0: GOSUB scribe
1126 RESTORE 1127: FOR q=0 TO 7: READ a: POKE USR
      "C"+q, a: NEXT q
1127 DATA 32,168,112,32,47,37,37,15
1128 PRINT AT 5,1 ;"C"
1131 IF b=1 THEN GO TO 1140
1132 IF b=2 THEN GO TO 1150
1133 IF b=3 THEN GO TO 1160
1140 IF b=1 THEN LET psip=0.6: LET doi=
      0.5*(rat+1)*psia
1141 IF psip <= doi THEN LET psid=psip: GO TO 1162
1142 IF doi <= psip THEN LET psid=doi: GO TO 1162
1150 IF b=2 THEN LET psip=1.2: LET doi=
      0,5*(rat+1)*psia: IF psip <= doi THEN LET psid=
      psip: GO TO 1162
1152 IF doi <= psip THEN LET psid=doi: GO TO 1162
1160 IF b=3 THEN LET psip=1.6: LET doi=
      0.5*(rat+1)*psia: IF psip <= doi THEN LET psid=
      psip: GO TO 1162
1161 IF doi <= psip THEN LET psid=doi
1162 PRINT AT 8,8; PAPER 7; INK 0;"C=";psid
1163 IF fel=1 THEN LET beta=0
1164 IF fel=2 THEN LET nec=3/(psid*z1): LET nec1=
      ASN(nec): LET beta=nec*90/PI
1165 PAPER 7: INK 0: PRINT AT 10,2;"9 ": CIRCLE
      19,91,7: PRINT AT 10,5;"COMPATIBILITATE
      MATERIAL"
1170 LET i$="0212290361": GOSUB box:LET a$=
      "I=(3sp)P(NA)NK(1+i)B/Ci <= IA,IJ": LET lin=13:
      LET col=2: GOSUB scribe
1180 IF fel=1 THEN LET c=1

```

```

1181 IF fel=2 THEN LET c=0.9: PRINT AT 13,5;"0.9"
1185 LET ro=c*put*N1*N1*kapa*(1+rat)*delta/
      (psid*rat): PRINT AT 16,8; PAPER 7; INK 0;
      "I=";ro
1190 IF ro <= rov AND ro<rod PRINT AT 20,7; PAPER 7;
      INK 0; FLASH 1;"MATERIAL COMPATIBIL"; FLASH 0:
      GOSUB apasa: GO TO 1210
1200 IF rov>rod THEN PRINT AT 19,6; PAPER 7; INK 0;
      FLASH 1;"MATERIAL INCOMPATIBIL !": FLASH 0;
      AT 20,6;"RELUATI CU ALT MATERIAL": BEEP .02,40:
      PAUSE 4: BEEP .02,-40: GOSUB apasa: GOSUB udg:
      GO TO 899
1210 GOSUB chl: PAPER 7: INK 0: PRINT AT 2,2;"10 ":
      CIRCLE 24,155,10 PRINT AT 2,5:"MOMENTUL DE
      RASUCIRE"
1220 LET i$="0604180329": GOSUB box: LET a$=
      "MO=70P/NAcosFG": LET lin=5: LET col= 7: GOSUB
      scrie
1230 LET Mt=70*put/(N1*(COS(beta*PI/180))3): PRINT
      AT 8,8; PAPER 7; INK 0;"MD=";Mt;" daNcm"
1232 PAPER 7: INK 0: PRINT AT 10,2;"11 ": CIRCLE
      24,91,10: PRINT AT 10,5;"MODULUL CALCULAT SI
      STAS"
1234 LET i$="0112300529": GOSUB box: PRINT AT 14,2;
      "mR[cm]="; AT 13,10;"1,31(2sp)3(2sp)Mok(1+i)E":
      PLOT 75,59: DRAW 40,0: PLOT 147,59: DRAW 88,0:
      PRINT AT 15,10;"k(8sp)CiDH": PLOT 120,64: DRAW
      10,-20: DRAW 15,30: DRAW 92,0
1250 LET mc=(1.31/z1)*(((Mt*kapa*(1+rat)*delta)/
      (psid*rat*sigma))1/3)
1260 PRINT AT 18,8; PAPER 7; INK 0;"mR[cm]=";mc
1259 REM cautarea modulului standardizat
1271 DIM m(41)
1272 RESTORE 1273: FOR j=1 TO 41: READ a: LET m(j)=a:
      NEXT j
1273 DATA 1,1.125,1.25,1.375,1.5,1.75,2,2.25,2.5,
      2.75,3,3.5,4,4.5,5.5,6,7,8,9,10,11,12,14,16, 18,
      20,22,25,28,32,36,40,45,50,55,60,70,80,90,100

```

```

1275 LET mc=10*mc: LET j=1
1280 IF m(j)>mc THEN GO TO 1310
1290 IF j <> 41 THEN LET j=j+1: GO TO 1280
1300 LET a$="NU exista modul standardizat": PRINT AT
    20,2;a$
1310 PRINT AT 19,10; PAPER 7; INK 0;"m[mm]=";m(j):
    LET mstas=m(j)
1311 IF mc <> mstas THEN LET psid=(1.32↑3)*Mt*kapa*
    (1+rat)*delta/((0.1*mstas*z1)↑3*rat*sigma)
1312 PRINT AT 20,2; PAPER 6; INK 2;"Se recalculeaza
    C=";psid: IF fel=1 THEN GO TO 1320
1313 PAUSE 200: IF fel=2 THEN PRINT OVER 0; AT 20,1;
    PAPER 7; INK 0;"(29sp)"; AT 20,1;"Se
    recalculeaza unghiul G=": LET nec=3/(psid*z1);
    LET nec1=ASN(nec1): LET beta=nec1*90/PI: PRINT
    AT 20,27; PAPER 7; INK 0; FN r(beta);" gr": LET
    beta=FN r(beta): LET vax=TAN (beta*PI/180)
1320 GOSUB apasa: GOSUB cortina
1330 PLOT 0,16: DRAW 255,0: DRAW 0,158: DRAW -255,0:
    DRAW 0,-158: PRINT AT 2,2;"12 ": CIRCLE
    24,155,10: PRINT AT 2,5;"ANTRAXUL DE REFERINTA"
1331 IF fel=1 THEN GO TO 1340
1332 IF fel=2 THEN LET bau=beta*PI/180: LET bubu=
    COS bau: LET z1=z1*bubu*bubu*bubu
1333 IF z1 <> INT z1 THEN LET z1= INT z1+1
1334 LET z2=rat*z1
1335 IF z2 <> INT z2 THEN LET z2= INT z2+1: LET rat=
    z2/z1
1340 LET i$="0204240319": GOSUB box: LET a$=
    "aP[mm]=mK(1+i)/2cosG": LET lin=5: LET col=3:
    GOSUB scrie
1341 IF fel=1 THEN LET a0=mstas*z1*(1+rat)/2: GO TO
    1360
1350 IF fel=2 THEN LET a0=mstas*z1*(1+rat)/(2*bubu)
1360 PRINT AT 8,4;"aP=";a0:" mm"
1370 IF a0=INT (a0) THEN PRINT AT 10,1;"Ungh.de
    angrenare : H=HP=20 gr"; AT 12,2 ; OVER 1;"angr.
    zero deplasat simetric"

```

```

1380 PRINT AT 18,2; PAPER 7; INK 0;"ANTRAXUL SE
ALINIAZA (d/n)?" : INPUT LINE e$
1381 IF e$="d" OR e$="D" THEN GO TO 2000
1382 IF e$="n" OR e$="N" AND fel=1 THEN LET x2=-x1:
GO TO 3500
1999 REM alinierea antraxului la angrenajele cu dinti
drepti
2000 INPUT FLASH 0;"(2sp)Introduceti antraxul
aQ[mm]",ar
2001 IF fel=2 THEN GO TO 2500
2002 LET alfar=ACS(0.94*a0/ar): LET x0=((TAN alfar-
alfar-TAN(20*PI/180)-(20*PI/180))/
(TAN(20*PI/180)))
2010 GOSUB chl: PRINT AT 1,1;"NOILE VALORI DE CALCUL
IN URMA": AT 2,6;"ALINIERII ANTRAXULUI"
2020 LET zlp=2*ar/(mstas*(1+rat)): IF zlp <> INT zlp
THEN LET zlp= INT zlp+1
2021 PRINT AT 4,1;"-nr.dintilor pinionului"; AT 6,8;
"zA=2aQ/m(1+i)=";zlp: LET z1=zlp
2030 LET z2p=rat*zlp
2031 IF z2p <> INT z2p THEN LET z2p= INT z2p+1
2032 PRINT AT 8,1;"-nr.dintilor rotii conduse"; AT
10,8;"zJ=izA=":z2p: LET z2=z2p
2033 LET ip=z2/z1: PRINT AT 12,1;"-raportul de
transmitere"; AT 14,8;"i=";ip
2034 IF ip <> i THEN psid=((1.31)↑3*Mt*kapa*(1+rat)*
delta/((.1*mstas*zlp)↑3*rat*sigma): PRINT AT
16,2;"-coeficientul diametral"; AT 18,8;
"C=";psid:IF fel=2 THEN LET nec=3/(psid*zlp):
LET beta=nec*90/PI
2035 GOSUB apasa: GOSUB chl
2036 LET alfar=ATN(0,36397023): LET alfaq=
(alfar*180/PI)
2045 PRINT AT 1,1;"-unghiul de angrenare"
2046 IF fel=1 THEN PRINT AT 3,8;"H=";alfag
2046 IF alfaq>24 THEN PRINT AT 11,4;"UNGHI DE
ANGRENARE > 24 gr"; AT 13,7; OVER 1; "SE REIA
CALCULUL CU"; AT 15,11;"zA=";z1+1;" dinti":

```

```

FOR w=1 TO 7: BORDER w: BEEP .002,w: NEXT w: LET
z1=z1+1: GOSUB apasa: GOSUB cortina: GOSUB udg:
GOSUB chl: GO TO 860
2050 LET x0=(TAN alfar-alfar-(TAN (20*PI/180)-
(20*PI/180)))/(TAN(20* PI/180))
2051 IF fel=1 THEN PRINT AT 5,1;"-functia de
calcul "; AT 8,1;"xP=";AT 7,4;" tgH-H-(tgHP-
HP)"; AT 9,9 ;"tgHP" ; AT 8,19;"=": PLOT 32,107:
DRAW 115,0: PRINT AT 11,10;x0
2053 IF criteriu=1 THEN LET x1=1-0.06*z1
2054 IF criteriu=2 THEN LET x1=1-0.03*z1
2055 IF criteriu=3 THEN LET x1=1
2057 LET xs=z1*(1+rat)*x0/2: LET x2=xs-x1: PRINT AT
13,1;"-depl.spec. suma x=":xs; AT 15,1;"-depl.
spec.pinion "; AT 16,8;"xA=";X1;AT 18,1;"-depl.
spec.roata condusa": AT 19,8;"xJ=";x2
2060 GOSUB apasa: GOSUB cortina: GO TO 3000
2500 REM alinierea antraxulului la angrenajele cu
dinti inclinati
2510 GOSUB chl: PRINT AT 1,1;"NOILE VALORI DE CALCUL
IN URMA": AT 2,6 ;"ALINIERII ANTRAXULUI"
2520 LET zlp=2*ar*bubu/(mstas*(1+rat)): IF zlp <> INT
zlp THEN LET zlp= INT zlp+1
2530 LET z2p=rat*zlp
2531 IF z2p <> INT z2p THEN LET z2p=INT z2p+1
2532 PRINT AT 8,1;"-ar.dintilor rotii conduse "; AT
10,8;"zJ=izA=";z2p: LET z2=z2p
2533 LET ip=z2/z1: PRINT AT 12,2;"-raportul de
transmitere"; AT 14,8;"i=";ip
2534 IF ip <> i THEN LET psid=((1.31)↑3*(Mt*kapa
*(1+rat)*delta/((.1*mstas*z1*bubu)↑3*rat*sigma):
PRINT AT 16,1;"-coeficientul diametral"; AT
18,8;"C=";psid: IF fel=2 THEN LET nec=
3/(psid*zlp): LET beta=nec*90/PI
2535 GOSUB apasa: GOSUB chl
2536 LET alfa0fr=(ATN(0.36397023/bubu)): LET alfa0fg=
(alfa0fr*180/PI)
2537 LET alfafr=ACS(a0*COS alfa0fr/ar):LET alfafg=

```

```

(alfafr*180/PI)
2538 LET alfanr=ATN(TAN alfafr*bubu):LET alfang=
alfanr*180/PI
2545 PRINT AT 1,1;"-unghiul de angrenare"
2547 IF fel=2 THEN PRINT AT 3,8;"HG=";alfang
2548 IF alfang>24 THEN PRINT AT 11,4;"UNGI DE
ANGRENARE >24 gr"; AT 13,7; OVER 1;"SE REIA
CALCULUL CU"; AT 15,11;"zA=";z1+1;" dinti":
FOR w=1 TO 7: BORDER w: BEEP .002,w: NEXT w:
LET z1=z1+1: GOSUB apasa: GOSUB cortina: GOSUB
udg: GOSUB ch1: GO TO 860
2550 LET x0=((TAN alfanr-alfanr-(TAN(20*PI/180)-
(20*PI/180)))/(TAN (20*PI/180)))
2552 PRINT AT 5,1;"-functia de calcul"; AT 8,1;"xP=";
AT 7,4;"tgHS-HS-(tgHPS-HPS)"; AT 9,9;" tgHPS";
AT 8,23;"="; PLOT 32,107: DRAW 148,0: PRINT AT
11,10 ;"=";x0
2553 IF criteriu=1 THEN LET x1=1-0.06**z1
2554 IF criteriu=2 THEN LET x1=1-0.03*z1
2555 IF criteriu=3 THEN LET x1=1
2557 LET xs=z1*(1+rat)*x0/(2*bubu13): LET x2=xs-x1:
PRINT AT 13,1;"-depl.spec.suma x=";xs; AT 15,1;
"-depl.spec.pinion"; AT 16,8;"xA=";x1; AT 18,1;
"-depl.spec.roata condusa"; AT 19,8;"xJ=";x2
2560 GOSUB apasa
2570 GO TO 3500
2999 REM geometria angrenajului cu dinti drepti
3000 GOSUB cortina: GOSUB ch1
3010 PRINT AT 1,2;"13 ": CIRCLE 24,163,9: PRINT AT
1,5;" GEOMETRIA ANGRENAJULUI"
3020 LET D1=mstas*z1: PRINT AT 2,6;"* Pinionul
(zA=(2sp)) *"; AT 2,21;z1; AT 3,2;"1)Diametrul
de divizare": LET i$="0105190319": GOSUB box: LET
a$="DA[mm]=mzA": LET lin=6: LET col=2: GOSUB
scrie: PAPER 7: INK 0: PRINT AT 8,2;"DA=";D1;
" mm="; INT ((10)*D1)/10;" mm"
3030 LET Dal=mstas*(z1+2+2*x1): PRINT AT 9,2;
"2)Diametrul de cap": LET i$="0111290319": GOSUB
box: LET a$="DHA[mm]=m(zA+2+2xA)": LET lin=12:

```

```

LET col=2: GOSUB scrie: PAPER 7: INK 0: PRINT AT
14,2;"DHA=";Da1;" mm =" ; INT ((10)* Da1)/10;
" mm"
3040 LET df1=mstas*(z1-2.5+2*x1): PRINT AT 16,2;
"3)Diametrul de picior": LET i$="0117300319":
GOSUB box: LET a$="DSA[mm]=m(zA-2,5+2xA)": LET
lin=18: LET col=2: GOSUB scrie: PRINT AT 20,2:
PAPER 7; INK 0;
"DSA=";Df1;" mm=" ; INT ((10)*Df1/10;" mm"
3050 GOSUB apasa: GOSUB cortina: GOSUB chl
3060 LET D2=mstas*z2: PRINT AT 1,1;"*Roata condusa
(zJ=((3sp) * "; AT 1,21;z2; AT 3,2;"1)Diametrul
de divizare": LET i$="0105190319": GOSUB box:
LET a$="DJ[mm]"=mzJ": LET lin=6: LET col=2:
GOSUB scrie: PAPER 7: INK 0: PRINT AT 8,2;"DJ
=";D2;" mm=" ; INT((10*D2)/10;" mm"
3070 LET Da2=mstas*(z2+2+2+x2): PRINT AT 9,2:
"2)Diametrul de cap ": LET i$="0111290319":
GOSUB box: LET a$="DHJ[mm]=m(zJ+2+2xJ)": LET
lin=12: LET col=2: GOSUB scrie: PAPER 7: INK 0:
PRINT AT 14,2;"DHJ=";Da2;" mm"; INT
((10)*Da2)/10;" mm"
3080 LET Df2=mstas(z2-2.5+2*x2): PRINT AT 16,2 ;
"3)Diametrul de picior": LET i$="0117300319":
GOSUB box: LET a$="DSJ[mm]=m(zJ-2,5+2xJ)": LET
lin=18: LET col=2: GOSUB scrie: PRINT AT 20,2;
PAPER 7; INK 0;"DSJ=";Df2;" mm"; INT
((10)*Df2/10;" mm"
3090 GOSUB apasa: GOSUB cortina
3100 GOSUB chl: PRINT AT 2,2;"LATIMEA ROTILOR": LET
i$="0204160349": GOSUB box: LET a$="B[mm]=CDA":
LET lin=5: LET col=3: GOSUB scrie: LET B=
psid*D1: PRINT AT 8,2; PAPER 7; INK 0;"B=";B;
" mm=" ; INT ((10)*B/10;" mm"
3101 IF e$ ="n" THEN LET ar=a0
3105 PRINT AT 4,21; PAPER 7; INK 0:"a=" ; INT
(10*ar)/(10);" mm"
3108 IF e$="n" THEN LET alfar=(20**PI/180)
3110 PAPER 7: INK 0: PRINT AT 12,2;"14 ":CIRCLE

```

```

24,76,10: PRINT AT 12,5:"FORTELE DIN ANGRENAJ";
AT 14,1:"-forta tangentiala": LET i$=
"0216280351": GOSUB box
3115 PAPER 7: INK 0: LET a$="FO=2MO=2MOcosH/DAcosHP":
LET lin=17: LET col=3: GOSUB scrie: LET Ft=2*Mt/
(.1*D1): PRINT AT 20,4;"FO="; FN r(Ft);" dan"
3130 GOSUB apasa: GOSUB cortina
3140 GOSUB chl: PRINT AT 2,1:"-forta normala
(radiala)": LET i$="0004320316": GOSUB box
3141 RESTORE 3142: FOR q=0 TO 7: READ a: POKE USR
"E"+q,a: NEXT q
3142 DATA 0,0,0,0,0,240,144,144
3145 IF fel=1 THEN LET a$="FE dan
=2MOsinHP/DA[cm]cosHP": LET lin=5: LET col=1:
GOSUB scrie: PAPER 7: INK 0: LET
Ft=2*Mt*SIN(20*PI/180)/
(.1*D1*0,94):PRINT AT 8,3;"FE="; FN r(Ft);" dan"
3150 GOSUB apasa: GOSUB cortina: GO TO 4000
3500 REM geometria angrenajului cu dinti inclinati
3510 PAPER 7: INK 0: CLS : GOSUB chl
3520 LET D1=mstas*z1/bubu: PRINT AT 2,6;"*Pinionul
(zA=(2sp)) *"; AT 2,21;z1; AT 3,2;"1)Diametrul
de divizare": LET i$=""0105190319": GOSUB box:
LET a$="DA[mm]=mzA/cosG": LET lin=6: LET col=2:
GOSUB scrie: PAPER 7: INK 0: PRINT AT 8,2;"DA=";
D1;" mm="; INT ((10)*d1)/10;" mm"
3530 LET dal=mstas*((z1/bubu)+2+2*x1): PRINT AT 9,2;
"2)Diametrul de cap": LET i$=""0111290319": GOSUB
box: LET a$="DHA[mm]=m((zA/cosG )+2+2xA)": LET
lin=12: LET col=2: GOSUB scrie: PAPER 7: INK 0:
PRINT AT 14,2;"DHA=";Dal;" mm="; INT ((10)*
Dal)/10;" mm"
3540 LET Dfl=mstas*((z1/bubu)-2.5+2*x1): PRINT AT
16,2 ;"3)Diametrul de picior": LET i$=
"0117310319": GOSUB box: LET a$="DAS[mm]=
m((zA/cosG)-2,5+2xA)": LET lin=18: LET col=2:
GOSUB scrie: PRINT AT 20,2; PAPER 7 ; INK 0;
"DSA=" ;Dfl;" mm="; INT ((10)*Dfl)/10;" mm"
3550 GOSUB apasa: GOSUB cortina: GOSUB chl

```



```

3560 LET D2=astas*x2/bubu: PRINT AT 1,1;" Roata
condusa (zJ=(3sp)) *"; AT 1,21;z2; AT 3,2
"1)Diametrul de divizare": LET i$="0105190319":
GOSUB box: LET a$="DJ[mm]=mzJ/cosG": LET lin=6:
LET col=2: GOSUB scrie: PAPER 7: INK 0: PRINT AT
8,2;"DJ=";D2;" mm="; INT ((10)*D2)/10;" mm"
3570 LET Da2=astas*(z2/bubu)+2+2*x2): PRINT AT 9,2;
"2)Diametrul de cap": LET i$="0111290319": GOSUB
box: LET a$="DHJ[mm]=m{(zJ/cosG)+2+2xJ}": LET
lin=12: LET col=2: GOSUB scrie: PAPER 7: INK 0:
PRINT AT 14,2 "DHJ=";Da2;" mm="; INT ((10)*Da2)
/10;" mm"
3580 LET Df2=astas*((z2/bubu)-2.5+2*x2): PRINT AT
16,2;"3)Diametrul de picior": LET i$=
"0117310319": GOSUB box: LET a$="DSJ[mm]=
m{(zJ/cosG)-2,5+2xJ}": LET lin=18: LET col=2:
GOSUB scrie: PRINT AT 20,2; PAPER 7; INK 0;
"DSJ=";Df2;" mm=": INT ((10)*Df2)/10;" mm"
3590 GOSUB apasa: GOSUB cortina
3600 GOSUB chl: PRINT AT 2,2;"LATIMEA ROTILOR": LET
i$="0204180349": GOSUB box: LET a$="B[mm]=CDA
cosG": LET lin=5: LET col=3: GOSUB scrie: LET
B=psid*D1*bubu: PRINT AT 8,2; PAPER 7; INK 0 ;
"B=";B;" mm=": INT ((10)*B)/10;" mm"
3601 IF e$="n" AND fel=2 THEN LET ar=a0
3610 PRINT AT 4,21; PAPER 7; INK 0;"a="; INT
((10)*ar)/10;" mm"; AT 6,1;"G="; FN
r(ATN vax*180/PI);" gr"
3900 REM fortele din angrenajul cu dinti inclinati
3910 PAPER 7: INK 0: PRINT AT 12,2;"14 ": CIRCLE
24,76,10: PRINT AT 12,5;"FORTELE DIN ANGRENAJ" ;
AT 14,1;"-forta tangentiala": LET i$=
"0216280351": GOSUB box
3920 PAPER 7: INK 0: LET a$="FOS=2MOCosHS/DACosHPS":
LET lin=17: LET col=3: GOSUB scrie: LET Ftf=
2*Nt*cosalfafr/(.1*D1*cos alfa0fr): PRINT AT
20,4;"FOS="; FN r(Ftf);" daN"
3930 GOSUB apasa: GOSUB cortina
3940 GOSUB chl: PRINT AT 2,1;"-forta

```

```

normala(radiala)": LET i$=0004320316": GOSUB box
3941 RESTORE 3942: FOR q=0 TO 7: READ a: POKE USR
    "E"+q,a: NEXT q
3942 DATA 0,0,0,0,0,240,144,144
3947 PAPER 7: INK 0: LET a$="FQ[daN]=FOstgHS": LET
    lin=5: LET col=2: GOSUB scrie: LET Fr=Ftf*TAN
    alfafr: PRINT AT 8,3;"FQ=";FN r(Fr);" daN"
3948 PRINT AT 10,1;"-forta axiala"
3952 LET i$="0112130319": GOSUB box: PAPER 7: INK 0:
    LET a$="FH=FOstgG": LET lin=13: LET col=2:
    GOSUB scrie: PAPER 7: INK 0: LET Fa=Ftf*vax:
    PRINT AT 16,3;"FH="; FN r(Fa);" daN"
3950 GOSUB apasa: GOSUB cortina
4000 GOSUB ch2: PAPER 6: INK 2: PRINT AT 11,2;
    "RELUATI CU ALTE DATE (d/n) ?": PAUSE 0
4010 IF INKEY$="d" OR INKEY$="D" THEN GOSUB udg:GO TO
    150
4020 PRINT AT 11,1; PAPER 7; INK 0;"(26sp)"; AT 11,4;
    "M.M.POPOVICI SOFTWARE 1992"; AT 13,4 "ROTI
    DINTATE CILINDRICE"; AT 15,1: OVER 0;"cu dinti
    drepti sau inclinati": PAUSE 0: STOP
7999 REM materialele pentru angrenaje
8000 DIM o$(38,40): FOR e=1 TO 38: READ o$(e): NEXT e
8002 DATA "39CA06","38MoCA09","18MoCN10","13CN30(1sp)
    18MC10 18MoCN13","13CN15 21TMC12","28TMC12",
    "OLC 55 OLC 60","OLC 10","OLC 45","41MoC11
    50VC11","40C10 OLC 15","OLC 20","20CO8": REM
    materiale grupa B
8003 DATA "Fc 40","OLC 55(NCR) OLC 60(NCR)","Fc 35",
    "OLC 45(N) OLC 45 (NCR)","OL 70(N)","Fc 30",
    Fc 25","50VC11","39CA06","OL 60(N)","33MoC11
    38MoCA09","OLC 35(N) OLC 35(NCR)","41MoC11",
    "34MoCN15","OLC 50(N)","40C10 41CN12 36MoCN10
    30MoCN20","OL 42(N)","OT60-3",Fgn 70-2",
    "OT 55-3","Fgn 60-2","OT 50-3","Fgn 50-7",
    "Fgn 45-5","OT 45-3": REM materiale grupa A
8004 RESTORE 8007
8005 DIM n(38): DIM d(38): DIM p(38): DIM r(38): DIM
    u(38)

```

```

8006 FOR e=1 TO 38: READ n(e),d(e),s(e),p(e),r(e),
    u(e): NEXT e
8007 DATA 1,12,1800,18000,80*104,22*106,2,13,1950,
    1000,80*104,22*106,3,15,2250,18000,200*104,
    32*106,4,16,2400,14800,200*104,32*106,5,17,
    2600,8000,200*104,32*106,6,18,2850,18000,
    200*104,32*106,7,19,1800,14000,200*104,
    32*106,8,20,1500,12000,200*104,32*106,9,21,
    1800,13000,200*104,32*106,10,22,1800,13000,
    200*104,32*106,11,23,1850,13000,200*104,
    32*106,12,24,1650,12000,200*104,32*106,13,35,
    2000,11000,200*104,32*106:REM materiale
    grupa B
8008 DATA 14,93,900,3800,0.4*104,8*106,15,94,1850,
    6400,80*104,22*106,16,101,780,3400,0.4*104,
    8*106,17,105,1680,5800,80*104,22*106,18,108,
    1500,5400,30*104,20*106,19,110,560,3000,
    0.4*104,8*106,20,115,560,2700,0.4*104,
    8*106,21,121,2000,5600,80*104,22*106,22,126,
    1950,2700,80*104,22*106,23,128,1350,4700,
    30*104,20*106
8009 DATA 24,136,1900,5400,80*104,22*106,25,137,
    1420,4650,30*104,20*106,26,114,2000,5400,
    80*104,22*106,27,145,2260,5700,80*104,
    22*106,28,150,1200,4100,30*104,20*106,29,151,
    2100,5400,80*104,22*106,30,153,1050,3800,
    8*104,12*106,31,163,1500,4400,0.6*104,
    9*106,32,166,1950,4200,0.8*104,107,33,
    184,1400,4000,0.6*104,9*106
8010 DATA 34,194,1680,3600,0.8*104,107,35,202,1250,
    3600,0.6*104,9*106,36,218,1400,3100,0.8*104,

```

```

10↑7,37,222,1250,2900,0.8*10↑4,10↑7,38,234,1440,
3200,0.6*10↑4,9*10↑6
8020 RETURN
8999 REM subroutine
9000 PAPER 9: INK 9: CLS : PLOT 0,0: DRAW 255,0: DRAW
-255,0: DRAW 0,-175
9001 LET sp=2: PLOT sp,sp: DRAW 255-2*sp,0: DRAW
0,175-2*sp: DRAW -255+2*sp,0: DRAW 0,-175+2*sp:
RETURN
9100 LET t$="(15sp)": PAPER 6: INK 0: BORDER 7: CLS :
PLOT 0,0: DRAW 255,0: DRAW 0,175: DRAW -255,0:
DRAW 0,-175
9101 PRINT AT 1,2; PAPER 5;t$;t$(2 TO): FOR i=2 TO
20: PRINT AT i,1; PAPER 7;t$;t$(2 TO);AT i-1,30:
PAPER 5;" ": NEXT i
9102 RETURN
9200 LET a$="(4sp)APASATI 0 TASTA OARECARE (4sp)"
9201 PRINT # 1; AT 1,0:a$
9202 LET a$=a$(32)+a$(TO 31)
9203 PAUSE 5: IF INKEY$="" THEN GO TO 9201
9204 RETURN
9300 LET xo=VAL i$(1 TO 2): LET yo=VAL i$(3 TO 4):
LET xl=VAL i$(5 TO 6): LET yl=VAL i$(7 TO 8)
9301 PAPER VAL i$(9): INK VAL i$(10)
9302 LET i=0: IF i THEN LET x=FN w(xo,yo,xl,yl)
9303 LET i$=x$(TO xl): FOR i=yo TO yo+yl-1: PRINT AT
i, xo;i$: NEXT i
9304 PLOT 8*xo+1,174-8*yo: DRAW 8*xl-3,0: DRAW
0,3-yl*8: DRAW 3-xl*8,0: DRAW 0,yl*8-3
9305 RETURN
9400 RESTORE 9401: LET a$="abcdefghijklmnopqr": FOR
i=1 TO LEN a$: FOR n=0 TO 7: READ k: POKE USR
a$(i)+n,k: NEXT n: NEXT k
9401 DATA 0,0,0,64,192,64,64,224,0,6,8,8,56,72,72,48
9402 DATA 32,168,112,32,47,37,37,15,0,0,62,64,64,68,
68,56
9403 DATA 0,0,4,36,20,24,16,0,112,16,48,16,112,0,0,0,
9404 DATA 0,28,34,60,34,34,76,0,0,0,0,56,72,72,60,0

```

```

9405 DATA 0,0,12,18,18,28,16,32,0,0,0,0,96,16,32,120
9406 DATA 0,240,16,32,66,246,2,7,1,242,20,32,246,2,7
9407 DATA 33,170,112,32,47,37,37,15,30,170,113,34,47,
37,37,15
9408 DATA 0,0,0,64,224,64,64,96,0,0,0,0,96,144,144,96
9409 DATA 0,0,0,0,224,128,128,128,0,0,0,0,96,128,128,
96
9410 DATA 0,0,0,96,64,224,64,64
9411 RETURN
9500 PRINT # 1; AT 1,0;"(32sp)": OVER 0: FOR f=0 21:
PRINT AT f,0; PAPER RND*7;"(32sp)": NEXT f:
PAUSE 10: BORDER 7: PAPER 7: INK 0 : CLS :
RETURN

```

Aplicații numerice

1) $P=96$ CP, $n_1=5000$ rot/min, clasa 6, pinion nesimetric, volum minim-minimorum, reductor cu dinți drepți, $i=3$.

Rezolvare : $z=12$ dinți ; $\delta_1=18$; materialul 28TMC12 (cu $\delta_2=18$; $\sigma_a=2850$ daN/cm², $\rho_1=200 \cdot 10^4$; $\rho_2=32 \cdot 10^6$); funcția de viteză $v=353,68421$ $k_d=1,0410628$, $k_r=1,25$, $k=1,3013215$, $\Psi_A=0,72111437$; $\Psi_D=1,2$; $\rho=62463,768$ (material compatibil); $M_t=1344$ daNcm, modulul calculat $m_c=0,25181436$ cm; modulul standardizat $m=2,75$ mm; $\Psi_D=0,92135004$ (recalculat); antraxul de referință $a_o=66$ mm (nealiniat); dimensiunile pinionului: $D_1=33$ mm, $D_{a1}=40$ mm; $D_{f1}=27,6$ mm; dimensiunile roții conduse ($z_2=36$): $D_2=99$ mm; $D_{a2}=102,9$ mm; $D_{f2}=90,5$ mm; lățimea roților $B=30,4$ mm; forțele din angrenaj $F_t=815$ daN, $F_n=297$ daN

2) $P=50$ CP, $n_1=3000$ rot/min, clasa 7, pinion simetric, criteriu optimal ponderat, reductor cu dinți drepți, $i=3$.

Rezolvare: $z_1=17$ dinți; $\delta_1=25,5$; materialul 28TMC12 (cu $\delta_2=18$, $\sigma_a=2850$ daN/cm², $\rho_1=200 \cdot 10^4$, $\rho_2=32 \cdot 10^6$); funcția de viteză $v=93,947368$, $k_d=0,88127544$, $k_r=1,25$; $k=1,1015943$, $\Psi_A=0,79943727$, $\Psi_D=1,598745$; $\rho=10541,411$ (material compatibil); $M_t=1166,6667$ daNcm; modulul calculat $m_c=0,16371372$ cm, modulul standardizat

$m = 1,75$ mm, $\Psi_D = 1,3090478$ (recalculat), antraxul de referință $a_o = 59,5$ mm care se alinează la valoarea $a_r = 60$ mm cînd $z_1 = 18$ dinți, $z_2 = 54$ dinți, $i = 3$, $\Psi_D = 1,1027695$, unghiul de angrenare $\alpha = 21,224882^\circ$; $x_o(\alpha) = 0,0083129676$, $x = 0,299266$; $x_1 = 0,46$; $x_2 = -0,16073317$; dimensiunile pinionului: $D_1 = 31,5$ mm, $D_{a1} = 36,6$ mm, $D_{f1} = 28,7$ mm; dimensiunile roții conduse: $D_2 = 94,5$ mm, $D_{a2} = 97,4$ mm, $D_{f2} = 89,5$ mm, lățimea roților $B = 34,7$ mm; forțele din angrenaj: $F_t = 741$ daN, $F_n = 270$ daN.

3) $P = 20$ CP, $n_1 = 3000$ rot/min, clasa 8, pinion în consolă, criteriul optimal ponderat, reductor cu dinți înclinați, $i = 3$.

Rezolvare : $z_1 = 17$ dinți, $\delta_1 = 25,5$; materialul 28TMC12 (cu $\delta_2 = 18$, $\sigma_a = 2850$ daN/cm², $\rho_1 = 200 \cdot 10^4$, $\rho_2 = 32 \cdot 10^6$); funcția de viteză $v = 32,210526$, $k_d = 1,0693563$, $k_r = 1,25$, $k = 1,3366954$, $\Psi_A = 0,82132023$, $\Psi_D = 0,6$; $\rho = 12270,864$ (material compatibil); $M_t = 482,1099$ daNcm, modulul calculat $m_c = 0,18031956$ cm, modulul standardizat $m = 2$ mm, $\Psi_D = 0,43973372$ (recalculat), $\beta = 9^\circ$, antraxul de referință $a_o = 68,8476$ 29 mm care se aliniaza la valoarea $a_r = 70$ mm cînd $z_1 = 18$ dinți, $z_2 = 54$ dinți, $i = 3$, $\Psi_D = 0,38446715$, $\alpha_f = 22,39952^\circ$, $x(\alpha_f) = 0,017338497$. $x = 0,64781971$, $x_1 = 0,46$, $x_2 = 0,18781971$; dimensiunile pinionului: $D_1 = 36,4$ mm, $D_{a1} = 42,2$ mm, $D_{f1} = 33,2$ mm; dimensiunile roții conduse: $D_2 = 109,3$ mm, $D_{a2} = 114$ mm, $D_{f2} = 104$ mm; lățimea roților $B = 13,8$ mm; unghiul de înclinare a dinților $\beta = 9^\circ$; forțele din angrenaj: $F_H = 261$ daN, $F_r = 109$ daN, $F_a = 42$ daN.

14. APLICAȚII ÎN DOMENIUL MATEMATICII

Întrucît matematica are un domeniu practic nelimitat, au fost alese acele capitole care sînt mai puțin tratate în lucrările apărute pînă în prezent.

14.1. PUTERI, FACTORIALI ȘI COMBINĂRI

Exemplul 14.1 : calculul lui x^n .

Capacitatea calculatorului este limitată la valoarea 10^{38} , dar această valoare poate fi depășită dacă se folosesc logaritmi.

```

8 REM calculul lui x la puterea n
10 CLS : INPUT "Introduceti x",x
12 IF x=0 OR x<0 THEN GO TO 10
20 INPUT "Introduceti n",n
22 IF n=0 OR n<0 THEN GO TO 20
25 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
29 GO TO 30*(INKEY$ ="d")+10*(INKEY$ ="n")
30 LET y=n*LN(x/LN(10))
40 IF y<38 THEN GO TO 70
50 CLS : PRINT AT 11,4;x;"^";n;"=";10^(y-INT(y));
   "E+"; INT(y)
60 GO TO 80
70 CLS : PRINT AT 11,4;x;"^";n;"=";x^n
80 PRINT AT 1,9 : "RELUATI (d/n) ?": PAUSE 0
90 GO TO 10*(INKEY$ ="d")+100*(INKEY$ ="n")
100 CLS : STOP

```

Exemplul 14.2 : Calculul lui $n!$ (pt. $n \leq 32$), unde $n! = n \cdot (n-1) \cdot (n-2) \dots 3 \cdot 2 \cdot 1$

```

8 REM calculul lui n!
10 CLS : INPUT "Introduceti n <= 32",n
15 IF n=0 OR n<32 AND n<0 THEN GO TO 10
20 LET f=1
30 FOR i=1 TO n
40 LET f=f*i
50 NEXT i
60 PRINT AT 11,4;n;"!=";f
70 PRINT AT 1,9;"RELUATI (d/n) ?"; PAUSE 0
80 GO TO 10*(INKEY$="d")+90*(INKEY$="n")
90 CLS : STOP

```

Aplicație : pentru $10! = 3628800$.

Întrucît programul calculează $n!$ pentru $n \leq 32$, se poate depăși limitarea

știind că logaritmul zecimal al lui $n!$ este $y = \sum_{i=1}^n \log_{10} i$. Rezultă că

$$n! = 10^{y - \text{INT}(y)} \cdot 10^{\text{INT}(y)}$$

Exemplul 14.3:

```

10 CLS : INPUT "Introduceti n",n
15 IF n=0 OR n<0 THEN GO TO 10
20 LET s=0
30 FOR i=1 TO n
40 LET s=s+LN(i)
50 NEXT i
60 LET y=s/LN(10)
70 IF y>8 THEN GO TO 120
80 CLS : PRINT AT 11,4;n;"!="; INT (10↑y)
100 GO TO 130
120 CLS : PRINT AT 11,4;n;"!=";10↑(y-INT(y));
"E+"; INT(y)
130 PRINT AT 1,9;"RELUATI (d/n)?: PAUSE 0

```



```
140 GO TO 10*(INKEY$="d")+150*(INKEY$="n")
150 CLS : STOP
```

Aplicații : pentru $20! = 2.43290E+18$ și pentru $100! = 9.3326205E+157$

Exemplul 14.4 : combinări $C(k,n) = C(k-1,n)(n-k+1/k)$

Dacă y este logaritmul zecimal al lui $C(k,n)$, atunci

$$y = \sum_{i=1}^n \log_{10} [(n-i+1)/i] \text{ și } C(k,n) = 10^{y - \text{INT}(y)} \times 10^{\text{INT}(y)}$$

```
8 REM combinari C(k,n)=C(k-1,n)(n-k+1/k)
10 CLS ; INPUT "Introduceti k", k
15 IF k=0 OR k<0 THEN GO TO 10
20 INPUT "Introduceti n", n
25 IF n=0 OR n<0 THEN GO TO 20
30 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
40 GO TO 50*(INKEY$="d")+10*(INKEY$="n")
50 LET s=0
60 FOR i=1 TO k
70 LET s=s+LN((n-i+1)/i)
80 NEXT i
90 LET y=s/LN(10)
100 IF y>8 THEN GO TO 130
110 CLS : PRINT AT 11,4;"C(" ;k;" , " ;n;" )=" ; INT #
    (10↑y)
120 GO TO 140
130 CLS : PRINT AT 11,4;"C(" ;k;" , " ;n;" )=" ; 10↑
    (y-INT(y));"E+" ; INT (y)
140 PRINT AT 1,9;"RELUATI (d/n) ?": PAUSE 0
150 GO TO 10*(INKEY$="d")+160*(INKEY$="n")
160 CLS : STOP
```

Aplicații: $C(5,10) = 252$

$C(7,22) = 170544$

14.2. ȘIRURI

Exemplul 14.5 : șirul tip $u_n = f(n)$; fie pentru $n \geq 1$, $u_n = n \sin(1/n)$ și

se cere calculul primilor 1000 termeni ai șirului.

```

8 REM sirul u=f(n);calculul primilor 1000 termeni
10 CLS : FOR n=1 TO 1000
20 PRINT n,n*SIN (1/n)
30 NEXT n

```

Aplicații: $u_{22}=0,99965568$; $u_{44}=0,99991391,0$, $u_{154}=0,99999297$; șirul converge spre 1.

Exemplul 14.6: pentru $n \geq 1$ fie $u_n = 1 + (1/2) + (1/3) + \dots + (1/n) - \log n$; se cer primii 1000 termeni ai șirului.

```

10 CLS : FOR n=1 TO 1000
20 LET s=0
30 FOR i=1 TO n
40 LET s=s+1/i
50 NEXT i
60 PRINT n,s-LN(n)
70 NEXT n

```

Aplicații: $u_{22}=0,59977087$; $u_{40}=0,58966359$; $u_{500}=0,57821532$.

Programul poate fi modificat pentru a se obține numai termenul u_n ; se obține programul următor:

```

10 CLS : INPUT "Introduceti n",n
15 IF n=0 OR n<0 THEN GO TO 10
20 LET s=0
30 FOR i=1 TO n
40 LET s=s+1/i
50 NEXT i
60 PRINT n,s-LN(n)

```

Aplicații: $u_{30}=0,59378975$; $u_{3000}=0,57738231$; $u_{10000}=0,57771558$; $u_{20000}=0,57724049$; șirul converge spre constanta lui Euler: 0,577218

Exemplul 14.7: Fie $n \geq 1$ și $u_{n+1} = \sqrt{5 + u_n}$ și se cere u_n

```

10 LET u=0

```

```

20 LET n=2
30 LET u=SQR(5+u)
35 PRINT AT 0,0;"NR,.ordine "; AT 0,16;"Valoarea"
40 PRINT ``n,u
50 LET n=n+1
60 GO TO 30

```

Pentru $n \geq 13$ se obține $u_n = 2.7912879$ (limita șirului).

Exemplul 14.8 : șirul $u_{n+2} = f(u_{n+1}, u_n)$: fie $u_{n+2} = [(u_{n+1} + 4n)/2]$
cu $u_1 = 1$ și $u_2 = 2$.

```

10 CLS : INPUT "Introduceti u1",u1
15 IF u1=0 OR u1<0 THEN GO TO 10
20 INPUT "Introduceti u2",u2
25 IF u2=0 OR u2<0 THEN GO TO 20
30 LET n=3
40 LET u=(u1+u2)/2
45 PRINT AT 0,0:"Nr.ordine "AT 0,16;"Valoarea"
50 PRINT ``n,u
60 LET u1=u2
70 LET u2=u
80 LET n=n+1
90 GO TO 40

```

Pentru $u_1 = 1$ și $u_2 = 2$ se obține 1,6666667

14.3. SERII

Exemplul 14.9: fie seria $s = \sum_{i=1}^n 1/2^i$; să se calculeze suma

primelor 100 termeni.

```

10 CLS : LET s=0
20 FOR n=1 TO 100
30 LET s=s+1/(2^n)

```

```
40 PRINT n, s
50 NEXT n
```

Se obține $s=1$ de la $n=28$.

Exemplul 14.10: seria armonică
$$s = \sum_{k=1}^n 1/k$$

```
10 CLS : LET s=0
20 FOR n=1 TO 10000
30 LET s=s+1/n
40 PRINT n, s
50 PRINT n
```

Rezultă: $s_{137} = 4,7893524$; $s_{388} = 5,0602235$; $s_{1000} = 7,4854709$;
 $s_{10000} = 9,787606$ (serie divergentă dar foarte lentă).

Exemplul 14.11 : pentru seria $s = \sum_{k=1}^n 1/[k(k+1)]$ se cer s_{22} și

s_{1000}

```
10 CLS : INPUT "Introduceti n", n
15 IF n=0 OR n<0 THEN GO TO 10
20 LET s=0
30 FOR k=1 TO n
40 LET s=s+1/k/(k+1)
50 NEXT k
60 PRINT s
```

Se obțin valorile: $s_{22} = 0,95652174$; $s_{100} = 0,99009901$; $s_{10000} = 0,999001$
 (serie convergentă spre 1).

Exemplul 14.12: Seria Rieman are forma $s = \sum_{k=1}^n 1/k^2$; se cer

s_{100} ; s_{10000} ; s_{20000} ; s_{30000} și s_{50000} .

```

10 CLS : INPUT "Introduceti n", n
15 IF n=0 OR n<0 THEN GO TO 10
20 LET s=0
30 FOR k=1 TO n
40 LET s=s+1/k/k
50 NEXT k
60 PRINT s

```

Rezultă: $s_{100} = 1,6349830$; $s_{10000} = 1,6448341$; $s_{20000} = 1,6448841$;
 $s_{30000} = 1,6449008$; $s_{50000} = 1,6449138$ (seria converge lent către
 $1,6449341 = \pi^2/6$)

14.4. FUNCȚII NUMERICE

Fie f o funcție numerică de o variabilă sau mai multe variabile. Pentru a calcula valoarea funcției într-un punct se definește funcția ca un șir de caractere și apoi cu ajutorul funcției VAL se evaluează șirul de caractere în punctul considerat.

Exemplul 14.13 : funcție numerică de o variabilă.

```

8 REM functie numerica de o variabila
10 CLS : INPUT "Introduceti functia f(x)", f$
15 PRINT AT 21,9;"CORECT (d/n)?" : PAUSE 0
20 GO TO 25*(INKEY$="d")+10*(INKEY$="n")
25 CLS : INPUT "Introduceti x0", x0
27 PRINT AT 21,9;"CORECT (d/n) ? : PAUSE 0
30 GO TO 40*(INKEY$="d")+25*(INKEY$="n")
40 LET x=x0
50 CLS : PRINT AT 11,2;"Valoarea in punctul x0=";
  x0; AT 13,2;"a functiei f(x)=";f$; AT 15,3 "este
  f(";x0;) =" ; VAL f$
60 PRINT AT 21,9:RELUATI (d/n) ?" : PAUSE 0
70 GO TO 10*(INKEY$="d")+80*(INKEY$="n")
80 CLS : STOP

```

Pentru: $f(x) = \cos(x) - 1/x^2$ și $x_0 = 0.3$ rezultă : -10.155775

$f(x) = x \sin x + x^2$ și $x_0 = 0,42$ rezultă : 0,34759939

Exemplul 14.14: functia numerica de trei variabile

```

8 REM functia numerica de trei variabile
10 CLS : INPUT "Introduceti functia f(x,y,z)",f$
15 PRINT AT 21,9:"CORECT (d/n)?" : PAUSE 0
20 GO TO 25*(INKEY$="d")+10*(INKEY$="n")
25 CLS : INPUT "Introduceti x0,y0,z0";x0,y0,z0
30 PRINT AT 21,9;"CORECT (d/n) ?" : PAUSE 0
35 GO TO 40*(INKEY$="d")+25*(INKEY$="n")
40 LET x=x0: LET y=y0: LET z=z0
50 CLS : PRINT "f(";x0;",";";"y0";",";";z0;"="; VAL f$
50 PRINT AT 21,9;"RELUATI (d/n) ? " : PAUSE 0
70 GO TO 10*(INKEY$="d")+80*(INKEY$="n")
80 CLS : STOP

```

Pentru: $f(x,y,z) = (3/x) + (3/y) + (3/z)$ și $x_0 = 1, y_0 = 2, z_0 = 3$ rezultă 5.5

$f(x,y,z) = 2x^2 - y \sin z$ și $x_0 = 1, y_0 = 2, z_0 = 1$ rezultă
0,84147098

$f(x,y,z) = x^2 + \tan(1/x) - 2z$ și $x_0 = y_0 = z_0 = 1$ rezultă
0,55740772

14.5. DERIVATE, POLINOAME, LUNGIMEA ARCULUI DE CURBĂ

Fie e un număr real pozitiv și foarte mic; luând $e = 0,001$ se poate calcula valoarea într-un punct x_0 a derivatei de ordinul întâi cu relația

$$\frac{[f(x_0 + e) - f(x_0 - e)]}{2e}$$

Exemplul 14.15

```

8 REM valoarea derivatei intii in punctul x0
10 CLS : INPUT "Introduceti functia",f$
15 PRINT AT 21,9:"CORECT (d/n) ?" : PAUSE 0
20 GO TO 25*(INKEY$="d")+10*(INKEY$="n")

```

```

25 CLS : INPUT "Introduceti x0",x0
27 PRINT AT 21,9;" CORECT(d/n) ? ": PAUSE 0
30 GO TO 40*(INKEY$="d")+25*(INKEY$="n")
40 LET e=.001
50 LET x=x0+e
60 LET g=VAL f$
70 LET x=x0-e
80 LET h=VAL f$
90 CLS : PRINT "Derivata in x0=";x0; AT 2,0;
  " a functiei:";f$; AT 4,0;"este ":"(g-h)/2/e
100 PRINT AT 21,9;"RELUATI (d/n) ? ": PAUSE 0
110 GO TO 10*(INKEY$="d")+120*(INKEY$="n")
120 CLS : STOP

```

Pentru: $2x^2/(3x-5)$ și $x_0=2$	rezultă -16,000152
$\sin(1/x)$ și $x_0=3,14$	rezultă -0.096329
$x/(1+x^2)$ și $x_0=1,5$	rezultă -0,118344316

Exemplul 14.16

```

8 REM valoarea derivatei a 2-a intr-un punct x0
10 CLS : INPUT " Introduceti functia",f$
12 PRINT AT 1,2;"Functia ":";f$
15 PRINT AT 21,9;"CORECT(d/n)?" : PAUSE 0
20 GO TO 25*(INKEY$="d")+10*(INKEY$="n")
25 INPUT "Introduceti x0",x0
26 PRINT AT 3,2:"Valoarea x0=";x0
27 PRINT AT 21,9;"CORECT(d/n) ? ": PAUSE 0
30 GO TO 40*(INKEY$="d")+25*(INKEY4+'n")
40 LET e=.001
50 LET x=x0
60 LET f=VAL f$
70 LET x=x0+2*e
80 LET g=VAL f$
90 LET x=x0-2*e
100 LET h=VAL f$
110 CLS : PRINT AT 0,0;"Derivata a 2-a a functiei ";
  AT 1,3;f$; AT 3,0;"in punctul x0=";x0; AT 5,0;

```

```

"este :";(g+h-2*f/4*e*e
120 PRINT AT 21,9;"RELUATI (d/n)?: PAUSE 0
130 GO TO 10*(INKEY$="d")+140*(INKEY$="n")
140 CLS : STOP

```

Programul se bazează pe determinarea derivatei a doua într-un punct x_0 cu ajutorul relației

$$f(x_0+2e)+f(x_0-2e)-2f(x_0)/4e^2 \quad \text{unde } e=0,001$$

Exemplul 14.17 : polinoame

Fie p un polinom oarecare de o variabilă; de exemplu $p(x)=a_4x^4+a_3x^3+a_2x^2+a_1x+a_0$. Conform schemei lui Horner se poate scrie: $p(x)=(((a_4x+a_3)x+a_2)x+a_1)x+a_0$. Se notează: $c_4 = a_4$; $c_3 = xc_4 + a_3$; $c_2 = xc_3 + a_2$, etc. și pe această bază se obține un polinom $p(x) = c$.

Deoarece indicele 0 nu poate fi dimensionat se vor lua $1 \leq i \leq n+1$ și $b_i = a_i - 1$, rezultând programul următor.

```

8 REM polinoame
10 CLS : INPUT "Introduceti gradul polinomului n",n
20 DIM b(n+1)
30 FOR i=n+1 TO 1 STEP -1
50 INPUT "Introduceti coeficientii",b(i)
60 CLS
70 NEXT i
80 INPUT "Introduceti x",x
90 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
95 GO TO 100*(INKEY$="d")+10*(INKEY$="n")
100 CLS : LET c=b(n+1)
110 FOR i=n TO 1 STEP -1
120 LET c=c*x+b(i)
130 NEXT i
140 CLS : PRINT AT 1,2;"Valoarea polinomului
    pentru "; AT 3,10;"x=";x;" este"; AT 5,11;
    "p(";x;")=";c
150 PRINT AT 21,9;"RELUATI(d/n) ?": PAUSE 0
160 GO TO 10*(INKEY$=d")+170*(INKEY$="n")
170 CLS : STOP

```


Pentru: $p(x) = 3x^3 + 3x^2 + 3x + 3$ și $x = 3$ rezultă $p(3) = 120$

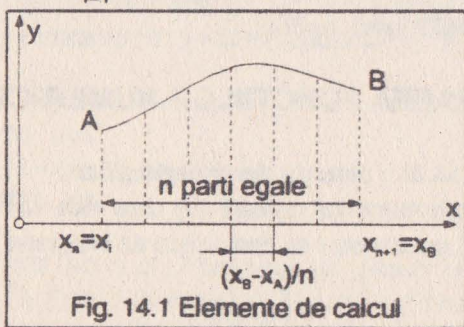
$p(x) = 2x^4 - x^3 + 5x^2 + 3x - 1$ și $x = 1,4$ se obține $p(1,4) = 17,9392$

$p(x) = x^8 - x - 1$ și $x = 2$ rezultă $p(2) = 253$.

Exemplul 14.18 : lungimea unui arc de curbă .

Se consideră un arc de curbă ale cărui puncte extreme A și B au abscisele x_A , x_B . Împărțind intervalul $[x_A; x_B]$ în n părți egale se obțin $n + 1$ puncte de diviziune: $x_1 = x_A; x_2; x_3; \dots x_{n+1} = x_B$ (fig. 14.1). Pentru $1 \leq i \leq n + 1$ o abscisă oarecare i va fi $x_i = x_A + (i-1)(x_B - x_A)/n$, iar ordonatele vor fi $y_1 = f(x_1); y_2 = f(x_2); \dots y_{n+1} = f(x_{n+1})$. Asimilînd lungimea l a arcului AB cu suma segmentelor A_i, A_{i+1} atunci este evidentă relația:

$$l = \sum_{i=1}^n \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}$$



```

8 REM lungimea unui arc de curba
10 CLS : INPUT "Introduceti functia f(x)", y$
20 INPUT "Introduceti abscisele punctelor extreme
   xA, xB", xa, xb
30 CLS : INPUT "Introduceti nr. partilor n", n
40 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
50 GO TO 60*(INKEY$="d")+10*(INKEY$="n")
60 CLS
100 LET d=(xb-xa) /n
110 LET l=0
120 FOR i=1 TO n

```

```

130 LET x1=xa+(i-1)*d
140 LET x2=xa+i*d
150 LET x=x1
160 LET y1=VAL y$
170 LET x=x2
180 LET y2=VAL y2
190 LET l=1+SQR(d*d+(y2-y1)*(y2-y1))
200 NEXT i
210 PRINT AT 1,3;"Lungimea arcului de curba";
    AT 3,10:y$; AT 5,3:"este ";l
220 PRINT AT 21,4;"ALT `n` (d/n)? ": PAUSE 0
230 GO TO 30*(INKEY$="d")+240*(INKEY$="n")
240 PRINT AT 21,6;"ALTA FUNCTIE (d/n)?": PAUSE 0
250 GO TO 10 (INKEY$="d")+260(INKEY$="n")
260 CLS : STOP

```

Pentru: $\sqrt{1-x^2}$ și $x_A=-1$, $x_B=1$ rezultă $l=3,115077$ luind $n=10$, respectiv $l=3,1414878$ luind $n=500$.

14.6. INTEGRAREA FUNCȚIILOR NUMERICE

Exemplul 14.19 : metoda dreptunghiurilor .

Fie o funcție numerică definită $f(x)$ care este continuu derivabilă pe intervalul $[a ; b]$ pentru care se cere valoarea integralei

$$\int_a^b f(x) dx$$

Reprezentînd grafic curba $f(x)$ pe intervalul $[a ; b]$, aria limitată de curbă, axa absciselor Ox și verticalele d_1 , d_2 reprezintă integrala s (fig.14.2). Dacă se descompune această arie în n dreptunghiuri de aceeași lățime, aria se poate calcula observînd că fiecare dreptunghi are ca lățime $(b-a)/n$ și ca înălțime $f(x_i)$, respectiv pentru $1 \leq i \leq n+1$ abscisa $x_i = a + (i-1)(b-a)/n$. Deci aria va fi:

$$s = \sum_{i=1}^n \frac{b-a}{n} f(x_i)$$

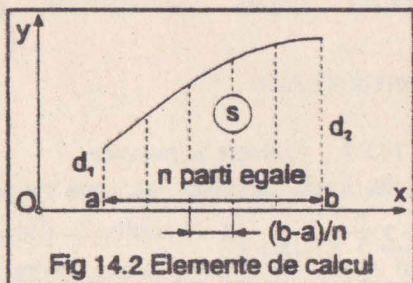


Fig 14.2 Elemente de calcul

```

8 REM valoarea integralei-met.dreptunghiurilor
10 CLS : INPUT "introduceti functia de integrat
  f(x)" ` f$
20 INPUT "Introduceti intervalul [a;b]", a,b
30 CLS : INPUT "Introduceti nr.dreptungh. n", n
35 PRINT AT 21,9;"CORECT (d/n) ?":PAUSE 0
40 GO TO 50*(INKEY$"d")+10*(INKEY$="n")
50 CLS
110 LET s=0
120 FOR i=1 TO n+1
130 LET x=a+(i-1)*(b-a)/n
140 LET s=s+VAL f$
150 NEXT i
160 CLS : PRINT AT 1,3;"Integrala functiei"; AT 3,5;
  f$; AT 5,3 "pe intervalul [";a,";";b;"]"; AT 7,3
  ;"este ";s*(b-a)/n
170 PRINT AT 21,10;"ALT n (d/n) ?": PAUSE 0
180 GO TO 25*(INKEY$='d')+190*(INKEY$="n")
190 PRINT AT 21,6;"ALTA FUNCTIE (d/n)? ": PAUSE 0
200 GO TO 10*(INKEY$="d")+210*(INKEY$="n")
210 CLS : STOP

```

Pentru: $s = \int_0^6 x^2 dx$ rezultă $s=74,1744$ pentru $n=50$; $s=72,216144$ luind
 $n=500$ și respectiv $s=72,108036$ dacă $n=1000$

$$s = \int_0^{\pi/2} \sin(x) dx \quad \text{rezultă: } s = 1,0078334 \quad \text{pentru } n=100 \quad \text{și}$$

$$s = 1,0003927 \quad \text{pentru } n=2000$$

Exemplul 14.20 : metoda trapezelor

Este similară metodei dreptunghiurilor, valoarea integralei fiind

$$s = \frac{b-a}{2n} \sum_{i=1}^n k \cdot f(x_i) \quad \left\{ \begin{array}{l} k=1 \text{ pentru } i=1 \text{ și } i=n+1 \\ k=2 \text{ în caz contrar} \end{array} \right.$$

```

8 REM valoarea integralei-met.trapezelor
10 CLS : INPUT "Introduceti functia de integrat
f(x)"`f$
20 INPUT "Introduceti intervalul [a ;b]",a,b
25 CLS : INPUT "Introduceti nr.trapezelor n",n
30 PRINT AT 21,9;:CORECT (d/n) ?": PAUSE 0
40 GO TO 50*(INKEY$="d")+10*(INKEY$="n")
50 CLS
110 LET s=0
120 FOR i=1 TO n+1
130 LET x=a+(i-1)*(b-a)/n
140 LET k=2
142 IF i=1 OR i=n+1 THEN LET k=1
145 LET s=s+k*VAL f$
150 NEXT i
160 CLS : PRINT AT 1,3;"Integrala functiei"; AT 3,5;
f$; AT 5,3:"pe intervalul [";a;"," b;"]"; AT 7,3
"este ";s*(b-a)/n/2
170 PRINT AT 21,10;"ALT n (d/n) ?"; PAUSE 0
180 GO TO 25*(INKEY$="d")+190*(INKEY$="n")
190 PRINT AT 21,6;"ALTA FUNCTIE (d/n)?" : PAUSE 0
200 GO TO 10*(INKEY$="d")+210*(INKEY$="n")
210 CLS : STOP

```

$$\text{Pentru: } s = \int_0^6 x^2 dx \quad \text{rezulta } \left\{ \begin{array}{l} s = 72,36 \quad \text{pentru } n = 10 \\ s = 72,0144 \quad \text{pentru } n = 50 \end{array} \right.$$

$$s = \int_1^2 dx/x \quad \text{rezulta} \quad \begin{cases} s = 0,69317218 & \text{pentru } n = 50 \\ s = 0,69314743 & \text{pentru } n = 500 \end{cases}$$

$$s = \int_0^2 3^x e^x dx \quad (\text{unde } e = 2,71828182) \quad \text{se obțin valorile}$$

$$\begin{cases} s = 31,2164 & \text{luind } n = 100 \\ s = 31,211864 & \text{luind } n = 1000 \end{cases}$$

Exemplul 14.21 : metoda Simpson

Relația pentru calculul integralei are forma

$$s = \frac{b-a}{6n} \sum_{i=1}^{2n+1} k \cdot f(x_i) \quad \text{unde } x_i = a + (i-1)(b-a)/2n$$

a și b sînt valorile intervalului de integrare, n este numărul părților iar $k=1$ pentru $i=1$ sau $i=2n+1$, respectiv $k=2$ pentru i impar și $k=4$ pentru i par.

```

8 REM valoarea integralei-met.Simpson
10 CLS : INPUT "Introduceti functia de integrat
f(x)"`f$
20 INPUT "Introduceti intervalul [a ;b]",a,b
30 CLS : INPUT "Introduceti nr,partilor n",n
35 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
40 GO TO 50*(INKEY$="d")+10*(INKEY$="n")
50 CLS
110 LET s=0
120 FOR i=1 TO 2*n+1
130 LET x=a(i-1)*(b-a)/2/n
135 LET k=2
136 IF i/2=INT(i/2) THEN LET k=4
138 IF i=1 OR i=2*n+1 THEN LET k=1
140 LET s=s+k*VAL f$
150 NEXT i
160 CLS : PRINT AT 1,2;"Integrala functiei"; AT 3,5;
f$; AT 5,3;"pe intervalul [";a;" ," ;b;"]"; AT
7,3;"este ";s*(b-a)/n/6
170 PRINT AT 21,10;"ALT n (d/n)?" : PAUSE 0

```

```

180 GO TO 25*(INKEY$="d")+190(INKEY$="n")
190 PRINT AT 21,6;"ALTA FUNCTIE (d/n)?: PAUSE 0
200 GO TO 10*(INKEY$="d")+210*(INKEY$="n")
210 CLS : STOP

```

Pentru: $s = \int_0^6 x^2 dx$ rezulta $s = 72$ pentru $n = 5$ și $n = 25$

$$s = \int_0^1 e^{-x^2} dx \text{ rezulta } \begin{cases} s = 0,74682495 \text{ pentru } n = 5 \\ s = 0,74682414 \text{ pentru } n = 25 \end{cases}$$

$$s = \int_0^2 3^x e^x dx \text{ rezulta } \begin{cases} s = 31,217088 \text{ pentru } n = 5 \\ s = 31,211826 \text{ pentru } n = 25 \end{cases}$$

(în ultimele două aplicații numărul $e = 2,71828182$).

Exemplul 14.22 : metoda CEBISEV

Calculul integralei în intervalul $[a ; b]$ se face cu relația

$$s = \frac{b-a}{9} \sum_{i=1}^9 f\left(\frac{b+a}{2} + \frac{b-a}{2} x_i\right)$$

unde: $x_1 = -x_9 = 0,91158931$; $x_2 = -x_8 = 0,60101866$;

$x_3 = -x_7 = 0,5286178$;

$x_4 = -x_6 = 0,16790618$; $x_5 = 0$ și $k = 2/9$.

```

8 REM valoarea integralei-met.CEBISEV
10 CLS : INPUT "Introduceti functia de integrat
f(x) " `f$
20 INPUT "Introduceti intervalul [a ; b)", a, b
30 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
40 GO TO 50*(INKEY$="d")+10*(INKEY$="n")
50 CLS
60 DIM x(9)
70 LET x(1)=0.91158931: LET x(9)=-x(1)
100 LET x(2)=0.60101866: LET x(8)=-x(2)
110 LET x(3)=0.52876178: LET x(7)=-x(3)
120 LET x(4)=0.16790618: LET x(6)=-x(4)
130 LET x(5)=0

```

```

150 LET s=0
155 LET c=(a+b)/2
156 LET d=(b-a)/2
157 FOR i=1 TO 9
158 LET x=c+d*x(i)
159 LET s=s+VAL f$
160 NEXT i
170 CLS : PRINT AT 1,3;"Integrala functiei"; AT 3,5;
    f$; AT 5,3 "pe intervalul [;a;"," ;b;]"; AT
    7,3;"este ";s*(b-a)/9
175 PRINT AT 21,6;"ALTA FUNCTIE (d/n) ?": PAUSE 0
180 GO TO 10*(INKEY$="d")+190*(INKEY$="n")
190 CLS : STOP

```

Pentru: $s = \int_0^2 3^x e^x dx$ unde $e=2,71828182$ rezultă $s=31,211809$.

14.7. ECUAȚII DIFERENȚIALE DE ORDINUL 1

Exemplul 14.23 : metoda Euler.

Fie y o funcție numerică definită prin condiția $y(x_0) = y_0$ și prin ecuația $y' = f(x, y)$: se presupune y definită și dezvoltată în serie Taylor pe intervalul $[x_0; x_0 + h]$. Dacă se păstrează primii doi termeni ai seriei se obține o valoare apropiată de $y(x)$ în vecinătatea lui x_0 dată de relația :

$$y(x) = y(x_0) + (x - x_0)y'(x_0).$$

Ținând seama de ecuația diferențială se obține $y(x) = y(x_0) + (x - x_0)f(x_0, y_0)$. Intervalul $[x_0; x]$ se împarte în n subintervale $(x_j; x_{j+1})$ de aceeași lungime l și pentru fiecare subinterval se aplică relația anterioară. Rezultă:

$$y(x_{j+1}) = y(x_j) + lf(x_j); \quad y(x_j) \quad \text{cu } l = (x - x_0)/n.$$

Dacă se notează $y(x_j) = y_j$ se poate scrie:

$$y_{j+1} = y_j + lf(x_j, y_j).$$

Prin urmare, calculul lui $y(x)$ constă în determinarea șirului $y_0, y_1, \dots, y_n = y(x)$.

```

8 REM ec.diferentiale de ord.1-met.Euler
10 CLS : INPUT "Introduceti x0,y0",x,y
20 INPUT "Introduceti ec.diferentiala(5sp)
   f(x,y)" `f$
30 INPUT "INTRODUCETI x",x
35 CLS : INPUT "Introduceti n",n
40 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
50 GO TO 60*(INKEY$="d")+10*(INKEY$="n")
60 CLS : LET y=y0
70 LET l=(x-x0)/n
80 LET x=x0
90 FOR j=0 TO n-1
100 LET f=VAL f$
110 LET y=y+l*f
120 LET x=x+l
130 NEXT j
140 CLS : PRINT AT 1,3;"Solutia y(1) a ecuatiei"; AT
   3,4;f$; AT 5,6;"este y(1)= ";y
150 PRINT AT 21,10;" ALT n (d/n) ?": PAUSE 0
160 GO TO 35*(INKEY$="d")+170*(INKEY$="n")
170 PRINT AT 21,6;"ALTA ECUATIE (d/n) ?": PAUSE 0
180 GO TO 10*(INKEY$="d")+ 190*(INKEY$="n")
190 CLS : STOP

```

Fie ecuația diferențială $y' = 2y + 2x + 1$ cu $x_0 = 0$ și $y_0 = 0$; calculînd $y(1)$ pentru $x=1$ rezultă:

$y(1) = 3,3782399$	luînd $n=5$
$y(1) = 5,1066833$	luînd $n=50$
$y(1) = 5,3160179$	luînd $n=200$
$y(1) = 5,3743126$	luînd $n=1000$

Se observă că precizia rezultatului crește cu creșterea numărului de su. intervale n .

Exemplul 14.24 : metoda Runge-Kutta

Cu aceleași ipoteze ca la metoda precedentă, noua metodă constă în a determina șirul $y_0, y_1 \dots y_n = y(x)$ folosind relațiile următoare:

$$y_{j+1} = y_j + (k_1 + 2k_2 + 2k_3 + k_4)/6 \text{ unde}$$

$$k_1 = f(x_1, y_1); k_2 = f[x_1 + (1/2), y_1 + (k_1/2)]$$

$$k_3 = f[x_1 + (1/2), y_1 + (k_2/2)]; k_4 = f(x_1 + 1, y_1 + k_3)$$

În program funcția $f(x, y)$ este introdusă sub forma unui șir de caractere care este evaluată cu ajutorul valorilor date lui x și y ; x_j și y_j sînt desemnați respectiv prin variabilele u și v .

```

8 REM ec.diferentiale de ord.1-met.Runge-Kutta
10 CLS : INPUT "Introduceti x0,y0",x0,y0
20 INPUT "Introduceti ec.diferentiala(5sp)
   f(x,y)"`f$
30 INPUT "Introduceti x",x
35 CLS : INPUT "Introduceti n",n
40 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
50 GO TO 60*(INKEY$="d")+10*(INKEY$="n")
60 CLS
70 LET l=(x-x0)/n
80 LET u=x0
85 LET v=y0
90 FOR j=0 TO n-1
92 LET x=u: LET y=v
94 LET k1=l*VAL f$
96 LET x=u+l/2
98 LET y=v+k1/2
100 LET k2=l*VAL f$
102 LET y=v+k2/2
104 LET k3=l*VAL f$
106 LET x=u+l
108 LET y=v+k3
110 LET k4=l*VAL f$
120 LET v=v+(k1+2*k2+2*k3+k4)/6
122 LET u=u+l
130 NEXT j
140 CLS : PRINT AT 1,3;"Solutia y(1) a ecuatiei";
   AT 3,4;f$; AT 5,6;"este y(1) "="v
150 PRINT AT 21,10;"ALT n (d/n) ? ": PAUSE 0
160 GO TO 35*(INKEY$/"d")+170*(INKEY$="n")
170 PRINT AT 21,6;"ALTA ECUATIE (d/n) ?": PAUSE 0
180 GO TO 10*(INKEY$="d")+190*(INKEY$="n")

```

190 CLS : STOP

Fie ecuația diferențială $y' = 2y + 2x + 1$ cu $x_0 = 0$ și $y_0 = 0$; pentru $x = 1$ rezultă soluția $y(1)$ cu următoarele valori :

$$y(1) = 5,3867937 \text{ luind } n = 5$$

$$y(1) = 5,3890448 \text{ luind } n = 20$$

$$y(1) = 5,3890558 \text{ luind } n = 50$$

Eroarea comisă este de ordinul 10^{-5} (se constată că metoda Runge-Kutta este mai precisă decît metoda Euler).

14.8. OPERAȚII CU VECTORI

Exemplul 14.25 : suma și diferența a doi vectori

Fie vectorii $\vec{u} (u_x, u_y, u_z)$ și $\vec{v} (v_x, v_y, v_z)$. Relațiile pentru vectorul:

$$\text{- sumă: } (u_x + v_x)\vec{i} + (u_y + v_y)\vec{j} + (u_z + v_z)\vec{k} = a\vec{i} + b\vec{j} + c\vec{k};$$

$$\text{modulul: } \sqrt{a^2 + b^2 + c^2}$$

$$\text{- diferență: } (u_x - v_x)\vec{i} + (u_y - v_y)\vec{j} + (u_z - v_z)\vec{k} = d\vec{i} + e\vec{j} + f\vec{k};$$

$$\text{modulul } \sqrt{d^2 + e^2 + f^2}$$

(unde $u_x, u_y, u_z; v_x, v_y, v_z$ - componentele vectorilor pe axe: $\vec{i}, \vec{j}, \vec{k}$ - versorii axelor).

Pentru a nu se complica programul definind caractere grafice noi pentru versorii s-a ales soluția indicării prin afișare a componentelor pe axe. În program s-au notat : $u_x = u(1); u_y = u(2); u_z = u(3); v_x = v(1); v_y = v(2); v_z = v(3); a = u(1) + v(1); b = u(2) + v(2); c = u(3) + v(3); d = u(1) - v(1); e = u(2) - v(2)$ și $f = u(3) - v(3)$.

8 REM suma si diferenta a doi vectori

10 CLS : DIM u(3): DIM v(3)

15 PRINT AT 0,0;"Introduceti componentele vectorului u"

20 FOR i=1 TO 3

30 INPUT u(i)

40 NEXT i

42 CLS : PRINT AT 0,0;"Introduceti componentele

```

    vecto- rului v"
44 FOR i=1 TO 3
46 INPUT v(i)
48 NEXT i
50 CLS : PRINT AT 1,1;"u(";u(1);";";u(2);";";u(3);
    ") "
60 PRINT AT 3,1;"v(" ;v(1);";";v(2);";";v(3);") "
70 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
80 GO TO 82*(INKEY$="d")+10*(INKEY$="n")
82 LET a=u(1)+v(1): LET b=u(2)+v(2): LET
    c=u(3)+v(3): LET d=u(1)-v(1): LET e=u(2)-v(2):
    LET f=u(3)-v(3)
90 CLS : PRINT AT 2,0;"Componentele vect.suma :";
    AT 4,12;"-pe Ox :";a; AT 5,12;"-pe Oy :";b; AT
    6,12;"-pe Oz :";c
100 PRINT AT 8,0;"Componentele vect.diferenta :"; AT
    10,12;"-pe Ox :";d; AT 11,12;"-pe Oy:";e; AT
    12,12 ;"-pe Oz :";f
105 PRINT AT 14,0;"Modulul vectorului : "; AT
    15,2;"-suma:";SOR(a*a+b*b+c*c)
107 PRINT AT 17,2;"-diferenta:" : SOR(d*d+e*e+f*f)
110 PRINT AT 21,7;"ALTI VECTORI (d/n)?" : PAUSE 0
120 GO TO 10*(INKEY$="d")+130*(INKEY$="n")
130 : CLS : STOP

```

Fie \vec{u} (1; 2; 4) și \vec{v} (2; 3; -2) rezultă:

- vectorul suma: $3\vec{i} + 5\vec{j} + 2\vec{k}$ cu modulul 6,164414

- vectorul diferență: $-\vec{i} - \vec{j} + 6\vec{k}$ cu modulul 6,164414

Exemplul 14.26 : produsul scalar și produsul vectorial
a doi vectori.

Fie vectorii \vec{u} (u_x, u_y, u_z) și \vec{v} (v_x, v_y, v_z). Se aplică relațiile:

- pentru produsul scalar: $u_x v_x + u_y v_y + u_z v_z$:

- pentru produsul vectorial:

$$(u_y v_z - u_z v_y)\vec{i} + (u_z v_x - u_x v_z)\vec{j} + (u_x v_y - u_y v_x)\vec{k}$$

(unde u_x, u_y, u_z : v_x, v_y, v_z - componentele vectorilor pe axe: $\vec{i}, \vec{j}, \vec{k}$ -

(versorii axelor Ox , Oy , Oz)

```

8 REM produs scalar si produs vectorial
10 CLS : DIM u(3): DIM v(3)
15 PRINT AT 0,0;"Introduceti componentele vecto-
    rului u"
20 FOR i=1 TO 3
30 INPUT u(i)
40 NEXT i
42 CLS : PRINT AT 0,0:"Introduceti componentele
    vecto- rului v"
44 FOR i=1 TO 3
46 INPUT v(i)
48 NEXT i
50 CLS : PRINT AT 1,3;"u(";u(1);";"u(2);";";u(3);
    ")"
60 PRINT AT 3,1;"v(";v(1);";";v(2);";";v(3);")"
70 PRINT AT 21,9 ";CORECT (d/n) ?": PAUSE 0
80 GO TO 90*(INKEY$="d")+10*(INKEY$="n")
90 CLS ; PRINT AT 2,5;"Produsul scalar  :": AT
    4,12;u(1)*v(1)+u(2)*v(2)+u(3)*v(3)
100 PRINT AT 6,1;"Componentele prod.vectorial  :";
    "(3sp)-pe Ox   :";u(2)*v(3)-u(3)*v(2);
    "(3sp)-pe Oy   :";u(3)*v(1)-u(1)*v(3);
    "(3sp)-pe Oz   :";u(1)*v(2)-u(2)*v(1)
110 PRINT AT 21,7;"ALTI VECTORI (d/n) ?": PAUSE 0
120 GO TO 10*(INKEY$="d")+130*(INKEY$="n")
130 CLS : STOP

```

Pentru :

a) $\vec{u}(1; 2; 4)$ și $\vec{v}(2; 3; -2)$ rezultă:

- produsul scalar: 0 (deci \vec{u} este perpendicular pe \vec{v});

- produsul vectorial: $-16\vec{i} + 10\vec{j} - \vec{k}$

b) $\vec{u}(1,2; 4; 5,5)$ și $\vec{v}(-2; 3; 4,5)$ rezultă :

- produsul scalar: 34, 35

- produsul vectorial: $1,5\vec{i} - 16,4\vec{j} + 11,6\vec{k}$

Exemplul 14.27 : produsul mixt si dublul produs vectorial.

Fie vectorii $\vec{u} (u_1, u_2, u_3)$, $\vec{v} (v_1, v_2, v_3)$ și $\vec{w} (w_1, w_2, w_3)$.

Produsul mixt:

$$\begin{aligned} \vec{u} \cdot (\vec{v} \times \vec{w}) &= \begin{vmatrix} u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \\ w_1 & w_2 & w_3 \end{vmatrix} = \\ &= u_1(v_2w_3 - v_3w_2) + u_2(v_3w_1 - v_1w_3) + u_3(v_1w_2 - v_2w_1) \\ &= a + b + c \end{aligned}$$

Dublul produs vectorial:

$$\begin{aligned} \vec{u} \times (\vec{v} \times \vec{w}) &= (d \cdot v_1 - e \cdot w_1)\vec{i} + (d \cdot v_2 - e \cdot w_2)\vec{j} + (d \cdot v_3 - a \cdot w_3)\vec{k} = \\ &= f\vec{i} + g\vec{j} + h\vec{k} \end{aligned}$$

unde: $d = u_1w_1 + u_2w_2 + u_3w_3$; $e = u_1v_1 + u_2v_2 + u_3v_3$.

```

8 REM produs mixt si dublul produs vectorial
10 CLS : DIM u(3): DIM v(3): DIM w(3)
20 FOR i=1 TO 3
30 INPUT "Introduceti componentele vectorului u"
  `u(i)
40 NEXT i
41 FOR i=1 TO 3
42 CLS : INPUT "Introduceti componentele vectorului v" `v(i)
43 NEXT i
44 FOR i=1 TO 3
45 CLS : INPUT "Introduceti componentele vectorului w" `w(i)
46 NEXT i
50 PRINT AT 1,3;"u(";u(1);";";u(2);";";u(3);")"
60 PRINT AT 2,3;"v(";v(1);";";v(2);";";v(3);")"
70 PRINT AT 3,3;"w(";w(1);";";w(2);";";w(3);")"
80 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
90 GO TO 100*(INKEY$="d")+10*(INKEY$="n")
100 CLS
110 LET a=u(1)*(v(2)*w(3)-v(3)*w(2))
120 LET b=u(2)*(v(3)*w(1)-v(1)*w(3))
130 LET c=u(3)*(v(1)*w(2)-v(2)*w(1))
140 LET d=u(1)*w(1)+u(2)*w(2)+u(3)*w(3)
150 LET e=u(1)*v(1)+u(2)*v(2)+u(3)*v(3)

```

```

160 CLS : PRINT AT 1,3;"Produsul mixt : "AT 3,5;a+b+c
170 PRINT AT 5,0;"Componentele dub.pr.vectorial:";AT
    7,1;"-pe Ox : ";d*v(1)-e*w(1); AT 8,1;"-pe Oy : "
    ;d*v(2)-e*w(2); AT 9,1;"-pe Oz : ";
    d*v(3)-e*w(3)
175 LET f=d*v(1)-e*w(1): LET g=d*v(2)-e*w(2): LET
    h=d*v(3)-e*w(3): PRINT AT 11,0:"Modulul dub.pr.
    vectorial : "; AT 13,11; SOR(f*f+g*g+h*h)
180 PRINT AT 21,6;"ALTI VECTORI (d/n) ?": PAUSE 0
190 GO TO 10*(INKEY$="d")+200*(INKEY$="n")
200 CLS : STOP

```

Pentru : a) $\vec{u}(2; 1; -1)$, $\vec{v}(3; 2; 5)$ și $\vec{w}(-1; 3; 2)$ rezultă:

- produsul mixt: -44

-dublul produs vectorial: $\begin{cases} -11\vec{j} - 11\vec{k} \\ \text{modulul } 15,556349 \end{cases}$

b) $\vec{u}(1; 2; 3)$, $\vec{v}(4; 5; 6)$ și $\vec{w}(7; 8; 9)$ rezultă:

- produsul mixt: 0 (vectori coplanari)

-dublul produs vectorial $\begin{cases} -24\vec{i} - 6\vec{j} + 12\vec{k} \\ \text{modulul } : 27,495454 \end{cases}$

Observatie: programele referitoare la vectori pot fi perfecționate definind caractere grafice pentru vectorii axelor conform datelor din tabelul următor

Vectorul	DATA
\vec{i}	2,255,2,32,32,32,32,28
\vec{j}	2,252,2,8,8,8,8,48
\vec{k}	2,252,2,36,40,48,40,36

14.9. MATRICE ȘI ECUAȚII LINIARE

Exemplul 14.28 : înmulțirea unei matrice cu un scalar

```

8 REM Inmultirea unei matrice cu un scalar
10 CLS : INPUT "Introduceti nr.liniilor n si al

```

```

coloanelor p pentru matricea A(n,p)
20 DIM a(n,p): DIM c(n,p)
25 INPUT "Introduceti scalarul k",k: PRINT AT 1,5;
   "Scalarul k=";k
27 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
28 GO TO 30*(INKEY$="d")+10*(INKEY$="n")
30 FOR i=1 TO n
40 FOR j=1 TO p
50 CLS : INPUT "Introduceti valorile matricei
   A(n,p)",a(i,j)
60 NEXT j
70 NEXT i
80 CLS : FOR i=1 TO n
90 FOR j=1 TO p
100 PRINT AT 1,3;"A(";i ;",";j ;")=";a(i,j)
110 NEXT j
120 NEXT i
130 PRINT AT 21,9;"CORECT (d/n)?": PAUSE 0
140 GO TO 150*(INKEY$="d")+130*(INKEY$="n")
270 CLS
280 FOR i=1 TO n
290 FOR k=1 TO p
300 PRINT "C(";i ;",";j ;")=";k*a(i,j)
310 NEXT j
320 NEXT i
330 PRINT AT 21,7;"ALTE VALORI (d/n)?": PAUSE 0
340 GO TO 10*(INKEY$="d")+350*(INKEY$="n")
350 CLS : STOP

```

$$\text{Pentru } A(3,3) = \begin{vmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{vmatrix} \text{ și } k=3 \text{ rezultă } C(3,3) = \begin{vmatrix} 3 & 6 & 9 \\ 12 & 15 & 18 \\ 21 & 24 & 27 \end{vmatrix}$$

Exemplul 14.29 : adunarea a 2 matrice cu n linii și p coloane.

Fie A și B două matrice cu n linii și p coloane fiecare: pentru $1 \leq i \leq n$ și $1 \leq j \leq p$ matricea sumă se determină folosind relația:

$$C(i,j) = A(i,j) + B(i,j).$$

```
8 REM adunarea a 2 matrice A(n,p) si B(n,p)
10 CLS : INPUT "Introduceti nr.liniilor n si al
    coloanelor p" `n,p
12 IF n=0 OR n<0 AND p=0 OR p<0 THEN GO TO 10
20 DIM a(n,p): DIM b(n,p)
30 FOR i=1 TO n
40 FOR j=1 TO p
50 CLS : INPUT "Introduceti valorile matricei(3sp)
    A(n,p)",a(i,j)
60 NEXT j
70 NEXT i
80 CLS : FOR i=1 TO n
90 FOR j=1 TO p
100 PRINT "A(";i:",",";j;")=";a(i,j)
110 NEXT j
120 NEXT i
130 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
140 GO TO 150*(INKEY$ ="d")+30*(INKEY4="n")
150 FOR i=1 TO n
160 FOR j=1 TO p
170 CLS : INPUT "Introduceti valorile matricei(3sp)
    B(n,p)",b(i,j)
180 NEXT j
190 NEXT i
200 CLS : FOR i=1 TO n
210 FOR j=1 TO p
220 PRINT "B(";i ;",",";j;")=";b(i,j)
230 NEXT j
240 NEXT i
250 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
260 GO TO 270*(INKEY$="d")+150*(INKEY$="n")
270 CLS
280 OR i=1 TO n: FOR j=1 TO p
290 PRINT"C(";i:",",";j;")=";a(i,j)+b(i,j)
300 NEXT j: NEXT i
310 PRINT AT 21,7;"ALTE MATRICE (d/n) ?": PAUSE 0
320 GO TO 10*(INKEY$ ="d")+330*(INKEY$="n")
330 CLS : STOP
```


$$\text{Pentru } A(3,2) = \begin{vmatrix} 3 & 1 \\ 5 & 2 \\ 2 & 0 \end{vmatrix} \text{ și } B(3,2) = \begin{vmatrix} 2 & 0 \\ 1 & 1 \\ -1 & -1 \end{vmatrix} \text{ rezultă } C(3,2) = \begin{vmatrix} 5 & 1 \\ 6 & 3 \\ 1 & -1 \end{vmatrix}$$

Exemplul 14.30 : înmulțirea a două matrice

Înmulțirea a două matrice $A(n,p)$ și $B(p,q)$ este posibilă numai dacă numărul coloanelor matricei A este egal cu numărul liniilor matricei B (adică $p=q$). Elementul $C(i,j)$ al matricei produs se obține înmulțind elementele liniei i din matricea A cu elementele coloanei j din matricea B și însumând:

$$C(i,j) = \sum_{k=1}^p A(i,k) \cdot B(k,j)$$

```

8 REM Inmultirea a 2 matrice
10 CLS : INPUT "Introduceti valorile n si p pentru
matricea A(n,p)" `n,p
20 CLS : INPUT "Introduceti valorile q,r pentru
matricea B(q,r)" `q,r
30 IF q <> p THEN BEEP .02,10: PAUSE 4: BEEP .02,-
20: CLS : PRINT AT 21,1; FLASH 1;"EROARE !":
FLASH 0; AT 21,16 ;"Trebuie q=p!": PAUSE 0: GO
TO 20
40 DIM a(n,p): DIM b(q,r): DIM c(n,r): CLS
50 FOR i=1 TO n: FOR j=1 TO p: INPUT "Introduceti
valorile matricei A";a(i,j): PRINT "A(;i;",
";j;)"=":a(i,j): NEXT j: NEXT i
60 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
70 GO TO 80*(INKEY$="d")+10*(INKEY$="n")
80 CLS : FOR i=1 TO q: FOR j=1 TO r: INPUT
"Introduceti valorile matricei B";b(i,j): PRINT
"B(;i;",";j;)"=":b(i,j): NEXT j: NEXT i
90 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
100 GO TO 110*(INKEY$="d")+80*(INKEY$="n")
111 CLS : FOR i=1 TO n: FOR j=1 TO r: LET c(i,j)=0:
FOR k=1 TO p: LET c(i,j)=c(i,j)+a(i,k)*b(k,j):
NEXT k: PRINT "C(;i;",";j;)"=":c(i,j): NEXT j:
NEXT i

```

```

120 PRINT AT 21,7;"ALTE MATRICE (d/n)?" : PAUSE 0
130 GO TO 10*(INKEY$="d")+140*(INKEY$="n")
140 CLS : STOP

```

Pentru: a) $A(2,3) = \begin{vmatrix} 2 & 1 & 0 \\ 0 & 3 & 1 \end{vmatrix}$ și $B(3,3) = \begin{vmatrix} -3 & 4 & -1 \\ 0 & 2 & -5 \\ 3 & 0 & 1 \end{vmatrix}$ rezultă

$$C(2,3) = \begin{vmatrix} -6 & 10 & 7 \\ -3 & 6 & -16 \end{vmatrix}$$

b) $A(2,3) = \begin{vmatrix} 3 & 1 & 2 \\ 4 & 0 & 1 \end{vmatrix}$ și $B(3,2) = \begin{vmatrix} 1 & 0 \\ 0 & 1 \\ 2 & 1 \end{vmatrix}$ rezultă $C(2,2) = \begin{vmatrix} 7 & 3 \\ 6 & 1 \end{vmatrix}$

Exemplul 14.31 : imaginea unui vector printr-o matrice patrată

Fie $A(n,n)$ o matrice patrată și \bar{x} un vector de componente $x_1, x_2 \dots x_n$.

Imaginea prin A a vectorului \bar{x} este un vector \bar{y} de componente $y_1, y_2 \dots y_n$ cu

$$y_i = \sum_{j=1}^n A(i,j) \cdot x_j$$

8 REM imaginea unui vector printr-o matrice patrata

10 CLS : INPUT "Introduceti nr.liniiilor si co-(3sp)loanelor matricei A(n,n)",n

12 IF n=0 OR n<0 THEN GO TO 10

20 PRINT AT 21,9;"CORECT (d/n) ?" : PAUSE 0

30 GO TO 40*(INKEY\$="d")+10*(INKEY\$="n")

40 DIM a(n,n): DIM x(n): DIM y(n): CLS

50 FOR i=1 TO n: FOR j=1 TO n: INPUT "Introduceti valorile matricei(3sp)A(n,n)"`a(i,j): PRINT "A(";i;",";j;")=";a(i,j): NEXT j: NEXT i

70 PRINT AT 21,9;"CORECT (d/n) ?" : PAUSE 0

80 GO TO 90*(INKEY\$="d")+40*(INKEY\$="n")

90 CLS : FOR i=1 TO n: INPUT "Introduceti valorile vectorului x",x(i): PRINT "X(";i;")=";x(i):

```

NEXT i
100 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
110 GO TO 120*(INKEY$="d")+90*(INKEY$="n")
120 CLS ; PRINT AT 0,5;"Vectorul imagine Y(i)":
PRINT
130 FOR i=1 TO n: LET y(i)=0: FOR j=1 TO n: LET
y(i)=y(i)+a(i,j)*x(j): NEXT j: PRINT "Y(";
i ;")="";y(i): NEXT i
140 PRINT AT 21,7;"ALTE VALORI (d/n)?" : PAUSE 0
150 GO TO 10*(INKEY$="d")+160*(INKEY$="n")
160 CLS ; STOP

```

Pentru: $A(3,3) = \begin{vmatrix} 3 & -2 & -1 \\ 2 & 5 & 0 \\ 1 & 0,5 & 2 \end{vmatrix}$ și $x(2; 1; -1)$ rezultă $y(5; 9; 0,5)$.

Exemplul 14.32 : inversarea unei matrice patrate

```

8 REM Inversarea unei matrice patrate
10 CLS : INPUT : Introdueți nr.liniilor si(6sp)al
coloanelor n",n
12 IF n=0 OR n<0 THEN GO TO 10
15 PRINT AT 21,9;"CORECT (d/n)?" : PAUSE 0
17 GO TO 20*(INKEY$="d")+10*(INKEY$="n")
20 DIM a(n+1,n): DIM b(n+1,n)
25 REM intrarea valorilor A(i,j) si a matricei
unitate
30 CLS : FOR i=1 TO n: FOR j=1 TO n: INPUT
"Introdueți valorile matricei(3sp)A(i,j)",
a(i,j): PRINT "A(";i ;",";j;")="";a(i,j)
40 LET b(i,j)=0
42 IF i=j THEN LET b(i,j)=1
44 NEXT j: NEXT i
52 PRINT AT 21,9 ;"CORECT (d/n) ?": PAUSE 0
55 GO TO 140*(INKEY$="d")+30*(INKEY$="n")
85 REM inversarea matricei A(i,j)
140 CLS : FOR k=1 TO n
150 IF a(k,k)=0 THEN LET c=0

```

```
160 IF a(k,k)=0 THEN GO TO 370
165 REM se imparte linia k prin A(k,k)
170 LET d=a(k,k)
180 FOR j=1 TO n
190 LET a(k,j)=a(k,j)/d
200 LET b(k,j)=b(k,j)/d
210 NEXT j
215 REM se transforma A(i,k) in 0 pt. i<>k
220 FOR i=1 TO n
230 IF i=k THEN GO TO 290
240 LET e=a(i,k)
250 FOR j=1 TO n
260 LET a(i,j)=a(i,j)-e*a(k,j)
270 LET b(i,j)=b(i,j)-e*b(k,j)
280 NEXT j
290 NEXT i
300 NEXT k
305 REM editarea matricei inverse
310 FOR i=1 TO n
320 FOR j=1 TO n
330 PRINT "B(";i;",";j;")=";b(i,j)
340 NEXT j
350 NEXT i
360 GO TO 500
365 REM cazul matricei A(k,k)=0
370 LET c=c+1
380 FOR j=1 TO n
390 LET a(n+1,j)=a(k,j)
400 LET b(n+1,j)=b(k,j)
410 NEXT j
420 FOR l=k TO n
430 FOR j=1 TO n
440 LET a(l,j)=a(l+1,j)
450 LET b(l,j)=b(l+1,j)
460 NEXT j
470 NEXT l
480 IF c <= n-k THEN GO TO 160
490 PRINT AT 11,2; INVERSE 1;"MATRICEA ESTE
NEINVERSABILA ! "
```

```

500 PRINT # 0; AT 0,7; PAPER 6; INK 2;"ALTA MATRICE
      (d/n) ?": PAUSE 0
510 GO TO 10*(INKEY$="d")+520*(INKEY$="n")
520 CLS : STOP

```

Pentru a) $A(3,3) = \begin{vmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{vmatrix}$ rezultă matrice neinvertibilă;

b) $A(3,3) = \begin{vmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{vmatrix}$ rezultă matricea $B(3,3) = \begin{vmatrix} 0,5 & 0,5 & -0,5 \\ -0,5 & 0,5 & 0,5 \\ 0,5 & -0,5 & 0,5 \end{vmatrix}$

Exemplul 14.33 : rezolvarea unui sistem de ecuații lineare.

Se consideră sistemul de n ecuații lineare cu n necunoscute :

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n \end{cases}$$

Pentru rezolvare se notează cu \bar{x} vectorul $(x_1, x_2 \dots x_n)$ și cu \bar{b} vectorul $(b_1, b_2 \dots b_n)$. Dacă A este matricea coeficienților necunoscutelor a_{ij} , atunci sistemul este echivalent cu $A(\bar{x}) = \bar{b}$ și el are soluție dacă matricea A este invertibilă, respectiv $\bar{x} = A^{-1}(\bar{b})$. În programul următor coeficienții matricei unitate și apoi cei ai matricei A^{-1} sînt notați $q(i, j)$.

```

8 REM Rezolvarea unui sistem de ecuatii lineare
10 CLS : INPUT "Introduceti nr.ecuatiilor n",n
12 IF n=0 OR n<0 THEN GO TO 10
15 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
17 GO TO 20*(INKEY$="d")+10*(INKEY$="n")
20 CLS : DIM a(n+1,n): DIM q(n+1,n): DIM x(n): DIM
      b(n)
25 REM Introducerea valorilor pentru A(i,j) si B(i)
30 FOR i=1 TO n: FOR j=1 TO n: INPUT "Introduceti
      coeficientii necunoscutelor A(i,j)",a(i,j):
      PRINT "A(";i;","j;")=";a(i,j)
32 LET q(i,j)=0

```

```
34 IF i=j THEN LET q(i,j)=1
40 NEXT j: NEXT i
52 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
55 GO TO 60*(INKEY$="d")+30*(INKEY$="n")
60 CLS : FOR i=1 TO n: INPUT "Introduceti termenii
    liberi B(i)",b(i): PRINT "B(";i;")=";b(i):NEXT i
70 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
80 GO TO 220*(INKEY$="d")+60*(INKEY$="n")
85 REM inversarea matricei A(i,j)
220 CLS : FOR k=1 TO n
230 IF a(k,k)=0 THEN LET c=0
240 IF a(k,k)=0 THEN GO TO 400
250 LET d=a(k,k)
260 FOR j=1 TO n
270 LET a(k,j)=a(k,j)/d
280 LET q(k,j)=q(k,j)/d
290 NEXT j
300 FOR i=1 TO n
310 IF i=k THEN GO TO 370
320 LET e=a(i,k)
330 FOR j=1 TO n
340 LET a(i,j)=a(i,j)-e*a(k,j)
350 LET q(i,j)=q(i,j)-e*q(k,j)
360 NEXT j
370 NEXT i
380 NEXT k
390 GO TO 540
400 LET c=c+1
410 FOR j=1 TO n
420 LET a(n+1,j)=a(k,j)
430 LET q(n+1,j)=q(k,j)
440 NEXT j
450 FOR l=k TO n
460 FOR j=1 TO n
470 LET a(l,j)=a(l+1,j)
480 LET q(l,j)=q(l+1,j)
490 NEXT j
500 NEXT l
510 IF c <= n-k THEN GO TO 240
```

```

520 PRINT AT 11,7; INVERSE 1;"NU EXISTA SOLUTIE !!":
    BEEP.02,40: PAUSE 4: BEEP 02,-20
530 GO TO 620
535 REM afisarea solutiei
537 CLS : PRINT AT 0,3; PAPER 5; INK 1;"SOLUTIILE
    SISTEMULUI :": PRINT
540 FOR i=1 TO n
550 LET x(i)=0
560 FOR j=1 TO n
570 LET x(i)=x(i)+q(i,j)*b(j)
580 NEXT j
590 NEXT i
610 FOR i=1 TO n: PRINT "x(";i;")=";x(i): NEXT i
620 BEEP .02,40: PAUSE 4: BEEP .02,-20: PRINT #0: AT
    0,7: PAPER 6; INK 2;"ALT SISTEM (d/n)?" : PAUSE 0
630 GO TO 10*(INKEY$="d")+640*(INKEY$="n")
640 CLS ; STOP

```

$$\text{Pentru: } \begin{cases} 7,6x_1 + 0,72x_2 + 0,53x_3 = 17,07 \\ -3,4x_1 + 6,82x_2 + 0,49x_3 = 32,16 \\ 0,3x_1 + 4,19x_2 + 7,8x_3 = 13,13 \end{cases}$$

rezultă: $x_1 = 1,8058517$; $x_2 = 5,7206562$ și $x_3 = -1,4591416$.

14.10. DISTRIBUTII STATISTICE

Exemplul 14.34 : indicatori statistici

Fie $x_1, x_2 \dots x_n$ o serie de n numere. Se pot calcula următorii indicatori:

$$\text{- media: } m = \frac{1}{n} \sum_{i=1}^n x_i$$

- dispersia (varianța):

$$v = \frac{1}{n} \sum_{i=1}^n (x_i - m)^2 = \frac{1}{n} \left(\sum_{i=1}^n x_i^2 \right) - m^2$$

- abaterea medie patratcă: $e = \sqrt{v}$

```

3 REM Indicatori statistici (var.1)
10 CLS : INPUT "Introduceti nr.de valori n",n
15 IF n<1 THEN GO TO 10
20 DIM x(n): DIM y(n)
30 FOR i=1 TO n: INPUT "Introduceti sirul de valori
  x(n)",x(i): PRINT "x(";i;")=";x(i): LET y(i)=
  x(i): NEXT i
40 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
50 GO TO 60*(INKEY$="d")+10*(INKEY$="n")
60 CLS : LET s=0: LET c=0
70 FOR i=1 TO n
80 LET s=s+s*x(i)
90 LET c=c+x(i)*x(i)
100 NEXT i
210 PRINT AT 6,0;"Media :";s/n
220 LET v=c/n-(s/n)*(s/n): PRINT AT 0,8;"Dispersia :
  " ;v
230 PRINT AT 10,0;"Abaterea medie patratice :"; AT
  12,6; SQR (v)
240 PRINT AT 21,7: INVERSE 1;"ALTE VALORI (d/n)?"":
  BEEP .02,40: PAUSE 4: BEEP .02,-20: PAUSE 0
250 GO TO 10*(INKEY$="d")+260*(INKEY$="n")
260 CLS : STOP

```

Pentru şirul de valori : 2 ; 2 ; 5 ; 8 ; 8 ; 9 ; 12 ; 15 rezultă :

- media :	7.625
- dispersia:	18.234375
- abaterea medie patratică:	4.2701727

Programul poate fi perfecţionat folosind frecvenţa de apariţie a valorilor.

Exemplul 14.35

```

3 REM Indicatori statistici (var.2)
10 CLS : INPUT "Introduceti nr.total de valori n",n
15 IF n<1 THEN GO TO 10
17 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0

```



```

20 GO TO 60*(INKEY$="d")+10*(INKEY$="n")
60 CLS : LET s=d: LET c=0: LET j=0
65 IF j=0 THEN PRINT "Prima valoare ?"
70 If j <> 0 THEN PRINT :Valoarea urmatoare ?"
80 INPUT x
90 PRINT "De cite ori (frecventa)?"
100 INPUT f
110 CLS : LET s=s+f*x: LET c=c+f*x*x: LET j=j+f
120 IF j<n THEN GO TO 70
130 PRINT AT 0,3: PAPER 5: INK 1;"INDICATORII
    STATISTICI"
140 PRINT "Media :";s/n
150 LET v=((c/n))-((s/n)*(s/n))
160 PRINT "Dispersia :";v
170 PRINT "Abaterea medie patratica : " "
    "(6sp)";SOR(v)
180 PRINT # 1; AT 1,7; PAPER 6; INK 2;"ALTE VALORI
    (d/n)?"BEEP .02,40: PAUSE 4: BEEP .02,-20:
    PAUSE 0
190 GO TO 10*(INKEY$="d")+200*(INKEYS="n")
200 CLS : STOP

```

Pentru datele înscrise în tabelul următor :

Valoarea	Frecvența
3	3
5	1
7	2
9	1
10	4
12	1

se obțin: media $m = 7,4166667$, dispersia $v = 9,5763889$ și abaterea medie patratică $e = 3,0945741$.

Exemplul 14.36 : coeficientul de corelație lineară.

Fie $x_1, x_2 \dots x_n$ și $y_1, y_2 \dots y_n$ două serii de n numere fiecare. Dacă m_x și m_y sînt mediile acestor două serii, atunci rezultă:

- dispersia valorilor $x(i)$: $v_x = \frac{1}{n} \sum_1^n (x_i - m_x)^2 = \left(\frac{1}{n} \sum_1^n x_i^2 \right) - m_x^2$

- dispersia valorilor $y(i)$: $v_y = \frac{1}{n} \sum_1^n (y_i - m_y)^2 = \left(\frac{1}{n} \sum_1^n y_i^2 \right) - m_y^2$

- covarianța valorilor $x(i)$, $y(i)$:

$$\text{cov}(x, y) = \frac{1}{n} \sum_1^n (x_i - m_x)(y_i - m_y) = \frac{1}{n} \sum_1^n x_i y_i - m_x m_y$$

- coeficientul de corelație: $r = \text{cov}(x, y) / \sqrt{v_x v_y}$

8 REM coef.de corelatie

```

10 CLS : INPUT "Introduceti nr.total de valori n",n
12 IF n<1 THEN GO TO 10
15 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
18 GO TO 20*(INKEY$="d")+10*(INKEY$="n")
20 LET s=0: LET t=0: LET u=0: LET v=0: LET w=0
30 CLS : FOR i=1 TO n: INPUT "Introduceti x(i)",x:
  PRINT AT i,0;"x(";i;")=";x: INPUT "Introduceti
  y(i)",y: PRINT AT i,16;"y(";i;")=";y: LET s=s+x:
  LET t=t+x*x: LET u=u+y: LET v=v+y*y: LET
  w=w+x*y: NEXT i
40 PRINT # 1; AT 1,9;"CORECT (d/n)?" : PAUSE 0
50 GO TO 120*(INKEY$="d")+30*(INKEY$="n")
120 CLS : LET mx=s/n: LET my=u/n: PRINT AT 0,0;
  "Media sirului x(i)"; AT 1,8;mx
130 PRINT AT 3,0;"Media sirului y(i)"; AT 4,8;my
140 LET vx=t/n-mx*mx: LET vy=v/n-my*my
150 PRINT AT 6,0;"Dispersia sirului x(i)"; AT 7,8;vx
160 PRINT AT 9,0;"Dispersia sirului y(i)" AT 10,8;vy
170 LET cov=w/n-mx*my
175 PRINT AT 12,0;"Covarianta :"; AT 13,8;cov
180 PRINT AT 15,0;"Coef.de corelatie :"; AT 16,8;
  cov/SOR(vx*vy)
190 PRINT # 1; AT 1,7; PAPER 6; INK 2;"ALTE VALORI
  (d/n)?" : BEEP .02,40: PAUSE 4: BEEP .020,20:

```

PAUSE 0

200 GO TO 10*(INKEY\$="d")+210*(INKEY\$="n")

210 CLS : STOP

Pentru: $x = \{5; 8; 9; 10; 12; 15\}$ și $y = \{1; 5; 7; 11; 13; 20\}$ rezultă:

- mediile:	$m_x = 9,83333333$; $m_y = 9,5$;
- dispersiile:	$v_x = 9,80555556$; $v_y = 37,25$;
- covariația:	$cov = 18,9166667$;
- coeficientul de corelație:	$r = 0,98979526$.

Exemplul 14.37: ajustări lineare.

Fie $x = \{x_1; x_2; \dots; x_n\}$ și $y = \{y_1; y_2; \dots; y_n\}$ două serii de câte n numere fiecare. Se caută determinarea ecuației unei drepte de forma $y = ax + b$ care să ajusteze cel mai bine cele n puncte de coordonate x_i și y_i . Metoda celor mai mici pătrate conduce la următoarele expresii:

$$a = \text{cov}(x,y)/v_x \quad ; \quad b = m_y - a m_x$$

8 REM ajustari lineare

10 CLS : INPUT :Introduceti nr.total de valori n",n

12 IF n<1 THEN GO TO 10

15 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0

17 GO TO 20*(INKEY\$="d")+10*(INKEY\$="n")

20 LET s=0: LET t=0: LET u=0: LET v=0: LET w=0

30 CLS : FOR i=1 TO n: INPUT "Introduceti x(i)",x:

PRINT AT i,0;"n(";i;")=";x: INPUT "Introduceti

y(i)",y: PRINT AT i,16;"y(";i;")=";y: LET s=s+x:

LET t=t+x*x: LET u=u+y: LET v=v+y*y: LET

w=w+x*y: NEXT i

40 PRINT # 1; AT 1,9;"CORECT (d/n) ?": PAUSE 0

50 GO TO 60*(INKEY\$="d")+30*(INKEY\$="n")

60 LET mx=s/n: LET my=u/n: LET vx=t/n-mx*mx: LET

vy=v/n-my*my: LET cov=w/n-mx*my: LET a=cov/vx:

LET b=my-a*mx

70 CLS : PRINT AT 0,8;"Ecuatia cautata :""y=";

a;";"x+";b

80 PRINT "Coef.de ajustare :": INT(ABS(100*cov/
SOR(vx*vy)))/100

90 PRINT # 1; AT 1,7: PAPER 6: INK 2;"ALTE VALORI

```

(d/n)?: BEEP .02,40: PAUSE 4: BEEP .02,-20:
PAUSE 0
100 GO TO 10*(INKEY$="d")+110*(INKEY$="n")
110 CLS : STOP

```

Pentru $x = \{3; 4; 6; 8; 8; 12\}$ și $y = \{3; 6; 5; 5; 8; 11\}$ rezultă ecuația dreptei

$$y = 0,7444795x + 1,2460568$$
și coeficientul de ajustare $r = 0,86$.

Exemplul 14.38 : histograme

Fie x_1, x_2, \dots, x_n o serie de k numere. Această serie de valori poate fi reprezentată grafic de o histogramă cu n clase de aceeași amplitudine. Programul următor permite construirea unei histograme cu $n \leq 12$.

```

8 REM histograme (var1)
10 CLS : INPUT "Introduceti nr.de valori al(5sp)
seriei k",k
12 IF k<1 THEN GO TO 10
30 DIM x(k)
35 REM introducerea valorilor x(k);cautarea
maximului si minimului
40 LET m=-10↑38: REM cea mai mare valoare negativa
50 LET p=-m
60 FOR i=1 TO k: INPUT "Introduceti x(k)",x(i):
PRINT "x(";i;")=";x(i)
70 IF x(i) >= m THEN LET m=x(i)
80 IF x(i) <= p THEN LET p=x(i)
90 NEXT i
100 PRINT # 1; AT 1,9;"CORECT (d/n) ?": PAUSE 0
110 GO TO 140*(INKEY$="d")+10*(INKEY$="n")
120 REM cautarea limitelor L(i) si al efectivului
e(i) al fiecarei clase
140 CLS : INPUT "Introduceti nr.claselor n <= 12",n
142 PRINT # 0; AT 0,9;"CORECT (d/n) ?": PAUSE 0
145 GO TO 150*(INKEY$="d")+140*(INKEY$="n")
150 CLS

```

```
160 DIM L(n+1): DIM e(n)
180 FOR i=1 TO n
190 LET L(i)=p+(i-1)*(m-p)/n
200 LET e(i)=0
210 NEXT i
220 LET L(n+1)=m+1
230 FOR j=1 TO n
240 FOR i=1 TO n
250 IF x(j) >= L(i) AND x(j)<L(i+1) THEN LET e(i)=
    e(i)+1
260 NEXT i
270 NEXT j
275 REM calculul frecventelor claselor f(i) si al
    inaltimilor dreptunghiurilor h(i)
280 DIM f(n): DIM h(n): LET r=10↑(-6)
310 FOR i=1 TO n
320 LET f(i)=100*e(i)/k
330 LET h(i)=INT(0,4*f(i)+0.5)
340 IF h(i) >= r THEN LET x=h(i)
350 NEXT i
360 FOR i=1 TO n
370 LET h(i)=h(i)*80/r
380 NEXT i
385 REM trasarea histogramei
400 BORDER 2: PAPER 0: INK 7: CLS : PLOT 0,0: DRAW
    255,0: DRAW 0,175: DRAW -255,0: DRAW 0,-175:
    LET x=0
420 FOR i=1 TO n
430 FOR j=0 TO 3
440 FOR y=3 TO h(i)
450 PLOT 5+x+j,y
455 IF f(i) <= 0.1 THEN LET h(i)=0: GO TO 510
460 NEXT y
470 NEXT j
490 IF n <= 10 THEN LET w=6
500 IF n >= 10 THEN LET w=5
510 LET x=x+w
520 NEXT i
530 OVER 1: PRINT AT 1,1;"Frecventele sint
```

```

urmatoarele: ": FOR z=1 TO n: PRINT AT z+2,20;
"f(";z;")=";INT (f(z)+0.5); : NEXT z
540 PRINT AT 18,12: PAPER 6;"ALTE VALORI (d/n)?" :
PAUSE 0
550 GO TO 10*(INKEY$="d")+560*(INKEY$="n")
560 CLS : STOP

```

Pentru: a) seria de valori ($k=20$): 2; 5; 8; 1; 4; 9; 3; 6; 4; 9; 7; 3; 1; 2; 5; 9; 8; 4; 5; 2 și $n=10$ clase rezultă frecvențele: 10; 15; 10; 15; 0; 15; 5; 5; 10; 15

b) seria de valori ($k=27$): 3; 5; 7; 8; 6; 4; 5; 3; 8; 7; 6; 5; 4; 1; 2; 3; 2; 5; 4; 8; 9; 5; 2; 1; 3; 7; 8 și $n=10$ clase rezultă frecvențele: 7; 11; 15; 11; 0; 19; 7; 11; 15; 4.

Valorile frecvențelor sînt rotunjite la întregul cel mai apropiat.

Exemplul 14.39

```

8 REM histograme (var.2)
10 CLS : INPUT "Introduceti nr.de clase n<=20",n
12 IF n<1 THEN GO TO 10
30 DIM t(n): LET mm=INT(255/(3*n+1)): LET
m1=mm+INT(255-mm-n*(3*mm))/n: LET k=0
40 CLS : FOR i=1 TO n: INPUT "Introduceti
efectivele claselor",t(i): PRINT "Clasa ";
i;"=";t(i):LET k=k+t(i): NEXT i
50 PRINT # 1; AT 1,9;"CORECT (d/n) ?": PAUSE 0
52 DIM f(n): FOR i=1 TO n: LET f(i)=100*t(i)/k:
NEXT i
60 GO TO 52*(INKEY$="d")+40*(INKEY$="n")
70 CLS : LET t1=0
80 FOR i=1 TO n
90 IF t(i)>t1 THEN LET t1=t(i)
100 NEXT i: LET t1=(175-mm)/t1
110 BORDER 2: PAPER 0: INK 7: CLS
120 PLOT 0,0: DRAW 255,0: DRAW 0,175: DRAW -255,0:
DRAW 0,-175
130 FOR i=1 TO n
135 OVER 1

```

```

140 PLOT mm*i+m1*(i-1),0
150 OVER 0: DRAW 0,t(i)*t1/2: DRAW m1,0: DRAW
    0,-t(i)*t1/2
160 NEXT i
180 OVER 1: PRINT AT 1,1;"Frecventele claselor sint
    ":" FOR i=1 TO n: PRINT AT (i+1),22;"f(";i;)"=";
    INT t(i)+0.5);"%";: NEXT i
190 PRINT #1, AT 1,7: PAPER 4; INK 6;"ALTE VALORI
    (d/n)?": PAUSE 0
200 GO TO 10*(INKEY$="d")+210*(INKEY$="n")
210 CLS : STOP

```

Pentru $n=10$ clase avînd efectivele: 20; 30; 40; 60; 90; 95; 100; 110; 120; 60; rezultă frecvențele: 3; 4; 6; 8; 12; 13; 14; 15; 17; 8.

Exemplul 14.40

```

8 REM multihistograme
10 BORDER 1: PAPER 1: INK 7: CLS
40 INPUT "Cite histograme (<9) ? ",n
45 IF n<1 THEN GO TO 40
50 IF n>8 THEN PRINT #1; AT 1,1 "Vrei prea
    multe !!": PAUSE 100: GO TO 40
70 INPUT AT 0,0;"Cite clase va avea fiecare
    va-(2sp)riabila( =";(INT(246/n/9));")?";v
80 LET v=INT v: IF v<1 THEN GO TO 70
90 IF v>246/n/9 THEN LET n=9: GO TO 50
100 LET L=INT(246/n/v)-1: REM largimea
110 DIM h(v,n)
120 LET b=INT((254-(n*v)*(L+1))/2): LET max=0: LET
    min=0
130 FOR j=1 TO n: FOR k=1 TO v
140 INPUT "Variabila ";(j)",valoarea ";(k);
    " = "h(k,i)
150 LET max=(max+h(k,j)+ABS(max-h(k,j)))/2
160 LET min=(min+h(k,j)-ABS(min-h(k,j)))/2
170 NEXT j: NEXT i
175 PRINT # 1; AT 1,9;"CORECT (d/n) ? ": PAUSE 0

```

```

176 GO TO 177*(INKEY$="d")+100*(INKEY$="n")
177 CLS ; PLOT 0,0: DRAW 255,0: DRAW 0,175: DRAW
    -255,0: DRAW 0,-175
180 IF min=max AND min=0 THEN LET max=145
190 IF min=max AND min<>0 THEN LET min=0
200 FOR j=1 TO v: FOR k=1 TO n
210 LET h(j,k)=INT ((h(j,k)-min)*145/(max-min))
220 FOR x=b TO b+L STEP k
230 FOR j=20 TO 20+h(j,k) STEP k
240 PLOT x,y
250 NEXT y
260 NEXT x
270 PLOT b,20: DRAW 0,h(j,k): DRAW L+1,0: DRAW
    0,-h(j,k): DRAW -L-1,0
280 LET b=b+L+1
290 NEXT k
300 NEXT j
310 INPUT AT 0,0;"Ce nume vrei sa dai graficului?":
    LINE n$
320 PRINT OVER 1; AT 20,(32-LEN n$)/2;n$
330 PRINT # 1; AT 1,7;"ALT GRAFIC (d/n) ?": PAUSE 0
340 GO TO 10*(INKEY$="d")+350*(INKEY$="n")
350 CLS : STOP

```

O reprezentare grafică *spațială* a histogramei este realizată de programul următor.

Exemplu 14.41

```

1 REM histograma spatiaala
2 BORDER 4: PAPER 5: INK 1: CLS : INPUT
    "Introduceti nr.de clase c <=12",c
3 IF c<1 OR c>12 THEN GO TO 2
4 DIM f(c): FOR i=1 TO c: INPUT "Introduceti
    frecventa claselor",f(i)
5 IF f(i)>100 OR f(i)<0 THEN PRINT AT 11,11;
    "INCORECT": PAUSE 30: CLS : GO TO 4
8 CLS : FOR i=1 TO c: PRINT AT i,2;"f("i;")=";

```



```

f(i): NEXT i: PRINT #0; AT 0,8;"CORECT (d/n)?" :
PAUSE 0
9 GO TO 10*(INKEY$="d")+2*(INKEY$="n")
10 CLS
11 LET a=INT(255/c)
12 OVER 0: FOR i=1 TO c
20 FOR n=1 TO f(i) STEP 1.2
30 PLOT (i*a)+5-(a+1),(10+n): PRINT AT 21,(i*a/8)-
a/8: PAPER 6; INK 2;f(i): DRAW 10,(n-n)*f(i):
DRAW 10,(n-n)*f(i)+30: NEXT n
40 DRAW -10,(n-n)*f(i): DRAW -10,(n-n)*f(i)-30
50 NEXT a
60 PRINT # 1; AT 1,9:"RELUATI (d/n)?" : PAUSE 0
70 GO TO 2*(INKEY$="d")+80*(INKEY$="n")
80 CLS : STOP

```

În linia 30 poate fi făcută modificarea; ... PRINT AT 21, (i*a/8)-a/8; PAPER (5+INT(RND*2)): INK INT(RND*4)+1;f(i): DRAW etc.

De pildă: a) c=7 clase cu frecvențele 4; 80; 100; 90 50; 30; 20;
b) c=9 clase cu frecvențele 20; 40; 60; 80; 100; 90; 70;
50; 30.

14.11. LEGI PROBABILISTICE

Exemplul 14.42 . Legea geometrică

Un eveniment se poate realiza cu o probabilitate $q < 1$ în timpul unei încercări (experiment). Probabilitatea p_1 de a obține acest eveniment numai la încercarea (repetarea) n este dată de relația:

$$p_1 = q(1-q)^{n-1}$$

Probabilitatea p_2 de a se realiza acest eveniment numai o dată în n repetări este: $p_2 = \sum_1^n q(1-q)^{i-1}$

Primul program calculează probabilitatea p_1 , iar al doilea probabilitatea p_2

```

10. CLS : INPUT "Introduceti probabilitatea q de
    realizare a unui eveniment",q
15 IF q<0 OR q >= 1 THEN GO TO 10
20 INPUT "Introduceti nr.de incercari n",n
25 IF n<1 THEN GO TO 20
30 PRINT AT 11,0;"Probabilitatea realizarii
    eveni-mentului la a ";n;" incercare este ";AT
    14,8;"p1=";q*((1-q)^(n-1))
40 PRINT AT 21,7;"ALTE VALORI (d/n)?" : PAUSE 0
50 GO TO 10*(INKEY$="d")+60*(INKEY$="n")
60 CLS : STOP

```

Pentru: a) $q=0,5$ și $n=5$ rezultă $p_1=0,03125$
 b) $q=0,999$ și $n=2$ se obține $p_2=0,00099900017$

```

8 REM legea geometrica (p2)
10 CLS : INPUT "Introduceti probabilitatea q de
    realizare a unui eveniment",q
15 IF q<0 OR q >= 1 THEN GO TO 10
20 INPUT "Introduceti nr.incercarii n",n
25 IF n<1 THEN GO TO 20
30 CLS : PRINT AT 11,0;"Probabilitatea realizarii
    eveni-mentului la a ";n;" incercare este ":";AT
    14,8;"p2=";1-(1-q)^(n-1)
40 PRINT AT 21,7;"ALTE VALORI (d/n)?" : PAUSE 0
50 GO TO 10*(INKEY$="d")+60*(INKEY$="n")
60 CLS : STOP

```

Pentru: a) $q=0,5$ și $n=5$ rezultă $p_2=0,96875$
 b) $q=0,999$ și $n=2$ se obține $p_2=0,999999$

Exemplul 14.43 : legea binomială

Fie un eveniment care se poate realiza cu o probabilitate $q < 1$. Probabilitatea p_1 de a se realiza acest eveniment de k ori în n încercări este

$$p_1 = C_n^k q^k (1-q)^{n-k} \quad \text{unde } C_n^k = n! / (k!(n-k)!)$$

Probabilitatea p_2 de a vedea realizat acest eveniment de k ori în plus în n încercări este dată de relația

$$p_2 = C_n^k q^k (1-q)^{n-k}$$

Exemplul 14.44 : conține programul de calcul al lui p_2 .

```

8 REM legea binomiala (p1)
10 CLS : INPUT "Introduceti probabilitatea q de
  realizare a evenimentului",q
15 IF q<0 OR q >= 1 THEN GO TO 10
20 INPUT "Introduceti nr. `n` de repetari a(1sp)
  evenimentului",n
25 IF n<1 THEN GO TO 20
26 INPUT "Introduceti nr.de repetitii k(4sp)unde
  doriti sa aiba loc eveni-(2sp)mentul" `k
27 IF k<0 THEN GO TO 26
28 PRINT AT 0,0;"Probabilitatea q=";q; AT 1,0;"Nr.
  de repetari n=" ;n; AT 2,0;"Nr.repetari(4sp)k=";
  k; AT 21,9;"CORECT (d/n)?" : PAUSE 0
29 GO TO 30*(INKEY$="d")+10*(INKEY$="n")
30 LET p=(1-q) ^ n
50 IF k=0 THEN GO TO 90
60 FOR j=1 TO k
70 LET p=p*p*(n-j+1)/j(1-q)
80 NEXT j
90 CLS : PRINT AT 11,0;"Probabilitatea ca
  evenimentul sa se realizeze de ";k;" ori in
  timpul a ";n;" repetari are valoarea " ` ` "
  TAB 8;"p1=";p
100 PRINT AT 21,7; INVERSE 1;"ALTE VALORI (d/n)?" :
  BEEP 1/30,40: PAUSE 4: BEEP 1/20,-20: PAUSE 0
110 GO TO 10*(INKEY$="d")+120*(INKEY$="n")
120 CLS : STOP

```

Pentru: a) $q=0,5$; $n=10$ și $k=2$ rezultă $p_1=0,043945312$

b) $q=0,2$; $n=15$ și $k=3$ rezultă $p_1=0,2501389$

Exemplul 14.44

```

8 REM legea binominala (p2)
10 CLS : INPUT "Introduceti probabilitatea q de
realizare a evenimentului ",q
12 IF q<0 OR q >= 1 THEN GO TO 10
20 INPUT "Introduceti nr. `n` de repetari a(1sp)
evenimentului " ` n
22 IF n<1 THEN GO TO 20
25 INPUT "Introduceti nr.repetarii k(5sp)unde
doriti sa aiba loc eveni-mentul " ` k
26 IF k<0 THEN GO TO 25
27 PRINT AT 0,0;"Probabilitatea q=";q; AT 1,0;"Nr.
de repetari n="; AT 2,0;"Nr.repetari(4sp)k=";k;
AT 21,9;"CORECT (d/n)?" : PAUSE 0
28 GO TO 30*(INKEY$="d")+10*(INKEY$="n")
30 LET p=(1-q)↑n
40 FOR i=1 to K: LET c=(1-q)↑n
50 FOR j=1 TO i: LET c=c*q*(n-j+1)/(1-q)/j
60 NEXT j
70 LET p=p+c
80 NEXT i
90 CLS : PRINT AT 11,0;"Probabilitatea ca
evenimentul sa se realizeze de ";k;" ori in
timpul a ";n;" repetari este p2=";p
100 PRINT AT 21,7; INVERSE 1;"ALTE VALORI (d/n)?" :
BEEP .02,40: PAUSE 4: BEEP .02,-20: PAUSE 0
110 GO TO 10*(INKEY$="d")+120*(INKEY$="n")
120 CLS : STOP

```

Pentru : a) $q=0,1$, $n=10$ și $k=5$ rezultă $p_2=0,9998531$

b) $q=0,4$, $n=5$ și $k=3$ rezultă $p_2=0,91296$

Exemplul 14.45 : legea lui Poisson

Fie un eveniment care într-un interval de timp poate surveni de $0,1,2,\dots,n$ ori. Presupunind că acest eveniment are loc cu probabilitate a în acest interval de timp, se poate determina probabilitatea p_1 ca

evenimentul să se realizeze de n ori. Relația de calcul este:

$$p_1 = a^n e^{-a} / n!$$

Probabilitatea de a realiza evenimentul de n ori în plus este:

$$p_2 = \sum_0^n \frac{a^j e^{-a}}{j!}$$

Primul program calculează p_1 iar programul 14.46 determină p_2 .

```

8 REM legea lui Poisson (p1)
10 CLS ; INPUT "Introduceti nr.de repetari n",n
15 IF n<0 THEN GO TO 10
20 INPUT "Introduceti probabilitatea a",a
25 IF a<1 OR a >= 1 THEN GO TO 20
30 PRINT AT 0,0;"Nr.de repetari n=";n;AT 1,0
   "Probabilitatea a=";a
40 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
50 GO TO 60*(INKEY$="d")+10*(INKEY$="n")
60 LET p=EXP(-a)
70 IF n=0 THEN GO TO 100
80 FOR j=1 TO n: LET p=p*a/j: NEXT j
100 CLS : PRINT AT 11,0;"Probabilitatea ca intr-un
   expe- riment repetat de n=";n;" ori sa se
   produca un eveniment `dorit` cu probabilitatea
   a=";a;" este deci " `` TAB 8;"p1=";p
110 PRINT AT 21,7;"ALTE VALORI (d/n)?" : PAUSE 0
120 GO TO 10*(INKEY$="d")+130*(INKEY$="n")
130 CLS : STOP

```

Pentru $n=1$ și $a=0,9$ rezultă $p_1=0,36591269$

Exemplul 14.46

```

8 REM legea lui Poisson (p2)
10 CLS : INPUT "Introduceti nr.de repetari n",n
15 IF n<0 THEN GO TO 10
20 INPUT "Introduceti probabilitatea a",a
25 IF a<0 OR a >= 1 THEN GO TO 20
30 PRINT AT 0,1;"Nr.de repetari n=";n;AT 1,0;

```

```

"Probabilitatea a=";a;AT 21,9;"CORECT (d/n) ?":
PAUSE 0
40 GO TO 60*(INKEY$="d")+10*(INKEY$="n")
60 LET p=EXP(-a)
70 IF n=0 THEN GO TO 100
75 LET c=p
77 FOR j=1 TO n: LET c=c*a/j
80 LET p=p+c
90 NEXT j
100 CLS : PRINT AT 11,0;"Probabilitatea ca intr-un
    expe- riment repetat de n=";n;" ori in timpul t,
    sa se produca un eve- niment dorit cu
    probabilitatea a=";" este p2=";p
110 PRINT AT 21,7;"ALTE VALORI (d/n)?" : PAUSE 0
120 GO TO 10*(INKEY$="d")+130*(INKEY$="n")
130 CLS : STOP

```

Pentru: a) $n=5$ și $a=0,999$ rezultă $p_2=0,99940888$
 b) $n=3$ și $a=0,7$ rezultă $p_2=0,99424654$

Exemplul 14.47 : legea normală

Fie x o variabilă aleatoare distribuită urmînd o lege normală.

Dacă a reprezintă un număr real oarecare, probabilitatea p de a avea $x \leq a$ este numărul dat de expresia:

$$p = \frac{1}{s\sqrt{2\pi}} \int_{-\infty}^a e^{-\frac{(x-m/s)^2}{2}} dx \quad (m \text{ -media; } s \text{ -abaterea medie pătratică})$$

Dacă $m=0$ și $s=1$ legea se numește centrală și redusă. Punînd $t=sx+m$ se obține

$$p = \frac{1}{\sqrt{2\pi}} \int_0^{a-m} e^{(-t^2/2)} dt$$

Întrucît $\frac{1}{\sqrt{2\pi}} \int_{-\infty}^0 e^{(-t^2)/2} dt = 0,5$ se obține $p = 0,5 + \frac{1}{\sqrt{2\pi}} \int_0^{a-m} e^{(-t^2)/2} dt$

Fie $z=(a-m)/s$; după dezvoltarea în serie a termenului $e^{(-t^2)/2}$ și integrarea termen cu termen se obține:

$$p = 0,5 + \frac{1}{\sqrt{2\pi}} \sum_0^{\infty} (-1)^j \frac{z^{2j+1}}{(2j+1) j! 2^j}$$

Notînd $C_j = (-1)^j \frac{z^{2j+1}}{(2j+1) j! 2^j}$ se poate scrie pentru $j \geq 1$:

$$C_j = -C_{j-1} \frac{2j-1}{2j(2j+1)} z^2 \text{ cu } C_0 = \frac{1}{\sqrt{2\pi}} z$$

```

8 REM legea normala (p)
10 CLS : INPUT "Introduceti media m",m
20 INPUT "Introduceti abaterea medie pa-(2sp)
   tratica s",s
30 INPUT "Introduceti nr.a",a
40 PRINT AT 0,0;"Media(21sp)m=";m; AT 1,0""Abaterea
   medie patratice s=";s; AT 2,0;"Numarul(19sp)
   a=";a
50 PRINT AT 21,9;"CORECT (d/n) ?": PAUSE 0
52 GO TO 53*(INKEY$="d")+10*(INKEY$="n")
53 LET z=(a-m)/s
54 IF ABS(z)>4 THEN GO TO 160
55 LET n=10
56 IF ABS(z) >= 2 THEN LET n=25: GO TO 80
80 LET c=z
90 LET t=z
100 FOR j=1 TO n
110 LET c=-c*(2*j-1)*z*z/(2*j*(2*j+1))
120 LET t=t+c
130 NEXT j
140 LET p=0.5+t/SOR(2*PI)
150 GO TO 180
160 IF SGN(z)=-1 THEN LET p=0
170 IF SGN(z)=1 THEN LET p=1
180 CLS : PRINT AT 11,6;"Probabilitatea este   :";
   AT 13,10;"p=";p
190 PRINT AT 21,7; PAPER 5; INK 1;"ALTE VALORI
   (d/n)?" : FOR x=1 TO 6: BEEP .05,10: BEEP .05,13:
   NEXT x: PAUSE 0

```

```

200 GO TO 10*(INKEY$="d")+210*(INKEY$="n")
210 CLS : STOP

```

- Pentru: a) $m=10$, $s=2$ și $a=10$ rezultă $p=0,5$
 b) $m=0$, $s=1$ și $a=2$ rezultă $p=0,97724987$
 c) $m=25$, $s=5$ și $a=18$ rezultă $p=080756659$

Exemplul 14.48 : problema inversă și anume: fie p un număr cuprins între 0,0001 și 0,9999; pentru ce valoare a , probabilitatea de a avea $x \leq a$ este p ?

```

8 REM legea normala (problema inversa)
10 CLS : INPUT "Introduceti media m",m: INPUT
  "Introduceti abaterea medie pa-(2sp)tratica s",
  s: INPUT "Introduceti probabilitatea p",p
15 PRINT AT 0,0;"Media(21sp)m=";m; AT 1,0;"Abaterea
  medie patratice s="s; AT 2,0 ;"Probabilitatea
  (11sp)p=";p; AT 21,9;"CORECT (d/n) ?": PAUSE 0
30 GO TO 40*(INKEY$="d")+10*(INKEY$="n")
40 CLS : LET h=-4: LET b=4
60 FOR i=1 TO 15
65 REM calculul probabilitatii q de a avea x<=z
70 LET z=(h+b)/2
80 LET n=10
90 IF ABS(z) >= 2 THEN LET n=25
100 LET c=z
110 LET t=z
120 FOR j=1 TO n: LET x=-c*(2*j-1)*z*z/(2*j*
  (2*j+1)): LET t=t+c: NEXT j
160 LET q=0.5+t/SQR(2*PI)
165 REM bucla noua?
170 IF ABS(p-q) <= 10^(-6) THEN GO TO 220
180 IF q>p THEN LET b=z
190 IF q<p THEN LET h=z
200 NEXT i
205 REM editarea rezultatului
210 CLS : LET a=(z*s)+m
220 PRINT AT 11,0;"Numarul are valoarea a="; SGN

```



```

a*INT(1000*ABS a)/1000
230 PRINT AT 21,7; PAPER 1; INK 6;"ALTE VALORI
(d/n)?" :BEEP .1,50: BEEP .1,40: BEEP .1,30: BEEP
.1,28: BEEP .1,10: BEEP .1,0: PAUSE 0
240 GO TO 10*(INKEY$="d")+250*(INKEY$="n")
250 CLS : STOP

```

Programul se bazează pe faptul că $p=0$ pentru $x < m-4s$ și $p=1$ pentru $x > m+4s$; prin urmare este suficient să se calculeze numărul a între aceste două valori prin dihotomie.

Pentru: a) $m=0, s=1, p=0,97724987$ rezultă $a=2$

b) $m=2, s=5, p=0,75$ rezultă $a=5,372$

c) $m=25, s=5, p=0,08756659$ rezultă $a=17,989$

Exemplul 14.49 : legea χ^2 (hi pătrat)

Fie $x \geq 0$ o variabilă aleatoare; dacă această variabilă este distribuită urmînd legea χ^2 cu n grade de libertate, probabilitatea p de a avea $x \geq a$ este dată de relația

$$p = \frac{1}{2^{n/2} \Gamma(n/2)} \int_0^a e^{-x/2} \cdot x^{n/2-1} dx$$

unde: $n \geq 1$, Γ semnifică funcția gamma definită pentru $x \geq 0$ prin expresia:

$$\Gamma(x) = \int_0^{\infty} e^{-t} \cdot t^{x-1} dt$$

Se demonstrează că $\Gamma(n/2) = (\frac{n}{2}-1)!$ pentru n par, respectiv

$$\Gamma(n/2) = \frac{n-2}{2} \cdot \frac{n-4}{2} \dots \frac{5}{2} \cdot \frac{3}{2} \cdot \Gamma(1/2) \text{ pentru } n \text{ impar și } n \geq 3, \text{ cu}$$

$$\Gamma(1/2) = \sqrt{\pi}$$

De asemenea se demonstrează că:

$$p = \left(\frac{a}{2}\right)^{n/2} \cdot \frac{e^{-a/2}}{\Gamma(n/2)} \left[1 + \sum_{k=1}^{\infty} \frac{e^k}{(n+2)(n+4)\dots(n+2k)} \right]$$

Se pot folosi logaritmi pentru a calcula pe $\Gamma(n/2)$. Se notează:

$$C_k = a^k / (n+2)(n+4)\dots(n+2k)$$

și atunci se obține $C_k = C_{k-1}a / (n+2k)$ pentru $k \geq 2$, cu $C_1 = a / (n+2)$.

În program variabila k variază de la 1 la 1000, dar programul se va opri când $C_k \geq 10^{-8}$.

```

8 REM legea hi patrat (p)
10 CLS : INPUT "Introduceti nr.gradelor de li-(2sp)
bertate n",n: INPUT "Introduceti nr. a",a: PRINT
AT 0,0:"Nr.gradelor de libertate n =";n: AT 1,0;
"Numarul(20sp)a=";a; AT 21,9;"CORECT (d/n) ?":
PAUSE 0
25 GO TO 30*(INKEY$="d")+10*(INKEY$="n")
30 CLS : LET m=n/2
40 IF m=INT(m) THEN GO TO 110
45 REM calculul lui L=LN gamma (n/2) pt.n impar
50 LET L=(LN(PI))/2
60 IF n=1 THEN GO TO 160
70 FOR i=1 TO n-2 STEP 2: LET L=L+LN(i)-LN(2):
NEXT i
100 GO TO 160
105 REM calculul lui LN gama (n/2) pt.n par
110 LET L=0
120 IF n=2 THEN GO TO 160
130 FOR i=1 TO m-1 LET L=L+LN(i): NEXT i
155 REM calculul sumei s de la k=1 la k=...
160 LET c=1
170 LET s=1
180 FOR k=1 TO 1000
190 LET c=c*a(n+2*k)
200 LET s=s+c
210 IF c <=10^(-8) THEN GO TO 230
220 NEXT k
225 REM editarea rezultatului
230 CLS : PRINT AT 11,0;"Probabilitatea are valoarea
:"AT 13,7;"p=";s*EXP(m*LN(a/2)-(a/2)-L-LN(m));
AT 21,7; INVERSE 1;"ALTE VALORI (d/n)?" : FOR w=

```

```

-40 TO 50 STEP 11: BEEP .9,w: NEXT w: FOR r=20
TO -20: BEEP .05,r: NEXT r: PAUSE 0
250 GO TO 10*(INKEY$="d")+260*(INKEY$="n")
260 CLS : STOP

```

- Pentru: a) $n=25$, $a=10$ rezultă $p=0,0033473589$
 b) $n=4$, $a=5$ rezultă $p=0,7127025$
 c) $n=100$, $a=82$ rezultă $p=0,095025019$

Exemplul 14.50 : problema inversă și anume: fie p un număr cuprins între 0 și 0,999; pentru care valoarea lui a probabilitatea de a avea $x < a$ este p ? Programul care urmează se bazează pe observația că media și abaterea medie pătratică a distribuției sînt respectiv n și $\sqrt{2n}$; prin urmare va rezulta practic $p=1$ pentru $x > n+8\sqrt{2n}$. Cum $p=0$ pentru $a=0$ este suficient să se determine a prin dihotomie între 0 și $n+8\sqrt{2n}$.

```

8 REM legea hi patrat (problema inversa)
10 CLS : INPUT "Introduceti nr.gradelor de li-(2sp)
bertate n",n: INPUT "Introduceti probabilitatea
p",p: IF p>0.999 OR n<1 THEN GO TO 10
15 PRINT AT 0,0:"Nr.gradelor de libertate n=";n; AT
1,0;"Probabilitatea p=";p; AT 21,9:"CORECT
(d/n) ?": PAUSE 0
25 GO TO 30*(INKEY$="d")+10*(INKEY$="n")
30 PRINT AT 21,9;"(14sp)"; AT 11,7: PAPER 5;INK
1;"Asteptati va rog !": LET h=0
40 LET b=n+8*SQR(2*n)
55 REM calculul probabilitatii de a avea x<=a
60 LET m=n/2
70 IF m=INT(m) THEN GO TO 140
80 LET L=(LN(PI/2))/2
90 IF n=1 THEN GO TO 190
100 FOR i=1 TO n-2 STEP 2
110 LET L=L+LN(i)-LN(2)
120 NEXT i
130 GO TO 190
140 LET L=0

```

```
150 IF n=2 THEN GO TO 190
160 FOR i=1 TO n-1
170 LET L=L+LN(i)
180 NEXT i
190 LET a=(b+h)/2
200 LET c=1
210 LET s=1
220 FOR k=1 TO 1000
222 LET c=c*a/(n+2*k)
224 LET s=s+c
225 REM editarea rezultatului
226 IF c <= 10↑(-8) THEN GO TO 229
227 NEXT k
229 LET q=s*EXP(m*LN(a/2)-(a/2)-L-LN(m))
230 REM bucla noua sau editarea rezultatului
240 IF ABS (p-q) <= 10↑(-6) THEN GO TO 320
250 IF p<q THEN LET b=a
260 IF p>q THEN LET h=a
270 GO TO 190
320 CLS : PRINT AT 11,0;"Numarul are valoarea :"; AT
    13,7;"a=";INT(1000*a)/1000
330 PRINT AT 21,7; INVERSE 1; FLASH 1; "ALTE VALORI
    (d/n)?:FOR w=1 TO 1: BEEP .0005,w+40: NEXT w:
    PAUSE 0
340 GO TO 10*(INKEY$="d")+350(INKEY$="n")
350 CLS : STOP
```

Pentru: a) $n=5, p=0,1$ rezultă $a=1,353$
b) $n=10, p=0,5$ rezultă $a=9,341$
c) $n=30, p=0,005$ rezultă $a=13,786$

15. APLICAȚII DE INTERES GENERAL

Prin programe de interes general se înțeleg produsele software care nu sînt strict legate de o anumită profesiune sau un anumit domeniu de cunoaștere. Ele oferă soluții care interesează pe orice posesor de calculatoare personale. Din această categorie se prezintă 5 programe comentate în cele ce urmează .

15.1. DIRECTORY

Sînt dese situațiile cînd un utilizator primește o casetă înregistrată cu programe pe calculator, care însă nu este însoțită de o fișă în care să fie consemnate titlurile programelor, blocurile componente cu adresele și lungimile lor, etc. Programul "directory" realizează acest deziderat.

Modul său de utilizare este foarte simplu și anume :

- se încarcă în memoria calculatorului programul "directory";
- se pornește caseta cu programe și se lasă a se derula pînă la sfîrșitul ei.

Pe ecranul TV se vor afișa informațiile dorite pentru orice bloc intilnit pe casetă.

```

8 REM directory
10 CLEAR 26895: BORDER 0: PAPER 0: INK 7: CLS
20 DATA "55", "62", "0", "221", "33", "120", "105", "17",
  "17", "0", "205", "86", "5", "201"
30 FOR a=26986 TO 26999: READ a$: POKE a, VAL a$:

```

```

NEXT a
40 DEF FN p(a): PEEK a+256*PEEK (a+1)
50 DEF FN a(p,c)=27000+17*(c-1)+170*(p-1)
60 LET p=1: LET c=1: LET g=300: POKE 27000,5
80 GOSUB 500
150 RANDOMIZE USR 26986: GOSUB 780: IF PEEK FN
a(p,c)=5 THEN GO TO 150
160 GOSUB 600
170 GOSUB 800
180 IF c<10 THEN LET c=c+1: LET et=150: GO TO 200
190 IF c=10 THEN LET p=p+1: LET c=1: LET et=80:
GO TO 200
195 STOP
200 POKE FN a(p,c),5: POKE 26992,INT (FN a(p,c)/256)
: POKE 26991, FN a(p,c)-256*INT (FN a(p,c)/256)
210 GO TO et
300 LET c=c-1: IF c=0 THEN LET p=p-1: LET c=10
310 IF p=0 THEN LET p=1: LET c=1: GOSUB 550:
GO TO 150
320 GO TO 170
500 REM titluri
520 CLS : PRINT PAPER 1;"DIRECTORY(7sp)M.M.
POPOVICI"
530 PLOT 0,165: DRAW INK 2;255,0
540 PLOT 0,166: DRAW INK 2;255,0
550 INPUT ;:PRINT # 1; PAPER 1; TAB 25;"PAGE ";p;
(" " AND p < 10)
560 RETURN
600 REM citirea
610 LET ad=FN a(p,c)
620 LET t=PEEK ad
630 DIM n$(10)
640 FOR i=1 TO 10: LET n$(i)=CHR$ PEEK (ad+i):
NEXT i
650 LET L=FN p(ad+11)
660 LET s=FN p(ad+13)
670 LET X=FN p(ad+15)
680 GOSUB 700
690 GO TO 710
```

```
700 PRINT "Prog. ": ; RETURN
701 PRINT "Num.A ": ; RETURN
702 PRINT "Chr.A "; : RETURN
703 PRINT "Bytes "; : RETURN
710 PRINT PAPER 1;n$; PAPER 0;" ";
720 IF t=3 THEN PRINT s;",";L; : GO TO 770
730 PRINT L;
740 IF t>0 THEN GO TO 770
750 PRINT "/" ;X;
760 IF a >= 0 AND s<10000 THEN PRINT "/" ;s;
770 PRINT ` `: RETURN
780 BEEP .1,-16: BEEP .1,32: BEEP .1,31: BEEP .1,48:
RETURN
805 INPUT ; : PRINT #1; FLASH 1;" "; FLASH 0; PAPER
1; TAB 25;"PAGE ";p;(" " AND p<10)
810 FOR w=1 TO 180
820 IF INKEY$="" THEN GO TO 1000
830 LET a$=INKEY$: GOSUB 780
840 LET cc=c: LET pp=p
850 INPUT ; : PRINT # 1; FLASH 1;" "; FLASH 0; PAPER
1;"(1sp)Up(2sp)Top(2sp)Down(2sp)Quit"; TAB 25;
"PAGE "; ;p; (" "AND p<10)
860 IF INKEY$="" THEN GO TO 860
870 LET a$=INKEY$: GOSUB 780
880 IF a$="q" THEN LET p=pp: GOSUB 500: FOR c=1 TO
cc: GOSUB 600: NEXT c: LET c=cc: RETURN
890 IF a$="u" THEN LET p=p-1: GO TO 920
900 IF a$="t" THEN LET p=1: GOSUB 920
910 LET p=p+1
920 IF p<1 THEN LET p=1
930 IF p>pp THEN LET p=pp
840 GOSUB 500
950 FOR c=1 TO (10 AND p<pp)+cc AND p=pp)
960 GOSUB 600
970 NEXT c: GO TO 850
1000 NEXT w
1010 GOSUB 550
1020 RETURN
9999 STOP
```

15.2. AGENDA TELEFONICĂ

Acest program este conceput pentru :

- crearea unei agende telefonice noi;
- introducerea de noi nume;
- afișarea de nume la cerere ;
- listarea alfabetică a numelor ;
- ștergerea memoriei.

Modificînd denumirile în mod corespunzător se poate transforma într-un alt afișier cu altă destinație (de ex.: realizarea unei programoteci de programe stocate pe diverse casete).

```

4 REM agenda telefonica
5 POKE 23658,8
10 BORDER 5: PAPER 5: INK 0: CLS
15 GO TO 360
20 DIM d$(100,32)
30 FOR d=1 TO 100
40 INPUT PAPER 6; INK 1; BRIGHT 1;"NUMELE SI
   PRENUMELE:" `LINE b$
90 PRINT PAPER 7; INK 2; BRIGHT 1; AT 0,0;"NUMELE
   ESTE:";INK 3 ` b$
110 INPUT INK 1;BRIGHT 1;"(3sp)TELEFONUL si
   PROFESIUNEA !" ` LINE c$
120 CLS
130 PRINT INK 2; AT 0,0;b$;" "; INK 1;c$
140 PRINT INK 3;`` "TASTATI:" ; INVERSE 1; PAPER 7;
   " ENTER "; INVERSE 0; PAPER 6;"DACA ESTE
   O.K.!!!" ` PAPER 7; ` "TASTATI " ; INVERSE 1;
   "**E**&ENTER"; INVERSE 0;"pt. CORECTARE"
150 INPUT LINE e$: CLS
160 IF e$ < > "" THEN GO TO 40
170 LET d$(d)=b$+" "+c$
180 PRINT INK 3;``"TASTATI :"; INVERSE 1; PAPER 7;
   BRIGHT 1;" ENTER "; PAPER 6; INVERSE 0;"PT. NOUL
   NUME,"; PAPER 7; INK 1; BRIGHT 0;` "SAU ORICE

```



```
TASTA PENTRU SORTARE !!"
190 PAUSE 0 : CLS
200 IF INKEY$="" THEN NEXT d
210 PRINT PAPER 7; BRIGHT 1; INK 4;"A POST SORTAT!":
    PAUSE 100
215 POKE 23692,0
220 LET b=0
230 LET g=d
240 LET z=1
250 LET b=z+1
260 IF b>g THEN GO TO 330
270 IF d$(b)>d$(z) THEN GO TO 290
280 LET z=z+1: GO TO 250
290 LET q$=d$(z)
300 LET d$(z)=d$(b)
310 LET d$(b)=q$
320 GO TO 280
330 PRINT INK 2;d$(g)
340 LET g=g-1
350 IF g>0 THEN GO TO 240
360 CLS : PRINT AT 1,5; PAPER 7; INK 3; FLASH 1;
    "CONTINUTUL PROGRAMULUI"
370 PRINT INK 2; BRIGHT 1;` "1-CREAREA UNEI AGENDE
    TELEFONICE(2sp)NOI"
380 PRINT INK 1;`"2-INTRODUCEREA DE NOI NUME"
390 PRINT INK 3;`"3-AFISARE DE NUME LA CERERE"
400 PRINT INK 2;`"4-"; PAPER 7;"SAVE"; PAPER 6;
    " PROGRAM"
405 PRINT INK 0;`"5-LISTAREA ALFABETICA A NUMELOR"
410 PRINT INK 2;`"6-STOP!"; INK 1; BRIGHT 1;"PENTRU
    CONTINUARE TASTATI"; BRIGHT 0; TAB 10;
    "< GO TO 360 >"
417 PRINT INK 0;`"7-"; PAPER 7;"STERGEREA MEMORIEI"
418 PRINT INK 0;`"8-INSTRUCTIUNI"
419 PRINT AT 20,6;"(APASATI 1... 8)"
420 PAUSE 0: CLS
422 IF INKEY$="8" THEN GO TO 4800
425 IF INKEY$="7" THEN GO TO 600
430 IF INKEY$="1" THEN GO TO 200
```

```
440 IF INKEY$="2" THEN NEXT d
450 IF INKEY$="3" THEN GO TO 500
460 IF INKEY$="4" THEN GO TO 998
462 IF INKEY$="6" THEN CLS:STOP
465 IF INKEY$="5" THEN FOR a=d TO 1 STEP -1:PRINT
    INK 2;d$(a): NEXT a
466 PRINT INVERSE 1;"(7sp)SFIRSITUL LISTEI(7sp)":
    PAUSE 0: CLS : GO TO 360
500 CLS : PRINT INK 1; BRIGHT 1;``"INTRODU NUMELE
    SI PRENUMELE"
510 INPUT LINE a$: LET f=LEN a$
520 CLS : PRINT AT 0,0;"NUMELE CERUT :",a$
530 FOR a=1 TO d
540 IF d$(a)(TO f)=a$ THEN PRINT INK 2;`d$(a)
    (f+1 TO):PAUSE 0:CLS:GO TO 360
550 NEXT a
560 PRINT AT 21,0; INK 2; FLASH 1; BRIGHT 1;"ACEST
    NUME NU ESTE IN MEMORIE !": PAUSE 0
570 GO TO 360
600 CLS : PRINT INK 2: BRIGHT 1;`` "NUME DE
    MODIFICAT?"
610 INPUT LINE a$: LET f=LEN a$
620 PRINT PAPER 7; INK 2;``"NUMELE : "` a$
630 FOR a=1 TO d
640 IF d$(a)(TO f)=a$ THEN PRINT INK 2;``d$(a)
    (f+1 TO): GO TO 657
650 NEXT a
655 PRINT AT 21,0: INK 2; FLASH 1; PAPER ;BRIGHT 1;
    "ACEST NUME NU ESTE IN MEMORIE!": PAUSE 0:
    GO TO 360
657 PRINT PAPER 7; INK 3; FLASH 1;"SINTETI GATA ?";
    FLASH 0;" D/N"
658 PAUSE 0: IF INKEY$="d" OR INKEY$="D" THEN GO TO
    660
659 GO TO 360
660 LET d$(a)=""
670 GO TO 360
1000 STOP
4800 CLS
```

```


4820 PRINT AT 4,0; PAPER 3; INK 6;"(3sp)
      I N S T R U C T I U N I(4sp)"
4840 PRINT AT 7,0; INK 2; PAPER 6;"(3sp)Acest program
      este o "" AGENDA TELEFONICA""(23sp)Pentru
      apelare se tasteaza numele si prenumele
      dorit(10sp)" ` "(3sp)Daca din greseala se
      tasteaza ""6"" la OPTIUNI sau se iese din(2sp)
      program, nu se apasa RUN, ci(4sp)"; AT 15,9;
      INVERSE 1; FLASH 1;"(1sp)< GO TO 360 >(1sp)"
4860 PRINT AT 17,7; PAPER 4; INK 7; FLASH 1;"APASATI
      ORICE TASTA":PAUSE 0: CLS
4880 PRINT AT 7,0; PAPER 6; INK 2;"(4sp)Se pot
      introduce"; PAPER 3; INK 7; FLASH 1; BRIGHT 1;
      "100"; FLASH 0; BRIGHT 0; PAPER 6; INK 2;"
      numere de telefon. Rindul nu trebuie sa
      depaseasca 32 caractere(5sp)"; AT 17,7; PAPER 4;
      INK 7; BRIGHT 1; FLASH 1;"APASA ORICE TASTA :
      PAUSE 0: CLS
4900 CLS : GO TO 360

```

15.3. CALENDAR

Există situații când este necesar să se cunoască *în ce zi a săptămînii cade o anumită dată* (de pildă aniversarea zilei de naștere a unui membru al familiei) sau *o anumită dată în ce zi a săptămînii a avut loc*. De asemenea, uneori este necesar să se cunoască *numărul de săptămîni și zile* - eventual *ani* - care se scurg între două date precizate. În aceste scopuri a fost conceput programul următor care afișează toate aceste valori începînd cu anul 1583 pornind de la faptul că în octombrie 1582 a fost introdus **calendarul gregorian** folosit și astăzi.

```

148 REM calendar
150 CLS
220 PRINT AT 1,5;"(2sp)C A L E N D A R(2sp)"; AT
      2,5;"19 taste ": PAUSE 5: REM tasta 3 in modul
      grafic
230 DIM c(24): DIM M$(12,12): DIM S$(7,9): DIM

```

```

N$(7,12): DIM F$(28,3)
270 RESTORE 280
280 DATA 0,31,59,90,120,151,181,212,243,273,304,334
290 DATA 0,31,60,91,121,152,182,213,244,274,305,335
300 DATA "IANUARIE(2sp)", "FEBRUARIE ", "MARTIE(4sp)",
"APRILIE(3sp)", "MAI(7sp)", "IUNIE(5sp)",
"AUGUST(4sp)", "SEPTEMBRIE", "OCTOMBRIE ",
"NOIEMBRIE ", "DECEMBRIE "
310 DATA "DUMINICA(2sp)", "LUNI(4sp)", "MARTI(5sp)",
"MIERCURI", "JOI(6sp)", "VINERI(3sp)",
"SIMBATA(2sp)"
320 DATA "(2sp)1(2sp)8 15 22", "(2sp)2(2sp)9 16 23",
"(2sp)3(2sp)10 17 24", "(2sp)4(2sp)6 13 20 27",
"(2sp)7 14 21 28"
340 DATA " 29", " 30", " 31", "", "", "", ""
350 DATA " 29", " 30", "", "", "", "", ""
360 DATA "", "", "", "", "", "", ""
370 FOR N=1 TO 24: READ C(N): NEXT N
380 FOR N=1 TO 12: READ M$(N): NEXT N
390 FOR N=1 TO 7: READ S$(N): NEXT N
400 FOR N=1 TO 7: READ N$(N): NEXT N
410 FOR N=1 TO 28: READ F$(N): NEXT N
420 PRINT AT 6,10;"OPTIUNI  :"; AT 10,3;"1)O LUNA
COMPLETA"; AT 12,15:"APASATI <1>"
430 PRINT AT 16,3;"2)TIMPUL INTRE 2 DATE:"; AT
18,15; "APASA <2>"
440 IF INKEY$="1" THEN CLS : GO TO 470
450 IF INKEY$="2" THEN CLS : GO TO 870
460 GO TO 440
470 PRINT : PRINT : FOR N=1 TO 12: PRINT TAB 5;
M$(N); NEXT N
480 INPUT "CE LUNA DORESTI ? (INDICA NR.) ",LUNA :
INPUT " DIN CE AN ? ",AN
490 IF LUNA<1 OR LUNA>12 THEN CLS : PRINT AT 12,0;
"DATE GRESITE! REIA !!!": PAUSE 200: CLS :
GO TO 470
500 IF AN<1 THEN CLS : PRINT AT 8,0 ;"REGRET! NU
SINT PROGRAMAT A CALCULA ANI ANTERIORI EREI
NOASTRE";

```

```

AT 20,6;"APASA O TASTA ....": PAUSE 0: CLS :
GO TO 420
510 BORDER 0: PAPER 0: INK 7: CLS : PRINT AT 0,0:
M$(LUNA); AT 0,25;AN: PRINT "31 taste ": REM
tasta 3 in modul grafic
520 IF AN<1583 THEN PRINT TAB 6;"CALENDARUL IULIAN":
PRINT: GO TO 540
530 PRINT TAB 5;"CALENDARUL GREGORIAN": PRINT
540 LET BIS=0
550 IF AN/4=INT(AN/4) THEN LET BIS=12
560 IF AN>1580 AND AN/100=INT(AN/100) THEN LET
BIS=0
570 IF AN/400=INT(AN/400) THEN LET BIS=12
580 LET z=0
590 IF LUNA=4 OR LUNA=6 OR LUNA =9 OR LUNA=11 THEN
LET z=7
600 IF LUNA=2 AND AN/4 <> INT(AN/4) THEN LET z=14
610 IF LUNA=2 AND AN/4 <> INT(AN/4) THEN LET z=21
620 IF AN > 1580 AND LUNA=2 AND AN/100=INT(AN/100)
THEN LET z=21
630 IF LUNA=2 AND AN/400=INT(AN/400) THEN LET z=14
640 LET D=AN*365+INT((AN-1)/4-INT(AN/100)+
INT(AN/400))
650 LET D=D+C(LUNA+BIS)
660 LET D=INT(D/7)-INT(D/7))*7.1)
670 FOR N=1 TO 7
680 LET W=D+N
690 IF AN=1582 AND LUNA<10 THEN LET W=D+N-4
700 IF AN=1582 AND LUNA=10 THEN CLS : PRINT
"OCTOMBRIE 1582 A FOST INSTAURAT CALENDARUL
GREGORIAN. A AVUT NUMAI 21 ZILE, SARIND DE LA
JOI 5 LA VINERI 15": GO TO 770
710 IF AN<1582 THEN LET W=D+N-4
720 IF W>7 THEN LET W=W-7: IF W>7 THEN GO TO 720
730 IF W<1 THEN LET W=W+7: IF W<1 THEN GO TO 730
740 PRINT TAB 4;S$(W);N$(N);F$(N+z)
750 PRINT TAB 4; INK W:"24 taste ": REM tasta 8 in
modul grafic
760 NEXT N

```

```

770 PRINT AT 19,0;"LUNA ANTERIOARA: A. URMATOAREA:
S.COPIE LA IMPRIMANTA: I.MENU:O.(10sp)
SFIRSIT:T."
780 IF INKEY$="A" OR INKEY$="a" THEN LET LUNA=LUNA-1
790 IF LUNA=0 THEN LET LUNA=12: LET AN=AN-1
800 IF INKEY$="S" OR INKEY$="s" THEN LET LUNA=LUNA+1
810 IF LUNA=13 THEN LET LUNA=1: LET AN=AN+1
820 IF INKEY$="A" OR INKEY$="a" OR INKEY$="S" OR
INKEY$="s" THEN GO TO 510
830 IF INKEY$="O" OR INKEY$="o" THEN BORDER 7: PAPER
7: INK 0: CLS : GO TO 412
840 IF INKEY$="T" OR INKEY$="t" THEN BORDER 7: PAPER
7: INK 0: CLS : STOP
850 IF INKEY$="I" OR INKEY$="i" THEN COPY
860 GO TO 780
870 PRINT TAB 3;"DIFERENTA INTRE DOUA DATE": PRINT
TAB 3;"25 taste ": PRINT : REM tasta 3 in modul
grafic
880 PRINT "AMBELE TREBUIE SA FIE POSTERIOARE
CALENDARULUI GREGORIAN(ANUL 1582)": PAUSE 50
890 PRINT : PRINT "LUNA TREBUIE INTRODUSA PRIN NU-";
"MARUL EI.....": PAUSE 50
900 PRINT : PRINT"ACEASTA PARTE A PROGRAMULUI NU
CONTROLEAZA ERORILE.ZIUA,LUNA SIANUL TREBUIE
INTRODUSE CU MA- RE ATENTIE!!!!"
910 PRINT AT 20,5; FLASH 1;"APASA O TASTA OARECARE":
PAUSE 0
920 CLS : INPUT "PRIMA DATA: ZIUA? ";D1;" LUNA ?
"M1,"ANUL? ";A1
930 INPUT "A DOUA DATA: ZIUA? ";D2;" LUNA? ";M2,
"ANUL ? ";A2
940 IF A1<1583 THEN PRINT AT 8,0;"DATE INCORECTE":
PRINT : PRINT :"AMBELE DATE TREBUIE SA FIE
POS-(1sp)TERIOARE ANULUI 1582": PRINT AT 20,5;
"APASA O TASTA OARECARE": PAUSE 0: CLS :
GO TO 420
950 LET D01=D1: LET D02=D2
960 LET A01=A1: LET A02=A2: LET M01=M1: LET M02=M2
970 LET AT=A2-A1: LET MT=M2-M1

```

```

980 IF MT<0 THEN LET AT=AT-1: LET MT=MT+12
990 LET S1=INT(D1/7): LET S2=INT(D2/7)
1000 LET ST=S2-S1
1010 LET BIS1=0
1020 IF A1/4=INT(A1/4) THEN LET BIS1=12
1030 IF A1/100=INT(A1/100) THEN LET BIS1=0
1040 IF A1/400=INT(A1/400) THEN LET BIS1=12
1050 LET BIS2=0
1060 IF A2/4=INT(A2/4) THEN LET BIS2=12
1070 IF A2/100=INT(A2/100) THEN LET BIS2=0
1080 IF A2/400=INT(A2/400) THEN LET BIS2=12
1090 LET D1=D1+A1*365+INT((A1-1/4)-INT(A1/100)+
    INT(A1/400))
1100 LET D1=D1+C(M1+BIS1)
1110 LET DP1=INT((D1/7)-INT(D1/7))*7.1): IF DP1<1
    THEN LET DP1=DP1+7
1120 LET D2=D2+A2*365+INT((A2-1/4)-INT(A2/100)+
    INT(A2/400))
1130 LET D2=D2+C(M2+BIS2)
1140 LET DP2=INT(((D2/7)-INT(D2/7))*7.1): IF DP2>7
    THEN LET DP2=DP2-7
1150 IF DP2<1 THEN LET DP2=DP2+7
1160 LET DT=D2-D1
1170 LET DPT=DP2-DP1
1180 LET W=INT(DT/7)
1190 IF DPT>7 THEN LET DPT=DPT-7: LET W=W+1
1200 IF DPT<1 THEN LET DPT=DPT-7: LET W=W-1
1210 PRINT "ZIU A DE ";S$(DP1): PRINT ;D01;" " ;
    M$(M01): PRINT "ANUL ";A01
1220 PRINT : PRINT "ZIU A "S$(DP2): PRINT ;D02;" " ;
    M$(M02):PRINT "ANUL ";A02
1230 PRINT: PRINT : PRINT : "SINT:": PRINT
1240 IF DPT=7 THEN LET DPT=DPT-7: LET W=W+1
1250 PRINT INT(DT/7);" SAPTAMINI SI(2sp)":DPT;
    " ZILE."
1260 PRINT : PRINT "=";DT;" ZILE."
1270 PRINT AT 20,9;"RELUATI(d/n)?"; PAUSE 0
1280 IF INKEY$="d" OR INKEY$="D" THEN GO TO 420
1300 CLS : STOP

```

15.4. BIORITM

După cum arată numele, programul oferă date asupra bioritmului unei persoane. Conform graficelor care vor fi afișate în urma rulării programului, se vor determina stările *emoționale, fizice și intelectuale* ale unui subiect testat.

```

8 REM bioritm
10 CLS : BORDER 2: PAPER 0: INK 7: CLS : PRINT AT
  6,10;"<BIORITM>"
20 INPUT "CUM VA NUMITI ?",LINE n$
30 PRINT AT 8,0;"Salut ";n$; AT 10,0"DATA NASTERII
  ?"
40 GOSUB 280: PRINT AT 10,19;d$: LET b$=d$: LET z=x
50 PRINT AT 12,0;"CE ZI DORINTI ?": GOSUB 280
60 PRINT AT 12,19;d$: LET d=x-z
70 INK 3: PRINT AT 15,0;"PINA LA ACEASTA DATA"
80 PRINT "AVETI ";d;" ZILE"
90 PRINT "ATI SERVIT ";3*d;" MESE,"
100 PRINT "SI ATI DORMIT TIMP DE " ` 8*d;" ORE"
110 PRINT "Apasati ENTER pentru bioritm",LINE i$
120 PAPER 0: INK 6: BORDER 2: CLS
130 PRINT TAB 8;"BIORITM PENTRU:"
140 PRINT TAB 15-LEN n$/2; FLASH 1;n$; FLASH 0; TAB
  12-LEN b$/2;"NASCUT ";b$
150 FOR a=1 TO 255 STEP 9: PLOT a,73: DRAW 0,3:
  NEXT a
160 FOR a=1 TO 255 STEP 63: PLOT a,71: DRAW 0,6:
  NEXT a
170 PLOT 127,0: DRAW 0,128
180 PRINT AT 21,0;"-14 zile"; TAB 22;" +14 zile"
190 PRINT AT 21,15-LEN d$/2;s$
200 INK 3: PRINT AT 3,0;"EMOTIONAL": LET c=28:
  GOSUB 380
210 INK 4: PRINT AT 3,11;"FIZIC": LET c=23:
  GOSUB 380
220 INK 5: PRINT AT 3,20;"INTELECTUAL": LET c=33:

```



```

GOSUB 380
230 PAPER 7
240 INPUT "Doriti reluarea programului? d/n", LINE
    a$
250 IF a$="d" OR a$="D" THEN GO TO 10
260 BORDER 0: PAPER 0: INK 7: CLS : PRINT AT 11,12;
    FLASH 1;"BIORITM": STOP
280 DATA 0,31,28,31,30,31,30,31,31,30,31,30,31
290 INPUT "ANUL ?",y: LET x=356*y+INT(y/4)-
    INT(y/100)
300 INPUT "LUNA(1-12)?",m:IF m<1 OR m>12 THEN GO TO
    300
310 RESTORE : FOR a=1 TO m: READ b: LET x=x+b: NEXT
    a
320 LET L=y+4*INT(y/4) AND y <> 100*INT (y/100)
330 IF L AND m>2 THEN LET b=29
340 READ b: IF L AND m=2 THEN LET b=29
350 INPUT ("ZIUA(1-";b;"?)"),d: IF d<1 OR d>b THEN
    GO TO 350
360 LET x=x+d: LET d$=STR$ d+"/"+STR$ m+"/"+STR$ y
370 RETURN
380 FOR a=0 TO 253: LET b=d-14+a/y
390 PLOT a,74+60*SIN(2*PI*b/c): DRAW 2,0
400 NEXT a
410 RETURN

```

15.5. CONVERSIE HEXAZECIMAL-ZECIMAL ȘI INVERS

O parte importantă din documentele care însoțesc programele elaborate de specialiști sau diversele articole din revistele de specialitate consacrate calculatoarelor folosesc sistemul de numerație hexazecimal. După cum se știe acest sistem are baza $b=16$ și utilizează 16 simboluri diferite pentru a reprezenta cele 16 cifre și anume:

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

unde literele semnifică valorile:

A=10; B=11; C=12; D=13; E=14; F=15.

Întrucât trecerea de la sistemul zecimal la cel hexazecimal și invers implică unele dificultăți, apare ca necesară elaborarea unui

program care să fie folosit de utilizator în aceste scopuri.

```

10 REM conversia heza-zecimal si invers
12 CLS : POKE 23658,8
14 RESTORE 18: DIM n(4): FOR F=1 TO 4: READ NUM
16 LET N(f)=num: NEXT F
18 DATA 4096,256,16,1
20 PAPER 0:INK 7: BRIGHT 0: BORDER 0: CLS
38 PRINT AT 19,1: INK 7; PAPER 1;"CONVERSIE(2sp)";
  AT 20,1;"HEXA <-->ZECIM"
39 PRINT # 0; AT 0,5;"PENTRU STOP TASTATI `S`"
40 PRINT AT 19,15; PAPER 4; INK 7;"H-CONVER.IN
  ZEC. "; AT 20,15;"D-CONV.IN HEXA.(2sp)"
42 PLOT 7,24: DRAW 106,0: DRAW 0,-17: DRAW -106,0:
  DRAW 0 17
44 PLOT 119,24: DRAW 136,0: DRAW 0,-17: DRAW
  -136,0: DRAW 0,17
46 PRINT AT 17,24; INK 0; PAPER 7;"OPTIUNI"
48 LET lin=5: LET O$=INKEY$
49 IF INKEY$="S" OR INKEY$="s" THEN STOP
50 IF O$="D" OR O$="d" THEN GOSUB 76: PRINT AT
  1,20;"IN"; AT 1,15; PAPER 6; INK 2;"HEXA"; AT
  1,23; PAPER 6; INK 1;"ZECIMAL"; AT 18,2; PAPER
  7; INK 0;"TASTATI NUMARUL DORIT....."
54 GO TO 48
56 LET c=6: GOSUB 154: LET dec=0: LET ex=1: FOR F=
  LEN N$ TO 1 STEP -1: IF CODE N$(F)<48 OR CODE
  N$(F)>57 THEN PRINT AT 16,5;"NU ESTE ZECIMAL":
  BEEP 2,-20: PRINT AT 16,4; FLASH 0;"(15sp): GO
  TO 56
58 LET DEC=((CODE N$(F)-48)*EX)+DEC: LET EX=EX*10:
  'EXT F: GOSUB 128
60 IF lin <= 19 THEN GO TO 56
62 PRINT AT 20,4; PAPER 6;"TASTATI `B`": FOR f=1 TO
  10: LET t$=INKEY$: NEXT f: IF t$="B" THEN FOR
  f=5 TO 19: PRINT AT f,21; PAPER 1;"(9sp)": NEXT
  f; PRINT AT 20,4: PAPER 7;"(12sp)": BEEP .02.40:
  LET lin=5: GO TO 56
64 GO TO 62

```

```

66 LET c=4: GOSUB 154: FOR F=LEN N$ TO 1 STEP -1:
  IF CODE N$(F)<48 OR CODE N$(F)>70 OR(CODE
  n$(f)>57 AND CODE n$(f)<65) THEN PRINT AT 12,6;
  "NU ESTE HEXA": BEEP 2,-20: PRINT AT 16,4; FLASH
  0;"(14sp)":GO TO 66
68 NEXT F: GOSUB 104
70 IF lin <= 19 THEN GO TO 66
72 PRINT AT 20,4; PAPER 6;"TASTATI `B`": FOR f=1 TO
  10: LET t$=INKEY$: NEXT f: IF t$="B" THEN FOR
  f=5 TO 19: PRINT AT f,21: PAPER 1;"(9sp)": NEXT
  f; PRINT AT 20,4; PAPER 7;"(12sp)": BEEP.02,40:
  LET lin=5: GO TO 66
74 GO TO 72
80 REM chenar
82 BORDER 4: INK 0: PAPER 7: CLS
84 PLOT 0,175: DRAW 255,0: DRAW 0,-175: DRAW
  -255,0: DRAW 0,175
86 PLOT 7,136: DRAW 153,0: DRAW 0,-33: DRAW-153,0:
  DRAW 0,33
88 PLOT 166,137: DRAW 83,0: DRAW 0,-123: DRAW-83,0:
  DRAW 0,123
90 PLOT 31,64: DRAW 41,0: DRAW 0,-9: DRAW -41,0:
  DRAW 0,9
92 PLOT 87,64: DRAW 41,0: DRAW 0,-9: DRAW -41,0:
  DRAW 0,9
94 PRINT AT 1,1;"CONVERSIUNE :"; PAPER 7; BRIGHT 1;
  AT 5,1;"Pt. reintoarcere la"; AT 6,1;"MENU(1sp)
  tastati `M` si "
96 PRINT BRIGHT 1; AT 7,1;"confirmati prin d/n"
98 FOR f=5 TO 19: PRINT AT f,21: PAPER 1; BRIGHT 1;
  "(9sp)"
100 NEXT f: PRINT PAPER 2; AT 14,4;"(4sp)": AT
  14,11; "(11sp)"
102 RETURN
104 REM subrutina de calcul H---->Z
106 LET h$=N$
108 IF LEN h$ <> 4 THEN LET h$="0"+h$: GO TO 108
110 LET dec=0
112 FOR f=1 TO 4

```

```

114 IF CODE h$(f)>47 AND CODE h$(f)<58 THEN LET dec=
    dec+((CODE h$(f)-48)*N(f))
116 IF CODE h$(f)>64 AND CODE h$(f)<71 THEN LET dec=
    dec+((CODE h$(f)-55)*N(f))
118 NEXT f
120 LET D$=STR$ DEC
122 PRINT INK 7; PAPER 2; AT 14,16-LEN D$;D$
124 PRINT AT lin,21: PAPER 1;INK 7; h$;"="; AT lin,
    (31-LEN d$);d$
126 LET lin=lin+1: RETURN
130 REM conversie Z---->H
134 LET d=dec: LET h$=""
136 LET coc=INT(d/16): LET rest=d-16*coc
138 IF rest >= 0 AND rest<9 THEN LET h$=STR$ rest+h$
140 IF rest >= 10 AND rest <= 15 THEN LET h$=
    CHR$(rest+55)+h$
142 IF coc <> 0 THEN LET d=coc: GO TO 136
144 IF LEN H$<4 THEN LET H$="0"+h$: GO TO 144
146 IF LEN H$>4 THEN PRINT AT 17,1;H$;" "; PAPER 3;
    FLASH 1;"IN AFARA LIMITELOR": FOR F=-20 TO 40:
    BEEP .02,F: NEXT F: PRINT AT 17,2; PAPER 7;
    FLASH 0;"(17sp)": RETURN
148 PRINT AT 14,11; INK 7; PAPER 2;" " ;h$
150 PRINT AT lin,26-LEN n$; INK 7; PAPER 1; n$;"=";
    h$: LET lin=lin+1: RETURN
152 PRINT h$: GO TO 134
156 REM rutina de scriere
160 LET n$=""
162 LET k$=INKEY$: IF k$="" THEN GO TO 162
164 IF k$="M" OR INKEY$="m" THEN GO TO 182
166 IF CODE k$=13 THEN RETURN
168 IF LEN n$=0 THEN PRINT AT 14,11; PAPER 2;
    (4sp)" ; AT 14,4;" "
170 LET N$=N$+k$
172 IF LEN N$>C THEN GO TO 160
174 BEEP .1,20: PRINT PAPER 2; INK 7; AT 14,9-LEN
    n$;n$
176 IF LEN n$<c THEN GO TO 162
178 PRINT AT 14,16-LEN n$; PAPER 2; INK 7;n$

```

```
180 GO TO 162
182 PRINT AT 20,5; FLASH 1;"MENU (d/n)?"
184 FOR F=1 TO 5: BEEP .01,F: LET t$=INKEY$: NEXT F:
  IF t$="M" OR t$="m" THEN FOR F=1 TO 5: BEEP
    .05,F: NEXT F: PRINT AT 19,11;"(9sp)": GO TO 160
186 IF t$="D" OR t$="d" THEN RUN
187 IF INKEY$="N" OR INKEY$="n" THEN CONTINUE
188 GO TO 184
```

Pentru: a) $16384_Z = 4000_H$; b) $23760_Z = 5CD0_H$; c) $32768_Z = 8000_H$;
d) $40000_Z = 9C40_H$; e) $65535_Z = FFFF_H$.

16. PROGRAME DE DIVERTISMENT (JOCURI)

16.1. PRELIMINARII

Programele de divertisment pe calculator reprezintă o categorie aparte de programe cu o mare audiență la utilizatorii de orice vîrstă. Realizate de echipe de profesioniști în informatică, reuniți sub diverse firme producătoare de jocuri, aceste programe solicită gîndirea, manipularea rapidă și imaginația celor care le joacă, elemente care le conferă un plus de activitate. Unele dintre aceste jocuri ating aproape perfecțiunea în ceea ce privește imaginea dinamică, fondul muzical, coloritul și complexitatea scenariului, motiv pentru care sînt asemuite cu desenele animate din cinematografie.

Subiectele acestor jocuri și caracteristicile lor (număr de participanți - joc individual, în echipe sau contra calculatorului, mulțimea și aspectul personajelor, multitudinea ecranelor în succesiunea jocului, ș.a.) fac aproape imposibilă o clasificare riguroasă a acestor programe.

Cu toate acestea, se pot considera convențional două categorii principale și anume:

a) **jocuri conversaționale**, în care, jucătorul este solicitat să găsească soluții la situațiile oferite și care, de regulă, necesită răspunsuri la întrebări privind deplasări la nord, sud, vest, stînga, dreapta, sus, jos și uneori declanșarea unui tir (fire-foc);

b) **jocuri neconversaționale** de tip:

- *static*, cînd se solicită abilitatea manuală, atenția și perspicacitatea jucătorului;

- *dinamic*, la care sarcinile precizate necesită spirit de observație, abilitate în mînuirea mișcărilor personajului central al jocului și elemente de decizie.

16.2. LOGICA JOCULUI

Orice joc se construiește pe baza unui "road mapping", adică pe o descriere generală alcătuită pe răspunsurile la o serie de întrebări dintre care cele mai importante sînt următoarele:

1) Care este *scopul jocului*? (ex: dezvoltarea logicii, a spiritului de observație, a rapidității mișcărilor utilizatorului, etc.)

2) *Cîți jucători* vor fi implicați și care este *rolul calculatorului* în acest sens? (de pildă calculatorul poate fi adversarul utilizatorului dacă jocul este de șah sau de tenis).

3) Care este *scenariul jocului*? La această întrebare se vor face referiri la:

- numărul personajelor care, alături de personajul principal, participă la joc și cum se dispune apariția lor în succesiunea evenimentelor jocului;

- numărul de ecrane care se succed și scopurile lor, numărul personajelor implicate, paleta coloristică și muzica aferentă;

- condițiile a căror îndeplinire permite trecerea la ecranul următor.

4) *Cum se sfîrșește jocul și se materializează reușita*, respectiv cum se trece la reluarea sau abandonarea jocului?

Este de remarcat că firmele producătoare de jocuri lansează concursuri pentru scenarii și apoi un scenariu aprobat este încredințat unei echipe formate din programatori, matematicieni, graficieni, muzicieni, psihologi, etc. a căror activitate comună se soldează într-un program cu secvențe diferite la același ecran intermediar, care se tastează în etape succesive pe grupuri de subiecți ce se manifestă critic, reușind prin sugestii să contribuie la perfecționarea programului. În final, programul este lansat pe piață în forma considerată cea mai adecvată scopului propus.

16.3. NOTIUNI SUMARE DESPRE ANIMATIE

A scrie despre animație pe calculator este o încercare temerară deoarece, pînă în prezent, nu s-au publicat "secretele" unor astfel de programe ultra performante. Mai mult decît atît, fiecare joc dispune de protecții din ce în ce mai sofisticate pentru a se împiedica "furtul de inteligență". De altfel, jocurile nu se scriu niciodată în BASIC ci numai în limbaj de asamblare care realizează viteze de deplasare și ritmuri în succesiunile de imagini net superioare, mișcările fiind cursive și nu "înterupte" ca în BASIC.

Din aceste motive, ceea ce se prezintă în continuare sînt numai unele sugestii la îndemîna utilizatorului care folosește în exclusivitate limbajul BASIC.

a) *Cum se deplasează aleator o mulțime de personaje*

Se consideră cazul unor "extraterestri" (fig.16.1) care apar aleator pe ecran și se deplasează dinspre partea inferioară spre cea superioară a ecranului.

Pozițiile aleatoare pot fi realizate utilizînd funcția RND . Astfel în secvența de program care urmează:

- liniile 10-30 definesc imaginea extraterestruului (care se obține tastînd în modul grafic litera A);
- liniile 40-90 definesc poziționarea aleatoare a unei mulțimi de extraterestri în continuă avansare.

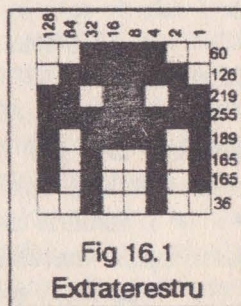


Fig 16.1
Extraterestru

4 PEM extraterestri

```

10 P WDER 0: PAPER 0: INK 7: CLS : RESTORE 30
20 FOR i=0 TO 7: READ a: POKE USR "a"+1,a: NEXT i
30 DATA 60,126,219,255,169,165,165,36
40 LET z=7: LET x=0: LET x=x+1
50 IF SCREEN$(0,z) <> "A" THEN GO TO 40: REM tasta
   A in modul grafic
60 BEEP .2,-30
70 LET a=INT(RND*30): LET b=INT(RND*29)

```



```

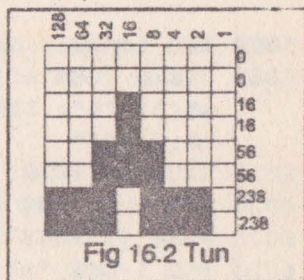
80 PRINT AT 21,a;"A";AT 21,b;"A": REM tasta A in
    modul grafic
90 POKE 23692,255: PRINT
100 IF INKEY$="" THEN GO TO 40

```

b) Cum se efectuează operațiunea "foc"

Se consideră cazul unui "tun" (definit pe tasta F) care poate emite raze laser. În programul următor:

- liniile 1000-1040 definesc tunul cu imaginea din fig.16.2;
- liniile 30-80 permit mișcarea tunului pe orizontală folosind tastele 5-stînga și respectiv 8-dreapta;
- liniile 500-525 determină apariția razei laser.



```

8 REM tir cu laser
10 BORDER 0: PAPER 0: INK 6: CLS
20 GOSUB 1000
30 LET J=8
40 LET J=J+(INKEY$="8" AND J<29)-(INKEY$="5" AND
    J>0)
50 IF j<0 THEN LET j=0: IF J>19 THEN LET J=19
60 PRINT AT 20,j;"(lsp)F(lsp)": REM tasta F in
    modul grafic
70 IF INKEY$ ="0" THEN GOSUB 500
80 GO TO 40
500 BRIGHT 1: PLOT 8*J+11,14
510 DRAW 0,154
515 PLOT INK 1;8*J+11,1
520 DRAW INK 1;0,151
525 BEEP .01,26: RETURN
1000 FOR I=0 TO 7: READ A
1010 POKE USR"F"+I,A
1020 NEXT I
1030 DATA 0,0,16,16,56,56,238,238
1040 RETURN

```

c) Cum se realizează animarea unui personaj

Se presupune că personajul ce se dorește animat este un iepure și că animația trebuie însoțită de un fond muzical vesel. Programul care urmează oferă o soluție posibilă. Întrucît lectura programului nu este dificilă s-a renunțat la comentarea lui.

```

6000 REM iepuras animat pe fond muzical
6005 CLEAR (USR"a")-100: DIM a$(2): DIM b$(2): LET
a$(1)="A": LET a$(2)="C": LET b$(1)="D": LET
b$(2)="B"
6010 RESTORE 6030: LET z=(USR"a")-99
6020 LET g=0: FOR i=z TO z+28
6025 READ j: POKE i,j: LET g=g+j: NEXT i
6026 FOR i=USR "a" TO USR "h"+7
6028 READ j: LET g=g+1: POKE i,j: NEXT i
6030 DATA 199,17,16,2,38,1,58,72,92,31,31,31,14,254,
238,16,237,121,67,16,254,37,32,244,1
6050 DATA 21,32,232,251,112,154,159,61,93,117,124,56,
8,62,93,157,21,116,119,7,14,89,249,188,186,174,
62,28,16,124,186,185,168,46,238,224,255,255,3
9301 PAPER 1: BORDER 1: CLS
9309 PRINT FLASH 1; INK 6; OVER 1: AT 1,0;"16 litere
A 16 litere D": INK 4: :REM tastele A si D in
modul grafic
9311 FOR j=28 TO 166 STEP 4: PLOT 0,j: DRAW 70,-3:
DRAW 90,3: DRAW 94,-3: NEXT j: POKE z,243
9313 INK 6: FOR j=3 TO 18: PRINT AT j,11;"(11sp)":
NEXT j
9315 FOR j=20 TO 126 STEP 6
9320 PLOT 150-j/2,48-j/5: DRAW j*.7,5: DRAW 5,-(j+5):
DRAW -(j*.7+7),0: NEXT j
9330 OR j=120 TO 255 STEP 6
9332 PLOT 150-j/2,48-j/5: DRAW j*.7+7,0: NEXT j: POKE
z+29,201
9607 FOR a=1 TO 32: BEEP.005,a+1/a: NEXT a: PRINT #1;
TAB 7; PAPER 3; INK 7;"APASATI TASTA 0!"
9608 FOR m=0 TO 12 STEP 12: RESTORE 9642: FOR k=1 TO
33:FOR L=1 TO 2: PRINT AT 19,2;a$(L); AT 20,2;

```

```

b$(L): READ j: BEEP .05,m+j: BEEP .05,m+j+12:
NEXT L: IF INKEY$="" THEN NEXT k: NEXT m: GO TO
9608
9642 DATA 2,2,0,0,4,4,7,7,12,11,9,9,7,7,4,7,7,7,12,12,
12,12,14,14,12,14,16,16,14,14,12,12,12,12,16,16,
14,14,12,12,9,9,7,7,4,4,2,0,0,0,7,7,7,7,7,7,7,7,
9,9,11,11,12,12,12,12
9650 IF INKEY$="0" THEN STOP

```

16.4. EXEMPLE DE JOCURI

Cu toate limitele impuse de limbajul BASIC, se pot concepe și în acest limbaj jocuri atractive cum sînt cele care urmează .

16.4.1. METEORIȚI

```

8 REM meteoriti
9 GO TO 3500
10 DIM y$(1)
30 BORDER 0: PAPER 0: INK 7: CLS
40 LET m=0
50 LET nf=1
60 LET b$="(1sp)BAZA(1sp)": LET e=0
100 LET v=0: CLS
110 LET a$="(5sp)C(5sp)C(5sp)C(5sp)C(5sp)C(2sp)":
REM tasta C in modul grafic
120 FOR i=1 TO 50: PLOT 255*RND,159*RND: NEXT i
130 PRINT AT 1,0;a$
140 LET f=9: LET b=5: PRINT AT 20,18-LEN b$; PAPER
6; INK 2;b$ PAPER 0;" "
150 PRINT AT 21,0;"(3sp)Scor=";e; TAB 12;"Maxim=";m;
TAB 24;"Cursa ";f
160 LET s=19: LET o=15
170 PRINT AT s,0;"A": REM tasta A in modul grafic
180 PAUSE 0
1000 IF nf>2 THEN LET a$=a$(2TO)+a$(1)
1010 PRINT AT 1,0;a$: LET o1=o: LET s1=s
1020 LET o=o+(INKEY$=r$ AND o<31)-(INKEY$=L$ AND o>0)

```

```

1030 LET s=s-1
1040 PRINT AT s,o;"A";AT s1,ol;" ": Rem tasta A in
modul grafic
1050 IF s<18 THEN PRINT AT s1+1,ol;" ": IF s>2 THEN
PRINT AT s+1,o: BRIGHT 1; INK 2;"B":REM tasta B
in modul grafic
1060 IF s>2 THEN GO TO 1000
1070 IF CODE a$(o+1) <> 146 THEN GO TO 2000
1080 LET a$(o+1)=" ": LET b=b-1: BEEP .2,50
1090 PRINT AT s-1,o;" "
1100 LET e=e+10
1110 IF m<e THEN LET m=e
1120 PRINT AT 21,8;e; AT 21,18;m
2000 LET ol=0: LET s1=s
2020 LET o=o+(INKEY$=r$ AND o<31)-(INKEY$=L$ AND o>0)
2020 LET s=s+1
2030 PRINT AT s,o;"A";AT s1,ol;" ": REM litera A in
modul grafic
2040 IF s<19 THEN GO TO 2000
2050 LET f=f-1: PRINT AT 21,30;f
2060 IF ATTR(s+1,o) <> 50 THEN LET v=2
2070 IF f=0 AND b>0 THEN LET v=1
2080 IF b=0 THEN GO TO 2600
2090 IF NOT v THEN BEEP .5,30: PAUSE 50: GO TO 1000
2500 PRINT INK 6; BRIGHT 1; AT 10,10;"N-AI REUSIT"
2510 PRINT INK 4; BRIGHT 1; AT 12,11 ;("AI CAZUT" AND
v=2);("SFIRSITUL CURSEI" AND v=1)
2520 LET i=10*RND: BEEP 1,i: BEEP 1,i-5
2530 PRINT AT 16,7;"INCA O DATA (d/n?)"
2535 PAUSE 0
2540 IF INKEY$="n" THEN STOP
2550 IF INKEY$="" THEN GO TO 2540
2560 CLS : GO TO 50
2600 PRINT AT 10,11; PAPER 7*RND; INK 9; FLASH 1;
"FELICITARI"
2610 LET e=e+(50*f)+50*nf: IF m<e THEN LET m=e
2620 LET nf=f+1
2630 IF nf=5 OR nf=6 THEN LET b$=b$(TO LEN b$-1)
2640 FOR i=1 TO 200: NEXT i

```

```
2650 GO TO 100
3500 BORDER 1: PAPER 0: INK 7: CLS : PLOT 0,0: DRAW
    0,175: DRAW 255.0: DRAW 0,-175
3510 FOR i=1 TO 120 STEP 5: PLOT 0,i: DRAW 255,-i:
    PLOT 0,i-175: DRAW 255-i,i: PLOT i,175: DRAW
    255-i,-i: NEXT i
3600 PRINT AT 10,12; INK 2;"METEORITI"
3610 INPUT "DORITI INSTRUCIUNI(d/n)?" ;y$
3620 IF y$ <> "d" THEN GO TO 4060
4000 BORDER 3: PAPER 5: INK 0: CLS : PRINT AT 0,11,
    FLASH 1;"METEORITI";AT 21,4: FLASH 0: PAPER 6;
    INK 1;"APASA O TASTA OARECARE"
4005 PRINT AT 3,1; PAPER 6; INK 0;"MISIUNEA:"; PAPER
    5; INK 2;"colectarea a 5 meteo-riti"
4010 PRINT AT 5,1;"Dispui de 9 incercari."
4020 PRINT AT 7,1;"Pentru fiecare meteorit capeti
    (2sp)10 puncte,iar pentru reusita un premiu."
4030 PRINT AT 11,0;Tu hotarasti momentul primei
    (4sp)decolari(dar numai a primei)"
4040 PRINT AT 15,1;"Dupa fiecare zbor trebuie sa te
    intorci la BAZA!!!! "
4050 PRINT AT 19,1;"De asemenea,trebuie sa dirijezi
    miscarile navei.": PAUSE 0
4060 CLS : PRINT AT 18,7;TASTE DE CONTROL": INPUT
    "Alege tasta pentru `stinga`= ";L$
4070 IF L$="" THEN LET L$="5"
4080 PRINT AT 21,1;"Stinga=" ;L$
4090 INPUT "Alege tasta pentru `dreapta`= ";r$
4100 IF r$="" THEN LET r$="8"
4110 PRINT AT 20,16;"Dreapta=" ;r$
4120 PRINT # 0;"Apasa o tasta cind putem incepe!"
4130 PAUSE 0
5000 RESTORE 5100
5010 FOR i=0 TO 23: READ j: POKE USR"a"+i,j: NEXT i
5050 GO TO 20
5100 DATA 24,24,24,60,60,126,126,126
5110 DATA 8,24,28,60,60,28,24,8
5120 DATA 129,66,40,0,0,40,66,129
```

Scopul jocului este să se culeagă meteorii pentru a fi aduși la baza de cercetare științifică. Utilizatorul trebuie să-și precizeze tastele de control pentru deplasarea navei la stînga și la dreapta (în lipsa opțiunii calculatorul impune tastele 5 și respectiv 8). Se dispune de 9 încercări, reușitele fiind punctate.

16.4.2 JOC LOGIC ȘI DE ÎNDEMÎNARE

Jocul constă în a ordona crescător numerele 1 ... 36 folosind tastele: o = stînga; p = dreapta; q = sus și a = jos.

```

1 REM joc logic si de indeminare
3 BORDER 0: PAPER 0: INK 7: CLS : PRINT AT 12,6;
  "I N D E M I N A R E"
5 PRINT # 0; AT 0,4; PAPER 6: INK 2; FLASH 1;
  "APASATI O TASTA OARECARE": PAUSE 0
10 DIM a$(2)
15 RESTORE : GOSUB 8000
20 LET j=0: LET k=0: LET j1=0: LET k1=0
30 LET a$=""
85 PAPER 4: BORDER 4: CLS
90 PAPER 5: INK 0
100 LET n=0: FOR x=1 TO 6; FOR y=1 TO 6: LET n=n+1:
  LET j=(x*3)+1 :LET k=(y*4)
110 PRINT AT j,k;"ABBC"; AT j+1,k;"D(2sp)E";AT j+1,
  k+1;n;; AT j+2,k"FGGH": NEXT x: REM tastele
  majuscule in modul grafic
120 PRINT AT 0,4;"AB21 litere BC": REM toate in
  modul grafic
13 PRINT AT 1,4;"DASTEAPTA!(2sp)MA ARANJEZ!E: REM D
  si E in modul grafic
135 PRINT AT 2,5;;"s=stop"
140 PRINT AT 2,4;"F21 litere GH": REM toate in modul
  grafic
150 LET x=6: LET y=6: LET x1=6: LET y1=6: GOSUB 2000
160 GOSUB 1500
170 PRINT AT 1,4;"Do=st;p=dr(2sp)q=sus:a=josE":
  PRINT # 0;AT 0,0; PAPER 7; INK 2; FLASH

```

```

1;"ORDONEAZA IN ORDINE CRESCATOARE!": REM
  literelul D si E in modul grafic
180 PRINT AT 2,4;"F21 litere GH": REM toate in modul
  grafic
1000 IF INKEY$="" THEN GO TO 1000
1005 IF INKEY$="S" OR INKEYS="s" THEN STOP
1010 LET y=y+(INKEY$="p" AND y<6)-(INKEY$="o" AND
  y>1)
1020 LET x=x+(INKEY$="a" AND x<6)-(INKEY$="q" AND
  x>1)
1030 GOSUB 2000: GO TO 1000
1500 INK 0: PAPER 5
1510 FOR n=1 TO 500: LET q=INT(RND*4)+5
1520 LET y=y+(q=5 AND y<6)-(q=8 AND y>1)
1530 LET x=x+(q=7 AND x<6)-(q=6 AND x>1)
1550 GOSUB 2000
1560 NEXT n: RETURN
2000 LET j=(x*3)+1: LET k=(y*4)
2005 LET j1=(x1*3)+1: LET k1=(y1*4)
2010 LET a$(1)=SCREEN$(j+1,k+1): LET a$(2)=SCREEN$
  (j+1,k+2)
2020 PRINT INK 0; AT j1,k1;"ABBC"; AT j1+1,k1;
  "D(2sp)E"; AT j1+1,k1+1;a$; AT j1+2,k1;"FGGH":
  REM majusculele in modul grafic
2030 PRINT PAPER 0; AT j,k;"(4sp)"; AT j+1,k;"(4sp)";
  AT j+2,k;"(4sp)"
2050 LET x1=x: LET y1=y
2060 RETURN
7000 PAPER 4: BORDER 4: INK 0: CLS
8000 FOR n=USR "A" TO USR "h"+7: READ u: POKE n,u:
  NEXT n
8010 RETURN
9000 DATA 255,255,224,192,192,192,192,192,255,255,
  0,0,0,0,0,255,255,7,3,3,3,3,3,192,192,192,
  192,192,192,192,192,3,3,3,3,3,3,3,192,192,
  192,192,192,224,255,255,0,0,0,0,0,255,255,3,
  3,3,3,3,7,255,255
9998 STOP

```

16.4.3. BALISTICA

Jucătorul trebuie sa arunce o bilă într-o țintă, apreciind distanța la care se află ținta în raport cu locul de lansare al bilei; el dispune de 10 încercări, reușitele fiind punctate.

```

5 REM balistica
10 BORDER 6: PAPER 6: INK 1: CLS : GOSUB 290: PRINT
  AT 0,10;"BALISTICA"```"Trebuie sa precizeze
  distanta"```(Min 200 m-max 450 m pentru"``` a
  arunca bila in tinta "``` Ai la dispozitie 10
  bile..."
20 FOR r=1 TO 4: BEEP .3,24 BEEP .3,12: NEXT r: FOR
  r=1 TO 2: FOR n=0 TO 36 STEP .5: BEEP .005,n:
  NEXT n: PRINT AT 18,5;:APASA O TASTA OARECARE":
  PAUSE 0: CLS : LET bile=0: LET hi=0: LET sc=0:
  LET b$=""
30 PRINT PAPER 4; INK 1; AT 0,0;"31 litere C"; AT
  21,0;"31 litere C": REM litera C in modul grafic
40 FOR m=0 TO 9: PRINT PAPER 4; INK 1; AT n,0;"C":
  NEXT n: FOR n= 17 TO 21: PRINT PAPER 4; INK 1;
  AT n,0;"C": NEXT n: FOR n=0 TO 21; PRINT PAPER
  4; INK 1;AT n,31;"C": NEXT n: REM litera C in
  modul grafic
50 FOR n=17 TO 20: PRINT PAPER 2; INK 1; AT n,1;"30
  semne ■" NEXT n: REM tasta S cu CS+SS si tasta
60 PRINT INK 0; AT 5,9;"A"; AT 5,10;"AAA"; AT 5,15;
  "AA": REM tasta A in modul grafic
70 PRINT BRIGHT 1; AT 18,6;"200"; AT 18,14;"300";
  AT 18,25;"400"; AT 20,10;"(2sp)TINTA(2sp)"
8. LET f=(RND*18)+10: FOR n=6 TO 9: PRINT PAPER 2;
  INK 1;AT n,1;"30 litere S": NEXT n: PRINT AT
  7,4;"10 BILE PENTRU 10 PUNCTE": REM litera S in
  modul grafic
90 FOR n=10 TO 16: PRINT INK 0; AT n,2;" ": NEXT n
100 PRINT AT 14,1; INK 0;"A ";AT 15,1;"B": PRINT
  PAPER 2; INK 0; AT 15,2;"E": REM literele A,B,E
  in modul grafic

```



```

110 IF bile=9 THEN GO TO 230
120 LET bile=bile+1: PRINT AT 1,1: PAPER 7: INK 1;
    "BILE:";bile; AT 3,9;:PUNCTAJUL=";sc; AT 1,9;
    "SCOR-MAX=";hi; AT 1,21;"(2sp)";b$
130 INK 0: PRINT AT 15,f;"CD": LET p=40: INPUT
    "DISTANTA ?(200-425",rng
140 IF rng>425 THEN PRINT FLASH 1; INK 0; AT 15,4;
    "DISTANTA MARE !!!": PAUSE 150: PRINT PAPER 6;
    AT 15,5;"(20sp)": GO TO 130
150 IF rng<200 THEN PRINT FLASH 1; INK 0; AT 15,4;
    "DISTANTA MICA !!!": PAUSE 150: FLASH 1: PRINT
    PAPER 6; AT 15,5;"(20sp)": GO TO 130
160 PRINT AT 9,2;"DISTANTA =" ;rng
170 LET a=rng*COS(PI*p/180): LET b=rng*SIN
    (PI*p/180): FOR x=0 TO b/16 STEP.5: LET c=.01*
    (b*x-16*x*x)
180 IF a*x>6200 THEN GO TO 220
190 PLOT PAPER 6; INK 0;.04*a*x+12,4*c+50: BEEP
    .005,c+25: NEXT x
200 IF ABS (A*b/3200-f)<1 THEN GO TO 220
210 PRINT AT 9,16: FLASH 1;"PIERDUT": BEEP .5,-20:
    PAUSE 150: CLS : GO TO 30
220 PRINT AT 10,10; INK 0; FLASH 1;"AI REUSIT!!":
    FOR N=10 TO 20: BORDER 1: BORDER 2: BORDER 3:
    BORDER 4: BORDER 5: BORDER 6: BEEP .03,N:: NEXT
    N: LET sc=sc+1: PAUSE 50: CLS : GO TO 30
230 IF sc>0 AND hi<sc THEN LET hi=sc; CLS : GO TO
    260
240 PRINT FLASH 1;AT 5,2;"JOCUL S-A TERMINAT"; FLASH
    0;"ALT JOC?(D/N)": INPUT a$
250 IF a$="d" OR a$="D" THEN LET sc=0: LET bile=0:
    CLS : GO TO 30
255 IF a$="n" OR a$="N" THEN CLS : PRINT AT 14,8;"LA
    REVEDERE!!!": STOP
270 PRINT AT 5,2;"CEL MAI BUN SCOR "; AT 7,3;"SCRIE
    NUMELE(max 8 litere)": INPUT b$
280 LET bile=0: LET sc=0: CLS : GO TO 30
290 FOR i=1 TO 5: FOR n=0 TO 7: READ a: POKE USR
    CHR$(i+143)+n,a: NEXT n: NEXT i: RESTORE 300

```

```

300 DATA 56,56,60,56,56,112,112,127
310 DATA 248,248,248,248,112,112,112,112
320 DATA 192,192,96,96,48,48,255,255
330 DATA 3,3,6,6,12,12,255,255
340 DATA 3,6,48,48,96,192,255,255
350 RETURN

```

16.4.4. ROBOT

Scopul jocului este ca robotul să curețe ecranul în timpul cel mai scurt. Tastele cu care se comandă deplasarea robotului sînt 5, 6, 7, 8. După terminarea curățirii unui ecran se apasă S, calculatorul întrebînd dacă se dorește același ecran sau schimbarea lui. Jocul se încheie după 5 misiuni de curățire a ecranului, afișînd punctajul și timpul consumat.

```

3 REM robot
5 BORDER 5: PAPER 5: INK 1: CLS
10 FOR I=0 TO 167: READ A: POKE USR "A"+I,A: NEXT I
13 PAPER 0: INK 7: PRINT AT 1,8;"ROBOT-JOC LOGIC":
   PAPER 7:INK 0: PRINT AT 3,0;"(7sp)K N A J J J J
   J(16sp)B B B B B B B B(16sp)C C C C C C C
   (8sp)": BEEP 3,20 REM literele majuscule in
   modul grafic
14 BORDER 5: PAPER 5: INK 1: PRINT AT 7,6;"SCOP:
   stergerea ecranului": PRINT "din minimum de
   mutari": PRINT "cu ajutorul robotului.": PRINT
   FLASH 1; AT 11,12;"SUCCES!"; FLASH 0: PRINT :
   PRINT : PRINT "TASTELE PT. DEPLASARE SINT:":
   PRINT "SUS-7": PRINT "JOS-6": PRINT "STINGA-5":
   PRINT "DREAPTA-8": PRINT "STOP-S(cind ati
   curatit ecranul in intregime)": PAUSE 200 : BEEP
   .3,12: BEEP .3,14: BEEP .3,15: BEEP 3,-4
15 BORDER 3
20 DATA 124,92,252,124,24,126,255,189,189,189,189,
   189,189,60,60,60,102,102,102,102,102,102,
   102,102,7,5,15,7,1,7,79,43,192,192,192,192,128,
   224,248,204,59,19,3,3,3,3,3,3,198,204,208,192,
   192,192,192,192,6,12,24,48,48,48,48,240,

```

```

96,48,24,24,31,15,1,1,126,90,102,126,24,126,255,
187,126,126,126,126,24,126,255,187,102,102,
102,102,102,96,96,96,102,102,102,102,102,6,6,6,
62,58,63, 62,24,126,255,187,3,3,3,3,1,7,31,51,
224,160,240,224,128,224,242,212,99,51,11,3,
3,3,3,3,218,200,192,192,192,192,192,192,6,12,24,
24,248,240,128,128,96,48,24,12,12,12,12,15,
255,255,255,255,255,255,255,255,255
25 GO TO 1800
30 LET D=2: LET X=10: LET Y=10: INK 1
35 LET C=0: LET t=0
40 IF d=1 THEN PRINT AT Y,X;"A";AT Y+1,X;"B"; AT
Y+2,X;"C": REM literеле A,B,C, in modul grafic
41 LET T=T+1: PRINT # 1; AT 0,20;"TIMP:";T
42 PRINT # 1; AT 0,0;"SCOR:";C
50 IF D=-1 THEN PRINT AT Y,X,"N";AT Y+1,X;"B"; AT
Y+2,X;"C": REM literеле N,B,C in modul grafic
60 IF D=2 THEN PRINT AT Y,X,"J"; AT Y+1,X;"B"; AT
Y+2,X;"C":REM literеле J,B,C in modul grafic
70 IF D=-2 THEN PRINT AT Y,X;"K" Y+1,X;"B";AT
Y+2,X;"C": REM literеле K,B,C,in modul grafic
80 IF INKEY$="6" THEN GO TO 200
85 IF INKEY$="s" OR INKEY$="S" THEN BEEP 2,10: GO
TO 1500
90 IF INKEY$="7" THEN GO TO 300
100 IF INKEY$="5" THEN GO TO 400
110 IF INKEY$="8" THEN GO TO 500
120 GO TO 40
200 IF D <> 2 THEN LET D=2: GO TO 40
201 LET C=C+1
210 IF Y >= 19 THEN GO TO 40
215 BEEP .01,-10
220 PRINT AT Y+2,X;"L": BEEP .01,-10: REM litera L
in modul grafic
230 PRINT AT Y,X;" "; AT y+1,X;"J"; AT Y+2,X;"B";
AT Y+3,X;"M": BEEP .01,-10: REM majusculele in
modul grafic (J,B,M)
240 LET Y=Y+1: GO TO 40
300 IF D <>-2 THEN LET D=-2: GO TO 40

```

```
301 LET C=C+1
310 IF Y<=0 THEN GO TO 40
320 BEEP .01,-10
330 PRINT AT Y+2,X;"M": BEEP .01,-10: REM litera M
    in modul grafic
340 PRINT AT Y+2,X;" "; AT Y-1,X;"K" ; AT Y,X;"B";
    AT Y+1,X;"L": BEEP .01,-10: REM majusculele in
    modul grafic (K,B,L)
350 LET Y=Y-1: GO TO 40
400 IF D <> 1 THEN LET d=1: GO TO 40
401 LET C=C+1
410 IF X <= 0 THEN GO TO 40
420 BEEP .01,-10
430 PRINT AT Y,X-1;"DE"; AT Y+1,X-1;"FG"; AT Y+2,
    X-1;"HI": BEEP.01,-10: REM literele D,E,F,G,H,I
    in modul grafic
440 PRINT AT Y,X;" "; AT Y+1,X;" "; AT Y+2,X;" ":
    BEEP .01,-10
450 LET X=X-1: GO TO 40
500 IF D <>-1 THEN LET D=-1: GO TO 40
501 LET C=C+1
510 IF X >-31 THEN GO TO 40
520 BEEP .01,-10
530 PRINT AT Y,X;"OP"; AT Y+1,X;"QR" AT Y+2,X;"ST":
    BEEP.01,-10: REM literele O,P,Q,R,S,T in modul
    grafic
540 PRINT AT Y,X;" " ; AT Y+1,X;" " ; AT Y+2,X;
    " ": BEEP .01,-10
550 LET X=X+1: GO TO 40
15 0 PRINT AT 10,3;"DORITI ACELASI ECRAN(D/N)?"; AT
    15,3;"SCOR: ";C+INT (T/100): PAUSE 0
1510 IF INKEY$="d" OR INKEY$="D" THEN GO TO 2999
1520 IF INKEY$="n" OR INKEY$="N" THEN GO TO 1800
1530 GO TO 1500
1800 DIM A(32)
1999 CLS
2000 FOR I=0 TO 31
2005 LET A(I+1)=INT (RND*22)
2010 PRINT AT A(I+1),I;"U": REM litera U in modul
```

```

    grafic
2020 NEXT I
2030 GO TO 30
2999 CLS
3000 FOR I=0 TO 31
3010 PRINT AT A(I+1).I;"U": NEXT I: REM litera U in
    modul grafic
3020 GO TO 30

```

16.4.5. TIR CU ARBALETA

Jocul are ca obiectiv ochirea a 10 ținte care se deplasează aleator, folosind o arbaletă al cărei tir se declanșează apăsînd tasta 0 (zero). Pentru fiecare țintă atinsă se obțin 100 puncte.

```

2 REM tir cu arbaleta
3 BORDER 0: PAPER 0: INK 6: CLS
4 PRINT AT 14,6; FLASH 1;:PROGRAMUL SE INCARCA":
  FOR I=0 TO 80: BORDER 0: BORDER 1: BORDER 2:
  BORDER 3: BORDER 4: BORDER 5: BORDER 6: BORDER
  0: NEXT I: PAUSE 50
5 GO TO 8
6 BORDER 2: BORDER 3: BORDER 4: BORDER 5: BORDER
  6; BORDER 0: PAUSE 1: IF INKEY$="" THEN GO TO 6
7 RETURN
8 POKE 23658,8
9 BORDER 0: PAPER 0: INK 7: CLS
15 LET L=0: LET S=0: LET m=0: GOSUB 8000
20 GOSUB 130
30 LET O=-1: LET X=30: LET V=-1
40 FOR B=1 TO 2
50 GOSUB 80
60 PRINT AT Y,X; INK 1;" AB ": BEEP .002,X+X
65 INK (RND*7)+1: PLOT 128,23: DRAW 0,24: PLOT 129,
  24: DRAW 0,2: PLOT 129,43: DRAW 0,2
70 IF INKEY$="0" THEN GO TO 200
75 GO TO 50
80 LET X=X+V

```

```
90 IF X=0 THEN GOSUB 100
95 RETURN
100 PRINT AT X,Y;"(3sp)": LET O=30: LET X=-1: LET
    V=1: GOSUB 130: NEXT B: GO TO 30
130 LET Y=INT(RND*15): IF Y<4 THEN GO TO 130
133 IF M=10 THEN CLS : GO TO 400
135 LET M=M+1
140 LET I=INT(RND*7)+2
145 RETURN
205 FOR T=14 TO 3 STEP -1
210 IF SCREEN$(T+1,15) <> " " THEN GO TO 260
220 PRINT AT T,15;"C": BEEP .002,T+T PRINT AT
    T,15;" "
230 PRINT AT Y,X; INK 1;" AB ": BEEP .002,X+X
240 GOSUB 80
250 NEXT T
255 GO TO 60
260 FOR A=0 TO 5: PRINT AT T+1,15-1: INK 7: BRIGHT
    1;"BINE": BEEP.01,A: PRINT AT T+1,15-1;"(4sp)":
    NEXT A
270 LET S=S+100: LET L=L+1: GOSUB 9600: GO TO 20
305 FOR A=50 TO 8 STEP -2: BEEP .002,A+3: BEEP .002,
    A-6: BEEP.002,A+16: BEEP .002,A-10: BEEP
    .001,50: NEXT A
310 PRINT AT 6,10; INK 6; PAPER 2;"EXCELENT"
320 PRINT AT 20,8; FLASH 1; INK 4;"APASA O TASTA"
322 FOR A=1 TO 7: BORDER A: BORDER 0: PRINT AT
    11,11; INK A;"10000": BEEP .002,PI*A: NEXT A: IF
    INKEY$="" THEN GO TO 322
325 LET S=S+10000: LET M=-10: LET L=-10
330 BORDER 0: CLS : GOSUB 9500: GO TO 20
410 PRINT AT 7,4; INK 6; PAPER 1;"(2sp)SCORUL :";S;
    "(1sp)PUNCTE(1sp)"
420 BEEP .002,10: BEEP .003,-5
425 PRINT AT 10,6; INK 6;"INCA UN JOC?(D/N)": PAUSE
    0
430 IF INKEY$="D" OR INKEY$="d" THEN CLS : LET M=0:
    LET S=0:LET L=0: GOSUB 9000: GO TO 20
435 IF INKEY$="N" OR INKEY$="n" THEN CLS : STOP
```

```
8005 FOR C=144 TO 153: FOR A=0 TO 7: READ N: POKE USR
    CHR$ C+A,N: BEEP .001,C-100: NEXT A: NEXT C
8010 DATA 0,0,6,29,58,0,125,15
8020 DATA 0,0,176,220,174,0,95,248
8030 DATA 8,28,28,8,8,8,20,20
8040 DATA 34,34,255,34,34,34,255,34
8050 DATA 82,180,173,75,10,144,0,0
8060 DATA 42,45,85,99,54,20,72,42
8070 DATA 0,0,1,1,1,1,3,3
8080 DATA 3,3,7,6,6,14,14,28
8090 DATA 60,126,255,223,206,238,125,57
8100 DATA 59,63,31,30,14,12,0,0
9000 PRINT AT 4,12; INK 7; BRIGHT 1;"T(2sp)I(2sp)
    R(1sp)"
9010 PRINT AT 6,0; INK 7;"TE AFLI LA O TRAGERE CU
    ARBALETA";INK 6; "(2sp)DACA OCHESTI 10 TINTE
    (2sp)POTI(6sp)OBTINE 10000 PUNCTE !!!!"
9015 PRINT AT 11,2; INK 7; BRIGHT 1;"TINTELE SE
    DEPLASEAZA ALEATOR(5sp)IN DREAPTA SI SPRE
    STINGA"
9020 PRINT AT 17,2; INK 6; INVERSE 1;"CU TASTA `0`
    DECLANSEZI TIRUL."
9025 PRINT AT 0,0; INK 2;"De 16 ori AB": REM literele
    AB in modul grafic
9027 PRINT AT 21,0; INK 7: BRIGHT 1;"31 litere C":
    REM C in modul grafic
9030 PRINT AT 21,9; FLASH 1; INK 6; PAPER 2;"APASA O
    TASTA.."
9040 PRINT AT 4,7; INK 4;"(1sp)AB"; AT 4,20;"(1sp)
    AB(1sp)"
9050 GOSUB 6
9060 CLS
9505 PRINT AT 1,0; INK 7; BRIGHT 1;"31 litere D": REM
    D in modul grafic
9510 PRINT AT 2,0: INK 4;"32 litere E": REM E in
    modul grafic
9520 PRINT AT 20,0; INK 4;"32 cifre 8": REM in modul
    grafic
9540 LET E=13: LET k=0
```

```
9545 FOR H=1 TO 2
9550 FOR N=1 TO 10
9560 LET A=INT(RND*4)+16
9565 LET B=INT(RND*E)+K
9570 PRINT AT A,B; INK(RND*6)+2; BRIGHT 1;"F": REM F
    in modul grafic
9575 NEXT N: LET E=14: LET K=17: NEXT H
9590 PRINT AT 16,15; BRIGHT 0; INK 7;"G"; AT 17,15;
    "H" AT 18,15;INK 2;"I";AT 19,15;INK 7;"J":REM
    literale G,H,I,J.in modul grafic
9610 PRINT AT 21,1; INK 2; PAPER 6; BRIGHT 1; INVERSE
    1;" SCOR=";S
9620 IF L=10 THEN CLS : GO TO 300
9630 RETURN
```


17. BETA BASIC

17.1. INTRODUCERE

Posibilitățile interpretorului BASIC al calculatoarelor compatibile cu XZ-SPECTRUM sînt limitate, motiv ce a determinat apariția programului **BETA BASIC** care introduce comenzi și funcții noi. Viteza de lucru este mărită deoarece programele sînt semicompilate. Sînt realizate două variante de BETA BASIC și anume:

- varianta **3.1** avînd un cod mașină de **18607** octeți la adresa **46930**
- varianta **1.8** avînd un cod mașină de **9567** octeți la adresa **55801**.

Încărcarea acestor programe se face cu comenzile cunoscute : **LOAD ""** dacă este primul program de pe casetă sau **LOAD "nume"**. Vor fi încărcate 3 linii de program numerotate 0,1 și 2 unde linia 2 intră în **auto-run** determinînd încărcarea codului mașină aferent programului. Încărcarea programului se termină cînd pe ecran apare mesajul firmei BETASOFT. Linia 1 conține o rutină ce permite crearea unei copii a programului BETABASIC, iar linia 0 conține definițiile noilor funcții din BETA BASIC. Alt semn că programul BETA BASIC este rezident este pointer-ul liniei curente care apare flashing (poate fi adus la forma inițială cu POKE 23609, 0).

Orice program încărcat cu **MERGE BASIC** va fi rulat normal cu excepția celor care folosesc UDG, pentru care se va comanda **"KEYWORDS 0"** (încărcarea prin LOAD va distruge linia 0 care este

absolut indispensabilă). Se va constata o creștere a vitezei de execuție mai ales în cazul programelor lungi care folosesc multe GO TO-uri sau GOSUB-uri; instrucțiunea RETURN este mult mai rapidă iar comanda NEW nu va șterge linia 0. Eliminarea lui BETA BASIC se face tastind RANDOMIZEUSR 0.

Comenzile noi sînt obținute prin trecerea în modul grafic (Caps Shift și 9) și apăsînd apoi o tastă indicată (uneori cu SHIFT, adică CS sau SS), iar noile funcții sînt obținute prin folosirea tastei FN împreună cu o literă indicată urmată de semnele "(" sau "\$". Ele se vor comporta precum cuvintele cheie ale limbajului BASIC.

O sinteză a acestor noi comenzi și funcții este prezentată în tabelele 17.1 și 17.2, comparativ pentru cele două variante de BETA BASIC.

TABELUL 17.1 - COMENZI BETA BASIC

TASTA	DENUMIREA la VERSIUNEA 1.8	DENUMIREA la VERSIUNEA 3.1
8	KEYWORDS	KEYWORDS
1	DEF PROC	DEF PROC
2	PROC	PROC
3	END PROC	END PROC
4	RENUM	RENUM
5	----	WINDOW
6	AUTO	AUTO
7	DELETE	DELETE
SHIFT 7	----	REF
SHIFT 6	JOIN	JOIN
SHIFT 5	EDIT	EDIT
SHIFT 4	KEYN	KEYN
SHIFT 3	----	LOCAL
SHIFT 2	----	DEFAULT
SHIFT 1	DEFKEY	DEFKEY
SHIFT 8	----	CSIZE
A	ALTER	ALTER

B	B (nu este functie)	BLANK
C	CLOCK	CLOCK
D	DO	DO
E	ELSE	ELSE
F	FILL	FILL
G	GET	GET
H	H (nu este functie)	BLANK
I	EXIT IF	EXIT IF
J	WHILE	WHILE
K	UNTIL	UNTIL
L	LOOP	LOOP
M	SORT	SORT
N	ON ERROR	ON ERROR
O	ON	ON
P	DPOKE	DPOKE
Q	POP	POP
R	ROLL	ROLL
S	SCROLL	SCROLL
T	TRACE	TRACE
U	USING	USING
V		RND
W	INKEY\$	INKEY\$
X	RND	PI
Y	FN	FN
Z	POINT	POINT

TABELUL 17.2. - FUNCTII BETA BASIC

AFISATE	TASTATE	TASTATE
	LA VERSIUNEA 1.8	LA VERSIUNEA 3.1
AND (FN A(FN A(
BIN\$	FN B\$	FN B\$
CHAR\$	FN C\$	FN C\$
COSE (FN C(FN C(
DEC (FN D(FN D(

DPEEK (FN P(FN P(
EOF	-	FN E(
FILLED (FN F(FN F(
HEX\$	FN H\$	FN H\$
INARRAY (-	FN U(
INSTRING(FN I(FN I(
ITEM	-	FN T(
MEM(FN M(FN M(
MEMORY\$	FN M\$	FN M\$
MOD(FN V(FN V(
NUMBER (FN N(FN N(
OR (FN O(FN O(
RNDM(FN R(FN R(
SCRN\$	FN K\$(FN K\$(
SHIFT\$	-	FN Z\$
SINE (FN S(FN S(
STRING\$	FN S\$	FN S\$
TIMES	FN T\$	FN T\$
USING\$	FN U\$	FN U\$
XOR(FN X(FN X(

Majoritatea acestor noi comenzi și funcții vor fi prezentate în cele ce urmează folosind ca bază versiunea 3.1 care este mai completă (ce este scris între paranteze pătrate este opțional).

17.2. COMENZILE/INSTRUCTIUNILE BETA BASIC

17.2.1. ALTER

Obținere	CS+9 și tasta A
Sintaxa	ALTER [atribut vechi] TO atribut nou
Efect	modifică atributele ecranului (INK, PAPER, FLASH, BRIGHT); în forma cea mai simplă ALTER schimbă valorile lui INK sau PAPER fără a șterge ecranul.

Exemple	<p>a) 100 PRINT AT 11,11;"BETA BASIC": PAUSE 50: ALTER TO PAPER 1(schimbă PAPER-ul fiecărui caracter în albastru)</p> <p>b) 100 PRINT AT 11,11;"BETA BASIC": PAUSE 50: ALTER PAPER 7 TO PAPER 2; INK 7; FLASH 1</p>
----------------	---

Exemplul 17.1: posibilitățile oferite de **ALTER** care schimbă atributele unei reprezentări grafice în maniera tablei de șah.

```

10 LET A=2: LET B=4
20 FOR L=1 TO 5
30 FOR N=1 TO 16
40 PRINT INK A; PAPER B;"XXXX"; PAPER A; INK
   B;"OOO";
50 NEXT N
60 LET C=A: LET A=B: LET B=C
70 NEXT L
80 LET T=30
90 ALTER INK A TO INK B: PAUSE T
100 ALTER PAPER A TO INK A: PAUSE T
110 ALTER INK A TO PAPER B: PAUSE T
120 ALTER INK B TO PAPER A: PAUSE T
130 LET T=T-T/10+1: GO TO 180

```

17.2.2. AUTO

Obținere	CS+9 și tasta 6
Sintaxa	AUTO [nr.linie] [,pas]
Efect	<p>permite numerotarea automată a liniilor; dacă nu se menționează pasul valoarea implicită este 10, iar dacă nu se specifică nici numărul liniei atunci valoarea implicită va fi aceea a liniei curente.</p> <p>AUTO este dezactivat dacă numărul liniei mai mic de 10 sau mai mare de 9983, ori după orice mesaj de eroare. Se iese din AUTO apăsînd BREAK mai mult de o secundă.</p>

Exemple **AUTO** numerotează de la linia curentă +10 cu pasul 10;
AUTO 100 numerotează de la linia 100 cu pasul 10;
AUTO 100,5 numerotează de la linia 100 cu pasul 5.

17.2.3. **CLOCK**

Obținere: **CS+9 și tasta C**

Sintaxa: **CLOCK număr sau șir**

Efect: afișează timpul curent în colțul din dreapta sus a ecranului (în ore, minute, secunde); când se atinge o oră prefixată se lansează un semnal sonor și / sau se poate executa o subrutină prin **GOSUB**.

Ceasul va merge în timpul rulărilor și editărilor, dar se va opri în timpul lucrului cu casetofonul, micro-drive-ul și în timpul execuției unui **BEEP**.

Pentru alegerea modului de lucru **CLOCK** va urma de o expresie numerică având valori în intervalul [0;7], care va seta ceasul conform datelor din tabelul următor;

Mod	Cheamă subrutină	Alarmă audibilă	Afișaj
0	-	-	-
1	-	-	da
2	-	-	-
3	-	-	da
4	da	-	-
5	-	-	da
6	da	da	-
7	da	da	da

Ceasul va porni din momentul lansării **BETA BASIC**-ului începînd cu ora 00:00:00; ora se modifică prin comenzi de forma **CLOCK "10:34:00"** (nu este nevoie de separatorul ":" deoarece **CLOCK** va lua în considerare primele 6 cifre întîlnite; dacă sînt mai puțin de 6 cifre, cele neexistente sînt considerate zero)

Exemple: **CLOCK** 3 va afișa ora curentă, sună și cheamă o subrutină

CLOCK "A06:20" va seta alarma pentru ora 6 și 20 minute, iar cînd se va ajunge la ora definită va lansa o alarmă sonoră dacă s-a tastat **CLOCK 2; 3; 6** sau **7**; dacă s-a tastat **CLOCK 4; 5; 6; 7** și numai dacă programul se rulează se poate apela o subrutină prin **GOSUB** (nu și în timpul editării programului).

Observație : este posibilă și forma **CLOCK : comandă, comandă, ...: RETURN** cînd, la ora stabilită pentru alarmă, este executat programul între **CLOCK** și **RETURN**.

17.2.4. CLS

Sintaxă:	CLS [nr.fereastră]
Efect:	șterge o fereastră din ecran (v. instrucțiunea WINDOW)
Exemplu:	CLS 2 (șterge fereastra 2)

17.2.5. CSIZE

Obținere:	CS+9 și SHIFT 8
Sintaxa:	CSIZE lățime, înălțime (în pixeli)
Efect:	modifică dimensiunile caracterelor din ecran
Exemple:	CSIZE 8,16 face caracterul de 2 ori mai înalt CSIZE 0 aduce caracterele la dimensiunea normală (8x8)

Exemplul 17.2 : folosirea instrucțiunii CSIZE pentru crearea de generice

```

5 BORDER 1: PAPER 1: INK 6: CLS
10 FOR f=0 TO 60 STEP 5: BEEP .1,f: NEXT f
15 FOR y=175 TO 170 STEP -1: PLOT 0,y: INK 5: DRAW
  255,0: NEXT y
20 FOR y=166 TO 163 STEP -1: INK 3: PLOT 0,y: DRAW
  255,0: NEXT y

```

```

25 PLOT 0,159: DRAW 255,0
30 CSIZE 11,20: PRINT AR 18,1; INK 6;"M.M.POPOVICI
SOFTWARE": PAUSE 30
40 CSIZE 32,50: PRINT AT 11,1;"LAGIRE"
50 FOR y=25 TO 20 STEP -1: PLOT 0,y: INK 5: DRAW
255,0: NEXT y
60 FOR y=16 TO 13 STEP -1: INK 3: PLOT 0,y: DRAW
255,0:NEXT y
70 PLOT 0,9: DRAW 255,0
80 CSIZE 0

```

17.2.6. DEFAULT

Obținere: CS+9 și SHIFT 2

Sintaxa: DEFAULT var=expresie,(var=expr)...

Efect: dacă nu există variabilă atunci DEFAULT înseamnă LET var=expresie

17.2.7. DEF KEY

Obținere: CS +9 și SHIFT 1

Sintaxa: DEF KEY șir de un caracter; șir

DEF KEY șir de un caracter; instrucțiune;instrucțiune:

Efect: afișarea unui șir sau a unor linii de program de orice lungime apăsând o singură tastă, caracterele afișate fiind memorate în calculator ori rămânând în partea de sus a ecranului pînă se apasă ENTER. Ultima variantă se obține tastînd ca ultim caracter al șirului ";", sau ultima linie de program fiind urmată de ":"

Exemplu: 5 DEF KEY "a" PRINT "BETA BASIC"

În acest exemplu ceea ce urmează după **DEF KEY "a"** este asignat tastei "a" și tastînd **GO TO 5** instrucțiunea nu se va executa imediat. Deoarece ultima instrucțiune nu este urmată de separatorul ":", se apasă mai întîi simultan Simbol Shift și Space (cînd se obține drept cursor semnul asterisc) și apoi tasta "a", instrucțiunea fiind acum executată.

Observații : 1) O tastă poate fi asignată în orice moment, vechea valoare fiind ignorată. Dacă se folosește în definiție un șir nul ori o instrucțiune vidă, atunci tasta nu va avea nici o definiție.

2) **DEF KEY ERASE** va șterge definițiile tastelor care sînt memorate deasupra RAMTOP-ului și care nu se șterg cu **NEW**.

17.2.8. DEF PROC

Obținere: CS+9 și tasta 1

Sintaxa: DEF PROC nume

Efect: permite definirea unui nume de procedură (v.**PROC**).
Instrucțiunea DEF PROC trebuie să fie primul cuvînt cheie dintr-o linie program, iar numele procedurii trebuie să aibă ca prim caracter o literă.

Exemplu: 5 DEF PROC init

O procedură poate avea același nume cu o variabilă fără a exista riscul unei confuzii.

17.2.9. DELETE

Obținere: CS+9 și tasta 7

Sintaxa: DELETE [nr.linie] TO [nr.linie]

Efect: șterge din program toate liniile din blocul specificat ; dacă "nr. linie" dinaintea lui TO nu este specificat atunci se va lua în considerare prima linie de după linia 0, iar dacă lipsește celălalt "nr.linie" se va considera ultima linie din program.

Exemple: **DELETE TO 100** - șterge toate liniile de după linia 0 inclusiv linia 100

DELETE 100 TO - șterge toate liniile începînd cu linia 100

DELETE 100 TO 100 - șterge numai linia 100

DELETE 0 TO 0 - șterge numai linia 0

DELETE TO - șterge tot programul exceptînd linia 0 (diferă de **NEW** prin aceea că nu face **CLEAR** asupra variabilelor)

Se menționează că în sintaxa instrucțiunii **DELETE** oricare din "nr.linie" trebuie specificat ; în caz contrar se afișează mesajul de eroare
U: No such line

17.2.10. DO

Obținere: CS+9 și tasta D

Sintaxa: DO WHILE condiție

DO UNTIL condiție (v.LOOP,EXIT IF)

Efect: DO și LOOP, împreună cu WHILE și UNTIL, oferă o structură de control asemănătoare buclelor FOR-NEXT; DO singur servește ca marker la care o comandă LOOP să sară înapoi.

Dacă condiția este îndeplinită,atunci liniile de program de după DO vor fi executate pînă la întîlnirea unui LOOP, iar dacă nu este îndeplinită atunci următoarea parte a programului este ignorată pînă la întîlnirea unui LOOP și execuția continuă cu instrucțiunea imediat următoare lui LOOP.

La sintaxa DO UNTIL conditia comandă este opusă adică partea de program dintre DO și LOOP va fi executată doar dacă această condiție nu este îndeplinită .

Exemplu:

```
10 LET TOTAL=0
20 DO UNTIL TOTAL 100
30 INPUT "TASTATI UN NUMAR", X
40 LET TOTAL=TOTAL+X: PRINT TOTAL
50 LOOP
60 PRINT "PESTE 100"
```

Tastînd pentru x valoarea 10 se afișează 10 și, reluînd programul, pentru x=110 se afișează "PESTE 100"

Observații : 1) Perechile DO-LOOP pot fi imbricate ca și ciclurile FOR-NEXT.

2) Deoarece adresa lui DO este memorată, nu trebuie ieșit în afara ciclului DO-LOOP decît cu EXIT IF sau POP (altfel adresa memorată va încărca inutil stiva calculatorului).

3) Dacă DO nu este urmat de LOOP se afișează mesajul de eroare "Missing LOOP".

17.2.11. DPOKE

Obținere: CS+9 și tasta P

Sintaxa: DPOKE adresă,număr

Efect: DPOKE înseamnă Double Poke și este echivalent în BASIC cu POKE adresă, număr-
 $INT(număr/256) * 256 : POKE$
 adresă+1, $INT(număr/256)$; altfel spus asupra celui mai puțin semnificativ bit al numărului (LSB) se face POKE la "adresă", iar asupra celui mai semnificativ bit (MSB) se face POKE la adresa umătoare (adresa +1). Funcția DPEEK va fi la fel ca DPOKE.

17.2.12. EDIT

Obținere: tasta 0 (nu modul grafic)

Sintaxa: EDIT [nr.linie]

EDIT a\$

EDIT a

Efect: aduce linia specificată în spațiul de editare (se intră în EDIT dacă prima tastă apăsată după ENTER este 0); dacă lipsește numărul liniei, va fi afișată pentru editare linia curentă.

Forma EDIT a\$ sau EDIT a aduce în spațiul de editare variabila, aceasta putînd fi modificată exact ca într-un INPUT.

17.2.13. ELSE

Obținere: CS+9 și tasta E

Sintaxa: ELSE : comandă:comandă:

Efect: ELSE face parte din structura IF-THEN:ELSE și anume dacă nu este adevărată condiția verificată de IF atunci este executată comanda ce urmează după ELSE, iar în caz contrar este executată comanda ce urmează după THEN, ELSE fiind ignorată.

Exemplu:

```
10 INPUT "Scrieti un numar",X
20 PRINT "Este acest numar egal cu 1 ?"
30 PAUSE 50
40 IF X=1 THEN PRINT "DA": ELSE PRINT "NU"
50 PAUSE -: CLS : GO TO 10
```

Instrucțiunea ELSE este întotdeauna precedată de separatorul ":".

17.2.14. END PROC

Obținere: CS+9 și tasta 3

Sintaxa: END PROC

Efect: marchează sfârșitul unei proceduri cu nume, permițând calculatorului să încheie comenzile între **DEF PROC** și **END PROC**.

Cînd o procedură este în execuție **END PROC** întoarce programul la comanda ce urmează după **PROC**. Folosirea lui **END PROC** fără **DEF PROC** determină mesajul de eroare W: "Missing DEF PROC".

17.2.15. EXIT IF

Obținere: CS+9 și tasta I

Sintaxa: EXIT IF condiție

Efect: face parte din structura DO-LOOP și este folosită pentru a părăsi ciclul DO LOOP în care se află, în orice punct din ciclu.

Dacă "condiție" este îndeplinită, programul execută un salt la comanda următoare lui LOOP.

Exemplu:

```
10 DO : PRINT "LINIA 10": PAUSE 30
20 EXIT IF INKEY$="S"
30 PRINT "LINIA 30": PAUSE 30: LOOP
40 PRINT "IESIRE DIN LOOP"
```

Programul va cicla pînă se apasă "S" (linia 30 nu se afișează cînd s-a ieșit din ciclu).

17.2.16. FILL

Obținere: CS+9 și tasta F

Sintaxa: FILL x,y

FILL [INK culoare]; x,y

FILL [PAPER culoare]; x,y

Efect: umple o zonă de PAPER cu INK dacă s-au folosit primele două forme , sau umple o zonă de INK cu PAPER dacă s-a folosit ultima formă, pornind de la coordonatele x, y. Se poate omite numărul culorii din specificare, considerându-se culoarea INK-ului sau a PAPER-ului curent (ex: FILL PAPER ; x, y)

Exemplu:

```

10 FOR N=1 TO 6
20 CLS
30 CIRCLE INK N;128,88,N*10
40 FILL INK N; 128,88
50 NEXT N

```

Evident că poate fi folosit și o formă mai completă a lui FILL :

FILL INK culoare a: PAPER culoare b; FLASH c; x, y
 (unde a,b={0, 7} și c={0, 1}).

O comandă FILL poate fi întreruptă folosind BREAK.

17.2.17. GET

Obținere: CS+9 și tasta G

Sintaxa: GET variabila numerică (adică GET a)

GET variabilă sir (adică GET a\$)

Efect: reprezintă, ca și INKEY\$, un mod de a citi tastatura, cu mențiunea că GET așteaptă pînă ce este apăsată o tastă înainte de a continua programul. La forma GET a, variabila numerică are valoarea $1 < a < 9$, iar la forma GET a\$ valoarea va fi 10 pentru "A", 11 pentru "B" etc.

Exemple: 10 GET a: PRINT a (afișează 1 dacă se tastează 1, respectiv 10 dacă se tastează A)

10 GET a\$:PRINT a\$ (afișează A dacă se tastează A, respectiv 1 dacă se tastează 1).

17.2.18. JOIN

Obținere: CS+9 și SHIFT 6

Sintaxa: JOIN [nr.linie]

Efect: concatenează (unește) linia specificată cu cea următoare (care își pierde numărul de linie), separarea celor două linii fiind făcută prin caracterul ":". Dacă linia nu este specificată atunci se consideră linia corespunzătoare pointer-ului de linii curent .

17.2.19. KEY IN

Obținere: CS+9 și SHIFT 4

Sintaxa: KEY IN șir de caractere

Efect: introduce în calculator un șir de caractere ca și cum ar fi fost tastate. Nu se folosește ca o comandă directă .

17.2.20. KEYWORDS

Obținere: CS+9 și tasta 8

Sintaxa: KEYWORDS 1

KEYWORDS 0

Efect: controlează afișarea UDG-ilor prin KEYWORDS 0, sau cuvintele cheie ale BETA BASIC-ului prin KEYWORDS 1. Inițial sistemul se află în starea KEYWORDS 1, iar dacă se dorește folosirea UDG-urilor se execută KEYWORDS 0; în ultimul caz se poate rămâne un timp nedefinit în modul grafic, în vreme ce din starea KEYWORDS 1 se iese după orice cuvânt cheie tastat.

17.2.21. LIST,LLIST

Obținere: în mod obișnuit

Sintaxa: LIST nr.linie TO nr.linie

LLIST nr.linie TO nr.linie

Efect: blocul specificat de linii va fi **LIST**-at (pe ecran) sau **LLIST**-at (la imprimantă). Primul sau al doilea număr de linie poate fi omis, în primul caz luându-se implicit prima linie după linia 0, iar în al doilea caz se ia implicit ultima linie a programului.

Exemple: **LIST 20 TO 100** (afișează blocul de linii 20-100)
LIST TO 200 (afișează blocul de linii 1-200)
LIST 100 TO 100 (afișează numai linia 100).

17.2.22. LOOP

Obținere: CS+9 și tasta L

Sintaxa: **LOOP**
LOOP WHILE condiție
LOOP UNTIL condiție

Efect: face parte din structura **DO-LOOP**; **LOOP** singur face ca programul să execute un salt înapoi la instrucțiunea **DO** corespunzătoare. Parametrii de condiție **WHILE** și **UNTIL** permit saltul condiționat, respectiv **LOOP WHILE** condiție face salt numai dacă această condiție este adevărată, iar **LOOP UNTIL** face saltul numai dacă condiția este falsă. Folosirea lui **LOOP** fără **DO** conduce la afișarea mesajului de eroare T: "LOOP without DO"

17.2.23. ON

Obținere: CS+9 și tasta 0

Sintaxa: **GO TO ON x; nr.linie,...** (sau **GOSUB ON x; nr.linie,...**)
ON x; comandă: comandă:...

Efect : la prima formă transferă controlul la o anumită linie în funcție de valoarea lui x, iar a doua formă execută comanda corespunzătoare lui x (prima pt.x=1, a doua pt.x=2, ș.a.m.d.).

Exemplu: 10 PRINT CHOICE: GO TO ON CHOICE:
 90,135,60,50
 20 PRINT "ALEGETI DE LA 1 LA 4": GO TO 10
 Linia 90 va fi apelată dacă s-a ales 1, linia 135 dacă s-a ales 2, ș.a.m.d. Dacă numărul tastat este negativ el se consideră în valoare absolută, iar dacă nu face parte din intervalul 1-4, nu se face salt la nici o linie, dar se trece la linia următoare.

17.2.24. ON ERROR

Obținere: CS+9 și tasta N

Sintaxa: ON ERROR nr.linie

ON ERROR :comandă : ...:RETURN (execută programul între ON ERROR și RETURN la detectarea unei erori)

Efect: produce saltul programului la linia cerută atunci când este detectată o eroare în timpul programului. O eroare ce trebuie tratată diferit este eroarea 21 (adică BREAK) cu mesajul "BREAK into program" conform exemplului dat.

Exemplu: 100 ON ERROR 5000

200 PRINT "SI IAR";:PAUSE 10:GO TO 200
 4999 STOP

5000 IF ERROR=21 THEN BEEP 1,69: BORDER
 RND*7: RETURN : ELSE POP : CONTINUE

(RETURN face revenirea la programul principal, iar POP reface adresa de revenire în program)

17.2.25. PLOT

Obținere: în mod obișnuit

Sintaxa: PLOT x, y, [; a\$]

Efect: Afișează un șir de caractere din orice punct de coordonate x,y al ecranului (coordonatele se referă la colțul din stînga sus al primului caracter al șirului de afișat). Se pot folosi și instrucțiuni ajutoare cum sînt **INVERSE**, **OVER**, **INK**, e.t.c. Dacă șirul se extinde în afara părții drepte a ecranului, se va înfășura în partea stîngă a acestuia (coordonatele se dau în pixeli).

Exemplu: `10 FOR I=16 TO 244: PLOT X,X/2;"< >": NEXT X`

Observații : 1) Poziția folosită de **DRAW** (prin **PLOT**) ca punct de plecare a unei linii drepte nu este afectată de **PLOT** cu șiruri.

2) Cursorii de control (**CHR\$8...CHR\$11**) pot fi incluși în comenzile **PLOT** cu șiruri.

17.2.26. POKE

Obținere: în mod obișnuit

Sintaxa: `POKE adresă,a$`

Efect: permite memorarea prin **POKE** a șirului, iar în combinația cu **MEMORY\$** permite manipularea rapidă a unor largi zone de memorie.

Exemplu: `10 LET SCREEN=16384`
`20 POKE SCREEN,STRING$(6144,"U")`
 (unde **STRING\$** este explicat la paragraful funcțiilor).
 Memoria ecran va fi umplută cu "U"-ri care apar ca modele, ecranul fiind dungat vertical.

Exemplul 17.3 : programul copiază începutul ROM-ului în zona atributelor, producînd efecte coloristice.

```
30 LET ATTRIBUTE=22528
40 POKE ATTRIBUTE, MEMORY$( ) (1 TO 704)
```

Exemplul 17.4 : programul următor construiește o figură memorată într-un șir și reafiată cu **POKE**.

```

10 CIRCLE 128,88,70
20 FILL 128,88
30 LET A$= MEMORY$( )(16384 TO 23295): REM întreg
   ecranul
40 CLS : PRINT "APASATI ORICE TASTA": PAUSE 0
50 POKE 16384,A$

```

Observații: 1) Pentru a se obține mai multe imagini afișate secvențial, o treime a ecranului poate fi memorată într-un șir după cum urmează:

- partea superioară : `LET A$=MEMORY$()(16384 TO 18431)`
- partea mijlocie : `LET A$=MEMORY$()(18432 TO 20479)`
- partea inferioară : `LET A$=MEMORY$()(20480 TO 22537).`

Pe această bază se poate realiza animația folosind POKE-uri secvențiale ale acestor structuri (se poate folosi o zonă, de ex: `DIM A$(10,2348)` pentru a menține datele).

2) Se poate face **CLEAR** pe o zonă de la adresele mari ale memoriei și apoi memorarea în acel loc a unui program. Astfel se face **CLEAR 33900** și se tastează programul :

```

10 POKE 34000, MEMORY$( )(23552 TO 33800 )
20 REM restul programului

```

Acum se poate folosi **NEW** urmînd încărcarea programului cu `POKE 23552, MEMORY$()(34000 TO 44248)`

Această linie poate fi ascunsă de un **NEW** printr-o tastă utilizator (folosită în UDG), care să fie definită după **CLEAR** și înainte de rularea programului :

```

DEF KEY "J": POKE 23552, MEMORY$( )(3400 TO 44248)

```

Cînd programul va fi reîncărcat cu **POKE**, va continua să ruleze din punctul unde a fost întrerupt, deoarece toate variabilele sistem au fost salvate la fel ca și variabilele normale.

17.2.27. POP

Obținere: CS+9 și tasta Q
Sintaxa: POP [variabilă numerică]
Efect: mută adresele din stiva GOSUB/DO-LOOP/PROC ; numărul liniei indicat este asignat variabilei numerice dacă ea există. Instrucțiunea face posibil saltul din subrutine, cicluri DO-LOOP și proceduri DEF PROC fără a încărca stiva cu adrese nefolosite. Folosit singur POP coboară doar pointer-ul stivei eliberând virful acesteia, iar POP LOC face utilizabilă valoarea din virful stivei, ca variabilă "LOC".

Exemplu: 100 GOSUB 500
 110 STOP
 500 POP LOC
 510 PRINT "SUBRUTINA APELATA DIN LINIA
 ";LOC
 520 GO TO LOC+1

Înlocuind linia 520 cu RETURN se va afișa mesajul de eroare "RETURN without GOSUB", deoarece se cere o adresă de revenire care nu mai este prezentă în stivă.

Folosirea instrucțiunii POP fără a exista date în stivă determină afișarea erorii V, "No POP date".

17.2.28. PROC

Obținere: CS+9 și tasta 2
Sintaxa: PROC nume
Efect: este un fel de GOSUB cu nume și trebuie să fie prima instrucțiune dintr-o linie de procedură. Numele procedurii poate fi orice nume valid de variabilă, iar tipul literelor (mari sau mici) nu contează.

DEFiniția PROCedurii poate avea oricâte linii de program și poate fi terminată în orice punct cu **END PROC**. Evident că **DEF PROC** este ignorat dacă nu este apelat cu **PROC**. Toate variabilele folosite de programul principal sînt utilizabile de către proceduri și reciproc.

Exemplu:

```

10 FOR N=1 TO 20: PROC DESENAREA PATRAT:
NEXT N
200 DEF PROC DESENEAZA PATRAT: LET
L=RND*20+20
210 PLOT RND*215, RND*135: DRAW L,0: DRAW
0,L: DRAW -L,0: DRAW 0,-L: END PROC
220 PRINT "TERMINAT !"
```

17.2.29. RENUM

Obținere: CS+9 și tasta 4

Sintaxa: RENUM [(linie start TO linie sfîrșit)][LINE linie nouă]
[STEP pas]

Efect: efectuează renumerotarea liniilor programului, avînd efect și asupra **GO TO** și **GOSUB**. Folosită singură, instrucțiunea **RENUM** renumerotează întreg programul începînd cu 10 pentru prima linie și continuînd cu pasul 10. Renumerotarea unui bloc de linii se face specificînd după **RENUM** intervalul ce se dorește numerotat. Pentru a se obține alt increment pentru linii se folosește **STEP**. Instrucțiunile **LINE** și **STEP** pot fi utilizate sau omise, dar ele trebuie să respecte ordinea indicată de sintaxă.

Exemple:

```

RENUM (100 TO 200) -renumerotează blocul specificat
RENUM (130 TO) -renumerotează linia 130 și restul
RENUM (TO 100) -renumerotează toate liniile pînă la 100
inclusiv, exceptînd linia 0
RENUM -renumerotează întreg programul
RENUM LINE 120 STEP 20 -renumerotează de la linia
100 cu pasul 20
RENUM (1540 TO) LINE 2000 -renumerotează blocul
de la linia 1540 pînă la sfîrșit, prima linie fiind acum 2000
```

RENUM (100 TO 176)LINE 230 STEP 5 -
renumerotează între liniile 100 și 176, prima linie fiind notată 230 și pasul 5

Referințele la blocurile numerotate sînt modificate, alături de GO TO și GOSUB și în instrucțiunile RESTORE, ON, ON ERROR, TRACE, LIST, LLIST, LINE și DELETE. Doar CLOCK nu este luat în considerare și deci aici corecția trebuie făcută manual.

17.2.30. ROLL

Obținere: CS+9 și tasta R

Sintaxa: ROLL cod-direcție [,pixeli][;x,y; lungime,înălțime]unde codul direcție rezultă din tabelul următor:

Cod direcție	Direcție	Se aplică pentru :
1	stînga	atribute
2	jos	atribute
3	sus	atribute
4	dreapta	atribute
5	stînga	caractere
6	jos	caractere
7	sus	caractere
8	dreapta	caractere
9	stînga	amebele
10	jos	amebele
11	sus	amebele
12	dreapta	amebele

Efect: ecranul sau o fereastră predefinită sînt mutate în sus, în jos, la dreapta sau la stînga. Orice mutare în afara ferestrei definite de ROLL va apare în partea opusă a ecranului. Pentru mișcarea cu un pixel se scrie instrucțiunea **ROLL cod-direcție** iar pentru o mișcare mare este recomandabil ca ROLL să fie într-un ciclu .

Mișcarea în diagonală se obține folosind succesiv **ROLL** cu un pixel la stînga și unul în sus (parametrul "înălțime" trebuie să fie mai mic de 256 pentru mișcări orizontale și mai mic de 176 pentru mișcări verticale, iar dacă nu este specificat se consideră implicit 1).

În mișcarea verticală viteza este proporțională cu numărul de pixeli de mișcat, iar pe orizontală mișcarea de 4 și 8 pixeli dă cea mai bună viteză.

O fereastră specifică ecranului poate fi mișcată dacă direcție este urmat de 4 parametri : coordonatele x, y (în pixeli) ale colțului din stînga sus a ferestrei, apoi *lungimea ferestrei* (în caractere) și *înălțimea ei* (în pixeli) .

Exemple:

a) după trasarea unui grafic se poate tasta programul

```
100 FOR d=5 TO 8: FOR p=1 TO 100
110 ROLL D
120 NEXT P: NEXT D
```

Schimbînd linia 110 în 110 ROLL D,4 imaginea va fi mutată cu 4 pixeli deodată

b) deplasarea ecranului spre stînga și apoi în jos

```
5 BORDER 5: PAPER 5: INK 1: CLS
10 CSIZE 32,50: PRINT AT 0,1;"ARBORI":
PAUSE 10
20 CSIZE 11,20: PRINT AT
17,1;"M.M.POPOVICI SOFTWARE": PAUSE 100
30 FOR i=0 TO 255: ROLL 5: NEXT i
40 FOR i=0 TO 125: ROLL 6: NEXT i
50 CSIZE 0
```

Exemplul 17.5 : efecte interesante se pot obține intersectînd mișcarea a două sau mai multe ferestre, ecranul fiind aparent răsucit și rupt în figuri bizare ; astfel programul următor rupe în patru listing-ul (sau orice imagine).

```
100 LIST
110 LET PIXEL=4
120 ROLL 5,PIXEL;0,175;32,68
130 ROLL 6,PIXEL;0,175,16,176
```

140 ROLL 7,PIXEL;0,87;32,88
 150 ROLL 8;PIXEL;128,176,16,176
 160 GO TO 120

17.2.31. SCROLL

Obținere: CS+9 și tasta S

Sintaxa: SCROLL cod-direcție [,pixeli][;x,y;lungime,inălțime]

Efect: mută ecranul (ca și ROLL) cu deosebirea că orice ieșire din ecran se pierde.

Exemplu:

```
100 LET A$="UN SIR FOARTE LUNG.."
110 FOR C=1 TO LEN A$
120 PRINT AT TO 10,31; INK 7;A$(C)
130 FOR P=1 TO 8: SCROLL 5;0,95,32,6: NEXT
    P
140 NEXT C
150 FOR P=1 TO 255: SCROLL 5;0,95;32,8:
    NEXT P
```

Linia 120 poate fi înlocuită prin

```
120 PLOT 148,95;A$(C)
```

17.2.32. SORT

Obținere: CS+9 și tasta M

Sintaxa: SORT

SORT A\$ (porțiune)

SORT INVERSE tablou de șiruri sau tablou numeric sau șiruri.

Ei. xt: aranjează șirurile, numerele sau literele în ordine crescătoare (cu INVERSE) sau descrescătoare. Șirurile sînt sortate după codul caracterelor. Se pot sorta blocuri (porțiuni) din tabloul de șiruri care să fie sortate [ex: SORT A\$ (1 TO 20) sortează primele 20 șiruri; SORT A\$(30 TO) sortează toate șirurile începînd cu al 30-lea; SORT A\$(2 TO) sortează întregul tablou de șiruri fiind supuse sortării al doilea și următoarele caractere ale fiecărui șir].

Exemplu: programul generează un tablou de 5 șiruri a câte 5 caractere fiecare (viteza programului crește folosind funcția RNDM):

```
110 DIM A$(5,5)
110 FOR S=1 TO 5: FOR L=1 TO 5
120 LET A$(S,L)=CHR$(RNDM*65+25)
130 NEXT L: NEXT S: GO TO 200
140 SORT A$
200 FOR S=1 TO 5: PRINT A$(S):NEXT S
```

Îndată ce tabloul a fost creat el va fi afișat .Făcînd GO TO 140 tabloul va fi sortat și afișat din nou (se va evita RUN pentru a nu se pierde tabloul). Modificînd linia 140 în **SORT INVERSE A\$** acțiunea de sortare va fi inversată.

Obsevații : 1) **SORT** face posibilă o dezvoltare rapidă și flexibilă a unei baze de date. Astfel se poate denumi tabloul ca "FISIER", iar șirurile lui ca "INREGISTRARI". Zonele fiecărui șir vor fi rezervate pentru diferite tipuri de informații și se vor numi "CIMPURI". De pildă, un fișier de nume, adrese și alte date ar putea fi alcătuit astfel: primele 20 caractere ale înregistrării ar fi numele persoanei, următoarele 20 caractere adresa (sau profesia) acestuia, ultimul caracter (al 41-lea) fiind vîrsta . Deoarece vîrsta trebuie să fie în intervalul de coduri 1-256, se poate folosi instrucțiunea

```
LET A$(S,41)=CHR$ vîrsta
```

plasîndu-se informația "vîrsta" în înregistrarea S.

2) Dacă trebuie memorate date mai complexe, cum ar fi balanța unei bănci, pentru a forma cîmpuri astfel încît unitățile, zecile și sutele să ocupe același loc în fiecare șir, se folosește funcția **USING\$** (v.par. 5.3):

```
LET A$(S,41 TO 46)=USING$ ("000.00",balan;a)
```

3) **SORT** funcționează și cu șiruri simple sau tablouri unidimensionale de șiruri. De ex:

```
INPUT S$: SORT S$: PRINT S$
```

va afișa "BFdeggors" dacă s-a tastat "Fred Bloggs).

4) Un tablou numeric bidimensional poate fi considerat ca avînd prima dimensiune linia și a doua coloana. De

exemplu :

SORT B(1 TO 20) (2)

va sorta primele 20 de linii ale lui B după numerele din a doua coloană. Se menționează că întotdeauna este nevoie de cel puțin o pereche de paranteze într-un **SORT** numeric, pentru a distinge tabloul B() de variabile B. A doua pereche de paranteze, dacă este folosită, va conține doar o valoare, exceptând cazul cu șiruri.

17.2.33. SPLIT

Obținere:	SS+4 (se tastează ca " < > ")
Sintaxa:	< >
Efect:	într-o linie tot ce urmează după " < > " rămâne în lucru în zona de editare, iar tot ce este înainte de " < > " va apare în listing.
Exemplu:	10 PRINT "HELLO": GO TO 10: < > "NU APARE" În listing apare: 10 PRINT "HELLO": GO TO 10, iar 10 < > PRINT "NU APARE" va rămâne în zona de editare.

17.2.34. TRACE

Obținere:	CS+9 și tasta T
Sintaxa:	TRACE nr.linie
Efect:	permite depanarea programelor BASIC afișând linia curentă, instrucțiunea din linie, starea și variabilele selectate și reduce viteza de execuție sau permite rularea pas cu pas. TRACE generează un apel tip <u>GOSUB</u> la o rutină specifică, imediat înaintea executării oricărei instrucțiuni, care utilizează variabilele LINE și STAT reprezentând numărul liniei și numărul instrucțiunii din linia respectivă a unei părți de program în curs de execuție . TRACE este dezactivat în timpul execuției subrutinei proprii de depanare și este reactivată la întâlnirea instrucțiunii RETURN din subrutină.

Exemplu: 9000 PRINT INVERSE 1; LINE ;":"; STAT
:RETURN

Se adaugă linia 9000 la un program și se inserează comanda **TRACE 9000** într-un punct unde se dorește depanarea.

Pentru dezactivare într-un punct al programului se tastează **TRACE 0**. Dacă se dorește vederea textului liniei executate, se include instrucțiunea

LIST LINE TO LINE

sau **LIST LINE-1 TO LINE** . Este permisă și forma **TRACE:comandă:.....:RETURN**, când este executat programul dintre **TRACE** și **RETURN**.

17.2.35. UNTIL

Obținere: CS+9 și tasta K

Sintaxa: UNTIL condiție (folosită în LOOP UNTIL condiție)
DO UNTIL condiție

Efect: permite executarea condiționată a lui DO și LOOP (v.DO)

17.2.36. USING

Obținere: CS+9 și tasta U

Sintaxa: PRINT USING format număr ; număr

Efect: atât USING cât și USING\$ permit specificarea cu format a numerelor ce trebuie afișate (aceasta este indicat într-un șir de caractere în care "#" înseamnă blank).

Exemplu: 100 FOR N=1 TO 20:LET X=RND+100

110 PRINT X,USING "###.##":NEXT N

Instrucțiunea USING rotunjește numărul la cel mai apropiat digit.

Observație : separatorul "#" este utilizat pentru spațiile directoare iar pentru zerourile directoare se poate folosi "0"; ambele semne pot fi

utilizate pentru numărul de cifre după punctul zecimal. De pildă pentru numărul 12,345 se pot folosi formele următoare cu rezultatele indicate :

```
"#.#.#" :      12,3
"#.#.#" :      12,3
"###.#.#" :    12,35
"000.00" :    012,35
"00" :        12
```

17.2.37. WHILE

Obținere:	CS+9 și tasta J
Sintaxa:	WHILE condiție
Efect:	permite execuția condiționată a lui DO și LOOP (v.D0)

17.2.38. WINDOW

Obținere:	CS+9 și tasta 5
Sintaxa:	WINDOW indice,[x,y, lungime,înălțime] (toate în pixeli iar indice={1 ;127})
Efect:	definește o fereastră (unde x,y sînt coordonatele colțului stînga sus, iar <i>lungimea</i> și <i>înălțimea</i> sînt dimensiunile ei) <u>WINDOW 0</u> reprezintă întreg ecranul.
Exemple:	WINDOW 1, 16, 111, 224, 80 WINDOW 2, 80, 31, 80, 32

17.2.39. XOS, XRG, YOS, YRG

Aceste 4 instrucțiuni nu sînt cuvinte cheie ci niște variabile speciale care permit schimbarea scalei și a originii folosite de **PLOT**, **DRAW**, **CIRCLE** și **FILL** (unde XOS=originea pentru axa X; YOS=originea pentru axa Y; XRG, YRG=scala pentru axele X și Y).

Ele sînt speciale deoarece **CLEAR** și **RUN** nu le șterg ci le inițializează la valori particulare.

Astfel, dacă se face **CLEAR** și apoi **PRINT XOS, YOS** se va afișa 0 întrucît XOS și YOS au inițial valoarea 0; ele pot fi însă setate la o altă

valoare cu instrucțiunea LET). Normal $XRG = 256$ și $YRG = 176$.

Exemple: a) **LET XOS=128:LET YOS=88** (se mută originea în centrul ecranului, ceea ce permite PLOT cu $-128 < x < 127$ și $-88 < y < 87$)

b) Schimbînd XRG și YRG se modifică scala cu care se lucrează; astfel programul următor desenează un pătrat mai întii normal, apoi alungit după axa X, apoi după X și în Y și în final numai după axa Y:

```
10 GOSUB 100: REM normal
20 LET XRG=128: GOSUB 100
30 LET YRG=88: GOSUB 100
40 LET XRG=256: GOSUB 100: STOP
100 CLS : PLOT 0,0: DRAW 50,0: DRAW 0,50: DRAW
    -50,0: DRAW 0,-50: PAUSE 100: RETURN
```

17.3. FUNCȚII BETA BASIC

Un număr de 21 funcții au fost adăugate în BETA BASIC (a se vedea linia 0). Aceste funcții sînt tratate ca funcții definite de utilizator și vor funcționa numai în prezența BETA BASIC-ului deci cu linia 0 și codul mașină). Din acest motiv salvarea unui program se face cu linia 0 și, bineînțeles, cu codul mașină aferent programului. Pentru a folosi un program BASIC care se dorește a fi adaptat pentru BETA BASIC, el se va încărca cu **MERGE** (nu cu **LOAD** care va șterge linia 0), respectiv se va tasta **NEW** pentru a șterge un program existent (care în BETA BASIC nu afectează linia 0) și apoi programul BASIC se încarcă cu **MERGE**.

17.3.1. AND

Obținere:	FN a(
Sintaxa:	AND (număr, număr)
Efect:	realizează un SI logic bit cu bit între două numere ce trebuie să fie în intervalul 0....65535
Exemple:	a) folosind funcția BIN\$ pentru înțelegerea operației SI logic BIN\$ (254) = "11111110"

```
BIN$ (121)="011110011"
```

```
BIN$ (AND(254,121))="01111000"
```

b) se poate folosi AND pentru a masca biții nedoriti; astfel
 PRINT AND (BIN 00000111, ATTR (linie,
 coloana))

are ca efect afișarea culorii INK pentru poziția "linie,
 coloană", prin mascarea celorlalți biți (se putea folosi 7 în
 loc de BIN 00000111).

c) programul va afișa "BANG !" dacă tasta F este
 apăsată, indiferent ce altă tastă este apăsată simultan

```
10 IF AND (BIN 00001000, IN 65022)=0 THEN  

  PRINT "BANG!"  

  20 GO TO 10
```

17.3.2. BIN\$

Obținere: FN B\$

Sintaxa: FN B\$ (număr)

Efect: furnizează echivalentul binar al unui număr ca un șir de 9
 caractere (dacă numărul este mai mic de 256), ori de 16
 caractere (dacă numărul este între 256 și 65535)

Exemplu: 10 PRINT AT 10,10;BIN\$(IN 65002):GO TO 10

Funcția BIN\$ este utilă în înțelegerea codului mașină și în
 operațiile bit cu bit ale funcțiilor AND, OR și XOR. De asemenea, poate fi
 utilă și în examinarea generatorului de caractere din ROM, a zonei UDG, a
 fișierului de atribute, a sistemului de variabile sau a tastaturii (așa cum se
 constată din exemplul dat).

17.3.3. CHAR\$

Obținere: FN C\$

Sintaxa: CHAR\$ (număr)

Efect: convertește întregii 0 la 65535 în șiruri de două caractere, permițând stocarea datelor numerice cu o sensibilă economie de memorie . Echivalentul în BASIC este LET A=INT(număr/256):LET B=număr-A*256: LET C\$=CHAR\$ A+ CHAR\$ B

Exemplu: programul următor folosește funcția NUMBER (șir) care convertește 2 caractere într-un întreg (număr cuprins între 0 și 65535) și se referă la o matrice de întregi

```
100 DIM A$ (5,2)
110 FOR E=1 TO 5: LET A$(E)=CHAR$(E*10):
    NEXT E
120 PRINT "MATRICE CREATA; APASATI ORICE
    TASTA(2sp)PENTRU AFISAREA EI": PAUSE 0
130 FOR E=1 TO 5:PRINT E,NUMBER
    (A$(E)):NEXT E
```

Se va afișa :

1	10
2	20
3	30
4	40
5	50

17.3.4. COSE

Obținere: FN c(

Sintaxa: cose (număr)

Efect: calculează cosinusul numărului indicat cu o precizie mai mică (4 cifre semnificative) dar de 6 ori mai repede .

17.3.5. DEC

Obținere: FN d(

Sintaxa: dec (hex\$)

Efect: are ca rezultat echivalentul zecimal al unui șir de caractere ce reprezintă un număr hexazecimal (nu contează dacă literele sînt mari sau mici)

Exemple: DEC ("FF")=255
DEC ("4000")=16384: DEC ("e")=14

Folosirea unui șir nul sau a caracterelor care nu fac parte din intervalele folosite în scrierea hexazecimală și anume: 0 ... 9, A ... F sau a ... f, determină afișarea mesajului "Invalid argument".

Pentru a face POKE cu caractere hexazecimale se pot folosi instrucțiunile INPUT A\$:POKE adresa,DEC(A\$).

17.3.6. DPEEK

Obținere: FN P(
Sintaxa: DPEEK (adresă)
Efect: face un PEEK dublu asupra unei adrese specificate și a următoarei. Echivalentul în BASIC este LET
VALOARE = PEEK (ADRESA) + 256 * PEEK (ADRESA + 1)
Exemplu: 100 LET NXT=DPEEK (23637): POKE NXT+5,65
110 REM x x x x x
Programul va citi adresa liniei 110 din variabila de sistem NXTLN. Primul caracter de după REM va fi modificat în "A" prin POKE (scrierea "+5" este folosită pentru a se sări peste numărul liniei, pointer-ul de linie și instrucțiunea REM)

17.3.7. FILLED

Obținere: FN F(
Sintaxa: FILLED (
Efect: afișează numărul de pixeli colorați cu ultima comandă FILL
Exemplu: 10 PLOT 0,0: DRAW 9,0: DRAW 0,9: DRAW -
9,0: DRAW 0,-9
20 FILL 3,5

30 PRINT FILLED ()

Programul desenează un patrat cu latura de 10 pixeli pe care îl umple cu culoarea INK-ului și apoi afișează cifra 64.

17.3.8. HEX\$

Obținere: FN H\$

Sintaxa: HEX\$(număr)

Efect: argumentul numeric este convertit într-un șir hexazecimal, care va avea o lungime de 2 caractere dacă numărul este cuprins în intervalul [-255;+256] și de 4 caractere dacă valoarea absolută a numărului depășește acest interval. Valorile peste 65535 conduc la afișarea mesajului "Integer out of range"

Exemple: HEX\$(32)="20"; HEX\$(255)="FF";
HEX\$(512)="0200"; HEX\$(-64)="C0"; HEX\$(-1024)="FC00"

17.3.9. INSTRING

Obținere: FN I(

Sintaxa: INSTRING(start, șir1,șir2)

Efect : caută șir2 în șir1 începînd de la caracterul *start* din șir1; dacă șirul este găsit se afișează poziția primului caracter din șir2 găsit în șir1 (altfel rezultatul este 0). Șir1 poate avea orice lungime dar șir2 trebuie să aibă sub 256 caractere (în caz contrar se afișează mesajul "Invalid argument"). Rezultatul 0 apare și dacă șir2 este mai lung decît șir1, dacă *start* este mai mare decît LEN șir1 sau dacă oricare din șiruri are LEN 0.

Exemplu: programul următor caută în șirul A\$ fiecare apariție a șirului "TEST"

```
100 DIM A$ (1000)
110 FOR N=1 TO RND*10+3
120 LET POS=RND*995
```

```

130 LET A$(POS TO POS+3)="TEST"
140 NEXT N
150 PRINT "TEST în A$-apasati orice
      tasta(2sp)pentru cautare"
160 PAUSE 0
170 LET LOC=1
180 LET LOC=INSTRING(LOC,A$,"TEST")
190 IF LOC <> 0 THEN PRINT "GASIT IN
      POZITIA(1sp)";LOC: LET LOC=LOC+1:GO TO
      180
200 PRINT "ASTA-I TOT !"

```

Șirul A\$ este căutat inițial din poziția 1(LOC=1) și secvențial de la poziția unde s-a găsit ultimul "TEST". Când INSTRING întoarce valoarea 0 înseamnă că s-au găsit toate aparițiile lui "TEST" în șirul A\$ (linia 190 se putea scrie: 190 LOC THEN PRINT "GASIT IN POZITIA ";LOC: LET LOC=LOC+1: GO TO 180, deoarece "0" este echivalent cu "NOT TRUE").

17.3.10. MEM

Obținere: FN M(
Sintaxa: MEM (
Efect: furnizează numărul de octeți de memorie liberă
Exemplu: 10 PRINT MEM(): DIM A\$ (100): PRINT
MEM()

17.3.11. MEMORY\$

Obținere: FN M\$
Sintaxa: MEMORY\$ ()

Efect: furnizează întreaga memorie sub formă de șir (locația 0 nu este inclusă astfel că instrucțiunea **CODE MEMORY\$ () (1)** este același lucru cu **PEEK 1**; de asemenea ultimii 3 octeți de memorie sînt excluși, adică funcția are **LEN 65532**),

Exemplu: **LET A\$=MEMORY\$ () (16384 TO 22527)**

17.3.12. MOD

Obținere: **FN V(**

Sintaxa: **MOD (număr1,număr2)**

Efect: efectuează împărțirea modulo a primului număr la al doilea număr (număr1 modulo număr2)

Exemple: **MOD (10,3)=1**

MOD (66,16)=2

17.3.13. NUMBER

Obținere: **FN N(**

Sintaxa: **NUMBER (șir)**

Efect: convertește un șir de două caractere într-un întreg (un număr cuprins între 0 și 65535); echivalentul său în **BASIC** este **:LET număr+256*CODE C\$(1)+CODE C\$(2)**

17.3.14. OR

Obținere: **FN O(**

Sintaxa: **OR (număr1, număr2)**

Efect: face un SAU logic bit cu bit a două numere ce trebuie să fie între 0 și 65535.

17.3.15. RNDM

Obținere: **FN R(**

Sintaxa: RNDM (număr)
Efect: Dacă "număr" este 0, atunci RNDM oferă un număr aleator cuprins între 0 și 1 (întocmai ca RND dar de 2,5 ori mai repede); dacă "număr" este diferit de 0, RNDM dă un număr aleator între 0 și acel număr inclusiv (de asemenea de 2,5 ori mai repede ca RND*număr).
Exemplu: 10 PLOT RNDM(255), RNDM(175)
 20 GO TO 10

RANDOMIZE (număr) va seta RNDM la o valoare particulară cu secvență de numere pseudoaleatoare, la fel ca și RND.

17.3.16. SCRN\$

Obținere: FN R\$(
Sintaxa: SCRN\$ (linie, coloană)
Efect: funcționează la fel ca SCREEN\$ cu excepția faptului că recunoaște și UDG-urile (blocul de caractere grafice ale calculatorului nu sînt recunoscute)

17.3.17. SINE

Obținere: FN S(
Sintaxa: SINE (număr)
Efect: oferă sinusul unui număr cu mai puțină precizie (4 zecimale) dar de 6 ori mai repede decît SIN.

7.3.18. STRING\$

Obținere: FN S\$(
Sintaxa: STRING\$ (număr, șir)
Efect: repetă de "număr" de ori "șir"-ul indicat.
Exemple: STRING\$ (32, "-") =32 de semne "-"
 STRING\$ (4, "AB")="ABABABAB"

```
PRINT STRING$ (704,"X")= un ecran plin cu
"X"
PRINT STRING$(3,"A"+CHR$ 13)=A
A
A
```

17.3.19. TIMES

Obținere: FN T\$
Sintaxa: TIMES ()
Efect: furnizează timpul curent ținut de ceasul BETA BASIC-ului, sub forma unui șir de caractere dispus în ore:minute:secunde. **TIMES** apelat de mai multe ori își va schimba constant valoarea.

17.3.20. USING\$

Obținere: FN U\$(
Sintaxa: USING\$ (șir formatat, număr)
Efect: furnizează șirul echivalent "număr-ului,formatat după specificația "șir formatat"; numărul de caractere directe sau de umplere se pot specifica, iar rotunjirea se face la cel mai apropiat digit afișat (v.USING).

17.3.21. XOR

Obținere: FN X(
Sintaxa: XOR (număr1, număr2)
Efect: face SAU EXCLUSIV bit cu bit a 2 numere ce trebuie să fie între 0 și 65535.

17.4. EXEMPLU DE UTILIZARE

Se prezintă un program care calculează media de examen a 500

de candidați la două obiecte de studiu și afișarea mediilor în ordine descrescătoare. Acest program se tastează după ce s-a încărcat versiunea 3.1.

Exemplul 17.6.

```

50 DEF PROC setw
60 WINDOW 1,16,14*8-1,10*8
80 CSIZE 8,16
90 PAPER 7: BRIGHT 1: INK 0: CLS
100 WINDOW 0
110 WINDOW 2,80,31,80,32
120 WINDOW 2: PAPER 1: INK 7: BRIGHT 1: CLS
130 WINDOW 0
140 END PROC
150 DEF PROC init
155 CSIZE 11,20
160 PRINT AT 0,1;"EXAMEN EXAMEN EXAMEN"
170 setw
180 PRINT AT 6,2: CSIZE 8,16;"POS NR.(2sp)NOTA 1
(2sp)NOTA 2(2sp)MEDIE"
190 DIM m(500,2)
200 DIM a(500)
210 DIM b(500)
220 END PROC
230 DEF PROC getn
240 CLS 2:PRINT WINDOW 2;AT 1,0;"g:greseala"
"s:sfisit"
250 LET ind=1
260 WINDOW 1
270 DO
280 PRINT AT 4,4;USING;"###"; ind
290 INPUT LINE a$
300 EXIT IF a$(1)="s"
310 EF a$(1)="g" THEN eroare: GO TO 430
320 LET a(ind)=VAL a$
330 PRINT AT 4,9;USING ;"# #.# #";a(ind)
340 INPUT a$
350 REM
360 REM

```

```
370 LET b(ind)=VAL a$
380 PRINT AT 4,16; USING ;"# #.# #";b(ind)
390 LET m(ind,2)=(a(ind)+b(ind))/2-0.003
400 LET m(ind1)=ind
410 PRINT AT 4,23; USING ;"# #.# #";m(ind,2)
420 LET ind=ind+1
430 SCROLL 7,16,14*8;28,80
440 LOOP
450 END PROC
460 DEF PROC sortare
470 SORT m(1 TO ind)(2)
480 END PROC
490 DEF PROC afisare nr
500 WINDOW 1
510 CLS
520 FOR i=nr TO nr+4
530 FOR i=nr TO nr+4
540 PRINT TAB 0; USING;"# # #";i; TAB 4; USING ;
"# # #";m(i,1); TAB 9; USING;"# #.# #";
a(m(i,1)); TAB 16; USING ;"# #.# #";b(m(i,1));
TAB 23; USING ;"# #.#.#";m(i,2)
550 NEXT i
560 END PROC
570 DEF PROC up
580 IF pos>1 THEN LET pos=pos-1: SCROLL 6,16;
16,14*8;28,80: PRINT AT 0,0; USING ;"# # #";pos;
TAB 4; USING ;"# # #";m(pos,1); TAB 9;
USING ;"# #.# #";a(m(pos,1)); TAB 16;
USING ;"# #.# #";b(m(pos,1)); TAB 23;
USING ;"# #.# #";m(pos,2)
590 END PROC
600 DEF PROC down
610 IF pos+4<ind THEN LET pos=pos+1: SCROLL 7,16;16,
14*8;28,80: PRINT AT 4,0; USING ;"# # #"; pos+4;
TAB 4; USING ;"# # #";m(pos+4,1); TAB 9; USING ;
"# #.# #";a(m(pos+4,1)); TAB 16;
USING ;"# #.# #";b(m(pos+4,1)); TAB 23;
USING ;"# #.# #";m(pos+4,2)
620 END PROC
```

```
630 DEF PROC display
640 CLS : PRINT WINDOW 2; AT 0,0;"6:jos"``"7:sus"``"j:
    jump"``"g:gresala"
650 LET pos=1
660 DEF PROC afisare pos
670 DO
680 GET a$
690 IF a$=7"THEN up
700 IF a$="6" THEN down
710 IF s$="g" THEN SCROLL 7,16;16,14*8;28,80:eroare
    1:sortare: LET pos=ner:afisare pos
720 IF a$="j" THEN jump:afisare pos
730 IF a$="f" THEN find:afisare pos
740 LOOP
750 END PROC
760 DEF PROC main
770 init
780 getn
790 LET ind=ind-1
800 sortare
810 display
820 END PROC
830 DEF PROC eroare
840 INPUT "NR.eroare;"ner
850 IF ner<1 OR ner>ind-1 THEN END PROC
860 PRINT AT 4,4; USING ;"# # #";ner
870 INPUT "NOTA1 :";n1
880 LET a (ner)=n1
890 PRINT AT 4,9; USING ;"# #.# #";a(ner)
900 INPUT "NOTA2 :";n2
910 LET b(ner)=n2
920 PRINT AT 4,16; USING ;"# #.# #";b(ner)
930 LET m(ner,2)=(ner)+b(ner))/2-0.003
940 PRINT AT 4,23; USING ;"# #.# #"; m(ner,2)
950 END PROC
960 DEF PROC eroarel
970 INPUT "POS.eroare :";ner
980 IF ner<1 OR ner>ind THEN END PROC
990 PRINT AT 4,0; USING ;"# # #";ner; AT 4,4;
```



```
      USING ;"# # #"; m(ner,1)
1000 INPUT "NOTA1 :";n1
1010 LET a(m(ner,1))=n1
1020 PRINT AT 4,9; USING ;"# #.# #";a(m(ner,1))
1030 INPUT "NOTA 2 :"; n2
1040 LET b(m(ner,1))=n2
1050 PRINT AT 4,16; USING ;"# #.# #"; b(m(ner,1))
1060 LET m(ner,2)=(a(m(ner,1))+b(m(ner,1)))/2-0,003
1070 PRINT AT 4,23; USING ;"# #.# #";m(ner,2)
1080 END PROC
1090 DEF PROC jump
1100 INPUT "POS. :";pos
1110 END PROC
1120 DEF PROC find
1130 INPUT "NR. :";num
1140 IF num<1 OR num>ind THEN GO TO 1130
1150 FOR i=1 TO ind
1160 IF m(i,1)=num THEN GO TO 1180
1170 NEXT i
1180 LET pos=1
1190 END PROC
```

18. COMPILATORUL HISOFT BASIC

18.1. INTRODUCERE

Programele scrise în libajul BASIC al calculatoarelor compatibile cu ZX-SPECTRUM pot fi *compilate* (transpuse) în cod mașină cu ajutorul compilatoarelor, ceea ce le sporește sensibil viteza de execuție. Există compilatoare care lucrează numai cu numere întregi și măresc viteza de execuție a programelor BASIC de circa 100 ori, dar prezintă dezavantajul că nu lucrează cu numere în virgulă mobilă (deci cu numere zecimale) și adesea au și alte restricții. De asemenea există și compilatoare care lucrează cu numere în virgulă mobilă dar fac programele BASIC doar de 3-5 ori mai rapide.

Programul **HISOFT BASIC** este un compilator care combină avantajele celor două tipuri de compilatoare descrise, eliminând însă dezavantajele lor. Astfel, HISOFT BASIC este un compilator în *virgulă flotantă* (mobilă) care poate obține viteza unui compilator care lucrează numai cu numere întregi dacă execută operații care nu necesită complexitatea unei aritmetici în virgulă mobilă. În același timp, el este *cel mai rapid compilator în virgulă mobilă* realizat pînă în prezent pentru calculatoarele compatibile cu ZX-SPECTRUM. Se menționează că HISOFT BASIC poate compila aproape toate programele BASIC *mai mici de 30 Ko* generînd un cod mașină rapid, putînd lucra și cu funcții definite de utilizator și cu matrice numerice bidimensionale.

De regulă alte compilatoare au un bloc de rutine în cod mașină de circa 5 Ko care trebuie să fie prezent în programul final pentru a fi

posibilă rularea codului compilat. Aceasta înseamnă că pînă și cel mai scurt program BASIC va avea, după compilare, mai mult de 5 Ko. Compilatorul HISOFT BASIC elimină acest dezavantaj important, *el incluzînd în programul final doar rutinele necesare pentru codul generat.* În acest fel, programele compilate nu vor avea lungimi mari. De asemenea, HISOFT BASIC permite amplasarea codului generat oriunde în RAM, chiar și în locațiile ocupate de compilator.

Instrucțiunile de bază sînt

- pentru începutul compilării : **REM: OPEN #**
- pentru sfîrșitul compilării : **REM: CLOSE #** (opțională dacă e dorește compilarea pînă la sfîrșitul programului BASIC).
- comanda pentru compilare : ***C** .
- comanda pentru lansarea în execuție a codului compilat : ***R** .

În timpul compilării HISOFT BASIC se va opri de două ori indicînd unele informații în partea de jos a ecranului; după această afișare se apasă de fiecare dată o tastă pentru continuare. La prima oprire marginile ecranului își vor schimba culoarea iar pe ecran vor apărea puncte și culori deosebite. După a doua apăsare a unei taste se vor indica lungimea codului compilat, numărul de octeți ce trebuie rezervat pentru variabilele cod mașină și numărul de octeți ocupat de programul BASIC fără variabile. Cea mai importantă informație oferită se referă la modalitatea de încărcare și salvare a codului mașină generat. Astfel adresa din linia de încărcare a programului (LOAD) trebuie să fie aceeași cu cea de lansare în execuție a codului mașină (RANDOMIZE USR). De exemplu dacă adresa de lansare a codului generat este **65000**, adresa de încărcare a acestui cod este tot **65000**, avînd însă înainte de încărcarea codului scrisă instrucțiunea **CLEAR 64999** (deci adresă-1).

Observații : 1) Programul BASIC poate fi recompilat cu aceeași comandă ***C** ; toate informațiile oferite de compilator și citite pe ecran vor fi aceleași cu excepția adresei unde se găsește codul compilat și anume vîrfurile memoriei (deasupra lui RAMTOP care este schimbat pentru a se afla în fața codului recent compilat). Pentru a readuce RAMTOP-ul la poziția inițială se tastează ***X**.

2) Reinițializarea compilatorului (care are două rutine în

cod mașină amplasate la adresele 60000 și lungime 77 octeți, respectiv la 23734 și lungimea de 11865 octeți) se face cu comanda
RANDOMIZE USR 23792 .

18.2. MODUL DE LUCRU CU HISOFT BASIC

1) Se încarcă programul HISOFT BASIC (prescurtat HiBASIC) cu toate blocurile sale.

2) Se încarcă programul BASIC ce urmează a fi compilat (având lungimea maximă de 30 Ko). Acesta trebuie să înceapă de la linia cu numărul cel mai mic (de exemplu dacă programul se execută cu RUN 9000, atunci la începutul lui se va insera instrucțiunea GO TO 9000).

3) Se verifică dacă programul BASIC nu conține nici o comandă neacceptată de HISOFT BASIC și anume : CLEAR, CONTINUE, ERASE, FORMAT, LIST, LLIST, LOAD, MERGE, MOVE, NEW, RUN, SAVE, VERIFY, VAL variabilă șir .

4) Se inserează la începutul programului BASIC o linie cu directiva

REM : OPEN #

care reprezintă instrucțiunea pentru începutul compilării, iar la sfârșitul programului de compilat instrucțiunea

REM : CLOSE #

care reprezintă instrucțiunea de sfârșit a compilării.

5) Se tastează

***T**

și se lansează în execuție programul BASIC (cu RUN), încercând diferite date de intrare. Apoi se tastează CLS și apoi din nou *T, ceea ce dătează afisarea listei de variabile a programului BASIC, în dreptul fiecărei variabile indicându-se tipul ei: Real (reală), Integer (întregă) sau Posint, respectiv Posinteg (combinație de pozitive și întregi). Pentru variabilele întregi respectiv întregi pozitive se introduce în programul BASIC o linie scrisă înainte de REM : OPEN #, având forma

REM: INT nume variabilă, nume variabilă, (pentru variabile întregi)

REM: INT+nume variabilă, nume variabilă, (pentru variabile întregi pozitive).

Dacă instrucțiunile DATA ale programului conțin valori întregi ele

se vor completa în linia respectivă de program cu

DATA INT valori numerice .

Pentru șirurile de caractere se inseră instrucțiunea

REM:LEN NŞ < = numărul caracterelor.

Efectuînd această operație de cunoaștere a variabilelor programului BASIC se vor obține două avantaje importante : scurtarea codului mașină compilat și sporirea vitezei de execuție.

6) Se compilează programul BASIC tastînd

***C**

iar în timpul compilării, după afișarea unui mesaj care nu este de eroare, se apasă o tastă oarecare.

7) La terminarea compilării corecte se afișează :

- numărul de octeți ocupat de codul compilat ;
- numărul de octeți necesari pentru variabilele codului compilat (care se adaugă la lungimea codului compilat cînd acesta se salvează);
- numărul de octeți al programului BASIC;
- comenzile ce trebuie folosite pentru salvarea codului (SAVE adresa,lungime) și încărcarea lui (LOAD).

La compilarea cu eroare se va citi lista mesajelor de eroare(v.18.3).

8) Se salvează codul compilat între instrucțiunile **REM:OPEN #** și **REM : CLOSE #**. Pentru execuția codului mașină obținut, în programul care apelează codul se va introduce instrucțiunea

RANDOMIZE **USR adresa**

(unde "adresa" este adresa codului compilat). Se precizează că acest cod compilat se va întoarce în BASIC în punctul unde îl întâlnește, în programul BASIC inițial, o instrucțiune **STOP** sau instrucțiunea de sfîrșit de compilare **REM : CLOSE #**. De reținut că înainte de încărcarea codului compilat, memoria trebuie ștearsă cu instrucțiunea

CLEAR adresa-1

18.3. COMENZILE SI MESAJELE DE EROARE ALE COMPILATORULUI HISOFT BASIC

Aceste comenzi și mesaje sînt prezentate în tabelul care urmează.

Tabelul 18.1- COMENZILE HIBASIC

Nr.crt.	Mnemonica	Efect
1	*C	Declanșează compilarea programului BASIC cuprins între instrucțiunile REM : OPEN # și REM : CLOSE #. Alte informații se obțin cu REM:LINE sau REM:LIST.
2	*X	Face efectul echivalent lui CLEAR 65367 dar fără CLS sau RESTORE. Prin aceasta se șterge codul compilat asigurându-se spațiu pentru noul cod. Dacă există unele coduri mașină în partea superioară a memoriei ce trebuie păstrate, atunci nu se utilizează *X ci CLEAR adresa cea mai mică-1.
3	*R	Execută codul compilat (deci face același lucru ca și RANDOMIZE USR adresă).
4	*T	Comandă folosită pentru a afla informații despre tipul variabilelor programului BASIC. Afișează tipul variabilelor numerice simple și lungimea maximă ocupată de variabilele șir simple, dar nu dă informații despre matrice.
5	*ERASE	Șterge programul BASIC și variabilele din memorie fără a afecta programul HIBASIC sau orice cod dincolo de RAMTOP.
6	BREAK	Oprește orice program, inclusiv codul compilat.
7	*D	La fel ca *C dar memorează doar codul generat de instrucțiunea DATA.
8	*E	La fel ca *C dar nu memorează codul generat de instrucțiunea DATA.

Cele două comenzi *D și *E sînt utile pentru a diviza un program de lungime mare ce conține instrucțiuni DATA în două părți: partea de cod și partea de date (nu contează ordinea comenzilor; astfel se poate

folosi mai întâi *D,SAVE, pentru partea de date și apoi *E, SAVE pentru partea de cod). Cele două părți vor forma un cod continuu în programul final.

Pentru o asemenea situație algoritmul de lucru este următorul :

1) Se tastează *D și rezultă informațiile SAVE "nume1" CODE: adr1,lungime1, LOAD "nume1" CODE adr2, după care se salvează CODE adr1,lungime1.

2) Se tastează *E și apare mesajul "Delete BASIC Y/N? "; se tastează Y și se afișează SAVE "nume2" CODE adr3, (lungime3+nr, număr octeți pentru variabile), respectiv LOAD "nume2" CODE adr4, după care se salvează CODE adr3,lungime3+nr,octeți pentru variabile.

3) În programul loader se introduc instrucțiunile :

```
CLEAR adr4-1: LOAD "nume1"CODE adr2,lungime1:
LOAD "nume2" CODE adr4,lungime3+nr,octeți pentru
variabile:
RANDOMIZE USR adr4
```

Tabelul 18.2 -MESAJELE DE EROARE HIBASIC

Nr.crt.	MESAJUL DE EROARE	SEMNIIFICAȚIA
1	Invalid compiler directive	Directivă de compilare invalidă (se obține acest mesaj când se declară o variabilă a doua oară).
2	Expecting a number	Se așteaptă un număr și se va introduce numărul respectiv
3	Expecting an integer	Se așteaptă un întreg (se va folosi INT după instrucțiunea DATA pentru care există valori întregi)
4	Non supported	Neadmis
5	Non existed line	O linie inexistentă (apare când o instrucțiune GO TO, GOSUB, RESTORE se referă la o linie inexistentă sau în afara porțiunii de program compilate.

6	To many variabile	Prea multe variabile (numărul maxim admis pentru variabilele numerice simple este 255).
7	USE *D, *E	Nu mai este spațiu de memorie.
8	Not enough room for m/c	Se cere folosirea comenzilor *D și apoi *E.
9	Not enough room for m/c	Spațiu insuficient pentru codul compilat (nu incipe în memorie după RAMTOP).
10	Exec. adress to high	Adresă de execuție prea mare.
11	Do not test	Nu testați (codul compilat nu se află în poziție potrivită și deci nu se poate executa ci doar salva și reincărca în poziția respectivă).
12	No file space	Nu există spațiu pentru fișier (apare în timpul executării lui *T)

Sînt de menționat următoarele :

1) Nici o expresie (exceptînd VAL"număr") nu este admisă în
instrucțiunile DIM și DATA.

2) Nu sînt acceptate expresii (altele decît VAL"număr") în GO TO,
GOSUB, RESTORE și nici matrice cu dimensiuni mai mari sau egale cu 3.

3) Atributele implicite PAPER 8, FLASH 0, BRIGHT 0 pe care
BASIC-ul le folosește pentru PLOT, DRAW, CIRCLE nu sînt introduse în
timpul compilării.

18.4. EXEMPLE DE UTILIZARE A COMPILATORULUI HISOFT BASIC

Pentru ilustrarea modului de lucru cu compilatorul HISOFT
BASIC se prezintă un număr de 5 exemple comentate în cele de urmează.

Exemplul 18.1 : programul afișează un șir de caractere unul sub

celălalt, diminuat de fiecare dată cu un caracter:

```
10 INPUT "Introduceti numele dvs",N$
20 LET L=LEN N$
30 FOR I=0 TO L-1
40 PRINT N$ (TO L-I)
50 NEXT I
```

În vederea compilării se introduce o linie nouă:

```
9 REM : OPEN #
```

după care se lansează în execuție cu comanda RUN pentru a se vedea dacă programul funcționează corect (tastînd " MIRCEA " ca răspuns la instrucțiunea INPUT). Apoi acesta se compilează cu comanda *C,compilerul afișînd :adresa și lungimea codului [64510,581(plus 277 octeți pentru variabilele codului)],adresa de LOAD (64510) și numărul de octeți al programului BASIC (104).

Dacă se inserează linia

```
8 REM : LEN N$ <=50
```

și se tastează succesiv *X și apoi *C se obține un cod mașină mai mic (581 + 72 octeți pentru variabilele codului compilat).

Se precizează că la programe mici rezultă coduri mașină mai lungi decît programul BASIC, dar la programe lungi codul mașină este mai mic decît acest program.

Exemplul 18.2 : programul desenează o spirală bazată pe hexagoane a căror latură se diminuează continuu:

```
10 REM : OPEN #
15 CLS
20 LET C=COS(PI/3)
30 LET S=SIN(PI/3)
40 LET C1=COS(PI/36)
50 LET S1=SIN(PI/36)
60 LET SF=.95
70 LET X=95
80 LET Y=0
90 LET CX=130
```

```

100 LET CY=88
110 LET SC=1.16
120 FOR J=1 TO 43
130 FOR I=0 TO 6
140 LET SX=X*SC+CX
150 LET SY=CY+Y
160 IF I=0 THEN GO TO 190
170 PLOT SX1,SY1
180 DRAW (SX-SX1),(SY-SY1)
190 LET SX1=SX:LET SY1=SY
200 LET IX=X*C-Y*S
210 LET Y=X*S+Y*C
220 LET X=IX
230 NEXT I
240 LET IX=SF*(X*C1-Y*S1)
250 LET Y=SF*(X*S1+Y*C1)
260 LET X=IX
270 NEXT J
280 STOP
290 SAVE "SPIRALA"

```

Tastînd *C pentru compilare se constată că în partea superioară a ecranului apar linia 290 și semnul ? (care clipește) și mesajul "Non suported" (în partea inferioară a ecranului). Prin acest mesaj compilatorul comunică neexecutarea compilării datorită prezenței instrucțiunii **SAVE** (deoarece nici una din comenzile sistemului de operare nu poate fi executată de HISOFT BASIC fiind specifice interpretorului BASIC). Din acest motiv se inserează linia

```
271 REM : CLOSE #
```

și programul se poate acum compila cu directiva *C (linia 290 nu mai este compilată). Rezultă un cod mașină la adresa 64376 și lungimea 837 (+105) octeți, pentru programul BASIC ce are lungimea de 497 octeți,

Tastînd *X și apoi *T urmat de **RUN**, programul este rulat; tastînd în continuare **CLS** și apoi *T se afișează setul de variabile al programului BASIC. Dacă acum se inserează linia

```
REM : INT+ CX, CY, J, I
```

și programul se recompilază cu directiva *C, rezultă un cod mașină mai

scurt (732+81) octeți. Rulînd codul mașină (cu comanda *R) se constată că spirala se desenează de 3 ori mai repede.

Exemplul 18.3 : programul desenează 13 pisici în diferite zone ale ecranului :

```

8 REM:INT+ N,J,I,X,Y,X0,Y0
9 REM:OPEN #
10 DATA INT 2,0,6,0,0,9,1,15,0,16,1,16,12,15,13 14,
    12,14,2,10,2,11,6,8,12,10,15,9,18,8,22,7,18,3
    18,2,22,1,18,0,15,2,12,0,6,0,3,2,0
14 CLS : LET N=0: RANDOMIZE
15 FOR J=0 TO 235 STEP 19
16 BEEP .01,N
20 RESTORE : READ X,Y
30 PLOT J+X,4+RND*150
40 FOR I=1 TO 23
50 LET X0=X: LET Y0=Y
60 READ X,Y
70 DRAW X-X0,Y-Y0
80 NEXT I
81 LET N=N+1
82 PRINT # I; AT 1,0;N;" PISICI";"" AND (N>1)
85 NEXT J
90 PAUSE 0
91 REM : CLOSE #

```

Programul are direct introduse în liniile 8 și 10 specificațiile privind variabilele întregi (N, J, I, X, Y, X0, Y0), respectiv valorile întregi din DATA. În urma compilării (cu *C) rezultă un cod mașină la adresa 64694 de lungime 652 (+22) octeți (pentru programul BASIC care are 724 octeți).

Exemplul 18.4 : programul umple ecranul cu un cuvînt ales de utilizator, din trei variante oferite :

```

8 REM : LEN C$<=5

```

```

9 REM : INT+ N,I
10 REM : OPEN #
20 CLS : PRINT "1(2sp)CAFEA" ` "2(2sp)CEAI" `
  "3(2sp)UNT"
30 INPUT "Apasati 1,2 sau 3",N
40 IF N<1 OR N>3 THEN GO TO 30
50 GOSUB 100*N
60 STOP
100 LET C$=CAFEA": GO TO 500
200 LET C$="CEAI": GO TO 500
300 LET C$=UNT": GO TO 500
500 CLS : FOR I=1 TO INT (704/LEN C$): PRINT C$;:
  NEXT I: RETURN

```

Tastînd *C pentru compilare se constată că se afișează linia 50 împreună cu mesajul "Non suported" deoarece conține o instrucțiune GOSUB calculată. Urmărind programul se observă că variabila N nu poate fi decît 1,2 sau 3, deci datele folosite în linia 50 sînt 100, 200 și 300. Compilarea va fi posibilă dacă se inserează linia

```
7 REM : GOSUB 100,200,300
```

Rezultă codul mașină de lungime 712(+15) octeți plasat la adresa 64641 (pentru un program BASIC de 305 octeți).

Exemplul 18.5 : programul desenează spațial (3D) un număr de 12 paralelipede de înălțimi aleatoare, dispuse pe 3 rînduri avînd fiecare 4 paralelipede :

```

7 REM : INT +a,y,o,i,rnd,q,f
8 REM : OPEN #
2) BORDER 7: PAPER 7: INK 0: CLS
100 FOR A=0 TO 60 STEP 20: PLOT A,A: DRAW 160,0:
  NEXT A
105 FOR A=0 TO 160 STEP 40: PLOT A,0: DRAW 60,60:
  NEXT A
990 DIM Q(12)
994 LET y=0: LET o=0: LET I=0
999 LET RND=INT(RND*70)+35: LET Q=RND : PRINT AT
  0,0;

```

```
"INALTIMEA :";Q;" "  
1000 FOR f=0 TO RND : PLOT INVERSE 1;40+I,f+40-y:  
DRAW 40,0: PLOT INVERSE 1;80+I, f+40-y: DRAW  
20,20: PLOT INVERSE 1;100+I,f+60-y: DRAW -40,0:  
DRAW -20,-20: NEXT f  
1005 PLOT INVERSE 1;40+I,f+40-y: DRAW INVERSE 1;40,0:  
PLOT INVERSE 1;80+I,f+40-y: DRAW INVERSE 1;  
20,20: PLOT INVERSE 1;100+I,f+60-y: DRAW INVERSE  
1;-40,0: DRAW INVERSE 1;-20,-20  
1010 PLOT INVERSE 1;40+i, f+40-y: DRAW INVERSE 1;0,-  
RND  
1049 LET o=o+1: LET I=1+41  
1060 IF o=4 THEN LET i=-20: LET y=y+20  
1080 IF o=12 THEN GO TO 1100  
1090 GO TO 999  
1100 PRINT # 1; AT 1,4;"APASATI O TASTA OARECARE":  
PAUSE 0: PRINT AT 0,0;"DORITI RELUAREA (d/n)?:"  
PAUSE 0: CLS  
1110 IF INKEY$="D" OR INKEY$="d" THEN GO TO 100  
1120 BEEP .2,40: PAUSE 4: BEEP .2,-40: STOP  
1121 REM : CLOSE #
```

În urma compilării se obține un cod-mașină de 1623 octeți (1541+82)
plasat la adresa 63745.

BIBLIOGRAFIE SELECTIVA

1. **Dumitrașcu Liviu**, *Microelectronica interactivă*, Editura Tehnică, București, 1989.
2. **GEE S.,M.**, *The Spectrum Programmer*, Granada, London, Toronto, Sidney, NEW York, 1982
3. **Petrescu Adrian, Francisc Iacob**, *Microcalculatorul personal HC-80*, AMC vol.49, Editura Tehnică, București, 1985
4. **Petrescu Adrian și colectiv**, *ABC de calculatoare personale și...nu doar atît*, Editura Tehnică, București, 1990
5. **Popovici Mircea-Mihail**, *Mecanica tehnică pentru muncitori* (3 volume). Editura Tehnică, București, 1980, 1982, 1985
6. **Rădulescu Octavian**, *Sinteze optimale în construcția de mașini*, Editura Tehnică, București, 1984
7. **Rădulescu Octavian**, *Organe de mașini- Sinteze optimale*, Editura Academiei Militare, București, 1982
8. **Vickers Steven**, *ZX Spectrum-Sinclair, Basic Programming* Sinclair Research Ltd, London, UK, 1983
9. Documentațiile programelor **BETA BASIC** și **HISOFT BASIC**

SUMAR

13. APLICAȚII ÎN TEHNICĂ

13.1. CALCULUL OPTIMAL AL BAREI DREPTÉ SOLICITATE LA ÎNTINDERÉ CENTRICĂ.....	4
13.2. CINEMATICÁ ANALITICÁ A MECANISMULUI MANIVELÁ-PISTON.....	9
13.3. ECHILIBRAREA STATICÁ A MECANISMULUI MANIVELÁ-PISTON.....	15
13.4. SINTEZA OPTIMALÁ A ARCULUI BARÁ DE TORSIUNE.....	20
13.5. SINTEZA OPTIMALÁ A ARCULUI LAMELAR SIMPLU.....	25
13.6. SINTEZA OPTIMALÁ A ARCULUI ELICOIDAL CILINDRIC GENERATOR DE FÓRTE.....	31
13.7. SINTEZA OPTIMALÁ A TRANSMISIILOR CU CURELE TRAPEZOIDALE ÎNGUSTE.....	42
13.8. SINTEZA OPTIMALÁ A ANGRENAJELOR CILINDRICE EXTERIOARE.....	50

14. APLICAȚII ÎN DOMENIUL MATEMATICII

14.1. PUTERI, FACTORIALI ȘI COMBINĂRI	85
14.2. ȘIRURI	87
14.3. SERII	89
14.4. FUNCȚII NUMERICE.....	91
14.5. DERIVATE, POLINOAME, LUNGIMEA ARCULUI DE CURBĂ.....	92
14.6. INTEGRAREA FUNCȚIILOR NUMERICE	96
14.7. ECUAȚII DIFERENȚIALE DE ORDINUL 1	101
14.8. OPERAȚII CU VECTORI.....	104
14.9. MATRICE ȘI ECUAȚII LINIARE.....	108

14.10. DISTRIBUTII STATISTICE.....	117
14.11. LEGI PROBABILISTICE	127
15. APLICAȚII DE INTERES GENERAL	
15.1. DIRECTORY	139
15.2. AGENDA TELEFONICĂ	142
15.3. CALENDAR	145
15.4. BIORITM.....	150
15.5. CONVERSIE HEXAZECIMAL-ZECIMAL ȘI INVERS... 151	
16. PROGRAME DE DIVERTISMENT (JOCURI)	
16.1. PRELIMINARII	156
16.2. LOGICA JOCULUI	157
16.3. NOTIUNI SUMARE DESPRE ANIMATIE	158
16.4. EXEMPLE DE JOCURI.....	161
16.4.1. METEORIȚI	161
16.4.2. JOC LOGIC ȘI DE ÎNDEMÎNARE.....	164
16.4.3. BALISTICA	166
16.4.4. ROBOT.....	168
16.4.5. TIR CU ARBALETA	171
17. BETA BASIC	
17.1. INTRODUCERE.....	175
17.2. COMENZILE/INSTRUCTIUNILE BETA BASIC.....	178
17.3. FUNCȚII BETA BASIC.....	203
17.4. EXEMPLU DE UTILIZARE.....	211
18. COMPILATORUL HISOFT BASIC	
18.1. INTRODUCERE.....	216
18.2. MODUL DE LUCRU CU HISOFT BASIC.....	218
18.3. COMENZILE SI MESAJELE DE EROARE ALE COMPILATORULUI HISOFT BASIC.....	219
18.4. EXEMPLE DE UTILIZARE A COMPILATORULUI HISOFT BASIC.....	222



Lucrarea contine o colectie de programe scrise în limbajul BASIC, care aplica si dezvolta notiunile prezentate în primul volum. Astfel:

Capitolul 13 contine 8 programe din domeniul Rezistentei materialelor, Mecanismelor si Organelor de masini, remarcându-se în special programele de sinteza optima a unor componente de masini.

Capitolul 14 contine 50 de programe de matematica diverse (puteri, factoriali, combinari, siruri, serii, functii umerice, ecuatii diferentiale de ordinul 1, operatii cu vectori, matrice si ecuatii lineare, distributii statistice si legi probabilitistice).

Capitolul 15 prezinta 5 programe de uz general (directory, agenda telefonica, calendar, bioritm, conversie hexazecimal-zecimal).

Capitolul 16 este consacrat programelor de divertisment (jocuri) oferindu-se notiuni însoțite de aplicatii privind animatia pe calculator; de asemenea sînt prezentate 5 jocuri atractive.

Capitolul 17 prezinta în detaliu programul BETABASIC (în variantele 3.1 si 1.8) însoțit de aplicatii

Capitolul 18 explica modul de utilizare a compilatorului HISOFTBASIC cu ajutorul caruia programele scrise în limbajul BASIC pot fi transpuse (compilate) în limbaj masina pentru a le spori viteza de executie.

Mircea Mihail Popovici

În EDITURA APH au aparut:

- **M. POPOVICI - BASIC pentru calculatoarele SPECTRUM, etc. INSTRUCTIUNI, EXERCITII, PROBLEME**
- **R. HRISTEV - Introducere în PROLOG.**
- **GHIDUL utilizatorului si programatorului SPECTRUM**
- **M. POPOVICI - LIMBAJUL MASINA pt. SPECTRUM, etc.**

ISBN 973-95175-7-9

LEI:

2500