

CHARGEUR . BAS

```

10 *****
20 ' VARIABLES LOCALES pour CPC '
30 ' copyright Y. GOUR (11/1990) '
40 *****
50 '
60 DEFINT a-z:MODE 1:CALL &BC02
70 INPUT"adresse de la routine";a:ad=a
80 MEMORY a-1
90 WHILE a<>"FIN"
100 READ a$
110 IF a$="" THEN READ a$:b=ad+VAL("&"a$):POKE a,b AND 255:POKE
a+1,INT(b/256)AND 255:a=a+2:GOTO 130
120 IF a$(">"FIN" THEN POKE a,VAL("&"a$"):a=a+1
130 WEND
140 IF a(">(ad+1E0) THEN PRINT"Erreur":END
150 o=PEEK(6):IF o=128 THEN j=1 ELSE IF o=123 THEN j=2 ELSE j=3
160 READ a$:a=ad+VAL("&"a$)
170 FOR i=1 TO j:READ cpc$,a$,b$:NEXT
180 POKE a,VAL("&"a$):POKE a+1,VAL("&"b$)
190 SAVE"local",b,ad,&1E0
200 END
210 '
220 DATA 21,$,34,01,$,9,C3,D1,BC,$,1A,C3,$,44,C3,$,7D,C3,$,115,C3
,$,AC,C3,$
230 DATA 151,53,54,41,43,CB,4C,4F,43,41,CC,45,4E,44,4C,4F,43,41,C
C,47,49,56,C5
240 DATA 47,45,D4,$,$,$,$,$,9,$,1DF,$,$,$,$,$,$,$,CD,$,EB,DD,
66,3,DD,6E,2
250 DATA DD,46,01,DD,4E,0,22,$,3C,22,$,40,9,22,$,42,23,22,$,3E,2B
,2B,ED,5B,$
260 DATA 3B,87,ED,52,DB,2A,$,3C,ED,5B,$,3A,2B,ED,52,D0,AF,32,$,3F
,3C,18,76,CD
270 DATA $,DF,CB,3F,2A,$,40,E5,DD,46,3,DD,4E,2,9,ED,5B,$,42,13,B7

```

```

,ED,52,30,$,40
280 DATA DD,66,1,DD,6E,$,D1,ED,80,EB,E,4,DD,9,3D,20,DD,22,$,40,C9
,CD,$,DF,CB
290 DATA 3F,2A,$,42,DD,46,3,DD,4E,2,ED,42,E5,ED,5B,$,40,ED,52,3B
,16,DD,66,1
300 DATA DD,6E,$,D1,D5,ED,80,E1,E,4,DD,9,3D,20,DD,22,$,42,C9,3E,3
,18,14,47,3A
310 DATA $,3F,87,78,20,4,3E,1,18,8,B7,28,5,CB,47,CB,3E,2,21,$,18B
,3C,F5,7E,23
320 DATA 87,20,FB,F1,3D,20,FG,7E,23,CD,5A,8B,B7,20,F8,CD,$,89,3E,
11,C3,$,$,3E
330 DATA 4,18,DE,CD,$,DF,67,3D,3D,28,6,87,4F,6,0,DD,9,7C,CB,3F,2A
,$,40,DD,46,3
340 DATA DD,4E,2,B7,ED,42,ED,5B,$,3C,ED,52,3B,D7,19,DD,56,1,DD,5E
,$,E5,ED,80,E1
350 DATA 1,FC,FF,DD,9,3D,20,DC,22,$,40,C9,CD,$,DF,67,3D,3D,28,6,8
7,4F,6,0,DD,9
360 DATA 7C,CB,3F,2A,$,42,DD,46,3,DD,4E,2,E5,9,ED,5B,$,3E,87,ED,5
2,30,9B,E1,DD
370 DATA 56,1,DD,5E,$,ED,80,1,FC,FF,DD,9,3D,20,DE,22,$,42,C9,$,D
,4E,6F,20,61
380 DATA 72,67,75,6D,65,6E,74,D,A,$,49,6C,6C,65,67,61,6C,20,73,74
,61,63,6B,D
390 DATA A,$,4F,64,64,20,6E,75,6D,62,65,72,20,6F,66,20,61,72,67,7
5,6D,65,6E
400 DATA 74,73,D,A,$,53,74,61,63,6B,20,66,75,6C,6C,D,A,$,53,74,61
,63,6B,20
410 DATA 65,6D,70,74,79,D,A,$,FIN
420 DATA 10F,464,93,CA,664,58,CB,6128,55,CB

```

ARBRE . BAS

```

10 ' transcription du programme en PASCAL "ARBRE" >LA
20 ' de Alain LAMBERT, publie dans SVM no 46 de >LB
30 ' Janvier 1988 page 91. >LC
40 ' >LD
50 MEMORY &3000:MODE 1:INK 0,14:INK 1,9:INK 2,6:INK 3,2 >QC
4:BORDER 0:CLS
60 :STACK,&4000,&1000 >PN
70 ' >LG
80 ' definition des types de variables et initialisation >LH
ns
90 DEG:DEFREAL g,c,d:DEFINT i,l,a,v,k,s,n,o,p >QA
100 g=g:c=c:d=d:i=i:l=l:2=12:a=a:2=2:v=v:k=k:s=s:n=n >MX
:n2:n2
110 ' >RC
120 ' fonctions pour tourner a gauche et a droite >RD
130 ' la tortue (coursur graphique) censee dessiner. >RE
140 ' son orientation est donnee par o. >RF
150 DEF FN lt(o,a)=(o+a) MOD 360 >XZ
160 DEF FN rt(o,a)=(o-a) MOD 360 >YA
170 ' >RJ
180 ' definition de la couleur en fonction du >RK
190 ' niveau: distinguer tronc/branche de feuille >TA
200 DEF FN col(n)=1+2*MAX(0,SGN(n-4)) >DG
210 ' >RD
220 INPUT"longueur du tronc";l >GB

```

```

230 PRINT
240 INPUT"coefficient gauche";g
250 INPUT"coefficient centre";c
260 INPUT"coefficient droit";d
270 PRINT
280 INPUT"angle";a
290 PRINT
300 INPUT"variation d'angle";v
310 PRINT
320 INPUT"courbure";k
330 PRINT
340 IF <k>0 THEN INPUT"meme sens (0) ou alterne (1) ";s
:PRINT
350 INPUT"niveau de ramification";n
360 '
370 ' definition de l'ecran graphique et
380 ' mise en position de la tortue
390 CLS:ORIGIN 320,0:MOVE 0,0:o=90
400 ' appel de la procedure de trace d'arbre
410 :GIVE,2,01,2,0a,2,0n:GOSUB 790
420 '
430 ' attente de frappe de touche
440 CALL &BB00:BORDER 10:WHILE INKEY*="" :WEND
450 END
460 '
470 '-----
480 ' FRUIT
490 '
500 :LOCAL,2,01,2,01
510 '
520 'un fruit est un octogone
530 IF RND%4 THEN pencol=2:l=1:FOR i=1 TO 8:GIVE,2,0
1:GOSUB 1140:o=FN rt(o,45):NEXT
540 '
550 :ENDLOCAL,2,0i,2,01
560 RETURN
570 '
580 '-----
590 ' EPAISSEUR
600 '
610 :LOCAL,2,0n,2,01,2,0n2
620 :GET,2,0n,2,01
630 '
640 IF n<0 THEN 730
650 ' epaisseur obtenue en tracent des rectangles
660 n2=n/2:GIVE,2,0n2:GOSUB 1140
670 o=FN lt(o,90):GIVE,2,01:GOSUB 1140
680 o=FN lt(o,90):GIVE,2,0n:GOSUB 1140
690 o=FN lt(o,90):GIVE,2,01:GOSUB 1140
700 o=FN lt(o,90):n2=ROUND(n/2):GIVE,2,0n2:GOSUB 1140
710 n2=n-2:GIVE,2,0n2,2,01:GOSUB 610
720 '
730 :ENDLOCAL,2,0n,2,01,2,0n2
740 RETURN
750 '
760 '-----
770 ' ARBRE
780 '
790 :LOCAL,2,01,2,0a,2,0n,2,012,2,0a2,2,0n2

```

```

>NH
800 :GET,2,01,2,0a,2,0n
>RK
810 '
>HQ
820 IF n=0 THEN 1000
>TB
830 '
>TC
840 ' donner de l'epaisseur
>UY
850 IF n>4 THEN pencol=3:o=FN rt(o,90):n2=n-3:GIVE,2,0
n2,2,01:GOSUB 610:o=FN lt(o,90)
>PX
860 pencol=FN col(n)
>VA
870 :GIVE,2,01:GOSUB 1140
>TG
880 '
>TH
890 ' branche de gauche
>HD
900 o=FN lt(o,a)
>PX
910 l2=ROUND(l*g):a2=a-v:n2=n-1:GIVE,2,012,2,0a2,2,0n2
:GOSUB 790
>HD
920 IF s=0 OR (n MOD 2)=1 THEN o=FN rt(o,a+k) ELSE o=FN
rt(o,a-k)
>TC
930 '
>TD
940 ' branche du centre
>XN
950 l2=ROUND(l*c):GIVE,2,012,2,0a2,2,0n2:GOSUB 790
>TF
960 '
>TG
970 ' branche de droite
>HK
980 IF s=0 OR (n MOD 2)=1 THEN o=FN rt(o,a-k) ELSE o=FN
rt(o,a+k)
>XU
990 l2=ROUND(l*d):GIVE,2,012,2,0a2,2,0n2:GOSUB 790
>XD
1000 '
>XE
1010 ' retour au noeud precedent
>LE
1020 o=FN lt(o,a)
>QA
1030 pencol=FN col(n)
>BN
1040 l2=-l:GIVE,2,012:GOSUB 1140
>XJ
1050 ' dessin possible de fruit au niveau 2
>TH
1060 IF n=2 THEN GOSUB 500
>YA
1070 '
>RD
1080 :ENDLOCAL,2,01,2,0a,2,0n,2,012,2,0a2,2,0n2
>FD
1090 RETURN
>XE
1100 '
>XF
1110 '-----
>XG
1120 ' "forward": en avant la tortue!
>XH
1130 '
>LC
1140 :LOCAL,2,01
>XK
1150 '
>AG
1160 :GET,2,01
>EK
1170 DRAWR l*CS(o),i*SIN(o),pencol
>YC
1180 '
>PZ
1190 :ENDLOCAL,2,01
>EG
1200 RETURN

```

FACT . BAS

```

10 MODE 2:STACK,7000,1000
>VB
20 n%=nX:n2X=n2X:fact!=fact!:prov!=prov!
>KF
30 '
>LC
40 INPUT "N (N<34) ";n%
>QB
50 :GIVE,2,0nX:GOSUB 140:GET,5,0fact!
>HL
60 PRINT n%;"!=":fact!
>QJ
70 PRINT
>HD
80 GOTO 40
>LB

```

9#		>LJ	28# 'calcul de p(n-1)(x)	>TA
10#	'-----	>RB	29# n2=n-1:GIVE,2,0n2,5,0x:GOSUB 19#	>FN
11#	'routine factorielle	>RC	30# 'recuperation de p(n-1)(x)	>RD
12#	'	>RD	31# :GET,5,0pn	>EA
13#	'definition des variables locales	>RE	32# 'calcul de p(n-2)(x)	>RF
14#	:LOCAL,2,0nX,2,0n2X,5,0fact!,5,0prov!	>KX	33# n2=n-2:GIVE,2,0n2,5,0x:GOSUB 19#	>FJ
15#	'variable en entree: n	>RG	34# 'recuperation de p(n-2)(x)	>RH
16#	:GET,2,0nX	>ZH	35# :GET,5,0p	>VC
17#	'	>RJ	36# 'p(n)(x)=((2*n-1)*p(n-1)(x)-(n-1)*p(n-2)(x))/n	>RK
18#	'si n=1 alors i!=1; retour ensuite de sous-programme	>RK	37# pn=((2*n-1)*pn-(n-1)*p)/n	>WG
e			38#	>TB
19#	IF nX=1 THEN fact!=1:GOTO 29#	>ZC	39# 'transfert de p(n)(x)	>TC
20#	'sinon	>RC	40# :GIVE,5,0pn	>LE
21#	'calcul de (n-1)!	>RD	41# 'restauration des variables locales	>RF
22#	n2X=nX-1:GIVE,2,0n2X:GOSUB 14#	>DT	42# :ENDLOCAL,2,0n,2,0n2,5,0pn,5,0p	>EP
23#	'recuperation de (n-1)!	>RF	43#	>RH
24#	:GET,5,0prov!	>NN	44# RETURN	>ZD
25#	'n!=(n-1)!*n	>RH		
26#	fact!=prov!*nX	>NT		
27#	'	>RK		
28#	'variable en sortie: n!	>TA		
29#	:GIVE,5,0fact!	>NA		
30#	'restauration des valeurs des variables locales avec	>RD		
nt appel				
31#	:ENDLOCAL,2,0nX,2,0n2X,5,0fact!,5,0prov!	>PK		
32#	'	>RF		
33#	RETURN	>ZB		

LEGENDE . BAS

1#	MODE 2:STACK,47000,41000	>WB		
2#	DEFINT i-n:DEFREAL p-x:n=n:n2=n2:pn=p:p=x	>KX		
3#	'	>LC		
4#	INPUT "degre N du polynome et X ";n,x	>RT		
5#	FOR i=1 TO n	>RB		
6#	' transfert de N et X et calcul de p(n)(x)	>LF		
7#	:GIVE,2,0i,5,0x:GOSUB 19#	>YV		
8#	' recuperation de p(n)(x)	>LH		
9#	:GET,5,0pn	>CE		
10#	PRINT "p(";i;)"=";pn	>TA		
11#	NEXT	>DH		
12#	PRINT	>NF		
13#	GOTO 40	>RE		
14#	'	>RF		
15#	'-----	>RG		
16#	'routine de calcul de polynome de Legendre	>RH		
17#	'	>RJ		
18#	'definition des variables locales	>RK		
19#	:LOCAL,2,0n,2,0n2,5,0pn,5,0p	>BA		
20#	'recuperation de n et x	>RC		
21#	:GET,2,0n,5,0x	>NN		
22#	'	>RE		
23#	'si n=# alors p(0)(x)=1	>RF		
24#	IF n=# THEN pn=1:GOTO 40#	>VV		
25#	'sinon, si n=1 alors p(1)(x)=x	>RH		
26#	IF n=1 THEN pn=x:GOTO 40#	>VQ		
27#	'sinon	>RK		
			28# 'calcul de x^(n-1)	
			29# n2X=nX-1:GIVE,2,0n2X,5,0x:GOSUB 17#	>JE
			30# 'recuperation de x^(n-1)	>RJ
			31# :GET,5,0prov!	>NR
			32# 'x^n=(x^(n-1))*x	>TA
			33# y!=prov!*x!	>FH
			34#	>RD
			35# 'transmission de x^n	>RE
			36# :GIVE,5,0y!	>LR
			37# 'restauration des variables locales	>RG
			38# :ENDLOCAL,2,0nX,2,0n2X,5,0y!,5,0prov!	>KU
			39#	>RJ
			40# RETURN	>ZE

PUISS . BAS

1#	MODE 2:STACK,47000,41000	>WB
2#	nX=nX:n2X=n2X:y!=y!:prov!=prov!	>DK
3#	'	>LC
4#	INPUT "reel X, puissance N ";x!,nX	>LJ
5#	'transmission des valeurs de N et X et appel de la r	>LE
outine		
6#	:GIVE,2,0nX,5,0x!:GOSUB 17#	>YF
7#	'recuperation du resultat X^n	>LG
8#	:GET,5,0y!	>UH
9#	PRINT x!;"^";nX="";y!	>TT
10#	PRINT	>ND
11#	GOTO 40	>RC
12#	'	>RD
13#	'-----	>RE
14#	'routine de calcul de puissance	>RF
15#	'	>RG
16#	'definition des variables locales	>RH
17#	:LOCAL,2,0nX,2,0n2X,5,0y!,5,0prov!	>GG
18#	'recuperation des valeurs de n et x	>RK
19#	:GET,2,0nX,5,0x!	>PF
20#	'	>RC
21#	'si n=# alors x^0=1:retour du sous-programme	>RD
22#	IF nX=# THEN y!=1:GOTO 32#	>WZ
23#	'sinon	>RF
24#	'calcul de x^(n-1)	>RG
25#	n2X=nX-1:GIVE,2,0n2X,5,0x!:GOSUB 17#	>JE
26#	'recuperation de x^(n-1)	>RJ
27#	:GET,5,0prov!	>NR
28#	'x^n=(x^(n-1))*x	>TA
29#	y!=prov!*x!	>FH
30#	'	>RD
31#	'transmission de x^n	>RE
32#	:GIVE,5,0y!	>LR
33#	'restauration des variables locales	>RG
34#	:ENDLOCAL,2,0nX,2,0n2X,5,0y!,5,0prov!	>KU
35#	'	>RJ
36#	RETURN	>ZE