

```

620 REM
621 REM
622 REM Affiche l'aide
623 REM
624 FOR S = 0 TO 22
625 IF RE$(S,0) = RE$(S,1) THEN
632
626 SH% = AF$(S,1) + 2 + (S =
17) * 2 + (S = 13) * 11
627 IF S > 18 THEN SH% =
AF$(19,1) + 4
628 VTAB AF$(S,0): HTAB SH%
629 INVERSE
630 PRINT "*";: CALL - 198
631 NORMAL
632 NEXT
633 REM
634 REM
635 REM Affiche score
636 REM
637 GOSUB 3
638 VTAB 22: HTAB 1
639 CALL - 958
640 IF S% = 0 THEN SC% = SC% +
20 - (S * (ES% - 1) * (NI% +
1))
641 IF S% < > 0 AND ES% = 2
THEN SC% = SC% - (S% * 5 * NI%)
642 SC$ = STR$(ABS(SC%))
643 IF LEN(SC$) < > 3 THEN
SC$ = "0" + SC$: GOTO 643
644 IF SC% < 0 THEN SC$ = "-" +
SC$: GOTO 646
645 SC$ = "+" + SC$
646 VTAB 5: HTAB 36
647 FLASH: PRINT SC$;: NORMAL
648 FOR S = 0 TO 3000: NEXT
649 HTAB 36
650 PRINT SC$;
651 RETURN
652 REM
653 REM
654 REM *****
655 REM * *
656 REM * Correction *
657 REM * *
658 REM *****
659 REM
660 REM
661 REM Correction ?
662 REM
663 VTAB 22: HTAB 1
664 PRINT "Voulez vous la
correction ? ";
665 POKE 49168,0: GET R$
666 IF R$ < > "N" AND R$ < >
"0" THEN CALL - 198: GOTO 663
667 IF R$ = "N" THEN GOTO 686

```

```

668 PRINT "OUI";
669 REM
670 REM
671 REM Correction
672 REM
673 FOR S = 0 TO 22
674 IF RE$(S,0) = RE$(S,1) THEN
684
675 CALL - 198: CALL - 198
676 VTAB AF$(S,0): HTAB
AF$(S,1)
677 IF S < > 13 THEN FLASH:
PRINT RE$(S,0);: GOTO 680
678 FLASH: PRINT LEFT$(
AD$(DA$(13)),1);: NORMAL
679 PRINT RIGHT$(
AD$(DA$(13)), LEN
(AD$(DA$(13))) - 1);
680 FOR S0 = 0 TO 4000: NEXT
681 HTAB AF$(S,1)
682 IF S < > 13 THEN INVERSE
: PRINT RE$(S,0);: NORMAL: GOTO
684
683 INVERSE: PRINT LEFT$(
AD$(DA$(13)),1);: NORMAL
684 NEXT
685 GOSUB 34
686 VTAB 22: HTAB 1
687 CALL - 958
688 RETURN
689 REM
690 REM
691 REM
692 REM *
*
693 REM * Programme principal
*
*
694 REM *
*
695 REM
696 REM
697 GOSUB 5: GOSUB 6
698 GOSUB 7
699 FOR QQ = 0 TO 9
700 GOSUB 8: GOSUB 9
701 GOSUB 10: GOSUB 11
702 IF S% = 0 THEN 705
703 IF ES% < > 2 THEN 701
704 IF NI% = 0 THEN GOSUB 12
705 NEXT
706 VTAB 22: HTAB 1
707 PRINT "Une autre partie ?
";
708 POKE 49168,0: GET R$
709 IF R$ < > "0" AND R$ < >
"N" THEN CALL - 198: GOTO 706
710 IF R$ = "0" THEN 698
711 END

```

MICRO7

LE
CAHIER
DULOGICIEL
n°216502
CAHIER DES ASJACKPOT 64
ADDCOLL
POTAGE
GENERMENU

EDITO

Vous êtes les meilleurs ! Nous recevons de plus en plus de programmes de très bonne qualité. Aussi ne désespérez pas si votre programme n'est pas encore publié. Nous vous rappelons que le cahier des as permet de primer les programmes de bon niveau. Par contre, vous pouvez très bien gagner dans le cahier des logiciels même si vous manquez de technique, si vous nous proposez un programme vraiment original. Éventuellement nous corrigeons les gros défauts du programme. Dans tous les cas, accompagnez votre programme d'un texte explicatif.

Avertissement : les imprimantes AZERTY ne possédant pas la fonte ascii américaine, certains signes utilisés pour la programmation n'apparaissent pas. Ils ont cependant un équivalent. C'est le cas pour l'arobas, remplacé par le caractère à et pour le signe dièse remplacé par le symbole £.

Basic : ZX
Difficulté : **
Adaptabilité : **



GENERMENU

Aujourd'hui le chef propose : panade, salsifis et profiterolles, sauce Sinclair, bien sûr. Bon Appétit !

Ce programme sur ZX 81 est chargé de donner des idées de menus à qui en aurait besoin. Il définit un menu à partir de trois critères :

- la saison,
- le temps de préparation des plats,
- leur prix de revient.

Ce programme fonctionne avec une extension de mémoire de 16 Ko. Les plats sont contenus dans un tableau P(150,30). Le codage des plats est l'ensemble des sept premiers caractères de chaque chaîne P(i) :

- Les trois premiers caractères indiquent la saison (trois blancs si la saison importe peu).
- Les trois suivants figurent la rapidité d'exécution.
- Le dernier représente le prix de revient approximatif.

Tout cela est, bien évidemment, très approximatif et l'utilisateur sera tenté de modifier ce code. On peut changer le plat se trouvant derrière les caractères de code afin d'en mettre un qui soit plus adéquat. Pour que le programme ne se plante pas et qu'il donne de bonnes performances il faut 5 entrées, plats, desserts qui puissent convenir pour chaque menu demandé. C'est le

cas avec le tableau proposé ici. Pour rentrer le programme il faut suivre la marche suivante : tout d'abord rentrer le programme en mémoire puis ajouter les lignes suivantes :

```
9000 FOR I=1 TO 150
9010 INPUT P(I)
9020 NEXT I
```

Ensuite vous faites un GOTO 9000 puis vous rentrez les plats tels qu'ils vous sont présentés, dans l'ordre présent, le tableau contient 50 entrées, 50 plats de résistance et 50 desserts. Puis vous supprimez les lignes qui vous ont servi à rentrer les plats, c'est-à-dire les lignes 9000 à 9020. Puis vous sauvez le programme sur cassette par un GOTO 792. En rechargeant le programme vous rechargerez le tableau en zone « Variables ». Pour exécuter le programme il faut faire un GOTO 10, surtout pas de RUN car vous effaceriez le tableau.

La structure du programme est très simple, le listing en donne une bonne idée. ■

Pierre MAUCLAIR

**PROGRAMME GAGNANT
LE MICRO-ORDINATEUR
DU MOIS**

```
533 FLASH : PRINT "*":: NORMAL
534 PRINT SPC( 4);
535 NT% = 22: GOSUB 2
536 IF CA% > 5 THEN 546
537 HTAB SH%
538 PRINT " ";RE$(SP%,1);
539 IF SP% = 19 THEN PRINT
RE$(20,1);RE$(21,1);RE$(22,1);
540 IF CA% = 0 THEN 575
541 SP% = SP% - (CA% > 3) + (CA%
< 4)
542 IF SP% = - 1 THEN SP% =
19
543 IF SP% = 20 THEN SP% = 0
544 IF SP% = 13 THEN 507
545 GOTO 531
546 IF SP% > 18 AND CA% > 7
THEN CALL - 198: GOTO 535
547 HTAB SH% + 1
548 C1$ = CHR$( TC%(CA%))
549 PRINT C1$;
550 GOSUB 2
551 IF (SP% > 18 AND CA% > 7)
OR CA% < 6 THEN CALL - 198:
GOTO 550
552 C2$ = CHR$( TC%(CA%))
553 PRINT C2$;
554 IF SP% = 17 OR SP% > 18
THEN 557
555 RE$(SP%,1) = C1$ + C2$
556 GOTO 535
557 GOSUB 2
558 IF (SP% > 18 AND CA% > 7)
OR CA% < 6 THEN CALL - 198:
GOTO 557
559 C3$ = CHR$( TC%(CA%))
560 PRINT C3$;
561 GOSUB 2
562 IF (SP% > 18 AND CA% > 7)
OR CA% < 6 THEN CALL - 198:
GOTO 561
563 C4$ = CHR$( TC%(CA%))
564 PRINT C4$;
565 IF SP% = 17 THEN RE$(SP%,1)
= C1$ + C2$ + C3$ + C4$: GOTO
535
566 RE$(19,1) = C1$
567 RE$(20,1) = C2$
568 RE$(21,1) = C3$
569 RE$(22,1) = C4$
570 GOTO 535
571 REM
572 REM
573 REM Confirmation sortie
574 REM
575 VTAB 23: HTAB 1
576 CALL - 958
577 PRINT "Autres modifications
```

```
? ";
578 POKE 49168,0: GET R$
579 IF R$ < > "0" AND R$ < >
"N" THEN CALL - 198: GOTO 575
580 IF R$ = "0" THEN PRINT
"OUI";: GOTO 507
581 RETURN
582 REM
583 REM
584 REM
*****
585 REM *
*
586 REM * Analyse des reponses
*
587 REM *
*
588 REM
*****
589 REM
590 REM
591 REM Nombre de reponses
fausses
592 REM
593 S% = 0
594 FOR S = 0 TO 22
595 IF RE$(S,0) < > RE$(S,1)
THEN S% = S% + 1
596 NEXT
597 REM
598 REM
599 REM Affiche reponse fausse
600 REM
601 VTAB 22: HTAB 1
602 CALL - 958
603 IF S% = 0 THEN PRINT
"Reponse exacte.": GOTO 637
604 IF N1% = 1 THEN PRINT
"Reponse fausse.": GOTO 612
605 PRINT "Iy y a ";S%;
"erreur";
606 IF S% > 1 THEN PRINT "s";
607 PRINT ". ";
608 REM
609 REM
610 REM Demande aide
611 REM
612 IF ES% = 2 THEN 637
613 VTAB 23: HTAB 1
614 PRINT "Voulez vous de
l'aide ? ";
615 POKE 49168,0: GET R$
616 IF R$ < > "0" AND R$ < >
"N" THEN CALL - 198: GOTO 613
617 IF R$ = "N" THEN PRINT
"NON";: GOTO 637
618 PRINT "OUI";
619 SC% = SC% - 10 * N1%
```

```

442 REM
443 IF AD% = 0 OR AD% = 2 OR
(AD% = 1 AND CO% = 1) OR (AD% >
3 AND (CO% = 1 OR CO% = 2)) THEN
450
444 DAZ(19) = (VAZ > 127)
445 DAZ(21) = (VAZ = 0)
446 REM
447 REM
448 REM Remplissage des
tableaux
449 REM
450 FOR S = 0 TO 22
451 IF DD%(S) = - 1 THEN
DD%(S) = RND (1) * 255
452 IF DAZ(S) = - 1 THEN
DAZ(S) = DD%(S)
453 NEXT
454 PC% = DAZ(17)
455 ES% = 0
456 RETURN
457 REM
458 REM
459 REM
*****
460 REM *
*
461 REM *Affichage et tableaux
reponse *
462 REM *
*
463 REM
*****
464 REM
465 FOR S = 0 TO 22
466 LZ = 2 - (S = 13 OR S > 18)
+ (S = 17) * 2
467 DZ = DAZ(S): GOSUB 4
468 RE$(S,0) = H$
469 DZ = DD%(S): GOSUB 4
470 RE$(S,1) = H$
471 VTAB AF%(S,0): HTAB
AF%(S,1)
472 IF S = 13 THEN PRINT SPC(
13);: GOTO 475
473 PRINT SPC( 9);
474 HTAB AF%(S,1): PRINT H$;
475 NEXT
476 RE$(13,1) = "0"
477 VTAB 16: HTAB 7
478 PRINT SPC( 12);
479 HTAB 7: PRINT IN$;
480 QUZ = QUZ + 1
481 VTAB 10: HTAB 36
482 FLASH
483 QU$ = STR$(QUZ)
484 IF QUZ < 10 THEN QU$ = "0"
+ QU$

```

```

485 FLASH : PRINT QU$;: NORMAL
486 FOR S = 0 TO 2000: NEXT
487 HTAB 36
488 PRINT QU$;
489 RETURN
490 REM
491 REM
492 REM
*****
493 REM *
*
494 REM * Entree des reponses
*
495 REM *
*
496 REM
*****
497 REM
498 REM
499 REM Reponse mode
d'adressage
500 REM
501 ES% = ES% + 1
502 VTAB 15: HTAB 37
503 FLASH : PRINT ES$;: NORMAL
504 FOR S = 0 TO 1000: NEXT
505 HTAB 37
506 PRINT ES$;
507 SP% = 13:SD% = VAL
(RE$(13,1))
508 VTAB 18: HTAB 20
509 FLASH : PRINT "*";: NORMAL
510 HTAB 21
511 PRINT SPC( 13);
512 HTAB 21
513 PRINT AD$(SD%);
514 NT% = 6: GOSUB 2
515 IF CA% < 2 THEN 520
516 SD% = SD% - (CA% > 3) + (CA%
< 4)
517 IF SD% = - 1 THEN SD% = 9
518 IF SD% = 10 THEN SD% = 0
519 GOTO 510
520 HTAB 7
521 PRINT SPC( 28);
522 HTAB 7
523 PRINT AD$(SD%);
524 RE$(SP%,1) = STR$(SD%)
525 IF CA% = 0 THEN 575
526 REM
527 REM
528 REM Reponse registres et
memoire
529 REM
530 SP% = SP% + 1
531 SH% = AF%(SP%,1) + 2 + (SP%
= 17 OR SP% > 18) * 2
532 VTAB AF%(SP%,0): HTAB SH%

```

```

10 REM
20 REM *****GENERMENU*****
-----
25 REM
30 REM ===INITIALISATION===
32 LET MOD3=0
35 DIM R$(9,30)
40 LET E$="POUS APPUYER SUR NE
*****
50 LET F$="VEUILLEZ RECOMMENCE
R, VOUS AVEZ COMMIS UNE ERREUR."
60 LET G$="*****"
65 REM
66 REM =====AFFICHAGE=====
70 PRINT AT 8,0;G$
80 PRINT AT 10,4;"PROGRAMME DE
GENERATION"
90 PRINT AT 12,10;"DE MENUS"
100 PRINT AT 14,0;G$
105 PRINT AT 20,11;"(C),PIERRE
MAUCLAIR."
110 PAUSE 200
120 CLS
121 REM
122 REM ==INTRODUCTION DES==
123 REM =====CRITERES=====
130 PRINT "QUELLE EST LA SAISON
?"
140 PRINT AT 3,0;"TAPER ""HIV""
POUR HIVER"
150 PRINT TAB 6;""PRI"" POUR P
RINTEMPS"
160 PRINT TAB 6;""AUT"" POUR A
UTOMNE"
170 PRINT TAB 3;"OU ""ETE""
180 PRINT AT 10,1;E$
190 INPUT S$
200 CLS
202 REM CONTROLE DES DONNEES
210 IF S$="HIV" OR S$="PRI" OR
S$="AUT" OR S$="ETE" THEN GOTO 2
40
220 PRINT F$
230 GOTO 140
231 REM
232 REM ==AUTRE CRITERE==
240 PRINT "QUELLE DUREE COMPTER
VOUS PASSERA LA PREPARATION DES
PLATS ?"
250 PRINT AT 5,0;"TAPER ""...""
POUR ASSEZ LONGUE"
260 PRINT TAB 7;""..."" POUR MO
YENNE"
270 PRINT TAB 8;""..."" POUR COU
RTE"
280 PRINT AT 10,1;E$
290 INPUT C$
300 CLS
302 REM CONTROLE DES DONNEES
310 IF C$="..." OR C$="..." OR C
$="..." THEN GOTO 340
320 PRINT F$
330 GOTO 250
331 REM
332 REM == IL FAUT QUE C$ ==
333 REM ==FASSE 3 DE LONGUEUR==
340 IF C$="..." THEN LET C$="...
350 IF C$="..." THEN LET C$="..."
351 REM
355 REM ==AUTRE CRITERE==
360 PRINT "ENTREZ MAINTENANT LE
PRIX DE "
370 PRINT "REVIENT APPROXIMATIF

```

```

QUI "
380 PRINT "VOUS CONVIENT."
390 PRINT AT 5,0;"TAPER ""1"" P
OUR ASSEZ ELEVE"
400 PRINT TAB 6;""2"" POUR MOY
EN"
410 PRINT TAB 6;""3"" POUR FAI
BLE"
420 PRINT AT 10,1;E$
430 INPUT D$
440 CLS
441 REM CONTROLE DES DONNEES
450 IF D$="1" OR D$="2" OR D$="
3" THEN GOTO 480
460 PRINT F$
470 GOTO 390
480 CLS
481 REM
482 REM ===TIRAGE ALEATOIRE===
485 LET MOD3=MOD3+1
490 LET I=INT (RND*49) +1
500 LET L=I
510 LET ALEA=490
512 LET IND=0
520 GOSUB 750
522 LET R$(MOD3)=P$(I)
530 LET J=INT (RND*49) +51
540 LET L=J
550 LET ALEA=530
552 LET IND=3
560 GOSUB 750
562 LET R$(MOD3+3)=P$(J)
570 LET K=INT (RND*49) +101
580 LET L=K
590 LET ALEA=570
592 LET IND=6
600 GOSUB 750
602 LET R$(MOD3+6)=P$(K)
603 REM
604 REM ==AFFICHAGE DU MENU==
610 PRINT AT 0,0;G$
620 PRINT AT 2,12;"MENU"
630 PRINT AT 4,0;G$
640 PRINT AT 7,0;P$(I,8 TO )
650 PRINT AT 10,0;P$(J,8 TO )
660 PRINT AT 13,0;P$(K,8 TO )
670 PRINT AT 16,0;"RECEPTEZ VOUS
CETTE PROPOSITION"
680 PRINT AT 18,0;"APPUYER SUR
""0"" OU ""N"" (AUTRE MENU AVE
C LES MEMES CRITERES)."
681 REM
682 REM (POSSIBILITE DE REFUS)
690 IF INKEY$("<" "N" AND INKEY$("<"
"0" THEN GOTO 690
700 IF INKEY$="N" THEN GOTO 480
701 REM
702 REM ==FORMULE DE FIN==
710 PRINT AT 16,0;G$
720 PRINT AT 18,0;"VOTRE 2X30 V
OUS SOUHAITE,"
730 PRINT AT 19,0;"BON APPETIT"
740 STOP
741 REM
750 REM == SP ADEQUATION ==
750 IF NOT (S$=P$(L,1 TO 3) OR
P$(L,1 TO 3)="") THEN GOTO AL
EA
770 IF NOT C$=P$(L,4 TO 6) THEN
GOTO ALEA
780 IF NOT D$=P$(L,7) THEN GOTO
ALEA
782 IF P$(L)=R$(IND+1) OR P$(L)

```



```

263 REM
*****
264 REM *
*
265 REM * Choix des donnees
question *
266 REM *
*
267 REM
*****
268 REM
269 REM
270 GOTO 399: REM Corps du sous
programme
271 REM
272 REM
273 REM Adressage indirect
274 REM
275 AZ = RND (1) * 10
276 DDZ(AZ) = RND (1) * 255
277 DDZ(AZ + 1) = 10 + RND (1)
* 100
278 DAZ(17) = DDZ(AZ + 1) * 256
+ DDZ(AZ)
279 IN$ = IN$ + "(00" + STR$
(AMZ) + STR$ (AZ) + ")"
280 RETURN
281 REM
282 REM
283 REM Adressage inhérent
284 REM
285 DAZ(17) = DDZ(17) + 1
286 IF COZ > 0 THEN 290
287 DAZ(15) = DDZ(15) - 1:VAZ =
DAZ(15)
288 IF VAZ = - 1 THEN VAZ =
255:DAZ(15) = 255
289 GOTO 293
290 IF COZ = 1 THEN DAZ(22) =
1: GOTO 293
291 IF COZ = 2 THEN DAZ(14) =
DDZ(15):VAZ = DAZ(14): GOTO 293
292 DAZ(16) = DDZ(16) + 1:VAZ =
DAZ(16)
293 RETURN
294 REM
295 REM
296 REM Adressage relatif
297 REM
298 PAZ = RND (1) * 50: IF PAZ
< 10 THEN PAZ = PAZ + 10
299 DZ = PAZ:LZ = 2: GOSUB 4
300 IN$ = IN$ + H$
301 SOZ = COZ: IF SOZ > 1 THEN
SOZ = SOZ - 2
302 DAZ(17) = DDZ(17) + 2 + (SOZ
= DDZ(21 + INT (COZ / 2))) *
PAZ

```

```

303 RETURN
304 REM
305 REM
306 REM Adressage immediat
307 REM
308 PAZ = RND (1) * 255
309 DZ = PAZ:LZ = 2: GOSUB 4
310 IN$ = IN$ + "#" + H$
311 DAZ(17) = DDZ(17) + 2
312 IF COZ < 3 THEN DAZ(14 +
COZ) = PAZ:VAZ = PAZ: GOTO 315
313 VAZ = ABS (DDZ(14) - PAZ)
314 DDZ(22) = (DDZ(14) > =
PAZ)
315 RETURN
316 REM
317 REM
318 REM Adressage absolu
319 REM
320 AZ = RND (1) * 10
321 DDZ(AZ) = RND (1) * 255
322 IN$ = IN$ + "00" + STR$
(AMZ) + STR$ (AZ)
323 DAZ(17) = DDZ(17) + 3
324 RETURN
325 REM
326 REM
327 REM Adressage page zero
328 REM
329 AZ = RND (1) * 10
330 DDZ(AZ) = RND (1) * 255
331 IN$ = IN$ + STR$ (AMZ) +
STR$ (AZ)
332 DAZ(17) = DDZ(17) + 2
333 RETURN
334 REM
335 REM
336 REM Adressage indexe
absolu
337 REM
338 AZ = RND (1) * 10
339 DDZ(AZ) = RND (1) * 255
340 IZ = 1: IF RND (1) < .5
THEN IZ = 0
341 PAZ = (AMZ * 16 + AZ) -
DDZ(15 + IZ)
342 DZ = PAZ:LZ = 4: GOSUB 4
343 IN$ = IN$ + H$ + ","
344 IF IZ = 0 THEN IN$ = IN$ +
"X"
345 IF IZ = 1 THEN IN$ = IN$ +
"Y"
346 DAZ(17) = DDZ(17) + 3
347 RETURN
348 REM
349 REM
350 REM Adressage indexe page
zero

```

```

60 NEXTX
62 PRINT"  "
64 PRINT"  "
66 PRINT"  "
68 PRINT" 100$  $ GAINS $ "
70 PRINT"  "
72 PRINT" 1  "
74 PRINT" 1  "
76 PRINT"  "
78 PRINT" 0  "
80 PRINT"  "
82 PRINT" RETURN  STOP--> * "
84 PRINT"  "
86 PRINT"TTTTTT"
90 POKE53269,7

```

```

92 REM-----
93 REM          SON N0 1
94 REM-----
100 POKE54277,9:POKE54278,0
101 POKE54273,90:POKE54272,220
102 POKE54276,33
103 FORX=1TO100:NEXTX
104 POKE54276,32
105 POKE54273,61:POKE54272,126
106 POKE54276,33
107 FORX=1TO100:NEXTX
108 POKE54276,32
114 GOTO120
115 REM-----
116 REM          TEST CLAVIER
117 REM-----
120 GETA$: IFA$="" THEN120
125 IFA$=CHR$(13) THEN130
127 IFA$=CHR$(32) ANDMI=100 THEN150
128 IFA$="*" THENGOTO500
129 GOTO120
130 MI=100:POKE54273,90:POKE54272,220
132 POKE54276,17:FORX=1TO100:NEXTX
134 POKE54276,16
136 GOTO120
137 REM-----
150 POKE54276,129:FORX=1TO100:NEXTX
152 POKE54276,128
154 REM-----
155 REM          TIRAGE AU SORT
156 REM-----
160 T1=INT(6*RND(TI))
161 T2=INT(6*RND(TI-100))
162 T3=INT(6*RND(TI-200))
165 REM-----
170 FORX=1TO4
172 FORY=0TO5
174 POKE2040,A+Y
175 POKE2041,A+Y
176 POKE2042,A+Y
180 GOSUB250
182 NEXTY
183 NEXTX
185 REM-----
190 FORY=0TOT1
192 POKE2040,A+Y
193 POKE2041,A+Y
194 POKE2042,A+Y
196 GOSUB250
198 NEXTY

```

```

200 FORY=0TOT2
202 POKE2041,A+Y
204 POKE2042,A+Y
206 GOSUB250
208 NEXTY
210 FORY=0TOT3
212 POKE2042,A+Y
214 GOSUB250
216 NEXTY
218 LETG=G+MI
220 IFT1=T2ANDT1=T3 THEN300
225 LETMI=0
230 GOTO120
250 REM-----
251 REM          SON ROULETTE
252 REM-----
255 POKE54277,2:POKE54278,100
256 POKE54276,17:FORZ=1TO50:NEXTZ
257 POKE54276,16
260 RETURN
300 REM-----
301 REM          JACK P0T *****
302 REM-----
310 PRINTTAB(14)G
315 POKE54277,76:POKE54278,9
320 FORX=1TO10
325 POKE54276,129:FORZ=1TO50:NEXTZ
330 POKE54276,128:FORZ=1TO20:NEXTZ
331 NEXTX
332 GETA$: IFA$="" THEN332
335 IFA$=CHR$(13) THEN340
337 GOTO332
340 MI=100:G=0
342 PRINT"TT"
344 PRINTTAB(14)" "
346 PRINT"TT"
350 GOTO120
500 REM-----
510 POKE53269,0
520 FORL=54272TO54296:POKEL,0:NEXTL
530 POKE53280,14:POKE53281,6
540 PRINT"J"
545 PRINT"J"
550 END
600 REM*****
900 REM          DATA  SPRITES
910 REM*****
1000 DATA0,0,0,0,0,17,0,240
1001 DATA15,195,240,15,255,240,0,255,192
1002 DATA0,40,0,0,170,0,0,170,0

```

1003 DATA2,170,128,2,170,128,10,170,160
 1004 DATA10,170,160,10,170,160,10,170,160
 1005 DATA10,170,160,2,170,128,2,170,128
 1006 DATA0,170,0,0,170,0,0,40,0
 1100 DATA0,0,0,0,0,0,0,60,0
 1101 DATA0,255,0,3,215,0,3,215,192
 1102 DATA15,85,240,15,85,240,15,85,240
 1103 DATA63,85,252,0,85,0,1,85,64
 1104 DATA1,85,64,1,85,64,1,85,64
 1105 DATA1,85,64,1,85,64,1,85,64
 1106 DATA1,85,64,0,85,0,0,20,0
 1200 DATA0,0,0,48,0,0,63,0,0
 1201 DATA15,240,0,12,252,0,12,207,0
 1202 DATA12,243,0,12,51,0,3,48,192
 1203 DATA3,48,192,3,48,192,3,12,48
 1204 DATA3,12,48,3,12,48,10,12,160
 1205 DATA10,12,160,10,40,160,0,40,0
 1206 DATA0,40,0,0,0,0,0,0,0
 1300 DATA60,0,60,63,195,252,63,255,252
 1301 DATA1,85,64,4,68,64,1,17,16
 1302 DATA4,68,64,1,17,16,4,68,64
 1303 DATA1,17,16,4,68,64,1,17,16
 1304 DATA4,68,64,1,17,16,4,68,64
 1305 DATA1,17,0,0,68,64,1,17,0
 1306 DATA0,68,64,0,17,0,0,4,0
 1400 DATA8,130,32,8,130,32,10,130,160
 1401 DATA10,130,160,10,130,160,10,130,160
 1402 DATA3,0,192,3,0,192,3,0,192
 1403 DATA3,3,192,3,3,0,3,3,3
 1404 DATA195,195,15,240,195,15,252,195,63
 1405 DATA252,243,60,60,51,60,15,51,60
 1406 DATA15,51,240,3,255,192,0,255,0

RUN DU NOUVEAU

INFORMATIQUE

DISPONIBLES:

COMMODORE 64, 1541, MPS 801,

L'AMSTRAD et ses jeux

les jeux sur MSX

demandez-les...

* dans la limite de nos stocks

new **MICROLIBRARY***

C 64 DISK COMPANION*
Pour apprendre à se servir du 1541 et commander les fichiers. 30060 149 F

ANATOMY OF THE 1541 DISK DRIVE *
Tout ce que vous voulez savoir sur le 1541, son utilisation, les fichiers séquentiels et relatifs. Plus la ROM du 1541 désassemblée. Un MUST... 30050 250 F

Et toujours

INSIDE THE COMMODORE 64 *
La ROM du 64 désassemblée. Que d'adresses utiles voire indispensables! 31005 190 F

THE WORKING COMMODORE 64 *
(D. LAWRENCE M. ENGLAND) 30010 110 F
Pour apprendre à bien utiliser le 64. De nombreux programmes (qui tournent)

COMMODORE 64 MACHINE CODE MASTER *
Un livre pour ceux qui pratiquent le langage machine 30011 129 F

COMMODORE 64 ADVENTURES *
Comment créer et programmer vos propres jeux d'aventures (textuels) sur CBM 64 30012 110 F

USING THE COMMODORE 64 *
Un livre très complet et précis sur le 64 30013 190 F

COMMODORE 64 EXPOSED *
Le tour du 64, en prime : comment programmer les touches de fonction. 30014 131 F

64 GAMES BOOK *
Des programmes de jeux de jeux 30015 145 F

SPRITES AND SOUND on the 64 * 30016 130 F

Et toujours

DR. WATSON COMPUTER LEARNING SERIES :
Beginners Assembly language VIC 20 5110 260 F
Beginners Assembly language CBM 64 5120 260 F
Beginners Assembly language APPLE II 5130 360 F

* Ces ouvrages sont en anglais facile à comprendre
Plus 5 F de port exceptionnel pour chaque ouvrage.

éducation

CONJUGAISON FRANÇAISE **TO SPEAK - I SPOKE**

Un logiciel conçu par une enseignante. Méthode active pour un élève ou une classe. Le moniteur choisit le verbe que l'élève doit conjuguer. Le clavier et l'écran sont accentués. 20006 K7 140 F 20016 disquette 170 F

Les fameux verbes irréguliers anglais. De la 6ème à la terminale. 3 niveaux : débutants, moyens, forts. Dictionnaire. Impression de la liste sur imprimante. 20005 K7 140 F 20015 disquette 170 F

des jeux

NIGHT MISSION (SUBLOGIC). Un vrai FLIPPER sur votre 64. Tout est paramétrable... dingue ! Disk 35003 K7 35013 480 F

FLIGHT PATH 737 (plus extension de 16 K). Un jeu très beau simulateur de vol. Le programme a été écrit par un pilote professionnel. Envolez-vous mais prenez garde à l'atterrissage ! K7 9031 135 F

METAGALACTIC LLAMAS BATTLE Les chercheurs de la Terre ont mis au point les Méta-Animaux, et les Llamas affrontent les Araignées venues de l'espace. K7 10010 99 F

INSECTOR Les étranges créatures de Terron font pleuvoir des bombes sur les humains, ainsi que des insectes affamés pour qui vous feriez un plat très appétissant. K7 2065 95 F

VIC CHESS (plus extension de 16 K). Un jeu d'échecs aux multiples possibilités. Exercez-vous avec un adversaire patient, infatigable... et à la hauteur ! Vous pouvez quitter la partie à tout moment, la sauvegarder sur K7 et la reprendre quand vous en avez envie ou le loisir ! K7 4012 135 F

du sérieux

ZOOM Moniteur langage machine très puissant. Parmi ses fonctions : "trace" rapide, transfert, compare, load, save, verify, assemble et disassemble. Disquette 6550 235 F

ARROW Cartouche autorisant le 64 à charger et sauvegarder à grande vitesse + fonction. Apprend et Verif + moniteur langage machine et autres fonctions 6060 670 F

PRINT LINK 64 Câble pour relier le 64 à toute imprimante parallèle (par exemple Epson) + logiciel permettant d'œuvrer comme une imprimante Commodore. Disquette 6570 540 F

BACK UP Utilitaire vous permettant de réaliser des copies exactes de vos disquettes avec un seul lecteur 6015 195 F

VIC TREE Module ajoutant plus de 40 commandes y compris des instructions se rapprochant du BASIC 4 et des fonctions d'aide à la programmation (manuel de 120 pages en anglais accessible). VIC TREE est un véritable outil pour le programmeur 6090 975 F

GRAPHIX Rend facile l'utilisation de la haute résolution sur votre 64. En langage machine. 20 commandes, se charge en RAM au-dessus du basic. Disquette 6565 235 F

MIKRO ASSEMBLEUR Un assembleur trois passes qui rend le langage machine aussi simple que le basic ! Le code source que vous écrivez est entré comme un programme basic équipé de fonctions de type AUTO, DELETE, FIND, et d'un moniteur langage machine. MIKRO est un bien bel outil ! (cartouche) 6095 950 F

des extensions **AUDIO COMPUTERS**

ZX 81 RAM 16K Dans un boîtier conçu spécialement pour supprimer les problèmes de fixation au ZX 81. RAM 16K 8050 325 F

CLAVIER MÉCANIQUE 8052 65 F (couleur ivoire, légendes rouge et bleu)

VIC 20 (cartouche SRC 32) 8003 32 K
Le VIC 20 africain *** 28159 BYTES FREE *** et donne au système 8192 octets utilisables en code machine... 849 F

8002 16 K (cartouche SRC 16) 549 F

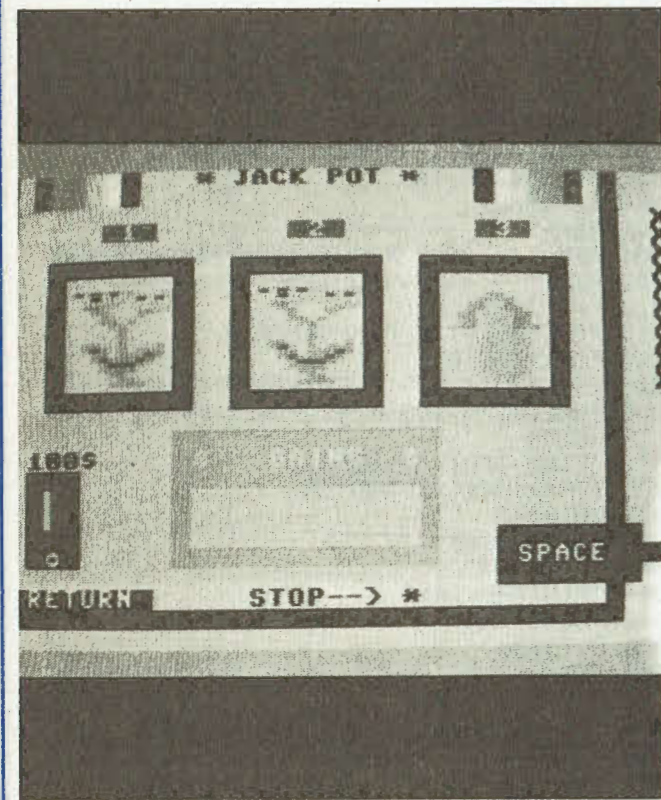
8100 CARTE MERE 349 F

Le QUICKSHOT 2
Équipe d'une option "TR CONTRÔLE" le QUICKSHOT 2 grâce à la nouvelle gâchette vous facilite la tâche. 3002 155 F

SLICKSTICK
Le joystick qui travaille tout en finesse. Une sensibilité extraordinaire. 13010 205 F

RAM 16K

1500 DATA85,84,21,102,100,85,22,81,153
 1501 DATA21,65,85,15,0,84,3,192,240
 1502 DATA0,192,192,0,243,0,0,60,0
 1503 DATA0,12,0,0,12,0,0,12,0
 1504 DATA15,12,12,12,14,204,252,3,207,236
 1505 DATA3,191,176,0,234,192,0,255,0
 1506 DATA0,60,0,0,15,0,0,63,192



1500 DATA85,84,21,102,100,85,22,81,153
 1501 DATA21,65,85,15,0,84,3,192,240
 1502 DATA0,192,192,0,243,0,0,60,0
 1503 DATA0,12,0,0,12,0,0,12,0
 1504 DATA15,12,12,12,14,204,252,3,207,236
 1505 DATA3,191,176,0,234,192,0,255,0
 1506 DATA0,60,0,0,15,0,0,63,192

instruction,"
 190 VTAB 16: HTAB 1
 191 PRINT "Inst:"
 192 IF VI% = 255 THEN 196
 193 GOSUB 3
 194 VTAB 20: HTAB 29: CALL -
 958
 195 PRINT "vous devrez trouver le mode d'adressage."
 196 VTAB 18: HTAB 1
 197 PRINT "M.Ad:"
 198 IF VI% = 255 THEN 219
 199 GOSUB 3
 200 VTAB 20: HTAB 1: CALL -
 958
 201 PRINT "Ensuite, vous exécuterez cette instruction en modifiant la mémoire et les registres du micro."
 202 GOSUB 3
 203 VTAB 20: HTAB 1: CALL -
 958
 204 PRINT "Il y a 10 questions."
 205 PRINT "2 essais par question."
 206 PRINT "Chaque question est notée sur 20."
 207 GOSUB 3
 208 VTAB 20: HTAB 1: CALL -
 958
 209 PRINT "Vous pourrez demander de l'aide ou une correction."
 210 PRINT "2 niveaux de jeu sont permis."
 211 GOSUB 3
 212 VTAB 20: HTAB 1: CALL -
 958
 213 PRINT "Niveau débutant: l'aide et la correction n'enlèvent pas de points."
 214 GOSUB 3
 215 VTAB 20: HTAB 1: CALL -
 958
 216 PRINT "Niveau supérieur: l'aide ainsi que les mauvaises réponses enlèvent des points. Aucune correction."
 217 GOSUB 3
 218 VTAB 20: HTAB 1: CALL -
 958
 219 VTAB 3: HTAB 35
 220 PRINT "Score";
 221 VTAB 8: HTAB 35

222 PRINT "Quest";
 223 VTAB 13: HTAB 35
 224 PRINT "Nb.Es
 225 VI% = 255: SPEED= VI%
 226 RETURN
 227 REM
 228 REM
 229 REM

 **
 230 REM *
 *
 231 REM * Initialisation debut de partie *
 232 REM *
 *
 233 REM

 **
 234 REM
 235 FOR S = 0 TO 9
 236 AD%(S) = 0
 237 NEXT
 238 PC% = RND (1) * 10000: IF PC% < 4000 THEN PC% = PC% + 4000
 239 AM% = RND (1) * 10: IF AM% < 5 THEN AM% = AM% + 5
 240 FOR S = 3 TO 15
 241 VTAB S: HTAB 20
 242 PRINT "00";AM%; CHR\$(TC%(S + 3));
 243 NEXT
 244 SC% = 0:QU% = 0
 245 VTAB 5: HTAB 36
 246 PRINT " ";
 247 VTAB 10: HTAB 36
 248 PRINT " ";
 249 VTAB 15: HTAB 37
 250 PRINT " ";
 251 VTAB 20: HTAB 1
 252 CALL - 958
 253 PRINT "Niveau 'D' ou 'S' ?";
 254 POKE 49168,0: GET R\$
 255 IF R\$ < > "D" AND R\$ < > "S" THEN CALL - 198: GOTO 251
 256 VTAB 20: HTAB 1
 257 CALL - 958
 258 IF R\$ = "D" THEN NI% = 0: PRINT "Niveau débutant."; GOTO 260
 259 NI% = 1: PRINT "Niveau supérieur."
 260 RETURN
 261 REM
 262 REM

```
*
105 REM
*****
*
106 REM
107 REM Titre
108 REM
109 HOME : HTAB 12
110 INVERSE : PRINT " LE JEU DU
6502 " : NORMAL
111 REM
112 REM Presentation du micro
?
113 REM
114 VTAB 10: HTAB 1
115 PRINT "Presentation du
micro ? ";
116 POKE 49168,0: GET R$
117 IF R$ < > "N" AND R$ < >
"O" THEN CALL - 198: GOTO 114
118 PRINT R$
119 VI% = 1: IF R$ = "N" THEN
VI% = 255
120 VTAB 9: CALL - 958
121 REM
122 REM Ecran de depart
123 REM
124 SPEED= VI%
125 IF VI% = 255 THEN 130
126 VTAB 17: HTAB 1
127 PRINT "Le microprocesseur
6502 comprend:"
128 VTAB 19: HTAB 1
129 PRINT "Un accumulateur de 8
bits."
130 VTAB 3: HTAB 1
131 PRINT "Accu: .."
132 IF VI% = 255 THEN 136
133 GOSUB 3
134 VTAB 19: HTAB 1: CALL -
958
135 PRINT "Un registre d'index
X de 8 bits."
136 VTAB 5: HTAB 1
137 PRINT "R.IX: .."
138 IF VI% = 255 THEN 142
139 GOSUB 3
140 VTAB 19: HTAB 1: CALL -
958
141 PRINT "Un registre d'index
Y de 8 bits."
142 VTAB 7: HTAB 1
143 PRINT "R.IY: .."
144 IF VI% = 255 THEN 148
145 GOSUB 3
146 VTAB 19: HTAB 1: CALL -
958
```

```
147 PRINT "Un compteur
programme de 16 bits."
148 VTAB 9: HTAB 3
149 PRINT "PC: ...."
150 IF VI% = 255 THEN 154
151 GOSUB 3
152 VTAB 19: HTAB 1: CALL -
958
153 PRINT "Un pointeur de pile
de 9 bits (1+8)."
154 VTAB 11: HTAB 4
155 PRINT "S: 1.."
156 IF VI% = 255 THEN 160
157 GOSUB 3
158 VTAB 19: HTAB 1: CALL -
958
159 PRINT "Un registre d'etat
de 8 bits, 4 bits
seulement
seront utilises."
160 VTAB 13: HTAB 4
161 PRINT "P: NVZC NVZC"
162 HTAB 7
163 PRINT "....."
164 IF VI% = 255 THEN 172
165 GOSUB 3
166 VTAB 17: HTAB 1: CALL -
958
167 PRINT "Le microprocesseur
6502 utilise:"
168 VTAB 19: HTAB 1
169 PRINT "Une memoire:
Adresses sur 16 bits."
170 HTAB 14
171 PRINT "Donnees sur 8
bits."
172 FOR S = 3 TO 15
173 VTAB S: HTAB 20
174 PRINT ".....: ..";
175 NEXT
176 VTAB 17: HTAB 1: CALL -
958
177 REM
178 REM Explications ?
179 REM
180 VTAB 20: HTAB 1
181 PRINT "Explications ? ";
182 POKE 49168,0: GET R$
183 IF R$ < > "O" AND R$ < >
"N" THEN CALL - 198: GOTO 180
184 PRINT R$
185 VI% = 1: IF R$ = "N" THEN
VI% = 255
186 SPEED= VI%
187 VTAB 20: HTAB 1: CALL -
958
188 IF VI% = 255 THEN 190
189 PRINT "A partir d'une
```

RUN ... TOUJOURS DU NOUVEAU...
INFORMATIQUE

62, rue Gérard 75013 PARIS

(1) 581 51 44
(1) 581 51 05

OUVERT DU LUNDI AU SAMEDI DE 9H à 19H

des jeux

RUN et U.S. GOLD ne vous prennent pas pour des Américains !..

AZTEC CHALLENGE (Jeu d'action). Les religions AZTEC étaient des plus barbares. Chaque année elles offraient en sacrifice des jeunes guerriers à leurs nombreux dieux. Une fois désigné, la seule façon d'échapper à la mort est de passer tous les obstacles et de triompher des pièges. Vous aurez ainsi conquis le fameux AZTEC CHALLENGE.
ATARI : 12002 K7 160 F / C 64 : 12102 K7 160 F 12023 Disk 190 F

BEACH HEAD (Jeu d'action). Une île pacifique est sous la domination d'un terrible dictateur et de ses troupes. Vous êtes le Commandant en Chef des forces d'intervention et devez obtenir une victoire navale rapide puis débarquer sur l'île et enfin capturer la forteresse de KUHN-LIN.
BEACH HEAD a reçu le titre de MEILLEUR JEU (son et graphisme) par le magazine U.S. BILLBOARD. C 64 : 12020 K7 160 F 12120 Disk 190 F

SLINKY (Arcade). Ouille! Slinky a des problèmes! Non seulement il doit changer les couleurs d'un labyrinthe de blocs, mais il doit aussi arrêter les pièges que lui tendent des "méchants" qui veulent que ses ressorts rouillent.
C 64 : 12300 K7 160 F 12301 Disquette 190 F

POOYAN. Un des plus fameux jeu d'arcade. Un jeu classique de bons contre les méchants. POOYAN combine des couleurs, de l'action rapide. POOYAN demande des réflexes pour battre les loups affamés qui veulent manger les petits cochons!
ATARI 35500 K7 160 F
C 64 35009 Disk 190 F - 35000 K7 160 F

THE DALLAS QUEST. A vous de vous mesurer à la fameuse personnalité de TEXAS. Vous serez entraîné de SOUTHFORK jusqu'en Amérique du Sud! L'appât est grand : deux milliards! En plus vous aurez été plus fort que le fameux J.R. EWING (marque déposée).
C 64 : 12021 Disk 250 F
ATARI : 12521 Disk 250 F

SOLO FLIGHT. Le simulateur de vol que vous attendiez, paysage en trois dimensions, planche de bord comportant tous les instruments nécessaires. Une fois que vous avez appris à piloter, testez votre habileté en réalisant des vols postaux entre villes (VOR et IFR). Fantastique simulateur, vous ne quitterez plus les commandes de votre avion.
C 64 : 12016 K7 250 F 35011 Disk 250 F

CAVERN OF KHAFKA. Partez à la recherche du fabuleux trésor de KHAFKA le pharaon. Au plus profond de la caverne des dangers et bien d'autres choses vous barrant l'accès aux richesses. Trouvez les clefs qui ouvrent les passages et finalement débouchez dans la chambre où les trésors attendent depuis des siècles.
ATARI :12522K7 160 F / C 64 :12022K7 160 F 35010 Disk 190 F

FORBIDDEN FOREST. Parti vous exercer au tir à l'arc dans la forêt vous vous êtes perdu. Le soir tombe et d'étranges bruits proviennent des buissons. Vous êtes au cœur de la FORÊT INTERDITE et a vous de vous en sortir.
ATARI : 12533 K7 160 F
C 64 : 12003 K7 160 F ... 12033 Disk 190 F

JEUX D'ÉCHECS

COLOSSUS CHESS : Une énorme bibliothèque d'ouvertures, un niveau de jeu fixé par vous (en déterminant le temps de réponse), la possibilité de chercher, aussi longtemps qu'il le faudra, le meilleur coup possible, ou d'étudier des problèmes.
35120 K7 170 F - 35020 disquette 220 F

GRAND MASTER CHESS : Un programme d'échecs avec 10 niveaux de jeu, de nombreuses options, et la possibilité, si vous êtes débutant, d'avoir à tout moment un véritable professeur à votre disposition.
6020 K7 165 F 6120 disquette - 195 F

ATARI Le fameux ZAXXON sur cassette. Un must pour tous les Atariens ludiques 35012 K7 : 195 F

RUN et SUPERSOFT moins cher qu'en Angleterre!.. **UN PARI TENU**

XERONS (A)
Commandant de la dernière base lunaire vous devez empêcher les ALIENS de débarquer. Les ALIENS attaquent par vagues de plus en plus rapides et meurtrières!!! Un jeu d'arcade signé Andrew TROTT, le maître de STIX (joystick plusieurs niveaux).
6037 K7 : 75 F 6537 Disk 125 F

BURGER CHASE (A)
Freddy est devenu "chef" d'un restaurant de Hamburgers. Il n'arrive pas à satisfaire toute la demande, à vous de l'aider... A vous de composer des hamburgers en recherchant les ingrédients à différents niveaux du restaurant... Attention aux méchantes saucisses, à l'œuf et aux pickles... (plusieurs niveaux; joystick et clavier).
6036 K7 : 85 F 6536 Disk : 135 F

STAR COMMAND. D'épouvantables combats dans l'espace, avec un graphisme 3D qui vous surprendra.
K7 6707 85 F

Et toujours...
HALLS OF DEATH (AG) 6043 K7 99 F 6543 Disquette 149 F
3 D GLOOPERS (A) 6023 K7 99 F 6523 Disquette 149 F
CRAZY KONG (A) 6005 K7 75 F 6505 Disquette 125 F
MANGROVE (A) 6013 K7 75 F 6513 Disquette 125 F

GOBLIN TOWERS (AT) 6042 K7 85 F 6052 Disquette 135 F
STIX (A) 6021 K7 99 F 6521 Disquette 149 F
KAKTUS (A) 6012 K7 85 F 6512 Disquette 135 F
TANK ATTACK (A) 6014 K7 85 F 6514 Disquette 135 F
LORD OF THE BALROGS (AT) 6025 K7 75 F 6055 Disquette 125 F
STREETS OF LONDON (AT) 6041 K7 85 F 6051 Disquette 135 F

PESKY PAINTER. Reprendre le palais du roi quel travail ! On peut bien sûr essayer de passer simplement de la peinture sur la poussière, sans la nettoyer, mais les gardes ont l'œil... Un sacré jeu !
K7 6707 99 F
(A) : ARCADE
(AG) : AVENTURE GRAPHIQUE
(AT) : AVENTURE TEXTUEL

TURTLE JUMP. EKIM (araignée géante) fut découverte par des pirates en 1647. Ceux-ci la dressèrent à la recherche et à la découverte de Trésors. La façon la plus rapide, mais aussi la plus dangereuse méthode est de sauter sur le dos des tortues de mer...
2006 120 F

ENCOUNTER. Cette planète est parsemée d'étranges obéïques noirs, et les habitants n'ont pas l'air très sociables... Ils cherchent même à vous détruire... Un jeu ultra-rapide, des graphismes 3D très impressionnants !
K7 35700 145 F

MULTISOUND SYNTHESIZER. Transformez votre 64 en synthétiseur de musique. Vous pourrez en jouer comme d'un orgue et créer vos propres harmonies, effets sonores et fonds musicaux. PLUS des effets sonores que l'on ne trouve que sur des appareils beaucoup plus sophistiqués...
2038 275 F

THE HOBBIT. La saga de BILBO sur 64. Retrouvez BILBO, GANDALF, THORIN, GOLUM, SMAUG et les autres ! THE HOBBIT : le JEU D'AVENTURE ! Manuel traduit en français.
35007 K7 250 F

TWIN KINGDOM VALLEY. Fantastique jeu d'aventure graphique de BUG BYTE. De très nombreuses chambres et des aventures... Si vous avez un ADMAN SPEECH SYNTHESIZER, le programme vous parlera !
4020 K7 150 F

SPACE PILOT. Voyagez dans le temps, de 1919 à l'an 2000, et affrontez les avions de ces époques pour devenir le maître du ciel. Venez en aide aux parachutistes, ce bon mouvement vous vaudra des points.
K7 9020 130 F

du sérieux

TEXTPRO Traitement de texte simple (pas d'accent) mais efficace. Fonctionne avec des imprimantes COMMODORE et par exemple la MT 80 de Mannesmann (avec PRINTLINK) programme à menu simple.
35005 K7 170 F
HOME MANAGER. Gestion famille : répertoire, agenda, dépenses/recettes (et analyse), comptes bancaires (emprunts ou prêts).
7009 K7 280 F 7010 Disk 280 F

LOGICIELS DE GESTION
DATAPRO La plus simple des bases de données et une des plus performantes pour son prix. Avec le même programme, vous pouvez utiliser pour mettre sur fichier vos collections de disques... vos programmes. Les commandes puissantes de DATAPRO vous permettent de rechercher rapidement toute "fiche" de la base de données. 35006 K7 170 F
FORTH 64. Un outil complet qui vous permettra de FORTHer. Le programme comprend un système d'exploitation sur cassette et un éditeur d'écran.
2026 K7 270 F

BUSICALC 3 Un des calcels les plus puissants du monde pour le 64. BUSICALC 3 possède toutes les fonctions de BUSICALC 2 + 3 dimensions + colonnes de largeur variable + histogrammes + capacité de fonctionnement avec VIZAWRITE et EASYSCRIPT + un mot de passe pour protéger les fichiers + envoi des caractères spéciaux à l'imprimante, par exemple : caractères condensés...
6701 Disk 975 F

BUSICALC UN VRAI « CALC » PROFESSIONNEL. La combinaison de toutes ses caractéristiques et fonctions lui confère un rapport qualité/prix exceptionnel. Plus l'on s'en sert et plus l'on découvre de nouvelles applications. Il n'arrête pas de s'auto-amortir.
VIC 20 K7 6001 299 F* / C 64 K7 6002 299 F*
Disk 6501 339 F / Disk 6022 339 F
CBM 4032 6003 Desq 630 F **CBM 8032** 6004 Disk 630 F

BUSICALC 2 Simple à utiliser, ultra-performant! Possesseurs de 64, vous avez maintenant le choix : BUSICALC ou BUSICALC 2. Quoique la nouvelle version utilise totalement les possibilités mémoire du 64, ce qui permet de travailler sur une plus grande feuille et beaucoup d'autres fonctions, elle conserve la SIMPLICITÉ d'utilisation de la première. BUSICALC 2 ne possède pas de page HELP car il n'en a pas besoin.
BUSICALC 2 vous propose 25 fonctions intrinsèques (incluant les fonctions conditionnelles, statistiques, trigonométriques et 3 fonctions définies par l'utilisateur. BUSICALC 2 possède une fonction "VU" (5 parties de la feuille peuvent être affichées simultanément. Une fonction "FIND" (trouver) ultra-rapide qui recherche dans la page en moins d'une seconde. Chaque "feuille" peut avoir 999 lignes ou 125 colonnes à condition que le nombre total de cases ne dépasse pas 2.500. *
K7 6601 Disquette 6602 600 F

VIZAWRITE
LE TRAITEMENT DE TEXTE EN FRANÇAIS
accents, cédilles, trémas affichés à l'écran. Se connecte aux imprimantes accentuées (Mannesmann TALLY, EPSON FX 80, etc.).
Notice en français. Disquette... 21011 1400 F

BON DE COMMANDE
à découper et à renvoyer à RUN INFORMATIQUE 62, rue Gérard 75013 PARIS

M _____ Je passe commande de : LOGICIEL JEUX Qté ___ N° ___ Qté ___ N° ___ Qté ___ N° ___
LOGICIEL GESTION Qté ___ N° ___ Qté ___ N° ___ Qté ___ N° ___
BUSICALC Qté ___ N° ___
EXTENSION Qté ___ N° ___
MICROLIBRARY N° _____ Matériel _____ Total _____
Ci-joint mon règlement par chèque bancaire ou CCP

Je désire recevoir votre documentation Préciser la machine : _____
Tous les prix comprennent la T.V.A. et les frais de port postaux (France Métropolitaine) Signature des parents pour les moins de 18 ans SIGNATURE : _____

Basic : Microsoft 1.0
 Difficulté : *
 Adaptabilité : **



ADDCOLL

Compter de tête, c'est bien. Avec une virgule baladeuse, c'est risqué. Heureusement TO virgule quatre plus trois est là. Vive la décimale !

Ce petit programme sur TO7 est assez simple mais très amusant, et il peut éventuellement aider à l'acquisition de mécanismes pour l'addition des décimaux.

Après le titre apparaissent deux nombres décimaux. Le premier occupe une place fixe, l'autre peut se déplacer à l'aide des touches flèche-droite et flèche-gauche. L'utilisateur doit d'abord disposer les nombres pour que l'addition soit correctement

posée, puis appuyer sur ENTREE. Si c'est correct la machine tire un trait et complète avec le signe plus. On doit alors effectuer les calculs comme « à la main » mais sous surveillance : tout chiffre inexact est refusé.

Reste alors le problème de la virgule ; elle apparaît sur la droite de l'écran et se déplace grâce à flèche-droite et flèche-gauche. Une fois en place, on valide par ENTREE. Après 5 opérations un bilan est proposé.

Remarques sur le programme :
 lignes 5 à 15 : définition d'un trait d'opération (on peut remplacer par LINE(72,58)-(232,58) en ligne 2120),
 ligne 990 : Compteurs de fautes mis à zéro,
 lignes 1005 à 1190 : espace-ment régulier des chiffres, ex : 72-3 devient 72,3,
 lignes 2000 à 1190 : présentation des deux nombres et test sur l'alignement des unités,

lignes 2500 à 2630 : test de calcul,
 lignes 2700 à 2774 : test sur la position de la virgule dans le résultat,
 lignes 3000 à 3080 : bilan,
 lignes 4000 à 4020 : temporisation, augmenter la longueur des boucles pour un affichage plus long des messages,
 ligne 10000 : liste des nombres à ajouter.

Michel BREUILLER

```
1 *ADDCOLL-----BREUILLER Michel-----
2 * 550 RUE DE L'OREE DU BOIS
3 * 45590 ST CYR EN VAL.TEL (38)65 25 91
4 * 2573 OCTETS
5 CLEAR,,1
10 DEFGR$(0)=0,255,0,0,0,0,0,0
15 FOR I=1 TO20:TR$=TR$+GR$(0):NEXT I
100 * TITRE
110 CLS:SCREEN1,4,4:BOXF(80,72)-(208,160),5:COLOR1,0
120 BOXF(88,80)-(199,152),0:COLOR1,0
130 ATTRB1,1:LOCATE12,12:PRINT"SOMMES":ATTRB0,0:LOCATE17,14:PRINT"DE":ATTRB0,1:LOCATE14,17:PRINT"DECIMAUX":ATTRB0,0:GOSUB4000
990 FT=0:FP=0:FV=0
1000 FOR G=1 TO5
1005 *
1010 READ N1$,N2$:N1=VAL(N1$):N2=VAL(N2$)
1020 L=LEN(N1$):N$=N1$:GOSUB1100:N1$=M$:L1=LEN(N1$):P1=INSTR(N1$,"."):IF P1=0 THEN P1=L1+1
1030 L=LEN(N2$):N$=N2$:GOSUB1100:N2$=M$:L2=LEN(N2$):P2=INSTR(N2$,"."):
1040 GOTO2000
1100 *
1110 M$=""
```

```
1120 FOR I=1 TO L-1
1130 M$=M$+MID$(N$,I,1)+" "
1140 NEXT I
1150 M$=M$+MID$(N$,L,1)
1160 FOR I=1 TO 2*L-1
1170 IF MID$(M$,I,1)=". " THEN M$=LEFT$(M$,I-2)+", "+RIGHT$(M$,2*L-I-2)
1180 NEXT I
1190 RETURN
2000 *
2010 CLS:SCREEN2,0,0:A1$=INKEY$
2020 LOCATE21-P1,3,0:PRINTN1$
2030 C=11
2040 LOCATE C,5,0:PRINTN2$
2050 AA$=INKEY$
2060 IF (AA$=CHR$(8))*(C>9)=1 THEN LOCATE C,5,0:PRINT SPC(14):C=C-2:LOCATE C,5,0:PRINTN2$
2070 IF (AA$=CHR$(9))*(C<25)=1 THEN LOCATE C,5,0:PRINT SPC(14):C=C+2:LOCATE C,5,0:PRINTN2$
2080 IF AA$<>CHR$(13) THEN2050
2090 IF (P2=0)*(C+L2=20)=1 THEN 2110
2100 IF SCREEN(20,5)<>44 THEN LOCATE0,22:PRINT"OPERATION MAL POSEE. RECOMMENCEZ":GOSUB4020:LOCATE0,22:PRINT SPC(38):FP=FP+1:GOTO2050
2110 LOCATE9,5:PRINT"+"
```

```
32 REM
*****
33 REM
34 SPEED=255:VTAB 24:HTAB 34
35 FLASH:PRINT"RETURN";:NORMAL
36 NT%=1:GOSUB 2
37 SPEED=VIX
38 RETURN
39 REM
40 REM
41 REM
*****
42 REM *
*
43 REM * Conversion D% ---> H%
*
44 REM *
*
45 REM
*****
46 REM
47 SD%=D%:H$=""
48 FOR SI=1 TO L%
49 SQ%=SD%/16:SR%=SD%-SQ%*16:SD%=SQ%
50 SH$=STR$(SR%):IF SR%>9 THEN SH$=CHR$(55+SR%)
51 H$=SH$+H$
52 NEXT SI
53 RETURN
54 REM
55 REM
56 REM
*****
57 REM *
*
58 REM * Initialisation debut de programme *
59 REM *
*
60 REM
*****
61 REM
62 DIM AD%(9),AF%(22,1),DA%(22),DD%(22),TC%(21)
63 DIM AD$(9),CO$(4,3),RE$(22,1)
64 FOR S=0 TO 22
65 READ AF%(S,0):READ AF%(S,1)
66 NEXT S
67 FOR S=0 TO 21
```

```
68 READ TC%(S)
69 NEXT S
70 FOR S=0 TO 9
71 READ AD$(S)
72 NEXT S
73 FOR S0=0 TO 4
74 FOR S1=0 TO 3
75 READ CO$(S0,S1)
76 NEXT S1
77 NEXT S0
78 REM
79 REM Donnees pour les tableaux
80 REM
81 REM Positions V et H ecran
82 REM
83 DATA 3,26,4,26,5,26,6,26,7,26,8,26,9,26,10,26,11,26,12,26,13,26,14,26,15,26
84 DATA 18,7,3,7,5,7,7,7,9,7,11,8,14,7,14,8,14,9,14,10
85 REM
86 REM Codes ASCII des touches autorisees
87 REM
88 DATA 13,32,21,10,8,11,48,49,50,51,52,53,54,55,56,57,65,66,67,68,69,70
89 REM
90 REM Modes d'adressage
91 REM
92 DATA Indirect,Inherent,Relatif,Immédiat,Absolu,Page zero,Indexe Abs,Indexe P.zero,Pre indexe X,Post indexe Y
93 REM
94 REM Codes operatoires
95 REM
96 DATA JMP,JMP,JMP,JMP,DEX,SEC,TXA,INY,BNE,BEQ,BCC,BCS,LDA,LDX,LDY,CMP,LDA,STA,STA,LDA
97 REM
98 RETURN
99 REM
100 REM
101 REM
*****
102 REM *
*
103 REM * Affiche ecran et explications *
104 REM *
```


Titre : 6502
Machine : Apple IIeLE CAHIER DES
AS

Le jeu du 6502 est un programme qui permet de tester ses connaissances sur le microprocesseur 6502. Le 6502 est le microprocesseur de l'Apple mais aussi de l'Oric, de l'Atari 600XL et 800XL, du Commodore Vic 20 et de bien d'autres ordinateurs. Tous les utilisateurs de ces ordinateurs sont invités à faire profit de ce programme, seulement s'ils sont intéressés par l'Assembleur bien entendu !

L'ordinateur affiche à l'écran

les registres internes du microprocesseur 6502, 13 cases mémoires et une instruction en langage d'assemblage. Vous devez trouver le mode d'adressage de cette instruction puis réaliser les modifications des registres ou de la mémoire qu'entraîne l'exécution de l'instruction. Le compteur ordinal de programme qui est affiché à l'écran est celui d'avant l'exécution de l'instruction, il faut donc l'incrémenter du nombre d'octets que nécessite l'instruction.

Une partie comprend 10 questions, chaque question est notée sur 20 points et vous avez droit à 2 essais par question. Il y a deux niveaux de jeu : débutant ou supérieur. Pour résoudre le problème posé, vous devez choisir un nombre d'adressage en utilisant la touche « flèche droite » ou « flèche basse » pour avancer et la touche « flèche gauche » ou « flèche haute » pour reculer. La touche « barre d'espace » valide le mode d'adressage. Pour

changer le contenu d'un registre ou d'une case mémoire, tapez le nouveau contenu en hexadécimal sur deux ou quatre digits. Pour passer au registre ou à la case mémoire suivante, utilisez la touche « flèche droite » ou « flèche basse » ou « barre d'espace ». Pour valider votre réponse, appuyez sur la touche « return ». Bonne chance dans votre apprentissage. ■

Richard ORTEGA

```

1 GOTO 697: REM Programme principal
2 GOTO 21: REM Lecture caractere clavier
3 GOTO 34: REM 'RETURN' pour continuer
4 GOTO 47: REM Conversion D% ---> H%
5 GOTO 62: REM Initialisation debut de programme
6 GOTO 109: REM Ecran de depart et explications
7 GOTO 235: REM Initialisation debut de partie
8 GOTO 270: REM Choix des donnees questions
9 GOTO 465: REM Affichage et tableaux reponse
10 GOTO 501: REM Entree des reponses
11 GOTO 593: REM Analyse des reponses
12 GOTO 663: REM Correction
13 REM

```

```

14 REM
15 REM
16 REM *
17 REM * Lecture un caractere au clavier *
18 REM *
19 REM
20 REM
21 POKE 49168,0: WAIT 49152,128: POKE 49168,0: CA% = PEEK (49152)
22 FOR SI = 0 TO NT%
23 IF CA% < > TC%(SI) THEN NEXT : CALL - 198: GOTO 21
24 CA% = SI: SI = NT%: NEXT
25 RETURN
26 REM
27 REM
28 REM
29 REM *
*
30 REM * 'RETURN' pour continuer *
31 REM *

```

```

2120 LOCATE8,7:PRINTTR$
2500 RP=0:R=0:T=0
2510 FOR I=29 TO 11 STEP-2
2520 IF (R=0)*((SCREEN(I,3)<48)+(SCREEN(I,3)>57))*((SCREEN(I,5)<48)+(SCREEN(I,5)>57))=-1 THEN2630
2530 COLOR0,3:LOCATEI,9:PRINT " "
2540 LOCATE I,9
2550 AA$=INKEY$:IF AA$="" THEN2530
2560 IF ASC(AA$)<48 OR ASC(AA$)>57 THEN2530
2570 T1=VAL(CHR$(SCREEN(I,3)))
2580 T2=VAL(CHR$(SCREEN(I,5)))
2590 T=T1+T2+RP
2600 U=T MOD 10:R=T@10
2610 IF VAL(AA$)<>U THEN LOCATE0,22:COLOR1,0:PRINT"RECOMMENCEZ":GOSUB4020:LOCATE0,22:COLOR0,0:PRINT SPC(12):FT=FT+1:GOTO 2530
2620 LOCATEI,9:PRINTAA$:RP=R
2630 NEXT I
2700 COLOR5,0:LOCATE0,22:PRINT"Maintenant, placez la virgule."
2710 C=34
2720 LOCATE C,9,0:PRINT", "
2730 AA$=INKEY$
2740 IF (AA$=CHR$(8))*((C>9)=1) THEN LOCATE C,9,0:PRINT " ":C=C-2:LOCATE C,9,0:PRINT " "
2750 IF (AA$=CHR$(9))*((C<36)=1) THEN LOCATE C,9,0:PRINT " ":C=C+2:LOCATE C,9,0:PRINT " "
2760 IF AA$<>CHR$(13) THEN2730
2770 IF SCREEN(20,9)<>44 THEN LOCATE0,22

```

```

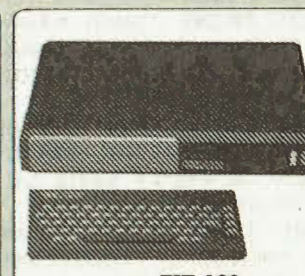
:PRINT"Virgule mal placee. Recommencez":GOSUB4020:LOCATE0,22:PRINT SPC(38):FV=FV+1:GOTO2730
2780 LOCATE0,22:PRINT SPC(13)"C'est BIEN" SPC(13)
2980 COLOR2,0
2990 GOSUB4000
2999 NEXT G
3000 '+++++++ BILAN ++++++++
3010 GOSUB4000:CLS
3020 LOCATE5,5:PRINT"POUR 5 OPERATIONS"
3030 IF FT<>0 THEN LOCATE0,8:PRINT"Vous avez fait ":PRINT:PRINTFT;"Faute";:IF FT>1 THEN PRINT"s de calcul" ELSE PRINT"de calcul"
3040 IF FP<>0 THEN LOCATE0,12:PRINT"Vous avez fait ":PRINT:PRINTFP;"Faute";:IF FP>1 THEN PRINT"s de position" ELSE PRINT"de position"
3050 IF FV<>0 THEN LOCATE0,16:PRINT"Vous avez fait ":PRINT:PRINTFV;"Faute";:IF FV>1 THEN PRINT"s de virgule" ELSE PRINT"de virgule"
3060 IF (FT=0)*(FP=0)*(FV=0)=-1 THEN LOCATE0,8:PRINT"Vous n'avez fait aucune faute"
3070 INPUTWAIT3080;30,A$
3080 CLS:LOCATE0,0,0:SCREEN4,6,6:END 4000 '
4010 FOR II=0 TO 400:NEXT II:RETURN
4020 FOR II=0 TO 230:NEXT II:RETURN
10000 DATA "2345.6","47.987","4.56","529","7658","57.345","0.456","0.9973","77.07","8300.1"

```

VISMO Sélection
Vente Informations Services Micro-Ordinateurs
Micro-ordinateurs familiaux

84, bd Beaumarchais - 75011 Paris
Métro Bastille ou Chemin Vert
De 10 h à 20 h sauf dimanche.

88, bd de Reuilly - 75012 Paris
Métro Daumesnil ou Dugommier
De 14 h à 20 h sauf dimanche et lundi

ZX SPECTRUM
péritel 48 KORIC ATMOS
48 KSANYO PHC 88
nouveau standard MSXTHOMSON MO8
micro-ordinateur françaisEKL 100
commande à distance par infrarouge
micro-ordinateur français

Ordinateur ZX 81 - coffret familial..... 650
Moniteur Zénith..... 1050
Imprimante GP 500 A..... 2450
Imprimante GP 50 A..... 1350
Lecteur K7 ordinateur..... 680
K7 MONASM VISMO pour Oric 1 et Atmos..... 160
Synthétiseur vocal Oric..... 550
Synthétiseur vocal ZX 81..... 435
Catalogue VISMO 20 F remboursable à la 1^{re} commande.
Interface imprimante Minitel. Périphériques. Accessoires. Logiciels. K7 jeux. Bibliographie. Nous consulter Tél. (1) 338.60.00

BON DE COMMANDE
à retourner accompagné de votre règlement à VISMO - 84, bd Beaumarchais - 75011 Paris

Contre-remboursement + 60 F
Règlement joint + 30 F

TOTAL : _____

Catalogue VISMO seul 20 F
(à déduire sur 1^{re} commande)

Nom _____

Prénom _____

Adresse _____

Tél. _____

Signature : _____

Basic : Basic Etendu
Difficulté : **
Adaptabilité : ***



POTAGE

Grand-mère veut faire de la cuisine, mais le jardin est occupé par un écureuil et un spoutnik. Heureusement, le criquet se dévoue pour faire les courses.

Des fruits sont répartis dans un labyrinthe-jardin, le criquet dont vous dirigez le déplacement avec le joystick doit aller chercher un par un et les déposer dans un panier. Chaque fruit déposé augmente le score de 250 points. A 1500 points, vous obtenez un criquet supplémentaire. Quand tous les fruits auront garni le panier,

vous devrez faire passer le criquet sur le téléphone pour commander la prochaine cargaison. Mais vous devrez prendre garde à ce que le spoutnik ne saute pas sur le criquet à un moment malheureux car alors c'en serait fait de lui. De plus, vit dans ce jardin un écureuil qui a la détestable manie de lancer des sacs sur le criquet. Il dépend de votre adresse

que le criquet survive ou trépace. Enfin, clou de la mise en scène, il y a dans ce jardin décidément bien spécial, un ascenseur. Mais par pitié faites attention en l'utilisant. En effet, il ne vous protège pas des sacs dont l'écureuil vous bombarde. Maintenant, le petit truc. Quand vous sentez qu'il pèse une mortelle menace sur votre protégé, vous pou-

vez appuyer sur la touche « bonne chance » le criquet sera peut-être épargné. Le programme en lui-même est assez simple, mais du fait de l'utilisation de caractères graphiques redéfinis, il est assez difficile de l'adapter sur un autre micro.

Fabrice LUCAS

```
80 REM *****
90 REM *** FABRICE LUCAS ***
95 REM *** TI-99/4A ***
99 REM *****
100 SCORE1=2000
110 CALL CLEAR
120 CALL SCREEN(9)
130 CALL COLOR(10,3,3)
140 CALL COLOR(9,3,9)
150 REM DESSIN DU JEU(ANGLE)
160 CALL CHAR(96,"071F3F7F7FFFFFFF")
170 CALL CHAR(98,"1F07030101000000")
180 CALL CHAR(99,"E0F8FCFEFEFFFFFF")
190 CALL CHAR(100,"FFFFFFFEFCFBE0")
200 CALL CHAR(97,"FBE0C0B0B0000000")
210 CALL CHAR(101,"000000010103071F")
220 CALL CHAR(102,"00000080B0C0E0F8")
230 CALL CHAR(103,"FFFFFF7F7F3F1F07")
240 CALL VCHAR(9,3,104,8):: CALL VCHAR(9,31,104,8):: CALL HCHAR(9,3,104,28):: CALL HCHAR(16,3,104,28)
250 DISPLAY AT(12,3):"VEUX-TU DES INSTRUMENTS?"
260 ACCEPT AT(14,14)VALIDATE("OUIN")BEEP =W$
270 IF W$="OUI" OR W$="NON" THEN 280 ELSE GOTO 260
280 REM presentation
290 PRINT "POSITIONNER ALPHA LOCK EN POSITION HAUT"
E" :: FOR T=1 TO 300 :: NEXT T :: CALL CLEAR
300 CALL CLEAR
310 PRINT "hhhhhhhhhhhhhhhhhhhhhhhhhhhhc"
```

```
320 PRINT "hhhhhhhhhhhhhhhhhhhhhhhhhhhhhhhhhhhh"
330 PRINT "hhha bhh"
340 PRINT "hhhh 'hhhhh c 'c 'hhhhh hh"
350 PRINT "hhhh ghhhhhhd gd ghhhhhhd hh"
360 PRINT "hhhhf eh"
370 PRINT "hhhhhhhhhhhhhhhhhhhhhhhhhhhhhhhhhhhh"
380 PRINT "hhhhhhhhhhhhhhhhhhhhhhhhhhhhhhhhhhhd"
390 PRINT "hhha"
400 PRINT "hhhh 'h c 'h c 'h c 'h c 'h c "
410 PRINT "hhhh habh bha habh ha ha "
420 PRINT "hhhh h h h h h h h "
430 PRINT "hhhh h h h h h h h "
440 PRINT "hhhh h h h hfeh h h "
450 PRINT "hhhh h h h hhhh h hf "
460 PRINT "hhhh h h h habhh 'hchh "
470 PRINT "hhhh h h h h h hbhahh "
480 PRINT "hhhh h h h h h h h ha "
490 PRINT "hhhh h h h h h h h h "
500 PRINT "hhhh h h h h h h h h "
510 PRINT "hhhh h h h h h h h h "
520 PRINT "hhhh hfeh h h h h h hf "
530 PRINT "ghhd ghhd ehf g d ghd ghd "
540 IF W$="NON" THEN 650
550 CALL MAGNIFY(2)
560 D$="LE BUT EST DE PRENDRE LES FRUITS EN EVITANT LE SAC ET LE SPOUTNIK . DEPOSER LE FRUIT DANS LE PANIER"
570 G$=D$
580 T$=" LE TELEPHONE SERT A DEMANDER DES FRUITS ET TOUJOURS S'AIDER DE LA TOUCHER BONNE CHANCE..."
590 FOR D=1 TO 99 :: INTRO$=SEG$(G$,D,1)
600 D1=D1+1 :: IF D1=11 THEN D1=1
610 YY=ASC(INTRO$):: CALL SPRITE(#D1,YY,
```

RECTIFICATIF

CADRAN SOLAIRE

Personne n'est parfait. Même pas nous. Nous avons publié, le mois dernier un programme dénommé « Cadran solaire », sur ZX-81: L'idée nous intéressant pour notre usage personnel, nous l'avons adapté sur plusieurs machines. Malheureusement c'est une version de travail qui a été publiée. Pour nous faire pardonner (?), nous publions une version destinée à l'Atmos et une autre au ZX.

ZX

```
10 REM CADRAN SOLAIRE
20 PRINT "RENTREE DES DONNEES : "
30 PRINT "LATITUDE DU MILIEU EN DEGRES ET MINUTES SEXAGESIMALES"
UX : "
40 PRINT "DEGRES : ";
45 INPUT D
50 PRINT "MINUTES : ";
55 INPUT M
60 PRINT "VILLE : ";
65 INPUT A$
70 PRINT "TRAITEMENT : ";
80 FAST
90 LET T=(D+M/60)*PI/180
100 LET B=ACS(-TAN T *TAN(23.45 *PI/180))
110 LET H=12
120 CLS
130 PRINT "GRADUATION D'UN CADRAN SOLAIRE A LA LATITUDE DE "
;A$
140 PRINT TAB 7 ;D;" DEGRES";
150 PRINT M;" MINUTES"
160 PRINT "HEURE HORIZONTAL VERTICAL";
170 FOR A=0 TO B STEP PI/24
180 LET X=SIN T
190 PRINT TAB 0 ;INT H;"H";30 AND H<>INT H;TAB 7 ;
200 GOSUB 300
210 LET X=COS T
220 IF A>PI/2 THEN GOTO 250
230 PRINT TAB 20;
240 GOSUB 300
250 LET H=H+.5
260 NEXT A
270 COPY
280 SLOW
290 STOP
300 LET X=ATN(X*TAN A)
310 LET X=(X+(X<0)*PI)*180/PI
320 LET A$=" "+STR$(INT X)
330 LET B$="0"+STR$(INT((X-INT X)*60))
340 PRINT A$(LEN A$-2 TO );"D";B$(LEN B$ -1 TO);"MN";
350 RETURN
```

ORIC Atmos

```
1 A0=1.5707288:A1=-.2121144:A2=.0742610:A3=-.0187293
5 CLS
10 REM CADRAN SOLAIRE
20 PRINT "RENTREE DES DONNEES : "
30 PRINT "LATITUDE DU MILIEU EN DEGRES ET MINUTES SEXAGESIMALES"
UX : "
40 INPUT "DEGRES : ";D
50 INPUT "MINUTES : ";M
60 INPUT "VILLE : ";A$
70 PRINT "TRAITEMENT : ";
90 T=(D+M/60)*PI/180
100 W = (-TAN(T)*TAN(23.45 *PI/180))
105 GOSUB 10000
110 H=12
120 CLS
130 PRINT "GRADUATION D'UN CADRAN SOLAIRE A LA LATITUDE DE "
;A$
140 PRINT TAB(7);D;" DEGRES";
150 PRINT M;" MINUTES"
160 PRINT "HEURE HORIZONTAL VERTICAL"
170 FOR A=0 TO B STEP .PI/24
180 X=SIN(T)
185 PRINT
190 PRINT TAB(0);INT(H);"H"; 30 AND H<>INT(H);TAB(12);
200 GOSUB 300
210 X=COS(T)
220 IF A>PI/2 THEN GOTO 250
230 PRINT TAB(25);
240 GOSUB 300
250 H=H+.5
260 NEXT A
290 END
300 X=ATN(X*TAN(A))
310 X=(X+ABS(X<0)*PI)*180/PI
320 A$=" "+STR$(INT(X))
330 B$="0"+STR$(INT((X-INT(X))*60))
340 PRINT MID$(A$,LEN(A$)-2,10);"D";MID$(B$,LEN(B$)-1,10);"
MN";
350 RETURN
10000 REM
10010 B=SQR(1-W)*((A3*W+A2)*W+A1)*W+A0
10099 RETURN
```

GERMENU - Suite de la page 68

```
=R$(IND+2) OR P$(L)=R$(IND+3)
TH EN GOTO ALEA
784 IF MOD3=4 THEN LET MOD3=1
790 RETURN
791 REM

792 SAVE "GENERMENU"
793 GOTO 10
794 STOP
```

```
..1HUITRES
..2SALADE NICOISE
..3SOUFFLE AU FROMAGE
..3CAROTTES RAPEES
..1PETITE FRITURE RIVIERE
..2SALADE PIEMONTAISE
..2SOUPE DE POISSON
..3CELERI REMOULADE
..1ESCARGOTS FARCIS
AUT ..3SOUPE AUX POIREAUX
..1LANGOUSTINES MAYONNAISE
..2SALSIFIS A LA CREME
..3GOUGERE
..3COMCOMBRE A LA CREME
..1CUISSES DE GRENOUILLES
..2COQUILLES DE POISSON
..2PIZZA
..3CHOUX-FLEURS EN SALADE
HIU ..1BOUCHEES A LA REINE
..1COQUILLES ST JACQUES
..1COEUR DE PALMIER CREME
..2ALLUMETTES AUX ANCHOIS
..3JAMBON CHAMPIGNONS
..3SALADE DE RIZ
..1COCKTAIL DE CRABE
..2CERVELLE MOUTON SAUTEE
..2AVOCAT MAYONNAISE
..3FRAISE DE VEAU POULETTE
PRI ..1TOURTE AUX OURSINS
..1ASPERGES A LA CREME
..1CRABE MAYONNAISE
..2CROISSANTS AU JAMBON
..3PANADE
..3SALADE DE CHOUX ROUGE
..1BARBARIN GRILLE
..2PIED DE MOUTON POULETTE
..2ANDOUILLE DE VIRE
..3MAQUEREAU GRILLE
ETE ..1PRAIRES FARCIES
..1SARDINES GRILLEES
..1SAUMON FUME
..2MOULES MARINIÈRES
..3LANGUE SAUCE PIQUANTE
..3ROULEAU DE PRINTEMPS
..1PAMPLEMOUSSE CREVETTES
..2CONSUMME
..2PATE
..3BETTERAVES EN SALADE
ETE ..1OUCHE LORRAINE
..2PASTEQUE
..1ESCALOPE A LA CREME
..2BEEFSTEAKS GRILLES
..3POULET AU BRIANI
..3CABILLAU HOLLANDAISE
..1FILET BOEUF MARINE ROTI
..2LAPIN CHOUX-FLEURS
..2FRAISE DE VEAU
..3COTELETTE PORC SAUCE
HIU ..1COG AU VIN
..1CHEUREUIL GRAND VENEUR
..1FOIE VEAU AUX OIGNONS
..2BROCHETTES TZIGANES
..3BOEUF EN DAUBE
..3POISSON COURT-BOUILLON
..1BOUDIN BLANC PUREE
..2BLANQUETTE DE VEAU
..2BROTI DE DINDONNEAU
..3RAGOUT
HIU ..1LOTTE A L'AMERICAINE
..1SANGLIER CHASSEUR
..1ROOSBEEF
..2CARRE DE VEAU EPINARDS
..3GALETTE BRETONNE OEUFS
```

```
..3BOUDIN NOIR PUREE
..1MORUE SAUCE BLANCHE
..2VEAU MARENGO
..2ROGNONS SAUTES PUREE
..3COLOMBO
ETE ..1BROCHET BEURRE BLANC
..2TOMATES FARCIES
..1TOURNEDOS GRILLES
..2PINTADE CHAMPIGNONS
..3RAGOUT VEAU PETITS POIX
..3OMELETTE
..1CAILLES RAISINS SECS
..2LONGE VEAU JARDINIÈRE
..2COTELETTES MOUTON RIZ
..3BOEUF AUX CAROTTES
HIU ..1COUSCOUS
..1CANARD A L'ORANGE
..1GIGOT FLAGEOLETS
..2QUEUE DE COCHON CHOUX
..3BOEUF BOURGUIGNON
..3OEUFS A LA COQUE
..1HOMARD A L'AMERICAINE
..2COLIN BEURRE BLANC
..2CARRE DE PORC ROTI
..3POULET AU CITRON
..1OSSO BOSSO
HIU ..1TRIPES MODE DE CAEN
..1ANANAS AU SUCRE
..2PUDDING
..3SAVARIN
..3MERINGUES
..1MOKA
..2FLAN AU COCO
..2COCKTAIL DE FRUITS
..3CREPES
HIU ..1BABA AU RHUM
..1BUCHE AU CHOCOLAT
..1PIECE MONTEE
..2TARTE
..3GAUFFRES CHANTILLY
..3FROMAGE BLANC
..1CAFE LIEGEOIS
..2BANANA SPLIT
..2BANANES FAMBES
..3RIZ AU LAIT
ETE ..1PARFAIT AU CHOCOLAT
..3CLAFOUTIS
..1FRUITS EXOTIQUES
..2CREME FOUETTEE VANILLE
..3BEIGNETS
..3CONFITURE
..1TARTE CREME PATISSIERE
..2OEUFS EN NEIGE
..2CREME
HIU ..3YAOURTIN
..1PROFITEROLLES
..1NOIX
..1VACHERIN
..2MOUSSE AU CHOCOLAT
..3GATEAU ROULE CONFITURE
..3YAOURT
..1GLACE PRALINEE
..2FLAN A L'ANANAS
..2PETITS SUISSES
..3SAVOIE
HIU ..1DIPLOMATE
..2GATEAU AUX MARRONS
..1FRUITS GIURES
..2FARD BRETON
..3KUGELHOP
..3MARMELADE DE FRUITS
..1PECHE MELBA
..2CHARLOTTE AUX 3 CREMES
..2FRUITS
..3PAIN PERDU
ETE ..1AMANDINE
..1SALADE DE GROSEILLE
```

```
16,168,24,-8,0):: CALL SOUND(100,123,17,
-8,19)
620 FOR U=1 TO 50 :: NEXT U
630 NEXT D
640 IF G$=0$ THEN G$=T$ :: GOTO 590 ELSE
CALL DELSPRITE(ALL)
650 FOR A=1 TO 500 :: NEXT A
660 CALL CLEAR
680 CALL COLOR(11,3,12)
690 CALL COLOR(8,10,12)
700 CALL COLOR(9,7,12)
710 CALL COLOR(10,7,7)
720 CALL COLOR(1,12,12)
730 CALL COLOR(3,12,7)
740 CALL COLOR(4,12,7)
750 CALL SCREEN(7)
760 REM
770 REM DESSIN DU JEU
780 REM
790 REM barreau
800 CALL CHAR(112,"C1E171793D1D0503")
810 CALL CHAR(113,"83878E9EBCB8A0C0")
820 CALL CHAR(114,"0101010101030303")
830 CALL CHAR(115,"80C0C0C080808080")
840 CALL VCHAR(1,1,104,72)
850 FOR T=3 TO 5 :: CALL HCHAR(T,4,104,5)
):: NEXT T
860 FOR T=17 TO 21 :: CALL HCHAR(T,4,104,5)
):: NEXT T
870 FOR T=8 TO 14 :: CALL HCHAR(T,6,104,3)
):: NEXT T
880 FOR T=3 TO 5 :: CALL HCHAR(T,11,104,10)
):: NEXT T
890 FOR T=3 TO 5 :: CALL HCHAR(T,23,104,4)
):: NEXT T
900 FOR T=3 TO 5 :: CALL HCHAR(T,29,104,3)
):: NEXT T
910 FOR T=9 TO 10 :: CALL HCHAR(T,11,104,6)
):: NEXT T
920 FOR T=9 TO 10 :: CALL HCHAR(T,19,104,5)
):: NEXT T
930 FOR T=9 TO 10 :: CALL HCHAR(T,26,104,6)
):: NEXT T
940 FOR T=13 TO 16 :: CALL HCHAR(T,11,104,6)
):: NEXT T
950 FOR T=13 TO 16 :: CALL HCHAR(T,19,104,5)
):: NEXT T
960 FOR T=16 TO 17 :: CALL HCHAR(T,26,104,6)
):: NEXT T
970 FOR T=20 TO 21 :: CALL HCHAR(T,22,104,10)
):: NEXT T
980 CALL VCHAR(11,26,104,3):: CALL HCHAR(13,26,104,4)
):: CALL VCHAR(1,32,104,24)
990 FOR T=20 TO 21 :: CALL HCHAR(T,11,104,9)
):: NEXT T
1000 CALL HCHAR(1,4,97):: CALL HCHAR(2,4,102)
):: CALL HCHAR(1,31,98)
1010 CALL HCHAR(2,31,101):: CALL HCHAR(3,8,99)
):: CALL HCHAR(5,8,100)
1020 CALL HCHAR(3,11,96):: CALL HCHAR(5,11,103)
):: CALL HCHAR(3,20,99)
1030 CALL HCHAR(5,20,100):: CALL HCHAR(3,23,96)
):: CALL HCHAR(5,23,103)
1040 CALL HCHAR(3,26,99):: CALL HCHAR(5,26,100)
):: CALL HCHAR(3,29,96)
1050 CALL HCHAR(5,29,103):: CALL HCHAR(6,4,97)
):: CALL HCHAR(6,31,98)
1060 CALL HCHAR(8,6,96):: CALL HCHAR(8,8,99)
):: CALL HCHAR(14,6,103)
```

```
1070 CALL HCHAR(14,8,100):: CALL HCHAR(16,4,102)
):: CALL HCHAR(17,8,99)
1080 CALL HCHAR(21,8,100):: CALL HCHAR(22,4,97)
):: CALL HCHAR(24,4,102)
1090 CALL HCHAR(9,11,96):: CALL HCHAR(10,11,103)
):: CALL HCHAR(9,16,99)
1100 CALL HCHAR(10,16,100):: CALL HCHAR(13,11,96)
):: CALL HCHAR(16,11,103)
1110 CALL HCHAR(13,16,99):: CALL HCHAR(16,16,100)
):: CALL HCHAR(20,11,96)
1120 CALL HCHAR(21,11,103):: CALL HCHAR(20,19,99)
):: CALL HCHAR(21,19,100)
1130 CALL HCHAR(20,22,96):: CALL HCHAR(21,22,103)
):: CALL HCHAR(16,26,96)
1140 CALL HCHAR(17,26,103):: CALL HCHAR(18,31,98)
):: CALL HCHAR(19,31,101)
1150 CALL HCHAR(22,31,98):: CALL HCHAR(24,31,101)
):: CALL HCHAR(15,31,101)
1160 CALL HCHAR(9,19,96):: CALL HCHAR(10,19,103)
):: CALL HCHAR(9,23,99)
1170 CALL HCHAR(10,23,100):: CALL HCHAR(13,19,96)
):: CALL HCHAR(16,19,103)
1180 CALL HCHAR(13,23,99):: CALL HCHAR(16,23,100)
):: CALL HCHAR(11,31,98)
1190 CALL HCHAR(8,31,101):: CALL HCHAR(9,26,96)
):: CALL HCHAR(11,27,97)
1200 CALL HCHAR(12,27,102)
1210 FOR T=3 TO 24 STEP 2 :: CALL VCHAR(T,9,112)
):: NEXT T
1220 FOR T=4 TO 24 STEP 2 :: CALL VCHAR(T,10,113)
):: NEXT T
1230 FOR T=7 TO 16 STEP 2 :: CALL VCHAR(T,4,112)
):: NEXT T
1240 FOR T=6 TO 16 STEP 2 :: CALL VCHAR(T,5,113)
):: NEXT T
1250 FOR T=4 TO 8 STEP 2 :: CALL VCHAR(T,21,112)
):: NEXT T
1260 FOR T=3 TO 8 STEP 2 :: CALL VCHAR(T,22,113)
):: NEXT T
1270 FOR T=4 TO 8 STEP 2 :: CALL VCHAR(T,27,112)
):: NEXT T
1280 FOR T=3 TO 8 STEP 2 :: CALL VCHAR(T,28,113)
):: NEXT T
1290 FOR T=9 TO 19 STEP 2 :: CALL VCHAR(T,17,112)
):: NEXT T
1300 FOR T=10 TO 19 STEP 2 :: CALL VCHAR(T,18,113)
):: NEXT T
1310 FOR T=12 TO 15 STEP 2 :: CALL VCHAR(T,31,113)
):: NEXT T
1320 FOR T=11 TO 15 STEP 2 :: CALL VCHAR(T,30,112)
):: NEXT T
1330 FOR T=20 TO 24 STEP 2 :: CALL VCHAR(T,20,112)
):: NEXT T
1340 FOR T=21 TO 24 STEP 2 :: CALL VCHAR(T,21,113)
):: NEXT T
1350 CALL VCHAR(6,24,114,14):: CALL VCHAR(6,25,115,14)
1360 REM haie
1370 CALL CHAR(88,"00000000FF24FF24")
1380 CALL HCHAR(8,12,88,4)
1390 CALL HCHAR(8,28,88,4)
1400 CALL HCHAR(2,5,88,3)
1410 CALL HCHAR(2,12,88,8)
1420 CALL HCHAR(2,24,88,2)
1430 CALL HCHAR(2,30,88,2)
1440 CALL HCHAR(12,20,88,3)
1450 CALL HCHAR(12,27,88,3)
1460 CALL HCHAR(16,6,88,2)
1470 CALL HCHAR(15,27,88,3)
```

```

1480 CALL HCHAR(19,12,88,5)
1490 CALL HCHAR(19,26,88,5)
1500 CALL HCHAR(24,11,88,9)
1510 CALL HCHAR(24,5,88,5)
1520 CALL HCHAR(19,23,88)
1530 CALL HCHAR(24,21,88,10)
1540 CALL HCHAR(12,12,88,4)
1550 CALL HCHAR(7,7,88)
1560 REM ecureil
1570 CALL CHAR(124,"1C7F7FFFFF7F1F070707
060E0F0F0F07000080C0C0C89E9ABF3F70FEE0E0
EOFC")
1580 REM sac de noisettes
1590 CALL CHAR(72,"543C1B1C3E3F7F7F3E000
00000000000000000000000000000000000000
000")
1600 REM marque
1610 CALL CHAR(64,"0000000000000000000000
00000000000000000000000000000000000000
000")
1620 REM pomme
1630 CALL CHAR(128,"00000102023B7FFFFFFF
FF7F3F3F1F0F40807B3C1CC2F0F8F8F8F8F0F0E0
C080")
1640 REM salade
1650 CALL CHAR(116,"0000061F6FF1FEFF8671
773A0D0F0707000040F4DE7EFEDFEFF1C63EFCF8
EOC0")
1660 REM panier
1670 CALL CHAR(140,"00000F1020404040E0FF
A8574827100F0000F0080402020207FF15EA12E4
0BF0")
1680 REM telephone
1690 CALL CHAR(80,"00003F7FFFF8F70F1C3B3
737377B7C7F0000FCFEFF1FEFF038DCECECECEDE3
EFE")
1700 REM bordure de l'ascenseur
1710 CALL CHAR(136,"0F312D5D5D5D62DCC080
B0A8A8884830F08CB4BABABA463B03010D151511
120C")
1720 CALL CHAR(132,"304888A8A8B080C0DC62
5D5D5D2D310F0C121115150D01033B46BABABAB4
BCF0")
730 REM spoutnik
1740 CALL CHAR(76,"60700B0509EF5E0F10234
C4F00030C0F060E0A090F76AF038C402F230C00
0F0")
1750 REM ASCENSEUR
1760 CALL CHAR(108,"071A2448488888888888
88888888888888888888888888888888888888
11FF")
1770 REM heros
1780 CALL CHAR(84,"6060100B05096F24120F3
32B2F2727E3060608D0A090F62448F0CCD4F4E4E
4C7")
1790 REM heros mort
1800 CALL CHAR(68,"081C0800102BC5C90F645
B27332B2FE70226722008D4A393F024DEE4CCD4F
4E7")
1810 CALL SPRITE(#1,84,9,176,40,#2,108,1
0,49,185,#12,124,14,1,32,0,4,#14,136,3,3
2,185,#13,132,3,144,185)
1820 CALL SPRITE(#16,72,13,12,44,0,4,#18
,80,5,136,216,#19,64,1,12,44,0,4)
1830 CALL SPRITE(#3,116,3,40,48,#11,128,
6,80,96,#4,116,13,48,152,#15,140,5,80,21
6)
1840 CALL SPRITE(#5,116,3,48,232,#9,128,
6,112,48,#6,116,13,104,216,#17,76,7,136,

```

```

88)
1850 CALL SPRITE(#7,116,3,136,112,#10,12
8,6,176,128,#8,116,13,176,216)
1860 CALL HCHAR(4,12,42,8):: HIGHT$=STR$
(SCORE1):: FOR B=4 TO 1 STEP -1 :: POT$=
(SEG$(HIGHT$,B,1))
1870 CALL HCHAR(4,13+B,(ASC(POT$))): NE
XT B
1880 CALL COLOR(7,16,7)
1890 CALL HCHAR(19,4,84):: CALL HCHAR(19
,5,86):: CALL HCHAR(20,4,85):: CALL HCHA
R(20,5,87):: CALL HCHAR(20,6,53)
1900 IF JEUX>=1 THEN CALL SOUND(600,110,
9,-8,0):: PO=8 :: GOTO 1930
1910 CALL MAGNIFY(3):: PID=1.5 :: TIC=96
:: TAC=128 :: DIR=49 :: PUS=185 :: KA=1
0000 :: PET=0
1920 SCORE=10 :: FIR=1 :: PO=4 :: PO1=4
:: LOC=123 :: LUC=123 :: FOG=41 :: PIG=1
77 :: CRIQ=5
1930 CALL DISTANCE(#1,#12,DIST):: CALL C
OINC(ALL,TRAC):: CALL KEY(1,KU,SU):: IF
SU<>0 AND TRAC=-1 THEN 2070
1940 CALL JOYST(1,X,Y):: CALL COINC(#1,#
2,4,TU):: IF (X+Y)<>0 THEN TOP=TOP+1 ::
GOTO 1960 ELSE TOP=0 :: IF TU=0 THEN 196
0
1950 CALL DELSPRITE(#16):: IF X+Y<>0 THE
N CALL MOTION(#1,0,0):: GOTO 1960 ELSE P
IG=DIR+(FIR*8):: POG=PUS :: GOTO 1990
1960 PIG=PIG+(VIT*(-Y/4)):: POG=POG+(VIT
*(X/4)):: CALL GCHAR((PIG/8)+1,(POG/8)+1
,ZX):: IF TOP<=2 THEN VIT=8 ELSE VIT=16
1970 IF ZX=104 OR ZX=88 OR ZX=96 OR ZX=1
03 OR ZX=99 OR ZX=100 OR PIG>183 OR POG>
239 THEN PIG=PIG-(VIT*(-Y/4)):: POG=POG-
(VIT*(X/4)):: TOP=0
1980 IF WW<=2 THEN CALL MOTION(#16,16,0)
:: CALL SOUND(50,2000+(WW*KA*2),5)ELSE C
ALL POSITION(#19,V1,V2):: CALL SPRITE(#1
6,72,13,V1,V2,0,4)
1990 DIR=DIR+(FIR*8):: CALL LOCATE(#1,PI
G,POG):: CALL LOCATE(#2,DIR,PUS):: CALL
COINC(#1,#16,12,DIR):: IF SORT=-1 THEN
2240
2000 CALL POSITION(#17,SPOUT1,SPOUT2)::
CALL COINC(#2,#14,16,CLAC):: IF CLAC=-1
THEN FIR=1 :: CALL LOCATE(#17,144,32)
2010 BIG1=INT(SPOUT1/8):: BIG2=INT(SPOUT
2/8):: CALL COINC(#2,#13,16,CLIC):: IF C
LIC=-1 THEN FIR=-1
2020 CALL COINC(#1,#16,12,TUOR):: WW=WW+
1 :: IF WW=4 THEN WW=0 ELSE IF TUOR=-1 T
HEN 2240
2030 CALL COINC(#1,#17,14,NAB):: IF NAB=
-1 THEN GOTO 2240 ELSE IF DOMGE=104 OR D
OMGE=88 THEN PO=PO*-1
2035 DISPLAY AT(7,4):PO;PO1;BIG1;BIG2
2040 IF BIG1<=0 OR BIG1>=24 OR BIG2<=0 O
R BIG2>=32 THEN PO=PO*-1 ELSE CALL GCHAR
(BIG1,BIG2,DOMGE)
2050 CALL MOTION(#17,PO,PO1):: CALL COIN
C(#1,#16,12,TORT):: IF TORT=-1 THEN 2240
2060 CALL SOUND(100,311,15,1100,25,-8,25
):: CALL SOUND(100,-8,20,165,20):: IF PI
0=1 OR NAB=-1 THEN 2180 :: IF PET=0 THEN
2170 ELSE 1930
2070 D=0 :: F=0
2080 D=D+1 :: CALL COINC(#1,#2+D,10,AB):

```

```

: IF AB=-1 THEN 2090 :: IF D+2=12 THEN 2
120 ELSE GOTO 2080
2090 IF D+2=9 XOR D+2=10 XOR D+2=11 THEN
CALL PATTERN(#1,128)ELSE GOTO 2110
2100 CALL DELSPRITE(#2+D):: PET=1 :: GOT
0 1940
2110 CALL PATTERN(#1,116):: GOTO 2100
2120 F=F+1 :: CALL COINC(#1,#10+F,10,BB)
:: IF BB=-1 THEN 2130 :: IF F+10=18 THEN
1940 ELSE GOTO 2120
2130 IF F+10=13 XOR F+10=14 THEN CALL SO
UND(120,189,3,-8,6):: GOTO 1940 ELSE GOT
0 2140
2140 IF F+10=15 AND PET=1 THEN CALL PATT
ERN(#1,84):: GOTO 2160 ELSE 2150
2150 CALL COINC(#1,#18,14,HALT):: IF HAL
T=-1 THEN JEUX=1 :: GOTO 1810 ELSE GOTO
1940
2160 CALL SOUND(180,131,6,-8,3,189,2)::
PET=0 :: SCORE=SCORE+250 :: GOTO 1940
2170 PET=.5
2180 IF SCORE<=9 THEN W=1 :: P00=3 :: GO
TO 2220 :: ELSE CALL SOUND(100,392,7,-8,
5)
2190 IF SCORE>=10 AND SCORE<=99 THEN W=2
:: P00=2 :: GOTO 2220 :: ELSE CALL SOUN
D(100,131,7,-8,7)
2200 IF SCORE>=100 AND SCORE<=999 THEN W
=3 :: P00=1 :: GOTO 2220 :: ELSE CALL SO
UND(100,123,5,-8,7)
2210 IF SCORE>=1000 AND SCORE<=9999 THEN
W=4 :: P00=0 :: GOTO 2220 :: ELSE CALL
SOUND(100,175,6,-8,6)
2220 STRESS$=STR$(SCORE):: FOR D=W TO 1

```

```

STEP -1 :: A$=(SEG$(STRESS$,D,1))
2230 CALL HCHAR(4,4,104,P00):: CALL HCHA
R(4,D+3+P00,(ASC(A$))): NEXT D :: GOTO
1930
2240 CALL SOUND(250,139,5):: CALL SOUND(
100,185,0):: CALL SOUND(100,139,0):: CAL
L SOUND(300,110,0,-8,0)
2250 CALL PATTERN(#1,68):: CALL DELSPRIT
E(#16):: PET=0
2260 IF SCORE>=1500 AND FOIS=0 THEN CRIQ
=CRIQ+1 :: FOIS=1
2270 CRIQ=CRIQ-1 :: IF CRIQ=0 THEN 2310
2280 CRIQ$=STR$(CRIQ):: QUET$=SEG$(CRIQ$
,1,1):: CALL HCHAR(20,6,ASC(QUET$))
2290 FOR I=1 TO 500 :: NEXT I
2300 CALL LOCATE(#1,176,40):: CALL PATTE
RN(#1,84):: GOTO 2000
2310 CALL DELSPRITE(ALL):: CALL MAGNIFY(
4):: CALL CLEAR
2320 IF SCORE>=SCORE1 THEN SCORE1=SCORE
:: DEST=1 ELSE DEST=0
2330 CALL SCREEN(12)
2340 ZZ=2 :: FOR U=1 TO 28 :: ZZ=ZZ+7 ::
CALL SPRITE(#U,128,(INT(RND*13)+3),ZZ,(
INT(RND*224)+1),U,U):: NEXT U
2350 CALL CLEAR :: CALL CHARSET
2360 IF DEST=0 THEN DISPLAY AT(17,3):"DO
MMAGE TU AS PERDU..." ELSE DISPLAY AT(17
,3):"BRAVO TU AS PULVERISER LE SCORE"
2370 DISPLAY AT(22,1):" VEUX-TU REJOU
ER ?" :: PRINT ,,,,,,,,,,
2380 ACCEPT AT(17,14)VALIDATE("NOUI")BEE
P:T$
2390 IF T$="OUI" THEN CALL DELSPRITE(ALL)
):: JEUX=0 :: GOTO 110 ELSE STOP

```

LES CHAMPIONS DU MOIS

TI 99 :
R. Malblanc (54 Jarny) - H. Raffa (13 La Ciotat) - H. Mezzasalma (40 Dax) - S.K. Cao Thian (4000 Liège) - A. Ruyer (75 Paris) - P. Jancourt (75 Paris) - F. Lequèvre (38 La Verpillière) - T. Menguy (97 St-Denis-la-Réunion) - S. Dubik (38 Grenoble) - O. Mignon (75 Paris) - J.P. Napoletano (77 Pontault-Combault) - T. Ollivier (95 Soisy) - M. Rousseau (93 Epinay-sur-Seine) - J.M. Pascal (97 Sainte-Clotilde, La Réunion) - A. Senhaji (02 Casablanca, Maroc) - A. Ruyer (75 Paris) - L. Kestelyn (73 Chambéry) - P. Taillepied (75019 Paris) - B. Galleli (92 Asnières) - R. Goiffon (80 Rosières en Santerre) - E. Ledoux (85 Le Château d'Olonne) - G. Jacq (68 Cernay).

SINCLAIR :
Y. Le Chequer (92 Malakoff) - L. Neveux (95 Luzarches) - C. Garaud (Beyrouth, Liban) - E. Szaleniec (95 Pontoise) - L. Depierre (70 Gray) - R. Girard (14 Cintheaux) - J.P. Briochon (80 Abbeville) - P. Malaise (87 Isle).

ORIC :
(67 Strasbourg) - A. Leber (78 Le Chesnay Parly II) - Y. Galois (59 Peronne-en-Maltois) - L. Hivet (60 Gouvieux) - O. Meyer (77 Champagne/Oise) - F. Bernardini (20166 Porticcio) - (67 Strasbourg-Rabertsau) - L. Delattre (80 Abbeville).

TO7 :
C. Levadou (33 Castillon la Bataille) - Y. Chochois (17 Charron) - L. Canals (42 Unieux) - D. Chausson (92 Meudon) - J. Gibanel (94 Cachan) - J.N. Pariolleau (17 Villedoux) - J. Dubouille (31 Tse) - J.C. Paulin (75 Paris) - J. Meunier (85 Vendée) - M. Carrier (38 Voiron) - J. Orsat (21 Dijon) - M. Szikman (59 Lille) - F. Cap (14 Caen) - C. Nava (94 Boissy-St-Léger).

ATARI :
A. Comte (12 Rodez) - P. Leborgne (34 Mauguio) - J.P. Gouttenegre (92 Fontenay-aux-Roses).

SEGA SC 300 :
P. Bordas (91 Ste-Geneviève-des-Bois) - P. Duroussin (38 Tignien Jameyziens).

COMMODORE :
A. Noury (91 Massy) - T. Becker (67 Gundershoffen) - S. Burri (57 Ceting) - T. Belmonte (71 Davayé) - G. Bidi (49 Cholet) - O. Blandin (45 St-Cyr-en-Val) - P. Leprince (35 Rennes) - P. Busuttill (61 Pau).

APPLE II :
C. Kreitz (78 Limay) - P. Daubeuf (78 Triel-sur-Seine) - Y. Le Bras (95 Argenteuil) - C. Degardin (93 Aulnay/Bois).

LYNX :
S. Butel (14 Falaise) - Y. Peres (38 Echirolles).

FX 702p :
L. Hanique (59 Ferrière la Petite) - D. Dubos (93 Villepinte) - F. Truchon-Bar-tès (92 Neuilly/Seine).

ALICE :
D. Girod (25 Morteau).

TRS 80 :
C. Delmas (33 La Réale).

MICRAL REE 8022 :
D. Dupire (62 Arras).

NON NOMMINÉE :
X. Carcaud (92 Colombes) - X. Carlotti (20 Porticcio) - F. Blondeau (78 Verneuil/Seine) - M. Tayssedre (76 Epuville).