RUI MANUEL TITO • MARCO PAULO CARRASCO

SUPER PROGRAMAS

EM BASIC E CÓDIGO MÁQUINA

edições SOCEDITE

RUI MANUEL TITO
MARCO PAULO CARRASCO

edições SOCEDITE

Título:
SUPER PROGRAMAS
Autores:
RUI TITO E MARCO PAULO
Capa:
FRANCO GOMES
Reservados todos os direitos por
Socedite — Edições e Publicações, Lda.
Av. da República, 47, 1.º Dt.º — 1000 LISBOA
1.ª Edição — Julho 1986
Composição — Fotolito e Montagem
SOCEDITE, LDA.
Impressão — Printipo, Lda.

PREFÁCIO

Existindo no mercado, uma variada gama de livros de programas, não achamos que este venha a ser, apenas mais um. Isto devido à qualidade dos programas nele apresentados, que o leitor poderá constatar.

Nesta obra o leitor poderá se divertir com jogos e outros programas, que utilizam por vezes a Linguagem Máquina, tornando-os mais perfeitos e atraentes. possuindo características semelhantes aos comerciais.

Poderá também enriquecer os seus conhecimentos, de Basic e Linguagem Máquina, onde nesta última, os iniciados obterão uma grande ajuda.

Para aqueles que pretendem se divertir, com os programas de jogo, encontrarão completas explicações de como introduzi-los. Existindo vários programas desde, jogos para dois jogadores, até programas onde a rapidez de resposta do leitor, é posta à prova.

O leitor que não quiser somente se divertir, mas também ampliar os seus conhecimentos, encontrará várias rotinas que poderá usar nos seus programas.

Esperamos que esta obra lhe seja de maior utilidade, quer para se divertir, como para ampliar os seus conhecimentos.

Marco Paulo dos Santos Carrasco
Rui Manuel dos Prazeres Tito

CAPÍTULO 1

Como introduzir os programas

Iremos neste Capitulo, esclarecer certas dúvidas, que o leitor possa possuir, quanto à introdução dos programas, que constituem esta obra.

Achamos que não deverá existir, qualquer problema de maior, na entrada dos programas, bastando o leitor seguir rigorosamente as operações que se seguem.

Operações a realizar para a entrada de um programa.

1 — Introduzir o programa em Basic.

2 — Gravá-lo em seguida, usando a informação, que se segue ao texto de introdução ao programa. No caso de possuir duas instruções de **SAVE** usará a primeira, que é a destinada, à gravação do Basic.

Estas são as operações que terá que seguir para introduzir e gravar, qualquer programa. No entanto, acontece que muitos dos programas, são acompanhados de mais uma listagem, além da do Basic. Sendo esta, uma listagem de hexadecimais, que constitui a Linguagem Máquina do programa.

A segunda listagem será introduzida, por um programa em Basic, que iremos descrever mais adiante no 2.º Capitulo (Carregador de hexadecimais). Continuaremos então, as operações necessárias para terminar de introduzir um programa, que seja também constituido por Linguagem Máquina.

Se o programa que estiver a introduzir, só ser escrito em Basic, isto é não possuir listagem de hexadecimais, deverá passar para a 6.ª operação.

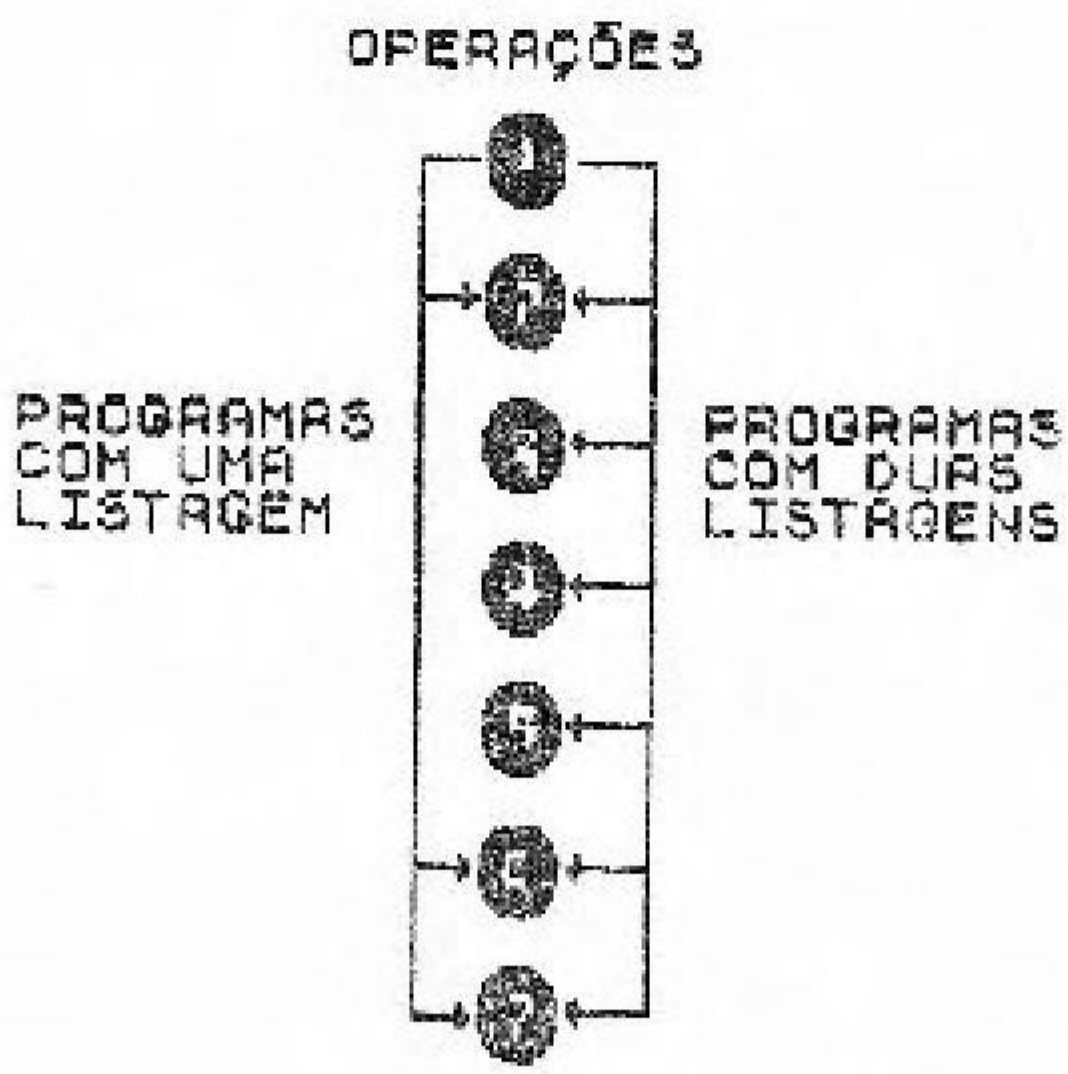
3 — Depois de gravado o Basic, deverá limpar a memória, usando a instrução **NEW**.

4 — Introduza agora o programa Carregador de hexadecimais, que se encontra, no 2.º Capítulo, que lhe irá servir para carregar na memória, os códigos hexadecimais, que constituem a Linguagem Máquina. Este programa encontra-se acompanhado de todas as informações necessárias, para a introdução dos códigos hexadecimais.

5 — Quando terminar de introduzir os códigos, deverá gravá-los, a seguir à gravação que fez do Basic, usando a segunda instrução de **SAVE**, que acompanha a introdução ao programa.

6 — Se tudo tiver sido feito correctamente, basta agora rebobinar a cassete, ao princípio e usar a instrução normal para carregar programas (**LOAD" "**).

7 — Verifique depois de receber o programa da cassete, se este funciona. No caso de não funcionar, é devido a que não o deve ter introduzido correctamente, devendo tentar corrigir os erros. Assim que encontrar os erros, grave o programa, da mesma forma que o fez anteriormente.



CAPÍTULO 2

PROGRAMAS ÚTEIS

Qualquer programador ou utente de microcomputadores, tem necessidade de utilizar, certos programas, que lhe são indispensáveis.

Descreveremos vários programas, que lhe serão úteis, desde um programa para introduzir hexadecimais, até um examinador de programas.

Neste Capítulo estão incluídos os seguintes programas:

Carregador de Hexadecimais
Gerador de Hexadecimais
Desenhador
Construtor de Gráficos
Super Letras
Examinador

CARREGADOR DE HEXADECIMAIS

A utilidade deste programa consiste, como já foi mencionado no 1.º Capitulo, na capacidade que ele possui, para introduzir e colocar os códigos hexadecimais, na memória.

Se o leitor pretende usar este livro a 100%, este programa será muito utilizado, pois os melhores programas, estão escritos em Linguagem Máquina. Sendo necessário para a introduzir, o uso constante deste programa. Por este motivo, deverá guardar este programa numa cassete à parte, para que sempre que dele necessite, não o tenha que introduzir novamente.

FUNCIONAMENTO DO PROGRAMA

As listagens da Linguagem Máquina são constituidas, por três colunas:

- A primeira coluna indica o endereço da memória, onde são introduzidos os códigos hexadecimais.

- A segunda indica os códigos hexadecimais.

- A terceira coluna é constituida pelo total da soma dos códigos hexadecimais, e tem como função verificar se os códigos foram bem introduzidos.

Quando o programa é corrido (**RUN**), este lhe pedirá para entrar o endereço inicial, servindo este para indicar ao programa, onde deve começar a introduzir os códigos hexadecimais no endereço que deve introduzir é o primeiro valor da 1.ª coluna da listagem de hexadecimais). Em seguida o programa lhe pedirá que introduza o endereço final, sendo este o último valor da mesma coluna. Depois desta operação, não necessitará de se preocupar mais com a primeira coluna, pois o programa daqui em diante, gerará ele próprio os

endereços onde são introduzidos os códigos. O programa está agora pronto para receber os códigos, que se encontram na 2.ª coluna e que são introduzidos linha por linha. Deverá introduzir a primeira linha da 2.ª coluna. Em seguida terá que entrar o total, que se encontra na mesma linha, mas na 3.ª coluna. O programa verificará se o total introduzido, corresponde efectivamente à soma dos códigos introduzidos, e só se estiver tudo correcto, o programa colocará os códigos na memória. Caso contrário uma mensagem de erro é produzida, e terá que entrar novamente a linha de códigos que errou.

Gravação

SAVE "C.HEX" LINE 0

```
  10 REM
  20 REM   © (MT) MARCO & TITO
  30 REM
  40 REM
  50 DEF FN H(H$)=16*(CODE H$(1)
-48-(7 AND H$(1)>"9"))+CODE H$(2
)-48-(7 AND H$(2)>"9")
  60 POKE 23658,8: INPUT "ENDERE
CO INICIAL ?? ";E: INPUT "ENDERE
CO FINAL ?? ";E2
  70 FOR N=E TO E2 STEP 10
  80 LET T=0
  90 PRINT N;" ";
 100 INPUT "CODIGOS HEX.";H$: IF
LEN H$<>20 THEN GO SUB 190: GO
TO 100
 110 PRINT H$;: FOR B=1 TO LEN H
$ STEP 2
 120 LET Z=FN H(H$): LET T=T+Z
 130 POKE N+INT (B/2),Z
 140 LET H$=H$(3 TO ): NEXT B
 150 PRINT " ";: INPUT "TOTAL ";
TX
 160 IF T<>TX THEN GO SUB 190: P
RINT : GO TO 60
 170 PRINT TX: NEXT N
 180 STOP
 190 PRINT #1;"   ERRO": BEEP 1,
50: RETURN
```

GERADOR DE HEXADECIMAIS

A utilidade deste programa reside, na possibilidade de criar listagens de hexadecimais, como aquelas que se encontram neste livro.

O leitor que, por algum motivo desejar produzir uma listagem de hexadecimais, deverá quando correr o programa, possuir a impressora ligada ao computador. O programa depois de corrido, pede-lhe que entre o endereço inicial e final, que indicará aó programa, a partir de que endereço da memória, deverá começar e terminar a listagem.

Sempre que usar este programa para produzir uma listagem, terá que usar o programa CARREGADOR DE HEXADECIMAIS para entrar os códigos da listagem.

Gravação

SAVE "G.HEXA" LINE 0

```
  10 REM
  20 REM   © (MT) MARCO & TITO
  30 REM
  40 REM
  50 INPUT "ENDERECO INICIAL ??
";E:  INPUT "ENDERECO FINAL ?? ";
E2:  FOR A=E TO E2 STEP 10
  50 PRINT A;"  ";: LPRINT A;"  ";
: LET TOT=0
  70 FOR G=0 TO 9: LET D=PEEK (A
+G)
  80 LET TOT=TOT+D
  90 LET F=4096: LET H$=""
 100 LET Y=INT (D/F)
 110 GO SUB 200
 120 LET D=D-Y*F: LET F=INT (F/1
6)
 130 IF F<1 THEN GO TO 150
 140 GO TO 100
 150 FOR N=1 TO 2: IF H$(1)="0"
THEN LET H$=H$(2 TO )
```

```
 160 NEXT N
 170 PRINT H$;: LPRINT H$;
 180 NEXT G
 190 PRINT " ";TOT: LPRINT " ";T
OT: NEXT A: STOP
 200 IF Y<=9 THEN LET H$=H$+CHR$
 (Y+48)
 210 IF Y>9 THEN LET H$=H$+CHR$
(Y+55)
 220 RETURN
```

DESENHADOR

Este programa vai dar-lhe a possibilidade de desenhar **SCREEN$** (imagens do écran), com muito mais rapidez.

O programa é capaz de construir circunferências, linhas e até rectângulos.

É de notar que alguns dos **SCREEN$** que se encontram neste livro foram feitos com a ajuda deste programa.

TECLAS DO DESENHADOR

Movimento do cursor A

O — Esquerda
P — Direita
Q — Subir
A — Descer

Movimento do cursor B

5 — Esquerda
8 — Direita
7 — Subir
6 — Descer

Outras funções:

W — Desenha uma linha do cursor **A** ao cursor **B.**

R — Coloca o programa no modo **OVER 1** ou **OVER 0.**
U — Constrói um rectângulo em que os cursores **A** e **B** são vértices diagonalmente opostos.

C — Desenha uma circunferência utilizando o cursor **A** como ponto sobre a circunferência e o cursor **B** como centro.

L — Dá entrada ao passo a ser efectuado pelos cursores.

1 — Move o cursor **A** para as coordenadas do cursor **B.**

2 — Limpa o écran.

3 — Guarda o **SCREEN$** na cassete.

Gravação

SAVE "DESENHA." LINE 0

DESENHADOR

```
  10 REM
  20 REM   © (MT) MARCO & TITO
  30 REM
  40 REM
  50 LET o=0
  60 INPUT "PAPER ? ";p: PAPER p
: INPUT "BORDER ? ";b: BORDER b:
 INPUT "INK ? ";i: INK i: CLS
  70 LET x1=10: LET y1=10: LET x
=100: LET y=100: LET l=1
  80 IF INKEY$="8" AND x+l<=254
THEN LET x=x+l
  90 IF INKEY$="5" AND x-l>=1 TH
EN LET x=x-l
 100 IF INKEY$="7" AND y+l<=174
THEN LET y=y+l
 110 IF INKEY$="6" AND y-l>=1 TH
EN LET Y=y-l
 120 IF INKEY$="p" AND x1+l<=254
 THEN LET x1=x1+l
 130 IF INKEY$="o" AND x1-l>=1 T
HEN LET x1=x1-l
```

```
 140 IF INKEY$="q" AND y1+l<=174
 THEN LET y1=y1+l
 150 IF INKEY$="a" AND y1-l>=1 T
HEN LET y1=y1-l
 160 IF INKEY$="3" THEN PRINT AT
 21,0;"
          ": INPUT "NOME DO SCREEN
?";a$: SAVE a$SCREEN$
 170 OVER 1: PLOT x1-1,y1: DRAW
2,0: PLOT x1,y1-1: PLOT x1,y1+1:
 PLOT x-1,y: DRAW 2,0: PLOT x,y+
1: PLOT x,y-1
 180 PLOT x1-1,y1: DRAW 2,0: PLO
T x1,y1-1: PLOT x1,y1+1: PLOT x-
1,y: DRAW 2,0: PLOT x,y+1: PLOT
x,y-1: OVER 0
 190 IF INKEY$="r" THEN LET o=NO
T o
 200 IF INKEY$="c" THEN GO SUB 2
90
 210 IF INKEY$="u" THEN GO SUB 3
30
 220 IF INKEY$="l" THEN INPUT "P
ASSO ? ";l
 230 IF INKEY$="2" THEN CLS
 240 IF INKEY$="1" THEN LET x=x1
: LET y=y1
 250 IF INKEY$="w" THEN DRAW OVE
R o;x1-x,y1-y
 260 IF o=1 THEN PRINT AT 21,0;"
OVER 1 "
 270 IF o=0 THEN PRINT AT 21,0;"
OVER 0"
 280 GO TO 80
 290 LET j=SQR ((ABS (x1-x)↑2)+(
ABS (y1-y)↑2))
 300 IF x+j>255 OR x-j<0 OR y+j>
175 OR y-j<0 THEN PRINT AT 21,0;
"MUITO GRANDE...": BEEP 1,5: PRI
NT AT 21,0;"                    ": RE
TURN
 310 CIRCLE OVER o;x,y,j
 320 RETURN
 330 PLOT OVER o;x,y: DRAW OVER
o;0,ABS y1-y: DRAW OVER o;-(ABS
x-x1),0: DRAW OVER o;0,ABS y-y1:
 DRAW OVER o;-(ABS x1-x),0
 340 RETURN
```

CONSTRUTOR DE GRÁFICOS

Um programa de extrema utilidade, que irá ajudá-lo a construir os seus gráficos com bastante facilidade.

Servindo-se de uma grelha de 8×8 bits, e de um cursor, você poderá construir o gráfico, e vê-lo ao mesmo tempo no seu tamanho normal. Além disso o programa indicar-lhe-á, os números correspondentes ao gráfico, caso queira utilizá-lo noutro programa.

O programa também grava automaticamente os gráficos, bastando apenas carregar na tecla indicada.

Gravação

SAVE "C.GRAF" LINE 0

```
CONSTRUTOR DE GRAFICOS

   10 REM
   20 REM   © (MT) MARCO & TITO
   30 REM
   40 REM
   50 POKE 23658,8
   60 GO SUB 450
   70 PAPER 1: BORDER 1: INK 9: C
LS
   80 PRINT AT 17,0;" D - DECIMAI
S "
   90 PRINT AT 19,0;" C - CONSTRU
IR GRAFICOS "
  100 PRINT AT 10,5; INVERSE 1;"C
ONSTRUTOR DE GRAFICOS"
  110 PRINT AT 21,0;" G - GRAVAR
GRAFICOS"
  120 PRINT AT 14,0;"5 <=";AT 14,
27;"=> 8"
  130 LET F=144
  140 PRINT AT 14,5;"ABCDEFGHIJKL
MNOPQRSTU"
  150 LET A=USR CHR$ F: LET H=INT
 (A/256): LET L=A-(H*256)
```

```
 160 PRINT AT 15,5;"ABCDEFGHIJKL
MNOPQRSTU"
 170 POKE 30000,12: POKE 30001,8
8: POKE 30002,L: POKE 30003,H
 180 LET L=USR 32000
 190 PRINT AT 4,6; PAPER 0; INK
5;CHR$ F
 200 PRINT AT 15,F-139; PAPER 0;
 INK 5;CHR$ F
 210 LET F=F+(INKEY$="8" AND F<1
64)-(INKEY$="5" AND F>144)
 220 IF INKEY$="D" THEN PRINT AT
 17,1; FLASH 1;"D": GO SUB 420
 230 IF INKEY$="C" THEN PRINT AT
 19,1; FLASH 1;"C": GO SUB 260
 240 IF INKEY$="G" THEN PRINT AT
 21,1; FLASH 1;"G": INPUT "NOME
?";O$: SAVE O$CODE USR "A",168:
PRINT AT 21,1; FLASH 0;"G"
 250 GO TO 150
 260 DIM X(8,8)
 270 FOR I=0 TO 7: PRINT AT I,12
; PAPER 0;"          ": NEXT I
 280 PRINT AT 21,0; PAPER 3;" 0
- PINTAR  9 - APAGAR F - FIM "
 290 LET K=0: LET L=0
 300 IF INKEY$="8" THEN LET K=K+
1: IF K=8 THEN LET K=0: LET L=L+
1: IF L=8 THEN LET L=0
 310 IF INKEY$="5" THEN LET K=K-
1: IF K=-1 THEN LET K=7: LET L=L
-1: IF L=-1 THEN LET L=7
 320 PRINT AT L,K+12; PAPER 8; I
NK 8;"+": BEEP .01,-40: PRINT AT
 L,K+12; PAPER 8; INK 8;" "
 330 IF INKEY$="0" THEN LET X(L+
1,K+1)=1: PRINT AT L,K+12; PAPER
 5;" "
 340 IF INKEY$="9" THEN LET X(L+
1,K+1)=0: PRINT AT L,K+12; PAPER
 0;" "
 350 IF INKEY$="F" THEN PRINT AT
 21,0;" G - GRAVAR GRAFICOS
        ": GO TO 370
 360 GO TO 300
 370 FOR D=0 TO 7: LET B=0: FOR
S=0 TO 7
 380 IF X(D+1,S+1)=1 THEN LET B=
B+2↑(7-S)
 390 NEXT S
 400 POKE A+D,B: NEXT D
 410 PRINT AT 19,1;"C": RETURN
 420 FOR I=A TO A+7: PRINT AT I-
A,22;PEEK I: NEXT I
```

```
 430 PAUSE 0: PAUSE 0: PRINT AT
17,1;"D": FOR U=0 TO 7: PRINT AT
 U,22;"    ": NEXT U
 440 RETURN
 450 FOR Z=32000 TO 32068: READ
A: POKE Z,A: NEXT Z
 460 RETURN
 470 DATA 175,50,63,125,42,50,11
7,6,8,237,91,48,117,125,203,23,2
20,50,126,212,57
 480 DATA 125,16,246,237,91,48,1
17,229,33,32,0,25,34,48,117,225,
35,55,63,126,254,7,200,50,50,63,
125
 490 DATA 24,213,245,62,40,18,24
1,19,201,245,175,18,241,19,201,0
,0,0,0,0,0
```

NOTA:

AS LETRAS QUE SE ENCONTRAM NA LINHA 160 SAO EM MODO GRAFICO

SUPER LETRAS

Em muitos microcomputadores pode-se encontrar, uma variada gama de tipos de letras. No entanto nos micros a que este livro se destina, estes só possuem um tipo de letra.

Vamos apresentar aqui, um programa que gera letras com o dobro da altura, que está acostumado a presenciar.

Poderá utilizar este programa integrado no seu programa (como nós fizemos no Capítulo 4, no programa COMBATE) ou utilizá-lo só para produzir **SCREEN$**, ou seja imagens de écran. Servindo também para o título de um texto que pretenda efectuar.

Gravação

SAVE "S.LETRAS" LINE 0
SAVE "CODE" CODE 31000,160

SUPER LETRAS

```
  10 REM
  20 REM   © (MT) MARCO & TITO
  30 REM
  40 REM
  50 LOAD ""CODE
  60 INPUT "MENSAGEM ? ";A$
  70 INPUT "LINHA ";B;" COLUNA "
;C
  80 IF C>31 OR B>22 THEN GO TO
70
  90 POKE 31000,C: POKE 31001,B
 100 LET ED=30000
 110 FOR A=1 TO LEN A$
 120 POKE ED,CODE A$(A): LET ED=
ED+1
 130 NEXT A
 140 POKE ED,255
 150 LET L=USR 31041
 160 GO TO 60
```

LINGUAGEM MAQUINA

```
31000  1100AF324079F579E618  1046
31010  F64067F1646778E6079F  1251
31020  0F0F816F1A773A40793C  718
31030  FE0813C3324079C31E79  1052
31040  07213875 7EFEFFC823E5  1304
31050  CD5079E118F447FE2038  1312
31060  4DFE3039492A355C1108  793
31070  001910FD0600811A5797E  738
31080  CB1F1213121313310F8ED  845
31090  4B187911A679CD1A7904  800
31100  11AE79CD1A79ED4B1879  1121
31110  0CED431879 79FE20C00E  1074
31120  000404ED431879 78FE18  855
31130  D8080 0ED431879C93E3F  907
31140  18AA00001E1E20201E1E  378
31150  01012121 11E1E00000000  128
```

EXAMINADOR

Programa de muita utilidade, para examinar os programas, que se pretendem duplicar ou que se necessite de conhecer mais sobre estes, além do simples nome.

O programa depois de ser corrido (**RUN**), fica procurando um programa para examinar (**LOAD**), assim que encontre um programa na cassete, este indicará ao nosso programa o seu nome, comprimento, endereço inicial e de que tipo se trata.

Não sendo possível conhecer, toda esta informação com o simples carregar do programa, reside aqui a utilidade deste Examinador de Programas.

Gravação :

SAVE "EXAMINADOR" LINE 0

```
E X A M I N A D O R

  10 REM
  20 REM   © (MT) MARCO & TITO
  30 REM
  40 REM
  50 PAPER 0: INK 5: BORDER 0
  60 CLEAR 31999: GO SUB 360
  70 LET FIM=32010: DEF FN F(X)=
PEEK (FIM+X)+256*PEEK (FIM+X+1)
  80 CLS : PRINT AT 3,0; PAPER 5
; INK 0;"          E X A M I N A D
O R          "
  90 PRINT AT 21,0;"
                    "
 100 RANDOMIZE USR 32000
 110 LET X=PEEK FIM
 120 IF X>3 THEN GO TO 80
 130 PRINT AT 5,0;
 140 PRINT INK 4;"NOME DO PROGRA
MA : "
```

```
 150 FOR A=FIM+1 TO FIM+10: PRIN
T CHR$ PEEK A;: NEXT A
 160 PRINT : PRINT INK 4;"TIPO :
"
 170 IF X=1 THEN GO SUB 250
 180 IF X=2 THEN GO SUB 260
 190 IF X=3 THEN GO SUB 300
 200 IF X=0 THEN GO SUB 220
 210 POKE FIM,255: GO SUB 420: G
O TO 80
 220 PRINT "Program": PRINT INK
4;"Tamanho :"; INK 5;FN F(11);"
Bytes"
 230 PRINT INK 4;"Tamanho do pro
grama :"; INK 5;FN F(15);" bytes
"
 240 IF FN F(13)>9999 THEN PRINT
 "Somente LOAD": RETURN
 250 PRINT "RUN na linha ";FN F(
13): RETURN
 260 PRINT "Number array"
 270 LET Z$="": GO TO 290
 280 PRINT "Character array": LE
T Z$="$"
 290 PRINT INK 4;"Tamanho do arr
ay :"; INK 5;FN F(11);" Bytes"
 300 LET P=PEEK (FIM+14)
 310 PRINT "Nome do array: ";CHR
$ (64+32*(P/32-INT (P/32)));Z$
 320 RETURN
 330 IF FN F(11)=6912 AND FN F(1
3)=16384 THEN PRINT "Imagem de e
cran (SCREEN$ )": RETURN
 340 PRINT "Bytes"
 350 PRINT INK 4;"Endereco inici
al :"; INK 5;FN F(13)
 360 PRINT INK 4;"Tamanho :"; IN
K 5;FN F(11);" Bytes"
 370 RETURN
 380 RESTORE 400
 390 FOR A=32000 TO 32009: READ
C: POKE A,C: NEXT A
 400 DATA 175,55,221,33,10,125,2
06,86,5,201
 410 RETURN
 420 PRINT INK 6;AT 21,3;"prima
uma tecla para LOAD"
 430 IF INKEY$="" THEN GO TO 430
 440 RETURN
```

CAPÍTULO 3

SUPER JOGOS PARA DOIS JOGADORES

Neste Capitulo, iremos apresentar alguns exemplos de jogos para dois jogadores.

Estes jogos são bastante divertidos, pois o adversário, deixa de ser a máquina, para ser um humano.

Com estes jogos você e os seus amigos, poderão desfrutar de momentos bastante divertidos, onde a competição, está presente, para decidir o vencedor.

O Capitulo é constituido por três programas:

Duelo
Abismo
Karting 2000

DUELO

Um jogo em que você defronta um amigo seu, para um duelo até à morte.

Neste duelo você e o seu adversário, estão equipados com Lasers nos discos voadores, que servem de suporte, para o combate.

Durante um período de tempo, tente acertar o mais vezes possível no seu adversário, do que ele em si. Para que consiga assim vencer o duelo.

BOA SORTE

Gravação

SAVE "DUELO" LINE 0
SAVE "DUELO CODE" CODE 31000,640

GRAFICOS

A-□ B-▲ C-▬

MT DUELO

POR: MARCO & TITO

D U E L O

```
  10 REM
  20 REM   © (MT) MARCO & TITO
  30 REM
  40 REM
  50 LOAD ""CODE : GO SUB 490
  60 PAPER 7: BORDER 7: INK 0: C
LS : PRINT AT 6,0; INK 4; INVERS
E 1;"              D  U  E  L  O
   "
  70 LET A$="                    
                    ": PRINT AT 5,0; IN
K 7;A$;AT 7,0;A$
  80 PRINT AT 10,0; INK 1;"     J
OGADOR 1       "; INK 2;"JOGADOR 2
"
  90 PRINT AT 12,5; INK 1;"Q-CIM
A";AT 12,19; INK 2;"O-CIMA"
 100 PRINT AT 13,5; INK 1;"A-BAI
XO";AT 13,19; INK 2;"L-BAIXO"
 110 PRINT AT 14,5; INK 1;"S-LAZ
ER";AT 14,19; INK 2;"ENTER-LAZER
"
 120 PRINT AT 21,0; INK 3;"  PRI
MA UMA TECLA PARA INICIAR"
 130 POKE 31539,2: POKE 31540,5:
 POKE 31545,1: POKE 31542,28: PO
KE 31543,7: POKE 31541,1
 140 FOR A=0 TO 32: LET L=USR 31
384: LET L=USR 31432: IF INKEY$<
>"" THEN GO TO 170
 150 NEXT A
 160 IF INKEY$="" THEN GO TO 130
 170 LET JOGO=31512: LET CO=3154
6: LET CO2=31548: LET MO=31544:
LET CL=31492
 180 LET PO=0: LET PO2=0: LET TE
MP=2000: GO SUB 470
```

```
190 PRINT AT 10,15; INK 0; PAPE
R 4;"[illegible]";AT 3,6;"[illegible]";AT 17,24;
"[illegible]"
200 PRINT AT 21,0; INK 3;"[illegible]"
210 POKE 31545,0: POKE 31541,0:
POKE CO,0: POKE CO+1,1: POKE CO
2,30: POKE CO2+1,18
220 PRINT AT 1,0; INK 1;"  ";AT
2,0;"  "; INK 2;AT 18,30;"  ";A
T 19,30;"  "
230 LET L=USR JOGO
240 LET D=PEEK 31544
250 IF D=1 THEN GO TO 310
260 IF D=2 THEN GO TO 350
270 PRINT AT 0,7; INK 1;PO; INK
2;AT 0,25;PO2
280 PRINT AT 20,15;TEMP;"  "
290 LET TEMP=TEMP-1: IF TEMP=-1
THEN GO TO 390
300 GO TO 230
310 FOR G=0 TO 10
320 BEEP .04,RND*-30: PRINT AT
PEEK (CO+1),PEEK CO; INK 3; OVER
1;"XX";AT PEEK (CO+1)+1,PEEK CO
;"XX"
330 POKE MO,0: NEXT G
340 LET L=USR CL: LET PO2=PO2+5
: GO TO 210
350 FOR G=0 TO 10
360 BEEP .04,RND*-30: PRINT AT
PEEK (CO2+1),PEEK CO2; INK 3; OV
ER 1;"XX";AT PEEK (CO2+1)+1,PEEK
CO2;"XX"
370 POKE MO,0: NEXT G
380 LET L=USR CL: LET PO=PO+5:
GO TO 210
390 BEEP .01,40: FOR A=0 TO 30:
BEEP .01,-20: BEEP .001,50: NEX
T A
400 IF PO>PO2 THEN GO TO 430
410 IF PO=PO2 THEN GO TO 440
420 PRINT AT 20,7; INK 2;"VENCE
DOR JOGADOR 2": GO TO 460
430 PRINT AT 20,7; INK 1;"VENCE
DOR JOGADOR 1": GO TO 460
440 PRINT AT 20,0;"TEMPO SUPLEM
ENTAR PARA DESEMPATE": FOR A=0 T
O 200: NEXT A
450 LET TEMP=500: FOR A=0 TO 31
: PRINT AT 20,A;" ": BEEP .01,-1
0: NEXT A: GO TO 230
460 FOR A=0 TO 500: NEXT A: GO
TO 60
```

```
470 FOR A=0 TO 21: PRINT AT A,0
; INK 7;"____________________________
____________": NEXT A
480 RETURN
490 FOR A=USR "A" TO USR "C"+7:
READ C: POKE A,C: NEXT A
500 DATA 255,129,129,129,65,65,
129,255,48,241,51,233,157,255,17
3,253,0,0,0,0,255,255,0,0
510 RETURN
```

LINGUAGEM MAQUINA

```
31000  780F0F0F47E6E0A95F78  1090
31010  E803EE5867C9CD2D7904  1238
31020  13AF322B7BF578E518F6  1275
31030  4067F1846778E6070F0F  1030
31040  0F616F1A7723131A773A  657
31050  2B7B3CFE08C8322B7B13  923
31060  C33179ED4B2C7BCD1879  1194
31070  3A2E7B577EBA2016237E  841
31080  BA20117A112000195 77E  644
31090  BA20072B7EBA2002AFC9  990
31100  3EFFC9322E7BCD18793A  1145
31110  2E7B77233771120001977  635
31120  2B77C93A2F7BFE00281F  916
31130  01FEDFED78CB4FCCD879  1658
31140  01FEBFED78CB4FCCE579  1640
31150  01FEBFED78CB47CC0A7A  1413
31160  C901FEFBED78CB47CCD5  1758
31170  7901FEFDED78CB47CCE5  1594
31180  7901FEFDED78CB4FCCF4  1716
31190  79C9ED4B2C7B3E01B6C8  1248
31200  05ED432C7BC9ED4B2C7B  1156
31210  3E12B6C804ED432C7BC9  1140
31220  3A397BFE01C83C323978  903
31230  ED4B2C7B0C0C0CED4333  870
31240  7BC93A357BFE01C83C32  1123
31250  357BED4B2C7B0D0DED43  985
31260  357BC9ED4B3A7BED432C  1219
31270  7B3E3932 2E7BCD5779FE  1128
31280  FFCAF87AAF322F7BCD93  1574
31290  79ED4B3A7B3E3FCD7F79  1192
31300  ED4B2C7BED433A7B3E39  1063
31310  CD7F79113E7BED4B3A7B  1148
31320  C32879ED4B3C7BED432C  1189
31330  7B3E3A322E7BCD5779FE  1129
31340  FFCAFE7A3E01322F7BCD  1321
31350  9379ED4B3C7B3E3FCD7F  1220
```

```
31360  79ED4B2C78ED433C7B3E  1149
31370  3ACD7F79115E7BED4B3C  1117
31380  7BC328793A3978FE00C8  1171
31390  ED4B337BCD187936 3F3A  1011
31400  3378FE1E28113C323378  799
31410  237EFE3A280CFE3F2003  677
31420  3639C9AF323978C93639  1029
31430  18F73A3578FE00C8ED4B  1271
31440  3678CD187936 3F3A367B  879
31450  FE01281130323678 2B7E  769
31460  FE39280CFE3F20033363A  827
31470  C9AF32357BC9363A18F7  1186
31480  3E0132387BC93E023238  663
31490  7BC9212058018002 7EFE  988
31500  202802363F230878B120  566
31510  F3C9CD1F7ACD5B7ACD7E  1551
31520  7BCD847BCD847BCD7E7B  1497
31530  C9071E123A012307C21E  581
31540  .1200010F000000011E12  83
31550  000003C006200E100A13  292
31560  0E2307C20AA23FFCCC33  992
31570  FFFF63FE19F804E003C0  1659
31580  00000000003C004600870  416
31590  C850C47043E04 5603FFC  1343
31600  CC33FFFF63FE19F804E0  1619
31610  03C00000CD987AC3987A  1143
31620  CDC87AC3C87ACDC87AC3  1768
31630  C87A00000000000000000  322
```

ABISMO

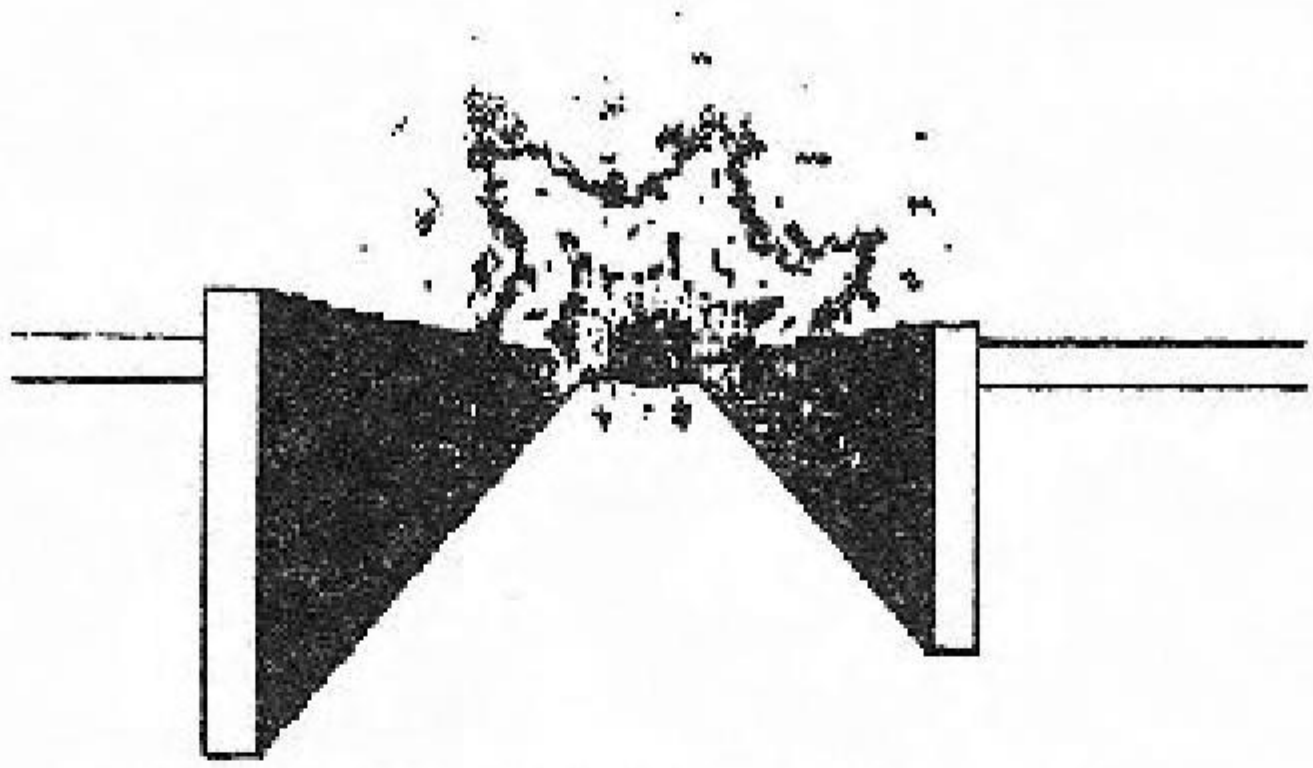
Um jogo em que se precisa ter, nervos de aço para se sair vencedor, neste incrível desafio. O objectivo é encurralar o adversário, entre as linhas, que os cursores dos dois jogadores vão criando. Uma das particularidades destes cursores, é que nunca ficam imóveis, estando sempre em movimento para um dos quatro sentidos.

Como o jogo é bastante rápido é necessário, possuir bons reflexos e pontaria, para escapar das armadilhas, criadas pelo adversário. Quando algum dos jogadores, chocar com a linha do adversário, ou com a sua própria linha, o adversário ganhará um ponto. O desafio é discutido em quatro níveis, até se decidir o vencedor.

BOA SORTE

Gravação

SAVE "ABISMO" LINE 0
SAVE "L/M" CODE 31000,220



ABISMO

MT

A B I S M O

```
  10 REM
  20 REM  © (MT) MARCO & TITO
  30 REM
  40 REM
  50 LOAD ""CODE
  60 PAPER 0: BORDER 0: INK 7: O
VER 1: GO TO 640
  70 OVER 0
  80 CLS
  90 LET p1=0: LET p2=0
 100 FOR x=1 TO 7
 110 CLS : GO SUB 150: GO SUB 38
0: GO SUB 430
 120 CLS : GO SUB 190: GO SUB 38
0: GO SUB 430
 130 CLS : GO SUB 260: GO SUB 38
0: GO SUB 430
 140 NEXT x
 150 PLOT 0,0: DRAW 255,0: DRAW
0,175: DRAW -255,0: DRAW 0,-175
 160 PLOT 1,1: DRAW 253,0: DRAW
0,173: DRAW -253,0: DRAW 0,-173
 170 FOR f=20 TO 155 STEP 15: PL
OT 40,f: DRAW 175,0: PLOT 40,f+1
: DRAW 175,0: NEXT f
```

```
 180 RETURN
 190 PLOT 0,0: DRAW 255,0: DRAW
0,175: DRAW -255,0: DRAW 0,-175
 200 PLOT 1,1: DRAW 253,0: DRAW
0,173: DRAW -253,0: DRAW 0,-173
 210 CIRCLE 129,88,50: CIRCLE 12
9,88,49
 220 PLOT 20,20: DRAW 0,135: PLO
T 21,20: DRAW 0,135
 230 PLOT 235,20: DRAW 0,135: PL
OT 236,20: DRAW 0,135
 240 PRINT AT 10,10;"
  ";AT 11,10;"                  "
 250 RETURN
 260 PLOT 0,0: DRAW 255,0: PLOT
0,1: DRAW 255,0: PLOT 0,175: DRA
W 255,0: PLOT 0,174: DRAW 255,0
 270 FOR f=5 TO 16 STEP 2: PRINT
 AT f,7; INK f/2;"
    ": NEXT f
 280 PRINT AT 2,5; PAPER 7; INK
0;"
 290 PRINT AT 16,5; PAPER 7; INK
 0;"
 300 RETURN
 310 FOR f=0 TO 8: FOR g=0 TO 7
STEP 3: PLOT PAPER f; INK g;PEEK
 30000,PEEK 30001: OUT 254,0: OU
T 254,16: NEXT g: NEXT f
 320 LET p2=p2+1: IF p2=10 THEN
GO TO 550
 330 GO SUB 470
 340 RETURN
 350 FOR f=0 TO 8: FOR g=0 TO 7
STEP 3: PLOT PAPER f; INK g;PEEK
 30003,PEEK 30004: OUT 254,0: OU
T 254,16: NEXT g: NEXT f
 360 LET p1=p1+1: IF p1=10 THEN
GO TO 850
 370 GO SUB 470: RETURN
 380 POKE 30000,10: POKE 30001,1
0: POKE 30002,4: POKE 30003,245:
 POKE 30004,165: POKE 30005,4
 390 PRINT AT 1,8;"C - PARA COME
CAR": PAUSE 0: PRINT AT 1,8;"
                    "
 400 FOR f=7 TO 0 STEP -.2: PLOT
 PAPER f; INK 9;10,10: PLOT PAPE
R f; INK 9;245,165
 410 OUT 254,16: OUT 254,0: NEXT
 f
 420 RETURN
 430 LET l=USR 31000
 440 IF PEEK 30002=255 THEN GO S
UB 310
```

```
 450 IF PEEK 30005=255 THEN GO S
UB 350
 460 RETURN
 470 CLS
 480 PRINT AT 0,5; FLASH (p1>=p2
);"JOGADOR 1";AT 0,18; FLASH (p2
>=p1);"JOGADOR 2"
 490 FOR f=0 TO 175 STEP 16: PLO
T 0,f: DRAW 255,0: NEXT f
 500 PLOT 128,0: DRAW 0,175
 510 FOR f=0 TO p1*16: PLOT 58,f
: DRAW INK 1;20,0: NEXT f
 520 FOR f=0 TO p2*16: PLOT 186,
f: DRAW INK 2;20,0: NEXT f
 530 IF INKEY$="" THEN GO TO 530
 540 RETURN
 550 CLS
 560 IF p1=p2 THEN PRINT AT 10,5
;"O JOGO ACABOU EMPATADO"
 570 IF p1>p2 THEN PRINT AT 10,4
;"O VENCEDOR E' O JOGADOR 1"
 580 IF p2>p1 THEN PRINT AT 10,4
;"O VENCEDOR E' O JOGADOR 2"
 590 BEEP .1,0: BEEP .1,4: BEEP
.1,7
 600 BEEP .1,0: BEEP .1,5: BEEP
.1,9
 610 BEEP .1,2: BEEP .1,4: BEEP
.1,7
 620 IF INKEY$<>"" THEN GO TO 60
 630 GO TO 590
 640 CLS : PRINT AT 10,11;"< ABI
SMO >"
 650 PRINT AT 2,0; INK 1;"    JOG
ADOR 1"; INK 2;"          JOGADOR 2
    "
 660 PRINT AT 3,0;"     A-ESQUERD
A        J-ESQUERDA"
 670 PRINT AT 4,0;"     S-BAIXO
         K-BAIXO"
 680 PRINT AT 5,0;"     D-CIMA
         L-CIMA "
 690 PRINT AT 6,0;"     F-DIREITA
         ENTER-DIREITA"
 700 PLOT 0,0: DRAW 255,0: DRAW
0,175: DRAW -255,0: DRAW 0,-175
 710 LET d=0: LET f=90
 720 PLOT 0,0: DRAW f,100: PLOT
255,0: DRAW -f,100
 730 IF d=1 THEN LET f=f+3
 740 IF INKEY$<>"" THEN GO TO 70
 750 IF f<2 OR f>175 THEN LET d=
NOT d
 760 IF d=0 THEN LET f=f-3
 770 GO TO 720
```

LINGUAGEM MAQUINA

```
31000  01FEFDED5B3275CD7079  1441
31010  323275ED4B3075CD9079  1164
31020  ED433075CDA479207DCD  1321
31030  BE7901FE6FED5B3875CD  1460
31040  7079323575ED4B337607  940
31050  070707CD9079ED433375  963
31060  CDA479205FCDBE7901F4  1378
31070  01ED5FCBDFD3FECB9FD3  1797
31080  FE0B78B120F118A6ED78  1384
31090  CB47280ECB4F2813CB67  959
31100  280CCB5F28057BC93E40  845
31110  C93E04C93E10C93E01C9  1011
31120  CB47200ECB572008CB67  956
31130  20020DC904C90CC905C9  872
31140  C5CDCF7947047E0710FD  1207
31150  E601C1C93EFF38327 5C9  1360
31160  3EFF323575C9CDCF7947  1342
31170  043EFE0F10FD477EA0A0  1129
31180  2F77C93EAF9047A71F37  1072
31190  1FA71FA8E6F8A8677907  1274
31200  0707A8E6C7A8070 76F79  1025
31210  E607C900000000000000  438
```

KARTING 2000

Resolveu participar numa corrida de karting, competindo com um amigo seu.

Trata-se de uma corrida onde a velocidade domina, por esse motivo, as pistas são separadas, para que nenhum dos participantes perturbe a corrida do adversário. Mas para não tirar a emoção da corrida, os pilotos partem ao mesmo tempo.

Poderá combinar com o seu adversário, o número de voltas pretendido, dependendo deste, a dureza da prova.

Espere pela luz verde e,

BOA SORTE

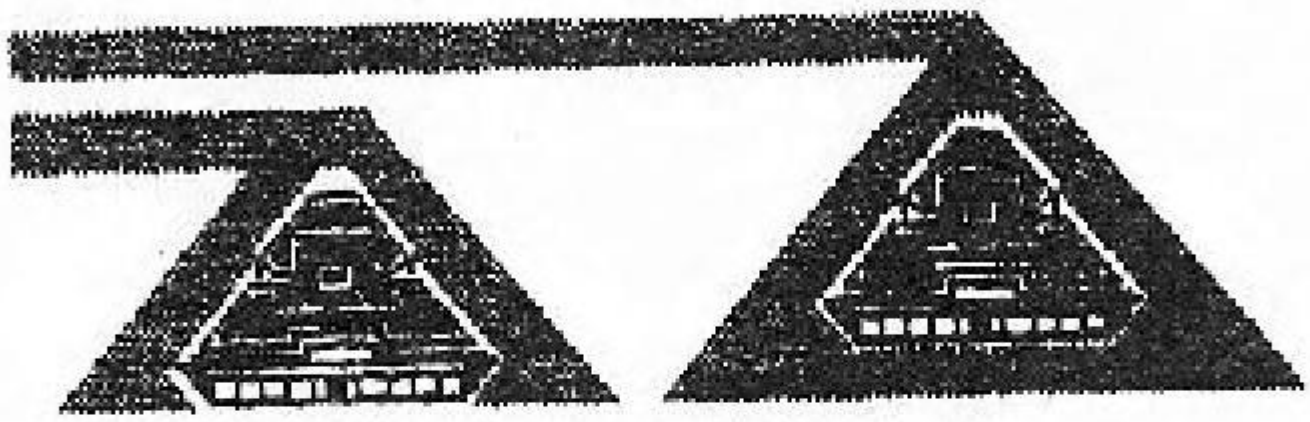
Gravação

SAVE "KARTING" LINE 0
SAVE "Kart.Code" CODE 30000,350

GRAFICOS

A- B- C- D- E- F- G- H-
I- J- K- L- M-

KARTING MT 2000



KARTING 2000

```
 10 REM
 20 REM   © (MT) MARCO & TITO
 30 REM
 40 REM
 50 LOAD ""CODE : GO SUB 810
 60 BORDER 1: PAPER 7
 70 INK 0: CLS
 80 GO SUB 360: GO SUB 270
 90 GO SUB 500: GO SUB 960
100 PRINT AT 21,0; FLASH 1; INK
1; PAPER 7;"              PRIMA UMA T
ECLA              ": PAUSE 0
110 FOR A=0 TO 31: PRINT AT 21,
A;" "
120 NEXT A
130 PRINT AT 10,22; PAPER 0; BR
IGHT 1; INK 2;"●": BEEP .2,30
140 FOR A=0 TO 200: NEXT A
150 PRINT AT 11,22; INK 6; PAPE
R 0;"●";AT 10,22;" ": BEEP .2,20
160 FOR A=0 TO 200: NEXT A
170 PRINT AT 12,22; PAPER 0; IN
K 4;"●";AT 11,22;" ": BEEP .2,10
180 IF ATTR (17,18)=68 THEN LET
V=V+1: PRINT AT PEEK 29505,PEEK
29504; INK 4; PAPER 0; OVER 1;"
"
```

```
 190 IF ATTR (15,14)=69 THEN LET
 V2=V2+1: PRINT AT PEEK 29507,PE
EK 29506; INK 5; PAPER 0; OVER 1
;" "
 200 LET L=USR 30000
 210 LET L=USR 30156: OUT 254,1
 220 PRINT AT 15,6; INK 1; PAPER
 7;"#";AT 7,25;"#"
 230 PRINT PAPER 0; INK 4;AT 11,
19;INT (V/2); INK 5;AT 10,19;INT
 (V2/2)
 240 IF VX=INT (V/2) THEN GO TO
700
 250 IF VX=INT (V2/2) THEN GO TO
 770
 260 GO TO 180
 270 POKE 29512,0: POKE 29513,0
 280 POKE 29512,0: POKE 29513,0
 290 POKE 29504,16: POKE 29505,1
7
 300 POKE 29506,14: POKE 29507,1
6
 310 LET H=121: LET L=29
 320 POKE 29508,L: POKE 29509,H
 330 POKE 29510,L: POKE 29511,H
 340 LET V2=0: LET V=0
 350 RETURN
 360 GO SUB 270: GO SUB 500
 370 GO SUB 960
 380 POKE 29512,1: POKE 29513,1
 390 LET L=USR 30156
 400 PRINT AT 21,8;"PRIMA UMA TE
CLA"
 410 PRINT AT 15,6; INK 1; PAPER
 7;"#";AT 7,25;"#"
 420 IF INKEY$="" THEN GO TO 380
 430 PAPER 0: CLS
 440 PRINT AT 10,5; INK 4;"JOGAD
OR1";AT 12,7;"L/ENTER";AT 10,20;
 INK 5;"JOGADOR2";AT 12,20;"D/F"
 450 GO SUB 960: INPUT "NUMERO D
E VOLTAS (1-99)?";VX
 460 IF VX>99 OR VX<1 THEN GO TO
 450
 470 FOR A=0 TO 100: NEXT A
 480 PAPER 7: CLS
 490 RETURN
 500 FOR A=9 TO 13: PRINT INK 1;
AT A,8;"[redacted]"
 510 NEXT A
 520 PRINT INK 5;AT 14,14;"[redacted]";AT
 16,14;"[redacted]"
 530 PRINT INK 4;AT 16,16;"[redacted]";AT
 18,16;"[redacted]"
```

```
540 FOR A=6 TO 14 STEP 2: PRINT
INK 0; PAPER 0;AT A,4;"©";AT A,
25;"©";AT A+2,6;"©";AT A+2,27;"©
"
550 NEXT A
560 FOR A=7 TO 13 STEP 2: PRINT
INK 0; PAPER 0;AT A,4;"©";AT A,
25;"©";AT A+2,6;"©";AT A+2,27;"©
"
570 NEXT A
580 LET A$="»»»»»»»»»»»»»»»»»»»»»»»»
»»»»"
590 PRINT AT 17,6; INK 0; PAPER
0;A$
600 PRINT AT 15,4; INK 0; PAPER
0;A$
610 LET A$="«««««««««««««««««««««
««««"
620 PRINT AT 7,6; INK 0; PAPER
0;A$
630 PRINT AT 5,4; INK 0; PAPER
0;A$
640 PRINT PAPER 0;AT 15,14; INK
5; OVER 1;" ";AT 17,16; INK 4;"
"
650 FOR A=10 TO 12: PRINT AT A,
22; INK 0; PAPER 0;" "
660 NEXT A
670 PRINT AT 15,6; INK 1; PAPER
7;"#";AT 7,25;"#"
680 PRINT PAPER 0; BRIGHT 1;AT
11,9; INK 4;"Voltas:J1:";AT 10,9
; INK 5;"Voltas:J2:"
690 RETURN
700 FOR A=0 TO 20: BEEP .01,A+R
ND*3
710 NEXT A
720 CLS : PRINT AT 10,7; INK 1;
"VENCEDOR JOGADOR 1"
730 PRINT AT 12,9;"■ ■ ■ ■ ■ ■
■"
740 PRINT AT 13,9;" ■ ■  ■  ■
■ "
750 FOR A=0 TO 300: NEXT A
760 GO TO 60.
770 FOR A=20 TO 0 STEP -1: BEEP
.01,A+RND*2
780 NEXT A
790 CLS : PRINT AT 10,7; INK 1;
"VENCEDOR JOGADOR 2"
800 GO TO 730
810 RESTORE 840: FOR A=USR "A"
TO USR "M"+7
820 READ S: POKE A,S
830 NEXT A
```

```
 840 DATA 24,60,126,102,36,60,10
2,90,0,204,126,167,167,126,204,0
 850 DATA 90,102,60,36,102,126,6
0,24,0,51,126,229,229,126,51,0
 860 DATA 24,60,126,102,36,60,10
2,66,0,204,126,39,39,126,204,0
 870 DATA 66,102,60,36,102,126,6
0,24,0,51,126,228,228,126,51,0
 880 DATA 24,60,126,126,126,126,
60,24,63,103,27,37,34,33,33,17
 890 DATA 252,254,248,226,196,13
2,132,136,17,11,4,8,5,9,19,31
 900 DATA 136,208,224,224,224,24
0,246,248
 910 RESTORE 940: FOR A=31005 TO
 31067
 920 READ S: POKE A,S
 930 NEXT A
 940 DATA 4,4,4,4,4,4,4,4,4,4,4,
1,1,1,1,1,1,1,1,1,8,8,8,8,8,8,
8,8,8,8,8,8,8,8,8,8,8,8,8,8,2,
2,2,2,2,2,2,2,2,2,4,4,4,4,4,4,4,
4,4,4,0
 950 RETURN
 960 PRINT AT 3,6; INK 2;"K A R
T I N G    2000": RETURN
 970 NEXT a
```

LINGUAGEM MAQUINA

```
30000  01FEFDED78F6F3FEF328  1891
30010  38ED78CB57281E01FEFD  1281
30020  ED78CB5F281F3ACB75FE  1358
30030  03202D3A4973C3C3249   569
30040  73AF32CB75181F3ACB75  1093
30050  CBC732CB75181D83ACB75  1390
30060  CBCF32CB75181D7AF32CB  1447
30070  753E05D3FECD7B7601FE  1350
30080  BFED78F6FCFEFC2836ED  1853
30090  78CB47281C01FEBFED78  1265
30100  CB4F281D3ACA73FE03C0  1177
30110  3A4873C3C32473AF32   827
30120  CA75C93ACA75CBC732CA  1551
30130  7518DA3ACA75CBCF32CA  1398
30140  7518D9AF32CA753E04D3  1179
30150  FEC37B7600023A4873FE  1191
30160  0028263D32473ED4640  754
30170  73ED433C732A4473223E  915
```

```
30180  733E44326A76CD2A76ED  1121
30190  4B3C73ED4340732A3E73  952
30200  2244733A4973FE00C83D  978
30210  324973ED4B4273ED433C  1095
30220  732A4873323E733E4532  734
30230  6A76CD2A76ED4B3C73ED  1313
30240  4342732A3E73224673C9  867
30250  ED4B3C73CD6B76ED4B3C  1269
30260  7336002A3E737EFE012B  809
30270  1FFE02281EFE04281DFE  938
30280  08281C211C79232B3E73  504
30290  ED433C73CD6B763A6A76  1191
30300  77C9051BED041BEA0C18  884
30310  E70D18E445780F0F0F47  801
30320  E8E0A96F78E603EE5867  1516
30330  C9063257ED5FE6F6B2D3  1543
30340  FE10F7C900000000000000  718
```

CAPÍTULO 4

SUPER JOGOS

Vamos agora apresentar diversos jogos, para um único jogador. Este tipo de jogo pode ser bastante divertido se o leitor o considerar como um desafio às suas capacidades.

Alguns programas são um pouco extensos, mas depois de introduzidos, verá que compensou o trabalho dispendido.

É apresentada uma variada gama de jogos, para que o leitor possa encontrar o seu favorito.

Os programas que acompanham este Capítulo são:

Pesquisador
Turbo
Lagarta
Invasão Chinha
Monstros
Perigo!!!
Combate
Zig Zag

PESQUISADOR

As fontes de minerais da Terra esgotaram-se, só existe uma solução, e todos, procuram a sorte no vasto Universo.

Você é mais sortudo que os seus competidores, encontrou o que procurava. Terá somente que recolher os minerais do vasto Universo, evitando os obstáculos que neste se encontram.

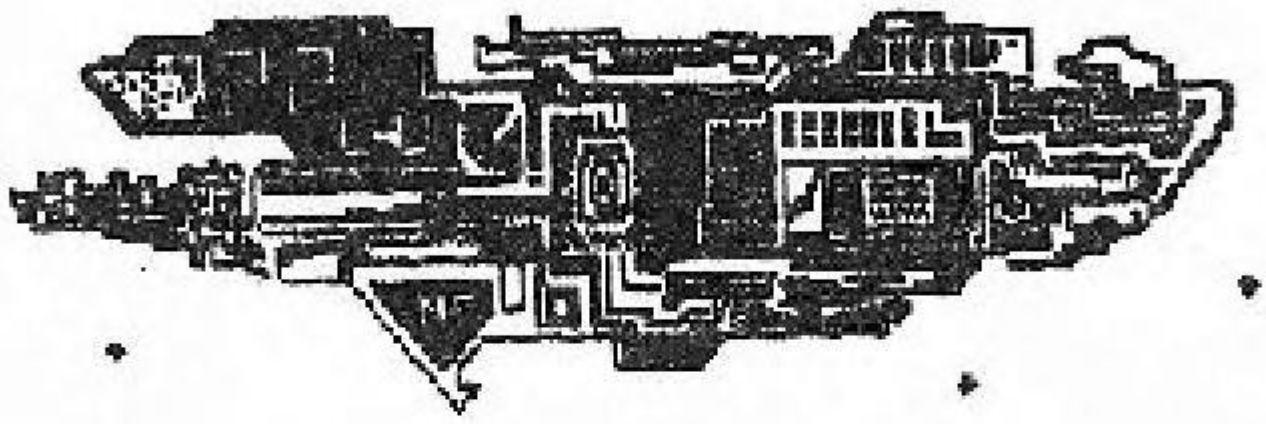
A sua nave trata-se do Pesquisador MT, equipado com potentes reactores. Possui assim tudo para ser bem sucedido no seu objectivo.

BOA SORTE

Gravaçao

SAVE "PESQUI." LINE 0
SAVE "Pesq.Code" CODE 31000,730

GRAFICOS



P E S Q U I S A D O R

```
   1 REM
   2 REM   © (MT) MARCO & TITO
   3 REM
   4 REM
   5 LOAD ""CODE : POKE 23675,66
: POKE 23676,123
  10 LET IN0=31510: LET CO=30000
: LET SOM=31260: LET ST=31490: L
ET NAVE=31116: LET LDIR=31352
  20 LET VIX=5: LET PO=0: LET V=
1: LET FASE=1500: LET T=0: LET C
X=4
  30 LET SX=40: LET SX2=60: GO S
UB 5000
  45 INK 0: LET L=USR IN0
  50 GO SUB FASE: POKE SOM+7,SX:
 POKE SOM+18,SX2
 150 INK 7
 155 POKE 30003,0: POKE CO,10: P
OKE CO+1,19: POKE 30002,1: LET L
=USR LDIR: LET L=USR ST
 160 LET D=INT (RND*24+1)
 161 IF D=4 OR D=13 OR D=22 THEN
 LET CX=3
```

```
 165 POKE 22531+D,CX: LET CX=CX+
1: IF CX=8 THEN LET CX=4
 170 LET L=USR NAVE: IF PEEK 300
02=0 THEN GO TO 1000
 181 IF T>400 THEN LET T=0: LET
FASE=FASE+100: GO TO 2400
 182 LET T=T+1: IF PEEK 30003=1
THEN LET PO=PO+10: POKE 30003,0
 190 GO TO 160
1010 LET CO2=PEEK CO: LET X=PEEK
 (SOM+7)
1015 POKE SOM+7,10
1020 FOR A=0 TO 30: PRINT AT PEE
K (CO+1),CO2; INK RND*7+1; OVER
1;"##";AT (PEEK (CO+1))+1,CO2;"#
#": LET L=USR SOM: NEXT A
1025 POKE SOM+7,X
1030 PRINT AT 19,CO2; INK 0;"  "
;AT 20,CO2;"  "
1040 LET L=USR LDIR: LET L=USR S
T: POKE 30002,1
1050 LET VIX=VIX-1: IF VIX=0 THE
N GO TO 1100
1060 GO TO 160
1110 FOR A=0 TO 20: LET L=USR SO
M: NEXT A
1120 PRINT AT 7,6; INK 5; BRIGHT
 1;"F I M  D E  J O G O"
1130 PRINT AT 10,5; INK 4;PO;" P
ONTOS";AT 11,5;V;" VOLTA";("S" A
ND V>1)
1135 FOR A=0 TO 500: NEXT A
1140 PAUSE 0: RUN 10
1300 RETURN
1510 LET X$="▼": GO SUB 9100
1515 LET A$=X$
1520 LET X$="▽": GO SUB 9120
1525 LET B$=X$
1530 GO TO 4000
1610 LET X$="▶": GO SUB 9100
1615 LET A$=X$
1620 LET X$="▷": GO SUB 9120
1625 LET B$=X$
1630 GO TO 4000
1710 LET X$="◀": GO SUB 9100
1715 LET A$=X$
1720 LET X$="◁": GO SUB 9120
1725 LET B$=X$
1730 GO TO 4000
1810 LET X$="▲": GO SUB 9100
1815 LET A$=X$
1820 LET X$="△": GO SUB 9120
1825 LET B$=X$
1830 GO TO 4000
1910 LET X$="◆": GO SUB 9100
```

```
1915 LET A$=X$
1920 LET X$="[illegible]": GO SUB 9120
1925 LET B$=X$
1930 LET X$="[illegible]": GO SUB 9100
1935 LET C$=X$
1940 LET X$="[illegible]": GO SUB 9120
1945 LET D$=X$
1950 GO TO 4100
2010 LET X$="[illegible]": GO SUB 9100
2015 LET A$=X$
2020 LET X$="[illegible]": GO SUB 9120
2025 LET B$=X$
2030 LET X$="[illegible]": GO SUB 9100
2035 LET C$=X$
2040 LET X$="[illegible]": GO SUB 9120
2045 LET D$=X$
2050 GO TO 4100
2110 LET X$="[illegible]": GO SUB 9100
2115 LET A$=X$
2120 LET X$="[illegible]": GO SUB 9120
2125 LET B$=X$
2130 LET X$="[illegible]": GO SUB 9100
2135 LET C$=X$
2140 LET X$="[illegible]": GO SUB 9120
2145 LET D$=X$
2150 GO TO 4100
2210 LET A$="[illegible]"
2220 LET B$="[illegible]"
2230 LET C$="[illegible]"
2240 LET D$="[illegible]"
2245 LET E$="[illegible]"
2250 LET F$="[illegible]"
2255 LET G$="[illegible]"
2260 LET H$="[illegible]"
2270 FOR A=0 TO 7 STEP 4
2260 PRINT AT 0+A,4;A$;A$;E$
2261 PRINT AT 1+A,4;B$;B$;F$
2262 PRINT AT 2+A,4;C$;C$;G$
2263 PRINT AT 3+A,4;D$;D$;H$
2264 NEXT A
2265 RETURN
2300 LET FASE=1500: LET U=U+1
2310 IF PEEK (CO+1)<3 THEN POKE
CO+1,20
2320 POKE CO+1,(PEEK (CO+1))-1
2330 GO TO 50
2400 FOR A=0 TO 30: LET L=USR SO
M: NEXT A
2420 INK 0: CLS
2430 PRINT AT 10,5; INK 5;"VOLTA
 N-";U;AT 11,5;PO;" PONTOS";AT 1
2,5;"PESQUISADORES:";UIX
2440 GO SUB 2500: IF SX>5 THEN L
ET SX=SX-5: IF SX2<100 THEN LET
SX2=SX2+5
```

```
2450 GO TO 45
2500 FOR A=0 TO 30: LET L=USR SO
M: LET L=USR SOM: LET L=USR SOM:
 NEXT A: RETURN
4010 FOR A=0 TO 7 STEP 2
4020 PRINT AT A,4;A$
4040 PRINT AT A+1,4;B$
4050 NEXT A
4060 RETURN
4110 PRINT AT 0,4;A$;AT 1,4;B$;A
T 2,4;C$;AT 3,4;D$
4120 PRINT AT 4,4;A$;AT 5,4;B$;A
T 6,4;C$;AT 7,4;D$
4130 RETURN
5010 BORDER 0: PAPER 0
5020 INK 7: CLS
5025 PRINT AT 10,5;"OBSTACULOS:"
;AT 11,5; INK 5;"● ♣ ▥ ▬ ♠ ♦ ♣",
5026 PRINT AT 13,5;"MINERAIS:";A
T 14,5; INK 3; BRIGHT 1;"● ●"
5030 PRINT AT 8,8; INK 4;"Por:";
" MARCO & TITO"
5035 PRINT AT 16,5;"TECLAS:";AT
17,5; INK 6;"◆ (1) (0) ◆"
5040 PRINT AT 6,5; INK 3;"[illegible]
[illegible]"
5050 PRINT AT 2,5; INK 3;"[illegible]
[illegible]"
5060 PRINT AT 4,8; INK RND*3+4;"
P E S Q U I S A D O R"
5070 POKE SOM+18,5+RND*5: LET L=
USR SOM: POKE SOM+18,60
5080 PRINT AT 21,5; INVERSE 1; I
NK RND*3+4;"     PRIMA UMA TECLA
"
5090 IF INKEY$="" THEN GO TO 506
0
5100 RETURN
9000 DIM Z$(24)
9010 FOR A=1 TO 24: LET Z$(A)=X$
: NEXT A
9020 LET Z$(4)=Y$: LET Z$(13)=Y$
: LET Z$(22)=Y$: RETURN
9100 LET Y$="●": GO SUB 9000: LE
T X$=Z$
9110 RETURN
9120 LET Y$="◎": GO SUB 9000: LE
T X$=Z$
9130 RETURN
```

LINGUAGEM MAQUINA

```
31000  780F0F0F47E6E0A98F78  1090
31010  E603EE5857C9CD2D7904  1238
31020  13AF325779F578E618F6  1317
31030  4067F1846778E6070F0F  1030
31040  0F816F1A7723131A773A  657
31050  57793CFE08C832577913  1007
31060  C33179073E1A32437932  748
31070  47793E7732447932487 9  855
31080  18BE3E7E32437932477 9  882
31090  3E1232447932487918AC  758
31100  F5CD1879F1772377112 0  1158
31110  0019772877C9CD1A7AED  1097
31120  463075AFC5CD7C79CD3A  1325
31130  7AC1C5CD18797EFE03CC  1449
31140  0A7A305C237EFE03CC0A  904
31150  7A305301FEEFED78CB47  1378
31160  282A01FEF7ED78CB4728  1255
31170  31C13E07188 4C111E27A  1073
31180  CD5879ED4B3075C511E2  1331
31190  7ACD6A79C1C511247BCD  1325
31200  5879185DFED4B307579FE  1308
31210  1A28CD0CED433075185D4  988
31220  ED4B307579FE0428C50D  1107
31230  ED433075185C4C1AF3232  1157
31240  75C936003A33753C3233  759
31250  75E5CD1A7AE137C93A48  1310
31260  5C1F1F1F06280EFE2520  568
31270  06EE10ED79265062D20F4  983
31280  EE10ED79 2E5410EC185F  1129
31290  3E00010400111818 21DF  388
31300  5715F578824779834FD5  1218
31310  112000041910FD412310  453
31320  FDD142C506004 8D5E511  1265
31330  2000A7ED52D1E5EDB8E1  1602
31340  D1C17810EAF143772810  1258
31350  FCC921004011004801 00  640
31360  08ED8021004811005001  624
31370  0006EDB0C911200006 18  701
31380  771910FCC921005 8E53A  1021
31390  0065FE0528133C3200 5B  610
31400  070707CD8F7AE1237DFE  1130
31410  0420E718073E0332005B  504
31420  18EA211F58E53A015BFE  1043
31430  0528123C32015B070707  288
31440  CD8F7AE1287DFE1820E7  1407
31450  C93E0332015B18EB3C3C  787
31460  244224BD24BDE7BD42BD  1227
31470  24421 83C3C3C3C7E3CFF  807
31480  3CFFFFFFF7EFF3C7E183C  1476
31490  11E27AED4B3075CD6A79  1274
31500  11247 9ED4B3078C35879  1067
```

```
31510  01FF022100581101E836  539
31520  00ED80C9018002400 7E0  1040
31530  06600990 13C815A81668  789
31540  1668166835AC4BD28C31  951
31550  9FF9AE75781E3C7EFFFF  1545
31560  C3DB423C3C7E8FAFAF8F  1362
31570  7E3C3C42DBC3FFFF7E3C  1422
31580  3C7EF1F5F5F17E3C3C3C  1464
31590  3C3CFF7E3C181030 7FFF  1031
31600  FF7F30101 83C7EFF3C3C  1031
31610  3C3C080CFEFFFFFE0C08  1178
31620  3C7EFFFFFFFF7E3C3C42  1518
31630  BDBDBDBD423CCCBDBDBD  1722
31640  D6D6D6C83828282828 28  1096
31650  2838000 00FFDB0000E7FF  1056
31660  001842A5A5FF81FF1615  1103
31670  3648 8C9FAE7868868ACD2  1311
31680  31F9751E3C242424E742  910
31690  241680C0E0F8B89C8EFF  1589
31700  FF8E9CB8F8E08080FF71  1833
31710  391D1F07030101030 71F  170
31720  1D3971FF000000000000  454
```

TURBO

Um jogo que exige do jogador, grande habilidade e rapidez, mas, não desespere, pois você é capaz. O objectivo é percorrer com o carro todo o labirinto. O carro vai deixando um rasto, que o orientará pelo labirinto, pois só assim, você saberá quais as zonas que já circulou. Acontece que existe um problema. o rasto vai-se deteriorando dificultando ainda mais a tarefa.

Outra coisa que terá de ter em conta, é a gasolina e qual o momento exacto para fazer o reabastecimento.

Quando terminar a tarefa, passará para uma fase de bónus. Nesta fase, o seu carro desloca-se ao dobro da velocidade e o rasto desaparece minimamente, assim como, a gasolina mantém-se constante. É só a velocidade que conta nesta fase, pois quanto mais rápido você for, mais será a sua pontuação.

Depois de terminar esta fase, passará para outro labirinto, e tudo se repetirá.

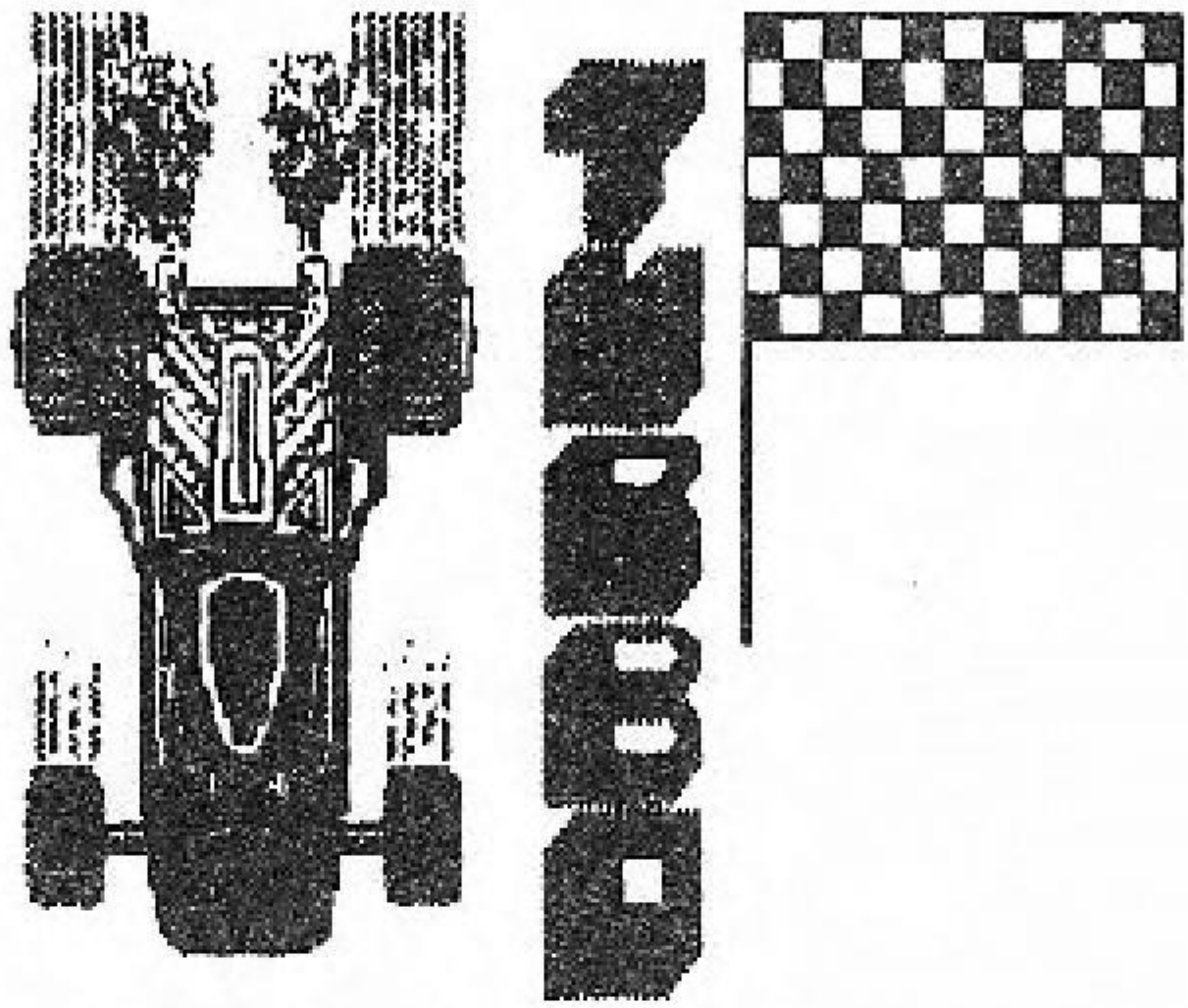
Arranque em primeira e.

BOA SORTE

Gravação

SAVE "TURBO" LINE 0
SAVE "Code" CODE 31000,490

GRAFICOS



TURBO

```
  10 REM
  20 REM   © (MT) MARCO & TITO
  30 REM
  40 REM
  50 LOAD ""CODE
  60 GO SUB 940
  70 PAPER 0: BORDER 0: INK 0: C
LS
  80 GO SUB 780: CLS
  90 LET FA=1
 100 FOR F=0 TO 14: PRINT AT F,0
,"[illegible]": NEXT F
 110 INK 7
 120 LET O=1: LET C=8: LET PO=0:
 LET C$=" [illegible] [illegible] [illegible]"
 130 LET L=1: LET Y=1: LET D=250
 140 POKE 31276,38: POKE 31285,1
8: POKE 32000,1: POKE 32001,1: P
OKE 32005,31: POKE 32006,0
 150 IF L THEN LET LAB=31370: PO
KE 32008,INT (LAB/256): POKE 320
07,(LAB/256-INT (LAB/256))*256
 160 IF NOT L THEN LET LAB=31430
: POKE 32008,INT (LAB/256): POKE
 32007,(LAB/256-INT (LAB/256))*2
56
```

```
170 LET W=USR 31324
180 IF Q=1 THEN POKE 32009,D
190 GO SUB 560
200 IF NOT Q THEN GO TO 290
210 LET Q=0
220 PRINT AT 20,0;"
                    "
230 LET W=USR 31000
240 IF PEEK 32005=255 THEN GO T
O 610
250 IF PEEK 32005=0 THEN GO TO
370
260 IF Y=1 THEN PRINT AT 7,15;
PAPER 0; INK 7; BRIGHT 1;"▶"
270 IF PEEK 32000=15 AND PEEK 3
2001=7 AND Y=1 THEN GO SUB 490
280 GO TO 230
290 LET Q=1: LET T=500
300 POKE 32009,255: POKE 31276,
27: POKE 31285,2
310 PRINT AT 20,0; PAPER 6; INK
9;"  ▶ ▶ ▶ ▶ FASE DE BONUS  ▶ ▶
▶ ▶ "
320 LET W=USR 31000
330 POKE 32005,31
340 IF PEEK 32005=255 THEN GO T
O 710
350 LET T=T-.2
360 GO TO 320
370 LET A$="[illegible]"
380 FOR F=7 TO 0 STEP -1
390 FOR G=1 TO LEN A$
400 PRINT AT PEEK 32001,PEEK 32
000; INK F;A$(G)
410 PRINT AT 16,23+C; INK G; PA
PER 6;A$(G)
420 BEEP .03,F: BEEP .03,G: NEX
T G
430 NEXT F
440 LET C$(C)=" ": LET C=C-2
450 IF C=0 THEN PAUSE 0: PRINT
AT 10,11; FLASH 1;"GAME OVER": P
AUSE 0: RUN 90
460 LET Y=1
470 GO SUB 560
480 GO TO 140
490 FOR F=PEEK 32005 TO 31
500 PRINT AT 18,F; INK 2;"[illegible]"
510 BEEP .01,F
520 NEXT F
530 POKE 32005,31
540 LET Y=0
550 RETURN
```

```
 560 PRINT AT 16,0; PAPER 4; INK
 9;"PONTOS ";PO;"    "; PRINT AT 1
6,17; PAPER 5; INK 9;"CARROS ";C
$;"
 570 PRINT AT 19,12;"FASE ";FA
 580 PRINT AT 17,11;"GASOLINA"
 590 PRINT AT 18,0; INK 2;"
                                  "
 600 RETURN
 610 LET FA=FA+1
 620 FOR F=0 TO 200
 630 LET W=USR 31308
 640 NEXT F
 650 FOR F=1 TO PEEK 32005
 660 LET PO=PO+10: GO SUB 560
 670 BEEP .02,F: NEXT F
 680 LET L=NOT L: LET D=D-3
 690 FOR F=0 TO 14: PRINT AT F,0
; INK 0;"                              "
                                  : NEXT F
 700 LET Y=1: GO TO 140
 710 IF T<0 THEN LET T=0
 720 PRINT AT 21,3; FLASH 1;"TEM
PO => ";500-INT T; FLASH 0;" ";
INVERSE 1;"BONUS ";INT T;" "
 730 LET PO=PO+INT T: GO SUB 560
 740 FOR F=0 TO 40: BEEP .03,RND
*10+30: NEXT F: PAUSE 0
 750 GO TO 690
 760 PRINT AT 4,0;
 770 PRINT "  ######  %    %  ######  %
#####  ######  "
 780 PRINT "    %    %    %  %    %  %
  %    %    %"
 790 PRINT "    %    %    %  ######  %
#####  %    %  "
 800 PRINT "    %    %    %  %    %  %
  %    %    %"
 810 PRINT "    %    ######  %    %  %
#####  ######"
 820 LET A=1: FOR F=0 TO 31: PRI
NT AT 0,F; INK A;"▶";AT 14,F;"▶"
: LET A=A+1: IF A=8 THEN LET A=1
 830 NEXT F
 840 LET A=1: FOR F=0 TO 14: PRI
NT AT F,0; INK A;"▶";AT F,31;"▶"
: LET A=A+1: IF A=8 THEN LET A=1
 850 NEXT F: PRINT AT 15,0; PAPE
R 5; INK 9;"TECLAS : "; PAPER 0;
 INK 7;"(Q)-SUBIR (A)-DESCER "
 860 PRINT AT 17,7; INK 9;"(S)-E
SQUERDA (D)-DIREITA"
 870 PRINT AT 20,0; INK 9; INVER
SE 1;"▶▶▶▶     PRIMA UMA TECLA
  ▶▶▶▶"
```

```
 880 IF INKEY$<>"" THEN GO TO 90
0
 890 LET L=USR 31306: GO TO 880
 900 FOR F=0 TO 50
 910 LET L=USR 31324: POKE 32000
,7: LET L=USR 31324: POKE 32000,
13: LET L=USR 31324: POKE 32000,
5
 920 NEXT F
 930 RETURN
 940 POKE 23675,88: POKE 23676,1
27: FOR F=USR "A" TO USR "F"+7:
READ A: POKE F,A: NEXT F
 950 DATA 189,255,165,165,60,189
,255,189,239,66,255,243,243,255,
66,239,247,66,255,207,207
 960 DATA 255,66,247,189,255,189
,60,165,165,255,189,153,51,102,2
04,153,51,102,204,32,56
 970 DATA 62,63,52,56,32,32
 980 RETURN
```

LINGUAGEM MAQUINA

```
31000  01FEFDED78CB47CCCD79  1659
31010  01FEFBED78CB47CCB979  1647
31020  01FEEFED78CB47CCA579  1615
31030  CB4FCC9179ED5B047D4A  1263
31040  0612CDE179362DEB1114  948
31050  00ED52EBED53047D2100  1036
31060  5801E0017EFE0028C23  761
31070  0678B120F53EFF32067D  1083
31080  C9CD2E7AED4B097D3A78  1198
31090  5CB93001C9ED4B785CED  1288
31100  5F98E80F47ED5F99E61F  1309
31110  4FCDE1797EFE97C83600  1415
31120  C9ED4B007DED43027D0D  1082
31130  CDE179C811607FCDF479  1561
31140  C9ED4B007DED43027D0C  1081
31150  CDE179C811587FCDF479  1559
31160  C9ED4B007DED43027D05  1074
31170  CDE179C811707FCDF479  1577
31180  C9ED4B007DED43027D04  1073
31190  CDE179C811587FCDF479  1553
```

```
31200  C9780F0F0F67E6E0A98F  1203
31210  7CE603EE58677EFE97C9  1518
31220  AF3248 7AF578E618F840  1351
31230  87F184677 8E6070F0F0F  981
31240  816F1A773A487A3CFE08  962
31250  13280632487AC3F879CD  1081
31260  E179ED43007D3607ED48  1148
31270  027DCDE179361 8C93A48  1090
31280  5C1F1F1F06022620 06EE  506
31290  10D3FE26222D20F4EE10  1128
31300  D3FE2EC810ECC9072100  1204
31310  5801E0017E3C77230B78  785
31320  B120F7C921005 8ED5B07  1113
31330  7D013C00C5060 81A0738  486
31340  0736002310F81 80538 97  594
31350  2310F1C10B137 8B120E5  1074
31360  C9000000000000000000  201
31370  FFFFFFFF800000018DF8  1569
31380  DFBDA0081005AFAAD5F5  1404
31390  A02A5405AFEB57F5A002  1195
31400  4005AFEB57F5A02A5405  1102
31410  AFAAD5F5A0081005 8DF8  1432
31420  DFBD800000001FFFFFFFF  1561
31430  FFFFFFFF800000018F7A  1462
31440  5EFD810A50818DEA57BD  1394
31450  840A5021 87EA57ED8008  1160
31460  100D877BDEE18D4002 5D  1178
31470  A157EA85AD500AB6AD57  1319
31480  EAB580000001FFFFFFFF  1564
```

LAGARTA

Neste jogo você controla uma lagarta comilona.

O objectivo é desviar-se das paredes, e alimentar a lagarta, com a fruta que vai aparecendo no écran.

Por cada fruta que a lagarta come o seu tamanho aumenta, tornando a tarefa mais dificil.

Quando a lagarta estiver crescida, passará para um nível mais dificil.

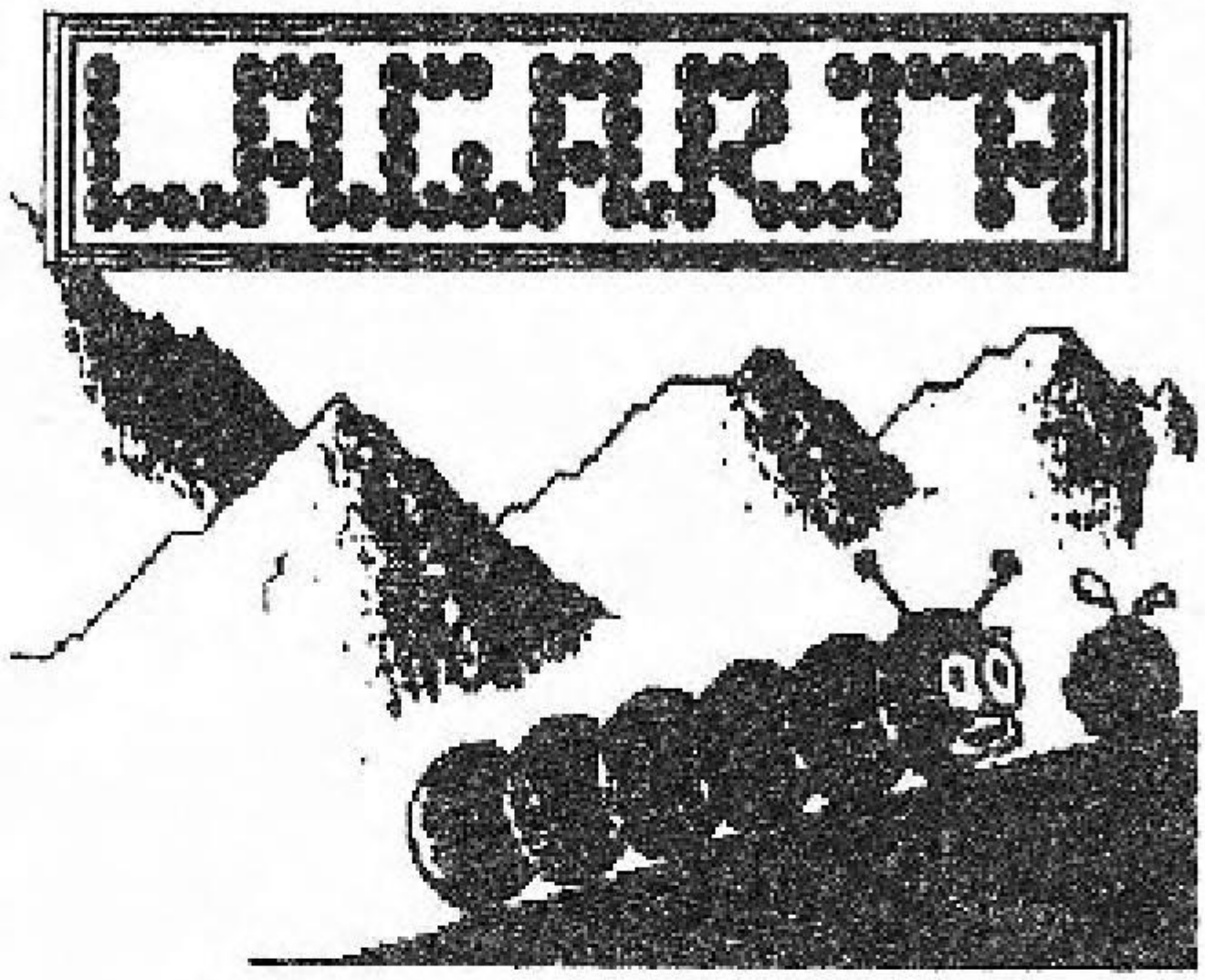
BOA SORTE

Gravação

SAVE "LAGARTA" LINE 0
SAVE "Code" CODE 31000,490

GRAFICOS

·-A ▫-B □-C ❑-D ●-E ◎-G ▒-H ▥-I



LAGARTA

```
  10 REM
  20 REM   © (MT) MARCO & TITO
  30 REM
  40 REM
  50 LOAD ""CODE : POKE 23658,8
  60 POKE 23658,8: POKE 23676,1:
 POKE 23675,123
  70 GO SUB 930
  80 GO TO 790
  90 PAPER 0: BORDER 0: INK 0: C
LS
 100 LET DI=10
 110 LET EFE=31437
 120 LET PAS=32006: LET COR=3200
0: LET POS=32002: LET COM=32003:
 LET MOR=32005
 130 LET PO=0: LET LA=3: LET FAS
=1
 140 FOR F=0 TO 21: PRINT AT F,0
; PAPER 0; INK 0;"[illegible]
[illegible]": NEXT F
 150 FOR F=32100 TO 32200: POKE
F,0: NEXT F
 160 POKE 32100,47: POKE 32101,6
9
```

```
 170 PRINT AT 9,15; BRIGHT 1; IN
K 4; OVER 1;" "
 180 IF FAS THEN GO SUB 290
 190 IF NOT FAS THEN GO SUB 420
 200 POKE PAS,0: POKE COR,15: PO
KE COR+1,9
 210 POKE COM,5: POKE MOR,0: POK
E POS,0
 220 RESTORE 920
 230 LET L=USR 31000
 240 IF PEEK 32005 THEN GO TO 54
0
 250 READ A: IF A=255 THEN LET A
=0: RESTORE 920
 260 IF PEEK PAS=DI THEN GO TO 5
10
 270 BEEP .04,A
 280 GO TO 230
 290 PAPER 1: INK 6: BRIGHT 1
 300 PRINT AT 0,0;"[redacted]
[redacted]";AT 21,0;"[redacted]
[redacted]";
 FOR F=0 TO 21: PRINT AT F,0;"[redacted]"
;AT F,31;"[redacted]": NEXT F
 310 PRINT AT 5,5;"[redacted]";AT
6,19;"[redacted]";AT 14,2;"[redacted]
[redacted]";AT 12,13;"[redacted]";AT 14,22;"[redacted]
[redacted]"
 320 PRINT AT 13,2;"[redacted]";AT 13,29;
"[redacted]"
 330 FOR F=1 TO 4
 340 PRINT AT F+2,12;"[redacted]";AT F+2,
19;"[redacted]";AT F+9,9;"[redacted]";AT F+9,22;"[redacted]
";AT F+6,5;"[redacted]";AT F+6,26;"[redacted]";AT
F+12,13;"[redacted]";AT F+12,18;"[redacted]"
 350 NEXT F
 360 PAPER 5: INK 2
 370 PRINT AT 2,2;".";AT 2,28;".
";AT 20,3;".";AT 18,10;".";AT 18
,22;".";AT 20,28;"."
 380 PRINT AT 8,1;".";AT 8,30;".
": PRINT AT 10,12;".";AT 10,19;"
."
 390 PAPER 0: INK 5: BRIGHT 0
 400 PRINT AT 19,12;"[redacted]"
 410 RETURN
 420 PAPER 2: INK 5: BRIGHT 1
 430 PRINT AT 0,0;"[redacted]
[redacted]";AT 21,0;"[redacted]
[redacted]";
 FOR F=0 TO 21: PRINT AT F,0;"[redacted]"
;AT F,31;"[redacted]": NEXT F
```

```
 440 PRINT AT 2,2;"▒▒▒▒▒▒▒▒▒▒▒▒▒▒";AT 2
,23;"▒▒▒▒▒▒▒▒▒▒▒▒▒▒";AT 6,6;"▒▒▒▒▒▒▒▒▒▒▒▒";AT
8,21;"▒▒▒▒▒▒▒▒▒▒▒▒";AT 17,3;"▒▒▒▒▒▒▒▒▒▒▒▒";A
T 17,22;"▒▒▒▒▒▒▒▒▒▒▒▒";AT 9,13;"▒▒▒▒▒▒▒▒▒▒▒▒";A
T 9,17;"▒▒▒▒";AT 15,13;"▒▒▒▒▒▒▒▒▒▒▒▒"
 450 FOR F=1 TO 5
 460 PRINT AT F+2,2;"▒▒▒";AT F+2,2
9;"▒▒▒";AT F+5,6;"▒▒▒";AT F+6,25;"▒▒▒"
;AT F+9,12;"▒▒▒";AT F+9,19;"▒▒▒"
 470 NEXT F
 480 PAPER 5: INK 2
 490 PRINT AT 14,3;".";AT 14,28;
".";AT 18,12;".";AT 18,19;".";AT
 8,12;".";AT 8,19;".";AT 6,14;".
";AT 6,17;".";AT 2,14;".";AT 2,1
7;"."
 500 PRINT AT 4,15;".."
 510 PAPER 0: INK 6: BRIGHT 0
 520 PRINT AT 13,3;"@"
 530 RETURN
 540 FOR F=0 TO 50: PRINT AT PEE
K (COR+1),PEEK COR; OVER 1;"@":
OUT 254,F: NEXT F
 550 FOR O=7 TO 0 STEP -1: LET H
=32100: FOR F=0 TO PEEK COM-2
 560 LET V=PEEK H+PEEK (H+1)*256
 570 POKE V,O: LET H=H+2: NEXT F
 580 NEXT O
 590 LET LA=LA-1: IF LA<0 THEN C
LS : PRINT AT 10,10; FLASH 1; IN
K 1; PAPER 7;"FIM DE JOGO";AT 15
,10; INK 2;"PONTOS >";PO: BEEP 5
,-10: GO TO 790
 600 GO TO 140
 610 FOR F=0 TO 15: LET L=USR EF
E: NEXT F
 620 FOR F=0 TO 21: PRINT AT F,0
,"
  ": NEXT F
 630 FOR F=0 TO 7: PRINT AT 10,F
;" PASSAGEM DE FASE": NEXT F
 640 PRINT AT 7,10; INVERSE 1;"L
AGARTAS"; INVERSE 0; INK 4;" ";L
A
 650 FOR F=0 TO 25
 660 PRINT AT 19,F; INK 4;"●●●●
●";AT 18,F+1;" 100": BEEP .04,RN
D*40: NEXT F
 670 PRINT AT 19,25;"      ";AT
18,27;"      "
 680 FOR F=0 TO 10: PRINT AT 2,F
; FLASH 0;" "; FLASH 1;"PONTOS":
 NEXT F
 690 FOR F=1 TO 100: PRINT AT 2,
18; INK 5;PO+F
```

```
 700 BEEP .01,F/2: NEX.
 710 LET PO=PO+100
 720 FOR F=10 TO 0 STEP -1: PRIN
T AT 2,F; FLASH 1;"PONTOS"; FLAS
H 0;" ";PO;" ": NEXT F
 730 PRINT AT 2,0;"                "
 740 FOR F=7 TO 0 STEP -1: PRINT
 AT 10,F;"PASSAGEM DE FASE ": NE
XT F
 750 PRINT AT 10,0;"
    "
 760 LET FAS=NOT FAS
 770 LET DI=DI+1: IF DI>40 THEN
LET DI=40
 780 GO TO 140
 790 PAPER 0: BORDER 0: INK 7: C
LS
 800 LET A=1: LET B=26
 810 LET A$=".□□.□□□.□□□.□□□.□□
□.□□□.□□□.□□□"
 820 PRINT AT 0,0; PAPER 1; BRIG
HT 1; FLASH 1; INK 7;A$;AT 21,0;
A$
 830 FOR F=1 TO 20: PRINT AT F,0
; BRIGHT 1; PAPER 1; INK 7; FLAS
H 1;"□";AT F,31;"□": NEXT F
 840 PRINT AT 6,7;"[redacted]
[redacted]";AT 10,7;"[redacted]
[redacted]";AT 8,7;"[redacted]";AT 9,7;"[redacted]";AT 7
,7;"[redacted]";AT 8,26;"[redacted]";AT 9,26;"[redacted]";A
T 7,26;"[redacted]"
 850 PRINT AT 7,8; PAPER 6; INK
4;"[redacted]";AT 8,8;"
[redacted]";AT 9,8;"[redacted]"
 860 PRINT AT 16,1; PAPER 2; INK
9;"Q-CIMA A-BAIXO P-DIRE. O-ESQ
U."
 870 PRINT AT 18,1; PAPER 6; INK
0;"[redacted] I- PARA INICIAR O JOGO [redacted]
[redacted]"
 880 PRINT AT 13,A; INK 4;"[redacted]
": LET A=A+1: IF A=27 THEN PRINT
 AT 13,A;"      ": LET A=1
 890 IF INKEY$="I" THEN GO TO 90
 900 PRINT AT 3,B; INK 4;"[redacted] "
: LET B=B-1: IF B=0 THEN PRINT A
T 3,B+1;"      ": LET B=26
 910 GO TO 860
 920 DATA 0,7,0,7,0,12,7,12,7,12
,4,7,12,4,7,12,255
 930 RESTORE 960: FOR F=USR "A"
TO USR "H"+7
 940 READ A: POKE F,A: NEXT F
 950 RETURN
```

```
 960 DATA 0,0,0,24,24,0,0,0,0,0,
60,36,36,60,0,0,0,126,66,66,66,6
6,126,0,255,129,129,129,129
 970 DATA 129,129,255,60,94,191,
191,191,191,94,60,60,66,153,173,
173,153,66,60,170,85,170,85,170,
85,170,85,170,170,170,170,170,17
0,170,170
```

LINGUAGEM MAQUINA

```
31000  01FEFDED78CB47CA6A7A  1569
31010  01FEFBED78CB47CA727A  1575
31020  01FEDFED78CB47CA827A  1563
31030  CB4FCA7A7A3A027DFE01  1168
31040  CA327AFE02CA527AFE03  1293
31050  CA427AFE04CA4A7AC9CD  1452
31060  5A7A7ECB77C28A7AFE06  1374
31070  CC927AED43007DE511C8  1347
31080  7D21647D015A00EDB011  904
31090  667D21C87D015A00EDB0  1089
31100  E122647D21647D3A037D  928
31110  477E6F237E573E441223  723
31120  10F5AF123A047DFEFF20  1182
31130  37AF32047DED4B007D11  863
31140  217BCDA67A3A785C473A  1048
31150  007D80E60F473A785C4F  918
31160  3A017D81E61F4FCD5A7A  1070
31170  7EFE0020E23508112978  879
31180  CDA67ACDDF7A010000C5  1241
31190  CD5A7A7ECB8FC4F579C1  1512
31200  0C79FE20CAEA79C3D579  1505
31210  0E00040478FE20C2D578  956
31220  C9FE63280DFE6A2814FE  1251
31230  71281BFE782822C9CD5A  1124
31240  7A366A11017BC3A67ACD  1111
31250  5A7A367111109 78C3A67A  1011
31260  CD5A7A36781111 17BC3A6  1109
31270  7ACD5A7A36631111 97BC3  1052
31280  A67AED4B007D0578FE00  1104
31290  C2537906 15C35379ED4B  1136
31300  007D04C35379ED4B007D  965
31310  0DC35379ED4B007D0CC3  1055
31320  53797 80F0F0F67E6E0A9  1095
31330  6F7CE603EE5867C93E03  1163
31340  32027DC33B793E013202  657
```

```
31350  7DC33B793E043202 7DC3  936
31360  3B793E02320 27DC33B79  796
31370  3EFF32067DC900003A03  759
31380  7D3C32037D3EFF32047D  859
31390  3A067D3C32067DC9AF32  858
31400  CC7AF578E616F64057F1  1599
31410  846778E6070F0F0F816F  877
31420  1A773ACC7A3CFE0613C8  1070
31430  32CC7AC3AA7A07210040  967
31440  01001B7E0777D3FE230B  791
31450  78B120F5C93A485C1F1F  1059
31460  1F061E0EFE25200 6EE10  564
31470  ED79261E2D20F4EE10ED  1236
31480  792EFF10ECC900000000  875
```

INVASÃO CHINHA

Os Chinhas são extraterrestres, provenientes de outro sistema solar.

O sistema solar dos Chinhas, encontra-se à beira da destruição, isto devido às grandes aglomerações de lixo, que os planetas possuem. É portanto por este motivo, que os Chinhas, como não suportam a atmosfera dos planetas do nosso sistema, enviaram um poderoso exército de autómatos, para dominar o nosso sistema, e fazer dele o depósito de detritos, do sistema dos Chinhas.

A sua missão será, como já deve calcular, de impedir que o exército enviado pelos Chinhas, venha a dominar o nosso planeta.

Os autómatos atacam primeiro com as forças menores que possuem. À medida que os vai destruindo, eles enviarão esquadrões, cada vez mais potentes.

Você encontra-se sobre um escudo terrestre, tentando evitar que o destruam, pois caso isso aconteça, terão o caminho livre, para dominar a Terra.

As armas que possui são: um escudo que somente pode ser activado de tempo a tempo, e um Laser que utilizará para neutralizar o inimigo.

Os autómatos estão programados, podendo por vezes estes atacar dois em dois, o que irá dificultar a sua tarefa, na medida que será mais difícil derrotá-los.

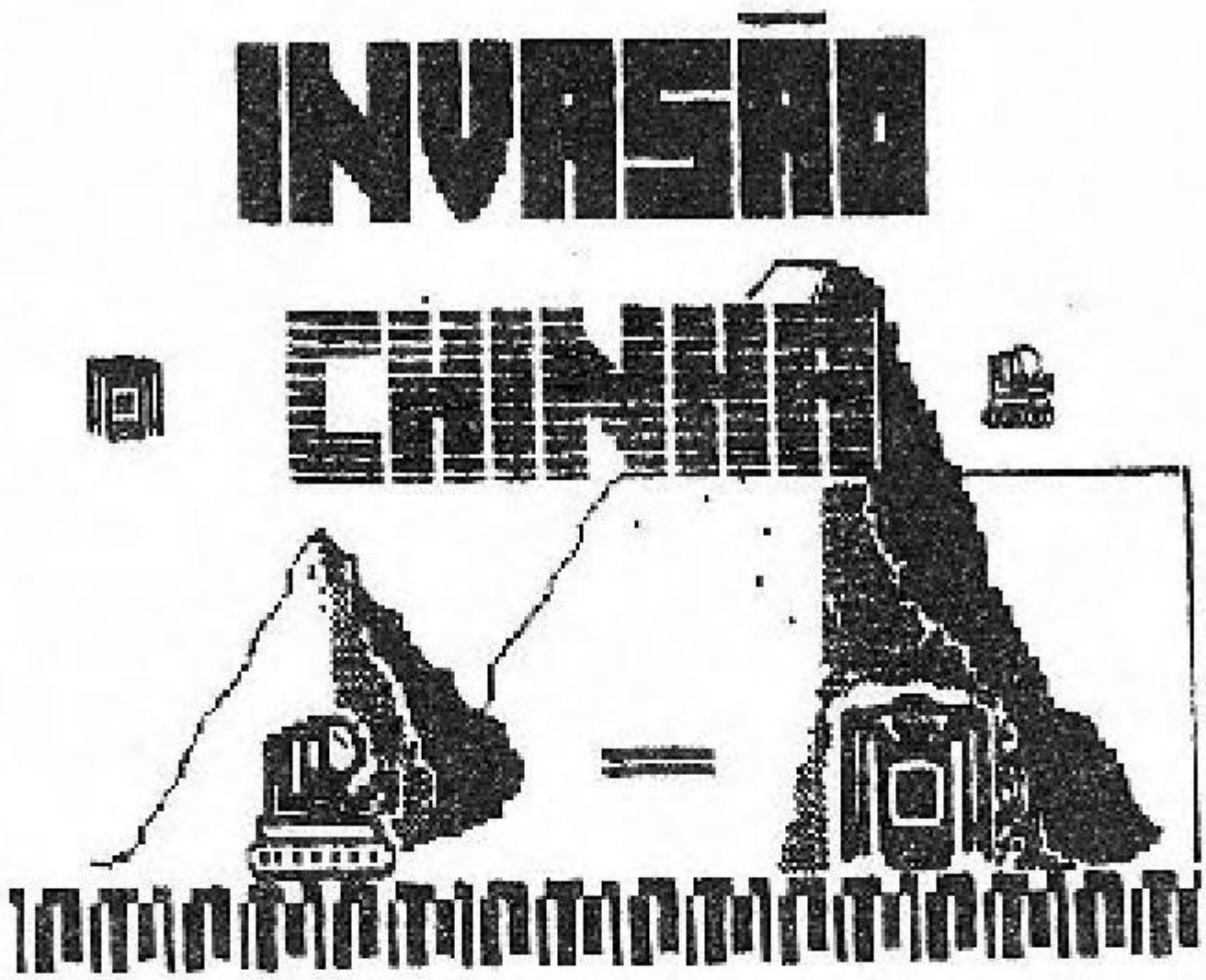
BOA SORTE

Gravação

SAVE "INVASÃO" LINE 0
SAVE "I-CODE" CODE 31000,980

GRAFICOS

A-III



INVASAO CHINHA

```
  10 REM
  20 REM   © (MT) MARCO & TITO
  30 REM
  40 REM
  50 LOAD ""CODE : POKE 23658,8
  60 GO SUB 820
  70 LET CO2=32116: LET CB=32102
  LET CO1=32114
  80 LET CO3=32118
  90 LET CO4=32120: LET SC=32122
 100 LET ACT2=32109: LET COR=321
11: LET ACT=32100
 110 LET VAR=32113
 120 LET VAR2=32099
 130 LET BONECO=31096: LET EF=31
941: LET FIREX=31277
 140 LET INI=31511
 150 POKE CB,10: POKE CB+1,19
 160 POKE COR,4: GO SUB 460
 170 GO SUB 600
 180 LET PO=0: LET VID=5
 190 LET T=50: LET FL=0
 200 POKE CB,10: POKE CB+1,19
 210 POKE CO1,0: POKE CO1+1,5
 220 POKE CO2,30: POKE CO2+1,6
```

```
 230 POKE CO3,8: POKE CO3+1,5
 240 POKE CO4,24: POKE CO4+1,5
 250 POKE ACT,0: POKE ACT2,0
 260 POKE SC,0: POKE VAR,1
 270 LET L=USR INI
 280 IF FL=1 THEN LET FL=0: PRIN
T INK 0;AT 20,0;"
                 "
 290 LET L=USR FIREX
 300 LET L=USR BONECO: IF PEEK V
AR=0 THEN GO TO 760
 310 LET L=USR FIREX
 320 IF INKEY$="N" AND T>60 THEN
 LET FL=1: LET T=0: LET L=USR EF
: PRINT AT 20,0; INK 2; BRIGHT 1
;"[redacted]
[redacted]"
 330 LET PO=PO+(PEEK SC)*10: POK
E SC,0
 340 PRINT AT 1,10; INVERSE 1;PO
 350 IF PO>=1000 THEN POKE COR,5
+2*8
 360 IF PO>=3000 THEN POKE COR,8
+8
 370 IF PO<1000 THEN POKE COR,68
 380 IF PO>=5000 THEN POKE COR,6
*5+1
 390 LET L=USR BONECO: IF PEEK V
AR=0 THEN GO TO 760
 400 LET L=USR FIREX
 410 LET T=T+1
 420 GO TO 270
 430 FOR A=5 TO 20: PRINT AT A,0
;"[redacted]
[redacted]"
 440 BEEP .01,-A*COS 0: PRINT IN
K 0;AT A,0;"
                 "
 450 NEXT A: RETURN
 460 PAPER 0: BORDER 0
 470 INK 0: CLS
 480 INK 7
 490 PRINT AT 5,8; INVERSE 1; BR
IGHT 1; INK 5;" I N V A S A O "
 500 PRINT AT 7,5; INK 4; BRIGHT
 1;"
```

CHINHA

```
                 "
 510 PRINT INK 6;AT 0,0; FLASH 1
;"    PRIMA    (I)   PARA   INICIAR
  "
 520 PRINT AT 12,9; INK 6;"(Z)-E
SQUERDA";AT 13,9;"(X)-DIREITA";A
T 14,9;"(M)-DISPARO";AT 15,9;"(N
)-ESCUDO"
```

```
530 PRINT AT 21,0; INK 5;"
"
540 OUT 254,0: LET L=USR FIREX
550 LET L=USR BONECO: LET L=USR
FIREX
560 IF INKEY$="N" THEN LET L=US
R EF
570 OUT 254,16
580 IF INKEY$<>"I" THEN GO TO 5
40
590 RETURN
600 INK 0: CLS
610 INK 5: PLOT 0,136
620 DRAW 25,39: DRAW 13,-20
630 DRAW 10,10: DRAW 14,-30
640 DRAW 10,15: DRAW 17,-10
650 DRAW 20,-5: DRAW 20,10
660 DRAW 7,5: DRAW 10,15
670 DRAW 13,-17: DRAW 5,-10
680 DRAW 4,10: DRAW 20,-15
690 DRAW 16,10: DRAW 10,10
700 DRAW 7,10: DRAW 7,-10
710 DRAW 10,-10: DRAW 16,-10
720 FOR A=0 TO 4: PRINT AT A,0;
OVER 1; INK 5; INVERSE 1;"  "
730 NEXT A: PRINT AT 21,0; INK
5;"
"
740 PRINT AT 0,9; PAPER 5; INK
0;"PONTOS";AT 0,20;"NAVES";AT 1,
22;"5"
750 RETURN
760 LET VID=VID-1: IF VID=-1 TH
EN GO TO 790
770 PRINT AT 1,22; INVERSE 1;VI
D;" "
780 GO SUB 430: GO TO 190
790 GO SUB 430: PRINT AT 10,5;
INVERSE 1;" F I M  D E  J O G O
"
800 FOR A=0 TO 200: NEXT A
810 GO TO 70
820 FOR A=USR "A" TO USR "A"+7:
READ B
830 POKE A,B: NEXT A
840 DATA 13,109,109,109,109,109
,109,97
850 RETURN
```

LINGUAGEM MAQUINA

```
31000  780F0F0F67E6E0A96F7C  1126
31010  E803EE5867C9CD2D7904  1238
31020  13AF324879CDEE7A1A77  1147
31030  23131A773A46793CFE08  772
31040  C83246791 3C3317907F5  1079
31050  CD1879F177237711 2000  913
31060  19772877C9F6CD1879F1  1343
31070  56BA28142356BA280F11  711
31080  20001956BA28072B56BA  691
31090  2802B7C937C9ED4B667D  1221
31100  3A6F7DCD5979DABE7AC5  1436
31110  01FEFEED78CB4FCCC879  1573
31120  01FEFEED78CB57CCDB79  1700
31130  01FE7FED78CB57CCEE79  1592
31140  C1AFCD4979ED4B667D3A  1364
31150  6E7DCB57200F117B7CCD  1041
31160  2879ED4B667D3E07C349  1037
31170  79115B7C18EFED4B667D  1155
31180  79FE01C80DED43667D3E  1182
31190  0432 6E7DC9ED4B667D79  1150
31200  FE1DC80CED43667D3E08  1096
31210  328E7DC93A6E7DCB5720  1101
31220  05CB6F201AC93A6C7DCB  1056
31230  47C0ED4B667D79FE01C8  1378
31240  0D3E01326C7DED43667D  892
31250  C93A6D7DCB47C0ED4B66  1373
31260  7D79FE1DC80C0C3E0132  866
31270  6D7DED436A7DC93A6C7D  1261
31280  CB472830ED4B667DC50D  1113
31290  79FE0028 5CCD18797EFE  1237
31300  0020 69ED43667DC1CD18  1092
31310  793600ED4B667D119B7C  1012
31320  CD0178ED4B667DCD1879  1220
31330  36423A6D7DCB47C8ED4B  1198
31340  6A7DC80C79FE1F2831CD  1140
31350  18797EFE00203AED436A  1025
31360  7DC1CD18793600ED4B6A  1140
31370  7D119B7CCD0178ED4B6A  1168
31380  7DCD18793642C9C1CD18  1218
31390  79AF77326C7D18BEC1CD  1310
31400  1879AF77326D7DC9CD18  1153
31410  79364218E4CD18793642  983
31420  18E8ED4B667DCDD07AAF  1505
31430  32717DED4B667DC34979  1216
31440  110000CDA37C3A00003C  827
31450  320000C9ED5F5F16001A  726
31460  835F3A785C8332707DC9  1115
31470  F578E618F64067F18467  1508
31480  78E6070F0F0F818FC9AF  1018
31490  324879CDEE7A1A77133A  1030
31500  48793CFE08C832487918  982
```

```
31510  EEED4B727DED43647DCD  1523
31520  787BED4B647DED43727D  1323
31530  ED4B747DED43647DCD78  1407
31540  7BED4B647DED43747DED  1442
31550  48767DED43647DCD7878  1295
31560  ED4B647DED43767DED48  1396
31570  787DED43647DCD787BED  1459
31580  4B647DED43767D2A9378  1161
31590  7DFE3B280311200001922  594
31600  9378C921DB7B18F73A65  1278
31610  7DFE133023ED4B647DC5  1215
31620  04ED43647DC1AFCD4979  1300
31630  ED4B647D11DB7BCD2879  1262
31640  ED4B647D3A6F7DC34979  1220
31650  ED4B647D3E42CDB97938  1135
31660  0DC53A667DB9380030C18  775
31670  CE0D18CB3A7A7D3C327A  983
31680  7DCDD07AAFED4B647DCD  1577
31690  49793A707DE51E4F0505  839
31700  ED43647DC3DE7A03C0FF  1518
31710  FF8811B99DAFF5AFF5AF  1765
31720  F5AFF5AE78EE77EFF7EF  2038
31730  F7EFF7AFF5AE750FF003  1702
31740  C0FFFF8811B99DAFF5EF  1856
31750  F7EFF7EC37EC37AC35AC  1712
31760  35AFF5EFF7AFF5AE750F  1685
31770  F003C0FFFFCFF3FFFFEF  2144
31780  F7EFF5A815A815A995A9  1595
31790  95A815A816EFF7AFF5AE  1807
31800  750FF003C0FFFFFFFFBF  1778
31810  FDAFF5AFF5A815ABD5AB  1837
31820  D5ABD5ABD5A816EFF7AF  1831
31830  F5AE750FF0000007B009  935
31840  F0115C115C115CCBDCE7  1221
31850  D4BC74C3843FFC40027F  1351
31860  FEAAABAAAB7FFE000001  1318
31870  E00F903A883A883A8836  1024
31880  D32BE72E3D21C33FFC40  1199
31890  027FFED655D5557FFE00  1360
31900  003C7E7E3C0000CD2879  736
31910  3A485C1F1F1F06280EFE  529
31920  252006EE10ED79263C2D  830
31930  20F4EE10ED792E6410EC  1286
31940  C93E0532EC7C21005A06  935
31950  203AEC7CFE00C83D32EC  1251
31960  7C5707070707B2CBF77723  1014
31970  10FCCDA67CCDA67C18DE  1504
```

MONSTROS 48K

Meteu-se numa aventura, que não lhe correu muito bem, e aqui está o resultado.

Encontra-se fechado num castelo enfeitiçado por uma bruxa, que o obriga a apanhar os seus diamantes, espalhados pelo seu castelo.

A bruxa possui três monstros de estimação que não simpatizaram muito consigo, e que tudo farão para lhe dificultar o seu trabalho. Mas não é assim tão mau, a bruxa, contando com a antipatia dos seus monstros, ofereceu-lhe uma arma enfeitiçada, que não mata os monstros, mas que alivia a situação, transportando-os para outro compartimento do castelo.

Trabalhe o melhor que possa e talvez a bruxa lhe dê a liberdade.

BOA SORTE

Gravação

SAVE "MONSTROS" LINE 0
SAVE "L/M" CODE 40000,1470

GRAFICOS

A- B- C- D- E- F- G- H-

M O N S T R O S 48K

```
  10 REM
  20 REM  © (MT) MARCO & TITO
  30 REM
  40 REM
  50 BORDER 0: PAPER 0
  60 INK 7
  70 CLS : LOAD ""CODE
  80 POKE 23675,129: POKE 23676,
161
  90 LET EF=41440: LET SX=41409:
 GO SUB 960
 100 LET XSC=0: LET FASE=1
 110 LET ROL=45015: LET CB=45002
 120 LET DIR1=45017: LET CO1=450
20: LET CO2=45022
 130 LET CO3=45024: LET SC=45026
 140 LET DIR2=45018
 150 LET DIR3=45019
 160 LET BX=4: LET BY=1
 170 LET PO=0: LET VID=5
 180 LET VAR=45015: LET ACT=4500
6
 190 LET ACT2=45009: LET BONECO=
40527
 200 LET FIREX=40801: LET COR=45
012
```

```
 210 LET INI=41091
 220 POKE ROL,1: POKE CB,27
 230 POKE CB+1,1: POKE CO1,10
 240 POKE CO1+1,10: POKE CO2,5
 250 POKE CO2+1,1: POKE CO3,20
 260 POKE CO3+1,10
 270 LET CX=3: POKE ACT,0
 280 POKE ACT2,0: POKE DIR1,4
 290 POKE DIR2,8: POKE DIR3,2
 300 POKE VAR,1
 310 RESTORE 760: READ Z
 320 GO SUB 770: GO SUB 660
 330 PRINT AT PEEK (CB+1),PEEK C
B; INK 7;"  ";AT PEEK (CB+1)+1,P
EEK CB;"  "
 340 LET L=USR BONECO: IF PEEK U
AR=0 THEN GO TO 630
 350 LET L=USR FIREX
 360 POKE COR,CX+64: LET CX=CX+1
 370 IF CX=8 THEN LET CX=4
 380 LET L=USR INI
 390 BEEP .02,Z: READ Z
 400 IF Z=255 THEN RESTORE 760:
READ Z
 410 LET L=USR FIREX
 420 LET PO=PO+(PEEK SC)*10: POK
E SC,0
 430 PRINT INK 6;AT 19,13;PO;"
": IF PO-XSC>1000 THEN LET FASE=
FASE+1: LET XSC=PO: GO SUB 490:
GO TO 220
 440 LET L=USR BONECO: IF PEEK U
AR=0 THEN GO TO 630
 450 LET L=USR FIREX
 460 LET X=PEEK (CB+1): LET Y=PE
EK CB
 470 IF ATTR (X,Y-2)=6 OR ATTR (
X,Y+3)=6 THEN GO SUB 660
 480 GO TO 340
 490 FOR A=0 TO 20: BEEP .01,-A
 500 BEEP .01,RND*10+60: BEEP .0
1,A
 510 NEXT A: RETURN
 520 FOR A=0 TO 10: LET L=USR SX
 530 NEXT A
 540 CLS
 550 PLOT 16,103: DRAW 224,0
 560 DRAW 0,-31: DRAW -224,0
 570 DRAW 0,31
 580 PRINT AT 10,3; INK 6; INVER
SE 1;"
"
 590 PRINT AT 11,3; INK 5; INVER
SE 1;"
"
```

```
 600 PRINT AT 10,5; INK 6; INVER
SE 1;"FIM DE JOGO";AT 11,5; INK
6;PO;" PONTOS"
 610 GO SUB 490: PAUSE 0
 620 GO TO 90
 630 LET VID=VID-1: IF VID=0 THE
N GO SUB 490: GO TO 520
 640 PRINT INK 4;AT 20,13;VID;"
"
 650 GO SUB 490: GO TO 220
 660 LET PO=PO+40: LET A=INT (RN
D*8)+1
 670 BEEP .01,50: PRINT AT BX,BY
;" ";AT BX+1,BY;" "
 680 IF A=1 THEN LET BX=4: LET B
Y=1
 690 IF A=2 THEN LET BX=10: LET
BY=1
 700 IF A=3 THEN LET BX=15: LET
BY=1
 710 IF A=4 THEN LET BX=1: LET B
Y=30
 720 IF A=5 THEN LET BX=7: LET B
Y=30
 730 IF A=6 THEN LET BX=13: LET
XY=30
 740 BEEP .01,30: PRINT AT BX,BY
; INK 6;"A";AT BX+1,BY;"V"
 750 RETURN
 760 DATA 14,13,8,10,13,10,8,10,
13,13,8,10,13,10,8,8,8,8,8,10,13
,13,8,10,13,255
 770 INK 0: CLS
 780 INK 7
 790 IF FASE=1 THEN LET A$="[illegible]"
 800 IF FASE=2 THEN LET A$="[illegible]"
 810 IF FASE=3 THEN LET A$="[illegible]"
 820 IF FASE=4 THEN LET A$="[illegible]"
 830 IF FASE=5 THEN LET A$="[illegible]": L
ET FASE=0
 840 LET B$=A$( TO 1)
 850 FOR A=0 TO 21 STEP 3: PRINT
 AT A,0; INK 5;A$
 860 NEXT A
 870 FOR A=0 TO 21: PRINT AT A,0
; INK 5;B$;AT A,31;B$;AT A,2;B$;
AT A,29;B$
 880 NEXT A
```

```
 890 PRINT AT 3,3; INK 0;"   ";AT
 6,12;"   ";AT 6,15;"   ";AT 9,7;"
  ";AT 9,15;"   ";AT 9,27;"   ";AT
 12,18;"  "
 900 PRINT AT 15,3; INK 0;"   ";A
T 15,15;"   ";AT 15,19;"   "
 910 FOR A=6 TO 12: PRINT AT A,1
4; INK 5;B$
 920 NEXT A
 930 PRINT AT 7,17; INK 5;B$;AT
8,17;B$;AT 13,18;B$;AT 14,18;B$;
AT 16,15;B$;AT 17,15;B$
 940 PRINT AT 19,5; INK 6;"PONTO
S: "; INK 4;AT 20,5;"VIDAS:   ";V
ID
 950 RETURN
 960 CLS : LET D=1
 970 PRINT INK 5;AT 21,0;"  POR
 M A R C O  &  T I T O    "
 980 PRINT AT 12,8; INK 6;"TECLA
S:"; INK 5;AT 13,8;"(Q)-CIMA";AT
 14,8;"(A)-BAIXO";AT 15,8;"(O)-E
SQUERDA";AT 16,8;"(P)-DIREITA";A
T 17,8;"(M)-DISPARO"
 990 FOR A=4 TO 6: PRINT AT A,7;
 INK D;" M O N S T R O S "
1000 NEXT A: FOR A=0 TO 31: PRIN
T AT 2,A; INK D;"  ";AT 10,A;"  ":
 LET D=D+1
1010 IF D=8 THEN LET D=1
1020 NEXT A
1030 LET A$="[illegible]
[illegible]": PRINT AT 10,0; I
NK 6;A$;AT 2,0;A$
1040 POKE SX+7,70: LET L=USR SX
1050 IF INKEY$<>"" THEN RETURN
1060 LET L=USR EF
1070 GO TO 1040
```

LINGUAGEM MAQUINA

```
40000 3E0F0F0F67E6E0A06F7C  1060
40010 E603EE6867C9CD5A9C04  1313
40020 13AF32709CCD5AA01A77  1112
40030 23131A773A709C3CFE08  847
40040 C332709C13C3599C07F5  1269
40050 CD409CF1772377112000  966
40060 19772B77C9ED48C8AFC5  1391
40070 CD2FA03E02CD159EDAE7  1309
```

```
40080  9D3ACBAFB8CADA9C3AD2  1621
40090  AFCB47C2DA9CCB4FC2DA  1711
40100  9C083E05CDC29D380F04  859
40110  3ACBAFB830083E0133D2  999
40120  AFC3DA9CED4BC8AF0404  1439
40130  3E05CDC29DDADA9CED4B  1527
40140  C8AF3ACBAFB8DADA9C3E  1649
40150  0232D2AFCD541FD24FA0  1205
40160  3AD2AFCB47C21E9DCB4F  1380
40170  C2329DCB57C2489DED4B  1426
40180  C8AF0C0C3E05CDD49DDA  1258
40190  5C9D0DED43C8AFC1AFCD  1514
40200  719CED4BC8AF1161A1CD  1436
40210  509CED4BC8AF3AD4AFC3  1583
40220  719CED4BC8AF053E08CD  1233
40230  C29DDA5C9DED43C8AFC3  1692
40240  059DED4BC8AF04043E05  924
40250  CDC29DDA5C9D0ED43C8  1532
40260  AFC3059DED4BC8AF0D3E  1294
40270  05CDD49DDA5C9DED43C8  1550
40280  AFC3059DED4BC8AF3ACA  1479
40290  AFB928238110C0C3E05  599
40300  CDD49D38183E0832D2AF  1169
40310  C3DA9C0D3E05CDD49D38  1279
40320  083E0432D2AFC3DA9CED  1315
40330  4BC8AF3ACBAFB8282338  1201
40340  1104043E05CDC29D3818  728
40350  3E0232D2AFC3DA9C053E  1135
40360  05CDC29D38083E0132D2  948
40370  AFC3DA9C3AD8AF32D2AF  1628
40380  CD2FA0C3DA9CF5CD409C  1651
40390  F156BA28072356BA2802  909
40400  B7C937C9F5CD409CF156  1637
40410  BA28F5112000195 68A28  887
40420  EDB7C9ED4BC8AFCD02A0  1675
40430  CDF89DED43C8AFC3059D  1346
40440  3AD6AFE61F4FCD1FA03A  1241
40450  D6AFE60F47AFCD329E38  1349
40460  EB3AE2AF3C32E2AFC9F5  1651
40470  CD409CF156BA28842356  1279
40480  BA28AF112000195 68A28  787
40490  A72856BA28A2B7C9F5CD  1518
40500  409CF156BA209723568A  1223
40510  2092112000195 6BA208A  694
40520  2B56BA2085B7C9ED4BCA  1378
40530  AF3E07CD329EDAF09FC5  1471
40540  01FEFBED78CB47CCC49E  1695
40550  01FEFDED78CB47CCD99E  1718
40560  01FEDFED78CB4FCCF09E  1719
40570  01FEDFED78CB47CC059F  1477
40580  01FE7FED78CB57CC1C9F  1420
40590  C1AFCD719CED4BCAAF3A  1589
40600  D3AFCB472017CB4F2018  1053
40610  CB572019114 1A1CD509C  1031
40620  ED4BCAAF3E07C3719C11  1239
```

```
40630  E1A018EF1101A118EA11  1102
40640  21A118E5ED4BCAAF053E  1203
40650  05CDC29DD8ED43CAAF3E  1520
40660  0132D3AFC9ED4BCAAF04  1331
40670  043E05CDC29DD805ED43  1152
40680  CAAF3E0232D3AFC9ED4B  1390
40690  CAAF0D3E05CDD49DD8ED  1484
40700  43CAAF3E0432D3AFC9ED  1384
40710  4BCAAF0C0C3E05CDD49D  1117
40720  D80DED43CAAF3E0832D3  1241
40730  AFC93AD3AFCB57200ECB  1350
40740  5F201DC93AD0AFCB47C0  1264
40750  ED4BCAAF0DCD409C7EFE  1507
40760  05C83E0132D0AFED43CC  1209
40770  AFC93AD1AFCB47C0ED4B  1596
40780  CAAF0C0CCD409C7EFE05  1211
40790  C83E0132D1AFED43CEAF  1382
40800  C93AD0AFCB47282FED4B  1315
40810  CCAFC50DCD409C7EFE05  1399
40820  2857FE0020C68ED43CCAF  1200
40830  C1CD409C3600ED4BCCAF  1353
40840  1189A1CD6DA0ED4BCCAF  1528
40850  CD409C36023AD1AFCB47  1197
40860  C8ED4BCEAFC50CCD409C  1527
40870  7EFE05282DFE00203AED  1061
40880  43CEAFC1CD409C3600ED  1387
40890  4BCEAF1189A1CD6DA0ED  1530
40900  4BCEAFCD409C3802C9C1  1331
40910  CD409CAF7732D0AF18BF  1367
40920  C1CD409CAF7732D1AFC9  1547
40930  CD409C360218E4CD409C  1158
40940  360218E8ED4BCAAFCD02  1208
40950  A0AF32D7AFED4BCAAFC3  1659
40960  719CC5110000CD509CCD  1129
40970  16A011E803C1CD509CC3  1253
40980  16A011E803210A00C3C1  865
40990  A1ED5F5F18001A838F3A  920
41000  785C8332D6AFC93AD6AF  1430
41010  E607C547CD3FA010FBC1  1393
41020  C31FA03AD8AF07FE10CC  1316
41030  4CA032D8AFC93E01C9C1  1335
41040  1164002164009CDC1A1C9  1010
41050  F578E618F64067F18487  1508
41060  78E6070F0F0F816FC9AF  1018
41070  32709CCD5AA01A77133A  995
41080  709C3CFE08C832709C18  1132
41090  EEED4BDCAFED43C8AF3A  1682
41100  D9AF32D2AFCD819CED4B  1629
41110  C8AFED43DCAF3AD2AF32  1567
41120  D9AFED4BDEAFED43C8AF  1780
41130  3ADAAF32D2AFCD819CED  1613
41140  4BC8AFED43DEAF3AD2AF  1594
41150  32DAAFED4BE0AFED43C8  1658
41160  AF3ADBAF32D2AFCD819C  1582
41170  ED4BC8AFED43E0AF3AD2  1658
```

```
41180  AF32DBAFC90FF03FFC7F  1517
41190  FE7FFE7FFE7FFE3FFC0F  1727
41200  F003C01FF873CE6FF66F  1503
41210  F65E7A1C381C360FF036  941
41220  1C700E66666666700E39  746
41230  9C0FF003C01E7873CE6F  1188
41240  F66FF65E7A1C381C380F  1002
41250  F039FC70EE666E666E70  1435
41260  6E38FC0FF003C006603E  1032
41270  60206020E01FE003C00F  945
41280  C00FF03F1C760E766676  1006
41290  66770E3F9C0FF003C006  910
41300  60067C06040704407F603  505
41310  C003F007E01FF813C91F  1195
41320  F807E00A503A5C781EF9  1118
41330  9FDE785FFA5E7A5DBA44  1412
41340  2204200E70FDFDFD00DF  1178
41350  DFDF00EEAAAAAAAAAAAA  1704
41360  8B007E425A5A427E0000  751
41370  183C7E7C3241800D6D6D  927
41380  6D6D6D6D611824242435A  785
41390  5A5A99A5A5995A5A2424  1068
41400  1800007E81FF7E00003A  718
41410  485C1F1F1F05450EFE25  538
41420  2006EE10ED79263C2D20  825
41430  F4EE10ED792E6410ECC9  1465
41440  210058010003 7EFE0728  552
41450  083CE607772308788 1C8  970
41460  18F03E0118F200000000  593
```

PERIGO!!!

Sendo o responsável por uma máquina, que assegura o abastecimento de alimentos, a uma mina. Você tem que apanhar as embalagens de comida, que descem pela mina, para cumprir a sua missão. Quando apanhar sete embalagens, é feita a distribuição e receberá os respectivos pontos.

Não deixe o tempo chegar a zero, pois a vida dos mineiros, encontra-se em PERIGO.

BOA SORTE

Gravação

SAVE "PERIGO!!!" LINE 0
SAVE "Code" CODE 31000,390

GRAFICOS

[illegible]-A

PERIGO!!!

```
  10 REM
  20 REM   © (MT) MARCO & TITO
  30 REM
  40 REM
  50 LOAD ""CODE : POKE 23658,8
  60 PAPER 0: BORDER 0: INK 7: C
LS
  70 GO SUB 800
  80 GO TO 620
  90 PAPER 0: BORDER 0: INK 0: C
LS
 100 LET PO=0: LET TE=99
 110 LET G=-1
 120 PRINT AT 21,8; INK 7;"TEMPO
";AT 21,19;"PONTOS"
 130 FOR F=0 TO 19: PRINT AT F,0
;"[redacted]
[redacted]": NEXT F
 140 POKE 31007,20
 150 PRINT AT 20,1; INK 5; PAPER
 5;"[redacted]
[redacted]"
 160 PRINT AT 0,0; PAPER 1; INK
7; FLASH 1;"*** PRIMA (C)-PARA
COMECAR ***"
 170 LET L=USR 31031
 180 IF INKEY$<>"C" THEN GO TO 1
60
 190 PRINT AT 0,0;"[redacted]
[redacted]"
 200 POKE 32000,RND*31: POKE 320
01,0: POKE 32002,3: POKE 32004,0
: POKE 32004,8: POKE 32005,18
 210 INK 9
 220 LET L=USR 31067
 230 IF INKEY$="1" THEN LET X=PE
EK 32000
 240 LET TE=TE-.1: IF TE<=0 THEN
 GO TO 540
 250 PRINT AT 21,14;INT TE;" "
 260 LET PO=PO+1: PRINT AT 21,26
;PO
 270 IF ATTR (17,PEEK 32004)=7 T
HEN GO SUB 290
 280 GO TO 220
 290 LET G=G+1
 300 LET X=PEEK 32004
 310 POKE 31007,50: LET L=USR 31
000: POKE 31007,20
 320 PRINT AT 17,X; PAPER 0; INK
 0; INK 0;"[redacted]"
```

```
 330 PRINT AT 21,G; INK G; PAPER
 G+1;"●"
 340 IF G=6 THEN GO TO 370
 350 POKE 32002,3: POKE 32001,0:
 POKE 32000,RND*31
 360 RETURN
 370 LET J=1: FOR F=23200 TO 232
06: POKE F,J: LET J=J+1: NEXT F
 380 FOR F=0 TO 63
 390 POKE 31007,40: LET L=USR 31
000: LET L=USR 31293: OUT 254,F:
 NEXT F: POKE 31007,20
 400 BORDER 0
 410 FOR F=0 TO 19: PRINT AT F,0
;"
  ": LET L=USR 31000: NEXT F
 420 IF G=-1 THEN GO TO 490
 430 FOR F=1 TO G+1: BEEP .2,F: :
 PRINT AT 2,F+10; INK F; PAPER 0
;"●";AT 21,F-1;" ": NEXT F
 440 BEEP .4,6: PRINT AT 4,0;"
    PONTOS CORRESPONDENTES    "
 450 FOR F=1 TO G+1: PRINT AT F*
2+4,8; PAPER 0; INK F;"●": LET L
=USR 31000
 460 PRINT AT F*2+4,13;F;" X 100
 = ";F*100
 470 LET PO=PO+(F*100): BEEP .1,
F*5: PRINT AT 21,26;PO
 480 NEXT F
 490 IF TE<=0 THEN GO TO 590
 500 FOR F=1 TO 5: LET TE=TE+F:
PRINT AT 21,14;INT TE: BEEP .4,F
: NEXT F
 510 FOR F=0 TO 100: LET L=USR 3
1000: NEXT F
 520 FOR F=0 TO 21: PRINT AT F,0
;"
  ": LET L=USR 31000: NEXT F
 530 LET G=-1: PAPER 0: INK 0: C
LS : GO TO 120
 540 FOR F=15 TO 255 STEP 10
 550 LET L=USR 31067
 560 POKE 31007,F
 570 FOR H=1 TO F/4: NEXT H: NEX
T F
 580 POKE 31007,20: GO TO 370
 590 BEEP 4,-10: CLS : PRINT AT
10,10; FLASH 1;"FIM DE JOGO"
 600 FOR F=0 TO 300: OUT 254,0:
OUT 254,16: NEXT F
 610 CLS
```

```
 620 PRINT AT 11,1; PAPER 2;"
                    ": FOR F=12 TO 16:
 PRINT AT F,1; PAPER 2;"
          ": NEXT F: PRINT AT 17,
1; PAPER 2;"                    "
 630 PRINT AT 12,2; INVERSE 1;"T
ECLAS:"
 640 PRINT AT 14,2;"L-ESQUERDA"
 650 PRINT AT 16,2;"'ENTER'-DIRE
ITA"
 660 PRINT AT 19,0; PAPER 8;"
                                ";A
T 21,0;"
                                "
 670 PRINT AT 20,1;"****** (I) P
ARA INICIAR ******"
 680 PRINT AT 20,0; PAPER 2;" ";
AT 20,31; PAPER 2;" "
 690 PRINT AT 0,0;"
  
  
 700 PRINT AT 1,0;"
  
 710 PRINT AT 2,0;"
  
 720 PRINT AT 3,0;"
  
 730 PRINT AT 4,0;"
  
 740 PRINT AT 5,0;"
  
 750 PRINT AT 6,0;"
  
 760 PRINT AT 7,27;"●"
 770 LET L=USR 31006
 780 IF INKEY$="I" THEN GO TO 90
 790 GO TO 770
 800 DATA 60,126,255,255,255,255
,126,60
 810 FOR F=USR "A" TO USR "A"+7:
 READ I: POKE F,I: NEXT F
 820 RETURN
```

LINGUAGEM MAQUINA

```
31000 3A485C1F1F1F06140EFE 609
31010 2520066E10ED79263C2D 630
31020 20F4EE10ED792E6410EC 1286
31030 C9212058116278013C00 552
31040 C506081A07380736002 398
```

```
31050  10F8180536 2D2310F1C1  877
31060  0B1378B120E6C9CD3D7A  1176
31070  CD187 9ED4B047DCD2D7A  1153
31080  350001FEBFED78CB47CC  1335
31090  E27901FEBFED78CB4FCC  1636
31100  F079ED4B047DCD2D7A35  1228
31110  24ED4B007DCD2D7A3500  999
31120  3A027DFE012835FE0228  829
31130  1B040478FE16287ACD2D  843
31140  7A7EFE2DCCFE7905CD2D  1381
31150  7A3507ED43007DC90400  829
31160  CD2D7A7EFE00CC167A05  1105
31170  CD2D7A3507ED43007DC9  1053
31180  040DCD2D7A7EFE00CC16  995
31190  7A05CD2D7A3507ED4300  864
31200  7DC9ED4B047D79FE1FC8  1373
31210  0CED43047DC9ED4B047D  1057
31220  79FE00C80DED43047DC9  1222
31230  3A785C57ED5F828 2E601  1180
31240  28083E023202 7DC93E01  551
31250  32027DC93E033202 7DC9  821
31260  ED5F473A785C80E81F4F  1141
31270  0600ED43007DC9780F0F  768
31280  0F67E6E0A96F7CE503EE  1447
31290  5867C921A05A0607 7E3C  874
31300  772310FAC921005 8 9140  807
31310  01F3ED5FF847777238578  1178
31320  8120F5FBC9000000000000  905
31330  00000000 07DEFDDF800000  833
31340  00003FBEF77C000000000  624
31350  7BF7DFEE00000000 3FBF  1045
31360  7EF000000000007DF7F7EE  1223
31370  0000000001F7DFDFC0000  681
31380  00006FBEFEFA00000000  805
```

COMBATE

Um jogo em que o público apela para um bom combate, entre si e o computador ou entre si e um amigo seu.

O leitor poderá colocar este jogo, num Capítulo imaginário, entre o 3.º e 4.º Capítulos, devido ao programa em causa, ser tanto para um jogador como para dois.

Você encontra-se num ringue, enfrentando o seu adversário, num combate, que só o mais forte pode vencer.

O combate é constituído por três assaltos, assim quem conseguir totalizar primeiro três assaltos ganhará o combate.

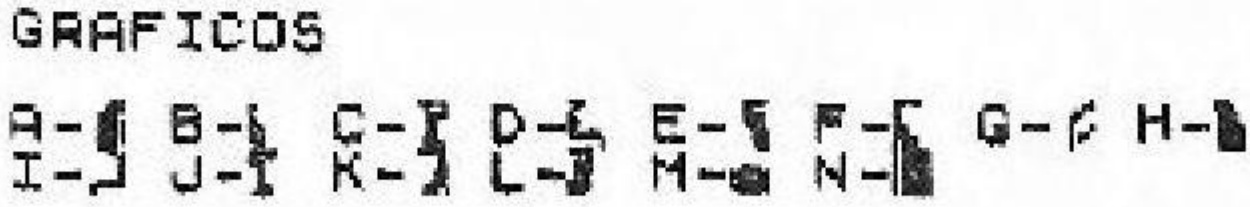
Existindo vários níveis de jogo, o leitor poderá se treinar em níveis mais baixos, até conseguir chegar e vencer o nível mais alto, e por conseguinte mais difícil.

BOA SORTE

Gravação

SAVE "COMBATE" LINE 0
SAVE "CODE" CODE 31000,850

GRAFICOS



COMBATE

```
  10 REM
  20 REM   © (MT) MARCO & TITO
  30 REM
  40 REM
  50 LOAD ""CODE : POKE 23675,24
4: POKE 23676,193
  60 LET SC=31001: LET VAL=31208
: LET RND=31000
  70 LET VEN=31002: LET FLAG=310
03: LET SC2=31005
  80 LET SOM=31523: LET FLAG2=31
004: LET JO=31006
  90 LET MTC=31005: LET CO=31730
 100 LET PX=31813: LET EF2=31554
 110 LET EF=31510: LET TEC=31007
 120 LET SCV1=0: LET SCV2=0
 130 POKE MTC,0: POKE SC,0
 140 POKE SC2,0: POKE VEN,88
 150 POKE FLAG,0: POKE FLAG2,0
 160 PAPER 7: INK 0
 170 CLS : BORDER 7
 180 GO SUB 510
 190 GO SUB 250: GO SUB 910
 200 LET RT=USR TEC: POKE VEN,88
 210 FOR A=0 TO 20: LET L=USR EF
 220 NEXT A
 230 IF RT=1 THEN GO TO 680
 240 IF RT=2 THEN GO TO 760
 250 GO SUB 890
 260 LET A$="[illegible]
[illegible]"
 270 PRINT AT 12,0; INK 1;A$; IN
K 7;AT 15,0;A$; PAPER 4; INK 4;A
T 14,0;A$;AT 13,0;A$
 280 LET L=USR EF
 290 GO SUB 440: FOR A=16 TO 18
 300 PRINT INK 2;AT A,0;"▓";AT A
,31;"▓": NEXT A
 310 FOR A=1 TO 6: PRINT AT A,1;
INK 1;"
      ": NEXT A
 320 LET A$=" C O M B A T E": GO
SUB 850
 330 POKE CO,0: POKE CO+1,3
 340 LET L=USR PX
 350 LET D=0: LET D2=7
 360 FOR A=0 TO 31: PRINT AT 0,A
; INK D;"▓";AT 7,A; INK D2;"▓"
 370 LET D2=D2-1: LET D=D+1
 380 IF D=8 THEN LET D=0
 390 IF D2=-1 THEN LET D2=7
 400 NEXT A
 410 LET D=7: FOR A=0 TO 7
```

```
 420 PRINT AT A,31; INK A;"▒";AT
 A,0; INK D;"▒": LET D=D-1
 430 NEXT A: RETURN
 440 LET A$=" JOGADOR 1  00    ":
 POKE CO,0
 450 POKE CO+1,22: GO SUB 650
 460 LET L=USR PX
 470 IF PEEK JO=2 THEN LET A$="
  JOGADOR 2  00": POKE CO,15: PO
KE CO+1,22: GO SUB 650: LET L=US
R PX: RETURN
 480 LET A$="  COMPUTADOR  00
": POKE CO,15
 490 POKE CO+1,22: GO SUB 650
 500 LET L=USR PX: RETURN
 510 FOR A=10 TO 15: PRINT AT A,
5; OVER 1; INK RND*1+1;"
          ": NEXT A
 520 LET A$="1-UM JOGADOR": GO S
UB 650
 530 POKE CO,5: POKE CO+1,10
 540 LET L=USR PX
 550 LET A$="2-DOIS JOGADORES":
GO SUB 650
 560 POKE CO,5: POKE CO+1,12
 570 LET L=USR PX
 580 PRINT AT 16,5; INK 1;"TECLA
S";AT 16,5; INK 3;"JOGADOR 1";AT
 17,5;"(D)e(F)";AT 16,15;"▒ JOGA
DOR 2";AT 17,15;"▒ (L)e(ENTER)"
 590 LET A$="P o r    M A R C O
&  T I T O  ©": GO SUB 650
 600 POKE CO,0: POKE CO+1,22
 610 LET L=USR PX: GO SUB 310
 620 IF INKEY$="1" THEN GO TO 10
10
 630 IF INKEY$="2" THEN CLS : PO
KE JO,2: RETURN
 640 LET L=USR EF2: LET L=USR SO
M: GO TO 620
 650 LET D=32000: FOR A=1 TO LEN
 A$: POKE D,CODE A$(A)
 660 LET D=D+1: NEXT A
 670 POKE D,255: RETURN
 680 LET SCV1=SCV1+1: GO SUB 940
: IF SCV1=3 THEN GO TO 710
 690 GO SUB 880: GO SUB 890
 700 GO SUB 910: GO TO 200
 710 LET A$="    VENCEDOR JOGADOR
 1": GO SUB 650
 720 POKE CO,4: POKE CO+1,3
 730 LET L=USR PX
 740 FOR A=0 TO 100: LET L=USR E
F: LET L=USR EF2: NEXT A
 750 GO TO 60
```

```
 760 LET SCV2=SCV2+1: GO SUB 940
: IF SCV2=3 THEN GO TO 820
 770 LET A$=CHR$ (SCV2+48): POKE
 CO,31
 780 POKE CO+1,22: GO SUB 650
 790 LET L=USR PX
 800 GO SUB 880: GO SUB 890
 810 GO SUB 910: GO TO 200
 820 IF PEEK JO=2 THEN LET A$="
  VENCEDOR JOGADOR 2": GO SUB 65
0: POKE CO,4: POKE CO+1,3: LET L
=USR PX: GO TO 850
 830 LET A$="    VENCEDOR COMPUTA
DOR": GO SUB 650
 840 POKE CO,4: POKE CO+1,3: LET
 L=USR PX
 850 FOR A=0 TO 100: LET L=USR E
F
 860 LET L=USR EF2: NEXT A
 870 GO TO 60
 880 FOR A=16 TO 18: PRINT AT A,
1;"
  ": NEXT A: RETURN
 890 PRINT AT 16,11;"▟▙";AT 16,1
2;"▟▙";AT 17,11;"▜▛▜▛";AT
18,11;"▜▛";AT 18,12;"▟▙"
 900 RETURN
 910 FOR A=0 TO 70: LET L=USR EF
: LET L=USR EF2
 920 NEXT A: BEEP 1.5,30
 930 RETURN
 940 LET A$=CHR$ (SCV1+48): POKE
 CO,13
 950 POKE CO+1,22: GO SUB 650
 960 LET L=USR PX
 970 LET A$=CHR$ (SCV2+48): POKE
 CO,31
 980 POKE CO+1,22: GO SUB 650
 990 LET L=USR PX
1000 RETURN
1010 CLS : POKE JO,1: INPUT "DIF
ICULDADE (0-7)??";A: IF A>7 OR A
<0 THEN GO TO 1010
1020 POKE VAL,255
1030 IF A=1 THEN POKE VAL,127
1040 IF A=2 THEN POKE VAL,63
1050 IF A=3 THEN POKE VAL,31
1060 IF A=4 THEN POKE VAL,15
1070 IF A=5 THEN POKE VAL,7
1080 IF A=6 THEN POKE VAL,3
1090 IF A=7 THEN POKE VAL,1
1100 RETURN
```

LINGUAGEM MAQUINA

```
31000  C4015801000013A7779  565
31010  C60A327779FE0ACC427B  1155
31020  01FEFDED78F6F3FEF3CA  2053
31030  D17901FEFDED78CB57CA  1687
31040  6179CB5FCA6C793A1879  1153
31050  FE03C2D1793A197906 05  995
31060  3C10FD321979AF321879  698
31070  C3D1793A1B79CBC73218  1210
31080  79C347793A1B79CBCF32  1174
31090  1B79C347792223A1D79FE  1031
31100  0138073D321D79CD857A  689
31110  01FE8FED78F8FCFEFCCA  2009
31120  EE7901FE8FED78CB47CA  1838
31130  8879CB4FCAC6793A1C79  1318
31140  FE03C2EE793A1D79060 5  1029
31150  3C10FD321D79AF321C79  903
31160  C3EE793A1C79CBC7321C  1241
31170  79C3A1793A1C79CBCF32  1265
31180  1C79C3A1793A1979FE01  1085
31190  D4147A3A1E79FE02CA76  1141
31200  79CDFC793A1879E5FFFE  1841
31210  01CC257A3A1A79FE0928  871
31220  3AFEA8263AC31F793A14  1003
31230  7B3C32147B5F3A157B03  804
31240  32157B5F3A7B5C8332 18  754
31250  79C93A1A793D321A793A  643
31260  19793D321979C3377A3A  833
31270  1A793C321A79C3B97A01  907
31280  0100C90102 00C93E1E32  546
31290  B57A32B87A3E5032B67A  1156
31300  3E1832B77A2AB57AED4B  1093
31310  B77AA7CB16 2B0 5C25 17A  1142
31320  2AB57ACD937A22B57AED  1303
31330  4BB77A0DC8ED43B77AC3  1397
31340  507A2AB57AED4BB77AA7  1331
31350  CB1E2305C2767A2AB57A  1052
31360  CD937A22B57AED4BB77A  1428
31370  0DC8ED43B77AC3757A7C  1360
31380  E607FE07CA9D7A24C97D  1341
31390  E6E0FEE0CAAC7A11E006  1676
31400  A7ED52C97CFE57C81120  1401
31410  0019C97E50011E3E0132  676
31420  B57A3E5032B67A3E1832  935
31430  B77A3E1E32B87A2AB57A  1098
31440  ED4BB77AA7CB1E2305C2  1251
31450  D57A2AB57ACDF27A22B5  1454
31460  7AED4BB77A0DC8ED43B7  1439
31470  7AC3D47A7CE607FE07CA  1475
31480  FC7A24C97DE6E0FEE0CA  1870
31490  0B7B11E008A7ED52C97C  1192
31500  FE57C81120001 9C93202  996
```

```
31510  2600ED5F6F11004801FF  826
31520  07ED803A485C1F1F1F06  741
31530  050EFE252006EE10ED79  960
31540  253C2D20F4EE10ED792E  1077
31550  C810ECCC9210058060 07E  906
31560  FE3S28063D772310F6C9  1034
31570  3E3F18F7AF327C78F678  1233
31580  E618F6406 7F1846779E8  1493
31590  070F0F0F818F1A773A7C  819
31600  783CFE0813C8327C78C3  1156
31610  5A7887210 07D7EFEFFC8  1213
31620  23E5CD8C78E118F447FE  1550
31630  20384DFE80304 92A365C  856
31640  110800191 0FD0808 11E2  676
31650  787EC81F121312132310  808
31660  F6ED48F27B11E27BCD88  1580
31670  7B0411EA78CD587BED46  1227
31680  F278 0CED43F27B79FE20  1453
31690  C00E000404ED43F27B78  1003
31700  FE18D80600ED43F278C9  1370
31710  3E3F18AA00001E1E2323  449
31720  2525292931311E1E0000  314
31730  0000070F1C1D3D3D393C  318
31740  80C08040A050D0203F11  1072
31750  0F0C0C0C1C1EE0408 0C0  717
31760  80FC02021F101E1E0E0F  520
31770  0F07FC40C0C0E0608888  1410
31780  01030C181112141 0E0F0  575
31790  78786C7C7C7C0C030303  725
31800  013F4040 7C88F0303030  836
31810  38783F0203030706 1D1D  318
31820  F8087878 70F0F0E00000  1312
31830  003CC33CC33C8FBFBFBF  1334
31840  BFBFBFBF000000000000  784
```

ZIG ZAG

Esta variante do clássico jogo **BREAK OUT**, tem uma particularidade; os tijolos não são destruídos. Assim o único objectivo é manter a bola em jogo, com a ajuda do taco.

O jogo pode ser jogado em vários níveis e a pontuação, é dada em relação ao tempo que a bola permaneceu em jogo.

BOA SORTE

Gravação

SAVE "ZIG ZAG" LINE 0
SAVE "Code" CODE 31000,220

GRAFICOS

●-A ◆-B ●-C ●-D •-E •-F

ZIG ZAG

```
  10 REM
  20 REM    © (MT) MARCO & TITO
  30 REM
  40 REM
  50 LOAD ""CODE : POKE 23658,8
  60 GO SUB 440
  70 GO SUB 230
  80 PAPER 0: BORDER 0: INK 0: C
LS
  90 PRINT AT 0,0; INK 3; PAPER
2;"▄▄▀▄▄▀                         ▙
▙▄"
 100 FOR F=0 TO 21: PRINT AT F,0
; INK 3; PAPER 2;"▟";AT F,31;"▙"
: NEXT F
 110 FOR F=1 TO 21: PRINT AT F,1
;"●●●●●●●●●●●●●●●●●●●●●●●●●●●●●●
": NEXT F
 120 PRINT AT 7,2; PAPER 4;"
";AT 7,25;"     ";AT 3,5;"     ";AT
 3,27;"    ";AT 9,12;"    ";AT 9,16
;"   "
 130 PRINT AT 3,13; PAPER 2;"
";AT 9,12;"     "; PAPER 6;AT 12,2
7;"   ";AT 3,23;"    "
 140 PRINT AT 0,11; PAPER 2; INK
7;"PONTOS         "
 150 POKE 32000,INT (RND*10+10):
 POKE 32001,5: POKE 32002,1: POK
E 32003,1: POKE 32004,10
 160 LET O=USR 31198
 170 LET PO=0
 180 LET O=USR 31000
 190 FOR F=0 TO L: NEXT F
 200 IF PEEK 32001=21 THEN GO TO
 380
 210 LET PO=PO+1: PRINT AT 0,18;
 PAPER 2; INK 7;PO
 220 GO TO 180
 230 PAPER 0: BORDER 0: INK 9: C
LS
 240 PRINT AT 3,5;"█████ █ ████";
AT 4,8;"█   █   █";AT 5,7;"█   █  █
";AT 6,6;"█  █ ██"
 250 PRINT AT 7,6;"█    █ █   █";
AT 8,5;"█████ █ ████"
 260 PRINT AT 9,5;"█████ █ ████"
 270 PRINT AT 10,11;"████ ███  █
███";AT 11,14;"█ ████ █   █";AT 1
2,13;"▙ █   █ █";AT 13,12;"█    █
███ █ ▙";AT 14,11;"█    █ ▙ █ █
 █";AT 15,11;"████ █ █ ████"
```

```
 280 PRINT AT 17,3; INK 7; PAPER
 1;"TECLAS : P > PARA DIREITA "
 290 LET U=1: PRINT AT 1,0; FLAS
H 1;"•●●●I - PARA INICIAR O JOGO
 ●●●●"
 300 FOR F=3 TO 16: PRINT AT F,0
; INK U; OVER 1;"
                ": LET U=U+1: IF U=7 T
HEN LET U=1
 310 IF INKEY$="I" THEN GO TO 34
0
 320 NEXT F
 330 GO TO 300
 340 INPUT "NIVEL (0-5)";A
 350 IF A<0 OR A>5 THEN GO TO 34
0
 360 LET L=A*5
 370 RETURN
 380 FOR F=7 TO 0 STEP -.2
 390 PRINT AT PEEK 32001,PEEK 32
000; INK F;CHR$ (144+(RND*5))
 400 BEEP .01,F: NEXT F
 410 CLS : PRINT AT 10,9; INK 9;
"PONTOS >"; BRIGHT 1;PO
 420 BEEP .1,0: BEEP .1,4: BEEP
.1,7: BEEP .1,12: IF INKEY$<>""
THEN GO TO 70
 430 GO TO 420
 440 FOR f=USR "A" TO USR "F"+7:
 READ A: POKE F,A: NEXT F
 450 RETURN
 460 DATA 60,126,255,255,255,255
,126,60
 470 DATA 24,60,126,255,255,126,
60,24
 480 DATA 0,60,126,126,126,126,6
0,0
 490 DATA 0,24,60,126,126,60,24,
0
 500 DATA 0,0,24,60,60,24,0,0
 510 DATA 0,0,0,24,24,0,0,0
```

LINGUAGEM MAQUINA

```
31000 ED4B007DCDCE79360011 1040
31010 0058210D7D01C002EDB0 867
31020 01FEDFED78CB47CAAD79 1605
31030 18673A047D4F0614CDCE 830
31040 79362D23362D23362D23 523
31050 362D23362DED4B007D05 675
31060 CDCE797EFE00C4BC79ED 1654
31070 4B007D0DCDCE797EFE00 1125
```

```
31060  C4C579ED4B007D0CCDCE  1374
31090  797EFE00C4C579ED4B00  1387
31100  7D04CDCE797EFE00C4BC  1425
31110  79ED4B007D3A027D8047  942
31120  3A037D814FED43007DCD  1028
31130  CE793807C93A047D3DFE  1091
31140  00CA3879320470188B3A  779
31150  047D3CFE1BCA3879320A  903
31160  7DC3387933A027DED443  1037
31170  027DC93A037DED443203  872
31180  7DC9780F0F0F67E6E0A9  1217
31190  8F7CE603EE6867C9110D  1128
31200  7D21005801C002EDB0C9  1055
31210  5801C002EDB0C9000000  897
```

CAPÍTULO 5

EFEITOS ESPECIAIS

Vamos neste Capítulo, descrever algumas rotinas, que servirão para melhorar a qualidade e apresentação dos seus programas.

Um efeito torna-se mais atraente se possuir, algo que não se esteja acostumado a presenciar.

Iremos assim vos fornecer, várias rotinas muito pequenas, que podem ser usadas sem complicações de maior, nos seus próprios programas.

As rotinas são escritas em Linguagem Máquina, mas estão inseridas num programa em Basic, para que o leitor as introduza com maior facilidade.

Acompanha a listagem de Basic, uma listagem de Assembler, que dará ao leitor a possibilidade de seguir a rotina passo a passo, tentando compreender o seu funcionamento.

Quanto à gravação, todos os efeitos poderão ser gravados da mesma forma: **SAVE "Nome do Efeito" LINE 0.**

EFEITO 1

```
  10 REM
  20 REM   © (MT) MARCO & TITO
  30 REM
  40 PAPER 0: BORDER 0: INK 7: C
LS
  50 FOR F=USR "A" TO USR "A"+7
STEP 2: POKE F,85: POKE F+1,170:
 NEXT F
  60 DATA 33,0,88,1,0,2,126,60,1
19,35,11,120,177,32,247,201
  70 FOR F=30000 TO 30015: READ
A: POKE F,A: NEXT F
  80 FOR F=0 TO 7: FOR G=0 TO 7:
 PRINT AT F,G*2; PAPER F; INK G;
"▓▓": NEXT G: NEXT F
  90 FOR F=0 TO 7: FOR G=0 TO 7:
 PRINT AT F,30-G*2; PAPER F; INK
 G;"▓▓": NEXT G: NEXT F
 100 FOR F=0 TO 7: FOR G=0 TO 7:
 PRINT AT 15-F,30-G*2; PAPER F;
INK G;"▓▓": NEXT G: NEXT F
 110 FOR F=0 TO 7: FOR G=0 TO 7:
 PRINT AT 15-F,G*2; PAPER F; INK
 G;"▓▓": NEXT G: NEXT F
 120 RANDOMIZE USR 30000: PAUSE
2: GO TO 120
```

LISTAGEM EM ASSEMBLER

```
100 ;  (MT) MARCO & TITO
110
120          ORG   30000
130 EFE1     LD    HL,22528
140          LD    BC,512
150 ST       LD    A,(HL)
160          INC   A
170          LD    (HL),A
180          INC   HL
190          DEC   BC
200          LD    A,B
210          OR    C
220          JR    NZ,ST
230          RET
```

EFEITO 2

```
  10 REM
  20 REM   © (MT) MARCO & TITO
  30 REM
  40 FOR F=31000 TO 31024: READ
A: POKE F,A: NEXT F
  50 DATA 243,1,255,0,33,0,88,17
,0,3,237,95,119,211,254,35,27,12
2,179,32,245,16,237,251,201
  60 FOR F=0 TO 21: PRINT AT F,4
;"TESTE TESTE TESTE TESTE": NEXT
 F
  70 RANDOMIZE USR 31000
```

LISTAGEM EM ASSEMBLER

```
100 ;  (MT) MARCO & TITO
110
120          ORG   31000
130 EFE2     DI
140          LD    BC,255
150 AG       LD    HL,22528
160          LD    DE,768
170 ST       LD    A,A
180          LD    (HL),A
190          OUT   (254),A
200          INC   HL
210          DEC   DE
220          LD    A,D
230          OR    E
240          JR    NZ,ST
250          DJNZ  AG
260          EI
270          RET
```

EFEITO 3

```
  10 REM
  20 REM   © (MT) MARCO & TITO
  30 REM
  40 REM
  50 FOR A=31000 TO 31074: READ
B: POKE A,B: NEXT A
  60 DATA 6,175,14,0,197,62,175,
144,71,167,31,55,31,167,31,168,2
30,248,168,103,121,7,7,7,168,230
,199,168,7,7,111,121,230,7,193
  70 DATA 62,32,86,95,122,211,25
4,123,54,0,35,61,194,61,121,205,
85,121,5,120,254,255,194,28,121,
201,197,6,0,122,211,254,211,254,
211,254,16,247,193,201
  80 FOR A=0 TO 21 STEP 1: PRINT
 AT A,0;"■TESTE TESTE TESTE TEST
E TESTE ■": NEXT A
  90 LET L=USR 31000
```

LISTAGEM EM ASSEMBLER

```
100     (MT) MARCO & TITO
110
120          ORG   31000
130 EF       LD    B,175
140          LD    C,0
150 AG       PUSH  BC
160          LD    A,#AF
170          SUB   B
180          LD    B,A
190          AND   A
200          RRA
210          SCF
220          RRA
230          AND   A
240          RRA
250          XOR   B
260          AND   #F8
270          XOR   B
280          LD    H,A
290          LD    A,C
300          RLCA
310          RLCA
320          RLCA
330          XOR   B
340          AND   #C7
350          XOR   B
```

```
380             RLCA
370             RLCA
360             LD      L,A
350             LD      A,C
400             AND     7
410             POP     BC
420             LD      A,32
430  AG2        LD      D,(HL)
440             LD      E,A
450             LD      A,D
460             OUT     (254),A
470             LD      A,E
480             LD      (HL),0
490             INC     HL
500             DEC     A
510             JP      NZ,AG2
520             CALL    SOM
530             DEC     B
540             LD      A,B
550             CP      255
560             JP      NZ,AG
570             RET
580  SOM        PUSH    BC
590             LD      B,0
600  AG4        LD      A,D
610             OUT     (254),A
620             OUT     (254),A
630             OUT     (254),A
640             DJNZ    AG4
650             POP     BC
660             RET
```

EFEITO 4

```
10 REM
20 REM  ① (MT) MARCO & TITO
30 REM
40 FOR F=31000 TO 31022: READ
A: POKE F,A: NEXT F
50 DATA 6,20,33,0,64,17,0,27,1
25,203,47,119,211,254,35,27,122,
179,32,244,16,236,201
60 FOR F=0 TO 21: PRINT AT F,0
; INK RND*6;" TESTE TESTE TESTE
TESTE TESTE ": NEXT F
70 RANDOMIZE USR 31000
```

LISTAGEM EM ASSEMBLER

```
100 ;  (MT) MARCO & TITO
110
120            ORG   31000
130 EFE4       LD    B,20
140 AG         LD    HL,16384
150            LD    DE,6912
160 ST         LD    A,(HL)
170            SRA   A
180            LD    (HL),A
190            OUT   (254),A
200            INC   HL
210            DEC   DE
220            LD    A,D
230            OR    E
240            JR    NZ,ST
250            DJNZ  AG
260            RET
```

EFEITO 5

```
10 REM
20 REM   © (MT) MARCO & TITO
30 REM
40 REM
50 FOR F=31000 TO 31026: READ
A: POKE F,A: NEXT F
60 DATA 17,0,0,243,175,211,254
,60,211,254,60,211,254,60,211,25
4,0,0,0,0,27,122,179,32,235,251,
201
70 RANDOMIZE USR 31000: PAUSE
0: GO TO 70
```

LISTAGEM EM ASSEMBLER

```
100 ; (MT) MARCO & TITO
110
120          ORG   31000
130 EFE5     LD    DE,0
140          DI
150 ST       XOR   A
160          OUT   (254),A
170          INC   A
180          OUT   (254),A
190          INC   A
200          OUT   (254),A
210          INC   A
220          OUT   (254),A
230          NOP
240          NOP
250          NOP
260          NOP
270          DEC   DE
280          LD    A,D
290          OR    E
300          JR    NZ,ST
310          EI
320          RET
```

CAPÍTULO 6

SUPER ROTINAS EM LINGUAGEM MÁQUINA

A inserção deste Capítulo, deve-se ao facto de muitos dos principiantes encontrarem uma certa dificuldade, em realizarem os seus programas em Linguagem Máquina. Devendo-se à necessidade de certas rotinas primárias, que por serem tão usadas, necessitam ser o mais reduzidas possível, e ainda a extrema rapidez de execução, que estas deverão possuir. Estas rotinas serão dificilmente feitas por um principiante, sem prática e domínio sobre Assembler.

O leitor que esteja interessado em Linguagem Máquina, deverá aprender todas as instruções da Linguagem Assembler, usando um programa próprio para introduzir Assembler, que lhe irá compilar esta linguagem em Linguagem Máquina.

As rotinas encontram-se apresentadas num programa em Basic, que irá depois de corrido, introduzir o código máquina na memória. Juntamos uma listagem de Assembler, para uma maior compreensão da rotina, por parte do leitor, como também pelo motivo, de o leitor pretender introduzir a rotina, directamente num programa Assemblador.

O leitor poderá por conseguinte, usar estas rotinas nos seus programas, com grande facilidade.

Aconselhamos ao leitor, que estiver mesmo interessado em Linguagem Máquina, que adquira um livro que explique todas as instruções Assembler, como também um bom programa, para introduzir os seus programas, e para compilar a Linguagem Assembler em Linguagem Máquina.

PLOT

Através desta rotina o leitor, poderá colocar pontos no écran. As coordenadas do ponto (256×176), a ser colocado no écran, deverão ser carregadas no registo **BC** e só depois executar a rotina.

Gravação

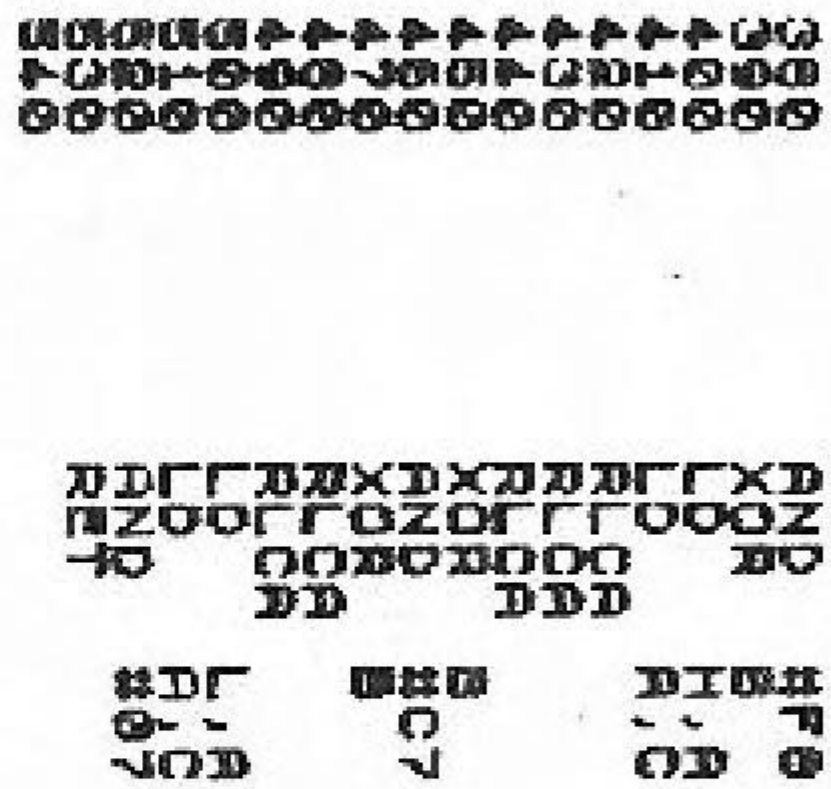
SAVE "PLOT" LINE 0

PLOT

```
10 REM
20 REM   © (MT) MARCO & TITO
30 REM
40 REM
50 FOR A=31000 TO 31045: READ
B: POKE A,B: NEXT A
80 DATA 205,40,191,71,4,69,254
,15,16,253,71,126,166,47,119,201
,62,175,144,71,167,31,56,31,167,
31,168,230,240,160,109,121,7,7,7
,168,230,169,168,7,7,111,121,230
,7,201
```

LISTAGEM EM ASSEMBLER

```
100 ;  (MT) MARCO & TITO
110
120 ;BC=COORDENADAS (175*255)
130
140          ORG  31000
150 PLOT     CALL PIXEL
160          LD   B,A
170          INC  B
180          LD   A,#FE
190 AG       RRCA
200          DJNZ AG
210          LD   B,A
220          LD   A,(HL)
230          XOR  B
240          CPL
250          LD   (HL),A
260          RET
270
280 PIXEL    LD   A,#AF
290          SUB  B
300          LD   B,A
310          AND  A
320          RRA
330          SCF
340          RRA
350          AND  A
360          RRA
370          XOR  B
```



ATTR

Para umas coordenadas carregadas no registo **BC,** o programa retorna com o respectivo endereço do **ATTRIBUTE FILE,** carregado no registo **HL.**

Desta forma o leitor poderá colocar ou retirar valores do **ATTRIBUTE FILE.**

Gravação

SAVE "ATTR" LINE 0

ATTR

```
10 REM
20 REM  © (MT) MARCO & TITO
30 REM
40 REM
50 FOR A=31000 TO 31015: READ
B: POKE A,B: NEXT A
60 DATA 120,15,15,15,71,230,22
4,169,111,120,230,3,238,88,103,2
01
```

LISTAGEM EM ASSEMBLER

```
100 ;  (MT) MARCO & TITO
110
120 ;BC=COORDENADAS (31*23)
130
140            ORG   31000
150 ATTR       LD    A,B
160            RRCA
170            RRCA
180            RRCA
190            LD    B,A
200            AND   #E0
210            XOR   C
220            LD    L,A
230            LD    A,B
240            AND   #03
250            XOR   #58
260            LD    H,A
270            RET
280
290 ;HL=ENDERECO DO ATTRIBUTE
```

PIXEL

Esta rotina permite-lhe calcular o endereço de um pixel, dado por coordenadas em alta resolução. O registo **BC** é carregado com as coordenadas (256×176). O registo **HL**, depois de ser corrido o programa, ficará com o endereço do byte, que contém o pixel, e o registo **A**, com o bit do byte correspondente ao pixel.

Gravação

SAVE "PIXEL" LINE 0

PIXEL

```
10 REM
20 REM  © (MT) MARCO & TITO
30 REM
40 REM
50 FOR A=31000 TO 31029: READ
B: POKE A,B: NEXT A
80 DATA 62,175,144,71,167,31,5
5,31,167,31,168,230,248,168,103,
121,7,7,7,168,230,199,168,7,7,11
1,121,230,7,201
```

LISTAGEM EM ASSEMBLER

```
100 ;  (MT) MARCO & TITO
110
120 ;BC=COORDENADAS (175*255)
130
140          ORG   31000
150 PIXEL    LD    A,#AF
160          SUB   B
170          LD    B,A
180          AND   A
190          RRA
200          SCF
210          RRA
220          AND   A
230          RRA
240          XOR   B
250          AND   #F8
260          XOR   B
270          LD    H,A
280          LD    A,C
290          RLCA
300          RLCA
310          RLCA
320          XOR   B
330          AND   #C7
340          XOR   B
350          RLCA
360          RLCA
370          LD    L,A
380          LD    A,C
390          AND   #07
400          RET
410
420 ;HL=ENDERECO DO BYTE
430 ;A=BIT DO BYTE
```

PRINT

Uma rotina que executa o mesmo que o Print em Basic, somente que o executa com uma maior rapidez.

Poderá assim escrever textos ou gráficos, com a surpreendente rapidez da Linguagem Máquina.

O leitor deverá carregar o registo **BC** com as coordenadas (32×24) onde deseja imprimir algo, e o registo **DE** carregado com o endereço onde se encontra o gráfico a imprimir.

Gravação

SAVE "PRINT" LINE 0

PRINT

```
   10 REM
   20 REM   © (MT) MARCO & TITO
   30 REM
   40 REM
   50 FOR A=31000 TO 31038: READ
B: POKE A,B: NEXT A
   60 DATA 175,50,62,121,245,120,
230,24,245,64,103,241,132,103,12
0,230,7,15,15,15,129,111,26,119,
58,62,121,60,254,8,19,200,50,62,
121,195,26,121,0
```

LISTAGEM EM ASSEMBLER

```
100 ;  (MT) MARCO & TITO
110
120            ORG   31000
130 PRINT      XOR   A
140            LD    (ROW),A
150 AG         PUSH  AF
160            LD    A,B
170            AND   #18
180            OR    #40
190            LD    H,A
200            POP   AF
210            ADD   A,H
220            LD    H,A
230            LD    A,B
240            AND   #07
250            RRCA
260            RRCA
270            RRCA
280            ADD   A,C
290            LD    L,A
300            LD    A,(DE)
310            LD    (HL),A
320            LD    A,(ROW)
330            INC   A
340            CP    8
350            INC   DE
360            RET   Z
370            LD    (ROW),A
380            JP    AG
390 ROW        DEFS  1
```

CLS

Da mesma forma que no Basic, esta rotina limpa todo o écran. Não deverá existir nenhuma dificuldade em trabalhar com esta rotina, pois não necessita de qualquer valor, para o seu funcionamento.

Gravação

SAVE "CLS" LINE 0

CLS

```
   10 REM
   20 REM   © (MT) MARCO & TITO
   30 REM
   40 REM
   50 FOR A=31000 TO 31028: READ
B: POKE A,B: NEXT A
   60 DATA 33,0,64,17,1,64,54,0,1
,255,23,237,176,33,0,88,17,1,88,
1,255,2,58,72,92,119,237,176,201
```

LISTAGEM EM ASSEMBLER

```
100 ;  (MT) MARCO & TITO
110
120            ORG   31000
130 CLS        LD    HL,16384
140            LD    DE,16385
150            LD    (HL),0
160            LD    BC,6143
170            LDIR
180            LD    HL,22528
190            LD    DE,22529
200            LD    BC,767
210            LD    A,(23624)
220            LD    (HL),A
230            LDIR
240            RET
```

SOM

Com esta rotina você poderá criar uma enorme variedade de efeitos sonoros, nos seus programas. A rotina é bastante fácil de manipular, tendo apenas que jogar com os valores da duração e frequência do som.

Gravação

SAVE "SOM" LINE 0

SOM

```
  10 REM
  20 REM   © (MT) MARCO & TITO
  30 REM
  40 REM
  60 FOR A=31000 TO 31022: READ
B: POKE A,B: NEXT A
  80 LET DURA=100: LET FREQ=50
  70 POKE 31007,DURA: POKE 31019
,FREQ
  80 DATA 58,72,92,31,31,31,6,16
0,14,254,37,194,34,121,238,16,23
7,121,38,255,16,244,201
  90 FOR A=0 TO 50: LET L=USR 31
000: NEXT A
```

LISTAGEM EM ASSEMBLER

```
100 ;  (MT) MARCO & TITO
110
120            ORG   31000
130 SOM        LD    A,(#5C48)
140            RRA
150            RRA
160            RRA
170            LD    B,#A0;DURACAO
180            LD    C,#FE
190 BX         DEC   H
200            JP    NZ,BX
210            XOR   #10
220            OUT   (C),A
230            LD    H,#FF;FREQ.
240            DJNZ  BX
250            RET
```

SUPERPROGRAMAS EM BASIC E LINGUAGEM MÁQUINA

Uma obra de indiscutível utilidade para todos aqueles que se interessam pela microinformática. Através de um conjunto de conselhos, programas, jogos, rotinas em Basic e linguagem máquina para aplicação nos micros ZX Spectrum, 16/48 K, ZX Spectrum +, TC 2068, o leitor tem a possibilidade, de uma forma agradável, ampliar significativamente os seus conhecimentos sobre o funcionamento deste tipo de microcomputadores.

edições SOCEDITE