

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

RUI MANUEL TITO • MARCO PAULO CARRASCO



**RUI MANUEL TITO
MARCO PAULO CARRASCO**

SUPER PROGRAMAS

**EM BASIC
E CÓDIGO MÁQUINA**

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.ª Dt. — 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 Capítulo, 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 lerá que seguir para introduzir o 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 Capítulo (Carregador de hexadecimais). Continuaremos então, as operações necessárias para terminar de introduzir um programa, que seja também constituído 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**.

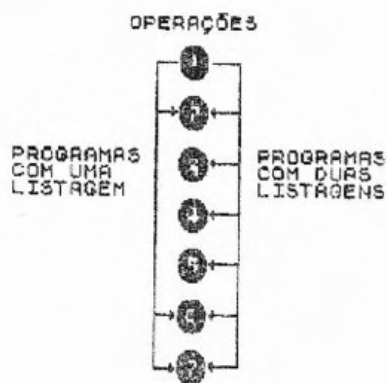
4 — Introduza agora o programa Carregador de hexadecimais, que se encontra, no 2.º Capítulo, que lhe irá servir para carregar na memória, os códigos hexadecimais, que constituem a Linguagem Máquina. Este programa encontra-se acompanhado de todas as informações necessárias, para a introdução dos códigos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

5 — Quando terminar de introduzir os códigos, deverá gravá-los, a seguir à gravação que fez do Basic, usando a segunda instrução do **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 Capítulo, 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 constituídas, 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 constituída 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
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$(2)>"9")
60 POKE 23658,8: INPUT "ENDERE
CO INICIAL ??":E: INPUT "ENDERE
CO FINAL ??":E2
70 FOR N=E TO E2 STEP 10
80 LET T=0
90 PRINT N:
100 INPUT "CÓDIGOS HEX.":H$: 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 ao 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
30 REM
40 REM
50 INPUT "ENDERECO INICIAL ??";
E: INPUT "ENDERECO FINAL ??";
E2: FOR A=E TO E2 STEP 10
60 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 **SCREENS** (imagens do écran), com muito mais rapidez.

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

É de notar que alguns dos **SCREENS** 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 retângulo 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 ecrã.

3 — Guarda o **SCREENS** na casete.

Gravação

SAVE "DESENHA," LINE 0

DESENHADOR

```

10 REM
20 REM
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$="F" THEN LET o=NO
T 0
200 IF INKEY$="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 OUE
R 0;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:
TURN
310 CIRCLE OVER 0;x,y,J
320 RETURN
330 PLOT OVER 0;x,y: DRAW OVER
0;0,ABS y1-y: DRAW OVER 0;-(ABS
x-x1),0: DRAW OVER 0;0,ABS y-y1:
DRAW OVER 0;-(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 8x8 bits, e de um cursor, você poderá construir o gráfico, e vê-lo ao mesmo tempo no seu tamanho normal. Além disso o programa indicará-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
30 REM
40 REM
50 POKE 23558,8
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;"S (=)"; AT 14,
27,0;"="
130 LET F=144
140 PRINT AT 14,5;"ABCDEFGHIJKL
MNOPQRSTU"
150 LET A=USR CHR$ F: LET H=INT
(A/255): LET L=A-(H*255)

```

```

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$="S" AND F<1
64)-(INKEY$="S" 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$="G" THEN PRINT AT
21,1: FLASH 1;"G": INPUT "NOME
7:0$ : SAVE O$CODE USR "A",158:
PRINT AT 21,1: FLASH 0;"G"
250 GO TO 150
260 DIM X(8,8)
270 FOR I=0 TO 7: PRINT AT I,12
; PAPER 0;" " : NEXT I
280 PRINT AT 21,0; PAPER 3;" 0
- PINTAR 9 - APAGAR F - FIM "
290 LET K=0: LET L=0
300 IF INKEY$="B" 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$="S" THEN LET K=K-
1: IF K=-1 THEN LET K=7: LET L=L
-1: IF L=-1 THEN LET L=7
320 PRINT AT L,K+12; PAPER 0; I
NK 8;" + " : BEEP 0,1,-40: PRINT AT
L,K+12; PAPER 0; INK 6;" "
330 IF INKEY$="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 320
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,8: 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 32655: READ
A: POKE Z,A: NEXT Z
460 RETURN
470 DATA 175,80,63,125,42,50,11
7,6,8,237,81,4,117,126,203,23,2
20,6,0,126,114,57
480 DATA 125,125,16,246,237,91,48,1
17,229,33,138,0,25,34,48,117,225,
33,63,126,264,7,200,60,50,63,
125
490 DATA 24,213,245,52,40,18,24
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

```

[illegible]

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 casete, 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

```

1000 PEEK 0; " : PRINT AT 3,0; DEF F(X) =
1010 CLS; " : PRINT E X A M I N A R D
1020 GO TO 21,0; "
1030 RANDOMIZE USR 32000
1040 LET X=PEEK FIN
1050 IF X>3 THEN GO TO 80
1060 PRINT AT 6,0;
1070 PRINT INK 4;"NOME DO PROGRA

```

```

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 200
180 IF X=2 THEN GO SUB 200
190 IF X=3 THEN GO SUB 200
200 IF X=0 THEN GO SUB 200
210 POKE FIM,255: GO SUB 420: G
O TO 30
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
a": 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)=15364 THEN PRINT "Imagem de e
cran (SCREENS)": 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
06,55,5,201
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 Capítulo, 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 Capítulo é constituído por três programas:

Duelo
Abismo
Karling 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

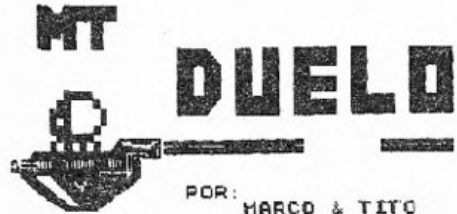
Gravação

SAVE "DUELO" LINE 0

SAVE "DUELO CODE" CODE 31000,640

GRAFICOS

A-□ B-■ C-—



DUELO

```

10 REM
20 REM
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: "
70 LET A$=""
80 PRINT AT 8,0: IN
K 7: A$; AT 7,0: A$
90 PRINT AT 10,0: INK 1: "
JOGADOR 1 " : INK 2: "JOGADOR 2
100 PRINT AT 12,5: INK 1: "Q-CIM
A": AT 12,19: INK 2: "O-CIMA"
110 PRINT AT 13,5: INK 1: "A-BAI
XO": AT 13,19: INK 2: "L-BAIXO"
120 PRINT AT 14,5: INK 1: "S-LAZ
ER": AT 14,19: INK 2: "ENTER-LAZER"
130 PRINT AT 21,0: INK 3: " PRI
MA UMA TECLA PARA INICIAR"
140 POKE 31539,2: POKE 31540,5:
POKE 31541,1: POKE 31542,28: PO
KE 31543,7: POKE 31544,1
150 FOR A=0 TO 32: LET L=USR 31
544: LET L=USR 31432: IF INKEY$<
>" THEN GO TO 170
160 NEXT A
170 IF INKEY$="" THEN GO TO 130
180 LET JOGO=31512: LET CO=3154
6: LET CO2=31548: LET MO=31544:
LET CL=31498
190 LET PO=0: LET PO2=0: LET TE
MP=2000: GO SUB 470

```

```

190 PRINT AT 10,15; INK 0; PAPE
R 4; "111"; AT 3,6; "111"; AT 17,24;
"111"
200 PRINT AT 21,0; INK 3; "111111"
210 POKE 31545,0: POKE 31541,0:
POKE CO,0: POKE CO+1,1: POKE CO
+30: POKE CO+1,15
220 PRINT AT 1,0; INK 1; " " AT
2,0; " " INK 2; AT 10,30; " " 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
3,25; 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
350 TO 210
360 FOR C=0 TO 10
370 BEEP .04,RND*-30: PRINT AT
PEEK (CO2+1),PEEK CO2; INK 3; OU
VER 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 SUPLEN
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 50

```

```

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 255,120,120,120,60,65,
120,255,48,241,31,233,157,233,17
0,253,0,0,0,255,255,0,0
510 RETURN

```

LINGUAGEM MAQUINA

```

01000 780F0F0F47E6E0A995F78 10000
01010 E8038E85857C60CD07904 10000
01020 10AF3220787570E610F6 10000
01030 40567F1846778E6070F07 10000
01040 0F816F1A7723131A773A 0007
01050 2B783CFE0808332B7513 0003
01060 C03179E04B82C7E0D1879 1144
01070 0A2E7B577E8A0016237E 841
01080 0A2E117A112000019577E 644
01090 0A2E070B7E8A00000AFC9 990
01100 0A7FC032E78CD18793A 1145
01110 0A7F87733771120001077 635
01120 0B77C03A2F787F000081F 916
01130 0B1FED78C84FC0000070 1658
01140 0B1FED78C84FC0000070 1640
01150 0B1FED78C84FC0000070 1413
01160 0B01FED78C84FC0000070 1758
01170 79001FED78C84FC0000070 1594
01180 79001FED78C84FC0000070 1716
01190 79001FED78C84FC0000070 1248
01200 05ED4320C78C84FC0000070 1156
01210 05ED4320C78C84FC0000070 1140
01220 0A097FE010C83C323978 0003
01230 ED482C7800000000ED4333 876
01240 78C000003678FE010C83C3 1183
01250 3578ED482C78000000ED433 995
01260 3578ED482C78000000ED433 1219
01270 78C000003678FE010C83C3 1183
01280 78C000003678FE010C83C3 1183
01290 78C000003678FE010C83C3 1183
01300 78C000003678FE010C83C3 1183
01310 78C000003678FE010C83C3 1183
01320 78C000003678FE010C83C3 1183
01330 78C000003678FE010C83C3 1183
01340 78C000003678FE010C83C3 1183
01350 9379ED482C78C84FC0000070 1220

```

```

31358 79ED4B2C7BED433C7B3E 1149
31370 3ACD7F79115E7BED4B3C 1117
31380 7BC328793A397BFE00C8 1171
31390 ED4B337BCD1879363F3A 1011
31400 337BFE1E28113C32337B 799
31410 8337BFE3A280CFE3F2003 877
31420 3638C9AF32337BC93639 1099
31430 18F73A387BFE00C8ED4B 1091
31440 367BCD1879363F3A367B 679
31450 FE0128113C323367B2B7E 769
31460 FE39280CFE3A363363A 837
31470 C9AF32337BC363A18F7 117
31480 FE0132387BC36323236 663
31490 7BC92128580180087EFE 988
31500 802802363F2308788120 588
31510 7BC9CD1F7AC05B7ACD7E 188
31520 7BCD847BCD847BCD7E7B 148
31530 C9071E123A812387C21E 581
31540 1200010F080000011E12 83
31550 000003C0006200E100A13 299
31560 0E2307C20AA23FFCC033 999
31570 FFFF63FE19F04E003C0 189
31580 00000000003C04600870 416
31590 0850C47043E45603FFC 134
31600 CC33FFFF63FE19F04E0 161
31610 03C000000CD957AC3987A 114
31620 CD067AC3C87ACDC67AC3 178
31630 C87A0000000000000000 322

```

ABISMO

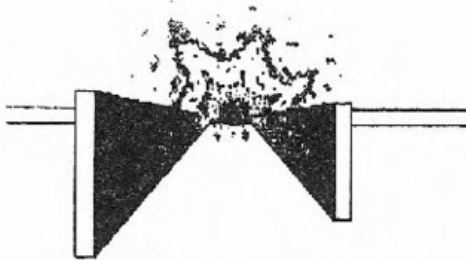
Um jogo em que se precisa ter nervos de aço para se sair vencedor, neste incrível desafio. O objectivo é encurralar o adversário, entre as linhas, que os cursores dos dois jogadores vão criando. Uma das particularidades destes cursores, é que nunca ficam imóveis, estando sempre em movimento para um dos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

ABISMO

```

10 REM
20 REM
30 REM
40 REM
50 REM
60 REM
70 REM
80 REM
90 REM
100 REM
110 REM
120 REM
130 REM
140 REM
150 REM
160 REM
170 REM
180 REM
190 REM
200 REM
210 REM
220 REM
230 REM
240 REM
250 REM
260 REM
270 REM
280 REM
290 REM
300 REM
310 REM
320 REM
330 REM
340 REM
350 REM
360 REM
370 REM
380 REM
390 REM
400 REM
410 REM
420 REM
430 REM
440 REM
450 REM
460 REM
470 REM
480 REM
490 REM
500 REM
510 REM
520 REM
530 REM
540 REM
550 REM
560 REM
570 REM
580 REM
590 REM
600 REM
610 REM
620 REM
630 REM
640 REM
650 REM
660 REM
670 REM
680 REM
690 REM
700 REM
710 REM
720 REM
730 REM
740 REM
750 REM
760 REM
770 REM
780 REM
790 REM
800 REM
810 REM
820 REM
830 REM
840 REM
850 REM
860 REM
870 REM
880 REM
890 REM
900 REM
910 REM
920 REM
930 REM
940 REM
950 REM
960 REM
970 REM
980 REM
990 REM

```

```

100 RETURN
110 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 25,20: DRAW 0,135: PLO
T 25,20: DRAW 0,135
230 PLOT 235,20: DRAW 0,135: PL
OT 235,20: 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 0,1: PLOT 0,174: DRAW 255,0
270 FOR f=8 TO 16 STEP 2: PRINT
AT f,7: INK f/2: "
NEXT f
280 PRINT AT 2,6: 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 550
370 GO SUB 470: RETURN
380 POKE 30000,10: POKE 30001,1
0: POKE 30002,4: POKE 30003,245:
POKE 30004,166: POKE 30005,4
390 PRINT AT 1,0: "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 3
UB 350
460 RETURN
470 CLS
480 PRINT AT 0,5; FLASH (p1)=p2
: "JOGADOR 1"; AT 0,15; 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-ESQUERDA
A
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=00
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 01FEFDE05B3275C07079 1441
31010 323275ED0483078C009079 1164
31020 ED433075ED0483078C009079 1304
31030 BE7901F7ED0483078C009079 1000
31040 707901F7ED0483078C009079 1000
31050 070707C09079ED0483078C009079 1000
31060 CDA47902009079ED0483078C009079 1000
31070 01ED05FC00C0FD3FECD09079 1000
31080 FE087801200F118A8007079 1000
31090 CB472000ECB4F2813CB067 1000
31100 2800CB5F280057BC003E40 1000
31110 C03E04C09079ED0483078C009079 1000
31120 CB472000ECB4F2813CB067 1000
31130 28002000C09079ED0483078C009079 1000
31140 C05C00C77947047E0710F0 1000
31150 FE087801200F118A8007079 1000
31160 C03E04C09079ED0483078C009079 1000
31170 CB472000ECB4F2813CB067 1000
31180 28002000C09079ED0483078C009079 1000
31190 070707C09079ED0483078C009079 1000
31200 070707C09079ED0483078C009079 1000
31210 FE087801200F118A8007079 1000

```

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

GRÁFICOS

A-2 B-10 C-8 D-4 E-2 F-2 G-8 H-4
I-2 J-8 K-4 L-2 M-1

KARTING

MT 2000



KARTING 2000

```

10 REM
20 REM
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 860
100 PRINT AT 21,0: FLASH 1: INK
1: PAPER 7: "PRIMA UMA T
ECLA
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: "
140 FOR A=0 TO 200: NEXT A
150 PRINT AT 11,22: INK 6: PAPE
R 0: "
160 FOR A=0 TO 200: NEXT A
170 PRINT AT 12,22: PAPER 0: IN
K 4: "
180 IF ATTR (17,16) = 6 THEN LET
V=V+1: PRINT AT PEEK 29505,PEEK
29504: INK 4: PAPER 0: OVER 1:

```



```

150 IF ATTR (15,14)=69 THEN LET
U2=U2+1: PRINT AT PEEK 29507,PE
EK 29506: INK 6; PAPER 0; OVER 1
;
2000 LET L=USR 30000
2010 LET L=USR 30155: OUT 254,1
2020 PRINT AT 15,6: INK 1; PAPER
7;"#":AT 7,25;"#":
2030 PRINT PAPER 0: INK 4;AT 11,
19:INT (U/2): INK 5;AT 10,19:INT
(U/2)
2040 IF UX=INT (U/2) THEN GO TO
7000
2050 IF UX=INT (U2/2) THEN GO TO
7000
2060 GO TO 150
2070 POKE 29512,0: POKE 29513,0
2080 POKE 29512,0: POKE 29513,0
2090 POKE 29504,16: POKE 29505,1
7
300 POKE 29506,14: POKE 29507,1
S
310 LET H=121: LET L=29
320 POKE 29506,L: POKE 29509,H
330 POKE 29510,L: POKE 29511,H
340 LET U2=0: LET U=0
350 RETURN
360 GO SUB 270: GO SUB 500
370 GO SUB 950
380 POKE 29512,1: POKE 29513,1
390 LET L=USR 30155
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
OR 1":AT 12,7:"L/ENTER":AT 10,20:
INK 5;"JOGADOR 2":AT 12,20:"D/F"
450 GO SUB 950: INPUT "NUMERO D
E VOLTAS (1-99)?" :UX
460 IF UX>99 OR UX<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:""
510 NEXT A
520 PRINT INK 5;AT 14,14;"[";AT
16,14;"["
530 PRINT INK 4;AT 16,16;"[";AT
18,16;"["

```

```

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$="XXXXXXXXXXXXXXXXXXXXXXXXXXXX"
590 PRINT AT 17,6: INK 0; PAPER
0;A$
600 PRINT AT 15,4: INK 0; PAPER
0;A$
610 LET A$="XXXXXXXXXXXXXXXXXXXXXXXXXXXX"
620 PRINT AT 7,6: INK 0; PAPER
0;A$
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 3; 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:U1":AT 10,9
; INK 5;"VOLTAS:U2:"
690 RETURN
700 FOR A=0 TO 20: BEEP .01,A+A
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 50
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 540: FOR A=USR "A"
TO USR "M"+7
820 READ S: POKE A,S
830 NEXT A

```


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!!!
Combato
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

Gravação

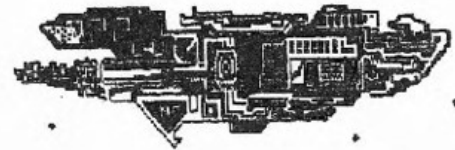
SAVE "PESQUI." LINE 0

SAVE "Pesq.Code" CODE 31000,730

GRAFICOS

A-B-C-D-E-F-G-H-I
J-K-L-M-N-O-P-Q-R

PESQUISADOR . MT



P E S Q U I S A D O R

```

1 REM
2 REM
3 REM
4 REM
5 LOAD ""CODE : POKE 23575,55
: POKE 23576,123
10 LET IN0=31510: LET CO=30000
: LET SOM=31258: LET ST=31499: L
ET NAUE=31116: LET LDIR=31382
20 LET UIX=5: LET PO=0: LET U=
1: LET FASE=1500: LET T=0: LET C
X=4
30 LET SX=40: LET SX2=60: GO 5
UB 5000
45 INK 0: LET L=USR IN0
50 GO SUB FASE: POKE SOM+7,SX:
POKE SOM+16,SX2
150 INK 7
155 POKE 30003,0: POKE CO,10: P
OKE CO+1,10: 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

```

© (MT) MARCO & TITO

55



```

00 REM
01 REM
02 REM
03 REM
04 LOAD ""CODE
05 GO SUB 840
06 PAPER 0: BORDER 0: INK 0: C
LS
00 GO SUB 780: CLS
01 LET FA=1
02 FOR F=0 TO 14: PRINT AT F,0
03 NEXT F
04 INK 7
05 LET C=0:1: LET C=5: LET PO=0:
06 LET C$=""
07 LET L=1: LET Y=1: LET D=255
08 POKE 31275,35: POKE 31255,1
09 POKE 32008,1: POKE 32001,1: P
OK 32008,31: POKE 32006,0
10 IF L THEN LET LAB=31370: PO
KE 32008,INT (LAB/255): POKE 320
07,(LAB/255-INT (LAB/255)):255
11 IF NOT L THEN LET LAB=31430
: POKE 32008,INT (LAB/255): POKE
32007,(LAB/255-INT (LAB/255))*2
56

```

```

170 LET U=USR 31324
180 IF W=1 THEN POKE 32009,D
190 GO SUB 350
200 IF NOT Q THEN GO TO 290
210 LET Q=0
220 PRINT AT 23,0;"

```

```

230 LET U=USR 31880
240 IF PEEK 32005=255 THEN GO TO
250
260 IF PEEK 32005=0 THEN GO TO
270
280 IF Y=1 THEN PRINT AT 7,15:
PAPER 7; INK 7; BRIGHT 1; "P"
290 IF PEEK 32000=15 AND PEEK 3
300=7 AND Y=1 THEN GO SUB 490
310 GO TO 230
320 LET Q=1: LET T=500
330 POKE 31880,Q:255: POKE 31276,
340 POKE 31280,Q:255: POKE 31276,
350 PRINT AT 20,0: PAPER 6; INK
360 "P" PAPER 6: BEEP DE BONUS P
370
380 LET U=USR 31880
390 POKE 32005,31
400 IF PEEK 32005=255 THEN GO TO
410
420 LET T=T-.2
430 GO TO 380
440 LET T=T+.2
450 IF T>=1 THEN STEP -1
460 FOR G=1 TO LEN A$
470 PRINT AT PEEK 32001,PEEK 32
480 INK F;A$(G)
490 PRINT AT 18,23+C; INK G; PA
500 PER B;A$(G)
510 BEEP .03,F: BEEP .03,G: NEX
520
530 T
540 NEXT F
550 LET C=C(C)=" ": LET C=C-2
560 IF C=0 THEN PAUSE 0: PRINT
570 AT 10,11: FLASH 1:"GAME OVER": P
580 AUSE 0: RUN 00
590 LET Y=1
600 GO SUB 560
610 GO TO 140
620 FOR F=PEEK 32005 TO 31
630 PRINT AT 18,F; INK 2;"■"
640 BEEP .01,F
650 NEXT F
660 POKE 32005,31
670 LET Y=0
680 RETURN

```

[illegible]

LINGUAGEM MAQUINA

[illegible]

[illegible]

66

```

K 4: PRINT AT 9,15: BRIGHT 1; IN
1300 OVER 1;
1310 IF PAS THEN GO SUB 290
1320 IF NOT PAS THEN GO SUB 430
1330 POKE PAS,0: POKE COR,15: PO
KE COR+1
1340 POKE COM,3: POKE MOR,0: POK
E POS,0
1350 RESTORE 920
1360 LET L=USR 31000
1370 IF PEEK 32000 THEN GO TO 54
0
1380 READ A: IF A=255 THEN LET A
=0: RESTORE 920
1390 IF PEEK PAS=DI THEN GO TO 6
10
1370 BEEP .04 A
1380 GO TO 230
1390 PAPER 1: INK 5: BRIGHT 1
1400 PRINT AT 0,0:
"AT 21,0:"
FOR F=0 TO 21: PRINT AT F,0:
"AT F,31:"; NEXT F
1410 PRINT AT 5,5:
"AT
5,19:"; AT 14,5:
"AT 14,13:"; AT 14,22:"
1420 PRINT AT 13,2:
"AT 13,29:"
1430 FOR F=1 TO 4
1440 PRINT AT F+2,12:
"AT F+2,
19:"; AT F+9,9:
"AT F+9,22:"
"AT F+6,5:"; AT F+6,26:
"AT
F+12,13:"; AT F+12,18:
1450 NEXT F
1460 PAPER 5: INK 3:
1470 PRINT AT 2,2:
"AT 2,28:"
"AT 20,3:"; AT 8,10:
"AT 10,
22:"; AT 20,8:
"AT 8,30:"
1480 PRINT AT 8,12:
"AT 10,19:"
1490 PAPER 0: INK 5: BRIGHT 0
1500 PRINT AT 19,12:
1510 RETURN
1520 PAPER 2: INK 5: BRIGHT 1
1530 PRINT AT 0,0:
"AT 21,0:"
FOR F=0 TO 21: PRINT AT F,0:
"AT F,31:"; NEXT F

```


LINGUAGEM MAQUINA

[illegible][illegible]

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 autômatos, 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 autômatos 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 ativado de tempo a tempo, e um Laser que utilizará para neutralizar o inimigo.

Os autômatos 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

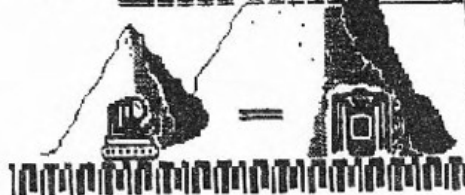
SAVE "I-CODE" CODE 31000,980

GRAFICOS

A-III

INVASÃO

CHINHA



INVASÃO CHINHA

```

10 REM
20 REM
30 REM
40 REM
50 LOAD "" CODE : POKE 23556,6
60 GO SUB 820
70 LET COR=32116: LET CB=32102
LET CO1=32114
80 LET CO3=32116
90 LET CO4=32120: LET SC=32122
100 LET ACT=32109: LET COR=321
11 LET ACT=32109
120 LET VAR=32113
130 LET BONECO=31096: LET EF=31
941 LET FIREX=31277
140 LET INT=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=0: 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+1,5
  
```

© (MT) MARCO & TITO

```

230 POKE CO3,0: 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 U
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
330 LET PO=PO+(PEEK SC)+10: POK
E SC,0
340 PRINT AT 1,10; INVERSE 1; PO
IF PO=1000 THEN POKE COA,5
+PO*8
350 IF PO>3000 THEN POKE COA,8
+PO*8
360 IF PO<1000 THEN POKE COA,5
370 IF PO>5000 THEN POKE COA,6
+PO*8
380 LET L=USR BONECO: IF PEEK U
AR=0 THEN GO TO 760
390 LET L=USR FIREX
400 LET T=T+1
410 GO TO 270
420 FOR A=5 TO 20: PRINT AT A,0
430 SLEEP .01,-A*COS 0: PRINT IN
K 0; AT A,0; "
440 NEXT A: RETURN
450 PAPER 0: BORDER 0
460 INK 0: CLS
470 INK 7
480 PRINT AT 5,5; INVERSE 1; BR
IGHT 1; INK 5; "I N U A 3 A O"
490 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$="N" THEN LET L=US
R
570 OUT 254,15
580 IF INKEY$<>"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 10,19: 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 10,-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,
20; "5"
750 RETURN
760 LET UID=UID-1: IF UID=-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 10,8;
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

[illegible]

003	1	E	E	E	D	4	B	7	2	7	D	E	D	4	3	6	4	7	0	C	D	15	5	2	3	
003	2	7	8	7	D	E	D	4	B	6	6	4	7	D	E	D	4	3	7	2	7	D	15	1	3	2
003	3	4	D	B	7	4	7	D	E	D	4	3	6	4	7	D	E	D	4	3	7	D	15	4	6	7
003	4	7	B	D	E	D	4	B	6	4	7	D	E	D	4	3	7	4	7	0	C	D	15	1	3	2
003	5	7	B	7	6	7	D	E	D	4	3	6	4	7	D	E	D	4	3	7	D	15	1	3	2	
003	6	E	D	B	6	6	4	7	D	E	D	4	3	7	6	7	D	E	D	4	B	15	1	3	2	
003	7	7	8	7	D	E	D	4	3	6	4	7	D	E	D	4	3	7	6	7	D	15	1	3	2	
003	8	4	B	6	4	7	D	E	D	4	3	7	6	7	D	E	D	4	3	7	D	15	1	3	2	
003	9	7	D	F	E	3	B	8	2	8	8	1	1	2	0	0	9	1	9	2	7	5	9	9	9	
003	10	9	3	7	D	E	3	9	2	1	0	8	1	1	2	0	7	7	3	8	1	1	2	7	5	
003	11	7	0	F	E	1	3	3	8	2	8	8	1	1	2	0	0	9	1	9	2	7	5	9	9	
003	12	4	E	D	4	3	6	4	7	0	C	D	E	D	4	3	6	4	7	0	C	D	15	1	3	2
003	13	4	B	6	4	7	0	C	D	E	D	4	3	6	4	7	0	C	D	E	D	15	1	3	2	
003	14	D	D	4	B	6	4	7	0	C	D	E	D	4	3	6	4	7	0	C	D	15	1	3	2	
003	15	4	B	6	4	7	0	C	D	E	D	4	3	6	4	7	0	C	D	E	D	15	1	3	2	
003	16	4	B	6	4	7	0	C	D	E	D	4	3	6	4	7	0	C	D	E	D	15	1	3	2	
003	17	4	B	6	4	7	0	C	D	E	D	4	3	6	4	7	0	C	D	E	D	15	1	3	2	
003	18	4	B	6	4	7	0	C	D	E	D	4	3	6	4	7	0	C	D	E	D	15	1	3	2	
003	19	4	B	6	4	7	0	C	D	E	D	4	3	6	4	7	0	C	D	E	D	15	1	3	2	
003	20	4	B	6	4	7	0	C	D	E	D	4	3	6	4	7	0	C	D	E	D	15	1	3	2	
003	21	4	B	6	4	7	0	C	D	E	D	4	3	6	4	7	0	C	D	E	D	15	1	3	2	
003	22	4	B	6	4	7	0	C	D	E	D	4	3	6	4	7	0	C	D	E	D	15	1	3	2	
003	23	4	B	6	4	7	0	C	D	E	D	4	3	6	4	7	0	C	D	E	D	15	1	3	2	
003	24	4	B	6	4	7	0	C	D	E	D	4	3	6	4	7	0	C	D	E	D	15	1	3	2	
003	25	4	B	6	4	7	0	C	D	E	D	4	3	6	4	7	0	C	D	E	D	15	1	3	2	
003	26	4	B	6	4	7	0	C	D	E	D	4	3	6	4	7	0	C	D	E	D	15	1	3	2	

MONSTROS 48K

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

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

A bruxa possui três monstros de estimação que não simpatizam muito consigo, e que tudo farão para lhe dificultar o seu trabalho. Mas não é assim tão má a bruxa, contando com a antipatia dos seus monstros, ofereceu-lhe uma arma enfeitada, 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-



MONSTROS 48K

```

10 REM
20 SCREEN 0
30 BORDER 0: PAPER 0
40 INK 7
50 CLS: LOAD ""CODE
60 POKE 23676,129: POKE 23676,
161
90 LET EF=41440: LET SX=41409:
GO SUB 260
100 LET XSC=0: LET FASE=1
110 LET A0L=45015: LET C0=45003
120 LET DIR1=45017: LET C01=450
20 LET C02=45022
130 LET C03=45024: LET SC=45026
140 LET DIR2=45018
150 LET DIR3=45019
160 LET BX=4: LET BY=1
170 LET PO=0: LET UID=5
180 LET VAR=45015: LET ACT=4500
8
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 ACT+0: POKE DIR1,4
290 POKE DIR2,0: POKE DIR3,2
300 POKE VAR,1
310 RESTORE 750: READ Z
320 GO SUB 770: GO SUB 660
330 PRINT AT PEEK (CB+1),PEEK C
340 INK 7;"",AT PEEK (CB+1)+1,P
350 CB;" "
360 LET L=USR BONECO: IF PEEK U
370 THEN GO TO 630
380 LET L=USR FIREX
390 POKE CO1,CX+84: LET CX=CX+1
400 IF CX=8 THEN LET CX=4
410 LET L=USR INI
420 BEEP .02,Z: READ Z
430 IF Z=255 THEN RESTORE 750:
440 READ Z
450 LET L=USR FIREX
460 LET PO=PO+(PEEK SC)*10: POK
470 SC,0
480 PRINT INK 6,AT 19,13,PC;"
490 IF PO-XSC>1000 THEN LET FASE=
490 FASE+1: LET XSC=PO: GO SUB 490:
500 GO TO 220
510 LET L=USR BONECO: IF PEEK U
520 THEN GO TO 630
530 LET L=USR FIREX
540 LET X=PEEK (CB+1): LET Y=PE
550 CB
560 IF ATTA (X,Y-2)=5 OR ATTA (
570 X,Y+3)=5 THEN GO SUB 660
580 GO TO 340
590 FOR A=0 TO 20: BEEP .01,-A
600 BEEP .01,RND*10+50: BEEP .0
610 A
620 NEXT A: RETURN
630 FOR A=0 TO 10: LET L=USR SX
640 NEXT A
650 CLS
660 PLOT 16,103: DRAW 224,0
670 DRAW 0,-31: DRAW -224,0
680 DRAW 0,31
690 PRINT AT 10,3: INK 6: INVER
700 SE 1;"
710 PRINT AT 11,3: INK 5: INVER
720 SE 1;"

```

```

600 PRINT AT 10,5: INK 6: INVER
610 SE 1;"FIM DE JOGO";AT 11,5: INK
620 SE 1;"PONTOS"
630 GO SUB 490: PAUSE 0
640 GO TO 00
650 LET VID=VID-1: IF VID=0 THE
660 N GO SUB 490: GO TO 650
670 PRINT INK 4,AT 10,13,VID;"
680 "
690 GO SUB 490: GO TO 220
700 LET PO=PO+40: LET A=INT (RN
710 D*5)+1
720 BEEP .01,50: PRINT AT BX,BY
730 ;" ",AT BX+1,BY;" "
740 IF A=1 THEN LET BX=4: LET B
750 Y=1
760 IF A=2 THEN LET BX=10: LET
770 BY=1
780 IF A=3 THEN LET BX=16: LET
790 BY=1
800 IF A=4 THEN LET BX=1: LET B
810 Y=30
820 IF A=5 THEN LET BX=7: LET B
830 Y=30
840 IF A=6 THEN LET BX=13: LET
850 XY=30
860 BEEP .01,30: PRINT AT BX,BY
870 ; INK 6;"A";AT BX+1,BY;" "
880 RETURN
890 DATA 14,13,8,10,13,10,8,10,
900 13,13,8,10,13,10,8,8,6,6,8,10,13
910 ,13,8,10,13,255
920 INK 0: CLS
930 INK 7
940 IF FASE=1 THEN LET A$="BONECO"
950 IF FASE=2 THEN LET A$="FIREX"
960 IF FASE=3 THEN LET A$="INICIALIZAÇÃO"
970 IF FASE=4 THEN LET A$="FIM DE JOGO"
980 IF FASE=5 THEN LET A$="PONTOS"
990 IF FASE=6 THEN LET A$="PONTOS"
1000 ET FASE=0
1010 LET B$=A$( TO 1)
1020 FOR A=0 TO 21 STEP 3: PRINT
1030 AT A,0: INK 5;A$
1040 NEXT A
1050 FOR A=0 TO 21: PRINT AT A,0
1060 ; INK 5;B$;AT A,31;B$;AT A,2;B$;
1070 AT A,29;B$
1080 NEXT A

```

```

890 PRINT AT 3,3; INK 0; " " AT
6,12; "AT 6,15; "AT 9,7; "
12,12; "AT 9,15; "AT 9,27; "AT
12,12; "
900 PRINT AT 15,3; INK 0; " "A
T 15,16; "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$; AT
9,17; B$; AT 13,16; B$; AT 14,16; B$;
AT 16,15; B$; AT 17,15; B$
940 PRINT AT 19,5; INK 5; "PONTO
5; "INK 4; AT 20,5; "VIDAS: "; U
10
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 5; "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) - DESPARO"
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$="": PRINT AT 10,0; I
NK 5; A$; AT 2,0; A$
1040 POKE SX+7,70: LET L=USR SX
1050 IF INKEY$<>" THEN RETURN
1060 LET L=USR 6F
1070 GO TO 1040

```

LINGUAGEM MAQUINA

```

400000 3E80F070F67E00000000F70 1000
400010 8E803000000000000000000 1000
400020 13AF3270000000000000000 1000
400030 23131A77000000000000000 1000
400040 C8327000000000000000000 1000
400050 C0400000000000000000000 1000
400060 19772B77000000000000000 1000
400070 C02FA63E000000000000000 1000

```

```

400080 903ACBAFB8CA0A9C3AD02 1621
400090 AF0C847C2DA9CC0B47C22DA 1711
400100 0C0833E865C0C290380F04 859
400110 0AC0B3F8000000000000000 999
400120 0AF0C320A900ED48C8AF0404 1439
400130 3E05C000000000000000000 1527
400140 0C08AF73300000000000000 1649
400150 00033200000000000000000 1805
400160 00022000000000000000000 1805
400170 00033200000000000000000 1805
400180 00033200000000000000000 1805
400190 00033200000000000000000 1805
400200 00033200000000000000000 1805
400210 00033200000000000000000 1805
400220 00033200000000000000000 1805
400230 00033200000000000000000 1805
400240 00033200000000000000000 1805
400250 00033200000000000000000 1805
400260 00033200000000000000000 1805
400270 00033200000000000000000 1805
400280 00033200000000000000000 1805
400290 00033200000000000000000 1805
400300 00033200000000000000000 1805
400310 00033200000000000000000 1805
400320 00033200000000000000000 1805
400330 00033200000000000000000 1805
400340 00033200000000000000000 1805
400350 00033200000000000000000 1805
400360 00033200000000000000000 1805
400370 00033200000000000000000 1805
400380 00033200000000000000000 1805
400390 00033200000000000000000 1805
400400 00033200000000000000000 1805
400410 00033200000000000000000 1805
400420 00033200000000000000000 1805
400430 00033200000000000000000 1805
400440 00033200000000000000000 1805
400450 00033200000000000000000 1805
400460 00033200000000000000000 1805
400470 00033200000000000000000 1805
400480 00033200000000000000000 1805
400490 00033200000000000000000 1805
400500 00033200000000000000000 1805
400510 00033200000000000000000 1805
400520 00033200000000000000000 1805
400530 00033200000000000000000 1805
400540 00033200000000000000000 1805
400550 00033200000000000000000 1805
400560 00033200000000000000000 1805
400570 00033200000000000000000 1805
400580 00033200000000000000000 1805
400590 00033200000000000000000 1805
400600 00033200000000000000000 1805

```


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

8-A

PERIGO!!!

```

100 REM
200 REM
300 REM
400 REM
500 LOAD ""CODE : POKE 23658,0
600 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 21,6: INK 7;"TEMPO
130 AT 21,19;"PONTOS"
140 FOR F=0 TO 19: PRINT AT F,0
150 NEXT F
160 POKE 31007,20
170 PRINT AT 20,1: INK 5; PAPER
180 PRINT AT 0,0: PAPER 1: INK
190 FLASH 1;"*** PRIMA (C)-PARA
200 COMEÇAR ***"
210 LET L=USR 31031
220 IF INKEY$<>"C" THEN GO TO 1
230 PRINT AT 0,0;"*****"
240 POKE 32000,AND*31: POKE 320
250 01,0: POKE 32002,3: POKE 32004,0
260 POKE 32004,6: POKE 32005,10
270 INK 0
280 LET L=USR 31087
290 IF INKEY$="1" THEN LET X=PE
300 EK 32000
310 LET TE=TE-.1: IF TE<=0 THEN
320 GO TO 540
330 PRINT AT 21,14:INT TE;" "
340 LET PO=PO+1: PRINT AT 21,26
350 ;PO
360 IF ATTR (17,PEEK 32004)=7 T
370 HEN GO SUB 200
380 GO TO 220
390 LET G=G+1
400 LET X=PEEK 32004
410 POKE 31007,50: LET L=USR 31
420 POKE 31007,20
430 PRINT AT 17,X: PAPER 0: INK
440 0: INK 0;"0"

```



```

330 PRINT AT 21,0; INK G; PAPER
G+1;" "
340 IF G=6 THEN GO TO 370
350 POKE 32002,3: POKE 32001,0:
POKE 32000,AND#31
360 RETURN
370 LET J=1: FOR F=23200 TO 232
00: 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 10: PRINT AT F,0
" : LET L=USR 31000: NEXT F
420 IF G=-1 THEN GO TO 400
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+4,0: 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 21,25;PO
480 NEXT F
490 IF TE<58 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
L=0: GO TO 120
540 FOR F=15 TO 255 STEP 10
550 LET L=USR 31007
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 254,15: 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
ECLA"
640 PRINT AT 14,2:"L-ESQUERDA"
650 PRINT AT 16,2:"ENTER"-DIRE
ITA"
660 PRINT AT 19,0: PAPER 2:"
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 7,0:"
770 LET L=USR 31005
780 IF INKEYS="1" 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 3A485C1F1F1F08140EFE 600
31010 252008FE10ED79263C2D 630
31020 00F4EE10ED7926E6410ED 1206
31030 C921205611527A013C08 650
31040 C506081A073607362023 000

```


COMBATE

```

10 REM
20 REM
30 REM
40 REM
50 LOAD
4: POKE 23675,100
60 LET SC=310001: LET VAL=31000
: LET AND=310000
70 LET VEN=310002: LET FLAG=310
80 LET SCOM=310003: LET FLAG2=31
90 LET MTC=310005: LET CO=31730
100 LET PX=31013: LET EF2=31054
110 LET EF=31010: LET TEC=31007
120 LET SCV1=0: LET SCV2=0
130 POKE MTC,0: POKE SC,0
140 POKE SCOM,0: POKE VEN,0
150 POKE FLAG,0: POKE FLAG2,0
160 POKE EF,0: INK 0
170 CLS: BORDER 7
180 GO SUB 610
190 GO SUB 250: GO SUB 910
200 LET AT=USR: TEC: POKE VEN,0
210 FOR A=0 TO 20: LET L=USR EF
220 NEXT A
230 IF AT=1 THEN GO TO 600
240 IF AT=2 THEN GO TO 750
250 GO SUB 800
260 LET AS=" "
270 PRINT AT 10,0: INK 1,AS: IN
K 7,AT 15,0,AS: PAPER 4: INK 4,A
T 14,0,AS: AT 13,0,AS
280 LET L=USR EF
290 GO SUB 440: FOR A=15 TO 10
300 PRINT INK 2,AT A,0," ";AT A
310 NEXT A
320 FOR A=1 TO 6: PRINT AT A,1:
INK 1:" "
330 NEXT A
340 LET AS="COMBATE": GO
SUB 600
350 POKE CO,0: POKE CO+1,3
360 LET L=USR PX
370 LET D=0: LET D2=7
380 FOR A=0 TO 31: PRINT AT 0,A
: INK 0,AT 7,A: INK 02," "
390 LET D2=D2-1: LET D=D+1
400 IF D=0 THEN LET D=0
410 IF D2=-1 THEN LET D2=7
420 NEXT A
430 LET D=7: FOR A=0 TO 7

```

© (MT) MARCO & TITO

```

420 PRINT AT A,31: INK A," ";AT
A,0: INK D:" "; LET D=D-1
430 NEXT A: RETURN
440 LET AS="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 AS="
JOGADOR 2 00": POKE CO,15: PO
KE CO+1,22: GO SUB 650: LET L=US
R PX: RETURN
480 LET AS="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 AND+1+1;"
520 NEXT A
530 LET AS="1-UM JOGADOR": GO S
UB 650
540 POKE CO,5: POKE CO+1,10
550 LET L=USR PX
560 LET AS="2-DOIS JOGADORES":
GO SUB 650
570 POKE CO,5: POKE CO+1,12
580 LET L=USR PX
590 PRINT AT 15,5: INK 1,"TECLA
S": AT 15,5: INK 3: JOGADOR 1": AT
17,5: "D": (F): AT 15,15: JOGA
DOR 2": AT 17,15: (L)(ENTER)"
600 LET AS="P O T M A R C O
& "
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
AS: POKE D,CODE AS(A)
660 LET D=D+1: NEXT A
670 POKE D,255: RETURN
680 LET SCV1=SCV1+1: GO SUB 940
690 IF SCV1=3 THEN GO TO 710
700 GO SUB 800: GO SUB 800
710 LET AS="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

```


ZIG ZAG

```

100 REM
101 REM
102 REM
103 REM
104 REM
105 REM
106 REM
107 REM
108 REM
109 REM
110 REM
111 REM
112 REM
113 REM
114 REM
115 REM
116 REM
117 REM
118 REM
119 REM
120 REM
121 REM
122 REM
123 REM
124 REM
125 REM
126 REM
127 REM
128 REM
129 REM
130 REM
131 REM
132 REM
133 REM
134 REM
135 REM
136 REM
137 REM
138 REM
139 REM
140 REM
141 REM
142 REM
143 REM
144 REM
145 REM
146 REM
147 REM
148 REM
149 REM
150 REM
151 REM
152 REM
153 REM
154 REM
155 REM
156 REM
157 REM
158 REM
159 REM
160 REM
161 REM
162 REM
163 REM
164 REM
165 REM
166 REM
167 REM
168 REM
169 REM
170 REM
171 REM
172 REM
173 REM
174 REM
175 REM
176 REM
177 REM
178 REM
179 REM
180 REM
181 REM
182 REM
183 REM
184 REM
185 REM
186 REM
187 REM
188 REM
189 REM
190 REM
191 REM
192 REM
193 REM
194 REM
195 REM
196 REM
197 REM
198 REM
199 REM
200 REM
201 REM
202 REM
203 REM
204 REM
205 REM
206 REM
207 REM
208 REM
209 REM
210 REM
211 REM
212 REM
213 REM
214 REM
215 REM
216 REM
217 REM
218 REM
219 REM
220 REM
221 REM
222 REM
223 REM
224 REM
225 REM
226 REM
227 REM
228 REM
229 REM
230 REM
231 REM
232 REM
233 REM
234 REM
235 REM
236 REM
237 REM
238 REM
239 REM
240 REM
241 REM
242 REM
243 REM
244 REM
245 REM
246 REM
247 REM
248 REM
249 REM
250 REM
251 REM
252 REM
253 REM
254 REM
255 REM
256 REM
257 REM
258 REM
259 REM
260 REM
261 REM
262 REM
263 REM
264 REM
265 REM
266 REM
267 REM
268 REM
269 REM
270 REM
271 REM
272 REM
273 REM
274 REM
275 REM
276 REM
277 REM
278 REM
279 REM
280 REM
281 REM
282 REM
283 REM
284 REM
285 REM
286 REM
287 REM
288 REM
289 REM
290 REM
291 REM
292 REM
293 REM
294 REM
295 REM
296 REM
297 REM
298 REM
299 REM
300 REM
301 REM
302 REM
303 REM
304 REM
305 REM
306 REM
307 REM
308 REM
309 REM
310 REM
311 REM
312 REM
313 REM
314 REM
315 REM
316 REM
317 REM
318 REM
319 REM
320 REM
321 REM
322 REM
323 REM
324 REM
325 REM
326 REM
327 REM
328 REM
329 REM
330 REM
331 REM
332 REM
333 REM
334 REM
335 REM
336 REM
337 REM
338 REM
339 REM
340 REM
341 REM
342 REM
343 REM
344 REM
345 REM
346 REM
347 REM
348 REM
349 REM
350 REM
351 REM
352 REM
353 REM
354 REM
355 REM
356 REM
357 REM
358 REM
359 REM
360 REM
361 REM
362 REM
363 REM
364 REM
365 REM
366 REM
367 REM
368 REM
369 REM
370 REM
371 REM
372 REM
373 REM
374 REM
375 REM
376 REM
377 REM
378 REM
379 REM
380 REM
381 REM
382 REM
383 REM
384 REM
385 REM
386 REM
387 REM
388 REM
389 REM
390 REM
391 REM
392 REM
393 REM
394 REM
395 REM
396 REM
397 REM
398 REM
399 REM
400 REM
401 REM
402 REM
403 REM
404 REM
405 REM
406 REM
407 REM
408 REM
409 REM
410 REM
411 REM
412 REM
413 REM
414 REM
415 REM
416 REM
417 REM
418 REM
419 REM
420 REM
421 REM
422 REM
423 REM
424 REM
425 REM
426 REM
427 REM
428 REM
429 REM
430 REM
431 REM
432 REM
433 REM
434 REM
435 REM
436 REM
437 REM
438 REM
439 REM
440 REM
441 REM
442 REM
443 REM
444 REM
445 REM
446 REM
447 REM
448 REM
449 REM
450 REM
451 REM
452 REM
453 REM
454 REM
455 REM
456 REM
457 REM
458 REM
459 REM
460 REM
461 REM
462 REM
463 REM
464 REM
465 REM
466 REM
467 REM
468 REM
469 REM
470 REM
471 REM
472 REM
473 REM
474 REM
475 REM
476 REM
477 REM
478 REM
479 REM
480 REM
481 REM
482 REM
483 REM
484 REM
485 REM
486 REM
487 REM
488 REM
489 REM
490 REM
491 REM
492 REM
493 REM
494 REM
495 REM
496 REM
497 REM
498 REM
499 REM
500 REM
501 REM
502 REM
503 REM
504 REM
505 REM
506 REM
507 REM
508 REM
509 REM
510 REM
511 REM
512 REM
513 REM
514 REM
515 REM
516 REM
517 REM
518 REM
519 REM
520 REM
521 REM
522 REM
523 REM
524 REM
525 REM
526 REM
527 REM
528 REM
529 REM
530 REM
531 REM
532 REM
533 REM
534 REM
535 REM
536 REM
537 REM
538 REM
539 REM
540 REM
541 REM
542 REM
543 REM
544 REM
545 REM
546 REM
547 REM
548 REM
549 REM
550 REM
551 REM
552 REM
553 REM
554 REM
555 REM
556 REM
557 REM
558 REM
559 REM
560 REM
561 REM
562 REM
563 REM
564 REM
565 REM
566 REM
567 REM
568 REM
569 REM
570 REM
571 REM
572 REM
573 REM
574 REM
575 REM
576 REM
577 REM
578 REM
579 REM
580 REM
581 REM
582 REM
583 REM
584 REM
585 REM
586 REM
587 REM
588 REM
589 REM
590 REM
591 REM
592 REM
593 REM
594 REM
595 REM
596 REM
597 REM
598 REM
599 REM
600 REM
601 REM
602 REM
603 REM
604 REM
605 REM
606 REM
607 REM
608 REM
609 REM
610 REM
611 REM
612 REM
613 REM
614 REM
615 REM
616 REM
617 REM
618 REM
619 REM
620 REM
621 REM
622 REM
623 REM
624 REM
625 REM
626 REM
627 REM
628 REM
629 REM
630 REM
631 REM
632 REM
633 REM
634 REM
635 REM
636 REM
637 REM
638 REM
639 REM
640 REM
641 REM
642 REM
643 REM
644 REM
645 REM
646 REM
647 REM
648 REM
649 REM
650 REM
651 REM
652 REM
653 REM
654 REM
655 REM
656 REM
657 REM
658 REM
659 REM
660 REM
661 REM
662 REM
663 REM
664 REM
665 REM
666 REM
667 REM
668 REM
669 REM
670 REM
671 REM
672 REM
673 REM
674 REM
675 REM
676 REM
677 REM
678 REM
679 REM
680 REM
681 REM
682 REM
683 REM
684 REM
685 REM
686 REM
687 REM
688 REM
689 REM
690 REM
691 REM
692 REM
693 REM
694 REM
695 REM
696 REM
697 REM
698 REM
699 REM
700 REM
701 REM
702 REM
703 REM
704 REM
705 REM
706 REM
707 REM
708 REM
709 REM
710 REM
711 REM
712 REM
713 REM
714 REM
715 REM
716 REM
717 REM
718 REM
719 REM
720 REM
721 REM
722 REM
723 REM
724 REM
725 REM
726 REM
727 REM
728 REM
729 REM
730 REM
731 REM
732 REM
733 REM
734 REM
735 REM
736 REM
737 REM
738 REM
739 REM
740 REM
741 REM
742 REM
743 REM
744 REM
745 REM
746 REM
747 REM
748 REM
749 REM
750 REM
751 REM
752 REM
753 REM
754 REM
755 REM
756 REM
757 REM
758 REM
759 REM
760 REM
761 REM
762 REM
763 REM
764 REM
765 REM
766 REM
767 REM
768 REM
769 REM
770 REM
771 REM
772 REM
773 REM
774 REM
775 REM
776 REM
777 REM
778 REM
779 REM
780 REM
781 REM
782 REM
783 REM
784 REM
785 REM
786 REM
787 REM
788 REM
789 REM
790 REM
791 REM
792 REM
793 REM
794 REM
795 REM
796 REM
797 REM
798 REM
799 REM
800 REM
801 REM
802 REM
803 REM
804 REM
805 REM
806 REM
807 REM
808 REM
809 REM
810 REM
811 REM
812 REM
813 REM
814 REM
815 REM
816 REM
817 REM
818 REM
819 REM
820 REM
821 REM
822 REM
823 REM
824 REM
825 REM
826 REM
827 REM
828 REM
829 REM
830 REM
831 REM
832 REM
833 REM
834 REM
835 REM
836 REM
837 REM
838 REM
839 REM
840 REM
841 REM
842 REM
843 REM
844 REM
845 REM
846 REM
847 REM
848 REM
849 REM
850 REM
851 REM
852 REM
853 REM
854 REM
855 REM
856 REM
857 REM
858 REM
859 REM
860 REM
861 REM
862 REM
863 REM
864 REM
865 REM
866 REM
867 REM
868 REM
869 REM
870 REM
871 REM
872 REM
873 REM
874 REM
875 REM
876 REM
877 REM
878 REM
879 REM
880 REM
881 REM
882 REM
883 REM
884 REM
885 REM
886 REM
887 REM
888 REM
889 REM
890 REM
891 REM
892 REM
893 REM
894 REM
895 REM
896 REM
897 REM
898 REM
899 REM
900 REM
901 REM
902 REM
903 REM
904 REM
905 REM
906 REM
907 REM
908 REM
909 REM
910 REM
911 REM
912 REM
913 REM
914 REM
915 REM
916 REM
917 REM
918 REM
919 REM
920 REM
921 REM
922 REM
923 REM
924 REM
925 REM
926 REM
927 REM
928 REM
929 REM
930 REM
931 REM
932 REM
933 REM
934 REM
935 REM
936 REM
937 REM
938 REM
939 REM
940 REM
941 REM
942 REM
943 REM
944 REM
945 REM
946 REM
947 REM
948 REM
949 REM
950 REM
951 REM
952 REM
953 REM
954 REM
955 REM
956 REM
957 REM
958 REM
959 REM
960 REM
961 REM
962 REM
963 REM
964 REM
965 REM
966 REM
967 REM
968 REM
969 REM
970 REM
971 REM
972 REM
973 REM
974 REM
975 REM
976 REM
977 REM
978 REM
979 REM
980 REM
981 REM
982 REM
983 REM
984 REM
985 REM
986 REM
987 REM
988 REM
989 REM
990 REM
991 REM
992 REM
993 REM
994 REM
995 REM
996 REM
997 REM
998 REM
999 REM
1000 REM

```

```

280 PRINT AT 17,3; INK 7; PAPER
1: TECLAS : P > PARA DIREITA
290 LET U=1: PRINT AT 1,0; FLAS
H 1: 0001 - PARA INICIAR O JOGO
300 FOR F=3 TO 15: PRINT AT F,0
INK U; OVER 1;
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 I=USR "A" TO USR "F"+7:
READ A: POKE F,A: NEXT F
450 RETURN
460 DATA 50,125,255,255,255,255
,125,50
470 DATA 24,50,125,255,255,125,
50,24
480 DATA 0,50,125,125,125,125,6
0,0
490 DATA 0,24,50,125,125,60,24,
0
500 DATA 0,0,24,50,50,24,0,0
510 DATA 0,0,0,24,24,0,0,0

```

LINGUAGEM MAQUINA

```

31000 ED450070DCDCE79368011 1040
31010 00550070D01C00000000 1050
31020 01FED7ED780B470C000000 1060
31030 1007300047047000100000 1070
31040 7000000000000000000000 1080
31050 3000000000000000000000 1090
31060 0000000000000000000000 1100
31070 4B0007D0DCDCE797E7E7E7 1110

```

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 Assembly, 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
30 REM
40 PAPER 0: BORDER 0: INK 7: C
L3
50 FOR F=USR "A" TO USR "A"+7
STEP 2: POKE F,55: POKE F+1,170:
NEXT F
60 DATA 33,0,55,1,0,2,125,60,1
12,35,11,120,177,321,247,201
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
130 EFER1 ORG 30000
140 LO HL,22528
150 ST LO BC,512
160 INC HL
170 LD HL,A
180 INC HL
190 DEC HL
200 LO BC
210 OR BC,B
220 ROR
230 RET

```

EFEITO 2

```

10 REM
20 REM
30 REM
40 FOR F=31000 TO 31024: READ
A: POKE F,A: NEXT F
50 DATA 213,1,255,0,33,60,17
13,237,95,110,211,255,4,58,27,12
170,32,248,15,237,255,1,58,27,12
60 FOR F=0 TO 31: 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 EFER1 ORG 31000
140 LO HL,22528
150 ST LO BC,512
160 INC HL
170 LD HL,A
180 INC HL
190 DEC HL
200 LO BC
210 OR BC,B
220 ROR
230 RET

```

[illegible]

```

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

```

[illegible]

SOM
AG4

[illegible]

EFREITO 4

[illegible]

LISTAGEM EM ASSEMBLER

[illegible]

EFEITO S

```

10 REM
20 REM
30 REM
40 REM
50 FOR F=31000 TO 31026: READ
A:POKER,F,A: NEXT F
60 DATA 17,0,0,1243,175,211,254
70 DATA 11,254,69,211,254,50,211,254
80 DATA 0,0,0,27,122,178,32,235,251,
201
70 RANDOMIZE USR 31000: PAUSE
8: GO TO 70

```

LISTAGEM EM ASSEMBLER

```

100 ; (MT) MARCO & TITO
110
120
130
140
150
160
170
180
190
200
210
220
230
240
250
260
270
280
290
300
310
320
330
340
350
360
370
380
390
400
410
420
430
440
450
460
470
480
490
500
510
520
530
540
550
560
570
580
590
600
610
620
630
640
650
660
670
680
690
700
710
720
730
740
750
760
770
780
790
800
810
820
830
840
850
860
870
880
890
900
910
920
930
940
950
960
970
980
990

```

CAPÍTULO 6

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 Assembler.

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 (256x176), a ser colocado no écran, deverão ser carregadas no registo **BC** e só depois executar a rotina.

Gravação

SAVE "PLOT:" LINE 0

PLOT

[illegible]

LISTAGEM EM ASSEMBLER

[illegible]

#7, 8, C7
#10, 11, C7

to CONDOCO RD DD
RDZOFZOFZOFZOFZ
RDZOFZOFZOFZOFZ

RDXRDXRDXRDXRDX

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

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

Gravação

SAVE "ATTR" LINE 0

```

10 REM
20 REM
30 REM
40 REM
50 FOR
60 POKE A,B: NEXT A
70 DATA 1,100,236,3,236,66,100,236
80

```

```

1000 ; (MT) MARCO & TITO
1001 ;BC=COORDENADAS (31*23)
1002
1003      ORG 31000
1004 ATTR    DB 0,000
1005
1006      RMO 0,0
1007      RMO 0,0
1008      RMO 0,0
1009      RMO 0,0
1010      RMO 0,0
1011      RMO 0,0
1012      RMO 0,0
1013      RMO 0,0
1014      RMO 0,0
1015      RMO 0,0
1016      RMO 0,0
1017      RMO 0,0
1018      RMO 0,0
1019      RMO 0,0
1020      RMO 0,0
1021      RMO 0,0
1022      RMO 0,0
1023      RMO 0,0
1024      RMO 0,0
1025      RMO 0,0
1026      RMO 0,0
1027      RMO 0,0
1028      RMO 0,0
1029      RMO 0,0
1030      RMO 0,0
1031      RMO 0,0
1032      RMO 0,0
1033      RMO 0,0
1034      RMO 0,0
1035      RMO 0,0
1036      RMO 0,0
1037      RMO 0,0
1038      RMO 0,0
1039      RMO 0,0
1040      RMO 0,0
1041      RMO 0,0
1042      RMO 0,0
1043      RMO 0,0
1044      RMO 0,0
1045      RMO 0,0
1046      RMO 0,0
1047      RMO 0,0
1048      RMO 0,0
1049      RMO 0,0
1050      RMO 0,0
1051      RMO 0,0
1052      RMO 0,0
1053      RMO 0,0
1054      RMO 0,0
1055      RMO 0,0
1056      RMO 0,0
1057      RMO 0,0
1058      RMO 0,0
1059      RMO 0,0
1060      RMO 0,0
1061      RMO 0,0
1062      RMO 0,0
1063      RMO 0,0
1064      RMO 0,0
1065      RMO 0,0
1066      RMO 0,0
1067      RMO 0,0
1068      RMO 0,0
1069      RMO 0,0
1070      RMO 0,0
1071      RMO 0,0
1072      RMO 0,0
1073      RMO 0,0
1074      RMO 0,0
1075      RMO 0,0
1076      RMO 0,0
1077      RMO 0,0
1078      RMO 0,0
1079      RMO 0,0
1080      RMO 0,0
1081      RMO 0,0
1082      RMO 0,0
1083      RMO 0,0
1084      RMO 0,0
1085      RMO 0,0
1086      RMO 0,0
1087      RMO 0,0
1088      RMO 0,0
1089      RMO 0,0
1090      RMO 0,0
1091      RMO 0,0
1092      RMO 0,0
1093      RMO 0,0
1094      RMO 0,0
1095      RMO 0,0
1096      RMO 0,0
1097      RMO 0,0
1098      RMO 0,0
1099      RMO 0,0
1100      RET
1101
1102 ;HL=ENDEREÇO DO ATTRIBUTE

```

Esta rotina permite-lhe calcular o endereço de um pixel, dado por coordenadas em alta resolução. O registo **BC** é carregado com as coordenadas (256x176). 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.

SAVE "PIXEL" LINE 0

(MT) MARCO & TITO

```

100 ; (MT) MARCO & TITO
110 ; SC=COORDENADOR (175+255)
120
130
140
150
160
170
180
190
200
210
220
230
240
250
260
270
280
290
300
310
320
330
340
350
360
370
380
390
400
410
420
430
440
450
460
470
480
490
500
510
520
530
540
550
560
570
580
590
600
610
620
630
640
650
660
670
680
690
700
710
720
730
740
750
760
770
780
790
800
810
820
830
840
850
860
870
880
890
900
910
920
930
940
950
960
970
980
990
1000
1010
1020
1030
1040
1050
1060
1070
1080
1090
1100
1110
1120
1130
1140
1150
1160
1170
1180
1190
1200
1210
1220
1230
1240
1250
1260
1270
1280
1290
1300
1310
1320
1330
1340
1350
1360
1370
1380
1390
1400
1410
1420
1430
1440
1450
1460
1470
1480
1490
1500
1510
1520
1530
1540
1550
1560
1570
1580
1590
1600
1610
1620
1630
1640
1650
1660
1670
1680
1690
1700
1710
1720
1730
1740
1750
1760
1770
1780
1790
1800
1810
1820
1830
1840
1850
1860
1870
1880
1890
1900
1910
1920
1930
1940
1950
1960
1970
1980
1990
2000
2010
2020
2030
2040
2050
2060
2070
2080
2090
2100
2110
2120
2130
2140
2150
2160
2170
2180
2190
2200
2210
2220
2230
2240
2250
2260
2270
2280
2290
2300
2310
2320
2330
2340
2350
2360
2370
2380
2390
2400
2410
2420
2430
2440
2450
2460
2470
2480
2490
2500
2510
2520
2530
2540
2550
2560
2570
2580
2590
2600
2610
2620
2630
2640
2650
2660
2670
2680
2690
2700
2710
2720
2730
2740
2750
2760
2770
2780
2790
2800
2810
2820
2830
2840
2850
2860
2870
2880
2890
2900
2910
2920
2930
2940
2950
2960
2970
2980
2990
3000
3010
3020
3030
3040
3050
3060
3070
3080
3090
3100
3110
3120
3130
3140
3150
3160
3170
3180
3190
3200
3210
3220
3230
3240
3250
3260
3270
3280
3290
3300
3310
3320
3330
3340
3350
3360
3370
3380
3390
3400
3410
3420
3430
3440
3450
3460
3470
3480
3490
3500
3510
3520
3530
3540
3550
3560
3570
3580
3590
3600
3610
3620
3630
3640
3650
3660
3670
3680
3690
3700
3710
3720
3730
3740
3750
3760
3770
3780
3790
3800
3810
3820
3830
3840
3850
3860
3870
3880
3890
3900
3910
3920
3930
3940
3950
3960
3970
3980
3990
4000
4010
4020
4030
4040
4050
4060
4070
4080
4090
4100
4110
4120
4130
4140
4150
4160
4170
4180
4190
4200
4210
4220
4230
4240
4250
4260
4270
4280
4290
4300
4310
4320
4330
4340
4350
4360
4370
4380
4390
4400
4410
4420
4430
4440
4450
4460
4470
4480
4490
4500
4510
4520
4530
4540
4550
4560
4570
4580
4590
4600
4610
4620
4630
4640
4650
4660
4670
4680
4690
4700
4710
4720
4730
4740
4750
4760
4770
4780
4790
4800
4810
4820
4830
4840
4850
4860
4870
4880
4890
4900
4910
4920
4930
4940
4950
4960
4970
4980
4990
5000
5010
5020
5030
5040
5050
5060
5070
5080
5090
5100
5110
5120
5130
5140
5150
5160
5170
5180
5190
5200
5210
5220
5230
5240
5250
5260
5270
5280
5290
5300
5310
5320
5330
5340
5350
5360
5370
5380
5390
5400
5410
5420
5430
5440
5450
5460
5470
5480
5490
5500
5510
5520
5530
5540
5550
5560
5570
5580
5590
5600
5610
5620
5630
5640
5650
5660
5670
5680
5690
5700
5710
5720
5730
5740
5750
5760
5770
5780
5790
5800
5810
5820
5830
5840
5850
5860
5870
5880
5890
5900
5910
5920
5930
5940
5950
5960
5970
5980
5990
6000
6010
6020
6030
6040
6050
6060
6070
6080
6090
6100
6110
6120
6130
6140
6150
6160
6170
6180
6190
6200
6210
6220
6230
6240
6250
6260
6270
6280
6290
6300
6310
6320
6330
6340
6350
6360
6370
6380
6390
6400
6410
6420
6430
6440
6450
6460
6470
6480
6490
6500
6510
6520
6530
6540
6550
6560
6570
6580
6590
6600
6610
6620
6630
6640
6650
6660
6670
6680
6690
6700
6710
6720
6730
6740
6750
6760
6770
6780
6790
6800
6810
6820
6830
6840
6850
6860
6870
6880
6890
6900
6910
6920
6930
6940
6950
6960
6970
6980
6990
7000
7010
7020
7030
7040
7050
7060
7070
7080
7090
7100
7110
7120
7130
7140
7150
7160
7170
7180
7190
7200
7210
7220
7230
7240
7250
7260
7270
7280
7290
7300
7310
7320
7330
7340
7350
7360
7370
7380
7390
7400
7410
7420
7430
7440
7450
7460
7470
7480
7490
7500
7510
7520
7530
7540
7550
7560
7570
7580
7590
7600
7610
7620
7630
7640
7650
7660
7670
7680
7690
7700
7710
7720
7730
7740
7750
7760
7770
7780
7790
7800
7810
7820
7830
7840
7850
7860
7870
7880
7890
7900
7910
7920
7930
7940
7950
7960
7970
7980
7990
8000
8010
8020
8030
8040
8050
8060
8070
8080
8090
8100
8110
8120
8130
8140
8150
8160
8170
8180
8190
8200
8210
8220
8230
8240
8250
8260
8270
8280
8290
8300
8310
8320
8330
8340
8350
8360
8370
8380
8390
8400
84
```

SAVE "PRINT" LINE 0

PRINT

```

10 REM
20 REM
30 REM
40 REM
50 FOR A=31000 TO 31039: READ
B: POKE A,B: NEXT
60 DATA 175,50,6,121,245,120,
70 300,4,245,54,103,122,113,245,120,
80 300,7,15,15,153,122,113,245,119,
90 300,121,6,15,15,153,122,113,245,
101,120,26,121,6

```

LISTAGEM EM ASSEMBLER

```

100 ; (MT) MARCO & TITO
110
120 PRINT
130
140
150
160
170
180
190
200
210
220
230
240
250
260
270
280
290
300
310
320
330
340
350
360
370
380
390
400
410
420
430
440
450
460
470
480
490
500
510
520
530
540
550
560
570
580
590
600
610
620
630
640
650
660
670
680
690
700
710
720
730
740
750
760
770
780
790
800
810
820
830
840
850
860
870
880
890
900
910
920
930
940
950
960
970
980
990

```

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

© (MT) MARCO & TITO

LISTAGEM EM ASSEMBLER

```

100 ; (MT) MARCO & TITO
110
120
130
140
150 CLS
160
170 ORG 31000
180 HL, 1000
190 DE, 1000
200 (HL), 1000
210 BC, 6143
220
230 IR
240
250 HL, 1000
260 DE, 1000
270 BC, 7007
280 (HL), 1000
290 (HL), 1000
300
310 RET IR
320

```

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
30 REM
40 REM
50 FOR A=31000 TO 31022: READ
B: POKE A, B: NEXT A
60 LET DURA=100: LET FREQ=50
70 POKE 31007, DURA: POKE 31019
, FREQ
80 DATA 58, 72, 90, 31, 31, 31, 5, 16
9, 14, 25, 4, 37, 19, 4, 34, 13, 1, 23, 5, 10, 23
7, 12, 1, 38, 5, 15, 24, 4, 38, 1
90 FOR A=0 TO 50: LET L=USR 31
000: NEXT A

```

LISTAGEM EM ASSEMBLER

```

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

```