

Tim Hartnell, já nosso conhecido de obras traduzidas nesta colecção, *Como Programar o seu ZX Spectrum* e a recente *ZX Spectrum na Educação*, continua a surpreender-nos com a qualidade dos seus trabalhos! Desta vez ele reúne 49 programas de jogos, da autoria dos melhores programadores do Reino Unido. Os programas, muitos dos quais utilizam subrotinas de código-máquina, incluem jogos verdadeiramente emocionantes entre os quais destacamos: «Protector», «Bombardeiro», «Pássaros», «Helicóptero», «Corridas», «Zombies», «Space Treck», «Lógicos», «Fantasmas», «Biorritmo» e muitos, muitos outros! Leitor, não perca mais tempo! Vá para casa fazer «explodir» o seu Spectrum!



EDITORIAL PRESENÇA

49 Jogos Explosivos para o ZX Spectrum

182

TIM HARTNELL
DAVID PERRY, GRAHAM CHARLTON, NEIL PELLINACCI e MALCOLM YOUNG

49 JOGOS EXPLOSIVOS

PARA O ZX SPECTRUM



CULTURA E TEMPOS LIVRES

1. ABC do Xadrez, *Petar Trifunovich e Sava Vukovich*
4. ABC do Bridge, *Pierre Jais e H. Lahana*
5. Guia Prático de Fotografia, *W. D. Emanuel*
6. ABC do Judo, *E. J. Harrison*
7. Como Fazer Cinema, *Paul Petzold*
8. Bridge Moderno, *Pierre Jais e H. Lahana*
9. Fotografia — Técnicas e Truques I, *Edwin Smith*
10. ABC dos Estilos, da Arquitectura ao Mobiliário, *A. Auzel*
11. Fotografia — Técnicas e Truques II, *Edwin Smith*
12. A Pesca Submarina, *António Ribera*
13. Teoria dos Finais de Partida, *Yuri Averbach*
14. Aprenda Rádio, *B. Fighiera*
15. Guia do Cão, *Louise Laliberté-Robert e Jean-Pierre, Robert*
16. ABC do Aquário, *Anthony Evans*
17. Iniciação à Electricidade e Electrónica, *Fernand Hurt*
18. Os Transistores, *Fernand Hurt*
19. Karatê I, *Albrecht Pflüger*
20. Iniciação ao Radiocomando dos Modelos Reduzidos, *C. Péricon*
21. Construa o seu Receptor, *B. Fighiera*
22. Montagens Electrónicas, *B. Fighiera*
23. O Berbequim Eléctrico, *Villy Dreier*
24. Cactos, *J. Nilauz Jensen*
25. Iniciação à Alta Fidelidade, *Peter Turner*
26. O Aquário de Água Doce, *Paulo de Oliveira*
27. ABC do Tênis, *Fonseca Vaz*
28. Karatê II, *Albrecht Pflüger*
29. ABC da Criação de Canários, *Curt Af Enehjelm*
30. Ginástica Feminina, *Sonja Helmer Jensen*
31. Cartomancia, *Rhea Koch*
32. Calculadoras Electrónicas de Bolso, *E. Dam Ravn*
33. O Pastor Alemão, *Gilles Legrand*
34. Xadrez — Teoria do Meio Jogo I, *Bondarevsky*
35. Manual do Super 8 — I, *Myron A. Matzkin*
36. ABC da Criação de Periquitos, *Cyril H. Rogers*
37. O Livro dos Gatos, *Barbel Gerber e Horst Bietfeld*
38. Manual do Super 8 — II, *Myron A. Matzkin*
39. ABC do Mergulho Desportivo, *Walter Mattes*
40. Circuitos Integrados/Aplicações Práticas, *F. Bergtoi I*
41. A Apicultura, *H. R. C. Riches*
42. ABC do Cultivo das Plantas, *H. G. Witham Fogg*
43. ABC da Criação de Pombos, *Karl R. Dahl*
44. Construção de Caixas Acústicas de Alta Fidelidade, *R. Brault*
45. Raças de Canários, *Klaus Speicher*
46. Jogos de Cartas, *Graciano Dolma*
47. Cocker Spaniels, *H. S. Lloyd*
48. ABC da Caça, *Fabian Abril*
49. Aprenda Televisão, *Gordon J. King*
50. Iniciação à Pesca, *Juan Nadal*
51. Basquetebol, *Marius Norregard*
52. Cães de Caça, *Santiago Pons*
53. Aprenda Electrónica, *T. L. Squires e C. M. Deason*
54. A Avicultura, *Jim Worthington*
55. A Produção de Coelho, *P. Surdeau e R. Henaff*
56. ABC dos Computadores, *T. F. Fry*
57. Natação para Crianças, *John Idorn*
58. O Boxer, *Anni Mortensen*
59. Voleibol, *Ole Hansen e Per-Göran Persson*
60. Iniciação à Vela, *Donald Law*
61. ABC da Filatelia, *Jacqueline Caurat*
62. A Pesca à Beira-Mar, *J. M. Böelle e D. Doyen*
63. Enxerto de Árvores de Fruto, *Alejo Rigau*
64. A Cultura do Morangueiro, *Luis Alsina Grau*
65. Emissores-Receptores (Walkies-Talkies), *P. Duranton*
66. Iniciação à Fotoelectrónica, *Heinz Richter*
67. Doces e Conservas de Fruta, *Robin Howe*
68. A Criação de Hamsters, *C. F. Snow*
69. A Criação de Porcos, *Roy Genders*
70. Calendário do Horticultor, *Luis Alsina Grau*
71. Jogos Electrónicos, *F. G. Raver*
72. Cultivo de Cogumelos e Trufas, *Alejo Rigau*
73. Aprenda Televisão a Cores, *Gordon J. King*
74. Gravação em Fita Magnética, *Ian R. Sinclair*
75. Poda de Árvores e Arbustos, *Roy Genders*
76. Como Treinar o Seu Cão, *E. Fitch Daglish*
77. Instrumentos de Medida e Verificação, *Heinrich Stöckle*
78. A Criação de Caracóis, *Mattias Josa*
79. Rádio — Fundamentos e Técnicas, *Gordon J. King*
80. Como Fazer Gelados, *Sylvie Thiebault*
81. Iniciação à Jardinagem, *Noel Claraso*
82. A Congelação dos Alimentos, *Suzanne Lapointe*
83. Windsurf — Prancha à Vela, *Ernstfried Prade*
84. Raças de Cães, *O. Hasselfeldt*
85. Rummy e Canasta, *Claus D. Grupp*
86. A Encadernação, *Annie Persus*
87. Aprenda Electricidade, *Heinz Richter*
88. Taxidermia, Embalsamento de Aves e Mamíferos, *Harry Hjortaa a*
89. Jogging — Correr para Manter a Forma, *Werner Sonntag*
90. ABC da Cozinha Chinesa, *Sonja Richmond*
91. Jogos T.V., *C. Tavernier*
92. Amplificadores de Som, *Richard Zierl*
93. O Livro do Poker, *Claus D. Grupp*
94. Aprenda a Desenhar, *Rose-Marie de Prémont e Nicole Philippi*
95. O Minitranspolim na Escola, *Sonja Helmer Jensen e Klaus Danø*
96. Jogos de Luzes e Efeitos Sonoros para Guitarras, *B. Fighiera*
97. O Cultivo do Tomate, *Louis N. Flawn*
98. Pilhas Solares, *F. Juster*
99. A Criação Doméstica de Coelho, *C. F. Snow*
100. Iniciação ao Futebol, *Wieland Männele e Heinz Arnold*
101. Horóscopos Chineses, *Feorg Haddenbach*
102. Guia Prático de Marcenaria, *Charles H. Hayward*
103. Andebol I, *Fritz e Peter Hatig*
104. Dispositivos Anti-Roubo, *H. Schreiber*
105. Perus, Pintadas e Codornizes, *Jérôme Sauze*
106. Crepes — Doces e Salgados, *Florence Arzel*
107. Aperitivos e Entradas, *Myrette Tiano*
108. Tênis de Mesa, *Leslie Woollard*
109. Aprenda Surf, *R. Abbott e M. Baker*
110. Futebol — Técnica e Tática, *Kurt Lavall*
111. A Vaca Leiteira, *Colin T. Whittemore*
112. O Cubo Mágico, *Josef Trajber*
113. O Perdigueiro Português, *José M. Correia*
114. Pizzas e Massas à Italiana, *Marieanne Ränk*
115. O Cubo Para Quem Já o Faz, *Josef Trajber*
116. A Pirâmide Mágica, A Torre, o Barril do Diabo, *M. Mrowka-W. J. Weber*
117. Gansos e Patos, *Marie Mourhe*
118. Iniciação ao Kung-Fu, *A. P. Harrington*
119. Electrónica e Fotografia, *Hanns-Peter Siebert*
120. O Livro da Fortuna, *Douglas Hill*
121. Construção de um Alimentador de Corrente, *Waldemar Baitinger*
122. Hóquei em Patins, *Francisco Velasco*
123. Técnicas de Tiro, *Anton Kovacic*
124. Aprenda a Tricotar, *Uta Mix*
125. ABC da Patinagem, *Christa-Maria e Richard Kerler*
126. A Pesca e os seus Segredos, *Armand Deschamps*
127. O Osciloscópio, *R. Rateau*
128. Guia Prático da Banda do Cidadão, *T. M. Normand*
129. Sumos e Batidos, *Manfred Donderski*
130. Introdução à Programação de Microcomputadores, *Peter C. Sanderson*
131. Aprenda Crochê, *Uta Mix*
132. ABC do Microprocessador, *P. Mchlusson*
133. Guia Prático de Basic, *Roger Hunt*
134. Introdução à Electrónica Digital, *Ian R. Sinclair*
135. ABC do Vídeo, *David K. Matthewson*
136. Fotografia em Movimento, *Don Morley*
137. Guia de Cobol, *Ray Welland*
138. Fotografia a Pequena Distância, *Sidney F. Ray*
139. Guia Moderno da Canaricultura, *Manuel Gonçalves*

140. Minieletrónica para Amadores, *Heinz Richter*
141. ABC da Programação de Computadores, *John Shelley*
142. Taito — O Futuro Pelas Cartas, *Edwin J. Nigg*
143. ABC da Equitação, *Dorothy Johnson*
144. Como Programar o Seu ZX 81, *Patrick Guille*
145. 100 Avarias TV e a Maneira Prática de as Detectar, *P. Duranton*
146. ABC da Horticultura, *Louis Giordano*
147. Basic para Microcomputadores, *A. P. Stephenson*
148. Como Programar o seu ZX Spectrum, *Tim Hartnell e Dilwyn Jones*
149. Iniciação aos Motores Diesel, *Dávid S. Maclean*
150. 60 Jogos para o ZX Spectrum, *David Harwood*
151. As Linhas da Mão, *Rose Hubert*
152. Cozinha Italiana, *Rotraud Degner*
153. Manual do ZX Spectrum, *R. J. Simpson e T. J. Terrel*
154. Z80 Assembler para o ZX Spectrum — Iniciação ao Código de Máquina.
John Paulo Fragoso
155. Aeróbica, *H. Schulz*
156. ABC do Atletismo, *Denis Watts*
157. 26 Programas Basic para Microcomputadores, *Derrick Daines*
158. Aprenda Pascal no seu Microcomputador, *Jeremy Ruston*
159. Guia Moderno da Suinicultura, *Colin Whittemore*
160. O Bar em Sua Casa — 888 Cocktails, *Aladar von Wesendonk*
161. Código de Máquina para Principiantes, *James Walsh*
162. Código de Máquina para Programadores Avançados, *Paul Holmes*
163. ABC da Fruticultura, *Henri Gosselin*
164. ABC da Canoagem, *Alan Hyde*
165. Guia de Fortran, *Philly Ridler*
166. Manual da Secretária, *Philippa Ramage*
167. ABC das Antenas, *Gordon J. King*
168. Programar Aventuras no Seu Computador, *Andrew Nelson*
169. Guia do Sinclair QL, *Boris Allan*
170. Novas Aventuras no seu ZX Spectrum, *Peter Shaw e James Mortleman*
171. O Computador no Escritório, *John Shelley*
172. Sobremesas, *Fred Timber*
173. Rádio — Do Circuito Oscilante ao Receptor de Ondas Curtas, *Richard Zierl*
174. ABC das Aberturas, *V. N. Panov*
175. A Construção de Pequenos Transformadores, *M. Dorian e F. Juster*
176. Gui de Pascal, *David Watt*
177. O ZX Spectrum na Educação, *Tim Hartnell, Christine Johnson e David Valentine*
178. Iniciação ao Snooker, *John Pulman*
179. Circuitos com Diacs, Tiristores e Triacs, *Fritz Bergtold*
180. O Jogo do Mahjong, *Ursula Eschenbach*
181. Cultura Hidropónica, Culturas sem solo, *Laura Fronty*
182. 49 Jogos Explosivos para o ZX Spectrum, *Tim Hartnell*

TIM HARTNELL

David Perry, Graham Charlton
Neil Pellinacci e Malcolm Young

49 JOGOS EXPLOSIVOS PARA O ZX SPECTRUM

EDITORIAL  PRESENÇA

PREFÁCIO

Os livros tão compactos como este não saem do Spectrum sem um certo investimento de tempo e esforço.

E gastou-se de facto bastante tempo e esforço tentando apresentar uma colecção de jogos para o Spectrum tão boa quanto possível.

Partindo do trabalho de alguns dos melhores programadores jovens do Reino Unido, juntei uma série de programas para o Spectrum que merecem ser considerados «de qualidade». Os principais contribuidores foram Malcolm Young, David Perry, Graham Charlton e Neil Pellinacci, cujo trabalho preenche a maior parte do livro.

Outras pessoas a quem pedi que produzissem obras para esta colecção foram Scott Vincent, Raymond Blake, Neal Cavalier-Smith, Paul Toland, Andrew Sweetland, Martin Jones, Graham White, Tim Rogers, Damian Steele, Neville Predebon e Michael Briggs. Também eles merecem os nossos parabéns pela qualidade dos programas que nos apresentaram.

É o momento de fazer «explodir» o seu Spectrum...

Tim Hartnell
Londres, 1983

Titulo original:

49 Explosives Games for the ZX Spectrum

© Copyright by Hartnell, 1983

Tradução de Conceição Jardim e Miguel Nogueira

Capa de António Marques

Reservados todos os direitos para a língua portuguesa à

EDITORIAL PRESENÇA, LDA:

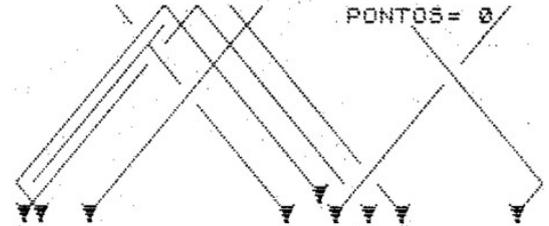
Rua Augusto Gil, 35-A — 1000 LISBOA

JOGOS DE PERÍCIA

PROTECTOR

O leitor já não precisa de gastar o seu dinheiro em jogos de perícia... «Protector» dá-lhe a possibilidade de salvar o mundo (ou pelo menos algumas cidades) sem sair de casa, como demonstram os exemplos de execução apresentados em seguida.

Os comandos do jogo são «M», «Q», «A» e «O». É o momento de realizar a sua missão e salvar o mundo...



```
10 POKE 23658,8: REM obter mai  
uscúlas  
20 CLS  
30 FOR F=1 TO 49  
40 BORDER F/7  
50 BEEP 0,05,F  
60 PRINT AT 5,10; INK 0; PAPER
```



```

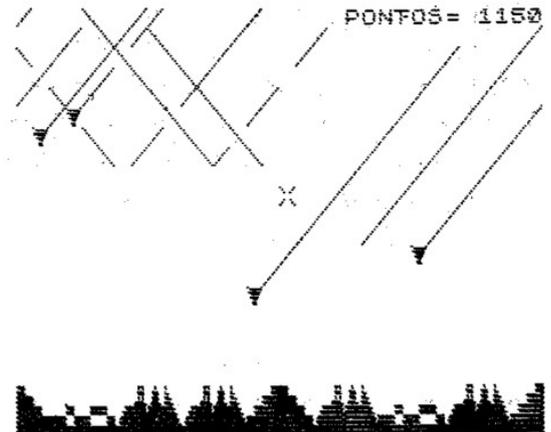
640 NEXT X
650 IF U=0 THEN GO TO 850
660 PRINT AT 10,0: FOR X=1 TO
U: PRINT AT 10,4*(X-1);R$;AT 11,
4*(X-1);U$: NEXT X
670 LET T=U*100: PRINT AT 11,(U
)+4;"= BONUS ";T: LET SC=SC+T
680 GO SUB 970
690 GO SUB 920: LET U=U+1: IF U
>15 THEN LET U=15
700 LET D=U: GO TO 290
710 IF L(B,1)<18 THEN GO TO 350
720 IF SCREEN$(L(B,1),L(B,2))<
>" THEN GO TO 300
730 LET Z=L(B,2)
740 IF Z>1 AND Z<6 THEN LET Z$(
1,1)=" ": LET Z$(1,2)=E$: GO
TO 840
750 IF Z>5 AND Z<10 THEN LET Z$(
2,1)=" ": LET Z$(2,2)=E$: GO
TO 840
760 IF Z>9 AND Z<14 THEN LET Z$(
3,1)=" ": LET Z$(3,2)=E$: GO
TO 840
770 IF Z>17 AND Z<22 THEN LET Z
$(4,1)=" ": LET Z$(4,2)=E$: G
O TO 840
780 IF Z>21 AND Z<25 THEN LET Z
$(5,1)=" ": LET Z$(5,2)=E$: G
O TO 840
790 IF Z>25 AND Z<30 THEN LET Z
$(6,1)=" ": LET Z$(6,2)=E$: G
O TO 840
800 LET L(B,1)=L(D,1): LET L(B,
2)=L(D,2): LET N(B)=N(D): LET X$(
B)=X$(D): LET D=D-1
810 IF D>0 THEN GO TO 350
820 IF U=0 THEN GO TO 850
830 GO TO 600
840 BEEP .15,2: GO SUB 920: GO
SUB 950: GO TO 800
850 CLS : FLASH 1: FOR X=1 TO 2
2: PRINT "=====
=====": NEXT X: PRINT AT 10
,9;"JOGO TERMINADO": FLASH 0
860 PRINT
870 FLASH 0: IF SC>=HS THEN PRI
NT "
CONSEGUIU A MELHOR PONTUACA
O!! ": LET HS=SC: INPUT "INDIQUE
O NOME: ";N$
880 PRINT : PRINT PAPER 0; INK

```

```

7;"PONTUACAO MAXIMA: ";N$
890 PRINT : PRINT FLASH 1;"CARR
EGUE EM B PARA JOGAR DE NOVO"
900 IF INKEY$<>"B" THEN BEEP .3
,1: GO TO 900
910 LET SC=0: GO TO 20
920 LET Q$=" "+Z$(1,1)+Z$(2,1)
+Z$(3,1)+" "+Z$(4,1)+Z$(5,1)+
Z$(6,1)+" "
930 LET Q$=Q$+" "+Z$(1,2)+Z$(2
,2)+Z$(3,2)+" "+Z$(4,2)+Z$(5,
2)+Z$(6,2)+" "
940 RETURN
950 PRINT AT 19,0; INK 4;Q$( TO
2); INK 3;Q$(3 TO 14); INK 4;Q$(
15 TO 18); INK 3;Q$(19 TO 30);
INK 4;Q$(31 TO 34); INK 3;Q$(35
TO 46); INK 4;Q$(47 TO 58); INK
3;Q$(51 TO 62); INK 4;Q$(63 TO )
960 RETURN
970 PRINT AT 20,13;"NIVEL ";H
980 FOR F=1 TO 200: NEXT F: CLS
990 RETURN

```

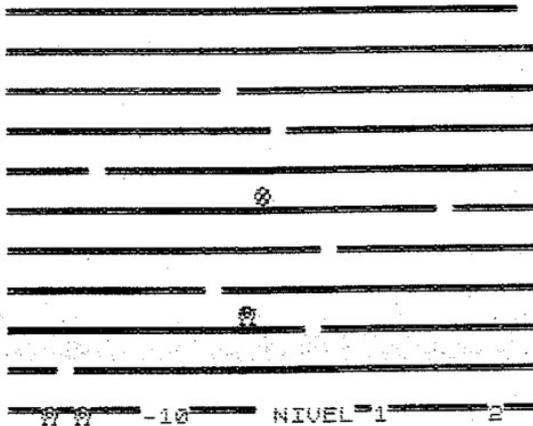


Paul Toland apresenta aqui um excitante jogo de pericia, que obrigará o leitor a saltar diversos andares através de orificios em movimento. A medida que se atinge o topo passa-se ao «screen» seguinte. Existem cinco «screens» diferentes, e o leitor poderá observar nas imagens que se seguem os objectivos a cumprir em cada um deles:

HI-RIZE

NIVEL 1

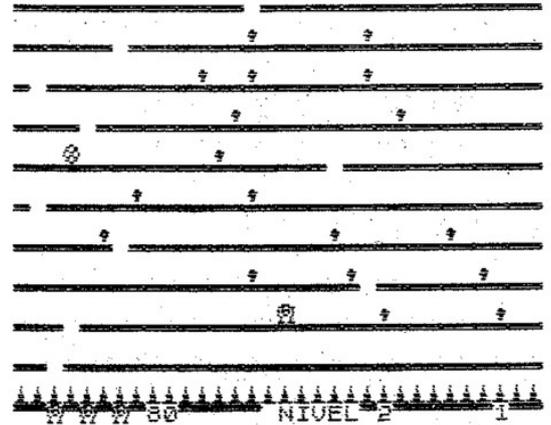
O seu objectivo consiste em atingir o topo a fim de passar ao nivel seguinte. Sobe saltando pelos intervalos das linhas superiores e nao caindo pelos da linha onde se encontra. Deve tambem evitar o barril que se move nas linhas, senao perde uma vida. Mova-se para a esquerda e a direita usando as teclas S e D, e salte com a tecla I



HI-RIZE

NIVEL 2

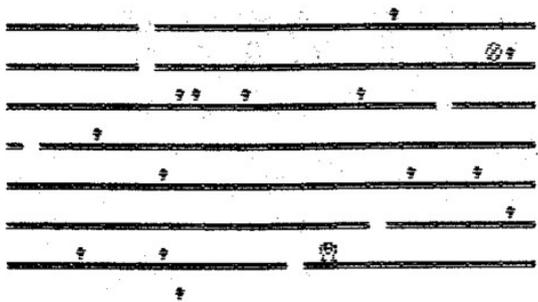
A partir deste nivel encontrara pontos vermelhos e ganhara pontuacao passando sobre eles. Neste nivel nao pode voltar para a linha na base.



HI-RIZE

NIVEL 3

Enquanto sobe, as linhas por baixo de si desaparecem, pelo que perdera uma vida se cair.



W W W 170

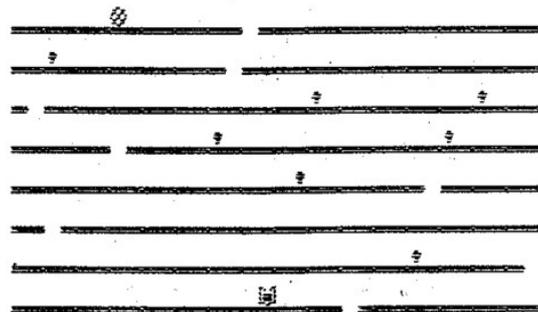
NIVEL 3

4

HI-RIZE

NIVEL 4

Tudo vai ser mais rapido daqui em diante.



W W W 370

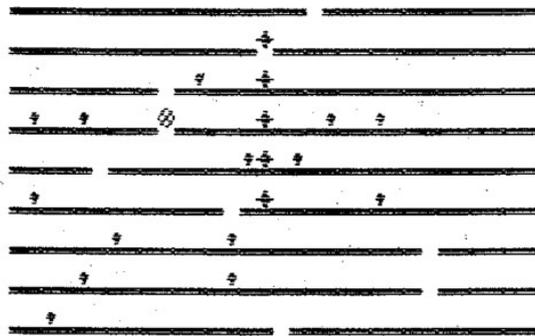
NIVEL 4

8

HI-RIZE

NIVEL 5

O toque dos picos na parte superior e' letal...



W W 270

NIVEL 5

11

Apresentamos em seguida a listagem deste programa:

```

1 REM SEM ALTO © P. TOLAND
80 GO SUB 800: GO SUB 2500
90 LET TOT=TOT+1: LET UN=UN+2:
GO SUB 920
100 FOR I=H TO L: PRINT AT I*2-
1,H(I,1);CHR$ 144: LET N=H(I,1)+
H(I,2): IF N<0 OR N>31 THEN LET
A=INT (RAND+.5): LET H(I,2)=A-(A=
0): LET N=(H(I,2)--1)*31
110 PRINT AT I*2-1,N;" ": LET H

```

```

(I,1)=N: NEXT I
120 PRINT AT Y,X," ": IF SCREEN
# (Y+1,X)=" " THEN LET SC=SC-20:
G: PRINT AT 21,0;SC;CHR# 144: BE
EP .1,-20: LET Y=Y+2: LET C=145:
LET T=0: IF L<10 THEN LET L=L+1
: IF G>2 THEN GO TO 300
130 LET T=T+1: IF T=UN THEN LET
C=145: BEEP .05,30
140 IF C=145 THEN GO TO 155
145 LET I#=INKEY#: IF I#<>"1" T
HEN GO TO 150
146 IF SCREEN# (Y-1,X)=" " THEN
BEEP .05,0: LET C=145: LET T=0:
GO TO 160
147 IF Y<20 THEN LET L=L-1: IF
L<10 AND G>2 THEN PRINT AT Y+1,0
;B#
148 LET SC=SC+10*G: PRINT AT 21
,0;SC;CHR# 144: BEEP .05,20: LET
Y=Y-2: LET X=X-H(Y/2+1,2)
150 LET X=X+(I#="0")-(I#="5"):
LET X=X+(X=-1)*32-(X=32)*32
154 IF SCREEN# (Y+1,X)=" " THEN
GO TO 120
155 LET N=ATTR (Y,X): IF N=50 T
HEN LET SC=SC+10*G: BEEP .05,5:
PRINT AT 21,0;SC;CHR# 144
157 IF N=48 OR N=51 THEN GO TO
310
160 IF G>3 THEN LET H=Y/2+(Y/2=
0)
162 PRINT INK 0;AT Y,X;CHR# C:
IF Y=0 THEN LET G=G+1: RESTORE 2
000: FOR I=1 TO 13: READ N: BEEP
.1,N: NEXT I: GO TO 90
170 PRINT AT BY,BX," ": LET BX=
BX+D: LET BX=BX+(BX=-1)*32-(BX=3
2)*32: LET A#=SCREEN# (BY+1,BX):
IF SCREEN# (BY+1,BX-D)=" " OR A
#=" " THEN LET BY=BY+2: LET D=-H
(BY/2+1,2): IF BY>11 OR BY>Y THE
N LET BY=0: LET BX=16: LET D=-H(
1,2)
175 LET N=ATTR (BY,BX): IF N=48
THEN GO TO 310
177 IF N=51 THEN LET D=-D: LET
BX=BX+D
180 PRINT INK 0;AT BY,BX;CHR# 1
47
185 IF G=2 AND Y<20 THEN PRINT
INK 0;AT 20,0;N#

```

```

190 GO TO 100
300 FOR I=L*2 TO 20: PRINT AT I
,X;CHR# 146;AT I-1,X," ": BEEP .
1,20-I: NEXT I
310 BEEP 2,-10: LET M=M-1: IF M
=0 THEN INPUT "OUTRO JOGO?" ;A#:
GO TO 1+(A#="N")+9998+(A#="n")+
9998
320 LET G=G-1+(G=1)
330 GO TO 90
300 RESTORE 300: LET T#="": FOR
I=1 TO 63: READ N: LET T#=T#+CH
R# (N+128): NEXT I
310 DATA 0,14,13,0,7,11,0,12,0,
11,0,0,7,11,0,3,3,11,0,11,0,0,10
,5,0,0,10,0,0,6,10,9,0,9,10,0,0,
0,0,0,11,3,0,10,5,0,13,14,0,0,0,
10,6,0,13,14,0,13,12,0,14,12
320 LET T#=" "+T#(TO 22)+
"+T#(23 TO 42)+
"+T#(43 TO )
900 RESTORE 900: FOR I=0 TO 55:
READ N: POKE USA "A"+I,N: NEXT
I
905 LET N=0: DIM B#(32): RANDOM
IZE : LET TOT=0: LET UN=5: LET S
C=0: LET M=4: LET G=1: RETURN
910 DATA 255,255,255,0,0,0,0,0
911 DATA 60,231,180,155,153,65,
65,231
912 DATA 195,165,120,100,100,18
0,231,24
913 DATA 24,102,102,153,153,102
,102,24
914 DATA 0,0,24,60,60,24,24,0,0
915 DATA 16,16,16,55,55,55,254,
0
916 DATA 24,24,60,255,60,24,24,
24
920 LET N#="": LET L#="": FOR I
=1 TO 32: LET N#=N#+CHR# 149: LE
T L#=L#+CHR# 144: NEXT I
930 IF P#="5" THEN GO SUB 2550
940 LET H=1: LET L=10: LET T=10
0: LET X=16: LET Y=20: LET C=145
950 INK 1: PAPER 6: BORDER 5: C
L3
960 LET BX=16: LET BY=0
970 IF G>1 THEN FOR I=1 TO 20:
PRINT INK 2;AT INT (RND*10)*2,RN
D*31;CHR# 148: NEXT I
980 IF G>4 THEN FOR I=2 TO 10 5

```

```

TEP 2: PRINT AT 1,15; INK 3;CHR#
150: NEXT I
1000 DIM H(10,2)
1010 FOR I=1 TO 21 STEP 2: PRINT
AT I,8;L#: NEXT I
1020 FOR I=1 TO 10: LET H(I,1)=I
NT (RND*31): LET A=INT (RND+.5):
LET H(I,2)=A-(A=0): NEXT I
1030 LET D=-H(1,2)
1040 PRINT AT 21,8;60;AT 21,15;"
NIVEL";CHR# 144;6;AT 21,29;TOT:
FOR I=1 TO H-1: PRINT AT 21,I*2
;CHR# 145: NEXT I
2000 RETURN
2010 DATA 2,2,2,14,14,14,11,11,1
1,7,11,7,2
2500 PAPER 1: INK 7: BORDER 1: C
LS : INPUT : PRINT AT 10,0;"Des
aja instrucoes?"
2505 LET I=0
2510 LET I#:=INKEY#: PRINT AT 0,0
; INK I;T#: IF I#="" THEN LET I=
I+1-(I=8)*8: GO TO 2510
2520 LET P#="N": IF I#="S" OR I#
="s" THEN LET P#="S"
2530 RETURN
2550 LET I=0: INK 7: PAPER 1: CL
S : BORDER 1: INPUT : PRINT AT
4,10; INK 6;" NIVEL ";6: GO SUB
6*100+2500
2560 LET I#:=INKEY#: PRINT AT 0,0
; INK I;T#: IF I#="" THEN LET I=
I+1-(I=8)*8: GO TO 2560
2570 BEEP .5,10: RETURN
2580 PRINT AT 6,0;" O seu object
ivo consiste em attingir o top
o a fim de passar ao nivel segu
inte. Sobee saltand
o pelos intervalos das linhas su
periores e nao caindo pelos
da linha onde se encontra.
Deve tambem
evitar o barril que se move nas l
inhas, senao perde uma vida."
2610 PRINT " Mova-se para a esqu
erda e a direita usando as tecl
as S e D, e salte com a tecla I"
: RETURN
2700 PRINT AT 6,0;" A partir des
te nivel encontrarapontos vermel
hos e ganhara pontuacao passando
sobre eles."

```

```

2710 PRINT " Neste nivel nao pod
e voltar pa-ra a linha na base."
: RETURN
2800 PRINT AT 6,0;" Enquanto sob
e, as linhas por baixo de si d
esaparecem, pelo que perdera u
ma vida se cair." : RETURN
2900 PRINT AT 10,0;" Tudo vai se
r mais rapido daqui em diante."
: RETURN
3000 PRINT AT 10,0;" O toque dos
picos na parte su- perior e' te
tal..": RETURN
9900 RETURN
9999 INK 0: PAPER 7: BORDER 7

```

BOMBARDEIRO

Nenhuma colecção de jogos para computador poderia estar completa sem um do tipo «Bombardeiro». Esta versão é bastante melhor do que muitas outras, e constitui um verdadeiro desafio para o jogador. Vejamos um exemplo de execução:



O cenário é familiar. Você encontra-se numa nave espacial que desce sobre a superfície de um planeta. Não dispõe de meios para subir novamente, e deve portanto «fabricar» à bomba uma

Você controla a sua base de lasers usando as teclas «Z» e «Y», além da tecla SPACE para disparar. Conseguirá 50 pontos por cada pássaro que atingir na descida, e 25 se o pássaro se encontrar na linha inferior. No entanto, cada pássaro atingido não será morto, apenas atordoado, e depressa voltará ao combate.

Você começa com três bases de lasers e perderá uma cada vez que for atingida por um pássaro ou uma bomba. O número de bases é indicado na linha superior da imagem, juntamente com a sua pontuação (à esquerda) e a pontuação máxima actual. Ganhará uma base extra quando tiver destruído três ondas de atacantes.

Neste ponto, você ver-se-á ameaçado por alienígenos mutantes. Estes deslizam como os pássaros, mas podem subir, largando mais bombas.

Para você ter um objectivo, talvez lhe interesse saber a pontuação máxima obtida por Neil Pellinacci — e só depois de bastante prática: 12 000.

O programa está escrito inteiramente em Basic, exceptuando uma rotina em código-máquina localizada no endereço 32500. Apresentamos em seguida uma listagem das instruções em código-máquina:

```

10 LD HL,704
20 LD DE,22526
30 LD BC,32
40 PUSH HL
50 ADD HL,DE
60 DEC HL
70 LD A,(HL)
80 C 126
90 JA Z,32539
100 POP HL
110 DEC HL
120 LD A,H
130 OR L
140 JA NZ,32569
150 LD HL,23263

160 LD B,32
170 LD A,(HL)
180 CP 126
190 JA NZ,32536
200 DEC HL
210 DJNZ 32528
220 LD C,0
230 RET
240 LD (HL),70
250 ADD HL,BC
260 LD (HL),126
270 JR 32517

```

Depois de terminar a introdução do programa, talvez lhe interesse alterar o endereço onde se encontra o código-máquina. Se assim acontecer não se esqueça de alterar as linhas 90, 317, 412 e 7020. A subrotina determina a posição de cada bomba e desloca-a para baixo uma posição de cada vez. Verifica ainda se o planeta se encontra já completamente coberto de bombas. O resultado deste teste é detectado ao voltar ao Basic na linha 317.

Deve introduzir o programa e em seguida gravá-lo em fita, antes de tentar executá-lo para o caso de ter cometido algum erro na escrita dos valores que constituem o código-máquina. É também conveniente gravar o programa de tal modo que entre automaticamente em execução ao ser carregado no computador.

```

10 RANDOMIZE ; GO TO 90
20 PRINT AT 0,6-LEN STR$ 5; IN
K 5;5; RETURN
30 LET BC=BC+(INKEY#="X" AND B
C<27)-(INKEY#="Z" AND BC>0); PRI
NT AT 21,BC; PAPER 8; INK 6; "
40 REM GRAFICOS EFG
35 IF INKEY#(">)" THEN RETURN
40 GO TO 520
50 FOR X=1 TO 3; FOR Y=1 TO 10
60 IF L(X,Y)<>-1 THEN LET L1=L
(X,Y); LET C1=C(X,Y); LET C2=Y;
LET L2=X; RETURN
70 NEXT Y; NEXT X; LET WF=1; R
ETURN
80 FOR X=20 TO 7+INT (AND#10)
STEP -1; PRINT AT X,BC+2; PAPER
7; " "; BEEP ,001,40-X; PRINT AT
X,BC+2; PAPER 0; " "; NEXT X; PRI
NT AT X,BC+2; INK 6; PAPER 7; " "
; RETURN
90 CLEAR 32499
95 BORDER 6; PAPER 0; CLS ; BO
RDER 0
100 INK 4; BRIGHT 1; PRINT AT 0
,26; INK 5; "000000"
105 GO SUB 7000; GO SUB 9000
107 LET HS=0
110 GO SUB 3000; LET BC=13; LET
S=0; LET LB=3; LET UU=0
120 LET FF=0
130 LET HF=0
140 LET BFF=0
150 LET WF=0
200 DIM L(3,10); DIM C(3,10)
205 FOR B=0 TO 2

```

```

210 FOR A=0 TO 9
220 LET L(B+1,A+1)=B*2+1: LET C
(B+1,A+1)=A*3+1
230 NEXT A
235 NEXT B
237 PRINT AT 0,0: INK 5;"000000
";AT 0,17;LB
: REM GRAFICOS EFG
240 FOR B=1 TO 3
250 FOR A=1 TO 10: PRINT AT L(B
A),C(B,A);"X": NEXT A: REM GRA
FICOS CD
255 NEXT B
260 LET BH=0
290 FOR K=1 TO INT (RAND*15+1)
300 LET B=INT (RAND*3+1): LET A=
INT (RAND*10+1)
305 IF L(B,A)=-1 THEN GO TO 315
310 PRINT AT L(B,A),C(B,A);"X"
: BEEP .01,-10: PRINT AT L(B,A)
,C(B,A);"X": BEEP .01,5: REM GRA
FICOS AB e CD
312 IF RAND>.6 THEN IF ATTR (22,
BC+2)=126 THEN GO SUB 30
315 IF INKEY#<>" " THEN GO SUB 3
0
317 IF NOT USA 32500 THEN GO TO
4000
318 IF ATTR (21,BC+1)=126 OR AT
TR (21,BC+2)=126 OR ATTR (21,BC+
3)=126 THEN GO TO 1000
320 NEXT K
330 IF BH>20 THEN GO SUB 50: IF
WF=0 THEN GO TO 405
340 IF WF=1 THEN GO TO 2000
400 LET L2=INT (RAND*3+1): LET C
2=INT (RAND*10+1)
402 LET L1=L(L2,C2): LET C1=C(L
2,C2)
404 IF L1=-1 THEN GO TO 400
405 PRINT AT L1,C1;" "
406 LET CD=INT (RAND*3-1)
410 PRINT AT L1,C1: OVER 1;"X"
: BEEP .001,L1+20: PRINT OVER 1;
AT L1,C1;"X": REM GRAFICOS CD
411 IF RAND>.8 THEN IF L1>6 THEN
PRINT AT L1+1,C1: INK 6: PAPER
7;" "
412 LET ZZ=USA 32500: IF INKEY#
<>" " THEN GO SUB 30: IF HF=1 THE
N LET HF=0: GO TO 290
413 IF ATTR (21,BC+1)=126 OR AT

```

```

TR (21,BC+2)=126 OR ATTR (21,BC+
3)=126 THEN GO TO 1000
415 LET C1=C1+CD: IF C1=32 THEN
LET C1=0
416 IF C1=-1 THEN LET C1=30
418 IF RAND>.85 THEN LET CD=INT
(RAND*3-1)
419 IF UU>=3 THEN IF RAND>.6 THE
N IF L1>8 THEN LET L1=L1-2
420 LET L1=L1+1: IF L1<>21 THEN
GO TO 410
425 IF C1=BC+1 OR C1=BC+2 THEN
LET BFF=1: GO TO 1000
430 PRINT AT L(L2,C2),C(L2,C2)
;"X": GO TO 290: REM GRAFICOS AB
520 PLOT BC*8+20,8: DRAW INK 3,
PAPER 0,0,128: PLOT OVER 1,BC*8
+20,8: DRAW OVER 1,0,128
530 IF BC+2=C1 OR BC+1=C1 THEN
GO TO 510
540 IF SCREEN# (5,BC+2)<>" " TH
EN PRINT AT 5,BC+1," " : LET S=
5+25: GO SUB 20
600 RETURN
610 PRINT AT L1,C1: INK 7;"X"
AT L1,C1;" " : LET C1=-1: LET L1
=-1: BEEP .05,-30: LET L(L2,C2)=
-1: LET S=5+50: GO SUB 20: REM G
RAFICOS KL
620 LET HF=1: LET BH=BH+1: RETU
RN
1000 REM BANG
1010 FOR Y=1 TO 4: FOR Z=1 TO 10
: BEEP .015,20-Z: PRINT AT 21,BC
: INK RAND*7;"X": NEXT Z: REM
GRAFICOS HIJ
1020 FOR Z=10 TO 1 STEP -1: BEEP
.015,20-Z: NEXT Z: NEXT Y
1030 LET LB=LB-1: PRINT AT 0,17;
: INK 5;LB
1040 IF LB=0 THEN GO TO 1060
1045 IF BFF=1 THEN LET BFF=0: GO
TO 430
1050 GO TO 415
1060 PRINT AT 0,17: INK 1: PAPER
7: FLASH 1;LB
1070 FOR X=1 TO 20: PRINT AT X,0
:
: NEXT X
1080 BEEP .2,50: BEEP .25,30
1090 FOR X=1 TO 7: PRINT AT X+7,
0: INK X;" " : JOGO TERMINAD

```

```

0      ": NEXT X
1100 BEEP .2,50: BEEP .25,30
1270 IF S>HS THEN LET HS=S
1280 PRINT AT 0,32-LEN STR# HS;
INK 5;HS
1290 IF S=HS THEN FOR Z=1 TO 5:
FOR X=20 TO -20 STEP -2: BEEP .0
1,X: NEXT X: NEXT Z
1300 FOR X=0 TO 20: BEEP .005,20
: NEXT X
1305 FOR X=0 TO 20: BEEP .005,0:
NEXT X
1310 FOR X=1 TO 300: NEXT X: BOR
DER 5: CLS : BORDER 0: PRINT AT
0,25: INK 5;"000000";AT 0,32-LEN
STR# HS; INK 5;HS: GO TO 110
1400 STOP
2000 REM FINAL DA ONDA
2010 FOR X=1 TO 20: PRINT AT X,0
:
": NEXT X
2020 FOR X=1 TO 7: PRINT AT 3+X*
2,2: INK X;" ONDA DE ATAQUE D
ESTRUIDA "
2030 NEXT X
2040 PAUSE 150: BEEP .2,50: FOR
X=-30 TO 35: BEEP .01,X: BEEP .0
15,20: NEXT X: BEEP .2,50
2050 FOR X=1 TO 20: PRINT AT X,0
:
": NEXT X
2060 LET UU=UU+1: IF UU=3 THEN B
ORDER 5: PAPER 0: CLS : BORDER 0
: PRINT AT 0,25: INK 5;"000000";
AT 0,32-LEN STR# HS; INK 5;HS: B
EEP .2,10: BEEP .2,-10: LET LB=L
B+1
2070 GO TO 120
3000 REM PAGINA DE TITULO
3010 RESTORE 3500
3020 READ X: IF X=999 THEN GO TO
3100
3040 READ Y: PLOT X,Y
3050 READ X: IF X=-999 THEN GO T
O 3020
3060 READ Y: DRAW X,Y: GO TO 305
0
3100 PRINT AT 10,4: INK 5;"© NEI
L PELLINACCI 1983"
3110 PRINT AT 13,7: INK 2: PAPER
7;"C O M A N D O S"
3120 PRINT AT 15,2: INK 6;"Z...

```

```

.....ESQUERDA"
3130 PRINT AT 17,2: INK 6;"X....
.....DIREITA"
3140 PRINT AT 19,2: INK 6;"<SPAC
E>.....DISPARAR"
3150 PRINT AT 21,6: INK 2: PAPER
7;"CARREGUE NUMA TECLA."
3160 INK 7: PLOT 2,2: DRAW 0,68:
DRAW 53,0: PLOT 175,70: DRAW 77
,0: DRAW 0,-68: DRAW -45,0: PLOT
2,2: DRAW 45,0
3165 INK 2: PLOT 0,83: DRAW 255,
0: DRAW 0,16: DRAW -255,0: DRAW
0,-16
3170 INK 4
3200 FOR X=30 TO -30 STEP -1: BE
EP .01,X: BEEP .01,-X: NEXT X: P
AUSE 0
3210 FOR X=1 TO 21: PRINT AT X,0
:
": NEXT X
3220 RETURN
3400 RETURN
3500 DATA 35,110,0,56,30,-14,-30
,-14,30,-14,-30,-14,-999
3510 DATA 36,110,0,56,-999
3520 DATA 100,110,0,56,30,-14,-3
0,-14,30,-20,-999
3530 DATA 151,110,0,56,30,-28,-3
0,-28,-999
3540 DATA 186,110,30,28,-30,0,30
,28,-999
3700 DATA 999
4000 REM ATERRAR
4010 BEEP .2,3: BEEP .15,-1: BEE
P .25,5
4020 FOR Z=1 TO 23
4030 FOR X=1 TO 3: FOR Y=1 TO 10
4040 IF L(X,Y)<0 OR L(X,Y)>20 TH
EN GO TO 4055
4050 PRINT AT L(X,Y),C(X,Y): INK
3:" " : LET L(X,Y)=L(X,Y)+1: PR
INT AT L(X,Y),C(X,Y): INK 3;"&"
: REM GRAFICOS CD
4060 NEXT Y: NEXT X
4070 NEXT Z
4080 FOR Z=20 TO -30 STEP -3: BE
EP .01,Z: BEEP .01,Z+1.5: BEEP
01,-Z+10: NEXT Z
4090 BEEP .15,-5: BEEP .15,-10:
BEEP .13,1
4095 FOR R=1 TO 5.

```

```

4100 FOR Z=1 TO 10 STEP 2: PRINT
AT Z,0; INK AND#6+1; " OS PA
SSAROS ATEERRAM. " : BEEP .0
1,Z#2: NEXT Z
4105 NEXT A
4200 BEEP 2,-10
4300 GOTO 1000
4900 STOP
7000 REM M/C
7020 RESTORE 7030: FOR A=1 TO 45
: READ B: POKE 32400+A,B: NEXT B
7030 DATA 33,162,22,17,0,00,1,02,
0,229,25,43,120,25,4,120,40,22
7040 DATA 22,5,43,124,181,30,242
7050 DATA 33,223,90,6,32,126,254
,125,32,5,43,10,248,14,0
7060 DATA 201,64,70,9,54,126,24,
227
8000 RETURN
9000 FOR C=0 TO 95: READ B: POKE
USR "A"+C,B: NEXT C
9010 DATA 102,113,26,7,1,2,2,0
9020 DATA 0,142,88,224,128,64,64
,0
9030 DATA 0,1,2,7,25,114,196,0
9040 DATA 0,128,64,224,152,78,35
,0
9050 DATA 0,0,0,0,4,14,31,255
9060 DATA 0,0,28,28,62,127,255,2
,55
9070 DATA 0,0,0,0,16,58,252,255
9080 DATA 0,2,0,0,18,0,0,255
9090 DATA 0,0,42,128,42,201,34,2
,55
9100 DATA 0,32,0,128,36,136,0,25
,0
9110 DATA 129,72,2,16,130,8,64,1
,33
9120 DATA 129,18,64,9,64,16,2,16
,1
9300 RETURN

```

Vejamos agora a função de cada parte do programa:

- 10 — elimina as subrotinas seguintes, que se encontram no princípio a fim de permitir ao programa ser executado a uma velocidade máxima.
- 20 — Subrotina que imprime a pontuação.
- 30 — Desloca a base de lasers. Se se carrega na tecla SPACE, o comando é enviado para a linha 520.

- 50 — Início da subrotina que só é chamada quando restam apenas alguns pássaros. Localiza o primeiro vivo, e selecciona-o para atacar. Se não restarem pássaros, passa WF para 1 a fim de indicar ao ciclo principal o que se passou
- 80 — Uma rotina para dificultar o jogo.
- 90 — Define a área que será ocupada pelo código-máquina.
- 100 — Inicialização do jogo.
- 120/160 — Define os pássaros.
- 200/235 — Define os pássaros.
- 240/255 — Imprime os pássaros.
- 290/320 — Leva alguns dos pássaros a baterem as asas aleatoriamente.
- 317 — Verifica estado das bombas; move-as para baixo.
- 318 — Verifica se há colisões entre a base bomba.
- 330 — Selecciona o pássaro seguinte sem recorrer à rotina de selecção aleatória, ver a subrotina 50.
- 340 — Verifica se terminou a onda atacante.
- 400/404 — Selecciona um pássaro aleatoriamente, e verifica se existe.
- 410/420 — Ciclo principal.
- 411 — Larga uma bomba.
- 419 — Se o pássaro é um mutante, pode movê-lo para cima.
- 520 — Rotina de disparo.
- 610/620 — Explosão do pássaro.
- 1000/1050 — Rotina de base atingida.
- 1060/1310 — Rotina de jogo terminado.
- 2000/2070 — Rotina de finalização da onda atacante.
- 2060 — Início das ondas de mutantes.
- 3000/3210 — Rotina da página de título.
- 3500/3700 — Dados para a página de título.
- 4000/4300 — Esta rotina é chamada quando o planeta se encontra completamente coberto de bombas; leva os pássaros a mudarem de cor e a aterrarem lentamente.
- 7020 — Carrega o código-máquina.
- 7030/7060 — Dados para o código-máquina.
- 9000 — Define os UDG's.
- 9010/9020 — Dados para os caracteres gráficos.

Finalmente, vejamos uma lista das principais variáveis usadas no programa:

- HS — Pontuação máxima
- BC — Posição da base
- S — Pontuação
- LB — Bases restantes
- WW — Número da onda atacante
- HF — Flag de tiro (1 — atingido; 0 — não)
- WF — Flag de final de onda atacante (1 — final; 0 — não)
- L(3,10), C(3,10) — Posições dos 30 pássaros.
- BH — Número de pássaros atingidos
- LI, CI — Posição dos pássaros em ataque
- CD — Direção horizontal dos pássaros

Tal como acontece com a maioria dos programas apresentados neste livro, existem muitas coisas que podem ser alteradas de acordo com os desejos do utilizador. Por exemplo, os números aleatórios podem ser alterados, modificando-se assim a probabilidade de ocorrência de diversos acontecimentos (é o que se pode fazer actuando, por exemplo, nos valores utilizados nas linhas 411 e 419). Note-se que é igualmente possível alterar as imagens apresentadas em várias fases do jogo.

HELICÓPTERO

Este jogo de Malcolm Young tem bastante acção. Você tripula um helicóptero de salvamento voando sobre o deserto.

Terá de salvar tantos soldados quantos puder antes de ficar sem combustível. Pode recolher um soldado quer aterrando sobre ele (o que lhe dará 50 pontos) ou, para tornar a operação mais fácil e rápida, descendo o gancho do helicóptero quando se encontra exactamente dois espaços acima do soldado. Neste caso obterá 25 pontos.

Pode igualmente usar o gancho para eliminar árvores. No entanto, ao fazê-lo você reduz a cobertura do helicóptero. Esta é importante porque lhe permite esconder-se dos inimigos que se deslocam em jeeps e disparam sobre si. Estes só podem disparar na vertical, e você pode destruir os jeeps largando bombas sobre eles. Você só os pode bombardear enquanto se encontra no céu, e fá-lo-á carregando também na tecla I (o programa sabe se você quer largar uma bomba ou o gancho, o que permite usar a mesma tecla para ambos os casos).

Depois de recolher um homem, a ambulância torna-se negra e você ganhará um bônus de 100 pontos se colocar o soldado nessa ambulância.

Para aumentar o número de coisas que você deve ter em conta ao jogar este programa, verificará que por vezes surge um camião-tanque na parte inferior da imagem. Se conseguir bombardeá-lo, ganhará vinte unidades de combustível extra, além de 10 pontos.

Este programa termina quando você fica sem combustível para o helicóptero ou quando tiver perdido todas as vidas de que dispõe inicialmente. No final do programa é apresentada uma tabela de pontuações, que você poderá ampliar facilmente até cinco posições.

```

5 GO SUB 550
10 REM a=1 b=2 c=3 d=4 e=5 f=6
g=7 h=8 i=9 j=10 k=11 l=12 m=13 n=14
o=15 p=16 q=17 r=18 s=19 t=20
20 DIM a$(5,10): DIM h(5)
30 LET tr=0: LET sf=0: LET dp=0:
LET sc=0: LET pk=0: LET vidas=3
32 BORDER 7: PAPER 7: INK 0: B
RIGHT 0: CLS
110 LET JF=0: LET ln=100: LET c
omb=200
120 BORDER 6: PAPER 5: INK 0: C
LS
130 LET h#="": LET r#="": L
ET a#="": LET j#="": LET p#="
": LET t#="":
140 FOR g=11 TO 21: PRINT PAPER
6;AT 9,0;"
": NEXT g
150 FOR t=0 TO 31: PAPER 8: LET
r=INT (AND*4): PRINT INK 2;AT 1
5-r,t;"Y"; INK 4;AT 15-5,t;"%"
NEXT t
160 LET y=INT (AND*10)+1: LET x
=INT (AND*30): LET a=1
170 PRINT PAPER 6;AT 11,a-1;"
"; INK 2-2*pk;AT 11,a;a#
175 LET comb=comb-1: PRINT PAPE
R 0; INK 7; BRIGHT 1;AT 0,0;"COM
B=";comb;" ";AT 0,10;"PONT=";sc
;AT 0,20;"VIDAS=";vidas
178 IF comb<0 THEN GO TO 1000
180 LET a=a+.5 AND (a<30): IF
NOT a THEN LET pk=a
190 PRINT OVER 0;AT y,x;"
191 IF y=11 AND x+1=a THEN LET
sc=sc+50: LET pk=0: BEEP ,5,-5

```

```

200 LET y=y+(INKEY#="8")-(INKEY
#="7"): LET x=x+(INKEY#="8")-(IN
KEY#="7")
200 IF y<11 THEN LET x=x+(INKEY
#="8")-(INKEY#="7")
210 LET y=y+(y<8)-(y>21): LET x
=x+(x<8)-(x>30)
220 LET c=ATTR (y,x): IF c<>40
AND c<>48 THEN GO TO 1000
230 LET c=ATTR (y,x+1): IF c<>4
0 AND c<>48 THEN GO TO 1000
240 PRINT OVER 0;AT y,x;h#: BEE
P .1,-10: PRINT OVER 1;AT y,x;f#
245 IF AND<.3 AND NOT pk THEN L
E T sf=1
240 IF sf THEN GO SUB 420+dp
250 IF AND<.5 THEN LET Jf=1
260 GO TO Jf*ln+170
270 LET Jy=16+INT (AND*4)+1
280 LET ln=100
290 LET Jd=AND: LET Jd=(Jd>.5)-
(Jd<.5)
300 LET a#=b# AND Jd=1: LET a#=
a#+(J# AND Jd=-1): IF AND<.333 T
HEN LET tr=1
305 IF tr=1 THEN LET a#=(t#(2 T
O ) AND Jd=1)+(t#( TO 2) AND Jd=
-1)
310 LET Jx=30 AND Jd=-1
320 PRINT INK 0;AT Jy,0;" "
325 PRINT INK 1;AT Jy,Jx;a#
330 LET Jx=Jx+Jd
340 IF Jx=0 OR Jx=31 THEN LET l
n=100: LET tr=0: LET Jf=0: PRINT
PAPER 0;AT Jy,0;" " GO TO 170
350 IF y<10 AND INKEY#="1" THEN
GO TO 610
355 IF tr=1 OR NOT (Jx>=INT x-1
AND Jx<=x+1) THEN GO TO 170
370 LET e=y+1: FOR f=Jy-1 TO e
STEP -1: LET b=ATTR (f,Jx+1): IF
b=40 OR b=48 THEN PRINT PAPER 0
; INK 0;AT f,Jx+1;" " BEEP .1,f
; PRINT INK 0;AT f,Jx+1;" " NEX
T f
380 PRINT INK 0;AT f,Jx+1;" "
390 IF (Jx+1=x+1 OR Jx+1=x) AND
f=y THEN BEEP 1,-5: GO TO 1000
400 GO TO 170
410 REM transportar soldado
420 LET sdx=INT (AND*30)+1: LET

```

```

sdy=12+INT (AND*5): LET dp=10:
LET t=INT (AND*20)+10
430 PRINT INK 1;AT sdy,sdx;"X":
BEEP .15,5
440 LET t=t-1: PRINT AT sdy,sdx
" " IF NOT t THEN LET sf=0: LE
T dp=0: RETURN
450 IF y=sdy AND x+1=sdx THEN L
E T sc=sc+50: BEEP 1,5: LET sf=0:
LET dp=0: LET pk=1: RETURN
460 IF INKEY#<"1" THEN RETURN
470 PRINT AT y+1,x+1;"1";AT y+2
,x+1;"1"
480 IF sdy=y+2 AND sdx=x+1 THEN
FOR m=y+2 TO y STEP -1: PRINT A
T m,x+1;"X": BEEP .3,0: PRINT AT
m,x+1;" " NEXT m: LET pk=2: LE
T sf=0: LET dp=0: LET sc=sc+25
490 PRINT AT y+1,x+1;" " AT y+2
,x+1;" "
500 RETURN
510 LET e=Jy: FOR f=y+1 TO e: L
E T b=ATTR (f,x+1): IF b=40 OR b=
48 THEN PRINT PAPER 0: INK 0;AT
f,x+1;"X": CHR# 8: BEEP .1,30-f:
PRINT INK 0;AT f,x+1;" " NEXT f
: GO TO 630
520 PRINT INK 0;AT f,x+1;" "
530 IF b=40 THEN BEEP 1,-5: LET
sc=sc+10: LET comb=comb+20*(tr=
1): PRINT AT Jy,Jx;" " LET J
x=0: GO TO 340
540 GO TO 170
550 REM
560 RESTORE
570 DATA "a",1,1,1,255,255,195,
219,24
580 DATA "b",15,124,8,63,63,195
,219,24
590 DATA "c",128,128,128,255,25
5,195,219,24
610 DATA "d",48,12,10,252,252,1
95,219,24
620 DATA "e",8,16,124,254,255,1
97,35,125
630 DATA "f",0,0,192,192,255,1,
0,0
640 DATA "g",255,0,0,0,0,0,0,0
650 DATA "h",7,8,8,8,8,8,8,0
660 DATA "i",126,97,97,255,255,
195,219,24
670 DATA "j",255,239,199,235,25

```


O objetivo consiste em conduzir a serpente num jardim — rodeado por uma sebe electricada — comendo simultaneamente os cogumelos que vão surgindo. Cada um destes leva a serpente a crescer, e você deve mover-se de tal modo que a serpente nunca toque em si mesma — senão o jogo termina.

E em seguida a listagem do programa:

```

0000 GO SUB 210
0001 REM ***Image# inicial***
0002 LET I=0: INVERSE 0: BRIGHT
0003 FLASH 0: BORDER 0: PAPER 0: I
NK 7: CLS
40 PRINT "      █      █      █      █      █
      █      █      █      █      █
50 PRINT " █      █      █      █      █
      █      █      █      █      █
60 PRINT " █      █      █      █      █
      █      █      █      █      █
70 PRINT " █      █      █      █      █
      █      █      █      █      █
80 PRINT " █      █      █      █      █
      █      █      █      █      █
90 PRINT " █      █      █      █      █
      █      █      █      █      █
100 PRINT " █      █      █      █      █
      █      █      █      █      █

110 PRINT AT 14,1:"Use as tecta
s 0,Z,I & P para subir, descer
, esquerda & direita. Sempre qu
e comer a cobra cresce!"
PRINT "Tenha cuidado em nao tro
pear em si mesmo ou na sebe!"
120 PRINT : PRINT INK 3;TAB 11;
"BOA SORTE!"
140 PRINT AT 10,5; INK I;"INDIO
UE NIVEL DE JOGO."
150 PRINT AT 12,12; INK I/2;"(1
-- 0)"
165 REM Definir nivel
180 LET A#=INKEY#: IF A#="" THE
N LET I=I+1: IF I>7 THEN LET I=0
170 IF A#<"1" OR A#>"9" THEN GO
TO 140
180 LET L=VAL A#
190 INK 0
200 GO TO 350
205 REM Graficos
210 FOR A=1 TO 5
220 READ A#
230 FOR N=0 TO 7
240 READ B

```

```

250 POKE USA A#+N,B
260 NEXT N
270 NEXT A
280 DATA "A",109,102,195,153,15
3,195,102,109
290 DATA "B",0,60,126,153,24,24
,24,24
300 DATA "C",90,36,189,102,102,
189,36,90
310 DATA "D",0,34,34,85,85,85,1
35,135
320 DATA "E",28,96,28,3,28,96,2
8,3
330 LET HS=0
340 RETURN
350 LET B#="p"
360 BORDER 7
370 PAPER 7
380 LET A=0
390 LET SC=0
400 CLS
405 REM
410 PRINT AT 0,0;"
411 PRINT AT 0,0; INK 5;"<PONTO
S:0>"; INK 1;AT 0,11;"M>"; INK 0
;"COBRA"; INK 1;"M"; INK 4;AT 0
,20;"<MAXIMO:";HS;AT 0,31;">": R
EM GRAFICO D
420 FOR N=1 TO 20
430 PRINT AT N,0; INK 1;"#
2": REM
GRAFICO E
440 NEXT N
450 PRINT AT 21,0; INK 1;"~~~~~
~~~~~": RE
M GRAFICO D
460 LET X=10
470 LET Y=10
480 LET A#=""
490 FOR N=10 TO 19
500 LET A#=A#+10+STR# N
510 PRINT BRIGHT 1; INK 2;AT 10
,N;A#": REM GRAFICO A
520 NEXT N
530 GO SUB 800
540 LET D#=INKEY#
545 REM ***ENTRADAS
550 IF D#="p" OR D#="q" OR D#="
" OR D#="z" THEN LET B#=D#
560 LET X=X+(B#="z")-(B#="q")
570 LET Y=Y+(B#="p")-(B#="i")

```

```

580 IF X=21 OR X=0 THEN GO TO 8
590 IF Y=31 OR Y=0 THEN GO TO 8
600 IF ATTR (X,Y)=122 OR X=22 OR
Y=32 THEN GO TO 830
610 GO SUB 750
620 LET C$=STR$ X
630 IF X<10 THEN LET C$=" "+STR
# X
640 LET D$=STR$ Y
650 IF Y<10 THEN LET D$=" "+STR
# Y
660 LET X1=VAL A$(1 TO 2)
670 LET Y1=VAL A$(3 TO 4)
680 IF ATTR (X1,Y1)<>122 THEN G
O TO 700
690 PRINT AT X1,Y1;" "
695 REM ***IMPRIMIR COBRA
700 PRINT BRIGHT 1; INK 2; AT X,
Y;"0": REM GRAFICO A
710 LET A$=A$+C$+D$
720 IF A=0 THEN LET A$=A$(5 TO
)
730 LET A=A-(A>0)
740 GO TO 540
750 LET M=ATTR (X,Y)
760 IF M<>60 THEN RETURN
770 LET A=A+L
780 LET SC=SC+L
790 PRINT AT 0,0; INK 5;"(PONTO
S:);SC;">" AND SC<100); INK 1;A
T 0,11;"M>"; INK 2;"COBRA"; INK
1;"<M"; INK 4; AT 0,20;"(MAXIMO:
)HS; AT 0,31;">": REM GRAFICO D
800 PRINT PAPER 7; INK 4; AT INT
(RND*18)+2,INT (RND*29)+2;"T":
REM GRAFICO B
810 FOR N=50 TO -20 STEP -10: B
EEP .01,N: NEXT N
820 RETURN
830 LET X=X-(X=22)+(X=-1)
840 LET Y=Y-(Y=32)+(Y=-1)
850 FOR F=50 TO -20 STEP -5: B
EEP .02,F: NEXT F
855 REM ***FIM
860 IF SC>HS THEN LET HS=SC
870 PRINT AT 10,6;"PONTUACAO MA
XIMA: ";HS
880 PRINT AT 12,6;"Carregue num
a tecla."
890 LET I=0
900 PRINT AT X,Y; INK I;"0": BE

```

```

EP .01,I+2: REM GRAFICO C
905 REM ***Pontuacao
910 PRINT AT 8,5; INK I; FLASH
1;">Conseguiu";SC;" pontos,(<
920 LET 0$=INKEY$: IF 0$="" THE
N LET I=I+1; IF I>7 THEN LET I=0
930 IF 0$="" THEN GO TO 900
940 CLS
950 GO TO 30

```

CORRIDAS

Este rápido jogo de acção foi escrito por Raymond Blake. Você conduz o seu automóvel por um percurso, passando sobre bandeirolas onde ganha pontos. Ao mesmo tempo, tenta evitar o automóvel suicida comandado pelo computador, que se move em direcção oposta à sua, tentando estragar o seu divertimento...

Você pode mudar de percurso em quatro pontos, tantas pistas quantas quiser. O carro suicida também pode mudar de pista, mas apenas uma de cada vez. Apesar disto, o leitor verificará que se trata de um jogo bastante difícil.

No início da corrida, você ganha pontos passando sobre bandeirolas vermelhas, que se tornam amarelas quando você passa sobre elas. Quando não tiver bandeirolas vermelhas (ou quiser simplesmente mudar um pouco), pode ganhar pontos com as bandeirolas amarelas desde que passe sobre a bandeirola magenta. Mas isto dará ao seu opositor Kamikaze uma possibilidade de o apanhar.

Aparecerá então uma outra bandeirola magenta noutra ponta do percurso, e passando sobre ela modificará novamente a cor das bandeirolas que dão pontos.

```

20 FOR I=1 TO 5: READ Z#: FOR
J=0 TO 7: READ Z: POKE USA Z#+J,
Z: NEXT J: NEXT I
30 LET HS=0
35 LET C=6: LET C1=2: LET SC=0
40 BORDER 0: PAPER 0: INK 7: B
RIGHT 1: CLS
50 PRINT "#####
#####"
60 PRINT "##"; INK 2;"++++++
++++"; INK 7;"##
": REM Entre as linhas 60 e 100
Usar o grafico A
70 PRINT "##"; INK 2;"#"; INK
7;"#####
#####";

```



```

330 IF a##="8" THEN LET a##="8":
LET n##="8": GO TO 400: REM grafi
CO C
340 IF a##="7" THEN LET a##="5":
LET n##="7": GO TO 400: REM grafi
CO E
350 IF a##="8" THEN LET a##="7":
LET n##="8": REM grafi
CO D
401 IF (x>8 AND x<13) OR (y>13
AND y<18) THEN PRINT AT x,y; INK
c1; " ": GO SUB 900: GO TO 410
402 LET a=ATTR (x+(a##="5")-(a##
="7"),y+(a##="8")-(a##="9"))
403 PRINT AT x,y; INK c;"4": RE
M grafi
CO A
405 IF a=108 THEN GO SUB 850
407 IF a=64+c1 THEN LET sc=sc+1
410 LET x=x+(a##="8")-(a##="7"):
LET y=y+(a##="8")-(a##="9")
420 PRINT AT x,y; INK 4;##
430 PRINT AT 10,17;sc: BEEP .00
1,20
450 IF x=x1 AND y=y1 THEN GO TO
500
500 IF flag>0 THEN LET flag=fla
g-1: BEEP .00,20
510 IF SCREEN$(x1+(b##="6")-(b#
="7"),y1+(b##="8")-(b##="9"))<>"#
" THEN GO TO 500
520 IF b##="5" THEN LET b##="7":
LET n##="5": GO TO 600: REM grafi
CO B
530 IF b##="8" THEN LET b##="5":
LET n##="8": GO TO 600: REM grafi
CO E
540 IF b##="7" THEN LET b##="8":
LET n##="7": GO TO 600: REM grafi
CO C
550 IF b##="8" THEN LET b##="5":
LET n##="8": GO TO 600: REM grafi
CO D
560 IF (x>8 AND x<13) OR (y>13
AND y<18) THEN PRINT AT x,y; " ":
GO SUB 900: GO TO 610
561 IF (x1=9 AND b##="6") OR (y1
=14 AND b##="8") OR (x1=12 AND b#
="7") OR (y1=17 AND b##="9") THEN
PRINT AT x1,y1; INK c1; " ": GO
SUB 700
562 IF (x1>8 AND x1<13) OR (y1>
13 AND y1<18) THEN PRINT AT x1,y
1; INK c1; " ": GO TO 610

```

```

600 LET z=ATTR (x1+(b##="6")-(b#
="7"),y1+(b##="8")-(b##="9"))
605 PRINT AT x1,y1; INK (z1-64
)-(128 AND z1-64>8); FLASH 1 AN
D z1-64>8;"4": REM grafi
CO A
606 LET z1=z
610 LET x1=x1+(b##="6")-(b##="7")
LET y1=y1+(b##="8")-(b##="9")
620 PRINT AT x1,y1; INK 5;n#
630 IF x=x1 AND y=y1 THEN GO TO
650
650 GO TO 300
700 IF traj=traj1 THEN RETURN
705 PRINT AT x1,y1; INK 5;n#
720 LET l=2*SGN (traj-traj1): L
ET traj1=traj+l/2
750 LET x1=x1+l*((b##="8")-(b##="
5"))
760 LET y1=y1+l*((b##="7")-(b##="
6"))
790 RETURN
800 FOR i=25 TO -10 STEP -1: BE
EP .02,i: NEXT i
810 PRINT AT 14,1; PAPER 4; INK
0; FLASH 1;"Carregue numa tecla
para jogar"
815 IF sc>#s THEN LET #s=sc
820 IF INKEY#<>" " THEN GO TO 82
0
830 IF INKEY#="" THEN GO TO 830
840 GO TO 35
850 LET c=8-c: LET c1=8-c1: LET
sc=sc-10: BEEP .01,30: BEEP .01
,30
855 LET flag=10
860 LET sc=sc+10
870 LET lin=INT (AND#20)+1: LET
col=INT (AND#30)+1: IF CODE SCR
EEN$(lin,col)<>0 THEN GO TO 870
875 PRINT AT (lin,col); INK 3; FL
ASH 1;"4"
880 RETURN
900 LET i#=INKEY#: IF y>13 AND
y<18 THEN GO TO 950
905 IF y>22 THEN GO TO 925
910 IF i#="5" AND y>2 THEN LET
y=y-2: LET traj=traj-1
915 IF i#="8" AND y<8 THEN LET
y=y+2: LET traj=traj+1
920 RETURN
930 IF i#="5" AND y>23 THEN LET
y=y-2: LET traj=traj+1

```

```

10000 IF I#="8" AND X<20 THEN LET
10001 LET traJ=traJ-1
10002 RETURN
10003 IF I#="5" AND X<7 THEN LET
10004 LET traJ=traJ+1
10005 IF I#="7" AND X>1 THEN LET
10006 LET traJ=traJ-1
10007 RETURN
10008 IF I#="5" AND X<20 THEN LET
10009 LET traJ=traJ-1
10010 IF I#="7" AND X>14 THEN LET
10011 LET traJ=traJ+1
10012 RETURN
10013 DATA "a",12,60,124,60,12,4,
10014 4,4
10015 DATA "b",24,90,126,90,24,15
10016 3,255,153
10017 DATA "c",224,78,68,255,255,
10018 68,78,224
10019 DATA "d",153,255,153,24,90,
10020 126,90,24
10021 DATA "e",7,114,34,255,255,3
10022 4,114,7
10023 IF flag>0 THEN LET flag=fla
10024 g-1

```

ZOMBIES

Zombies, por David Perry, é um jogo em que você deve tentar atrair Zombies para os pântanos. Os Zombies são cegos, e movem-se sempre em direção a si.

O próprio programa contém as instruções necessárias. Use as teclas de cursos para se mover.

```

1 REM
2 REM
3 REM
4 REM
5 REM
6 REM
7 REM
8 REM
9 REM
10 REM
11 LET NUM=10: DIM N(NUM+1): D
12 IM N#(NUM+1,8): FOR N=1 TO NUM:
13 LET N(N)=1100-(N*100): LET N#(N)
14 ="SPECTRUM": NEXT N

```

```

12 BORDER 7: PAPER 7: INK 0: B
13 RIGHT 0: CLS
14 LET LEU=5: GO SUB 55
15 LET SPC=0
16 BORDER 7: PAPER 7: INK 0: B
17 RIGHT 0: CLS: DIM A(LEU): DIM B
18 (LEU): LET KILL=0
19 PRINT AT 0,3:
20
21 FOR N=1 TO 20: PRINT AT N,3
22 "; INK 4:
23 "; INK 0: "; NEXT N
24 PRINT AT 21,3:
25
26 BRIGHT 0: BORDER 7
27 PRINT AT 2,26;"7":AT 3,27:
28 "5": INK 0;"X": INK 0;"8":AT 4,28
29 "0":AT 5,26;"Hover":AT 1,26;"-
30 -":AT 6,26;"Chave":AT 7,26;"-
31 -"
32 PRINT AT 8,26;"Voce=": INK
33 0;"t":AT 9,26: INK 0;"Zomb=": IN
34 K 0;"0":AT 10,26: INK 0;"Pant.":
35 INK 0;"@": REM GRAFICOS A, B e
36 C
37 PRINT AT 12,27;"BOA":AT 13,
38 26;"BOATE":AT 14,26:
39 PRINT AT 15,28: "; LEU-KILL:
40 AT 17,27;"PARA":AT 18,26;"MATAA:
41
42 FOR N=1 TO LEU
43 LET A(N)=INT (RAND*20): LET
44 B(N)=INT (RAND*20): IF ATTR (A(N)
45 +1,B(N)+4) <>124 THEN GO TO 25
46 PRINT AT A(N)+1,B(N)+4: INK
47 0: PAPER 4;"M": NEXT N: REM gra
48 fico B
49 FOR N=1 TO LEU*5
50 LET A=INT (RAND*20): LET B=I
51 NT (RAND*20): IF ATTR (A+1,B+4) <>
52 124 THEN GO TO 28
53 PRINT AT A+1,B+4: INK 0: PA
54 PER 4;"@": NEXT n: REM grafico
55 C
56 LET X=INT (RAND*20): LET Y=I
57 NT (RAND*20): IF ATTR (X+1,Y+4) <>
58 124 THEN GO TO 30
59 PRINT AT X+1,Y+4: INK 0: PA
60 PER 4;"t": FOR N=0 TO 50 STEP 5:
61 BEEP .02,N: NEXT N: REM grafico
62 A
63 PRINT AT X+1,Y+4: INK 0: PA
64 PER 4;"t": REM grafico A

```

```

33 FOR N=1 TO LEV: IF A(N)=100
AND N=LEV THEN GO TO 49
34 IF A(N)=100 THEN NEXT N
35 IF KILL=LEV THEN GO TO 91
36 IF N>LEV THEN LET N=LEV
37 PRINT AT A(N)+1,B(N)+4; BRIGHT 1; INK 4; " "
38 IF A(N)<X THEN LET A(N)=A(N)+1
39 IF B(N)>Y THEN LET B(N)=B(N)-1
40 IF A(N)>X THEN LET A(N)=A(N)-1
41 IF B(N)<Y THEN LET B(N)=B(N)+1
42 LET A=ATTR(A(N)+1,B(N)+4)
43 IF A=124 THEN PRINT AT A(N)+1,B(N)+4; INK 2; PAPER 4; BRIGHT 1; "M"; BEEP .01,N; NEXT N; REM grafico B
44 IF A=99 THEN LET SC=SC+10; FOR I=7 TO 0 STEP -1: PRINT AT A(N)+1,B(N)+4; BRIGHT 1; PAPER 4; INK I; "B"; BEEP .02,I*7; NEXT I; PRINT AT A(N)+1,B(N)+4; INK 3; PAPER 4; BRIGHT 1; "B"; LET KILL=KILL+1; PRINT AT 16,20; " "; LEV=KILL; " "; LET A(N)=100; IF N<=LEV THEN NEXT N; REM grafico C
45 IF N>=LEV THEN LET N=LEV
46 IF N<=LEV AND A(N)=X AND B(N)=Y THEN FOR I=0 TO 7: PRINT AT A(N)+1,B(N)+4; BRIGHT 1; PAPER 4; INK I; FLASH 1; "†"; BEEP .1,I*7; NEXT I; GO TO 87; REM grafico A
47 IF KILL=LEV THEN GO TO 91
48 IF N<=LEV THEN NEXT N
49 IF LEV=KILL THEN PAUSE 30; GO TO 91
50 FOR G=1 TO 2: PRINT AT X+1,Y+4; INK 4; BRIGHT 1; " "
51 LET X=X+(INKEY#="6")-(INKEY#="7"); LET Y=Y+(INKEY#="8")-(INKEY#="5")
52 LET A=ATTR(X+1,Y+4)
53 IF A=124 THEN PRINT AT X+1,Y+4; INK 6; PAPER 4; BRIGHT 1; "†"; BEEP .04,20; NEXT G; GO TO 32; REM grafico A
54 IF A=99 THEN GO TO 103
55 LET LEV=5; CLS; PRINT AT 1

```

```

,12;"ZOMBIES!";AT 2,12;"=====
56 PRINT AT 3,0;"VOCE ESTA A DESLIZAR LENTAMENTE"
57 FOR I=0 TO 7: PRINT AT I+6,6; PAPER 7-I; INK I;"NO PANTANO!"; NEXT I
58 PRINT AT 16,6;"MAS CONSEGUI U OBTER-"; PRINT AT 19,10;60;" PONTOS!"
59 PRINT #1,TAB 2;"© Copyright DAVID PERRY 1983"
60 LET I=0
61 PRINT AT 21,6; INK I;"CARRE QUE NUMA TECLA,"
62 LET I=I+1; IF I>7 THEN LET I=1
63 IF INKEY#="" THEN GO TO 61
64 LET LEV=5; PAUSE 0
65 GO TO 106
66 CLS; PRINT AT 0,12;"ZOMBIE 51";AT 1,12;"=====
67 PRINT; PRINT TAB 3;"VOCE A CABE DE ATERRAR NUMA"
68 PRINT TAB 6;"ILHA POUADA P OR..."
69 RESTORE 70; FOR N=0 TO 31: READ A; POKE USA "A"+N,A; NEXT N
70 DATA 0,24,126,24,24,24,0,129,195,165,165,102,24,36,124,129,126,66,90,90,66,126,129,153,90,60,231,231,60,90,153
71 FOR N=7 TO 13: PRINT AT N,12; INK N-7;"ZOMBIES!"; BEEP .5,N; NEXT N
72 PRINT AT 14,12;"ZOMBIES"; BEEP .5,35
73 PRINT AT 17,2;"© COPYRIGHT DAVID PERRY 1983"
74 PAUSE 10; FOR N=0 TO 27: PRINT AT 19,N; INK 2;"M"; INK 5;"†"; BEEP .02,N; NEXT N; REM graficos B e A
75 PRINT AT 19,N;" "; PAUSE 10; BEEP .5,40; PAUSE 5; FOR N=27 TO 0 STEP -1: PRINT AT 19,N; INK 2;"M"; INK 5;"†"; BEEP .02,N; NEXT N
76 CLS
77 PRINT AT 0,12;"ZOMBIES!";AT 1,12;"=====
78 PRINT AT 2,1;"Os Zombies sa o comedores de carne... Feliz

```

```

mente sao cegos, e essa sera a s
ua unica vantagem. Procure atraio
los para o pantano e tenha cuida
do: se voce cair tambem morrera
. Como so tem uma vida-
79 PRINT AT 10,10;"ACAUTELE-SE
"
80 PRINT AT 14,4;"Teclas de mo
vimento: 5-8"
81 PRINT AT 16,3;"VOCE="; INK
5;"1"; INK 0;" ZOMBIE="; INK 2;"
%"; INK 0;" PANTANO="; INK 3;"B"
; INK 5;"": REM GRAFICOS A, B e
C
82 LET I=0: PRINT #1,TAB 2;"©
Copyright DAVID PERRY 1983"
83 PRINT AT 19,5; INK I; PAPER
2; BRIGHT 1;" CARREGUE NUMA TEC
LA.
84 LET I=I+1: BEEP .01,I*7: IF
I>7 THEN LET I=0
85 IF INKEY#="" THEN GO TO 83
86 CLS : RETURN
87 CLS : PRINT AT 1,12;"ZOMBIE
5!"; AT 2,12;"=====
88 FOR N=0 TO 7: PRINT AT 4+N,
5; PAPER N; INK 7-N;"ELES APANHA
RAM-NO!!!": NEXT N
89 GO TO 58
90 STOP
91 CLS : PRINT AT 1,12;"ZOMBIE
5!"; AT 2,12;"=====
92 FOR N=7 TO 0 STEP -1: PRINT
AT 4+N,7; PAPER N; INK 7-N;" P
A R A B E N S " : NEXT N
93 LET SC=SC+(LEV*20)
94 PRINT AT 13,1;"VOCE PONTUOU
";SC;" ATE AGORA!"
95 LET LEV=LEV+5: FOR N=1 TO 1
0 STEP 2: BEEP .02,N: NEXT N: PR
INT AT 16,4;"TENTE O NIVEL SEGUI
NTE": PRINT : PRINT " CO
M ";LEV;" ZOMBIES!"
96 PRINT AT 19,7;"BONUS = ";(L
EV-5)*20;"
97 PRINT #1,TAB 2;"© Copyright
DAVID PERRY 1983"
98 LET I=0
99 PRINT AT 21,5; INK I;" CARR
EGUE NUMA TECLA.
100 LET I=I+1: BEEP .02,I*5: IF
I>7 THEN LET I=0
101 IF INKEY#="" THEN GO TO 99

```

```

102 GO TO 15
103 CLS : PRINT AT 1,12;"ZOMBIE
5!"; AT 2,12;"=====
104 FOR N=7 TO 0 STEP -1: PRINT
AT 4+N,3; PAPER N; INK 7-N;"VOC
E ACABOU DE SE SUICIDAR": NEXT N
105 GO TO 58
106 BORDER 0: PAPER 0: INK 7: B
RIGHT 1: CLS
107 IF SC<=N(10) THEN GO TO 123
108 LET NUM=11: IF SC>=N(NUM) T
HEN INPUT "ESCREVA 8 INICIAIS ";
P#: IF LEN P#>8 THEN GO TO 108
109 PRINT AT 7,0; FLASH 1; BRIG
HT 0; INK 7; PAPER 2;"ESPERE ALG
UNS SEGUNDOS!"
110 IF SC>=N(NUM) THEN LET N(NU
M)=SC: LET N#(NUM)=P#
111 FOR A=1 TO (NUM-1): LET B#=#
N#(A): LET C#=#N#(A+1): LET B=N(A
): LET C=N(A+1): IF B<C THEN LET
N(A)=C: LET N(A+1)=B: LET N#(A)
=C#: LET N#(A+1)=B#
112 NEXT A: FOR N=1 TO NUM-1: I
F N(N)<N(N+1) THEN GO TO 111
113 NEXT N
114 CLS
115 PRINT AT 2,6;"TABELA DE PON
TUACOES"
116 PRINT AT 3,5;"=====
=====
117 FOR N=1 TO NUM-1: PRINT AT
N+5,7; INK 6;"("); INK 2;N; INK 5
;" )"; AT N+5,12; INK 7;N(N): PRIN
T AT N+5,17; INK 5;N#(N): NEXT N
118 PLOT 0,0: DRAW 255,0: DRAW
0,175: DRAW -255,0: DRAW 0,-175:
PRINT AT 17,5; PAPER 2; INK 7;"
CARREGUE NUMA TECLA. " : LET I=0
119 LET I=I+1: IF I>7 THEN LET
I=0
120 PRINT AT 2,6; INK I;"TABELA
DE PONTUACOES"
121 BEEP .01,I*7: PAUSE 2: IF I
NKEY#="" THEN GO TO 119
122 PAPER 7: INK 0: BORDER 7: B
RIGHT 0: CLS : GO TO 14
123 PRINT AT 10,4;"PONTUACAO MU
ITO BAIXA!!!"
124 PRINT AT 13,5;" CARREGUE NU
MA TECLA. "
125 PAUSE 0: GO TO 114

```

AVENTURAS

O GIGANTE E A CIDADE

Estes dois programas de aventuras, escritos por Neil Pellinacci, são jogados de forma semelhante, se bem que o cenário e a solução sejam — evidentemente — diferentes. Vale a pena tentar convencer um amigo a introduzir os programas na máquina a fim de você não conhecer as soluções antes de jogar.

A CIDADE inclui instruções no programa que precede o jogo. Para terminar a aventura, você deve dizer a palavra de passe ao seu contacto. Você não sabe quem é o seu contacto, se bem que não haja muitas possibilidades.

Na primeira aventura, O GIGANTE, você deve tentar fugir do covil do Gigante. Este é bastante mau, e só o deixará fugir se estiver morto. Você tem portanto de matá-lo para ganhar.

Em ambos os programas, tal como acontece de resto na maior parte dos jogos de aventuras, você dá as suas ordens em frases de duas palavras (verbos e substantivos): por exemplo LARGAR MACHADO. Depressa descobrirá quais as palavras que o programa reconhece.

Note que cada programa possui um carregador (que no caso de A CIDADE serve também como manual de instruções). Os possuidores de máquinas de 48 K podem desprezar a instrução CLEAR 32767, mas devem manter a CLS. Se você utilizar as instruções de SAVE incorporadas no programa, note que deve definir as cores e realizar os POKE's antes de gravar o programa.

Os carregadores foram usados para o espaço de memória livre para o programa principal. Para introduzir os programas, introduza primeiro, e grave, o carregador, e em seguida introduza e grave o programa principal, imediatamente a seguir. Ao gravar ambos os programas, ou ao executá-los, não se esqueça de que a linha de auto-execução é a 1 no caso dos carregadores e 900 no dos programas principais.

Vejamos agora o carregador de O GIGANTE:

```
10 REM AVENTURA © NEIL  
PELLINACCI 1983  
20 BORDER 0: PAPER 0: INK 7: B  
RIGHT 1: CLS  
30 CLEAR 32767  
40 POKE 23609,100: POKE 23658,  
8: POKE 23624,71  
50 LOAD ""
```

E o programa principal:

```
12 PRINT "ESTOU JUNTO A PORTA  
A DO GIGANTE."  
13 IF 0(9) THEN PRINT "NÃO TENHO  
ARMAS E DEVO FUGIR DE-PRESSA."  
14 LET S=P: RETURN  
18 PRINT "ESTOU NUMA PEQUENA  
LOJA, QUE VENDE COMIDA PARA O  
RES." : LET S=P: RETURN  
33 PRINT "ESTOU NUMA GRUTA ES-  
CURA. VEJO UMA PORTA PARA NORT-  
E, E UMA CLA-REIRA PARA ESTE."  
34 LET N=P: LET S=P: LET E=P:  
RETURN  
36 PRINT "ESTOU NUMA CLAREIRA  
A GRUTA EN-CONTRA-SE A OESTE."  
: LET E=P: RETURN  
39 PRINT "ESTOU NUM CAMINHO Q-  
UE SE DIRIGE PARA SUL. VEJO UMA  
LOJA A NORTE, E UMA FONTE A ESTE."  
40 LET N=P: LET S=P: LET E=P:  
LET W=P: RETURN  
42 PRINT "ESTOU JUNTO A UMA F-  
ONTE. EXISTE UMA ESTRADA PARA SU-  
L." : LET S=P: LET W=P: RETURN  
54 PRINT "ESTOU NUMA PASSAGEM  
ESTREITA QUE SOBE LIGEIRAMENTE P-  
ARA NORTE. VEJO LA EM CIMA A  
LUZ DO SOL." : LET N=P: LET U=P:  
RETURN  
60 PRINT "ESTOU NUM CAMINHO E-  
M PEDRA SOLTAE VEJO UMA ESTRADA  
PARA ESTE." : LET N=P: LET E=P  
61 IF 0(4)=LOC THEN PRINT "VEJ-  
O UM TUMULO A MEUS PES."  
62 RETURN  
63 PRINT "EXISTE AQUI UMA EST-  
RADA QUE COR-RE NO SENTIDO NORTE-  
-SUL, A NORTEVEJO UMA FONTE." : L
```

```

ET N=P: LET S=P: RETURN
89 PRINT "VEJO UMA PAQUENA CL
AREIRA COM UMA GRANDE ARVORE."
: LET S=P: LET E=P: RETURN
72 PRINT "ESTOU NUMA FLORESTA
EXISTEM CLAREIRAS A ESTE E
A SUL."
73 LET W=P: LET E=P: LET S=P:
RETURN
75 PRINT "ESTOU AGORA NUMA CL
AREIRA, VEJO UM POCO PROFUNDO NO
SOLO." : LET W=P: LET D=P: RETUR
N
84 PRINT "A ESTRADA TERMINA A
QUI, VEJO UM RIO PARA SUL, COM M
ONTANHAS EM FUNDO." : LET N=P: R
ETURN
87 PRINT "UM ENORME LOBO SALT
OU DAS ARVO- RES E MATOU-ME!" : G
O TO 9000
90 PRINT "ESTOU NUMA CLAREIRA
, OICO RUIDOSA OESTE." : LET N=P:
LET E=P: LET W=P: RETURN
93 PRINT "A FLORESTA TERMINA
AQUI, A SUL ENCONTRA-SE UM RIO,
TALVEZ SEJA POSSIVEL ATRAVESSA-
LO." : LET N=P: LET W=P: RETURN
114 PRINT "ESTOU NA MARGEM DO
RIO, VEJO UM PORTAO ABERTO, GUAR
DADO POR UM CAO, E UMA MANSAO M
AIS ADIANTE."
115 LET S=DOG: IF NOT S THEN PR
INT "O CAO PODE TRAZER PROBLEMAS
..."
116 RETURN
117 PRINT "ESTOU PERTO DE UMA
MANSAO, VEJO UMA PORTA A MINHA
FRENTE." : LET N=P: RETURN
138 PRINT "ESTOU DENTRO DA MAN
SAO, NAO GOS-TARIA DE VIVER AQUI
, ESTA TUDO SUJO E CHEIRA MAL."
LET W=P: RETURN
800 PRINT "VEJO...": LET FL=0:
FOR Z=P TO 9: IF D(Z)=LOC THEN
GO TO 830
810 NEXT Z: IF NOT FL THEN PRIN
T "NADA"
820 RETURN
830 IF Z=4 AND LOC=20 THEN GO T
O 810
840 PRINT O$(Z): LET FL=1: GO T
O 810

```

```

850 IF A$(P)=" " THEN LET A#-A#
(2 TO ): GO TO 850
860 IF A$(LEN A$)=" " THEN LET
A#-A$( TO LEN A$-P): GO TO 860
870 LET B#="": FOR Z=P TO LEN A
$: IF A$(Z)=" " THEN LET B#-A$(Z
+P TO ): LET A#-A$( TO Z-P): RET
URN
880 NEXT Z: RETURN
900 GO SUB 9500: CLS
908 LET N=0: LET S=0: LET E=0:
LET W=0: LET U=0: LET D=0
999 GO SUB LOC+3: PRINT "SAIDA
S: "; ("NORTE " AND N); ("SUL " AN
D S); ("ESTE " AND E); ("OESTE" AN
D U): GO SUB 800
1000 INPUT INK 5: "QUE DEVO FAZER
?" : LINE A#: IF A#="" THEN GO T
O C
1005 GO SUB 850: PRINT INK 6; A#:
" "; B#: PRINT
1007 IF LOC=4 AND AND>.88 THEN P
RINT "O GIGANTE SALTOU SOBRE MIM
E ES-MAGOU-ME O CRAMEO, MORRI."
: GO TO 9000
1008 IF LOC=39 AND NOT DOG AND R
ND>.9 THEN PRINT "O CAO ATACOU-M
E E MATOU-ME." : GO TO 9000
1010 IF A#="N" AND N THEN LET LO
C=LOC-7: GO TO C-P-P
1020 IF A#="S" AND S THEN LET LO
C=LOC+7: GO TO C-P-P
1030 IF A#="E" AND E THEN LET LO
C=LOC+P: GO TO C-P-P
1040 IF A#="O" AND U THEN LET LO
C=LOC-P: GO TO C-P-P
1045 IF A#="O" OR A#="E" OR A#="
N" OR A#="S" THEN PRINT "NAO POS
SO IR NESTA DIRECCAO." : GO TO C
1050 IF A#="I" THEN GO SUB 2000:
GO TO C
1060 IF A#="OLHAR" THEN CLS : GO
TO C-P-P
1070 IF A#="DESCER" THEN GO TO 3
000
1080 IF A#="SUBIR" THEN GO TO 31
00
1090 IF A#="APANHAR" OR A#="AGAR
RAR" THEN GO TO 2100
1100 IF A#="LARGAR" THEN GO TO 2
200
1110 IF A#="MATAR" THEN GO TO 23
00

```

```

1120 IF A$="EMPURRAR" THEN GO TO 2400
1130 IF A$="ATRAVESSAR" THEN GO TO 2500
1140 IF A$="ABRIR" THEN GO TO 2600
1150 IF A$="ALIMENTAR" THEN GO TO 2700
1160 IF A$="COMPRAR" THEN GO TO 2800
1170 IF A$="GRAVAR" THEN SAVE "AVENTURA" LINE 998: PRINT "VERIFI CAR...": VERIFY "AVENTURA": GO TO C
1180 IF A$="DESISTIR" THEN GO TO 9970
1190 IF A$="SOCORRO" THEN PRINT "O GIGANTE NAO ME DEIXARA SAIR, MESMO QUE EU SEJA BEM EDUCADO,": GO TO C
1200 IF A$="CAVAR" THEN GO TO 2900
1300 PRINT "NAO COMPREENDO,": GO TO C
2000 PRINT "TENHO O SEGUINTE...": LET FL=0: FOR Z=1 TO 9: IF O(Z)=0 THEN PRINT O$(Z): LET FL=P
2010 NEXT Z: IF NOT FL THEN PRINT "NADA."
2020 RETURN
2100 IF B$="" THEN INPUT INK 5: (A$); " COMO ??": LINE B$: GO TO 2100
2105 IF OB=4 THEN PRINT "NAO POS SO TRANSPORTAR MAIS,": GO TO C
2110 FOR Z=P TO 9: IF O$(Z) (TO LEN B$)=B$ AND O(Z)=LOC THEN GO TO 2130
2120 NEXT Z: PRINT "NAO POSSO FAZER ISSO,": GO TO C
2130 IF B$="OSSO DE CAO" AND LOC=5 THEN PRINT "ISSO SERIA ROUBO!": GO TO C
2140 PRINT "OK...": LET O(Z)=0: LET OB=OB+P: GO TO C
2200 IF B$="" THEN INPUT INK 5: "QUE DEVO EU "; (A$); "??": LINE B$: GO TO 2200
2210 FOR Z=P TO 9: IF O$(Z) (TO LEN B$)=B$ AND O(Z)=0 THEN PRINT "OK...": LET O(Z)=LOC: LET OB=OB+P: GO TO C

```

```

2220 NEXT Z: PRINT "NAO TENHO "; B$; "!": GO TO C
2300 IF B$="" THEN INPUT INK 5: "MATAR QUEM?": LINE B$: GO TO 2300
2310 IF B$(">")"GIGANTE" THEN PRINT "NAO SOU UM ASSASSINO!": GO TO C
2315 IF LOC<>4 THEN PRINT "NAO U EUJO NENHUM GIGANTE!": GO TO C
2320 IF O(9) THEN PRINT "NAO TENHO NENHUMA ARMA PARA MATAR O GIGANTE,": GO TO C
2330 PRINT "OPTIMO, MATAMOS O GIGANTE E CON-SEGUIMOS A LIBERDADE,": GO TO 9980
2400 IF B$="" THEN INPUT INK 5: "QUE DEVO EMPURRAR?": LINE B$: GO TO 2400
2410 IF B$(">")"PORTA" THEN PRINT "NAO POSSO FAZER ISSO,": GO TO C
2420 IF LOC<>39 THEN PRINT "QUE PORTA?": GO TO C
2430 IF NOT DO THEN PRINT "ESTA FECHADA A CHAVE,": GO TO C
2440 PRINT "OK...": LET LOC=LOC+P: GO TO C-P-P
2500 IF B$="" THEN INPUT INK 5: "QUE QUER ATRAVESSAR?": LINE B$: GO TO 2500
2505 IF B$(">")"RIO" THEN PRINT "NAO SOU ASSIM TAO ESPERTO!": GO TO C
2507 IF LOC<>31 AND LOC<>32 THEN PRINT "SE ME MOSTRAR O RIO, EU TENTO...": GO TO C
2510 IF O(2) THEN PRINT "VOU PRECISAR DE QUALQUER COISA PARA ME AJUDAR,": GO TO C
2520 PRINT "OK...": LET LOC=LOC+(LOC=31)-(LOC=32): GO TO C-P-P
2530 IF B$="" THEN INPUT INK 5: "POR FAVOR CONTINUE A ORDEM,": LINE B$: GO TO 2500
2505 IF B$(">")"PORTA" THEN PRINT "TALVEZ VA LEVAR ALGUM TEMPO,": GO TO C
2610 IF LOC<>39 THEN PRINT "NAO ESTOU A VER PORTA NENHUMA!": GO TO C
2620 IF DO THEN PRINT "IRRA! JA ESTA ABERTA!": GO TO C

```

```

2625 IF O(4) THEN PRINT "NAO TEN
HO CHAVE...": GO TO C
2630 PRINT "OK...": LET DO=P: GO
TO C
2700 IF B#(">")"CAO" THEN PRINT "NA
O ME PARECE QUE VALHA A PENHA...":
GO TO C
2710 IF LOC<>32 THEN PRINT "QUAL
CAO?": GO TO C
2720 IF O(P) THEN PRINT "COM O O
US? OS DEDOS?": GO TO C
2730 PRINT "O CAO ADORMECEU...":
LET DOG=P: LET S=P: LET O(P)=LOC
: LET OB=OB-P: GO TO C
2800 IF O(P) THEN PRINT "NAO TEN
HO DINHEIRO...": GO TO C
2810 IF B#="" THEN INPUT INK 5:"
ALGUMA COISA EM ESPECIAL?": LINE
B#: GO TO 2810
2820 IF B#="NAO" THEN PRINT "MAU
! JA ESTOU A PERDER A MINHA PAC
IENCIA!": GO TO C
2830 IF B#(">")"OSSO DE CAO" THEN P
RINT "PARA QUE SE E GRATIS?": G
O TO C
2840 IF LOC<>6 THEN PRINT "NAO H
A AQUI NENHUM OSSO...": GO TO C
2845 IF OB=4 THEN PRINT "NAO POS
SO TRANSPORTAR MAIS NADA...": GO T
O C
2850 PRINT "OK...": LET O(P)=0:
LET OB=OB+P: GO TO C
2900 IF LOC<>20 THEN PRINT "AIND
A NAO POSSO FAZER ISSO...": GO TO
C
2910 IF O(4)<>LOC THEN PRINT "NA
O HA NADA AQUI...": GO TO C
2920 PRINT "DESCOBI UMA CHAVE...":
GO TO C
3000 IF LOC<>25 THEN PRINT "PARA
ONDE?": GO TO C
3010 IF O(5) THEN PRINT "VOU PAR
TIR O PESCOÇO!": GO TO C
3020 PRINT "OK...": LET LOC=LOC-
P: GO TO C-P-P
3100 IF LOC<>18 THEN PRINT "AIND
A NAO POSSO FAZER ISSO...": GO TO
C
3110 IF O(5) THEN PRINT "COMO ME
TROUXE PARA AQUI? PODE -ME EXP
LICAR?": GO TO C
3120 PRINT "OK...": LET LOC=LOC+

```

```

F: GO TO C-P-P
9970 CLS : INPUT "TEM A CERTEZA
(S/N)? ": LINE A#
9980 IF A#="S" OR A#="SIM" THEN
GO TO 9980
9990 GO TO C
9990 INPUT "QUER OUTRA AVENTURA
(S/N)? ": LINE A#
9910 IF A#="S" OR A#="SIM" THEN
RUN 900
9920 IF A#="N" OR A#="NAO" THEN
STOP
9930 GO TO 9900
9940 LET O=PI-PI: LET P=PI/PI
9910 LET F=VAL "7": LET LOC=VAL
"24": LET OB=0
9920 LET DO=0: LET DOG=0: LET C=
VAL "1000": DIM O$(9,11): DIM O(
9)
9930 FOR Z=P TO INT (PI#PI): REA
D O$(Z),Y#: LET O(Z)=VAL Y#
9940 NEXT Z
9950 DATA "OSSO DE CAO","S","TAB
UA","11","ANEL","14","CHAVE","20
","CORDA","23","PEDRA","28","BOL
SA","30","LIVRO","39","PISTOLA",
"40"
9960 RETURN

```

Apresentamos agora o carregador de A CIDADE:

```

10 REM A Cidade
20 REM Um Jogo de Aventura
30 REM De Neil Pallinacci 1983
40 REM Spectrum 16 K
50 REM
60 REM
70 BORDER 0: PAPER 0: INK 7: B
RIGHT 1
80 CLEAR 32767: CLS
90 POKE 23609,100: POKE 23658,
0: POKE 23624,71
95 GO SUB 200
100 LOAD ""
200 DIM A$(32): PAPER 6: INK 0:
LET A#="" : PARE O GRAVADOR
: LET L=9: GO SUB 900
210 PAUSE 100: PAPER 0: CLS
220 LET A#="" : A CIDAD
E: LET L=3: INK 6: GO SUB 900
230 LET A#="UMA AVENTURA PARA O
ZX SPECTRUM...": LET L=5: GO SUB

```



```

A MINHA FRENTE, A RUA CONTINUA
PARA LESTE." : LET E=P: LET DOI=0
: RETURN
24 PRINT "ESTOU NUMA RUA DESE
RTA, ONDE VEJO UM TAXI, A SUL
FICA O MER- CADO." : LET E=P: LE
T U=P: LET S=P: RETURN
27 PRINT "ESTOU NO INTERIOR D
E UMA GRANDE LOJA, NAO VEJO NING
UEM, EXISTE UM CAFE PARA LESTE,
" : LET N=P: LET S=P: LET U=P: LE
T E=P: RETURN
30 PRINT "ESTOU NUM PEQUENO C
AFE, DESERTO, A LOJA GRANDE FICA
A OESTE." : LET U=P: RETURN
33 PRINT "ESTOU NO INTERIOR D
O BANCO, NAO OICO QUALQUER RUIDO
" : IF 60 THEN PRINT "VEJO UM
GUARDA DE COSTAS PARA MIM,"
37 PRINT : LET N=P: LET S=NOT
60: RETURN
39 PRINT "ESTOU NUM MERCADO D
E RUA, A ESTE VEJO UMA ESQUADRA,
E UM LOCAL DE CONSTRUCAO A SUL,"
40 LET N=P: LET S=P: LET E=P
RETURN
42 PRINT "CHEGUEI A ESQUADRA
, NAO ESTA DESERTA,"
43 IF NOT 0(7) THEN PRINT "UM
POLICIA VE O DINHEIRO E DA-ME O
EM DE PRISAO." : GO TO 9000
44 PRINT "UM POLICIA TIRA-ME O
QUE RECOLHE COLOCA TUDO NOS SI
TIOS ONDE ESTAVA." : LET N=P: L
ET U=P: GO TO 700
45 PRINT "ESTOU NUMA PEQUENA
ESTACAO DE METRO, VEJO UM BURR
CO NA PAREDE A SUL." : LET S=P: R
ETURN
48 PRINT "ESTOU JUNTO DA PORT
A DO COFRE, QUE ESTA FECHADA,"
49 IF VC THEN PRINT "EXISTE UM
BURACO NELA,"
50 LET E=P: LET S=VC: RETURN
51 PRINT "VEJO DUAS PORTAS, U
MA A OESTE, OUTRA A SUL, A ENTR
ADA DO COFRE ESTA A NORTE," : LE
T N=P: LET S=P: LET U=P: RETURN
54 PRINT "ESTOU NUM LOCAL DE
CONSTRUCAO, RODEADO DE MUROS AL
TOS." : LET N=P: RETURN
57 PRINT "ESTOU NO HALL DE UM

```

```

GRANDE HOTELE VEJO A MINHA FRE
NTE A RECEP- CAO." : LET E=P: RET
URN
60 PRINT "ESTOU NA CAVE DE UM
GRANDE EDI- FICIO." : LET N=P: L
ET U=P: RETURN
63 PRINT "ESTOU NO INTERIOR D
O COFRE, PARATINGIR O DINHEIRO,
DEVO INDICAR CODIGO, EXISTEM 9
BOTOES, E TENHO DE CARREGAR N
O CORRECTO,"
64 LET N=P: RETURN
66 PRINT "SURTIU UMA PAREDE A
TRAS DE MIM, FUI APANHADO." : GO
TO 9000
78 PRINT "ESTOU NO INTERIOR D
A PRINCIPAL SALA DO COFRE." : LE
T N=P: RETURN
800 IF A$(P)=" " THEN LET A$=A$
(2 TO ) : GO TO 600
810 IF A$(LEN A$)=" " THEN LET
A$=A$( TO LEN A$-P) : GO TO 810
820 LET S$="" : FOR Z=P TO LEN A
$ : IF A$(Z)=" " THEN LET B$=A$(Z
+P TO ) : LET A$=A$( TO Z-P) : RET
URN
830 NEXT Z: RETURN
700 RESTORE 8000: FOR Z=P TO 7:
READ A$,B$: IF 0(Z)=0 THEN LET
0(Z)=VAL B$
710 NEXT Z: LET 08=0: RETURN
800 PRINT "SAIDAS:" : ("NORTE "
AND N); ("SUL " AND S); ("ESTE "
AND E); ("OESTE " AND U)
810 PRINT "VEJO..."
820 LET FL=0: FOR Z=P TO 7: IF
0(Z)=LOC THEN PRINT 0$(Z) : LET F
L=P
830 NEXT Z: IF NOT FL THEN PRIN
T "NADA DE INTERESSANTE,"
840 RETURN
900 GO SUB 8000: CLS
999 LET N=0: LET S=0: LET E=0:
LET U=0: GO SUB LOC*3: GO SUB 80
0
1000 INPUT INK 5: "QUE DEVO FAZER
?" : LINE A$: IF A$="" THEN GO TO
0
1010 IF LOC=12 AND AND >.9 AND 60
THEN PRINT "O GUARDA VIROU-SE,
VIU-ME E DIS-PAROU SOBRE MIM, NO
ARI." : GO TO 9000

```

```

1050 GO SUB 600: PRINT " INK 5)A
#:"; B$
1060 IF A#="N" AND N THEN LET LO
C=LOC-F: GO TO C-P
1070 IF A#="S" AND S THEN LET LO
C=LOC+F: GO TO C-P
1080 IF A#="E" AND E THEN LET LO
C=LOC+P: GO TO C-P
1090 IF A#="O" AND U THEN LET LO
C=LOC-P: GO TO C-P
1100 IF A#="N" OR A#="S" OR A#="
E" OR A#="O" THEN PRINT "NAO POS
SO IR NESTA DIRECCAO.": GO TO C
1110 IF A#="APANHAR" OR A#="ABAR
RAR" THEN GO TO 2000
1120 IF A#="LARGAR" THEN GO TO 2
000
1130 IF A#="I" THEN GO SUB 2200:
GO TO C
1140 IF A#="OLHAR" THEN CLS: GO
TO C-P
1150 IF A#="SOCORRO" THEN GO TO
2300
1160 IF A#="ABORDAR" OR A#="PERS
EGUIR" THEN GO TO 2400
1170 IF A#="ABRIR" THEN GO TO 25
00
1180 IF A#="EMPURRAR" THEN GO TO
2600
1190 IF A#="DESCER" THEN GO TO 2
700
1200 IF A#="DESISTIR" OR A#="PAR
AR" THEN GO TO 2800
1210 IF A#="LER" OR A#="EXAMINAR
" THEN GO TO 2800
1220 IF A#="MATAR" THEN GO TO 29
00
1230 IF A#="CORTAR" THEN GO TO 3
000
1240 IF A#="CARREGAR" OR A#="PRE
MIR" THEN GO TO 3100
1250 IF A#="GRAVAR" THEN SAVE "C
IDADE" LINE 999: PRINT "VERIFICA
R...": VERIFY "CIDADE": GO TO C
1260 IF A#="DIZER" THEN GO TO 32
00
1300 PRINT "NAO COMPREENDO.": GO
TO C
2000 IF B#="" THEN INPUT INK 5)"
APANHAR O QUE ??": LINE B$: GO T
O 2000
2002 IF LEN B#>13 THEN PRINT "NA

```

```

O POSSO": GO TO C
2005 IF OB=3 THEN PRINT "NAO POS
SO TRANSPORTAR MAIS NADA.": GO T
O C
2010 FOR Z=P TO 7: IF O$(Z) ( TO
LEN B#)=B# AND O(Z)=LOC THEN LET
O(Z)=0: LET OB=OB+1: PRINT "OK.
...": GO TO C
2020 NEXT Z: PRINT "NAO VEJO "; B
#;"!": GO TO C
2100 IF B#="" THEN PRINT A#;" O
QUE?": GO TO C
2110 IF LEN B#>13 THEN PRINT "NA
O POSSO.": GO TO C
2120 FOR Z=P TO 7: IF O$(Z) ( TO
LEN B#)=B# THEN PRINT "OK...": L
ET O(Z)=LOC: LET OB=OB-1: GO TO
C
2130 NEXT Z: PRINT "NAO TENHO.":
B#;"!": GO TO C
2200 PRINT "TENHO O SEGUINTE...":
LET FL=0: FOR Z=P TO 7: IF O(Z
)=0 THEN PRINT O$(Z): LET FL=P
2210 NEXT Z: IF NOT FL THEN PRIN
T "NADA."
2220 GO TO C
2300 IF LOC=17 THEN PRINT "PERGU
NTE A POPULACAO...": GO TO C
2310 PRINT "SIGA A ESTRADA NACIO
NAL NO. 2...": GO TO C
2400 IF B#="" THEN PRINT A#;" O
QUE?": GO TO C
2410 IF B#<>"COMBOIO" AND B#<>"T
AXI" THEN PRINT "NAO POSSO.": GO
TO C
2420 IF B#="COMBOIO" AND LOC<>3
AND LOC<>15 THEN GO TO 2490
2430 IF B#="TAXI" AND LOC<>3 AND
LOC<>8 THEN GO TO 2490
2440 IF B#="COMBOIO" THEN GO TO
2470
2450 IF O(2) THEN PRINT "NAO TEN
HO DINHEIRO SUFICIENTE.": GO TO
C
2460 PRINT "O TAXI PARTE COMIGO
DENTRO.": LET LOC=LOC+(F AND LOC
=3)-(F AND LOC=8): GO TO C-P
2470 IF O(3) THEN PRINT "PRECISO
DE BILHETE!": GO TO C
2480 PRINT "ENTRA E O COMBOIO PA
RTE.": LET LOC=(3 AND LOC=15)+(1

```

```

5 AND LOC=3): GO TO C-P
2490 PRINT "AINDA NAO POSSO FAZER ISSO.": GO TO C
2500 IF B#="" THEN PRINT "QUE DE VO ABRIR?": GO TO C
2510 IF B#<>"PORTA" THEN PRINT "NAO POSSO FAZER ISSO.": GO TO C
2520 IF LOC<>7 THEN PRINT "QUAL PORTA?": GO TO C
2530 IF DO THEN PRINT "IARR! JA ESTA ABERTA!": GO TO C
2540 IF O(5) THEN PRINT "PRECISO DA CHAVE.": GO TO C
2550 PRINT "OK,..": LET DO=P: GO TO C
2560 IF B#="" THEN PRINT "EMPURRA O QUE?": GO TO C
2570 IF B#<>"PORTA" THEN PRINT "NAO POSSO,..": GO TO C
2580 IF LOC<>7 THEN PRINT "NAO VEUJO NENHUMA PORTA,..": GO TO C
2590 IF NOT DO THEN PRINT "ESTA FECHADA A CHAVE.": GO TO C
2640 LET DO1=P: PRINT "OK,..": VEJO UMAS ESCADAS PARA BAIXO,..: GO TO C
2700 IF LOC<>7 THEN GO TO 2490
2710 IF NOT DO1 THEN PRINT "HA UMA PORTA PARA ATRAVESSAR!": GO TO C
2720 PRINT "OK,..": LET LOC=LOC+7: GO TO C-P
2800 IF B#="" THEN INPUT INK 5; (A#); "O QUE?": LINE B#: GO TO 2800
2810 IF B#<>"PISTA" THEN PRINT "NAO POSSO FAZER ISSO!": GO TO C
2820 IF O(6) THEN PRINT "NAO TENHO NENHUMA PISTA.": GO TO C
2830 PRINT "A PISTA ESTA ESCRITA NUMA FOLHA DE PAPEL A4. DIZ: QUEM CONDUZ NAO BEBE.": GO TO C
2890 IF B#="" THEN PRINT "MATAR QUEM?": GO TO C
2910 IF B#<>"GUARDA" THEN PRINT "RECUSO-ME POR RAZOES MORAIS.": GO TO C
2920 IF LOC<>12 THEN PRINT "NAO VEJO O GUARDA.": GO TO C
2930 IF O(P) THEN PRINT "NAO TENHO ARMA.": GO TO C
2940 PRINT "OK,..": LET GU=0: GO

```

```

TO C
3000 IF B#="" THEN INPUT INK 5; "QUE DEVO CORTAR?": LINE B#: GO TO 3000
3010 IF B#<>"PORTA DO COFRE" THEN PRINT "NAO POSSO FAZER ISSO.": GO TO C
3020 IF LOC<>15 THEN PRINT "NAO VEJO A PORTA DO COFRE.": GO TO C
3030 IF O(4) THEN PRINT "NAO TENHO NADA COM QUE CORTAR.": GO TO C
3040 PRINT "OK,..": LET UC=P: GO TO C-P
3100 IF LOC<>21 THEN PRINT "NAO VEJO NADA PARA "; A#; "!": GO TO C
3110 IF LEN B#<>P THEN PRINT "PRECISO DE UM NUMERO ENTRE 1 E 9": GO TO C
3120 IF B#<>"1" OR B#>"9" THEN PRINT "ISSO NAO E UM NUMERO!": GO TO C
3130 LET Z=VAL B#: IF Z<>3 THEN LET LOC=LOC+P: GO TO C-P
3140 LET LOC=LOC+P: PRINT "O PORTAO DE FERRO DESLIZA LENTA-MENTE,..": GO TO C-P
3200 IF B#="" THEN PRINT "DIZER O QUE?": GO TO C
3210 IF B#<>"PALAVRA DE PASSE" THEN PRINT "NAO POSSO DIZER ISSO": GO TO C
3220 IF LOC=15 AND NOT O(7) THEN PRINT "ENCONTREI O CONTACTO, E TERMINARIA MISSAO, O CONTACTO ERA O CON-DUTOR DO COMBOIO.": GO TO 9000
3225 IF LOC=3 OR LOC=8 THEN PRINT "O CONDUTOR DO TAXI DIZ ALGO BASTANTE DESAGRADAVEL.": GO TO C
3230 PRINT "NINGUEM PARECE INTERESSAR-SE.": GO TO C
3000 LET O=PI-PI: LET P=NOT O
3010 LET F=VAL "5": LET LOC=VAL "13"
3020 LET OB=0: LET DO=0: LET GU=P: LET UC=0: LET C=VAL "1000"
3030 DIM O(7): DIM O$(7,17)
3040 RESTORE 8000: FOR Z=P TO 7: READ O$(Z),A#: LET O(Z)=VAL A#: NEXT Z
8050 DATA "FACA","4","BOLSA","9"

```

```

"BILHETE", "10", "LANTERNA", "18",
"CHAVE", "19", "PISTA ESCRITA", "20",
"MAIA COM DINHEIRO", "25",
8060 PRINT TAB 11; INK 5; "A CI
DADE" ESCRITA POR NEIL PELLI
NACCI" CARREGUE EM QUALQ
UER TECLA"
8070 PAUSE P: PAUSE 0: FOR Z=50
TO -20 STEP -5: BEEP .01,Z: BEEP
.01,Z-3: BEEP .01,Z+6: NEXT Z
8080 RETURN
8900 INPUT "TEM A CERTEZA (S/N)?
"; LINE A$: IF A$="" THEN GO TO
8900
8910 IF A$(P)="S" THEN GO TO 900
0
8920 GO TO C
9000 INPUT "QUER NOVA AVENTURA (
S/N)? "; LINE A$: IF A$="" THEN
GO TO 9000
9010 IF A$(P)="N" THEN STOP
9020 RUN 900

```

PORTAS DO DESTINO

Trata-se de uma obra-prima, escrita por Malcolm Young, que recompensará plenamente o tempo que você gastará a introduzir o programa na máquina.

Esta aventura gráfica baseia-se na rotina de labirinto que ocupa as linhas 1000 a 1400. A rotina que se inicia na linha 2000 trata os gráficos.

Quando Malcolm começou a escrever este programa, pensava simplesmente construir um programa que apresentasse um labirinto a três dimensões. No entanto, quando o terminou resolveu que poderia torná-lo muito mais interessante, e o resultado está aqui para ser visto. São incluídas todas as instruções necessárias. Um aviso: o programa produz labirintos bastante difíceis de resolver. Não use este programa se quiser um jogo que possa dominar numa hora ou pouco mais.

```

2 REM © 1983, Malcolm Young
10 DEF FN r(x)=INT (RND*x)+1
15 LET set=0
30 GO SUB 9000: REM instrucce
S
40 BORDER 4: PAPER 5: INK 0: C

```

```

L8
50 GO SUB 1000: REM construir
labirinto
60 GO SUB 8000: REM inicializa
cao
70 GO TO 200
80 NEXT a
90 REM rotina principal
100 LET k#=INKEY#: IF k#="" THE
N GO TO 100
110 LET f=f+(k#="d")-(k#="e")+2
*(k#="r")
120 LET f=f+4*(f<1)-4*(f>4)
130 IF k#="s" THEN GO SUB 5000
135 IF k#="a" THEN LET ##(py,px
)="": LET l(11)=l(11)-l(13): I
F l(11)<0 THEN GO TO 500
140 LET px=px+((k#="e")*((f=2)-
(f=4)))
150 LET py=py+((k#="a")*((f=3)-
(f=1)))
155 IF ##(py,px)="#" THEN LET s
w=1: LET px=px-(f=2)+(f=4): LET
py=py-(f=3)+(f=1)
160 IF ##(py,px)=CHR# 128 THEN
GO SUB 6000
170 IF ##(py,px)>"," THEN GO TO
2000
175 IF g1=py AND g2=px THEN GO
TO 800
180 IF k#="p" THEN LET re=1-re:
LET k#="a": GO TO 140
190 IF k#="l" THEN GO SUB 7000
200 IF l(7) AND k#="x" THEN GO
SUB 7200
210 IF l(8) AND k#="t" THEN GO
SUB 5500
220 IF k#="s" THEN INPUT "Quer
gravar o jogo? "; LINE a$: IF a$
="s" THEN GO SUB 7500: CLS
230 CLS
240 PRINT #0; INVERSE 1;"NORTE"
AND f=1;"ESTE" AND f=2;"SUL" AN
D f=3;"OESTE" AND f=4; INVERSE 0
;" " FLASH 1;"REPETIR" AND re;
FLASH 0;TAB 15;"Energia=";l(11)
250 BEEP .1,10: GO SUB 2000
260 IF INKEY#="" AND (re#0) THE
N LET k#="a": GO TO 135
270 IF sw THEN PRINT AT 10,11;
FLASH 1;"MURO SOLIDO": BEEP 1,-1

```

```

0: LET sw=NOT sw
0000 GO TO 100
0010 REM fim do Jogo
0020 PRINT AT 10,0;"Voce morreu
de fome..."
0030 PRINT "Os seus ossos junta
-se aos de muitos outros."
0040 GO TO 4530
0050 REM final
0060 PAPER 0: INK 7: CLS
0070 GO SUB 5150: PRINT AT 4,0
0080 FOR i=40 TO 45 STEP .5: SEE
P
25,i: PRINT "Optimo! Consegui
UI:"
NEXT i
0090 FLASH 1
0100 PRINT INK 6;"Executou a sua
missao com"
0110 INK 4
0120 IF l(10)>200 THEN PRINT "er
a de capacidade."
0130 IF l(10)<200 AND l(10)>100
THEN PRINT "bastante merito."
0140 IF l(10)<100 THEN PRINT "co
nsideravel sorte."
0150 GO TO 4842
0160 REM construir labirinto
0170 PAPER 0: INK 5: CLS : PRINT
"O labirinto pode levar bastant
e tempo a ser definido, pelo que
convem guardar os maiores em
cassette."
0180 PRINT "por favor indique
as dimensoes do labirinto prete
ndido, com umalargura entre 30 e
100 e um com-primento entre 20
e 100"
0190 INPUT "Quer carregar um lab
irinto s/n? "; LINE a#
0200 IF a#="s" THEN GO SUB 7500:
RETURN
0210 INPUT "Indique largura: ";w
: LET w=INT (w/2): IF w<15 OR w>
50 THEN GO TO 0210
0220 INPUT "Indique comprimento:
";l: LET l=INT (l/2): IF l<10 O
R l>50 THEN GO TO 0220
0230 LET s=INT (w*(V3)): LET s=IN
T (s+s/10): DIM l(s): DIM d(s)
0240 LET s=s+INT (RND*s/3): LET
t=INT (s/60): LET t1=(s-t*60)/10
0: LET t=t+t1

```

```

1033 POKE 23674,0: POKE 23673,0:
POKE 23672,0
1035 PRINT "Demora aproximada
mente ";t,"minutos."
1040 LET m=w*2+1: LET n=l*2+1: D
IM m$(n,m)
1050 FOR i=1 TO m: LET m$(1,i)="#"
: LET m$(n,i)="#" : NEXT i
1060 FOR i=2 TO n-1: LET m$(i,1)
="#" : LET m$(i,m)="#" : NEXT i
1070 LET nn=2+INT (RND*(l-2))+3
1080 LET mm=2+INT (RND*(w-2))+3
1090 LET m$(nn,mm)="#" : LET k=0
1100 IF m$(nn-2,mm)="#" THEN GO
TO 1210
1110 IF m$(nn+2,mm)="#" THEN GO
TO 1210
1120 IF m$(nn,mm-2)="#" THEN GO
TO 1210
1130 IF m$(nn,mm+2)="#" THEN GO
TO 1210
1140 IF k-1=0 THEN GO TO 1420
1150 IF d(k)=1 THEN LET nn=nn+l(
k)
1160 IF d(k)=2 THEN LET nn=nn-l(
k)
1170 IF d(k)=3 THEN LET mm=mm+l(
k)
1180 IF d(k)=4 THEN LET mm=mm-l(
k)
1190 LET k=k-1
1200 GO TO 1100
1210 LET l1=2+INT (RND*3)+2
1220 IF l1=8 THEN GO TO 1210
1230 LET d1=INT (RND*4)+1
1240 IF d1>2 THEN GO TO 1230
1250 LET s=-1+2*(d1=2)
1260 LET t=0
1270 GO TO 1300
1280 LET s=0
1290 LET t=-1+2*(d1=4)
1300 FOR i=2 TO l1 STEP 2
1310 IF m$(nn+s*i,mm+t*i)="#" TH
EN GO TO 1210
1320 NEXT i
1330 FOR i=1 TO l1
1340 LET m$(nn+s*i,mm+t*i)="#"
1350 NEXT i
1360 LET nn=nn+s*t1
1370 LET mm=mm+t*(l1
1380 LET k=k+1
1390 LET l(k)=l1

```

```

1400 LET d(k)=d1
1410 GO TO 1100
1420 REM no. de portas
1425 LET nd=INT ((l(4)+l(5)/25):
LET st=nd/14
1430 LET l(4)=m: LET l(5)=n
1435 REM esconder objectos
1440 FOR o=1 TO 4
1450 FOR i=1 TO st
1460 LET l=FN r((l(5)-3)+1): LET c
=FN r((l(4)-3)+1): LET kn=FN r(5)*
10+o
1470 IF m$(l,c)<>"#" THEN GO TO
1480 LET m$(l,c)=CHR$(48+kn)
1490 NEXT i
1500 NEXT o
1510 REM esconder alimentos e vi
loes
1520 FOR o=5 TO 7 STEP 2
1530 FOR i=1 TO st*3
1540 LET l=FN r((l(5)-3)+1): LET c
=FN r((l(4)-3)+1): LET kn=FN r(5)*
10+o
1550 IF m$(l,c)<>"#" THEN GO TO
1560 LET m$(l,c)=CHR$(48+kn)
1570 NEXT i: NEXT o
1580 REM esconder dinheiro
1590 FOR i=1 TO st*4
1600 LET l=FN r((l(5)-3)+1): LET c
=FN r((l(4)-3)+1): LET kn=FN r(5)*
10+o
1610 IF m$(l,c)<>"#" THEN GO TO
1620 LET m$(l,c)=CHR$(48+kn)
1630 NEXT i
1640 LET m$(l(5)/2,(l(4)/2)=CHR$(
48+FN r(5)*10+8)
1650 REM esconder chaves
1660 FOR i=1 TO 4*st
1670 LET l=FN r((l(5)-4)+2): LET c
=FN r((l(4)-4)+2)
1680 IF m$(l,c)<>" " THEN GO TO
1690 LET m$(l,c)=CHR$ 128
1700 NEXT i
1705 LET set=1
1710 LET time=(PEEK 23672+255*PE
EK 23673+65536*PEEK 23674)
1720 LET time=time/50: BEEP 2,10
1730 INPUT "O labirinto demorou

```

```

";(time);" segundos";"a construi
r) quer grava-lo? ";a#
1740 IF a$(1)="s" THEN GO SUB 75
00
1750 CLS : RETURN
2000 REM image#
2010 BORDER 1: PAPER 0: BRIGHT 1
2020 LET x=24: LET y=16: REM ang
ulo
2030 FOR i=1 TO 6
2040 LET ly=py+((f=3)-(f=1))*i
2050 LET lx=px+((f=2)-(f=4))*i
2060 IF m$(ly,lx)<"#" OR m$(ly,l
x)=CHR$ 128 THEN NEXT i
2065 LET dt=(m$(ly,lx)=" ")
2070 LET vw=i-1-(i>5): REM maxim
a distancia de vista
2075 IF NOT vw AND m$(ly,lx)>"#"
THEN GO SUB 2500

```

```

2080 LET wx=vw*x: LET wy=vw*y
2090 FOR d=0 TO vw: REM divisao
de paineis
2100 LET h=255-d*x*2
2110 PLOT d*x,d*y: DRAW 0,175-(y
*d)*2: IF d=vw THEN DRAW h,0
2120 PLOT 255-d*x,175-d*y: DRAW
0,-(175-y*2*d): DRAW -h,0
2125 PLOT 254-d*x,175-d*y: DRAW
0,-(175-y*2*d)
2130 NEXT d
2140 FOR i=0 TO vw-1: REM desenh
ar muros
2150 LET ly=py+((f=3)-(f=1))*(i+
1)
2160 LET lx=px+((f=2)-(f=4))*(i+
1)
2170 LET d$=m$(ly-(f=2)+(f=4),lx
-(f=1)+(f=3))
2175 LET lwall=(d$="#"): LET ldr
=(d$>"#")
2180 LET d$=m$(ly+(f=2)-(f=4),lx
+(f=1)-(f=3))
2190 LET rwall=(d$="#"): LET rdr
=(d$>"#")
2195 IF (ldr+rdr) THEN GO SUB 23
90
2197 GO TO ldr*7+2200
2200 PLOT i*x,i*y+y*NOT lwall: D
RAW x,y*(lwall)
2205 PLOT i*x,175-(i*y+y*NOT lwa

```

```

2200 (1): DRAW X,-(y#lwall)
2207 GO TO rdr#20+2210
2210 PLOT 255-i*x,i*y+4*NOT rwal
(1: DRAW -x,y*(rwall)
2220 PLOT 2000-i*x,175-(i*y+y*NOT
rwal(1): DRAW -x,-(y*rwal(1)
2230 NEXT i
2240 INK 5: LET p=5
2250 LET v=250/p: LET e=(255-2*v*
x)/p
2260 FOR l=1 TO p-1
2270 LET h=(v*x+e*(1-v)*l
2280 PLOT v*l,0: DRAW h,w*y
2300 NEXT l
2310 IF 4t THEN PRINT FLASH 1: 0
VER 1:AT 10,15;"AB":AT 11,15;"CD
": REM graficos A, B, C e D
2320 IF NOT re THEN RETURN
2325 LET r=1
2330 REM verificar repeticao
2340 FOR i=0 TO 1
2350 LET r0=py+((f=3)-(f=1))*i:
LET rx=px+((f=2)-(f=4))*i
2360 LET r1=r*(r0+(y+(f=2)-(f=4)
rx+(f=3)-(f=1))="#")
2370 LET r2=r*(r0+(y-(f=2)+(f=4)
rx-(f=3)+(f=1))="#")
2380 NEXT i: RETURN
2390 REM desenhar porta
2400 LET k=1/(i+1)*5
2410 FOR j=1 TO 7
2420 IF ldr THEN PLOT i*x+j*3,i*
y+j*2
2425 IF ldr THEN DRAW INK 5;0,17
5-(i*y+j*2)*2
2430 IF rdr THEN PLOT 255-(i*x+j
*3),i*y+j*2
2440 IF rdr THEN DRAW INK 5;0,17
5-(i*y+j*2)*2
2450 NEXT j
2460 RETURN
2480 PLOT 100,100+k: DRAW k,0,-P
I*1.75
2490 DRAW k/2.5,-k*1.5: DRAW -k/
2.5*2-k,0,-PI: DRAW k/2.5,k*1.5
2491 RETURN
2500 REM porta
2510 PAPER 6: INK 0: CLS
2520 PRINT AT 10,4;"B PORTA": SE
EP .5,10
2530 LPRINT 0: INK 5
2540 RETURN

```

```

2600 REM
2610 IF l(12) THEN GO TO 3000
2620 PRINT AT 2,4: FLASH 1;"Se#
chave, nao entra..."
2630 LET x=px-(f=2)+(f=4): LET y
y=py-(f=3)+(f=1)
2640 BEEP 1,-20: GO TO 100
2700 REM
2710 IF py<>INT (l(5)/2) AND px<
>INT (l(4)/2) THEN RETURN
2720 CLS: RESTORE 2000
2730 PRINT "A porta abre-se com
um ruído..."
2740 FOR i=-30 TO -20 STEP .5: B
EEP .1,i: NEXT i
2750 FOR i=1 TO FN r(8): READ t#
: NEXT i
2760 PRINT "E no centro da sala
voce ve o que o trouxe aqui..."
"TAB 2;t#;"
2770 PRINT "Agora deve levar es
te objecto para uma saida situ
ada"
2775 LET ##(py,px)=" "
2780 LET g1=FN r(l(5)): LET g2=FN
r(l(4))
2785 LET ##(g1,g2)=" "
2790 GO SUB 9500
2810 LET b=FN r(100): LET l(11)=
l(11)+b
2820 PRINT "Encontrou tambem um
a vitamina" "que aumenta a sua e
nergia de ";b
2840 BORDER 0: PAPER 0: INK 5
2850 PAUSE 0
2860 GO TO 230
2900 DATA "um escravo para toda
a vida","um bilhete para Tombuct
u","um aspirador automatico","a
bola da final dos Campeoes","Eu
ropeus","a solucao do problema d
a casa"
2910 DATA "Um ZX 83 gratis..."
"um abastecimento eterno de tinto
","12 milhoes em barras de ouro"
3000 REM abrir porta
3010 LET j=CODE ##(py,px)-48: LE
T ln=INT (j/10)
3020 PRINT AT 0,3: INK 5;"A port
a tem uma fechadura no. ";ln
3025 IF l(9) THEN PRINT "As suas
chaves abrem a porta!": PAUSE 1

```

```

80: GO TO 3070
3030 IF I(12)<>"n" THEN PRINT AT
21,0; INK 5; "A sua chave nao ser
ve..."; GO TO 2630
3040 PRINT AT 20,0; "A chave serv
e! Quer abrir a fe- chadura (s/n
)?"; BEEP 1,10
3050 IF INKEY#="n" THEN GO TO 20
80
3060 IF INKEY#<>"s" THEN GO TO 3
050
3065 GO SUB 2700
3070 LET fate=j-ln*10; BORDER 4;
PAPER 5; INK 1; CLS
3073 IF fate=8 THEN GO TO 2700
3075 PRINT "NO INTERIOR, esta"...
"; BEEP .5,3; BEEP 1,10
3080 GO TO fate*200+3090
3090 REM vazio
3100 PRINT AT 3,5; "NADA!"; AT 5,0
; "Alguem esteve aqui antes."
3110 GO TO (AND>.5)*80+3120
3120 REM perder chave
3130 PRINT "A chave esta presa
na fechadu- ral! Quer tentar lib
erta-la?"
3140 PAUSE 0: IF INKEY#<>"s" THE
N GO TO 3200
3150 IF I(6) THEN PRINT "OK, nao
deve demorar muito usan- do o la
ser."; LET I(6)=I(6)-1; LET i=1;
GO TO 3190
3160 PRINT "OK, espero um moment
o enquanto voce tenta."
3170 PAUSE 70; FOR i=0 TO 10+FN
r(50); BEEP .01,AND*30-15; NEXT
i
3180 IF AND>.7 THEN LET I(13)=I(
13)-1; PRINT "Tera de a deixar a
qui."; GO TO 3190
3185 PRINT "Conseguiu!"; "Mas ";
3190 PRINT "Perdeu ";i;" unidade
s de energia"; "na sua tentativa.
"; LET I(11)=I(11)-i
3200 GO TO 4900
3290 REM laser
3300 LET u=FN r(8)+2; PRINT AT 3
,6;q#(1); "com pilhas suficientes
para";u;" utilizacoes!"
3310 IF I(6) THEN PRINT AT 3,0;"
Outro
";
3315 PRINT TAB 12;"XXXX" TAB 1

```

```

5;"XXXX"
3320 PRINT AT 9,0;"O laser tem u
m factor de consumo igual a 9. Qu
ere-o?";
3330 IF INKEY#="n" THEN GO TO 48
90
3335 IF INKEY#<>"s" THEN GO TO 3
330
3340 IF NOT I(6) THEN LET I(13)=
I(13)+9; GO TO 3350
3350 IF I(6) THEN PRINT "Como ja
tem um laser tera de le-var as
pilhas."
3360 LET I(6)=I(6)+U; BEEP .2,3;
BEEP .2,10; BEEP .2,3; BEEP .7,
10
3370 PRINT "Voce e agora o org
uothoso pos- suidor de um laser
com ";I(6);"utilizacoes garantid
as!"
3380 GO TO 4900
3490 REM desmaterializador
3500 PRINT AT 3,5;q#(2); AT 3,0;"
Outro " AND I(7)
3510 PRINT AT 5,11;"XXXX" TAB 5;"
XXXX" TAB 12;"XXXX"
3520 PRINT "Isto permite-lhe de
sintegrar pa-redes solidas com d
ois disparos e esturrar qualquer
malandro queencontre."
3530 LET u=FN r(6)+2
3540 PRINT "Este tem ";u;" carg
as";q#(2);" tem um""factor de c
onsumo de 14"
3550 PRINT "Quer leva-lo consig
a?";
3560 IF INKEY#="n" THEN GO TO 48
90
3565 IF INKEY#<>"s" THEN GO TO 3
560
3570 CLS; FOR i=0 TO 10; BEEP .
05,i; NEXT i
3580 PRINT FLASH 1; AT 2,11;"PARA
SENS!"
3590 PRINT "Voce nao precisou d
e regatear muito para ficar co
m esta formi-davel arma, a melho
r de qualquerarmaria."
3600 IF I(7) THEN PRINT "Voce n
ao quer certamente andar com do
is so" para ter as cargas gratis
...";

```

```

3610 PRINT "Para usar esta arma
, aponte parao seu alvo e carreg
ue em 'x'."
3620 PRINT "Pode usa-la ";l(7)+
u;" vezes."
3630 PRINT INVERSE 1;"AVISO DO
GOVERNO: ESTA ARMA PODEFAZER-LH
E MAL A SAUDE."
3670 LET l(13)=l(13)+14: LET (l(
7)+u)
3680 GO TO 4900
3690 REM correia transportadora
3700 PRINT AT 2,6;q$(3);AT 2,0;"
Outro " AND l(6)
3710 PRINT INK 4; PAPER 2;AT 4,1
5;" ";AT 5,7; INVERSE 1;" +=:t
:=: "; FLASH 1;"T"; FLASH 0; INV
ERSE 1;" +=:t:=: ";AT 5,15;" "
3720 PRINT "Isto permite-lhe via
jar no hi- perespaco para difer
entes partesdo labirinto, mas na
o tera qual-quer controle sobre
o destino, esse parar numa parede
sufocara imediatamente"
3730 LET u=FN r(13)+2
3740 PRINT "Este e' mais um pro
duto moderno, de deitar fora depo
is de usar, devendo ser largado
ao fim de ";u;" utilizacoes."
3750 PRINT q$(3); " tem um facto
r de "; "consumo de 5"
3760 PRINT AT 20,3;"Deseja usar
este produto?"
3770 IF INKEY$="n" THEN GO TO 49
00
3775 IF INKEY$("<"s" THEN GO TO 3
770
3790 CLS : LET l(8)=u: LET l(13)
=l(13)+5
3800 PRINT AT 1,0;"UENDIDO! A c
usada pessoa senta-da atras do Z
X Spectrum."
3810 PRINT "Use esta correia ap
enas ";u;"vezes." "Para saltar
no hiperespaco car- regue em 't'
"
3820 PRINT INVERSE 1;"O FABRIC
ANTE NAO PODE SER RES- PONSABIL
IZADO PELO USO IMPROPRIODESTA TR
ANSPORTADOR."
3830 FOR i=1 TO 5: BEEP .75,i: N
EXT i

```

```

3840 GO TO 4900
3890 REM chave
3900 PRINT 2,0;q$(4);" "
3910 PRINT TAB 15;" ";TAB 11;"
";TAB 15;" "
3930 PRINT "Esta chave da-lhe
acesso a qual-quer porta do labi
rinto!"
3940 PRINT "Pode usa-la tantas
vezes quantas quiser." "Tem um f
actor de 3"
3950 PRINT FLASH 1;"Quer leva-la
consigo?"
3960 IF INKEY$="n" THEN GO TO 48
90
3970 IF INKEY$("<"s" THEN GO TO 3
950
3980 FOR i=1 TO 10: BEEP .1,5: P
AUSE 10: NEXT i
3990 CLS
4010 IF NOT l(9) THEN LET l(13)=
l(13)+3
4020 LET l(9)=1
4030 GO TO 4900
4090 REM alimentos (energia)
4100 PRINT AT 2,10;"Comida!"
4110 PRINT "Voce corre para a m
essa e engote "; "toda a comida."
4120 LET am=FN r(25)+10
4130 PRINT AT 10,6; INVERSE 1;"C
arregue numa tecla.";AT 7,0;"A c
omida dev-lhe "; PAUSE 0
4140 IF am>25+FN r(5) THEN PRINT
"Indigestao! Nao consegue mais
- quer pontos de energia!"; BEEP
1,-20: GO TO 4900
4150 PRINT am;" valiosos pontos
de energia."
4160 LET l(11)=l(11)+am
4200 GO TO 4900
4290 REM tesouro
4310 FOR i=1 TO FN r(5): READ t$
,am: NEXT i
4320 PRINT AT 2,3;q$(5);AT 2,17;
"i"; "Voce tem ";t$
4330 LET am=FN r(am)+5: LET en=I
NT (am/5)
4340 PRINT "Tem o valor de ";am;
"pontos"; "de tesouro e um factor
de con- sumo de ";en
4350 PRINT "O tesouro pode serv

```

```

ir para pagaras monstros que eu
ardam estas passagens."
4360 PRINT "Quer leva-lo consi
o?"
4370 IF INKEY$="n" THEN GO TO 43
50
4380 IF INKEY$>"s" THEN GO TO 4
370
4390 LET L(13)=L(13)+en: LET L(1
0)=L(10)+am
4400 PRINT "Voce ganhou agora "
;L(10);"pontos de tesouro."
4410 PRINT "O seu factor de con
sumo e "Ja""de";L(13);". Cada
movimento gasta";L(13);" unidad
es de energia."
4420 GO TO 4300
4430 DATA "Ouro!",60,"Um saco de
prata!",30,"Uma bolsa cheia de
diamantes",40,"Uma espada de pla
tina",20,"Um vaso com rubis",20
4490 REM mais
4500 RESTORE 4500
4510 DATA "Um irado","Um louco",
"Um feroz","Um horrival","Um esf
omeado","Um violento","Um malvad
o","Um hediondo"
4520 FOR i=1 TO FN r(8): READ t#
: NEXT i: RESTORE 4530
4530 DATA "gorila","dragao","gig
ante","androide","lobisomem","de
monio",n#
4540 FOR i=1 TO FN r(7): READ t#
: NEXT i
4550 PRINT INK 3;AT 2,5;a#;" ";t
#
4560 PRINT INK 1;"Ele vai arranc
ar-lhe os bracos por te-lo acor
dado!"
4570 PRINT INK 0;"Que vai fazer
?";"Ira:"
4580 PRINT INK 1;"1/ fugir""2/
pagar-lhe""3/ correr""4/ rezar
para que ele desista"
4590 IF INKEY$<"1" OR INKEY$>"4"
THEN GO TO 4590
4595 CLS
4600 GO TO VAL INKEY$*50+4550
4610 PRINT "Voce tem ";
4612 IF L(6) THEN PRINT "um lase
r."
4614 IF L(7) THEN PRINT "um raio

```

```

antimateria."
4616 IF NOT (L(6)+L(7)) THEN PRI
NT "nada."
4620 LET pr=((L(7)>0)+.8*(L(6)>0
)): IF NOT pr THEN LET pr=RND/2
4622 PRINT AT 21,1;"Carregue num
a tecla para lutar"; PAUSE 0
4624 FOR i=1 TO 5*(1/pr)
4626 LET cl=FN r(8)-1: BORDER cl
: PAPER cl: INK 7-cl: LET mc=USR
invert: BEEP .05,10-FN r(20)
4628 IF L(6) THEN BORDER 2: PAPE
R 6: BEEP .075,30: LET mc=USR in
vert
4630 IF L(7) THEN BORDER 0: PAPE
R 7: BEEP .1,0: LET mc=USR inver
t
4631 LET L(11)=L(11)-2
4632 NEXT i: BORDER 5: INK 4: PA
PER 0: BRIGHT 5: CLS
4634 IF RND<pr THEN GO TO 4648
4636 PRINT AT 5,1;"PERDEU! O ";a
#(4 TO );" ";t4;" atirou o seu c
orpo para um canto onde se juntou
aos muitos outrosque ousaram vi
r aqui."
4638 FOR i=0 TO -30 STEP -1: BEE
P .5,1: NEXT i
4640 FLASH 1: CLS : GO SUB 5160
4642 POKE 23893,56: PRINT AT 1,2
;"Quer tentar de novo?"
4644 IF INKEY$="n" THEN CLS : ST
OP
4646 IF INKEY$="s" THEN GO TO 60
4647 GO TO 4644
4648 PRINT AT 5,10;"Ganhou!"
4650 IF L(7) THEN PRINT "mas com
o raio antimateria nao podia t
er perdido!": LET L(7)=L(7)-1: G
O TO 4900
4652 IF L(6) THEN PRINT "Trabalh
ou bem com o laser!": LET L(6)=L
(6)-1: GO TO 4900
4654 PRINT "Que golpe! O ";a#(4
TO );" ";t#;" deve sentir-se hoje #a
l disposto."
4656 GO TO 4900
4658 PRINT "Que vai dar ao ";a#(
4 TO );t#;"?"
4660 PRINT "Tem consigo.": FOR
a=1 TO 5: PRINT a$(a) AND L(a+5)
: NEXT a

```

```

4670 DIM g$(18): INPUT g#
4673 FOR i=1 TO 6: BEEP .1,5: IF
g#=#(i) THEN GO TO 4680
4676 NEXT i: PRINT AT 12,0:"Nao
percebi, Tente de novo!": BEEP .
5,5: PAUSE 0: DIM g$(64): PRINT
AT 12,0;g#: GO TO 4670
4680 IF i=5 THEN GO TO 4690
4682 PRINT "O nao tao ";a$(4 TO
);":t# "agradece e aceita o
";a$(i,3 TO ) "desaparecendo em s
eguida."
4684 LET l(i+6)=0: LET l(13)=(l(1
3)-e(i)
4686 GO TO 4900
4690 INPUT "Quanto vai dar ao ";
(t#);":?";a#
4693 IF a#>0 THEN GO TO 4700
4694 PRINT "O ";t#;" ficou furio
so" "por voce tentar engana-lo,
e ti-rou-lhe tudo excepto metade
do seu dinheiro."
4696 FOR i=0 TO -20 STEP -1: BEE
P .05,i: NEXT i
4698 LET t=(l(11))/2: LET l=(l(10)
): LET m=l(4): LET n=l(5): DIM l(1
3)
4697 LET l(13)=1: LET l(4)=m: LE
T l(5)=n: LET l(10)=l: LET l(11)
=t
4698 GO TO 4900
4700 IF a#>FN r(20)+10 THEN PRIN
T "O ";t#;" aceita a sua magra"
"oferta e diz-lhe que sera' aind
a mais ";a$(4 TO );":da proxima ve
z." : BEEP 1,10: GO TO 4900
4702 PRINT "O ";t#;" "insultad
o pela sua oferta, fica mais ";a
$(4 TO ) "e rouba-lhe o ";
4704 LET o=FN r(5): IF o=5 THEN
LET o=6
4706 LET l(13)=l(13)-e(o): LET l
(o+5+(o=6))=0
4708 PRINT a$(o,4 TO )
4709 GO TO 4900
4710 IF l(8) THEN PRINT "Voce ac
tivou o seu transportadore zzzap
! Desapareceu!": BEEP 1,50: GO T
O 6500

```

```

4711 FOR i=0 TO FN r(40)+10 STEP
.S: BORDER FN r(7): BEEP .01,i:
LET l(11)=l(11)-.5: NEXT i: FOR
i=0 TO FN r(50): BEEP .01,-i: N
EXT i
4714 LET r=2-FN r(3)
4718 IF NOT r THEN GO TO 4730
4722 PRINT "Nao foi suficiente me
nte rapido! O ";t#;"corre muito,
... "Agora tera' de lutar."
4725 GO TO 4610
4735 PRINT "Conseguiu! Voce devi
e ir "as Olimpíadas fazer a m
araton!"
4740 PRINT "Depois de um rapido
sprint voce vem parar ao mesmo s
itio!"
4745 PRINT AT 20,0:"Descanse um
pouco, e depois car-regue numa t
ela."
4750 BEEP .S,-50: BEEP .3,-40
4755 IF INKEY#="" THEN GO TO 475
0
4758 GO TO 4900
4760 FOR i=0 TO 5: PAPER FN r(7)
: BEEP .1,50: CLS : BEEP .S,-50:
NEXT i
4765 PAPER 6: INK 0: CLS
4770 PRINT "Nao resultou! O ";t#
"ainda esta' aqui!"
4773 LET w=FN r(10): LET l(11)=l
(11)-w
4775 PRINT "Gastou ";w;" unidade
s de energia" "a tentar."
4780 GO TO 4570
4800 PRINT "OK."
4900 GO SUB 3400: REM esperar
4920 BEEP 1,5
4930 BORDER 0: PAPER 0: INK 5: B
RIGHT 1: CLS
4940 LET m$(py,px)=" "
4950 GO TO 230
5000 REM maps
5005 IF l(10)<10 THEN PRINT INK
6;AT 0,0:"Nao tem dinheiro sufic
iente parapagar. Sem dinheiro, n
ao ha' ma-pe.": BEEP .5,-30: GO
TO 100
5010 BORDER 5: BRIGHT 0: CLS : P
RINT AT 4,9: INK 6;"****NORTE***
*"
5020 IF INKEY#<>"" THEN GO TO 50
20

```

```

5030 LET #X=(PX-5)*(PX>5)+1: LET
#Y=(PY-5)*(PY>5)+1
5040 IF #X+10>(4) THEN LET #X=(
4)-10
5050 IF #Y+10>(5) THEN LET #Y=(
5)-10
5060 LET d#=#$(PY,PX): LET #$(PY
,PX)="*"
5070 FOR l=#Y TO #Y+10: PRINT IN
K 6;TAB 9;"*";
5080 FOR c=#X TO #X+10
5090 IF CODE #$(L,C)>48 AND CODE
#$(L,C)<128 THEN PRINT "E";: GO
TO 5120: REM grafico E; imprimi
r porte
5100 IF #$(L,C)>"5" THEN PRINT F
LASH 1; INK 6;"K";: GO TO 5120:
REM grafico K
5110 PRINT #$(L,C);
5120 NEXT c: PRINT INK 6;"*";
5130 NEXT l: PRINT INK 6;TAB 9;"
*****SUL*****"
5140 LET l(13)=l(13)-2: LET l(10
)=l(10)-10: REM pagar $10 pelo #
apa
5150 LET #$(PY,PX)=d#
5160 PRINT INK 4; BRIGHT 1; AT 8,
0;"Energia restante=";l(11)"Pon
tos de tesouro=";l(10)"O seu co
nsumo por movimento e ";l(13)
5170 PRINT INK 5;AT 17,0;"Voce t
raz consigo:"
5180 IF l(6) THEN PRINT INK 5;q#
(1);l(6);" disparos."
5190 IF l(7) THEN PRINT q#(2);l(
7);" disparos."
5200 IF l(8) THEN PRINT INK 5;q#
(3);" Vida de ";l(8)
5210 IF l(9) THEN PRINT INK 4;q#
(4)
5212 IF l(12) THEN PRINT INK 5;"
Uma chave"
5220 IF NOT (l(12)+l(6)+l(7)+l(8
)+l(9)) THEN PRINT FLASH 1;"Nada
"
5225 IF INKEY#<>" " THEN GO TO 52
26
5227 GO SUB 5500
5230 PAUSE 500: RETURN
5500 IF NOT g1 THEN RETURN
5505 PRINT "A SAIDA esta' a ";N
ORTE-" AND PY>g1;"SUL-" AND PY<g
1;

```

```

5510 PRINT "OESTE-" AND PX>g2;"E
STE-" AND PX<g2;" "
5520 RETURN
6000 REM chave
6010 PRINT AT 1,5;"Voce encontro
u uma chave!"
6020 BEEP .75,10
6030 IF NOT l(12) THEN PRINT AT
21,3;"Quer apanha-la?"
6040 IF l(12) THEN PRINT AT 21,3
;"Quer trocar de chaves?"
6050 LET #$(PY,PX)=" ": REM espa
co
6060 IF INKEY#="n" THEN RETURN
6070 IF INKEY#<>"s" THEN GO TO 6
060
6075 IF l(13)<1 OR NOT l(12) THE
N LET l(13)=l(13)+1
6080 BORDER 2: PAPER 4: INK 0: C
LS
6090 LET key=FN r(5): IF key=l(1
2) THEN GO TO 6090
6095 LET l(12)=key
6100 BEEP .1,3: BEEP .4,10: BEEP
.2,3: BEEP .5,10
6110 PRINT AT 10,2;"TEM AGORA UM
A CHAVE MARCADA" l(12)"Guarde-a
bem - a sua vida pode depender
dela."
6120 GO SUB 8400
6140 BORDER 0: PAPER 0: INK 5: C
LS
6150 RETURN
6500 REM transportador
6510 LET PX=FN r(1(4)): LET PY=FN
r(1(5)): LET f=FN r(4)
6520 FOR i=0 TO 10: LET cl=FN r(
9)-1: BORDER cl: PAPER 7-cl: BEE
P .01,10-FN r(20): CLS: NEXT i
6530 FOR i=1 TO 5
6540 LET cl=FN r(8)-1: BEEP .1,4
0
6550 BORDER cl: PAPER 7-cl: CLS
6560 PAUSE 20: POKE 23693,56: CL
S
6570 NEXT i: BORDER 1
6580 PRINT AT 6,8;"Voce foi tran
sportado para outra parte do labi
rinto."
6590 LET l(8)=l(8)-1: IF NOT l(8
) THEN LET l(13)=l(13)-5
6600 GO SUB 8400: RETURN

```

```

7000 REM largar objecto
7005 PAPER 0: INK 0: BORDER 4: C
LS
7010 INPUT ("Que quer largar?")
n#;"?";g#
7020 IF g#(1)=" " THEN RETURN
7030 FOR i=1 TO 6
7040 IF g#=#(i) THEN GO TO 7070
7050 BEEP .1,10-FN r(20): NEXT i
7060 PRINT "Perdao, nao sei o qu
e e " ;g# "Tente de novo.": GO T
O 7010
7070 IF (i+5) THEN GO TO 7090
7080 PRINT "Mas voce nao tem ";g
# "Tente de novo.": GO TO 7010
7090 IF i=5 THEN GO TO 7140
7100 PRINT "OK, diga adeus ao se
u " ;g#(4 TO )
7110 LET l(13)=l(13)-e(i)
7120 LET l(i+5)=0
7130 BEEP 1,10: GO TO 7190
7140 INPUT "Quanto dinheiro quer
deixar? " ;tr
7150 IF tr>l(10) THEN PRINT "Voc
e nao tem tanto, ";n#": BEEP 1,-
10: GO TO 7140
7160 LET l(13)=l(13)-tr/5: LET l
(10)=l(10)-tr
7170 PRINT "Diga adeus a " ;tr;"
pontos" ;"de tesouro."
7180 IF l(13)<=0 THEN LET l(13)=
2
7190 BORDER 0: PAPER 0: INK 5: C
LS: RETURN
7200 REM desmaterializador
7210 LET dy=py+(f=3)-(f=1): LET
dx=px+(f=2)-(f=4)
7212 IF dy>l(5)-1 OR dy<2 OR dx<
2 OR dx>l(4)-1 THEN PRINT "Nao p
ode desmaterializar a pare-de ex
terior": PAUSE 0: RETURN
7215 LET ##(dy,dx)=" "
7220 FOR i=40 TO 55: LET cl=3-FN
r(8)
7230 PAPER cl: BORDER 7-cl: CLS
7240 BEEP .05,i: NEXT i
7250 LET l(7)=l(7)-1
7260 BEEP .5,-20: RETURN
7500 REM gravar
7505 LET l(1)=py: LET l(2)=px: L
ET l(3)=f
7510 PRINT FLASH 1;AT 15,12;"Gra
vacao"

```

```

7520 PRINT INVERSE 1;"Nome ""Lab
irinto""";AT 20,0;"Carregue numa
tecla ate limpar aimage#."
7530 SAVE "Labirinto" DATA l():
CLS: SAVE "Labirinto" DATA ##()
7540 INPUT "Verificar?";a#: IF a
#<>"s" THEN RETURN
7550 PRINT "Rebobine a cassette
e carregue em PLAY." ;"se a ver
ificacao der erro, es- creva: G
O TO 7500"
7560 VERIFY "Labirinto" DATA l()
: VERIFY "Labirinto" DATA ##()
7570 RETURN
7600 PRINT FLASH 1;AT 10,11;"Ler
dados."
7620 PRINT #0; FLASH 1;AT 1,0;"P
onha gravador em funcionamento."
7630 LOAD "Labirinto" DATA l():
LOAD "Labirinto" DATA ##()
7640 LET py=l(1): LET px=l(2): L
ET f=l(3)
7650 CLS
7660 RETURN
8000 REM variaveis
8010 IF NOT set THEN GO TO 8070
8020 LET px=2: LET py=px: LET f=
2: REM posicao do jogador
8030 DIM l(13): LET l(4)=#: LET
l(5)=n
8040 LET l(10)=200: REM tesour
o
8050 LET l(13)=1: REM consumo de
energia
8060 LET l(11)=INT (l(4)*l(5)/5)
8070 LET r=0: LET re=r: REM repe
tir
8080 LET g1=0: LET sw=0
8090 REM definir graficos
8100 RESTORE 8250
8110 FOR i=1 TO 7: READ c#
8120 FOR e=0 TO 7: READ n: POKE
USR c#+e,n
8130 NEXT e: NEXT i
8140 FOR i=USR "s" TO USR "s"+16
8150 READ b: POKE i,b: NEXT i
8160 LET invert=USR "s"
8170 DIM a#(6,18): DIM e(5)
8180 FOR i=1 TO 6: READ a#,e: LE
T a#(i)=a#: LET e(i)=e: NEXT i
8190 BORDER 0: PAPER 0: INK 5: B
RIGHT 1: CLS

```

```

3200 RETURN
3250 DATA "a",0,0,3,15,12,12,12,
8
3260 DATA "b",0,0,120,224,96,48,
48,96
3270 DATA "c",1,1,1,1,0,1,1,0
3280 DATA "d",224,192,120,120,0,
120,120,0
3290 DATA "e",255,120,120,120,12
0,120,120,255
3300 DATA "k",0,2,0,255,67,226,6
4,0
3310 DATA "h",24,60,120,60,24,24
,24,24
3320 DATA 33,0,64,6,24,197,6,0,1
25,238,255,110,36,16,249,193,15,
243,201
3330 DATA "Um laser",9,"Um raio
antimateria",14,"Uma correia tra
nsportadora",5,"Uma gazuza",3,"al
gum dinheiro",0,"Uma chave",1
3400 PRINT AT 21,1: FLASH 1; INK
9;"Carregue em 'c' para continu
ar"
3410 IF INKEY#<>"c" THEN GO TO 3
410
3420 BEEP 1,10: RETURN
9000 REM instrucoes
9002 POKE 23600,100
9005 INPUT "Ola! Indique o seu
nome: "; LINE n#: IF n#="" OR n#="
"nao" THEN GO TO 9005
9007 IF n#(1)>"Z" THEN LET n#(1)
=CHR#(CODE n#-32)
9010 BORDER 1: INK 1: PAPER 0: C
LS
9020 FOR b=0 TO 10
9030 PLOT 127,175: DRAW b*25-125
,-175
9040 PLOT 127,0: DRAW b*25-125,1
75
9050 NEXT b: POKE 23600,50
9060 PLOT 0,112: DRAW 255,0
9070 PLOT 0,70: DRAW 255,0
9080 PRINT AT 3,0:
9090 OVER 1: INK 2
9100 PRINT TAB 5;" "
9101 PRINT TAB 5;" "
9102 PRINT TAB 5;" "
9103 PRINT TAB 5;" "

```

```

9104 PRINT TAB 5;" "
9105 PRINT
9110 PRINT TAB 13;" "
9111 PRINT TAB 13;" "
9112 PRINT TAB 13;" "
9114 PRINT TAB 13;" "
9115 PRINT
9120 PRINT TAB 4;" "
9121 PRINT TAB 4;" "
9122 PRINT TAB 4;" "
9123 PRINT TAB 4;" "
9124 PRINT TAB 4;" "
9150 INK 1: OVER 0: BEEP .4,20:
BEEP .5,5: BEEP .5,10
9160 INPUT "Deseja instrucoes (s
/n)? "; LINE a#: IF a#="n" THEN
RETURN
9170 BORDER 5: PAPER 1: INK 4: B
RIGHT 1: CLS
9180 PRINT TAB 7: FLASH 1;"PORT
AS DO DESTINO"
9190 PRINT "Bemvindo nas PORTAS
DO DESTINO,"
9220 PRINT n#;"."
9230 PRINT "Voce veio aqui 'a pr
ocura do Objectivo Final que
se encontra depois de uma das po
rtas do Des-tino, no corecao do
labirinto."
9240 PRINT "Deve viajar pelas t
ongas e enga-nadoras passagens,
tentando des-cobrir o caminho co
rrecto para o interior."
9250 PRINT "Encontrara portas q
ue conduzem a passagens adjacent
es, mas atraz delas pode est
ar o PERIGO!"
9260 GO SUB 8400: PAPER 0: CLS
9270 PRINT "As portas encontram-
se fechadas 'a chave, so' podend
o ser abert- as se dispuser da c
have certa."
9280 PRINT INVERSE 1"DIRECCOES
->": INK 6
9290 PRINT "Pode usar Um mapa q
ue mostra as paredes 'a volta e

```

```

o trajecto por si seguido. Mos
trara, tambem as portas e chaves
proximas."
9300 PRINT "Sempre que pedir o #
opa ("##") perdera 10 pontos d
e tesouro."
9310 GO SUB 8400: BORDER 0: CLS
9320 PRINT TAB 0;"Comandos:": PL
OT 62,167: DRAW INK 4;72,0
9330 PRINT "M - Esquerda D -
Direita R - Virar para tras."
9340 PRINT "A - Avancar"
9350 PRINT "S - Gravar jogo"
- Repetir (se estiver num cor-
- redor comprido)."
9360 PRINT INK 5;"Cada objecto r
ecolhido tera um factor de cons
umo indicando a energia que o
obrigara a perderem cada passo."
9370 PRINT INK 7;"Voce pode esse
cificar as dimen- soes do labiri
nto desejado, mas os maiores dem
oracao cerca de 20 minutos a ser
gerados."
9380 PRINT INK 5;"Pode gravar os
labirintos em cassette para
uso futuro."
9390 GO SUB 8400
9400 RETURN

```

JOGOS COM GRÁFICOS MÓVEIS

CORRIDA AO OURO

A sua tarefa, neste jogo escrito por Neil Pellinacci, consiste em transportar seis sacos cheios de ouro de uma caixa-forte de um banco para um cofre. Ambos estão separados por um rio. Este deve ser atravessado através de uma rede de pequenos diques. Infelizmente, o futuro destes não é garantido, porque troncos de árvore, descendo o rio, destroem ocasionalmente partes do sistema de diques. Você deve transferir todos os sacos para o cofre antes de o caminho ser cortado.

Você joga usando as teclas seguintes:

Q — Subir
Z — Descer
I — Esquerda
P — Direita

A tarefa que você está a executar é indicada pela cor do seu homem. O branco indica que pegou num saco, e o amarelo que o está a transportar. Para pegar um saco, você desloca-se para o mesmo quadrado onde este se encontra. Coloca-o depois no cofre entrando neste. Em ambos os casos ouvirá um som, e o homem mudará de cor. Dispõe de três homens por jogo, e o número destes que resta em cada momento é indicado na parte superior da imagem.

Os troncos eliminam todas as secções de diques que encontram no caminho. No entanto, as secções acima e abaixo também são danificadas, e se você passar por uma destas e entrar numa zona do rio por onde tenha passado um tronco, perderá uma vida. Esta é a única maneira de você perder uma vida.

Depois de ter colocado os seis sacos no cofre, você passa ao nível seguinte. Neste existem duas formações de diques, que alternam entre si. A medida que o jogo progride, o número de troncos aumenta, até finalmente você descobrir que o jogo se torna praticamente impossível...

Verificará então que os diques quase não existem, e terá de recorrer a alguns truques. Você pode saltar para um tronco que se encontre imediatamente acima ou abaixo. De início será difícil fazê-lo, mas não impossível. Não se esqueça de que você não se desloca juntamente com o tronco; estará apenas a usá-lo como um pedaço de dique. Se tiver cuidado, pode até deslocar-se para a direita e para a esquerda usando os troncos.

O final do jogo é a parte mais difícil, sendo esta a razão principal para tentar obter a maior pontuação possível antes de o dique ser destruído.

O sistema de pontuação é simples. Obtém-se 100 pontos por ir do cofre até à caixa-forte, e outros 100 por conseguir trazer um saco no sentido contrário. Se perder uma vida enquanto transporta o saco, não pontua. A maior pontuação de Neil é de cerca de 3300, o que lhe dá já uma ideia do que deve tentar atingir.

O programa está escrito essencialmente em Basic, mas utiliza uma rotina em código-máquina localizada no endereço 32530 que desloca os troncos da direita para a esquerda do visor, parando-os depois de colidirem com a parede de um dique. Os dados correspondentes a este código-máquina encontram-se nas duas últimas linhas do programa.

Vejamos agora a estrutura do programa:

- 100 — Subrotina que imprime a pontuação
- 1000/1030 — Preparação do jogo
- 1040/1045 — Inicialização de cada jogo
- 1050/1100 — Inicialização de cada nível do jogo
- 1510/1560 — Ciclo principal
- 1600/1630 — Detecta colisões
- 1700/1740 — Final de cada nível
- 1800/1820 — Rotina de perda de um homem
- 2000/2100 — Rotina de jogo terminado
- 6000/6650 — Desenha os diques
- 7000/7100 — Desenha o rio e as margens
- 8000/8100 — Página de título e instruções
- 9000/9120 — POKE caracteres gráficos
- 9200/9310 — Poke código-máquina

Vejamos agora a listagem:

```

10 CLEAR 32529: RANDOMIZE : GO
TO 1000
99 REM pontuacao
100 PRINT AT 0,11-LEN STR# S; I
NK 7; PAPER 2; BRIGHT 1;S: RETUR
N

```

```

1000 REM
1010 BORDER 2: PAPER 4: CLS : GO
SUB 3000
1020 PRINT AT 0,0; INK 7; PAPER
2; "PONT.:"; BRIGHT 1;"00000"; BR
IGHT 0;" VIDAS:"; BRIGHT 1;"3";
BRIGHT 0;" MAXIMO:"; BRIGHT 1;"0
0000"
1030 LET H3=0
1040 GO SUB 3000: PRINT AT 0,6;
PAPER 0; INK 7; BRIGHT 1;"00000"
:AT 0,10;"3"
1045 LET MM=0: LET MEN=3: LET S=
0: LET RN=.992
1050 LET SK=0: LET L=2: LET C=20
: LET G=7
1100 GO SUB 7000: GO SUB 6000
1500 REM CICLO
1510 LET L1=L: LET C1=C
1517 LET A#=INKEY#: IF A#="" THE
N GO TO 1555
1520 LET L=L+(A#="Z")-(A#="0")
1530 LET C=C+(A#="P")-(A#="I")
1540 LET AT=ATTR(L,C): IF AT<>2
4 THEN GO TO 1500
1545 BEEP .0025,10
1550 PRINT AT L1,C1; PAPER 0; IN
K 0;" " :AT L,C: INK 6; PAPER 3;"
C" : REM GRAFICO C
1555 IF AND>RN THEN PRINT AT 7+I
NT (AND*9),30; INK 0; PAPER 0;"
": LET RN=RN-.0005
1560 LET Z=USR 32530: GO TO 1510
1565 IF AT=30 THEN IF G=7 THEN L
ET G=6: BEEP .1,20: LET S=S+100:
GO SUB 100: LET SK=SK+1: GO TO
1550
1570 IF AT=31 THEN IF G=6 THEN B
EEP .1,40: LET G=7: LET S=S+100:
GO SUB 100: IF SK=6 THEN GO TO
1700
1580 IF AT=45 THEN GO TO 1800
1585 IF AT=0 THEN GO TO 1545
1590 LET L=L1: LET C=C1: GO TO 1
500
1700 FOR A=-10 TO 20 STEP 5: BEE
P .1,A: BEEP .1,A-3: NEXT A
1710 FOR A=15 TO -10 STEP -5: BE
EP .1,A: BEEP .1,A-3: NEXT A: BE
EP .1,-15
1720 FOR A=1 TO 21: PRINT AT A,0
; PAPER A/3;" ": NEXT A

```

```

1730 BEEP .1,20; BEEP .1,25; FOR
A=1 TO 21: PRINT AT A,0; PAPER
0;": NEXT A
1740 GO TO 1050
1800 FOR A=10 TO -25 STEP -4: PR
INT AT L,C; PAPER 5; INK 0;"C";A
T L,C; PAPER 5; INK 5;": BEEP
.01,A; NEXT A
1805 PRINT AT L1,C1; PAPER 3; IN
K 0;":
1810 LET MEN=MEN-1; PRINT AT 0,1
0; PAPER 2; INK 7; BRIGHT 1;MEN
1815 IF MEN=0 THEN GO TO 2000
1817 IF SK=6 THEN GO TO 1700
1820 LET L=2; LET C=23; LET G=7;
GO TO 1500
2000 REM FINAL
2010 FOR A=0 TO 7: FOR B=1 TO 21
: PRINT AT B,0; PAPER A;": NEXT
B; NEXT A
2020 FOR A=1 TO 12: BEEP .1,3; B
EEP .1,6; BEEP .1,13; NEXT A
2030 IF S>HS THEN LET HS=S; PRIN
T AT 0,30-LEN STR$ HS; PAPER 2;
INK 7; BRIGHT 1;HS: BEEP .1,2; B
EEP .2,10
2035 FOR A=20 TO 2 STEP -1: PRIN
T AT A,0; BRIGHT 1; PAPER AND*7;
INK 9;": JOGO TERMINADO JOGO TE
RMINADO;": NEXT A
2040 FOR A=0 TO 7: BORDER A; SEE
P .2,7-A; NEXT A; FOR A=7 TO 0 3
TEP -1; BORDER A; BEEP .2,7-A-1;
NEXT A
2100 GO TO 1040
3000 STOP
3000 REM PONTES
3010 IF MM=0 THEN GO TO 6600
3020 LET MM=0
3050 PAPER 3; INK 0; PRINT AT 7,
9;": "/AT 7,18;":
3060 PRINT AT 8,9;": "/AT 8,12;":
"/AT 8,15;":
3070 PRINT AT 9,9;": "/AT 9,15
"/AT 10,9;": "/AT 10,17;
"/
3080 PRINT AT 11,9;": "/AT 11,12;
"/AT 12,6;": "/AT
12,14;": "/AT 12,22;": "/
3090 PRINT AT 13,6;": "/AT 13,14
"/AT 13,21;":

```

```

6100 PRINT AT 14,6;": "/A
T 14,18;": "/AT 15,9;": "/
AT 15,18;": "/
6500 RETURN
6550 PAPER 3; INK 0; PRINT AT 7,
12;": "/AT 7,20;": "/AT 8,10;":
6610 PRINT AT 9,10;": "/AT 9,12;":
"/AT 9,16;": "/AT 9,20;": "/A
T 10,10;": "/AT 10,13;": "/AT 10,1
6;": "/AT 10,20;": "/
6620 PRINT AT 11,10;": "/AT
11,20;": "/AT 12,10;": "/AT 12,1
6;": "/
6630 PRINT AT 13,9;": "/AT 13,1
6;": "/AT 13,20;": "/AT 14,9;": "88
"/AT 14,20;": "/
6640 PRINT AT 15,9;": "/AT 15,1
7;": "/
6650 LET MM=1; RETURN
7000 REM IMAGEM
7010 FOR A=1 TO 21: PRINT AT A,0
; PAPER 4;": "/
": NEXT A
7015 FOR A=6 TO 16: PRINT AT A,0
; PAPER 5; INK 0;": "/
": NEXT A
7020 INK 0; PAPER 3; FOR A=1 TO
3: PRINT AT A,23;": "/
7030 PRINT AT 2,9;": "/
"/AT 3,9;": "/AT 3,18;": "/AT 4,9
"/AT 4,18;": "/
7040 PRINT AT 5,13;": "/AT 6,13;":
"/AT 5,21;": "/AT 6,21;": "/
7050 PRINT AT 16,10;": "/AT 16,18
"/
7060 PRINT AT 17,9;": "/
AT 18,18;": "/AT 19,23;": "/A
T 20,14;": "/
7070 FOR A=19 TO 21: PRINT AT A,
0;": "/
": NEXT A
7080 FOR A=19 TO 21: PRINT AT A,
0; INK 5;"AA"; NEXT A; PRINT AT
2,25; INK 7;"B"; REM GRAFICOS A
E B
7100 RETURN
8000 REM TITULO
8005 BORDER 3
8010 FOR A=1 TO 21: PRINT AT A,0
; PAPER 6;": "/
": NEXT A
8012 DIM I(8); FOR A=1 TO 8: LET

```

```

I(A)=A-1: NEXT A
8013 FOR A=1 TO 8: LET II=INT (A
ND+0)+1: LET AA=I(A): LET I(A)=I
(II): LET I(II)=AA: NEXT A
8018 GO TO 8030
8018 FOR A=1 TO 8: INK I(A): BOR
DER I(A): PRINT AT 4,7;"AAAAAAA
AAAAAAAA": REM GRAFICO A
8020 PRINT AT 6,7;"AAAAAAAAAAAAA
AAAAA": REM GRAFICO A
8022 PRINT AT 5,7;"A"; AT 5,24;"A
": BEEP .4,I(A)+10: REM GRAFICO
A
8024 IF INKEY$("<") THEN BORDER 2
: RETURN
8026 NEXT A: GO TO 8016
8030 PRINT AT 10,7: PAPER 0: INK
0: BRIGHT 1;" CORRIDA AO OURO "
8040 INK 0: PAPER 5: PRINT AT 12
,12;"COMANDOS": PLOT 0,75: DRAW
96,0: PLOT 255,75: DRAW -96,0
8050 INK 1: PRINT AT 14,9;"0,....
....SUBIR"; AT 15,9;"Z.....DESC
ER"
8060 PRINT AT 16,9;"I.....ESQUER
DA"; AT 17,9;"P.....DIAEITA"
8070 PAPER 5: INK 0: PLOT 0,24:
DRAW 255,0
8080 PRINT AT 19,6: PAPER 4: INK
7: BRIGHT 1;"CARREGUE NUMA TECL
A."
8090 PRINT AT 21,0: PAPER 2: INK
7;"Escrito por Neil Pellinacci
1983"
8100 GO TO 8016
90000 REM UGD
90005 RESTORE 9100
9010 FOR A=USR "A" TO USR "A"+23
: READ B: POKE A,B: NEXT A
9100 DATA 0,126,60,24,50,125,255
,126
9110 DATA 255,129,103,165,137,12
9,255,102
9120 DATA 18,56,16,124,16,40,68,
68
9200 REM MCODE
9210 FOR A=0 TO 47: READ B: POKE
32530+A,B: NEXT A
9220 RETURN
9300 DATA 33,192,88,6,11,14,31,3
0,126,254,0,32,15,64,45,43,126,2
54,24

```

```

9310 DATA 32,4,54,45,24,2,54,0,3
5,13,32,232,35,16,227,33,192,88,
17,32,0,6,11,54,45,25,16,251,201

```

TARÂNTULA

No grande programa de Malcolm Young que se segue, Tarântula, você encontra-se num poço escuro com as dimensões aproximadas de um estádio de futebol, lutando contra temíveis aranhas.

As tarântulas deslizam pelos lados do poço procurando comida, e na opinião delas os seres humanos entram na categoria «comida»... Você está equipado com uma lanterna e um laser, e desloca-se pelo poço num tanque pouco seguro.

O programa contém todas as instruções necessárias, e você verificará que depressa descobre o modo de jogar depois de iniciar a execução do programa.

```

5 DIM P$(15): DIM W$(5,15): D
IM 4(5)
8 PRINT "Por favor espere um
pouco.": GO SUB 8615: REM defini
r caracteres
10 GO SUB 9000
20 BORDER 5: PAPER 0: INK 7: C
LS
30 PRINT AT y,x;i$
35 PRINT INK 0: PAPER 5: AT 21,
2;"PONT.=";score; AT 21,12;"ARANH
AS=";nb;" VIDAS=";vidas
40 LET a=x: LET b=y
50 LET i$=INKEY$
60 GO TO (i$("<" OR i$(">")))*20
+70
70 BEEP .05,0: GO SUB VAL i$*1
000
80 PRINT AT b,a;" "; BRIGHT 1:
AT y,x: INK rk;T$
90 GO SUB (RAND<lev)*100+100
100 GO TO (d$(y+1,x+1)="A")*390
+110: REM grafico A
110 LET cn=cn*(cn<n)+1
120 GO TO (n>0)*85+40
125 IF NOT s(cn,1) THEN GO TO (
n>1)*70+40
130 LET sx=s(cn,1): LET sy=s(cn
,2)

```

```

140 LET d$(sy,sx)=" "
150 LET sy=sy+(sy<y+1)-(sy>y+1)
: LET s(cn,2)=sy
160 LET sx=sx+(sx<x+1)-(sx>x+1)
: LET s(cn,1)=sx
170 LET d$(sy,sx)="A": REM graf
ico A
180 GO TO 40
200 REM nova tarantula
210 IF n=15 THEN LET lev=.05: R
ETURN
220 LET n=n+1: LET nb=nb+1
230 LET sx=s(n,1): LET sy=s(n,2)
240 LET d$(sy,sx)="A": REM graf
ico A
250 LET c=INT (AND*5)+2
260 PRINT BRIGHT 1; INK c; AT sy
-1, sx-1; "A"; CHR$ 0; BEEP .4, 1:
PRINT " ": REM grafico A
270 PRINT INK 0; PAPER 5; AT 21,
2; "PONT.="; score; AT 21, 12; "ARANH
AS="; nb; " VIDAS="; lives
280 RETURN
300 PRINT PAPER 7; BRIGHT 1; FL
ASH 1; INK 2; AT 10, 5; "Arrgh!! AP
ANHARAM-NO.": LET lives=lives-1
310 FOR t=1 TO n: IF s(t,1)=x+1
AND s(t,2)=y+1 THEN GO TO 530
320 NEXT t: STOP
330 LET d$(s(t,2),s(t,1))=" "
340 LET s(t,1)=0: LET nb=nb-1
350 FOR t=1 TO 10: PRINT INK 2;
AT y,x;"A"; CHR$ 0; BEEP .25, 1:
PRINT "A": BEEP .25, 0: NEXT t
360 PAPER 0: INK 7: FLASH 0: CL
S
365 IF n=15 AND nb=0 THEN GO SU
B 9250
370 PRINT INK 0; PAPER 5; AT 21,
2; "PONT.="; score; AT 21, 12; "ARANH
AS="; nb; " VIDAS="; lives
375 LET rk=rk+(rk<7)
380 IF lives THEN RETURN
390 IF score<q(5) THEN GO TO 54
0
400 INPUT PAPER 5; INK 1; AT 0, 0
; "Voce ficou nos "; FLASH 1; "cin
co primeiros!"; FLASH 0; "Indiqu
e o nome: "; LINE 0$
410 FOR r=5 TO 1 STEP -1: BEEP
.1, r: IF score>q(r) THEN NEXT r

```

```

420 FOR g=5 TO r+2 STEP -1: BEE
P .1, g: LET q(g)=q(g-1): LET w$(
g)=w$(g-1): NEXT g
430 LET q(g)=score: LET w$(g)=p
#
440 BORDER 3: PAPER 5: INK 2: C
LS
450 PRINT AT 3, 2; "Campeoes de h
oje:"
460 FOR r=1 TO 5: BEEP .5, r
470 PRINT AT 7+r, 4; FLASH (w$(r
)=p# AND score=q(r)); w$(r); q(r):
NEXT r
480 PRINT AT 10, 3; FLASH 1; INK
0; "Quer Jogar de novo?": PAUSE
0
490 IF INKEY#="n" THEN STOP
495 GO TO 10
500 REM promovover
510 BORDER 2: PAPER 6: INK 1: C
LS
520 PRINT AT 3, 11; "PARABENS !"
TAB 1; "Voce matou "; score/10; "
tarantulas"
530 PRINT TAB 5; "Voce e o porta
nto"
540 LET rk=rk-1
550 PRINT TAB 3; "promovido ao
grau "; rk
560 RESTORE 3660: FOR m=1 TO 25
: READ n: BEEP .2, n: NEXT m
570 PRINT "Voce muda agora de
tanque devido"
580 PRINT "a sua nova posicao"
590 PRINT "Comandara agora o
tanque"
600 FOR c=6 TO rk STEP -1: READ
c#: NEXT c
610 PRINT BRIGHT 1; PAPER rk; I
NK 0; c#
620 DIM d$(22, 32): LET s=7: LET
r=s/3: LET n=0: LET nb=0
630 GO SUB 9250: PRINT AT 20, 6;
"Carregue numa tecla.": PAUSE 50
0
640 BORDER 5: PAPER 0: INK rk:
CLS
650 LET p=0: RETURN
1000 LET UD=(T#="E")-(T#="D"): R
EM graficos E, D
1010 LET HD=(T#="C")-(T#="F"): R
EM graficos C, F

```

```

1020 LET v=y+1: LET h=x+1: LET w
th=1
1025 PAPER 5: INK 0
1030 GO TO ABS vd=110+1040
1040 LET dt=(s#5d)
1050 LET d=h+dt: LET d=d OR (d<1)
): LET d=d-(d>32)*(d-32)
1070 FOR t=h+hd TO 4 STEP hd
1080 LET vl=v+wth OR v-wth<1
1090 LET ll=(v+wth)-(v+wth>21)*(v+wth-21)
1100 PRINT AT vl-1,t-1;d#(vl,t);
CHR# 0): IF vl>1 THEN PRINT OVER
1;"J" AND hd=1;"I" AND hd=-1: R
EM graficos J, I
1110 FOR #=vl+1 TO ll-1: PRINT F
LASH (d#(w,t)<>"");AT #-1,t-1;d
#(w,t);: NEXT #
1120 PRINT AT ll-1,t-1;d#(ll,t);
CHR# 0): IF ll<21 THEN PRINT OVE
R 1;"H" AND hd=1;"G" AND hd=-1:
REM graficos H, G
1130 LET wth=wth+1: NEXT t
1140 GO TO 1030
1150 LET dt=(s#vd)
1160 LET d=v+dt: LET d=d OR (d<1)
): LET d=d-(d>21)*(d-21)
1170 FOR t=v+vd TO 4 STEP vd
1180 LET ll=(h-wth) OR (h-wth<1)
1190 LET vl=(h+wth)-((h+wth-32)*
(5+wth>32))
1200 PRINT AT t-1,(l-1);d#(t,(l
T
O vl)
1202 IF ll>1 THEN PRINT OVER 1;A
T t-1,(l-1);"J" AND vd=1;"H" AND
vd=-1: REM graficos J, H
1205 IF vl<32 THEN PRINT OVER 1;
AT t-1,ul-1;"I" AND vd=1;"G" AND
vd=-1: REM graficos I, G
1210 LET wth=wth+1
1220 NEXT t
1230 BEEP .2/5: PAPER 0: INK 7:
CLS: PRINT AT y,x;t#
1240 PRINT INK 0; PAPER 5; AT 21,
2;"PONT.=";score;AT 21,12;"ARANH
AS=";nb;" VIDAS=";lives
1250 LET s=s-RND/3: RETURN
2000 LET UD=(T#="E")-(T#="D"): L
ET HD=(T#="C")-(T#="A"): REM gra
ficos E, D, C, F
2010 LET U=Y+1: LET H=X+1
2020 FOR T=1 TO R

```

```

2025 LET U=U+UD: LET H=H+HD
2030 LET U=U+(U<1)-(U>21): LET H
=H+(H<1)-(H>32)
2035 BEEP .05,10
2040 PRINT AT U-1,H-1;"K": REM g
raficos K
2050 GO TO (D#(U,H)<>"")*20+205
0
2060 PRINT INK 0;AT U-1,H-1;"█":
NEXT T
2070 LET r=(r>2)/10: RETURN
2080 FOR T=0 TO 7: BEEP .1,t: PR
INT INK t;AT U-1,H-1;"A";CHR# 0);
: BEEP .1,t+7: PRINT INK 7-t;"B"
;CHR# 0): BEEP .1,t-7: NEXT T: R
EM graficos A, B
2090 LET score=score+10: LET s=s
+RND/4: LET r=INT (s/1.5)+1
2100 FOR t=1 TO n: IF vas(t,2) A
ND h=s(t,1) THEN GO TO 2120
2110 NEXT T: STOP
2120 LET s(t,1)=0: LET nb=nb-1
2130 LET d#(v,h)=" "
2140 PRINT INK 0; PAPER 5; AT 21,
2;"PONT.=";score;AT 21,12;"ARANH
AS=";nb;" VIDAS=";lives
2150 LET lev=lev+.01*(lev<.2)
2160 LET p=p+1: IF p=15 THEN GO
TO 700
2170 IF n=15 AND nb=0 THEN PRINT
BRIGHT 1; PAPER 4; INK 0; AT 15,
0;"Por favor espere 2 segundos p
elaproxima onda.": GO SUB 9250
2190 RETURN
5000 LET T#="F": REM graficos F
5010 LET X=X-(X>0)
5020 RETURN
6000 LET T#="E": REM graficos E
6010 LET Y=Y+(Y<20)
6020 RETURN
7000 LET T#="D": REM graficos D
7010 LET Y=Y-(Y>0)
7020 RETURN
8000 LET T#="C": REM graficos c
8010 LET X=X+(X<31)
8020 RETURN
9500 REM DEFINIR CARACTERES
9510 DATA 66,90,126,60,126,153,6
6,36
9520 DATA 129,155,126,60,126,153
,153,129
9530 DATA 0,126,60,55,55,60,126,
6

```

```

00940 DATA 16,16,146,254,206,254,
00941 130
00950 DATA 130,254,254,230,254,14
00951 10,10
00960 DATA 0,126,60,236,236,60,12
00961 0
00970 DATA 1,3,7,15,31,63,127,255
00980 DATA 128,192,224,240,248,25
00981 4,255
00990 DATA 255,127,63,31,15,7,3,1
00991 0
01000 DATA 255,254,252,248,246,22
01001 4,192,128
01010 DATA 0,0,0,24,24,0,0,0
01020 RESTORE
01030 FOR d=97 TO 107
01040 FOR e=0 TO 7
01050 READ n: POKE USA CHR$ d+e,n
01060 NEXT e: NEXT d
01070 RETURN
01080 DATA 0,-1,-2,-1,1,1,0,0,0
01081 1,1,1,1,0,-2,0,1,1,1,0,-1,1,0,-
01082 1
01070 DATA "AMARELO","CIARO","VERD
01071 ","MAGENTA","VERMELHO","AZUL","
01072 "PRETO"
01080 REM TITULOS
01090 RANDOMIZE RND*65535
01100 LET H$="A B A B A B A B A B
01101 A B A B A B A": REM graficos A,
01102 B
01130 INK 1: PAPER 5: BORDER 1: C
01131 L$
01140 PRINT H$(2 TO 31): AT 0,0;
01150 FOR X=1 TO 11: PRINT OVER 1
01151 TAB 31;"AB": NEXT X: REM grafic
01152 os A,B
01160 PRINT AT 21,0:H$(1 TO 32)
01170 PRINT AT 10,10: BRIGHT 1; F
01171 LASH 1;"TARANTULA"
01180 LET M$="Voce esta num poco
01181 escuro com o tamanho de um camp
01182 o de futebol lutando pela vida c
01183 ontra as temiveis tarantulas. Es
01184 tas deslizam pelas paredes do po
01185 co em busca de comida. Tera de
01186 lutar com elas se preza a sua vi
01187 da."
01190 LET M$=M$+" Voce esta equi
01191 pado apenas com uma lanterna e U
01192 m laser, e desloca-se num pequen
01193 o veiculo blindado que nao basta

```

```

para o defender das mandibulas
das tarantulas."
0100 LET M$=M$+" As suas armas f
0101 uncionam a suco de tarantula, e
0102 se nao tiverem suficiente perdem
0103 potencia."
0110 LET M$="
0111 "+M$+"
0120 PRINT INK 0: PAPER 5: AT 10,
0121 0;"Carregue E para terminar inst
0122 ru-coes; outra tecla para aceler
0123 ar."
0125 PAUSE 500
0130 FOR l=1 TO LEN M$-27: PRINT
0131 AT 15,2;M$(l TO l+27)
0140 IF INKEY$="e" OR INKEY$="E"
0141 THEN GO TO 0160
0150 PAUSE 10: NEXT l
0160 BORDER 4: PAPER 5: BRIGHT 0
0161 : CLS
0170 PRINT AT 5,2;"Use as teclas
0171 5 e 0 para mover.";"Tecla 1 pa
0172 ra acender lanterna.";"Tecla 2
0173 para disparar."
0180 DIM d$(21,32): DIM s(15,2)
0190 LET s=7: LET r=s/1.5+1
0200 LET x=INT (RND*32): LET y=I
0201 NT (RND*21)
0210 LET score=0: LET lives=3: L
0211 ET nb=0
0220 LET rk=7: LET t$="C": REM g
0221 rafico C
0230 LET cn=
0240 LET lev=.05: LET p=0
0245 GO SUB 9250: INPUT " Carreg
0246 ue em ENTER. " r$: RETURN
0250 FOR z=1 TO 15
0260 LET sx=INT (RND*32)+1: LET
0261 sy=INT (RND*21)+1: LET o=RND
0270 IF o<.5 THEN LET s(z,1)=sx:
0271 LET s(z,2)=20*(RND)>.5)+1
0280 IF o>.5 THEN LET s(z,2)=sy:
0281 LET s(z,1)=31*(RND)>.5)+1
0290 NEXT z: LET n=nb
0300 RETURN

```

A SELVA

Trata-se de um programa bastante difícil da autoria de Andrew Sweetland e Martin Jones. Na primeira fase, você deve guiar a sua canoa por um rio de floresta. Encontrará vários rochedos pelo caminho, e deve obviamente evitá-los. Se chocar com um deles, será ingerido por um peixe comedor de homens.

Na segunda fase, encontrar-se-á numa desagradável estrada como a indicada em seguida:



O seu carro não tem travões, e necessitará de toda a sua habilidade para conseguir evitar as árvores.

Use as seguintes teclas para controlar o barco e o automóvel:

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

1 REM *** A SELVA ***

©Martin Jones & Andrew Sweetland

```

2 REM episodio I
3 GO SUB 9991: LET U=3
4 POKE 23998,0
5 FOR t=1 TO 3: BORDER 0: BAI
GHT 1: PAPER 6: INK 7: CLS: BAI
GHT 0
10 FOR f=14 TO 21: PRINT AT f,
0: PAPER 1,,, NEXT f
15 PRINT #1,AT 0,0: PAPER 1,,,
20 PRINT AT 0,0: BRIGHT 1: INK
3)
30 FOR f=10 TO 13: PRINT BRIGHT
T 1:AT f,0: PAPER 3,,, NEXT f
40 PRINT AT 1,20: BRIGHT 1: IN
K 6): "AT 2,20": REM grafi
cos DE FG
45 FOR f=15 TO 18: LET p=5+(RN
D*15): PRINT AT f,p: PAPER 1: IN
K 6): "AT f,p+10": NEXT f:
PRINT AT 20,AND*30: INK 5: PAPE
R 1): REM graficos HI
50 LET x=17: LET y=0
55 IF y<20 THEN IF ATTA (x,y)
=14 OR ATTA (x,y+1)=14 OR ATTA (
x,y+2)=14 THEN GO SUB 130
60 PRINT AT x,y: INK 7: PAPER
1): REM graficos AB
70 LET y=y+1
80 IF INKEY#="q" THEN IF (x>15
AND y<20) THEN IF ATTA (x-1,y)<
>14 AND ATTA (x-1,y+1)<>14 AND A
TTA (x-1,y+2)<>14 THEN PRINT AT
x,y: PAPER 1: INK 7): LET x
=x-1
83 IF INKEY#="s" THEN IF (x<20
AND y<20) THEN IF ATTA (x+1,y)<
>14 THEN IF ATTA (x+1,y+1)<>14 T
HEN IF ATTA (x+1,y+2)<>14 THEN P
RINT AT x,y: PAPER 1: INK 7):
LET x=x+1
85 IF INKEY#="o" AND y<20 THEN
LET y=y-5
86 OUT 254,0: OUT 254,15
90 FOR z=1 TO 10: NEXT z: IF y
=30 THEN NEXT t: GO TO 105
95 GO TO 55
100 REM FIM DO EPISODIO I
105 PRINT AT x,y: PAPER 1: INK
7): FOR f=50 TO 60: FOR g=0
TO 32: OUT 254,g+15: OUT 254,0:

```

```

NEXT g: NEXT f: REM graficos AB
110 GO TO 500
120 REM COOPS!
130 PRINT AT x,y: PAPER 1; INK
7;"": FOR f=0 TO 50: OUT 254,1
6: OUT 254,0: NEXT f: PRINT AT x
+1,y+1: PAPER 1; INK 4;"00";AT x
-1,y+1: INK 7;"0000": LET (i=li-1:
IF (i=0 THEN GO TO 3000: REM gr
aficos JK,CO,LM
140 FOR f=1 TO 500: NEXT f: GO
TO 5
500 REM EPISODIO II
505 CLS: PAPER 1: INK 4: CLS:
FOR f=0 TO 21: PRINT "++++++"
++++++": NEXT f
: REM graficos N
510 FOR f=-50 TO 50: BEEP .01,f
: NEXT f
515 FOR f=1 TO 3: PRINT PAPER 4
: INK 1;AT f,0;"
": NEXT f
520 FOR f=5 TO 6: PRINT AT f,10
: PAPER 4; INK 1;"":AT f,23;"
":AT f,28;"": NEXT f
525 FOR f=7 TO 12: PRINT AT f,2
3: PAPER 4; INK 1;"":AT f,28;"
": NEXT f
530 FOR f=3 TO 4: PRINT AT f,23
: PAPER 4; INK 1;"": NEX
T f
535 FOR f=8 TO 10: PRINT AT f,3
: PAPER 4; INK 1;"":
": NEXT f
540 FOR f=10 TO 18: PRINT AT f,
1: PAPER 4; INK 1;"": NEXT f
545 FOR f=15 TO 17: PRINT AT f,
11: PAPER 4; INK 1;"":
AT f,25;"": NEXT f
550 FOR f=18 TO 20: PRINT AT f,
3: PAPER 4; INK 1;"":AT
f,28;"": NEXT f
555 FOR f=13 TO 15: PRINT AT f,
20: PAPER 4; INK 1;"":AT f,27
": NEXT f
560 PRINT PAPER 4; INK 1;AT 2,2
0;"":AT 2,24;"":AT 3,20;"
":AT 4,18;"":AT 7,18;"
":AT 8,20;"":AT 9,2;"":AT 9,8
":AT 11,20;"":AT 12,21;"":AT
13,23;"":AT 13,20;"

```

```

565 PRINT PAPER 4; INK 1;AT 14,
19;"":AT 14,23;"":AT 14,28;"
":AT 15,10;"":AT 15,21;"":A
T 16,28;"":AT 17,4;"":AT 17,9;"
":AT 18,11;"":AT 18,20;"":
AT 19,2;"":AT 19,11;"":AT 19,2
0;"":
570 LET x#="": LET x=2: LET y=0
: LET d#="0"
580 IF x=(18 AND y=31) OR (x=19
AND y=31) OR (x=20 AND y=31) TH
EN GO TO 7000
581 IF ATTR (x,y)=12 THEN GO TO
590
583 PRINT AT x,y: PAPER 4; INK
0;d#
585 OUT 254,0: OUT 254,15
590 IF INKEY#="0" THEN LET x#="
c": LET d#="#" : REM grafico 0#="
595 IF INKEY#="a" THEN LET x#="
a": LET d#="#" : REM grafico p#="
610 IF INKEY#="q" THEN LET x#="
q": LET d#="#" : REM grafico p#="
620 IF INKEY#="v" THEN LET x#="
p": LET d#="#" : REM grafico 0
630 PRINT AT x,y: PAPER 4; INK
0;"": REM grafico T
640 IF x#="c" THEN LET y=y-1
650 IF x#="p" THEN LET y=y+1
660 IF x#="a" THEN LET x=x+1
670 IF x#="q" THEN LET x=x-1
680 GO TO 580
700 GO TO 900
800 REM coops!
810 PRINT AT x,y: INK 2; FLASH
1;"": FLASH 0;AT x-1,y: PAPER 4
: INK 0;"": FOR f=0 TO 50: OUT
254,15: OUT 254,0: NEXT f: LET
(li=li-1: IF (li>0 THEN FOR r=0 TO
500: NEXT r: GO TO 500
820 IF (li=0 THEN GO TO 3000
7000 REM exito!
7010 BORDER 0: PAPER 0: INK 7: C
LS: PRINT AT 10,11;"PARABENS !"
:AT 12,2;"Voce conseguiu abrir c
aminho";AT 13,10;"pela SELVA !"
7020 FOR f=0 TO 30: BEEP .01,f:
BEEP .01,-f: NEXT f
7030 GO TO 8050
7040 STOP
8000 REM fracasso
8010 RESTORE : BORDER 0: PAPER 0

```

```

: INK 7: CLS
0020 PRINT AT 10,1;"Ah! Fascista
! Desperdicou todas as hipoteses
que lhe dei!"
0030 FOR f=1 TO 12: READ a,b: BE
EP a,b: NEXT f
0040 DATA 0,0,0,0,0,0,0,0,0,0,0,0
0,0,0,0,0,0,0,0,0,0,0,0
0,0,0,0,0,0,0,0,0,0,0,0
0050 PRINT AT 15,0;"Deseja outro
jogo (s/n)?"
0060 IF INKEY#="s" THEN LET li=3
0070 TO 5
0070 IF INKEY#="n" THEN PRINT US
A 0
0080 GO TO 0060
0090 STOP
0090 REM Ud
0091 RESTORE 0000: FOR f=0 TO 15
p: READ a: POKE USR "a"+f,a: NEX
T f
0100 DATA 0,0,1,1,40,96,127,63
0110 DATA 0,0,192,192,134,131,25
0120 DATA 240,56,168,252,124,63,
0130 DATA 0,0,0,3,15,15,31,31
0140 DATA 0,0,0,192,240,240,240,
0150 DATA 31,31,15,15,3,0,0,0
0160 DATA 240,240,240,240,192,0,
0170 DATA 0,0,0,1,0,7,31,63
0180 DATA 0,0,0,192,192,224,240,
0190 DATA 0,24,40,56,28,14,7,2
0200 DATA 0,12,6,14,28,56,112,64
0210 DATA 0,234,170,230,0,0,0,0
0220 DATA 0,234,140,234,0,0,0,0
0230 DATA 24,60,126,60,24,24,60,
0240
0250 DATA 0,230,60,255,255,58,20
0260 DATA 90,126,90,24,90,126,90
0270 DATA 0,110,34,255,255,34,11
0280 DATA 24,90,126,90,24,90,126
0290
0300 DATA 0,0,70,0,0,40,240,254
0310 DATA 0,0,0,24,24,0,0,0
0320 RETURN

```

LÓGICA

Neste rápido jogo de lógica e habilidade, você encontra-se extremamente bem disfarçado sob a forma de um asterisco... Terá de evitar os Horríveis Pontos Negros que surgem aleatoriamente à sua volta.

Como verificará ao executar o programa, terá de indicar inicialmente o nível de dificuldade. Este pode variar entre 1 e 100, correspondendo os números menores aos jogos mais fáceis. O nível 100 é apenas para peritos...

Mantenha-se em movimento contínuo a fim de não ser apanhado pelos Pontos Negros. Use as teclas de cursor para se deslocar.

```

20 INPUT "Movimentos - teclas
de cursor. Indique dificuldade
(1/100):"p
30 BORDER 4: LET x=10: LET y=1
5: LET s=10: LET a=53
40 PRINT " ": POKE 23692,255
50 IF SCREEN$(x,y)="#" THEN G
O TO 150
60 PRINT AT x,y;"*": LET s=s+1
70 PRINT AT x+INT (RND*3)-1,y+
INT (RND*3)-1; INVERSE 1;" "
80 PRINT p
90 IF CODE INKEY#<53 OR CODE I
NKEY#>56 THEN GO TO 40
100 LET a=CODE INKEY#
110 IF a=53 AND y>0 THEN LET y=
y-1
120 IF a=54 AND y>20 THEN LET x
=x+1
130 IF a=55 AND y>0 THEN LET x=
x-1
140 IF a=56 AND y<30 THEN LET y
=y+1
150 GO TO 40
160 PRINT "Voce pontuou
";s: PAUSE 500: INPUT "Carregue
em ENTER ";s: CLS : RUN

```

NA HORTA

A mente desorganizada de Neal Cavalier-Smith concebeu este estranho jogo cujo objectivo é plantar couves. Você recebe

um ponto por cada couve plantada, e perde 10 se tentar plantar uma couve em cima de uma casa. Perderá ainda três pontos se as plantar perto de uma casa ou debaixo de uma árvore (não se esqueça, leitor, de que eu disse que era estranho...).

```

3 LET s=0
4 PAPER 0: INK 7: BORDER 0
5 PRINT "15000 USR 12123100"

```

PTA NA HO

6 PRINT "Voce ganha 1 ponto p
or cada cou-ve que consiga plant
ar, mas per-de 10 se plantar uma
sobre um poste (nao crescerá
ai'), e 3 se as plantar junto
a uma casa ou sob uma arvore."

```

10 LET a#="█ █": LET s=0: LE
T h=0
20 DIM b$(12,32)
30 LET c#=""
50 FOR x=1 TO 12
60 LET c#=c#+""
70 LET b$(x)=c#+a$
80 NEXT x
90 LET y=13
100 LET x=(AND*12+1)
101 PRINT AT 17,13; FLASH 1;"IN
ICIO"
110 PRINT AT 21,0: PRINT INK 5;
b$(x)
120 LET x=x+(AND*2-1)
125 LET s=s+1
130 IF x>12 THEN LET x=x-1
140 IF x<1 THEN LET x=x+1
145 IF INT x=6 AND y=12 OR y=10
THEN PRINT AT 20,4; INK 2;"█"
: PRINT AT 21,4; INK 7;"█": LE
T s=s-2
146 IF INT x=9 AND y=14 THEN PR
INT AT 20,20; INK 4;"█": PRINT
AT 21,20; INK 4;"█": LET s=s-3
150 IF INKEY$="5" THEN LET y=y-
1
160 IF INKEY$="8" THEN LET y=y+
1
170 IF SCREEN$(11,y)<>" " THEN
GO TO 220
180 PRINT AT 11,y; INK 4;"0"

```

```

190 POKE 23692,255
200 GO TO 110
220 BEEP 1,1: LET h=h+1: LET s=
s-10: IF h=3 THEN PRINT "Pontuac
ao=";s: FOR x=1 TO 10: LET z=TAN
x: NEXT x: PAUSE 50: RUN
230 GO TO 110

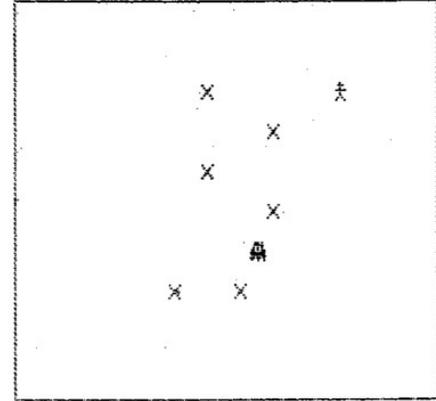
```

FANTASMAS

O fantasma persegue-o. A pequena zona de jogo apresenta algumas minas (X), e você deve tentar levar o fantasma a entrar em contacto com elas. Se tal acontecer, o fantasma será electrocutado...

As minas também constituem um perigo para si, pelo que deve evitá-las ao mesmo tempo que foge do fantasma.

Escrito por Neal Cavalier-Smith e Graham White, este programa mostra como é possível dar uma boa dose de inteligência a um fantasma.



Não conseguirá certamente fugir-lhe durante muito tempo... O programa é de facto muito rápido. Se for demasiado rápido para si das primeiras vezes que jogar, inclua alguns ciclos FOR/NEXT no programa a fim de o tornar mais lento.

```

1 GO SUB 600
2 GO SUB 300
3 LET q=0: LET s=0: LET p=5:
LET hs=0
5 BORDER 0: PAPER 0: INK 7: C
L5
10 LET a=INT (RND*18)+2
15 PLOT 38,0: DRAW 205,0: DRAW
0,160: DRAW -205,0: DRAW 0,-160
20 LET b=INT (RND*12)+10
30 LET x=2*(INT (RND*8))+2
40 LET y=2*(INT (RND*10))+10
43 LET e=2*(INT (RND*6))+6: LE
T f=2*(INT (RND*8))+10
44 IF e=x OR e=a OR f=y OR f=b
THEN GO TO 43
45 PRINT AT e,f: PAPER 2:"X"
46 IF q=p THEN GO TO 49
47 LET q=p: FOR z=0 TO 7-p: GO
TO 43: NEXT z
49 IF y>29 THEN LET y=29
50 PRINT AT x,y: INK 5:CHR$ 15
6: LET v=x: LET w=y
55 REM CHR$ 155 e' u# udg
60 PRINT AT a,b: INK 6:CHR$ 14
4: LET c=a: LET d=b
65 REM CHR$ 144 e'um udg
70 LET a=a-(x<a)+(x>a)
80 LET b=b-(y<b)+(y>b)
90 LET x=x+2*(INKEY$="6")-2*(I
NKEY$="7")
95 LET x=x+2*(x<1)-2*(x>20)
100 LET y=y+2*(INKEY$="8")-2*(I
NKEY$="9")
105 LET y=y+2*(y<5)-2*(y>29)
110 PRINT AT v,w:" ";AT c,d:" "
115 IF a=x AND b=y THEN GO TO 4
80
120 IF SCREEN$(a,b)="X" THEN G
O TO 200
130 IF SCREEN$(x,y)="X" THEN G
O TO 300
140 GO TO 49
200 PRINT AT a,b: FLASH 1:CHR$
145
205 REM CHR$ 145 e' u# udg
210 FOR d=2 TO 40 STEP 4
211 BEEP d/300,-15
212 NEXT d
215 PRINT AT a,b:" "
216 LET s=s+10
217 PRINT AT 10,0: PAPER 4: FLA
SH 1:"pt."

```

```

218 PRINT AT 12,0: FLASH 1: PAP
ER 0:s
220 GO TO 10
300 PRINT AT x,y: FLASH 1:"X":
REM grafico B
310 FOR d=20 TO 1 STEP -1
311 BEEP d/300,25
312 NEXT d
315 PRINT AT x,y:" "
316 LET p=p-1: IF p=0 THEN GO T
O 500
320 GO TO 10
400 PRINT AT x,y: FLASH 1:"X":
REM grafico B
410 FOR d=1 TO 10
411 BEEP .01,INT (RND*50)-25
412 NEXT d
415 PRINT AT x,y:" ": PRINT AT
e,f:" "
416 LET p=p-1: IF p=0 THEN GO T
O 500
420 GO TO 10
500 PAPER 1: FLASH 1: CLS
510 PRINT AT 10,7:"S E M U I
D a s"
515 FOR n=40 TO -20 STEP -1
516 BEEP .01,n
517 NEXT n
620 FOR d=1 TO 200: NEXT d
630 FLASH 0: CLS
635 PAPER 0: CLS
640 IF hs<s THEN LET hs=s
650 PRINT AT 5,10: PAPER 3:"MAX
IMO ";hs;AT 10,6: PAPER 5: INK 0
;"Carregue numa tecla."
660 LET a$=INKEY$: IF a$="" THE
N GO TO 660
665 LET p=5: LET s=0
670 GO TO 5
680 FOR n=0 TO 7
610 READ d: POKE USR "a"+n,d
620 NEXT n
630 DATA 25,62,42,107,127,1
00,73
640 FOR n=0 TO 7
650 READ d: POKE USR "b"+n,d
660 NEXT n
670 DATA 145,82,0,195,0,74,137,
0
680 FOR n=0 TO 7
690 READ d: POKE USR "m"+n,d
700 NEXT n

```

```

710 DATA 8,28,8,62,8,8,20,34
720 RETURN
800 BORDER 0: PAPER 0: INK 7: 0
L3
805 PRINT TAB 5; INK 6; FLASH 1
;" F A N T A S M A S !"
810 PRINT "" Voce deve mover o
seu homem ("; INK 5;CHA# 150; I
NK 7;")pelo visor ate' obrigar o
fan- tasma ("; INK 6;CHA# 144;
INK 7;") que o persegue, a cair
sobre as bombas ("; PAPER 2;"X";
PAPER 0;"). Mas tenha cuidado,
nao caia voce nelas! Tera 8
vidas. Boa sor
te.....
820 PRINT ""Movimentos: teclas
5 a 8"
830 PRINT TAB 6; FLASH 1;"Carr
egue numa tecla."
840 LET a#=INKEY#: IF INKEY#=""
THEN GO TO 840
850 RETURN

```

SAPO

O Sapo é outro grande programa de Malcolm Young. O autor afirma que teve grande prazer na criação deste programa porque pôde explorar nele algumas técnicas de programação novas. «Por exemplo», escreveu, «tenho experimentado rotinas em código-máquina e produzi uma rotina de «scroll» para a esquerda completamente minha... O computador não pode normalmente reconhecer os gráficos definidos pelo utilizador usando SCREEN\$, e a maior parte das pessoas tem resolvido este problema recorrendo à função ATTR.

«Verifiquei que fazendo a variável de sistema CHAR\$ apontar para a memória de UDG, a função SCREEN\$ pode actuar do modo habitual, se bem que seja necessário usar as letras correspondentes aos caracteres gráficos. Usei esta técnica no programa entre as linhas 300 e 490. Esta técnica tem a vantagem de reconhecer quaisquer caracteres UDG, incluindo os negros sem FLASH ou BRIGHT.

«Construi o programa de tal modo que pudesse ser introduzido nas máquinas de 16 K sem quaisquer modificações.

O objectivo do programa de Malcolm consiste em apanhar tantos insectos quanto possível usando as teclas I e P para

esquerda e direita, e CAPS SHIFT para saltar. Deve-se evitar sair da imagem ou cair dos troncos ou chocar com a aranha, o que resultará na perda de uma vida. Cada jogo começa com 200 unidades de energia, diminuindo lentamente até zero. Quando atinge zero, o sapo morre.

Podem aparecer borboletas, e se forem apanhadas valem 10 pontos de energia e 50 pontos de bônus. O jogador possui apenas de uma probabilidade de 50% de apanhar cada borboleta, por muito rigoroso que seja o salto.

O leitor verificará que este jogo é bastante aditivo, e pode modificá-lo de modo a incluir uma opção de níveis de dificuldade, alterando as dimensões e o número dos troncos. Depois de introduzir o programa, grave-o com a instrução SAVE «sapo» LINE 5000.

```

10 REM SAPO
20 DEF FN r(x)=INT (AND*x)+1
30 CLEAR USR "a"-43
40 GO SUB 9000: REM Udg
45 GO SUB 8110
50 GO SUB 8000: REM imagem
60 GO TO 500
70 POKE 23506,0: POKE 23607,60
80 PRINT AT 0,0: BRIGHT 1: INK
7: PAPER 2;"PONT. ";sc;TAB 11;
"ENERGIA: ";en;TAB 24;"SAPOS: ";f
FOR
90 RETURN
100 REM ler teclado
110 LET k=PEEK 23556
120 LET x=x+(k=80)-(k=73)
130 IF x<1 OR x>31 THEN GO TO 3
30
140 LET k=IN 65278
145 GO TO (k=254 OR Jp)*ds+150
150 PRINT AT y,x-1;" S ": REM g
grafico B
160 IF NOT POINT (x#8,31) THEN
GO TO 450
165 RETURN
170 REM salto
180 LET Jp=1: LET bp=10: LET ds
=40: PRINT AT y,x-1;"
190 POKE 23506,lw: POKE 23607,h
h
200 FOR l=0 TO 1
210 LET y=y-Jp
215 GO TO (y=17)*180+220
220 IF SCREEN$ (y,x)"f" THEN G
O TO 300

```

```

230 PRINT AT y-1,x;"a" AND Jp<0
: AT y,x;"c"; AT y+1,x;"d"; AT y+0,
x;"a" AND (Jp>0 AND y<16): BEEP
.005,5p
240 NEXT l
250 LET bp=bp+Jp
270 LET Jp=Jp-2*(y<3)+(y>15)
2800 POKE 23696,0: POKE 23697,60
2900 RETURN
300 LET a#=SCREEN$(y,x)
310 IF a#="i" OR a#="j" THEN LE
T e=(y-1)*32+x-2: LET i#(e TO e+
2)="": GO TO 350
320 IF a#="g" OR a#="h" THEN LE
T b=0: LET en=en+10: LET sc=sc+3
0: GO TO 350
330 FOR z=20 TO 30: BEEP .05,-z
: BEEP .1,-z-.5: NEXT z
340 LET frog=frog-1: LET s=0: B
EEP 1,-10: IF NOT frog THEN GO T
O 1000
345 LET en=200: GO TO 6000
350 LET sc=sc+10: LET Jp=-1
360 BEEP .5,10
370 GO SUB 70
380 RETURN
400 IF SCREEN$(18,x)<>"e" THEN
GO TO 450
410 GO SUB 8030
420 PRINT AT y-1,x;"c"; AT y,x;"
d"
430 PRINT AT 16,x-1;"aaa"; AT 17
,x;"b"
440 RETURN
450 BEEP .05,-5
460 PRINT AT 16,x;" "; AT 17,x;"
C"; AT 18,x;"D": REM graficos C,
D
470 BEEP .05,-5
480 PRINT AT 17,x;" "; AT 18,x;"
C": BEEP .1,-5: PRINT AT 18,x;"
": REM grafico C
490 GO TO 330
500 REM rotina principal
520 IF AND>.05 THEN GO TO 550
530 LET st=32*(FN r(6)+9)-1
540 LET i#(st TO st+1)="IU": RE
M grafico IU
550 PRINT AT 1,0: INK 1:i#
570 LET i#=i#(3 TO )
580 LET x=x-NOT Jp: LET l=USR #

```

```

600 BEEP .005,10
610 GO SUB 100
620 GO TO (NOT b)*40+530
630 LET d=FN r(4): LET b=b-1
640 LET by=by+(d=1)-(d=2): LET
bx=bx+(d=3)-(d=4)
650 LET by=by-(by>15)+(by<1): L
ET bx=bx+(bx<0)-(bx>30)
660 PRINT AT by,bx: INK 2: OVER
1,"GH": REM graficos G, H
670 IF NOT b THEN LET b=(AND>.9
)*(FN r(20)+5): IF b THEN LET by
=FN r(15)+1: LET bx=FN r(30)
680 REM areinha
690 LET s=s+(AND>.4)*((s<x)-(s>
x))
700 PRINT INK 0; AT 16,0; a#; AT 1
6,s;"LM": REM graficos L, M
710 IF y>15 AND (x=s OR x=s+1)
THEN GO TO 810
720 LET en=en-1
730 PRINT AT 6,19: BRIGHT 1: IN
K 7: PAPER 2; en;"
740 IF en<0 THEN PRINT PAPER 2:
AT 6,7: FLASH 1;"Energia nula!":
GO SUB 330
790 GO SUB 100
800 GO TO 500
810 PRINT INK 0; AT 16,s;" "; AT
17,s;"LM": REM graficos L, M
820 FOR z=0 TO 1 STEP .1
830 BEEP z,-z*10: NEXT z
840 GO SUB 340
850 GO TO 500
1000 REM tabela de pontuacao
1010 GO SUB 3000: GO SUB 70
1020 IF sc<=hs THEN GO TO 1050
1030 INPUT "Tem a melhor pontuac
ao de hoje! Indique nome: " LIN
E n#
1040 LET hs=sc
1050 DIM a#(1)
1060 INPUT "Deseja novo jogo? ":
a#
1070 IF a#="n" OR a#="N" THEN PO
KE 23693,60: PRINT AT 20,6;"OK.
ate outra vez.": PAUSE 50: BOR
DER 7: RANDOMIZE USR 4750
1080 GO SUB 3120
1090 GO TO 50
5000 REM titulo
5010 BORDER 6: PAPER 1: INK 4: C

```


TEMPOS LIVRES

PARES DE CARTAS

Este programa — escrito por Raymond Blake — baseia-se no jogo Pangolines, no qual é necessário escolher pares de cartas de um conjunto de 20 até não sobrar nenhuma. O objectivo consiste em fazê-lo no menor número de movimentos possível.

Quando lhe for pedido, indique as coordenadas da carta a virar, escrevendo primeiramente o número da coluna. A carta escolhida será então virada. Proceda novamente do mesmo modo, virando uma segunda carta. Se ambas forem iguais, são eliminadas. Se não, são viradas de novo ao contrário. O jogo termina quando não resta nenhuma carta, sendo indicada a pontuação do jogador. Quanto menor for a pontuação melhor. Raymond Blake desafia-o a bater o seu melhor: 17 lances.

```

30 BORDER 0: PAPER 0: INK 7: C
L5
50 DIM a(4,5) RANDOMIZE
50 LET a#="1112131415212223242
531323334354142434445"
50 FOR I=1 TO 10
50 FOR J=1 TO 2
100 LET l=LEN a#: LET b=INT (RN
D*(l/2)*2+1: LET b#=a$(b TO b+1)
120 LET x=VAL b$(1): LET y=VAL
b$(2)
130 LET a#=a$( TO b-1)+(a$(b+2
TO ) AND b<>(l-1)
150 LET a(x,y)=1: NEXT J: NEXT
I
200 LET sc=0
210 FOR i=1 TO 5: PRINT AT 0,4*
i-1: INK 2: INVERSE 1: BRIGHT 1:
i: NEXT i
220 FOR j=0 TO 3: PRINT AT j*5+
3,0: INK 2: INVERSE 1: BRIGHT 1:
j+1: FOR i=0 TO 4: FOR k=2 TO 5:
PRINT AT j*5+k,i*4+2: INK 6:

```

120

```

NEXT K: BEEP .03,30: NEXT I:
NEXT J
330 PRINT AT 6,23;"Virar";AT 8,
25,""
340 IF INKEY#<>"" THEN GO TO 34
0
350 LET a#=INKEY#: IF a#<"1" OR
a#>"4" THEN GO TO 350
360 PRINT AT 8,24;a#: BEEP .05,
20: LET q1=VAL a#
370 IF INKEY#<>"" THEN GO TO 37
0
380 LET a#=INKEY#: IF a#<"1" OR
a#>"5" THEN GO TO 380
390 PRINT AT 8,26;a#: BEEP .05,
20: LET q2=VAL a#
400 IF a(q1,q2)<>0 THEN GO TO 4
50
410 PRINT AT 10,23: INVERSE 1;"
Repita"
420 BEEP 1,-10: PRINT AT 8,24;"
";AT 10,23;"
430 GO TO 330
450 PRINT AT 6,23;" ";A
T 8,24;"
470 LET x1=(q1-1)*5+2: LET y1=(
q2-1)*4+2: LET x=x1: LET y=y1: L
ET c=q1: LET d=q2: GO SUB 800
490 LET q1=c: LET q2=d
530 PRINT AT 6,23;"Virar";AT 8,
25,""
540 IF INKEY#<>"" THEN GO TO 54
0
550 LET a#=INKEY#: IF a#<"1" OR
a#>"4" THEN GO TO 550
560 PRINT AT 8,24;a#: BEEP .05,
20: LET p1=VAL a#
570 IF INKEY#<>"" THEN GO TO 57
0
580 LET a#=INKEY#: IF a#<"1" OR
a#>"5" THEN GO TO 580
590 PRINT AT 8,26;a#: BEEP .05,
20: LET p2=VAL a#
600 IF a(p1,p2)<>0 THEN GO TO 6
50
610 PRINT AT 10,23: INVERSE 1;"
Repita"
620 BEEP 1,-10: PRINT AT 8,24;"
";AT 10,23;"
630 GO TO 530
650 PRINT AT 6,23;" ";A
T 8,24;"
670 LET x2=(p1-1)*5+2: LET y2=(

```

121


```

H 1;AT 21,0;"Nao possui nenhuma
carga!"; GO TO 550
680 PRINT AT 2,21; PAPER 5; INK
0;"AMARELO "
690 FOR a=1 TO 7: BORDER 7: BEE
P .3,-1: BORDER 5: PAUSE 5: NEXT
a
700 PRINT FLASH 1; INK 7; PAPER
3;AT 21,0;"Cargas de profundida
de prontas!"
710 INPUT "Indique profundidade
(1/10) ";dp
720 IF dp<1 OR dp>10 THEN INPUT
(dp;" e' fora de alcance. ");dp
GO TO 720
730 LET dp=INT dp
740 BEEP .3,-10: BEEP .2,20
750 FOR f=10 TO -dp STEP -1
760 BEEP .02,f
770 NEXT f
780 BEEP .8,-10: PAUSE 20
790 IF sx=r AND sy=p THEN BEEP
.5,1: IF sd=dp THEN GO TO 1000
800 LET s(n,3)=s(n,3)-1
810 PRINT FLASH 1;AT 21,0;"Fa(h
ou)"; IF sx=r AND sy=p THEN PRIN
T AT 21,0;"O submarino esta por
baixo de si"
820 PAUSE 100
830 PRINT AT 21,0;"
840 GO TO 110
1000 REM sub destruido
1010 BORDER 1: PAPER 1: BRIGHT 1
: INK 7: CLS
1020 PRINT AT 2,3;"PARABENS !"
1030 PRINT AT 4,3;"Voce consegui
u destruir o"
1040 PRINT AT 6,3;"submarino em
"/nb;" tentativas"
1050 PRINT "Eficiencia: ";pc/nb*
100;"%"
1060 INPUT "Deseja novo Jogo? ";
LINE i#
1070 IF i#="s" THEN RUN
1080 BORDER 7: PAPER 7: INK 0: B
RIGHT 0: CLS : STOP
3000 REM colisao
3010 BEEP 1,-12
3020 PRINT AT 21,0;"COLISAO !!";
3030 IF ATTR (p,r)=40 THEN PRINT
AT 20,0; FLASH 1; INK 0; PAPER

```

```

7;s#(n);" afundou-se!"; LET s(n,
1)=-1
3040 IF ATTR (p,r)<n*8+n-1 THEN
GO TO 3100
3050 PRINT AT 20,0; FLASH 1; INK
0; PAPER 7;s#(n);" afundou-se!"
: LET s(n,1)=-1
3060 FOR a=1 TO 0 STEP -.1: BEEP
a,a*5-10: NEXT a
3070 PAUSE 100
3080 PRINT AT 20,0;TAB 31;" ")TA
B 31;"
3090 GO TO 110
3100 LET a=ATTR (p,r)-40
3110 STOP
4000 REM ++++++
5000 REM ataque do sub
5005 IF AND>.7 THEN GO TO 250
5010 BORDER 2: LET u=1: PRINT AT
2,21; PAPER 2; INK 7;"VERMELHO"
5020 FOR a=1 TO 7: BORDER 4: BEE
P .5,-5: BORDER 2: PAUSE 5: NEXT
a
5030 IF AND<.5 THEN GO TO 5500
5040 PRINT AT 21,0; INK 7; BRIGH
T 1;s#(c);" sob ataque!"
5045 PAUSE 50
5050 LET p=5
5060 PRINT AT 21,0; PAPER 5; FLA
SH 1;"Carregue em F e tenha espe
ranca."
5070 FOR t=1 TO 2
5080 LET t1=FN r(10): LET pf=(t1
*8+164)-200
5090 PRINT AT 8,20+p; PAPER 5; I
NK 1;"+"
5100 LET p=p*(p<10)+1
5110 PAUSE 10
5120 PRINT AT 8,20+p; PAPER 5; I
NK c-(c<>6);d#(c)
5130 PAUSE 10
5140 IF INKEY$<>"f" THEN GO TO 5
090
5150 FOR i=7 TO 1 STEP -1
5160 PLOT 200,62: DRAW INVERSE 1
: INK i;pf,40
5170 BEEP .05,a: NEXT i
5180 PLOT 200,62: DRAW INK 1;pf,
40: PAUSE 50
5190 IF t1=p THEN GO TO 5300
5200 PRINT AT 21,0; PAPER 2; INK
7;"
Boa accao evasiva!"

```

```

6210 PAUSE 100: PRINT AT 21,4;"O
vidado! Outro torpedo!": FOR a=
1 TO 7: BEEP .05,a: NEXT a
6220 PAUSE 50: PRINT AT 21,0;TAB
31;" "
6230 NEXT t
6240 PRINT AT 20,0;"Falso alarme
! Danos: ";U-1
6250 PAUSE 50: PRINT AT 20,0;TAB
31;" " ;TAB 31;" "
6260 GO TO 2500
6270 REM atirado
6280 IF u=1 THEN PRINT AT 21,0;"
Atirado!": BEEP .0,-5: PAUSE 50
: PRINT AT 20,0;"0 ";s#(c);" eja
ctou todas as "sua cargas de p
ofundidade.": LET s(c,3)=0
6290 IF u=2 THEN PRINT FLASH 1;
INK 7; PAPER 0; BRIGHT 1; AT 21,0
:s#(c);" perdido!"
6300 LET usu+1: PAUSE 100
6310 IF u=2 THEN NEXT t
6320 PRINT BRIGHT 1; AT s(c,2);s(
c,1); INK 5; PAPER 5;"|"
6330 LET s(c,1)=-1
6340 FOR a=1 TO 5 STEP -.1
6350 BEEP a;a*5-10
6360 NEXT a
6370 FOR a=1 TO 20: BEEP .0025,2
0: NEXT a
6380 GO TO 2500
6390 REM submarino muito proxim
o
6400 PRINT AT 21,0; FLASH 1; BAI
GHT 1; INK 7;"Alerta!"
6410 PAUSE 100
6420 PRINT AT 21,0;s#(c);" avist
ou submarino!"
6430 PAUSE 100
6440 GO TO 2500
6450 GO TO 9999
7000 REM i# a#
7010 BORDER 4: PAPER 5: INK 1: B
RIGHT 0: CLS
7020 FOR i=0 TO 19: PRINT BRIGHT
1;"|"; REM 2
0 graficos F
7030 NEXT i
7040 PLOT 5,15: DRAW 150,0: DRAW
0,150
7050 REM minas
7060 LET m=FN r(10)+10

```

```

7070 FOR i=1 TO m: LET mx=FN r(1
9): LET my=FN r(19): IF ATTR (my
,mx)>64 THEN PRINT AT my,mx;"*"
7075 NEXT i
7080 REM navios
7090 FOR a=1 TO 6: PRINT AT s(2
,2);s(a,1); OVER 1; BRIGHT 1; PAP
ER 7; INK a-1;d#(a): NEXT a
7100 PRINT AT 5,21; FLASH 1;"SUB
MARINO!": FLASH 0; AT 1,21;"Situa
cao"; AT 2,21; PAPER 4;"VERDE "
7110 REM periscopio
7120 CIRCLE 200,100,40
7130 FOR p=40 TO 0 STEP -1
7140 PLOT 200-p,100: DRAW p*2,0,
PI
7150 NEXT p
7160 FOR i=0 TO 9: PRINT OVER 1;
AT 4+i,25;" + "; LET i=i+(i=3): N
EXT i: REM graficos G, H
7170 PRINT OVER 1; AT 6,21;"++++
++++"
7180 RETURN
8000 REM instrucoes
8010 BORDER 1: PAPER 5: INK 1: B
RIGHT 1: CLS
8020 DIM s$(5,15): DIM a(5,3): D
IM i$(1): LET u=0: LET i=1: LET
d$="|||||": REM graficos bcccc
a
8030 FOR a=1 TO 5: BEEP .1,-a: P
RINT AT 1,0; INK 2; INVERSE 1;"S
UBMARINO!": BEEP .25,a: LET i=1-
i: NEXT a
8040 INPUT BRIGHT 1;"Deseja inst
rucoes (s/n)? ";i#
8050 IF i#<>"s" THEN GO TO 8210
8060 FOR a=1 TO 6: READ p#: READ
a#
8070 PRINT AT 3,0;p#;"a#"
8080 DIM b$(LEN p#): DIM t$(LEN
a#)
8090 INPUT "Carregue e# "; FLASH
1;"ENTER"; FLASH 0;" "; LINE p#
8100 PRINT AT 3,0;b$;"t#": NEXT
a
8110 IF p#="n" THEN GO TO 8210
8120 IF p#<>"s" THEN INPUT "Quer
der um nome a sua frota (s/n
)? ";p#: GO TO 8110
8130 REM nomear navios
8140 POKE 23658,8: RESTORE 9300:

```

```

BORDER 4: PAPER 5: BRIGHT 0: CL
5
8150 FOR a=1 TO 6: READ c#
8160 PRINT AT 0,0;"Que nome gost
aria de dar a" c#;"
INPUT n#
8170 IF LEN n#>15 OR n#="" THEN
INPUT "Repita ";n#: GO TO 8170
8180 LET s#(a)=n#
8190 PRINT PAPER 0-a; INK a-1; AT
a+5,0;n#;" ";d#(a); NEXT a
8200 POKE 23658,0: GO TO 8230
8210 REM nomear navios
8220 RESTORE 9310: FOR a=1 TO 6:
READ c#: LET s#(a)=c#: NEXT a
8230 FOR i=1 TO 6: READ d: LET s
(i,1)=i-1: LET s(i,2)=20: LET s
(i,3)=d: NEXT i
8240 LET sx=FN r(20)-1: LET sy=F
N r(20)-1: LET sd=FN r(10)
8250 LET pc=FN b(): LET sb=0: RE
TURN
9000 REM data
9010 REM definir graficos
9020 FOR c=1 TO 14: READ c#
9030 FOR b=0 TO 7: READ bt: POKE
USR c#+b, bt: NEXT b
9040 NEXT c: RETURN
9050 DATA "a",0,0,0,12,20,127,12
0,0: REM corveta
9060 DATA "b",16,16,50,136,255,2
55,128,0: REM cruzador
9070 DATA "c",0,0,0,204,92,255,2
54,0: REM fragata
9080 DATA "d",137,74,44,31,248,5
0,32,145: REM explosao
9090 DATA "e",0,40,40,120,30,116
,20,0: REM mina
9100 DATA "f",255,128,128,128,12
0,128,128,128: REM rede
9110 DATA "g",0,0,1,1,7,1,1,0: R
MM 4
9120 DATA "h",0,0,0,0,224,0,0,0:
REM -
9130 DATA "i",0,0,0,24,24,0,0,0:
REM dado #1
9140 DATA "j",0,95,95,0,0,5,5,0:
REM dado #2
9150 DATA "k",0,3,0,24,24,0,192,
192: REM dado #3
9160 DATA "l",0,102,102,0,0,102,
102,0: REM dado #4

```

```

9170 DATA "m",195,195,0,24,24,0,
195,195: REM dado #5
9180 DATA "n",102,102,0,102,102,
0,102,102: REM dado #6
9190 DATA "Voce comanda uma frot
a composta por 1 cruzador, 2 cor
vetas e 3 fragatas."
9200 DATA "A sua missao consiste
em des- truir um submarino in
imigo que frequenta as rotas co
merciais."
9210 DATA "O seu cruzador esta
equipado com um sonar, que pro
duz um som mais agudo ao aproxim
ar-se do inimigo."
9220 DATA "Cada navio pode trans
portar ape-nas uma quantidade li
mitada de cargas de profundidad
e; quando as perde deve voltar
ao porto.
Cruzador - 4 cargas
Fragata - 3 cargas
Corveta - 2 cargas.
"
9230 DATA "O submarino pode torp
edear qual-quer navio a uma dist
ancia de 5 unidades, medida vert
ical ou ho-rizionalmente."
9240 DATA "O submarino move-se u
ma unidade em cada lance, e pode
atingir a profundidade de 10 un
idades. Se um dos seus navios se
encontrar directamente por cima
dele, soa-ra um alarme."
9250 DATA "O numero de movimento
s de que dispoe em cada lance
depende de um dado. Podera usa-
los para quaisquer navios ate
perfazer o total."
9260 DATA "O computador pergunta
r-lhe-a sequer mover cada navio
. Se nao quiser, carregue em E
NTER. As direccoes sao: N, S,
E, O. Pode disparar as cargas ca
rregando emD."
9270 DATA "Se houver colisoes, s
era' afun- dado o navio mais peq
ueno, o mesmo acontecendo em
choques cominas."
9280 DATA " ", "Navios sob o seu
comando: Cte. Jobo Belo (n
avio almirante) Antonio Enes (fra
gata 1) Joao Coutinho (fr

```

```

9290 DATA "Deseja dar nomes a s
ua frota (s/n)?"
9300 DATA "o seu cruzador", "a fr
agata 1", "a fragata 2", "a fragat
a 3", "a corveta 1", "a corveta 2"
9310 DATA "CTE, JOAO BELO", "ANTO
NIO ENES", "JOAO COUTINHO", "PERO
ESCOBAR", "BOAVISTA", "BRAVA"
9320 DATA 4,3,3,3,2,2
9999 BORDER 7: PAPER 7: INK 0: B
RIGHT 0: INVERSE 0: CLS : STOP

```

O CARRASCO

Apresentamos aqui uma versão bastante boa (escrita por Raymond Blake) de um jogo muito popular. Foi pensado de tal modo que o leitor possa adaptar a lista de palavras, que aliás foi mantida deliberadamente pequena para o encorajar.

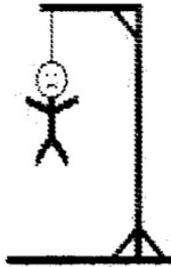
SPECTRUM HANGMAN

Da próxima vez que quiser ajuda,
não venha ter comigo!

-A--I-A

Letras erradas

BCDEFGHJKL



Para alterar as palavras, modifique simplesmente o número 30 da linha 70, substituindo-o pelo número total de palavras da sua lista, e escreva-as em declarações DATA no final do programa, tal como acontece nas linhas 1500 a 1520. O programa

dar-lhe-á maior prazer ainda se pedir a um amigo que altere a lista de palavras, de modo a você poder jogar sem as conhecer.

```

20 POKE 23558,8
25 LET F=0: RANDOMIZE
30 BORDER 6: PAPER 7: INK 0: C
LS: RESTORE
40 PRINT AT 0,7:"SPECTRUM HANG
MAN": INK 2: AT 1,7:"-----"
50 IF F=0 THEN PRINT AT 3,0:"O
ccorri! Estes malucos vao en-
forçar-me se voce nao conseguir a
divinhar a palavra!"
55 IF F=1 THEN PRINT AT 3,0:"O
h, nao! Agora querem enforçar om
eu amigo! Voce vai ter de nos a
judar!"
60 LET W=0: LET W$="": LET C=0
70 FOR I=1 TO INT (RND*30+1):
READ A$: NEXT I
75 FOR I=-20 TO 30: BEEP .02,I
: NEXT I
80 LET L=LEN A$: LET B$="-----"
" ( TO L)
90 PRINT AT 12,3: INK 2;B$
100 PRINT AT 3,0: INK 1:"Indiqu
e letra"
120 INPUT C$: IF LEN C$<>1 AND
LEN C$<>L THEN GO TO 120
130 IF C$<>"A" OR C$>"Z" THEN GO
TO 120
140 PRINT AT 3,0:"
150 IF LEN C$=1 THEN GO TO 180
160 IF C$=A$ THEN LET B$=C$
170 IF C$<>A$ THEN LET W=L
175 GO TO 200
180 LET W=0: FOR I=1 TO L: IF A
$(I)=C$ THEN LET B$(I)=C$
190 IF C$<>A$(I) THEN LET W=W+1
195 NEXT I
200 PRINT AT 12,3: INK 2;B$: IF
W=L THEN LET C=C+1: GO SUB 730+
C*20: BEEP .6,5*(10-C)-25: LET W
#=W#+(C# AND LEN C$=1)
210 IF W<>L THEN FOR I=10 TO 30
: BEEP .02,I: NEXT I
220 PRINT AT 17,0:"Letras errad
as": PRINT AT 19,2: INK 4;W$
240 IF A$=B$ THEN GO TO 500
250 IF C=10 THEN GO TO 500
260 GO TO 100
500 PRINT AT 3,0: INK 3:"Da pro

```

```

xima vez que quiser ajuda, nao ve
nha ter comigo!
520 FOR I=20 TO -30 STEP -1: BE
EP .02, I: NEXT I: GO TO 550
600 PRINT AT 3,0; INK 3; " Muito
obrigado. Devo-lhe um co-po, am
igo!"
520 FOR I=-20 TO 30: BEEP .02, I
: NEXT I
650 PRINT AT 10,2; INK 1; "A pal
avra era: "; AT 12,3; A$
660 PAUSE 120
690 LET F=1: GO TO 30
800 FOR J=10 TO 12: PLOT 150, J:
DRAW 30,0: NEXT J
810 FOR J=212 TO 214: PLOT J, 13
: DRAW 0, 100: NEXT J: RETURN
820 FOR J=201 TO 203: PLOT J, 13
: DRAW 10, 10: NEXT J
830 FOR J=21 TO 23: PLOT 215, J:
DRAW 10, -10: NEXT J: RETURN
840 FOR J=111 TO 113: PLOT 211,
J: DRAW -45,0: NEXT J
850 FOR J=201 TO 203: PLOT J, 11
0: DRAW 10, -10: NEXT J: RETURN
860 PLOT 171, 110: DRAW 0, -20: R
ETURN
880 CIRCLE 171, 83, 7: PLOT 169, 8
5: PLOT 173, 85
890 PLOT 169, 80: DRAW 4, 0, .5*-P
I: RETURN
900 FOR J=170 TO 172: PLOT J, 75
: DRAW 0, -14: NEXT J: RETURN
920 FOR J=70 TO 72: PLOT 169, J:
DRAW -10, 4: NEXT J: RETURN
940 FOR J=70 TO 72: PLOT 173, J:
DRAW 10, 4: NEXT J: RETURN
960 FOR J=59 TO 62: PLOT 171, J:
DRAW -7, -10: NEXT J: RETURN
980 FOR J=59 TO 62: PLOT 171, J:
DRAW 7, -10: NEXT J: RETURN
1500 DATA "ISOTOPO", "POKER", "PLA
NETA", "ESFERA", "FALACIA", "TRIUNF
O", "CANGURU", "ATRIBUTO", "PRAZO",
"CAMINHO"
1510 DATA "HOTEL", "GALAXIA", "UNI
VERSO", "MIRAGEM", "ESTREMECER", "C
OLORIDO", "SORRISO", "VISAO", "ESTR
ADA", "SALTERDO"
1520 DATA "ARDOSIA", "TEMPORAL", "
COLEGA", "IMPRESSORA", "MAQUINA", "
DESDOBRAR", "DIVISAO", "REVISTA", "
PALMEIRA", "CORREDOR"

```

JOGOS NO ESPAÇO

FUGA ENTRE AS ESTRELAS

Este pequeno programa, escrito por Graham Charlton, produz um jogo surpreendentemente interessante. Deve usar as teclas «Z» e «M» para se deslocar para a esquerda e direita, evitando os asteriscos que se movem verticalmente.

```

10 BORDER 2: PAPER 2: CLS
20 LET a=10: LET b=15: LET c=2
0: LET t=0
30 POKE 23692,0
40 LET t=t+1
50 PRINT AT 21,31; " "
60 FOR z=1 TO 2
70 INK 7: PRINT AT c, AND*31; "*"
80 NEXT z
90 LET b=b+(INKEY$="M" AND b<3
0)-(INKEY$="Z" AND b>1)
100 IF SCREEN$(a,b)="#" THEN B
EEP 2,8: PRINT "Pontuou "; t: FOR
x=1 TO 200: NEXT x: RUN
110 PRINT AT a,b,"@"
120 GO TO 30

```

SPACE TREK

Apresentamos agora uma outra contribuição de David Perry para este livro. Neste jogo, você voa através das galáxias, evitando as estrelas, tentando disparar contra os cruzadores de batalha inimigos. Quando tiver terminado, deve abandonar a galáxia através do Buraco Negro, afastando-se para as profundezas do espaço...

O programa inclui todas as instruções necessárias. Os comandos são dados usando as teclas de cursor, e zero para disparar.

```

1 REM
2 REM SPACE TALK!
3 REM #####
4 REM
5 REM COPYRIGHT
6 REM DAVID PERRY
7 REM 1983
8 REM
9 POKE 23858,8: LET num=10: D
IM n$(num+1,0): DIM n(num+1): FO
R n=1 TO num: LET n(n)=(1000-(10
0*n)): LET n$(n)="SPECTRUM": NEX
T n: GO SUB 2000: LET n$(1)="DAV
E OK!": LET n(1)=5000
10 BORDER 1: PAPER 0: INK 7: B
RIGHT 1: CLS
20 LET gal=1: LET bon=3: LET s
c=0: LET lev=3: LET men=3
30 PRINT AT 0,0: INK 6: BRIGHT
0:
"; AT 21,0;"
35 FOR n=1 TO 3: PRINT INK 5;A
T n,0: "; AT n,31:" ": NEXT n: F
OR n=4 TO 20: PRINT INK 5; AT n,0
: "; AT n,31:" ": NEXT n
40 IF men=0 THEN GO TO 1000
42 FOR n=1 TO lev
45 LET x=INT (RAND*14)+6: LET y
=INT (RAND*29)+1: IF SCREEN$(x,y
)<>" " THEN GO TO 45
50 PRINT AT x,y: INK 2;"K"
51 NEXT n
54 FOR j=1 TO bon*2
55 LET x=INT (RAND*14)+6: LET y
=INT (RAND*30)+1: IF SCREEN$(x,y
)<>" " THEN GO TO 55
60 PRINT AT x,y: INK 6;"*"
70 NEXT j
75 LET x=INT (RAND*14)+6: LET y
=INT (RAND*29)+2: IF SCREEN$(x,y
)<>" " THEN GO TO 75
80 PRINT AT x,y: INK 7;"O"
90 PRINT AT 1,3: INK 4;" Nav
es: "; FOR n=1 TO men: PRINT I
NK 7;"a";: NEXT n: PRINT AT 1,19
: INK 6;"Pont: ";sc
100 PRINT AT 2,8: INK 3;"Pont.
maxima: ";n(1): PRINT AT 4,0: IN
K 5;"

```

```

105 PRINT AT 3,4: INK 2;"Galaxi
a: ";gal: PRINT AT 3,14: INK 6;"K
lingons: ";lev
110 LET d=2: LET a=5: LET b=1
120 PRINT AT a,b;" "
130 IF d=1 THEN LET a=a-1
140 IF d=2 THEN LET b=b+1
150 IF d=3 THEN LET a=a+1
160 IF d=4 THEN LET b=b-1
163 IF b<1 THEN LET b=30
164 IF b>30 THEN LET b=1
165 IF a<5 THEN LET a=20
166 IF a>20 THEN LET a=5
167 IF lev=0 THEN BEEP .01,50:
IF SCREEN$(a,b)="O" THEN LET bo
n=bon+2: LET gal=gal+1: PRINT AT
3,11;gal: GO SUB 300: LET lev=b
on: GO TO 40
170 IF SCREEN$(a,b)<>" " THEN
FOR i=7 TO 0 STEP -1: PRINT AT a
,b: INK i;"X": BEEP .02,i*7: NEX
T i: GO SUB 300: PRINT AT 1,13;"
": LET men=men-1: FOR n=1 TO
men: PRINT AT 1,13+n;"a": NEXT
n: GO TO 40
180 IF INKEY$="7" THEN LET d=1
190 IF INKEY$="8" THEN LET d=2
200 IF INKEY$="9" THEN LET d=3
210 IF INKEY$="5" THEN LET d=4
220 IF d=1 THEN PRINT AT a,b: I
NK 6;"a"
230 IF d=2 THEN PRINT AT a,b: I
NK 6;"b"
240 IF d=3 THEN PRINT AT a,b: I
NK 6;"c"
250 IF d=4 THEN PRINT AT a,b: I
NK 6;"d"
255 IF INKEY$="0" THEN GO SUB 3
00
260 BEEP .01,d*10: GO TO 120
300 IF d=1 THEN GO TO 400
310 IF d=2 THEN GO TO 500
320 IF d=3 THEN GO TO 600
330 IF d=4 THEN GO TO 700
340 STOP
400 IF a=5 THEN RETURN
405 FOR n=a-1 TO 5 STEP -1
410 LET a$=SCREEN$(n,b)
420 IF a$="K" THEN LET lev=lev-
1: PRINT AT 3,23;lev;" ": LET s
c=sc+100: PRINT AT 1,25;sc: FOR
i=7 TO 0 STEP -1: PRINT AT n,b)

```

```

INK I;"K": BEEP .01,i*7: NEXT I:
PRINT AT n,b;" ": NEXT n
430 IF a#="*" OR a#="0" THEN BE
EP .03,10: LET n=6: RETURN
440 PRINT AT n,b; ("f" AND n>5):
BEEP .01,n
450 PRINT AT n,b; (" " AND n>5):
NEXT n: RETURN
500 IF b=30 THEN RETURN
505 FOR n=b+1 TO 30
510 LET a#=SCREEN$(a,n)
520 IF a#="K" THEN LET lev=lev-
1: PRINT AT 3,28;lev;" ": LET s
c=sc+100: PRINT AT 1,25;sc: FOR
i=7 TO 0 STEP -1: PRINT AT a,n;
INK I;"K": BEEP .01,i*7: NEXT I:
PRINT AT a,n;" ": NEXT n
530 IF a#="*" OR a#="0" THEN BE
EP .03,10: LET n=30: RETURN
540 PRINT AT a,n; ("e" AND n<31)
: BEEP .01,n
550 PRINT AT a,n; (" " AND n<31)
: NEXT n: RETURN
600 IF a=20 THEN RETURN
605 FOR n=a+1 TO 20
610 LET a#=SCREEN$(a,n)
620 IF a#="K" THEN LET lev=lev-
1: PRINT AT 3,28;lev;" ": LET s
c=sc+100: PRINT AT 1,25;sc: FOR
i=7 TO 0 STEP -1: PRINT AT a,n;
INK I;"K": BEEP .01,i*7: NEXT I:
PRINT AT a,n;" ": NEXT n
630 IF a#="*" OR a#="0" THEN BE
EP .03,10: LET n=20: RETURN
640 PRINT AT n,b; ("f" AND n<21)
: BEEP .01,n
650 PRINT AT n,b; (" " AND n<21)
: NEXT n: RETURN
700 IF b=1 THEN RETURN
705 FOR n=b-1 TO 1 STEP -1
710 LET a#=SCREEN$(a,n)
720 IF a#="K" THEN LET lev=lev-
1: PRINT AT 3,28;lev;" ": LET s
c=sc+100: PRINT AT 1,25;sc: FOR
i=7 TO 0 STEP -1: PRINT AT a,n;
INK I;"K": BEEP .01,i*7: NEXT I:
PRINT AT a,n;" ": NEXT n
730 IF a#="*" OR a#="0" THEN BE
EP .03,10: LET n=1: RETURN
740 PRINT AT a,n; ("e" AND n>0):
BEEP .01,n
750 PRINT AT a,n; (" " AND n>0):

```

```

NEXT n: RETURN
801 LET b=2: FOR n=20 TO 5 STEP
-1
810 BEEP .02,n*2: PRINT AT n,0;
INK 5;" "
820 NEXT n
830 PRINT AT 3,31; INK 5;" ": P
RINT AT 21,1; INK 5;" ": RETURN
1000 PRINT AT 10,4; PAPER 7; INK
0; BRIGHT 0;"Voce pontuou ";sc)
" "
1010 BEEP .1,24: BEEP .1,23: BEE
P .04,24: BEEP .5,21: BEEP .5,20
: BEEP .05,21
1500 IF sc<=n(10) THEN PAUSE 50:
GO TO 1540
1502 LET num=11: IF sc>=n(num) T
HEN INPUT "Indique 8 iniciais! "
; p#: IF LEN p#>8 THEN GO TO 1500
1505 IF sc>=n(num) THEN LET n(num
)=sc: LET n$(num)=p#
1510 FOR a=1 TO (num-1): LET b#=
n$(a): LET c#=n$(a+1): LET b=n(a
): LET c=n(a+1): IF b<c THEN LET
n(a)=c: LET n(a+1)=b: LET n$(a)
=c#: LET n$(a+1)=b#
1520 NEXT a: FOR n=1 TO num-1: I
F n(n)<n(n+1) THEN GO TO 1510
1530 NEXT n
1540 CLS
1550 PRINT AT 2,7;"PONTUAC0ES MA
XIMAS"
1560 PRINT AT 3,7;"=====
====="
1580 FOR n=1 TO num-1: PRINT AT
n+5,7; INK 6; "("; INK 2;n; INK 5
; ")" AT n+5,12; INK 7;n(n); PRIN
T AT n+5,17; INK 5;n$(n): NEXT n
1590 PLOT 0,0: DRAW 55,0: DRAW 0
,175: DRAW -255,0: DRAW 0,-175:
PRINT AT 17,6; PAPER 2; INK 7;"C
arregue numa tecta.": LET i=0
1600 LET i=i+1: IF i>7 THEN LET
i=0
1610 PRINT AT 2,7; INK I;"PONTUA
COES MAXIMAS"
1620 BEEP .01,i*7: PAUSE 2: IF I
NKEY#="" THEN GO TO 1600
1630 GO TO 10
2000 BORDER 1: PAPER 0: INK 7: B

```

```

RIGHT 1: CLS
2030 PRINT AT 2,6;"S P A C E . T
R E K !"
2040 PRINT AT 3,6;"#####
#####"
2050 PRINT AT 5,1;"Comande a sua
nave usando as teclas de cur-
sor e "O" para dis-parar. O obje-
ctivo do jogo e mover-se entr-
e as estrelas per-seguindo os c-
ruzadores Klingon. Quando estive-
rem destruidos, po-dera sair po-
r um buraco negro."
2065 RESTORE 2085: FOR c=0 TO 7:
READ hat: POKE USR "a"+c,hat: N
EXT c
2070 PRINT : PRINT " Voce: "a"
Buraco negro: "; INK 7;"O"
2080 PRINT : PRINT " Estrelas: "
"; INK 6;"*"; INK 7;"Klingon:
"; INK 2;"K"; INK 7;" "
2081 PLOT 0,0: DRAW 255,0: DRAW
0,175: DRAW -255,0: DRAW 0,-175
2084 FOR c=0 TO 39: READ hat: PO
KE USR "b"+c,hat: NEXT c
2085 DATA 24,36,36,36,60,90,102,
66
2086 DATA 0,224,94,49,49,94,224,
6
2087 DATA 66,102,90,60,36,36,36,
24
2088 DATA 0,7,122,140,140,122,7,
6
2089 DATA 0,0,0,255,255,0,0,0
2090 DATA 24,24,24,24,24,24,24,2
4
2110 LET i=0
2120 LET i=i+1: IF i>7 THEN LET
i=0
2130 PRINT AT 2,6: INK i;"S P A
C E T R E K !"
2140 BEEP .01,50-(i*7): PAUSE 2:
IF INKEY#="" THEN GO TO 2120
2150 RETURN
9990 FOR n=0 TO 7: PRINT INK n;"
"; NEXT n
9999 RETURN

```

Trata-se de um jogo fascinante onde, uma vez mais, a nossa missão consiste em patrulhar um sector da galáxia. Como terá adivinhado, existem alienígenos que você deve encontrar e destruir.

O programa mostra um mapa do sector galáctico onde você se encontra, juntamente com as suas características mais relevantes em termos de jogo. Depressa aprenderá a ler este mapa, à medida que for executando o jogo.

Em qualquer momento pode mover-se, tentar detectar inimigos ou disparar sobre eles. Os seus detectores actuam de duas formas diferentes. O detector de curto raio de acção, que investiga os oito quadrados contíguos àquele onde a sua nave se encontra, consome pouca energia. Os detectores de longo alcance investigam dois quadrados numa única direcção, e gastam uma maior quantidade de energia.

O jogo termina se você aterrar em cima de uma nave alienígena... Os inimigos não se deslocam dos mesmos sectores durante cada jogo. Você dispõe de reservas de energia limitadas, e deve tentar eliminar tantos alienígenos quanto possível antes de perder completamente a energia.

O programa inventa nomes diferentes para os alienígenos de cada vez que é executado. Cada um destes só pode disparar contra si depois de você próprio usar o laser sobre ele (revelando assim a posição em que se encontra), e apenas no caso de se encontrar a uma distância de um único quadrado. Os danos provocados na sua nave por um tiro inimigo são indicados em termos de energia (ou seja, cada tiro eficaz drena as baterias da sua nave). O jogo continua até você abater uma nave inimiga ou ficar sem energia. Tim Rogers converteu este jogo, a partir de um original meu.

```

10 BORDER 0: RANDOMIZE
20 INK 6
30 PAPER 0
40 CLS
50 GO SUB 2000: GO SUB 1000
70 GO SUB 3000
110 GO TO 70
1010 DIM g(10,10)
1020 DIM a(10,10)
1050 FOR a=1 TO 20
1060 LET g(FN a(x),FN a(x))=1
1100 NEXT a
1110 LET g(U,P)=2

```

```

1120 LET s(U,p)=2
1125 LET rn=FN a(8)
1130 LET rn1=FN a(8)
1135 LET z#=FN a#(0)
1140 RETURN
1510 LET b1=FN a(20)
1520 GOTO 1,b1
1530 GOTO 1,b1-12
1540 RETURN
2000 DEF FN a(x)=INT (RND*x)+1
2010 DEF FN a#( )=x#(rn, TO t(rn,
1)))+x#(rn1, TO t(rn1,2))
2015 LET x=10
2020 LET u=5: LET p=5
2040 LET a##="  " : REM grafico
a, b, c
2100 DIM c#(5,9)
2110 FOR a=1 TO 5
2120 READ c#(a)
2130 NEXT a
2140 LET at=0
2150 DIM b(8): DIM c(8)
2160 FOR a=1 TO 8
2170 READ b(a): READ c(a)
2180 NEXT a
2190 LET e=1000+2000*RND
2200 DIM x#(10,8)
2210 DIM y#(10,8)
2215 DIM t(8,2)
2220 FOR a=1 TO 8
2225 READ o#: LET t(a,1)=LEN o#
2230 LET x#(a)=o#
2235 READ o#: LET t(a,2)=LEN o#
2240 LET y#(a)=o#
2250 NEXT a
2255 DIM s#(10,11)
2270 FOR a=1 TO 9
2280 LET rn=FN a(8)
2290 LET rn1=FN a(8)
2300 LET s#(a)=FN a#( )
2310 NEXT a
2320 FOR a=1 TO 3
2330 FOR b=0 TO 7
2340 READ c
2350 POKE USA CHR# (143+a)+b,c
2360 NEXT b: NEXT a
2400 RETURN
3010 OLS
3020 PRINT AT 0,10;"MAPA ";
3030 GO SUB 1500
3050 PRINT "ESTELAR"
3060 GO SUB 1500

```

142

```

3100 FOR a=1 TO 32
3110 PRINT "*";
3130 NEXT a
3140 GO SUB 1500
3150 PRINT AT 3,0;" 1234567890"
3160 FOR a=1 TO 10
3170 IF a<10 THEN PRINT a: GO TO
0 3190
3180 PRINT "0";
3190 FOR b=1 TO 10
3195 LET b##=a#(s(a,b)+1)
3200 PRINT INVERSE 1;b#;
3205 IF b#<>" " THEN GO SUB 1500
3210 NEXT b
3215 IF a<10 THEN PRINT a: GO TO
3230
3220 PRINT "0"
3230 NEXT a
3240 PRINT " 1234567890"
3310 PRINT AT 5,14; INK 2; PAPER
7; FLASH 1;"Baterias: ";INT e
3320 PRINT AT 6,14;
3330 FOR a=1 TO e/200
3340 PRINT INK 5;" ";
3350 NEXT a
3360 GO SUB 1500
3370 IF e<1 THEN GO TO 4300
3380 IF at>0 THEN PRINT AT 8,14;
"Naves inimigas";AT 9,14;"destru
idas ";at: GO SUB 1500
3390 PRINT AT 10,14;"Nave locali
zada"
3400 PRINT AT 11,14;"no subquadr
ante"
3610 PRINT AT 12,14; FLASH 1;"t
";U;"I";p;"j"
3620 GO SUB 1500
3630 PRINT AT 13,14;"no sector
";AT 14,14;s#(INT (U*p/10+.5));
3640 PRINT AT 16,0;
3645 GO SUB 1500
3650 GO TO 4500
4010 GO SUB 5000
4020 PRINT FLASH 1;"Baterias com
pletamente gastas"
4030 PRINT at;" inimigos mortos"
4110 PRINT ",c#(FN a(5));": dese
ja ser" "comandante de novo?"
4120 IF INKEY#<>" " THEN GO TO 41
20
4130 IF INKEY#="" THEN GO TO 413
0

```

143

```

4140 IF INKEY#<>"n" AND INKEY#<>
"n" THEN RUN
4150 PRINT ",c#(FN a(5))": voce
4160 BEEP .01
4170 PRINT "Um pouco ("
4180 LET n=FN a(3)
4190 LET p1=FN a(3)
4200 PRINT FN a#(1)
4210 PRINT end
4220 GO SUB 5000
4230 PRINT AT 16,0;c#(FN a(5)):"
- Ordens, comandantes?"
4240 BEEP .01,RND*50
4250 PRINT AT 17,12;"1 Deteccao"
4260 GO SUB 1500
4270 PRINT TAB 12;"2 Movimento"
4280 GO SUB 1500
4290 PRINT TAB 12;"3 Disparar"
4300 GO SUB 1500
4310 PRINT AT 21,0;"Carregue 1,"
D, 3 na sua consola"
4320 LET i#=INKEY#
4330 IF CODE i#<49 OR CODE i#>51
THEN GO TO 4399
4340 GO SUB 1500
4350 GO SUB 5000
4360 IF i#"1" THEN GO TO 5000
4370 GO SUB 5000+1000*VAL i#
4380 FOR a=0 TO 20
4390 PRINT AT 21,a;"#####"
4400 BEEP .01,a+12
4410 PRINT AT 21,a;" "
4420 NEXT a
4430 PRINT AT 21,a;" "
4440 RETURN
4450 PRINT AT 16,0;
4460 FOR a=1 TO 5
4470 PRINT "

```

```

5000 NEXT a
5010 PRINT AT 16,0;
5020 RETURN
5030 PRINT TAB 10; FLASH 1; INK
7;"DETECCAO"
5040 PRINT c#(FN a(5)):"-"
5050 PRINT "longinqua (2) ou pro
xima (1)?"
5060 LET i#=INKEY#
5070 BEEP .01,0
5080 IF CODE i#<49 OR CODE i#>50
THEN GO TO 5030

```

```

5090 LET k=VAL i#
5100 LET e=e-10*k
5110 IF k=2 THEN GO TO 5500
5120 PRINT AT 16,0;" Deteccao,"
5130 LET f=0
5140 FOR a=-(u>1) TO (u<10)
5150 FOR b=-(p>1) TO (p<10)
5160 BEEP .01,0
5170 IF g(u+a,p+b)=1 THEN LET f=
1
5180 NEXT b: NEXT a
5190 IF f=1 THEN PRINT "Nave ";z
#;" na vizinhanca!": GO TO 5150
5200 PRINT "Nada na vizinhanca,"
5210 GO SUB 1500
5220 PRINT AT 21,0;"Desligar det
eccao?"
5230 BEEP .1,AND#50
5240 IF INKEY#="" THEN GO TO 517
0
5250 IF INKEY#="n" OR INKEY#="N"
THEN GO SUB 5000: GO TO 5010
5260 GO SUB 1500
5270 GO TO 4500
5280 PRINT AT 16,0;c#(FN a(5)):"
Direccao?"
5290 PRINT "Norte:1 Este:2 SUL
3 Oeste:4"
5300 IF INKEY#<>"" THEN GO TO 55
40
5310 LET i#=INKEY#
5320 BEEP .01,0
5330 IF CODE i#<49 OR CODE i#>52
THEN GO TO 5520
5340 LET k=(VAL i#-1)*2+1
5350 LET z=(g(u+b(k)*2,p+c(k)*2)
=1)
5360 IF z=1 THEN PRINT FLASH 1;"
positivo": GO TO 5590
5370 PRINT "negativo"
5380 GO SUB 1500
5390 IF INKEY#<>"" THEN GO TO 55
20
5400 GO TO 5150
5410 LET fl=0
5420 LET e=e-50
5430 PRINT "RU#0:"
5440 GO SUB 1500
5450 CIRCLE 40,24,16
5460 PLOT 40,24
5470 DRAW 0,16

```

```

7050 GO SUB 1500
7060 PRINT AT 16,16;"000"
7080 PLOT 40,24
7090 DRAW 16,0
7100 GO SUB 1500
7110 PRINT AT 18,16;"000"
7130 PLOT 40,24
7140 DRAW 0,-16
7150 GO SUB 1500
7160 PRINT AT 20,16;"100"
7180 PLOT 40,24
7190 DRAW -16,0
7200 GO SUB 1500
7210 PRINT AT 18,14;"270"
7230 INPUT "Graus? ";r
7240 IF r>315 OR r<0 THEN GO TO
7250
7250 LET r=INT (r/45)+1
7255 IF r=1 THEN RETURN
7260 LET g(u,p)=0
7270 LET s(u,p)=3
7280 LET u=u+b(r)
7290 LET p=p+c(r)
7300 IF g(u,p)<>1 THEN GO TO 111
0
7310 PRINT AT AND#21,0; FLASH 1;
INK 2; PAPER 6;"Voce aterrou no
na nave ";z#;"!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!"
7320 BEEP 1,3
7325 IF AND>.98 THEN GO TO 7310
7330 FOR a=1 TO AND*5
7340 PLOT 128,87
7350 DRAW 128-RND*255,87-RND*175
7360 NEXT a
7370 BEEP 3,-23
7380 GO TO 7320
0005 LET fl=1
0010 PRINT AT 14,14; INK 6; PAPE
R 2; FLASH 1;"ALERTA"
0020 GO SUB 1500; GO SUB 1500
0030 PRINT AT 15,0;c#(FN a(5));"
- Direccao de tiro:"
0040 GO SUB 7015
0045 LET e=e-100
0050 IF g(u+b(r),p+c(r))<>1 THEN
GO TO 0300
0070 BEEP 1,20
0080 PRINT AT 15,0; FOR a=1 TO
7: PRINT "
;0);

```

```

0090 PRINT c#(FN a(5));" - Atins
!000 0 ";z#;"!"
0095 LET s(u+b(r),p+c(r))=4
0097 LET g(u+b(r),p+c(r))=4
0100 LET at=at+1
0120 PRINT c#(FN a(5));" - Desli
00 c
alerta?"
0130 IF INKEY#<>" THEN GO TO 81
00
0135 IF INKEY#="" THEN GO TO 813
00
0140 GO TO 1110
0300 GO SUB 5000
0310 PRINT c#(FN a(5));" ";
0320 LET s(u+b(r),p+c(r))=3
0330 PRINT " Falhasos."
0340 FOR a=0 TO 30
0350 BEEP 1,a
0360 NEXT a
0370 BEEP 1,35
0380 FOR a=1 TO 50
0390 BORDER AND*7
0400 NEXT a
0410 BORDER 6
0420 PRINT ",c#(FN a(5));" Os "
;z#;" estao","a responder ao tir
o!"
0430 GO SUB 1500
0450 IF AND>.6 THEN GO TO 8760
0460 PRINT "ATINGIRAM-NOS!!!"
0470 LET e=e-100*AND
0480 GO TO 8110
0700 PRINT c#(FN a(5));" : mas fe
lizmente os"
0710 PRINT z#;" falharam."
0720 GO TO 8110
9020 DATA "Dr Sock","Spottie","T
. Loohu","Checkout","Zulu"
9030 DATA -1,0,-1,1,0,1,1,1,0,
1,-1,0,-1,-1,-1
9040 DATA "Glob","Ulian","Frax",
"tloid","Mesh","nik","Rom","ulan
","Grup","lish","Krett","on","ou
es","tar","Coch","rane"
9050 DATA 255,165,195,153,153,19
9,165,255
9060 DATA 255,169,219,255,255,21
9,189,255
9070 DATA 248,152,152,255,47,57,
249,223

```

TEMPESTADE LUNAR

Este programa, escrito por David Perry, utiliza métodos sofisticados de tratamento de cadeias alfanuméricas para obter os meteoros e naves que se movem lateralmente no visor.

Por baixo encontra-se a superfície do planeta McClariana (que deve o seu nome ao intrépido explorador McClaren que o descobriu em 2068). Você deve tentar descer através destes obstáculos, e aterrizar numa das três pistas.

Para simplificar a sua tarefa, extremamente difícil, você dispõe de um botão que regula o impulso dos motores permitindo-lhe manter-se na mesma posição quando a confusão por baixo é demasiado grande. O jogo utiliza bastante o som e a cor, além de alguns interessantes gráficos definidos pelo autor.

TEMPESTADE LUNAR

Indique nível (1 a 3)

5-Esquerda 8-Direita
 0-Impulso
Tente aterrizar nas pistas

```

1 REM
2 REM TEMPESTADE LUNAR
3 REM Copyright:
4 REM DAVID PERRY
5 REM
6 REM
7 REM

```

8 BORDER 0: PAPER 0: INK 7: C

9 LET sc=0: LET bo=100: LET i
=7: GO TO 45
10 CLS: RESTORE: GO SUB 29

```

11 LET x=0: LET y=INT (RND*31)
+1
12 PRINT AT x,y;" ": PRINT AT
3,0: INK 8;a##;AT 7,0;a##;AT 11,0;
b##;AT 15,0;b##;AT 19,0;b##
13 PRINT AT 5,0: INK 8;b##;AT 9
,0;b##;AT 13,0;b##;AT 17,0;b##
14 IF INT x>10 THEN GO TO 79
15 IF SCREEN$(x,y)<>" " THEN
GO TO 3: PRINT AT x,y;"@": REM 9
grafico C
16 BEEP .01,x*2
17 PRINT AT x,y;" "
18 LET x=x+.5: IF IN 51438=254
THEN LET x=x-.5: LET sc=sc-1
19 LET y=y+(INKEY$="8")-(INKEY
#="5")
20 IF y<0 THEN LET y=31
21 IF y>31 THEN LET y=0
22 IF x>10 THEN GO TO 79
23 IF SCREEN$(x,y)<>" " THEN
GO TO 38
24 LET sc=sc+1
25 PRINT AT x,y: INK 7: BRIGHT
0;"@": REM grafico C
26 LET a##=##(2 TO )+a##(1)
27 LET b##=b##(31)+b##( TO 30)
28 GO TO 12
29 PRINT AT 20,0: INK 2;"
30 PRINT AT 19,0: INK 3;"
FLASH 1;"": FLASH 0: INK 7;"
INK 3: FLASH 1;"": FLASH 0;"
FLASH 0;"": FLASH 1;"": INK 7;"
FLASH 0;"": INK 3: FLASH 1;"":
FLASH 0;"": FLASH 1;"":
SH 1;"": FLASH 0;"":
31 FOR n=1 TO 3: READ c#: FOR
a=0 TO 7: READ b: POKE USR,c#+a,
b: NEXT a: NEXT n
32 DATA "c",24,36,90,165,219,2
19,189,195
33 DATA "a",0,60,126,171,213,1
26,60,0
34 DATA "b",195,189,90,126,126
90,189,195
35 RETURN
36 FOR n=7 TO 0 STEP -1
37 PRINT AT x,y: INK n;"#": RE
M grafico B
38 BEEP .01,n*3: BEEP .01,50-n
*3

```

```

30 NEXT n
40 LET i=7
41 PRINT AT 2,8; INK 5;"PONTUA
CAO: ";sc
42 PRINT INK 1; BRIGHT 8; AT 0,
1;"Carregue em qualquer tecla!!!
43 LET i=i-1: IF i<0 THEN LET
i=7
44 IF INKEY$="" THEN GO TO 42
45 LET sc=0: CLS : LET d=0: BR
IGHT 1: INK 0
46 PRINT "
47 PRINT "
48 PRINT "
49 PRINT "
50 PRINT "
51 PRINT "
52 PRINT "
53 PRINT " INK 5; BRIGHT 0
54 PRINT "
55 PRINT "
56 PRINT "
57 PRINT "
58 PRINT "
60 PRINT AT 15,6; INK 1;"Indiq
ue nivel (1 a 3)"
61 PRINT AT 18,6; INK 1/2;"===
=====
62 PRINT AT 18,3; INK 4; INK 7
;"0-Esquerda      0-Direita"
63 PRINT AT 19,11; INK 4;"0-Im
pulso"
64 PRINT AT 20,4; INK 6;"Tente
aterrar nas pistas"
65 PRINT #1; BRIGHT 8; INK 7;"
TEMPESTADE LUNAR © D. PERRY 1983
66 PRINT AT 15,6; INK 1;"Indiq
ue nivel (1 a 3)"

```

```

67 LET a#=INKEY$: IF a#<"1" OR
a#>"3" THEN GO TO 76
68 IF a#="" THEN GO TO 76
69 IF a#="1" THEN LET a#=""
70 IF d=1 THEN LET a#=""
71 IF a#="2" THEN LET a#=""
72 IF d=2 THEN LET a#=""
73 IF a#="3" THEN LET a#=""
74 IF d=3 THEN LET a#=""
75 GO TO 10
76 LET i=i-1: IF i<0 THEN LET
i=7
77 GO TO 66
78 STOP
79 PRINT AT x,y; INK 7; BRIGHT
8;"0": IF y=4 THEN LET bo=150:
GO TO 83: REM grafico C
80 IF y=16 THEN LET bo=100: GO
TO 83
81 IF y=28 THEN LET bo=150: GO
TO 83
82 GO TO 36
83 LET sc=sc+bo: FOR n=1 TO 3:
FOR i=0 TO 7
84 PRINT AT 1,7; INK i;"P A R
A B E N S I !"
85 PRINT AT 3,7;"Bonus: ";bo;"
"
86 PRINT AT 5,11;"Pontuacao: "
;sc
87 NEXT i: NEXT n
88 FOR n=-20 TO 50 STEP 5: SEE
P .005,n: BEEP .001,50-ABS (n):
BEEP .005,n/2: NEXT n
89 CLS : GO TO 10

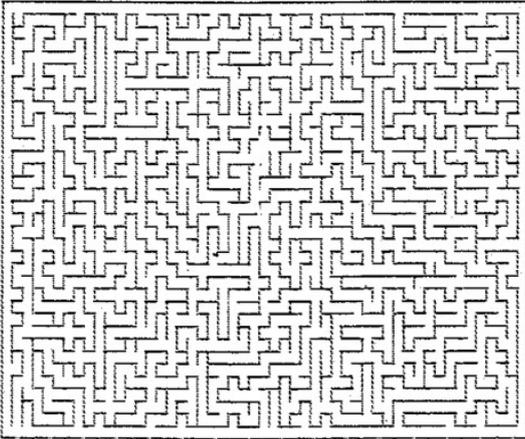
```

O PRAZER DO LABIRINTO

GERADOR DE LABIRINTOS

Vamos apresentar agora uma série de programas interessantes sobre labirintos, a maior parte dos quais escritos por Graham Charlton.

Começamos por um programa que desenha um labirinto de quaisquer dimensões, como o seguinte:

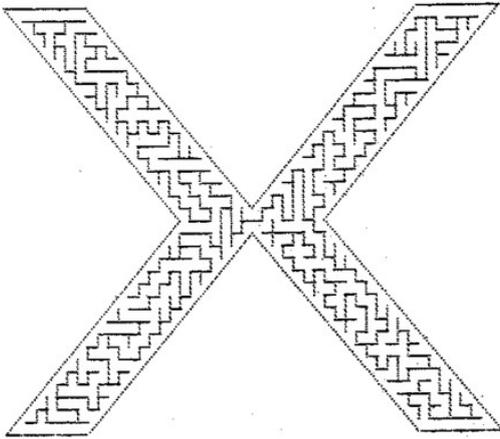


Quando o programa o pedir, indicamos um número qualquer entre 2 e 23 a fim de definir a complexidade do labirinto obtido. Notar que a produção do labirinto é demorada.

```
10 DIM a$(9999)
20 DIM b$(9999)
30 INPUT "Numero (2 a 23)? ";s
40 LET t=INT (255/s)*s
45 LET w=INT (175/s)*s
50 PLOT 0,0
60 DRAW t,0
70 DRAW 0,#
80 DRAW -t,0
90 DRAW 0,-#
100 LET x=s
110 LET y=s
120 LET z=1
130 PLOT x,y
140 IF (POINT (x+s,y)+POINT (x,
y-s)+POINT (x,y+s)+POINT (x-s,y)
)<>4 THEN GO TO 200
150 LET z=z-1
160 IF z=0 THEN STOP
170 LET x=CODE a$(z)
180 LET y=CODE b$(z)
190 GO TO 130
2000 LET a$(z)=CHR# x
2010 LET b$(z)=CHR# y
2020 LET z=z+1
2030 LET r=INT (RND*4)
2040 LET c=s*((r=0)-(r=1))
2050 LET d=s*((r=2)-(r=3))
2060 IF POINT (x+c,y+d) THEN GO
TO 230
2070 DRAW c,d
2080 LET x=x+c
2090 LET y=y+d
2100 GO TO 130
```

LABIRINTOS IRREGULARES

Não há qualquer razão para que as paredes do labirinto sejam perpendiculares entre si. Vamos apresentar em seguida quatro exemplos de labirintos irregulares, todos eles usando paredes com inclinações diversas.

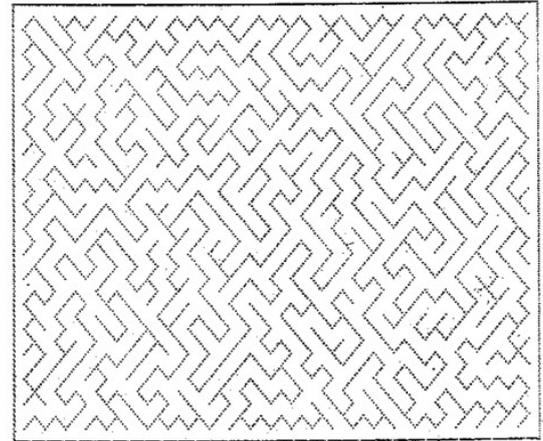


```

1000 DIM a#(10000)
1001 DIM b#(10000)
1002 INPUT "Numero (0 a 23)? "; s
1003 LET t=INT (200/s) #
1004 LET w=INT (170/s) #
1005 PLOT X,Y
1006 IF (POINT (x+s,y)+POINT (x-s,y)+POINT (x,y+s)+POINT (x,y-s)) <> 4 THEN GO TO 2000
1007 LET z=z-1
1008 IF z=0 THEN STOP
1009 LET x=COORD a#(z)
1010 LET y=COORD b#(z)
1011 GO TO 1300
2000 LET a#(z)=CHR# x
2010 LET b#(z)=CHR# y
2020 LET z=z+1
2030 LET r=INT (RND*4)
2040 LET c=s*((r=0)-(r=1))
2050 LET d=s*((r=2)-(r=3))
2060 IF POINT (x+c,y+d) THEN GO TO 2300
2070 DRAW c,d
2080 LET x=x+c
2090 LET y=y+d
2100 GO TO 1300
    
```

```

110 LET y=s*10
120 LET z=1
130 PLOT X,Y
140 IF (POINT (x+s,y)+POINT (x-s,y)+POINT (x,y+s)+POINT (x,y-s)) <> 4 THEN GO TO 2000
150 LET z=z-1
160 IF z=0 THEN STOP
170 LET x=COORD a#(z)
180 LET y=COORD b#(z)
190 GO TO 1300
2000 LET a#(z)=CHR# x
2010 LET b#(z)=CHR# y
2020 LET z=z+1
2030 LET r=INT (RND*4)
2040 LET c=s*((r=0)-(r=1))
2050 LET d=s*((r=2)-(r=3))
2060 IF POINT (x+c,y+d) THEN GO TO 2300
2070 DRAW c,d
2080 LET x=x+c
2090 LET y=y+d
2100 GO TO 1300
    
```



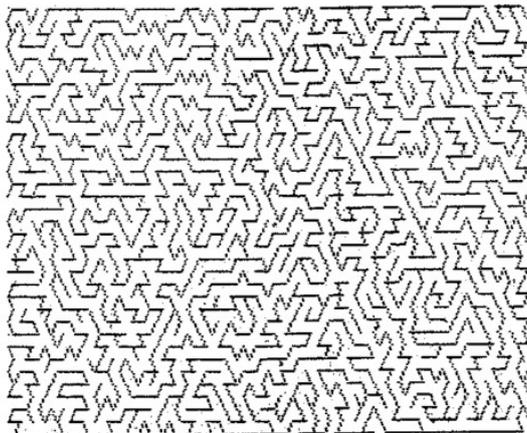
```

10 DIM a#(10000)
20 DIM b#(10000)
30 INPUT "Numero (0 a 23)? "; s
40 LET t=INT (200/s) #
50 LET w=INT (174/s) #
    
```

```

60 DRAW t,0
70 DRAW 0,#
80 DRAW -t,0
90 DRAW 0,-#
100 LET X=#
110 LET Y=#
120 LET Z=1
130 PLOT
140 IF (POINT (X+S,Y+S)+POINT (
X+S,Y-S)+POINT (X-S,Y+S)+POINT (
X-S,Y-S))<>4 THEN GO TO 200
150 LET Z=Z-1
160 IF Z=0 THEN STOP
170 LET X=CODE a#(Z)
180 LET Y=CODE b#(Z)
190 GO TO 130
2000 LET a#(Z)=CHR# X
2010 LET b#(Z)=CHR# Y
2020 LET Z=Z+1
2030 LET r=INT (RND*4)
2040 LET c=s*((r<2)-(r>1))
2050 LET d=s*((r=0)+(r=3)-(r=1)-
(r=2))
260 IF POINT (X+C,Y+D) THEN GO
TO 230
270 DRAW C,D
280 LET X=X+C
290 LET Y=Y+D
300 GO TO 130

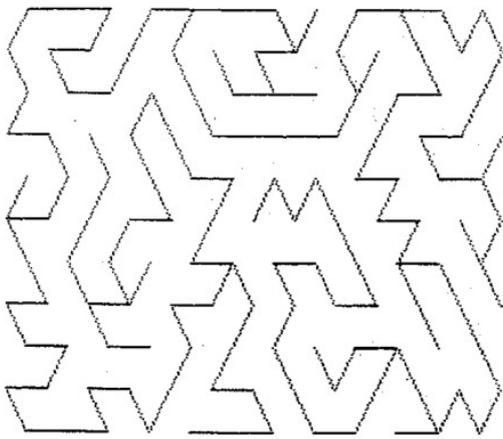
```



```

10 DIM a#(30000)
20 DIM b#(30000)
30 LET X=1000
40 LET Y=000
50 LET Z=1
60 INPUT "Numero (2 a 23)? ";s
70 LET t=s*2
80 LET q=INT (SOR (t*t-s*s))
100 PLOT X,Y
200 FOR c=-t TO t STEP t*2
210 IF X+C>255 OR X+C<0 THEN GO
TO 230
220 IF POINT (X+C,Y)=0 THEN GO
TO 3000
230 NEXT c
250 FOR c=-s TO s STEP t
260 IF X+C>255 OR X+C<0 THEN GO
TO 310
270 FOR d=-q TO q STEP q*2
280 IF Y+D>175 OR Y+D<0 THEN GO
TO 300
290 IF POINT (X+C,Y+D)=0 THEN G
O TO 3000
300 NEXT d
310 NEXT c
2000 LET Z=Z-1
2010 IF Z=0 THEN STOP
2020 LET X=CODE a#(Z)
2030 LET Y=CODE b#(Z)
2040 GO TO 100
3000 LET a#(Z)=CHR# X
3010 LET b#(Z)=CHR# Y
3020 LET Z=Z+1
3030 LET r=INT (RND*5)
3040 LET c=s*((r=1)+(r=5)-(r=0)-
(r=4))+t*((r=3)-(r=2))
3050 LET d=q*((r<2)-(r>3))
3060 IF X+C>255 OR X+C<0 THEN GO
TO 3030
3070 IF Y+D>175 OR Y+D<0 THEN GO
TO 3030
3080 IF POINT (X+C,Y+D) THEN GO
TO 3000
3090 DRAW C,D
3100 LET X=X+C
3110 LET Y=Y+D
3120 GO TO 100

```



MUROS DO SUSPENSE

Neste programa é necessário competir com dois opositores, controlados pelo computador, na tentativa de resolver o labirinto. Os dois opositores são representados por um «O», e você por um*. Deve usar as teclas de cursor, e partirá do canto superior esquerdo tentando atingir o canto inferior direito.

Ganha quem primeiro resolver o labirinto.

```

5 CLS : PRINT TAB 7;"Aguarde
um momento"
10 DIM c$(23,33)
12 DIM a$(1000)
14 DIM b$(1000)
20 LET x=13
30 LET y=17
40 LET z=1
50 FOR #=2 TO 22
60 LET c$(#)= "#####"
#####
70 NEXT #
100 LET c$(x,y)= " "
110 IF c$(x+2,y)="#" OR c$(x-2,

```

```

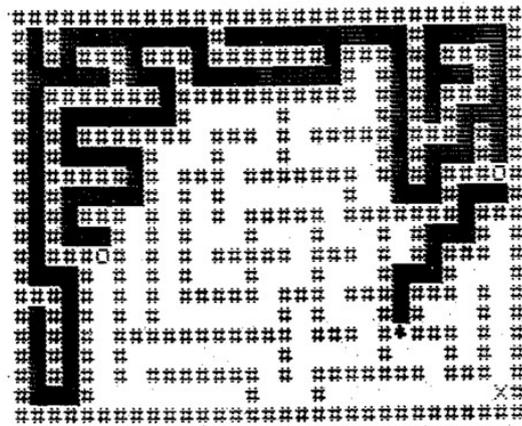
y)="#" OR c$(x,y+2)="#" OR c$(x,
y-2)="#" THEN GO TO 1000
1200 LET z=z-1
1300 IF z=0 THEN GO TO 2000
1400 LET x=ORDEN a$(z)
1500 LET y=ORDEN b$(z)
1600 GO TO 100
1700 LET a$(z)=CHR# x
1800 LET b$(z)=CHR# y
1900 LET z=z+1
2000 LET r=INT (RND*4)
2100 LET c=(r=0)-(r=1)
2200 LET d=(r=2)-(r=3)
2300 IF c$(x+2*c,y+2*d)="#" THEN
GO TO 1000
2400 LET c$(x+c,y+d)="#"
2500 LET c$(x+2*c,y+2*d)="#"
2600 LET x=x+2*c
2700 LET y=y+2*d
2800 GO TO 100
2900 CLS : FOR #=2 TO 22
3000 PRINT INK 1;c$(#,2 TO )
3100 NEXT #
3200 PRINT AT 19,29;"X"
3300 LET x=1
3400 LET y=1
3500 LET u=1
3600 LET v=1
3700 LET s=0
3800 LET t=0
3900 PRINT AT x,y;"*"
4000 LET z#=INKEY#
4100 LET c=(z#="6")-(z#="7")
4200 LET d=(z#="8")-(z#="5")
4300 IF SCREEN#(x+c,y+d)="#" TH
EN BEEP .1,0: GO TO 3500
4400 PRINT AT x,y; INK 5;"■"
4500 LET x=x+c
4600 LET y=y+d
4700 PRINT AT x,y;"*"
4800 IF x=19 AND y=29 THEN INPUT
"Desseja outro jogo?";v#: IF v#
="s" OR a#="6" THEN RUN
4900 IF x=19 AND y=29 THEN STOP
5000 LET s=s-1
5100 IF s=-1 THEN LET s=3
5200 PRINT AT p,q; INK 6;"■"
5300 IF s=4 THEN LET s=0
5400 LET c=(s=1)-(s=3)
5500 LET d=(s=0)-(s=2)

```

```

4030 IF SCREEN$(p+c,q+d)="#" TH
EN LET s=s+2: GO TO 4055
4050 LET p=p+c
4055 LET q=q+d
4060 PRINT AT p,q;"O"
4070 IF p=10 AND q=20 THEN INPUT
"Desaja outro jogo?";v$: IF v$
="s" OR a$="S" THEN RUN
4072 IF p=10 AND q=20 THEN STOP
5000 LET t=t-1
5070 IF t=-1 THEN LET t=3
5080 PRINT AT u,v; INK 5;"■"
5090 IF t=4 THEN LET t=0
5010 LET c=(t=0)-(t=2)
5020 LET d=(t=1)-(t=3)
5030 IF SCREEN$(u+c,v+d)="#" TH
EN LET t=t+2: GO TO 5055
5050 LET u=u+c
5055 LET v=v+d
5065 PRINT AT u,v;"O"
5070 IF u=10 AND v=20 THEN INPUT
"Desaja outro jogo?";v$: IF v$
="s" OR a$="S" THEN RUN
5072 IF u=10 AND v=20 THEN STOP
5080 GO TO 3100

```



Nesta versão do jogo, você enfrenta apenas um único oponente, igualmente controlado pelo computador. No entanto, este é bastante mais inteligente do que os do jogo anterior, e move-se ao dobro da velocidade a que você se pode deslocar.

```

5 CLS : PRINT TAB 7;"Aguarde
u# momentos"
10 DIM c$(23,33)
12 DIM a$(1000)
14 DIM b$(1000)
20 LET x=13
30 LET u=17
40 LET z=1
50 FOR w=2 TO 22
60 LET c$(w)="#" *****
***** "
70 NEXT w
100 LET c$(x,y)="#"
110 IF c$(x+2,y)="#" OR c$(x-2,
y)="#" OR c$(x,y+2)="#" OR c$(x,
y-2)="#" THEN GO TO 1000
120 LET z=z-1
130 IF z=0 THEN GO TO 2000
140 LET x=CODE a$(z)
150 LET y=CODE b$(z)
160 GO TO 100
1000 LET a$(z)=CHR$ x
1010 LET b$(z)=CHR$ y
1020 LET z=z+1
1030 LET r=INT (RND*4)
1040 LET c=(r=0)-(r=1)
1050 LET d=(r=2)-(r=3)
1060 IF c$(x+2*c,y+2*d)="#" THEN
GO TO 1000
1070 LET c$(x+c,y+d)="#"
1080 LET c$(x+2*c,y+2*d)="#"
1090 LET x=x+2*c
1100 LET y=y+2*d
1110 GO TO 100
2000 CLS : FOR w=2 TO 22
2010 PRINT INK 1;c$(w,2 TO )
2020 NEXT w
2030 PRINT AT 10,20;"X"
3000 LET x=1
3005 LET p=1
3010 LET y=1

```



```

AT 20,30;"X"
3020 LET Z#="LNKEY#"
3040 LET C#=(Z#="5")-(Z#="7")
3050 LET D#=(Z#="0")-(Z#="9")
3060 IF SCREEN# (X+C,U+D)="#" TH
EN GO TO 3020
3070 PRINT AT X,U;"■"
3080 LET X=X+C
3090 LET U=U+D
3120 IF U=0 THEN GO SUB 5000
3130 IF U=31 THEN GO SUB 5010
3140 IF X=0 THEN GO SUB 5020
3150 IF X=21 THEN GO SUB 5030
3160 PRINT FLASH 1;AT X,U;"■"
3170 IF X=20 AND U=30 AND P=23 A
ND Q=33 THEN INPUT "Desseja b
otctr o
Jogo? ";V#; IF V#="S" OR B#="0"
THEN RUN
3171 IF X=20 AND U=30 AND P=23 A
ND Q=33 THEN STOP
3180 GO TO 3010
5000 RANDOMIZE USA 60021
5010 LET U=U+1
5015 LET Q=Q-1
5020 FOR #=0 TO 21
5030 PRINT AT #,0;C#(P+#,Q)
5040 NEXT #
5050 RETURN
5100 RANDOMIZE USA 60030
5110 LET U=U-1
5115 LET Q=Q+1
5120 FOR #=0 TO 21
5130 PRINT AT #,31;C#(P+#,Q+31)
5140 NEXT #
5150 RETURN
5200 RANDOMIZE USA 60111
5210 LET X=X+1
5215 LET P=P-1
5220 PRINT AT 0,0;C#(P,Q TO Q+31)
)
5250 RETURN
5300 RANDOMIZE USA 60043
5310 LET X=X-1
5315 LET P=P+1
5320 PRINT AT 21,0;C#(P+21,Q TO
Q+31)
)
5350 RETURN
9000 DATA 33,0,54,85,52,19,92,6,31
9010 DATA 35,24,43,115,35,15,24,9
9020 DATA 114,35,51,32,24,251
9030 DATA 33,25,87,20,6,50,19,2
9040 DATA 5,31,43,24,35,115,43
9050 DATA 16,24,114,43,51,32

```

168

```

0130 DATA 2,2,2,51
0140 DATA 3,9,2,17,3,0
0150 DATA 14,4,2,7,5,3,2,5,4,2,10
0160 DATA 14,4,2,7,5,3,2,5,4,2,10
0170 DATA 14,4,2,7,5,3,2,5,4,2,10
0180 DATA 14,4,2,7,5,3,2,5,4,2,10
0190 DATA 14,4,2,7,5,3,2,5,4,2,10
0200 DATA 14,4,2,7,5,3,2,5,4,2,10
0210 DATA 14,4,2,7,5,3,2,5,4,2,10
0220 DATA 14,4,2,7,5,3,2,5,4,2,10
0230 DATA 14,4,2,7,5,3,2,5,4,2,10
0240 DATA 14,4,2,7,5,3,2,5,4,2,10
0250 DATA 14,4,2,7,5,3,2,5,4,2,10
0260 DATA 14,4,2,7,5,3,2,5,4,2,10
0270 DATA 14,4,2,7,5,3,2,5,4,2,10
0280 DATA 14,4,2,7,5,3,2,5,4,2,10
0290 DATA 14,4,2,7,5,3,2,5,4,2,10
0300 DATA 14,4,2,7,5,3,2,5,4,2,10
0310 DATA 14,4,2,7,5,3,2,5,4,2,10
0320 DATA 14,4,2,7,5,3,2,5,4,2,10
0330 DATA 14,4,2,7,5,3,2,5,4,2,10
0340 DATA 14,4,2,7,5,3,2,5,4,2,10
0350 DATA 14,4,2,7,5,3,2,5,4,2,10
0360 DATA 14,4,2,7,5,3,2,5,4,2,10
0370 DATA 14,4,2,7,5,3,2,5,4,2,10
0380 DATA 14,4,2,7,5,3,2,5,4,2,10
0390 DATA 14,4,2,7,5,3,2,5,4,2,10
0400 DATA 14,4,2,7,5,3,2,5,4,2,10

```

LABIRINTO ROLANTE

O programa de labirinto que se segue foi escrito por Neil Bellinacci. Foi escrito para a máquina de 16 K. Também pode ser executado no computador de 48 k, mas convirá não o modificar dada a dificuldade de fazer alterações nas rotinas em código.

O objectivo do jogo consiste em atravessar um labirinto produzido aleatoriamente, primeiro de cima para baixo e depois voltando novamente ao topo no menor tempo possível. Para dar maior vivacidade ao jogo, partes do labirinto rolam para a esquerda e o resto para a direita. Você, no entanto, não rola para lado algum.

Você dispõe de três vidas, e perde uma sempre que colide com uma parede à esquerda ou à direita. Ouve neste caso um som. O número de vidas restantes, juntamente com o tempo gasto e o melhor tempo conseguido anteriormente, é apresentado no topo da imagem.

No início é apresentada uma seqüência de instruções, indicando quais as teclas que devemos usar, os dez melhores tempos e os nomes dos jogadores que os obtiveram.

169

Se carregar em «K» em qualquer momento desta sequência inicial pode alterar as teclas de comando. Siga simplesmente as instruções que lhe forem dadas para o efeito. «S» permitir-lhe-á sair das páginas de título e iniciar o jogo.

O programa tem cinco níveis de dificuldade. Sugiro-lhe que comece pelo primeiro, e que observe depois o nível cinco para se aperceber da diferença. O relógio começa a funcionar assim que o labirinto surge. Se o seu tempo final se encontrar entre os dez primeiros, ser-lhe-á pedido que indique o seu nome, que depois será incorporado na lista dos dez melhores.

Para introduzir o jogo na máquina, proceda do modo indicado em seguida, não se esquecendo de verificar sempre as gravações que fizer:

1 — Introduza o programa um, que carrega o código-máquina, e execute-o. Deve obter no final a mensagem «9 STOP statement» se o programa estiver correcto. Grave em seguida este programa em cassette, porque pode necessitar dele mais tarde.

```

100 REM código-máquina
110 POKE 20000,100
120 CLEAR 0
130 READ w:addr,byte
140 READ byte: IF byte=-1 THEN
GO TO 300
150 IF byte=000 THEN STOP
160 POKE w,byte
170 LET addr=addr+1
180 GO TO 140
190 DATA 30,200,00,00,04,200,100
200,00,100,04,200,100,100,00,00,200
210,04,200,100,100,00,100,04,200,200
220,100,00,200,200,100,00,200,00,
230,00,200,200,100,00,200,100,1
240 DATA 25,200,200,100,20,200,
250,100,20,200,200,100,20,200,10
260,100
270 DATA 30,0,00,200,200,100,20
280,100,100,100,200,200,00,100,20,20
290,100,100,100,200,200,100,100,1
300 DATA 30,200,107,213,200,245,
310,100,0,0,107,40,20,100,0,31
320,245,107,207,02,30,241,0,30,0,31

```

```

330,30,10,200,107,207,00,30,100,1
340,30,10,200,107,207,00,30,100,1
350,30,10,200,107,207,00,30,100,1
360,30,10,200,107,207,00,30,100,1
370,30,10,200,107,207,00,30,100,1
380,30,10,200,107,207,00,30,100,1
390,30,10,200,107,207,00,30,100,1
400,30,10,200,107,207,00,30,100,1
410,30,10,200,107,207,00,30,100,1
420,30,10,200,107,207,00,30,100,1
430,30,10,200,107,207,00,30,100,1
440,30,10,200,107,207,00,30,100,1
450,30,10,200,107,207,00,30,100,1
460,30,10,200,107,207,00,30,100,1
470,30,10,200,107,207,00,30,100,1
480,30,10,200,107,207,00,30,100,1
490,30,10,200,107,207,00,30,100,1
500,30,10,200,107,207,00,30,100,1
510,30,10,200,107,207,00,30,100,1
520,30,10,200,107,207,00,30,100,1
530,30,10,200,107,207,00,30,100,1
540,30,10,200,107,207,00,30,100,1
550,30,10,200,107,207,00,30,100,1
560,30,10,200,107,207,00,30,100,1
570,30,10,200,107,207,00,30,100,1
580,30,10,200,107,207,00,30,100,1
590,30,10,200,107,207,00,30,100,1
600,30,10,200,107,207,00,30,100,1
610,30,10,200,107,207,00,30,100,1
620,30,10,200,107,207,00,30,100,1
630,30,10,200,107,207,00,30,100,1
640,30,10,200,107,207,00,30,100,1
650,30,10,200,107,207,00,30,100,1
660,30,10,200,107,207,00,30,100,1
670,30,10,200,107,207,00,30,100,1
680,30,10,200,107,207,00,30,100,1
690,30,10,200,107,207,00,30,100,1
700,30,10,200,107,207,00,30,100,1
710,30,10,200,107,207,00,30,100,1
720,30,10,200,107,207,00,30,100,1
730,30,10,200,107,207,00,30,100,1
740,30,10,200,107,207,00,30,100,1
750,30,10,200,107,207,00,30,100,1
760,30,10,200,107,207,00,30,100,1
770,30,10,200,107,207,00,30,100,1
780,30,10,200,107,207,00,30,100,1
790,30,10,200,107,207,00,30,100,1
800,30,10,200,107,207,00,30,100,1
810,30,10,200,107,207,00,30,100,1
820,30,10,200,107,207,00,30,100,1
830,30,10,200,107,207,00,30,100,1
840,30,10,200,107,207,00,30,100,1
850,30,10,200,107,207,00,30,100,1
860,30,10,200,107,207,00,30,100,1
870,30,10,200,107,207,00,30,100,1
880,30,10,200,107,207,00,30,100,1
890,30,10,200,107,207,00,30,100,1
900,30,10,200,107,207,00,30,100,1
910,30,10,200,107,207,00,30,100,1
920,30,10,200,107,207,00,30,100,1
930,30,10,200,107,207,00,30,100,1
940,30,10,200,107,207,00,30,100,1
950,30,10,200,107,207,00,30,100,1
960,30,10,200,107,207,00,30,100,1
970,30,10,200,107,207,00,30,100,1
980,30,10,200,107,207,00,30,100,1
990,30,10,200,107,207,00,30,100,1

```

2 — Escreva novamente RUN. Quando o programa parar, escreva NEW para o eliminar. O código-máquina manter-se-á mesmo assim em memória.

3 — Escreva agora o programa que se segue, o programa principal. Depois grave-o em cassette, imediatamente à frente do anterior. Para o gravar utilize a instrução SAVE «Labirinto» — não recorra à linha 9600 do programa.

```

1 REM ** Labirinto **
2 © Neil Pellinacci
300 CLS : LET q#="Indique difi-
4 cidade (1-5)"
500 INPUT (q#) lev
600 IF lev<1 OR lev>5 OR lev<>I
700 NT lev THEN LET q#="Indique um n-
8 umero inteiro!": GO TO 510
900 POKE 32551,8 : POKE 32555,1 :
1000 POKE 32557,1 : POKE 32555,1
1100 BORDER 2 : PAPER 7 : CLS
1200 PRINT AT 0,0 : PAPER 2 : INK
1300 7 : " LABIRINTO ROLANTE
1400 "
1500 PRINT AT 1,0 : INK 6 : PAPER
1600 0 : "Top.> 0 Vidas> 3 Max.>
1700 "
1800 PRINT INK 6 : PAPER 0 : OVER
1900 1 : AT 1,4 : "<"; AT 1,17 : "<"; AT 1,27
2000 : "<"

```

```

040 PRINT AT 1,29; PAPER 0; INK
0:;H(1)
050 PRINT AT 2,0; INK 1;S#;AT 2
1,0;S#
060 GO SUB 8000
1000 INK 7
1010 FOR a=4 TO 20 STEP 2: PRINT
AT a,0; "*****"
*****": NEXT a: REM grafico
1030 FOR a=1 TO lev: FOR b=3 TO
20 STEP 2
1040 LET b1=INT (RND*30)+1: PRIN
T AT b,b1; " "
1050 NEXT b: NEXT a
1060 FOR a=1 TO lev: FOR b=4 TO
20 STEP 2
1070 LET b1=INT (RND*30)+1: PRIN
T AT b,b1; INK 2; " "
1080 NEXT b: NEXT a
1090 FOR a=3 TO 20: PRINT AT a,0
; OVER 1; INK 2;S#; NEXT a
1100 INK 0: RETURN
1000 GO SUB 800
1910 LET lives=3: LET t=0
2000 LET fin=21: LET l=2: LET c=
15
2010 PRINT AT 1,6; PAPER 0; INK
0;t;AT l,c; INK 2; PAPER 7;" "
2020 LET a=USR 32300: IF a=5 THE
N GO TO 2120
2025 IF a=1 THEN IF l>2 THEN IF
SCREEN# (l-1,c)<>" THEN LET l=l
-1
2030 IF a=2 THEN IF l<21 THEN IF
SCREEN# (l+1,c)<>" THEN LET l=
l+1
2035 LET c=c+(a=4 AND c<31)-(a=3
AND c>0)
2040 IF SCREEN# (l,c)="" THEN LE
T a=USR 32550: GO TO 2100
2050 IF l<>fin THEN PRINT AT l,c
; INK 1;"A": LET t=t+1: GO TO 20
10: REM grafico A
2060 IF fin=21 THEN LET fin=2: G
O TO 2010
2070 FOR b=1 TO 2: FOR a=40 TO -
20 STEP -2: BEEP ,004,a: BEEP ,0
04,a-15: BEEP ,004,30-a: BEEP ,0
04,30-a-15: NEXT a: NEXT b
2080 GO TO 2200
2100 LET lives=lives-1: PRINT AT

```

```

1,19; PAPER 0; INK 8;lives
2110 IF lives<>0 THEN GO TO 2000
2120 PRINT AT 6,0; INK 1;S#;AT 1
5,0;S#
2130 PRINT AT 6,9; INK 8;"JOGO T
ERMINADO";AT 16,9;"JOGO TERMINAD
O"
2140 FOR a=1 TO 256: LET b=USR 3
2300: NEXT a
2150 FOR a=1 TO 150: NEXT a: GO
TO 4000
2200 FOR a=1 TO 10: IF t<h(a) TH
EN GO TO 2230
2210 NEXT a: GO TO 4000
2230 FOR b=0 TO a STEP -1: LET h
(b+1)=h(b): LET h#(b+1)=h#(b): N
EXT b
2240 LET h(a)=t: INPUT "Escreva
o seu nome:"; LINE n#: LET h#(a)
=d#: LET h#(a, TO LEN n#+1)=n#
2250 GO SUB 8000: GO TO 4070
4000 BORDER 0: PAPER 0: CLS: RE
M paginas de titulo
4010 GO SUB 5100: LET ink=6: GO
SUB 4500: REM titulo
4020 FOR c=1 TO 20: GO SUB 4500:
IF CHR# PEEK 23556="5" OR CHR#
PEEK 23556="K" THEN GO TO 4100
4030 NEXT c
4040 GO SUB 5000: REM opcoes
4050 FOR c=1 TO 20: GO SUB 4500:
IF CHR# PEEK 23556="5" OR CHR#
PEEK 23556="K" THEN GO TO 4100
4060 NEXT c
4070 GO SUB 5500: REM pontuacoes
maximas
4080 FOR a=1 TO 500: IF CHR# PEE
K 23556="5" OR CHR# PEEK 23556="
K" THEN GO TO 4100
4090 NEXT a: GO TO 4000
4100 LET a=USR 32500: IF CHR# PE
EK 23556="5" THEN GO SUB 4200: G
O TO 1900
4110 GO SUB 4200: GO TO 4300
4200 FOR a=1 TO 34: LET b=USR 32
250+USR 32400: NEXT a: RETURN
4300 REM mudar teclas
4310 INK 0: BORDER 6: PAPER 6: C
LS: BORDER 3
4315 POKE 32557,1: POKE 32551,10
: POKE 32556,1: POKE 32555,10
4320 PRINT AT 1,9; INK 0;"DEFINI

```

```

R TECLAS"
4330 PRINT " INK 1;"Indique nov
a tecla quando for";TAB 9;"pedid
o"
4340 PRINT " INK 2;TAB 11; PAP
ER 7;"Subir";TAB 21;
4350 LET k=PEEK 23555: IF k=255
THEN GO TO 4330
4360 POKE 32311,k: PRINT PAPER 7
;CHR# k
4365 LET a=USR 32550: FOR a=1 TO
150: NEXT a
4370 PRINT " INK 2;TAB 11; PAPER
7;"Descer";TAB 21;
4380 LET k=PEEK 23555: IF k=255
THEN GO TO 4350
4390 POKE 32319,k: PRINT PAPER 7
;CHR# k
4395 LET a=USR 32550: FOR a=1 TO
150: NEXT a
4400 PRINT " INK 2;TAB 11; PAPER
7;"Esquerda";TAB 21;
4410 LET k=PEEK 23555: IF k=255
THEN GO TO 4400
4420 POKE 32327,k: PRINT PAPER 7
;CHR# k
4425 LET a=USR 32550: FOR a=1 TO
150: NEXT a
4430 PRINT " INK 2;TAB 11; PAPER
7;"Direita";TAB 21;
4440 LET k=PEEK 23555: IF k=255
THEN GO TO 4430
4450 POKE 32335,k: PRINT PAPER 7
;CHR# k
4455 LET a=USR 32550: FOR a=1 TO
150: NEXT a
4460 PRINT " INK 2;TAB 11; PAPER
7;"Abortar";TAB 21;
4470 LET k=PEEK 23555: IF k=255
THEN GO TO 4470
4480 POKE 32343,k: PRINT PAPER 7
;CHR# k
4490 LET a=USR 32550: GO TO 4000
4500 INK ink: RESTORE 4000: FOR
a=0 TO 2: READ a#: PRINT AT a,14
;a#: NEXT a
4510 FOR a=3 TO 5: READ a#: PRIN
T AT a,4;a#: NEXT a
4520 FOR a=7 TO 10: READ a#: PRI
NT AT a,5;a#: NEXT a
4530 PRINT INK 7;AT 10,2;"Escrit
o por Neil Pellinacci."

```

```

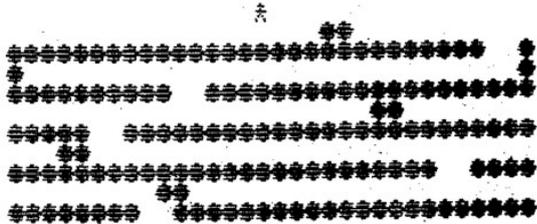
4540 LET ink=ink+1: IF ink=8 THE
N LET ink=1
4550 INK 0: RETURN
4560 DATA " "
4570 DATA " "
4580 DATA " "
4590 DATA " "
4600 DATA " "
4610 DATA " "
4620 DATA " "
4630 DATA " "
4640 DATA " "
4650 DATA " "
4660 DATA " "
4670 DATA " "
4680 DATA " "
4690 DATA " "
4700 DATA " "
5000 REM opcoes
5010 FOR a=10 TO 21: PRINT AT a,
0;a#: NEXT a
5020 INK 5: PRINT AT 15,10;"Carr
egue:";AT 17,10; BRIGHT 1;"S"; B
RIGHT 0;" para comecar";AT 19,7;
BRIGHT 1;"K"; BRIGHT 0;" para #
udar teclas"
5030 RETURN
5100 INK 6: PRINT AT 15,8;"COMAN
DOS DE JOGO";AT 17,5;"- ESQUERDA
";AT 17,21;"- SUBIR";AT 19,5;"-
DIREITA";AT 19,21;"- DESCER"
5110 PRINT BRIGHT 1;AT 17,3;CHR#
PEEK 32327;AT 17,19;CHR# PEEK 3
2311;AT 19,3;CHR# PEEK 32335;AT
19,19;CHR# PEEK 32319
5120 PRINT AT 21,11; BRIGHT 1;CH
R# PEEK 32343; BRIGHT 0;"- ABORT
AR"
5130 INK 0: RETURN
5500 BORDER 0: PAPER 0: CLS : IN
K 7: PRINT AT 0,6; INK 5; BRIGHT
1;"Melhores pontuacoes:"
5510 FOR a=1 TO 10: PRINT " " A
ND a<>10;a;" " ;h$(a);". " ;h(a):
NEXT a
5515 PRINT #0; INK 5;" Carr
egue "; BRIGHT 1;"S"; BRIGHT 0;"
ou "; BRIGHT 1;"K"
5520 RETURN
6000 REM

```

```

8010 RESTORE 8400
8020 READ a,b: IF a=99 THEN RETU
RN
8025 IF a=88 THEN PAUSE 5+5: GO
TO 8020
8030 BEEP a/3,b-3+lev: GO TO 802
0
8400 DATA .3,4,.3,4,.6,10,.3,4,.
3,4,.6,10,.3,4,.3,4,.9,10.1
8410 DATA 80,19
8420 DATA .4,10,.4,12,.4,6,.4,6,
.8,4,5
8430 DATA 88,20,.5,0,.6,4
8500 DATA 99,0
9000 REM U49
9010 RESTORE 9000: FOR a=USR "a"
TO USR "b"+7: READ b: POKE a,b:
NEXT a
9030 DATA 10,55,15,124,16,40,68,
68,24,126,106,255,255,126,126,24
9050 LET s#=" "
9060 DIM h(10): DIM h$(10,20)
9070 FOR a=1 TO 10: LET h$(a)="N
eil Pellinacci...": NEXT a
9080 FOR a=1 TO 10: READ b: LET
h(a)=b: NEXT a
9090 DATA 350,390,430,470,510,55
0,590,630,720,1000
9100 LET d$="....."
9200 GO TO 4800
9500 FOR z=32550 TO 32580: PRINT
z;"",PEEK z: NEXT z
9610 STOP
9620 SAVE "labirinto" LINE 9700:
SAVE "codigo"CODE 32000,500
9610 STOP
9700 CLEAR 31999: LOAD ""CODE :
GO TO 9000

```



4 — Escreva RUN 9000. Quando surgir a página de título, carregue em «S» para jogar. Se o programa estoirar, isso significará que você cometeu um erro em algum lado. Se o programa executar normalmente, tente redefinir as teclas. Verifique tudo.

5 — Quando tiver a certeza de que o programa está bom, faça a sua cópia final, usando RUN 9600. Depois verifique esta cópia usando as instruções VERIFY"" e VERIFY ""CODE. Execute novamente o jogo, quer carregando-o, quer escrevendo RUN 9000.

O programa, como é óbvio, encontra-se escrito em código-máquina e em Basic. A parte em código-máquina lê o teclado, desloca o labirinto, trata os sons e executa certos outros efeitos especiais. Diversos POKE's existentes no programa principal controlam a execução destas rotinas. A linha 9500 pode ser usada a fim de verificar os valores que constituem o código-máquina, desde que o leitor defina os endereços apropriados. Existe suficiente espaço no mapa da memória usada para o código-máquina para lhe permitir incorporar as suas próprias rotinas de efeitos especiais, se o quiser fazer.

Se dispuser de um computador com uma memória de 48 K e quiser colocar o código-máquina num endereço mais elevado, deve recordar o seguinte:

- 1) É necessário alterar os endereços da função USR citados no programa principal.
- 2) Deve alterar alguns dos bytes do programa em código-máquina, dado que algumas das rotinas chamam outras. Para aqueles que sabem alguma coisa do assunto, será necessário alterar os dois bytes que se seguem às instruções CALL (205, em decimal).
- 3) Faça os ajustamentos necessários no programa carregador do código-máquina.

LABIRINTO A 3 DIMENSÕES

Este interessante programa foi escrito por Scott Vincent. Necessitará de um computador de 48 k para o executar. Na memória do computador é guardado um labirinto de dez por dez, com uma única entrada/saída. O objectivo consiste em percorrer o labirinto, descobrir o tesouro e voltar depois ao ponto por onde se começou.

Se o leitor perde o sentido da direcção em que se move, pode carregar na tecla «R» para repetir todos os seus passos. O programa repetirá então todos os seus movimentos desde o início, mostrando-os rapidamente. Quando a repetição tiver terminado, você pode continuar o seu caminho.

Deverá usar as seguintes teclas para se deslocar no interior do labirinto:

- 1 — Mover para a esquerda
- 2 — Mover para a direita
- 0 — Avançar
- L — Virar-se

Note que deve carregar na tecla 4 para rodar o corpo; não pode usar para este efeito as teclas 1 ou 2.

Se e quando conseguir sair do labirinto trazendo o tesouro consigo, o computador imprimirá o número de movimentos de que necessitou para o efeito.

Note que o labirinto não é produzido aleatoriamente, ao contrário do que se passa nos programas do mesmo tipo já apresentados anteriormente. O jogador tem assim oportunidade de se habituar progressivamente a ele, dado que no caso contrário seria extremamente difícil resolver o labirinto. No entanto, o tesouro é colocado de cada vez num local diferente.

```

5 BORDER 5: PAPER 5: INK 1: 0
L5
10 RANDOMIZE : DIM m(9): DIM n
(9): DIM m$(201,9): DIM n$(200,1
2)
20 FOR x=0 TO 31: READ y: POKE
USR "a"+x,y: NEXT x
30 FOR x=1 TO 9: READ m(x): NE
XT x
40 FOR x=1 TO 9: READ n(x): NE
XT x
50 FOR x=1 TO 200: READ m$(x):
NEXT x
60 FOR x=1 TO 200: READ n$(x):
NEXT x
70 LET q$="001": LET p$="": LE
T l=1: LET tr=0: LET x=1+INT (RN
D*7)
80 LET la=201*(x=1 OR x=4 OR x
=6 OR x=7)+81*(x=2)+127*(x=3)+16
9*(x=5)
90 LET lb=201*(x=4 OR x=5)+100
*(x=1)+84*(x=2)+122*(x=3)+172*(x
=5)+195*(x=7)
100 LET lc=94*(x=1)+98*(x=2)+12

```

```

4*(x=3)+131*(x=4)+117*(x=5)+191*
(x=6)+199*(x=7)
110 CLS : FOR x=1 TO 9: IF m$(l
,x)="" THEN GO SUB m(x)
120 NEXT x
130 FOR x=1 TO 9: IF m$(l,x)=""
THEN GO SUB n(x)
140 NEXT x
150 IF l=2 OR l=la OR l=lb OR l
=lc THEN GO SUB 220: IF l=2 AND
tr=1 THEN GO TO 70
160 LET a$=INKEY$: IF a$="" THE
N GO TO 160
170 LET p=VAL n$(l, TO 3)*(a$=""
1")+VAL n$(l,4 TO 6)*(a$=""0")+VA
L n$(l,7 TO 9)*(a$=""2")+VAL n$(l
,10 TO )*(a$=""l")
180 IF a$=""r" THEN GO TO 300
190 IF p=0 THEN GO TO 160
200 BEEP ,008,0: LET l=p: LET q
#=q#+("0" AND (x100)+("0" AND (x
10)+STR$ l
210 LET p=p#+a$: GO TO 110
220 IF l=2 AND tr=0 THEN CLS
PRINT AT 10,12: FLASH 1:"ENTRADA
": RETURN
230 IF l=2 THEN CLS : PRINT AT
8,4:"Optimo, necessitou de";AT 1
0,11;LEN p$:" movimentos.";AT 14
,6:"Carregue numa tecla.": PAUSE
464: RETURN
240 IF l=la THEN PLOT 116,99: D
RAW OVER 1;22,0: PLOT 116,76: DR
AW OVER 1;22,0: RETURN
250 IF l=lb THEN PLOT 100,115:
DRAW OVER 1;54,0: PLOT 100,68: D
RAW OVER 1;54,0: RETURN
260 IF l=lc THEN PLOT 72,143: D
RAW OVER 1;110,0: PLOT 72,32: DR
AW OVER 1;110,0: RETURN
270 IF tr=1 THEN PRINT AT 9,11:
"Voce ja";AT 10,12:"tem o";AT 1
1,12:"tesouro.": RETURN
280 LET tr=1: PRINT AT 9,11:"Vo
ce acaba";AT 10,11:"de descobrir
":AT 11,11:"o tesouro."
290 RETURN
300 LET y=1
310 FOR n=1 TO LEN q$ STEP 3
320 IF n=1 THEN GO TO 390
330 IF p$(y)=""1" THEN PRINT #1;
AT 0,6;"←"

```

```

340 IF p$(y)="0" THEN PRINT #1;
AT 0,6;↑
350 IF p$(y)="2" THEN PRINT #1;
AT 0,6;↓
360 IF p$(y)="(" THEN PRINT #1;
AT 0,6;↓
370 LET y=y+1
380 FOR x=1 TO 25 STEP 5: FOR s
=x TO x+4 STEP 1.5: BEEP .004,x:
BEEP .004,x+s: NEXT s: NEXT x
390 LET l=VAL a$(n TO n+2)
400 CLS: FOR x=1 TO 9: IF ##(l
,x)="0" THEN GO SUB #(x)
410 NEXT x
420 FOR x=1 TO 9: IF ##(l,x)="1
" THEN GO SUB n(x)
430 NEXT x
440 IF l=2 OR l=1a OR l=1b OR l
=c THEN GO SUB 220
450 NEXT n: GO TO 160
5000 PLOT 40,175: DRAW 32,-32: P
LOT 40,0: DRAW 32,32: RETURN
5005 PLOT 24,175: DRAW 0,-175: D
RAW 0,32: DRAW 48,0: DRAW 0,111:
DRAW -48,0: RETURN
5010 PLOT 183,143: DRAW 32,32: P
LOT 183,32: DRAW 32,-32: RETURN
5015 PLOT 231,175: DRAW 0,-175:
DRAW 0,32: DRAW -48,0: DRAW 0,11
1: DRAW 48,0: RETURN
5020 PLOT 73,142: DRAW 27,-27: P
LOT 73,33: DRAW 27,27: RETURN
5025 PLOT 73,142: DRAW 11,-11: D
RAW 0,-37: DRAW -11,-11: PLOT 85
,88: DRAW 15,0: DRAW 0,55: DRAW
-15,0: RETURN
5030 PLOT 155,115: DRAW 27,27: P
LOT 182,33: DRAW -27,27: RETURN
5035 PLOT 182,33: DRAW -11,11: D
RAW 0,87: DRAW 11,11: PLOT 170,6
0: DRAW -15,0: DRAW 0,55: DRAW 1
5,0: RETURN
5040 PLOT 101,114: DRAW 19,-19:
PLOT 101,61: DRAW 19,19: RETURN
5045 PLOT 101,114: DRAW 8,-8: DR
AW 0,-37: DRAW -8,-8: PLOT 110,9
9: DRAW 6,0: DRAW 4,-4: DRAW -4,
4: DRAW 0,-23: DRAW -6,0: DRAW 6
,0: DRAW 4,4: RETURN
5050 PLOT 154,114: DRAW -19,-19:
PLOT 154,61: DRAW -19,19: RETUR
N

```

```

5255 PLOT 154,114: DRAW -8,-8: D
RAW 0,-37: DRAW 8,-8: PLOT 145,9
9: DRAW -8,0: DRAW -4,-4: DRAW 4
,4: DRAW 0,-23: DRAW 6,0: DRAW -
6,0: DRAW -4,4: RETURN
5060 PLOT 73,143: DRAW 110,0: DR
AW OVER 1;0,-110: DRAW 0,-1: DR
AW -111,0: DRAW OVER 1;0,110: RET
URN
5065 PLOT 101,115: DRAW 54,0: DR
AW OVER 1;0,-54: DRAW 0,-1: DRAW
-55,0: DRAW OVER 1;0,54: RETURN
5070 OVER 1: PLOT 117,77: DRAW 3
,3: PLOT 138,98: DRAW -3,-3: PLO
T 138,77: DRAW -3,3: PLOT 128,95
: DRAW -4,4: OVER 0: DRAW 23,0:
DRAW OVER 1;0,-22: DRAW 0,-1: DR
AW -23,0: DRAW OVER 1;0,23: RETU
RN
5080 DATA 16,48,96,255,255,96,48
,16,24,60,126,219,24,24,24,24
5090 DATA 8,12,6,255,255,6,12,8,
24,24,24,24,219,126,60,24
5000 DATA 5000,5010,290,5020,503
0,290,5040,5050,290
5010 DATA 5005,5015,5060,5025,50
35,5065,5045,5055,5070
5020 DATA "111","000000010","
101","000010000","010101","01000
0011","000010101","000010111","0
00011","000000010","011","100000
000","111","000100000","101","10
1","001","010101","101","100001"
,"011","001","011","101"
5030 DATA "100000000","011","010
011","000110011","011","011","10
1","100011","101","000111","111"
,"011","111","000111","000011","
1000000111","010000101","101","11
1","000101","010101","101","1000
10101","001","000100010"
5040 DATA "011","010111","111","
011","100010000","011","001","10
1","110011","111","011","000111"
,"001","001","110001","110000111
","001","101","110001","111","00
0010100","101","010100101","0101
01","111"
5050 DATA "100101","000010101","
010000000","101","100000010","10
0000001","010100000","011","0000
01","111","111","000000111","111

```

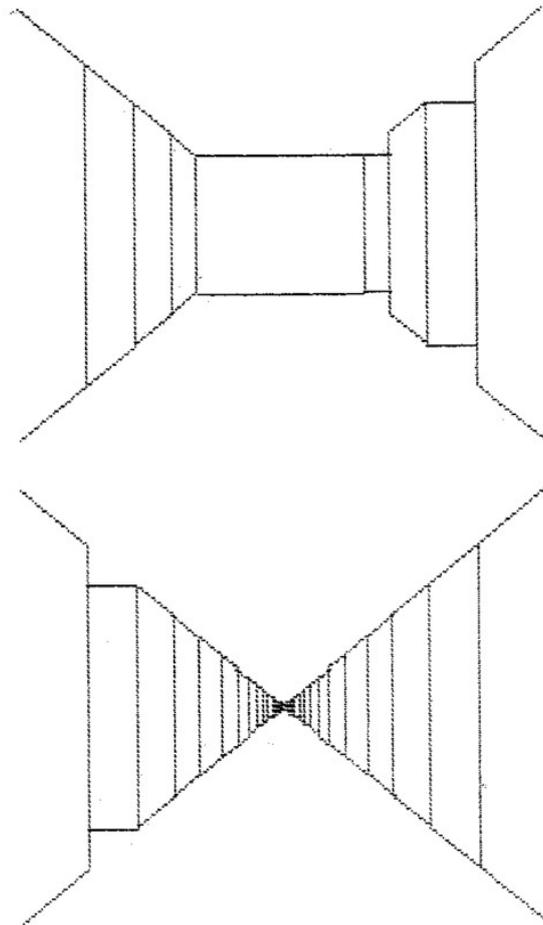
"100111", "000111", "000100111",
"111", "000000100", "001", "001", "1
00001", "010011", "001", "000010011
", "010001"
6050 DATA "010110000", "100000010
", "110000000", "011", "000000101",
"111", "000101", "000111", "101", "1
10101", "000000111", "101", "011",
000000110", "111", "000101", "001",
"11000000", "100000101", "111", "0
11", "000001", "100011", "001", "000
100011"
6070 DATA "100011", "010000001", "
011", "101", "101", "001", "111", "01
1", "010101", "011", "101", "101", "1
01", "000000110", "011", "000110000
", "000011", "110000000", "00000011
1", "100011", "000000011", "0000001
10", "110000000", "000110111", "000
110000"
6080 DATA "110111", "110011", "000
011", "011", "011", "011", "000011"
", "101", "101", "111", "101", "011", "0
00100000", "011", "101", "100000100
", "101", "011", "000010001", "011",
"101", "010001", "000111", "1101000
11", "111"
6090 DATA "000110100", "000000110
", "000000100", "000010011", "10000
0000", "010011", "000100000", "0100
00010", "100000101", "000010000", "
101", "000101", "00000010", "111",
"011", "001", "01", "101", "101", "0
10000000", "000001", "000010000", "
011", "001", "000011"
6100 DATA "003000004002", "000000
000001", "0000005000005", "02500000
0026", "000007000008", "0000040020
03", "000010000011", "000000000005
", "0000052000053", "000012000013",
"000000000007", "000000014015"
6110 DATA "000011000010", "018000
017016", "000013000012", "01500000
0014", "020000000021", "0000000000
19", "000017016018", "0230000000022
", "016018000017", "0000000021020",
"000000000024", "0000000022023"
6120 DATA "023000000027", "0020003
000004", "0000000025025", "000030002
9031", "0000050000060", "00000000320
33", "0000000027028", "035000000034
", "029031000030", "0330000000032",
"000036000037", "0390000038040"

6130 DATA "0000000034035", "0420000
043041", "0000053000052", "000003700
0036", "040039000038", "00004500440
46", "054000000055", "0710000072070
", "000047000048", "000043041042",
"049000000050", "044046000045"
6140 DATA "000000000051", "0000048
000047", "0000000050049", "000003804
0039", "011000010000", "0000000570
56", "041042000043", "000000055054
", "000000000053", "056000000057",
"0530061064062", "0310000030029"
6150 DATA "0000000067066", "0000000
000059", "000000000060", "000000000
0065", "0520063061064", "05400520630
61", "000000000060", "056000000067
", "051054062063", "0480000045044",
"000073000074", "1010000000102"
6160 DATA "000076075077", "0000072
070071", "121000122120", "07807900
0080", "000074000073", "0000000000
86", "081000000082", "075077000076
", "085084000083", "0000000078079",
"0000000082081", "000000000097"
6170 DATA "0860000087088", "0790000
080078", "000000000089", "084000000
3085", "088000000087", "0000000000
91", "000000000090", "0950000094093
", "000001000092", "000000000096",
"000000000100", "0930095000094"
6180 DATA "0000030000034", "000000
000099", "000097000098", "00000400
3095", "000100105104", "0700710000
72", "146144147145", "000000102101
", "000107000106", "1040000103105",
"000109000108", "000106000107"
6190 DATA "110000000111", "114112
115113", "000100000109", "153000000
0152", "0000000111110", "0001500000
51", "1160000117118", "000171000172
", "000000000119", "113114112115",
"118116000117", "0770000076075"
6200 DATA "000000126127", "000124
000123", "120121000122", "00000000
0125", "000123000124", "1301200001
29", "000122120121", "0000000135134
", "127000000126", "1310000000132",
"000000000133", "1280000129130"
6210 DATA "000000132131", "000129
130128", "0000000136137", "13900000
0138", "134000000135", "1370000000
36", "000141000140", "0000000138139

```

", "000143000142", "000140000141",
", 144147145146", "000173000174",
6220 DATA "135104000103", "000142
000143", "000149000148", "14514514
4147", "000151000150", "0001480001
49", "112115113114", "115113114112",
", "000154000155", "0000000150157",
", "000000152153", "0000000150158",
6230 DATA "000155000154", "157000
000155", "150000000151", "15300015
4152", "153000000159", "00000001511
50", "000130000131", "0000000155155",
", "153000000157", "152153000154",
", "155000000155", "000000159170",
6240 DATA "000172000171", "000000
167168", "170000000169", "00011711
8116", "000175000176", "1471451451
44", "173000179177", "000174000173",
", "000176000175", "000182000183",
", "000181000180", "177178000179",
6250 DATA "000164162163", "000184
000185", "000179177178", "18618700
0188", "000183000182", "1900000001
89", "000194000195", "000185000184",
", "187000188186", "000000191192",
", "000000000193", "000000189190",
6260 DATA "192000000191", "195000
000197", "000188186187", "00019900
0198", "000195000194", "0000001971
95", "000000000200", "000198000199

```



LABIRINTO A 3 DIMENSÕES — II

Quando tiver dominado o programa de Scott Vincent, pode progredir para o que apresentamos em seguida. Neste programa, escrito por Graham Charlton, você vê novamente o labirinto em três dimensões. No entanto, ao contrário do que acontece no anterior, o labirinto é agora diferente de cada vez que o programa é executado.

O labirinto é constituído por uma rede de 22 X 32, necessitando o computador de aproximadamente um minuto para o criar. Vejamos alguns exemplos de execução deste programa:


```

THEN PLOT 128-k,88+l: DRAW D**k,0
: PLOT 128-k,88-l: DRAW D**k,0: G
0 TO 03
3120 LET #J=#J+1
3130 THEN PLOT #J+1
L/4: DRAW D**k/4,-
/4: GO TO 03
3140 PLOT 128-k,88+l: DRAW K
/4,0: DRAW D**k/4,-
3150 THEN PLOT #J+1: DRAW D**k/4,-
L/4: DRAW D**k/4,-
/4: GO TO 03
3160 PLOT 128-k,88+l*3/4: DRAW -
k/4,0: DRAW D**k/4,0
3170 LET K=K*3/4
3180 LET L=L*3/4
3190 TO 03
3200 PLOT X+1 AND Y+1
3210 PLOT X-k,88+l: DRAW D**k,-
3220 PLOT X+1 AND Y=01 THEN STOP
3230 LET X=X+1 AND Y=01 THEN STOP
3240 PLOT X+1 AND Y=01 THEN STOP
3250 PLOT X+1 AND Y=01 THEN STOP
3260 PLOT X+1 AND Y=01 THEN STOP
3270 PLOT X+1 AND Y=01 THEN STOP
3280 PLOT X+1 AND Y=01 THEN STOP
3290 PLOT X+1 AND Y=01 THEN STOP
3300 PLOT X+1 AND Y=01 THEN STOP
3310 PLOT X+1 AND Y=01 THEN STOP
3320 PLOT X+1 AND Y=01 THEN STOP
3330 PLOT X+1 AND Y=01 THEN STOP
3340 PLOT X+1 AND Y=01 THEN STOP
3350 PLOT X+1 AND Y=01 THEN STOP
3360 PLOT X+1 AND Y=01 THEN STOP
3370 PLOT X+1 AND Y=01 THEN STOP
3380 PLOT X+1 AND Y=01 THEN STOP
3390 PLOT X+1 AND Y=01 THEN STOP
3400 PLOT X+1 AND Y=01 THEN STOP
3410 PLOT X+1 AND Y=01 THEN STOP
3420 PLOT X+1 AND Y=01 THEN STOP
3430 PLOT X+1 AND Y=01 THEN STOP
3440 PLOT X+1 AND Y=01 THEN STOP
3450 PLOT X+1 AND Y=01 THEN STOP
3460 PLOT X+1 AND Y=01 THEN STOP
3470 PLOT X+1 AND Y=01 THEN STOP
3480 PLOT X+1 AND Y=01 THEN STOP
3490 PLOT X+1 AND Y=01 THEN STOP
3500 PLOT X+1 AND Y=01 THEN STOP
3510 PLOT X+1 AND Y=01 THEN STOP
3520 PLOT X+1 AND Y=01 THEN STOP
3530 PLOT X+1 AND Y=01 THEN STOP
3540 PLOT X+1 AND Y=01 THEN STOP
3550 PLOT X+1 AND Y=01 THEN STOP
3560 PLOT X+1 AND Y=01 THEN STOP
3570 PLOT X+1 AND Y=01 THEN STOP
3580 PLOT X+1 AND Y=01 THEN STOP
3590 PLOT X+1 AND Y=01 THEN STOP
3600 PLOT X+1 AND Y=01 THEN STOP
3610 PLOT X+1 AND Y=01 THEN STOP
3620 PLOT X+1 AND Y=01 THEN STOP
3630 PLOT X+1 AND Y=01 THEN STOP
3640 PLOT X+1 AND Y=01 THEN STOP
3650 PLOT X+1 AND Y=01 THEN STOP
3660 PLOT X+1 AND Y=01 THEN STOP
3670 PLOT X+1 AND Y=01 THEN STOP
3680 PLOT X+1 AND Y=01 THEN STOP
3690 PLOT X+1 AND Y=01 THEN STOP
3700 PLOT X+1 AND Y=01 THEN STOP
3710 PLOT X+1 AND Y=01 THEN STOP
3720 PLOT X+1 AND Y=01 THEN STOP
3730 PLOT X+1 AND Y=01 THEN STOP
3740 PLOT X+1 AND Y=01 THEN STOP
3750 PLOT X+1 AND Y=01 THEN STOP
3760 PLOT X+1 AND Y=01 THEN STOP
3770 PLOT X+1 AND Y=01 THEN STOP
3780 PLOT X+1 AND Y=01 THEN STOP
3790 PLOT X+1 AND Y=01 THEN STOP
3800 PLOT X+1 AND Y=01 THEN STOP
3810 PLOT X+1 AND Y=01 THEN STOP
3820 PLOT X+1 AND Y=01 THEN STOP
3830 PLOT X+1 AND Y=01 THEN STOP
3840 PLOT X+1 AND Y=01 THEN STOP
3850 PLOT X+1 AND Y=01 THEN STOP
3860 PLOT X+1 AND Y=01 THEN STOP
3870 PLOT X+1 AND Y=01 THEN STOP
3880 PLOT X+1 AND Y=01 THEN STOP
3890 PLOT X+1 AND Y=01 THEN STOP
3900 PLOT X+1 AND Y=01 THEN STOP
3910 PLOT X+1 AND Y=01 THEN STOP
3920 PLOT X+1 AND Y=01 THEN STOP
3930 PLOT X+1 AND Y=01 THEN STOP
3940 PLOT X+1 AND Y=01 THEN STOP
3950 PLOT X+1 AND Y=01 THEN STOP
3960 PLOT X+1 AND Y=01 THEN STOP
3970 PLOT X+1 AND Y=01 THEN STOP
3980 PLOT X+1 AND Y=01 THEN STOP
3990 PLOT X+1 AND Y=01 THEN STOP
4000 PLOT X+1 AND Y=01 THEN STOP

```

JOGOS PARA DOIS JOGADORES

QUADRADOS

Este programa, escrito por Neil Pallinacci, inclui versões para um ou dois jogadores. Em ambas as versões o jogo é jogado num mapa de dez por dez.

Você deve colocar marcas no mapa, tendo em conta que não pode colocar nenhuma delas junto às do computador.

O jogo para um jogador é chamado «AREA», e nele o leitor joga contra o computador. Perde quem primeiro ficar impossibilitado de colocar uma pedra no tabuleiro. Será rejeitado qualquer movimento que envolva a colocação de uma peça adjacente a uma já colocada pela máquina.

O computador coloca as suas peças aleatoriamente, a não ser que tenha problemas em descobrir um movimento, caso em que procurará metodicamente determinar o seu movimento. Se não conseguir executá-lo, admitirá a derrota.

O jogo para dois jogadores é chamado «LINHAS» e ganha quem primeiro conseguir colocar cinco peças em linha, horizontal ou verticalmente. Os movimentos serão rejeitados se o quadrado sugerido já estiver preenchido. No entanto, se um jogador colocar a sua peça adjacente às do adversário, a peça passa a ser deste. É possível usar várias técnicas para ganhar (tal como acontece na versão anterior). Mas basta um pequeno erro para perder.

É essencial que o leitor coloque o computador no modo «maiúsculas» (CAPS LOCK) imediatamente depois de ter indicado a opção de jogo. Quando o programa começa, imprime uma página de título e pede que seja escolhida uma opção (A a D).

As duas primeiras opções, para um ou dois jogadores, são óbvias. A terceira opção, alterar cores, permite-lhes definir as cores de ambos os jogadores carregando na tecla apropriada, 1 a 7, quando lhe for pedido. As duas cores devem obviamente ser diferentes.

Se pedir informação o computador descreverá rapidamente o objectivo do jogo.

Quando começar a jogar, o computador far-lhe-á a seguinte pergunta:

NOME DO JOGADOR 1

Escreva o seu nome, ou carregue em ENTER no caso de o nome já ter sido indicado anteriormente. Proceda do mesmo modo para o segundo jogador, quando necessário.

O visor apresenta o tabuleiro enquanto o jogo progride, ao mesmo tempo que indica a cor correspondente a cada jogador, a respectiva pontuação e a quantidade de quadrados dessa cor que se encontram actualmente no tabuleiro. Na parte inferior direita da imagem é indicado a quem pertence o movimento seguinte. Por baixo é apresentado o último movimento indicado ao computador, a branco.

Escreva o seu movimento indicando primeiro um número e depois uma letra, por exemplo 7A, 1D, sempre por esta ordem. Se escrever AA terminará o jogo, voltando à página inicial para definição de opções.

Este programa encontra-se escrito inteiramente em Basic, sendo formado por vários módulos diferentes ligados entre si pela rotina principal. Esta ocupa as linhas 540 a 1000, encontrando-se os diversos módulos a seguir.

```

500 REM Linhas
510 REM *****
530 GO SUB 9000: GO SUB 8000
540 GO SUB 9500: BORDER 0: PAPE
R 0: CLS: INK 7
600 GO SUB 9200: PAPER 0: INK 7
700 GO SUB 8000: GO SUB 2000: R
EM Jogador UM
705 IF #f OR #f THEN GO TO 5000
707 GO SUB 8000
710 IF #p=1 THEN GO SUB 3000
720 IF #p=2 THEN GO SUB 4000
725 IF #f OR #f THEN GO TO 5000
740 GO TO 700
1000 STOP
2000 REM Jogador UM
2010 BEEP .3: -10: PRINT INK 8: AT
16, 22: "JOGADOR UM": AT 17, 22: #
2020 PRINT AT 18, 22: INK 8: "Move
":
2030 GO SUB 2600: PRINT AT 19, 23
: #
2031 IF #f THEN LET #f=1: RETURN

```

```

2032 IF b(x,y) <> 0 THEN GO TO 203
0
2034 IF #p=1 THEN GO TO 2100
2035 LET ll=(y-2)*2+1: LET cc=(x
-2)*2+1
2040 LET p1s=p1s+1: LET b(x,y)=1
: PRINT AT ll,cc: OVER 1: PAPER
c1: " ": AT ll+1,cc: " "
2050 IF b(x,y+1)=2 OR b(x,y-1)=2
OR b(x+1,y)=2 OR b(x-1,y)=2: THE
N LET b(x,y)=2: PRINT AT ll,cc
OVER 1: PAPER c2: " ": AT ll+1,cc
: " ": LET p2s=p2s+1: LET p1s=p1
s-1
2052 BEEP .2, 10
2055 IF #p=1 THEN LET #f=0: RETU
RN
2060 LET #f=0: GO SUB 7500
2070 RETURN
2100 IF b(x,y+1)=2 OR b(x,y-1)=2
OR b(x+1,y)=2 OR b(x-1,y)=2: THE
N GO TO 2030
2110 GO TO 2035
2250 REM mover
2500 INPUT LINE a#: IF LEN a# <> 2
THEN GO TO 2600
2605 IF a#="AA" THEN LET #f=1: R
ETURN
2610 IF CODE a#(1) < 48 OR CODE a#
(1) > 57 THEN GO TO 2600
2620 IF CODE a#(2) < 65 OR CODE a#
(2) > 74 THEN GO TO 2600
2630 LET x=CODE a#(1)-48
2640 LET y=CODE a#(2)-53
2650 RETURN
3000 REM spectrum
3010 PRINT AT 16, 22: INK 6: "JOG.
DOIS": AT 17, 22: 9#: AT 18, 22: "Mo
ve: "
3015 FOR #=1 TO 15
3020 LET x=INT (RAND*10)+2: LET y
=INT (RAND*10)+2
3030 IF b(x,y) <> 0 THEN GO TO 302
0
3040 IF b(x+1,y)=1 OR b(x-1,y)=1
OR b(x,y+1)=1 OR b(x,y-1)=1: THE
N NEXT #: GO SUB 3500: IF #f THE
N RETURN
3045 LET ll=(y-2)*2+1: LET cc=(x
-2)*2+1: PRINT AT ll,cc: OVER 1:
PAPER c2: " ": AT ll+1,cc: " "
3050 LET b(x,y)=2: LET p2s=p2s+1

```

```

: RETURN
3500 REM
3510 FOR x=2 TO 11: FOR y=2 TO 1
4: IF b(x,y)<>0 THEN GO TO 3540
3520 IF b(x+1,y)=1 OR b(x-1,y)=1
OR b(x,y+1)=1 OR b(x,y-1)=1 THE
N GO TO 3540
3530 RETURN
3540 NEXT y: NEXT x: LET wf=1: R
RETURN
3600 RETURN
3600 REM Jogador dois
4010 BEEP ,3,-10: PRINT INK 6; AT
15,22;"JOG. DOIS"; AT 17,22; q#
4020 PRINT AT 18,22; INK 6;"Move
r:"
4030 GO SUB 2600: PRINT AT 19,23
;a#
4031 IF a# THEN LET a# = 2: RETURN
4032 IF b(x,y)<>0 THEN GO TO 403
0
4035 LET l=(y-2)*2+1: LET cc=(x
-2)*2+1
4040 LET p2s=p2s+1: LET b(x,y)=2
: PRINT AT l,cc; OVER 1; PAPER
c2;"": AT l+1,cc;"
4050 IF b(x+1,y)=1 OR b(x-1,y)=1
OR b(x,y+1)=1 OR b(x,y-1)=1 THE
N LET b(x,y)=1: PRINT AT l,cc;
OVER 1; PAPER c1;"": AT l+1,cc
;"": LET p1s=p1s+1: LET p2s=p2
s-1
4055 BEEP ,2,20
4060 LET wf=0: GO SUB 7500
4070 RETURN
5000 REM jogo terminado
5010 BRIGHT 1: PAPER 1: FOR a=13
TO 21: PRINT AT a,22;"
": NEXT a
5020 PRINT AT 13,23;"TERMINADO"
5030 IF a# THEN PRINT AT 15,24;"
PARADO"; AT 16,23;"PELO JOG."; AT
17,27;a#
5040 IF wf THEN PRINT AT 15,23;"
O JOG. ";wf; AT 16,26;"GANHA"
5050 PRINT AT 20,23; INK 6;"PRIM
A VEM"; AT 21,26;"TECLA"
5060 BRIGHT 0: PAPER 0: GO SUB 6
5070 FOR k=-20 TO 40 STEP 2: BEE
P ,05,k: BEEP ,01,30-k: NEXT k
5070 PAUSE 0: GO TO 540

```

```

6000 REM pontuacoes
6010 PRINT AT 2,22; INK 7;"JOGAD
OR UM"; AT 3,22; p#; AT 4,23; INK 7
; PAPER c1;"": PAPER 0; INK 5;"
"; p1s;"": REM grafico E
6020 PRINT AT 9,22; INK 7;"JOG.
DOIS"; AT 10,22; q#; AT 11,23; INK
7; PAPER c2;"": PAPER 0; INK 5
;""; p2s;"": REM grafico E
6030 RETURN
7000 RETURN
7500 REM verificar vitoria
7510 FOR x=2 TO 11: FOR y=2 TO 1
1
7520 IF b(x,y)<>0 THEN LET a=b(x
,y): GO SUB 7800: IF wf=1 THEN R
ETURN
7530 NEXT y: NEXT x: RETURN
7800 LET x1=x: LET y1=y
7810 FOR z=1 TO 4: LET x1=x1+1:
IF b(x1,y1)<>a THEN GO TO 7830
7820 NEXT z: LET wf=a: RETURN
7830 LET x1=x: LET y1=y
7840 FOR z=1 TO 4: LET y1=y1+1:
IF b(x1,y1)<>a THEN GO TO 7860
7850 NEXT z: LET wf=a: RETURN
7860 RETURN
7900 RETURN
8000 REM variaveis
8010 DIM p$(10): DIM q$(10): LET
c1=4: LET c2=5
8020 DIM b(12,12)
8030 POKE 23609,100
8500 RETURN
8600 REM info
8610 BORDER 1: PAPER 1: INK 6: C
LS
8620 PRINT AT 1,13;"AREA"; AT 2,1
3;"-----"
8630 PRINT "TAB 4;"JOGO PARA UM
SO JOGADOR"""" VOCE DEVE GANHAR
A AO COMPUTADOR.""""COLOGUE QUAD
RADOS DE MODO A QUE""""NAO FIQUE
M ADJACENTES AOS DO""""COMPUTADO
R.""
8640 PRINT "" O COMPUTADOR JOGAR
A DO MESMO """"MODO E INDICARA
QUEM GANHA.""
8650 PRINT "" INK 7;" "" CARRE
QUE NUMA TECLA. ""
8660 PAUSE 0: BORDER 0: PAPER 0:
CLS : INK 0

```

```

8670 PRINT AT 1,12;"LINHAS";AT 2
8680 PRINT "TAB 4;"JOGO PARA DO
IS JOGADORES""VOCE DEVE GANH
AR AO Oponente""DISPONDO CINCO
QUADRADOS EM FILA""ANTES DELE
8690 PRINT "" AS LINHAS PODEM SE
R HORIZONTAIS""OU VERTICAIS.
8700 PRINT "" INK 2;"" CARRE
QUE NUMA TECLA.
8710 PAUSE 0: BORDER 4: PAPER 4:
CLS : INK 0
8715 PRINT ""TAB 5;"QUADRADOS c
ontinuacao";TAB 5;""
8720 PRINT "" SE COLOCAR UM SUA
DRADO JUNTO A""UM DOS DO SEU O
ponente, PASSARA""A SER DELE.
8730 PRINT "" INK 7;"" CAR
REGUE NUMA TECLA.
8740 PAUSE 0: GO TO 9500
9800 REM cd's
9810 FOR a=0 TO 6: POKE USR "a"+
a,1: POKE USR "b"+7-a,1: NEXT a
9820 POKE USR "a"+7,255: POKE US
R "b",255
9830 FOR a=0 TO 6: POKE USR "c"+
a,128: POKE USR "d"+7-a,128: NEX
T a
9840 POKE USR "c"+7,255: POKE US
R "d",255
9850 POKE USR "e",255: FOR a=1 T
O 6: POKE USR "a"+a,129: NEXT a:
POKE USR "e"+7,255
9860 RETURN
9865 REM image#
9870 PAPER 0: CLS : INK 7
9880 FOR a=1 TO 20 STEP 2: PRINT
AT a,1;"|";AT a+1,1;"|";
NEXT a: REM graficos 80 e 84
9890 PAPER 7: INK 2: FOR a=0 TO
9: PRINT AT 0,a#2+1;a;"|";AT 21,
a#2+1;a;"|";NEXT a
9900 FOR a=1 TO 10: LET a1=a#2
9910 PRINT AT a1,0;CHR$(a+64);A
T a1,21;CHR$(a+64);AT a1+1,0;"
";AT a1+1,21;"|";NEXT a
9920 PRINT AT 0,0;"|";AT 1,0;"|
";AT 1,21;"|";AT 0,21;"|";

```

```

9930 RETURN
9935 REM opcoes
9940 LET a1=0: LET a2=0: LET a3
=0: LET a4=0: DIM b(12,12)
9950 BORDER 2: PAPER 2: INK 7: C
LS
9960 PRINT AT 1,9;"**QUADRADOS**
";AT 3,0;"Escrito por Neil Pelli
";
9970 PRINT ""TAB 12;"OPCOES"
9980 PRINT ""TAB 5;"[A] Um Jogad
or"
9990 PRINT ""TAB 5;"[B] Dois Joga
dores"
10000 PRINT ""TAB 5;"[C] Mudar cor
es"
10010 PRINT ""TAB 5;"[D] Informaca
o"
10020 PRINT "" Escolha a opcao
carregando"" na tecla apr
opriada:
10030 IF INKEY#="a" THEN LET np=1
: GO TO 9700
10040 IF INKEY#="b" THEN LET np=2
: GO TO 9800
10050 IF INKEY#="d" THEN GO TO 96
00
10060 IF INKEY#<>"c" THEN GO TO 9
900
10070 BORDER 7: PAPER 7: CLS : IN
K 1: PRINT ""Cor do Jogador 1:
""Carregue na tecla correspond
ente a cor"
10080 LET a=CODE INKEY#
10090 IF a<49 OR a>55 THEN GO TO
9900
10100 LET c1=a-48: BEEP .5,c1
10110 PRINT ""Cor do Jogador 2:
""Carregue na tecla correspond
ente a cor"
10120 LET a=CODE INKEY#: IF a<49
OR a>55 THEN GO TO 9900
10130 LET c2=a-48: IF c2=c1 THEN
GO TO 9900
10140 BEEP .5,c2: GO TO 9500
9700 INPUT "Escreva o seu nome:
";LINE a$
9710 IF a$="" THEN GO TO 9730
9720 LET p$=a$
9730 LET q$="SPECTRUM": RETURN
9800 INPUT "Nome do Jogador 1:
";LINE a$

```

```

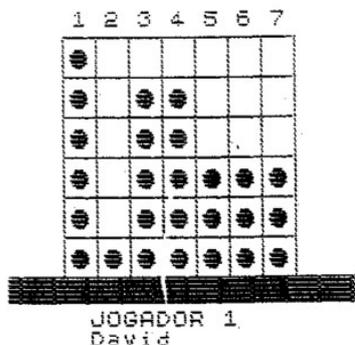
0010 IF a#="" THEN GO TO 9830
0020 LET p#:=a#
0030 INPUT "Nome do Jogador 2: "
; LINE a#
0040 IF a#="" THEN GO TO 9850
0050 LET a#:=a#
0060 RETURN

```

QUATRO EM LINHA

Da fértil imaginação de David Perry saiu esta versão de Quatro em Linha. O programa contém todas as instruções necessárias para ser executado.

Apresentamos um exemplo de execução seguido da listagem do programa.



```

2 FOR n=0 TO 7: READ a: POKE
USR "a"+n,a: NEXT n: GO SUB 5000
: DIM r(7,7)
5 DATA 50,126,255,255,255,255
,126,50
10 BORDER 0: PAPER 0: INK 7: B
RIGHT 1: CLS
20 PRINT AT 3,7;"1 2 3 4 5 6 7
"
40 FOR n=52 TO 166 STEP 16
50 PLOT n,40
60 DRAW 0,100
80 NEXT n

```

```

90 FOR n=44 TO 142 STEP 16
100 PLOT 52,n: DRAW 110,0
110 NEXT n
120 FOR n=34 TO 44: PLOT 25,n:
DRAW 100,0: NEXT n
130 LET play=1
140 BRIGHT 0
200 LET play=1: PRINT AT 10,0;
INK 7; PAPER 2; FLASH 1;"JOGADOR
1 ": PRINT AT 10,0;a#;"
210 INPUT "Mover Jogador 1? ";#
0
220 IF #>7 OR #<1 THEN GO TO
210
230 GO SUB 1000
250 LET play=2: PRINT AT 10,0;
INK 7; PAPER 2; FLASH 1;"JOGADOR
2 ": PRINT AT 10,0;b#;"
260 INPUT "Mover Jogador 2? ";#
0
270 IF #>7 OR #<1 THEN GO TO
260
280 GO SUB 1000
290 GO TO 200
1050 FOR n=6 TO 1 STEP -1: IF r(
n,#0)=1 THEN NEXT n: RETURN
1070 LET r(n,#0)=1
1080 IF play=1 THEN PRINT AT 3+(
n*2),(#0*2)+5; INK 2; BRIGHT 1;"
●": REM grafico A
1081 IF play=2 THEN PRINT AT 3+(
n*2),(#0*2)+5; INK 5; BRIGHT 1;"
●": REM grafico A
1090 RETURN
5000 BORDER 1: CLS : PLOT 0,0: D
RAW 0,175: DRAW 255,0: DRAW 0,-1
75: DRAW -255,0
5010 PRINT AT 2,6; BRIGHT 1;"QUA
TRO EM LINHA !!!"
5020 PRINT AT 3,6;

```

```

5030 PRINT AT 4,0;" Neste jogo d
eve tentar derrotar seu oponent
e dispondo quatro marcas numa l
inha, VERTICAL, HORIZONTAL ou
DIAGONAL... Por exemplo:
"
5050 PLOT 0,0: DRAW 0,175
5061 PLOT 255,0: DRAW 0,175
5060 PRINT AT 13,4;"●●●●": PRINT
AT 10,11;"●";AT 11,11;"●";AT 12
,11;"●";AT 10,11;"●"

```

```

5070 PRINT AT 10,15;"@";AT 11,15
5080 PRINT AT 12,17;"@";AT 13,18;"@";
5090 PRINT AT 10,20;"@";AT 11,25
5100 PRINT AT 12,24;"@";AT 13,23;"@";
5110 INPUT "Nome do Jogador 1? "
5120 IF LEN a#>10 THEN GO TO 509
5130 PRINT AT 15,1; INK 7; PAPER
2; FLASH 1;"JOGADOR 1: "; PRIN
T a#
5140 INPUT "Nome do Jogador 2? "
5150 IF LEN b#>10 THEN GO TO 509
5160 PRINT AT 16,1; INK 7; PAPER
2; FLASH 1;"JOGADOR 2: "; PRIN
T b#
5170 PRINT AT 15,22;" = "; INK 2
5180 PRINT AT 16,22;" = "; INK 5
5190 PRINT AT 20,7; FLASH 1; INK
2;"Carregue a# ENTER."; PAUSE 5
: RETURN

```

TANQUES

Trata-se de um jogo para duas pessoas, no qual é necessário disparar sobre a torre do tanque adversário. Vejamos a lista dos comandos a usar pelo primeiro jogador:

Esquerda — Z
Direita — X
Subir — 2ª linha/esquerda
Descer — 3ª linha/esquerda
Disparar — Linha superior/esquerda

E agora os comandos do segundo jogador:

Esquerda — M
Direita — Symbol Shift
Subir — 2ª linha/direita
Descer — 3ª linha/direita
Disparar — Linha superior/direita

Convirá notar que a versão aqui apresentada utiliza para a leitura dos «ports» correspondentes ao teclado os valores ade-

quados à versão 3 do Spectrum. Estes valores são diferentes no caso das versões 1 e 2 deste computador. Se o leitor possuir uma destas últimas versões do computador, deve alterar, nas linhas onde existem REM's seguidas de número (por exemplo, REM 255) o valor que se encontra à frente da instrução IN, substituindo-o pelo número indicado na REM.

Note que, ao jogar, os tanques desaparecem durante um momento quando se dispara, mas que mesmo assim podem ser atingidos ou fugir da bala. Não se pode passar sobre as minas «largadas» pelas balas. Cada jogador dispõe de três vidas, e o primeiro a morrer, como é óbvio, perde. Este programa foi escrito por Neal Cavalier-Smith.

```

10 REM Tanques
15 REM Neal Cavalier-Smith
20 LET x1=INT (RND*20): LET y1
=5: LET x2=INT (RND*20): LET y2=
25
30 LET y1b=1: LET x1b=0: LET y
2b=-1: LET x2b=0
40 LET cc=0: LET c1=0: LET (if
e2=3: LET life1=3
50 GO SUB 1000
60 GO SUB 100
70 GO SUB 50
100 REM movimento
101 PRINT AT x1,y1;" "
102 IF x1b=0 THEN PRINT AT x1,y
1+y1b;" "
103 IF y1b=0 THEN PRINT AT x1+x
1b,y1;" "
110 IF IN 65022<191 AND x1<20 T
HEN LET x1=x1+1: LET x1b=1: LET
y1b=0: REM 255
120 IF IN 64510<191 AND x1>0 TH
EN LET x1=x1-1: LET x1b=-1: LET
y1b=0: REM 255
130 IF IN 65278=189 AND x1>0 TH
EN LET y1=y1-1: LET y1b=-1: LET
x1b=0: REM 255
140 IF IN 65278=187 AND y1<30 T
HEN LET y1=y1+1: LET y1b=1: LET
x1b=0: REM 255
150 PRINT AT x2,y2;" "
152 IF x2b=0 THEN PRINT AT x2,y
2+y2b;" "
153 IF y2b=0 THEN PRINT AT x2+x
2b,y2;" "
160 IF IN 49150<191 AND x2<20 T
HEN LET x2=x2+1: LET x2b=1: LET
y2b=0: REM 255

```

```

170 IF IN 57342<191 AND x2>0 TH
EN LET x2=x2-1: LET
y2=y2-1: REM
180 IF IN 30756=187 AND y2>0 TH
EN LET y2=y2-1: LET
x2=x2-1: REM
190 IF IN 30766=189 AND y2<30 T
HEN LET y2=y2+1: LET
x2=x2+1: REM
200 REM definir tanque
210 IF IN 61430<191 THEN LET cc
=c: REM
220 IF cc>0 THEN GO SUB 300
230 IF SCREEN# (x2+x2b,y2+y2b)=
"." THEN GO TO 2000: PRINT AT x1
,y1:CHR# 144
240 PRINT AT x1,y1:CHR# 144
250 IF IN 63466<191 THEN LET c1
=c: REM
260 IF c1>0 THEN GO SUB 400
270 IF SCREEN# (x1+x1b,y1+y1b)=
"." THEN GO TO 2200: PRINT AT x2
,y2:CHR# 144
280 IF x1b=0 THEN PRINT AT x1,y
1+y1b:CHR# 146
290 IF y1b=0 THEN PRINT AT x1+x
1b,y1:CHR# 146
300 IF SCREEN# (x1+x1b,y1+y1b)=
"." THEN GO TO 2200: PRINT AT x1
,y1:CHR# 144
310 PRINT AT x2,y2:CHR# 144
320 IF x2b=0 THEN PRINT AT x2,y
2+y2b:CHR# 146
330 IF y2b=0 THEN PRINT AT x2+x
2b,y2:CHR# 146
340 RETURN
350 REM
360 IF cc=3 THEN LET x2s=x2+x2b
: LET y2s=y2+y2b
370 FOR c=0 TO 9
380 IF c=0 THEN GO TO 335
390 IF y2s<1 THEN LET y2b=1: LE
T x2b=INT (RAND*3)-1
400 IF y2s>29 THEN LET y2b=-1:
LET x2b=INT (RAND*3)-1
410 IF x2s>10 THEN LET x2b=-1:
LET y2b=INT (RAND*3)-1
420 IF x2s>16 THEN LET x2b=-1:
LET y2b=INT (RAND*3)-1
430 IF x2s<1 THEN LET x2b=1: LE
T y2b=INT (RAND*3)-1
440 IF x=x1 AND y=y1 OR x=x1+x1
b AND y=y1+y1b THEN GO SUB 2200

```

```

330 PRINT AT x,y: " "
335 PRINT AT x2s,y2s: "."
340 LET x=x2s: LET y=y2s
350 LET x2s=x2s+x2b
360 LET y2s=y2s+y2b
370 NEXT c
380 LET cc=cc-1
390 RETURN
400 REM
410 IF c1=3 THEN LET x1s=x1+x1b
: LET y1s=y1+y1b
420 FOR d=0 TO 9
430 IF d=0 THEN GO TO 435
440 IF y1s<1 THEN LET y1b=1: LE
T x1b=INT (RAND*3)-1
450 IF y1s>29 THEN LET y1b=-1:
LET x1b=INT (RAND*3)-1
460 IF x1s>10 THEN LET x1b=-1:
LET y1b=INT (RAND*3)-1
470 IF x1s>16 THEN LET x1b=-1:
LET y1b=INT (RAND*3)-1
480 IF x1s<1 THEN LET x1b=1: LE
T y1b=INT (RAND*3)-1
490 IF x=x2 AND y=y2 OR x=x2+x2
b AND y=y2+y2b THEN GO SUB 2000
500 PRINT AT x,y: " "
510 PRINT AT x1s,y1s: "."
520 LET x=x1s: LET y=y1s
530 LET x1s=x1s+x1b
540 LET y1s=y1s+y1b
550 NEXT d
560 LET c1=c1-1
570 RETURN
580 STOP
1000 REM definir graficos
1010 FOR z=144 TO 146
1020 FOR x=0 TO 7
1030 READ a: POKE USA (CHR# z)+x
,a
1040 NEXT x
1050 NEXT z
1060 DATA 60,126,255,255,255,255
,126,60,24,24,24,24,24,24,24,24,
0,0,0,255,255,0,0,0
1070 RETURN
2000 PRINT AT x2,y2: FLASH 1;"S"
:CHR# 147
2010 FOR z=5 TO 100: NEXT z
2020 PRINT AT x2,y2: FLASH 0:"
"
```

```

0001 FLASH 1;"T E R M I N A D O
0002 PRINT TAB 7;"Ganha o Jogador
0003 STOP
0004 GO TO 100
0005 PRINT AT x1,y1; FLASH 1;"G
0006
0007 FOR x=5 TO 100: NEXT x
0008 PRINT AT x1,y1; FLASH 0;"
0009
0010 LET life1=life1-1
0011 IF life1=0 THEN PRINT INVER
0012 FLASH 1;"T E R M I N A D O
0013 PRINT TAB 7;"Ganha o Jogador
0014 STOP
0015 GO TO 100

```

AUXILIARES EM CÓDIGO-MÁQUINA

MONITOR DE MEMÓRIA

O programa que se segue, escrito por David Perry, permite ao leitor examinar o conteúdo da ROM e da RAM. Possui boas rotinas de conversão de hexadecimal para decimal e vice-versa, e dá ao utilizador a possibilidade de deslocar valores de um endereço para outro, de gravar em fita e carregar sectores de código.

ESTE PROGRAMA REVELARÁ O CONTEÚDO DA ROM E RAM SOB A FORMA DE ENDEREÇO DECIMAL E HEXADECIMAL, VALOR NESSE ENDEREÇO EM DECIMAL E HEXADECIMAL, E CARACTER CORRESPONDENTE A ESTE VALOR. O PROGRAMA PERMITIRÁ TAMBÉM DESLOCAR LIVREMENTE CÓDIGOS EM MEMÓRIA, CONVERTER HEXADECIMAL PARA DECIMAL E VICE-VERSA, ALEM DE EXECUTAR QUALQUER ROTINA EM CÓDIGO.

Ender+hex:	dec:	peekhex:	chr#:
0 0000	243	00F3	NEXT
1 0001	175	00AF	CODE
2 0002	17	0011	?
3 0003	255	00FF	COPY
4 0004	255	00FF	COPY
5 0005	105	00C3	NOT
6 0006	255	00C5	THEN
7 0007	17	0011	?
8 0008	42	002A	*
9 0009	93	005D	J
10 000A	92	005C	^
11 000B	34	0022	^
12 000C	95	005F	^
13 000D	92	005C	^

```

14 000E 04 0018 3
16 000F 07 0043 C
18 0010 10 0070 NOT
17 0011 14 0072 PAUSE
D: Novo ender. H: hex-dec
E: Exec. código 0: dec-hex
S: Save Bytes L: Load By.

```

M-Desl. Memor. C-continuar

```

1 REM
2 REM
3 REM
4 REM
5 REM
6 REM
7 REM
8 REM
9 REM

```

```

M O N I T O R
© DAVID PERRY
©1983

```

```

10 BRIGHT 0: GO SUB 720: INPUT
11 "Numero inicial? ";add
12 GO TO 330
13 REM hex-dec
14 PRINT AT 20,0;"
"
60 PRINT AT 20,0;
70 LET a=0: LET e=0
80 INPUT "Caracteres hex.? ";a
# LET c#="0"
90 IF a=0
100 LET a=0
110 IF LEN a#>3 THEN GO TO 120
120 LET a#-0#+a#: GO TO 100
130 LET #=4096: LET a#e
140 FOR r=1 TO 4
150 IF FN p(a#(r))<e OR FN p(a#
(r))>16 THEN GO TO 100
160 LET a#a+#FN p(a#(r)): LET
e=e/16
170 NEXT r
180 PRINT " Hexadecimal: ";a
#;" "
190 PRINT " Decimal: ";a;"
"
200 PAUSE 0: RETURN
210 REM dec-hex
220 PRINT AT 20,0;"
230 INPUT "Numero decimal? ";a
240 IF a>65535 OR a<0 THEN GO T
O 230

```

```

250 PRINT AT 20,5;" Decimal: ";a
#;" "; PRINT AT 21,5;" Hexadecim
al: ";
260 LET u=4096: LET y=a
270 LET t=INT (y/u)
280 PRINT PAPER 7; BRIGHT 1;CHR
# (148+t+7*(t>9));
290 LET y=y-u*t
300 LET u=u/16
310 IF u=1 THEN GO TO 270
320 RETURN
330 REM
340 PRINT AT 0,0; PAPER 5; INK
0;"Ender+hex: dec: peekhex: chr#
";
350 LET go=add: LET to=add+17:
FOR l=go TO to
360 LET dec=PEEK add: LET c#-CH
R# (dec AND dec>32)
370 PRINT TAB (1 AND add<10)+(1
AND add<100)+(1 AND add<1000)+(
1 AND add<10000); INK 7; PAPER 4
;add; PAPER 7;"
380 LET a=add: GO SUB 260
390 PRINT " "; PAPER 5;dec; PA
PER 7;TAB 17;; LET a=dec: GO SUB
260: PRINT PAPER 7;TAB 24; PAPE
R 6;c#
400 LET add=add+1: IF add>65535
THEN LET add=0
410 NEXT l
420 PRINT AT 10,2; PAPER 2; INK
7;" A: Novo ender. H: hex-dec "
430 PRINT AT 20,2; PAPER 2; INK
6;" E: Exec. código 0: dec-hex "
440 PRINT AT 21,2; INK 7; PAPER
2;" S: Save Bytes L: Load By. "
450 PRINT #1;AT 0,0;" M-Desl.
Memor. C-continuar"
460 LET i#-INKEY#: IF i#="" THEN
N GO TO 450
470 IF i#="A" OR i#="a" THEN RU
N
480 IF i#="H" OR i#="h" THEN GO
SUB 49: GO TO 410
490 IF i#="S" OR i#="s" THEN GO
SUB 920
500 IF i#="D" OR i#="d" THEN GO
SUB 299: PAUSE 0: GO TO 410
510 IF i#="E" OR i#="e" THEN GO
TO 570
520 IF i#="L" OR i#="l" THEN GO
SUB 880: GO TO 410

```

```

520 IF i#="C" OR i#="c" THEN CL
530 GO TO 330
530 IF i#="E" OR i#="H" THEN GO
TO 600
540 IF i#<>"S" OR i#<>"s" OR i#
<>"L" OR i#<>"l" OR i#<>"H" OR i#
<>"h" OR i#<>"d" OR i#<>"D" OR
i#<>"e" OR i#<>"E" OR i#<>"C" OR
i#<>"c" OR i#<>"M" OR i#<>"m" T
HEN GO TO 450
550 GO TO 410
560 REM executar rotina
570 INPUT "Endereco de execucao
? ";ex
580 RANDOMIZE USA ex: PAUSE 0:
RUN
590 REM deslocar memoria
600 INPUT "Deslocar do endereco
? ";#f
610 IF #f>65535 OR #f<0 THEN GO
TO 600
620 INPUT "Deslocar para endere
co? ";#t
630 IF #t>65535 OR #t<0 THEN GO
TO 620
640 INPUT "No. de bytes a deslo
car? ";n
650 IF #t+n>65535 THEN PRINT #1
: "Demasiados bytes. Tente de nov
o " : PAUSE 0: GO TO 600
660 LET add=#t
670 FOR p=1 TO n
680 POKE #t,PEEK #f
690 LET #t=#t+1: LET #f=#f+1: N
EXT p
700 CLS
710 GO TO 330
715 REM instrucoes
720 DEF FN p(a#)=(CODE a#-48-7*
(a#>"9")-32*(a#>"Z"))
730 BORDER 7: PAPER 7: INK 0: C
LS: PRINT AT 1,7;"
"
740 PRINT AT 2,7;" M O N I T O
R.
"
750 PRINT AT 3,7;" © DAVID PE
RAY
"
760 PRINT AT 4,7;" 1983
"
770 PRINT AT 5,7;"
"
780 PRINT AT 8,0;" ESTE PROGRAM

```

```

A REVELARA O CONTEUDO DA ROM
E RAM SOB A FORMA"
790 PRINT "DE ENDERECO DECIMAL
E HEXADECIMAL, VALOR NESSE END
ERECO EM DE-"
800 PRINT "CIMAL E HEXADECIMAL,
E CHARACTER"
810 PRINT "CORRESPONDENTE A EST
E VALOR."
820 PRINT "O PROGRAMA PERMITIR
A TAMBEM DESLOCAR LIVREMENTE
CODIGOS EM MEMORIA, CONVERTER H
EXADECIMAL"
830 PRINT "PARA DECIMAL E VICE-
VERSA, ALEM DE EXECUTAR QUALQUER
ROTINA EM CODIGO."
840 REM Carregar programa
850 INPUT "Quer carregar um pro
grama (s/n)? ";t#
860 IF t#="S" OR t#="s" THEN GO
TO 880
870 CLS: RETURN
880 INPUT "Endereco de carga? "
:t
890 IF t>65535 OR t<0 THEN GO T
O 880
900 PRINT FLASH 1; PAPER 2; INK
7; AT 21,0;" Ponha o gravador em
movimento."
910 LOAD ""CODE t: CLS: RETURN
920 REM gravar
930 INPUT "Endereco do codigo a
gravar? ";sf
940 IF sf>65535 OR sf<0 THEN GO
TO 930
950 INPUT "Numero de bytes? ";n
960 IF n>65535 OR n<1 THEN GO T
O 950
970 INPUT "Nome? ";s#
980 IF s#="" OR LEN s#>10 THEN
GO TO 970
990 SAVE s#CODE sf,n
1000 RETURN

```

Apresentamos agora uma versão do programa anterior escrita inteiramente em Basic. Esta versão foi igualmente concebida para o computador de 48 K, e escrita pelo mesmo autor — David Perry.

GUARDAR «SCREENS» EM CÓDIGO-MÁQUINA

Esta rotina, escrita por David Perry, foi concebida para o computador de 48 K. Permite ao utilizador guardar em memória e imprimir instantaneamente no visor qualquer imagem. Pode ser usada pelo leitor em qualquer programa de sua autoria.

```

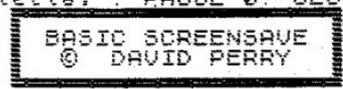
10 BORDER 0: PAPER 0: INK 7: C
L5
15 RESTORE
16 FOR n=40000 TO 40028
20 READ a: POKE n,a
30 NEXT n
40 DATA 33,0,64,17,48,117,1,0,
27,237,176,201,33,48,117,17,0,64,
1,0,27,237,176,201,195,144,234,
0,0
50 PRINT AT 0,0;"Carregue imag
em do gravador e aguarda-la-ei
imediatamente!": PRINT "podera'
depois recupera-la quan-do quise
r!"
50 PRINT : PRINT "Para guardar
a imagem escreva:"
70 PRINT "      RANDOMIZE USA 40
000"
80 PRINT : PRINT "Para imprimi
r a imagem escreva:"
90 PRINT "      RANDOMIZE USA 40
012"
100 PRINT : PRINT : PRINT "Veja
Um exemplo."
120 PRINT FLASH 1;" Ponha o gra
vador em movimento. "
130 LOAD ""SCREEN$: RANDOMIZE
USA 40000
140 PAUSE 0: CLS : PRINT AT 0,0
;" Carregue numa tecla para impr
i-mir a imagem no visor.": PAUSE
0: RANDOMIZE USA 40012: GO TO 1
40

```

```

1 CLS : PRINT "Este programa
guarda a imagem emmemoria e em s
eguida recupera-a.Como exemplo,
vou listar este programa, guar
da-lo e imprimi-lo de novo no vis
or."
2 PRINT : PRINT "O programa e
sta" completamente escrito em B
asic e pode ser usa-do para muit
os fins."
3 PRINT : PRINT TAB 5;"Carreg
ue numa tecla.": PAUSE 0: CLS
10 REM
20 REM
30 REM
40 REM
45 CLS : LIST 50
50 LET a=30000: REM endereco o
nde se guarda a imagem
60 FOR x=0 TO 21
70 FOR y=0 TO 31
80 LET a$=SCREEN$(x,y)
85 PRINT AT x,y;" "
90 POKE a,CODE a$
100 LET a=a+1
110 NEXT y
120 NEXT x
130 REM
140 REM ***imprimir image****
150 REM
160 LET a=30000: REM endereco o
nde esta a imagem
170 FOR x=0 TO 21
180 FOR y=0 TO 31
190 LET b=PEEK a
200 PRINT AT x,y;CHR# b
210 LET a=a+1
220 NEXT y
230 NEXT x

```




```

10,1,0
1140 DATA 4,0,0,0,0,PI/3,-1,4,PI
1150 DATA 0,0,0,0,0,PI/3,0,0,PI,-1
1160 DATA 0,0,0,0,0,PI/3,0,0,PI,-1
1170 DATA 1,0,0,PI/4,0,0,0,0
1180 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1190 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1200 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1210 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1220 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1230 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1240 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1250 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1260 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1270 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1280 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1290 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1300 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1310 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1320 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1330 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1340 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1350 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1360 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1370 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1380 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1390 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1400 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1410 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1420 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1430 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1440 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1450 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1460 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1470 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1480 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1490 DATA 1,0,0,PI/6,0,0,PI/4,-0,0
1500 DATA 1,0,0,PI/6,0,0,PI/4,-0,0

```

COPIADOR

Este programa de David Perry permitir-lhe-á examinar os cabeçalhos de qualquer programa, e indicará o modo como é possível fazer a sua cópia. No caso dos programas em código-máquina, guarda-os em memória e indica a instrução a utilizar para a sua gravação.

Se o programa em código tiver um carregador em Basic, deve primeiramente fazer MERGE deste (MERGE""), e depois SAVE "nome" LINE "linha por onde inicia a execução". Para usar o programa, siga as indicações nele contidas.

```

5 GO SUB 31
6 CLEAR 32747
7 FOR I=32748 TO 32761
8 READ X: POKE I,X
9 DATA 55,52,0,221,33,0,125,1
7,20,0,205,86,0,201
10 NEXT I
11 PRINT AT 5,2; FLASH 1;"Esto
u a carregar o cabeçalho"; FLASH
0; AT 7,10;"Um momento."
12 RANDOMIZE USA 32748
13 LET U=1: GO SUB 19: LET U=0
14 CLS : PRINT AT 0,0;"Dados s
obre programa"
15 IF PEEK 32000=0 THEN GO TO
64

```

```

16 IF PEEK 32000=1 THEN GO TO
59
17 IF PEEK 32000=2 THEN GO TO
73
18 IF PEEK 32000<>3 THEN RUN
19 LET Z#="": FOR B=32001 TO 3
2010
21 LET Z# = Z# + CHR$(PEEK B)
22 NEXT B
23 LET O# = Z#
24 LET A = PEEK 32011: LET B = PEE
K 32012
25 LET A = A + (B * 256)
26 LET A = PEEK 32013: LET B = PEE
K 32014
27 LET A = A + (B * 256)
28 LET A = PEEK 32015: LET B = PEE
K 32016
29 LET A = A + (B * 256)
30 IF U=1 THEN RETURN
31 GO TO 47
32 BRIGHT 1: BORDER 0: PAPER 0
: INK 7: CLS : PRINT TAB 12;"COP
IADOR"
33 PRINT TAB 12;"-----"
34 PRINT
35 PRINT "Este programa copia
programas em
36 PRINT "código-máquina autom
aticamente."
37 PRINT
38 PRINT "Se existir um carreg
ador Basic, deve copia-lo primei
ro escreven-
39 PRINT "do MERGE """" e depo
is"
40 PRINT " SAVE ""nome"" LIN
E ""linha que","carrega o código
-máquina""
41 PRINT : PRINT "Este program
a le o cabeçalho,"
42 PRINT "carrega o programa e
em seguida"
43 PRINT "indica o modo de o s
cavar."
44 PRINT : PRINT "1....Rebobin
e a cassette para o início
da secção em código"
45 PRINT "2....Verifique ligac
ões"
46 PRINT "3....Carregue em qua
lquer tecla"

```

```

46 PAUSE 0: CLS: RETURN
47 CLS: PRINT AT 1,12;"COPIAD
OR"
48 PRINT AT 2,12;"====="
49 PRINT AT 3,0;"Para gravar o
programa deve antes do mais
gravar o seu"
50 PRINT "carregador Basic, re
bobinando depois a cassette pa
ra o inicio do programa em codig
o-maquina."
51 PRINT "Em seguida 'NEW' est
e programa e escreva:"
52 PRINT: PRINT "LOAD """;z
#;"CODE";#;"",n
53 PRINT "Verifique as ligaco
es, ponha o gravador em moviment
o."
54 PRINT: PRINT "Quando tiver
carregado escreva:"
55 PRINT: PRINT "SAVE """;z
#;"CODE";#;"",n
56 PRINT "Coloque cassette vir
gem, verifique os cabos, e po
nha o gravador a gravar."
57 INPUT "Copia na impressora
(s/n)?";a#
58 IF a#="s" THEN COPY
59 IF a#="n" THEN PRINT AT 21,
0; BRIGHT 0;"Talvez convenha ent
ao escrever."
60 INPUT "Existem mais seccoes
em codigo?";a#
61 IF a#="s" THEN RUN
62 PRINT AT 21,1; FLASH 1; PAP
ER 2;"Carregue numa tecla para
new"
63 PAUSE 1: PAUSE 0: NEW
64 CLS: PRINT "Acaba de carre
gar um cabeçalho Basic."
65 PRINT "Titulo:"
66 PRINT z#;"executa da linha
",#
67 PRINT "Comprimento:";o
68 PRINT: PRINT "Carregue num
a tecla para execu- tar novament
e...": PAUSE 0: RUN
69 CLS: PRINT "Acaba de carre
gar o cabeçalho deum ARRAY numer
ico."
70 PRINT "Titulo:"
71 PRINT z#

```

```

72 PRINT: PRINT "Carregue num
a tecla para execu- tar novament
e...": PAUSE 0: RUN
73 CLS: PRINT "Acaba de carre
gar o cabeçalho deum ARRAY alfan
umerico."
74 PRINT "Titulo:"
75 PRINT z#
76 PRINT: PRINT "Carregue num
a tecla para execu- tar novament
e...": PAUSE 0: RUN

```

COPIADOR =====

Este programa copia programas em código-maquina automaticamente.

Se existir um carregador Basic, deve copia-lo primeiro escreven- do MERGE "" e depois "SAVE "nome" LINE "linha que carrega o código-maquina"

Este programa te o cabeçalho, carrega o programa e em seguida indica o modo de o gravar.

- 1...Rebobine a cassette para o inicio da seccao em codigo
- 2...Verifique ligacoes
- 3...Carregue em qualquer tecla

BIORRÍTMOS

Poderá agora representar graficamente os seus momentos altos e baixos utilizando este programa.

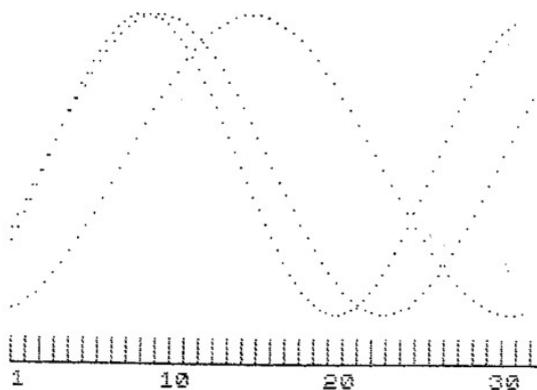
Existem três ciclos, que se iniciam quando você nasce e se mantêm durante toda a sua vida, que muitos pensam ter um efeito sobre esta. Os três ciclos são os seguintes:

Físico — Um ciclo de 23 dias que controla a sua força e resistência físicas.

Emocional — Um ciclo de 28 dias, que governa as suas sensações de optimismo e pessimismo.

Mental — Ciclo de 33 dias que comanda a sua capacidade de raciocínio lógico.

Físico Mental Emocional



```

1 RESTORE
2 PRINT AT 0,0;"          BI
ORRITMO.
10 INPUT "Indique data de nasc
imento:";"Dia ";a;" Mes (no.) "
;b;" Ano ";c
20 INPUT "Indique data actual:
";"Mes ";d;" Ano ";e
25 CLS
30 LET t=INT (((e-c)*365.25)+(
(d-b)*30.35)-a)
300 FOR r=0 TO 255
810 PLOT r,10
815 IF r=INT (r/b)*b THEN FOR u
=10 TO 20: PLOT r,u: NEXT u
820 NEXT r
830 PRINT AT 21,0;"1          10
20          30"
840 PRINT AT 0,0; INK 1;"Físico
"; INK 2;"Mental "; INK 4;"
Emocional"
900 FOR r=1 TO 3
905 READ U
910 LET l=2*PI*(t-(INT (t/u)*u
)/u
920 LET k=2*PI*(33-u)*.03
1000 FOR a=l TO k+(2*PI) STEP
.1
1010 PLOT INK ((1 AND u=23)+(2 A
ND u=28)+(4 AND u=33));(a-l)*(35
-28+u),90+51N a#60
1020 NEXT a
1030 NEXT r
1040 DATA 23,28,33
1050 INPUT "Novo biorritmo?"a#
: IF a#(1)="s" THEN GO TO 1

```

Este programa de Graham Charlton preenche uma forma qualquer com uma determinada cor. As linhas 10 a 60 desenham a forma, e a subrotina que se inicia na linha 9000 preenche-a. Tome as cadeias A\$ e B\$ tão compridas quanto possível no seu computador; note que ocupam bastante memória.

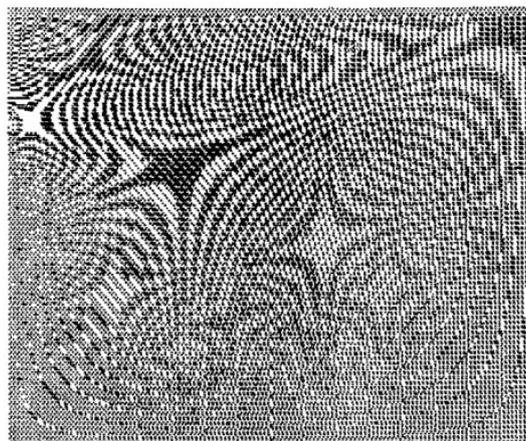
Se quiser preencher outra parte do visor, altere as linhas 9030 e 9040, onde x e y se referem às coordenadas em pixels do ponto por onde se inicia o enchimento das formas.

```

10 CIRCLE 120,80,15
20 PLOT 100,50
30 DRAW 40,0
40 DRAW 0,40
50 DRAW -40,0
60 DRAW 0,-40
70 GO SUB 9000
100 STOP
9000 DIM a$(90000)
9010 DIM b$(90000)
9020 LET z=1
9030 LET x=105
9040 LET y=80
9050 PLOT x,y
9060 IF POINT (x+1,y)+POINT (x-1
,y)+POINT (x,y+1)+POINT (x,y-1)<
>4 THEN GO TO 9120
9070 LET z=z-1
9080 IF z=0 THEN RETURN
9090 LET x=CODE a$(z)
9100 LET y=CODE b$(z)
9110 GO TO 9050
9120 LET a$(z)=CHR# x
9130 LET b$(z)=CHR# y
9140 LET z=z+1
9150 FOR c=-1 TO 1
9160 FOR d=-(c=0) TO (c=0) STEP
2
9170 IF POINT (x+c,y+d)=0 THEN G
O TO 9200
9180 NEXT d
9190 NEXT c
9200 LET x=x+c
9210 LET y=y+d
9220 GO TO 9050

```

O programa que se segue, de Damian Steele, cria algumas imagens de bastante impacto, que demonstram bem (se ainda necessitar de ser convencido) a eficácia do tratamento gráfico do seu computador.



```
10 PAPER 5: BORDER 1: INK 1: L
```

```

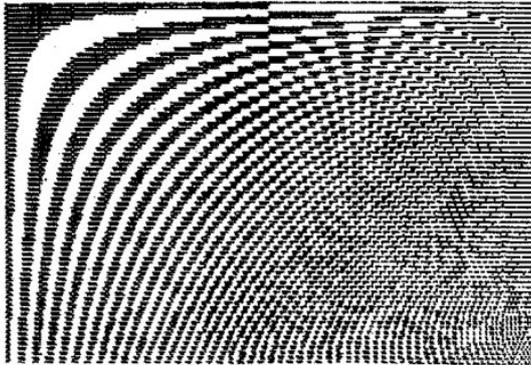
ET d=1: LET a=1: LET f=0: LET x
0: LET y=175: LET s=a: LET b=0:
LET c=0: OVER 1
20 CLS: PRINT #0;AT 0,1;"Carr
egue numa tecla se estiver aborr
ecido..."
100 FOR a=x TO y STEP s
110 IF a>=128 THEN GO TO 400
120 IF a>=87 THEN GO SUB 200
130 PLOT 0,a: DRAW (y*(1.457))-b,
a-c
135 IF INKEY$(0) THEN GO TO 10
00
140 NEXT a
200 LET b=b+5.1: LET c=c+2
300 RETURN

```

```

400 LET s=2
450 GO SUB 700
500 OVER 1
510 FOR a=1 TO 255 STEP s
515 IF INKEY#<>"" THEN GO TO 10
520 PLOT a,0: DRAW 0,175
530 NEXT a
535 IF b=1 THEN GO TO 600
537 GO SUB 700
540 FOR a=1 TO 175 STEP s
545 IF INKEY#<>"" THEN GO TO 10
550 PLOT 0,a: DRAW 255,0
560 NEXT a
565 GO SUB 700
570 IF b<>1 THEN LET b=1: GO TO
450
600 IF e=1 THEN LET e=0: GO TO
537
620 LET f=f+1: IF f<=1 THEN LET
s=1.4: LET b=0: GO TO 410
650 GO SUB 700: GO TO 1000
700 FOR a=1 TO 10: BEEP .007, (R
ND*30)+35: NEXT a
800 RETURN
1000 PAUSE 0: CLS
1001 LOAD ""
9500 STOP
9990 PRINT FLASH 1; AT 10,9; "PARE
O GRAVADOR": PRINT AT 21,6; "CAR
REGUE NUMA TECLA.": PAUSE 0: GO
TO 1

```



Este programa, bastante simples, de David Perry, imprime todas as cores produzidas pelo seu Spectrum, permitindo-lhe por exemplo sintonizar o receptor de televisão do melhor modo possível.

```

10 BORDER 2: PAPER 0: BRIGHT 1
INVERSE 0: OVER 0: FLASH 0: CL
20 LET i=-1: FOR n=0 TO 31 STE
4
30 LET i=i+1
40 FOR y=0 TO 3
50 FOR x=0 TO 21
60 PRINT AT x,n+y: INK i; "■"
70 NEXT x: NEXT y: NEXT n
80 INK 7

```

DEMONSTRAÇÃO DE PORTAS LÓGICAS

Este programa, escrito por Neville Predebon, mostra-nos o resultado do uso de portas lógicas AND, NAND, OR e NOR num circuito. Para usar o programa, responda às perguntas que faz.

```

1 REM Portas
5 REM © Predebon 1983
7 BORDER 6
10 PRINT INK 1; AT 2,13; "PORTAS
"; INK 0; AT 10,2; "Ilustracao de
portas logicas"
12 INK 0
15 PRINT AT 10,6; "Carregue num
a tecla.": PAUSE 0: CLS
20 PRINT AT 5,1; "Indique a por
ta.": PRINT
25 PRINT TAB 5; "AND = 1"
30 PRINT TAB 5; "NAND = 2"
35 PRINT TAB 5; "OR = 3"
40 PRINT TAB 5; "NOR = 4"
42 IF INKEY#<>"" THEN GO TO 42
43 IF INKEY#="" THEN GO TO 43
45 LET z#=INKEY#: IF z#<"1" OR
z#>"4" THEN GO TO 42
46 CLS

```

```

47 GO SUB 115
50 PRINT AT 3,6;"SAIDA 0000"
55 PRINT TAB 2;"Computacao A - 0011"
60 PRINT TAB 2;"Computacao B - 0101"
65 IF Z#="2" THEN GO TO 85
70 IF Z#="3" THEN GO TO 95
75 IF Z#="4" THEN GO TO 95
80 PRINT TAB 6;"Saída - 0000"
85 GO TO 100
85 PRINT TAB 6;"Saída - 0100"
90 GO TO 100
90 PRINT TAB 6;"Saída - 0100"
95 GO TO 100
95 PRINT TAB 6;"Saída - 1000"
100 GO TO 100
100 INPUT INK 1;"Nova execucao?"
105 IF a#="s" OR a#="S" THEN GO TO 15
110 PRINT : PRINT INK 2;TAB 9;"Terminado":STOP
115 PRINT AT 3,1;"PORTA"
116 IF z#="1" THEN PRINT AT 3,6;" AND "
120 IF z#="2" THEN PRINT AT 3,6;" NAND "
125 IF z#="3" THEN PRINT AT 3,6;" OR "
130 IF z#="4" THEN PRINT AT 3,6;" NOR "
135 RETURN

```

SONS

Cada rotina deste pequeno programa, chamada por uma das instruções RANDOMIZE que se encontram no final, produz um som diferente. Os cinco sons aqui apresentados poderão ser usados por si em qualquer programa de sua autoria.

```

10 CLEAR 32399
20 FOR a=32400 TO 32549
30 READ n: POKE a,n
40 NEXT a

```

```

50 DATA 6,3,197,33,15,0,17,40,
0,200,205,101,3,225,17,10,0,167,2
07,00,100,204,200,32,207,193,16,
200,201,0
60 DATA 6,5,197,33,0,3,17,1,0
07,00,200,101,3,225,17,10,0,167,2
0,0,0,204,193,16,200,201,0,0
70 DATA 6,5,197,33,15,0,17,40,
00,200,205,101,3,225,17,10,0,167,
03,00,100,204,200,32,207,193,16
,00,0,201,0
80 DATA 6,0,197,33,0,6,17,5,0
00,0,101,3,225,17,0,0,167,207
00,0,204,193,16,200,201,0,0,0
90 DATA 6,50,197,33,0,1,17,1,0
200,205,101,3,225,17,10,0,167,2
07,00,0,204,193,16,200,201,0,0,0
0,0
100 RANDOMIZE USR 32400
110 RANDOMIZE USR 32400
120 RANDOMIZE USR 32400
130 RANDOMIZE USR 32400
140 RANDOMIZE USR 32500
150 GO TO 100

```

ARCO-ÍRIS

Este programa permitir-lhe-á verificar como o seu computador pode misturar as suas cores básicas de modo a produzir 128 tonalidades diferentes.

São usadas para este efeito três caracteres gráficos. Um deles, com um padrão em xadrez, é muito usado noutros programas, mas tende a produzir o tipo de distorção conhecido pelo nome de «moiré». Os padrões constituídos por linhas rectas melhoram a estabilidade de cor, eliminando o movimento aparente da imagem.

SPECTRUM DE 128 CORES!

Este programa demonstra como é possível obter 128 tonalidades diferentes a partir das oito cores do ZX Spectrum.

Alguns programas que recorrem a este efeito utilizam um caracter grafico com um padrao em xadrez para a mistura das cores. Tambem o apresentamos aqui, mas em geral as series de linhas do mel- hor resultado, reduzindo o apa- rente movimento dos pontos im- pressos.

Carregue em qualquer tecla.



1. Linhas horizontais
2. Xadrez
3. Linhas quebradas

Carregue numa das opcoes 1, 2, 3

A combinacao de cores de cada tom pode ser determinada atraves dos eixos da tabela. Convem notar que as cores de 'tinta' e de 'papel' nao podem ser trocadas, isto e, tinta azul sobre papel vermelho e diferente da combi- nacao inversa.

```

2 FOR a=0 TO 7 STEP 2: POKE U
SR "a"+a,BIN 11111111: POKE UR
"a"+a+1,BIN 00000000: NEXT a
3 FOR a=0 TO 7 STEP 2: POKE U
SR "b"+a,BIN 10101010: POKE UR
"b"+a+1,BIN 01010101: NEXT a
4 FOR a=0 TO 7 STEP 2: POKE U
SR "c"+a,BIN 11110000: POKE UR
"c"+a+1,BIN 00001111: NEXT a
5 PAPER 7: INK 1: BORDER 7: C
GO TO 100
10 PAPER 0: INK 7: BORDER 0: C

```

```

L3 : PRINT AT 10,0;" 0 1 2
3 4 5 6 7 Paper"
100 FOR c=0 TO 1
105 PRINT AT 10,10;"Bright ";c;
PRINT #0;AT 0,0;" Carregue em
qualquer tecla, ou em OPTION
para obter o menu"
110 FOR a=0 TO 7: FOR b=0 TO 7
115 BEEP .003,RND*40-10
120 IF b=0 THEN PRINT AT a*2+1,
b*4;a;AT 0,0;"Ink"
130 IF c=1 THEN PRINT BRIGHT c
INK a; PAPER b;AT a*2,b*4;
;AT a*2+1,b*4;
140 PAPER 7: INK 1: BORDER 7: C
145 CLS
150 PAPER 0: INK 7: BORDER 0: C
110 PRINT AT 2,1;"Este programa
demonstra como e possivel obter
120 tonalidades diferentes a p
artir das oito cores do ZX Sp
ectrum."
120 PRINT "Alguns programas qu
e recorrem a este efeito utiliza
m um caracter grafico com um padr
ao em xadrez para a mistura das
cores. Tambem o apresentamos aqui
, mas em ge- ral as series de li
nhas do me- lhor resultado, red
uzindo o apa- rente movimento dos
pontos im- pressos."
130 PRINT AT 21,3;"Carregue em
qualquer tecla."
140 PAUSE 0
145 CLS
150 PAPER 7: INK 1: BORDER 7: C

```

```

L5 : PRINT AT 1,10; INVERSE 1;"M
ENU"
150 PRINT AT 5,0;"1. Linhas hor
izontais"; AT 7,0;"2. Xadrez";
AT 9,0;"3. Linhas quebradas";
AT 11,0;"Carregue numa das co
res 1, 2, 3": REM graficos AA,
BB, CC
170 PRINT AT 10,1;"A combinacao
de cores de cada tom pode ser
determinada atraves dos eixos da
tabela. Convem notar que as co
res de tinta e de papel nao p
odem ser trocadas, isto e, tint
a azul sobre papel vermelho e d
iferente da combi- nacao inverse
"
180 PRINT #0; AT 0,1; INK 1;"As
rotinas que geram os UDG's enco
ntre-se nas linhas 2 a 4."
190 IF INKEY#<>" THEN GO TO 19
0
200 LET a#=INKEY#
205 IF a#>"0" OR a#<"1" THEN GO
TO 200
210 LET c1=VAL a#: GO TO 10

```

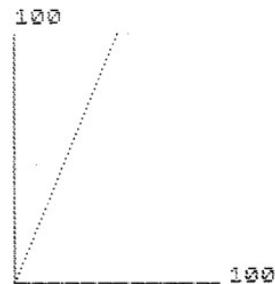
REGRESSÃO LINEAR

Este útil programa, escrito por Neal Cavalier-Smith, permite-lhe indicar um certo número de pontos (à sua escolha) e obter a recta que melhor os representa, dando a equação correspondente.

Apresentamos alguns exemplos de execução, seguidos da listagem do programa:

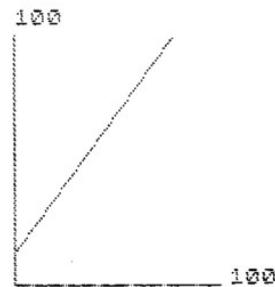
A equacao da melhor recta e':

$$Y=2X+0$$



A equacao da melhor recta e':

$$Y=1,12X+12,7$$



```

10 LET a=0: LET b=0: LET n=0
20 LET k=0: INK 7: PAPER 0
30 CLS : BORDER 0
40 INPUT "Quantos pontos? "; n
50 DIM x(n)
60 DIM y(n)
70 FOR c=1 TO n
75 INPUT "X(", c); " "; x(c); "Y
(", c); " "; y(c)
76 PLOT x(c), y(c)
80 LET a=a+x(c)
90 LET b=b+y(c)
100 NEXT c
110 LET a=a/n: LET b=b/n
120 FOR c=1 TO n

```

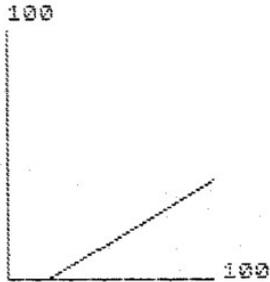
```

130 LET m=m+(((x(c)-a)*(y(c)-b)
)/(ABS(x(c)-a))+2))
140 NEXT c
145 LET m=m/n
150 LET k=b-m*a
160 LET c=0
195 PLOT 0,0: DRAW 100,0: PRINT
AT 21,13;"100"
196 PLOT 0,0: DRAW 0,100: PRINT
AT 0,0;"100"
200 PRINT AT 0,0;"A equacao da
melhor recta e':"
210 PRINT AT 2,0;"Y=";INT (m*10
0+.5)/100;"X+(";INT (k*10+.5)/10
;")"
220 FOR c=1 TO 100
225 LET d=m*c+k
230 IF d<100 AND m*c+k>=0 THEN
PLOT c,d
240 NEXT c
250 STOP

```

A equacao da melhor recta e':

$Y=0.5X+(-9.9)$



QUADRO DE PONTUAÇÕES

Este quadro de pontuações pode ser utilizado em qualquer jogo de pericia por si programado. Como exemplo, escreva um número maior do que 100, seguido do seu nome, e em pouco tempo o computador calculará e imprimirá a nova tabela de pontuações. O programa foi escrito por David Perry.

```

1 REM QUADRO DE PONTUACOES
2 REM © DAVID PERRY 1983
3 REM
4 REM Num=Numero de pontuacao
5
6 REM
7 REM
8 REM
9 BORDER 0: PAPER 0: INK 7: B
RIGHT 1: CLS
10 LET num=10: DIM n(num+1): D
IM n$(num+1,8): FOR n=1 TO num:
LET n(n)=1100-(n*100): LET n$(n)
="SPECTRUM": NEXT n
15 CLS: INPUT "Para exemplo,
indique pontuacao Pontuacao
:";sc
20 IF sc<=n(10) THEN GO TO 200
30 LET num=11: IF sc>=n(num) T
HEN INPUT "8 iniciais! ";p$: IF
LEN p$>8 THEN GO TO 20
35 PRINT AT 7,0: FLASH 1: BRIG
HT 0: INK 7: PAPER 2:"
Um momento.
40 IF sc>=n(num) THEN LET n(nu
m)=sc: LET n$(num)=p$
50 FOR a=1 TO (num-1): LET b#=
n$(a): LET c#=n$(a+1): LET b=n(a
): LET c=n(a+1): IF b<c THEN LET
n(a)=c: LET n(a+1)=b: LET n$(a)
=c#: LET n$(a+1)=b#
60 NEXT a: FOR n=1 TO num-1: I
F n(n)<n(n+1) THEN GO TO 50
70 NEXT n
80 CLS
90 PRINT AT 2,6:"QUADRO DE PON
TUACOES"
100 PRINT AT 3,6:"=====
=====
110 FOR n=1 TO num-1: PRINT AT
n+5,7: INK 6;"("; INK 2;n: INK 5
;")";AT n+5,12: INK 7;n(n): PRIN
T AT n+5,17: INK 5;n$(n): NEXT n
120 PLOT 0,0: DRAW 255,0: DRAW

```

```

0,175: DRAW -255,0: DRAW 0,-175:
PRINT AT 17,8: PAPER 2: INK 7:"
Carregue numa tecla.": LET i=0
130 LET i=i+1: IF i>7 THEN LET
i=0
140 PRINT AT 2,6: INK i:"QUADRO
DE PONTUACOES"
150 BEEP .01,i*7: PAUSE 2: IF I
NKEY#="" THEN GO TO 130
160 GO TO 15
200 PRINT AT 10,4:"Pontuacao mu
ito baixa.,"
210 PRINT AT 13,6:"Carregue num
a tecla."
220 PAUSE 0: RUN

```

FICHEIROS

Este interessante programa de ficheiros foi escrito por Michael Briggs. Armazena dados de uma forma bastante compacta, assegurando um uso bastante eficaz da memória disponível.

Quando o programa é executado, é-lhe perguntado o seu nome, o título do ficheiro, os títulos dos campos, etc. Note que nenhuma entrada pode ter um comprimento superior a 256 caracteres, e que um campo deve ter 28 caracteres ou menos.

É então impresso um menu, indicando as opções seguintes:

1. Iniciar ficheiro: Esta opção faz correr o programa, eliminando todos os dados que contenha.
2. Introduzir registos: Esta opção pede-lhe que introduza um registo, campo a campo. Em seguida colocará esse registo em ordem alfabética relativamente ao resto do ficheiro. Se cometer um erro ao introduzir qualquer registo, escreve simplesmente «Q» para voltar ao menu.
3. Procura de registos: Esta opção verifica todo o ficheiro, procurando uma cadeia de letras ou um registo numerado.
4. Imprimir todos os registos: Esta opção imprime o primeiro campo de todos os registos, atribuindo um número específico a cada um deles.
5. Gravar ficheiro: Esta opção grava ficheiro em cassette, usando o nome anteriormente atribuído àquele. Se o título não for válido, a gravação é realizada sob o nome «Index».

6. Estado: Esta opção indicar-lhe-á vários factos sobre o estado do programa, como o espaço disponível, a quantidade de registos, etc.

O leitor verificará que apesar de tudo isto parecer complexo, o programa é, de facto, bastante fácil de utilizar:

```

1 REM Michael Briggs
2 POKe 23509,40: BORDER 6: IN
K 1: PAPER 7: CLS : DIM z$(35000)
3 LET z=1: LET co=0
4 INPUT "Nome...":c#
5 INPUT "Apelido...":s#
6 INPUT "Titulo do ficheiro:
":t#
7 INPUT "Numero de campos: "
8
9 DIM a$(7,20): FOR a=1 TO f:
PRINT "Titulo do campo: ";a: IN
PUT b$: LET a$(a,2 TO )=b$: LET
a$(a,1)=CHR$(LEN b$): PRINT "-";
INK 2;b$: PRINT : NEXT a
10 PAUSE 50
11 POKe 23509,40: BORDER 6: CL
S : INK 0: PLOT 0,0: DRAW 255,0:
DRAW 0,175: DRAW -255,0: DRAW 0
,-175: INK 3
12 PRINT INK 0: BRIGHT 1: AT 1,
13-LEN t$/2;t#:"MENU"
14 PRINT AT 3,3:" No.1 Inicia
r ficheiro."
15 PRINT AT 5,3:" No.2 Acresc
entar registo."
16 PRINT AT 7,3:" No.3 Procur
ar registo."
17 PRINT AT 9,3:" No.4 Imprim
ir registos."
18 PRINT AT 11,3:" No.5 Grava
r ficheiro."
19 PRINT AT 13,3:" No.6 Estad
o do ficheiro."
20 PRINT INK 2: AT 15,8:"© Mich
ael Briggs"
21 PRINT INK 1: AT 17,11-(LEN c
#+LEN s#)/2:"FICHEIRO DE ";c#":
";s#
22 INPUT "Opcao...":c
23 CLS
24 INK 1
25 IF 0<c AND c<8 THEN GO TO c
*1000
26 GO TO 100

```

```

1000 PRINT AT 10,2;"Confirma, ";
c$;"?"
1010 INPUT y$
1020 IF y$(1)="s" OR y$(1)="S" T
HEN RUN
1030 GO TO 100
2000 INK 2: FOR a=1 TO f
2010 PRINT AT a*2-1,13-(CODE a#(
a,1)/2);"Indique ";a$(a,2 TO )
2020 NEXT a
2021 PRINT AT 21,0; INK 1;"Escre
va '0' para voltar a menu."
2030 LET e$=""
2040 FOR a=1 TO f
2050 PRINT AT a*2-1,13-(CODE a#(
a,1)/2);"Indique "; FLASH 1;a$(a
,2 TO 1+CODE a$(a))
2060 INPUT f$
2061 IF f$="0" THEN GO TO 100
2070 LET e$=e#+CHR# LEN f#+f$
2080 PRINT AT a*2-1,13-(CODE a#(
a,1)/2);"Indique ";a$(a,2 TO 1+C
ODE a$(a))
2090 PRINT INK 0;AT a*2,15-LEN f
#/2;f$
2100 NEXT a
2110 LET x1=1: FOR a=1 TO co: LE
T x#=z#(x1+2 TO x1+CODE z#(x1+1)
+1)
2115 IF x#>e$(2 TO ) THEN GO TO
2400
2120 LET x1=x1+CODE z#(x1)
2125 NEXT a
2126 LET e#=CHR# (LEN e#+1)+e$:
LET z#(z TO z+LEN e#)=e#
2131 LET z=z+LEN e#
2140 LET co=co+1: CLS : GO TO 20
00
2400 LET e#=CHR# (LEN e#+1)+e#
2410 LET z#(x1+LEN e# TO z+LEN e
#)=z#(x1 TO z)
2420 LET z#(x1 TO x1+LEN e#-1)=e
#
2430 GO TO 2131
2999 REM Busca
3000 PRINT TAB 6;"PROCURA DE REG
ISTOS, ";AT 2,0;"Escreva, ";AT 3
,0;" 'n' se quiser numero de reg
isto ' ' se quiser apenas caract
eres": INPUT "Nome ";k$
3004 LET m$="" : IF k$(1)="n" TH

```

```

EN LET m#=k$(1): LET k#=k$(2 TO
)
3005 PRINT TAB 6;"Nome=";k$
3006 LET ret=3040
3010 LET x=1: FOR a=1 TO co
3015 LET x1=x: LET v#=z#(x TO (C
ODE z#(x)+x-1))+": IF m#="N" T
HEN GO TO 3300
3016 FOR j=1 TO LEN v#-LEN k#
3020 IF v$(j TO j-1+LEN k#)=k# T
HEN GO TO 3200
3030 NEXT j
3040 LET x=x1+LEN v#-1: NEXT a
3050 PRINT " INK 0;"Carregue n
uma tecla,": PAUSE 0: GO TO 100
3200 PRINT " INK 4;"Entrada no,
.";a: LET x2=x1: LET x1=x1+1: F
OR k=1 TO f
3210 PRINT INK 2;a$(k,2 TO CODE
a$(k)+1);": PRINT INK 1;TAB
32-CODE z#(x1);z#(x1+1 TO x1+COD
E (z#(x1)))
3220 LET x1=x1+1+CODE z#(x1)
3225 NEXT k
3240 LET x1=x2: GO TO ret
3300 IF a<>VAL k# THEN GO TO 304
0
3310 LET ret=3050
3320 GO TO 3200
4000 PRINT "Qualquer tecla inter
ompe o rola-mento da ima
gem."
4005 PAUSE 0
4010 PRINT AT 18,15-LEN t$/2; IN
K 2;t$
4020 INK 1: PRINT
4030 PRINT
4040 LET x=1: FOR a=1 TO co
4049 PRINT AT 21,0: POKE 23692,-
1
4050 PRINT "0000";AT 21,4-LEN (S
TR# a);a;":z#(x+2 TO x+1+CODE
(z#(x+1)))
4060 LET x=x+(CODE z#(x))
4065 IF INKEY#<>" THEN GO TO 40
65
4070 NEXT a
4075 PRINT "TAB 6;"Carregue num
a tecla."
4080 PAUSE 0: GO TO 100
5000 PRINT "O ficheiro sera gra
vado sob o nome: "; PRINT AT 10

```

```

,15-LEN (t#/2) FLASH 4/t#
5005 IF t#="" OR LEN (t#)>10 THEN
PRINT AT 10,1;" " IND
EX " ". SAVE "INDEX" L
INE 100: GO TO 100
5010 SAVE t# LINE 100
5020 GO TO 100
5000 PLOT 0,0: DRAW 255,0: DRAW
0,175: DRAW -255,0: DRAW 0,-175
5010 PRINT AT 1,10-(LEN t#)/2) B
RIGHT 1; INK 2;t#;" ESTADO"
5020 PRINT AT 3,2;"No momento :
";INT (((PEEK 23672+255*PEEK 236
73)+65536*PEEK 23674)/50);" segs.

5030 PRINT AT 5,10-(LEN s#+LEN c
#)/2);"FICHEIRO DE ";c#;" "s#
5040 PRINT AT 7,8;"No. de regist
os: ";c#
5050 PRINT AT 9,2;"Espaco livre:
";35000-z
5060 PRINT AT 11,4;"Caracteres n
o ficheiro: ";z
5070 PRINT AT 13,3;"Programa © M
ICHAEL BRIGGS"
5075 PRINT AT 20,6;"Carregue num
a tecla."
5080 PRINT AT 3,2;"No momento: "
;INT (((PEEK 23672+255*PEEK 2367
3)+65536*PEEK 23674)/50);" segs.
"
5085 IF INKEY$<>"" THEN GO TO 61
00
5090 GO TO 5030
6100 GO TO 100

```

PROGRAMAÇÃO ESTRUTURADA

ESBOÇO DE UM PROGRAMA

Escrevi já muitos artigos e capítulos de livros com o objectivo de melhorar a capacidade de programar daqueles que os lêem. Mas de cada vez que apresento uma lista das «coisas que devem ser feitas quando se programa», ou explico o conteúdo desses textos a alguém, recordo uma velha história de um lavrador.

Uma vez foi abordado por um jovem que vendia cursos por correspondência sobre agricultura. Disse-lhe este: «Não compreende que se tirar este curso ficará a saber muito mais sobre o seu trabalho?». O lavrador respondeu: «Nem sequer consigo trabalhar tão bem como sei...»

O mesmo acontece, parece-me, com os conselhos sobre programação. Sei perfeitamente que não programa «tão bem quanto sei». Muitas vezes contrário as regras todas, entrando directamente na codificação de um programa complexo sem sequer me lembrar de que existe uma «programação estruturada».

Sugiro portanto que o leitor tenha presente, ao ler este apêndice, que não acredito verdadeiramente que você — ou qualquer outra pessoa — vá aproveitar as ideias aqui apresentadas como regras fixas, que devam ser obedecidas em todas as circunstâncias. A melhor forma de considerar o que aqui se apresenta, ou qualquer outro material do mesmo tipo que encontre, consiste em lê-lo cuidadosamente e fazer uma avaliação própria do interesse de cada sugestão dada. Depois, aplique as ideias que lhe parecem mais apropriadas.

A ideia básica que desejo transmitir neste apêndice é que se pode estruturar um programa a partir de um esboço prévio deste.

A ideia é simples, mas muito valiosa na medida em que nos permite escrever programas complicados, como muitos dos apresentados neste livro. Como é óbvio, o leitor talvez esteja já a criar programas complexos sem precisar da ideia aqui sugerida. Mas mesmo que tal aconteça, sugiro que pense cuidadosamente no que está contido neste apêndice, tentando descobrir algo que possa facilitar o seu trabalho.

A ideia fundamental da programação estruturada está contida na frase «programação de cima para baixo». Isto sugere que se começa a programar definindo o objectivo geral do programa que se quer escrever. Em seguida tomam-se algumas notas — cada uma das quais acabará por ser transformada numa subrotina — que cobrirão cada uma das tarefas que o computador deverá realizar.

Para aclarar melhor este último parágrafo vou dar um exemplo concreto. Vejamos então os passos que devem ser dados quando se pretende escrever um programa sobre o Jogo do Galo (note que a programação estruturada produz listagens muito mais extensas do que o habitual. No entanto, este maior comprimento do programa é contrariado pelo facto de os programas assim escritos serem muito fáceis de emendar e melhorar, e de a sua estrutura ser fácil de compreender por outros programadores. É provável que o leitor não possa dizer o mesmo de alguns dos seus programas actuais.

DEFINIR TABULEIRO INICIAL

Lance do Computador

- (A) VERIFICAR SE O QUADRADO CENTRAL ESTÁ VAZIO; SE NÃO COLOCAR NELE UMA PEÇA
 - (B) VERIFICAR SE EXISTE UM MOVIMENTO GANHANTE DO COMPUTADOR, E SE HOVER EXECUTÁ-LO
 - (C) VERIFICAR SE O JOGADOR GANHARÁ NO LANCE SEGUINTE, E SE SIM BLOQUEAR O SEU MOVIMENTO
 - (D) SE NÃO HOVE MOVIMENTO DO JOGADOR, VERIFICAR SE O COMPUTADOR PODE FAZER UM MOVIMENTO. SE SIM MOVER, SE NÃO DECLARAR UM EMPATE.
- IMPRIMIR TABULEIRO
 ACEITAR O LANCE DO JOGADOR
 IMPRIMIR TABULEIRO
 VERIFICAR SE O JOGADOR GANHOU. SE SIM PARAR
 «GO TO» LANCE DO COMPUTADOR.

Como o leitor pode verificar se dedicar alguns momentos ao estudo deste esboço, pode saber exactamente quais os passos da execução feita pelo computador, as decisões que deverá tomar, e os resultados alternativos destas decisões.

O leitor já deu o seu primeiro passo no sentido da escrita de um programa do Jogo do Galo. A propósito, note que não interessa se já sabe ou não a forma concreta como o computador irá executar as funções necessárias. Apenas interessa que saiba por agora quais as operações gerais que deverão ser executadas por uma dada ordem.

O segundo passo deste processo consiste em transformar o esboço feito numa série de chamamentos de subrotinas. Em linguagens mais estruturadas do que a Basic empregue no Spectrum torna-se relativamente fácil chamar um certo número de subrotinas no interior de um ciclo sem fim, usando uma estrutura como a DO/WHILE ou a REPEAT/UNTIL. Estas permitem ao programador repetir uma série de passos de programa indefinidamente, até ser satisfeita uma determinada condição ou se alterar o estado de um indicador («pointer» ou «flag»).

Na Basic por nós empregue, no entanto, teremos de nos contentar com o humilde GO TO. Grande parte das críticas à GO TO, que os «programadores sérios» vêem como o maior insulto à verdadeira arte da programação alguma vez feito, devem-se às primeiras e relativamente primitivas versões da Basic, nas quais só era possível usar uma IF/THEN seguida de GO TO. Isto significava que o programa saltava repetidamente de uma instrução para outra, conduzindo a uma codificação «spaghetti» extremamente difícil de interpretar.

A segunda fase consiste portanto em transformar o esboço já preparado numa série de chamamentos de subrotinas, realizados num ciclo sem fim produzido por uma GO TO da seguinte forma:

```

10 REM Nome
20 GO SUB 9000: REM inicializa
30 GO SUB 1000: REM lance do c
  computador
40 GO SUB 8000: REM imprimir i
  magem
50 GO SUB 2000: REM aceitar la
  nce do jogador
60 GO SUB 8000: REM imprimir i
  magem
70 IF jogador-nao-ganha AND co
  mputador-nao-ganha THEN GO TO 30
80 PRINT "PARABENS"
```

Como se pode ver, temos já a estrutura geral de um programa do jogo do Galo, apesar de ainda não termos qualquer noção de como funcionará de facto o programa. Podemos começar agora a escrever o programa partindo do princípio.

Existem outras vantagens deste método de «esboço». Se existir alguma coisa que não possamos, nesta fase, programar (como o modo como o computador descobrirá quem ganha), poderemos simplesmente colocar uma instrução PRINT no local apropriado, por exemplo PRINT «Verificar vitória», e continuar a desenvolver outras partes do programa. Depois, de cada vez que o computador deva verificar a vitória, imprimirá esta mensagem. Isto significa que se pode desenvolver o programa sem ficar retido por uma qualquer dificuldade secundária que não possa ser resolvida no momento.

A segunda vantagem ocorre no final do desenvolvimento do programa, na fase de correcção de erros. Sempre considereei esta como a mais frustrante, e de muitos pontos de vista a menos compensadora fase do trabalho de programação. Se bem que consiga muitas vezes pôr o programa em funcionamento em condições razoáveis com alguma rapidez (se bem que o meu programa de xadrez, em Basic, tenha demorado a corrigir cerca de seis meses), desde um funcionamento em condições razoáveis até um verdadeiro bom funcionamento pode demorar bastante tempo. Quando porém se constrói o programa da forma indicada, a fase de correcção de erros é bastante simplificada.

Por exemplo, o leitor poderá verificar no caso do programa do Jogo do Galo que o computador tende a ignorar o canto inferior do lado direito, quando um movimento para esta posição lhe permitiria ganhar, ou bloquear um movimento ganhante do jogador. Da sequência de rotinas definida anteriormente sabemos que o lance do computador deve ser realizado num qualquer ponto entre as linhas 1000 e 1999. Isto permite-nos imediatamente concentrar a procura do erro. Se cada subrotina tiver sido programada também como uma série de subrotinas, cada uma delas construída da forma já citada, será muito fácil determinar a instrução errada.

Vejamos um pouco melhor esta ideia da transformação de cada subrotina numa série de outras subrotinas. Observemos então a subrotina que se inicia na linha 1000, aquela em que o computador realiza de facto o seu movimento, dado que esta é a mais importante (e a mais difícil de programar) de todas.

A subrotina pode iniciar-se do seguinte modo:

```
2000 REM lance do computador
2010 LET mover=0: REM Se passa a
1, descoberto lance valido
2020 GO SUB 2200: REM Verificar
se o quadrado central esta
vazio
2030 IF mover=1 THEN RETURN
2040 GO SUB 2400: REM Verificar
se existe lance vencedor
```

```
2050 IF mover=1 THEN RETURN
2060 GO SUB 2600: REM Verificar
se e' possivel evitar a vitoria
do jogador
2070 IF mover=1 THEN RETURN
2080 GO SUB 2800: REM Verificar
se existe movimento possivel
2090 IF mover=1 THEN RETURN
2100 REM O retorno com mover=0
significa que nao e' possivel
qualquer movimento
2110 RETURN
```

Como já disse, os programas construídos usando o «ciclo sem fim de subrotinas», juntamente com «subrotinas dentro de subrotinas» pode aumentar bastante o comprimento do programa. No entanto, não há normalmente necessidade de nos preocuparmos com a memória disponível (ao contrário do que acontecia com o ZX81 ou o TS1000, quando a contagem dos bytes restantes se transformava no aspecto dominante do trabalho de programação). O maior trabalho tido a escrever um maior número de instruções será completamente recompensado pela maior clareza das listagens obtidas, e pelo menor tempo perdido na correcção dos erros que inevitavelmente ocorrerão.

O leitor verificará certamente que, se tiver a preocupação de começar por fazer um esboço da estrutura do programa a escrever, todo o processo de construção do programa em causa se tornará de facto muito mais simples do que se não tiver esse cuidado. Certamente observará também que o tempo gasto na construção do programa será muito mais produtivo do que no caso de se lançar imediatamente no trabalho de programação, sem dedicar algum tempo previamente à definição de um esboço daquilo que lhe interessa fazer.

O melhor conselho que lhe posso dar consiste portanto em ignorar o velho agricultor, e tentar desde já «programar tão bem quanto sabe»...

ÍNDICE

PREFÁCIO	
JOGOS DE PERÍCIA	
Protector	9
Bem alto	14
Bombardeiro	21
Passaros	23
Helicóptero	32
Serpente	37
Corridas	41
Zombies	46
AVENTURAS	
O Gigante e a Cidade	52
Portas do Destino	68
JOGOS COM GRÁFICOS MÓVEIS	
Corrida ao ouro	91
Tarântula	97
A Selva	104
Lógica	109
Na horta	109
Fantasmas	111
Sapo	114
TEMPOS LIVRES	
Pares de cartas	120
Submarino	123
O Carrasco	132

JOGOS NO ESPAÇO	
Fuga entre as estrelas	135
Space Trek	135
Sonda espacial	141
Tempestade lunar	148
A O PRAZER DO LABIRINTO	
Gerador de labirintos	152
Labirintos irregulares	154
Muros do suspense	158
Muro do suspense	161
Labirinto a 2 níveis	162
Labirinto móvel	165
Labirinto rolante	169
Labirinto a 3 dimensões	177
Labirinto a 3 dimensões — II	184
JOGOS PARA DOIS JOGADORES	
Quadrados	189
Quatro em linha	196
Tanques	198
AUXILIARES EM CÓDIGO-MÁQUINA	
Monitor de memória	203
Guardar Screens em código-máquina	208
Guardar Screens em Basic	209
Irlanda	210
Copiador	212
UTILITÁRIOS E DEMONSTRAÇÕES	
Biorritmos	216
Pintura	218
Timothy Leary	219
Teste de cores	221
Demonstração de portas lógicas	221
Sons	222
Arco-Íris	223
Regressão Linear	226
Quadro de Pontuações	229
Ficheiros	230
PROGRAMAÇÃO ESTRUTURADA	
Esboço de um programa	235

Este livro foi fotocomposto
 Por José Maria Marques—Lisboa
 e impresso na
 Tipografia Nunes, Lda. — 4000 Porto
 para a
 EDITORIAL PRESENÇA, LDA.