Eis uma obra que lhe permite explorar todas as potencialidades do seu ZX SPECTRUM, desde aventuras aliciantes com os programas para jogos — de guerra, de aventuras, de casino, espaciais — até à música e à poesia, não esquecendo os programas utilitários que lhe permitirão resolver equações, converter unidades, construir gráficos e mesmo verificar a sua conta bancária! Muitas ideias que podem ser aplicadas para melhoramento dos seus próprios programas!

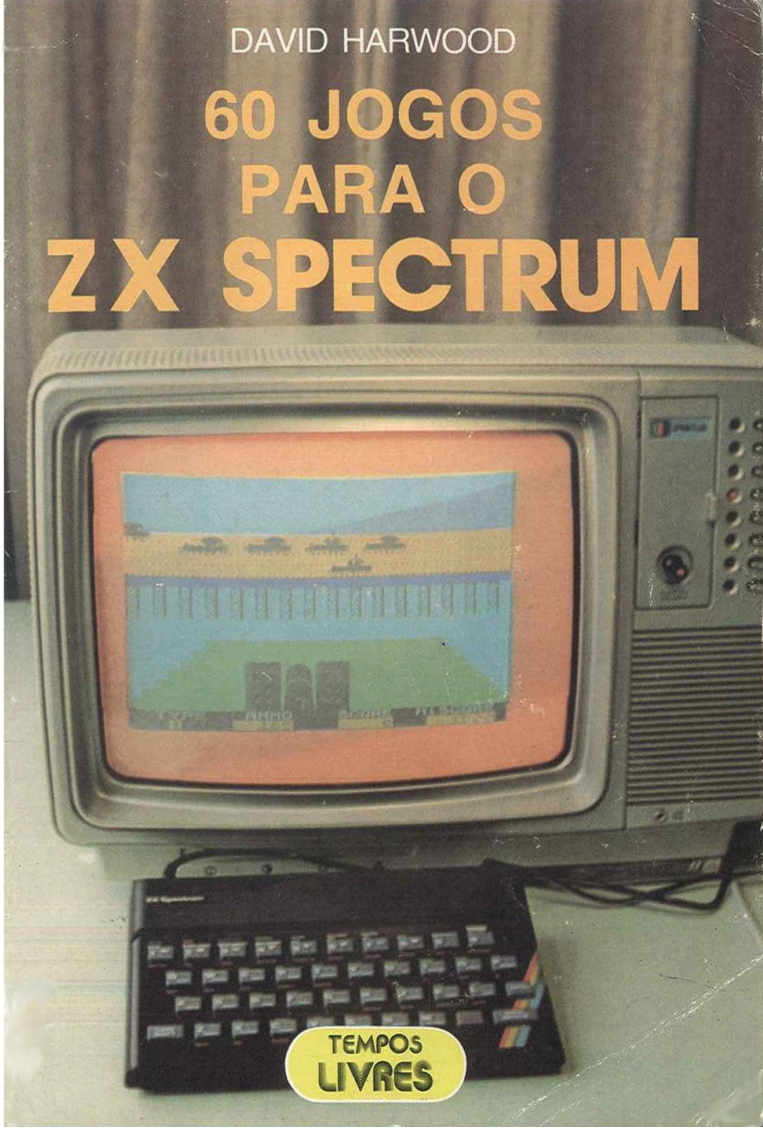
EDITORIAL PRESENCA / MARTINS FONTES

150

60 Jogos Para o ZX SPECTRUM

DAVID HARWOOD

60 JOGOS PARA O ZX SPECTRUM



TEMPOS LIVRES

CULTURA E TEMPOS LIVRES

1. ABC do Xadrez, *Petar Trifunovitch e Sava Vukovitch*
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. Aussel*
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 Huré*
18. Os Transistores, *Fernand Huré*
19. Karaté I, *Albrecht Pflüger*
20. Iniciação ao Radiocomando dos Modelos Reduzidos, *C. Péricone*
21. Construa o seu Receptor, *B. Fighiera*
22. Montagens Electrónicas, *B. Fighiera*
23. O Berbequim Eléctrico, *Villy Dreier*
24. Cactos, *J. Nilaus Jensen*
25. Iniciação à Alta Fidelidade, *Peter Turner*
26. O Aquário de Água Doce, *Paulo de Oliveira*
27. ABC do Ténis, *Fonseca Vaz*
28. Karaté II, *Albrecht Pflüger*
29. ABC da Criação de Canários, *Curt Af Anehjelm*
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, *Bärbel Gerber e Horst Bielfeld*
38. Manual do Super 8 — II, *Myron A. Matzkin*
39. ABC do Mergulho Desportivo, *Walter Mattes*
40. Circuitos Integrados/Aplicações Práticas, *F. Bergtold*
41. A Apicultura, *H. R. C. Riches*
42. ABC do Cultivo das Plantas, *H. G. Witham Fogg*
43. ABC da Criação de Pombos, *Kai 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, *Fabián 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 Coelhos, *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. Boëlle e B. 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. Rayer*
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, *Matias Josa*
79. Rádio — Fundamentos e Técnicas, *Gordon J. King*
80. Como Fazer Gelados, *Sylvie Thiébault*
81. Iniciação à Jardinagem, *Noel Clarasó*
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 Persuy*
87. Aprenda Electricidade, *Heinz Richter*
88. Taxidermia. Embalsamamento de Aves e Mamíferos, *Harry Hjortaa*
89. Jogging — Correr para Manter a Forma, *Werner Sonntag*
90. ABC da Cozinha Chinesa, *Sonya 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 Minitrampolim na Escola, *Sonja Helmer Jensen e Klaus Dano*
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 Coelhos, *C. F. Snow*
100. Iniciação ao Futebol, *Wieland Männle e Heinz Arnold*
101. Horóscopos Chineses, *Georg Haddenbach*

102. Guia Prático de Marcenaria, *Charles H. Hayward*
103. Andebol, *Fritz e Peter Hattig*
104. Dispositivos Anti-Roubo, *H. Schreiber*
105. Perus, Pintadas e Codornizes, *Jerome Sauze*
106. Crepes — Doces e Salgados, *Florence Arzel*
107. Aperitivos e Entradas, *Myrette Tiano*
108. Ténis de Mesa, *Leslie Woollard*
109. Aprenda Surf, *R. Abbott e M. Baker*
110. Futebol — Técnica e Táctica, *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 Mourthe*
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. Mélusson*
133. Guia Prático de Basic, *Roger Hunt*
134. Introdução à Electrónica Digital, *Ian R. Sinclair*
135. ABC do Video, *David Matthewson*
136. Fotografia em Movimento, *Don Morley*
137. Guia Prático de Cobol, *Ray Welland*
138. Fotografia a Pequena Distância, *Sidney F. Ray*
139. Guia Prático da Canaricultura, *Dr. Manuel Gonçalves*
140. Minielectrónica para Amadores, *Heinz Richter*
141. ABC da Programação de Computadores, *John Shelley*
142. TAROT — O Futuro pelas Cartas, *Edwin J. Nigg*
143. ABC da Equitação, *Dorothy Johnson*
144. Como programar o seu ZX 81, *Patrick Gueulle*
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, *David S. Maclean*
150. 60 Jogos para o ZX Spectrum, *David Harwood*

DAVID HARWOOD

60 JOGOS PARA O ZX SPECTRUM

EDITORIAL PRESENÇA

Título original:
60 GAMES AND APPLICATIONS FOR THE ZX SPECTRUM
© *Copyright by* Harwood, 1982
Tradução de Conceição Jardim e Eduardo Nogueira
Capa de R. Silva

Reservados todos os direitos para Portugal
à EDITORIAL PRESENÇA, LDA.
Rua Augusto Gil, 35-A — 1000 LISBOA

INTRODUÇÃO

Com este livro, e o seu ZX Spectrum, o leitor pode lançar-se numa grande aventura — ou em várias. O Spectrum, apesar das suas pequenas dimensões, é um computador com um imenso poder... e este livro contém bastantes programas que demonstram esse poder, e a flexibilidade de uso que o Spectrum permite.

Encontrará aqui uma vasta gama de jogos, desde o Ouro do Dragão (aventura no passado) até Entre as Estrelas (exploração no futuro); desde Zombies até jogos de casino. Todos os programas contêm ideias que podem ser adaptadas para melhoramento dos seus próprios programas.

Mas o livro não é inteiramente composto por jogos, pois temos a certeza de que o leitor não deseja passar toda a sua vida jogando... Se bem que os jogos sejam bastante divertidos, e sejam aliás uma das coisas que tornaram os computadores tão populares como são hoje, muitos jogos nem se quer dão uma pálida ideia das enormes potencialidades do computador. Para mostrar a sofisticação da máquina que possui, incluimos uma vasta gama de programas que lhe podem ser úteis na sua vida quotidiana. Os programas utilitários justificam, por si só, a compra de um Spectrum. Se lhe interessa construir gráficos ou resolver equações, existem programas para o fazer. A conversão entre unidades, a ordenação alfabética ou numérica, treino em código Morse, renumeração das linhas do seu programa — tudo pode

encontrar aqui, numa forma pronta a ser usada. E se quiser vigiar de perto a sua conta bancária pode usar ainda o programa incluido para esse fim.

É chegado o momento de ligar a alimentação do seu Spectrum e passar ao que se segue.

Gostaria antes do mais de agradecer as ideias e programas originais das seguintes pessoas:

Roger Macintyre, D. C. Owen, Martin Frobisher, Paul Toland, Stuart Roberts, Tim Rogers, I. S. Howson, Andrew Lovering, R. Stubbs, John Knight, Said Hassan, Jeremy Ruston, Paul Stevens, J. Elliot, M. Charlton, K. Mahogany, Daniel King e Tim Hartnell.

Feliz programação...

David Harwood

ENTRE AS ESTRELAS

Você é responsável pela segurança de um cubo de espaço medindo 10 X 10 X 10. A Federação Terrestre, não poupando qualquer gasto no que toca à defesa do planeta, pôs à sua disposição uma nave espacial equipada com um Spectrum funcionando como computador de bordo. Cabe-lhe agora defender as vias de comunicação usadas na área que lhe foi atribuída.

Quando executa o programa, começa por saber a sua posição no interior do cubo, indicada por três coordenadas sob o título "Nave em cruzeiro nas coordenadas:".

A primeira coordenada é a sua posição Norte/Sul (com os números mais pequenos a sul); a segunda é a sua posição transversal, isto é leste/oeste; a tèrceira é a sua posição no interior do cubo (frente/trás). É fácil compreender que a sua capacidade para visualizar em três dimensões lhe será aqui muito útil.

A nave inimiga move-se muito lentamente no interior do cubo, mas apesar de você conhecer sempre a direcção em que ela se encontra não conhece a distância. Deve atingi-la tantas vezes quanto possível antes de o contador de tempo atingir zero, mas sem colidir com a nave inimiga. Se ficar sem energia o jogo terminará. Você sabe quando está suficientemente próximo para disparar porque o computador lhe dirá que a nave inimiga está a disparar sobre a sua. De cada vez que é atingido as suas reservas de energia diminuirão drasticamente.

O jogo é bastante fácil. Basta-lhe carregar na tecla correspondente à direcção em que se deseja mover:
"N","S","E" ou "W" para mover-se em direcção a norte, sul, leste ou oeste; "A" para avançar, "R" para retirar, "L" para disparar o seu laser sobre a nave inimiga. Se por exemplo você sabe que o inimigo se encontra a norte, pode carregar na tecla "N" continuamente até se encontrar na mesma trajectória norte/ /sul do inimigo, verificando depois a distância a que se encontra através de disparos.

Verificará que o próprio programa lhe ensinará o modo mais fácil de jogar. Não se esqueça de que deve aproximar-se tanto quanto possível do inimigo para disparar, e de que a sua tarefa consiste em obter a maior pontuação possível antes de o tempo terminar.

```
  10 REM Entre as estrelas
  20 REM Por Roger Macintyre
  25 BORDER RND*7: PAPER 7: CLS
  30 GO SUB 1070
  40 GO SUB 800
  50 IF t<0 THEN GO TO 500
  80 PRINT AT 17,0; INK RND*6;"
Indique ordens"
  90 PRINT AT 18,2; INK RND*6; F
LASH 1;"N,S,E,W,(L)aser,"'"
(A)vançar,(R)etirar"
 100 LET t=t-.25
 120 IF INKEY$="" THEN GO TO 120
 125 BEEP .1,RND*20
 130 IF INKEY$="l" THEN GO SUB 3
20
 140 LET x=x+(INKEY$="s")-(INKEY
$="n")
 150 LET y=y+(INKEY$="e")-(INKEY
$="w")
 160 LET z=z-(INKEY$="a")+(INKEY
$="r")
 195 PRINT AT 5,0;t$;t$
 200 GO SUB 620
 210 IF RND>.5 THEN GO TO 40
 240 LET a=a+INT ((RND*3)-(RND*3
))
```



```
 250 IF a<1 THEN LET a=1: IF a>1
0 THEN LET a=10
 260 LET b=b+INT ((RND*3)-(RND*3
))
 265 IF b<1 THEN LET b=1: IF b>1
0 THEN LET b=10
 270 IF RND>.5 THEN GO TO 40
 280 LET c=c+INT ((RND*3)-(RND*3
))
 290 IF c<1 THEN LET c=1: IF c>1
0 THEN LET c=10
 310 GO TO 40
 320 REM ** disparar laser
 330 LET t=t-.75
 480 LET t=t+1
 482 PAUSE 100
 485 PRINT AT 1,0;t$
 490 RETURN
 500 CLS : BORDER RND*6: PRINT
TAB 3; INK RND*6; FLASH 1;"
FINAL"
 530 IF t1<=0 THEN PRINT INK RND
*6;"Estivemos no espaço muito tempo"

 540 IF t>0 THEN PRINT INK RND*6
;" Fomos derrotados"
 555 PRINT AT 10,0; INK RND*6;"
Energia restante  =";t;" ergs"
 560 IF t<=0 THEN PRINT INK RND*
6; FLASH 1;"          Reservas de energia esgo
tadas          "
 570 STOP
 580 CLS : PRINT ' INK RND*6;
"Colidimos com a nave"; TAB 8,"INIMIGA"

 610 STOP
 620 REM ** Disparos inimigos **
 630 IF ABS (a-x)>3 OR ABS (b-y)
>3 OR ABS (c-z)>3 THEN RETURN
 650>IF RND>.75 THEN RETURN
 660 BEEP .3,RND*20: PRINT AT 1,
0; INK RND*6; FLASH 1;"       AL
IENS FIRING AT US      "
 670 PAUSE 100
 690 PRINT AT 1,0;t$
 700 IF RND>.7 THEN GO TO 770
 710 BEEP .3,RND*20: PRINT AT 1,
0; INK RND*6; FLASH 1;" ALIEN F
IRE HAS HIT US "
 720 LET t=t-7
```

```
730 IF l<=0 THEN GO TO 500
740 PAUSE 100
755 PRINT AT 1,0;t$
760 RETURN
770 BEEP .3,RND*20: PRINT AT 1,
0; INK RND*6; FLASH 1;"
Inimigo falhou! "
780 PAUSE 100
792 PRINT AT 1,0;t$
795 RETURN
800 REM ** Estado **
810 BORDER RND*6
850 PRINT AT 10,0;" Energia restante =
";l;" ergs"
870 LET t1=t1-1: IF t1=0 THEN G
O TO 500
890 PRINT AT 19,20;" tempo : ";t1;
" "
900 IF l<3 THEN PRINT AT 12,4;
FLASH 1;" ENERGIA REDUZIDA "
920 PRINT AT 20,19;"Relatório: ";t
930 PRINT AT 14,0;" Nave em cruzeiro nas
coordenadas:"
935 PRINT TAB 4;x;" ";y;" ";z
940 IF a=x AND b=y AND c=z THEN
GO TO 580
960 PRINT AT 5,0;" ";AT 5,0;"
";AT 5,0;" "
965 PRINT AT 5,0;"A nave inimiga está"
970 IF a<>x OR b<>y THEN PRINT
"a"
980 IF a<x THEN PRINT " Norte ";
990 IF a>x THEN PRINT " Sul ";
1000 IF b>y THEN PRINT " Leste ";
1010 IF b<y THEN PRINT " Oeste ";
1030 IF c>z THEN PRINT " atrás de nós"
1040 IF c<z THEN PRINT " à nossa frente"
1060 RETURN
1070 REM ** Inicializar **
1090 LET l=25+INT (RND*30): LET
l=0: LET t1=35
1140 LET a=INT (RND*10)+1: LET b
=INT (RND*10)+1
1150 LET c=INT (RND*10)+1: LET x
=INT (RND*10)+1
1160 LET y=INT (RND*10)+1: LET z
=INT (RND*10)+1
1180 LET t$="
"
1210 RETURN
```


Esta versão do jogo "Mastermind" utiliza as letras "A", "B", "C", "D", "E" e "F". Escolhe quatro letras (que podem ser repetidas no interior do código), sendo necessário adivinhar o código no menor número de tentativas possível. Como verá ao executar o programa, uma letra correcta na posição errada produzirá um "+", enquanto que a letra correcta na posição certa produzirá um "*".

As tentativas não válidas são rejeitadas.

FASTERMIND foi escrito por D. C. Owen.

```
80 REM fastermind
90 REM por D. C. Owen
95 BORDER RND*7: INK RND*6: CL
S
100 LET l=0: DIM a(4): DIM b(4)
130 FOR z=1 TO 4
140 LET b(z)=INT (RND*6)+1
150 NEXT z
160 POKE 23692,-1
170 PRINT AT 21,0;"Fastermind a
b c d e f";AT 21,0; OVER 1;"____
_____"
180 LET l=l+1
185 PRINT "" Indique número "; l
190 INPUT LINE a$
195 LET k=0: LET j=0
200 FOR z=1 TO 4
210 LET a(z)=CODE a$-96
220 IF a(z)>6 THEN GO TO 165
230 PRINT CHR$ (a(z)+96);" ";
235 LET b(z)=ABS b(z)
240 IF a(z)<>b(z) THEN GO TO 26
0
250 LET k=k+1: LET a(z)=0: LET
b(z)=-b(z)
280 LET a$=a$(2 TO )
290 NEXT z
300 FOR h=1 TO 4: FOR z=1 TO 4
310 IF a(h)<>b(z) THEN GO TO 36
0
330 LET j=j+1: LET b(z)=-b(z)
```

```
 350 GO TO 370
 360 NEXT z
 370 NEXT h
 400 PRINT TAB 10; INVERSE 1;"
Pontuou ";
 410 IF k=0 THEN GO TO 450
 420 FOR z=1 TO k
 430 BEEP .2,RND*20: PRINT FLASH
 1; INK RND*6;"*"; FLASH 0;" ";
 440 NEXT z
 450 IF j=0 THEN GO TO 490
 470 BEEP .2,RND*20: PRINT FLASH
 1; INK RND*6;"+"; FLASH 0;" ";
 480 NEXT z
 490 IF k<4 THEN GO TO 180
 500 PRINT '' PAPER RND*6; INK 9
; FLASH 1;"!!!!!!!!!!!  CONSEGUIU  !
!!!!!!!!!!"
```

Fastermind a b c d e f

Indique número 1
a b c d Pontuou * +
Indique número 2
a b d e Pontuou * * *
Indique número 3
a b d a Pontuou * * +
Indique número 4
a b b e Pontuou * *
Indique número 5
a b d d Pontuou * *
Indique número 6

"TRAILER"

Este jogo baseia-se num outro intitulado SNAKE, no qual é necessário mover-nos pelo visor sem chocar com a nossa cauda... Em TRAILER dispomos de três vidas, sendo a pontuação e o número de vidas restantes impressos na linha superior. Quando se executa o programa deve-se escolher uma velocidade entre 0 e 10, onde 0 é a mais rápida. A direcção do movimento é comandada pelas teclas de cursor (5, 6, 7 e 8).

```
  1 REM TRAILER por D. Harwood
  4 REM ** Definir caracter — executar linhas 5 a 7
 antes de escrever o resto do programa *
  5 FOR t=1 TO 7: READ n
  6 POKE USR "k"+t,n: NEXT t
  7 DATA 60,126,219,219,255,255
,102,0
  8 REM Para obter ● escrever 'k' em modo gráfico
  9 LET p$="● ● ●"
 10 LET s=0: LET c=3
 11 BORDER 7: PAPER 7: INK 0: C
LS
 12 REM instruções
 15 PRINT TAB 8; INK 1;"T ";: P
AUSE 10: PRINT INK 2;"r ";: PAUS
E 10: PRINT INK 3;"a ";: PAUSE 1
0
 20 PRINT INK 4;"i ";: PAUSE 10
: PRINT INK 1;"l ";: PAUSE 10: P
RINT INK 2;"e ";: PAUSE 10: PRIN
T INK 3;"r"
 30 PRINT '''" Este é um jogo de perícia. Deve"
 40 PRINT " mover o seu homem pelo visor"
 50 PRINT " sem passar pelos mesmos sítios"
 60 PRINT " Dispõe de três vidas."
 65 INPUT " Velocidade (0-10):0 fast
:10 slow ";sp
 70 PRINT '' INK 2; FLASH 1;AT
10,10;" BOA SORTE! "
 80 PAUSE 300: CLS
```

```
  85 PLOT 248,168: DRAW -241,0:
DRAW 0,-161: DRAW 241,0: DRAW 0,
160
  90 LET a=11: LET b=16: LET a$=
STR$ (INT (RND*4)+5)
  95 PRINT AT 0,20;p$
 100 PRINT AT a,b;"@"
 105 FOR z=1 TO sp*2: NEXT z
 110 IF INKEY$<>"" THEN LET a$=I
NKEY$
 115 PRINT AT a,b;"*"
 120 LET a=a+(a$="6")-(a$="7")
 130 LET b=b+(a$="8")-(a$="5")
 140 IF a=21 OR a=0 THEN LET a=A
BS (a-20)
 150 IF b=31 OR b=0 THEN LET b=A
BS (b-30)
 160 BEEP .01,RND*20
 170 LET s=s+1: PRINT AT 0,0;"Sc
ore: ";s
 180 IF SCREEN$ (a,b)="*" THEN G
O TO 200
 190 GO TO 100
 200 CLS : PRINT PAPER 1; INK 7;
 FLASH 1;AT 11,0;"!!!!!!!!!!!!!!!!
BANG!!!!!!!!!!!!!!!"
 210 LET c=c-1
 220 IF c=0 THEN GO TO 270
 230 LET p$="": FOR i=1 TO c: LE
T p$=p$+"@ ": NEXT i
 240 PRINT AT 0,0;" Só restam";c;" homens "
 250 PRINT AT 0,0;"     PREPARE-SE!"
 260 GO TO 80
 270 PRINT AT 13,8; FLASH 1;"
JOGO TERMINADO"
```


BOMBARDEIRO

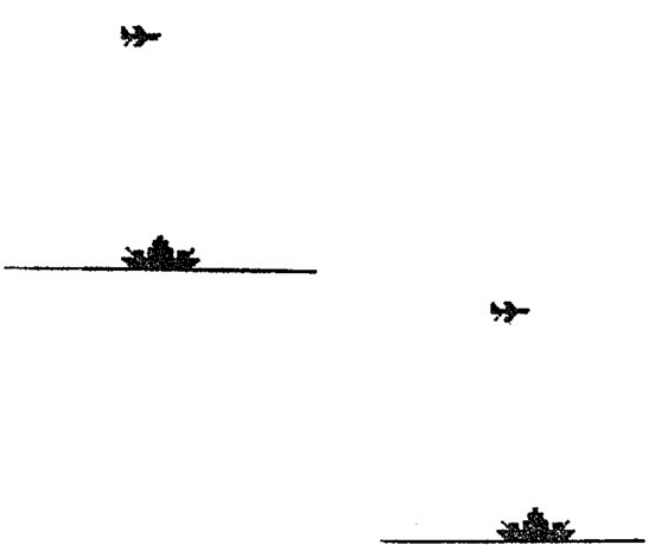
Este programa é mais difícil de jogar do que se pode pensar de início. Em BOMBARDEIRO é necessário destruir um couraçado bombardeando-o ao sobrevoá-lo com o nosso avião. Para tornar a tarefa mais difícil, o avião desloca-se no ar a uma velocidade dupla da do navio, sendo por vezes afectado por ventos que aumentam ainda mais a sua velocidade.

O jogador dispõe de 20 bombas nesta versão do jogo, mas é possível alterar facilmente este número modificando a variável m na linha 110. Dispara carregando em "F". Não se vê nada a cair do avião, mas se atingir o navio este explode e em seguida reaparece miraculosamente e continua a sua viagem incansável da esquerda para a direita. Existe um limite de tempo de 300 segundos, terminando o jogo quando acabam as bombas ou o tempo. A pontuação máxima é 5340, mas é praticamente impossível atingi-la.

O programa ilustra as excelentes potencialidades gráficas do Spectrum, que neste caso imprime no visor um avião e um navio. BOMBARDEIRO foi escrito por Martin Frobisher.

```
  1 REM  Bombardeiro
  2 REM  Por Martin Frobischer
 10 REM  Escreva esta rotina e execute-a antes de
escrever o resto.
 20 FOR a=1 TO 8: READ a$
 30 FOR t=0 TO 7: READ b
 40 POKE USR a$+t,b
 50 NEXT t: NEXT a
 60 DATA "a",135,195,227,255,25
5,17,35,71
 65 DATA "s",0,128,192,255,254,
192,128,0
 70 DATA "d",32,23,15,7,255,127
,63,31
 75 DATA "f",31,159,159,159,255
,255,255,255
 80 DATA "g",248,249,249,249,25
5,255,255,255
 85 DATA "h",4,232,240,224,255,
```

```
254,252,248
  90 DATA "j",0,0,0,1,1,7,7,7
  95 DATA "k",0,0,0,128,128,224,
224,224
 100 BORDER RND*6: PAPER RND*5+1
  INK 9: CLS
 105 PLOT 0,39: DRAW 255,0
 110 LET m=20: LET t=300: LET s=
0: LET a=1
 120 LET b=INT (RND*15)+1
 130 LET a=a+2: LET b=b+1
 140 IF RND>.7 THEN LET a=a+1
 150 IF a>26 THEN PRINT AT 5,a-3
;"    ": LET a=3
 160 IF b>26 THEN PRINT AT 15,b-
1;"    ";AT 16,b-1;"    ": LET
b=1
 165 REM  Avião construído com 'a' e 's' em modo gráfico.
 170 PRINT AT 5,a-3;"    "
 180 LET t=t-1: PRINT AT 0,0; PA
PER 2;"Tempo: =";t;"  "
 185 REM Navio: parte superior com 'j' e 'k', e parte inferior
 com 'd', 'f', 'g' e 'h' em modo gráfico
 190 PRINT AT 15,b-1; INK 0;"  "
";AT 16,b-1;"    "
 200 IF INKEY$="f" THEN GO TO 21
0
 205 LET t=t-1: GO TO 130
 210 LET m=m-1: PRINT AT 0,10; P
APER 2;"  Mísseis =  ";m;"  "
 220 IF ABS (a-b)>2 THEN GO TO 1
70
 230 IF m=0 OR t<1 THEN STOP
 240 FOR q=1 TO 7
 250 BORDER RND*6: BEEP .02,-q:
BEEP .02,q
 260 PRINT AT 15,b-1; INK 0;"
";AT 16,b-1;"    ";AT 15,b-1;
INK 7;"    ";AT 16,b-1;"    "
 270 NEXT q
 280 LET s=s+267
 290 PRINT AT 1,6; PAPER 2;"Pontuação
=";s;"  "
 300 IF INKEY$<>"" THEN GO TO 26
5
 310 GO TO 170
```

PROTECTOR

Neste programa, o jogador tem a tarefa de proteger uma parte defeituosa do campo de forças que protege a Terra. Certos inimigos já detectaram o ponto fraco das nossas defesas, e tentam penetrar no campo de forças.

O campo de forças pode suportar um ataque em qualquer sector, mas o sector em causa é enfraquecido ao sofrer o ataque. Um segundo ataque num sector já fraco provoca o colapso de todo o campo, deixando portanto a Terra sem qualquer defesa. O campo será também anulado quando tiver mais de nove pontos fracos.

A sua tarefa consiste portanto em contrariar quaisquer ataques com a sua nave. Pode restaurar qualquer parte enfraquecida que se encontre directamente por baixo de si carregando em "F". As teclas "5" e "8" controlam o seu movimento, no sentido das setas nelas impressas.

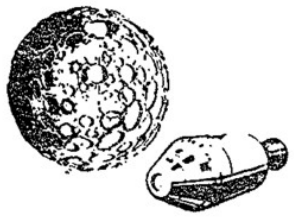
O jogo tem cinco níveis de habilidade, sendo o primeiro (1) o mais fácil. Como os primeiros níveis só servem de facto para prática, a pontuação aumenta bastante nos níveis mais elevados.

Julgo que o fracasso é inevitável, porque os inimigos renovam continuamente os seus ataques. A pontuação é indicada no final do jogo. Este programa foi escrito por Paul Toland.

```
  1 FOR a=1 TO 4: READ a$
  2 FOR t=0 TO 7: READ b
  3 POKE USR a$+t,b
  4 NEXT t: NEXT a
  5 DATA "a",255,0,255,0,255,0,
255,0
  6 DATA "s",0,0,0,0,255,0,255,
0
  7 DATA "d",0,0,0,0,60,60,60,6
0
  8 DATA "f",255,255,195,195,19
5,195,195,195
  9 REM Para obter o gráfico da linha 10 escreva 'a' em
 modo gráfico.
 10 CLS : LET d$="
                                                  "
 20 PRINT AT 19,3; INK 4;"≡";TA
B 27;"≡"
 30 PRINT AT 0,10; PAPER 2; INK
 9; FLASH 1;"Protector"
 40 PRINT AT 20,5; INK RND*6: F
LASH 1;" Indique nível, 1 a 5."
 50 IF INKEY$="" OR INKEY$<"0"
OR INKEY$>"5" THEN GO TO 50
 60 LET s=VAL INKEY$*2-1
 70 LET b=16: LET v=1: LET w=0
 80 FOR i=1 TO 20
 90 LET a=INT (RND*23)+4
100 FOR h=s TO 19 STEP v
105 REM Gráfico da linha 110 obtido com 'd' em modo
 gráfico.
```



```
110 PRINT AT 20,1; INK 4;d$;AT
h,a; INK RND*6;"■";AT 19,b;" "
120 LET b=b+(INKEY$="8")-(INKEY
$="5")
130 LET b=b+(b=3)-(b=27)
135 REM Gráfico da linha 140 obtido com 'f' em modo
 gráfico.
140 PRINT AT 19,b;"■"
150 IF INKEY$<>"f" OR d$(b)="≡"
THEN GO TO 170
160 LET w=w-1: LET d$(b)="≡": B
EEP .01,RND*20+10
170 PRINT AT h,a;" "
180 NEXT h
190 IF b=a THEN GO TO 220
195 REM Gráfico da linha 200 obtido com 's' em modo
 gráfico.
200 IF d$(a)="=" OR w=9 THEN GO
TO 240
210 LET w=w+1: LET d$(a)="=": B
EEP .0102,RND*20
220 NEXT i
230 LET v=v*2: GO TO 60
240 PRINT AT 5,1; INK RND*6;"
Fracassou!      ";: BEEP 1,RND*20
250 PRINT INK RND*6;"- O campo de
forças depois de atingida a pontuação      ";: BEEP
 2,RND*20: PRINT INK RND*6;((v-1
)*20+i)*s;"
260 PRINT INK RND*6; FLASH 1;"
Deseja proteger outro planeta?"
270 IF INKEY$="" THEN GO TO 270
280 IF INKEY$<>"n" THEN RUN
290 STOP
```



LETRAS *(Letter Chaser)*

O leitor dispõe de uma bola que deve movimentar no visor passando por cima das letras nele impressas por ordem alfabética. Indique a velocidade do jogo (1 — rápido; 2 — lento) e em seguida use as teclas de cursor para a deslocar. O jogo termina se atingir uma letra fora da sequência, indicando qual deveria ter sido a letra seguinte. A dificuldade do jogo vai aumentando. O programa foi escrito por Paul Toland.

```
  1 FOR t=0 TO 7: READ z
  2 POKE USR "a"+t,z: NEXT t
  3 DATA 0,60,126,126,126,126,6
0,0
 10 LET x=1: LET y=1: LET a=1
 20 LET d=0: LET nc=65
 30 LET p=RND*7: PAPER p: BORDE
R p: INK 9: CLS
 40 INPUT FLASH 1;" Indique velocidade
(1 RÁPIDO; 2 LENTO)"; S
 50 PRINT FLASH 1;"          LETT
ER-CHASER          "
 60 FOR t=1 TO 20
 70 PRINT FLASH 1;"■";AT t,31;"
■"
 80 NEXT t
 90 PRINT FLASH 1;"          LETT
ER-CHASER          "
110 FOR i=65 TO 90
120 LET rx=INT (RND*30)+1: LET
ry=INT (RND*19)+2
130 PRINT AT ry,rx;CHR$ i
140 NEXT i
150 LET i$=INKEY$
160 LET a=(i$="8")-(i$="5")+(i$
="")*a
170 LET d=(i$="6")-(i$="7")+(i$
="")*d
180 PRINT AT y,x;" "
190 LET x=x+a: LET y=y+d
195 LET x=x+(x=0)-(x=31): LET y
=y+(y=0)-(y=21)
200 LET n=CODE SCREEN$ (y,x)
205 REM  Para obter gráfico da linha 210 escreva 'a' em
modo gráfico
```



```
210 PRINT AT y,x;"●"
220 PAUSE 3*s
230 IF n=32 THEN GO TO 150
240 IF n<>nc THEN GO TO 320
250 LET nc=nc+1: BEEP .1,nc-65
260 IF nc<91 THEN GO TO 150
270 FOR i=1 TO 8
280 BEEP .1,i: BEEP .1,8-i
290 NEXT i
300 PRINT AT 10,10; FLASH 1;"
CONSEGUIU!    "
310 GO TO 330
320 PRINT AT 10,10; FLASH 1;CHR
$ (nc);" - POUCA SORTE! "
330 INPUT FLASH 1;"Outra tentativa?"
;a$
340 IF a$<>"n" THEN RUN
350 PAPER 7: BORDER 7
```

O OURO DO DRAGÃO

O objectivo deste jogo consiste em acumular tanto ouro quanto possível enquanto se desloca por um labirinto complexo formado por túneis, grutas e portas e tentando simultaneamente evitar o dragão e os poços nos túneis. Escreva "A" para avançar, "L" para se mover para a esquerda, "R" para se mover para a direita. Se encontrar o dragão ou cair num dos poços, o computador tocará uma pequena melodia antes de terminar o jogo. O programa foi escrito por D. C. Owem.

```
  1 REM  O ouro do Dragão
  2 REM      Por D. C. Owen
  3 RANDOMIZE : LET g=0
 10 BORDER RND*6: INK RND*5: CL
S
 15 POKE 23692,-1
 20 PRINT FLASH 1;'' AT 21,8;" O
OURO DO DRAGÃO ";AT 21,8; OVER 1;"___
_____________"
 30 PRINT '' '' 'Você tem:    : -"'g;" barras
de ouro"
```

```
 40 PRINT '" à sua frente está"; GO SUB 1000
 60 LET b$=a$
 70 PRINT ' à sua esquerda está"; GO SUB 1000
 80 LET l$=a$
 85 PRINT '"e à sua direita está "; : GO SUB
1000

 86 LET r$=a$
 90 PRINT '' FLASH 1; "Por onde deseja ir?".

100 PRINT ' ' FLASH 1;" — A —em frente, L
— esquerda, R — direita"
110 INPUT k$

120 IF k$="a" AND b$="D" OR k$=
"r" AND r$="D" THEN GO SUB 2000
130 IF k$="a" AND b$="T" OR k$=
"l" AND l$="T" OR k$="r" AND r$=
"T" THEN GO SUB 3000
140 IF k$="a" AND b$="C" OR k$=
"l" AND l$="C" OR k$="r" AND r$=
"C" THEN GO SUB 4000
150 IF NOT (k$="l" OR k$="r" OR
k$="a") THEN GO TO 90
160 GO TO 15
999 BORDER 7: PAPER 7
1000 RESTORE
1010 FOR t=1 TO INT (RND*3)+1
1020 READ a$
1030 NEXT t
1040 PRINT INK RND*6; a$: LET a$=
a$(2)
1050 DATA " Porta "," Túnel "," Gruta"

1060 RETURN
2000 REM ** PORTA **
2010 GO TO 2000+INT (RND*4+1)*10
0
2100 PRINT '" Está fechada. Continua." RETURN"

2200 LET q=INT (RND*9+1)*100
2210 PRINT "Abrirá. Existem";q;" barras de ouro aqui."
2220 LET g=g+q: RETURN
2300 PRINT "Existe aqui um lago. Não consegue ver
a margem oposta."

2310 PRINT "Vai tentar atravessá-lo?"
```



```
2320 INPUT c$
2325 CLS
2330 IF c$(1)<>"y" THEN RETURN
2340 LET k=INT (RND*3)+1
2350 IF k=2 THEN PRINT "Fugiu com";g;"
 barras de ouro":GO TO 2370

2360 PRINT AT 10,4; FLASH 1;
"Infelizmente, você"
2370 PRINT AT 11,9; INK 2; FLASH
 1;"... AFOGOU-SE!!! ": GO TO 5000
2400 LET k=INT (RND*9)+1*50
2410 PRINT '" Esta sala contém um
 "; INK 2; FLASH 1;" DRAGÃO.."
2420 PRINT "Exige": k :"barras de ouro""ou comê-lo-á!"
2425 PAUSE 200: POKE 23692,-1
2430 FOR j=1 TO 20
2435 BEEP .2,j
2440 PRINT INK RND*6;TAB j;
"PREPARE-SE"
2450 NEXT j
2460 IF g>k-1 THEN PRINT " Já tem o
suficiente. ": LET g=g-k: PAUSE 10
0: RETURN
2470 PRINT FLASH 1; INK 2;"...
...MAS não tem o suficiente... ADEUS!": GO TO 5000
3000 REM ** TUNEL **
3010 IF RND>.85 THEN RETURN
3020 PRINT'—"Fugiu com";g;" barras de ouro"
:STOP
4000 REM ** GRUTA **
4010 GO TO 4000+INT (RND*3+1)*10
0
4020 PRINT "A gruta está vazia. Continue.": RETURN
ove on.
4200 LET q=INT (RND*10+1)*100
4210 PRINT ' "Encontra"; q :"barras de ouro"

4220 PRINT "para acrescentar às que tem"
4230 LET g=g+q: RETURN
4300 IF RND>.9 THEN GO TO 4400
4305 POKE 23692,-1
4310 FOR h=1 TO 24: PRINT : NEXT
 h
4320 PRINT INK 2; FLASH 1;"OH, NÃO!"

4330 FOR j=1 TO 15
4335 BEEP .2,j
```

```
4340 PRINT TAB 2*j; INK RND*5;"■
■"
4350 NEXT j
4360 PRINT INK 2; FLASH 1;
"É um poço..."""...Você morreu."  GO TO 5000
4400 PRINT    "Há ruídos à frente"
4410 PRINT    " Quer investigar? "

4420 INPUT k$
4430 IF k$(1)<> "s" THEN RETURN
4440 GO TO 4000
5000 RESTORE 5020
5005 FOR a=1 TO 10
5010 READ b,c: BEEP b,c
5015 NEXT a
5020 DATA 1,0,1,0,.5,2,.5,0,.5,3
,.5,0,.5,-1,.5,0,.5,1,1,0
```

PACIÊNCIA

Este programa, de Ken Mahogany, permite-lhe uma rede de 4 x 4 posições que contém letras do alfabeto. É necessário dispô-las por ordem alfabética, isto é:

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

deixando um espaço no canto inferior direito. O movimento é executado indicando um número (é indicado um código no visor) correspondente à letra que se quer deslocar, seguido do quadrado para onde deve ser movida. O programa conta o número de movimentos. O leitor deve conseguir resolver a paciência em cerca de 40 movimentos. Se deseja alterar a ordem das letras no início do jogo, modifique os valores da instrução DATA da linha 420.


```
  10 REM  Paciência
  20 REM © K Mahogany, 1982
  30 GO SUB 330
  40 GO SUB 200
  50 GO SUB 200
  90 INPUT INK 7; "Qual deseja mover?"; x
 110 IF a(x)=32 THEN GO TO 90
 130 INPUT INK 7;TAB 8;  " Para onde
?";y
 140 IF a(y)<>32 THEN GO TO 130
 150 LET a(y)=a(x)
 160 LET a(x)=32
 170 LET go=go+1
 180 GO TO 50
 200 REM ***     Impressão     ***
 210 PRINT AT 0,3; PAPER 7; INK
3;"  Movimento:  "; INK 2;go
 220 PRINT '' INK RND*4;CHR$ a(1
);CHR$ a(2);CHR$ a(3);CHR$ a(4),
"   1   2   3   4"
 230 PRINT INK RND*4;CHR$ a(5);C
HR$ a(6);CHR$ a(7);CHR$ a(8),"
5   6   7   8"
 240 PRINT INK RND*4;CHR$ a(9);C
HR$ a(10);CHR$ a(11);CHR$ a(12),
"   9  10  11  12"
 250 PRINT INK RND*4;CHR$ a(13);
CHR$ a(14);CHR$ a(15);CHR$ a(16)
,"  13  14  15  16"
 320 RETURN
 330 REM ***    Iniciação     ***
 340 DIM a(16)
 350 FOR b=1 TO 16
 360 READ m
 370 LET a(b)=m+64
 380 NEXT b
 390 LET go=1
 400 PAPER 6: BORDER 2
 405 CLS
 410 RETURN
 420 DATA 9,14,5,2,11,6,1,4,12,7
,-32,10,13,8,3,15
```

HANGMAN

Trata-se de um jogo muito simples em que o computador escolhe uma palavra e em seguida nos permite realizar algumas tentativas (dependendo do comprimento da palavra) para a adivinharmos. O vocabulário pode ser facilmente ampliado. Este programa foi escrito por Ken Mahogany.

```
 10 REM Spectral Hangman
 20 REM © K Mahogany, 1982
 30 FOR g=1 TO RND*25+1
 40 READ a$
 50 NEXT g
 60 LET n=LEN a$
 70 DIM b(n): DIM d(n)
 80 FOR g=1 TO n
 90 LET b(g)=CODE a$(g)
100 LET d(g)=b(g)
110 NEXT g
120 FOR j=1 TO n+n/3
140 GO SUB 410
150 IF h=n THEN GO TO 300
190 PRINT '''' INK RND*6;" Indique
a sua tentativa ";j
200 INPUT c$
210 LET f=CODE c$
220 FOR g=1 TO n
230 IF d(g)=f THEN LET d(g)=0
240 NEXT g
250 NEXT j
265 GO SUB 410
270 PRINT '' INK 3; "É pena, acabou
o tempo!"
280 GO TO 330
300 REM ** Ganhar **
310 PRINT ' INK 4;TAB 4; "Conseguiu!"
320 PRINT ' INK 4;TAB 4; "Adivinhou
em";j-1" tentativas"
330 PRINT ' INK 2; "A palavra era
";a$
335 PRINT '' INK RND*5; "Carregue em
qualquer tecla para novo jogo"
340 PAUSE 4e4
350 RUN
360 DATA "aspecto","spectrum","cambrige","
"acaso","abóbora"
370 DATA " "pergunta","adivinha","enfrentar"
,"tios","gravador"
380 DATA "basic","formula","amistoso","recursos
","melhor"
390 DATA "manteiga","morango","átomo"
,"feiticeiro","enfrentar"
400 DATA "sabio","maldoso","imaginativo"
,"desejar","inimigo"
410 LET h=0
415 FOR e=1 TO n
420 IF b(e)=d(e) THEN PRINT INK
RND*6;"_";
430 IF b(e)<>d(e) THEN PRINT IN
K RND*6;CHR$ (b(e));: LET h=h+1:
BEEP .25,h
440 NEXT e
450 BORDER RND*6
460 PRINT ' INK 0; PAPER 6;"
ADIVINHOU "; INK RND*5;h; IN
K 0;" letras ";
470 IF h<>1 THEN PRINT PAPER 6;
"s"
480 PRINT
490 RETURN
```


REACÇÃO

Este programa serve para testar os seus reflexos. O computador imprime uma letra ou número numa posição aleatória. O leitor deve carregar imediatamente na tecla correspondente. Começa com cinco pontos e ganha-os se escrever correctamente a letra, ou perde-os se for muito lento ou carregar numa tecla errada. Pode parecer fácil, particularmente na medida em que existe um comando de velocidade, mas o computador não apaga as letras já impressas, pelo que você é forçado a recordar as que foram impressas anteriormente. O jogo pára quando você perde todos os pontos.

```
  1 REM Reacção por D. Harwood
  2 LET t=5
 10 PRINT TAB 10; INK RND*5; FL
ASH 1;" REACÇÃO "
```

```
  20 PRINT    "Vou imprimir um caracter no visor e você
deve carregar imediatamente na  tecla correspondente"
  30 PRINT   "Se não acertar dou-lhe outra oportunidade,
mas de início só terá 5 pontos"
  40 PRINT  '  "Perde 2 pontos quando não é suficiente
mente rápido"

  45 INPUT  "Velocidade (1-10) ?";t
  50 PRINT  '' INK 2; FLASH 1;"
Espero que esteja pronto"
  60 PAUSE 200
  70 CLS : POKE 23562,-1
  75 PRINT AT 0,0; INK 1; FLASH
1;"REACÇÃO REACÇÃO REACÇÃO"

  76 PRINT AT 2,0; INK 2; FLASH
1;" Pontuação ";l
  77 IF l<1 THEN GO TO 230
  80 LET a=INT (RND*26)+96
  90 IF RND>.7 THEN LET a=INT (R
ND*9)+48
 100 LET x=RND*16+3: LET y=RND*3
1
 105 BEEP .1,0: PRINT AT x,y; IN
K RND*5; FLASH 1;CHR$ a
 110 PAUSE t*7
 120 LET a$=INKEY$
 125 IF a$="" THEN GO TO 210
 130 IF CODE a$=a THEN GO TO 170
 140 PRINT AT 0,0; INK 2; FLASH
1;" Errado, perdeu um ponto"

 150 PAUSE RND*200
 160 LET l=l-1: GO TO 75
 170 PRINT AT 0,0; INK 2; FLASH
1;"                ACERTOU "

 180 LET l=l+1
 190 FOR t=1 TO 10: BEEP .1,t: N
EXT t
 200 GO TO 75
 210 PRINT AT 0,0; INK 3; FLASH
1;"              MUITO LENTO "

 215 PAUSE RND*200
 220 LET l=l-2: GO TO 75
 230 CLS : PRINT AT 10,8; INK RN
D*5; FLASH 1;"    Perdeu todos os pontos...
Adeus  "
 240 BEEP 1,-2: BEEP 2,4
```


O programa que se segue transforma o seu Spectrum num William Shakespeare . . . ou quase. Escolhendo aleatoriamente palavras nas instruções DATA, e espaçando-as usando as linhas 40 a 60, o programa consegue juntar frases de uma forma surpreendentemente boa. As frases são impressas no visor em diferentes cores, sendo algumas realçadas usando a instrução BRIGHT. O programa foi escrito por Tim Hartnell.

```
  5 RANDOMIZE
 10 REM    Poesia
 20 POKE 23692,-1
 30 IF RND>.7 THEN GO TO 70
 40 FOR j=1 TO RND*3
 50 PRINT
 60 NEXT j
 70 LET a$="   "
 75 RESTORE
 80 FOR a=1 TO RND*12
 90 READ b$
100 NEXT a
110 LET x=LEN a$: LET y=LEN b$
120 IF a$(x-1)=b$(y-1) THEN GO
TO 75
130 IF x+y>=32 THEN GO TO 160
140 LET a$=a$+b$
150 GO TO 75
160 PRINT INK RND*5; BRIGHT (RN
D>.8);a$
170 GO TO 10
200 DATA "livre","iniciar","cedo","apesar de","..."
210 DATA " "discípulo","chorando","apenas",
"tentou alcançar"
220 DATA     "sózinha","ganha por","então"
```

BOLAS E CRUZES

Julgo que não há necessidade de explicar o modo de jogar este jogo.

O leitor e o computador realizam os seus movimentos alternadamente, tentando colocar três Os ou três Xs em linha. Ao contrário do que acontece na maior parte das versões computadorizadas deste jogo, este permite-lhe ganhar de vez em quando! Em muitos casos o melhor que se consegue é um empate...

Os movimentos são realizados indicando o número do quadrado para onde nos queremos mover. O leitor é o primeiro a jogar, mas o computador ignorá-lo-á se não escrever 5 (o quadrado central) como seu primeiro movimento.

Esta versão do jogo é capaz de suportar cerca de 40 000 desenvolvimentos, cerca de um oitavo de todas as combinações possíveis.

```
  1 REM Bolas e cruzes
  2 REM Adaptado por I. Hartnell e D. Harwood
  3 REM de um programa ZX80
  4 REM Por Stuart Roberts
 10 DIM b(9): DIM p(9)
 20 FOR a=1 TO 9
 30 LET b(a)=a
 40 NEXT a
 50 LET e=0: LET q=0
 60 LET n=0
 70 LET x=4
 80 PRINT AT 5,0;
 90 FOR a=1 TO 9
100 IF a=x THEN LET x=x+3: PRIN
T ";"
110 IF b(a)=0 THEN GO TO 320
120 IF b(a)=10 THEN GO TO 350
130 PRINT INK RND*6;b(a);" ";
140 NEXT a
150 IF n=1 THEN GO TO 380
160 IF e=8 THEN PRINT "" INK 1;
FLASH 1;" GANHEI ": GO TO 730
170 IF q=5 THEN PRINT "" INK 1;
FLASH 1;" É UM EMPATE ": GO TO
730
180 INPUT " Movimento? ";z
190 IF b(z)<>z THEN GO TO 180
200 LET q=q+1
210 IF z=11 THEN LET b(5)=10: G
O TO 60
220 LET n=1: LET b(z)=0
230 GO TO 70
240 FOR a=c TO d
250 IF b(a)=a THEN LET b(a)=10
260 LET a=a+f
270 NEXT a
280 GO TO 60
290 IF b(5)=5 THEN GO TO 20
300 LET b(d)=10
310 GO TO 60
320 PRINT INK 1; FLASH 1;"O"; F
LASH 0;" ";
330 LET p(a)=1
340 GO TO 140
350 PRINT INK 2; FLASH 1;"X"; F
LASH 0;" ";
360 LET p(a)=4
370 GO TO 140
380 LET g=0
390 LET c=1: LET d=9: LET f=3
400 GO SUB 580
410 LET c=3: LET d=7: LET f=1
420 GO SUB 580
430 LET d=9: LET f=2
440 GO SUB 580
450 LET c=2: LET d=8
460 GO SUB 580
470 LET c=1: LET d=7
480 GO SUB 580
490 LET d=3: LET f=0
500 GO SUB 580
510 LET c=4: LET d=6
520 GO SUB 580
530 LET c=7: LET d=9
540 GO SUB 580
550 LET g=g+1
560 IF g=5 THEN GO TO 60
570 GO TO 390
580 LET e=0
590 FOR a=c TO d
600 LET e=e+p(a): LET a=a+f
610 NEXT a
620 IF e=3 THEN GO TO 720
```

```
630 IF g=0 THEN RETURN
640 IF e=3 THEN GO TO 240
650 IF g=1 THEN RETURN
660 IF e=2 THEN GO TO 240
670 IF g=2 THEN RETURN
680 IF e=5 THEN GO TO 240
690 IF g=3 THEN RETURN
700 IF e=1 OR e=4 THEN GO TO 29
0
710 RETURN
720 FOR a=1 TO 10. BEEP .1,a: B
EEP .1,10-a: NEXT a
725 PRINT INK 2; FLASH 1;" GANHOU"
730 BEEP .1,0: BEEP .2,0
735 INPUT FLASH 1;"   Deseja outro jogo?"
740 IF k$(1)<>"n" THEN CLS : RU
N
750 PRINT ''   "OK. Obrigado pelo jogo."
```

```
1 2 3        1 2 3        0 2 3
4 5 6        4 0 6        4 0 6
7 8 9        7 8 X        X 8 X

      0 X 3        0 X 3
      4 0 6        0 0 X
      X 0 X        X 0 X
```


Este programa simula uma oitava de um teclado de piano, usando gráficos que representam as teclas deste. Faça executar a linha 3 antes de escrever o resto do programa a fim de obter imediatamente os caracteres gráficos, depois obtidos escrevendo 'P' e 'I' em modo gráfico.

As teclas "Q" a "P" representam as teclas brancas do piano, e as teclas "2", "3", "5", "6", "7", "9" e "0" representam as teclas negras. Se carregar em alguma destas teclas obterá um som. Altere a duração da nota modificando o valor .3 da instrução BEEP da linha 105.

```
  1 REM     Música, por D. Harwood
  2 REM     Execute linhas 3-4 antes de escrever
o resto do programa
  3 FOR n=0 TO 7: POKE USR "p"+
n,16: POKE USR "i"+n,252: NEXT n
  4 LET p$="": LET n$="": LET p
=0: LET x=0
  5 PAPER 0: INK 7: BORDER 0: C
LS
  7 PRINT
 10 FOR m=1 TO 5
 15 REM  Para obter escreva 'p' em modo gráfico e para
escreva 'i' em modo grafico.
 20 PRINT TAB 10;"|■■|■■■|■■"
 30 NEXT m
 40 FOR m=1 TO 4
 50 PRINT TAB 10;"|||||||||||"
 60 NEXT m
 70 LET k$=INKEY$
 80 IF k$="" THEN GO TO 70
 85 IF p$=k$ THEN GO TO 100
 90 GO SUB 1000
100 PRINT OVER 1;AT x,9+p;n$
105 BEEP .3,a
110 LET p$=k$
115 PRINT OVER 0;AT 0,10;"
                ";AT 8,10;"|||||||
|||"
120 GO TO 70
```

```
1000 IF k$="1" THEN LET a=1: LET
 p=2
1005 IF k$="2" THEN LET a=3: LET
 p=3
1010 IF k$="4" THEN LET a=6: LET
 p=5
1015 IF k$="5" THEN LET a=8: LET
 p=6
1020 IF k$="6" THEN LET a=10: LE
T p=7
1025 IF k$="8" THEN LET a=13: LE
T p=9
1030 IF k$="9" THEN LET a=15: LE
T p=10
1035 IF k$="q" THEN LET a=-1: LE
T x=8: LET p=1
1040 IF k$="w" THEN LET a=0: LET
 p=2
1045 IF k$="e" THEN LET a=2: LET
 p=3
1050 IF k$="r" THEN LET a=4: LET
 p=4
1055 IF k$="t" THEN LET a=5: LET
 p=5
1060 IF k$="y" THEN LET a=7: LET
 p=6
1065 IF k$="u" THEN LET a=9: LET
 p=7
1070 IF k$="i" THEN LET a=11: LE
T p=8
1075 IF k$="o" THEN LET a=12: LE
T p=9
1080 IF k$="p" THEN LET a=14: LE
T p=10
1090 LET x=9: LET n$="▮"
1100 IF k$="1" OR k$="2" OR k$="
4" OR k$="5" OR k$="6" OR k$="8"
 OR k$="9" THEN LET x=0: LET n$=
"▮"
1110 RETURN
```



BREAKOUT

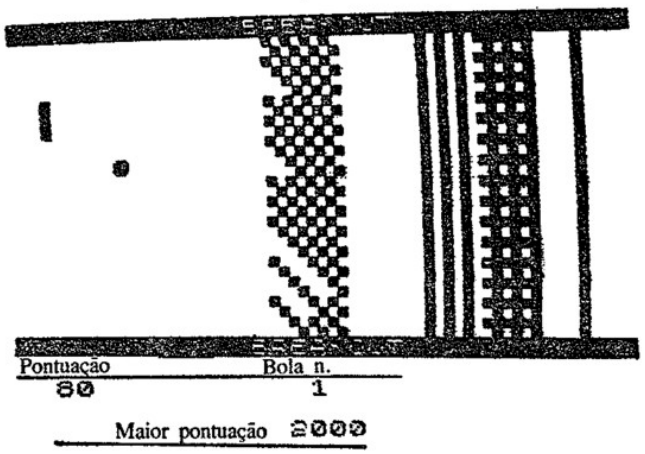
Nesta versão de BREAKOUT, controla-se a raquete no lado esquerdo do visor usando as teclas '6' e '7', tentando manter a bola em jogo. O objectivo consiste em demolir a maior parte possível das três paredes coloridas. As paredes mais afastadas produzem maiores pontuações. Pode utilizar 5 bolas em cada jogo. Necessitará de escrever LET H = 0 antes de executar o programa para definir o estado do contador, fazendo executar o programa com GO TO 5 em vez de RUN.

Se gravar o programa em cassete depois de o ter jogado, reinicie o jogo com GO TO 5; a maior pontuação anterior estará assim ainda à sua disposição. Os gráficos, os sons e as cores usados neste jogo são bastante interessantes.

```
  1 REM      Execute linhas 2 a 4 antes de escrever o
resto do programa
  2 FOR t=0 TO 7: READ z
  3 POKE USR "a"+t,z: NEXT t
  4 DATA 0,60,126,126,126,126,6
0,0
  5 RANDOMIZE
 10 LET t=0: LET z=0
 20 LET tb=0: LET r=8
 30 BORDER RND*7: INK 9: CLS
 40 PRINT AT 1,0; INVERSE 1;"
        BREAKOUT                "
 50 PRINT AT 17,0; INVERSE 1;"
        BREAKOUT                "
 60 PRINT AT 18,0;  "Pontuação
— Bola n.  ";AT 18,0; OVER 1;"_____
_____"

 70 PRINT AT 21,0;"    Maior pontuação "
  ;h;AT 21,0; OVER 1;"_____
_____"
 80 FOR i=2 TO 16
 90 PRINT AT i,13; INK 3;"▮▮▮▮";
 INK 0;"    "; INK 4;"▮▮▮"; INK
5;"▮▮▮"; INK 0;"   "; INK 6;"▮"
100 NEXT i
110 LET a=-1: LET d=1: LET x=11
```

```
  LET y=INT (RND*10)+5
 120 LET tb=tb+1
 130 IF tb=6 THEN GO TO 310
 140 PRINT AT 19,15;tb
 150 PRINT AT r,1;" ";AT r+1,1;"
"
 160 LET r=r+(INKEY$="6")-(INKEY
$="7")
 165 IF r=1 THEN LET r=r+1
 167 IF r=16 THEN LET r=r-1
 170 PRINT AT r,1; INK 1;"▐";AT
r+1,1;"▐"
 190 LET x=x+a: LET y=y+d
 200 LET n=ATTR (y,x)
 202 REM Para obter o gráfico da linha 205 escrevá 'a' em
 modo gráfico
 205 PRINT AT y,x; INK 2;"●"
 210 IF y=2 OR y=16 THEN LET d=-
d
 220 IF n=57 OR x=30 THEN LET a=
-a: BEEP .01,20+RND*10
 230 PRINT AT r,1;" ";AT r+1,1;"
"
 240 LET r=r+(INKEY$="6")-(INKEY
$="7")
 245 IF r=1 THEN LET r=r+1
 247 IF r=16 THEN LET r=r-1
 250 PRINT AT r,1; INK 1;"▐";AT
r+1,1;"▐"
 260 IF n<56 THEN GO TO 300
 270 LET t=t+(n-57)*5
 275 BEEP .01,20+RND*10
 280 PRINT AT 19,2;t
 290 LET a=-a
 300 PRINT AT y,x;" ": GO TO (x>
0)*20+110
 320 PRINT AT 10,3; FLASH 1;
"FIM DO JOGO"
 330 IF t>h THEN LET h=t
 340 PRINT '"      Maior pontuação actual
;h
 350 INPUT "    Outro jogo?    ";a$
 360 IF a$="n" THEN STOP
 370 CLS : GO TO 10
```

ZAP

O leitor terá de evitar que os ubíquos extraterrestres aterrem no nosso planeta (o que parece ser a principal ocupação da humanidade...). Podem descer segundo três direcções diferentes: verticalmente ou diagonalmente para a direita ou para a esquerda.

Deve colocar a sua base usando as teclas '5' e '8', disparando o seu míssil usando 'F' e tentando intersectar a trajectória da nave inimiga.

Não conseguira qualquer resultado bloqueando o inimigo com a sua própria nave. No início do jogo deverá indicar um nível de dificuldade (0 a 5), sendo 0 o mais fácil. O programa foi escrito por Paul Toland.

```
    1 REM      Execute as linhas 2-6 antes de escrever o
resto do programa
    2 FOR a=1 TO 2: READ a$: FOR
t=0 TO 7
    3 READ b: POKE USR a$+t,b
    4 NEXT t: NEXT a
    5 DATA "a",0,0,0,24,60,255,25
5,255
    6 DATA "s",0,24,60,90,255,90,
129,129,0
    7 LET c$="": LET b$="
```



```
    8 FOR z=1 TO 32
    9 LET c$=c$+CHR$ 16+CHR$ (RND
*6)+b$(z): NEXT z
   10 LET t=0
   20 INPUT FLASH 1; "Indique nível (0-5): "
   30 LET bp=999: LET g=16
   35 LET b=-1
   40 CLS
   50 PRINT AT 19,0;c$
   60 LET t=t+1: LET r=INT (RND*3
)+1
   70 PRINT AT 0,0;t
   80 IF r=3 THEN LET p=INT (RND*
12)
   90 IF r=2 THEN LET p=INT (RND*
16)*2
  100 IF r=1 THEN LET p=INT (RND*
12)+20
  110 LET i=r-2
  120 FOR j=1 TO 19
  125 REM  Para obter o gráfico da linha 130 escreva 's'
em modo gráfico
  130 PRINT AT j,p; INK 1;"A"
  140 LET a$=INKEY$
  150 PRINT AT 18,g;" "
  160 LET g=g+(a$="8")*2-(a$="5")
*2
  165 REM  para obter o gráfico da linha 170 escreva 'a'
em modo gráfico
  170 PRINT AT 18,g; INK 2;"A"
  180 IF a$="f" AND b=-1 THEN BEE
P .01,0: LET b=17
  190 IF b=17 THEN LET bp=g
  200 IF b>-1 THEN PRINT AT b,bp;
" ": LET b=b-1: PRINT AT b,bp; I
NK 3;"."
```



```
  205 FOR z=5 TO 1*3 STEP -1: NEX
T z
  210 IF (b=j OR b=j+1) AND bp=p
THEN GO TO 35
  220 PRINT AT j,p;" "
  230 LET p=p+i
  240 NEXT j
  250 PRINT AT j-1,p;" "; INK 1;"
A"; INK 0;" "
  260 LET d$="    Os inimigos aterraram "
  270 FOR z=1 TO 22
  280 BEEP .1,z
  290 PRINT AT 21,z-1;d$(z)
  300 NEXT z
  310 PAUSE 10: BEEP .1,0: PAUSE
10: BEEP .2,0
  320 PRINT AT 0,0; FLASH 1; INK
   "Carregue em 's' para novo jogo "
  330 PAUSE 4e4
  340 IF INKEY$="s" THEN CLS : GO
TO 7
```

CENTROPÓIDE

Você, representado por uma marca azul, viaja pelo visor da sua televisão passando por todos os asteriscos e evitando os quadrados vermelhos. Deve passar por todos os asteriscos antes de terminar o tempo, que lhe será indicado no final no caso de ter êxito. Este programa foi escrito por Paul Toland.

```
   10 RANDOMIZE
   20 FOR i=0 TO 31
   30 PRINT AT 0,i; INK 2;"■";AT
21,i;"■"
   40 NEXT i
   50 FOR i=0 TO 21
   60 PRINT AT i,0; INK 2;"■";AT
i,31;"■"
   70 PRINT AT RND*19+1,RND*29+1;
 INK 2;"■"
   90 NEXT i
  100 FOR i=1 TO 10
```

```
 110 PRINT AT RND*19+1,RND*29+1;
 INK 3;"*"
 120 NEXT i
 130 LET t=0: LET h=0: LET x=3
 140 LET y=3: LET a=0: LET d=1
 150 LET a$=INKEY$
 160 IF a$="5" OR a$="8" THEN LE
T d=0: LET a=SGN (VAL a$-6)
 170 IF a$="6" OR a$="7" THEN LE
T a=0: LET d=SGN (VAL a$-6.5)*-1
 180 LET x=x+a: LET y=y+d
 190 LET n=ATTR (y,x)
 200 PRINT AT y,x; INK 1;"■"
 205 BEEP .2,x
 210 IF n=59 THEN LET h=h+1
 220 IF h=10 THEN GO TO 250
 230 IF n=57 OR n=58 THEN GO TO
260
 240 LET t=t+1: GO TO 150
 250 PRINT AT 10,8; INK 5; FLASH
 1;"   Terminado em    ";t;"  segs. "
 260 INPUT "   Outro jogo?";a$
 270 IF a$="n" THEN STOP
 280 CLS : RUN
```

PERÍCIA

Conduza a sua cobra, em crescimento contínuo, usando as teclas '5', '6', '7' e '8' de tal modo que evite a margem, os pontos por onde já passou e os sinais '+'. O objectivo do jogo consiste em durar tanto tempo quanto possível, sendo o tempo indicado no final do jogo. Este programa foi escrito por Paul Toland.

```
   1 REM      Execute as linhas 2 a 6 antes de escrever o
resto do programa
   2 FOR x=1 TO 2: READ a$
   3 FOR t=0 TO 7: READ a
   4 POKE USR a$+t,a: NEXT t: NE
XT x
   5 DATA "p",0,8,8,8,255,255,12
8,60
   6 DATA "o",170,85,170,85,170,
85,170,85
```



```
   7 LET h=1
   8 POKE 23692,-1
   9 LET s=0: LET u=15: LET p=10
  10 PRINT AT u-1,p;"  "
  20 LET p=p-(p>1)
  30 LET p=p+(INKEY$<>"")*2*(p<2
0)
  40 LET n=ATTR (u,p)
  50 IF n=58 THEN LET u=u-1: BEE
P .1,u
  55 IF u=0 THEN GO TO 100
  58 REM     Para obter o gráfico da linha 60 escreva 'p'
em modo gráfico
  60 PRINT AT u,p;"■"
  70 PRINT AT 21,0;
  75 REM     Para obter o gráfico da linha 80 escreva 'o'
em modo gráfico
  80 PRINT TAB RND*20; INK 2;"▒▒
▒▒"
  83 PRINT
  85 FOR j=1 TO 10: NEXT j
  90 LET s=s+u: GO TO 10
 100 IF h<s THEN LET h=s
 110 PRINT AT 0,0;; INK 1; FLASH
 1;;"SURGE  :";s'"  A sua pontuação é ";s'"
  Maior pontuação    ";h
 120 INPUT          " Outro jogo?    ?";a$
 130 IF a$="y" THEN CLS : GO TO
8
```

SURGE

Neste jogo, escrito por Tim Rogers, a sua nave encontra-se perto de um estranho cinturão de asteróides de forma achatada. A sua nave possui um escudo protector, o que significa que não pode ser atingida pelos asteróides. O único problema é que a nave é empurrada para cima sempre que entra em contacto com qualquer dos asteróides. O objectivo do jogo consiste em manter-se no visor durante tanto tempo quanto possível. Quanto mais baixa for a sua posição maior é a pontuação obtida.

```
  10 RANDOMIZE
  20 FOR i=0 TO 31
  30 PRINT AT 0,i; INK 2;"■";AT
21,i;"■"
  40 NEXT i
  50 FOR i=0 TO 21
  60 PRINT AT i,0; INK 2;"■";AT
i,31;"■"
  70 PRINT AT RND*19+1,RND*29+1;
 INK 3;"+"
  90 NEXT i
 100 LET t=0: LET h=0: LET d=1
 110 LET x=2: LET a=0: LET y=5
 120 LET a$=INKEY$
 130 IF a$="5" OR a$="8" THEN LE
T d=0: LET a=SGN (VAL a$-6)
 140 IF a$="6" OR a$="7" THEN LE
T a=0: LET d=SGN (VAL a$-6.5)*-1
 150 IF a$="7" THEN LET d=-1
 160 LET x=x+a: LET y=y+d
 170 LET n=ATTR (y,x)
 175 PRINT AT y,x; INK 1;"■"
 177 BEEP .2,x
 180 IF n=59 THEN LET h=h+1
 190 IF n=57 OR n=58 OR h=6 THEN
 GO TO 210
 200 LET t=t+1: GO TO 120
 210 PRINT AT y,x;"*"
 220 PRINT AT 10,6; INK 4; FLASH
 1;"  Sobreviveu      ";t;"  segs.
 "
 230 INPUT     Outro jogo?   ";a$
 240 IF a$="n" THEN STOP
 250 CLS : RUN
```

CAMPO DE MINAS

Você comanda um grupo de tanques neste jogo escrito por I. S. Howson. À sua frente encontra-se em campo de minas inimigo através do qual deve passar. Quantos tanques perderá ao tentar descobrir um caminho seguro?

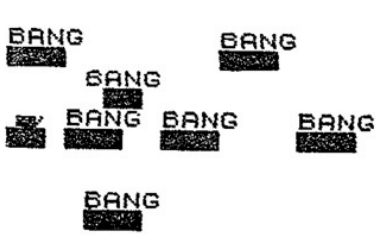
Ao iniciar o programa, a máquina produzirá um campo de minas aleatório — diferente de cada vez que jogar. Conduza o seu tanque pelo visor, usando as teclas '6', '7' e '8'. Se atingir uma mina, o tanque explode e você deve tentar de novo com outro tanque. As minas são múltiplas, destruindo todos os tanques que as atingirem. Se tiver êxito na descoberta do caminho seguro, o programa pára, tocando uma pequena melodia.

Verificará que consegue melhorar bastante os seus resultados à medida que o jogo progride.

```
  1 REM      Execute as linhas 2 a 5 antes de escrever o
resto do programa.
  2 FOR t=0 TO 7: READ y,z
  3 POKE USR "a"+t,y: POKE USR
"s"+t,z: NEXT t
  4 DATA 0,0,0,0,0,0,0,0
  5 DATA 15,242,15,244,15,248,1
5,240
 10 LET s=0
 20 RANDOMIZE
 30 LET b=0: LET a=10
 40 PRINT AT 0,0; INK 2; FLASH
1;" Tanques destruídos:  ";s
 60 PRINT AT a-1,b;"  "
 65 REM  Para obter o gráfico da linha 70 escreva 'a' e 's'
em modo gráfico
 70 PRINT AT a,b; INK 1;"  ▰"
 80 PRINT AT a+1,b; INK 1;"▰■"
 90 PRINT AT a+2,b;"  "
100 LET b=b+(INKEY$="8")
110 LET a=a+(INKEY$="6")-(INKEY
$="7")
120 IF b=INT (RND*30) OR a=INT
(RND*20) THEN GO TO 150
130 IF b=28 THEN GO TO 210
140 GO TO 60
150 PRINT AT a,b; INK RND*6; FL
ASH 1;"BANG"
155 FOR t=-20 TO -10: BEEP .01,
t
157 BORDER RND*7
160 PRINT AT a+1,b; INK 7;"■■■"
;AT a+1,b;"■■■"
165 BEEP .01,10+t
170 PRINT AT a+1,b; INK 0;"■■■"
;AT a+1,b;"■■■"
175 NEXT t
180 LET s=s+1
190 PRINT AT 1,8; INVERSE 1;"TA
'KS"
```

```
  200 IF s>19 THEN STOP
  205 GO TO 20
  210 PRINT AT 10,0; INK 2;
"SOBREVIVEU!"
  220 BEEP 2,20: BEEP 1.5,15: BEE
P 1,10: BEEP .5,5
```

Tanques destruídos: 12



BILLBOARD

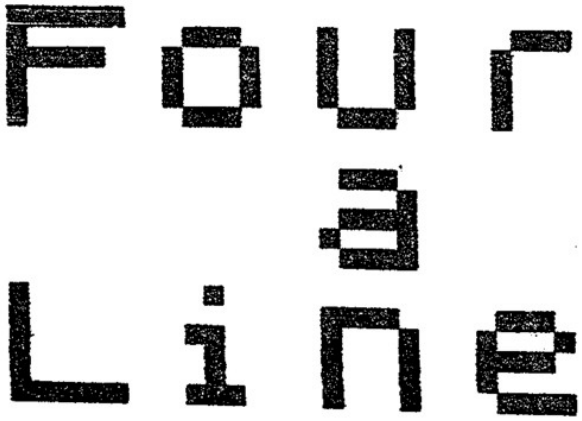
Este programa permite-nos obter uma versão ampliada de qualquer caracter: normal, gráfico ou definido por nós. Não permite o uso das palavras-chave ou funções.

Só é possível dispor de quatro caracteres por linha, podendo-se escolher a cor deles. A variável X armazena o endereço de início, na ROM, do conjunto de caracteres.

```
    1 REM Bill Board
    2 REM     por D. Harwood
    4 LET d=0
    5 LET x=PEEK 23606+256*PEEK 2
3607
   10 INPUT " Cor de tinta     ? ";i: IN
K i
```



```
   20 INPUT "    Escreva 4 letras";a$
   25 FOR k=0 TO 3
   30 LET p1=x+CODE a$(k+1)*8
   40 FOR g=0 TO 7
   50 LET p=PEEK (p1+g)
   60 FOR j=0 TO 7
   70 IF p/2<>INT (p/2) THEN PRIN
T AT 13+g,8-j+(k*8); INK i;"■"
   80 LET p=INT (p/2)
   90 NEXT j: NEXT g: NEXT k
  100 INPUT " Novamente?  ";s$
  110 POKE 23692,-1
  120 IF s$(1)="s" THEN PRINT ''
'''': GO TO 10
```



CATCH

Neste programa de Andrew Lovering, pode-se ver uma caixa aberta na parte inferior do visor. Carregando em "8" movemo--nos para a direita, e em "5" para a esquerda. É necessário tentar apanhar um balão que desce muito rapidamente, caindo de uma posição aleatória no cimo do visor. Para confundir o jogador, e para introduzir um elemento de sorte no jogo, surge uma tampa em posições aleatórias sobre a caixa.

Se o balão atinge o fundo, ou se toca na tampa, é impressa a pontuação obtida e o jogo recomeça quando se carrega em qualquer tecla.

```
  1 REM     execute as linhas 2 a 4 antes de escrever o
resto do programa
  2 FOR t=0 TO 7: READ z
  3 POKE USR "p"+t,z: NEXT t
  4 DATA 56,124,254,124,56,16,1
6,96
  5 LET d=0
 10 LET a=14: LET c=INT (RND*30
)+1
 15 LET x=RND*5: LET y=RND*5
 20 LET a=14: LET c=INT (RND*30
)+1: LET b=0
 30 LET b=b+1
 35 REM     para obter o gráfico da linha 40 escreva 'p'
 em modo gráfico
 40 PRINT AT b,c; INK y;"p"
 50 LET e=INT (RND*30)
 60 PRINT AT 19,e; INK x;"    "
 70 PRINT AT 20,a; INK x;" ■  "
 80 PRINT AT 21,a; INK x;"    "
 90 IF INKEY$="5" AND a>0 THEN
LET a=a-1
100 IF INKEY$="8" AND a<29 THEN
 LET a=a+1
110 IF b=18 AND c=e+1 THEN GO T
O 190
120 IF b=20 AND c=a+1 THEN GO T
O 150
130 IF b=21 THEN GO TO 190
```



```
140 CLS : GO TO 30
150 LET d=d+1
160 PRINT AT 0,0; INK 1; FLASH
1;" pontuação ";d
170 FOR j=10 TO 20: BEEP .1,j
175 BEEP .1,20-j: NEXT j
180 GO TO 10
190 PRINT AT 0,0; INK 1; FLASH
1;" Pontuação ";d
200 PAUSE 4e4: RUN
```

VIAGEM ENTRE AS ESTRELAS

Este jogo foi escrito por Tim Rogers, sendo necessário nele conduzir uma nave entre as estrelas, deixando um rasto enquanto se atravessa o universo.

De cada vez que o computador cria o universo é-nos permitido ver onde as estrelas são colocadas, de tal modo que quando a nave penetra no universo a uma altura aleatória no lado direito do visor já podemos ter alguma ideia sobre o melhor modo de a conduzir para evitar as estrelas. A pontuação é fornecida no final do jogo. Quando se toca uma estrela, ocorre uma explosão ruidosa.

```
  1 REM     Execute as linhas 2 a 4 antes de escrever o
resto do programa
  2 FOR t=0 TO 7: READ a
  3 POKE USR "p"+t,a: NEXT t
  4 DATA 24,60,126,255,126,90,1
53,24
 10 LET s=0
 30 PAPER 0: BORDER 0: INK 7: C
LS
 40 FOR a=1 TO 30
 50 PRINT AT INT (RND*20),INT (
RND*32); INK 4;"*"
 55 NEXT a
 60 LET p=INT (RND*21)
 70 FOR a=31 TO 0 STEP -1
 75 REM     Para obter o caracter da linha 80 escreva 'p'
 em modo gráfico
```

```
  80 PRINT AT p,a;"▲"
  90 IF ATTR (p,a-1)=4 THEN GO T
O 160
 100 PRINT AT p,a;"■"
 110 LET p=p+(INKEY$="6")-(INKEY
$="7")
 120 LET s=s+1
 121 FOR g=1 TO 100: NEXT g
 130 NEXT a
 140 FOR g=1 TO 20: NEXT g
 150 CLS : GO TO 40
 160 FOR s=1 TO 10
 165 BEEP .1,s
 170 PRINT AT p,a-2; INK 0;"■■"
 180 BEEP .1,10-s
 190 PRINT AT p,a-2; INK 7;"■■"
 200 NEXT s
 210 FOR g=1 TO 100: NEXT g
 240 PRINT FLASH 1;" Pontuação =";s
```

PATRULHA GALÁCTICA

Este jogo coloca-nos ao comando de uma nave de patrulha. Esta começa com 50 galões de combustível, diminuindo esta quantidade lentamente. A quantidade de combustível restante em qualquer momento é indicada no canto superior direito do visor.

Pode-se reabastecer a nave tocando em qualquer dos depósitos (quadrados verdes) distribuídos aleatoriamente. De cada vez que se passa por um destes depósitos obtêm-se 25 galões. Controla-se a nave carregando na tecla "M", o que nos permite deslocar a nave para a direita; não tocando na tecla a nave deriva lateralmente.

O jogo termina, sendo a pontuação final apresentada, quando se gasta todo o combustível ou se toca num asteróide vermelho. O programa foi escrito por R. Stubbs.

```
  1 REM      Execute as linhas 2 a 4 antes de escrever o
resto do programa
  2 FOR t=0 TO 7: READ a
  3 POKE USR "p"+t,a: NEXT t
```



```
  4 DATA 24,60,126,255,126,90,1
53,24
 10 LET f=50: LET s=0: LET a=0:
LET b=13
 20 PRINT AT 21,RND*30; INK 2;"
*"
 30 LET r=INT (RND*10)+1
 35 IF r=9 THEN PRINT AT 8,RND*
30; INK 4;"■"
 40 LET s=s+1: LET f=f-1
 50 POKE 23692,-1: PRINT '
 60 IF b>2 THEN LET b=b-1
 70 IF INKEY$="m" AND b<28 THEN
LET b=b+2
 72 IF f<10 THEN PRINT AT 0,20;
INK 3; INVERSE 1;"LOW FUEL ": B
EEP .01,0: GO TO 75
 73 PRINT AT 0,20;"          "
 75 REM    Para obter o caracter da linha 80 escreva 'p'
em modo gráfico
 80 PRINT AT 0,b;"▲";TAB 29;f
 90 LET p=ATTR (1,b)
100 IF p=58 THEN GO TO 140
110 IF p=60 THEN LET f=f+25
120 IF f=0 THEN GO TO 150
130 GO TO 20
140 PRINT INK 2; FLASH 1;
" Pouca sorte, chocou "
                    : GO TO 160
150 PRINT INK 1; FLASH 1;"    Pouca
sorte, gastou todo o combustível"
160 FOR t=1 TO 10: BEEP .1,t: N
EXT t
170 INPUT "     Outro jogo?          ";a$
180 IF a$(1)="y" THEN RUN
```

ZOMBIES

Nesta versão de ZOMBIES, o utilizador parte do canto superior esquerdo do visor e tenta atingir o canto inferior direito. Pode-se mover para a esquerda, para a direita e para baixo — usando as teclas "5", "6" e "8". Não choque com nenhum ZOMBIE. Tenha cuidado, porque os ZOMBIES seguem-no continuamente!

O programa utiliza um caracter em forma de face para representar o jogador, e um ponto para representar um ZOMBIE.

```
  1 REM     Zombies, por D. Harwood
  2 REM     Execute as linhas 3 a 6 antes de escrever o
resto do pròg rama
  3 FOR t=0 TO 7: READ a,b
  4 POKE USR "p"+t,a: POKE USR
"o"+t,b: NEXT t
  5 DATA 60,60,66,126,165,255,1
65,255
  6 DATA 129,255,189,255,66,125
,60,60
 10 PRINT INK 1; FLASH 1;"ZOMBI
ESZOMBIESZOMBIESZOMBIES"
 20 PRINT '' INK RND*5;" O jogador
deve tentar chegar a casa, mas tenha cuidado porque está a,ser

 30 PRINT INK RND*5; " perseguido por
 "; INK 2; FLASH 1;"ZOMBIES"; I
NK RND*5; FLASH 0;"     Não toque neles! "

 40 PRINT '' TAB 7; INK 2; FLASH
1;" PREPARE-SE "
 50 PAUSE 300
 60 CLS
 70 BORDER RND*5
 80 PRINT AT 0,0; INK 1; FLASH
1;"S";AT 21,31; INK 1; FLASH 1;"
H"
 90 LET x=0: LET y=1: LET c=4:
LET d=4
 110 IF INKEY$="" THEN GO TO 110
 120 PRINT AT x,y; INK 1;" "
 130 LET x=x+(INKEY$="6")-(x>21)
 140 LET y=y+(INKEY$="8")-(INKEY
$="5")+(y<0)-(y>30)
 150 LET p=ATTR (x,y)
 160 IF p=58 THEN GO TO 300
 161 FOR g=1 TO 2
 162 IF c<20 THEN LET c=RND*3+x
 164 IF d<28 THEN LET d=RND*6-3+
y
 165 REM   Para obter o caracter da linha 170 escreva 'o'
em modo gráfico
 170 PRINT AT c,d; INK 2;"●"
 171 NEXT g
 175 REM   Para obter o caracter da linha 180 escreva 'p'
em modo gráfico
 180 PRINT AT x,y; INK 1;"@"
 190 IF x=21 AND y=31 THEN GO TO
 210
 200 GO TO 110
```



```
210 PRINT AT 21,0; INK 1; FLASH
1;"   Óptimo, sobreviveu! "
220 FOR t=1 TO 10
230 BEEP .1,t: BEEP .2,10-t
240 NEXT t
250 STOP
300 PRINT AT 10,0; INK 1; FLASH
1;"    Apanharam-no..."
320 FOR a=1 TO 10
330 READ b,c: BEEP b,c
340 NEXT a
350 DATA 1,0,1,0,.5,2,.5,0,.5,3
,.5,0,.5,-1,.5,0,.5,1,1,0
3660 STOP
```

MORSE

O leitor pode tornar-se um mestre em código Morse com o auxílio deste programa escrito por John Knight. Quando o executar, verá aparecer um menú que lhe permite optar entre escrever uma mensagem, depois reimpressa e "tocada" em Morse, ou levar a máquina a produzir um som e um símbolo Morse aleatoriamente. O programa permite-lhe fazer três tentativas para descobrir o caracter equivalente ao símbolo, ou terminar a execução.

A rotina na linha 9000 divide A$ em elementos de C$. Para simplificar o processamento anterior, C$(97) é o equivalente a CHR$(97), isto é, à letra 'a'. Este programa mostra-nos uma boa aplicação da ordem BEEP.

```
 10 REM    Treino de Morse
 20 REM © J Knight, 1982
 30 GO SUB 9000
 40 POKE 23692,-1
 70 PRINT '' INK 2; FLASH 1;
"FAÇA A SELECÇÃO";
 80 PRINT '''"1-    Português para Morse
 90 PRINT ''"2-    Morse para Português
"
100 PRINT ''"3-    Fim "
110 IF INKEY$="" THEN GO TO 110
```

```
 120 GO SUB VAL (INKEY$)*1000
 130 GO TO 40
1000 REM     Português para Morse
1010 PRINT INK 1;'''''"                    PORTUGUÊS
PARA MORSE"
1020 PRINT '' INK 2;"Escreva a mensagem, e"

1030 PRINT 'TAB 3; INK 2; FLASH
1;" Carregue em ENTER"
1040 INPUT w$
1045 PRINT
1050 FOR g=1 TO LEN w$
1060 IF w$(g)" " THEN GO TO 1090
1070 PRINT ''
1080 GO TO 1100
1090 PRINT c$(CODE w$(1));
1092 LET z$=c$(CODE w$(1)): GO S
UB 5000
1100 LET w$=w$(2 TO )
1110 NEXT g
1120 IF INKEY$="" THEN GO TO 112
0
1130 RETURN
2000 REM     Morse para Português
2010 PRINT '''''"          Vou indicar-lhe uma letra"
2020 PRINT "     em Morse, e você deve escrever a que
corresponde"
2030 PRINT ' INK 2; FLASH 1;"
Carregue em ENTER quando estiver preparado para começar"
2040 IF INKEY$="" THEN GO TO 204
0
2050 LET s=0
2060 FOR z=1 TO 10
2070 LET j=97+INT (RND*26)
2080 PRINT '''''"  A que letra corresponde?"
; C$(j)
2090 LET z$=c$(j): GO SUB 5000
2100 FOR h=1 TO 3
2140 INPUT INK RND*5;"     A sua resposta:
"; k$
2150 IF CODE (k$)=j THEN GO TO 2
200
2160 IF h<3 THEN PRINT '' INK 1;
;"  Não, experimente de novo"
2170 IF h=3 THEN PRINT ''c$(j);
" representa ";CHR$ j
2175 LET z$=c$(j): GO SUB 5000
2180 NEXT h
2190 GO TO 2220
```



```
2200 PRINT ' INK 2; FLASH 1;"
  Está certo!"
2210 LET s=s+1
2220 PRINT '' INK RND*5;"    A sua pontuação é ";s;" num total de ";z
2230 PRINT '' INK 3;"
2240 NEXT z
2250 RETURN
2260 STOP
5000 FOR c=1 TO 5
5010 IF z$(c)="." THEN BEEP .1,0
5020 IF z$(c)="-" THEN BEEP .5,0
5030 PAUSE 10: NEXT c
5040 RETURN
9000 LET a$=".-*-...*-.-.*-..*.*
..-.*--.*....*..*.---*-.-*.-..*-
-*-.*---*.--.*--.-*.-.*...*-*..-
*...-*.--*-...-*-.--*--..**"
9010 DIM c$(122,5)
9020 FOR b=97 TO 122: LET b$=""
9030 FOR c=1 TO 5
9040 IF a$(1)="*" THEN GO TO 907
0
9050 LET b$=b$+a$(1): LET a$=a$(
2 TO )
9060 NEXT c
9070 LET c$(b)=b$: LET a$=a$(2 T
O )
9080 NEXT b
9090 RETURN
```

INVERSO

Neste programa é indicado ao utilizador um número de dez algarismos (0 a 9), devendo-se colocá-lo por ordem ascendente (isto é, 1234567890) por "inversão". Um exemplo esclarecerá melhor o que significa aqui "inverter". Se tivermos o número 1234567980 e quando nos for pedido o número invertido escrevermos 9, obteremos 1234567908. Se em seguida escrevermos 8, obteremos 1234567809. Se finalmente escrevermos 9, obteremos 1234567890, terminando o jogo.

No final o computador diz-nos o número de operações de que necessitámos para colocar os algarismos pela ordem correcta.

```
  1 LET x=0: LET a$=""
 10 INK 1: CLS
 20 PRINT TAB 7;"I ";: PAUSE 10
: PRINT "N ";: PAUSE 10: PRINT "
V ";: PAUSE 10: PRINT "E ";
 30 PAUSE 10: PRINT "R ";: PAUS
E 10: PRINT "S ";: PAUSE 10: PRI
NT "O"
 40 PRINT '''" Deve tentar escrever o número de 10
algarismos em "
 60 PRINT "ordem ascendente, i.e. "; INK 2;
 FLASH 1;"1234567890"
 70 PRINT '''"e.g. Se tivesse o número
       ' INK 1; FLASH 1;"1234
567098"
 80 PRINT "e no momento adequado"

 90 PRINT " escrevesse     "; INK 2; F
LASH 1;"8"; INK 1; FLASH 0;"
 transformaria o número original"
100 PRINT "em              "; INK 1;
FLASH 1;"1234567890"; FLASH 0;"
terminando"
110 PRINT "o jogo."
120 PRINT '' INK 3; FLASH 1;"
Carregue numa tecla para começar "
130 PAUSE 10000
140 CLS : POKE 23562,-1
150 FOR t=0 TO 9
160 LET l=INT (RND*10)+48
170 GO SUB 500: LET a$=a$+CHR$
l
180 NEXT t
190 PRINT '' INK RND*5;a$
210 INPUT "Inverter número?      ";r
220 PRINT INK RND*5;" Número invertido —
    "; FLASH 1;r
230 GO SUB 1000: LET x=x+1
240 IF a$<>"1234567890" THEN GO
TO 190
250 PRINT '''' INK RND*5; FLASH
1;a$;" "; INK RND*5;a$;" "; INK
RND*5;a$
260 FOR t=10 TO 30: BEEP 1/t,t:
NEXT t
270 PRINT '' INK RND*5; Conseguiu
    em ";x;" tentativas."
280 STOP
```



```
500 FOR i=1 TO t
510 IF a$(i)=CHR$ l THEN GO TO
160
520 NEXT i
530 RETURN
1000 LET b$=""
1010 FOR i=10 TO r STEP -1
1020 LET b$=b$+a$(i)
1030 NEXT i
1040 LET a$=a$( TO r-1)+b$
1050 RETURN
```

PUBLICIDADE

Este programa é útil para apresentar mensagens publicitárias. Em primeiro lugar cria-se um visor, numa dada cor, usando as teclas "5", "6", "7" e "8" para controlar as direcções. "1" serve para escrever, "2" para introduzir espaços. Depois de o visor estar terminado, carregue em "S" para parar, escreva em seguida a mensagem, e a linha onde deseja vê-la impressa. Esta mensagem pode ter qualquer comprimento. Depois de ter escrito a mensagem volta ao menú, escolhendo a opção seguinte. Escreva "S" para interromper a apresentação da mensagem.

O programa guarda 10 visores, mas apenas uma mensagem devido ao comprimento variável desta. O programa foi escrito por Paul Toland.

```
  1 DIM s$(5,704)
  2 LET c$="■"
  5 CLS
  7 POKE 23692,-1: PRINT AT 21,
0;
 10 PRINT INK 1; "1 - Criar novo visor""2 -
Apresentar visor existente""3 - Alterar mensagem""Parar"
 20 INPUT ;c
 30 IF c<1 OR c>4 THEN GO TO 20
 40 IF c=4 THEN STOP
 50 IF c=3 THEN GO TO 230
```

```
  60 INPUT INK 2; FLASH 1;" "Visor
  No. ? (1-5)";no
  70 IF no<1 OR no>10 OR no<>INT
no THEN GO TO 60
  80 IF c=2 THEN GO TO 300
  85 INPUT " Cor de tinta?      ";d
  90 PRINT INK 2;"  Criar visor usando 5, 6, 7,
8 . para controlar o cursor"

 100 PRINT ' INK 1;"    Use 1 para ■
2 para espaço e S para parar "
 110 LET x=16: LET y=10
 120 LET s$(no,1 TO 704)="  "
 121 FOR i=1 TO 100: NEXT i
 125 PRINT AT 0,0; INK d;s$(no);
AT y,x; INK 1; FLASH 1;"+"
 130 LET i$=INKEY$
 140 IF i$="1" THEN LET c$="■"
 150 IF i$="2" THEN LET c$=" "
 160 LET x=x-(i$="5")+(i$="8")
 170 LET x=x+(x<0)-(x>31)
 180 LET y=y-(i$="7")+(i$="6")
 190 LET y=y+(y<0)-(y>21)
 200 LET s$(no,y*32+x+1)=c$
 210 IF i$="s" THEN GO TO 230
 220 GO TO 125
 230 INPUT INK 1;"  Em que linha deseja
imprimir  a mensagem (0-21)?"
 240 IF l<0 OR l>21 THEN GO TO 2
30
 245 INPUT INK 1;"  Em que cor ?";z
 246 INPUT FLASH 1;' "Cintilando? (1 - sim,
0 - não)" q
 250 INPUT INK 2; FLASH 1;  Escreva a
mensagem...
                    ";i$
 260 LET l$="
                "+i$+"
                "
 270 GO TO 5
 300 PRINT AT 0,0; INK d;s$(no)
 310 FOR i=1 TO LEN l$-32
 320 PRINT AT l,0; INK z; FLASH
q;l$(i TO i+31)
 330 IF INKEY$="s" THEN STOP
 335 PAUSE 3
 340 NEXT i
 350 GO TO 310
```


Este programa pode ajudá-lo a ter a certeza de que não passa cheques em branco... Permite-lhe dar entrada a informações relativas à sua conta, dá-lhe a opção de alterar uma dada entrada se o quiser, e finalmente imprime informações sobre todos os cheques passados (a quem o foram, porquê, e em que quantia). Em seguida o programa imprime o saldo final da conta e, se necessário, dá-lhe a boa notícia de que a conta está a zero...

Este programa foi escrito por Tim Hartnell.

```
  5 LET a$=CHR$ 8+CHR$ 8+CHR$ 8
+CHR$ 8+CHR$ 8+CHR$ 8+CHR$ 8
  7 INK 1: CLS
 10 REM     Livro de cheques
 20 REM © T Hartnell 1982
 30 POKE 23692,-1
 40 INPUT INK 2; FLASH 1;"
    Indique último saldo conhecido
                    ";b
 50 PRINT "     Saldo inicial, $ ";
b
 60 PRINT '"     Depósito ";a$; OVER 1
; INK 2;"_____"
 65 PRINT "     Saldo     ";a$; OVER 1;
INK 2;"_____"
 70 INPUT INK 2; FLASH 1;"
    Indique depósito feito (0 para terminar)
                    ";d
 80 IF d=0 THEN GO TO 120
 90 LET b=b+d
100 PRINT d,b
110 GO TO 70
120 PRINT '" Saldo anterior aos""""últimos cheques
passados"; INK 2; FLASH 1;"$";b
130 INPUT INK 2; FLASH 1;"
Quantos cheques passou desde então ";n

140 DIM a$(n,22): DIM b(n)
150 FOR g=1 TO n
160 INPUT INK 2: FLASH 1;"
Indique nome
                    ";n$
170 INPUT INK 2; FLASH 1;"
Indique justificação
                    ";m$
```

```
180 LET a$(g)=n$+" -"+m$
190 INPUT INK 2; FLASH 1;"How m
uch is the cheque? ";q
200 LET b(g)=q
210 PRINT a$(g);"- "; INK 2; F
LASH 1;"£";b(g)
220 INPUT INK 2;" Se estiver correcto, carregue
em ENTER; senão, escreva 'e' ";u$
230 IF u$="e" THEN GO TO 160
240 LET b=b-q
250 NEXT g
260 PRINT '''''' Registo dos seus cheques
até ao momento ";
270 PRINT "_____
_____"
280 PRINT '';
290 FOR g=1 TO n
300 PRINT a$(g);" - "; INK 2; F
LASH 1;"£";b(g)
310 NEXT g
315 PRINT "_____
_____"
320 PRINT '" Saldo final: ";
INK 2; FLASH 1;"$";b
325 PRINT "_____
_____"
326 POKE 23692,-1
330 PRINT '''' Escreva 'r' para nova execução ";

340 PRINT " desde início "
350 PRINT '" ou 'p' para uma impressão ";

360 PRINT " detalhada dos cheques ";

370 PRINT "escritos "
380 PRINT '' ou 'e' para terminar "
390 INPUT u$
400 IF u$="r" THEN RUN
410 IF u$="b" THEN GO TO 50
420 IF u$="p" THEN GO TO 450
430 IF u$="e" THEN PRINT INK 3,
FLASH 1;TAB 7;" PROGRAMA TERMINADO
STOP ";:
440 GO TO 390
450 REM **, Impressão
460 FOR g=1 TO n
470 LPRINT a$(g),"£";b(g)
480 NEXT g
490 GO TO 330
```


CONDUTOR

Neste programa, escrito por Said Hassan, é necessário conduzir um carro verde por uma estrada bastante acidentada. O carro desloca-se atravessando o visor; depois de completar cada secção, o visor é limpo e surge um novo troço de estrada. O programa guarda informações sobre as maiores pontuações obtidas.

```
  2 FOR t=0 TO 7: READ a,b
  3 POKE USR "p"+t,a: POKE USR
"o"+t,b
  4 NEXT t
  5 DATA 170,255,85,255,170,60,
85,60
  6 DATA 170,60,85,60,170,255,8
5,255
  7 LET c$="C R A S H "
 10 LET h=0
 20 LET s=0
 25 LET x=5: LET a=3: CLS
 30 FOR n=0 TO 20
 35 REM Para obter o caracter gráfico da linha 40 escreva 'p'
em modo gráfico
 40 PRINT TAB a; INK 1;"▩"; INK
2;"▒▒▒▒▒▒▒"; INK 1;"▩"
 50 LET a=a+(a<7 AND a)*SGN (RN
D-.5)+(NOT a)-(a=7)
 60 NEXT n
 70 LET a=x
 80 FOR n=1 TO 20
 90 LET p=ATTR (n,x)
105 REM Para obter o caracter gráfico da linha 110 escreva 'o'
em modo gráfico
110 PRINT AT n-1,a; INK 2;"▒";A
T n,x; INK 4;"▙"
120 IF p=57 THEN GO TO 170
130 LET a=x
140 LET x=x+(INKEY$="m")-(INKEY
$="z")
145 FOR j=1 TO 30: NEXT j
150 NEXT n
160 LET s=s+n: GO TO 25
170 LET s=s+n
175 FOR a=1 TO 10: BORDER RND*7
: BEEP .1,a-10
```

```
180 PRINT AT 2,10+a; INK RND*5;
c$(a)
183 NEXT a
185 PRINT AT 4,11;"PONTUAÇÃO";s
190 IF h<s THEN LET h=s
200 PRINT AT 6,11; INK 1; FLASH
1;"   Maior pontuação: ";h
210 INPUT "   Deseja outro jogo? ";a$
230 IF a$(1)="y" THEN GO TO 20
```



O Spectrum possui excelentes potencialidades gráficas dispondo ainda de cor e de som, e é também útil no campo das operações matemáticas. Pode ser levado a realizar tarefas bastante úteis e complexas. Os programas que se seguem dão uma ideia da variedade de modos de utilização do Spectrum.

O primeiro programa é muito simples, e calcula o resto de uma divisão. Usa-se o sinal de divisão habitual, definido nas linhas 3 a 5.

O segundo faz cálculo estatístico, determinando a área sob uma curva de Distribuição Normal, no ponto indicado para a variável X na linha 20. Este programa tem apenas um rigor de cerca de 10^{-4}.

O terceiro programa é um programa de demonstração, que utiliza o esplêndido modo de tratamento de cadeias deste computador, permitindo o uso de ordens que "contam" palavras entre zero e 99. Ouve-se um som por cada contagem.

O quarto programa converte as temperaturas de graus Centígrados em Fahrenheit.

O quinto programa desenha um gráfico de conversões, usando uma escala a cores fácil de aprender, transformando Fahrenheit em Centígrados.

O sexto programa determina a quantia que é necessário pagar por um empréstimo quando a quantia envolvida e o juro variam.

```
  1 REM  Calculo de restos
  2 REM  Execute as linhas 2 a 4 antes de escrever o
resto do programa
  3 FOR t=0 TO 7: READ a
  4 POKE USR "p"+t,a: NEXT t
  5 DATA 24,24,0,255,255,0,24,2
4
 15 POKE 23692,-1
```

```
  20 INPUT INK 1;" Indique número a dividir"
  30 INPUT ( INK 2;a;"      Dividir por?
");b
  40 PRINT ' INK RND*5;a;" ÷ ";b
;" = ";INT (a/b);"      resto      ";a
-b*INT (a/b)
  60 GO TO 20
```

3212 ÷ 32 = 100 resto 12

234 ÷ 9 = 26 resto

213 ÷ 6 = 35 resto 3

```
   5 REM *    Distribuição normal
   7 POKE 23692,-1
  10 PRINT INK 1; "DISTRIBUIÇÃO NORMAL"
  20 INPUT INK 1; FLASH 1;"X=";x
  30 LET t=1/(1+0.231642*x)
  40 LET q=1/SQR (2*PI)*EXP (-x↑
2/2)
  50 LET a=0.319382: LET b=-0.35
6564
  60 LET c=1.781480: LET d=-1.82
1256
  70 LET e=1.330274
  80 PRINT INK RND*5;"PX=";1-q*(
a*t+b*t↑2+c*t↑3+d*t↑4+e*t↑5)
  90 GO TO 10
```

```
   5 POKE 23692,-1: PRINT AT 21,
0;
  10 LET t$=' "Zero um dois três quatro
 cinco seis sete oito nove "
  20 FOR z=0 TO 9: FOR x=0 TO 9
  30 LET k=1+6*z
  40 IF z=0 THEN LET q$="": GO T
O 60
  50 LET q$=t$(k TO k+5)
  60 LET k=1+6*x
  70 LET q$=q$+t$(k TO k+5)
  80 PRINT INK RND*5;q$
  85 FOR t=1 TO z*10+x: BEEP .1,
10: NEXT t
  90 NEXT x: NEXT z
```


Zero
um
dois
três
quatro
cinco
seis
sete
oito
nove
um Zero
um um
um dois
um três
um quatro
um cinco
um seis
um sete
um oito
um nove

```
   1 REM Centígrado para Fahrenheit
   2 REM Execute as linhas 3 a 5 antes de escrever o
resto do programa
   3 FOR t=0 TO 7: READ a
   4 POKE USR "p"+t,a: NEXT t
   5 DATA 64,160,64,0,0,0,0,0
  10 POKE 23692,-1: PRINT AT 20,
0;
  20 INPUT INK 2; FLASH 1;"
Temperatura Celsius?
  30 LET f=1.8*c+32
  35 REM     Para obter o caracter da linha 50 escreve 'p'
em modo gráfico
```

```
  50 PRINT ' INK RND*5;c;"°
centigrados = ";f;"° fahrenheit"
  60 GO TO 20
```



```
100° centrigrado = 212° fahrenheit
50° centrigrado = 122° fahrenheit
32° centrigrado = 89.6° fahrenheit
12° centrigrado = 53.6° fahrenheit
```

```
  10 REM    Gráfico de conversão
  20 REM    Fahr. para Cent.
  30 FOR f=30 TO 220 STEP 10
  40 IF f<100 THEN PRINT " ";
  50 PRINT f;
  60 FOR c=1 TO 5*(f-32)/45
  65 IF c/6-1=INT (c/6-1) THEN L
ET i=0
  66 LET i=i+1
  70 PRINT INK i;"■";
  75 NEXT c
  76 LET i=0: PRINT
  80 NEXT f
  90 PRINT 'TAB 3;"1020304050607
08090100"
```



```
 30
 40
 50 ■■
 60 ■■■
 70 ■■■■
 80 ■■■■■
 90 ■■■■■■
100 ■■■■■■■
110 ■■■■■■■■
120 ■■■■■■■■■
130 ■■■■■■■■■■
140 ■■■■■■■■■■■
150 ■■■■■■■■■■■■
160 ■■■■■■■■■■■■■
170 ■■■■■■■■■■■■■■
180 ■■■■■■■■■■■■■■■
190 ■■■■■■■■■■■■■■■■
200 ■■■■■■■■■■■■■■■■■
210 ■■■■■■■■■■■■■■■■■■
220 ■■■■■■■■■■■■■■■■■■■
    10203040506070809010O
```

```
  2 INK 1: CLS
  5 POKE 23692,-1: PRINT AT 20,
0
  10 PRINT ''"    Juro de empréstimo      "
  20 INPUT INK 2; FLASH 1;"
Indique quantia ";a
  30 INPUT INK 3; FLASH 1;"
   Indique taxa de juro    ";i
  40 LET i=i/100
  50 INPUT INK 4; FLASH 1;"
 Indique duração do emprestimo em anos      ";y
  60 LET y=12*y
  70 LET b=a*(i/12)/(1-(1+(i/12)
)↑-y)
  80 LET b=INT (b*100+.5)/100
  90 PRINT "'   O pagamento mensal é
   "; INK 2; FLASH 1;"$";b
 100 IF INKEY$="" THEN GO TO 100
 110 GO TO 10
```

ALPHASORT

O programa Alphasort ordena palavras ou frases por ordem alfabética. Indica-se primeiro (linha 10) o número de elementos que se deseja ordenar, e em seguida as palavras, uma de cada vez (linha 50), carregando em ENTER depois de cada palavra.

```
 10 REM Alphasort
 20 INPUT INK 2; FLASH 1;"
Indique número de elementos
";n
 30 DIM s$(n+1,15)
 40 FOR t=1 TO n
 50 INPUT INK RND*5; ("Elemento "; t;
" ");  LINE s$(t)
 60 NEXT t
 70 PRINT INK RND*5;"  Um momento...
...estou a ordenar"
 80 FOR i=1 TO n: FOR t=1 TO n
 85 LET b$=s$(t)
 90 IF s$(t+1)>=s$(t) THEN GO T
O 110
100 GO TO 130
110 LET s$(t)=s$(t+1)
120 LET s$(t+1)=b$
130 NEXT t: NEXT i
140 FOR t=n TO 1 STEP -1
150 PRINT INK RND*5;s$(t)
160 NEXT t
```

DATA SORT

O programa DATA SORT ordena números por ordem ascendente, imprimindo-os em seguida.

```
  5 LET y=1: LET x=5: LET a=5
10 INPUT INK RND*5; FLASH 1;"
Quantos elementos?";n
20 DIM a(n+1)
30 FOR g=1 TO n
40 INPUT INK RND*5; FLASH 1,("
Elemento ";g;" ");a(g)
45 NEXT g
47 FOR g=1 TO n: FOR t=1 TO n
50 IF a(t+1)<a(t) THEN GO TO 8
0
60 LET b=a(t)
70 LET a(t)=a(t+1): LET a(t+1)
=b
80 NEXT t: NEXT g
90 FOR t=n TO 1 STEP -1
100 PRINT INK RND*5;"Elemento ";n-t
+1;": ";a(t)
110 NEXT t
```


RAÍZES QUADRADAS

Este programa determina a raiz quadrada de qualquer número. Primeiramente escolhe um número qualquer entre zero e o número indicado (linha 50). Em seguida aproxima o resultado por divisões sucessivas, verificando de cada vez se o número tratado é a resposta correcta. A linha 80 existe simplesmente para que o leitor observe a acção do computador enquanto procura a resposta (verifique isto eliminando a linha 80).

Como o Spectrum possui uma função SQR, este programa não tem grande interesse prático, mas é interessante executá-lo e observar como se aproxima do resultado correcto. O programa foi escrito por Tim Hartnell.

```
 10 REM    Raizes quadradas
 20 REM © Hartnell 1982
 30 INPUT INK RND*5; FLASH 1;"
De que número devo achar a raiz? "; b
 40 IF b<0 THEN GO TO 30
 45 PRINT INK 1; FLASH 1;" O
numero é "; b
 50 LET a=INT (RND*b)+1
 60 LET x=b/a: LET y=(x+a)/2
 70 PRINT INK RND*5;y
```

```
  80 PRINT INK RND*5;" O erro é"
;ABS (y-SQR b)
  90 IF a=y THEN GO TO 110
 100 LET a=y: GO TO 60
 110 PRINT '' INK RND*5; FLASH 1
;"     A raiz quadrada é ";y
```

```
 O número é 10
3.6333333
 O erro é     0.67105567
3.2210145
 O erro e     .058736833
3.1628132
 O erro é     .00053554773
3.1622777
 O erro é     4.5634806E-8
3.1622777
 O erro é     0
3.1622777
 O erro é     0
```

A raiz quadrada é 3.1622777

PROBABILIDADES

O programa que se segue, escrito por Martin Frobisher, imprime uma tabela e um gráfico mostrando os possíveis resultados do lançamento de três dados. A primeira coluna da tabela mostra o total dos três dados, a segunda o número de combinações possíveis que podem dar este total, e a terceira dá a probabilidade de obtenção deste total.

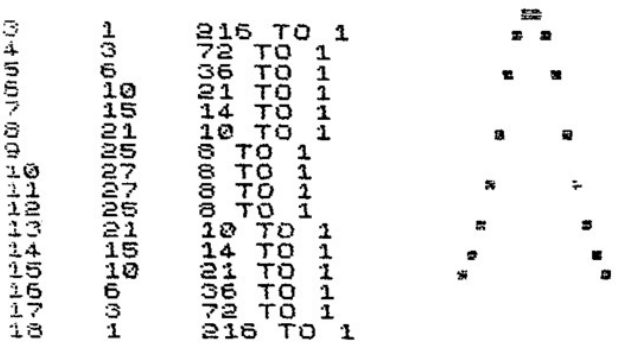
O gráfico impresso ao lado da tabela é a curva de distribuição, e mostra a probabilidade de os dados darem um qualquer resultado.

```
  1 LET j=RND*6
  2 FOR b=3 TO 18
  5 PRINT AT b,20; PAPER j;"
    "
```



```
  6 NEXT b
 10 DIM a(16)
 20 FOR b=1 TO 6
 30 FOR c=1 TO 6
 40 FOR d=1 TO 6
 50 LET e=b+c+d
 60 LET a(e)=a(e)+1
 70 NEXT d: NEXT c: NEXT b
 80 FOR b=3 TO 18
 90 PRINT AT b,0;b;TAB 5;a(b);T
AB 10;INT (216/a(b));" TO 1"
100 PLOT INK 9;2*b+200,50+3*a(b
)
110 NEXT b
```

```
3      1      215 TO 1
4      3      72 TO 1
5      6      36 TO 1
6      10     21 TO 1
7      15     14 TO 1
8      21     10 TO 1
9      25     8 TO 1
10     27     8 TO 1
11     27     8 TO 1
12     25     8 TO 1
13     21     10 TO 1
14     15     14 TO 1
15     10     21 TO 1
16     6      36 TO 1
17     3      72 TO 1
18     1      216 TO 1
```



CORRELAÇÃO / REGRESSÃO

Este programa aceita os nossos dados, X, terminados por 999, e depois os dados Y.

Em seguida calcula os vários sub-totais (soma de X, Y, etc.), o coeficiente de correlação de Pearson, e a equação de regressão (linear).

Neste momento deve-se carregar em qualquer tecla para continuar. Pode-se então interpolar/extrapolar a equação da linha indicando valores de X, terminando com 999.

Nesta fase do programa, enfrenta-se a opção de enviar os resultados para a impressora, desde que se tenha respondido com "s" à pergunta "IMPRESSORA LIGADA?". Pode-se provocar uma nova execução do programa carregando apenas em ENTER, ou pará-lo escrevendo "N" e carregando em ENTER. O programa foi escrito por Paul Toland.

```
  1 INK 1: CLS
  5 POKE 23692,-1: PRINT AT 20,
0;
 10 PRINT "     Correlação / Regressão"
 20 INPUT "     IMPRESSORA LIGADA? ";p$
 30 GO TO 70
 31 IF p$="n" THEN RETURN
 33 INPUT FLASH 1;" Imprimir esta página?
";a$
 40 IF a$<> "S" THEN RETURN
 50 COPY
 60 RETURN
 70 DIM x(20): DIM y(20)
 80 PRINT '' INK RND*5;"
Indique valores X por ordem —  Termine com 999
 90 FOR i=1 TO 20
100 INPUT INK 1;("     Valor ";i;" "
);x(i)
110 IF x(i)=999 THEN GO TO 150
130 NEXT i
140 PRINT '' INK 2; FLASH 1;"
  20 é o maximo ";
150 LET i=i-1
160 PRINT "AT 21,12;i;"     Valores
170 PRINT '' INK RND*5;"
  Indique valores Y por ordem"
180 FOR j=1 TO i
190 INPUT INK 1;("     Valor ";j;" "
);y(j)
200 NEXT j
210 GO SUB 31
220 LET c=0: LET d=0
225 LET e=0: LET f=0: LET g=0
230 FOR j=1 TO i
240 LET c=c+x(j)
245 LET d=d+y(j)
250 LET e=e+x(j)*x(j)
260 LET f=f+y(j)*y(j)
270 LET g=g+x(j)*y(j)
280 NEXT j
290 LET r=(i*g-c*d)/SQR ((i*e-c
↑2)*(i*f-d↑2))
300 CLS
310 PRINT " Soma X ";c' INK 2;"
Soma Y";d;' INK 3;"  Soma de X↑2 ";e;'
INK 4;"  Soma de Y↑2 ";f;' INK 5;"
Soma de XY ";g
320 PRINT '"     Coeficiente de correlação de
Pearsons"
330 PRINT TAB 6;r
340 PRINT '"     Coeficiente de determinação"
350 PRINT TAB 6;r*r
360 LET b=(i*g-c*d)/(i*e-c↑2)
365 LET a=c/i-d/i*b
370 PRINT '"     Equação de regressão linear"
380 PRINT INK RND*5;"Y=";a;("+"
AND SGN b>-1);("-" AND SGN b<0)
;ABS b;" X"
390 IF INKEY$="" THEN GO TO 390
400 GO SUB 31
410 CLS
420 PRINT "     Interpolação / Extrapolação
999 para terminar"
430 PRINT "     X                Y"
440 INPUT x
450 IF x=999 THEN GO TO 490
460 LET y=a+b*x
```

```
470 PRINT x;TAB 10;y
480 GO TO 440
490 GO SUB 31
500 INPUT INK 2; FLASH 1;"
Nova execução? ";a$
510 IF a$(1)<>"n" THEN RUN
```

COMBINAÇÕES E ARRANJOS MATEMÁTICOS

Este programa, escrito por Said Hassan, calcula combinações e arranjos. É pedido ao utilizador que indique a operação que deseja ver realizada.

As linhas 300 a 400 verificam se os valores indicados são numéricos e se se encontram dentro da gama processada pela máquina. O arranjo de **n** elementos **r** a **r** é dada pela fórmula:

$$nAr = \frac{n!}{(n-r)!}$$

Por exemplo, considere-se cinco cartas de jogar que devem ser divididas em grupos de três (n = 5, r = 3, e 5A3 = 60).

A combinação de **n** elementos **r** a **r** é dada pela fórmula:

$$nCr = \frac{n!}{(n-r)!\ r!}$$

De quantos modos é possível escolher três títulos de livros num conjunto de cinco livros?

5C3 = 10

```
10 PRINT INK RND*5; FLASH 1;"'
'C' Combinações " ' "P' arranjos
```



```
 20 LET a$=INKEY$
 30 IF a$="" OR a$<>"p" AND a$<
>"c" THEN GO TO 20
 40 PRINT INK 1; FLASH 1;'"
 Indique N"
 50 GO SUB 300: LET n=VAL n$
 70 PRINT INK 2; FLASH 1;"
 Indique R"
 80 GO SUB 300: LET r=VAL n$
 90 IF r<=n THEN GO TO 120
100 PRINT ' INK RND*5; FLASH 1,
" Muito elevado "
110 GO TO 70
120 PRINT '' INK RND*5;"
Um momento»
130 LET z=n
135 GO SUB 410
140 LET b=a: LET z=n-r
150 GO SUB 410
155 LET c=a
160 IF a$="p" THEN GO TO 195
170 LET z=r
180 GO SUB 410
190 LET c=c*a
195 CLS
200 PRINT INK RND*5;n;" ";a$;"
";r;" = ";b/c
210 IF INKEY$="" THEN GO TO 210
220 CLS : RUN
300 INPUT n$
310 IF n$="" THEN GO TO 300
320 CLS
330 FOR x=1 TO LEN n$
340 IF (n$(x)>="0" AND n$(x)<="
9") THEN GO TO 370
350 PRINT INK RND*5;"   Erro ";F
LASH 1;"   Escreva de novo "
360 GO TO 300
370 NEXT x
380 IF VAL n$<34 THEN RETURN
390 PRINT INK RND*5;"   Muito elevado"
, FLASH 1;"   Escreva de novo "
400 GO TO 300
410 LET a=1
420 FOR x=1 TO z
430 LET a=a*x
440 NEXT x
450 RETURN
```



NÚMEROS PRIMOS

Este programa, como o título sugere, produz números primos. Quando se executa o programa, este pergunta-nos a quantidade de números primos que desejamos obter. Começará em seguida a gerá-los, imprimindo-os simultaneamente. Se deseja um registo permanente dos resultados altere a linha 110 para LPRINT d,.

```
  10 LET x=1: LET y=2: LET d=3
  20 INPUT INK 2; FLASH 1;"
Quantos números primos?";a
  30 PRINT INK 2; FLASH 1;a;"
 Primos"; FLASH 0;1,2,3,
  40 FOR b=x TO a-d
  50 LET d=d+y: LET c=y+x
  60 LET e=INT (d/c): LET f=d-e*
c
  70 IF f=0 THEN GO TO 50
  80 IF c>=e THEN GO TO 110
  90 LET c=c+y
 100 GO TO 60
 110 PRINT INK RND*5;d,
 120 NEXT b
```


SISTEMAS DE EQUAÇÕES

Este programa, escrito por Paul Stevens, resolve equações simultâneas na forma:

$$Ax + By = e$$
$$CX + Dy = f$$

Ao executar este programa, é perguntado ao utilizador o valor das constantes. O programa determina em seguida os valores de X e Y.

```
   5 REM     Sistemas de equações
   6 REM     Por Paul Stevens
  10 PRINT AT 0,0;" Sistemas de Equações "
  20 PRINT AT 0,0; OVER 1; INK R
ND*5;"                                "
  30 PRINT INK RND*5;"  ""AX+BY=E"
,"CX+DY=F"
  40 INPUT INK 1; FLASH 1;"A=";a
, INK 1; FLASH 1;"B=";b, INK 1;
FLASH 1;"E=";e
  50 INPUT INK 2; FLASH 1;"C=";c
, INK 2; FLASH 1;"D=";d, INK 2;
FLASH 1;"F=";f
  60 LET g=(c*b)-(d*a)
  70 LET h=(-d*e)+(b*f)
  80 LET i=(c*e)+(-a*f)
  90 LET x=h/g: LET y=i/g
 100 PRINT '' INK RND*5;a;"X+";b
;"Y=";e
```

```
 110 PRINT '' INK RND*5; c; "X+"; d
; "Y="; f
 120 PRINT '' INK 1; "   A resposta é
"; FLASH 1; "X=", x
 130 PRINT TAB 10; FLASH 1; "Y=";
y
 140 INPUT INK 2; FLASH 1; " Nova
execução?"; a$
 150 IF a$(1)="y" THEN RUN
```

Sistemas de equações

```
3X+5Y=E
2X+4Y=F

2X+3Y=14

5X+9Y=38
```

A resposta é X=4
Y=2

SOLUÇÃO DE EQUAÇÕES

Esta rotina, escrita por Jeremy Rushton, usa o método de Newton para resolver equações. Indica-se a equação que se deseja ver resolvida em ordem a X quando a máquina pede "Indique função". Depois, em resposta a "Indique valor inicial", indica-se o ponto de partida do computador. Deve ser um valor próximo daquilo que pensamos ser a resposta correcta, ou — no caso de haver mais do que uma resposta correcta - um número próximo daquele que procuramos. Em seguida carrega-se em ENTER e

observa-se o computador aproximando-se progressivamente da resposta.

Para experimentar o programa, indique X * X − 5 (para determinar a raiz quadrada de 5) ou X ↑ 3 − 27.6 para determinar a raiz cúbica de 27.6.

```
  1 REM      Solução de equações pelo método de Newton
  2 REM      Por Jeremy Rushton
 10 INPUT INK 2; FLASH 1; "
Indique uma função"; LINE f$
 20 PRINT INK RND*5; "    Função    : "
; f$
 30 INPUT INK 1; FLASH 1; "
Indique valor inicial ";      "; s
 40 PRINT INK RND*5; " Valor inicial"

 50 INPUT INK 3; FLASH 1; "
Indique máximo erro    "; err
 60 PRINT INK RND*5; "    Erro máximo
  : "; err
 70 PRINT AT 10,10; INK 1; "X=";
s
 80 LET x=s
 90 IF ABS (VAL (f$)) <err THEN
STOP
100 LET t=VAL f$: LET x=x+.0000
1
150 LET b=(VAL (f$)-t)/0.00001
160 LET s=s-t/b
170 GO TO 70
```

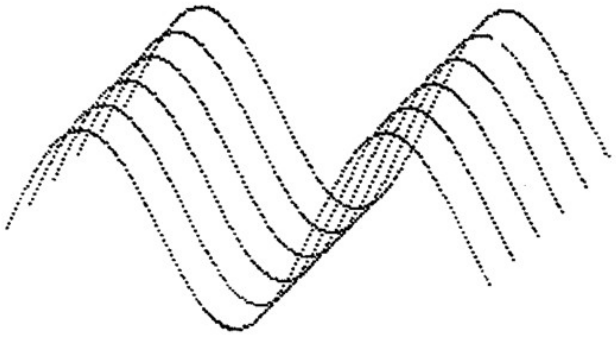
EXECUÇÃO DE FIGURAS

Os programas que se seguem demonstram as potencialidades do Spectrum na produção de figuras. O primeiro desenha uma série de círculos, usando a ordem CIRCLE para criar um cone. Experimente alterar **t, a** e **b** — e veja o que acontece.

O segundo programa desenha rectas, de comprimento aleatório, em cores também aleatórias, a partir dos quatro cantos do visor.

O terceiro programa desenha linhas rectas de tal modo que estas formam uma curva. O utilizador indica a cor de tinta e a espessura e o Spectrum faz o resto.

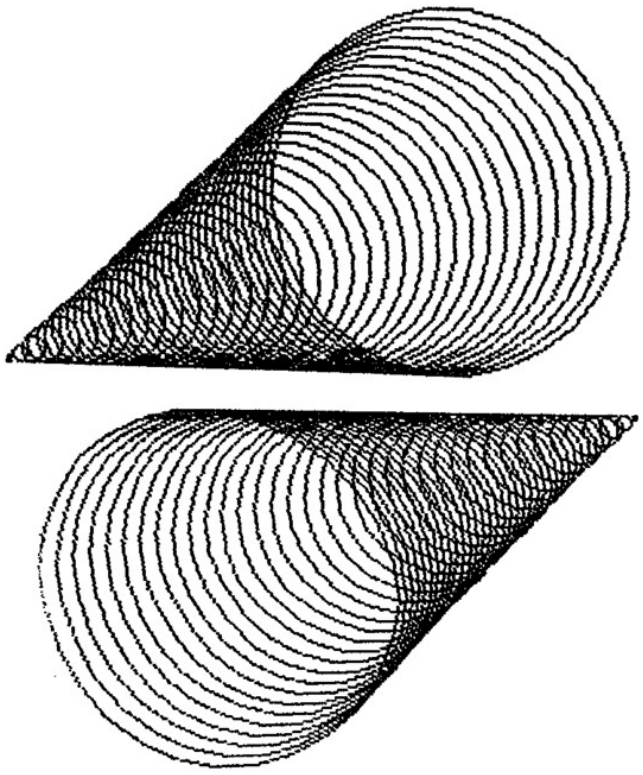
O quarto programa desenha uma série de curvas sinusoidais, produzindo o resultado da figura seguinte. As linhas são em seguida apagadas. Experimente alterar os valores 20 e 40 e o passo nas linhas 40 e 50.



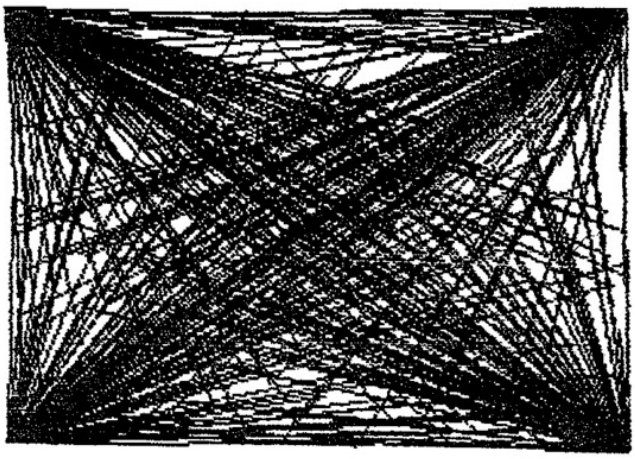
O quinto programa representa um gráfico depois de se indicar qualquer equação na linha 10.

O sexto programa desenha círculos, em diferentes coordenadas, e com diferentes pontos iniciais e finais, o que nos permite obter círculos sobrepostos como no exemplo apresentado.

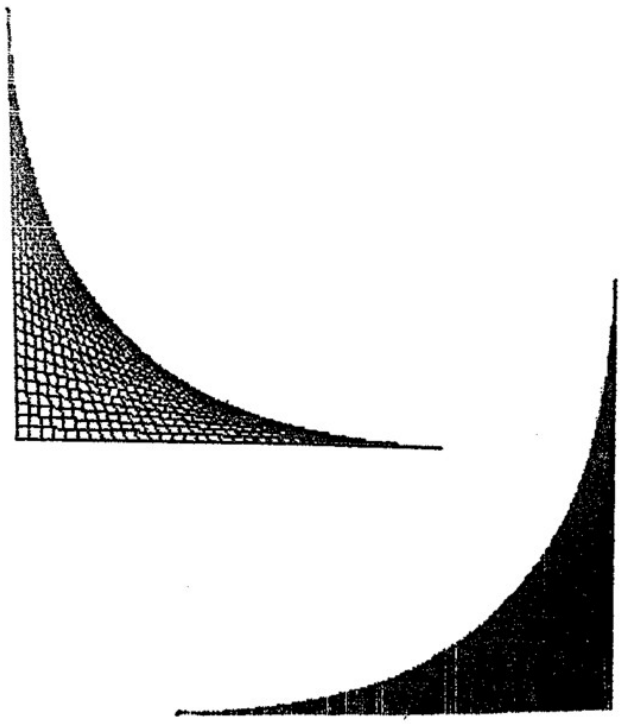

```
  1 BORDER RND*6: PAPER RND*6:
INK 9: CLS
 10 LET a=1: LET b=1
 20 FOR t=1 TO 185 STEP 5
 30 CIRCLE t,a,b
 40 LET a=a+2: LET b=b+2
 50 NEXT t
```



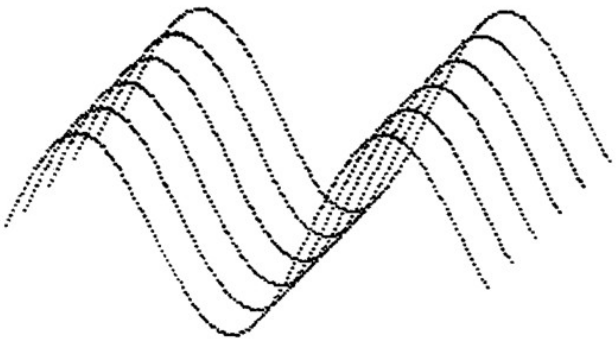
```
  10 PLOT 0,0: DRAW INK RND*6;RN
D*255,RND*175
  20 PLOT 255,175: DRAW INK RND*
6;-RND*255,-RND*175
  30 PLOT 255,0: DRAW INK RND*6;
-RND*255,RND*175
  40 PLOT 0,175: DRAW INK RND*6;
RND*255,-RND*175
  60 GO TO 10
```




```
  10 INPUT " Cor de tinta      ? ";c
  20 BORDER 7: PAPER 7: INK c: C
LS
  30 INPUT "    Escreva número entre 1 (fino)
e 25 (espesso) ";b
  40 LET a=0
  50 PLOT 0,175: DRAW 0,-175
  60 PLOT 0,0: DRAW 175,0
  70 FOR t=175 TO 0 STEP -b
  80 PLOT 0,t: DRAW a,-t
  90 LET a=a+b
 100 NEXT t
```



```
10 LET x=RND*5: PAPER x: INK 9
BORDER x: CLS
15 FOR d=1 TO 2
20 FOR a=0 TO 50 STEP 10
30 FOR x=0 TO 10 STEP .05
40 IF d=1 THEN PLOT 20*x+a,40*
SIN x+40+a
50 IF d=2 THEN PLOT OVER 1;20*
x+a,40*SIN x+40+a
60 NEXT x: NEXT a: NEXT d
```



```
10 INPUT INK 2;"  Escreva uma função
de X ";  LINE a$
15 LET c=RND*5: PAPER c: BORDE
R c: INK 9: CLS
20 PLOT 0,175: DRAW 0,-175: DR
AW 255,0
30 FOR x=0 TO 255
40 LET y=VAL a$
50 IF x>255 OR y>175 THEN GO T
O 80
60 PLOT x,y
70 NEXT x
80 PRINT AT 19,10; FLASH 1;"Y=
";a$
```



```
1 REM  Desenho de círculos
10 INPUT "  Coordenada X?  ";x:
INPUT "  Coordenada Y?  ";y
15 INPUT "  Início (primeiro raio)?
";s
20 INPUT "  Final (último raio)?
";f
30 INPUT "  Cor?  ";c
40 FOR z=s TO f
50 CIRCLE INK c;x,y,z
60 NEXT z
70 GO TO 10
```



SOMBRAS

Este programa produz um padrão em contínua transformação, imprimindo um quadrado azul quando encontra um quadrado negro e um quadrado negro quando encontra um quadrado azul.

```
1 REM  Sombras, por D. Harwood
5 FOR t=0 TO 21
6 PRINT "
```

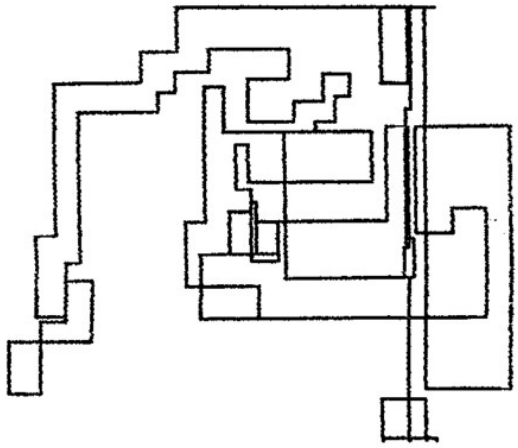
```
  7 NEXT t
 10 LET a$="▒"
 20 LET z=RND*21: LET a=RND*21:
LET b=RND*31
 30 IF ABS (z-b)>=21 THEN LET z
=b-10
 40 FOR t=0 TO b
 50 LET d=0
 60 LET p=ATTR (a,t)
 70 IF p=56 THEN LET d=1
 80 PRINT AT a,t; INK d;a$
 90 LET d=0
100 LET p=ATTR (ABS (z-t),b-1)
110 IF p=56 THEN LET d=1
120 PRINT AT ABS (z-t),b-1; INK
d;a$
130 NEXT t
140 GO TO 20
```


ETCHASKETCH

Este pequeno programa permite-nos usar as teclas '5', '6', '7' e '8' para deslocar uma linha pelo visor, desenhando imagens escolhidas por nós.

Em execução, a linha continua a mover-se até carregarmos numa tecla.

```
  1 LET a$="": LET a=127: LET b
=87
 10 IF INKEY$<>"" THEN LET a$=I
NKEY$
 20 LET b=b+(a$="7")-(a$="6")
 30 LET a=a+(a$="8")-(a$="5")
 40 IF a=256 OR a=0 THEN LET a=
ABS (a-255)
 50 IF b=176 OR b=0 THEN LET b=
ABS (b-175)
 60 PLOT a,b: GO TO 10
```



BLOCO DE DESENHO

Este programa permite-nos desenhar algumas imagens bastante interessantes. Controlamos a direcção usando as teclas '5' a '8'. Modifica-se o caracter carregando em 'H' e escrevendo em seguida o novo caracter. Altera-se a cor de tinta carregando em C e escrevendo depois a nova cor. Pode-se fazê-la cintilar carregando em 'F'; o efeito é eliminado carregando em 'N'.

```
10 REM      Bloco de Desenho
20 REM    Por D. Harwood
30 PRINT INK 1;TAB 6;"
BLOCO DE DESENHO"
40 PRINT '' INK RND*5;"
  Mover usando as teclas de cursor  "
50 PRINT '' INK RND*5;"
Se quiser alterar a cor "
60 PRINT INK RND*5;" Carregue em "; I
NK 2; FLASH 1;"C"; INK RND*5; FL
ASH 0;" e indique a nova cor "

70 PRINT '' INK RND*5;"
Se quiser alterar o "
80 PRINT INK RND*5;"     caracter,
carregue em "; INK 2; FLASH 1;"H"; IN
K RND*5; FLASH 0;" e indique o novo caracter "

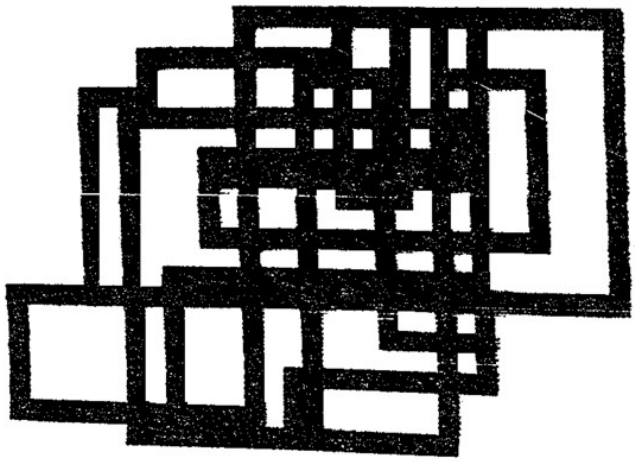
85 PRINT '' INK RND*5;" Se
quiser cintilação carregue em "
86 PRINT INK 2; FLASH 1;"F"; I
NK RND*5; FLASH 0;"      (para desligar
carregue em )          ";  INK 2; FLASH 1;"
N"; FLASH 0;")"
90 PRINT '' INK 1; FLASH 1;"
  CARREGUE EM QUALQUER TECLA  "
100 PAUSE 10000: CLS
110 LET x=10: LET y=10: LET b=0
120 LET c=2: LET h$="■"
130 PRINT AT x,y; INK c; FLASH
b;h$
140 LET y=y+(INKEY$="8")-(INKEY
$="5")
145 IF y=-1 THEN LET y=0
146 IF y=32 THEN LET y=31
```



```
150 LET x=x+(INKEY$="6")-(INKEY
$="7")
155 IF x=-1 THEN LET x=0
156 IF x=22 THEN LET x=21
160 IF INKEY$="c" THEN GO SUB 2
00
170 IF INKEY$="h" THEN GO SUB 3
00
175 IF INKEY$="f" THEN LET b=1
177 IF INKEY$="n" THEN LET b=0
180 GO TO 130
200 INPUT  "Escreva nova cor"; c

210 RETURN
300 INPUT    "Escreva novo caracter"; $ h

310 RETURN
```



WORDSQUARE

Neste programa, indicam-se algumas palavras que o computador esconde depois numa rede, cujas dimensões dependem do comprimento da palavra mais comprida da lista. Se lhe parecer muito difícil descobrir as palavras, o computador pode fazê-lo em seu lugar, indicando-as em cor inversa.

A palavra mais comprida da lista não deve ter mais de 18 letras, senão a rede não caberá no visor. Podem-se indicar cerca de 20 palavras de comprimento variável, obtendo a rede em 5 a 10 minutos. Uma lista de palavras muito longa pode provocar períodos de espera bastante frustrantes.

É boa ideia indicar as palavras por ordem descendente de comprimentos, pois apressar-se-á assim a execução. É muito interessante observar a execução deste programa.

```
  1 REM Wordsquare
  2 REM     Por J. Eliott
  5 BORDER RND*6: PAPER RND*6:
INK 9: CLS
 10 PRINT   " Se não deseja ver "
 20 PRINT   " as respostas carregue em
";  FLASH 1;"n";  FLASH 0;"' "
 30 PRINT   " agora. Senao carregue em qualquer
 tecla. "
 50 LET r$=INKEY$: IF r$="" THE
N GO TO 50
 60 CLS
 65 POKE 23692,-1
 70 PRINT AT 0,10;"WORDSQUARE";
AT 0,10; OVER 1;"          ";;;
 80 INPUT   " Quantas palavras? "          ;a
 90 INPUT   " Indique palavra mais comprida "
b$
100 DIM c$(a,LEN b$)
110 LET c$(1)=b$
120 FOR c=2 TO a
130 INPUT     Indique número da palavra
;c);d$
140 IF LEN d$>LEN b$ THEN PRINT
 AT 20,0; FLASH 1;  " COMPRIMENTO NÃO
ACEITÁVEL";. GO TO 130
```



```
160 LET c$(c)=d$
190 NEXT c
199 REM     A linha seguinte contém 22 espaços
200 PRINT AT 19,0;"
                      "
210 LET d=LEN b$+2
220 FOR e=1 TO d: FOR f=1 TO d
230 PRINT AT e,f;"*"
240 NEXT f: NEXT e
250 DIM h$(d,d)
260 FOR q=1 TO a
270 LET j$=c$(q)
280 PRINT AT 19,0;j$
310 LET x=INT  (RND*d)+1: LET y=
INT  (RND*d)+1
320 LET z=INT  (RND*3): LET w=IN
T  (RND*3)
330 IF z=0 AND w=0 THEN GO TO 3
20
360 IF z=2 THEN LET z=-1
370 IF w=2 THEN LET w=-1
380 DIM k(LEN j$,2)
390 FOR l=1 TO LEN j$
400 IF j$(l)=" " THEN GO TO 460
410 LET x=x+z: LET y=y+w
430 IF x<1 OR x>d OR y<1 OR y>d
THEN GO TO 270
440 IF (NOT h$(x,y)=" ") AND (N
OT  (h$(x,y)=j$(l))) THEN GO TO 2
70
445 LET k(l,1)=x: LET k(l,2)=y
450 PRINT AT 19,l-1; INVERSE 1;
INK 9;j$(l)
460 NEXT l
470 FOR m=1 TO LEN j$
480 IF j$(m)=" " THEN GO TO 500
485 LET h$(k(m,1),k(m,2))=j$(m)
490 IF r$="n" THEN GO TO 500
495 BEEP .01,RND*10-RND*20: PRI
NT AT k(m,1),k(m,2);j$(m)
500 NEXT m: NEXT q
510 REM     A linha seguinte contém 15 espaços
520 PRINT AT 19,0;"
               "
530 FOR n=1 TO d: FOR p=1 TO d
535 BEEP .01,RND*10-RND*20
540 IF NOT h$(n,p)=" " THEN GO
TO 565
```

```
 550 LET p$=CHR$ (INT (RND*26)+9
7)
 560 PRINT AT n,p;p$: GO TO 570
 565 PRINT AT n,p;h$(n,p)
 570 NEXT p: NEXT n
 580 PRINT AT 19,10; FLASH 1;"
Terminado "
 590 PRINT AT 20,0; FLASH 1;"
Carregue em qualquer tecla para obter as respostas.";
 600 LET q$=INKEY$: IF q$="" THE
N GO TO 600
 610 FOR n=1 TO d: FOR p=1 TO d
 620 IF h$(n,p)=" " THEN GO TO 6
40
 630 PRINT AT n,p; INVERSE 1; IN
K 9;h$(n,p)
 640 NEXT p: NEXT n
 650 PRINT AT 19,10;"
                    "
```


Terminado.
Carregue em qualquer tecla para obter as respostas

WORDSQUARE

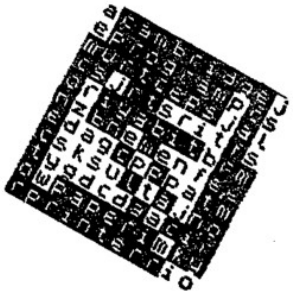
```
acambridgej
eprogrampcs
murtcepsjal
csjrtsritfs
oriyabitbrm
nzbnemenfee
edagcpppatm
csksultajno
tygdcdaacir
owpaperimky
rprinterrio
```



Terminado.
Carregue em qualquer tecla para obter as respostas

WORDSQUARE





PAPEL DE PAREDE

Este programa, escrito por Mark Charlton, utiliza o seu nome (ou qualquer outra cadeia alfanumérica com até 16 caracteres de comprimento) para produzir um colorido padrão de papel de parede.

```
 10 REM     Papel de Parede
 20 REM @ M Charlton
 30 INPUT " Indique o seu nome      ";a$
 40 LET a$=a$+" "
 45 IF LEN a$<16 THEN GO TO 40
 47 LET a$=a$( TO 16)
 48 LET b$=a$
 50 BORDER 7: PAPER 7: CLS
 90 FOR h=1 TO 16
100 FOR a=-16 TO 16
145 IF a=0 THEN GO TO 160
150 PRINT INK (h/3);a$(ABS a);
160 NEXT a
180 LET a$=a$(2 TO )+a$(1)
190 NEXT h
195 POKE 23692,-1
200 GO TO 90
```



ANAGRAMAS

Este programa, escrito por Ken Mahogany, mostra a flexibilidade do Spectrum no tratamento de cadeias alfanuméricas. Quando é executado, indica-se uma palavra (por exemplo o seu nome), e o computador produz todas as combinações possíveis das letras que o constituem. O exemplo de execução mostra alguns anagramas obtidos a partir do nome do autor.

```
 10 REM     Anagramas
 20 REM @K Mahogany
 30 INPUT INK 2;  "Indique uma palavra
";: LINE a$
 35 POKE 23692,-1
 40 LET n=LEN a$: DIM a(n)
 45 FOR x=1 TO 7
 50 LET a(1)=INT (RND*n)+1
 60 FOR z=2 TO n
 70 LET a(z)=INT (RND*n)+1
 80 FOR j=1 TO z-1
 90 IF a(j)=a(z) THEN GO TO 70
100 NEXT j: NEXT z
110 LET b$=""
120 FOR b=1 TO n
130 LET b$=b$+a$(a(b))
140 NEXT b
150 PRINT TAB 4; PAPER x; INK 9
b$
160 NEXT x
170 GO TO 45
```

```
Maghnyao
ayMagnho
aaonyghM
oMnhayga
ghaayoMn
aahygoMn
ghnaMyoa
yahMnoga
hoynMaga
Maygoahn
```

```
aognhyMa
ngoayaMh
aoMnghya
gayonaMh
Mahoyagn
yaaonhgM
ohMaayng
ahaMgony
nagaoyMh
ahnMyoga
nahMygao
```

AVIÃO DE CAÇA

Você é o piloto de um avião a jacto. Deve alinhar o aparelho inimigo no visor, usando as teclas '5' a '8', que deslocam na direcção das sectas nelas representadas. Destrói o inimigo carregando em "F". O aparelho inimigo não se mantém parado à espera que você o destrua; o leitor ver-se-á forçado a contrariar os seus movimentos evasivos. Como só pode controlar o seu próprio avião, o inimigo parecerá mover-se no sentido contrário àquele em que o jogador se move.

O jogo é apresentado aqui a um nível de principiante. A velocidade pode ser aumentada retirando a linha 150 (indicador de tempo). O programa foi escrito por Paul Toland.

```
  1 REM      Execute as linhas 2 a 4 antes de escrever
o resto do programa
  2 FOR t=0 TO 7: READ a
  3 POKE USR "a"+t,a: NEXT t
  4 DATA 24,60,126,255,126,90,1
53,24
 10 LET t=0
 20 LET x=INT (RND*32): LET y=I
NT (RND*22)
 30 PRINT AT 10,14; INK 1; FLAS
H 1;">  <"
 40 IF t<>INT (t/2)*2 THEN GO T
O 80
 50 IF x>-1 AND x<32 AND y>-1 A
ND y<22 THEN PRINT AT y,x;" "
 60 LET y=y+INT (RND*3)-1
 70 LET x=x+INT (RND*3)-1
 80 LET a$=INKEY$
 85 IF x>-1 AND x<32 AND y>-1 A
ND y<22 THEN PRINT AT y,x;" "
 90 LET x=x+(a$="5")-(a$="8")
100 LET y=y+(a$="7")-(a$="6")
115 REM    Para obter o caracter da linha 120 escreva 'a'
 em modo gráfico
120 IF x>-1 AND x<32 AND y>-1 A
ND y<22 THEN PRINT AT y,x;"▟"
130 IF a$="f" THEN BEEP .01,0
140 IF a$="f" AND x=15 AND y=10
 THEN GO TO 170
150 LET t=t+1: PRINT AT 0,0; IN
K 2;t
160 GO TO 30
170 FOR v=1 TO 10
180 PRINT AT 10,15;"■": BEEP .0
1,-10-v
190 PRINT AT 10,15;" ": BEEP .0
1,v
200 BORDER RND*7
210 NEXT v
220 PRINT AT 18,0; INK 2;"
Conseguiu em ";t;"    segundos ! "
230 PRINT ' FLASH 1;"    Novo jogo?"
240 IF INKEY$="" THEN GO TO 240
250 IF INKEY$="y" THEN CLS : RU
N
```

MAIOR / MENOR

Este programa é um jogo muito simples, no qual se tenta prever se a carta seguinte terá um valor maior ou menor do que a anterior. Se a previsão estiver errada, perdem-se 10 pontos; se estiver certa, somam-se dez pontos.

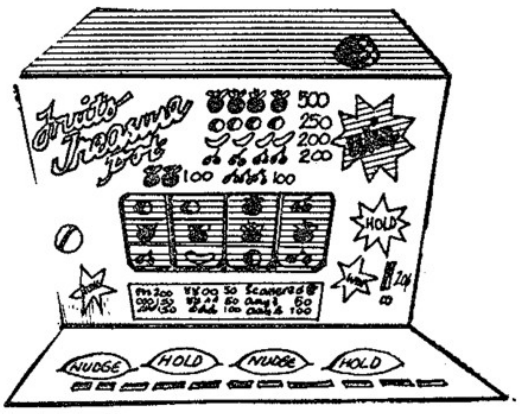
```
  1 REM      Maior / Menor
  2 REM      Por D. Harwood
 10 LET p=100: POKE 23692,-1
 20 PRINT AT 21,0; INK RND*5;"
MAIOR / MENOR ";  OVER 1;AT 21,0;"___
_____________"
 30 PRINT '' INK RND*5;" Vou
  indicar um número, e você deve prever se o seguinte será
maior ou menor?
 40 INPUT INK 1; FLASH 1;
 " Indique gama de valores dos "'" números ";
a; INK 1; FLASH 1;"  a ";b
 50 LET x=INT (RND*b+a)
 55 PRINT '' INK 1; FLASH 1;"
 Você tem ";p;"  pontos. "
 57 IF p=0 THEN GO TO 190
 60 PRINT ' INK RND*5;"  O número é:
 ";x
 70 INPUT "  Julga que o numero seguinte será
 maior ou menor?"
  '' INK 1; FLASH 1;"  (H — maior,
L — menor) ";a$
 80 LET y=INT (RND*b+a)
 90 IF x<=y THEN LET b$="  Maior
"
100 IF x>=y THEN LET b$="  Menor "
110 IF a$="h" AND x<=y OR a$="l
" AND x>=y THEN GO TO 150
120 PRINT INK RND*5;"  Errado, o
número era "; FLASH 1;b$
130 PRINT INK RND*5;" Tinha o valor";y
140 LET p=p-10: LET x=y: GO TO
55
150 FOR t=10 TO 30: BEEP 1/t,t:
NEXT t
160 PRINT INK RND*5;"  Correcto, o
número seguinte era "; FLASH 1;b$
170 PRINT INK RND*5;" Tinha o valor";y
180 LET p=p+10: LET x=y: GO TO
55
190 CLS : POKE 23692,-1
200 FOR t=1 TO 10
210 FOR g=0 TO 15
220 PRINT INK RND*5;TAB g;"
Perdeu todos os pontos "
225 BEEP .01,g+t*2: BORDER RND*
5
230 NEXT g
240 NEXT t
```


JACKPOT

Este programa custa 5 escudos por jogo, e pode-se ganhar 8 ou 15 escudos. Começa-se com 100 escudos, e a quantia de que se dispõe é continuamente apresentada no visor. É também possível parar uma ou mais das três rodas. O programa foi escrito por Daniel King.

```
  1 LET t$="nnn": DIM a$(3): LE
T b$="OX*▗▄▖▝▀▘": LET p=100
  2 PAPER 7: BORDER 7: INK 9: C
LS
 10 PRINT AT 3,12; PAPER 1; INK
 9;"JACKPOT";AT 5,12; INK 2;"▄▄▄▄
▄▄▄";AT 6,12; INK 2;"▌ ▌ ▐▗ ▌";A
T 7,12;"▀▀▀▀▀▀▀▀▀"
 20 PRINT AT 6,12; INK 2;"▌ ▌ ▐
 ▌";AT 7,12;"▀▀▀▀▀▀▀▀▀";AT 8,12;"▀
▀▀▀▀▀▀▀"
 30 PRINT '' FLASH 1; INK 1;"
 Carregue em R para um jogo. Custo: 5$ "
 40 PRINT ''" Dois pagam 8$... Três pagam 15$."
 45 IF INKEY$<>"r" THEN GO TO 4
5
 50 LET p=p-5
```

```
  55 PRINT AT 16,0; INK RND*5;"M
oney:";p;" "
  80 IF t$(1)<>"h" THEN LET a$(1
)=b$(INT (RND*7)+1)
  90 IF t$(2)<>"h" THEN LET a$(2
)=b$(INT (RND*7)+1)
 100 IF t$(3)<>"h" THEN LET a$(3
)=b$(INT (RND*7)+1)
 110 FOR f=1 TO 3
 120 FOR j=1 TO 7
 125 IF t$(f)="h" THEN GO TO 160
 130 PRINT AT 6,11+2*f;b$(j)
 140 NEXT j
 150 PRINT AT 6,11+2*f;a$(f)
 160 NEXT f
 170 IF a$(1)=a$(2) AND a$(1)=a$
(3) THEN LET p=p+15: BEEP .5,0:
BEEP .5,0: BEEP .5,0: GO TO 190
 180 IF a$(2)=a$(3) OR a$(1)=a$(
2) THEN LET p=p+8: BEEP .5,0: BE
EP .5,0
 185 PRINT AT 7,13; INK 2;"■";AT
 7,15;"■";AT 7,17;"■"
 190 LET t$="nnn"
 192 PRINT AT 16,0; INK RND*5;"M
oney:";p
 195 IF RND>.3 THEN GO TO 45
 200 PRINT INK 1; FLASH 1;AT 7,1
3;"H";AT 7,15;"H";AT 7,17;"H"
 210 INPUT INK 1;" Carregue em "; FLA
SH 1;"H"; FLASH 0;" Se quiser para uma roda;
senão carregue em                ";; FLASH 1;"
N"; FLASH 0;" "; LINE t$
 220 IF t$(1)="h" THEN PRINT AT
7,13; PAPER 1;"H": GO TO 230
 225 PRINT AT 7,13; INK 2;"■"
 230 IF t$(2)="h" THEN PRINT AT
7,15; PAPER 1;"H": GO TO 240
 235 PRINT AT 7,15; INK 2;"■"
 240 IF t$(3)="h" THEN PRINT AT
7,17; PAPER 1;"H": GO TO 250
 245 PRINT AT 7,17; INK 2;"■"
 250 PAUSE 100
 260 GO TO 50
```

COLOURMIND

Este programa baseia-se no jogo Mastermind, na qual o computador escolhe quatro cores aleatórias e o utilizador deve adivinhar essas cores. Quando lhe for pedido, escreva o número correspondente à cor pretendida. Ser-lhe-á então dito quantas adivinhou na posição correcta (negro) e quantas adivinhou na posição errada (branco).

Nenhuma cor é repetida entre as quatro escolhidas.

```
  2 REM      Por D. Harwood
  4 DIM a(4): DIM b(4)
  5 LET g=1
  6 LET a$=CHR$ 8+CHR$ 8+CHR$ 8
+CHR$ 8+CHR$ 8+CHR$ 8+CHR$ 8+CHR
$ 8+CHR$ 8+CHR$ 8+CHR$ 8
  7 POKE 23692,255
 10 OVER 1: PRINT AT 21,8;"
COLOURMIND";
```

```
  20 PRINT a$;"_____"
  30 PRINT '' " Vou escolher quatro cores. Você"
  40 PRINT "    deve adivinhá-las em"
  50 PRINT " vinte tentativas. "
  60 PRINT '' "    Carregue em qualquer tecla!"
  70 PAUSE 10000
  80 FOR t=1 TO 4
  90 LET a(t)=INT (RND*8)
  95 IF a(t)=a(1) AND t<>1 OR a(
t)=a(2) AND t<>2 OR a(t)=a(3) AN
D t<>3 OR a(t)=a(4) AND t<>4 THE
N GO TO 90
 100 NEXT t
 110 CLS
 120 LET s=0: LET r=0: INPUT ("
 Escreva número da cor ";g;" ");g$
 130 LET b(1)=VAL g$(1): LET b(2
)=VAL g$(2)
 135 LET b(3)=VAL g$(3): LET b(4
)=VAL g$(4)
 140 PRINT ' INK b(1);"■ "; INK
b(2);"■ "; INK b(3);"■ "; INK b(
4);"■"
 150 LET g=g+1
 160 FOR t=1 TO 4: FOR l=1 TO 4
 170 IF a(t)=b(l) AND t<>l THEN
LET s=s+1
 180 NEXT l: NEXT t
 190 FOR t=1 TO 4
 200 IF b(t)=a(t) THEN LET r=r+1
 210 NEXT t
 220 PRINT "    ■ negras -";r;"
brancas - ";s
 230 IF r=4 THEN GO TO 260
 240 IF g=20 THEN GO TO 270
 250 GO TO 120
 260 CLS : PRINT AT 11,0; FLASH
1; INK 4;"    Acertou na cor! ": GO TO 280
 270 PRINT AT 11,0; INK 4;"
 Pouca sorte. Esgotou as suas vinte tentativas"
 280 PRINT AT 13,0; FLASH 1; INK
 a(1);"■ "; INK a(2);"■ "; INK a
(3);"■ "; INK a(4);"■ "
 290 FOR g=1 TO 20: BEEP .2,RND*
20: NEXT g
```


Este programa, escrito por Tim Hartnell, produz um jogo bastante bom.

Indicamos os nossos movimentos escrevendo a letra da parte inferior do quadrado alinhado pela posição da nossa peça, e o número que se encontra na posição correspondente do lado direito; em seguida carrega-se em ENTER. Repete-se depois este procedimento para o quadrado para onde desejamos deslocar a peça.

O computador faz os seus saltos múltiplos automaticamente. Depois de você capturar uma peça, o computador pedir-lhe-á que salte de novo. Se puder fazê-lo carregue em qualquer tecla e em ENTER, e ser-lhe-á oferecido um novo movimento. Se não, carregue apenas em ENTER e o computador fará o seu movimento.

O jogo pode terminar de três maneiras. Ou o computador aceita a derrota quando julga que a sua posição é desesperada, ou proclama a sua vitória ao capturar todas as peças do jogador; pode acontecer também que só conceda a vitória depois de o jogador lhe ter capturado todas as peças. O leitor verificará que, apesar de a máquina jogar com bastante lentidão no início, a velocidade aumenta à medida que o jogo prossegue.

A minha pontuação é 0, a sua é 0.



```
  10 REM ********************
  20 REM *     DAMAS      *
  30 REM *  _ Por Tim Hartnell  *
  40 REM ********************
  50 GO SUB 940
  60 LET Z=24
  70 LET Q=0
  80 REM **    Dobrar peças          **
  90 FOR G=69 TO 72: IF A(G)=-1
THEN LET A(G)=-2
 100 NEXT G
 110 GO SUB 720
 120 REM *    Spectrum captura *

 130 IF A(Z)=99 OR A(Z)<1 THEN G
O TO 220
 140 IF Z<28 AND A(Z)=1 THEN LET
 A(Z)=2
 150 LET Y=1
 160 IF A(Z+X(Y))<0 AND A(Z+2*(X
(Y)))=0 THEN LET Q=X(Y)
 170 IF A(Z)=2 AND A(Z-X(Y))<0 A
ND A(Z-2*X(Y))=0 THEN LET Q=-X(Y
)
 180 IF Q<>0 AND Z+2*Q>23 THEN G
O TO 250
 190 LET Q=0
 200 IF Y=2 THEN GO TO 220
 210 LET Y=2: GO TO 160
 220 LET Z=Z+1
 230 IF Z<73 THEN GO TO 130
 240 IF Q=0 THEN GO TO 340
 250 LET A(Z+Q)=0: LET A(Z+2*Q)=
A(Z): LET A(Z)=0
 260 LET Z=Z+2*Q: LET COM=COM+1:
 GO SUB 720
 270 LET Q=0: LET Y=1
 280 IF A(Z+X(Y))<0 AND A(Z+2*(X
(Y)))=0 THEN LET Q=X(Y)
 290 IF A(Z)=2 AND A(Z-X(Y))<0 A
ND A(Z-2*X(Y))=0 THEN LET Q=-X(Y
)
 300 IF Q<>0 AND Z+2*Q>23 THEN G
O TO 250
 310 IF Y=1 THEN LET Y=2: GO TO
280
 320 GO TO 480
 330 REM *    Spectrum não captura *

 340 LET U=0
```



```
 350 LET Z=24+INT (RND*49): LET
U=U+1
 360 IF A(Z)=1 OR A(Z)=2 THEN GO
 TO 380
 370 GO TO 350
 380 LET Y=1
 390 IF A(Z+X(Y))=0 AND (A(Z+2*X
(Y))>-1 AND A(Z+2*X(Y)+1)>-1 AND
 A(Z+2*X(Y)-1)>-1) THEN LET Q=X(
Y)
 395 IF A(Z+X(Y))=0 AND U>250 TH
EN LET Q=X(Y)
 400 IF A(Z)=2 AND A(Z-X(Y))=0 A
ND (A(Z-2*X(Y))>-1 AND A(Z-2*X(Y
)+1)>-1 AND A(Z-2*X(Y)-1)>-1 OR
U>600) THEN LET Q=-X(Y)
 410 IF Q<>0 THEN GO TO 450
 420 IF Y=1 THEN LET Y=2: GO TO
390
 430 IF U<1000 THEN GO TO 350
 440 PRINT INK 2; FLASH 1;"
  Desisto do jogo     !": STOP
 450 LET A(Z+Q)=A(Z): LET A(Z)=0
 470 GO SUB 720
 480 PRINT '''TAB 6; INK 2;"
Indique o seu movimento
 490 PRINT TAB 5; INK 2;"
(por exemplo A9 e ENTER"
 500 PRINT TAB 4; INK 2;"
    e B6 e ENTER)."
 510 BEEP .2,RND*10
 520 INPUT FLASH 1,"  De ?   ";A$:
BEEP .2,RND*10: INPUT FLASH 1,"
Para?";B$: FOR U=1 TO 2: LET Z=0
 525 PRINT AT 15,0;"


                    "
 530 IF U=1 THEN LET C$=A$
 540 IF U=2 THEN LET C$=B$
 550 LET Z=-24*(C$="G9")-25*(C$=
"E9")-26*(C$="C9")-27*(C$="A9")-
30*(C$="H8")-31*(C$="F8")-32*(C$
="D8")-33*(C$="B8")-37*(C$="G7")
-38*(C$="E7")-39*(C$="C7")-40*(C
$="A7")-43*(C$="H6")-44*(C$="F6"
)-45*(C$="D6")-46*(C$="B6")-50*(
C$="G4")
 560 IF Z<>0 THEN GO TO 580
 570 LET Z=-51*(C$="E4")-52*(C$=
"C4")-52*(C$="A4")-56*(C$="H3")-
```

```
57*(C$="F3")-58*(C$="D3")-59*(C$
="B3")-63*(C$="G2")-64*(C$="E2")
-65*(C$="C2")-66*(C$="A2")-69*(C
$="H1")-70*(C$="F1")-71*(C$="D1"
)-72*(C$="B1")
 580 IF W=1 THEN LET D=-Z
 590 IF W=2 THEN LET E=-Z
 600 NEXT W
 620 LET A(E)=A(D)
 630 LET A(D)=0
 650 IF ABS (D-E)>7 THEN LET A(D
+(INT (E-D)/2))=0: LET HUM=HUM+1
 670 GO SUB 720
 680 IF ABS (D-E)>7 THEN INPUT ;
 FLASH 1;" Pode saltar de novo? Se Sim,
 carregue em qualquer tecla e ENTER"
S)";U$: IF U$<>"" THEN GO TO 520
 690 IF HUM<12 AND COM<12 THEN G
O TO 60
 700 IF HUM=12 THEN PRINT FLASH
1; BRIGHT 1; INK 2; PAPER 6;"
          Ganhou! ": STOP
 710 IF COM=12 THEN PRINT FLASH
1; BRIGHT 1; INK 2; PAPER 6;"I h
              Venci-o! ": STOP
 740 FOR M=24 TO 72
 750 LET A(M)=145*(A(M)=1)+147*(
A(M)=2)+32*(A(M)=0)+144*(A(M)=-1
)+146*(A(M)=-2)+99*(A(M)=99)
 760 NEXT M
 770 PRINT AT 1,0; INK RND*7; PA
PER 9;" A minha pontuação é ";COM;",
  e a sua é   ";HUM
 780 PRINT AT 3,0;
 790 LET T=-2: FOR K=0 TO 3: FOR
 J=0 TO 3
 800 PRINT INK 2;"■"; INK 1;CHR$
 (A(72-J-13*K));
 810 NEXT J: LET T=T+1
 820 PRINT INK 6; PAPER 2;INT ((
J+K)/2)+T
 830 FOR J=0 TO 3
 840 PRINT INK 1;CHR$ (A(66-J-13
*K)); INK 2;"■";
 850 NEXT J: LET T=T+1
 860 PRINT INK 6; PAPER 2;INT ((
J+K)/2)+T
 870 NEXT K
 880 PRINT INK 6; PAPER 2;"ABCDE
FGH "
 890 FOR M=24 TO 72
```



```
 900 LET A(M)=(A(M)=145)+2*(A(M)
=147)+0*(A(M)=32)-(A(M)=144)-2*(
A(M)=146)+99*(A(M)=99)
 910 NEXT M
 920 RETURN
 930 REM **********************
 940 REM *   Inicializar variáveis   *
 950 DIM A(99): DIM X(2): LET X(
1)=-6: LET X(2)=-7
 960 FOR Z=1 TO 99: LET A(Z)=99
 970 IF Z<73 AND Z>55 AND NOT (Z
=67 OR Z=68 OR Z=60 OR Z=61 OR Z
=62) THEN LET A(Z)=1
 980 IF Z<54 AND Z>42 AND NOT (Z
=47 OR Z=48 OR Z=49) THEN LET A(
Z)=0
 990 IF Z<41 AND Z>23 AND NOT (Z
=34 OR Z=35 OR Z=36 OR Z=28 OR Z
=29) THEN LET A(Z)=-1
1000 NEXT Z
1010 LET COM=0
1020 LET HUM=0
1030 FOR J=0 TO 7
1040 READ A: READ B
1050 READ C: READ D
1060 POKE USR "A"+J,A
1070 POKE USR "B"+J,B
1080 POKE USR "C"+J,C
1090 POKE USR "D"+J,D
1100 NEXT J
1110 RETURN
1120 DATA 0,0,0,0,24,8,84,84
1130 DATA 60,20,124,124,126,42,1
24,84
1140 DATA 126,42,124,108,60,20,1
24,84
1150 DATA 24,8,254,170,0,0,0,0
```

RENUMERADOR

O último programa apresentado neste livro é uma rotina muito útil que permite renumerar linhas, mas não as instruções GO TO e GO SUB. Escreva a rotina a partir da linha 9990. O programa permite-lhe definir o primeiro número de linha e a diferença entre cada duas linhas.

```
9990 REM    Renumerador
9991 LET x=PEEK 23635+256*PEEK 2
3636
9992 INPUT "     Primeiro número de linha ";s
9993 INPUT "Diferença entre linhas";d
9994 IF PEEK (x+1)+256*PEEK (x)=
9990 THEN STOP
9995 POKE x,INT (s/256)
9996 POKE x+1,s-255*INT (s/256)
9997 LET s=s+d
9998 LET x=4+x+PEEK (x+2)+256*PE
EK (x+3)
9999 GO TO 9994
```


ÍNDICE

Introdução 7
Entre as Estrelas 9
Fastermind 13
Trailer 15
Bombardeiro 17
Protector 19
Letras (Letter Chase) 22
O ouro do Dragão 23
Paciência 26
Hangman 28
Reacção 29
Poesia 31
Bolas e cruzes 32
Música 35
Breakout 37
Zap 39
Centropóide 41
Perícia 42
Surge 43
Campo de minas 44
Billboard 46
Catch 48
Viagem entre as estrelas 49
Patrulha galática 50
Zombies 51
Morse 53
Inverso 55
Publicidade 57

Livro de cheques 59
Condutor 61
Utilitários (Cálculo de restos, distribuições normais, contagem, conversão de temperaturas, gráficos de temperatura, juros de um empréstimo) 63
Alphásort 68
Data Sort 68
Raízes quadradas 69
Probabilidades 70
Correlação/Regressão 72
Combinações e arranjos matemáticos 74
Números primos 76
Sistemas de equações 77
Solução de equações (usando o método de Newton) 78
Execução de figuras (Cones, Linhas rectas, Curvas, Curvas sinusoidais Representação de funções, Círculos) 79
Sombras 85
Etchasketch 87
Bloco de desenho 88
Wordsquare 90
Papel de parede 94
Anagramas 95
Avião de caça 96
Maior/menor 98
Jackpot 99
Colourmind 101
Jogo de Damas 103
Renumerador 108

Este livro acabou de se imprimir
em 1984
para
EDITORIAL PRESENÇA, LDA
na
Tipografia Nunes, Lda 4000 Porto