

# Think



## Bombenjagd mit Grips

Bevor eine versteckte Bombe entschärft werden kann, muß man sie erst einmal finden. In unserem Fall gehört eine Menge Scharfsinn und Logik dazu, damit es bei der Jagd nach den Bomben nicht zu bösen Überraschungen kommt. Machen Sie mit bei "Think", einem alle einfachen wie eindrucksvollen Denkspiel für alle CPCs.

Ziel des Spiels "Think" ist, in einem Spielfeld, das aus 36, 64, 100 oder 144 Feldern besteht, eine festgelegte Anzahl von Bomben zu finden und die entsprechenden Felder zu markieren. Die Anzahl der Felder und die Anzahl der Bomben hängen dabei vom gewählten Spiellevel ab. Dabei gilt:

Level	Felder	Bomben
Beginner	36	4
Amateur	64	9
Expert	100	20
Professional	144	48

Um die Bomben zu finden, kann man mit einem Cursor jedes Feld auf dem Spielfeld anspringen. Ein so gewähltes Feld kann auf zwei verschiedene Weisen behandelt werden. Ist man der Meinung, auf dem Feld befindet sich eine Bombe, sperrt man es mit einem Totenkopf, indem man die Leertaste drückt. Andernfalls deckt man das Feld durch <COPY> auf. Findet sich unter einem mit <COPY> aufgedeckten Feld eine Bombe, ist das Spiel unweigerlich zu Ende. In diesem Fall werden alle Felder

offengelegt, und man kann die Positionen der Bomben erkennen. Ist das aufgedeckte Feld "sauber", erscheint in seiner linken oberen Ecke eine Ziffer, die die Anzahl der direkt an das Feld angrenzenden Bomben zeigt. Diese Zahl bezieht sich auf alle Bomben rund um das gewählte Feld. Ist keine Zahl zu sehen, so sind alle Felder um das gewählte Feld herum bombenfrei, auch die diagonalen Nachbarn. Ein bereits aufgedecktes Feld kann logischerweise nicht mehr markiert werden. Die Sperrung mittels Totenkopf läßt sich jedoch durch nochmaliges Drücken der Leertaste rückgängig machen.

### Gesteuert wird per Menü

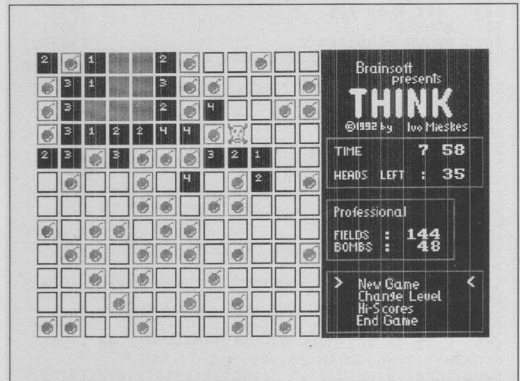
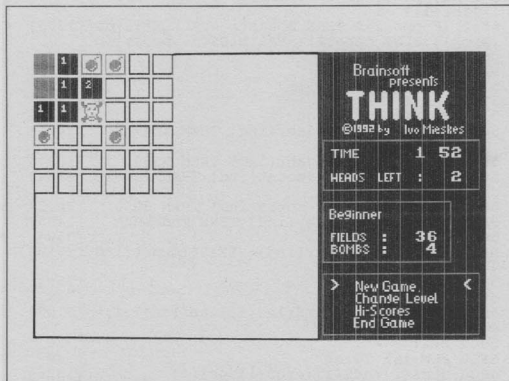
Während des Spiels läuft eine Uhr mit, die im Kontrollkasten auf der rechten Bildschirmseite unter dem Eintrag "time" eingeblendet ist. Hat man alle Bomben gefunden und als Freigabesignal <RETURN> gedrückt, stoppt die Zeit. Die schnellsten Bombensucher können sich in der High-Score-Tabelle verewigen.

Über ein kleines Menü wird das gesamte Spielgeschehen kontrolliert. Die vier Menüpunkte haben folgende Bedeutung:

- New Game** – startet ein neues Spiel
- Change Level** – wechselt den Schwierigkeitsgrad
- Hi-Scores** – eröffnet einen Blick auf die besten Plazierungen
- End Game** – bringt den Bombensucher auf den Boden des BASIC zurück

Die einzelnen Menüpunkte werden mit den Cursorstasten angewählt und mit <COPY> aktiviert. Die folgende Tabelle zeigt alle Tasten und deren Funktion innerhalb des Spiels "Think".

Taste	Bedeutung
Cursorstasten	Steuerung des Spielcursors und der Pfeile im Hauptmenü
<COPY>	dient zum Aufdecken eines Feldes und zum Anklicken einer Option im Hauptmenü
<SPACE>	kenzeichnet ein Feld mit einem Totenkopf oder löscht die Totenkopfmarkierung
<RETURN>	beendet das Spiel, wenn man meint, alle Bomben markiert zu haben. Es wird die Kontrollfunktion aktiviert, die die markierten Bomben mit den tatsächlich gesetzten vergleicht. Die Zeit wird gestoppt, und man kann sich gegebenenfalls in die High-Score-Liste eintragen
<DEL>	dient zum Abbrechen des Spiels. Es werden alle Bomben aufgedeckt, und man gelangt zurück ins Hauptmenü



"Think" ist eine Herausforderung an das Logikverständnis. Gefährliche Bomben müssen geortet werden. Verschiedene Levels vom Beginner ...

... bis zum Professional bringen eine spannende Langzeiterhaltung mit sich

## Zwei Listings sorgen für Bombenunterhaltung

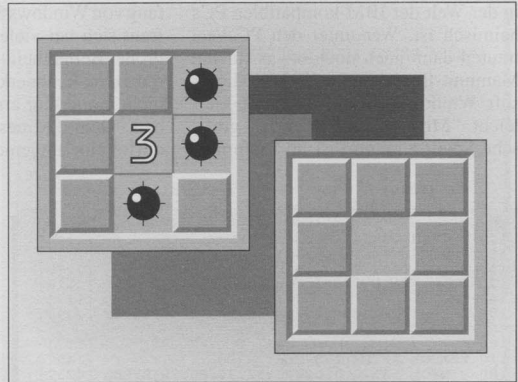
"Think" besteht aus zwei Listings. Das eine heißt THINK.PR.G, das andere SCORE.BAS. Letzteres ist ein kleines BASIC-Programm, das vor dem ersten Spiel gestartet werden muß. SCORE. BAS richtet eine ASCII-Datei namens THINK.SCO auf der Diskette ein, die die besten Ergebnisse für die Ewigkeit festhält. Diese Datei wird vom Hauptprogramm gebraucht. THINK.PR.G ist dank des leicht verständlichen BASIC-Codes in vielen Bereichen zu modifizieren. So lassen sich beispielsweise die Farben in der Zeile 800 den individuellen Wünschen anpassen. Wer Zeit beim Abtippen sparen will, kann die REM-Zeilen weglassen, sie werden von GOTO- oder GOSUB-Befehlen nicht direkt angesprochen.

### Im Falle einer Fehlermeldung

Wer ohne Diskettenlaufwerk auskommen muß, wird zu Programmbeginn vom Rechner eine Fehlermeldung erhalten."Think" erwartet die High-Score-Tabelle von der Diskette. In diesem Fall reicht es, den GOSUB-Befehl in Zeile 790 zu streichen. Das Spiel läuft dann zwar ohne High-Score-Tabelle, aber es läuft.

Wenn Sie vor dem ersten Start von "Think" das Zusatzprogramm SCORE.BAS nicht ausführen lassen, beschwert sich der Rechner, daß er die

*Eine Ziffer in einem Feld bezieht sich auf alle angrenzenden Felder einschließlich der diagonalen Nachbarn. Besitzt ein Feld keine Ziffer, so sind alle angrenzenden Felder bombenfrei*



High-Score-Tabelle nicht finden kann. Abhilfe: Starten von SCORE.BAS.

Im "Professional-Level" muß der gute alte CPC sehr viele Berechnungen durchführen. Dadurch wird unter anderem die Zeit des Spielfeldaufbaus verlängert. Es tut sich einige Zeit nichts auf dem Bildschirm. Etwas Geduld ist angesagt, nicht das Drücken der <ESC>-Taste. Auch die Berechnungen der Bombenanzahl benötigen in den hohen Levels etwas Zeit, auch hier gilt: Geduld ist besser als Abbruch.

### Langzeit-Logik-Unterhaltung

In der Regel läßt sich "Think" ohne großes Risiko lösen. Verschiedene Kniffe und Strategien entblößen immer mehr vom Spielfeld. Nur in seltenen

Fällen, wie beim Spielstart, muß man blind ins Blaue zielen – mit dem Risiko, eine Bombe zu treffen. Um den Geist von "Think" besser einfangen zu können, geben wir einige kleine Tips: Felder ohne Ziffern sind die beliebtesten Felder. Um sie herum können alle angrenzenden Felder gefahrlos geöffnet werden. Liegen um ein Feld mit einer 1 bereits sieben aufgedeckte Felder, muß im achten eine Bombe versteckt sein. Die Umgebungen aller "1"-Felder, die an ein Bombenfeld angrenzen, können gefahrlos enttarnt werden.

### Kleines "Who's Who?" der Variablen

Der Übersicht halber haben wir zum Abschluß noch einmal alle Variablen und ihre Bedeutung aufgelistet. Das erleichtert ein Verändern des Programms:

a,b,d,l,m,q,s,z	Zählervariablen
low, high, hi	legen Grenzwerte in einem DIMensionierten Feld fest
cl, mi, sec, ti	Zeitvariablen
t\$, t	Tastaturabfragen
name\$	High-Score-Variablen
l\$	Level-Variable
bm	Anzahl der zu suchenden Bomben im Spiel
xm, ym	doppelte Breite und Höhe des Spielfelds
x, y	aktuelle Cursorposition
kx, ky	Totenkopfkoordinaten
bx, by	Bombenkoordinaten
d(x,y)	Feldvariable (sie ist 0, wenn das Feld noch nicht aufgedeckt wurde und 1 im anderen Fall)

Übrigens: Es soll ja vorkommen, daß der eine oder andere unserer Leser auch

### Warum nicht mal BASIC?

Ivo Mieskes über sein Programm "Think"



Der Grund, warum ich dieses Spiel geschrieben habe, ist nicht nur die Spielidee, die mir gefallen hat, sondern vielmehr die Tatsache, daß immer weniger Spiele in reinem BASIC-Code verfaßt werden.

Oft hört man, BASIC sei viel zu langsam. Doch diesen "Schönheitsfehler" kann man nicht selten durch geschickte Programmierung umgehen. Sieht man sich das Programm "Think" an, so fallen

zum Beispiel die auf den ersten Blick umständlich programmierten Symboldefinitionen oder die vielen PRINT- und LOCATE-Anweisungen auf. Natürlich könnte man alle Daten schön sauber in DATA-Zeilen schreiben, dies würde aber die Ablaufgeschwindigkeit vervielfachen. Hier sieht man, daß die konventionellen Programmiermethoden die "sauberen" in puncto Geschwindigkeit bei weitem übertreffen können.

Dieses Programm ist absichtlich auf Schnelligkeit getrimmt und zeigt damit, daß auch BASIC "pur" eine durchaus taugliche Programmiersprache sein kann.

in der Welt der IBM-kompatiblen PCs heimisch ist. Wer unter den PC-Vertrauten dann auch noch etwas für die Mammut-Benutzeroberfläche Microsoft Windows übrig hat, kennt vielleicht "Minesweeper". Dieses grafische Spielprogramm ist im Lieferum-

fang von Windows 3.1 enthalten und erfreut sich bei vielen "Windowianern" großer Beliebtheit – denn auch mit teuren Büro-Rechnerlöten mag man ja nicht immer nur ernsthafte Dinge treiben. Dem "Minesweeper"-Kundigen dürfte eine ungemein starke Ähnlich-

keit mit unserem CPC-"Think" auffallen – will sagen: Wer sich zur fröhlichen Bombensuche aufmacht, braucht dazu nicht unbedingt immer die teuerste Computer-Umgebung!...

(Ivo Mieskes/sz)

```

100 ' THINK.BAS ' Hauptprogramm [1196]
110 ' CPC 664/6128(plus) [1532]
120 ' (c) 1992 Ivo Mieskes [1279]
130 ' & CPC International [2118]
140 ' [117]
150 ' --- Zeichensatz und Symbole definier [2399]
en
160 '
170 SYMBOL AFTER 135
180 SYMBOL 135,0,0,140,82,94,80,76:SYMB [4257]
L 136,0,0,0,115,74,74,74,74
190 SYMBOL 137,0,0,29,36,165,157,5,57:SYMB [3748]
OL 138,0,224,144,147,228,151,148,227
200 SYMBOL 139,0,240,128,135,228,132,132,2 [4827]
44:SYMBOL 140,0,4,4,28,164,164,164,156
210 SYMBOL 141,0,28,34,32,38,34,34,28:SYMB [3383]
OL 142,0,0,0,115,146,146,146,122
220 SYMBOL 143,0,0,0,70,169,175,168,166:SY [4750]
MBOL 144,0,96,80,86,100,68,68,68
230 SYMBOL 145,0,1,2,98,150,146,146,98:SYM [4346]
BOL 146,0,0,0,99,148,242,129,30,2
240 SYMBOL 147,0,4,0,52,69,37,21,100:SYMB [4035]
L 148,0,0,0,206,41,41,41,201
250 SYMBOL 149,0,1,1,57,73,73,73,61:SYMBOL [3963]
150,0,112,64,69,101,66,69,117
260 SYMBOL 151,0,0,0,99,84,103,68,67:SYMB [3866]
L 152,0,2,2,55,162,162,34,34
270 SYMBOL 153,0,48,72,75,122,74,74,74:SYM [3616]
BOL 154,0,0,0,71,169,169,169,167
280 SYMBOL 155,0,32,32,102,41,47,40,166:SY [3996]
MBOL 156,0,0,0,75,74,74,74,50
290 SYMBOL 157,0,72,74,72,122,74,74,74:SYM [2789]
BOL 158,0,14,16,16,204,2,2,28
300 SYMBOL 159,0,0,0,51,68,68,68,51:SYMBOL [3173]
160,0,0,0,51,164,167,164,35
310 SYMBOL 161,0,0,0,24,160,144,8,48:SYMB [3921]
L 162,0,58,66,67,66,66,66,58
320 SYMBOL 163,0,0,0,142,82,82,82,79:SYMB [4136]
L 165,0,0,1,114,74,73,72,75
330 SYMBOL 166,0,0,128,152,164,188,160,24: [4454]
SYMBOL 167,0,16,16,16,16,16,16,30
340 SYMBOL 168,0,0,0,101,149,245,133,98:SY [3990]
MBOL 169,0,2,2,50,74,122,66,51
350 SYMBOL 170,0,62,100,84,85,85,77,68:SYM [3387]
BOL 171,0,0,0,84,81,23,10,138
360 SYMBOL 172,0,14,17,16,19,17,14,84:SYMB [3337]
OL 173,0,0,0,57,73,73,73,61
370 SYMBOL 174,0,0,0,163,84,87,84,83:SYMB [3729]
L 175,0,0,0,0,128,128,0,0
380 SYMBOL 176,0,0,117,69,101,69,69,69,0:SYM [3436]
BOL 177,0,209,17,145,17,17,221,0
390 SYMBOL 178,0,198,41,36,34,41,198,0:SYM [4193]
BOL 179,0,113,74,114,74,74,113,0
400 SYMBOL 180,0,145,91,85,81,81,145,0:SYM [4738]
BOL 181,0,113,74,113,72,74,113,0
410 SYMBOL 182,0,128,64,0,128,64,128,0:SYM [3700]
BOL 183,0,75,74,74,123,74,75,0
420 SYMBOL 184,0,153,37,37,61,37,165,0:SYM [4685]
BOL 185,0,142,80,72,68,66,156,0
430 SYMBOL 188,0,142,136,136,140,136,238,0 [3804]
:SYMBOL 189,0,238,132,132,196,132,132,0
440 SYMBOL 190,0,125,17,17,17,17,17,0:SYMB [3589]
OL 191,0,69,109,85,69,69,69,0
450 SYMBOL 192,0,224,0,0,192,0,224,0:SYMB [3743]
L 193,0,128,128,202,170,198,2,12
460 SYMBOL 194,0,32,32,42,42,42,36,0:SYMB [3273]
L 195,0,2,3,66,162,162,66,0
470 SYMBOL 196,0,32,105,162,43,42,41,0:SYM [4138]
BOL 197,0,2,26,162,147,10,178
480 SYMBOL 198,0,0,145,170,57,160,147:SYMB [5040]
OL 199,0,0,128,0,0,128,0
490 SYMBOL 200,0,0,15,31,31,15,0,0:SYMBOL [3668]
201,0,0,240,248,248,240,0,0
500 SYMBOL 202,0,0,255,255,255,255,60,60:S [4420]
YMBOL 203,60,60,60,60,60,60,60,60
510 SYMBOL 204,60,60,60,60,60,24,0,0:SYMB [4880]
L 205,60,60,63,63,63,63,60,60

```

```

520 SYMBOL 206,60,60,252,252,252,252,60,60 [3999]
:SYMBOL 207,0,0,24,60,60,60,60,60
530 SYMBOL 208,0,0,24,60,60,62,62:SYMB [3266]
L 209,63,63,63,61,61,61,60,60
540 SYMBOL 210,60,60,188,188,188,252,252,2 [4134]
52:SYMBOL 211,124,124,124,60,60,24,0,0
550 SYMBOL 212,0,0,12,30,30,60,60,120:SYMB [3912]
OL 213,120,240,240,224,224,240,240,120
560 SYMBOL 214,120,60,60,30,30,12,0,0:SYMB [4869]
OL 215,0,0,15,31,31,15,1,1
570 SYMBOL 216,0,0,252,254,254,252,224,224 [3298]
:SYMBOL 217,1,1,1,1,1,1,1,1
580 SYMBOL 218,224,224,224,224,224,224,224 [4788]
:224:SYMBOL 219,1,1,1,1,1,0,0,0
590 SYMBOL 220,224,224,224,224,224,192,0,0 [4922]
:SYMBOL 221,0,0,230,148,148,228,132,128
600 SYMBOL 222,0,0,99,148,242,129,102,0:SY [4469]
MBOL 223,0,0,51,74,122,66,50,0
610 SYMBOL 224,16,56,147,84,82,81,86,0:SYM [4451]
BOL 225,240,136,136,243,138,138,138,242
620 SYMBOL 226,0,1,0,57,73,73,73,61:SYMBOL [5073]
227,0,0,0,113,74,73,72,75
630 SYMBOL 228,0,0,0,153,36,36,164,24:SYMB [4772]
OL 229,0,80,144,184,144,144,144,144
640 SYMBOL 230,0,0,0,192,128,128,128,128:S [2722]
YMBOL 231,255,255,255,251,243,251,251,241
650 SYMBOL 232,255,255,255,249,246,253,251 [3938]
:240:SYMBOL 233,255,255,255,241,254,249,25
4,241
660 SYMBOL 234,255,255,255,246,246,240,254 [4813]
:254:SYMBOL 235,255,255,255,240,247,241,25
4,241
670 SYMBOL 236,255,255,255,249,247,241,246 [4762]
:249:SYMBOL 237,255,255,255,240,254,253,25
1,251
680 SYMBOL 238,255,255,255,246,246,249,246 [4837]
:249:SYMBOL 239,0,127,64,64,64,64,64,64
690 SYMBOL 240,0,254,2,2,2,2,2:SYMBOL 24 [3662]
1,64,64,64,64,64,64,127,0
700 SYMBOL 242,2,2,2,2,2,2,254,0:SYMBOL 24 [4156]
3,0,127,64,64,64,67,71,79
710 SYMBOL 244,0,254,2,26,34,194,226,242:S [3464]
YMBOL 245,79,79,79,70,67,64,127,0
720 SYMBOL 246,242,210,178,98,194,2,254,0: [2567]
SYMBOL 247,0,96,231,248,56,18,22,16
730 SYMBOL 248,0,6,231,31,28,40,104,8:SYMB [4012]
OL 249,16,8,27,124,115,48,0
740 SYMBOL 250,136,144,16,216,62,206,12,0: [4640]
SYMBOL 251,255,255,255,255,255,255,255
750 SYMBOL 252,0,187,170,187,136,187,0,0:S [5117]
YMBOL 253,0,184,136,184,160,184,0,0
760 ' [117]
770 ' --- Menue aufbauen [1540]
780 ' [117]
790 GOSUB 2430:POKE 30009,1:POKE 30010,12: [5208]
ENV 1,15,-1,10:ENV 2,15,-1,2
800 MODE 1:BORDER 0:INK 0:INK 0:INK 1,14:INK 2 [6296]
,25:INK 3,17:WINDOW#2,25,38,1,24:WINDOW#3,
26,37,20,23
810 WINDOW#2,25,38,1,24:WINDOW#3,26,37,20, [2632]
23
820 PEN 3:PEN#1,3:PEN#2,0:PEN#3,0:PAPER#1, [4673]
0:PAPER#2,1:PAPER#3,1:CLS#2:CLS#3
830 LOCATE#2,4,2:PRINT#2,CHRS(225) :CHRS(22 [7048]
6) :CHRS(227) :CHRS(228) :CHRS(229) :LOCATE#2,
7,3:PRINT#2,CHRS(221) :CHRS(222) :CHRS(223) :
CHRS(224)
840 LOCATE#2,3,4:PRINT#2,CHRS(215) :CHRS(21 [13072]
6) :CHRS(207) :CHRS(207) :CHRS(207) :CHRS(208)
:CHRS(207) :CHRS(207) :CHRS(212) :LOCATE#2,3,
5:PRINT#2,CHRS(217) :CHRS(218) :CHRS(205) :CH
RS(206) :CHRS(203) :CHRS(209) :CHRS(210) :CHRS
(205) :CHRS(213)
850 LOCATE#2,3,6:PRINT#2,CHRS(219) :CHRS(22 [10327]
0) :CHRS(204) :CHRS(204) :CHRS(204) :CHRS(204)
:CHRS(211) :CHRS(204) :CHRS(214) :LOCATE#2,3,

```

```

7:PRINT#2,CHR$(164);CHR$(252);CHR$(253);CHR [7332]
RS(193)
860 LOCATE#2,8,7:PRINT#2,CHR$(194);CHR$(19 [7332]
5);CHR$(196);CHR$(197);CHR$(198);CHR$(199)
:LOCATE#2,2,9:PRINT#2,CHR$(190);CHR$(191);
CHR$(192)
870 LOCATE#2,2,11:PRINT#2,CHR$(183);CHR$(1 [5923]
84);CHR$(185);CHR$(32);CHR$(188);CHR$(189)
:LOCATE#2,2,16:PRINT#2,CHR$(176);CHR$(177)
;CHR$(178)
880 LOCATE#2,2,17:PRINT#2,CHR$(179);CHR$(1 [6693]
80);CHR$(181);CHR$(182):LOCATE#2,6,16:PRIN
T#2,CHR$(58)
890 LOCATE#2,6,17:PRINT#2,CHR$(58):LOCATE# [5066]
2,9,11:PRINT#2,CHR$(58)
900 GRAPHICS PEN 2 [948]
910 MOVE 390,280:DRAWR 208,0:DRAWR 0,-66:D [3850]
RAWR-208,0:DRAWR 0,66
920 MOVE 390,198:DRAWR 170,0:DRAWR 0,-80:D [2523]
RAWR-170,0:DRAWR 0,80
930 MOVE 390,102:DRAWR 208,0:DRAWR 0,-80:D [2914]
RAWR-208,0:DRAWR 0,80
940 GRAPHICS PEN 1 [949]
950 ' [117]
960 ' --- Spielfeld aufbauen [847]
970 ' [117]
980 GOTO 1950 [341]
990 CLEAR:WINDOW#0,1,24,1,25:CLS#0:xm=PEEK [6878]
(30010);ym=xm:bm=INT((1/(11-2*PEEK(30009)
)*xm/2)^2)
1000 DIM kx(bm+1),ky(bm+1),d(xm,ym),bx(bm) [3011]
,by(bm)
1010 GOSUB 1830:PEN 1:FOR y=1 TO ym STEP 2 [4229]
:FOR x=1 TO xm STEP 2
1020 LOCATE x,y:PRINT CHR$(239);CHR$(240): [3029]
LOCATE x,y+1:PRINT CHR$(241);CHR$(242);:NE
XT x,y
1030 ' [117]
1040 ' --- Bombenkoordinaten festlegen [1463]
1050 ' [117]
1060 RANDOMIZE TIME:FOR a=1 TO bm [1678]
1070 x=2*INT(RND*(xm/2)+1)-1:y=2*INT(RND*( [4068]
ym/2)+1)-1:bx(a)=x:by(a)=y
1080 FOR s=1 TO a-1:IF x=bx(s)AND y=by(s)T [4593]
HEN 1070 ELSE NEXT:NEXT
1090 FOR a=1 TO bm-1:FOR b=a+1 TO bm:IF bx [3786]
(a)=bx(b)GOTO 1110
1100 hi=bx(b):bx(b)=bx(a):bx(a)=hi:hi=by(b) [5379]
):by(b)=by(a):by(a)=hi
1110 NEXT b,a:GOTO 1320 [642]
1120 ' [117]
1130 ' --- Anzahl der Bomben neben einem F [3837]
eld ausrechnen
1140 ' [117]
1150 IF d(x,y)>0 THEN GRAPHICS PEN 2:RETUR [1974]
N
1160 GOSUB 1790:FOR a=1 TO z:IF kx(a)=x AN [4214]
D ky(a)=y THEN RETURN ELSE NEXT
1170 FOR a=1 TO bm:IF x=bx(a)AND y=by(a)TH [6653]
EN GOSUB 1540:CLEAR INPUT:GOTO 2000:ELSE N
EXT:GOSUB 1190:RETURN
1180 IF x=bx(a)AND y=by(a)THEN GOSUB 1540: [6244]
CLEAR INPUT:GOTO 2000:ELSE NEXT:GOSUB 1190
:RETURN
1190 FOR a=1 TO bm:IF bx(a)=x-2 THEN low=a [4332]
:GOTO 1220:ELSE NEXT
1200 FOR a=1 TO bm:IF bx(a)=x THEN low=a:G [3091]
OTO 1220:ELSE NEXT
1210 FOR a=1 TO bm:IF bx(a)=x+2 THEN low=a [3654]
ELSE NEXT
1220 FOR a=bm TO 1 STEP-1:IF bx(a)=x+2 THE [6160]
N high=a:GOTO 1250:ELSE NEXT
1230 FOR a=bm TO 1 STEP-1:IF bx(a)=x THEN [5131]
high=a:GOTO 1250:ELSE NEXT
1240 FOR a=bm TO 1 STEP-1:IF bx(a)=x-2 THE [4574]
N high=a:ELSE NEXT
1250 FOR a=low TO high:IF by(a)=y+2 OR by ( [5490]
a)=y OR by(a)=y-2 THEN d=d+1
1260 NEXT:low=0:high=0 [781]
1270 IF d=0 THEN PRINT#1,CHR$(251);CHR$(25 [5083]
1);CHR$(251);CHR$(251);:d(x,y)=1:GOSUB 153
0:RETURN
1280 PEN#1,1:PRINT#1,CHR$(230+d);CHR$(251) [6154]
;CHR$(251);CHR$(251);:d(x,y)=1:d=0:GOSUB 1
530:PEN#1,3:RETURN
1290 ' [117]
1300 ' --- Zeit anzeigen [926]
1310 ' [117]
1320 x=1:y=1:WINDOW#1,x,x+1,y,y+1:GRAPHICS [5487]
PEN 2:GOSUB 1530:ti=TIME
1330 cl=INT((TIME-ti)/300):mi=INT(cl/60):s [3970]

```

```

ec=cl-(mi*60)
1340 IF mi<10 THEN LOCATE#2,8,9 ELSE LOCAT [3626]
E#2,7,9
1350 PRINT#2,mi [381]
1360 IF sec<10 THEN LOCATE#2,11,9:PRINT#2, [5219]
CHR$(48)+CHR$(sec+48):ELSE LOCATE#2,10,9:P
RINT#2,sec
1370 ' [117]
1380 ' --- Tastaturabfrage [2297]
1390 ' [117]
1400 ts=UPPER$(INKEYS):IF ts=""THEN 1330 E [7008]
LSE t=ASC(ts):IF t<240 OR t>244 THEN:IF t<
>224 THEN:IF t<<" THEN:IF t<>13 THEN:IF
t<>127 THEN 1330
1410 IF t=240 AND y>2 THEN GRAPHICS PEN 0: [3686]
GOSUB 1530:y=y-2:GOSUB 1530
1420 IF t=241 AND y<ym-2 THEN GRAPHICS PEN 0: [2228]
GOSUB 1530:y=y+2:GOSUB 1530
1430 IF t=242 AND x>2 THEN GRAPHICS PEN 0: [3295]
GOSUB 1530:x=x-2:GOSUB 1530
1440 IF t=243 AND x<xm-2 THEN GRAPHICS PEN 0: [3490]
GOSUB 1530:x=x+2:GOSUB 1530
1450 IF t=224 THEN GRAPHICS PEN 2:GOSUB 15 [3109]
30:GOSUB 1150
1460 IF t=127 THEN GRAPHICS PEN 0:GOSUB 15 [3345]
30:GOSUB 1570:GOTO 2000
1470 IF ts="" THEN GRAPHICS PEN 1:GOSUB 16 [4179]
30:GOSUB 1530
1480 IF t=13 THEN GOSUB 1870 [1034]
1490 WINDOW#1,x,x+1,y,y+1:GOTO 1330 [2529]
1500 ' [117]
1510 ' --- Rahmencourer zeichnen [1045]
1520 ' [117]
1530 MOVE(2*x-2)*8,398-(2*y-2)*8:DRAWR 0,- [5829]
30:DRAWR 30,0:DRAWR 0,30:DRAWR-30,0:GRAPHI
CS PEN 2:RETURN
1540 ' [117]
1550 ' --- Game over + alle bomben zeigen [2146]
1560 ' [117]
1570 GOSUB 1780 [855]
1580 PEN 2:FOR a=1 TO bm:LOCATE bx(a),by(a) [4557]
):PRINT CHR$(243);CHR$(244)
1590 LOCATE bx(a),by(a)+1:PRINT CHR$(245); [4938]
CHR$(246):NEXT:PEN 1:RETURN
1600 ' [117]
1610 ' --- Bomben mit Kreuz kennzeichnen [3358]
1620 ' [117]
1630 IF d(x,y)>0 THEN GRAPHICS PEN 2:RETUR [1974]
N
1640 IF bm=z THEN FOR a=1 TO z ELSE 1660 [3385]
1650 IF kx(a)=x AND ky(a)=y THEN 1680 ELSE [4166]
NEXT a:GRAPHICS PEN 2:RETURN
1660 z=z+1:FOR a=1 TO z:CLS#1:IF kx(a)=x A [3053]
ND ky(a)=y THEN 1680 ELSE NEXT a
1670 GOSUB 1770:PEN#1,3:PRINT#1,CHR$(247); [6620]
CHR$(248);CHR$(249);CHR$(250);:kx(z)=x:ky(
z)=y:GOSUB 1830:GRAPHICS PEN 2:RETURN
1680 GOSUB 1770:PEN#1,1:PRINT#1,CHR$(239); [6753]
CHR$(240);CHR$(241);CHR$(242);:kx(a)=0:l=1
+1:PEN#1,3:GRAPHICS PEN 2:ELSE RETURN
1690 FOR q=1 TO 1:FOR s=1 TO z [1243]
1700 IF kx(s)=0 THEN FOR a=s TO z:kx(a)=kx [1894]
(a-1):ky(a)=ky(a+1):NEXT a
1710 NEXT s,q:g=z:z=0:FOR a=1 TO g:IF kx(a) [3769]
)>0 THEN z=z+1
1720 NEXT a:q=0:GOSUB 1830:RETURN [1287]
1730 ' [117]
1740 ' --- Sound [877]
1750 ' [117]
1760 SOUND 1,500,15,15:SOUND 1,400,15,15:S [3873]
OUND 1,300,15,15:SOUND 1,200,15,15:RETUR
N
1770 SOUND 1,120,5,15:SOUND 1,100,5,15:SOU [3222]
ND 1,80,5,15:SOUND 1,60,5,15:RETURN
1780 SOUND 1,900,0,15,1,15:RETURN [2234]
1790 SOUND 1,650,0,15,2:RETURN [2322]
1800 ' [117]
1810 ' --- Anzahl der Kreuze anzeigen [2247]
1820 ' [117]
1830 IF bm-z<10 THEN LOCATE#2,11,11:PRINT# [6336]
2,bm-z;:RETURN ELSE LOCATE#2,10,11:PRINT#2
,bm-z;:RETURN
1840 ' [117]
1850 ' --- Kontrolle ob Kreuze auch wirklic [4860]
h Bomben sind
1860 ' [117]
1870 IF z<>bm THEN RETURN [1014]
1880 GRAPHICS PEN 0:GOSUB 1530:FOR a=1 TO [3324]
z:FOR b=1 TO z
1890 IF bx(a)=kx(b)AND by(a)=ky(b)THEN v=v [1700]
+1
1900 NEXT b,a:m=5-PEEK(30009):hi=(PEEK(299 [5019]

```

