

SPRING VOGEL, S

Bild 5. Szene fünf ist nicht unlösbar, man braucht aber Ausdauer.

Bild 3. Die dritte Spielszene zeigt bereits neue Elemente wie Rutschbänder und Förderbänder

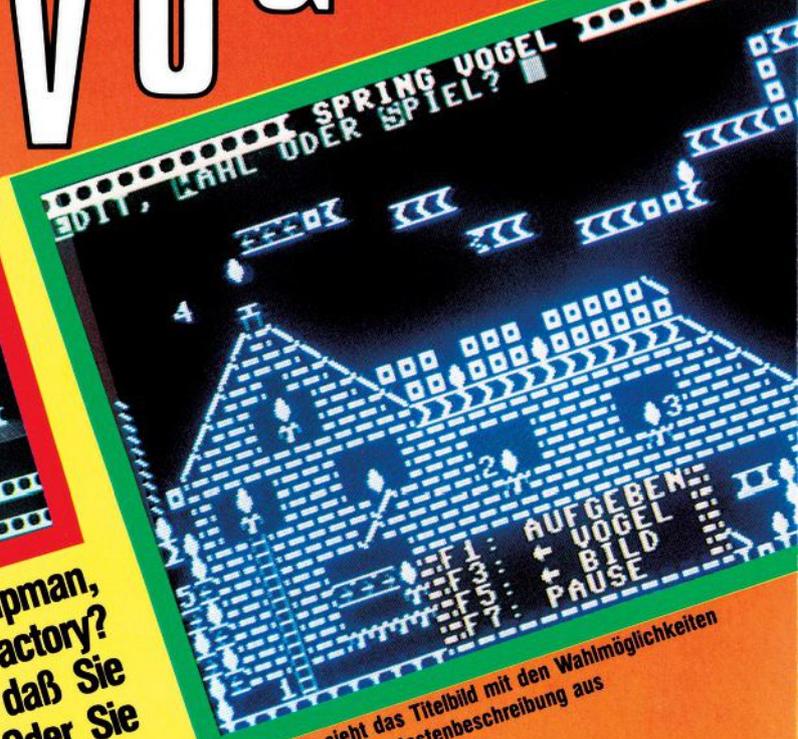
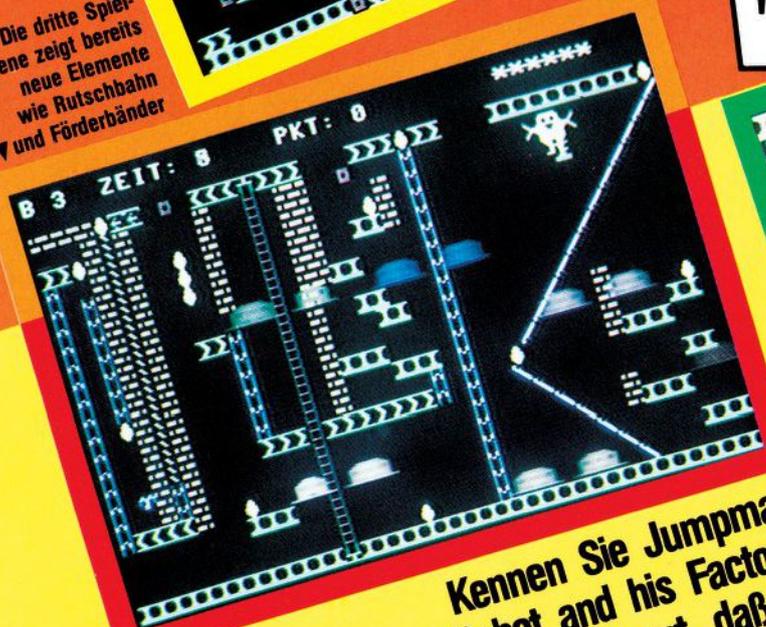


Bild 1. So sieht das Titelbild mit den Wahlmöglichkeiten und der Funktionstastenbeschreibung aus

Kennen Sie Jumpman, Miner 2049 oder Mister Robot and his Factory? Dann haben Sie sich sicherlich geärgert, daß Sie nie die letzten Bilder erreicht haben. Oder Sie haben sich über die eintönigen Spielszenen geärgert. Mit dem Listing des Monats wäre das nicht passiert. Spring-Vogel ist eine Kombination aus den drei Spielen. Eines kann jedoch nicht mehr passieren! Wird eines der Bilder langweilig, dann machen Sie sich einfach ein neues.

Tatsächlich kann man Spring-Vogel fast als Spielgenerator bezeichnen. Mit Spring-Vogel gelingt es Ihnen, sich jedes beliebige Spielfeld aufzubauen. Das klingt sehr vielversprechend, und das ist es auch.

Spring-Vogel ist ein Vertreter der Jump and Run-Kategorie. Damit lassen sich also keine Schieß- oder Abenteuerspiele erzeugen, aber innerhalb der Spring- und Laufgruppe bleibt kein Wunsch offen. Worum handelt es sich bei Spring-Vogel nun eigentlich? Zunächst die Story. Ein heftiger Sturm hat einen Vogel — unseren Helden

— mitsamt seinen Eiern aus dem Nest geweht. Durch den harten Aufprall auf die Erde hat er sich seine Flügel gebrochen. Er kann also nicht mehr richtig fliegen, sondern nur noch auf dem Boden laufen, hüpfen und springen. Die Aufgaben des Vogels (die genaue Klassifizierung bleibt Ihnen überlassen) besteht nun darin, in einem Labyrinth aus Aufzügen, Transportbändern, Seilen, Einbahnstraßen, Trampolinen, Rutschbahnen,

Gummiwänden, magischen Flügeln, gemeinen Vogelfallen und mißgestimmten Monstern alle Eier wieder einzusammeln. Gelingt es dem Vogel, die Eier in einer Spielszene aufzunehmen, muß er mit der nächsten Überraschung fertig werden; es waren nicht alle. Das nächste vom Winde verwehte Bild erwartet ihn mit weiteren schwierigen Aufgaben. Die Anzahl der Torturen für unseren leidgeprüften Helden bestimmen Sie selbst. Doch Vorsicht, unser Vogel verfügt nicht wie eine Katze über sieben, sondern »nur« über sechs Leben. Und diese sechs Leben sind schnell, durch Kontakt mit den Monstern oder Verzehrern einer roten Tollkirsche, ausgehaucht.

PRING

Bild 2.
Die erste Spiel-
szene ist relativ
leicht zu
bewältigen

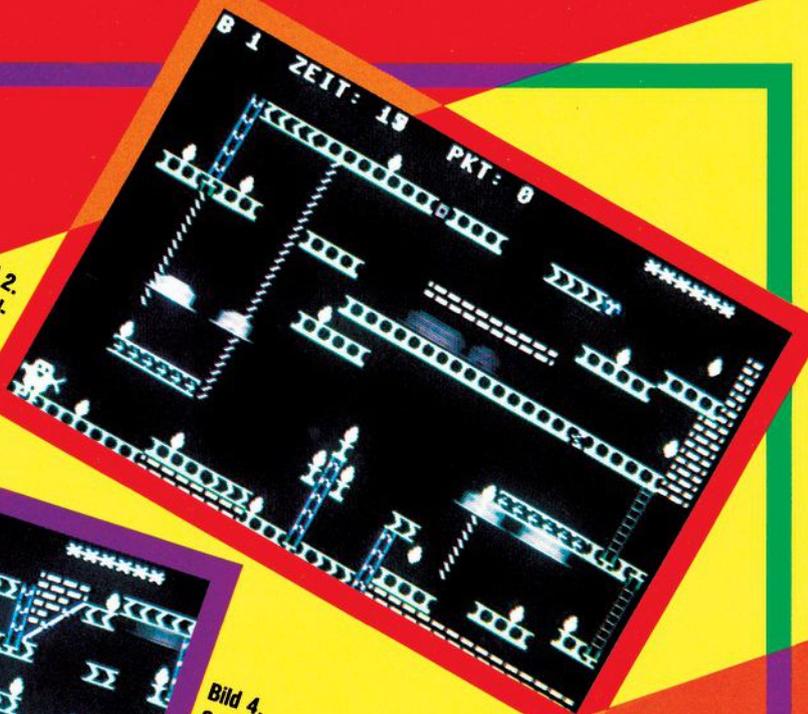
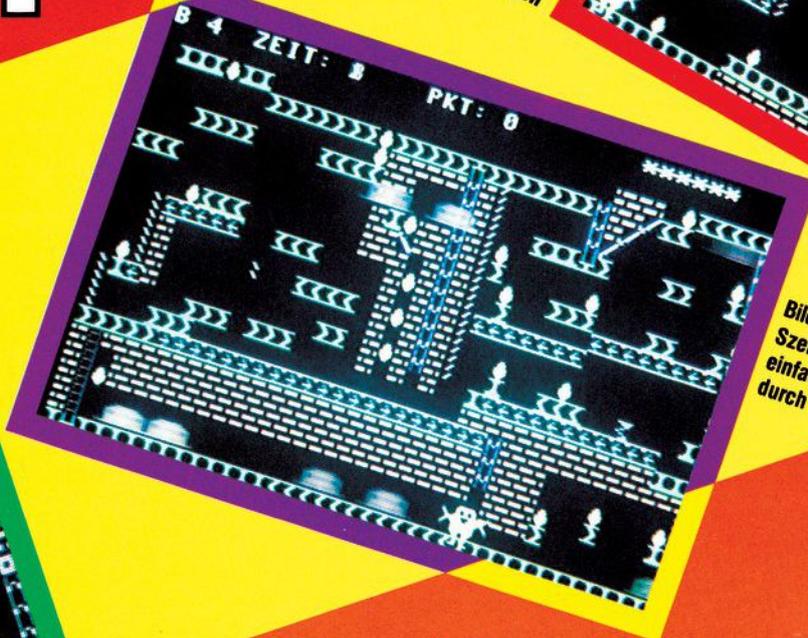


Bild 4.
Szene vier. Nicht
einfach und da-
durch reizvoll



Der

Aussicht auf 2000 Mark vor allem der Ärger über Spiele wie »Jumpman Jnr.« oder »Miner 2049er« verantwortlich, bei denen ich die letzten Bilder nie zu sehen bekam. Nach kleinen Änderungen an diesen Programmen, die die Helden mit Unsterblichkeit versorgten, war zwar das Ende einfach zu erreichen, aber leider wurden die Spiele dadurch auch schnell langweilig. Deshalb wollte ich ein ähnliches Spielprogramm entwickeln, daß diese Nachteile vermeidet.

Zunächst sollte es möglich sein, jedes einzelne Bild extra anzuwählen und vor allem sollten sich einfach neue Bilder erstellen lassen sobald die alten langweilig werden. Durch konsequente Parametrisierung und eine Spielroutine, die in der Art eines Interpreters die Bilder und Tabellen abarbeitet, entwickelte sich aus dieser Idee schließlich ein regelrechter »Generator« für solche »Jump an Run«-Spiele. Das Ergebnis liegt nun nach rund einem Monat Entwicklungszeit in Form von »Spring-Vogel« vor, einem Programm, mit dem Sie sicher viel Spaß haben werden. (Matthias Törk)

Der Sturz über mehr als vier Etagen ist ebenfalls für den flugunfähigen Vogel lebensbedrohend. Es sei denn, er findet die von seinem Tierschützer verstreuten »magischen Flügel«. Ausgestattet mit deren zauberhaften Fähigkeiten kann er in einem ökologischen Flecken beliebig umherfliegen. Durch Betätigen des Feuerknopfes wird er wieder in seinen behinderten Zustand zurückversetzt, erlangte allerdings vorher die Fähigkeit an unzugänglichen Stellen landen zu können. Diese magischen Flügel gestatten es, Bilder so aufzubauen, daß nur der taktisch gezielte Einsatz dieser Flughilfe es ermöglicht, ein Bild vollständig abzuräumen. Dabei kann es vorkommen, daß sich der Vogel in eine völlig aussichtslose Situation manövriert, und in einer Sackgasse landet. Auch hier bietet das Programm einen Ausweg an, die Funktionstasten. Diese können jederzeit (außer während der Flugphase) benutzt werden.

Die genaue Bedeutung können Sie der Beschreibung zum Listing entnehmen. Um zu sehen, welcher Vogel von welchem Spieler

der bessere ist, gibt es auch hier eine Punktezahl. Bei den bereits mitgelieferten Bildern ist diese in Zehnerschritten ab dem ersten Bild gestaffelt.

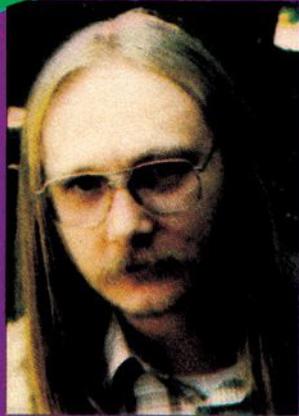
Entscheidungskriterien

Weshalb wurde dieses Programm zum Listing des Monats auserkoren? Das Hauptkriterium war die Realisierung eines Spielegenerators. Mit dem Editiermodus von Spring-Vogel lassen sich nahezu beliebig viele Bilder konstruieren. Wird ein Bild langweilig, so kann man ohne großen Aufwand ein neues mit noch größeren Schwierigkeiten entwickeln (versuchen Sie aber erst einmal, die beiden bereits existierenden Bilder 5 und 6 zu bewältigen). Der Phantasie sind keine Grenzen gesetzt.

Warum alleine spielen?

Vorstellbar ist ein Wettbewerb unter Freunden, wer das lustigste, das interessanteste, das schwierigste oder das raffinierteste Bild entwirft. Der Programmieraufwand hält sich für ein solch komplexes Programm in relativ bescheidenen Grenzen. Die vorgegebenen Bausteine bieten eine derartige Vielfalt an verschiedenen Bildern, daß jeder nach seinem eigenen Geschmack entwerfen kann. Insgesamt ein Programm, das durch Witz und Programmierkunst überzeugt.

(aa/Matthias Törk)



Gewinner

Sein Ärger brachte Matthias Törk 2000 Mark ein.

Ich bin am 18.4.1955 geboren und studiere derzeit Germanistik, Philosophie und Informatik. Mein erster Computer war ein VC 20, der dann Anfang 1983 durch einen C 64 ersetzt wurde.

Für die Entwicklung des Programms »Spring-Vogel« ist neben der verlockenden

Spring Vogel, spring

Das Listing des Monats zur Erzeugung von »Jump and Run«-Spielen für Commodore 64.

Bei dem ersten Start des Programms dauert es einige Zeit (bei weiteren Starts geht es dann schneller), bis sich das Programm mit der Frage »Edit, Wahl oder Spiel?« meldet.

Mit »W« kann man eines der sechs Bilder zum Spielen auswählen, mit »S« wird ein vollständiges Spiel ab Bild 1 gewählt.

Die Funktionstasten haben folgende Bedeutung:

- f1: Dieses Spiel aufgeben, zurück zum Menü
- f2: diesen Vogel opfern und mit dem nächsten Vogel neu an der Startposition beginnen
- f3: mit dem nächsten Bild weitermachen
- f5: Pause (weiter mit SPACE)

Für jedes Ei gibt es je nach Bild 10, 20 ... 60 Punkte. Wird ein Bild in einer bestimmten Zeit beendet, so erhält der Spieler einen Bonus, der um so größer ist, je schneller das Bild beendet wurde.

Der Editor

Der Editor dient dazu, eigene Spiele zu entwickeln. Mit ihm können neue Bilder erstellt werden, die im Programmtext selbst die alten Bilder überschreiben. Deshalb sollte das Programm vor dem Editieren, das neue Spiel nach dem Editieren unter neuem Namen gespeichert werden. Dazu einfach das Programm mit »STOP« abbrechen. Der Editor fragt zunächst, welches Bild geändert werden soll. Dann erscheint dieses Bild mit einem Cursor. Es stehen folgende Funktionen zur Verfügung:

—: die vorgenommenen Änderungen endgültig in den Programmtext übernehmen

f1: Editor verlassen, ohne Änderungen zu übernehmen. Das Bild bleibt erhalten, wie es vor dem Editoraufruf war. In beiden Fällen fragt das Programm, ob das Bild ausprobiert werden soll. Aus dem Spiel gelangt man mit f1 und E jederzeit in den Editor zurück.

Die Cursorsteuerung erfolgt wie gewohnt über die Cursorstasten. SHIFT CLR: Bild löschen

Die Startposition des Vogels wird durch »0« festgelegt. In jedes Bild können zwischen 0 und 7 Monster gesetzt werden, die sich horizontal auf einer Breite von acht Spalten hin und her bewegen. Das linke Ende dieser Bewegungsbahn kann mit den Tasten 1 bis 7 festgelegt werden.

Mit den Tasten A bis T können die 20 verschiedenen Bausteine, die zur Verfügung stehen, gesetzt werden. Der Vogel reagiert immer nur auf das Zeichen unter seinen Füßen. Bei Sprüngen reagiert er (außer bei Wänden) erst in der absteigenden Phase wieder auf Zeichen. Eine Übersicht über die Bausteine bietet die Tabelle 1. Dazu einige ergänzende Bemerkungen:

Die Bausteine A bis D sind durch verschiedene Bewegungsmöglichkeiten gekennzeichnet. Springen ist möglich:

- A: rechts, links
- B: rechts, links, runter
- C: rechts, links, rauf
- D: alle vier Richtungen

Bei E bis H ist eine Bewegung nur in jeweils eine Richtung und kein Sprung möglich.

Bei I bis N wird der Vogel automatisch bewegt. Er selbst kann nur springen.

Der Trampolin O bewirkt einen zufälligen Sprung nach rechts oder links.

Bei der Gummwand S wird der Vogel auf das letzte Zeichen, das weder Wand noch Luft war, zurückgeworfen.

Hinweise zum Eintippen

Da das Problem in die Interruptroutine eingreift und seinen eigenen Programmtext verändert, sollte es auf jeden Fall vor dem ersten Start abgespeichert werden.

Um das Abtippen des Listings zu erleichtern, wurden im Listing alle Steuerzeichen innerhalb von Printbefehlen durch leicht lesbare Worte entsprechend Tabelle 2 ersetzt. Wenn zum Beispiel im Listing <CD> steht, dann ist an dieser Stelle CURSOR DOWN einzugeben.

Um Tippfehler bei den zahlreichen DATA-Zeilen besser lokalisieren zu können, wurden diese in verschiedene Blöcke mit jeweils eigener Prüfsumme aufgeteilt.

Um das Eintippen der sechs Bilder zu erleichtern, wurden die Zeilen ab 60000 im Kleinschriftmodus gelistet. Hier ist für jeden Großbuchstaben (nicht bei den Ziffern!) die entsprechende Taste zusammen mit SHIFT zu drücken. Hier ist vor allem darauf zu achten, daß jeder PRINT-Befehl genau 39 Zeichen enthält. Fehlende Zeichen können zur Zerstörung des Programms führen, da der Editor diese Zeilen überschreibt.

Daß die jeweils 39 Zeichen genau mit dem Listing übereinstimmen, ist dagegen nicht so wichtig. Durch Abweichungen wird hier allenfalls die Spielbarkeit des vorgegebenen Spiels beeinflusst, ein Mangel, der jederzeit mit dem Editor behoben werden kann. Im Prinzip ist es auch möglich, auf das Abtippen der Bilder 2 bis 6 zu verzichten und an anderer Stelle einfach mit Hilfe des Bildschirmeditors fünfmal das erste Bild zu kopieren. Auf jeden Fall aber muß die Zahl der Zeilen, die Zeilennummer und die Länge der Zeilen mit dem Listing übereinstimmen.

Um das Eintippen der Bilder zu erleichtern, wurden diese in der vorliegenden Programmversion in den Programmtext selbst gelegt. Deshalb ist die Zahl der Bilder auf sechs (diese Zahl wurde gewählt, um die Tipparbeit in erträglichen Grenzen zu halten) festgelegt. Die Anzahl der Bilder kann aber folgendermaßen erhöht werden:

Der Variablen BM in Zeile 50070 ist die gewünschte Anzahl zuzuweisen. Für jedes weitere Bild sind dem Programm ent-

TASTE	: BAUSTEIN
A	: BODEN
B	: LEITER UNTERES ENDE
C	: LEITER OBERES ENDE
D	: LEITER MITTELSTÜCK
E	: SEIL HOCH
F	: SEIL RUNTER
G	: EINBAHNSTRASSE RECHTS
H	: EINBAHNSTRASSE LINKS
I	: TRANSPORTBAND RECHTS
J	: TRANSPORTBAND LINKS
K	: AUFZUG HOCH
L	: AUFZUG RUNTER
M	: RUTSCHBAHN RECHTS RUNTER
N	: RUTSCHBAHN LINKS RUNTER
O	: TRAMPOLIN
P	: LUFT, FREIER FALL
Q	: TOEDLICHE FALLE
R	: EI
S	: GUMMI-WAND
T	: MAGISCHE FLÜGEL

Tabelle 1. Die verschiedenen Bausteine und ihre Bedeutung

sprechend dem Schema der ersten sechs Bilder 22 PRINT-Zeilen, die jeweils 39 beliebige Zeichen ausgeben, und ein abschließendes RETURN, anzufügen. Die erste Zeilennummer eines Bildes berechnet sich nach der Formel $59900 + \text{Bildnummer} * 100 + 1$, zum Beispiel 60601 für Bild 7.

(Matthias Törk/aa)

Im Listing	: zu drückende Taste
<HO>	:HOME
<CLR>	:CLR
<RON>	:RVS ON
<ROF>	:RVS OFF
<CD>	:CURSOR DOWN
<CU>	:CURSOR UP
<CR>	:CURSOR RIGHT
<CL>	:CURSOR LEFT
<F1>...<F8>	:FUNKTIONSTASTEN
<BLK>...<YEL>	:FARBEN 0-7
[1]...[8]	:FARBEN 8-15, ZIFFER + COMMODORE-TASTE

Tabelle 2. Darstellung der Steuerzeichen im Listing

Die wichtigsten Variablen	
XS,YS	: Spritekoordinaten des Helden
XG,YG	: Koordinatengrenzen
OX,OY	: Koordinatenoffset Sprite/Zeichen
TG	: Zeitlimit
BM	: maximale Bildzahl
BI	: aktuelles Bild
UP ()	
DO (),RI ()	
LE (),DX ()	
DY ()	: Bewegungstabellen
JU ()	: Sprungtabelle
CO ()	: Farbtabelle
JP,JS	: Joystick
EZ,EM	: Anzahl Eier
SC	: Punkte
MZ	: Anzahl Leben
CP	: Cursorposition
BS	: Bildschirmadresse
FR	: Adresse Farb-RAM
KO	: Sprite-Kollisionsregister
V	: VIC-Chip
ZS	: Adresse Zeichensatz
EA	: Adresse Bilder
SI,W1,GL,GH	: Sound

Tabelle 3. Die wichtigsten Variablen von Spring-Vogel

Die wichtigsten Routinen	
50000:	INITIALISIERUNG
48000:	MENUE
49000:	EDITOR
45000:	AB BILD BI SPIELEN
1050:	EIN BILD SPIELEN
46000:	BILD BI INITIALISIEREN
500:	JOYSTICK
600:	FIGUREN BEWEGEN
40000:	SPRINGEN
11000:	EI
2000:	LUFT
12000:	TOEDLICHE FALLE
15000:	FLIEGEN

Tabelle 4. Die wichtigsten Routinen nach Zeilennummern aufgeschlüsselt

```

10 REM *** SPRING-VOGEL 9.3.LIST
11 REM *** M.TOERK      17.6.84
20 POKE53280,0:POKE53281,0
25 PRINT"<CLR>NICHT WUNDERN UND WARTEN!!"
"
30 GOSUB 50000:GOTO 48000
499 REM ** JOYSTICK **
500 JS=PEEK(JP):IF (JSAND16)=0 THEN GOSUB
B 40000:GOTO 600
505 IF (JSANDQ1)=Q0 THEN POKEW1,SW:IF UP(C
M) THEN YS=YS-Q8:GOTO 590
510 IF (JSANDQ2)=Q0 THEN POKEW1,SW:IF DO(C
M) THEN YS=YS+Q8:GOTO590
520 IF (JSANDQ4)=Q0 THEN POKEW1,SW:IF LE(C
M) THEN XS=XS-Q4:GOTO590
530 IF (JSANDQ8)=Q0 THEN POKEW1,SW:IF RI(C
M) THEN XS=XS+Q4
590 POKEW1,Q0
600 IFXS<OXTHENXS=OX
620 SYS(SR) Q0,XS,YS:RETURN
1000 REM *** MAIN LOOP **
1050 IF PEEK(TA) THEN1200
1055 IF PEEK(KO)ANDQ1 THEN CM=16:POKEKO,
0:GOTO1100
1060 CM=USR(XS-OX) ,YS-OY:IF CM=Z8 THEN
XS=8*PEEK(49426)+OX:YS=8*PEEK(49427)+OY
1070 IF CM>Q7 THEN XS=XS+DX(CM):YS=YS+DY
(CM)
1075 IF TI<TG THEN PRINT"<HO>",INT(TI/10
0)
1080 IF CM<Z4 THEN GOSUB 500:GOTO 1050
1100 ON CM-13 GOTO 3000,2000,12000,11000
,1050,15000
1200 GETB$:IFB$="<F1>" THEN RETURN
1210 IF B$="<F3>" THEN CM=16:GOTO 1100
1215 IF B$="<F5>" THEN 11040
1217 IF B$="<F7>" THEN POKE 198,0:WAIT19
8,1
1220 GOTO1050
2000 I=0:POKEW1,33
2010 I=I+1:YS=YS+4:IF I>8 OR YS>YG THEN
CM=16:POKEW1,0:POKESI+1,GH:GOTO 1100
2015 POKESI+1,40-2*I
2020 GOSUB 600:CM=USR(XS-OX) ,YS-OY:IF C
M=15 THEN 2010
2025 IFCM=18THENXS=FNX(XS)
2030 POKEW1,0:POKESI+1,GH:YS=FNX(YS):GOT
O 1050
3000 RX=2-4*INT(RND(1)*2):GOSUB40100:GOS
UB600:GOTO1050
10999 REM ** GEGENSTAND POSITIV, EI**
11000 POKE BS+40*PEEK(49427)+PEEK(49426)
,80:SC=SC+BI*10:EZ=EZ+1
11030 IF EZ<EM THEN PRINT"<HO>",,SC:GOTO
1050
11035 IF TI<TG THEN SC=SC+INT((TG-TI)/10
)
11040 BI=BI+1:IF BI>BM THEN RETURN
11050 GOSUB 46000:GOTO 1050
12000 MZ=MZ-1:IF MZ=0 THEN RETURN
12050 PRINT"<HO>",,SC,MID$("*****
",7-MZ,6)
12052 POKEW1,17
12055 FOR XS=XS TO XX STEP SGN(XX-XS):GO
SUB 600:POKESI+1,49+(XSAND1):NEXT
12057 FOR YS=YS TO YY STEP SGN(YY-YS):GO
SUB 600:POKESI+1,52+(YSAND1):NEXT
12060 XS=XX:YS=YY:POKEKO,0:POKEW1,0
12070 GOTO 1050
15000 POKEV+29,255:POKE BS+40*PEEK(49427
)+PEEK(49426) ,80:CM=3:XS=XS-10:SW=17
15050 JS=PEEK(56320) Listing »Spring-Vogel«

```

```

15052 IF (JSAND16)=0 THEN XS=XS+10:POKEV+2
9,254:GOSUB600:POKEK0,0:SW=129:GOTO1050
15055 IF YS<OY THEN YS=OY
15056 IF YS>YG THEN YS=YG
15057 IF XS>XG THEN XS=XG
15060 GOSUB 505:GOTO 15050
39999 REM **** JUMP ****
40000 IF CM>3 AND CM<8 THEN RETURN
40005 RX=0:IF (JSAND4)=0 THEN RX=-2
40010 IF (JSAND8)=0 THEN RX=2
40100 POKESI+6,16*15+13:POKEW1,17:POKEW1
,16:FOR I=0 TO 13
40120 POKESI+1,104-JU(I):YS=YS-JU(I):XS=
XS+RX:GOSUB 600
40130 CM=USR(XS-OX),YS-OY:
40140 IF CM=18 THEN XS=FNX(XS):GOTO 4030
0
40150 IF I<9 THEN POKEK0,0:GOTO 40250
40235 IFCM<>15OR (PEEK(K0)AND1) THEN4030
0
40250 NEXT
40300 IF CM<>15 THEN YS=FNY(YS)
40310 POKEW1,0:POKESI+6,16*15:POKESI+1,6
H:RETURN
45000 MZ=6:SC=0:GOSUB 46000
45100 GOSUB 1050:POKEV+21,0
45150 FORI=1TO3000:NEXT
45160 PRINT"<CLR><CD><CD><CD><CD><YEL>",
"<GRN>DAS WARS!!!":PRINT"<CD>ERREICHTE P
UNKTZAHL:<RON>",SC
45163 IFSC>HSTHENHS=SC
45165 PRINT"<CD>HIGH SCORE:<RON>",,HS
45170 PRINT"<RED>-----[71]:PRINT,"<CD>*** JUMP!!
***
45180 IF (PEEK(56320) AND 16)<>0 THEN 45
190
45190 RETURN
46000 SYS(49155) 0:EZ=0:XX=120:YY=120
46010 POKE W1,0:POKESI,6L:POKESI+1,6H:SW
=129:REM SOUND
46015 A=0:IFTI<TGTHENA=INT((TG-TI)/10)
46020 PRINT"<CLR><YEL> BONUS:",A,"<CL>
<CL><CL><CL>PKT:";SC,MID$("*****
",7-MZ,6):PRINT
46030 FORI=1TO7:POKEXH+I,1:POKEXL+I,170:
POKETY+I,0:NEXT
46050 ON BI GOSUB 60000,60100,60200,6030
0,60400,60500
46100 EM=0:FORI=80TO958:C=PEEK(BS+I):IFC
=80THENNEXT:GOTO46200
46105 IFC=82THENEM=EM+1
46110 IFC<64ANDC<85THENPOKEFR+I,CO(C-65)
:NEXT:GOTO46200
46120 POKEFR+I,0:C=C-48:XS=(I-INT(I/40)*
40)*8+OX:YS=INT(I/40)*8+OY
46121 IFC<00RC>7THEN46140
46130 POKETY+C,YS+12:POKE XH+C,XS/256:PO
KEXL+C,XSAND255:IFC=0THENXX=XS:YY=YS
46140 NEXT
46200 TI$="000000":TG=5000+2000*BI:YS=YY
:XS=XX
46220 PRINT"<HO>B";BI;"<CL> ZEIT:"; "
":RETURN
48000 REM *** MENUE ***
48040 POKEV+21,0:PRINT"<CLR>[71]<CD>":GOS
UB 60500
48041 PRINT"<CU><CU><CU><CU><CU>","[1]<C
R>#F1: AUFGEBEN#":PRINT,"<CR>#F3: _ VOGE
L #
48042 PRINT,"<CR>#F5: _ BILD #":PRINT,"
<CR>#F7: PAUSE #

```

```

48043 PRINT"<HO><YEL>***** SPRING
VOGEL *****"
48044 PRINT"<GRN><RON>E<ROF>DIT, <RON>W<
ROF>AHL ODER <RON>S<ROF>PIEL";
48045 INPUT B$:IF B$="E"THEN GOSUB 49000
:GOTO 48040
48050 IF B$="S" THEN BI=1: GOSUB 45000:G
OTO 48040
48060 IF B$<>"W" THEN 48040
48070 INPUT"<CLR><CD>WELCHES BILD SPIELE
N";B$:BI=VAL(B$):IFBI<1 OR BI>BM THEN 48
070
48080 GOSUB 45000:GOTO 48040
48999 REM *** EDITOR ****
49000 INPUT"<CLR><CD><CD><CD>WELCHES BIL
D";B$:BI=VAL(B$):IF BI<0 OR BI>BM THEN 4
9000
49020 PRINT"BITTE WARTEN":ZN=59900+BI*10
0:ZA=EA
49050 IF FNAD(ZA+2)<ZN THEN ZA=FNAD(ZA):
GOTO 49050
49055 POKE53281,1:PRINT"<CLR><CD>":POKE5
3281,0
49060 ON BI GOSUB 60000,60100,60200,6030
0,60400,60500
49069 REM FARBE SETZEN
49070 FOR I=80 TO 958
49071 C=PEEK(BS+I)-65:IFC=>0ANDC<20THENP
OKE55296+I,CO(C):NEXT:GOTO49095
49075 POKE55296+I,1:NEXT
49095 CP=80:PRINT"<HO>[7]ZEICHEN: <RON>A
-T<ROF> POS: <RON>0-7<ROF> <RON>CURS.<
ROF> <RON>CLR<ROF>"
49096 PRINT"AENDERUNGEN UEBERNEHMEN: <RO
N>_<ROF> QUIT: <RON>F1<ROF>"
49100 POKE 646,PEEK(CP+55296):POKE204,0
49105 GET B$:IF B$="" THEN 49105
49110 POKE207,0:POKE204,1:POKEBS+CP,PEEK
(BS+CP)AND127
49120 C=ASC(B$):IFC>64ANDC<85THENPOKE 64
6,CO(C-65):PRINTCHR$(C+32);:GOTO 49140
49125 IFB$=""ANDB$<"8"THENPRINT"<WHT>"
;B$;
49130 IF B$="<CD>" OR B$="<CU>" OR B$="<
CR>" OR B$="<CL>" THEN PRINTB$;
49140 IF PEEK(211)>38 THEN PRINT"<CL>";
49150 CP=PEEK(211)+40*PEEK(214):IF CP>=2
4*40 OR CP<80 THEN CP=80:PRINT"<HO><CD>"
49155 IFB$="<CLR>"THEN POKE53281,1:PRINT
"<CLR><CD>":POKE53281,0:GOTO 49095
49160 IF B$<>"_" AND B$<>"<F1>" THEN 491
00
49170 IF B$="<F1>" THEN RETURN
49200 B=BS+40:PRINT"<HO> **** BITTE
WARTEN *** "
49210 FOR I=0 TO 21:ZA=FNAD(ZA):A=ZA+6:B
=B+40
49220 FOR J=0 TO 38:C=PEEK(B+J)
49221 IF C>=48ANDC<=55THENC=C-32:GOTO492
30
49229 IF C<65 OR C>84 THEN C=80
49230 POKE A+J,C+32:NEXTJ,I
49350 INPUT"<CLR><CD>BILD SPIELEN (J/N)"
;B$:IF B$="N" THEN RETURN
49370 GOSUB 45000:RETURN
49499 RETURN
49999 REM
50000 REM ** INIT **
50010 CM=0:XS=0:YS=0:JS=0:OX=14:OY=36:T=
0:TG=2000
50020 Q0=0:Q8=8:Q4=4:Q1=1:Q2=2:Q7=7:Z8=1
8:Z4=14:JP=56320:SR=49249:TA=198

```

Listing
»Spring-Vogel«
(Fortsetzung)

```

50040 XH=49400:XL=49392:TY=49432
50042 DEFFNY(Y)=PEEK(49427)*8+DY
50045 DEFFNX(X)=PEEK(49426)*8+DX
50050 V=53248:BS=256*204:XG=320+OX:YG=20
0+DY
50060 KO=V+30:FR=55296:REM SPR.KOLLIS.RE
G / FARBRAM
50070 POKE 786,193:POKE 785,32:BM=6: REM
USR ADR.,MAX.BILDER
50090 DIM JU(13):FORI=0TO13:READJU(I):NE
XT
50100 DATA 5,4,3,2,1,1,0,0,-1,-1,-2,-3,-
4,-5
50215 FORI=BS+1017 TO BS+1023:POKEI,16:N
EXT:POKEBS+1016,14
50220 POKE V+23,0:POKE V+29,254:POKEV+39
,7
50230 DIMX(7),Y(7),RX(7),RY(1):FORI=0TO7
:X(I)=100:Y(I)=100:NEXT
50249 REM MAPRO-TABELLEN
50250 FORI=1TO7:POKEV+39+I,I:POKE49432+I
,30*I:POKE49392+I,35*I:POKE49400+I,0
50251 POKE49416+I,1:POKE49408+I,5+RND(1)
*45:NEXT:POKE49401,1
50252 POKE 49424,1:REM SB-FLAG
50410 DIM UP(20),DO(20),RI(20),LE(20),DX
(20),DY(20)
50422 RI(0)=1:LE(0)=1:RI(1)=1:LE(1)=1:UP
(1)=1:RI(2)=1:LE(2)=1:DO(2)=1
50437 UP(3)=1:DO(3)=1:RI(3)=1:LE(3)=1:UP
(4)=1:DO(5)=1:RI(6)=1:LE(7)=1:DX(8)=4
50477 DX(9)=-4:DY(10)=-8:DY(11)=8:DY(12)
=8:DX(12)=8:DY(13)=8:DX(13)=-8:DY(15)=4
50599 REM SOUND
50600 SI=54272:W1=SI+4
50620 POKE SI+24,15
50625 POKE SI+5,0:POKE SI+6,15*16+0
50630 GL=180:GH=18:A=0
50710 FORI=13TO16:FORN=0TO62:READ Q:A=A+
Q:POKE 49152+I*64+N,Q:NEXT
50720 IF A<>16572 THEN PRINT"DATAERROR":
STOP
50800 DATA0,255,0,1,255,128,1,153,128,1,
153,128,3,255,192,31,195,248
50805 DATA 63,227,252,103,243,230,195,24
9,195,131,253,193,3,255,192
50807 DATA 3,255,192,1,231,128,0,231,0,0
,102,0,0,102,0,0,102,0,1,231,128
50809 DATA ,,,,,,
50810 DATA0,255,0,1,255,128,129,153,129,
193,153,131,99,255,198,63,195,252
50815 DATA 31,199,248,7,207,224,3,159,19
2,3,191,192,3,255,192
50817 DATA 3,255,192,1,231,128,0,231,0,0
,102,0,0,102,0,0,102,0,1,231,128
50819 DATA ,,,,,,
50850 DATA 0,192,0,1,224,0,3,240,0,2,
208,0,2,208,0,3,240,0,3,240,0
50855 DATA 3,48,0,15,112,0
50857 DATA ,,,,,,,,,,,,,,,,,,,,,,
,,,,,
50870 DATA 0,192,0,1,224,0,3,240,0,2,20
8,0,2,208,0,3,240,0,3,240,0
50875 DATA 3,48,0,3,188,0
50880 DATA ,,,,,,,,,,,,,,,,,,,,,,
,,,,,
51000 DIM CO(19):FOR I=0 TO 19:READ CO(I
):NEXT
51010 DATA 5,5,5,2,10,8,13,13,5,5,4,4,4,
4,12,1,2,7,8,14
52000 REM EDIT-ANFANGSADR.
52010 DEFFNAD(X)=PEEK(X)+256*PEEK(X+1)

```

```

52020 EA=2049
52050 IF FNAD(EA+2)<59800 THEN EA=FNAD(E
A):GOTO 52050
58000 REM ZEICHENSATZ NACH #E0000 (ZS),
VIDERRAM NACH #CC00 (BS)
58020 ZS=14*4096:ZV=53248
58040 IF(PEEK(53272)AND254)=56 THEN RETU
RN
58050 POKE 53272,56:POKE 56576,148:POKE
648,204:BS=204*256
58055 PRINT"<CLR>" GEDULD":PRINT:GO
SUB 60400
58060 POKE 56334,0:POKE 1,51
58070 FOR I=0 TO 2047:POKE ZS+I,PEEK(ZV+
I):NEXT
58075 POKE 1,55:POKE 56334,1:A=0
58090 READ C:A=A+C:IF C>255 THEN 59010
58095 H=ZS+8*(C+65):FOR I=0 TO 7:READ C:
A=A+C:POKE H+I,C:NEXT
58098 GOTO 58090
58100 DATA 0,255,60,24,24,24,60,255,255
58110 DATA 1,66,66,66,66,255,24,24,255
58120 DATA 2,255,24,24,255,66,66,66,66
58130 DATA 3,66,66,126,66,66,66,126,66
58140 DATA 4,12,24,48,0,12,24,48,0
58150 DATA 5,48,24,12,0,48,24,12,0
58160 DATA 6,255,129,8,4,62,4,8,255
58170 DATA 7,255,129,16,32,124,32,16,255
58180 DATA 8,255,56,28,14,14,28,56,255
58190 DATA 9,255,28,56,112,112,56,28,255
58200 DATA 10,66,66,90,102,102,66,66,66
58210 DATA 11,66,66,102,102,90,66,66,66
58220 DATA 12,192,208,112,56,28,14,7,3
58230 DATA 13,3,7,14,28,56,112,208,192
58240 DATA 14,255,62,12,48,192,48,12,62
58250 DATA 15,0,0,0,0,0,0,0,0
58260 DATA 16,0,126,126,102,102,126,126,
0
58270 DATA 17,24,60,60,126,126,126,60,24
58280 DATA 18,0,223,223,0,0,251,251,0
58290 DATA 19,0,102,255,153,153,56,56,0
58299 DATA 999
59000 REM ** MAPROS **
59010 IF A<>14182 THENPRINT"DATA-ERROR I
N 58100FF":STOP
59014 A=0:FOR I=49152 TO 49378:READ N:PO
KE I,N:A=A+N:NEXTI
59016 IF A<>27365THENPRINT"DATA ERROR":S
TOP
59020 DATA 80,70,83,32,158,183,224,8,176
,31,189,219,192,45,21,208,141
59022 DATA 21,208,96,32,158,183,224,16,1
76,14,134,2,32,253,174,32,235
59024 DATA 183,165,21,201,2,144,3,76,72,
178,138,72,166,2,32,10,192
59026 DATA 189,219,192,45,16,208,141,16,
208,70,21,144,9,189,211,192,13
59028 DATA 16,208,141,16,208,138,10,170,
104,157,1,208,165,20,157,0,208
59030 DATA 166,2,189,211,192,13,21,208,1
41,21,208,96
59099 REM
59100 DATA 32,20,192,173,16,193,73,25
4,141,16
59102 DATA193,162,7,189,248,207,24,109
,16,193
59104 DATA157,248,207,202,16,243,169,7
,133,2
59106 DATA166,2,189,0,193,24,125,8,1
93,157
59108 DATA 0,193,201,64,144,11,189,8,
193,73

```

Listing »Spring-Vogel« (Fortsetzung)

