



*BoDaCT – Boulder Dash Construction Tool*

# Rockfords neue Höhlen

*Er gehört zur Diskettensammlung jedes Spiele-Fans wie das Endmonster zum letzten Level: der Game-Klassiker Boulder Dash. Mit unserem Tool kann man jetzt nach Herzenslust eigene Spielfelder entwerfen!*

**E**s gibt kaum Spiele-Freaks, die Rockford nicht kennen – dieses winzige, aber um so flinkere Kerlchen, das mit schlafwandlerischer Sicherheit jeden noch so gut versteckten Diamanten in unzähligen Minenstollen findet und einsackt. Als 1984 die Urversion dieses Geschicklichkeitsspiels für den C64 auf den Markt kam (Abb. 1), hockten selbst eingefleischte und abgebrühte Joystick-Zocker tagelang vor dem Bildschirm.

Viele Nachfolgeversionen wurden in der Zwischenzeit veröffentlicht (Boulder Dash II bis V). Ebenso hat man die Spielidee unzählige Male kopiert (z.B. Emerald Mine). Eines ist aber klar – und das betrifft im Prinzip jedes Computer-Game: Nach einiger Zeit kennt man die vertracktesten und schwierigsten Levels auswendig und weiß, wo's langgeht. Das vorher heißgeliebte Game wird plötzlich langweilig und verschwindet im hintersten Winkel der Diskettenbox.

Das muß nicht sein. Wir bieten Ihnen ein komplexes Programmwerkzeug, um beliebig viele neue Spielfelder zu entwerfen; mit mehr Diamanten oder noch verzwickteren Fallen.

Mit »BoDaCT« können Sie eigene Caves (Höhlen) kreieren und zu Spielen zusammenstellen. Die neu entworfenen Levels kann man an andere Computerfans weitergeben – sie fallen nicht unter die Copyright-Beschränkungen. Zum Spielen braucht man allerdings die Originalversion von Boulder Dash II (Abb. 2, erhältlich bei N. Erdem, c/o 64er-Magazin, Postfach 100518, 8000 München 1, Tel. 089/46 13-5020, Best.-Nr. 641102, 9,80 Mark).



[2] Bessere Grafik, raffiniertere Levels: Boulder Dash II

Der Level-Designer besteht aus drei separaten Programmen:

- EDITOR: Damit entwirft man eigene Höhlen und stellt sie zu einem Spiel zusammen,
- TESTER: ... Probespiel für selbstentworfenen Levels,
- LINKER: ... verbindet maximal acht Spiele zu einer Sammlung.

## Die Höhlen von Boulder Dash

Zunächst geben wir Ihnen einen Überblick zum Aufbau der Höhlen von Boulder Dash. Wir verwenden bewußt den Begriff »Höhlen«, da die Spielstufen in Caves (= Höhlen) und Levels (= Schwierigkeitsgrade) eingeteilt sind. Für jede Höhle gibt's also fünf Schwierigkeitsgrade; die Grundstruktur der Höhlen bleibt aber unverändert bestehen. Unterschiede macht lediglich die Zeit, die man zum Lösen der Levels braucht, die Anzahl der Diamanten, die man finden muß und die Position ausgewählter Spielelemente (Hindernisse, denen man im Spiel begegnet: Steine, Sand, Schmetterlinge, Diamanten usw.).

Peter Liepa, der Boulder-Dash-Programmierer, entwickelte beim Definieren der Caves eine gewisse Raffinesse: Bei den meisten C-64-Spielen stehen die fertigen Levels bereits in vollem Umfang im Speicher, bevor sie auf den Bildschirm geholt werden. Bei Boulder Dash undenkbar: Der Speicherbedarf wäre gigantisch. Jede Höhle besteht aus 40 x 22 Spielele-



[1] Damit fing's an: Urversion von Boulder Dash



menten mit jeweils fünf Levels. Insgesamt gibt's 16 Höhlen und vier Bonus-Caves. Summa summarum sind das 80 000 Byte (40 x 22 x 5 x (16 + 4)). Da ginge auch dem C64 die Puste aus!

Nimmt man die Boulder-Dash-Caves genauer unter die Lupe, erkennt man schnell, wie sie definiert wurden: Man findet häufig nur gerade Linien und rechteckige Spielelemente! Es ist also unkomplizierter, als man denkt: Eine Höhle besteht lediglich aus Rechtecken und Linien (auf die übrigen Elemente gehen wir später ein).

Ergänzt wird das Höhlenobjekt durch zusätzliche Infos wie z.B. Ausgangspunkt, Größenangaben und welches Spielelement zum Zeichnen verwendet wird. Dadurch läßt sich eine Höhle (normalerweise 880 Byte groß) in nur 212 Byte pressen. Wichtig: Boulder Dash hat sich bereits auf dieses Limit (212 Byte pro Cave) eingerichtet. Man kann's also weder ignorieren noch verändern.

Das sind aber noch nicht alle Unterschiede in den einzelnen Levels: Jetzt kommt der gesteuerte Zufall ins Spiel. Damit werden bis zu vier verschiedene Spielelemente in der Höhle verteilt, abhängig von einem bestimmten Anfangswert.

### Levels werden im Speicher aufgebaut

Damit haben Sie die wichtigsten Informationen, um die Struktur der Höhlen zu verstehen. Bevor man aber eine Höhle für Boulder Dash aktivieren kann, muß sie zunächst verdeckt im C-64-Speicher gezeichnet werden:

- die Höhle füllt sich mit dem Spielelement »Sand«,
- jetzt werden die Zufallselemente der Höhle verteilt,
- anschließend bildet sich ein Rand ums gesamte Spielfeld (Abb. 3).

Das ist die Basis, auf der die weitere Definition der Höhle aufbaut. Alle Objekte der Höhle werden nun in derselben Reihenfolge plziert, in der man sie krieert hat. Abb. 4 zeigt die Höhle ohne, Abb. 5 mit Zufallselementen.

### Der Editor

Genug der Theorie, kommen wir jetzt zur Praxis! Laden Sie den ersten Programmteil mit:

```
LOAD "EDITOR", 8
```

und starten Sie mit RUN. In Sekundenbruchteilen ist man im DOS-Modus des Editors. Hier lassen sich Spiele laden, speichern oder der Edit-Modus aktivieren.

#### DOS-Modus des Editors

Funktionen, die man per Tipp auf die entsprechende Taste aktiviert:

- <F8>: Editor-Oberfläche beenden und Rückkehr ins Basic,
- <#>: Laufwerksnummer ändern (von 8 bis 11). Künftige Diskettenbefehle beziehen sich jetzt auf die Floppy mit der angegebenen Nummer.
- <\$>: Directory anzeigen,
- <@>: beliebige Diskettenbefehle (inkl. Disk-Inhaltsverzeichnis),
- <E>: Editor-Modus aktivieren. Geben Sie bei Frage nach der Nummer den gewünschten Kennbuchstaben (A bis T) der Höhle an, die Sie im Editor bearbeiten möchten (also keine Zahl!).
- <F>: Höhlen färben. Auch hier muß man den Kennbuchstaben angeben.
- <H>: Handicaps berechnen (s. Beschreibung),
- <K>: Höhle kopieren,
- <L>: laden,
- <N>: löscht alle bislang definierten Höhlen nach Rückfrage,
- <S>: speichern,
- <T>: Höhlen vertauschen,

- <SPACE>: Anzeige des freien Speicherplatzes.

Alle Aktionen lassen sich mit <RUN/STOP> abbrechen, außer Speichern und Laden: Da genügt ein Tipp auf <RE-TURN> (quasi die Eingabe eines leeren Dateinamens).

#### Edit-Modus:

Den typischen Bildschirmaufbau sieht man in Abb. 6. Die oberen 22 Zeilen sind für die verkleinerte Darstellung der Höhle vorgesehen, darunter befinden sich zwei Info-Zeilen. Die letzte gilt als Statuszeile. Hier erscheinen alle Fehlermeldungen und Eingabe-Aufforderungen. In der unteren, rechten Ecke wird man über die aktuelle Cursor-Position informiert.

#### Infozeilen:

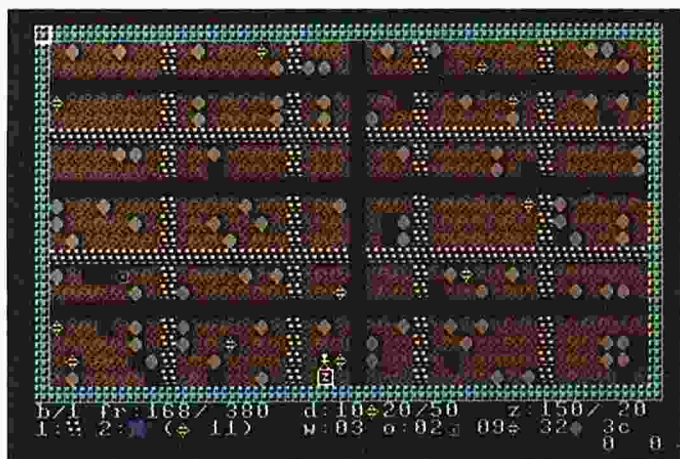
... dienen der Programm-Koordination. Beispiel:

```
H/L fr:HÖH/GES. d:DD PD/PZ z:ZSL/ZMS
```

```
1:E 2:E (ENNN) w:ZW o:P1E P2E P3E P4E
```

Die Übersetzung der Parameter ist simpler, als man denkt:

- H: ... ist die Bezeichnung der Höhle (A bis T), wobei die Nummern ab Q Bonushöhlen repräsentieren. Die Werte lassen sich mit <F3> / <F4> erhöhen bzw. reduzieren.
- L: Levelnummer (1 bis 5). Per <F1> / <F2> ändern!
- HÖH: Freier Speicherplatz dieser Höhle (maximal 212),
- GES: Freier Gesamtspeicherplatz (maximal 1289),
- DD: Anzahl der Diamanten (je mehr, desto schwieriger ist der Level zu lösen!). Ändern: mit <D>.
- PD: Punktwertung für jeden Diamanten (läßt sich mit <SHIFT D> regulieren),
- PN: Bonuspunkte für jeden zusätzlich gesammelten Diamanten (mit <SHIFT D> ändern).
- ZSL: Maximalzeit, in der man alle in der Höhle versteckten Diamanten finden und einsammeln sollte. Der jeweilige Wert läßt sich mit <Z> ändern.
- ZMS: Zeit für Mauer und Schleim. Die Zahl gibt an, wie lange die Zaubermauer (verwandelt Steine in Diamanten) aktiv bleibt und wie schnell sich der Schleim verbreitet (je kleiner die Werte, desto fixer). Ändern per Tastenkombination <SHIFT Z>.
- E: Spielelement. Mit den beiden Elementen 1 und 2 lassen sich die Zeichenbefehle ausführen und der Bildschirm mit den entsprechenden Mustern füllen. Tipp auf <1> bzw. <2> stellt die gewünschten Pixelinhalte ein.
- NNN: ... gibt an, wie oft das Element E in der Höhle auftauchen soll. Außerst praktisch, um Diamanten zu zählen. Die Werte lassen sich mit <SHIFT 1 bis 3> einstellen.
- ZW: ... ist die Zufallswurzelzahl. Davon hängt die Verteilung der genannten Zufallselemente ab. (ändern mit <W>, <+> oder <->). Der Wert erscheint hexadezimal.
- Pn: ... Wahrscheinlichkeitswert für das dahinter angegebene Spielelement. Man muß sie der Größe nach sortieren und kann sie mit <O> ändern. Achtung: Es dürfen niemals identische Werte vorkommen!



[3] Jede Höhle muß abgeschlossenen umrandet sein





[4] Höhle ohne ...

Dazu ein Beispiel:

Boulder Dash ermittelt für jede Bildschirmposition einen Zufallswert und checkt damit die Intervalle. Ist der Wert kleiner als P1 (z.B. \$0E, also 14), erscheint Element 1 (Mauer). Ist die Zahl größer als P1, aber kleiner als P2 (\$13), greift der Computer auf Spielelement 2 (Schmetterling) zurück.

#### Spielelemente:

Jeder Boulder-Dash-Fan kennt sie zur Genüge. Da die Höhlen im Editor jedoch stark verkleinert erscheinen, wollen wir sie hier nochmals erwähnen:

Die Elemente werden bei Auswahl der Zeichen (1 oder 2) auf dem Bildschirm (in der untersten Zeile) gezeigt. Von links nach rechts:

- freier Platz, Sand (braun), Mauer (weiß), Zaubermauer (gelb), Ziel (weiß), Randelement (türkis), viermal Firefly (blau), Stein (grau), Diamant (gelb), Rockford (weiß), wachsende Wand (violett), viermal Schmetterling (hellrot), Schleim (grün) und Schleimwand (blau).

Schmetterlinge und Fireflies gibt's jeweils viermal, da man für jede der vier möglichen Richtungen ein separates Element vorgesehen hat. Deshalb besitzen Fireflies eine dreieckige Spitze und Schmetterlinge einen Strich (in die Richtung, in die sie sich nach dem Spielstart bewegen sollen). Da es für jede Richtung ein gesondertes Element gibt, kann man diese Tierchen quasi lenken.

#### Editor-Anweisungen:

Die Editorbefehle teilt man in drei Gruppen ein:

- Zeichnen,
- Ändern und
- Einstellungen.

#### Zeichenbefehle:

Sechs Anweisungen kümmern sich ums Zeichnen der Objekte. Jeder Befehl beginnt an der aktuellen Cursor-Position und braucht eventuell noch weitere Parameterwerte. Braucht man zum Zeichnen eines Objekts mehr als einen Punkt, erzeugt man alle weiteren per Cursor-Tasten: Das Objekt erscheint unmittelbar auf der Editorfläche - Änderungen kann man also an Ort und Stelle begutachten. Man zeichnet Objekte mit dem Zeichenmuster 1. Für zwei Befehle braucht man zusätzlich noch Zeichen 2.

Die folgende Befehlsliste zeigt die Eingabeaufforderungen des Editors (in Klammern):

- <S>: ... setzt Zeichen 1.
- <L>: (nach Punkt). Zieht eine horizontale, vertikale oder diagonale Linie mit Zeichen 1.
- <R>: (nach Punkt). Zeichnet ein Rechteck (quasi einen Rand) mit Zeichen 1.
- <F>: (nach Punkt). Das Rechteck wird ebenfalls mit Zeichen 1 umrandet, aber mit Zeichen 2 gefüllt.
- <P>: (Abstand, nach Punkt). Erzeugt ein Raster mit Zei-

chen 1. Damit gibt man den horizontalen und vertikalen Abstand der Punkte an, mit »nach Punkt« die Ausdehnung des Rasters.

- <E>: (Abstand). Dieser Befehl ist mit Vorsicht zu genießen: Er durchforstet die gesamte, bisher gezeichnete Höhle nach Zeichen 1. Wird das Tool fündig, platziert es Zeichen 2 »Abstand« Positionen weiter rechts. Am äußersten rechten Rand geht's dann in der nächsten Zeile weiter. Will man also Zeichen 2 direkt unter Zeichen 1 platzieren, ist der Punkteabstand exakt »40«. Achten Sie außerdem darauf, daß der Rand ums Spielfeld nicht zerstört wird - notfalls müssen Sie ihn neu zeichnen. Es wäre nämlich äußerst ungewöhnlich für den Spielablauf von Boulder Dash, wenn Rockford die Höhle so ohne weiteres verlassen könnte!

#### Beispiel:

Abb. 5 zeigt ein Höhlenbeispiel. Jeder Zeichenbefehl wurde mindestens einmal verwendet. Aus Gründen der Übersichtlichkeit haben wir auf Zufallselemente verzichtet. In Abb. 6 sieht man die selbe Höhle, jedoch mit Zufallselementen.

Oben links erkennt man Rockford (den Helden der Boulder-Dash-Games), platziert per SETZ-Befehl <S>. Das Ziel befindet sich links unten. Neben Rockford sehen Sie ein Rechteck aus Mauern, per FÜLL-Anweisung <F> mit Steinen gefüllt. Es wird von einem anderen Rechteck überlappt, das sich aus Diamanten zusammensetzt - erzeugt per RAND-Befehl.

Als nächstes fällt die diagonale Linie, bestehend aus Fireflies, ins Auge. Unterhalb jedes Glühwürmchens entdeckt man jeweils einen Stein. Die Quader wurden per ERSETZ-Befehl und mit dem Abstand »40« erzeugt. Bleibt noch, das PUNKTE-Raster zu erwähnen, das unser Diamanten-Rechteck teilweise überlagert.

#### Änderungsbefehle:

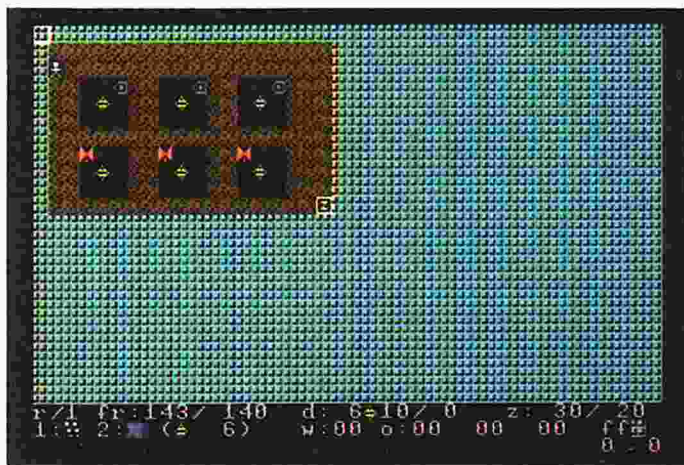
Objektentwürfe lassen sich ändern; dazu kennt der Editor entsprechende Anweisungen. Nur ein aktuelles Objekt läßt sich bearbeiten - man macht es zum momentan gültigen, indem man den Cursor darauf platziert. Das geht auch mit den Funktionen »Blättern« oder »Memo« (s. Beschreibungen dazu). Dann kann man sich die Cursor-Bewegung sparen.

- <DEL>: Löschen. Entfernt das aktuelle Objekt aus der Höhle.

- <I>: Undo. Macht die letzte Änderung rückgängig.

- <A>: Ändern (die Abfragen richten sich nach dem aktuellen Objekt). Man darf alle Eigenschaften verändern. Es erscheinen dieselben Abfragen wie beim Zeichnen des Objekts - mit einem Unterschied: Der Ausgangspunkt läßt sich verschieben. Mögliche Abfragen: »von Punkt«, Zeichen 1 und evtl. Nr. 2.

- <SHIFT A>: wie A. Das Objekt wird aber direkt an die aktuelle Cursor-Position geholt.



[5] ... und mit Spielelementen





[6] Typischer Bildschirmaufbau im Editor-Modus

- <B>: Blättern. Damit bewegt man sich durch die einzelnen Objekte. Zum Blättern benutzt man die Cursor-Tasten. Jedes gewählte Objekt hebt sich dadurch vom restlichen Bildschirm ab, daß es neu gezeichnet und der Cursor auf seine Ausgangsposition gesetzt wird. Mit <RUN/STOP> oder <RETURN> beendet man diese Funktion. <RETURN> bewirkt außerdem, daß das soeben hervorgehobene Objekt zum aktuellen gekürt wird. Der Cursor erscheint aber an der Position vor dem Blättern!
- <SHIFT B>: ... im Prinzip wie B. Der Unterschied: Die Höhle baut sich Schritt für Schritt auf (wie es auch Boulder Dash macht). Diese Funktion ist vor allem für Einsteiger sehr nützlich, um den Höhlenaufbau zu verstehen. Zurückblättern kann man allerdings nicht mehr.
- <M>: Memo (Speicher Nr). Sichert das Objekt in einem der RAM-Speicher 0 bis 9.
- <SHIFT M>: wie M. Das Objekt wird aber zusätzlich aus der Höhle getilgt.
- <H>: Holen (Speicher Nr). Bringt ein gespeichertes Objekt zurück und fügt es an der alten Position ein: Jetzt ist es das aktuelle Objekt. Es läßt sich also direkt mit <A> oder <SHIFT A> ändern.
- <N>: (Zahl). Jedes Objekt erhält eine fortlaufende Nummer, die man mit dieser Funktion ändern kann. In der Nummernfolge werden die Objekte später gezeichnet.
- <U>: Undo Info ... macht alle Änderungen in den beiden Infozeilen rückgängig.
- <SHIFT U>: Undo Höhle. Die Höhle wird restauriert und befindet sich wieder in dem Zustand wie beim Start des Edit-Modus oder beim letzten Sichern.
- <F5>: Sichern. Speichert den aktuell gültigen Zustand der Höhle. Er läßt sich mit <SHIFT U> wieder laden.
- <F8>: Ende. Edit-Modus verlassen und Rückkehr zu den DOS-Funktionen.

#### Befehle für Einstellungen:

Die Mehrzahl dieser Anweisungen wurde bereits bei der Erläuterung zu »Infozeilen« berücksichtigt. In unserer Tabelle finden Sie eine komfortable Gesamtübersicht.

#### Handicaps:

Dazu einige wichtige Infos: Boulder Dash benutzt für jede Höhle ein Flag-Byte. Darin wird vermerkt, auf welche Handicaps (Schmetterling, Firefly, Schleim und Schleimwand) man in der Höhle stößt. Verwendet man diese Spielelemente zum Zeichnen, berechnet der Edit-Modus das Flag-Byte korrekt. Werden sie jedoch nur als Zufallselemente eingebaut, entfällt die Berechnung: Es würde viel zu lange dauern, denn das Programm müßte die Höhle fünfmal zeichnen (einmal pro Level). Die Berechnung des Flag-Byte läßt sich bequemer mit dem H-Befehl nachholen.

## Spielelemente in verschiedenen Farben

Der Farbe-Modus dient dazu, die aktuellen Farben der Spielelemente zu ändern. Wie im Edit-Modus erscheint eine verkleinerte Darstellung der Höhle auf dem Bildschirm. Dadurch bekommt man im Farbe-Modus jedoch nur einen ungefähren Überblick der Höhle im späteren Spiel. Dennoch reicht die Anzeige völlig aus, sich einen ersten Eindruck zu verschaffen.

Man kann zwischen drei Farben wählen. Farbe 1 und 2 lassen sich beliebig auf eine der 16 C-64-Farben einstellen; für Farbe 3 kann man jedoch nur die ersten acht Farben verwenden. Grund: Boulder Dash benutzt den Multicolor-Modus.

Per entsprechendem Tastendruck (also <1>, <2> oder <3>) in der Ziffernreihe auf dem Keyboard) aktiviert man den nächsthöheren Farbcode. Drückt man <SHIFT> zusätzlich zur Zahlentaste, läuft's genau umgekehrt.

Wie im Edit-Modus kann man auch hier die Levelnummer mit <F1> bzw. <F2> und die Höhlennummer mit <F3 / F4> ändern. Verlassen wird der Farbe-Modus ebenfalls mit <F8>.

## Levels testen

Der Tester ist ein eigenständiges Programm im Paket von Boulder Dash Construction Tool und erlaubt, selbstentwickelte, zu Spielen zusammengestellte Höhlen auszuprobieren. Das Programm lädt Boulder Dash II nach und startet es. Im Menü von Boulder Dash kann man nun jede Höhle einzeln auswählen und starten. Die Anzeige für »Leben« verringert sich während des Spiels nicht.

#### Bedienung:

Bevor man den Tester lädt, muß man die zu testende Höhlengrafik direkt in den Speicher holen (z.B. »Boulder Dash 1« von der beiliegenden Sonderheftdiskette):

```
LOAD "BOULDER DASH 1",8,1
```

Nach der Eingabe von NEW kommt jetzt das Testprogramm an die Reihe:

```
LOAD "TESTER",8
```

Achtung: Entfernen Sie jetzt die Sonderheftdisk und schieben Sie Ihre Version von Boulder Dash II in den Laufwerkschacht. Anschließend starten Sie den Tester mit RUN. Die Floppy lädt die benötigten Dateien des Originalspiels. Kurz darauf kann man mit dem Test seiner eigenen Höhlen beginnen.

#### Fehlermeldungen:

Beim Start des Testers können diverse Fehlermeldungen auftauchen:

BITTE ZUERST DAS SPIEL LADEN... DANN ERNEUT STARTEN

- Sie haben vergessen die Spiele-Datei zu laden (in unserem Beispiel also »Boulder Dash 1«). Das muß man jetzt schleunigst nachholen (LOAD »(spielename)«,8,1). »Tester« bleibt unverändert im Speicher und muß nicht erneut geladen werden!

FEHLER! KONNTE DAS PROGRAMM 2 NICHT LADEN  
BOULDER DASH II DISK EINLEGEN... DANN ERNEUT  
STARTEN

- Sie haben übersehen, die Originaldiskette von Boulder Dash II ins Laufwerk zu legen (oder es ist eine Raubkopie: dann wird die Disk vom Programm als defekt eingestuft). Stellen Sie die Fehlerursache ab (meist genügt schon, die Raubkopie gegen das Original auszutauschen) und starten Sie wieder mit RUN.

## Boulder-Spiele zusammenfügen

Bleibt noch die Erläuterung des dritten und letzten Programmteils von BoDaCT: Mit dem »Linker« können Sie maxi-



mal acht verschiedene Spiele laden und sie als Spielesammlung speichern, die automatisch mit einem Ladeprogramm versehen wird:

LOAD "LINKER", 8

Gestartet wird mit RUN.

## Programmbedienung:

Acht Spiele kann der Linker zu einem verbinden. Beim Laden müssen Sie angeben, welches man belegen will.

Die Befehlsliste:

- <F8>: ... beenden und zurück zum Basic 2.0,
- <#>: Floppyadresse ändern,
- <\$>: Directory,
- <@>: Diskettenbefehle,

Editor-Modus (Einstellungen)	
Tasten	Funktionen
<F1 / F2>	einen Level weiter bzw. zurück,
<F3 / F4>	eine Höhle weiter oder zurück,
<F8>	Edit-Modus verlassen und Rückkehr zum Editor-DOS,
<D>	Anzahl der Diamanten
<SHIFT D>	Punktwertung für Diamanten
<Z>	Zeit, in der die Aufgabe pro Höhle gelöst sein muß
<SHIFT Z>	Zeit für Zaubermauer und Schleim
<1 / 2>	Zeichen 1 bzw. 2 einstellen
<SHIFT 1 / 2>	Zähler, wie oft Zeichen 1 bzw. 2 in der Höhle vorkommen
<SHIFT 3>	dto., für das Zeichen, auf dem der Cursor plziert ist
<W>	Startwert (Wurzel) des Zufallsgenerators
<O>	Ändern aller Zufallsobjekte
<T>	Tropf (Wert). Besonderheit: Obwohl damit ein Zeichenbefehl aktiviert wird, bleibt das Objekt auf dem Bildschirm unsichtbar. Der Wert gibt die Wahrscheinlichkeit an, mit der ein Stein oder Diamant durch die Schleimwand fällt. Dabei benutzt Boulder Dash ein ungewöhnliches Erkennungsmerkmal: Nicht die Höhe ist ausschlaggebend, sondern die Anzahl der eingeschalteten Bits der Binärzahl des Werts: Je mehr aktiviert wurden, desto geringer ist die Wahrscheinlichkeit, daß etwas durch die Wand dringt (z.B. Wert 255 = 1111 1111 binär).



[7] Beispiel-Höhle auf Disk: Prof Sieben Dash

- <?>: Liste aller geladenen Spiele anzeigen,
- <L>: Spiel laden
- <CBM L>: Spielesammlung laden (z.B. »Boulder-Spiele« von der Sonderheftdiskette). Achtung: Damit werden alle bisher geladenen Spiele und Höhlen überschrieben!
- <S>: Spielesammlung speichern. Ein Spiele-Sample wird



[8] An Schwierigkeit kaum überbietbar: Brutalo Dash im Testmodus

stets mit dem Namen BOULDER-SPIELE auf Diskette abgelegt. Es hindert Sie jedoch nichts daran, die Dateibezeichnung mit der Floppy-DOS-Anweisung »Rename« zu ändern!

- <CBM S>: Einzelspiel sichern. Damit kann man Spiele wieder aus der Sammlung auskoppeln.
- <N>: Name ändern. Jedes Spiel wird im Computer unter dem Dateinamen registriert, den man beim Laden verwendet. Per N-Befehl kann man es umbenennen.
- <T>: Spiele tauschen,
- <SPACE>: ... zeigt den freien Speicherplatz an.

## Spielesammlungen:

Hat man eine Spielesammlung geladen, z.B.:

LOAD "BOULDER-SPIELE", 8, 1

muß man vor dem Start die Boulder-Dash-II-Originaldisk einlegen. Sollten Sie's vergessen haben, macht Sie das Programm darauf aufmerksam. Ist der Fehler behoben, muß man erneut mit RUN starten. Nach dem Laden von Boulder Dash II befindet man sich im Menü und kann jetzt das gewünschte Spiel per <F5> und <F7> wählen. Die übrigen Spiele bleiben unversehrt im Computerspeicher erhalten. So kann man jederzeit (z.B. nach Ende einer Runde) ein anderes ausprobieren! Die gewohnten Programmfunktionen von Boulder Dash II haben sich nicht geändert.

## Übung macht den Meister

Damit man die Bedienung des Editors schnell checkt und den verzwickten Aufbau der Boulder-Dash-Höhlen versteht, empfehlen wir, mit den Spieledateien auf der Diskette zu diesem Sonderheft zu experimentieren:

- Boulder Dash 1,
- Boulder Dash 2,
- Prof sieben Dash (Abb. 7),
- Brutalo Dash (Abb. 8),
- Boulder-Spiele.

Vor allem sollten Sie den Blätterbefehl des Editors (<SHIFT B>) aktivieren, so oft es geht: Zum Verständnis des Höhlenaufbaus ist er äußerst nützlich.

Dennoch ist die Bedienung von BoDaCT ziemlich komplex. Dies liegt hauptsächlich an der ungewöhnlichen Speicher-methode der Boulder-Dash-Höhlen. Aber: Der vom Programm-ator gewählte, objektorientierte Editor ist die optimale Lösung. Sipel programmierte Höhlen-Editoren, die nur das Setzen von Spielelementen erlauben, wären zwar leichter zu bedienen, spätere Level-Änderungen aber ungleich arbeits-intensiver, wenn nicht gar unmöglich. (bl)

## Kurzinfo: Boulder Dash Construction Tool

**Programmart:** Editor, Tester und Linker zum Ändern von Levels zu »Boulder Dash II«  
**Laden:** LOAD "EDITOR", 8  
**Starten:** nach dem Laden mit RUN  
**Steuerung:** Tastatur/Joystick Port 1  
**Besonderheiten:** benötigt die Spielversion Boulder Dash III  
**Benötigte Blocks:** 200  
**Programmautor:** Stephan Hradek