

Compute mit

40/84

Das unabhängige Magazin für Anwender und Interessenten von Commodore Computern

2,80 DM
24 öS
2,80 sF

Tips & Tricks

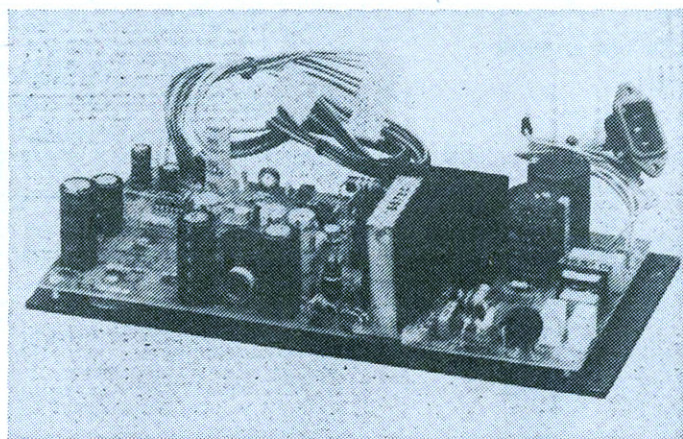
Unser kleiner Workshop bringt wieder praktische Programmierhilfen für Könnern und solche, die es schnellstens werden wollen. "ZS-Copy" und "Merge CBM 64" sind zwei Routinen, die jeder Commodore-User unbedingt drauf haben sollte. Auf Seite 33 erfahren Sie mehr.

Logik im Basic-Programm

Wie man Systematik in sein Selbstgestricktes bringt Teil 4

Heute beschäftigen wir uns mit einem ebenso wichtigen wie interessanten Element jeglicher Programmierarbeit, nämlich dem Thema "Entscheidungen". Was es da an Wissenswerten gibt, finden Sie bestimmt auf Seite 54.

Neues Schaltnetzteil mit Spitzenwerten



Speziell für die Erfordernisse der Mikroprozessor-Anwendung wurde das SC 1142 von bitronic entwickelt. Lesen Sie auf Seite 5, mit welchen Werten das neue Gerät aufwartet.

4. CFA in Frankfurt/M



Vom 6. bis 8. September fand in Frankfurt die diesjährige Commodore-Fachausstellung statt, zu der neben vielen Anbietern der breitgefächerten Zubehör- und Software-Industrie auch Ihre "Compute mit" angetreten war, um sich der staunenden Öffentlichkeit zu präsentieren. Lesen Sie unseren großen Messebericht auf Seite 34

Schach dem Computer

Wir haben für Sie zwei anspruchsvolle Schachprogramme getestet, nämlich Colossus 2.0 von CDS und CAISSA von Dynamics. Beide Programme sind in ihrer Art außergewöhnlich und sehr spielstark. Schlagen Sie sofort unseren Test auf Seite 30 auf.

The Pit für den VC-20 + 16K

In 88 Tagen sollte es bei Jules Verne um die Welt gehen. In 99 Sekunden müssen Sie bei diesem spannenden Gräberspiel den Schatz geborgen haben. Wenn Sie diesen tollen Vertreter einer erfolgreichen Spielidee näher kennenlernen wollen, sollten Sie sich schnurstracks auf Seite 44 begeben.

— Assembler-Kurs (4. Teil) — Berichte — News — Software — CM Lexikon —
— Basic-Kurs (3. Teil) — Rätsel — Logik im Basicprogramm (Teil 4) —
- 10 Spielprogramme für C-64 und VC-20 z.B. Universum, Pferderennen, The Pit -

Soft-Service 3

Telex

HiFi-Video: Kooperationsvertrag zwischen dem Otto Maier Verlag und der Firma Sonny

Festival der Mikrofans:

4. Internationale Commodore Fachausstellung 4

Business-Basic für Commodore

Für Mikro-P und Terminals:

Neues Schaltnetzteil 62 5

Neue Doppel-Floppy für den Commodore 64

IEEE-Aufwertung der Computer SX-64/C-64 6

CBS Computerspielprogramme

Eine sinnvolle Ergänzung der Produktpalette

Ali – das intelligente Algebra-Programm für den C-64 7

Software

Danger (C-64)	8
Frësser (C-64)	12
Universum (C-64)	17
Olivaro (C-64)	22
Pferderennen (C-64)	25
Cometen Falle (VC-20)	36
Zahlen-Senso (VC-20)	37
Delta-Race (VC-20)	39
Kranführer (VC-20)	41
The Pit (VC-20)	44

Softreviews

Valhalla (C-64) 30
Atom Smasher (VC-20)

Schachprogramme im Vergleich.

Colossus und Caissa für den Commodore 64 31

Adventure-Ecke

The Dallas Quest für den Commodore 64 32

Bücher

Heyne: Computer Lexikon 32
Menzel: Dateiverarbeitung mit Basic

Tips & Tricks 33

Report

CFA '84
Commodore-Fachausstellung vom 6. bis 8. September
Wir waren für Sie dabei und berichten über das Neueste bei Commodore 34

CM-Lexikon

In unserem CM-Lexikon bringen wir in dieser Ausgabe Computerbegriffe mit dem Anfangsbuchstaben "B" 50

Werkstatt

Basick Kurs Teil 4 51

Assembler Kurs 58

Leserbriefe 61

Kleinanzeigen 62

Rätsel 63

Impressum

Compute mit
erscheint wöchentlich im:
Roeske Verlag, Eschwege

Herausgeber:
Ralph Roeske

Redaktion:
Ralph Roeske (Chefredakteur, verantwortlich)
Gertrud Marx-Fischer (stellvertr. Chefredakteurin)
Horst Franke, Stefan Kaus, Thomas Morgen,
Stefan Ritter, Gisela Schwellach

Freie Mitarbeiter:
D. Gütlich, A. Hazy, A. Bachler, E. Wehner,
A. Kroneberger, M. Hilke, M. Lissy, J. Struller,
Th. Schlaeger, Th. Sielenkämper

Gestaltung u. Fotografie:
Renate Wells, Gerd Köberich

Herstellung:
Roeske Verlag, Eschwege

Satz und Reproduktion:
Roeske Verlag, Eschwege

Druck:
Parzeller, Fulda

Vertrieb:
Inland (Groß-, Einzel- und Bahnhofsbuchhandel),
sowie Österreich und Schweiz:
Verlagsunion
Friedrich-Bergius-Straße 7
6200 Wiesbaden
Tel.: 06121/2660

Anfragen nicht an Vertrieb oder Druckerol, sondern nur an den Verlag!

Anschrift:
Roeske Verlag
Compute mit
Fuldaer Straße 6
3440 Eschwege
Tel. Sa. Nr. 05651/8558

Anzeigenerweiterung:
Roeske-Verlag, Eschwege
Christian Widuch,
Tel.: 05651/8559

Erscheinungsweise:
Compute mit erscheint wöchentlich, Mittwoch

Anzeigenpreise:
Bitte Mediaunterlagen anfordern

Urheberrecht:
Alle in Compute mit veröffentlichten Beiträge sind urheberrechtlich geschützt. Alle Rechte, auch Übersetzungen, vorbehalten.
Reproduktionen jeder Art (Fotokopie, Microfilm, Erfassung in Datenverarbeitungsanlagen, usw.) bedürfen der schriftlichen Genehmigung des Verlags. Alle veröffentlichte Software wurde von Mitarbeitern des Verlages oder von freien Mitarbeitern erstellt.
Aus ihrer Veröffentlichung kann nicht geschlossen werden, daß die beschriebenen Lösungen oder Bezeichnungen frei von Schutzrechten sind.

© Roeske Verlag, Eschwege
Redaktion "Compute mit"

Bezugspreise:
Einzelheft: 2,80 DM

Autoren, Manuskripte:
Der Verlag nimmt Manuskripte und Software zur Veröffentlichung gerne entgegen.
Sollte keine andere Vereinbarung getroffen sein, so gehen wir davon aus, daß Sie mit einem Honorar von DM 100,- pro abgedruckter Seite im Heft, einverstanden sind.

Bei Zusendung von Manuskripten und Software, erteilt der Autor dem Verlag die Genehmigung zum Abdruck und Versand der veröffentlichten Programme auf Datenträger.

Rücksendung erfolgt nur gegen Erstattung der Unkosten. Zusendungen von Software zur Veröffentlichung soll bitte folgendes enthalten:

Kopierfähige Kassette oder Diskette mit dem Programm (Computer-Bezeichnung), von Drucker erstelltes Listing oder Serie von Bildschirmfotos (keine Schreibmaschinenlistings), evtl. Bildschirmfotos von einem Probelauf, ausführliche Programmbeschreibung (Erklärung der Variablenliste, Beschreibung des Bildschirmaufbaues, Farbe, Grafik usw.)

Für eingesandte Programmunterlagen kann keinerlei Haftung übernommen werden.

DatEx 64

DatEx 64 ist eine universelle Daten- und Textverarbeitung, vorwiegend für den Privatanwender, mit überdurchschnittlichen Leistungen. Sie können mit **DatEx 64** Ihre Videocassetten, Briefmarken, Rezepte, Schallplatten, Bücher etc. verwalten und Texte bearbeiten, mit deutschen Sonderzeichen. Durch einfache Anwendbarkeit ist die Arbeit mit **DatEx 64** auch für den Anfänger schnell zu erlernen. Die wichtigsten Eigenschaften in Stichworten:

Datensätze frei erstellbar (von einem Zeichen bis zu einer DIN-A4-Seite in der selben Datei) – verschiedene Suchoptionen – Zugriff über jeden Eintrag (Datenbankähnliche Funktion) – Erweiterungsmöglichkeiten – Textbreite bis 80 Zeichen (Scrolling) – Übernahme des letzten Wortes (kein Trennen notwendig) – nachträgliches Einfügen – **deutsche Sonderzeichen auf den Druckern: MPS 801, VC 1525, Seikosha GP 100 VC** – Bildschirmfarben jederzeit änderbar – Daten- und Textverarbeitung wird nur einmal geladen – Plausibilitätskontrollen – deutsches Handbuch – nur Diskettenversion.

DM 69,90

Disketten 64

Disketten 64 braucht jeder Anwender, der mehr als fünfzig Programme oder Dateien auf Diskette besitzt. Mit **Disketten 64** hat die Sucherei nach Programmen ein Ende und außerdem sind Sie jederzeit über Ihren Dateien-Programmbestand informiert. Der Vorteil von **Disketten 64** besteht eigentlich darin, daß die Eingaben nicht über die Tastatur erfolgen, sondern von **Disketten 64** selbständig eingelesen werden.

Die wichtigsten Eigenschaften in Stichworten: Erfassung nach max. 40 verschiedenen Gebieten – Sortieren nach Programmnamen (alphabetisch) – Folgende Informationen werden von Disketten 64 eingelesen und ausgegeben: Länge des Programmes (Blocks), Anzahl der Teile des Programmes, Name der Diskette, fortlaufende Programmnummer – zusätzlich kann ein kurzer Kommentar eingegeben werden – deutsches Programm.

DM 69,90

AdrEx 64

AdrEx 64 ist ein menügesteuertes Adressenverwaltungsprogramm.

AdrEx 64 zeichnet sich durch eine hohe Anwenderfreundlichkeit aus. Es können 450 Adressen pro Datei verwaltet werden. Die wichtigsten Eigenschaften in Stichworten: 450 Adressen pro Datei (Cassettenversion 150) – zusätzlich 3 Textzeilen – Direktzugriff auf alle Kriterien einer Adresse, auch Textzeilen – komfortable Suchmöglichkeiten – löschen, ändern, ausdrucken einer Adresse – Listen und Etiketten drucken – Absicherung gegen Fehlbedienungen – Bildschirmfarben jederzeit änderbar – deutsches Programm.

Cassettenversion (150 Adressen)
Diskettenversion (450 Adressen)

**DM 69,-
DM 79,-**

Demnächst erscheint unser komfortabler "DISK-MONITOR" für den C-64.

BITTE INFO ANFORDERN.

Unser C-64 Katalog liegt für DM 2,- in Briefmarken bereit.

HÄNDLERANFRAGEN ERWÜNSCHT. Alle Programme für den Commodore 64. Computeranwendungen Seucan. Postf. 24 44, 8600 Bamberg 1, Tel.: 0 95 42/83 48

BESTELLCOUPON

Bitte senden Sie mir _____

☐ per Nachnahme (zzgl. DM 5,50) ☐ per Vorauszahlung (zzgl. DM 3,50)

Auslandsbestellungen nur gegen Vorkasse möglich (zzgl. DM 7,-)

Name: _____ Adresse: _____

PLZ: _____ Ort: _____ Unterschrift: _____

Bitte einsenden an: Computeranwendungen Seucan, Postfach 24 44, 8600 Bamberg 1.

soft service

Unser Softwareversand bietet Ihnen zu wirklich einmalig günstigen Preisen jede Woche die neuesten Programme, auf Kassette an.

Wenn Sie die hier aufgeführten Bedingungen beachten, können auch Sie von unserem Angebot Gebrauch machen.

Bestellungen Inland:

Gegen Einsendung eines Schecks oder Vorauszahlung auf unser Konto bei der Kreissparkasse Eschwege, Bankleitzahl 522 500 30 Kto.-Nr. 45 22 934

senden wir Ihnen die gewünschten Programme schnellstmöglich zu. Geben Sie bitte unbedingt die genaue Bezeichnung der Kassette (z.B. Compute mit 37/

84 C-64) und den Computer an.

Bestellungen Ausland:

Nur Vorauskasse, Schein
Keine Schecks oder Überweisungen!



aus Compute mit 40/84

Commodore 64 K 14,- DM

Universum
Fresser
Danger
Olivaro
Pferdereennen

VC-20

K 14,- DM

Cometen Falle
Zahlen-Senso
Delta-Race
Kranführer
The Pit

Sony und Ravensburger kooperieren bei "Edutainment" - MSX-Software

HiFiVideo: Einen Kooperationsvertrag im Bereich "Edutainment" für Homecomputer-Software nach dem MSX-Standard (Microsoft Extended Basic) werden der Otto Meier Verlag, Ravensburg, und die Sony Deutschland GmbH, Köln abschließen. "Edutainment", eine Zusammensetzung aus "education" und "entertainment" ist der neue Sammelbegriff für "educational software", die auf Basis des

Fachwissens des angesehenen Verlages von den beiden Firmen gemeinsam entwickelt werden soll. Dem Angebot an spielerischen Lernprogrammen dürfte breites Interesse sicher sein, denn Computerprogramme mit Nutzen und Lehrcharakter finden im Markt steigende Aufmerksamkeit. Im Gerätebereich zeichnet sich mit dem internationalen MSX-Standard auf dem bisher unübersichtlichen Homecomputer-Markt

eine Systemvereinheitlichung ab, da sich die bedeutendsten Firmen der Unterhaltungs- und Computerelektronik dem MSX-Standard angeschlossen haben.

Im einzelnen wurde beschlossen, die Zusammenarbeit beider Firmen auf der Verpackung deutlich zu machen, gemeinsame Werbung und Öffentlichkeitsarbeit zu betreiben und die Lernprogramme jeweils exklusiv über

den Fachhandel der Bundesrepublik zu vertreiben. Eine Ausdehnung des Vertriebs auf Österreich und Schweiz über die dortigen Sony-Niederlassungen ist geplant. Ein regelmäßiger Informationsaustausch soll die frühzeitige Erkennung von Trend sowohl im Geräte- als auch im Datenträgerbereich sicherstellen. Sony gibt Ravensburger voll entwicklungs-technische Unterstützung bis hin zu Daten-Endgeräten.

Vierte Internationale Commodore-Fachausstellung (CFA) Festival der Mikrofans



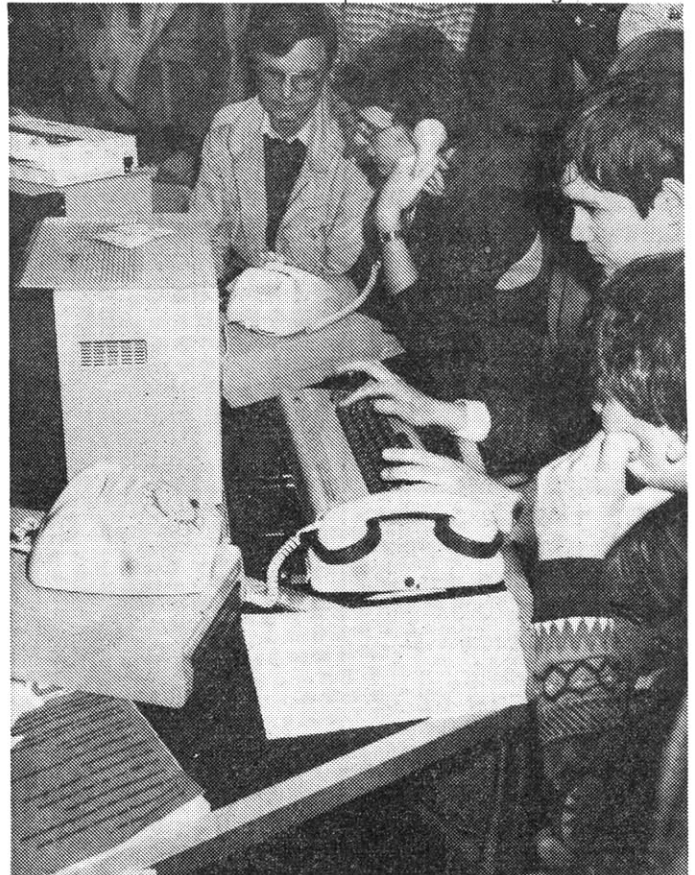
Schätzungsweise weit über 20.000 Fachinteressenten, Hobbynutzer, und "Freaks" besuchten die "Vierte Internationale Commodore-Fachausstellung" (CFA), die jetzt in Frankfurt zu Ende ging. Der Mikromarktführer präsentierte auf dem Messegelände der Main-Metropole sein gesamtes Produkt- und Leistungsangebot auf publikumsnahe Weise. Rund weitere 100 Aussteller demonstrierten vielfältige Hard- und Softwareneuheiten für Commodore-Anwender. Resümiert der für den Deutschland-Vertrieb zuständige Geschäftsführer Alwin Stumpf: "Die CFA erwies sich erneut als idealer Platz, jedermann bewußt zu machen: **Mikros sind heute**

praktische Alltagswerkzeuge wie Telefon, Radio oder Schreibmaschine. Hier bot sich eine ausgezeichnete Möglichkeit, den Anwendernutzen der Kleinrechner vor Ort jedermann konkret zu verdeutlichen!" Tatsächlich entwickelte sich die CFA in den letzten Jahren zu einem Mekka der Mikrocomputer aus der Home- wie Profi-Szene. Commodore wandte diesmal ganz erhebliche Mühe auf, um die Mikro-Messe zu einem attraktiven Anziehungspunkt zu machen. Neben einer aktuellen Leistungsschau führte das Unternehmen auch eine Reihe von Seminaren und Workshops durch, um den Informationswünschen der Besucher aus allen Be-

völkerungsschichten und Berufssparten gerecht zu werden.

Mindestens ebenso reizvoll war der illustre Fundus der Software-, Zubehör- und Peripherie-Aussteller. Zahlreiche EDV-Service- und Pro-

grammierfirmen aus mehreren europäischen Ländern offerieren an ihren Ständen praxisnahe Lösungen für Commodore-Kunden. Die Palette reichte dabei von der Fertigungssteuerung über die Meßwerterfassung bis hin



zum Radsporttraining. Stärker als im Vorjahr waren auf der CFA diesmal die Fachverlage aktiv. Die Häuser Langenscheidt, Otto Maier und Georg Westermann stellten neue Lernpro-

gramme und Anleitungsbücher für Mikros vor; die Volkshochschule warb für ihr stark erweitertes EDV-Aus- und Fortbildungsangebot. Commodore hatte schließlich auch eine spezielle Ge-

brauchtcomputer- und Programmabörse eingerichtet; außerdem konnten sich die "Freaks" mit Computer- und Videospielen amüsieren. Das Unternehmen betrachtet die CFA "als eines der schlag-

kräftigsten Instrumente des Consumer-Marketings für Mikros" (Stumpf). Bislang führt in der Bundesrepublik kein weiterer DV-Hersteller eine solche eigene Messe durch.

Brandaktuell! Ada von Data Becker

ADA ist eine Programmiersprache der jüngsten Generation. Sie erhielt ihren Namen zu Ehren der Countess Ada Lovelace, der Tochter des englischen Dichters Lord Bryon. Die Countess lebte im achtzehnten Jahrhundert in England und ist die erste bekannte Frau, die darüber nachdachte, wie eine von Charles Babbage entwickelte Rechenmaschine zu programmieren sei. Sie war sozusagen die erste Programmiererin der Welt (Zitat aus der Einführung).

Das Programm:

Im Grunde genommen ist das Programm noch gar kein richtiger Interpreter, sondern eher als Lernsystem vorgesehen. Es lassen sich kleinere Programme schon sehr gut programmieren. Auf der Programmdiskette befinden sich 5 Programme:

a) Der Editor:

Mit dem Editor können Sie Ihre ADA Programme schreiben.

Der Editor beinhaltet aber auch eine komplette Diskettenverwaltung. Sie können damit auf elegante Art und Weise Befehle an Ihre Diskettenstation übermitteln, sich die Inhaltsangaben Ihrer Disketten auf Bildschirm und Drucker ausgeben lassen und noch vieles mehr.

b) Der Syntax-Prüfer:

Dieses Programm überprüft einen mit dem Editor erstellten Algorithmus auf seine syntaktische Richtigkeit.

c) Der Semantik-Prüfer und Compiler:

Dieses Programm prüft den Algorithmus auf semantische Richtigkeit und erzeugt ein sehr schnelles Assemblerprogramm.

d) Der Assembler:

Den Assembler können Sie in Verbindung mit dem ADA Compiler, aber auch getrennt von ihm benutzen. Sie können mit ihm die Assemblerprogramme des Übersetzers und Ihre eigenen Assemblerprogramme und Maschinensprache übersetzen.

e) Der Disassembler:

Mit dem Disassembler können Sie die Maschinenprogramme, die sich im Rechner befinden, in die mnemonische Assemblerschreibweise zurückübersetzen.

DAS LERNPROGRAMM:

Ziel des Lernprogrammes ist es, mit der sehr jungen Sprache ADA zurechtzukommen. Gleichzeitig soll es aber auch helfen, mit dem CBM 64 besser arbeiten zu können. Ein großer Teil des Trainingskurses befaßt sich deshalb mit dem Programmieren in Assembler-Sprache und der Nutzung von Betriebssystemroutinen. Dies nicht zuletzt deshalb, weil der Assembler die Schnittstelle zwischen der höheren Programmiersprache 'ADA' und dem Mikroprozessor 6510, dem 'Herz' ihres Commodore 64, ist.

Das Durcharbeiten der Assembler Routinen gibt eine wertvolle Hilfe, wie eigene Assemblerprogramme geschrieben werden können.

Wie zu erkennen ist, arbeitet das Trainingsprogramm auf zwei Ebenen:

1. Auf der Ebene der 'Höheren' Sprache ADA
2. Auf der Ebene der Maschinensprache

Die Sprache ADA ist im Gegensatz zu BASIC strukturiert (wie Pascal oder Fortran) und somit recht einfach im Aufbau der Programme und sehr schnell in der Ausführung. Durch den Aufbau des Trainingskonzeptes ist sie für Leute, die Interesse an weiterführenden Sprachen haben, ideal.

Auch Anfänger können durch den Kurs diese Sprache erlernen.

Das 'Zubehör', Assembler und Disassembler, kann auch unabhängig vom Programm verwendet werden und rechtfertigt allein schon den Preis von DM 189.-.

Bezugsquelle: Data Becker,
Düsseldorf

Für Mikro-P und Terminals: Neues Schaltnetzteil 62W

Speziell für Mikroprozessoranwendungen wurde das neue Schaltnetzteil Modell SC1142 entwickelt.

Mit drei Ausgängen bietet bitronic/Sierracin ein Gerät für internationalen Einsatz, 90-132 V bzw. 180-264 Vac, 47-63 Hz Eingangsspan-

nung, und 5V/7A, +12V/2A und -12V/0,25A. Damit können also die Spitzenströme für den Anlauf von zwei 5 1/4"-Floppy-Disk-Laufwerken gezogen werden. Bewährte Flyback-Technologie vereint dies Netzgerät in 285 x 140 x 70mm, es entspricht den

Sicherheitsbestimmungen
UL, CSA, IEC wie VDE und
den Störstrahlungsbedingun-
gen VDE 0871 "B".

Auffällig sind die schon vorgegebenen Kabel mit Steckverbindern zur einfachen Installation. Welligkeit und Rauschen liegen unter 1%

Spitze/Spitze, Reglungsgenauigkeit über alles bei $\pm 1\%$ an 5V und $\pm 3\%$ an 12V. Implementiert sind Schutz gegen Überspannung und Kurzschluß sowie Netzausfallsignal.

CBS Computerspielprogramme

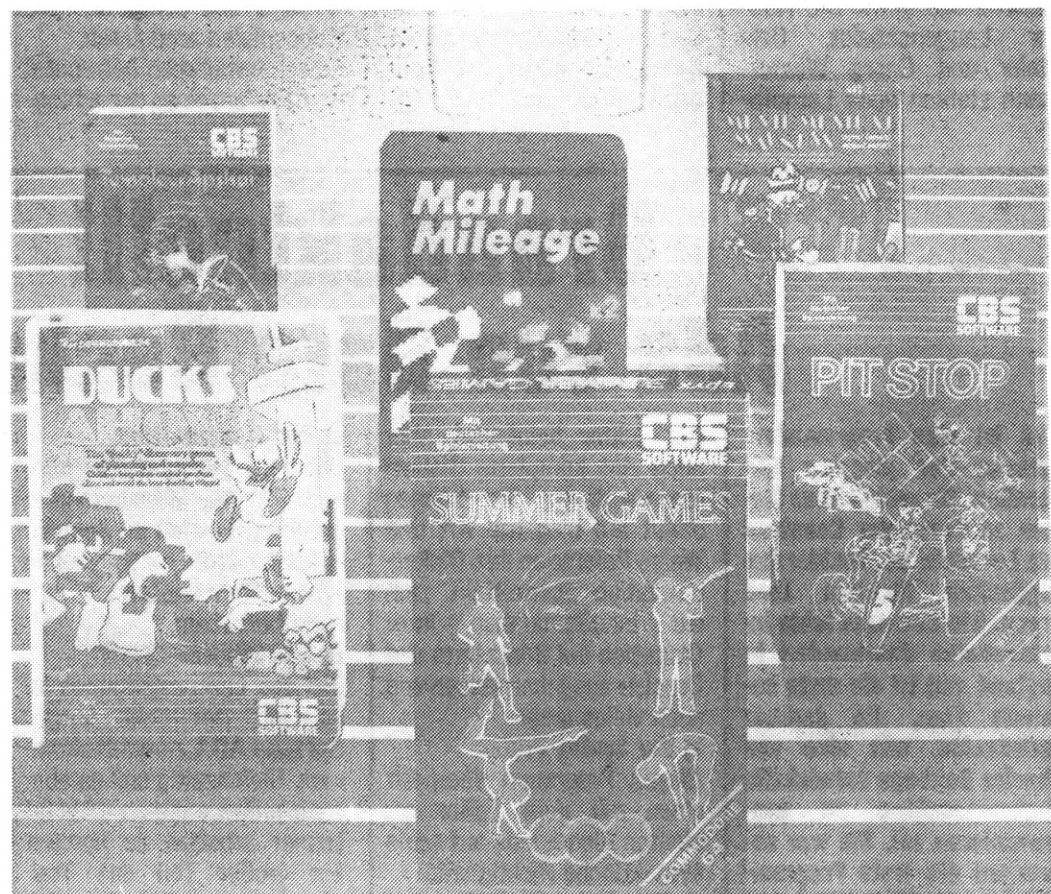
Eine sinnvolle Ergänzung der Produktpalette

Mit insgesamt 15 Computerspielprogrammen, kompatibel für die Homecomputer **COM-MODORE 64** und **COMMO-DORE VC-20** sowie für die **Atari Hardware von 400 bis 1200** kommt **CBS Electronics** auf den Markt.

Diese Computerspiel-Software ist für die ganze Familie entwickelt. Sieben Vorschulprogramme regen die Entdeckerfreude an und fördern das Lernen. Außerdem bringen sie Kindern den optimalen Einstieg in die Welt der Computer.

Die Lernprogramme sind von Pädagogen erarbeitet worden, die Fachleute auf dem Gebiet der computergestützten Didaktik sind. Außerdem gibt es in der Serie insgesamt acht, in den USA teilweise preisgekrönte Strategie- und Taktikspiele, die auch ihren Reiz auf die Erwachsenen ausüben.

Mit dieser Computerspiel-Software dringt die CBS-Tochter **ARXON** in einen gewaltig anwachsenden Markt ein. Wurde noch in der Periode Februar/Mai 1983 insgesamt ca. 20.000 Homecomputer



verkauft, so waren im Vergleichszeitraum 1984 insgesamt ca. 136.000 Geräte. Die CBS Electronics hat ein jahrzehntelanges Know How im Bereich der Unterhaltungselektronik.

Die neuesten Computerspiele werden von den besten Spieleentwicklern programmiert. Durch diese qualitativen Fakten und nicht zuletzt durch die Tatsache, daß das neue Software-Programm für

Computer engagiert beworben wird, rechnet sich **ARXON** einen guten Marktanteil für diesen Geschäftsbereich aus.

Mit Hilfe dieses Programms wird der Mathematikunterricht an Gymnasium unterstützt. Die Aufgaben reichen von der Unterstufe bis zum Abitur. Um einen besseren Überblick über die Fähigkeiten von Ali zu bekommen, besteht zunächst die Möglichkeit einer Demonstration. Schon hier zeigt Ali, daß einiges in ihm steckt. Gleichungen, Faktorisieren, Wertetabellen, Termvereinfachungen sind nur einige Leistungsmerkmale von Ali. Durch seine enorme Vielseitigkeit kann Ali für viele Schüler eine optimale Hilfestellung sein.

Info: Heureka-Software in München.

(sr)

ALI – das intelligente Algebraprogramm für den Commodore 64

Macht die Mathematik zum Kinderspiel



Heureka-Software
Vom Lehrer für Schüler

ALI macht Schule: Durch Übung zum Meister aller Klassen

$\begin{pmatrix} C \\ \uparrow \\ 0 \\ \downarrow \\ S \end{pmatrix}$

ALI

$\begin{pmatrix} C \\ \downarrow \\ \pi \\ \uparrow \\ C \end{pmatrix}$

Unterstützt den Mathematikunterricht am Gymnasium von der Unterstufe bis zum Abitur

Neue Doppel-Floppy für den Commodore 64

Unter der Bezeichnung "Micropower 2000" vertreibt die Firma WCS, Aumühle ein Doppellaufwerk für den C-64

Die Speichereinheit Micropower 2000 Double Disk Drive kann ohne Interface an alle Commodore-Rechner der Serien VC-20, C-64, 3000, 4000, 8000, 600 und 700 angeschlossen werden.

Dieses erlaubt die Verwendung des seriellen IEC-Bus und der IEEE-488 Schnittstelle. Das Gerät ist schreib- und lesekompatibel mit den Commodore-Geräten VC-1541 und CBM 4040.

Mit der Diskettenstation Micropower 2000 lassen sich Disketten in weniger als 2 Minuten duplizieren. Eine höhere Datengeschwindigkeit zwischen dem VC-20, C-

64, CBM 264 und der Diskettenstation kann mit einem IEEE-488 Interface realisiert werden.

Der Fehlerkanal wird über eigene Leuchtdioden angezeigt. Der Controller wird von einem intelligenten Mikroprozessor gesteuert.

Technische Daten:

Disketten: Standard 5 1/4 Zoll

Single Side/Si.-Density

Totale Speicherkapazität: 174 848 Bytes P. Disk.

Sequentielle Files: 168 656 Bytes P.Disk.

Relative: 167 132 Bytes P. Disk.

64535 Records P. File

Einträge in die Directory: 144 P. Disk.

Sektoren pro Spur: 17-21

Bytes pro Sektor: 256

Spuren: 35



Blöcke: 683 (664 frei)

Mikroprozessor: Serie 6500

Puffer: 6 K RAM

Betriebssystem: 16 K ROM

Laufwerke: 2 Stück

Spannung: 220 Volt Wechselspannung

Frequenz: 50/60 Hertz

Leistungsaufnahme: 45 Watt

Höhe: 145mm

Tiefe: 300mm

Breite: 220mm

Preis: DM 2.698,- + 14% MwSt.

IEEE-Aufwertung der Computer SX-64/C-64

Mit dem SX-64 ist seit einiger Zeit der "Renner" unter den Heimcomputern, der C-64, auch in "seriöser" Version erhältlich.

Bei beiden Rechnern stört – trotz all ihrer Vorteile – das Fehlen eines IEEE-488-Ausgangs wie z.B. am legendären PET 2001 & CBM 8032.

Diesen Mangel behebt das IEEE-488-Steckmodul von te-wi, ein Lizenzprodukt der TECHNOFOR GmbH: ausgereift, weltweit im Einsatz. Für DM 239,- incl. MwSt. mit 60seitigem Betriebshandbuch direkt von te-wi und allen anspruchsvolleren Commodore-Händlern zu beziehen!

Einsatzbereiche allgemein:

...große IEEE-kompatible CBM-Peripherie auch am SX-64/C-64!

...simultan VC-serieller/IEEE-paralleler Datenverkehr

Schulen:

...problemlose Mehrbenutzersysteme mit C-64 am IEEE-Bus

...rationelle Mehrfachbenutzung eines teuren Peripheriegeräts

Industrie

...preiswerte IEEE-Meß/Steuersysteme mit C-64 als Controller

...voll IEEE-normkonformer Modul – elektrisch wie funktionell

Anwendungshilfen:

Für Turnkey-Systeme: Leerdisketten und Lesekennzeichen

für Adreßverlegung: Utility-Disketten

für Basic-4-Befehle: nachladbare Zusatzsoftware auf Disk

Hi-Tec-Varianten:

Modell II hat gegenüber Modell I Goldkontakte

Modell III kann C-64 in Non-Controller am IEEE-Bus verwandeln!

Für C-64 als Multi-Con-

troller/Redundanter Controller

Für C-64 als IEEE-Bus-Monitor und-Analysator!



Danger

für den Commodore 64

Versuchen Sie, den vorgegebenen Weg zu durchlaufen. Doch Vorsicht! Auf dem Weg zum Ziel erwarten Sie einige gefährliche Hindernisse. Weichen Sie den hochstoßenden Pfählen aus und hüten Sie sich vor den zuschnappenden Fallen!

Anmerkungen zum Programm:

Der veränderte Zeichensatz ist ab Adresse 12288 abgelegt.

In den Zeilen 290-310 des Programmes wird die Musik gelesen und bestimmt.

In den Zeilen 500-560 wird der Zeichensatz geändert.

In den Zeilen 1020-1550 gehen die Pfähle per Zufalls-generator hoch und runter.

In den Zeilen 2000-2430 gehen die Fallen per Zufalls-generator auf und zu.



```

10 REM*****
20 REM*
30 REM* (C)BY ACHIM KRONEBERGER *
40 REM*      &      *
50 REM*      EBERHARD WEHNER      *
60 REM*      *
90 REM*****
100 POKE53281,0:POKE53280,0
110 PRINT"Q"
120 PRINT"3"
130 PRINT"#####";
140 PRINT"#####";
150 PRINT"#####";
160 PRINT"#####";
170 PRINT"#####";
180 PRINT"#####";
190 PRINT"#####";
200 A1=A1+1
210 ONA1GOSUB380,390,400,410,420,430,440,450,460
220 PRINT"#####WRITTEN IN 1984"
230 PRINT"#####BY"
240 PRINT"#####EBERHARD WEHNER & ACHIM KRONENBERGER"
250 PRINT"#####PRESS SPACE BAR TO START GAME!"
260 GETA$
270 IFA$=" "THEN290
280 GOTO120
290 SI=54272:FL=SI:FH=SI+1:TL=SI+2:TH=SI+3:W=SI+4:A=SI+5:H=SI+6:L=SI+24
300 POKEL,15:POKETH,13:POKETL,15:POKEA,3*16+15:POKEH,9
310 READX:READY:READO
320 IFX=-1THEN470
330 POKEFH,X:POKEFL,Y
340 POKEW,65
350 FORT=1TOD:NEXT
360 POKEW,0:B0=2000
370 GOTO310
380 PRINT"Q":RETURN
    
```



```

1140 J=PEEK(56320)
1150 IFJ=123THENPOKE2040,13:C=C-6
1160 IFJ=119THENPOKE2040,11:C=C+6
1170 IFC<20ANDPEEK(V+16)=0THENC=20
1180 IFC>254ANDPEEK(V+16)=0THENPOKEV+16,1:C=2:POKEV+27,63
1190 IFC>64ANDPEEK(V+16)=1THENGOTO1700
1200 IFC<2ANDPEEK(V+16)=1THENPOKEV+16,0:C=254
1210 POKEV,C
1220 KO=PEEK(V+30)
1230 IFKO<>0THENGOTO1630
1240 POKEV+30,0
1250 GOTO1020
1260 FORW=250TO228STEP-1
1270 POKEV+9,W:NEXT
1280 RETURN
1290 FORW=250TO228STEP-1
1300 POKEV+3,W:NEXT
1310 RETURN
1320 FORW=250TO228STEP-1
1330 POKEV+5,W:NEXT
1340 RETURN
1350 FORW=250TO228STEP-1
1360 POKEV+7,W:NEXT
1370 RETURN
1380 FORW=250TO228STEP-1
1390 POKEV+11,W:NEXT
1400 RETURN
1410 FORW=228TO250
1420 POKEV+9,W:NEXT
1430 RETURN
1440 FORW=228TO250
1450 POKEV+3,W:NEXT
1460 RETURN
1470 FORW=228TO250
1480 POKEV+5,W:NEXT
1490 RETURN
1500 FORW=228TO250
1510 POKEV+7,W:NEXT
1520 RETURN
1530 FORW=228TO250
1540 POKEV+11,W:NEXT
1550 RETURN
1560 GOTO1020
1570 GOTO1570

1580 FORJJ=0TO62:READQ:POKE896+JJ,Q:NEXT
1590 FORJ=0TO62:READQ:POKE704+J,Q:NEXT
1600 FORJ=0TO62:READQ:POKE832+J,Q:NEXT
1610 FORJ=0TO62:READQ:POKE960+J,Q:NEXT
1620 RETURN
1630 FORVV=1TO2000:NEXT
1640 POKEV+30,0
1650 C=20:POKEV,C:POKEV+1,223
1660 MM=MM-1
1670 PRINT"#####";MM
1680 IFMM=0THENGOTO2540
1690 GOTO1020
1700 S1=S1+1000+MM*100
1710 POKEV+16,0:POKE56334,1:POKE1,55:POKEV+24,21
1720 POKEV+21,1:POKE2040,11:C=20
1730 POKEV,C:POKEV+1,223
1740 POKEV+39,1
1750 PRINT"J":POKE53280,0
1760 PRINT"200"
1770 PRINT"30"
1780 PRINT"30"
1790 PRINT"30"
1800 PRINT"30"
1810 PRINT"30"
1820 PRINT"30"
1830 PRINT"30"
1840 PRINT"30"
1850 PRINT"30"
1860 PRINT"30"
1870 PRINT"30"
1880 PRINT"30"
1890 PRINT"30"
1900 PRINT"30"
1910 PRINT"30"
1920 PRINT"30"
1930 PRINT"30"
1940 FORT=0TO1:POKE1984+T,160:POKE56256+T,0:NEXT:PRINT"MAENNCHEN:";MM
1950 PRINT"#####PUNKTE:";S1
1960 FORT=0TO11:POKE1984+T,160:POKE56256+T,0:NEXT
1970 FORT=16TO21:POKE1984+T,160:POKE56256+T,0:NEXT
1980 FORT=26TO31:POKE1984+T,160:POKE56256+T,0:NEXT
1990 FORT=36TO39:POKE1984+T,160:POKE56256+T,0:NEXT
2000 E=INT(RND(1)*24+1)
2010 IFE=1THENGOTO 2210
2020 IFE=2THENGOTO 2240
2030 IFE=3THENGOTO 2270
2040 IFE=4THENGOTO 2300
2050 E1=INT(RND(1)*27+1)
2060 IFE1=1THENGOTO 2330
2070 IFE1=2THENGOTO 2360
2080 IFE1=3THENGOTO 2390
2090 IFE1=4THENGOTO 2420
2100 J=PEEK(56320)
2110 IFJ=123THENPOKE2040,13:C=C-8
2120 IFJ=119THENPOKE2040,11:C=C+8
2130 IFC<20ANDPEEK(V+16)=0THENC=20
2140 IFC>254ANDPEEK(V+16)=0THENPOKEV+16,1:C=2
2150 IFC>64ANDPEEK(V+16)=1THEN2520
2160 IFC<1ANDPEEK(V+16)=1THENC=254:POKEV+16,0
2170 POKEV,C
2180 KO=PEEK(V+31)
2190 IFKO=0THENGOTO2450:POKEV+31,0
2200 GOTO2000
2210 FORA=2TO5
2220 POKE1984+A,160:POKE56256+A,0
2230 NEXT:GOTO2050
2240 FORA=12TO15
2250 POKE1984+A,160:POKE56256+A,0
2260 NEXT:GOTO2050
2270 FORA=22TO25
2280 POKE1984+A,160:POKE56256+A,0
2290 NEXT:GOTO2050
2300 FORA=32TO35
2310 POKE1984+A,160:POKE56256+A,0
2320 NEXT:GOTO2050
2330 FORA=5TO2STEP-1
2340 POKE1984+A,32
2350 NEXT:GOTO2000
2360 FORA=15TO12STEP-1
2370 POKE1984+A,32
2380 NEXT:GOTO2000
2390 FORA=25TO22STEP-1
2400 POKE1984+A,32
2410 NEXT:GOTO2000
2420 FORA=35TO32STEP-1
2430 POKE1984+A,32
2440 NEXT:GOTO2000
2450 FORH=223TO255
2460 POKEV+1,H
2470 NEXT:FORW=1TO1000:NEXT

```

```

2480 POKEV+16,0:MM=MM-1:PRINT"#####";MM
2490 C=20:POKEV+1,223:KO=1:POKEV,C:POKEV+31,0
2500 IFMM=0THENGOTO2540
2510 GOTO2000
2520 PRINT"J":POKEV+21,0
2530 S1=S1+1000+MM*100:GOSUB2720
2540 POKEV+16,0:A=1:X=100
2550 PRINT"#####NOCH EIN SPIEL? (J/N)
2560 PRINTTAB(10)"ERREICHTE PUNKTE:";S1
2570 IFMM=0THENS1=0
2580 POKEV+21,3:POKE2041,15:POKEV+23,2:POKEV+29,2:POKEV+3,70
2590 X=X+A
2600 IFX>250THENA=(-1)
2610 IFX<50THENA=1
2620 POKEV+2,X
2630 GETA$
2640 IFA$="J"THENGOTO2680
2650 IFA$="N"THENPOKEV+21,0:PRINT"J":END
2660 GOTO2590
2670 RETURN
2680 POKE56334,PEEK(56334)AND254:POKE1,PEEK(1)
2690 UI=1:POKEV+21,0
2700 POKE1,PEEK(1)OR4:POKE56334,PEEK(56334)OR1
2710 POKE53272,(PEEK(53272)AND240)OR12:GOTO560
2720 PRINT"
2730 PRINT"X #####";
2740 PRINT"#####";
2750 PRINT"#####";
2760 PRINT"#####";
2770 PRINT"#####";
2780 PRINT"#####";
2790 PRINT"#####";
2800 PRINT"#####";
2810 PRINT"#####";
2820 PRINT"#####";
2830 PRINT"#####";
2840 PRINT"#####";
2850 PRINT"#####";
2860 PRINT"#####";
2870 PRINT"#####";
2880 PRINT"#####";
2890 PRINT"#####";
2900 PRINT"#####";
2910 PRINT"#####";
2920 PRINT"#####";
2930 PRINT"#####";
2940 PRINT"#####";
2950 PRINT"#####";
2960 PRINT"#####";
2970 FORL=0TO3:
2980 POKE1984+L,160:POKE56256+L,0
2990 NEXT
3000 POKEV,200:POKEV+1,222
3110 DATA 34,207,410,23,59,70,26,20,70,23,59,70,17,103,125
3120 DATA 31,3,125,29,69,300,26,20,300,23,59,500
3130 DATA -1,-1,-1
3140 DATA 0,7,15,31,63,127,255,255,255
3150 DATA 28,224,240,248,252,254,255,255,255
3160 DATA 39,255,255,255,254,252,248,240,224
3170 DATA 65,255,255,255,127,63,31,15,7
3180 DATA 83,255,255,255,255,255,255,255,255
3190 DATA-1
3200 DATA0,255,0,1,255,128,3,231,192,7,129,224,15,231,240,15,231,240,7
3210 DATA231,224,3,231,192,3,231,192,1,255,128,1,255,128,0,255,0,0,255
3220 DATA0,0,255,0,0,255,0,0,255,0,0,255,0,0,255,0,0,255,0,0,255
3230 DATA0
3240 DATA0,0,0,0,0,0,0,252,0,1,246,0,1,255,128,1,252,0,0,254,0,1,255,0
3250 DATA1,239,0,1,239,0,1,239,0,1,239,0,1,243,0,1,255,0,1,255,0,1,255
3260 DATA0,1,255,0,0,252,0,0,255,192,0,255,192,0,0,0
3270 DATA0,0,0,0,0,0,0,63,0,0,111,128,1,255,128,0,63,128,0,127,0,0,255
3280 DATA128,0,247,128,0,247,128,0,247,128,0,247,128,0,207,128,0,255,128
3290 DATA0,255,128,0,255,128,0,255,128,0,63,0,3,255,0,3,255,0,0,0,0
3300 DATA0,0,0,0,0,0,124,70,222,64,165,80,65,20,80,93,20,92,85,244,80,69
3310 DATA20,80,125,20,94,0,0,0,0,0,0
3320 DATA0,57,23,188,69,20,34,68,164,34
3330 DATA68,167,60,68,164,48,68,68,40,56,71,164,0,0,0,0,0,0

```


für den Commodore 64

FRESSER

Das Programm "Fresser" läuft auf dem Commodore 64 mit Joystick.

Nach dem Starten werden Sie gefragt, ob Sie die Instruktionen lesen wollen. Danach ertönt eine Melodie. Ist sie zu Ende, wird nach dem Level (1-9) gefragt. Haben Sie ihn bestimmt und auf den Feuerknopf gedrückt, erscheint ein großer Monsterkopf, der erst mit dem Auge zwinkert und dann die Zunge rausstreckt. Danach beginnt das Spiel.

Sie sind in der Mitte des Spielfeldes und müssen Herzen fressen. Aber Vorsicht,

denn es stehen überall Monster herum. Wenn Sie gegen ein Monster oder gegen die Wand laufen, verlieren Sie ein Leben. Immer, wenn Sie 10 Herzen aufgesammelt haben, beginnt ein neues Bild, nur mit 25 Herzen und 25 Monster mehr. Haben Sie 70 Herzen aufgesammelt (700 Punkte), bekommt man 100 Punkte Bonus. Sie beginnen mit 3 Leben. Bei Bonus 100 erhält man eins dazu.

Variablenliste:

V = Basisadresse des VIC



KL = Anzahl der Herzen und Monster

MA = Anzahl der Leben

SI = Basisadresse des SID

J = Adresse des Joysticks an Port 2

SC = Score

HI = Highscore

X,Y = Die Position von "Fresser"

AS = Level

Programmaufbau:

8-13: Titelbild

100-111: Spielfeldaufbau

112-121: Joystickabfrage

200-207: "Fresser"-LINKS

300-307: "Fresser"-RECHTS

400-407: "Fresser"-HOCH

500-507: "Fresser"-RUNTER

600-610: Auge u. Zunge von Monster

1000-1003: "Fresser"-gegen die Wand gelaufen

1100-1103: "Fresser"-gegen ein Monster gelaufen

10000-11000: Spielstand

30000-30010: Game over

40000-50012: Bonus 100

60000-60040: Melodie

62000-62003: Level

63100-63108: Spritedefinition

63200-63505: Instruction

```

0 GOTO8:*****
1 *-----(C)1984-----*
3 *-----*
6 *-----*
8 PRINT"███";:KL=50:MA=3:V=53248:GOSUB63199
9 POKE53280,12
10 PRINT"███ F R E S S E R███"
11 GOSUB60000:GOSUB60000:GOSUB60000:GOSUB60000:GOSUB60000:GOSUB63100
12 PRINT"██████████BY MARCUS HILKA"
13 PRINT"██████LEVEL(1-9)?":GOSUB62000
14 J=PEEK(56320)
15 IF(JAND16)=0THENGOTO17
16 GOTO14
17 PRINT"███"
20 GOTO 10000
90 REM*****
91 REM*SPIELFELD*
92 REM*****
100 PRINT"███":FORL=1TOKL:POKEW,0
101 X=INT(RND(1)*1000)
102 POKE1024+X,83:POKE55296+X,2
104 X=INT(RND(1)*1000)
105 POKE1024+X,88:POKE55296+X,5
110 NEXT:POKE55296+20+40*12,1
111 X=20:Y=12:POKE1024+X+40*Y,42:FORA=1TO1000:NEXT
112 J=PEEK(56320)
113 IF(JAND4)=0THENAB=1
114 IF(JAND8)=0THENAB=2
115 IF(JAND1)=0THENAB=3
116 IF(JAND2)=0THENAB=4
117 IFSC=100ORSC=200ORSC=300ORSC=400ORSC=500ORSC=600THEN ZX=1
118 IFSC=700THEN40000
119 IFZX=1ANDD=1THEN11000
121 ONABGOTO200,300,400,500
177 REM*****
178 REM*LINKS*
179 REM*****
200 POKE1024+X+40*Y,32:POKEFH,29:POKEFL,69
201 X=X-1:POKEW,65
203 IFX=-1ORX=40THEN1000
204 IFPEEK(1024+X+40*Y)=88THEN1100
205 IFPEEK(1024+X+40*Y)=83THENSC=SC+10:D=1:ZX=0
206 POKE1024+X+40*Y,42:POKEW,0
207 GOTO112
288 REM*****
289 REM*RECHTS*
290 REM*****
300 POKE1024+X+40*Y,32:POKEFH,29:POKEFL,69
301 X=X+1:POKEW,65
303 IFX=-1ORX=40THEN1000
304 IFPEEK(1024+X+40*Y)=88THEN1100
305 IFPEEK(1024+X+40*Y)=83THENSC=SC+10:D=1:ZX=0
306 POKE1024+X+40*Y,42:POKEW,0
307 GOTO112
389 REM*****
390 REM*HOCH*
391 REM*****
400 POKE1024+X+40*Y,32:POKEFH,29:POKEFL,69
401 Y=Y-1:POKEW,65
402 IFY=-1ORY=25THEN1000
404 IFPEEK(1024+X+40*Y)=88THEN1100
405 IFPEEK(1024+X+40*Y)=83THENSC=SC+10:D=1:ZX=0
406 POKE1024+X+40*Y,42:POKEW,0
407 GOTO112
489 REM*****
490 REM*RUNTER*
491 REM*****
500 POKE1024+X+40*Y,32:POKEFH,29:POKEFL,69
501 Y=Y+1:POKEW,65
503 IFY=-1ORY=25THEN1000
504 IFPEEK(1024+X+40*Y)=88THEN1100
505 IFPEEK(1024+X+40*Y)=83THENSC=SC+10:D=1:ZX=0
506 POKE1024+X+40*Y,42:POKEW,0
507 GOTO112
597 REM*****

```


programme

```

598 REM*AUGE UND ZUNGE*
599 REM*****
600 PRINT"*****POL"
601 FORA=1TO400:NEXT
602 PRINT"***** 0 3
603 PRINT"*****
604 FORA=1TO900:NEXT
605 PRINT"***** 3"
610 RETURN
998 REM*****
999 REM*GEGEN DIE WAND GELAUFEN*
990 REM*****
1000 POKEV,0
1001 PRINT"VRESSER IST GEGEN DIE WAND GELAUFEN":FORA=1TO2000:NEXT
1002 MA=MA-1:IFMA=0THENGOTO30000
1003 GOTO10000
1089 REM*****
1090 REM*GEGEN EIN MONSTER GELAUFEN*
1091 REM*****
1100 POKEV,0
1101 PRINT"VRESSER IST GEGEN EIN MONSTER GELAUFEN":MA=MA-1
1102 FORA=1TO2000:NEXT
1103 IFMA=0THENGOTO30000
9980 REM*****
9981 REM*SPIELSTAND*
9982 REM*****
10000 POKEV+21,60:PRINT"V":D=0:ZX=0
10003 PRINT"VRESSER":MA
10004 PRINT"XPUNKTE":SC+G*800
10005 GOSUB60000:GOSUB600
10006 FORA=1TO2000:NEXT:POKEV+21,0:GOTO100
11000 KL=KL+25:GOTO10000
29980 REM*****
29981 REM*GAME OVER*
29982 REM*****
30000 PRINT"V"
30001 FORH=1TO9:PRINTMID$("GAME OVER",H,1)+" ";:FORHG=1TO100:NEXTHG,H
30002 IFSC>HITHENHI=SC:PRINT"VH I G H S C O R E!!!V"
30003 PRINT:PRINT"VXPUNKTE":SC+G*800
30004 PRINT"VHIGH":HI+G*800:G=0:PRINT"VSIE SPIELTEN IN LEVEL ":AS
30005 PRINT"VNOCH EINMAL(J/N)"
30006 GETA$:IFA$=""THEN30006
30007 IFA$="J"THENKL=50:MA=3:SC=0:GOTO13
30008 PRINT"VVIELLEICHT DAS NAECHSTE MAL
30009 PRINT"VTSCHUESSSSSSSS!!!!"
30010 PRINT"VENDE":END
39900 REM*****
39901 REM*BONUS 100*
39902 REM*****
40000 PRINT"V":MA=MA+1
40001 A1=983:B1=1024
40002 A2=1024:B2=2064
40003 A3=2063:B3=2023
40004 A4=2023:B4=983
40005 FORJ=1TO5
40006 A1=A1+41:B1=B1+39
40007 FORI=A1TOB1:GOSUB63000:NEXTI
40008 A2=A2+39:B2=B2+41
40009 FORI=A2TOB2STEP40:GOSUB63000:NEXTI
40010 A3=A3+41:B3=B3+39
40011 FORI=A3TOB3STEP-1:GOSUB63000:NEXTI
40012 A4=A4+39:B4=B4+41
40013 FORI=A4TOB4STEP-40:GOSUB63000:NEXTI
40014 NEXTJ
50000 PRINT"V"
50001 PRINT"V"
50002 PRINT"V"
50003 PRINT"V"
50004 PRINT"V"
50005 PRINT"V"
50006 PRINT"V"
50007 PRINT"V"
50008 PRINT"V"
50009 PRINT"V"
50010 PRINT"V"

```

```

50011 PRINT"#####" :RESTORE:SC=0
50012 GOSUB60000:GOSUB60000:GOSUB60000:GOSUB60000:GOSUB60000:G=G+1:GOTO10000
60000 REM*MALODIE*
60002 SI=54272:FL=SI:FH=SI+1:TL=SI+2:TH=SI+3:W=SI+4:A=SI+5:H=SI+6:L=SI+24
60003 POKEL,15:POKETH,13:POKETL,15:POKEA,15+3*16:POKEH,9
60004 READX:READY:READO
60005 IFX=-1THENRETURN
60006 IFX=-2ANDAS=0THENRETURN
60007 IFX=-2ANDAS>0THENRESTORE:RETURN
60008 POKEFH,X:POKEFL,Y
60009 POKEW,65
60010 FORT=1TOO/3:NEXT
60011 POKEW,0:GOTO60004
60012 DATA17,103,250,19,137,250,21,237,250,17,103,250,19,137,250,21,237,500
60014 DATA17,103,250,19,137,250,17,103,250,21,237,250,17,103,250,19,137,250
60015 DATA21,237,500
60016 DATA17,103,250,19,137,250,17,103,250,21,237,250,17,103,250,19,137,250
60017 DATA21,237,500
60019 DATA16,109,250,19,137,500,17,103,1000,-1,-1,0
60021 DATA9,196,250,10,89,250,10,247,250,17,103,500,10,247,250,17,103,500
60023 DATA10,247,250,17,103,1000,17,103,250,19,137,250,20,178,250
60024 DATA21,237,250,17,103,250,19,137,250,21,237,500,16,109,250
60025 DATA19,137,500,17,103,1000,-1,-1,0
60026 DATA9,196,250,10,89,250,10,247,250,17,103,500,10,247,250,17,103,500
60027 DATA10,247,250,17,103,1000
60028 DATA14,162,250,13,10,250,12,78,250,14,162,250,17,103,250,21,237,500
60029 DATA19,137,250,17,103,250,14,162,250,19,137,1000,-1,-1,0
60030 DATA9,196,250,10,89,250,10,247,250,17,103,500,10,247,250,17,103,500
60031 DATA10,247,250,17,103,1000,17,103,250,19,137,250,20,178,250
60032 DATA21,237,250,17,103,250,19,137,250,21,237,500,16,109,250
60033 DATA19,137,500,17,103,1000,-1,-1,0
60034 DATA17,103,250,19,137,250,21,237,250,17,103,250,19,137,250,21,237,500
60035 DATA17,103,250,19,137,250,17,103,250,21,237,250,17,103,250,19,137,250
60036 DATA21,237,500
60037 DATA17,103,250,19,137,250,17,103,250,21,237,250,17,103,250,19,137,250
60039 DATA21,237,500
60040 DATA16,109,250,19,137,500,17,103,1000,-2,-2,0
60100 REM DATA FUER MONSTERKOPF
60101 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
60102 DATA24,0,1,20,32,18,34,48,42,33,40,74,32,148,10,32,66,10,0,33,10
60103 DATA0,16,203,1,137,60,2,70,8,4,36,0,4,24,24,4,8,36,4,8,66,0,32,90,0,32,66
60110 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
60111 DATA128,0,0,64,0,3,1,1,133,2,130,137,4,68,145,8,8,161,16,16,64,16,32
60112 DATA128,45,97,0,194,154,8,0,4,24,0,3,40,24,0,72,36,0,32,60,0,64,90,0,128
60113 DATA60,0,64
60120 DATA0,32,36,0,16,24,0,32,0,0,16,0,0,32,0,0,64,45,0,128,82,0,64,128
60121 DATA0,32,128,0,16,128,0,8,64,0,16,59,0,32,4,0,16,0,0,12
60122 DATA0,0,2,0,0,2,0,0,1,0,0,0,131,0,0,76,0,0,48
60130 DATA36,0,128,24,1,0,0,0,128,0,0,64,4,0,128,171,193,0,65,34,0,0
60131 DATA33,0,0,32,128,1,65,0,34,146,128,247,21,0,8,34,0,0,32,0,0,16,0
60032 DATA21,237,250,17,103,250,19,137,250,21,237,500,16,109,250
60033 DATA19,137,500,17,103,1000,-1,-1,0
60034 DATA17,103,250,19,137,250,21,237,250,17,103,250,19,137,250,21,237,500
60035 DATA17,103,250,19,137,250,17,103,250,21,237,250,17,103,250,19,137,250
60036 DATA21,237,500
60037 DATA17,103,250,19,137,250,17,103,250,21,237,250,17,103,250,19,137,250
60039 DATA21,237,500
60040 DATA16,109,250,19,137,500,17,103,1000,-2,-2,0
60100 REM DATA FUER MONSTERKOPF
60101 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
60102 DATA24,0,1,20,32,18,34,48,42,33,40,74,32,148,10,32,66,10,0,33,10
60103 DATA0,16,203,1,137,60,2,70,8,4,36,0,4,24,24,4,8,36,4,8,66,0,32,90,0,32,66
60110 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
60111 DATA128,0,0,64,0,3,1,1,133,2,130,137,4,68,145,8,8,161,16,16,64,16,32
60112 DATA128,45,97,0,194,154,8,0,4,24,0,3,40,24,0,72,36,0,32,60,0,64,90,0,128
60113 DATA60,0,64
60120 DATA0,32,36,0,16,24,0,32,0,0,16,0,0,32,0,0,64,45,0,128,82,0,64,128
60121 DATA0,32,128,0,16,128,0,8,64,0,16,59,0,32,4,0,16,0,0,12
60122 DATA0,0,2,0,0,2,0,0,1,0,0,0,131,0,0,76,0,0,48
60130 DATA36,0,128,24,1,0,0,0,128,0,0,64,4,0,128,171,193,0,65,34,0,0
60131 DATA33,0,0,32,128,1,65,0,34,146,128,247,21,0,8,34,0,0,32,0,0,16,0
60132 DATA0,16,0,0,224,0,0,128,0,128,128,0,199,0,0,56,0,0
61988 REM*****
61989 REM*SCHWIRIGKEITSGRAD*
61990 REM*****

```

```

61999 :
62000 GETA$:AS=INT(VAL(A$))
62001 IFA$="" THEN 62000
62002 IFAS>SORAS<1 THEN 62000
62003 PRINTAS
62004 KL=AS*25 :PRINT"  FIRE  ":RETURN
62100 :
63000 POKEI,127:POKEI+54272,2:RETURN
63002 :
63100 V=53248:POKE2044,15:POKEV+43,5:POKE2045,255:POKEV+21,60
63101 POKE2042,11:POKE2043,13:POKEV+42,5:POKEV+41,5:POKEV+44,5
63102 FORN=0TO62:READQ:POKE 704 +N,Q:NEXT
63103 FORN=0TO62:READQ:POKE 832 +N,Q:NEXT
63104 FORN=0TO62:READQ:POKE 960 +N,Q:NEXT
63105 FORN=0TO62:READQ:POKE16320+N,Q:NEXT
63106 POKEV+23,60:POKEV+29,60
63107 POKEV+4,100:POKEV+5,100:POKEV+6,150:POKEV+7,100:POKEV+8,100:POKEV+9,140
63108 POKEV+10,150:POKEV+11,140:POKEV+21,0:RESTORE:RETURN
63199 POKE53281,0:POKE53280,0
63200 PRINT"  INSTRUCTIONS(J/N)? "
63201 GETA$:IFA$="" THEN 63201
63202 IFA$="J" THEN 63204
63203 RETURN
63204 GOSUB63500
63205 PRINT"  SIE SIND IN DER MITTE DES SPIELFELDES  UND MUESSEN HERZEN FRESSE
N.
63206 PRINT"  ABER VORSICHT,DENN ES STEHEN UEBERALL  MONSTER HERUM.
63207 PRINT"  WENN SIE GEGEN EIN MONSTER ODER GEGEN  DIE WAND LAUFEN,VERLIEREN
63208 PRINT"  SIE EIN LEBEN.
63210 PRINT"  SPACE "
63211 GETA$:IFA$="" THEN 63211
63212 IFA$=" " THEN 63214
63213 GOTO63211
63214 GOSUB63500
63215 PRINT"  IMMER,WENN SIE 10 HERZEN AUFGESAMMELT  HABEN(100 PUNKTE),
63216 PRINT"  BEGINNT EIN NEUES BILD,NUR MIT 25 HERZEN UND 25 MONSTER MEHR.
63217 PRINT"  BEI 70 HERZEN BEKOMMEN SIE BONUS 100.
63218 PRINT"  SIE BEGINNEN MIT 3 LEBEN.BEI BONUS 100  BEKOMMEN SIE 1 LEBEN DAZU.
63219 PRINT"  (MARCUS HILKA)":PRINT"  SPACE "
63220 GETA$:IFA$="" THEN 63220
63221 IFA$=" " THEN RETURN
63222 GOTO63220
63500 PRINT"  ":FORA=1TO40:X=INT(RND(1)*16):POKE646,X
63501 PRINT"  ";:NEXT:PRINT
63502 PRINT"  FRESSER "
63503 FORA=1TO40:X=INT(RND(1)*16):POKE646,X
63504 PRINT"  ";:NEXT:PRINT
63505 RETURN
READY.

```

Achtung Programmierer!

Haben Sie eigentlich schon bedacht, daß Ihre eingesandten Programme für "Compute mit" bei uns eine viermal so große Abdruckchance haben, als bei anderen Verlagen. Warum? Weil die "Compute mit" nicht einmal – wie alle anderen Computerzeitschriften – sondern viermal im Monat, sprich wöchentlich, erscheint. Außerdem können Sie 100%ig sicher sein, daß Ihr Programm von uns nach der Qualität beurteilt und natürlich auch dementsprechend honoriert wird. Trauen Sie sich – und schicken Sie uns Ihre Programme!

Universum

für den Commodore 64 mit Simon's Basic

Dieses Programm ist ein etwas ungewöhnliches, aber ganz toll gemachtes Dokumentationsprogramm. Es ist in Simon's Basic geschrieben. Mehr sollte vorher nicht verraten werden!

```

10 SI=54272:POKE SI+24,15
20 POKE SI+5,31:POKE SI+6,5*16+6
30 :
100 REM #####
110 REM ##
120 REM # DAS UNIVERSUM TEIL I #
130 REM ##
140 REM #####
200 :
210 POKE 53281,6:POKE 53280,6:PRINT "X"
220 PRINT " "
230 PRINT " "
240 PRINT " DAS UNIVERSUM TEIL I "
250 PRINT " "
260 PRINT " "
270 PRINT " MIT DEN THEMEN "
280 PRINT " DIE ERDE, "
290 PRINT " SCHWARZE LOECHER UND "
300 PRINT " DIE DREI URSPRUNGSTHEORIEN "
310 PRINT " "
320 PRINT " "
322 FLASH 1,40
330 GETT$:IFT$="" THEN 330
335 OFF
340 DIMM(8),M$(8),M2$(8)
1000 REM ***** ERDE
1005 POKE 53281,14:POKE 53280,14
1010 PRINT " ":PRINT AT(11,1)" D I E E R D E ":FLASH 6,200
1020 Z=2:FOR I=1 TO 16:READ T$:Z=Z+2
1025 IF Z=14 THEN Z=4:GOSUB 9000:GOSUB 7000
1030 GOSUB 8000
1040 NEXT
1045 GOSUB 9000:PRINT " ":OFF
1046 FOR I=1 TO 8:READ M(I):M2$(I)=STR$(M(I))+
1051 M$(1)="MAGNESIUM "
1052 M$(2)="KALIUM "
1053 M$(3)="NATRIUM "
1054 M$(4)="KALZIUM "
1055 M$(5)="EISEN "
1056 M$(6)="ALUMINIUM "
1057 M$(7)="SILIZIUM "
1058 M$(8)="SAUERSTOFF "
1060 HIRES 0,0:MULTI 6,7,2
1070 YD=5:XS=85:ZF=1
1080 FOR I=1 TO 8
1090 FOR J=0 TO INT(M(I)+.5)
1100 LINE J,YD,J,YD+15,ZF
1110 NEXT
1115 TEXT 45,YD+4,M2$(I),ZF,1,7
1120 TEXT XS,YD+4,M$(I),ZF,1,7
1130 YD=YD+20
1140 ZF=ZF+1:IF ZF=4 THEN ZF=1
1150 NEXT
1152 LINE 50,0,50,160,1
1153 LINE 82,0,82,160,1
1156 TEXT 52,168,"98.5",1,1,7
1157 LINE 50,178,82,178,1
1158 LINE 50,180,82,180,1
1159 BLOCK 0,185,159,199,1

```

```

1160 TEXT30,190,"ANTEILE IN %",0,1,8
1165 GETT$:IFT$=""THEN1165
1170 CSET1:PRINTCHR$(142)
1200 REM ***** SCHWARZE LOECHER
1201 POKE53281,0:POKE53280,0
1202 PRINTAT(12,1)"SCHWARZE LOECHER":FLASH 13,200
1205 Z=2
1210 FORI=1TO39:READTE$:Z=Z+2
1220 IFZ>=14THENZ=4:GOSUB9000:GOSUB7000
1230 GOSUB8000
1240 NEXT
1250 GOSUB9000:PRINT"J":OFF
1260 HIRES0,0:MULTI7,1,8
1261 CIRCLE79,50,25,40,2:PAINT79,50,1
1262 FORA=1TO150
1263 XX=INT(RND(1)*160)
1264 YY=INT(RND(1)*120)
1265 IFTEST(XX,YY)=0THENPLOTXX,YY,2
1266 NEXT
1285 FORI=0TO3
1290 LINE65+I,65+I,30+I,110+I,1
1300 NEXT
1310 LINE30,110,25,120,1
1320 LINE25,120,33,113,1
1330 PAINT29,113,1:PLOT26,119,1
1332 PLOT27,118,1
1335 LINE0,123,159,123,2
1340 YP=126:EZ=0
1360 READTE$:TEXT0,YP,TE$,3,1,8
1365 FORW=1TO1000:NEXT
1370 YP=YP+9
1380 EZ=EZ+1
1390 IFEZ<8THEN1360
1395 GETT$:IFT$=""THEN1395
1405 :
1410 HIRES0,0:MULTI8,2,1
1420 CIRCLE79,50,12,20,1
1430 PAINT79,50,1
1440 FORA=1TO150
1450 XX=INT(RND(1)*160)
1460 YY=INT(RND(1)*120)
1470 IFTEST(XX,YY)=0THENPLOTXX,YY,3
1480 NEXT
1490 FORI=1TO2:LINE72+I,58+I,30+I,110+I,2:NEXT
1500 LINE31,111,25,120,2
1510 LINE25,120,32,113,2
1520 PAINT30,113,2:PLOT28,116,2
1530 PLOT29,115,2
1540 LINE0,123,159,123,3
1550 YP=126:EZ=0
1560 READTE$:TEXT0,YP,TE$,3,1,8
1570 FORW=1TO1000:NEXT
1580 YP=YP+9
1590 EZ=EZ+1
1600 IFEZ<8THEN1560
1610 GETT$:IFT$=""THEN1610
1620 CSET1:PRINTCHR$(142)
1630 Z=2:FORI=1TO8
1640 READTE$:Z=Z+2
1650 IFZ>=14THENZ=4:GOSUB9000:GOSUB7000
1660 GOSUB8000
1670 NEXT
1680 GOSUB9000:PRINT"J"
1685 :
1690 HIRES0,0:MULTI7,1,0
1695 BLOCK50,20,110,100,3
1700 FORA=1TO300
1710 XX=INT(RND(1)*160)
1720 YY=INT(RND(1)*120)
1730 IFTEST(XX,YY)=0THENPLOTXX,YY,2
1740 NEXT
1750 FORX=0TO78STEP.5
1752 YY=5*SQR(78-X)
1760 Y1=53-YY:Y2=56-YY
1762 LINEX,Y1,X,Y2,1
1770 X1=159-X
1772 Y3=66+YY:Y4=69+YY
1774 LINEX1,Y3,X1,Y4,1
1776 X2=71-YY:X3=74-YY
1778 Y5=(205-X)/2
1780 LINEX2,Y5,X3,Y5,1
1800 NEXT
1810 LINE0,123,159,123,2
1820 YP=126:EZ=0
1822 READTE$:TEXT0,YP,TE$,2,1,8
1824 FORW=1TO1000:NEXT

```

```

1826 YP=YP+9
1828 EZ=EZ+1
1830 IFEZ<8THEN1822
1840 GETT$:IFT#=""THEN1840
1850 CSET1:PRINTCHR$(142)
1860 REM ***** URSPRUNGTHEORIEN
1862 POKE53281,6:POKE53280,6
1864 PRINTAT(9,1)"DIE URSPRUNGTHEORIEN":FLASH 14,200
1866 Z=2:FORI=1TO8
1868 READTE$:Z=Z+2
1870 IFZ>=14THENZ=4:GOSUB9000:GOSUB7000
1872 GOSUB8000
1874 NEXT
1876 GOSUB9000:PRINT" ":OFF
1879 :
1880 POKE53281,5:POKE53280,5
1890 HIRES0,0:MULTI0,7,2
1892 TEXT15,1,"THEORIE DES PUL-",1,2,8
1894 TEXT1,19,"SIERENDEN UNIVERSUMS",1,2,8
1895 GOSUB4500
1898 GOSUB6000:SM=200
1900 XG=150:XA=3:YG=90:YA=43:GOSUB6500
1910 GOSUB6000
1912 XG=70:XA=45:YG=40:YA=70:GOSUB6500
1916 SM=150:XG=20:XA=70:YG=20:YA=80:GOSUB6500
1918 GOSUB6000
1920 GOSUB4000
1942 XG=20:XA=70:YG=20:YA=80:GOSUB6500
1944 SM=200:XG=70:XA=45:YG=40:YA=70:GOSUB6500
1946 XG=150:XA=3:YG=90:YA=43:GOSUB6500
1947 GETT$:IFT#=""THEN1947
1948 :
1950 :
1952 HIRES0,0:MULTI0,7,2
1954 TEXT13,10,"URKNALL - THEORIE",1,2,8
1956 GOSUB4500
1960 GOSUB6000:GOSUB4000:GOSUB6000
1962 SM=150:XG=20:XA=70:YG=20:YA=80:GOSUB6500
1966 SM=200:XG=70:XA=45:YG=40:YA=70:GOSUB6500
1970 XG=150:XA=3:YG=90:YA=43:GOSUB6500
1971 GETT$:IFT#=""THEN1971
1972 :
1974 :
1976 HIRES0,0:MULTI0,7,2
1978 TEXT25,1,"STEADY STATE",1,2,8
1979 TEXT53,19,"THEORIE",1,2,8
1980 GOSUB4500
1982 GOSUB6500
1984 GOSUB6000
1990 GETT$:IFT#=""THEN1990
1991 CSET1:PRINTCHR$(142)
1992 POKE53281,6:POKE53280,14:PRINT" "
1994 Z=2:FORI=1TO4
1996 READTE$:Z=Z+2
1998 IFZ>=14THENZ=4:GOSUB9000:GOSUB7000
2000 GOSUB8000
2002 NEXT
2004 GETT$:IFT#=""THEN2004
2006 IFT#="J"THEN RUN
2008 PRINT"OK":END
4000 BLOCK2,42,157,138,1
4002 POKESI+4,129
4004 POKESI,147:POKESI+1,8
4006 UU=1:RR=2
4008 FORA=1TO6
4010 ARC 80,90,0,360,30,RR,RR,2
4011 IFA>1THENARC 80,90,0,360,30,AR,AR,1
4012 AR=RR:UU=UU+2:RR=AR+UU
4016 NEXT
4018 ARC 80,90,0,360,30,37,37,1
4030 POKESI+4,0:RETURN
4500 LINE0,40,159,40,1
4510 LINE0,140,159,140,1
4520 RETURN
5000 DATAUNSER HEIMATPLANET HAT EINEN DURCH-
5001 DATASCHNITTlichen DURCHMESSER VON 12742 KM.
5002 DATASIE DREHT SICH IN 365,2422 TAGEN IN
5003 DATAEINER MITTLEREN ENTFERNUNG VON AN-
5004 DATANAEHERND 149.6 MIO. KM EINMAL UM DIE
5005 DATASONNE. DIE DURCHSCHNITTliche DICHTe DER
5006 DATAERDE IST 5.5MAL SO GROSS WIE DIE DICHTe
5007 DATAVON WASSER. IHR VOLUMEN BETRAEGT
5008 DATA1 083 218 990 000 KMt3 - IHRE MASSE
5009 DATA5973 TRILLIARDEN TONNEN. DAS ALTER DES
5010 DATABLAUEN PLANETEN WIRD AUF MINDESTENS 4.3
5011 DATAMILLIARDEN JAHRE GESCHAETZT.

```


5012 DATAALLES IM UNIVERSUM BESTEHT AUS GRUND-
 5013 DATASUBSTANZEN. UNS SIND HEUTE WEIT UEBER
 5014 DATAHUNDERT BEKANNT. DENNOCH BESTEHEN ETWA
 5015 DATA98 % DER ERDKRUSTE AUS NUR 8 ELEMENTEN.
 5016 DATA2.1/2.6/2.8/3.6/5.0/8.1/27.7/46.6
 5020 :
 5021 DATAEINE SUPERNOVA STELLT DEN ANFANG EINER
 5022 DATALETZTEN EXISTENZSTUFE - UND NICHT DAS
 5023 DATAVOELLIGE ENDE - EINES STERNS DAR.,ASTRONOMEN HALTEN DIESE PULSARE
 5024 DATAFUER NEUTRONENSTERNE. WENN EIN
 5025 DATARIESENSTERN ZUSAMMENBRICHT UND IN
 5026 DATAEINER SUPERNOVA EXPLODIERT - WERDEN DIE
 5027 DATATOMS IN SEINEM KERN ZUSAMMENGEPRESST -
 5028 DATASO DASS ELEKTRONEN IN DIE ATOMKERNE EIN-
 5029 DATADRINGEN UND NEUTRONEN ENTSTEHEN LASSEN.
 5030 DATADER STERN ENDET ALS NEUTRONENBALL -
 5031 DATADESSEN DURCHMESSER NUR WENIGE KILOMETER
 5032 DATAMISST - ABER SO DICHT IST - DASS EIN
 5033 DATATEIL IN DER GROESSE EINES ZUCKERSTUECKS
 5034 DATAMILLIONEN VON TONNEN WIEGEN WUERDE.
 5035 DATAWENN EIN STERN ZUSAMMENBRICHT NIMMT
 5036 DATAEINE SCHWERKRAFT UNTER DEM EXTREMEN
 5037 DATADRUCK - UNTER DEM SEINE MATERIE STEHT -
 5038 DATAZU. WUERDE DER ZUSAMMENBRUCH UEBER DAS
 5039 DATASTADIUM EINES NEUTRONENSTERNS HINAUS-
 5040 DATAREICHEN - HAETTE SELBST LICHT KEINE
 5041 DATACHANCE MEHR - SEINEM SCHWERKRAFTFELD ZU
 5042 DATAENTFLIEHEN! SOMIT WUERDE DER KOERPER
 5043 DATAVOELLIG SCHWARZ WERDEN. JEDER NAEHERNDE
 5044 DATAGEGENSTAND WUERDE IN DIESES SCHWARZE
 5045 DATALOCH HINEINGEZOGEN WERDEN. OBWOHL VIELE
 5046 DATAHINWEISE DIE EXISTENZ DER SCHWARZEN
 5047 DATALOECHER BESTAETIGEN - BLIEBEN SIE BIS
 5048 DATAJETZT UNERWIESENE THEORIE. DENNOCH
 5049 DATAWERDEN SIE IN DER MODERNEN ASTRONOMIE
 5050 DATASTRENDIG ALS ARBEITSMODELL EINGESETZT.
 5051 DATAEIN SOLCHES LOCH KOENNTE JAEHRlich EINE
 5052 DATASONNENMASSE IN FORM BENACHBARER MATERIE
 5053 DATAEINFANGEN. BEOBACHTUNGEN AN DER 60 MIO.
 5054 DATALICHTJAHRE ENTFERNTEN GALAXIS M 87 IM
 5055 DATASTERNBILD DER JUNGFRAU GEBEN BERECHTIG-
 5056 DATATEN ANLASS ZU DER ANNAHME - DASS IM KERN
 5057 DATADIESER GALAXIE EIN OBJEKT VON 5 MRD.(!)
 5058 DATASONNENMASSEN SITZT - EIN SCHWARZES LOCH.
 5060 :
 5061 DATALICHT - DAS EINEN,NORMALEN STERN VER-,LAESST - WIRD VON
 5062 DATADESSEN SCHWEREFELD,KAUM BEEINTRACHTIGT,- BEI EINEM NORMALEN
 5063 DATASTERN WIE DER SONNE,IST DAS LICHT GELB.
 5064 DATAAM ENDE SEINES LE-,BENS NIMMT DIE AN-,ZIEHUNGSKRAFT ENORM
 5065 DATAZU. DARAUSS FOLGT,EINE BREITE ROTVER-,SCHIEBUNG - DIE SEIN
 5066 DATALICHT 'ROETER' ER-,SCHEINEN LAESST.
 5067 DATAUNTER ROTVERSCHIEBUNG VERSTEHT MAN DIE
 5068 DATAVERLAENGERUNG DER WELLENLAENGE DES
 5069 DATALICHTS - DAS VON EINEM SICH ENTFERNEN-
 5070 DATADEN OBJEKT AUSGESTRAHLT WIRD. SIE WIRD
 5071 DATADURCH DEN DOPPLER-EFFEKT VERURSACHT.
 5072 DATADAS AUSMASS DER ROTVERSCHIEBUNG SAGT
 5073 DATAUS - MIT WELCHER GESCHWINDIGKEIT SICH
 5074 DATADER KOERPER BEWEGT.
 5075 DATAWENN DER STERN ZU-,SAMMENBRICHT UND ZU,EINEM SCHWARZEN LOCH
 5076 DATAWIRD - KOENNEN,SELBST LICHTSTRAHLEN,DEM SCHWERKRAFT-
 5077 DATABEREICH NICHT,WIDERSTEHEN.
 5078 :
 5080 DATAIM FOLGENDEN LETZTEN KAPITEL WERDEN DREI
 5081 DATATHEORIEN BEHANDELT - DIE AUFGESTELLT
 5082 DATAWURDEN - UM DEN URSPRUNG DES UNIVERSUMS
 5083 DATAZU ERKLAEREN. SELBSTVERSTANDLICH
 5084 DATALEIBT JEDE THEORIE NUR EIN
 5085 DATAGEDANKENSPIEL - OBWOHL ES HINWEISE
 5086 DATAGIBT - DIE ZUMINDEST DIE WAHRSCHEINLICH-
 5087 DATAKEIT EINIGER BEHAUPTUNGEN ERHOEHEN.
 5090 :
 5091 DATANACH DIESER THEORIE,WIRD SICH DIE GEGEN-,WAERTIGE AUSDEHNUNG
 5092 DATADES UNIVERSUMS VER-,LANGSAMEN UND EINMAL,GANZ AUFHOEREN.
 5093 DATAENDE
 5094 DATADANN WIRD SICH DAS,UNIVERSUM ZUSAMMEN-,ZIEHEN.,ENDE
 5095 DATASCHLIESSLICH WIRD ES,ZU EINER GEWALTIGEN,EXPLOSION KOMMEN.
 5096 DATADABEI WIRD DIE,MATERIE AUSEINANDER-,GERISSEN UND BILDET
 5097 DATANEUE GALAXIEN - DIE,VONEINANDER FORT-,STREBEN - BIS
 5098 DATAUCH DIESE AUS-,DEHNUNG STAGNIERT -,UND SICH DER VORGANG
 5099 DATAWIEDERHOLT,WIEDERHOLT,WIEDERHOLT...ENDE
 5100 DATADIESE THEORIE GEHT,DAVON AUS - DASS NUR,EINE EXPLOSION VOR
 5101 DATAETWA 20000 MIO.,JAHREN STATTFAND.,ENDE
 5102 DATAES DEHNT SICH,FORTWAEREND AUS.,ENDE
 5103 DATADIESE THEORIE DES,STATIONAEREN UNIVER-,SUMS GEHT DAVON AUS
 5104 DATADASS DIE AUSDEHNUNG,UNENDLICH IST. ES,WIRD BEHAUPTET -

```

5105 DATADASS DAS UNIVERSUM KEINEN URSPRUNG HAT,UND NIE ENDEN WIRD.
5106 DATANEUE GALAKSIEN BILDEN,SICH SPONTAN UND,NEHMEN DIE DURCH DIE
5107 DATAEXPANSION ENT-,STANDENEN LEER-,STELLEN EIN.,ENDE
5108
5109 DATAICH HOFFE - SIE HABEN EINIGE DINGE ER-
5110 DATAFAHREN - DIE IHNEN UNBEKANNT WAREN.
5111 DATA-----
5112 DATA      WIEDERHOLUNG ?      -      'J'
5500 RETURN
6000 YP=142.EZ=0
6005 BLOCK0,142,159,199,0
6010 READTE$:IFTE$="ENDE"THENRETURN
6015 TEXT0,YP,TE$,3,1,8      7030 FCHR 4,10,10,10,96:FORW=1T050:NEXT
6020 FORW=1T01000:NEXT      7040 FCHR 4,30,10,10,96:FORW=1T050:NEXT
6030 YP=YP+9      7050 RETURN
6040 EZ=EZ+1      8000 REM ***** TEXTAUSGABE
6050 IF EZ<6THEN6010      8010 L=LEN(TE$):PRINT" "
6060 GETT$:IFT$=""THEN6060      8012 RE=(40-L)/2
6070 BLOCK0,142,159,199,0      8020 FORJ=0TOL
6080 GOTO6000      8030 POKE211,J+RE:POKE214,2:SYS58640:PRINTMID$(TE$,J+1,1):NEXT
6090 BLOCK2,42,157,138,1      8040 RETURN
6505 FA=2:FORA=1TOSM      9000 REM ***** TASTENDRUCK ABWARTEN
6510 XX=INT(RND(1)*X0)+XA      9010 PRINTAT(0,22)"  " WEITER -- EINE TASTE DRUECKEN "
6520 YY=INT(RND(1)*Y0)+YA      9020 GETT$:IFT$=""THEN9020
6530 PLOTXX,YY,FA      9024 POKESI+4,33:POKESI,100:POKESI+1,200
6540 IFFA=2THENFA=3:ELSE FA=2:9026 FORW=1T0200:NEXT:POKESI+4,0
6550 NEXT      9030 RETURN
7000 REM ***** TEXTFELD LOESCHEN
7010 FCHR 4,0,10,10,96:FORW=1T050:NEXT
7020 FCHR 4,20,10,10,96:FORW=1T050:NEXT

```

Computer im Fernsehen

01.10.84	ZDF
Mikroprozessoren (2)	16.04 h
28.09.84	SW3
Mikro-Prozessoren-Computer	21.15 h
2. Computer in der Produktion	

Alles, was sich im Fernsehen mit dem unerschöpflichen Thema Homecomputer beschäftigt, nennen wir Ihnen jeweils an dieser Stelle, damit Sie genau zur rechten Zeit Ihren Fernseher einschalten können und nichts wesentliches versäumen. Ein toller COMPUTE MIT...-Service – denn COMPUTE MIT...-Leser wissen Bescheid und können mitreden.

Liebe Freunde von COMPUTE MIT....!

WERSIBOARD FÜR COMMODORE C 64/SX 64

MUSIC 64

PERSONAL
COMPUTER
MUSIC

Das WERSIBOARD-SYSTEM MUSIC 64 besteht aus einem Orgel-Manual im Commodore Design, einem Interface-Modul und zugehöriger Software. Gemeinsam mit dem Commodore VC 64 entsteht ein Musikinstrument mit bemerkenswerten Fähigkeiten.

Komplett-Paket bestehend aus:

KEYBOARD

- 49 Tasten, 4 Oktaven C-C
- PROFIFORMAT
- Gehäuse aus stoßfestem Kunststoff im Commodore-Design
- Interface-Modul mit Verbindungskabel, anschlussfertig

SOFTWARE

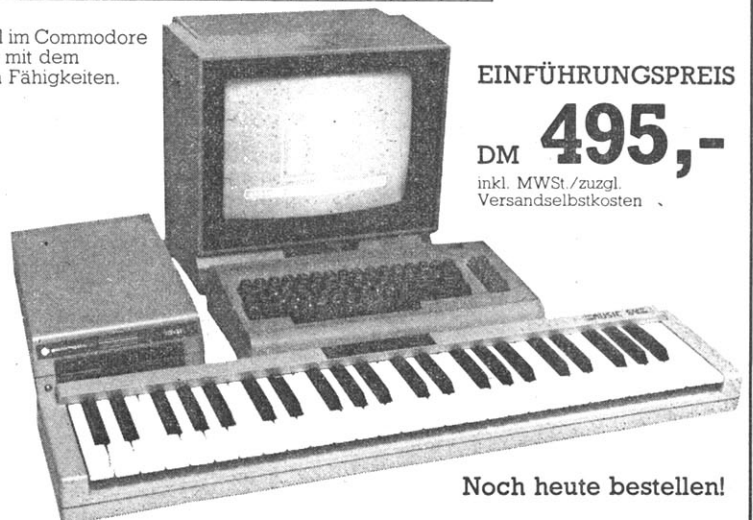
- auf 5¼" Diskette
- Programm MONO 64 — monophoner Synthesizer
- Programm POLY 64 — polyphones Keyboard
- Klangfarben direkt am PC veränderbar

WERSI Orgel- und Piano-Bausätze Industriestraße · 5401 Halsenbach
Telefon (06747) 7131 Telex 42323

BESTELL-COUPON abtrennen und senden an: WERSI electronic, Industriestraße, 5401 Halsenbach, Telefon 06747/7131
Hiermit bestelle ich/wir zu Ihren Allgemeinen Verkaufs-, Lieferungs-, Zahlungs- und Servicebedingungen zur sofortigen Lieferung

☐ Stück ☐ WERSIBOARD MUSIC 64 Einzelpreis DM 495,- ☐ Gesamtpreis DM _____
Zahlung erfolgt: ☐ per Nachnahme ☐ per beigefügtem Scheck ☐ zuzüglich Versandselbstkosten ☐ per Vorauskasse ☐ Überweisungsbeleg ist beigefügt
☐ Übersenden Sie mir vor Lieferung die Rechnung ☐ Überweisungsbeleg ist beigefügt

Absender: _____
21 E _____ Ort/Datum _____ Unterschrift _____



EINFÜHRUNGSPREIS

DM **495,-**

inkl. MwSt./zugl. Versandselbstkosten

Noch heute bestellen!



für den Commodore 64

Es ist nicht leicht einen Planeten zu besiedeln, und genau das bekommen Sie in diesem Spiel so richtig zu spüren. Kaum steht ein Teil der Raumstation, werden Sie auch schon von feindlichen Raumfahrzeugen angegriffen. Waffen konnte Ihr Schiff jedoch nicht mehr transportieren, denn sonst wäre kein Platz für die Baumaterialien der Station dagewesen.

Da gibt es also nur eine einzige Lösung: Schutzschirme auf volle Leistung und dann ran an den Feind. Versuchen Sie so die Lander und heimtückischen Minen aufzuhalten, damit sie Ihnen nicht den Platz streitig machen können! Ihr Raumschiff "Olivaro", daß Sie am Controlport 2 steuern können, hilft Ihnen dabei.

Für jede durchlaufende Runde, die von den Treffern Ihres Raumers abhängen, gibt es einen Bonus, der optisch und musikalisch angezeigt wird. Jede Runde steigt aber auch das Tempo, was die ganze Mission noch schwieriger gestaltet.

Sollte es jedoch einem der Landungsfahrzeuge trotz Ihrer Abwehrversuche gelingen auf dem Planeten zu landen, ist für Sie das Spiel zu Ende.

Das Programm besitzt außerdem noch eine Highscore-Wertung und eine MC-Bewegungsroutine, die das Spiel realistischer, schneller und flüssiger gestaltet.

Variablendefinition:

HY,I,Z,X,T,K,LA = Schleifenvariablen

V = Basicadresse des VIC

YS = Highscore

RO = Rundenanzahl

U = bestimmt Art der Angreifer (Mine o. Lander)

SP = Grundtempo der Angreifer

X2 = legt Register für X-Koordinaten der Sprites 0 und 1 fest

X1 = legt Register für y-Koordinaten der

Sprites 0 und 1 fest

EN = Variable für Sprite ein/aus der Angreifer

XK = x-Koordinate der Gegner

RA = beinhaltet und kontrolliert augenblicklichen Stand des Gegners auf der y-Koordinate

PU = Punktzahl für getroffenen Gegner

AG = kontrolliert Trefferanzahl je Runde

S1 = kontrolliert Anzahl der Bonusrunden

SK = erhöht Spieltempo nach jeder dritten Runde

Musik-Konstanten:

FH,FL = Frequenz

TH,TL = Tastverhältnis

W = Wellenform

A = Anschlag/Abschwellen

H = Halten/Ausklängen

L = Lautstärke

Zeilenerläuterungen:

10-20: Konstanten für Video- und Musikkontrollen festlegen

30: Unterprogramm: Sprites, Maschinenprogramme und neue Zeichen einlesen

40-50: Spritedefinitionen in Blöcke unterbringen

60-80: Farbendefinition der Sprites

90-100: Sprites und Kollisionsregister löschen

110-290: Highscore festlegen und Bildschirmausgabe des Anfangsbildes

300-310: Joystickabfrage für Spielbeginn

320-340: Variablen und Bildschirm löschen

350-510: Spielstart und Aufbau des Spielbildes

520: MC-Routine: X-Register mit Raumerposition laden

530: Raumerposition festlegen

540-560: Spielwertungen festlegen

570-590: Aussehen der Angreifer per Zufall bestimmen

600-610: Konstanten für Angreifer definieren

620: Sprites einschalten

630-660: Position des Gegners bestimmen und gleichzeitig auf Überschreitung kontrollieren

670: Kollisionsabfrage

680: MC-Routine: Raumerbewegung

690: erneuter Durchlauf des Bewegungsforganges

700-730: Kollisionssound

740: Punkt- und Trefferzahl erhöhen

750: Trefferzahl überprüfen

760: zurück für weiteren Durchlauf

770: Rundenzahl und Tempo erhöhen

780: Sprites ausschalten

790-830: Bildschirmausgabe: Bonus

840-880: Punktzahl erhöhen und Bonus-sound spielen, danach zurück zum erneuten Spieldurchlauf

890-940: Schlußmelodie und zurück zum Anfangsbild

1000-1080: MC-Routine: Verschiebung und Copierung des Zeichengenerators

1090-1270: Neue Zeichen einlesen

1280-1330: MC-Bewegungsroutine einlesen

1340-1550: Sprites definieren und einlesen, zurück zum weiteren Spieldurchlauf

Anmerkung:

Mit Raumer ist das Gefährt des Spielers gemeint!

Gegner oder Angreifer sind die feindlichen Objekte, die der Spieler zu neutralisieren hat.


```

1 REM =====
2 REM = =
3 REM = ANDREAS BACHLER =
6 REM = =
7 REM =====
10 V=53248
20 SI=54272;FL=SI;FH=SI+1;TL=SI+2;TH=SI+3;W=SI+4;A=SI+5;H=SI+6:L=SI+24
30 GOSUB 1000
40 REM SPRITES BEEINFLUSSEN
50 POKE2040,11:POKE2041,13:POKE2042,14
60 POKEV+28,6
70 POKEV+37,10:POKEV+38,6:POKEV+39,1
80 POKEV+40,15:POKEV+41,12
90 REM SPRITES LOESCHEN
100 POKEV+21,0:POKEV,0:POKEV+1,0:POKEV+2,0:POKEV+3,0:POKEV+30,0
110 IFYS>HITHENHI=YS
120 REM BILDAUFBAU
130 POKEV+32,0:POKEV+33,0
140 PRINT"....."
150 PRINT"....."
160 PRINT"....."
170 PRINT"....."
180 PRINT"....."
190 PRINT"....."
200 PRINT"....."
210 PRINT"....."
220 PRINT"....."
230 PRINT"....."
240 PRINT"....."
250 PRINT"..... 1984 BY ANDREAS BACHLER"
260 PRINT".....PRESS FIRE FOR START"
270 PRINT"....."
280 PRINT".....HIGHSCORE";HI
290 PRINT".....LAST SCORE";YS
300 IFPEEK(56320)=111THENGOTO320
310 GOTO300
320 REM VARIABLEN LOESCHEN
330 YS=0:SK=0:RO=0:AG=0:S1=0
340 PRINT":RA=0:REM SPIELAUFB AU
350 FORHY=1TO25:K=INT(RND(1)*800):LA=INT(RND(1)*800)
360 POKE1024+K,46:POKE55296+K,7:POKE1024+LA,138:POKE55296+LA,1:NEXT
370 PRINT":FORI=1TO24:PRINTTAB(32)"":NEXT
380 PRINTTAB(32)"":
390 PRINT"CEAEAEAEAEAEAEAEAEAEAEAEAEAEAEAEAH"
400 PRINT"DFBFBFBFBFBFBFBFBFBFBFBFBFBFBFBFBG";
410 POKE1119,137:POKE55391,1
420 POKE1210,137:POKE55482,1
430 POKE1391,137:POKE55663,1
440 POKE1482,137:POKE55754,1
450 POKE1609,137:POKE55881,1
460 POKE1632,137:POKE55904,1
470 PRINT"KMORNP"
480 PRINT"SPC(34)"SCORE"
490 PRINT"SPC(34)"HIGH-"
500 PRINTSPC(34)"SCORE"
510 PRINT"SPC(34)"ROUND"
520 SYS51252
530 POKEV+4,128:POKEV+5,215
540 PRINT"SPC(33);YS
550 PRINT"SPC(33);HI
560 PRINT"SPC(35);RO
570 REM SPIELBEGINN
580 U=INT(RND(1)*2)+1
590 ONUGOTO600,610
600 EN=1:PU=7:SP=4:X1=1:X2=0:GOTO620
610 EN=2:PU=5:SP=3:X1=3:X2=2
620 POKEV+21,EN+4
630 XK=INT(RND(1)*231)+25
640 POKEV+X2,XK
650 RA=RA+SP+SK
651 POKEV+X1,RA
660 IFRA>212THENGOTO890
670 IFPEEK(V+30)=EN+4THENGOTO700
680 SYS49152
690 GOTO650

```

```

700 REM KOLLISION
710 POKEV+21,PEEK(V+21)-EN:POKEV+30,0:POKEV+X1,0:POKEV+X2,0:RA=0
720 POKEL,15:POKEW,33:POKEA,36:POKEH,255
730 FORX=1TO10:FORZ=40TO80:POKEFH,Z:NEXTZ,X:POKEW,0
740 YS=YS+PU:AG=AG+1
750 IFAG=15THENAG=0:GOTO770
760 GOTO540
770 RO=RO+1:S1=S1+1:IFS1=3THENS1=0:SK=SK+1
780 POKEV+21,0
790 PRINT"KOLLISION"
800 PRINT"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
810 PRINT"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
820 PRINT"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
830 PRINT"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
840 YS=YS+RO*10
850 POKEL,15:POKEW,17:POKEA,36:POKEH,255
860 POKETH,8:POKETL,8
870 FORI=1TO15:POKE53281,I:FORZ=0TO40STEP.5:POKEFH,Z:NEXTZ,I
880 POKEW,0:POKE53281,0:FORI=1TO1500:NEXT:GOTO340
890 REM SCHLUSSMELODIE
900 POKEL,15:POKEW,17:POKEA,36
910 POKEH,255:POKETH,8:POKETL,8
920 FORI=1TO10:FORZ=80TO40STEP-1
930 POKEFH,Z:NEXTZ,I:POKEW,0
940 GOTO100
1090 FORX=0TO135:READT:POKE13312+X,T:NEXT
1100 REM ZEICHEN-DATAS
1110 DATA 60,66,153,161,161,153,66,60
1120 DATA 255,255,112,56,28,14,199,195
1130 DATA 195,199,14,28,56,112,255,255
1140 DATA 255,255,240,240,220,206,199,195
1150 DATA 195,199,206,220,248,240,255,255
1160 DATA 255,255,14,28,56,112,224,195
1170 DATA 195,224,112,56,28,14,255,255
1180 DATA 195,227,115,59,31,15,255,255
1190 DATA 255,255,15,31,59,115,227,195
1200 DATA 0,0,16,56,254,56,16,0
1210 DATA 0,0,0,0,16,0,0,0
1220 DATA 0,0,1,3,63,99,199,100
1230 DATA 63,8,7,3,1,0,0,0
1240 DATA 60,255,255,255,255,255,255,0
1250 DATA 255,0,255,255,255,255,60,0
1260 DATA 0,0,128,192,252,230,243,22
1270 DATA 252,16,224,192,128,0,0,0
1280 REM BEWEGUNGS-ROUTINE IN MC
1290 FORI=0TO43:READT:POKE49152+I,T:NEXT
1300 DATA 169,224,141,2,220,173,0,220
1310 DATA 41,8,240,8,173,0,220,41,4,240
1320 DATA 11,96,138,24,105,3,176,17,170,76,40,192,138
1330 DATA 56,233,33,144,7,138,233,3,170,153,4,208,96
1340 REM SPRITES DEFINIEREN
1350 FORI=0TO62:READT:POKE704+I,T:NEXT
1360 FORI=0TO62:READT:POKE832+I,T:NEXT
1370 FORI=0TO62:READT:POKE896+I,T:NEXT
1380 REM SPRITE DATAS
1390 DATA 0,0,0,3,0,0,1,224,0,3,192,0
1400 DATA 0,96,0,0,60,0,0,255,0,3,255,192
1410 DATA 119,255,238,118,73,46,126,73,62
1420 DATA 118,73,46,118,255,46,35,255,196,38
1430 DATA 255,100,3,153,192,0,219,0,1,129,128
1440 DATA 3,0,192,6,0,96,31,129,248
1450 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0:REM 16 STUECK
1460 DATA 32,0,0,32,0,0,168,0,2,230,0
1470 DATA 11,229,128,47,223,96,191,229,88
1480 DATA 170,170,168,37,111,224,9,111,128
1490 DATA 2,110,0,0,168,0,0,136,0,0,136,0,0,32,0,0,32,0
1500 DATA 0,32,0,0,32,0,0,32,0,0,168,0,0,168,0
1510 DATA0,168,0,0,168,0,2,170,0,2,2,0
1520 DATA2,170,0,2,2,0,10,34,128,10,170,128
1530 DATA10,170,128,10,170,128,4,16,64
1540 DATA17,73,16,20,16,80,1,69,0,0,0,0,0,0,0
1550 RETURN
READY.

```

FFERDE RENNEN

für den Commodore 64

Das folgende Programm ist eine Simulation, bei der die teilnehmenden Spieler jede gültige Wettart der Rennbahnen spielen dürfen, mit Ausnahme der Systemwetten.

Die Zeilen 20 bis 35 dienen zum Einlesen der Sprites, sowie des Maschinenprogrammes.

In den Zeilen 60 bis 75 werden den Variablen Anfangs- bzw. Festwerte zugewiesen. Die Zeilen 79 bis 139 beinhalten die Instruktionen. Dabei wird in den Zeilen 86-95, 113-117 und 135-139 die Speicherplatz-Nr. der Sprites zum Ein- oder Ausschalten der Geräuschgeneratoren benutzt. Dadurch erklingen hufschlagähnliche Laute. In den Zeilen 150 bis 225 werden die Anzahl der Spieler festgestellt, sowie das Startgeld erteilt. Außerdem werden die Namen der Spieler festgehalten, wobei darauf geachtet wird, daß kein Name länger als elf Buchstaben ist. Die Zeilen 270 bis 450 dienen der Feststellung über Wettart, Einsatz und Tip. Bei der Frage nach der Wettart befindet sich der Cursor schon auf dem Wort 'Sieg'. Möchte man nun auf

Sieg setzen, so genügt der Druck auf die Returntaste. Ansonsten steuert man mit 'Cursor ab' die gewünschte Zeile an und drückt 'Return'. Will der Spieler das kommende Rennen auslassen, so gibt er 'N' ein, gefolgt von 'Return'.

Der Einsatz wird auf ganze Zahlen begrenzt, um einen gleichmäßigen Ausdruck zu erhalten.

Das Rennen selbst ist in den Zeilen 780 bis 1010 festgehalten. Dazu wird den Pferden nach den Startvorbereitungen (780-800) in der Zeile 805 eine zufällige Bewegungsform zugeteilt, um im Rennen keinen Gleichlauf der Pferde zu erhalten. Hier setzt nun das oben erwähnte Maschinenprogramm ein (820). Es sorgt dafür, daß bei jedem dritten Durchlauf die Pferde ihr Bewegungsbild ändern, und so einen flüssigen Ablauf erhalten. Nun wird durch Zufallsentscheid ein Pferd gewählt und nach Prüfung, ob eine bestimmte

Position erreicht wurde (910), eine kleine Strecke vorgerückt (850 bzw. 920). Sollte die festgelegte Position erreicht sein, so wird entweder das MSB im Reg. 16 des VIC für das entsprechende Sprite gesetzt (940), oder es wird in die Zeilen 950 bis 990 verzweigt, um die Reihenfolge des Einlaufs festzustellen. Wenn dies dreimal geschehen ist, so wird in den Zeilen 1020 bis 1130 das Ergebnis mitgeteilt und ausgewertet, Gewinne angezeigt und zum Besitz dazu gezählt. In den Zeilen 2000 bis 2295 werden der Besitz, Einsatz und Tip tabellarisch festgehalten. Dies geschieht mit Hilfe einer nachgeahmten 'Printusing'-Anweisung (2200 und 2240).

Die Zeilen 3000 bis 4040 enthalten das Unterprogramm zur Erzeugung der Hufgeräusche.

BEI SIEG MUSS DEIN PFERD GEWINNEN!

BEI PLATZ MUSS ES UNTER DEN ERSTEN

DREI SEIN *WEITER EINE TASTE*

RENNEN NR. 1. ES STARTEN 7 PFERDE

SPIELER	BESITZ	EINSATZ	TIP
DETLEF	1000	0	

DETLEF WAS MOECHTEST DU WETTEN:

? SIEG

PLATZ

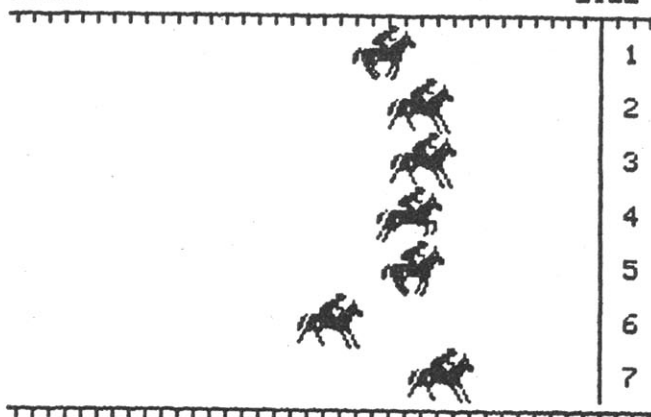
ZWEIERWETTE

DREIERWETTE

START

RENNEN NR. 1

ZIEL




```

0 REM *****
1 REM *
2 REM * P F E R D E - R E N N E N *
3 REM *
4 REM * F U E R D E N C 6 4 *
5 REM *
6 REM * C 1984 BY DETLEF GUETLICH *
7 REM *
8 REM *
9 REM *
10 REM *****
11 :
15 A=0:B=0:AA=704:X=RND(-TI):POKE53280,14:POKE53281,14
16 PRINT "*****"SPC(11)"EINEN MOMENT BITTE!!!"
17 :
18 REM ***** SPRITE'S UND MP INITIALISIEREN *****
19 :
20 FORI=0TO62:READJ:POKE832+I,J:NEXT
25 FORI=0TO62:READJ:POKE896+I,J:NEXT
30 FORI=0TO62:READJ:POKE960+I,J:NEXT
35 FORI=0TO63:READJ:POKE704+I,J:NEXT
40 DATA0,3,128,0,3,192,0,15,136,0,62,8,0,126,28,0
41 DATA251,180,0,240,126,0,120,255,0,29,251,23,255,240,111,255
42 DATA240,207,255,224,222,255,224,158,255,124,29,129,228,59,0,36
43 DATA46,0,36,72,0,40,80,0,64,144,0,0,32,0,0
45 DATA0,3,128,0,3,192,0,15,136,0,62,8,0,126,28,0
46 DATA251,180,0,240,126,0,120,255,23,253,251,111,255,240,111,255
47 DATA240,222,255,224,222,255,224,157,135,96,59,129,176,33,0,144
48 DATA65,0,72,64,128,72,128,128,36,0,64,36,0,0,13
50 DATA0,3,128,0,3,192,0,15,136,0,62,8,0,126,28,0
51 DATA251,180,0,240,126,0,120,255,0,29,251,23,255,240,63,255
52 DATA240,111,255,224,111,191,224,111,191,224,71,65,192,7,193,192
53 DATA2,129,64,3,130,64,1,66,128,0,164,128,0,65,0
56 :
57 REM ***** VARIABLEN FESTLEGEN *****
58 :
59 DIMPN$(10),PB(10),PE(10),PT$(10),PW$(10),PS(13),PM(6),PO(6)
65 V=53248:S1=54272:FORI=0TO13:PS(I)=V+I:NEXTI
68 Y=1:X=2040:FORI=0TO6:PM(I)=X+I:PO(I)=Y:Y=Y*2:NEXTI
70 Y=75:FORI=3TO13STEP2:Y=Y+23:POKEPS(I),Y:POKEPS(I-1),14:NEXTI
72 POKEV+39,0:POKEV+40,1:POKEV+41,2:POKEV+42,15
74 POKEV+43,9:POKEV+44,7:POKEV+45,11
75 FORI=0TO6:POKEPM(I),14:NEXT
76 :
77 REM ***** INSTRUKTIONEN *****
78 :
79 GOSUB3530:POKEPS(0),168:POKEPS(1),68:POKEV+21,1
80 PRINT "*****"
81 PRINT "*****"
82 PRINT "***** P F E R D E - R E N N E N *****"
83 PRINT "*****"
84 PRINT "*****";
85 PRINT "*****MOECHTEST DU INSTRUKTIONEN (J/N)";
86 POKEI+4,PEEK(2040)+3:ONPEEK(2040)-12GOTO87,83,89
87 POKE2040,14:GOTO90
88 POKE2040,15:GOTO90
89 POKE2040,13
90 GETA$:IFA$="N"THEN140
95 IFA$(">")THENPOKEI+11,PEEK(2040)+3:GOTO80
100 PRINT "*****HERZLICH WILLKOMMEN AUF DER RENNBahn!!"
102 PRINT "***** DU KANNST SIEG UND PLATZ WETTEN. "
104 PRINT "***** AUSSERDEM DIE ZWEIER-WETTE UND DIE "
106 PRINT "DREIER-WETTE!"
108 PRINT "BEI SIEG MUSS DEIN PFERD GEWINNEN!"
110 PRINT "BEI PLATZ MUSS ES UNTER DEN ERSTEN"
112 PRINT "DREI SEIN! *WEITER EINE TASTE*";
113 POKEI+4,PEEK(2040)+3:ONPEEK(2040)-12GOTO114,115,116
114 POKE2040,14:GOTO117
115 POKE2040,15:GOTO117
116 POKE2040,13
117 GETA$:IFA$=" "THENPOKEI+11,PEEK(2040)+3:GOTO104
120 PRINT "***** BEI DER ZWEIER-WETTE MUESSEN DIE "
122 PRINT "ERSTEN ZWEI, BEI DER DREIER-WETTE"
124 PRINT "***** Sogar die ersten drei in der richtigen "
126 PRINT "REIHENFOLGE GETROFFEN WERDEN!"
128 PRINT "DIE QUOTEN BETRAGEN FUER:"

```

VERWENDETE BILDSCHIRM- STEUERZEICHEN

␣	BILDSCHIRM LOESCHEN
␣	CURSOR HOME
␣	CURSOR AB
␣	CURSOR RECHTS
␣	CURSOR AUF
␣	CURSOR LINKS
■	FARBE SCHWARZ
■	FARBE WEISS
■	FARBE ROT
■	FARBE BLAU
■	FARBE GELB
■	FARBE GRUEN

VARIABLEN-LISTE

A,B	HILFSVARIABLEN
A\$	ZEILE LOESCHEN
AA	STARTADRESSE MP
AA\$	TEXT 'START----
AP()	ANZAHL SPIELER
B\$	ABFRAGE J/N
EL	ZAehler REIHENFOLGE
I,J	SCHLEIFENZAehler
P\$	HILFSSTRING
PB()	BESITZ
PE()	EINSATZ
PM()	SPRITE-SPEICHERADR.
PN\$()	NAMEN DER SPIELER
PO()	MSB VON REG. 21
PS()	SPRITE-POSITION
PT\$()	TIP
PW\$()	WETTART
PY()	KONTR. SPRITE-POS.
RH	RENNEN NR.
SI	STARTADR. SOUNDCHIP
TP\$()	ERGEBNIS
V	STARTADR. VIDEOCHIP
X	HILFSVARIABLE
X1\$	HILFSSTRING
X2\$	HILFSSTRING
Y	HILFSVARIABLE

```

130 PRINT"■ DEN 3SIEG 7:1, DEN 3PLATZ 2:1 "
132 PRINT"■ DIE 3ZWEIER-WETTE 20:1, DIE 3DREIER-"
134 PRINT"■ WETTE 120:1";
135 POKES1+4,PEEK(2040)+3:ONPEEK(2040)-12GOTO136,137,138
136 POKE2040,14:GOTO139
137 POKE2040,15:GOTO139
138 POKE2040,13
139 GETA$:IFA$=""THENPOKES1+11,PEEK(2040)+3:GOTO124
140 A$=""
143 PRINT"#####";FORI=1TO13:PRINTA$:NEXT
145 PRINTA$:POKEV+21,0:POKES1+4,16:POKES1+11,16
147 :
148 REM ***** ANZAHL SPIELER *****
149 :
150 PRINT"##### WIEVIEL SPIELER MOECHTEN MITMACHEN";:INPUTAP
155 IFAP>1ANDAP<11ORAP=1THEN170
160 PRINT"##### NUR ZWISCHEN 1 UND 10 BITTE!! ";
165 FORI=1TO2000:NEXTI:GOTO150
170 AP=INT(AP):A$="" START --- "
180 :
182 REM ***** STARTGELD ZUWEISEN *****
184 :
190 FORI=1TOAP:PB(I)=1000:NEXT
192 :
194 REM ***** NAMEN FESTSTELLEN *****
196 :
200 FORI=1TOAP
210 PRINT"#####"A$"■ NAME DES"11. SPIELERS";
220 INPUTP$:X=23:IFI=10THENX=24
225 PN$(I)=MID$(P$,1,11):NEXTI
227 :
228 REM ***** DIE RENNEN ZAEHLEN *****
229 :
230 FORRN=1TO10
240 :
250 REM ***** WETTART, EINSATZ UND TIP FESTHALTEN *****
260 :
270 GOSUB2000
280 FORI=1TOAP:IFPB(I)=0THEN430
290 PRINT"#####";PN$(I);" WAS MOECHTEST DU WETTEN: "
300 PRINT"■ 3SIEG"
310 PRINT"■ 3PLATZ"
320 PRINT"■ 3ZWEIERWETTE"
330 INPUT"■ 3DREIERWETTE#####";P$
340 PW$(I)=MID$(P$,1,1)
350 IFPW$(I)="S"ORPW$(I)="P"ORPW$(I)="2"ORPW$(I)="D"THEN370
355 IFPW$(I)="N"THEN430
360 GOSUB2000:PRINT"##### DIESE WETTART GIBT ES NICHT!!! "
365 FORJ=1TO2000:NEXTJ:GOTO290
370 GOSUB2000
380 PRINT"#####";PN$(I);" WIEVIEL RISKIERST DU":
385 PRINT"DAS LIMIT LIEGT BEI 1000";:INPUTX:IFX=0THEN430
390 IFX>1000THENPRINT"■";A$:GOTO380
395 IFX<=PB(I)THENPE(I)=INT(X):GOTO410
400 PRINT"##### SOVIEL HAST DU NICHT ":PRINTA$
405 FORJ=1TO2000:NEXTJ:GOTO380
410 PB(I)=PB(I)-PE(I):GOSUB2000
420 PRINT"#####";PN$(I);" WAS TIPST DU (KEINE KORREKTUR"
425 PRINT"BEI EINGABEFehler!!";:INPUTPT$(I)
430 GOSUB2000:NEXTI
440 PRINT"##### WIR WOLLEN NUN ZUR RENNBahn GEHEN"
450 FORI=1TO3000:NEXTI
460 :
470 REM ***** RENNBahn *****
480 :
490 PRINT"■":POKE53280,5:POKE53281,5
500 PRINT"■ RENNEN NR.":RN
510 PRINT"START ZIEL"
520 PRINT"■-----";
530 PRINT"■ | "
540 PRINT" | 1"
550 PRINT" | "
560 PRINT" | "
570 PRINT" | 2"
580 PRINT" | "

```

```

590 PRINT"
600 PRINT"
610 PRINT"
620 PRINT"
630 PRINT"
640 PRINT"
650 PRINT"
660 PRINT"
670 PRINT"
680 PRINT"
690 PRINT"
700 PRINT"
710 PRINT"
720 PRINT"
740 PRINT"
750 :
760 REM ***** RENNEN *****
770 :
780 POKEPS(1),75
785 EL=0:TP$="":FORI=0TO6:PY(I)=254:POKEPS(I*2),14:POKEPM(I),14:NEXTI
790 POKEV+21,127
800 PRINT"SPC(36)";:FORI=1TO1000:NEXT:GOSUB3000
805 FORI=0TO6:POKEPM(I),INT(RND(1)*3)+13:NEXTI
810 FORI=1TO21:FORJ=1TO10
820 A=INT(RND(1)*7):B=A*2:POKE781,A:SYSAA
830 POKEPS(B),PEEK(PS(B))+2
835 FORX=1TO10:NEXTX:POKE51+11,PEEK(PM(A))+3:NEXTJ
845 PRINT"SPC(26)MID$(A$,1,10);
870 NEXTI:PRINT"
880 A=INT(RND(1)*7):B=A*2:POKE781,A:SYSAA
890 IFPEEK(PS(B))=PY(A)THEN835
900 POKEPS(B),PEEK(PS(B))+2
910 POKE51+11,PEEK(PM(A))+4:GOTO880
935 IFPY(A)=35THEN850
940 PY(A)=35:POKEV+16,PEEK(V+16)ORPO(A):POKEPS(B),1:GOTO930
944 :
945 REM ***** REIHENFOLGE DES EINLAUFS FESTSTELLEN *****
946 :
950 EL=EL+1:ONELGOTO960,980,990
960 PRINT"ERSTER:";TP$=RIGHT$(STR$(A+1),1):PRINTA+1;
970 GOTO920
980 PRINT"- ZWEITER:";TP$=TP$+"-"+RIGHT$(STR$(A+1),1):PRINTA+1;GOTO920
990 PRINT"- DRITTER:";TP$=TP$+"-"+RIGHT$(STR$(A+1),1):PRINTA+1;
1000 FORI=1TO3000:NEXTI
1010 POKEV+21,0:POKEV+16,0:GOSUB4000
1014 :
1015 REM ***** ERGEBNIS AUSWERTEN *****
1016 :
1020 POKE53260,14:POKE53281,14:GOSUB2000
1030 PRINT"DER ENDGUELTIGE RICHTERSPRUCH: ";TP$
1040 FORI=1TOAP
1050 IFPE(I)=0THEN1120
1052 IFPW(I)="N"THEN1120
1054 IFPW(I)="S"THEN1070
1056 IFPW(I)="P"THEN1080
1058 IFPW(I)="Z"THEN1090
1060 IFPW(I)="D"THEN1100
1070 IFPT(I)<>LEFT$(TP$,1)THEN1120
1075 PE(I)=PB(I)+PE(I)*7:PRINTPN(I)" GEWINNT";PE(I)*7:GOTO1120
1080 IFPT(I)=LEFT$(TP$,1)ORPT(I)=MID$(TP$,3,1)THEN1085
1083 IFPT(I)<>RIGHT$(TP$,1)THEN1120
1085 PB(I)=PB(I)+PE(I)*2:PRINTPN(I)" GEWINNT";PE(I)*2:GOTO1120
1090 IFPT(I)<>LEFT$(TP$,3)THEN1120
1095 PB(I)=PB(I)+PE(I)*20:PRINTPN(I)" GEWINNT";PE(I)*20:GOTO1120
1100 IFPT(I)<>TP$THEN1120
1110 PB(I)=PB(I)+PE(I)*120:PRINTPN(I)" GEWINNT";PE(I)*120
1120 NEXTI
1130 FORI=1TO4000:NEXTI
1140 :
1150 REM ***** EINSATZ- UND TIPVARIABLE AUF NULL SETZEN *****
1160 :
1170 FORI=1TO10
1180 PE(I)=0:PT(I)="":PW(I)=""
1190 NEXTI

```



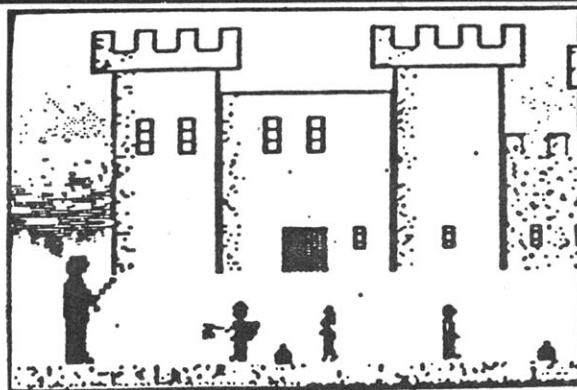
```

1200 :
1210 REM ***** NÄCHSTES RENNEN? *****
1220 :
1230 NEXT RN
1240 RN=0:GOSUB2000
1250 :
1260 REM ***** NEUES SPIEL? *****
1270 :
1280 PRINT "***** DAMIT IST DIESER RENTTAG BEENDET!!"
1290 PRINT "EIN NEUER BESUCH (J/N)?"
1300 GETB$:IFB$="J" THEN RUN
1310 IFB$(">N") THEN 1300
1320 PRINT "SCHADE!! AUF WIEDERSEHEN!!":END
1337 :
1338 REM ***** BESITZ-ANZEIGE *****
1339 :
2000 PRINT "RENNEN NR. ";RN;" ES STARTEN 7 PFERDE"
2010 PRINT " "
2020 PRINT "SPIELER | BESITZ | EINSATZ | TIP"
2030 PRINT "-----|-----|-----|-----"
2040 PRINT " "
2050 PRINT " "
2060 PRINT " "
2070 PRINT " "
2080 PRINT " "
2090 PRINT " "
2100 PRINT " "
2110 PRINT " "
2120 PRINT " "
2130 PRINT " "
2140 PRINT "-----|-----|-----|-----"
2150 PRINT " "
2160 FOR J=1 TO 4
2170 PRINT " ";PN$(J)
2180 NEXT J
2190 PRINT " "
2200 X1$=" ":FOR J=1 TO 4:X2$=X1$+STR$(PB(J))
2210 PRINT TAB(14);RIGHT$(X2$,8)
2220 NEXT J
2230 PRINT " "
2240 X1$=" ":FOR J=1 TO 4:X2$=X1$+STR$(PE(J))
2250 PRINT TAB(25);RIGHT$(X2$,5)
2260 NEXT J
2270 PRINT " "
2280 FOR J=1 TO 4
2290 PRINT TAB(33);LEFT$(PT$(J),5)
2295 NEXT J:RETURN
2337 :
2338 REM ***** GERÄUSCHKULISSE *****
2339 :
3000 POKES1+24,15:POKES1+14,250:POKES1+15,0:POKES1+19,24:POKES1+20,240
3010 POKES1+17,21:POKES1+16,6:POKES1+18,145
3020 RETURN
3500 POKES1+24,15:POKES1,250:POKES1+1,1:POKES1+5,21:POKES1+6,16
3510 POKES1+7,29:POKES1+8,2:POKES1+12,21:POKES1+13,16
3520 RETURN
3937 :
3938 REM ***** ABSCHWELLEN DES TONES *****
3939 :
4000 FOR I=14 TO 0 STEP -1:FOR J=1 TO 40:NEXT J
4010 POKES1+24,1
4020 NEXT I
4030 POKES1+4,128:POKES1+11,128:POKES1+18,128:POKES1+19,0
4040 RETURN
4337 :
4338 REM ***** DATA MASCHINENPROGRAMM *****
4339 :
5000 DATA 139,248,7,105,3,141,4,212,173,255,2,201,3,200,33,162,6,169,248,7
5001 DATA 201,13,208,6,254,248,7,76,237,2,201,14,208,6,254,248,7,76,237,2,169
5002 DATA 13,157,248,7,202,16,225,169,1,141,255,2,96,238,255,2,96,0,0,0,0
5003 DATA 1
END

```

Valhalla

für den Commodore 64



An dieser Stelle möchten wir Ihnen wieder ein Adventure vorstellen. Nach dem Auftakt in unserer Erstaussgabe, als wir das Adventure "Ulysses and the golden Fleece" besprachen, folgt heute die Legende von Valhalla.

Valhalla ist eine nordische Sage. Drachen, Wölfe, Götter, Riesen und Zwerge sind die Figuren dieses Phantasieadventures. Dabei hat jede Figur eine eigene Persönlichkeit, die in folgende Charaktere unterteilt ist:

- 1.) Güte, Redlichkeit
- 2.) göttliche Gabe
- 3.) Körperkraft
- 4.) Tapferkeit
- 5.) Intelligenz

Das Besondere an diesem Adventure ist die bewegte Grafik. Im Gegensatz zu normalen Text-/Grafikadventures bewegen sich die Figuren unabhängig von der Tastatureingabe weiter. Ebenso können die Figuren per Kommando eingabe zu Aktionen veranlaßt werden, die dann auf dem Bildschirm zu beobachten sind. Gibt man z.B. den Befehl "drink wine",

marschiert eine der Figuren zu der Weinflasche und nimmt einen Schluck.

Es gibt eine Vielzahl verschiedener Screens und Eingabemöglichkeiten. Schlösser, Burgen, Wälder und Sümpfe sind grafisch hervorragend gelungen. Da auf dem Bildschirm immer Bewegung herrscht, ist allein das Zuschauen schon faszinierend.

Da kommen und gehen die verschiedensten Gestalten (insgesamt 36 Figuren), essen und trinken, ohne das in das Spielgeschehen eingegriffen wird. Allerdings ganz ohne Eingabe geht es nicht, sonst ist das Spiel vorzeitig beendet.

So ist es z.B. unerlässlich Nahrung aufzunehmen, um nicht den qualvollen Hungertod zu sterben.

Desweiteren kann ein reger Warenhandel betrieben werden, es kann ge- aber auch verkauft werden.

Überhaupt ist Valhalla ein sehr durchstrukturiertes Programm, das machen schon die Eingabemöglichkeiten deut-

lich.

Beispiel:

"Sell axe to thor for 20 crowns."

Sinn des Adventures ist, sechs magische Gegenstände zu finden.

Diese sind überall verstreut und sind ohne Hilfe der anderen Charaktere nicht zu finden.

Um diese Hilfe zu bekommen, muß man sich aber enorm anstrengen.

Valhalla ist nicht einfach zu spielen, fasziniert aber durch qualitativ hochwertigen Programmierstil, tolle Grafik und nimmermüde Action. Dem Programm liegt ein 52-Seiten umfassendes Anleitungsheft bei, in dem sämtliche Figuren mit ihren Charakteren verzeichnet sind. Das im Moment wohl einzigartige Adventure bietet neben der garantierten Spielfreude auch genügend Anregungen für Programmautoren, die sich an Adventures versuchen wollen.

Atom Smasher

für den VC-20 ohne Erweiterung

Unseren Lesern, die in die Spuren von Otto Hahn, Ilse Meitner und Fritz Straßmann treten wollen, sei dieses Spiel wärmstens empfohlen. Von Romik aus England haben wir schon manches hübsche Spiel bekommen – was sie uns diesmal über den Kanal geschickt haben, halten wir jedoch für besonders originell.

Statt uns in galaktischen Dimensionen herumzutreiben, begeben wir uns diesmal in die Welt des Kleinsten: Ihre Aufgabe besteht darin, mit Hilfe eines Protons den Atomkern zu treffen, ohne dabei mit den Kern umkreisenden Elektro-

nen Kontakt aufzunehmen. Jede Kollision zwischen Ihrem Proton und einem Elektron führt (in Abwechslung zu ähnlichen Spielen) zu einem Elektron mehr, wodurch sich Ihre Arbeit mit der Zeit erheblich erschweren kann. Dazu kommt, daß Ihr Proton eine Halbwertszeit von 1,5 besitzt, nach Einsatz von drei "Leben" somit kampfunfähig wird.

Programmtechnisch ist Atom Smasher zwar nicht in die Spitzenkategorie der Romik-Software einzuordnen, aber dennoch ein Spiel, das man immer wieder gern laden wird.

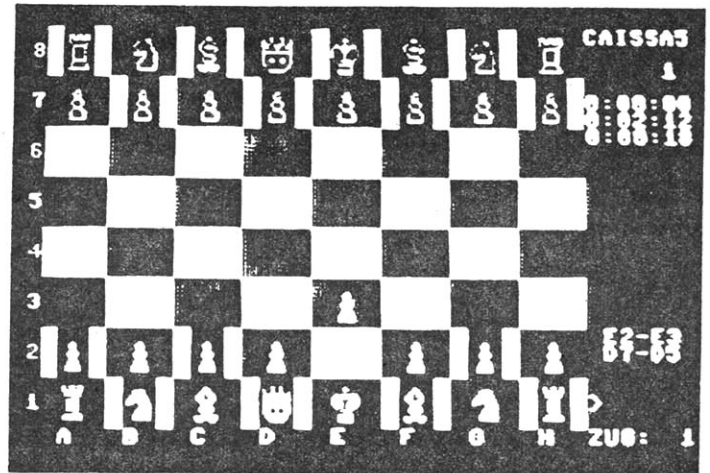
Schachprogramme im Vergleich

Das "königliche" Spiel mit dem Computer ist eines der beliebtesten Anwendungsgebiete.

Von der Spielstärke gibt es bei einigen Programmen kaum Unterschiede zu reinen Schachcomputern.

Leidenschaftliche Schachspieler greifen immer häufiger zu der Möglichkeit, ihre Können um Strategie mit dem Computer zu messen. Dafür spricht auch, daß der Computer immer Lust zum Spielen hat.

Wir stellen Ihnen zwei der anspruchvollsten Schachprogramme vor, Colossus 2.0 von CDS und Caissa von Dynamics.



Colossus

Beim Schachprogramm Colossus gibt es keine Spielstufen im eigentlichen Sinne. Die Schwierigkeitsgrade können in Form von Bedenkzeit gewählt werden (Turnier-Modus).

Desweiteren gibt es den sogenannten Equality-Mode, in dem der Computer die gegnerische Bedenkzeit als Zeitlimit annimmt.

Spielanalysen ohne Zeitangabe können ebenfalls von Colossus bewältigt werden.

Die Möglichkeit, Aufgabenstellungen z.B. aus Zeitungen nachzuspielen ist eine lobenswerte Eigenschaft.

Colossus kann Problemstellungen mit Zugvorgabe lösen (Beisp.: Schwarz zieht und ist matt in 3 Zügen), Brief- und Blitzschach spielen. Schon die Möglichkeiten der Zugeingabe sind überraschend. Neben der Tastatureingabe (d7-d5) kann auch mittels Cursor die Zugfolge gesteuert werden.

Züge können beliebig und jederzeit zurückgenommen werden, was durchaus nicht selbstverständlich ist.

Bis zu 120 Züge eines Spieles kann Colossus speichern und problemlos ab einer vorgegebenen Figurenstellung wiederholen.

Das laufende Spiel kann jederzeit unterbrochen, aufgezeichnet und später fortgesetzt werden.

Diese Eigenschaft machen das Programm für Schachclubs interessant, Spiele können detailliert analysiert

werden.

Interessant ist die Fähigkeit, alle möglichen Züge für eine bestimmte Figur aufzuzeigen.

Außergewöhnlich auch die Teilung von Text- und Grafik bei Colossus. Während auf der Grafikseite das Schachbrett dargestellt wird, sind sämtliche Informationen auf der Textseite abzurufen.

Hier kann die Bedenkzeit beider Parteien sowie die 7 letzten Züge entnommen werden.

Weiterhin können Informationen zur Analyse von Spielen abgelesen werden, z.B. Vorausberechnung von Zügen und deren Bewertung.

Allerdings verfügt Colossus nicht über Hilfsroutinen, wo normalerweise Zugvorschläge gemacht werden.

Der geübte Schachspieler kann jedoch anhand der Informationen von der Textseite einige Schlüsse ziehen.

Text- und Grafikseite sind jederzeit umschaltbar, Spielstufen, Farbe der Figuren lassen sich ebenfalls während des Spieles ändern.

Caissa

Caissa zählt zu den außergewöhnlichen Schachprogrammen.

Durch Interrupt-Steuerung ist Caissa in der Lage, bis zu 19 Halbzüge im voraus zu berechnen.

Er nutzt die Bedenkzeit seines Gegners, um die optimale Zugfolge festzulegen. Diese Züge werden ebenso angezeigt,

wie die bereits ausgeführten.

So kann ein Zug dem Gegner durchaus als unsinnig erscheinen, Caissa "denkt" im voraus.

Es ist wirklich beeindruckend, die "Denkphase" dieses Programmes zu beobachten und analysieren.

Jederzeit (auch während des Spieles) können Figurenstellungen geändert werden, Caissa stellt sich blitzschnell darauf ein und zeigt dies auch in seinen Berechnungen. Neben einer Autoplayfunktion, bei der Caissa gegen sich selbst spielt, ist auch eine Funktion zur Lösung von Schachproblemen vorhanden. Das Programm kann Matt in ein bis zehn Zügen problemlos lösen. Die verschiedenen Schwierigkeitsgrade werden in Form von Zeitvorgaben bestimmt, die Zugzeit läßt sich von 4 Sekunden bis 10 Stunden einstellen.

Es ist ebenfalls möglich, Problemstellungen vorzugeben und somit eventuell Stellungen aus Zeitschriften nachzuspielen. Caissa macht auch Zugvorschläge, gerade für Anfänger ist dies sehr hilfreich.

Bei einer Fehleingabe besteht die Möglichkeit den letzten Zug zurückzunehmen, insgesamt lassen sich bis zu 30 Züge zurücknehmen.

Selbstverständlich kann der aktuelle Spielstand auf Datenträger abgespeichert werden, Parteien können jederzeit weitergespielt oder analysiert werden.

The Dallas Quest



Dallas kommt nun jeden Tag in die heimischen Stuben.

J.R. und seine Mannen treiben diesmal in Form eines Adventures ihr Unwesen. Nach dem Laden erscheinen J.R., Bobby und Pamela als Titelbild, die Dallas-Titelmelodie wird gespielt. Die Grafik dieses Adventures ist wirklich faszinierend, ebenso wie die Spielidee. Als Privatdetektiv, angeheuert von Sue

Ellen, begibt man sich auf die abenteuerliche Suche nach einem Lageplan für ein Ölfeld.

Doch, wie auch anders, J.R. weiß wieder von dem Vorhaben. Mit allen erdenklichen Mitteln versucht der Bösewicht dies zu verhindern.

Man benötigt schon eine Menge Ideenreichtum, um J.R. zu entgehen. So muß z.B. einer Eule eine Sonnenbrille ver-

paßt werden, um diese auch bei Tageslicht sehen zu lassen.

Alles in allem, ein tolles Adventure für lange Tage, gespielt mit Humor und tollen Szenenbildern.

Aber Vorsicht!!

Lassen Sie sich nicht vom Dallas-Fieber packen!

bücher

Heyne: Computer-Lexikon

Das Heyne Computer-Lexikon ist ein umfangreiches Sammelwerk der wichtigsten und meistgebrauchten Begriffe aus der Computerwelt. Da jeder Begriff wirklich verständlich und klar definiert wird, ist dieses Buch nicht nur für Einsteiger und Laien interessant, sondern auch für den Fachmann ein exzellentes Nachschlagewerk.

Positiv, daß bei den Definitionen auf unnütze und verwirrende Formulierungen verzichtet wurde.

Zudem verdeutlichen einige Bilddokumentationen die Definitionen, und ein Vergleich englischer und deutscher Bezeich-



nungen ist ebenfalls hilfreich.

Die Daten: Heyne 1984, 287 Seiten, ISBN S-453-47041-9, DM 12,80

für die Leistung eines Rechners geworden sind.

Nun eignet sich die Programmiersprache Basic aufgrund vieler Charakteristika ihrer Syntax nicht besonders zur Organisation eines echten großrechneranalogen Datenpools, doch hat sich der Autor bewußt dieser Problematik gestellt, da Basic zumindest heute noch die am meisten verbreitete problemorientierte höhere Programmsprache ist. Klaus Menzel teilt seinen Band ein in die Abschnitte Generieren, Segmentieren, Sortieren, Dateiverbund und formatierten Output. Er beschließt ihn mit einem Exkurs über statistische Auswertung und einem Anhang über Dateitypen und Besonderheiten der einzelnen Betriebssysteme. "Dateiverarbeitung mit BASIC" ist für jeden, der etwas aus seinem Rechner herausholen und ein echtes data processing betreiben möchte, empfehlenswert. Die Daten: B.G. Teubner (Stuttgart) 1983, mit 6 Basicprogrammbausteinen auf Diskette, vielen Zeichnungen, 237 Seiten, 62,- DM (Buch allein 28,80) ISBN 3-519-02513-2.

Dateiverarbeitung — mit BASIC

Klaus Menzel widmet sein Buch dem Bereich der sogenannten relationalen Datenbanken, wie Basic sie vom Aufbau her ermöglicht. Wir alle, die wir tagtäglich mit unserem Home- oder Personalcomputer arbeiten, stellen immer wieder fest, daß

ohne die Organisation einer Datei bzw. Datenbank der Verwendbarkeit unseres Systems erhebliche Grenzen gesetzt sind. Das ist auch ganz natürlich, da schließlich die Verwaltung von Daten und die Verarbeitung derselben ein Synonym

(sr)

Tips & Tricks

für den Commodore 64

Assembler-Listing

'ZS-Copy'

Commodore-User finden hier eine Utility, die ihrem Namen alle Ehre macht.

Sie stellt wirklich eine 'nützliche' Maschinen-Routine dar, mit deren Hilfe sich der Zeichensatz nicht nur kopieren, sondern auch editieren läßt.

Sie ist durch das nachfolgende Basic-Listing gut dokumentiert.

```

0      12COPY LDX #0      24 STA NZ+2
1 *= 49152      130Z LDA 53248,X      25 LDA 1
2 LDA #48      14NZ STA 12288,X      26 EOR #4
3 STA 52      15 DEX      27 STA 1
4 STA 56      16 BNE OZ      28 LDA 56334
5 LDA 56334      17 INC OZ+2      29 EOR #1
6 AND #254      18 INC NZ+2      30 STA 56334
7 STA 56334      19 DEY      31 LDA 53272
8 LDA 1      20 BNE COPY      32 AND #240
9 AND #251      21 LDA #208      33 EOR #12
10 STA 1      22 STA OZ+2      34 STA 53272
11 LDY #8      23 LDA #48      35 RTS
  
```

★ ★ ★ ★ ★ ★ ★ ★

```

10 REM ZEICHENSATZ-COPY IN MACHINE-CODE
20 FORX=49152TO49228:READMC:POKEX,MC:NEXT
22 DATA169,48,133,52,133,56,173,14,220,41,254,141,14,220,165
24 DATA1,41,251,133,1,160,8,162,0,189,0,208,157,0,48,202,208
26 DATA247,238,26,192,238,29,192,136,208,236,169,208,141,26
28 DATA192,169,48,141,29,192,165,1,73,4,133,1,173,14,220,73
30 DATA1,141,14,220,173,24,208,41,240,73,12,141,24,208,96
40 :
50 SYS49152
60 :
  
```

70 REM DEMO EINES NEUEN ZEICHENS

80 Y=1:FORX=1TOY:READZN:FORJ=0TO7:READDN:POKE12288+ZN*8+J,DN:NEXT:NEXT

90 DATA1,255,129,129,129,129,129,129,255:PRINT"Q A A A A A A A"

100 :

110 REM ERLAEUTERUNG:

120 REM IN ZEILE 80 WIRD IN DER VARIABLEN Y DIE MENGE DER ZEICHEN ABGELEGT,

130 REM DIE GEAENDERT WERDEN SOLLTEN. IN DIESER ZEILE WIRD DANN DAS NEUE ZEICHEN

140 REM AUS DEN DATA-STATEMENTS EINGELESEN. IN ZEILE 90 GIBT DIE ERSTE ZIFFER

150 REM DEN BILDSCHIRMCODE DES ZEICHENS AN, WELCHES GEAENDERT WERDEN SOLL. DIE

160 REM UEBRIGEN 8 DATAS GEBEN DEN BINAERCODE DES NEUEN ZEICHENS AN. DER NEUE

170 REM ZEICHENSATZ STEHT AB ADRESSE 12288 UND IST VOR DEM UEBERSCHREIBEN DURCH

180 REM BASIC ENTSPRECHEND GESCHUETZT.

Merge für Commodore 64

Einer der Basic-Befehle die dem Commodore-Benutzer bisher fehlen, nämlich Programm zu mergen, wird durch die folgende kurze MC-Routine ermöglicht. Nach dem Laden des ersten Programms muß folgende Routine eingegeben werden:

Routine

1000 REM ** MERGE FOR CBM 64

1010 REM POKE 52,159:POKE 56,159

1020 FOR I=40448 TO 40478:READ A : POKE 1,A:NEXT

1030 DATA 216,56,173,45,0,233,2,141,43,0,173,46

1040 DATA 0,141,44,0,96,234,234,169,1,141,43

1050 DATA 0,169,8,141,44,0,96

Danach muß man RUN eintasten. Nun kann man die Routine mit SYS 40448 aufrufen, das zweite Programm laden und mit SYS 40468 beenden. Zuletzt muß man nur noch die Routine mit Delete löschen, und beide Programme sind zu einem zusammengefaßt. Man sollte darauf achten,

daß die Zeilennummern des zweiten Programms höher liegen als die des ersten.

Commodore

64

Tips:

POKE 808,239 setzt RUN/STOP außer Betrieb.

POKE 808,237 bewirkt eine Wiederaufnahme dieser Funktion.

POKE 775,200 unterdrückt das List-Kommando.

POKE 775,167 bringt es wieder zur Funktion.

CFA '84



Commodore-Fachausstellung Frankfurt/M
vom 6. - 8. September 1984

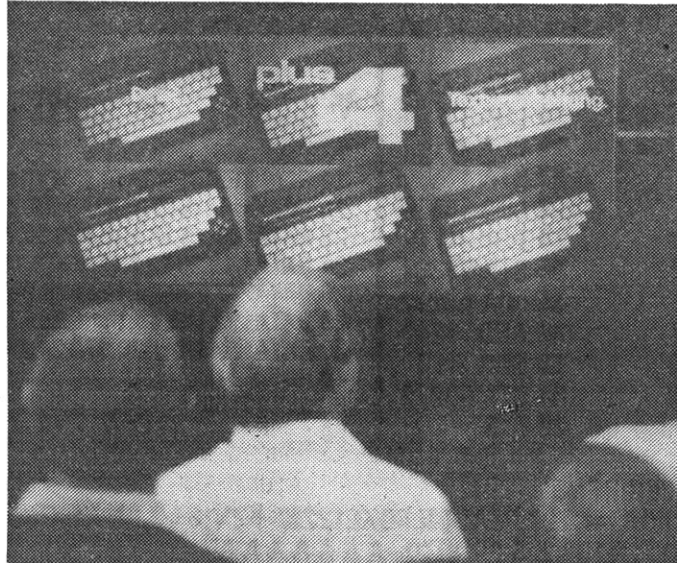
Nachdem die Computer-Freaks sich nahezu alle auf der HiFi-Video in Düsseldorf gesehen hatten, trafen sie sich Anfang September in Frankfurt auf der Commodore-Fachausstellung, die vom Computer-Giganten aus Frankfurt-Niederrad schon zum 4. Mal ausgerichtet worden war.

Jüngste im Bunde der Aussteller war natürlich Ihre "Compute mit", die vom Roeske-Verlag der Öffentlichkeit präsentiert wurde und mit Sekt ihren Geburtstag feiern durfte.

Wenn sich eine Firma im Rahmen einer Sonderveranstaltung dem breiten Publikum präsentiert, so empfiehlt es sich im allgemeinen, dieses Unternehmen auch innerhalb eines Ausstellungsberichtes vorzustellen. Nun, bei dem Initiator der vorliegenden Messe dürfen wir uns das wohl schenken. Von der Frankfurter Tochter des amerikanischen Computer-Konzerns hat sicher auch derjenige schon etwas gehört, der RAM für eine falsch geschriebene Margarine-Marke hält.

Verweilen Sie also nicht lange, sondern begleiten Sie uns auf unserem kleinen Rundgang durch Halle 1 des Frankfurter Messegeländes.

Die CFA, die schon mit einer guten Portion Vorschußlobernen bedacht worden war, öffnete am 6. September zum 4. Mal ihre Pforten. Aussteller aus Deutschland, Österreich,



der Schweiz, Schweden und England suchten nicht nur Kontakt zueinander und den diversen Händlern in der Branche. Sie wollten vor allem auch den eigentlichen Kunden, d.h. professionelle wie private Computer-Anwender, erreichen. Auf über 4000 Quadratmetern wurde denn auch so ziemlich alles geboten, was sich annähernd in die Warenbereiche Hardware, Software, Zubehör und Literatur einordnen läßt.

Der Commodore-Stand war natürlich besonders umlagert. Die Computer-Spezialisten aus der Lyoner Straße in Frankfurt-Niederrad hatten sich denn auch mehr als die Organisation der Ausstellung einfallen lassen. Commodore wartete mit einem kleinen Profi-Pavillon auf, in dem man die unterschiedlichen Rechner für Klein- und Mittelbetriebe bestaunen und auch fachlich in Augenschein nehmen konnte. Darüber hinaus

war ein schon von anderen Messen her bekanntes und beliebtes Rondell aufgebaut, an dem die Homecomputer-Freaks auf ihre Kosten kommen konnten. Und dies umso mehr, als viermal täglich wertvolle Software und C-64 Computer verlost wurden. Die ernsthaften Interessenten hatten zusätzlich Gelegenheit, an diversen "Workshops" teilzunehmen, die tiefer in die Materie der Commodore-Rechner und aller Zusatzgeräte einführten. So sprach Burkhard Rosin aus Braunschweig über "Datenstrukturen und -formate der Floppy 1541", Jim Butterfield (Toronto) über die neue Commodore-Generation (plus/4, C-16, 116), Horst Labusch, der Autor des 'Grandmaster'-Schachprogrammes, über Commodore-Assembler, und Dr. Clemens Deisenhammer aus Salzburg informierte die Anwender von Commodore-Computern über die Mög-

lichkeiten von BTX, DFÜ und Teletex.

Angesichts dieser massiven Öffentlichkeitsarbeit hätten Beobachter versucht sein können, für die übrigen Aussteller von schlechten Startchancen zu sprechen. Dem war aber ganz und gar nicht so, sondern beflügelte sie sogar, zumal mancher im Wind des Organisators segeln konnte (man denke an die vielen Anbieter speziellen Commodore-Zubehörs).

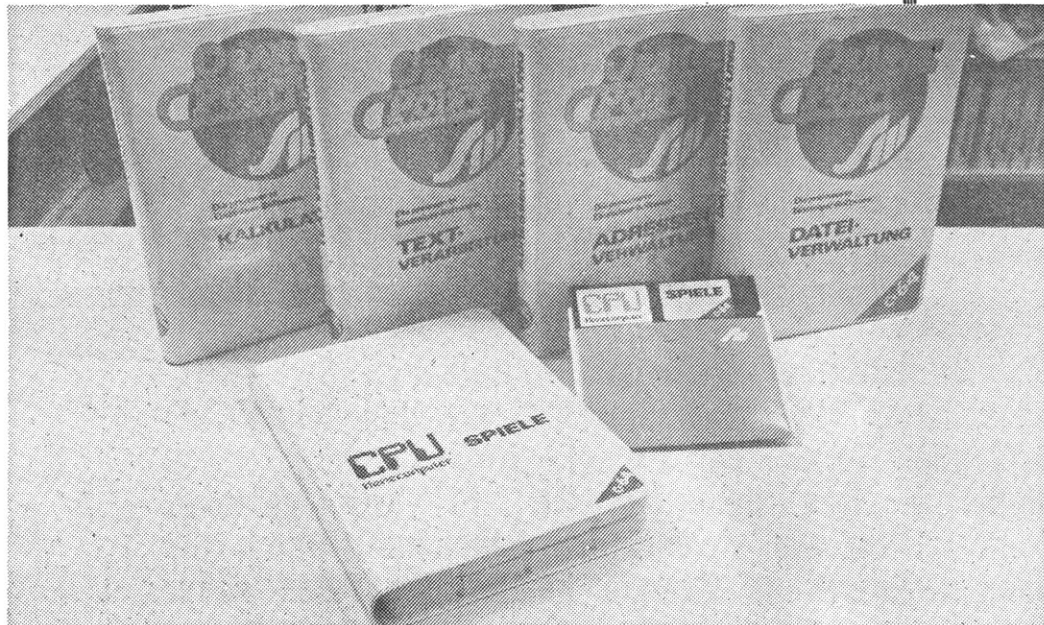
Der Software-Händler "Micro-Händler" aus Mönchengladbach präsentierte seine neuesten Software-Hits für die Commodore-Renner VC-20 und C-64 und hatte viel jugendliches Publikum um seinen Stand versammelt.

Der Schulbuchverlag Westermann, den man früher ausschließlich mit Atlanten und verschiedensten Unterrichtswerken in Verbindung brachte, betätigt sich inzwischen mit großem Engagement in Sachen Lern- und Ausbildungsprogrammen für Homecomputer. Ähnliches hatten wir in den letzten Ausgaben von HOMECOMPUTER bereits vom Spiele-Fabrikanten Otto Maier aus Ravensburg gemeldet. Um bei Verlagen zu bleiben, so hat sich Langenscheidt auf der Basis der Space-Invaders-Idee etwas Drolliges einfallen lassen. Keine Außerirdischen greifen an, sondern unsympathische Gesellen, die schon ganze Schüler-Generationen zur Verzweiflung brachten, nämlich Vokabeln. Man kann nur

von Glück sagen, daß sie nicht einfach hingenommen werden müssen: Langenscheidt stellt eine "Vokabelfalle" vor, die einer gut bestückten Weltraum-Basis gleichkommt.

Vom Lernen zur Musik: Das Thema "Computer als Synthesizer" fesselt immer mehr Menschen. Schon jetzt kann man sagen, daß der Heimcomputer eines der beliebtesten Musikinstrumente ist. Kein Wunder, daß die Stände zweier Firmen umlagert waren, die sich auf derartige Angebote spezialisiert haben. Wersi-electronic gab dem interessierten Publikum Gelegenheit, das fabelhafte WERSI-BOARD unter die Lupe zu nehmen, mit dem man schon für weniger als fünfhundert Mark aus seinem 64er eine Top-Orgel machen kann. Orgel-Schmidt wartete mit einer ganzen Palette leistungsstarker Programme auf, die aus Ihrem Commodore einen vollwertigen Konzert-Synthesizer machen können.

Adcomp Datensysteme aus München präsentierten neben vielen anderen interessanten Features für Profis und Semi-Profis auch ihr Produkt "64+", ein Steckmodul, das nicht nur den Commodore 64-Befehlssatz auf Basic 4.0 erweitert, sondern gleichzeitig einen vollwertigen IEEE-488-Bus mit Platinenstecker bietet.



Aztec-Software wartete mit einem Sprachmodul auf, das nicht nur eine ernstzunehmende Konkurrenz zu Currahs Microspeech darstellt, sondern – und das ist für engagierte Programmierer sicher interessant – in unbegrenzter Zahl Vokabeln erzeugen kann. Da Aztec sein "Speakeasy" für die bekanntesten Heimcomputer anbietet, erübrigt sich fast der Hinweis, daß man auch eine C-64er Version zur Verfügung stellt.

SmallBusiness erhebt in Sachen Fremdsprachen einen hohen Anspruch. Die Software-Spezialisten von SM verweisen darauf, daß ihr neues Lernkonzept "Softlearning" nicht irgend ein Computer-Sprachkurs, sondern ein System sei, das

auf jüngsten tiefenpsychologischen Erkenntnissen basiere. Nicht der Einsatz des Computers sei das Entscheidende, vielmehr der Umstand, daß "Softlearning" die Technik des Vokabel-Lernens als solche revolutioniere. 2000 wirklich behaltene Vokabeln in nur 14 Tagen stellen somit kein unerreichbares Ziel mehr dar. Bislang sind die SM-Sprachkurse für Englisch, Italienisch, Spanisch und Französisch zu haben – weitere Sprachen sollen aber in Kürze folgen.

Die Zubehörfirma COMTEC bietet nicht nur "garantiert stückgeprüfte" Disketten (8 bzw. 5,25"/m. Verstärkungsring), sondern darüber hinaus ein umfangreiches Sortiment an Aufbewahrungssystemen für wertvolle Datenträger: Safes und Tresore, die sowohl feuergeschützt als auch sturz- und löschsicher sind. Groß war das Angebot der verschiedenen Software-Häuser. KINGSOFT aus Roetgen bei Aachen war mit "Zaga" vertreten, einem Videospiel der allerjüngsten Generation: 60K reine Maschinensprache, fast schon zu realistische 3D-Grafik und ein Sound, der alle Register moderner Heimcomputer zieht.

Aus Schweden kam 'handic-soft', bekannt durch das Finanzplanungsprogramm

"Calcresult". Die "advanced" Version hat statt zwei nun drei Dimensionen (32 Seiten, 254 Zeilen, 64 Spalten, soll jedoch mindestens ebenso bedienerfreundlich sein wie die konventionelle Ausgabe. HANDIC bot aber auch eine Menge anderer Software, vor allem für den Commodore 64, an: FORTH-64 (eine Programmiersprache der "vierten Generation"), STAT-64 (ein Basic-Toolkit für statistischen Einsatz), REL-64 (programmgesteuerte Apparatekontrolle, beispielsweise für die Steuerung der elektrischen Eisenbahn-Anlage), MON-64 (die "perfekte" Terminplanung für mittleres Management), GRAF-64 (grafische Analyse von Gleichungen) und vieles mehr.



Wir hoffen, daß unser Rundgang, die gegebenen Informationen wie auch die zahlreichen Bilder bei Ihnen ein wenig von der Stimmung aufkommen zu lassen, die an allen Ausstellungstagen auf der 4. CFA in Frankfurt herrschte.

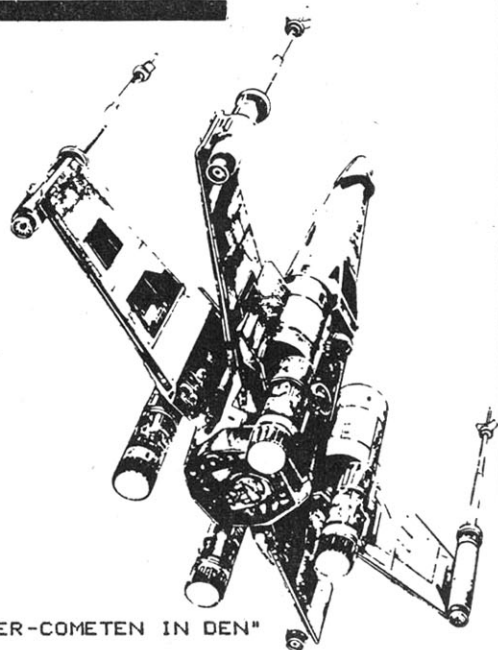
(sk)



Cometen-Falle

für den VC-20 ohne Erweiterung

Sie sind Astronaut der Raumfähre Galactica. Beschützen Sie Ihr Raumschiff vor den feindlichen Zyklonen. Vernichten Sie die Killer-Cometen, diese blockieren Ihre Steuerung. Mit "F1" wird gefeuert, doch leisten Sie sich keinen Fehlschuß! Durch die dann freiwerdende Energie kommt der Feind näher heran.



```

10 POKE36879,42
20 REM"*****"
30 REM"***** CO BY ANDREAS HAZY
70 REM"*****"
80 REM"*****"
90 PRINT"COMETEN - FALLE":PRINT
100 PRINT"SIE SIND ASTRONAUT DERRAUMFAEHRE 'GALECTICA'"
110 PRINT"DIE FEINDLICHEN ZYKLO-NEN STELLTEN IHNEN KILLER-COMETEN IN DEN"
120 PRINT"WEG, DIE IHRE STEUER-UNG TOTAL BLOCKIEREN."
130 PRINT"SIE MUESSEN NUN VER- SUCHEN DIE COMETEN AB-ZUSCHIESSEN (MITTELS"
140 PRINT"DER TASTE 'F 1'). SIE DUERFEN SICH KEINEN FEHLSCHUSS LEISTEN,"
150 PRINT"DENN DURCH DIE DADURCHFREIWERDENDE ENERGIE, KANN DER FEIND NAEHER"
160 PRINT"AN SIE HERANKOMMEN."
170 FORTT=1TO10000:GETTT$:IFTT$<>" THEN TT=10000
180 NEXT TT
190 DIM A(15),B(15)
200 POKE36879,14
210 PRINT"Q"
220 FOR I=1TO15
230 A(I)=INT(RND(1)*418)+7680
240 B(I)=90
250 IF PEEK(A(I))<>32 THEN 230
260 POKE A(I),B(I):POKE A(I)+30720,3
270 NEXT I
280 FOR I=38863TO38926
290 A=0
300 POKE I,4:NEXT I
310 Q=8142
320 FOR I=QT08163
330 FOR J=1TO15:NEXT J
340 GOSUB 600
350 A=0
360 POKE I-1,32
370 U=1
380 POKE I+21,32
390 IF PEEK(I)=90 OR PEEK(I)=128+32 THEN 1040
400 POKE I,113
410 POKE I+22,128+32
420 NEXT I
430 REM
440 Q=8163:I=Q
450 REM
460 I=I-1:IF I<=8142 THEN 530
470 Q=I
480 FOR J=1TO15:NEXT J
490 GOSUB 600
500 REM
510 A=1
520 POKE I+1,32
530 U=1
540 POKE I+23,32
550 IF PEEK(I)=90 OR PEEK(I)=128+32 THEN 1040
560 POKE I,113
570 POKE I+22,128+32
580 GOTO 460
590 POKE 8165,32:POKE 8165-22,32:GOTO 310
600 GETG$:IFG$<>CHR$(133) AND G<=0 THEN RETURN
610 IFG<>0 THEN 680
620 IFA=1 THEN 770
630 G=I-22:
640 IF PEEK(G)=90 THEN 820
650 POKE G,46
660 POKE G+30720,7
670 RETURN
680 POKE G,32
690 G=G-22
700 IF PEEK(G)=90 THEN 820
710 POKE G,46
720 POKE G+30720,7
730 IFG<7680 THEN 750
740 RETURN
750 GOTO 970
760 RETURN
770 G=I-23
780 IF PEEK(G)=90 THEN 820
790 POKE G,46
800 POKE G+30720,7
810 RETURN
820 POKE G,86
830 POKE G+30720,2
840 POKE 36877,220

```

```

850 FORL=15T00STEP-1
860 POKE36878,L
870 FORM=1T0300
880 NEXTM,L
890 POKE36877,0:POKE36878,0
900 YY=YY+1:IFY=15THEN1170
910 FORO=1T015
920 IFA(O)=GTHENB(O)=32
930 NEXTO
940 POKEG,32
950 G=0
960 RETURN
970 G=0
980 FORY=1T015
990 POKEA(Y),32
1000 A(Y)=A(Y)+22
1010 POKEA(Y),B(Y):POKEA(Y)+30720,3
1020 NEXTY
1030 RETURN
1040 POKEI+30720,5
1050 POKEI,42
1060 POKE36877,220
1070 FORL=15T00STEP-1
1080 POKE36878,L
1090 FORM=1T075
1100 POKEI,86
1110 POKEI,42
1120 NEXTM,L
1130 POKE36877,0:POKE36878,0:POKEI,32:POKEI+22,32
1140 POKE36879,254
1150 PRINT"WOLLENSIE SIND LEIDER AN EINEM KILLER-COMETEN ZERSCHELT."
1160 GOTO1190
1170 POKE36879,172
1180 PRINT"WOLLENBRAVO SIE HABEN ALLE KILLER-COMETEN ERWISCHT!!!"
1190 PRINT:PRINT:PRINT
1200 PRINT"WOLLEN SIE ES NOCH EINMAL MIT DEN ZYKLONEN AUFNEHMEN? (J/
N)"
1210 GETUU$:IFUU$=""THEN1210
1220 IFUU$="J"THENCLEAR:GOTO190
1230 IFUU$="N"THEN1250
1240 GOTO1210
1250 POKE36879,59
1260 PRINT"WOLLEN SIE AUFNEHMEN SIE FEIGLING!"
1270 END
READY.

```

Zahlen-Senso

für den VC-20 (Grundversion)

Bei diesem Spiel geht es um Konzentration. Es werden nacheinander 5 Zahlen dargestellt, die mit Hilfe eines Zählers, der mit der G-Taste (Go) betätigt wird, abgeschossen werden können (mit der F-Taste (Feuer)).

Aber es muß schnell gehen, damit nie 5 Zahlen auf einmal vorhanden sind, denn dann sind Sie abgeschossen.

Für jede abgeschossene Zahl erhält man einen Punkt. Als M. Ziffer gibt es ein Raumschiff. Für das Abschießen eines Raumschiffes bekommt man 10 Punkte.

```

0 REM*****
1 REM ANDREAS HAZY
5 REM*****
10 DIM P1(15)
20 DIM WJ$(6)
30 POKE36879,110:PRINT" ";
40 PRINT"-----ZAHLEN-SENSO-----";
50 PRINT"BEI DIESEM SPIEL GEHT ES UM KONZENTRATION. AUF DER TASTATUR MUES-";
60 PRINT"SEN DIE TASTEN 23 (=GO)";
70 PRINT"UND 24 (=FEUER) BETAET-TIGEN. BEI GO WIRD DER";
80 PRINT"DER ZAEHLER JEWEILS UMEINE STELLE VORGE- RUECKT, WENN SIE 24";
90 PRINT"DRUECKEN, WERDEN ALLE ZIFFERN, AUF DER DER ZAEHLER STEHT ABGE-";
100 PRINT"SCHOSSEN. SIE MUESSEN AUFFASSEN, DASS NICHT MEHR ALS 5 ZIFFERN";
110 PRINT"VORHANDEN SIND, SONST WERDEN SIE ABGESCHOS- SEN. ALS 11. ZIFFER";
120 PRINT"ERSCHEINT EIN RAUM- SCHIFF (= ). FUER DAS GIBT ES 10 PUNKTE.";
130 FOR I=1 TO 8000:GETQQ$:IFQQ$<>" " THEN I=8000
140 NEXT I
150 POKE36879,25:PRINT" ";
160 FOR II=1 TO 15:READ P1(II):NEXT
170 DATA 120,115,110,105,100,95,90,85,80,75,70,65,60,55,50
180 IG=IG+1
190 POKE36879,25:PRINT" ";
200 KB=0
210 IF IG=16 THEN 730
220 PRINT"-----ZAHLEN-SENSO-----";
230 Q$=""
240 W$=""
250 R1=TI
260 PRINT" "
270 PRINT" | "
280 PRINT" RUNDE" IG
290 GETYY$
300 GOTO 400
310 IF YY$="G" THEN 350
320 IF YY$="F" THEN 610
330 PRINT" ";RIGHT$(Q$,1);TAB(15);YY$="":GOTO 480
340 IF YY$=" " THEN 290
350 Q=Q+1
360 IF Q<10 THEN 390
370 IF Q=11 THEN Q=0:GOTO 390
380 Q$=" ":GOTO 330
390 Q$=STR$(Q):GOTO 330
400 IF TI<(R1+P1(IG)) THEN 310
410 R1=TI
420 IF PP/10=INT(PP/10) AND PP<>0 THEN Y$=" ":GOTO 450
430 Y=INT(RND(TI)*10)
440 Y$=STR$(Y)
450 W$=W$+RIGHT$(Y$,1)
460 IF LEN(W$)>=6 THEN 560
470 GOTO 310
480 LL=LEN(W$)
490 IF LL>=5 THEN 540
500 LL=5-LL
510 FOR I=1 TO LL
520 PRINT". ";
530 NEXT I
540 PRINT W$
550 GOTO 340
560 PRINT" "
570 PRINT" SIE WURDEN LEIDER ABGESCHOSSEN"
580 PP=PP-2
590 FOR II=1 TO 1500:NEXT II
600 GOTO 180
610 LL=LEN(W$)
620 FOR II=1 TO LL
630 WJ$(II)=MID$(W$,II,1)
640 IF WJ$(II)=RIGHT$(Q$,1) THEN GOSUB 850:WJ$(II)="" :PP=PP+1:KB=KB+1:P2=1
650 NEXT II
660 W$=""
670 FOR II=1 TO LL
680 W$=W$+WJ$(II)
690 NEXT II
700 IF KB>=15 THEN 180
710 IF P2=0 THEN GOSUB 880
720 GOTO 330
730 PRINT" ";
740 POKE36879,8
750 PRINT" SIE ERREICHTEN: " :PRINT PP "PUNKTE"
760 PRINT:PRINT:PRINT"WOLLEN SIE NOCH EINE RUNDE SPIELEN? (J/N)"
770 IG=0
780 GETX$:IF X$=" " THEN 780
790 IF X$="J" THEN 180
800 IF X$<>"N" THEN 780
810 END
820 BB=BB+1
830 IF BB=3 THEN BB=0:PP=PP-1:RETURN
840 RETURN
850 IF WJ$(II)="" THEN PP=PP+10
860 RETURN
870 END

```


39

programme

[illegible]

```

602 IF SC > 2 THEN Z = SC
605 GOTO 5
700 IF K = 1 THEN S00
702 IF K = 2 THEN SC = SC + 1 : K = 32 : RETURN
704 IF K = 3 THEN F = F + 75 : K = 32 : POKEN = 3, 200 : FORE = 1 TO 500 : NEXTE : POKEN = 3, 0 : RETURN
705 IF K = 6 THEN S00
706 RETURN
750 PRINT "***** GAME IN PAUSE ***";
760 WAIT 203, 15 : GOTO 70
800 REM# =====#
802 REM# DELTA-RACE #
804 REM# WRITTEN BY: #
806 REM# TH. SCHLAEGER#
808 REM# =====#
READY.

```



Ziel des Spieles ist es, die auf dem Fließband stehende Kiste auf den Wiegebalken zu heben, indem man den Lasthaken mit den Tasten 'W,A,D,X' so steuert, daß man genau über den Ring der Kiste fährt, um durch Drücken der S-Taste, die Kiste einzuhaken. Dann fährt

man so über den Wiegebalken, daß die Kiste ohne überzustehen darüber hängt, und hakt sie dann durch Drücken der Leertaste wieder aus. Beim Transport der Kiste ist darauf zu achten, daß Sie nirgends anstößt. Für jede richtig abgesetzte Kiste gibt es einen, im Rahmen

des Schwierigkeitsgrades vorgesehene Punktzahl. Hierbei ist es möglich, z.B. bei Schwierigkeitsgrad 13 eine niedrigere Punktzahl zu erreichen, als bei Schwierigkeitsgrad 5; wie man aus nachstehender Tabelle ersehen kann:

SCHWIERIGKEITSGRAD	1	2	3	4	5	6	7	8	9	10	11	12	13
NIEDR. PUNKTZAHL	19	38	57	76	96	115	134	153	173	192	211	230	250
HOECHSTE PUNKTZAHL	56	113	170	227	284	341	398	455	512	569	626	683	740

Die Spielzeit ist auf 2 Minuten begrenzt.

Programmaufbau:

1-246: Erläuterung der Steuerung, setzen des Schwierigkeitsgrades
250-460: Aufbau der Grafik, Spielzeit wird auf Null gesetzt
500-590: Abfrage von Tastatur und Zeit,

Ausgabe der Spielzeit

600-930: Steuerung des Hakens
1000-1090: Bewegung der Kiste auf dem Fließband
1100: Signalton beim Anstoßen des Hakens
1200-1310: Zerschellen der Kiste mit akustischer Untermalung
1500-1510: Einhaken der Kiste mit

Signalton

1520-1533: Kiste löschen
1535-1541: Abfrage, ob die Kiste angestoßen ist
1542-1570: Bewegung der Kiste am Haken
1600: Abfrage, ob sich die Kiste in der richtigen Position befindet, um mit Signalton auszuhaken


```

1006 IFF=15THEN1050
1010 POKEE,102:POKEE+22,102:POKEE+44,102:POKEE+Q,4:POKEE+22+Q,4:POKEE+44+Q,4
1015 IFF=1THEN1080
1020 POKEE-1,102:POKEE+21,102:POKEE+43,102:POKEE-1+Q,4:POKEE+21+Q,4:POKEE+43+Q,4
1022 POKEE-23,87:POKEE-23+Q,4
1025 IFF=2THEN1080
1030 POKEE-2,102:POKEE+20,102:POKEE+42,102:POKEE-2+Q,4:POKEE+20+Q,4:POKEE+42+Q,4
1033 POKEE-24,32
1035 IFF=3THEN1080
1040 POKEE-3,32:POKEE+19,32:POKEE+41,32:GOTO1080
1050 E=8101:F=0
1080 IFPEEK(B)<>10THEN1100
1090 RETURN
1100 POKE36875,200:FORK=1TO333:NEXTK:POKE36875,0:RETURN
1200 G=G-22:J=0
1210 J=J+1:IFJ=1THENJ=127
1230 IFJ=2THENJ=32:GOTO1260
1240 POKEG-22+Q,4:POKEG+Q,4:POKEG+22+Q,4:POKEG-23+Q,4:POKEG-1+Q,4:POKEG+21+Q,4
1250 POKEG-24+Q,4:POKEG-2+Q,4:POKEG+20+Q,4
1260 POKEG-22,J:POKEG,J:POKEG+22,J:POKEG-23,J:POKEG-1,J:POKEG+21,J
1270 POKEG-24,J:POKEG-2,J:POKEG+20,J
1275 POKE36876,0
1280 IFJ=127THENFORI=1TO250:POKE36877,128:NEXTI:POKE36877,0
1300 IFJ=127THENJ=1:GOTO1210
1310 G=E:S=0:H=0:GOTO550
1500 IFPEEK(B+22)<>87THENRETURN
1505 IFE<81040RE>8112THENRETURN
1510 S=1:G=E:POKE36876,222
1520 POKEG-22,32:POKEG,32:POKEG+22,32:POKEG-23,32:POKEG-1,32:POKEG+21,32
1530 POKEG-24,32:POKEG-2,32:POKEG+20,32
1533 IFG=ETHENPOKEG+42,32:POKEG+43,32:POKEG+44,32:POKE36876,0
1535 G=G+H
1536 IFPEEK(G)<>32THEN1200
1537 IFPEEK(G+20)<>32THEN1200
1538 IFPEEK(G+22)<>32THEN1200
1539 IFPEEK(G-22)<>32THEN1200
1540 IFPEEK(G-23)<>32THEN1200
1541 IFPEEK(G-24)<>32THEN1200
1542 POKEG-22,102:POKEG-22+Q,4:POKEG,102:POKEG+Q,4:POKEG+22,102:POKEG+22+Q,4
1545 POKEG-23,102:POKEG-23+Q,4:POKEG-1,102:POKEG-1+Q,4:POKEG+21,102:POKEG+21+Q,4
1550 POKEG-24,102:POKEG-24+Q,4:POKEG-2,102:POKEG-2+Q,4:POKEG+20,102:POKEG+20+Q,4
1570 POKEB,10:E=8101:F=0:RETURN
1600 IFPEEK(G+42)<>170THEN1800
1640 POKE36876,222:J=1:W=INT(RND(1)*99)+50:KG=KG+INT(W*Z)
1650 PRINT"X100"TAB(14)"":PRINT"X1000000GEWICHT:"INT(W*Z)"KG":PRINT"X100"
1690 GOTO1210
1700 PRINT"X10":POKE36879,8:POKE36878,15:PRINTSPC(133)"X10 PUNKTZAHL:"
1705 FORI=128TO254:POKE36874,I:V=V+41:PRINT"X10 SPC(167)V"X10 *":IFV=KGTHEN1710
1707 FORT=1TO100:NEXTT:NEXTI
1710 PRINT"X10 SPC(167)KG"X10":POKE36874,0
1720 END
1800 X=0:H=22:POKE36877,254:GOSUB1520:GOTO1800
2000 POKE36874,200:POKE0,32:POKE0+1,32:POKE0+2,32
2020 N=INT(RND(1)*12):O=8161:O=O-(22*N):IFPEEK(O)=102THENPOKE36874,0:X=1
2030 POKE0,170:POKE0+Q,2:POKE0+1,170:POKE0+1+Q,2:POKE0+2,170:POKE0+2+Q,2:IFX=1TH
EN1800
2040 POKE36874,0:RETURN

```

ACHTUNG!

Bei uns finden Sie
fast alles für Ihren
VC-20/64

Hardware:

Speichererweiterungen 8K-64K RAM,
Moduladapter 2/3/5-fach, Recorder-
interface, Schnell-Save Module, Tool-
kitmodule, 40/80 Zeichenkarten,
Epromprogrammierer, Epromkarten,
PIO IN/OUT Module, Interface, Joy-
sticks, Paddle, und...und...und...

Zubehör:

Kassetten, Disketten, Staubschutz-
hauben, Bücher, Bausätze, Resetaster,
Alle Stecker, Bauteile, und...und...

Software:

Außergewöhnliche Spiele und Pro-
gramme, z.B.: Flugsimulatoren, Date-
programme, Krankheitsdiagnose,
Biorhythmus, Lottoberechnung und
Spiele...Spiele...Spiele...

Prüfen Sie unser Angebot der Tiefstpreise.

„Schnell das neue Spitzeninfo
anfordern, für 2 DM in Briefmarken.
Garantiert 24 Std. Infoversand.“

Händleranfragen erwünscht.

mükra
DATEN-TECHNIK

Rotdornweg 15
1000 Berlin 45
☎ 030-817 38 57
341 45 73

Der ideale Schönschreibdrucker für Ihren C 64* von **iti** OLYMPIA electronic compact 2

- eingebautes Interface für seriellen Port
- Adresse 4 oder 5 einstellbar
- Schreibmaschine mit Korrekturspeicher
- ON-OFF-Line Taste trennt Betriebsarten
- Zeichenabstand (10, 12 und 15 Z/Zoll) und Zeilenabstand (1, 1 1/2 und 2 zeilig) vom Rechner einstellbar
- ohne Tastatur als compact 2 RO
- 100 Zeichentypenrad
- 14 Zeichen/Sekunde
- Leerstellenunterdrückung
- sofort lieferbar
- enorm preiswert
- viele Typenräder, Gewebe-Karbon-, Multikarbonbänder
- bundesweiter Werkkundendienst

* auch für andere Rechner mit Centronics, V.24 und IEC-Bus.

Alle OLYMPIA-Drucker (ESW 102, 103 u. 3000) für den C 64 lieferbar

iti -Datentechnik

Telemannstraße 18
7250 Leonberg (Höfingen)
☎ 07152/6305



The Pit

für den VC-20 + 16K

Bei diesem Spiel müssen Sie nach unterirdischen Schätzen graben. Dabei ist auf verschiedene Gefahren zu achten:

- a) auf Steine, die auf einen herabfallen, wenn man sie unterquert
 - b) auf den Höhlenbären
 - c) in der Schatzhöhle auf herabfallende Felsen
 - d) und zum Schluß: auf die, hinter einem zusammenbrechende Brücke
- Für jeden kleinen Schatz, den man er-

reicht, bekommt man 100 Punkte. Für jeden großen Schatz, den man aus der Schatzhöhle bringt, 200 Punkte.

Für jeden großen Schatz, den man an die Erdoberfläche bringt, erhält man 500 Punkte.

99 Sekunden stehen Ihnen zur Verfügung, um die Schätze zu bergen. Es beginnt eine neue Spielrunde, wenn mindestens ein großer Schatz an die Erdoberfläche befördert wurde

Die Tastaturbelegung wird im Spiel erklärt, es ist aber auch möglich, mit dem Joystick zu spielen.

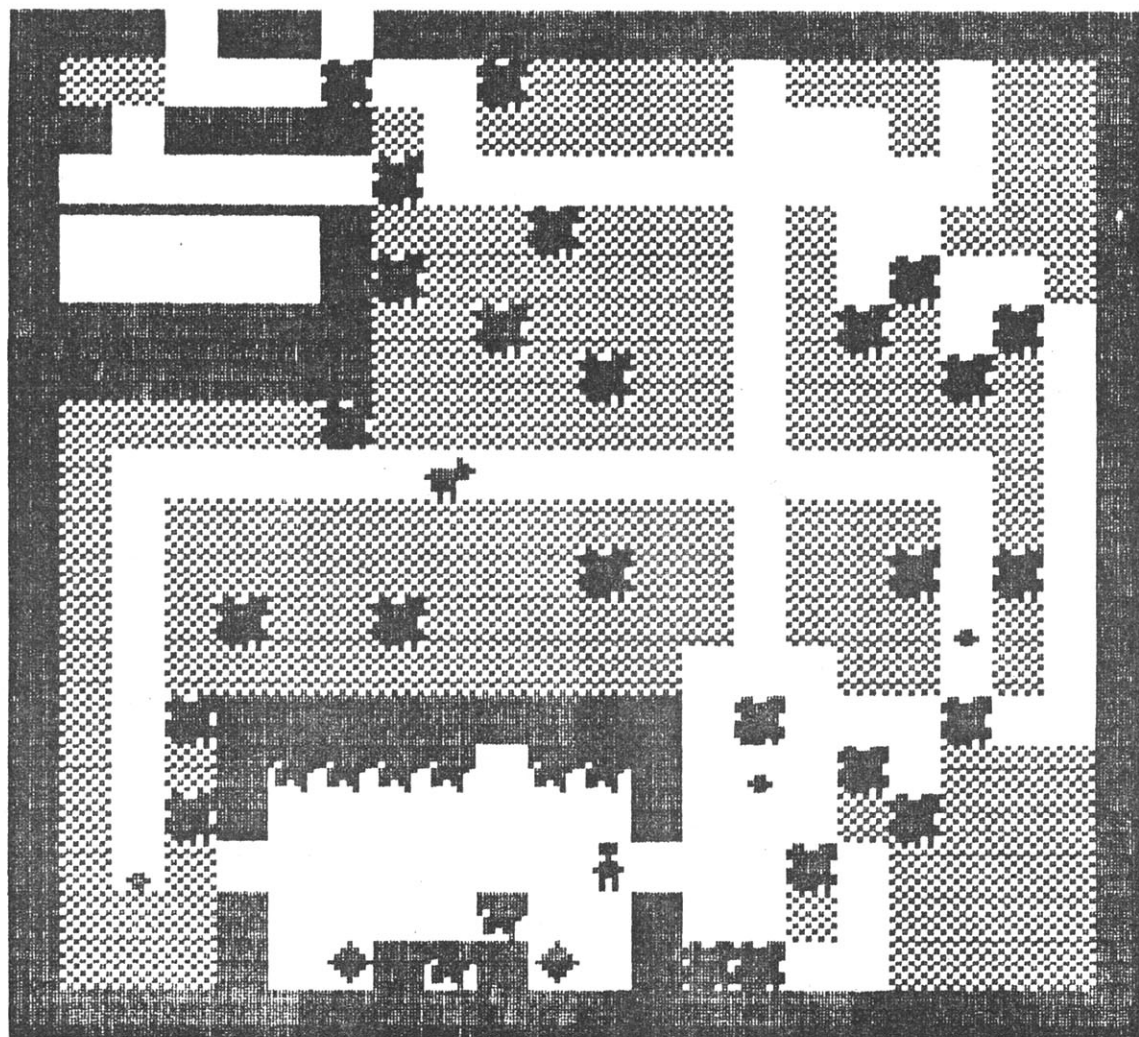
Vor dem Laden des Programmes müssen die Befehle:

Poke 44,28:

Poke 7168,0:

NEW

einggegeben werden.




```

1 REM *** THE PIT ***
2 REM FREI NACH EINEM          SPIELAUTOMATEN
3 REM VON
4 REM JOHANNES
5 REM STRULLER
6 REM VOR LADEN
7
8 REM POKE44,28:POKE7168,0:NEW EINGEBEN
9 GOSUB11000:REM TITELBILD & NEUER ZEICHENSATZ
10 GOSUB14000:REM JOYSTICKROUTINE
11 GOSUB13000:REM SPIELERKLAERUNG
12 POKE36879,8:PRINT"J":POKE36878,15:POKE649,0:REM TASTATURPUFFER AUS
13 GOSUB1400:REM SIRENE
14 ET=99:GOSUB1000:REM PUNKTEANZEIGE
15 R=0:H=1:Q1=0:I1=0:Q0=0:M1=0:K1=1:K2=1:PJ=75:HJ=77:SP=0
16 GOSUB10000:REM SPIELFELDAUFBAU
17 HB=4352:PI=4121:S1=TI
18 ET=99-INT((TI-S1)/60):IFET<10THENPOKE36876,240
19 REM ABFRAGE NACH HOEHLNBAER
20 IFPI=HBORPI-1=HBORPI+1=HBORPI+22=HBORPI-22=HBTHEN1300:REM VOM BAER GEFASST
21 POKEPI,PJ
22 GOSUB1000
23 IFPI=4121ANDM1=1THENP=P+T1:T1=0:GG=1:GOTO30:REM NEUE SPIELRUNDE
24 IFET<0THEN500:REM ZEIT UM
25 R=R+W
26 IFQ0=1THENGOSUB900
27 IFQ=1THENGOSUB1500
28 REM JOYSTICKABFRAGE
29 JS=USR(0)
30 REM TASTATURABFRAGE
31 TS=PEEK(197)
32 IFJS=40RTS=33THENPI=PI-1:GOSUB300:GOSUB2010:POKEPI+1,32:IFQ=1THENQ=0:PI=PI+1
33 IFJS=80RTS=26THENPI=PI+1:GOSUB300:GOSUB2050:POKEPI-1,32:IFQ=1THENQ=0:PI=PI-1
34 IFJS=10RTS=37THENPI=PI-22:GOSUB300:GOSUB2030:POKEPI+22,32:IFQ=1THENQ=0:PI=PI+22
35 IFJS=20RTS=30THENPI=PI+22:GOSUB300:GOSUB2070:POKEPI-22,32:IFQ=1THENQ=0:PI=PI-22
36 IFJS=0ANDTS=64THENPJ=75
37 IFR=2THENW=0:R=0:GOTO400
38 POKEHB,HJ:POKE33792,3:GOSUB600
39 GOTO90
40 IFPEEK(PI)=64ORPEEK(PI)=65ORPEEK(PI)=119ORPEEK(PI)=76THENQ=1:RETURN:REM WAND
,STEIN..
41 IFPEEK(PI-22)=65THENSE=PI:W=1:REM STEIN UNTERLAUFEN
42 IFPEEK(PI)=90THENT1=T1+500:T2=T2+200:M1=1:GOSUB750:RETURN:REM GR. SCHATZ ERR
EICHT
43 IFPEEK(PI)=66THENP=P+100:GOSUB700:RETURN:REM KLEINEN SCHATZ ERREICHT
44 IFPI=4519ORPI=4525THENG=1:REM SCHATZHOEHLE ERREICHT
45 IFPI=4211ORPI=4208THENQ0=1:REM BRUECKE ERREICHT
46 IFPEEK(PI)=32ORPEEK(PI)=66ORPEEK(PI)=119THENRETURN
47 POKEPI,230:POKE33792+PI,4:FORO=1TO15:POKE36877,150
48 NEXTO:POKE36877,0:POKE33792+PI,1:RETURN
49 IFPI=4518ORPI=4526THENG=0
50 REM STEINFALL
51 POKESE-22,32:POKE33770+SE,1:SE=SE+22
52 IFSE-22=PITHENSP=1:REM VOM STEIN ERSCHLAGEN
53 IFSP<1THEN435
54 IFPEEK(SE+22)<32THEN450
55 GOTO440
56 IFPEEK(SE)<32THEN450
57 POKESE,65:POKE33792+SE,5:IFSP=1THENPOKESE+22,PJ:POKE33814+SE,1
58 GOTO410
59 IFSP<1THEN455:POKESE-44,65:POKE33758+SE,5:GOTO460
60 POKESE-22,65:POKE33770+SE,5
61 POKE36877,220:FORL=6TO0STEP-1:POKE36878,L
62 FORM=1TO12:NEXTM:NEXTL
63 POKE36877,0:POKE36878,15:IFSP=1THEN500
64 GOTO90
65 REM MANNVERLUST
66 FORL=15TO0STEP-1,5:FORO=1TO10:POKE36878,L:POKE36874,128+10*O:NEXTO:NEXTL:POK
E36874,0
67 POKE36878,15:PI=4121:HB=4158:G=0:GG=1:T1=0:T2=0:Q0=0:I1=0
68 GOTO1100
69 REM HOEHLNBAERSTEUERUNG
70 ONHGOTO620,640,660,680
71 HB=HB+22:Z=RND(1):IFPEEK(HB)<32ORZ>.7THENH=INT(RND(1)*4)+1:HB=HB-22:RETURN

```

```

630 POKEHB-22,32:GOSUB2090:GOTO695
640 HB=HB-1:Z=RND(1):IFPEEK(HB)<>32ORZ>.7THENH=INT(RND(1)*4)+1:HB=HB+1:RETURN
650 POKEHB+1,32:GOSUB2110:GOTO695
660 HB=HB-22:Z=RND(1):IFPEEK(HB)<>32ORZ>.7THENH=INT(RND(1)*4)+1:HB=HB+22:RETURN
670 POKEHB+22,32:GOSUB2130:GOTO695
680 HB=HB+1:Z=RND(1):IFPEEK(HB)<>32ORZ>.7THENH=INT(RND(1)*4)+1:HB=HB-1:RETURN
690 POKEHB-1,32:GOSUB2150
695 RETURN
700 REM KLEINEN SCHATZ ERREICHT
710 FORL=15TO0STEP-1.5:FORO=1TO7
720 POKE36878,L:POKE36876,220
730 NEXTO,L:POKE36876,0:POKE36878,15
740 RETURN
750 REM GROSSEN SCHATZ ERREICHT
760 FORL=15TO0STEP-3:FORO=0TO8
770 POKE36878,L:POKE36876,245-4*O
780 NEXTO,L:POKE36876,0:POKE36878,15
790 RETURN
900 REM ZUSAMMENSTUERZENDE BRUECKE
910 Q1=Q1+1:IFQ1=3THEN940
920 IFQ1=4THEN970
930 RETURN
940 Q1=2:I1=I1+1:IFI1>5THENQO=0:Q1=3:RETURN
950 POKE4234-I1,32:IFPI+22=4234-I1THEN970
960 RETURN
970 FORI=1TO3:POKEPI,74:FORO=1TO50:NEXTO
980 POKEPI,32:PI=PI+22:NEXTI:POKEPI,64
990 Q1=0:QO=0:GOTO500
1000 REM *PUNKTEANZEIGE*
1010 IFET=9THENPRINT"8"TAB(19)" "
1020 PRINT"SPUNKTE";P:PRINT"*****ZEIT";ET
1030 POKE36876,0:RETURN
1100 REM NEUER MANN
1110 POKE36879,127:POKE36869,194:AB=AB+1
1120 ONABGOTO1130,1140,1150
1130 PRINT"DEINER FOLGT DER" NZWEITE MANN." :FORO=1TO1500:NEXTO:GOTO30
1140 PRINT"DEINER FOLGT DER" MLETZTE MANN." :FORO=1TO1500:NEXTO:GOTO30
1150 REM SPIELEND
1160 IFP>HITHENHI=P
1170 POKE36879,127:POKE36869,194:POKE649,10
1180 PRINT"DEIN NACHSTPUNKTZAHL:" :PRINT"*****HI:PRINT"DEIN PUNKTE: "P
1190 INPUT"DEIN NACHEINMAL ";C$
1200 IFC$="JA"THENPRINT"J":AB=0:P=0:GOTO30
1210 IFC$<>"NEIN"THEN1190
1220 SYS64818
1300 REM KAMPF HOEHLNBAER-MANN
1310 POKEHB,32:FORL=1TO20:PJ=INT(RND(1)*5)+74:POKEPI,77
1320 POKE36877,200
1330 FORO=1TO100:NEXT
1340 POKE36877,0:PJ=INT(RND(1)*5)+69
1350 POKEPI,PJ:POKE36876,200
1360 FORO=1TO100:NEXT:POKE36876,0
1370 NEXTL
1380 POKEPI,77
1390 GOTO500
1400 REM SIRENE
1410 FORL=1TO5:FORM=180TO235STEP3
1420 POKE36876,M:FORM=1TO10
1430 NEXTM
1440 POKE36876,0:FORM=1TO100
1450 NEXTM,L
1460 POKE36878,15
1470 RETURN
1500 REM STEINGFALL IN DER SCHATZHOEHLE
1510 ONGGGOTO1520,1560
1520 I=INT(RND(1)*7)+1
1530 FORL=15TO0STEP-3:FORO=1TO2
1540 POKE36878,L:POKE36875,230:POKE36877,230:NEXTO,L
1550 POKE36875,0:POKE36877,0:POKE36878,15
1560 POKE38266+I,1:POKE4474+I,32:I=I+22
1570 IFI>88THENG=1:POKE4474+I,76:POKE38266+I,4:POKE4474+I-88,76:POKE38266+I-88,
4:RETURN
1580 POKE4474+I,76:POKE38266+I,4:G=2
1585 IFPI=45180RPI=45260RPI=45170RPI=4527THENG=0:P=P+T2:T2=0
1590 IFPI=4474+I0RPI=4496+ITHEN1600

```

```

1595 RETURN
1600 REM VOM FELSEN ERFASST
1610 IF I>66 THEN 500
1620 POKE 4452+I,32:POKE 4452+22+I,32
1630 I=I+22
1640 POKE 4474+I,76:POKE 4474+I+22,PJ:POKE 38266+I,4:POKE 38288+I,1
1650 GOTO 1610
2000 REM BEINBEWEGUNGEN
2010 K1=-K1:PJ=67:IF K1=-1 THEN PJ=68
2020 RETURN
2030 K1=-K1:PJ=71:IF K1=-1 THEN PJ=72
2040 RETURN
2050 K1=-K1:PJ=69:IF K1=-1 THEN PJ=70
2060 RETURN
2070 K1=-K1:PJ=73:IF K1=-1 THEN PJ=74
2080 RETURN
2090 K2=-K2:HJ=81:IF K2=-1 THEN HJ=82
2100 RETURN
2110 K2=-K2:HJ=79:IF K2=-1 THEN HJ=80
2120 RETURN
2130 K2=-K2:HJ=83:IF K2=-1 THEN HJ=84
2140 RETURN
2150 K2=-K2:HJ=77:IF K2=-1 THEN HJ=78
2160 RETURN
10000 REM *SPIELFELDAUFBAU*
10010 RESTORE:POKE 36869,205:FOR I=1 TO 462:READ F1:POKE 4139+I,F1
10020 IF I>133 AND I<139 THEN F2=3:GOTO 10100
10030 IF F1=65 THEN F2=5:GOTO 10100
10040 IF F1=102 THEN F2=7:GOTO 10100
10050 IF F1=119 OR F1=32 THEN F2=1:GOTO 10100
10060 IF F1=66 OR F1=90 THEN F2=6:GOTO 10100
10070 IF F1=76 THEN F2=4:GOTO 10100
10080 IF F1=64 THEN F2=2
10100 POKE 37931+I,F2:NEXT I
10110 RETURN
11000 REM *TITELBILD*
11010 POKE 36879,25:PRINT "          "
11020 PRINT "          "
11030 PRINT "          "
11040 PRINT "          "
11050 PRINT "          "
11060 PRINT "          "
11070 PRINT "          "
11080 PRINT "          "
11090 PRINT "          "
11100 PRINT "          "
11110 PRINT:PRINT "          BITTE WARTEN:"
11120 PRINT "          "
11130 PRINT "          "
11140 PRINT "          "
11150 PRINT "          "
11160 PRINT "          "
11170 PRINT "          "
11180 PRINT "          "
11190 PRINT "          "
11200 PRINT "          "
11210 PRINT "          "
11999 REM *NEUER ZEICHENSATZ*
12000 FOR I=1 TO 470:READ A:B=B+A:NEXT I
12005 IF B>36243 THEN PRINT "          FEHLER IN DEN SPIELFELDDATEN"
12010 B=0:FOR I=0 TO 2047:POKE I+5120,PEEK(32768+I):NEXT I
12020 DX=5632
12030 FOR L=0 TO 20
12040 FOR I=0 TO 7:READ A:POKE DX+I,A:B=B+I:NEXT I
12050 DX=DX+8:B1=B+B
12060 NEXT L:IF B1>6468 THEN PRINT "          FEHLER IN DEN          ZEICHENDATEN":END
12070 B=0:B1=0
12100 RETURN
13000 REM SPIELBESCHREIBUNG
13010 POKE 36879,42:POKE 36869,194
13020 PRINT "          SPIELBESCHREIBUNG          WER WUENSCHT ?          "
13030 POKE 198,0:WAIT 198,1:GET A$
13040 IF A$="N" THEN RETURN
13050 IF A$="J" THEN 13030
13060 PRINT "          "

```



```

13070 PRINT"MAEI DIESEM *PIEL MUES-SEN *IE NACH UNTER-   IRDISCHEN *CHAETZEN
GRABEN."
13080 PRINT"ABEI IST AUF VER-   SCHIEDENE IEFAHREN ZU ACHTEN : "
13090 PRINT"NAUF *TEINE, DIE AUF   EINEN HERABFALLEN,   WENN MAN SIE UNTER-   Q
UERT."
13100 PRINT"NAUF DEN   IOEHLENBAEREN."
13110 GOSUB13500
13120 PRINT"MAN DER *CHATZHOEHLE   AUF HERUNTERFALLENDE   ELSEN."
13130 PRINT"UND ZUM *CHLUSS NOCH   AUF DIE, HINTER EINEM,ZUSAMMENBRECHENDE   I
RUECKE."
13140 GOSUB13500
13150 PRINT"***** 7//17-IL//I ***"
13160 PRINT"JUER JEDEN KLEINEN   *CHATZ, DEN MAN   ERREICHT BEKOMMT MAN
100";
13165 PRINT" PUNKTE."
13170 PRINT"JUER JEDEN GROSSEN   *CHATZ, DEN MAN AUS   DER *CHATZHOEHLE"
13180 PRINT"BRINGT, BEKOMMT MAN   200 PUNKTE."
13190 PRINT"JUER JEDEN GROSSEN   *CHATZ DEN MAN AN DIE *ROBERFLAECHE BRINGT"
13200 PRINT"BEKOMMT MAN   500 PUNKTE."
13210 GOSUB13500
13220 PRINT"MAN HAT 99 *EKUNDEN   *EIT, UM DIE   *CHAETZE ZU BERGEN."
13230 PRINT"UNTS BEGINNT EINE NEUE *PIELRUNDE, WENN MAN   MINDESTENS EINEN"
13240 PRINT"GROSSEN *CHATZ AN DIE *ROBERFLAECHE BRINGT."
13250 GOSUB13500
13260 PRINT"***** 7//17-IL//I *****"
13270 PRINT"UNTESTEUERT WIRD MIT DEN IASTEN : "
13280 PRINT"UNTS : NACH LINKS":PRINT"UNTS : NACH RECHTS"
13290 PRINT"UNTS : NACH OBEN":PRINT"UNTS : NACH UNTEN"
13300 PRINT"UNTS MIT DEM JOYSTICK"
13310 PRINT"UNTS STEHEN EINEM   3 *RENNER ZUR   VERFUEGUNG!"
13500 PRINT"UNTS IASTE RUECKEN."
13510 POKE198,0:WAIT198,1
13520 RETURN
14000 REM JOYSTICKROUTINE
14010 POKE1,60:POKE2,3
14020 FORI=0TO63:READA:POKE828+I,A:B=B+A:NEXT
14030 IFB<6738THENPRINT"UNTSFEHLER IN DEN   JOYSTICKDATEN":END
14040 B=0
14050 RETURN
20000 REM *DATEN FUER SPIELFELD*
20010 DATA64,64,64,32,64,64,65,64,64,64
20020 DATA64,64,64,64,64,64,64,64,64,64
20030 DATA64,64,64,102,102,102,102,102,102,102
20040 DATA102,65,102,102,102,102,102,65,102,102
20050 DATA32,102,102,64,64,64,32,64,64,64
20060 DATA64,102,102,102,102,102,102,102,102,102
20070 DATA102,102,32,102,102,64,64,32,32,32
20080 DATA32,32,32,65,102,102,102,102,102,102
20090 DATA32,32,32,32,32,102,102,64,64,119
20100 DATA119,119,119,119,64,102,102,102,65,102
20110 DATA102,102,32,102,66,65,102,102,102,64
20120 DATA64,32,32,32,32,32,64,65,102,102
20130 DATA102,102,102,102,32,102,102,102,102,65
20140 DATA102,64,64,64,64,64,64,64,64,102
20150 DATA102,65,102,102,102,102,32,102,65,102
20160 DATA102,102,102,64,64,64,64,64,64,64
20170 DATA64,102,102,102,102,65,102,102,32,102
20180 DATA102,102,65,102,102,64,64,102,102,102
20190 DATA102,102,65,102,102,102,102,102,102,102
20200 DATA32,102,102,102,102,102,102,64,64,102
20210 DATA32,32,32,32,32,32,32,32,32,32
20220 DATA32,32,32,32,32,32,32,102,102,64
20230 DATA64,102,32,102,102,102,102,102,102,102
20240 DATA102,102,102,32,102,102,102,32,102,102
20250 DATA64,64,102,32,102,102,102,102,102,102
20260 DATA102,102,65,102,102,32,102,102,65,32
20270 DATA65,102,64,64,102,32,102,65,102,102
20280 DATA65,102,102,102,102,102,102,32,102,102
20290 DATA102,66,102,102,64,64,102,32,102,102
20300 DATA102,102,102,102,102,102,102,102,102,32
20310 DATA102,102,102,65,102,102,64,64,102,32
20320 DATA65,64,64,64,64,64,64,64,64,64
20330 DATA102,32,102,65,102,102,102,102,64,64
20340 DATA102,32,102,64,76,76,76,76,76,76
20350 DATA76,64,102,66,102,102,102,102,102,102

```

```

20360 DATA64,64,102,32,65,64,32,32,32,32
20370 DATA32,32,32,64,102,102,65,102,65,102
20380 DATA102,102,64,64,102,66,102,32,32,32
20390 DATA32,32,32,32,32,32,65,102,102,102
20400 DATA102,102,102,102,64,64,102,102,102,64
20410 DATA32,32,32,32,32,32,64,102
20420 DATA65,102,102,102,102,102,64,64,102,102
20430 DATA102,64,32,90,64,90,64,90,32,64,102,102,102,102,102,102,102
20440 DATA64,64,64,64,64,64,64,64,64,64
20450 DATA64,64,64,64,64,64,64,64,64,64
20460 DATA64,64,64,64,64,64,64,64,64,64
30000 REM *DATEN FÜR NEUEN ZEICHENSATZ*
30010 DATA255,255,255,255,255,255,255,255
30020 DATA103,255,255,126,254,255,255,122
30030 DATA0,0,0,0,0,24,60,24
30035 DATA28,28,8,124,28,28,18,34
30040 DATA28,28,8,124,30,28,36,34
30050 DATA56,56,16,62,120,56,36,66
30060 DATA56,56,16,62,56,56,72,68
30070 DATA0,8,200,59,63,123,200,0
30080 DATA0,136,72,59,63,59,192,0
30090 DATA0,9,222,252,220,19,16,0
30100 DATA0,3,222,252,222,17,16,0
30110 DATA28,28,8,28,62,28,20,20
30120 DATA255,255,255,191,126,124,108,4
30130 DATA0,4,6,127,254,120,120,40
30140 DATA0,4,6,127,254,120,120,68
30150 DATA0,32,96,254,127,30,30,20
30160 DATA0,32,96,254,127,30,30,34
30170 DATA16,120,248,120,248,30,28,8
30180 DATA16,248,120,120,120,158,28,8
30190 DATA8,28,30,248,120,248,120,16
30200 DATA8,28,158,120,120,120,248,16
40000 REM *DATEN FÜR JOYSTICKROUTINE*
40010 DATA173,19,145,72,169,0,141,19
40020 DATA145,173,17,145,41,28,74,74
40030 DATA133,99,173,17,145,41,32,10
40040 DATA10,5,99,133,99,104,141,19
40050 DATA145,162,127,142,34,145,173,32
40060 DATA145,162,255,142,34,145,41,128
40070 DATA74,74,74,74,5,99,73,143
40080 DATA133,99,169,0,133,98,162,144
40090 DATA56,32,73,220,96

```

TASTATUR-MEISTER® ist da!

Gibt die schnelle Referenz für Befehle, Optionen und Formate direkt an den Arbeitsplatz –

BASIC Funktionen

STEUER-TASTEN Schneller Überblick



ASCII Tabelle zum Aufhängen
(Nicht abgebildet)

TASTATUR-MEISTER®

- Befehle vollständig vorhanden, mit voll ausgeschriebener Befehlsbeschreibung
- Alle Steuertasten übersichtlich

SCHABLONE aus: ● Widerstandsfähigem Kunststoff
● Dauerhaftem Druck
● Eingeteilt nach Nutzungsbereichen

Händeranfragen erwünscht.

Hersteller:

FÜR COMMODORE VC 20 und C-64

- BASIC x á DM 29,90
 - SIMON'S BASIC x á DM 29,90
 - MULTIPLAN x á DM 29,90
 - BLANKO x á DM 27,00
- (für eigene Anwendungen)

Schablonen auch für IBM-PC erhältlich.

Das 1.1/2.0, BASIC 1.1/2.0 WORDSTAR, MULTIPLAN
LOTUS 1,2,3, BLANKO

FRIWA-Vertrieb · Reisingerstr. 6 · 8 München 2 · Tel. 0 89/53 04 50 · Telex 5 213 775

Senden Sie bitte TASTATUR-MEISTER:

- ☐ per Nachnahme (zzgl. NN-Gebühr und 5,- DM Versandgebühr)
- ☐ per Scheck (zzgl. 5,- DM Versandgebühr) (liegt bei)

Name

Straße

PLZ Ort

CM-LEXIKON

Es folgt Teil 2 unseres Wörterbuches, diesmal ist der Buchstabe B an der Reihe.
Haben Sie sich das letzte CM-Wörterbuch aufgehoben?
Bleiben Sie am Ball, weitere wichtige Begriffe aus der Computerwelt werden auch in den nächsten Ausgaben erklärt!

Wenn Sie sich alle Teile unseres kleinen Wörterbuches sammeln, so haben Sie am Ende unserer Reise durch das Computeralphabet ein schönes Sammelwerk, in dem alles Wissenswerte verzeichnet ist.

Baud:

ist die Maßeinheit für die Schrittgeschwindigkeit bei der Datenübertragung. Sie gibt an, wieviele Zeichen pro Sekunde vom Computer zu Peripheriegeräten übertragen werden.

Bit:

Bit ist die kleinste Einheit des Dualsystems. Der Wert eines Bits kann nur 0 oder 1 sein. Durch Aneinanderreihen von Bits entstehen Informationen, die der Computer versteht und verarbeitet. Dabei werden je 8 Bits zu einem Byte zusammengefaßt.

Befehlssatz:

Gesamtheit aller Befehle, die ein bestimmter Microprozessortyp versteht und bearbeiten kann.

Basic:

Die wohl beliebteste und meistgebrauchte Programmiersprache für Personal- und Heimcomputer. Basic steht für "Beginners All purpose Symbolic Instruction Code". Basic hat immer den gleichen Grundsatz, kann sich aber durch erweiterten Befehlssatz von anderen Basic-"Dialekten" unterscheiden. Da alle Basic-Programme erst vom Interpreter in Machinensprache umgesetzt werden, gehört Basic zu den langsamen Programmiersprachen. Da der Programmablauf in kleine Einzelschritte zerlegt wird und Basic sich am Schulenglisch orientiert, ist es aber auch gleichzeitig die einfachste Sprache.

BDSG:

Abkürzung für Bundesdatenschutzgesetz

Bus:

Unter Bus versteht man ein System von zusammengehörigen Leitungen, die eine bestimmte Funktion bei der Datenübertragung ausüben: Es gibt Steuerbusse, Adreßbusse und Datenbusse, auf den die jeweiligen Bits übertragen werden. Mit Bussen werden Geräte und Baugruppen innerhalb des Computers miteinander verbunden.

Boole'sche Algebra:

Rechenregeln logischer Verknüpfungen, wurde vom Mathematiker Boole erfunden. Boole hat für das Dualsystem Regeln aufgestellt, wie die Operationen "und", "oder", die Negation "nicht" bzw. "nicht oder" festgelegt werden. Dazu noch die Regeln "falsch" oder "wahr". Damit lassen sich alle logischen Beziehungen untereinander darstellen und verknüpfen.

BTX:

steht für Bildschirmtext. Mit diesem System ist der interaktive Betrieb unter Computern möglich. Als Datensichtgerät dient der Fernsehapparat, als Abfrage- und Eingabegerät können neben speziellen BTX-Tastaturen auch angepaßte Personal- und Heimcomputer verwendet werden. Zur Datenleitung wird das öffentliche Telefonnetz.

BDOS:

Teil des Betriebssystems CP/M. Es hat die Aufgabe, den Datentransport zwi-

schen Computer und externen Speichern (z.B. Floppy) zu übernehmen.

Das BDOS organisiert interne Dateien und überwacht den Datenverkehr. Folge bei Unregelmäßigkeiten des Datenverkehrs – Fehlermeldung.

BCD-Code:

steht für binär codiertes Dezimalsystem. Mit vier Bits läßt sich jede Ziffer des Dezimalsystems eindeutig darstellen. Die Umwandlung erfolgt durch die Umrechnung, welche Potenz der Zahl 2 in der Dezimalziffer vorhanden ist.

Beispiel: Die Zahl fünf ist $0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$. Im BCD-Code wäre die fünf also 0101.

Byte:

besteht aus 8 bits. Da zum Speichern eines Zeichens 8 bit (egal, ob es sich dabei um einen Buchstaben, ein Symbol oder eine Zahl handelt) benötigt werden, mißt man die Kapazität eines Speichers in Byte. Eine Speichergröße von 1000 Bytes bedeutet daß 1000 Zeichen gespeichert werden können.

Binärcode:

bedeutet, daß alle Daten, Ziffern und Zeichen und alle Vorgänge in einem Computer auf die Binärzahlen 0 und 1 reduziert und umcodiert werden. Der Binärcode ist die Sprache, die der Computer versteht und verarbeitet. Der Binärcode wird auch als Dual-Code bezeichnet, weil die Reihenfolge der "Einer" und "Nullen" von Potenzen der Zahl 2 abhängig ist.

**** COMMODORE 64 BASIC V2 ****
64K RAM SYSTEM 38911 BASIC BYTES FREE

Basic-Kurs Teil 4

Wie letzte Woche angekündigt, wollen wir ab dieser Lektion mit dem richtigen Programmieren beginnen. Das bedeutet, daß das Zahlengeplänkel der Grundübung abgeschlossen ist und wir uns nun mit einem Programmablauf (ALGORITHMUS) befassen wollen. Doch zuvor müssen wir noch einen Befehl lernen, der für das erste Programm unerlässlich ist. Es ist der Input-Befehl.

Der Input-Befehl

Dieser Befehl steht immer in der Form:

INPUT "ANWEISUNG";
VARIABLE

im Programm, und ist etwas ganz Komfortables, da er gleichzeitig einen PRINT-Befehl sowie eine VARIABLEN-WERT-Zuweisung beinhaltet.

Findet der Interpreter einen Input-Befehl, so sieht er nach, ob in der Anweisungszeile ein Text steht. Findet er etwas, so schreibt er den Text aus und geht danach in die Eingabebereitschaft. Alles was nun eingegeben wird, weist der Computer nach Drücken der RETURN-Taste der angegebenen Variablen zu. Geben Sie also ein:

INPUT "WIEVIEL STOCKWERKE HAT EIN DREISTOECKIGES HAUS ?";A

Nachdem Sie RETURN das erstmal gedrückt haben, erscheint auf dem Bildschirm:

WIEVIEL STOCKWERKE
HAT EIN DREISTOECKIGES
HAUS?

?

Wenn Sie nun die Lösung errechnen haben, so geben Sie die korrekte Zahl (sie ist größer als 2 und kleiner als 4) über die Tastatur ein. Sie werden sehen, daß sie hinter dem Fragezeichen erscheint. Wenn Sie nun Return drücken, meldet sich der Computer mit dem ganz gewöhnlichen READY.

Allerdings steht die Zahl nicht nur hinter dem Fragezeichen auf dem Bildschirm, sondern auch im Speicher unter der Variablen A.

Dies können wir ganz leicht beweisen, indem wir die Variable A einfach abfragen, was wir ja schon aus früheren Lektionen her können und tippen ein:

PRINT A
und der Computer meldet sich tatsächlich mit dem in der INPUT-Zeile zugewiesenen Wert für A.

Natürlich funktioniert das

auch, wenn Sie die Textzeile weglassen und nur:

INPUT A

eingeben. Der Computer schreibt auch dann das FRAGEZEICHEN und wartet darauf, daß Sie Ihre Eingabe mit RETURN abschließen.

Betrachten wir uns allerdings BASIC-Programme hier im Heft, so fallen uns zunächst einmal kleine Zahlen am Anfang jeder Zeile auf, die wir noch nie in einem unserer Programme verwendet haben. Dies soll nun anders werden, und wir werden auch diese Zeilennummern verwenden und damit unser eigenes Programm schreiben.

Der Algorithmus

Wenn Sie sich morgens einen Kaffee kochen, gehen Sie immer in einer bestimmten Reihenfolge vor. Diese Reihenfolge sieht laut einer Umfrage bei über 90% unserer Leser folgendermaßen aus:

1. Wasser in die Kaffeemaschine geben
2. Filtertüte in den Filter stecken
3. Kaffee mahlen
4. Kaffeepulver in Filtertüte streuen
5. Alles laut Bedienungsanleitung zusammenbauen
6. Maschine anstellen

Kaum einer unserer Leser würde auf die Idee kommen, erst das heiße Wasser durchlaufenzulassen und dann die Bohne hineinzugeben und das ganze in die Kaffeemühle zu schütten.

Erstens geht davon die Mühle kaputt und zweitens schmeckt es auch einfach nicht so richtig.

Diese notwendige Folge von Abläufen, die uns jeden Morgen einen Kaffee beschert, nennen wir ALGORITHMUS. Ein Algorithmus ist also eine Folge logischer Operationen, die uns ein gewünschtes Ergebnis liefern soll.

Genauso werden wir es jetzt im Programmieren halten. Denn auch hier hat es wenig Zweck, erst eine Variable abzufragen und sie dann erst

zu definieren.

Nun weiß der Computer jedoch nicht so ohne weiteres, wie er sich in einem solchen Algorithmus zurechtfinden soll. Er benötigt dafür Zeilennummern. Während des Programmablaufs verarbeitet der Interpreter seine Befehle nach den Zeilennummern.

Diese beginnen im BASIC V2,0 mit Null und gehen bis Fünfundsechzigtausend aufwärts. Wenn wir z.B. eingeben:

10 S=66

50 PRINT S

und diesen Algorithmus dann auch noch mit RUN starten, so führt der Computer zuerst den Befehl in der niedrigsten Zeile aus und da steht S=66. Also schreibt der Computer in seinen Variablenspeicher für S eine 66 auf, und dann sucht er sich die nächste Zeile. In unserem Fall ist dies die 50. Es könnte auch jede andere Zeilennummer sein, die

höher ist als 10, der Computer sieht nur die nächst höhere Zeile, die im Programm vorkommt, als nächsten Befehl an. In Zeile 50 steht nun PRINT A, was der Computer auch sofort ausführt und uns eine 66 auf den Bildschirm schreibt. Danach ist dieses Programm beendet und der Interpreter teilt uns durch ein READY mit, daß er fertig ist. Ein einmal geschriebenes Programm kann man sich durch den Befehl LIST jederzeit wieder ansehen.

Das wäre im großen und ganzen schon alles, was es über einen Algorithmus zu sagen gibt, man kann in diesem Algorithmus auch noch beliebig herumspringen, aber dazu kommen wir nächste Woche. Für diesmal haben wir noch ein Programm, welches alle bisher gelernten Befehle beinhaltet und Ihr Alter in Tagen berechnet:

3 x duffe Spiele,
Tips und Tricks, News,
Berichte, Software-News,
in: Homecomputer, CPU
und Computerposter,
das Superposter mit dem
Spiel auf der Rückseite.
Jeden Monat neu bei
Ihrem Zeitschriftenhändler.

Roeske Verlag
Fuldaer Straße 6
3440 Eschwege

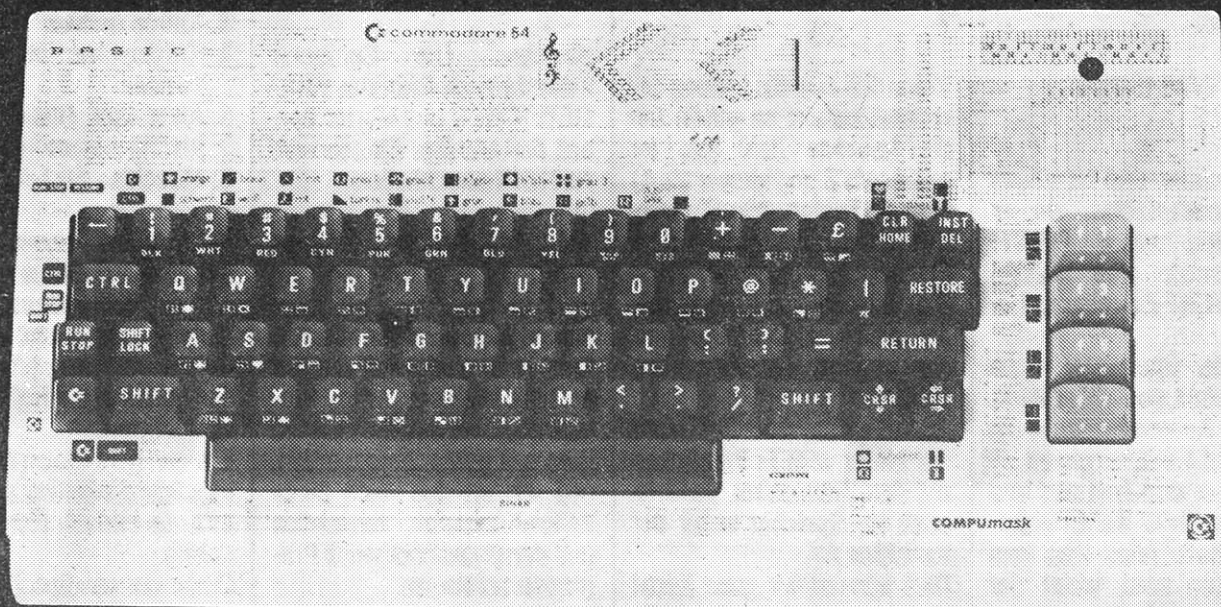


```
10 TAGE=365
20 INPUT "WIE ALT SIND
SIE IN JAHREN";ALTER
30 ERGEBNIS=ALTER*
TAGE
40 PRINT "SIE SIND ";ER-
GEBNIS;" TAGE ALT"
50 PRINT "AUF WIEDER-
SEHEN"
```

Das Ergebnis stimmt nicht auf den Tag genau, aber das ist auch eine etwas komplizierte Sache und kommt in einer der nächsten Folgen.

Aufgabe zur nächsten Lektion:

**Verändern Sie das Programm
soweit, daß es außer den Ta-
gen auch noch die Anzahl der
Monate angibt.**



SCHLUSS mit dem unproduktiven Stöbern in überall herumliegenden Programmierunterlagen.

SCHLUSS MIT FRUST UND ZEITVERLUST!

Von einigen hunderttausend VC20- und C64-Besitzern sind nur wenige bis zu Sound und Grafik vorgedrungen.

DAS SOLL SOFORT ANDERS WERDEN!

I. Dinkler, Idee-Soft, Am Schneiderhaus 7,
D-5760 Amsberg 1, Tel.: 02932/32947

C-64 – OBERSEITE

50 BASIC-Begriffe. Ausgabezeichen für Zeichenfarben, F-Tasten, Cursor, INST/DEL und CLR/HOME. POKEs für Bildschirm-Rahmen, -hintergrund, -positionen, Zeichenfarben, Sound (3 Stimmen, Geräusch, 4 Oktaven). Fehler. BIN/DEZ-Umrechnung. Uhrzeit. 18 Floppy- und Druckerbefehle. Hüllkurve, Welle, Filter, Notenwerte. – Steuertastenerläuterung. Sprite-Pokes.

RÜCKSEITE: Felder, Edier- und Formationskommandos. Logische und Rechenoperationen. BIN/DEZ/HEX-Tabelle. Ungeeignete Variablen. Bildschirm-, ASCII- und CHR\$-Codes. Scrolling-POKEs Sound- und Grafik-Demoprogramme. Sprite-Data-Tafel.

VC 20

Wie C 64 ohne die für diesen geltenden Besonderheiten. Dazu: Farbentabelle für Bildschirmrahmen und -grund. Speicherorganisation.

COMPUMask zeichnet sich aus durch guten Sitz, abrieb- und reinigungsmittelfesten Mehrfarben-Kunststoff in HI - RES unter flexiblem, reißfestem Edelkunststoff in computerdeckenden Maßen.

Erhältlich in allen Fachgeschäften zum Preis von ca. DM 30,-. Für Lieferrückweis genügt Postkarte oder Anruf.

C64/IEEE-488 Steckmodul

Dieser ausgereifte, weltweit erprobte IEEE-488-Modul eröffnet dem Commodore 64 über seinen parallelen Ausgang ungeahnte Einsatzmöglichkeiten wie:

große, IEEE-kompatible CBM-Peripherie am C-64, simultanen (seriell – VC/paralleler – IEEE) Datenverkehr. Konfliktfreie, speicherverschiebbliche Modulsoftware. Im Einsatz beispielsweise **in Schulen** ermöglicht der IEEE-488-Steckmodul problemlose Mehrbenutzersysteme am IEC-Bus wie auch durch die rationell genutzte Peripherie: z. B. zahlreiche Computer an einer Doppelfloppy.

In der Industrie bietet der IEEE-488-Steckmodul die Möglichkeit für preisgünstige IEC-Meß-/Steuersysteme mit dem Commodore 64 als Controller. Zu diesem Modul wird ein **Betriebshandbuch** geliefert, in dem Beschreibungen zu fast sämtlichen Anwendungsfällen mit Programmbeispielen, Belegungstabellen, Angaben zum erforderlichen Kabel- und Steckmaterial, Literatur etc. aufgeführt sind. Zusätzlich können zum IEEE-488-Steckmodul **Anwendungshilfen** wie u. a. Disketten mit Lesekennzeichen, Utility-Disketten usw. bezogen werden.

IEEE-Steckmodul für Commodore 64
einschließlich Betriebshandbuch DM 239,— inkl. MwSt.



te-wi

te-wi Verlag GmbH
Theo-Prosel-Weg 1
8000 München 40

Weiterführende Literatur...



NEU! C-64 Computerhandbuch

Ein Handbuch für jeden Erfahrungsstand von der ersten Begegnung bis zum professionellen Einsatz des COMMODORE 64 bzw. 1541. Das Werk ist sehr bildreich und bietet somit eine schnelle Übersicht – als echtes Nachschlagewerk werden Sie es stets in der Nähe Ihres Computers finden.
Raeto West, ca. 400 Seiten, Softcover, DM 56,—, 4. Qu. 84



NEU! C-64 Akustik und Graphik

Ein planvoller Lehrgang – keine Beispielsammlung – in anschaulichem Stil – daher für jedes Alter. Dieses Werk eröffnet dem C-64 Benutzer die Welt der Graphiken und Klangbilder. Es enthält Programmbibliotheken und wird abgerundet durch zahlreiche Anhänge.
John Anderson, ca. 200 Seiten, Softcover, DM 49,—, 4. Qu. 84



6502 - Programmieren in Assembler

Dieses Buch behandelt ausführlich die Assemblersprachen-Programmierung für den weitverbreiteten Mikroprozessor 6502. Er steckt auch in Ihrem C-64.
Lance Leventhal, 704 Seiten, Softcover, DM 59,—



CBM Computer Handbuch

Dieses unentbehrliche Nachschlagewerk bietet eine wahre Fundgrube – mit einer schrittweisen Einführung bis hin zur Darstellung aller professionellen Möglichkeiten dieses beliebten Computers.
Osborne Danahue, 544 Seiten, Softcover, DM 59,—



NEU! LOGO Computersprache für Kinder und Eltern

Dieses Buch beweist: **Jeder kann programmieren.** LOGO ist die Computersprache für Eltern und Kinder. Nicht umsonst wurde dieser Titel zum „Buch des Jahres 1983“ in den USA. LOGO ist das Ergebnis der Erforschung menschlicher Intelligenz; entwickelt von einem Pädagogen und Mathematikprofessor. LOGO ist die erste Computersprache, die bewußt Strategien menschlichen Denkens dient.
Daniel Watt, ca. 400 Seiten, Softcover, DM 59,—, 4. Qu. 84

Coupon

Hiermit bestelle(n) ich (wir):

Datum

Unterschrift

Name/Firma

Straße/Postfach

Ort

Der Sensible C-64
CP/M und WordStar
C-64 Programmsammlung
VisiCalc (mit CBM Diskette)
77 BASIC Programme
Mikrocomputer-Grundwissen
C-64 Computer für Kinder
VC20 Computer für Kinder

4. Q. 84, DM 29,80
DM 29,80
4. Q. 84, DM 29,80
DM 79,—
DM 39,—
DM 36,—
4. Q. 84, DM 29,80
4. Q. 84, DM 29,80

Teil 4

Logik im Basic-Programm

Wie man Systematik in sein Selbstgestricktes bringt

Computer- und damit auch Basic-Programme bestehen nur zum Teil aus Rechenanweisungen. Mindestens ebenso wichtig sind die verschiedenen Formen der Entscheidung, die vom Rechner im Verlauf des jeweiligen Programmes getroffen werden sollen. Wir wollen uns deshalb heute ein wenig mit diesem wichtigen Element der Programmierung beschäftigen...

1. Einseitige Entscheidung
Ohne unser bekanntes Beispiel, das "Rechnung-Schreibprogramm", über Gebühr strapazieren zu wollen, möchte ich es doch noch einmal heranziehen, weil es sich zur Verdeutlichung unseres heutigen Themas recht gut eignet.

Wir nehmen dazu an, daß wir im Fall eines geringen Bestellwertes eine Versandkostenpauschale in Rechnung stellen. Es gilt also, eine Routine zu schreiben, die z.B. 7,50 DM Versandkosten addiert, wenn der Gesamtwarenwert kleiner als 150,- DM ist.

Was hat eine solche Routine exakt zu tun?

Zunächst muß G (Variable für den Gesamtwarenwert) dahin geprüft werden, ob es kleiner, gleich oder größer als 150 ist. Mathematisch formuliert heißt diese Frage dann:

Ist $G < 150$? (so geschrieben, kann sie leicht mit JA bzw. NEIN beantwortet werden). Lautet die Antwort JA, so muß die folgende Anweisung gelten:

$G = G + 7.5$

Im gegenteiligen Fall geschieht nichts, und das Programm läuft wie gehabt weiter.

Natürlich möchte ich als fairer Geschäftsmann meinen Kunden nicht über den Grund für die zusätzliche Belastung im Unklaren lassen. Daher muß für den Fall eines JA auch der Ausdruck eines entsprechenden Hinweises sichergestellt sein.

Wir schreiben also zwei Handlungen des Rechners vor:

1. Handlung: IF $G < 150$ THEN $G = G + 7.5$

2. Handlung: IF $G < 150$ THEN PRINT "ZUZUEGL:VERSAND 7,50DM"

Prüfen Sie mal die Lauffähigkeit dieser Anweisungen. Dabei müssen Sie natürlich nicht das komplette Rechnungsprogramm eintippen. Es reicht, wenn Sie mittels INPUT beim Programmstart die Eingabe des Ausgangs-G gewährleisten. Also:

```
10 INPUT G
20 IF G < 150 THEN
  G = G + 7.5
30 IF G < 150 THEN PRINT
  "ZUZUEGL.VERSAND
  7,50DM"
40 PRINT G
```

Was tut sich? Zunächst wird es sicher einwandfrei funktionie-

nieren. Spielen Sie aber das kleine Programm mit verschiedenen G-Werten durch, die schrittweise über die 150 Mark-Schwelle hinaus führen. Probieren Sie es mit 140, 142, 145, 149, 155, 160 Mark.

Was stellen Sie fest?

Überlegen Sie kurz, wann das Programm beginnt, falsche Ergebnisse zu liefern. Machen Sie sich Gedanken über den Grund für dieses Fehlverhalten! (Lesen Sie nicht gleich weiter).

Unser Beispiel zeigt anschaulich einen Mangel der Programmiersprache BASIC auf, mit dem man zwar leben kann, den man aber erkennen sollte. BASIC legt den Programmablauf sequentiell fest, d.h. in einem steten Nacheinander (Beweis: die Zeilennummern). Leider haben wir es bei Programmabläufen aber auch mit Gleichzeitigkeiten zu tun, mit deren Beschreibung sich unsere Programmiersprache ein wenig schwer tut. Analysieren wir kurz die Art und Weise, wie einerseits wir, andererseits BASIC unser VERSANDKOSTEN-PROGRAMM verstanden haben.

10 INPUT G: Hier ist alles klar.

G wird angefordert. Beide Parteien sehen noch keine Schwierigkeiten.

20 IF $G < 150$ THEN

$G = G + 7.5$: Auch hier ist die Anweisung beiderseitig noch klar: Das Problem bahnt sich aber schon an. Gehen wir nämlich vom Fall $G = 145$ aus, so wird in dieser Zeile G auf 152,5 erhöht. In der Folgezeile

30 IF $G < 150$ THEN PRINT "ZUZUEGL.VERSAND 7,50DM": wird dann nicht mehr mit dem "alten" sondern bereits aufgestockten G gearbeitet, das inzwischen größer als 150 geworden ist.

Dadurch haben wir zwar die Versandkostenpauschale berechnet, aber nicht ausgewiesen.

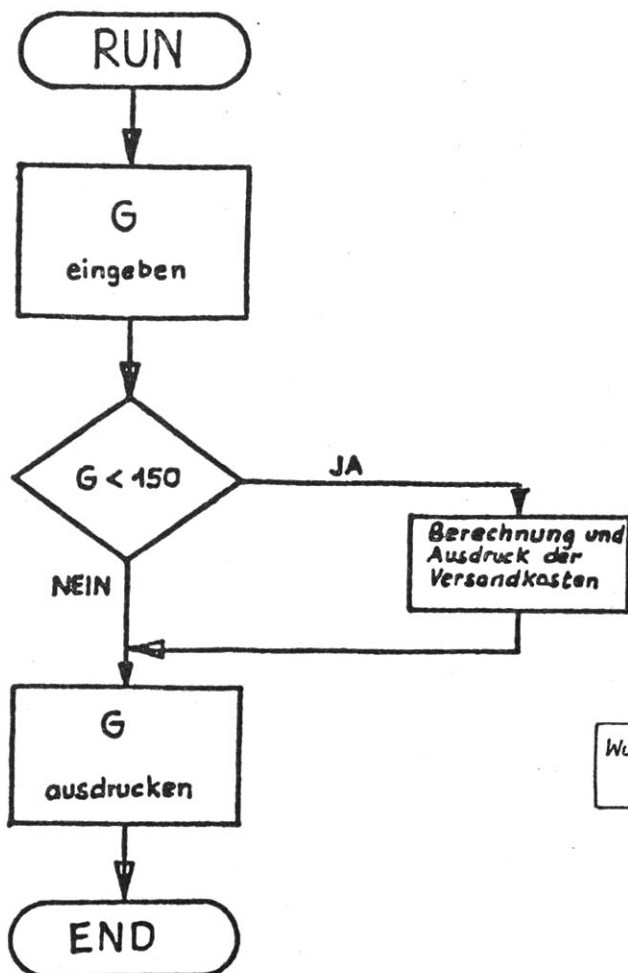
40 PRINT G: beweist das.

Während also wir die Gleichzeitigkeit zweier nötiger Handlungen im Sinn hatten, hält sich der Basic-Interpreter strikt an die chronologische Folge. Wir müssen somit von unserer Schreibweise abgehen und so strukturieren, daß unser BASIC eine gegebene Anweisung ebenso interpretiert wie wir.

Mit unseren bisherigen Kenntnissen lösen wir das am einfachsten durch die Umset-

zung der nötigen Handlungen in ein Unterprogramm, das bei positivem Testergebnis ($G < 150$) angesprungen wird. Unser Mini-Programm könnte nun so aussehen:

```
10 INPUT G
20 IF G < 150 THEN GOSUB 50
50: Wenn Test positiv
   Sprung in Zeile 50
30 PRINT G
40 END: Nötig, weil weitere
   Zeilen folgen
50 REM UNTERPROGRAMM
60 G=G+7.5
70 PRINT "ZUZUEGLVERSAND
   7,50DM"
80 RETURN: Sprung zu-
   rück, Programm fährt mit
   Zeile 30 fort.
Ein Programmablaufplan
nach DIN 66001 sähe dann
etwa wie folgt aus:
```



2. Zweiseitige Entscheidung
Den Unterschied zwischen ein- und zweiseitiger Entscheidung im Ablauf eines Basic-Programmes machen wir uns anhand eines kleinen Mathematik-Listings deutlich, das sich in diesem Umfang nicht besonders nützlich ausnimmt, andererseits aber für unsere Zwecke gute Dienste leistet. Es geht um folgende Problematik: wir wollen nach Eingabe eines Kürzels entweder Quadratwurzel (W) oder Quadrat (Q) einer Zahl abfragen können

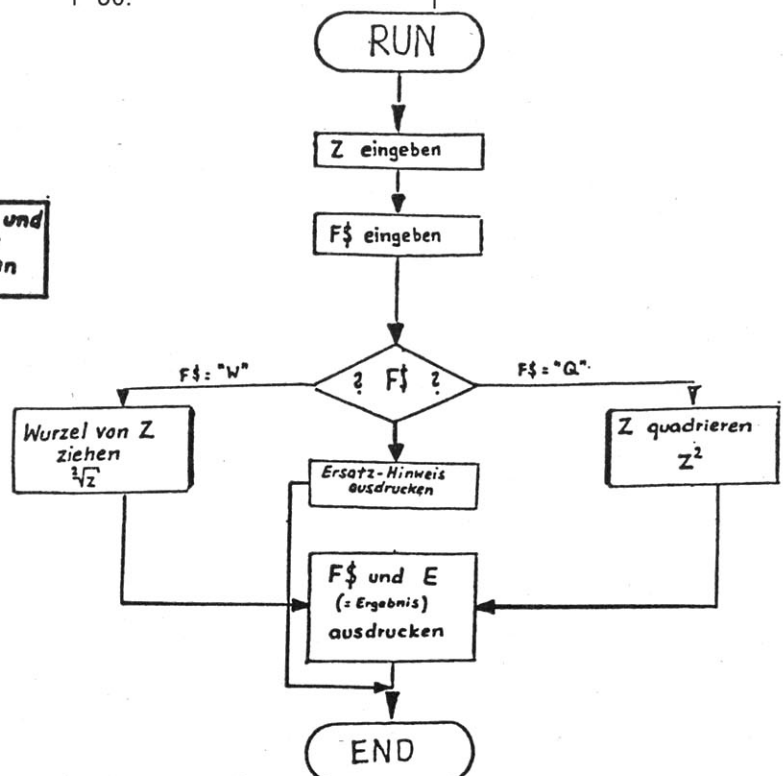
```
10 INPUT "ZAHL";Z:INPUT
  "FUNKTION";F$
20 IF F$="W" THEN
  E=SQR(Z):GOTO 60
30 IF F$="Q" THEN E=Z
```

```
Z:GOTO 60
40 PRINT "FUNKTION NICHT
  VORGESEHEN"
50 GOTO 80
60 PRINT "GEWUENSCHTE
  FUNKTION";F$
70 PRINT "ERGEBNIS";E
80 END
```

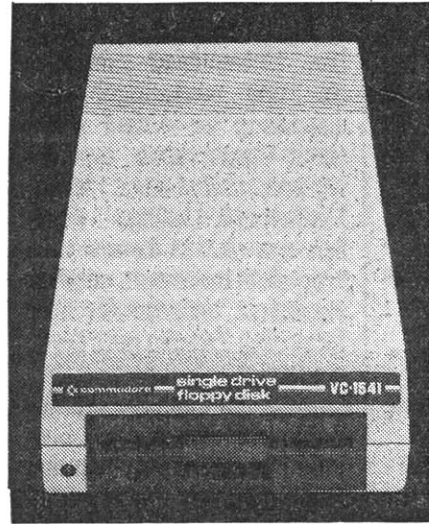
Das Listing birgt so gut wie keine Schwierigkeiten. In Zeile 10 werden Ausgangswert (ZAHL) und gewünschte Berechnung (FUNKTION) erfragt. Da die zweite Eingabe-Anweisung Buchstaben anfordert, muß eine String-Variable folgen. Zeile 20 und 30 bilden den Kern unseres Programmes, da hier die zweiseitige Entscheidung, um die es uns geht, getroffen wird: entweder erhält der Rechner ein Q, was er als Quadratur-Befehl auffaßt – oder F\$ wird W, worauf der Computer die (Quadrat-) Wurzel des jeweiligen Ausgangswertes berechnet. Ist F\$ weder Q noch W, so kann unser 64er damit nichts anfangen: er sagt uns, daß die angegebene Funktion nicht vorgesehen ist. Damit dieser Hinweis jedoch nicht in jedem Fall erfolgt, stehen die GOTO's in den Zeilen 20 und 30.

Zeile 60 gibt noch einmal (zur Kontrolle) die erhaltene Rechenanweisung aus, Zeile 70 bringt das Ergebnis. Die Inhalte der Zeilen 50 und 80 könnten sicher auch vertauscht sein (50:END/80:GOTO 50), aus grundsätzlichen Erwägungen sollte das END jedoch, soweit möglich, am Schluß eines Programmes stehen. Zweiseitig ist die Entscheidung innerhalb unseres kleinen Anwender-Programmes, will sowohl für die eine als auch die andere Situation eine besondere Handlung vorgeschrieben wird. Konkret: der Rechner prüft eine Bedingung (ist F\$ "Q" oder "W"?), und in jedem Fall folgt eine besondere Handlung (einmal Quadrieren, das andere Mal Radizieren). Der PAP (Programm-Ablaufplan) sieht somit wie folgt aus.

(sk)



Die Floppy 1541



Viele Anwender von Heimcomputern schwören inzwischen auf die schnelle Datenverarbeitung mit einem Diskettenlaufwerk. Doch wie funktioniert so ein Ding überhaupt und was spielt sich im Innern einer Diskettenstation ab?

Allgemeines:

Diskettenlaufwerke sind, wie der Name schon sagt, mechanische und elektronische Antriebseinheit für Disketten. Floppylaufwerke können in der Computereinheit integriert sein, oder als externes Gerät angeschlossen werden. Da mehrere Diskettenstationen hintereinander geschaltet werden können, erhöht sich damit zwangsläufig der verfügbare Speicherplatz.

Disketten sind magnetisch beschichtete Folienscheiben, etwa in der Größe einer Single-Schallplatte. Der Vorteil liegt im direkten Datenzugriff. In der Schreibdichte können Disketten durchaus verschieden sein, es gibt Single-Density und Double-Density Disketten (einfache + doppelte Dichte)

Diskettenspeicherung basiert ähnlich wie die Bandaufzeichnung auf magnetischen und unmagnetischen Feldern. Ist ein solches Feld einmal magnetisiert worden, so wird dieses Stadium solange aufrecht erhalten, bis es mit einem Elektromagneten zerstört wird. Nun kann man diese Magnetisiervorgänge auch ganz gezielt vornehmen.

Der Magnet wird an eine bestimmte Stelle der Metallscheibe gebracht und löst dort die Magnetisierung aus. Dabei bewegt sich der Magnet vor- und zurück, während die Diskette sich dreht.

Daten lassen sich auf dieser Basis sehr schnell an bestimmte Positionen schreiben, die dann ebenso schnell und gezielt wieder gelesen werden können. Es entsteht ein Datenträger von hoher Qualität, die Daten sind nur durch Entmagnetisieren zu löschen.

Der Unterschied zur Bandaufzeichnung liegt nicht unbedingt in der Geschwindigkeit des Speichers und Einladens

von Daten, da einige Tapes durchaus schneller sind als Diskettenlaufwerke.

Vorteil ist der wahlfreie Datenzugriff, d.h. Datenfelder können ganz gezielt angesprochen werden.

Formatieren:

Auf einer unbeschriebenen leeren Diskette sind keine Schreibfelder (Tracks) vorhanden, Daten können nicht eingeschrieben werden.

Zum Beschreiben einer Diskette muß diese erst "formatiert" werden, d.h. für die Daten muß Platz geschaffen werden.

Sinn und Zweck des Formatierens ist es, die Diskette magnetisch mit einer Serie von Bereichen (Sektoren) auf-

zubereiten. Dabei entspricht die Länge eines Sektors ca. 1cm.

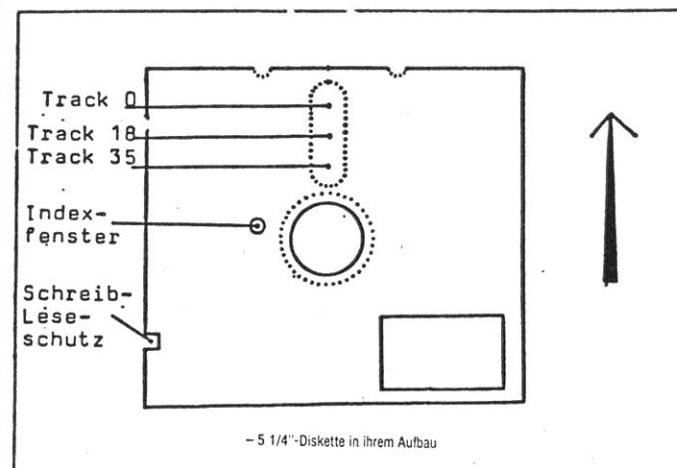
Diese Sektoren werden in Ringen zusammengefaßt, die sich kreisförmig um den Mittelpunkt der Diskette anordnen (ähnlich wie bei einer Langspielplatte).

Dabei laufen die Nummern der Tracks von der Mitte der Diskette beginnend (Track Nr. 1), dem kürzesten Track, bis zum äußeren Rand der Diskette mit dem Track 35. Dabei ist Track Nr.1 der kürzeste, der auch die wenigsten Sektoren hat, während der längste Track mit den meisten Sektoren Nr. 35 ist.

Dieser Vorgang hilft der Steuerlogik des Laufwerks, Daten auf der Diskette wiederzufinden.

In einem Sektor stehen 256 Bytes zur Verfügung, der Track 18 wird für das Directory benötigt (Inhaltsverzeichnis).

Der letzte Sektor des Directory wird dadurch festgesetzt, daß die beiden Bytes, die am Anfang jedes Sektors stehen und auf den nächsten korrespondierenden Sektor hinweisen, auf 0 (Null) gesetzt werden. Das bedeutet, daß der Hinweis den das DOS (Disk Operating System)



beim Lesen der beiden Bytes findet, auf die Spur 0 und den Sektor 255 zeigt. Diese existieren jedoch nicht.

Ist jedoch das gesamte Byte 0, also keines der 8 bits gesetzt, so ist der bezeichnete Sektor bereits belegt.

Das BAM (Block Allocation Map) nimmt diese Speicherdaten auf, und prüft, ob auf dem Sektor bereits etwas steht. Das BAM liegt in Track 0 von Sektor 18 und ist 140 Bytes lang, die sich in 35 à 4 Bytes aufteilen.

Aus diesen Bytes können folgende Informationen entnommen werden:

1. Byte: Nummer der Sektoren aus den 35 Tracks

2. Byte: hält den Status (beschrieben oder unbeschrieben) der Sektoren 0-7 des korrespondierenden Tracks.

3. Byte: hält den Status der Sektoren 8-16 des korrespondierenden Tracks.

4. Byte: hält den Status der Sektoren 17-23 des korrespondierenden Tracks

Es ist noch zu bemerken, daß die BAM für jeden Track 24 Sektoren (0-23) rechnet, ob-

wohl es maximal 21 Sektoren auf dem längsten Track der Außenseite von der Diskette gibt und diese zur Mitte immer weniger werden.

Die BAM übergeht diese potentielle Schwierigkeit dadurch, daß sie nicht vorhandene Sektoren als beschrieben anerkennt.

Speichern:

Ist die Diskette in ihrer Struktur aufgebaut und das Directory initialisiert, so ist die Diskette zur Datenspeicherung bereit. Diese Speichereinheiten nennt man "Files".

Die meistverwendeten Files sind der

1. Programmfile – **Programmsave**

2. sequentieller File – geöffnete File für die Speicherung von **Daten**.

Beide werden auf die gleiche Art gespeichert.

a.) Wird Save vom Benutzer angewählt, so eröffnet der Controller einen Programmfile mit dem ihm angewiesenen Namen.

b.) Das DOS prüft im

Directory, ob bereits ein File mit gleichem Namen existiert.

c.) Angenommen, es existiert kein gleicher Name, so schreibt das DOS den Filenamen in das Directory. Als Starttrack und Sektor werden 0 und 255 reserviert, die nicht existieren.

d.) Danach sucht das DOS den nächsten freien Sektor im BAM, welches sich zu jeder Zeit im Datenbuffer des Floppy-Controllers befindet. Ist ein freier Sektor gefunden, so wird er im BAM markiert.

e.) Anschließend fährt die Floppy den Lesekopf zur Position des freien Sektors und holt sich 254 Bytes des abzuspeichernden Programms, aus dem Arbeitsspeicher des Computers.

f.) Nun sucht das DOS in Vierschritten den Sektor im BAM, der dem vorher lokalisierten Sektor am nächsten steht.

g.) Die Adresse des zuletzt festgestellten Sektors wird nun in die ersten 2 Bytes des zuerst gefundenen Sektors eingetragen.

h.) Jetzt werden die 254 Programmbytes in den Sektor geschrieben. Der in Punkt f gefundene Sektor wird nun anstelle des zuerst abgelegten Sektors in das BAM geschrieben. Dieser Vorgang wiederholt sich solange, bis das Programm komplett auf die Sektoren verteilt ist.

i.) Nach Beendigung dieses Vorgangs wird im letzten Sektor die Suchadresse auf 0,255 gesetzt. Dies zeigt dem Floppy-Controller an, daß das Programm hier abschließt.

j.) Anschließend werden die beiden ersten Track- und Sektorenzeiger, die immer noch im BAM stehen, gegen die 0,255 im Directory ausgetauscht.

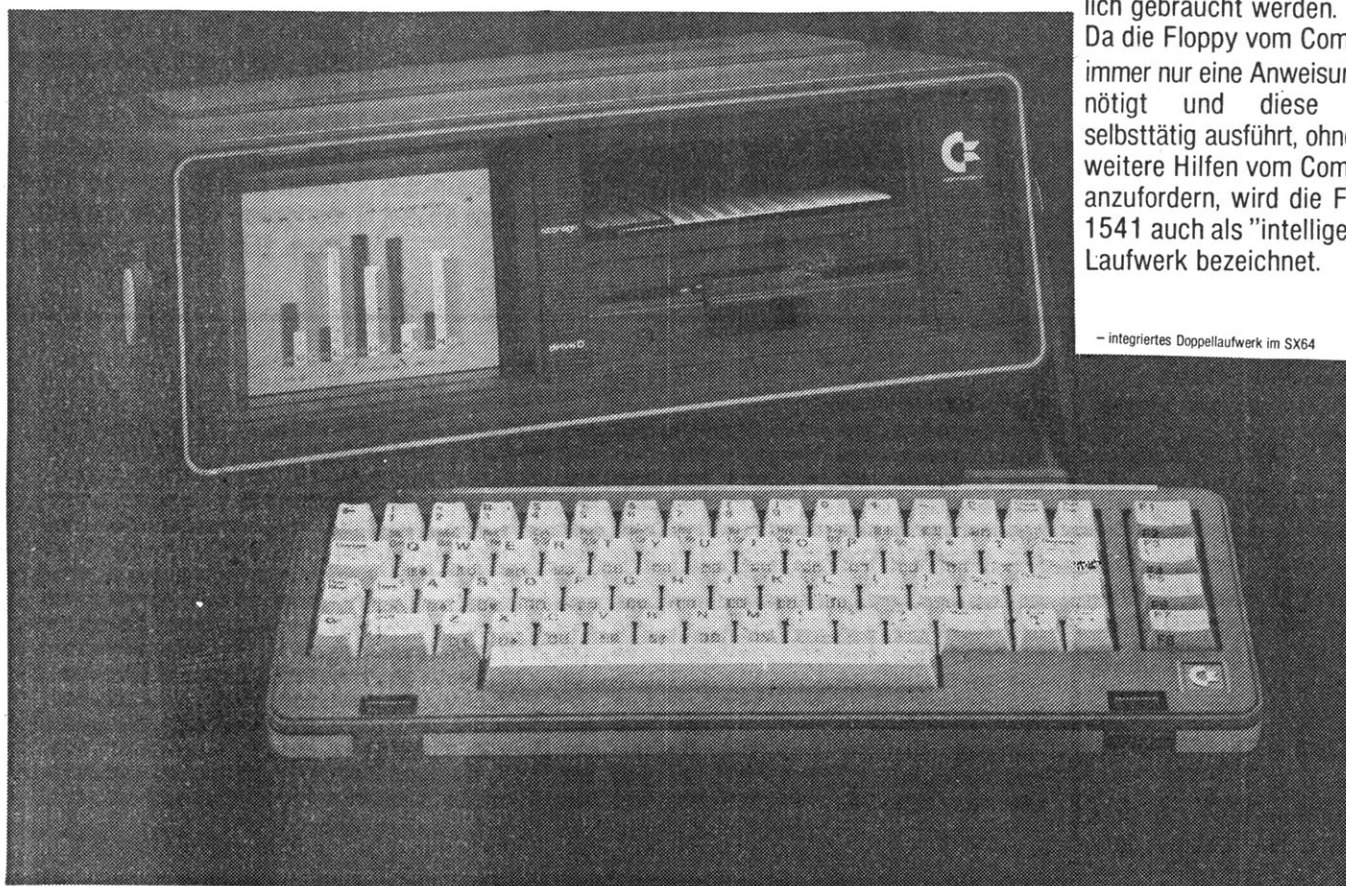
k.) Nun liest das DOS noch einmal das Directory und aktualisiert die Zahl der freien Blocks auf der Diskette.

Die Floppy 1541 läuft mit einem 6502 Mikroprozessor und wird durch eigenes DOS gesteuert. Dabei geht kaum Speicherplatz verloren, da Teile des DOS nur dann geladen werden, wenn sie wirklich gebraucht werden.

Da die Floppy vom Computer immer nur eine Anweisung benötigt und diese dann selbsttätig ausführt, ohne sich weitere Hilfen vom Computer anzufordern, wird die Floppy 1541 auch als "intelligentes" Laufwerk bezeichnet.

(sr)

— integriertes Doppellaufwerk im SX64



Assemblerkurs

Teil 4

Wir hoffen, daß jeder der an diesem Kurs teilnimmt, den Editor und den Assembler eingetippt hat. Diese Programme werden wir während unseres gesamten Kurses verwenden. Sie sind unverzichtbar.

Nun zum 1. Assemblerprogramm. Wir laden unseren Editor und tippen folgendes Programm ein:

```
1  ORG 828
2  LDA #65
3  STA 1024
4  LDA #3
5  STA 55296
6  RTS
```

Danach save'n wir das Programm ab und laden den Assembler. Dieser übersetzt das Assemblerprogramm in Maschinensprache. Mit **SYS 828** können Sie das Programm starten.

Was ist passiert?

In der linken oberen Ecke erscheint ein weißes Pik-Zeichen.

Wie funktioniert das?

Nun wollen wir uns das Assemblerprogramm einmal ansehen:

Zeile 1 ORG 828

Diese Anweisung ist kein Maschinenbefehl, sondern eine Anweisung an den Assembler, das erzeugte MC-Programm ab Adresse 828 abzulegen.

Zeile 2 LDA #65

Dieser Befehl LDA ist eine Abkürzung für "Lade Akkumulator (A-Register)"; das

Doppelkreuz läßt erkennen, daß der folgende Wert direkt geladen werden soll. Würde das Doppelkreuz nicht stehen, so würde die nachfolgende Zahl als Adresse angesehen werden und der Inhalt dieser Adresse danach im Akku stehen. So aber steht im Akku 65.

In Basic würde man sagen

Let A = 65 oder einfach **A = 65**

Zeile 3 STA 1024

Beim C-64 beginnt der Bildschirmspeicher für Zeichen bei 1024 und endet bei 2023. Dies sind genau 1000 Bytes. Das ist auch einfach zu erklären, nämlich 40 Buchstaben x 25 Zeilen. Zurück zu Assembler:

Was bedeutet STA? STA bedeutet STORE Akku. Das heißt, der Inhalt des Akkus wird in die darauffolgende Adresse abgespeichert.

Sehen wir im Handbuch Seite 134, unter Poke 65 Satz 1 nach, so finden wir das Pik-Zeichen.

Aha – daher. Nun gut – weiter.

Zeile 4 LDA #3

Diesen Befehl kennen wir schon. In den Akku wird eine 3 geladen.

Zeile 5 STA 55296

Diesen Befehl kennen wir ebenfalls. Der Inhalt des Akkus (zur Zeit 3) wird in dem Byte mit der Adresse 55296 gespeichert.

Sie ist verantwortlich für die Farbe des 1. Bytes im Bildschirmspeicher.

Zeile 6 RTS

Diese Zeile ist unverzichtbar. Sie bedeutet Return from Subroutine. Diese Zeile muß bei jeder Unteroutine in 65XX-Assembler am Schluß stehen, sodaß der Rechner weiß, jetzt muß ich zurück zu Basic. Andernfalls wäre es recht wahrscheinlich, daß der Rechner in einen unbestimmten Zustand verfällt (abstürzt) und nur noch ein Ausschalten hilft.

Die Abspeicherung der einzelnen Werte in diese Adresse nennt man in unserem Falle übrigens – **direkte Adressierung**. Dies deshalb, weil wir

```
10 PRINT"Q";
100 DATABRK,ORA,???,???,???,ORA,ASL,???,PHP,ORA,ASL,???,???,ORA,ASL,???,
102 DATABPL,ORA,???,???,???,ORA,ASL,???,CLC,ORA,???,???,???,ORA,ASL,???,
104 DATAJSR,AND,???,???,BIT,AND,ROL,???,PLP,AND,ROL,???,BIT,AND,ROL,???,
106 DATABMI,AND,???,???,???,AND,ROL,???,SEC,AND,???,???,???,AND,ROL,???,
108 DATARTI,EOR,???,???,???,EOR,LSR,???,PHA,EOR,LSR,???,JMP,EOR,LSR,???,
110 DATABVC,EOR,???,???,???,EOR,LSR,???,CLI,EOR,???,???,???,EOR,LSR,???,
112 DATARTS,ADC,???,???,???,ADC,ROR,???,PLA,ADC,ROR,???,JMP,ADC,ROR,???,
114 DATABVS,ADC,???,???,???,ADC,ROR,???,SEI,ADC,???,???,???,ADC,ROR,???,
116 DATA???,STA,???,???,STY,STA,STX,???,DEY,???,TXA,???,STY,STA,STX,???,
118 DATABCC,STA,???,???,STY,STA,STX,???,TYA,STA,TXS,???,???,STA,???,???,
120 DATALOY,LOA,LOX,???,LOY,LOA,LOX,???,TAY,LOA,TAX,???,LOY,LOA,LOX,???,
122 DATABCS,LOA,???,???,LOY,LOA,LOX,???,CLV,LOA,FSX,???,LOY,LOA,LOX,???,
124 DATACPY,CMP,???,???,CPY,CMP,DEC,???,INY,CMP,DEX,???,CPY,CMP,DEC,???,
126 DATABNE,CMP,???,???,???,CMP,DEC,???,CLD,CMP,???,???,???,CMP,DEC,???,
128 DATACPX,SBC,???,???,CPX,SBC,INC,???,INX,SBC,NOP,???,CPX,SBC,INC,???,
130 DATABEQ,SBC,???,???,???,SBC,INC,???,SED,SBC,???,???,???,SBC,INC,???,
```

direkt die Adresse angegeben haben.

Hier noch als letztes Hilfsprogramm ein Disassembler.

Wer in Maschinensprache programmiert – früher oder später beginnen die meisten Microcomputer-Freunde damit – der muß überprüfen können, was er programmiert hat.

Dazu verwendet man einen Disassembler. Der C-64 hat ja nur die Möglichkeit vorgesehen, mit PEEK, POKE, USR und SYS direkt auf seine Bytes zuzugreifen und an vor-

her bestimmte Speicherstellen zu springen, um Maschinenprogramme aufzurufen.

Der hier vorgestellte Disassembler ist zwar in der Speicherausnutzung etwas verschwenderisch angelegt, ist aber dafür als Basicprogramm sehr schnell.

Nach dem Eintasten des Programms wird eine Dezimalzahl oder ein \$-Zeichen, gefolgt von einer Hexzahl, eingegeben. Danach folgt ein < L> , welches für List steht.

Nach Druck auf die Taste RETURN wird der C-64 eine Bildschirmseite mit einem Assemblerlisting drucken.

In der nächsten Ausgabe zeigen wir Ihnen, wie man den Bildschirm löscht, beim Programmieren Bytes sparen kann, um Geschwindigkeit und Speicherausnutzung zu erhöhen, und wie man Ramroutinen benutzen kann.

★ ★ ★ ★ ★ ★ ★ ★

```

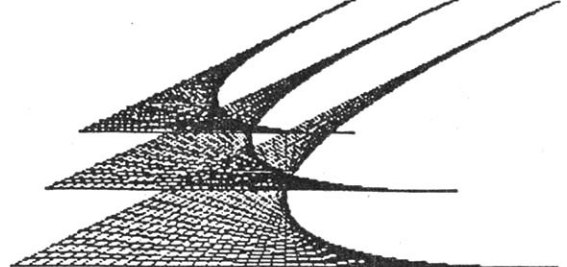
200 DATA1,2,1,1,1,2,2,1,1,2,1,1,1,3,3,1,2,2,1,1,1,2,2,1,1,3,1,1,1,3,3,1
202 DATA3,2,1,1,2,2,2,1,1,2,1,1,3,3,3,1,2,2,1,1,1,2,2,1,1,3,1,1,1,3,3,1
204 DATA1,2,1,1,1,2,2,1,1,2,1,1,3,3,3,1,2,2,1,1,1,2,2,1,1,3,1,1,1,3,3,1
206 DATA1,2,1,1,1,2,2,1,1,2,1,1,3,3,3,1,2,2,1,1,1,2,2,1,1,3,1,1,1,3,3,1
208 DATA1,2,1,1,2,2,2,1,1,1,1,1,3,3,3,1,2,2,1,1,2,2,2,1,1,3,1,1,1,3,1,1
210 DATA2,2,2,1,2,2,2,1,1,2,1,1,3,3,3,1,2,2,1,1,2,2,2,1,1,3,1,1,3,3,3,1
212 DATA2,2,1,1,2,2,2,1,1,2,1,1,3,3,3,1,2,2,1,1,1,2,2,1,1,3,1,1,1,3,3,1
214 DATA2,2,1,1,2,2,2,1,1,2,1,1,3,3,3,1,2,2,1,1,1,2,2,1,1,3,1,1,1,3,3,1
300 DATA0,10,0,0,0,4,4,0,0,2,1,0,0,3,3,0
302 DATA9,11,0,0,0,5,5,0,0,8,0,0,0,7,7,0
304 DATA3,10,0,0,4,4,4,0,0,2,1,0,3,3,3,0
306 DATA3,11,0,0,0,5,5,0,0,8,0,0,0,7,7,0
308 DATA0,10,0,0,0,4,4,0,0,2,1,0,3,3,3,0
310 DATA9,11,0,0,0,5,5,0,0,8,0,0,0,7,7,0
312 DATA0,10,0,0,0,4,4,0,0,2,1,0,12,3,3,0
314 DATA9,11,0,0,0,5,5,0,0,8,0,0,0,7,7,0
316 DATA0,10,0,0,4,4,4,0,0,0,0,0,3,3,3,0
318 DATA9,11,0,0,5,5,6,0,0,8,0,0,0,7,0,0
320 DATA2,10,2,0,4,4,4,0,0,2,0,0,3,3,3,0
322 DATA9,11,0,0,5,5,6,0,0,8,0,0,7,7,8,0
324 DATA2,10,0,0,4,4,4,0,0,2,0,0,3,3,3,0
326 DATA9,11,0,0,0,5,5,0,0,8,0,0,0,7,7,0
328 DATA2,10,0,0,4,4,4,0,0,2,0,0,3,3,3,0
330 DATA9,11,0,0,0,5,5,0,0,8,0,0,0,7,7,0
400 DATA,,,#,$,$,$,$,"X",$,"Y",$,"X",$,"Y",$,,$,$,"X"),($,"Y"),($,),
500 DIMM$(255),L$(255),F$(255),F1$(12),F2$(12)
510 FORI=0TO255:READM$(I):NEXT:FORI=0TO255:READL$(I):NEXT:FORI=0TO255:READF$(I):
NEXT
515 FORI=0TO12:READF1$(I),F2$(I):NEXT
530 PC=0
600 INPUT"☐";A$:IFA$="X"THENEND
605 IFA$="L"THEN1000
610 IFRIGHT$(A$,1)="L"THENGOSUB800:GOTO1000
630 GOTO600
800 ILEFT$(A$,1)="$"THEN810
805 PC=VAL(A$):RETURN

```



GÖRLITZ COMPUTERBAU

Ihr Commodore 64 kann viel mehr



wenn Sie ihn mit unserem Görlitz-VC-EPSON-INTERFACE an einen Epson-Drucker anschließen.

Unser VC-EPSON-INTERFACE Best.-Nr. 8422 bringt alle CBM-Grafikzeichen in vierzig verschiedenen Schriftarten und -breiten * **HARDCOPY** mit SIMON's BASIC sauber und hochauflösend * hat einen eingebauten Selbsttest * einen 2K Pufferspeicher * einen eigenen Z80-Mikroprozessor * zwei IEC-Buchsen (wie Floppy) * IEC-Adresse einstellbar * keine Modifikation des Druckers * definierbarer Zeichensatz (FX) bleibt frei * ein Interface für C-64, VC-20 an alle Epson-Drucker * Lieferung komplett mit DIN-Verbindungskabel und ausführlicher Bedienungsanleitung.

EPSON RX-80	mit VC-Interface Best.-Nr. 8422 DM 1330,-
EPSON RX-80 F/T	mit VC-Interface Best.-Nr. 8422 DM 1502,-
EPSON FX-80	mit VC-Interface Best.-Nr. 8422 DM 1889,-
EPSON FX-100	mit VC-Interface Best.-Nr. 8422 DM 2362,-

Preise inkl. MwSt., Katalog 1/84 kostenlos.

Besuchen Sie unsere Ausstellung in der Rhein-Mosel-Halle in Koblenz vom 29. Okt. bis 31. Okt. 84. Wir freuen uns auf Ihren Besuch!

GÖRLITZ COMPUTERBAU-Postf. 852 - 5400 Koblenz - Tel. 0261-27500

```

810 PC=0:FOR I=2TO LEN(A$)-1
820 A=ASC(MID$(A$,I,1))-48:IFA>9THENA=A-7
830 PC=PC*16+A:NEXT:RETURN
900 FOR J=3TO 0STEP -1:A=INT(DZ/16+J):DZ=DZ-A*16+J:A=A+48:IFA>57THENA=A+7
910 PRINTCHR$(A):NEXT:RETURN
950 FOR J=1TO 0STEP -1:A=INT(DZ/16+J):DZ=DZ-A*16+J:A=A+48:IFA>57THENA=A+7
960 PRINTCHR$(A):NEXT:RETURN
1000 FOR I=1TO 11:DZ=PC:PRINT" ";:GOSUB900:PRINT"- ";:P1=PEEK(PC)
1005 ONL%(P1)GOSUB1100,1200,1300
1010 NEXT:GOTO600
1100 DZ=P1:GOSUB950:PRINT" ";TAB(10);M$(P1):PC=PC+1:RETURN
1200 DZ=P1:GOSUB950:PRINT" ";:P2=PEEK(PC+1):DZ=P2:GOSUB950
1205 PRINT" ";TAB(10);M$(P1);
1210 IFF%(P1)=9THENP2=PEEK(PC+1):IFP2<128THENP2=PC+P2+2:GOTO1220
1215 IFF%(P1)=9THENP2=PC-(255-P2)+1
1220 PRINT" ";F1$(F%(P1)):DZ=P2:IFF%(P1)=9THENGOSUB900:GOTO1240
1230 GOSUB950
1240 PRINTF2$(F%(P1)):PC=PC+2:RETURN
1300 DZ=P1:GOSUB950:PRINT" ";:P2=PEEK(PC+1):DZ=P2:GOSUB950:PRINT" ";
1310 P3=PEEK(PC+2):DZ=P3:GOSUB950:PRINT" ";TAB(10);M$(P1);
1320 PRINT" ";F1$(F%(P1)):DZ=P3*256+P2:GOSUB900:PRINTF2$(F%(P1)):PC=PC+3:RETURN
READY.

```

FUNKTIONIERT OHNE FRUST: KOMFORTABLE SOFTWARE VON DYNAMICS.

Weil es nicht nur Computer-Freaks gibt, bietet Dynamics anwenderfreundliche Software – komfortabel, menuegesteuert über den Bildschirm.

Wer also auf seinem Commodore C64, dem TI99/4A oder den Atari 400/800- oder XL-Modellen nicht lange programmieren will oder null Bock auf Listings hat, der geht ins Fachgeschäft oder Kaufhaus.

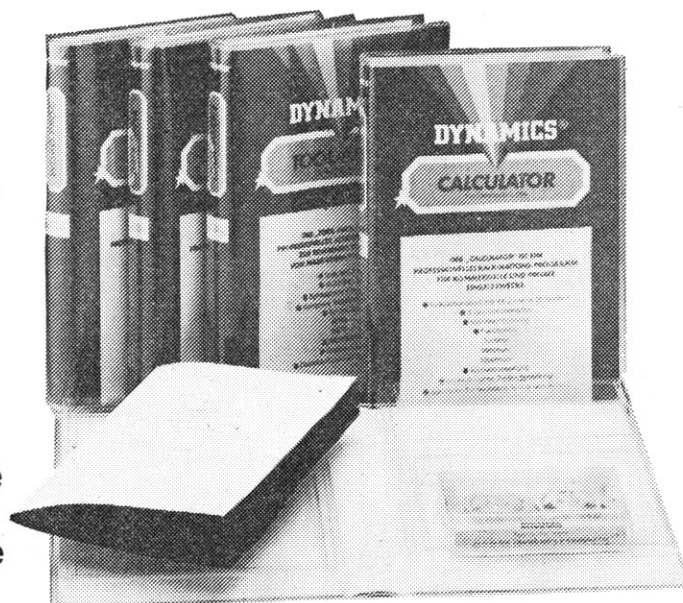
Dort gibt es das Dynamics „Adressen-Archiv“, das „Video-“, „Bücher-“ und „Schallplatten-“

Archiv“. Programme, die Ordnung halten – einfach und flink. Da gibt es neu den „Vokabel-Trainer“, der Lernen zum Vergnügen macht. Fragen Sie danach!

Wer mehr will, sollte sich die Dynamics Arbeits-Programme zeigen lassen. Aber auch „Word-Proc“, „Tool Pack I und II“, „Calculator“ oder „Data-Bank“ bleiben immer anwenderfreundlich.

Gut verständliche Bedienungsanleitungen (keine Wälzer!), logischer Aufbau und einfache Anwendung garantieren, daß Computern mit Dynamics Spaß macht!

Also: auf ins Fachgeschäft, hin zum Kaufhaus. Weil Dynamics Software einfach komfortabel ist. Ausprobieren!



Dynamics-Software gibt es auf Computer-Cassette oder Diskette.

DYNAMICS®

COMPUTER-SOFTWARE UND
COMPUTER-ZUBEHÖR.

FÜR ATARI 400/800 UND
600 XL/800 XL,
TEXAS INSTRUMENTS TI 99/4A
UND COMMODORE C 64.

Dynamics Marketing GmbH,
Große Bäckerstraße 11,
2000 Hamburg 1.

Leserbriefe

Wie kann ich beim VC-20 den Text nach unten scrollen lassen? Gibt es dafür Zusatzprogramme oder einfache Routine? Da ich ein Programm schreibe, in dem Text nach unten scrollen soll, wäre ich für baldige Antwort sehr dankbar.

P. Braune, Ettlingen

Redaktion:

Das folgende kleine Basic-Programm veranlaßt einen Down-Scroll beim VC-20.

```
10 A$=" ": For T=0 to 18: A$ + A$ + Chr$(29): NEXT
15 A$ = CHR$(19) + A$ + Chr$(160) + Chr$(157) + Chr$(157)
20 A$ = A$ + Chr$(157) + Chr$(157) + Chr$(148) + Chr$(148) + Chr$(148)
1000 REM Scroll nach unten
1005 Print A$, POKE 217,158: POKE 218,158
```

Zunächst einmal herzlichen Glückwunsch für Ihre gelungene Zeitschrift. Da ich noch nicht lange mit Computern arbeite, (ich habe einen VC-20) bin ich doch teilweise im Nachteil. In Ihrer Zeitschrift dagegen wird auch für Einsteiger und Anfänger einiges geboten. Super sind übrigens sämtliche Listings! Nun zu meinem Problem: Ich bin auf der Suche nach einem Sprachsynthesizer für den VC-20, entweder als Modul oder Software. Bisher konnte ich leider keine Bezugsquelle finden, vielleicht können Sie weiterhelfen.

G. Schneyderman, Bonn

Redaktion:

Wir geben diese Frage gern an unsere Leser weiter und wären für kurzen Erfahrungsbericht dankbar.

Wie kommt es, daß ich große Probleme beim Laden von Programmen habe?

Ich arbeite mit einem C-64 und einer Datasette. Die größten Probleme habe ich bei gekaufter Software. Bereits 2 Kassetten mußte ich umtauschen. Meine Datasette ist in Ordnung, da meine abgespeicherten Programme problemlos wieder eingeladen werden.

T. Kiebler, Kassel

Redaktion:

Ihre Probleme sind mit einer Tonkopfstückung zu lösen. Drücken Sie dazu die Play-Taste Ihrer Datasette. Unter dem Einschubfach für die Kassetten wird nun die Tonkopfschraube sichtbar. Mit einem kleinen Schraubenzieher müssen Sie die Schraube verstellen und nun einfach ausprobieren. Haben Sie die richtige Einstellung gefunden, so überziehen Sie die kleine Schraube mit etwas Nagellack. So kann sich die Schraube nicht mehr verstellen und diese Art von Problemen gibt es für Sie nicht mehr.

Ich besitze einen Commodore 64 und habe mir kürzlich eine Diskettenstation zugelegt. Diese arbeitet auch einwandfrei, doch ist in letzter Zeit häufig ein Klappern und Rattern zu hören. Ist dies normal oder sollte ich das Gerät zur Reparatur bringen?

K. Steinbach, Flensburg

Redaktion:

Sie haben eine völlig intakte Floppy. Das Rattern kommt durch den Schreib-/Lesekopf, der zunächst einmal das Directory anfährt. Treten hierbei Lesefehler auf, so fährt der Schreib-/Lesekopf erneut zum Anfang, dort wird dann das Geräusch verursacht.

NEU FÜR DEN COMMODORE C-64



**Die kaufmännisch
orientierte Software-Serie
für kleinere Betriebe.**

SM SOFTWARE AG / MÜNCHEN

V P C	Völzke Computer Peripherie	V P C
		
Eprom-Programmer V128 für C 20, C 64 u. SX-64 für Eproms 2508/16/32 u. 2758/16/32/64/128. Professionelle Ausführung m. komfortabler Treiber-Software auf Kas- sette: DM 249,- Neu: Eprom-Programmer V128-G im Pult-Gehäuse DM 349,- Uniment-C 64-Befehlserweiterung: über 50 zus. Befehle u. Funktionen für Assembler, Centronics-Druckanschl., Graphik-, Sprite-, Sound- und Disketten-Anwendung; mit Beispielprogrammen u. ausführlicher Bedienungsanlei- tung DM 99,- Diskette zzgl. DM 7,- UNIMENT-Steckmodul DM 199,- Weiteres aus unserem Programm: — Eprom-Karten und -Löschgerät — 80-Zeichenkarten Hagen Völzke, Ahornallee 4, 8023 Pullach Versandhandel Tel. 089/7934534		
Info gegen Rückporto		

Liebe Rätselfreunde von Compute mit,

nach der kleinen Abwechslung mit dem Silbenrätsel der letzten Woche, machen wir in dieser Ausgabe wieder das beliebte Suchrätsel. Unser Lösungswort besteht diesmal aus 16 Buchstaben und ist in jedem Computer zu finden.

Da es diesmal wirklich nicht sehr schwierig ist, werden wir auch keine weiteren Hilfen geben.

Wir wünschen auf jeden Fall viel Glück bei der Suche im Buchstabensalat! Im übrigen werden wir ab und zu veränderte Rätsel bringen, so wie z.B. das Silbenrätsel, um auch diese Rubrik abwechslungsreich zu gestalten.

1. CPU (14)
2. Elektronische Datenverarbeitung (Abk.) (3)
3. Schnittstelle (9)
4. Kfm. Programmiersprache (5)
5. Gegenteil von Software (8)
6. Lösch- und programmierbarer Festwertspeicher (5)
7. Zusammenschaltung mehrerer Computer (8)
8. Computerzeichnung (6)
9. Simulation eines Mikroprozessors (9)
10. Anderes Wort für ROM (15)
11. Zahl 14 in Hexadezimal (1)
12. Schreib/Lesespeicher (3)
13. Telefonmodem (14)
14. Keyboard (8)
15. Andere Bezeichnung für Maschinencode (10)
16. Befehl, der den Rechner im Ausgangszustand versetzt (5)

D F L V R E N I H T I E R N E C E M E S I
H A E E D O C T K E J B O H O H A V M V S
O L E R C E N E I V D V N C I I N E D A C
R A S T R A R H L Z H B D A N T E E S H E
A R T N S E F A A O G N E N P D F T I G H
H M R E E I D R E S E T R N M A G H T R R
A N W E D S I D E D R A E O D M E A I E D
T K V A B A L W S T E R R P V N D D I L F
L P E L S G M A R E N P E F D I A M A P E
O E I Z E N T R A L E I N H E I T O E P K
Z S H V A E E E I R I D W L F E N O S O I
V V O L H T A S T A T U R L T I S F S K S
I A S I O Z Z T S V E L O E K T A B R K B
H D S F H W U T I E O B D I E C C M S I E
L A O Z V E E D O S O E S N I K S H I T L
K I H P A R G E W C O C R S E N A E A S U
D N O A F K M R H E I N H M F L L N A U A
O D R F E E N E A N E T N H C I M C D K A
C H T S I E N R C S D R D I H C H I R A L
A E F N R E H C I E P S E S E L R U N G L
S V I R W R E V E R R I N O I T A L U M E

Die Gewinner aus Heft 37/84

1. Preis

N. Stegmaier, 7075 Mutlangen

2. Preis

U. Langer, 5778 Meschede

3. Preis

St. Benas, 8720 Schweinfurt

Hier die Gewinner für je 1 Gutschein:

K.-J. Ziegler, 5400 Koblenz
D. Herr, 6228 Eltville
V. Wolff, 6305 Buseck
W. Twrdy, 7315 Weilheim
K. Hahn, 660 Saarbrücken
G. Feidl, 8000 Mahn
M. Jolowicz, 3320 Salzgitter
H. Hensel, 4018 Langenfeld
L. Corsten, 4000 Düsseldorf 30
J. Ottenweller, 8770 Lohr
Th. Flicker, 6228 Eltville
G. Drieves, 4600 Dortmund
B. Junski, 4000 Düsseldorf
K. Ollech, 2406 Stockelsdorf
J. Henning, 6334 Asslar
W. Pohland, 1000 Berlin
R. Sabellek, 5300 Bonn
G. Stender, 2360 Bad Segeberg
J. Kuthleick, 2400 Lübeck
M. Neu, 6670 St. Ingbert
H. Thiel, 3579 Willingshausen
J. Teske, 2401 Ratekau

**...Und wieder können Sie Gutscheine aus unserem
Softwareservice gewinnen!**

Einsendeschluß ist der 6. Oktober 1984

Ihre Lösung senden Sie an:

Roeske Verlag, CM-Rätsel, Fuldaerstraße 6, 3440 Eschwege

DM 35.00
DM 45.00
DM 35.00
DM 35.00