

Bartel

**DAS
IDEEENBUCH
ZUM
COMMODORE 64**

**REZEPTE
FÜR KREATIVES COMPUTERN**

EIN DATA BECKER BUCH

Bartel

**DAS
IDEEBUCH
ZUM
COMMODORE 64**

REZEPTE

FÜR KREATIVES COMPUTERN

EIN DATA BECKER BUCH

ISBN 3-89011-027-4

Copyright (C) 1984 DATA BECKER GmbH
Merowingerstr. 30
4000 Düsseldorf

Alle Rechte vorbehalten. Kein Teil dieses Buches darf in irgendeiner Form (Druck, Fotokopie oder einem anderen Verfahren) ohne schriftliche Genehmigung der DATA BECKER GmbH reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

Wichtiger Hinweis!

Die in diesem Buch wiedergegebenen Schaltungen, Verfahren und Programme werden ohne Rücksicht auf die Patentlage mitgeteilt. Sie sind ausschließlich für Amateur- und Lehrzwecke bestimmt und dürfen nicht gewerblich genutzt werden.

Alle Schaltungen, technische Angaben und Programme in diesem Buch wurden von den Autoren mit größter Sorgfalt erarbeitet bzw. zusammengestellt und unter Einschaltung wirksamer Kontrollmaßnahmen reproduziert. Trotzdem sind Fehler nicht ganz auszuschließen. DATA BECKER sieht sich deshalb gezwungen, darauf hinzuweisen, daß weder eine Garantie noch die juristische Verantwortung oder irgendeine Haftung für Folgen, die auf fehlerhafte Angaben zurückgehen, übernommen werden kann. Für die Mitteilung eventueller Fehler sind die Autoren jederzeit dankbar.

VORWORT

Was kann man mit dem COMMODORE 64 eigentlich alles machen? Nachdem der Reiz der ersten Spiele verflogen ist, stellen sich immer mehr Anwender diese Frage.

Beantwortet wird sie von Rainer Bartel, der in diesem Buch unzählige Einsatzmöglichkeiten des 64ers beschrieben hat. Die Bandbreite reicht von der Dateiverwaltung über Schaufensterwerbung bis zu Diät- und Strickprogrammen. Einer Anwendung, der Textverarbeitung ist das Buch selbst zu verdanken: Es wurde komplett mit TEXTOMAT auf dem 64er geschrieben und auf dem EPSON LQ 1500 ausgedruckt.

Das angenehmste neben dem reichhaltigen Wissen über die Anwendungsmöglichkeiten des 64ers ist aber, daß das Buch äußerst spannend und interessant geschrieben ist. Und das kann man sonst von Mikrocomputerbüchern nicht immer sagen. Viel Spaß beim Lesen!

Düsseldorf, im Juni 84



(Dr. Becker)

INHALTSVERZEICHNIS

0.	Holen Sie Ihren 64er aus der Ecke	1
1.	So lernt man sich kennen	3
1.1	Das kann der C64 wirklich	7
1.2	Speichern tut not	12
1.3	Auch ein 64er braucht Druck	14
1.4	Wenig Software - wenig Nutzen	17
1.5	Fernseher oder Monitor ?	22
1.6	Nützliche Kleinigkeiten	25
2.	Grundrezepte	28
2.1	Textverarbeitung	30
2.2	Rechnen, Planen, Kalkulieren	39
2.3	Daten speichern und sortieren	47
2.4	Musizieren	56
2.5	Malen, Zeichnen und Entwerfen	58
2.6	Selbst programmieren	61
3.	Texten, Drucken und Versenden	66
3.1	Gedichte vom Computer (mit Programm "Reim")	67
3.2	Einladung zur Party	74
3.2.1	Eine einfache Adressen-Datei	75

	(mit Programm "Adress-Datei")	
3.2.2	Ein ganz einfaches Textprogramm (mit Programm "Simpel-Text")	82
3.3	Diplomarbeit - professionell gestaltet	84
3.4	Individuelle Werbebriefe	93
4.	Tabellen erstellen, Rechnen und Planen	104
4.1	Haushaltsbuch-Tips	105
4.1.1	Und so sieht es in der Praxis aus	110
4.1.2	Haushaltsbuch als Software-Paket	117
4.2	Autokosten im Griff (mit Programm "Autokosten")	120
4.3	Wie hoch wird die monatliche Belastung ? (mit Programmier-Hilfen)	133
4.4	Erster, Zweiter, Dritter (Fallbeispiel: Computer beim Sportfest)	142
4.5	Nebenbei ein Taschenrechner (vier Lösungswege)	153
5.	Daten aller Arten	158
5.1	Rezeptkartei für Hobbyköche	159
5.2	Ordnung in der Tiefkühltruhe	164
5.3	Gartenpflege-Kalender	168
5.4	Lagerliste für Handwerksbetriebe	170
5.5	Ein ganz persönliches Gesundheitsarchiv	173
5.6	Egal, was Sie sammeln - der Computer behält den Überblick	176
5.7	Fotohobby - Erfahrungsdaten sammeln	183
5.8	Jogging - ein Trainingsplaner	186
5.9	FdH - Diätplan elektronisch (mit Programm "EffDeHa")	188

5.10	Ein intelligentes Fremdwörterbuch (mit Programm "Lexikon")	198
5.11	Literaturdatenbank fürs Abitur	207
6.	Musizieren, Malen und Entwerfen	209
6.1	Kleine Notenschule (mit Programm "64 Musik Maschine")	211
6.2	Hausgemachte Hits	218
6.3	CAD für Handarbeit	221
6.4	Von Hamburg nach München (mit 2 Programmen)	226
6.5	Laufschrift und bewegte Bilder (mit Programmier-Beispielen)	238
6.6	Geo-Spiele	242

0. Holen Sie Ihren 64er aus der Ecke

Kennen Sie folgenden Dialog ?

"Warum hast Du Dir eigentlich einen Computer gekauft ?"

"Äh, da kann ich meine Adressen ... und Nachhilfe, außerdem Telespiele; alles solche Sachen."

"Und dafür gibst Du zigtausend Mark aus ? So ein Blödsinn !"

"Aber ohne Computer kann man heute nicht mehr mithalten."

"Na, ich kauf mir jedenfalls so eine Kiste nicht."

So oder ähnlich laufen viele Gespräche zwischen Computer-Besitzern und Computer-Skeptikern ab. Wenn Sie dann in der unangenehmen Lage sind, Ihre Motivation zum Kauf eines C64 rechtfertigen zu müssen, können Sie ganz schön ins Schleudern kommen.

Natürlich, Sie wissen vielleicht ganz gut, wie Sie Ihren 64er einsetzen. Oder ?

Gehören Sie möglicherweise zu den rund 20.000 Schnellentschlossenen, die das Gerät unter den Christbaum gestellt haben ? Oder zu denen, die die Nase voll davon haben, mit dem Computer nur zu spielen ?

Bitte, prüfen Sie sich doch einmal selbst.

```
*****
*
*   Wie möchten Sie Ihren C64 am liebsten einsetzen ? *
*
*   1. Nur zum Spielen                               JA/NEIN *
*   2. Fürs Hobby                                    JA/NEIN *
*   3. Im Haushalt                                   JA/NEIN *
*   4. Für Beruf bzw. Geschäft                       JA/NEIN *
*   5. Um Computer besser zu verstehen              JA/NEIN *
*
*****
```

Wenn Sie bei der Frage 1. mit NEIN geantwortet haben, sind Sie hier richtig; wenn Sie mehr als zweimal das JA angekreuzt haben, erst recht. Sollten Sie jedoch den Computer hauptsächlich als Spielzeug benutzen wollen, dürfen Sie dieses Buch getrost gegen eine Spielkassette umtauschen (wenn Ihr Händler das mitmacht).

Um es ganz deutlich zu machen: auf den folgenden Seiten finden Sie Hinweise auf ernsthafte, nützliche Anwendungen, die teilweise sogar echt Geld sparen helfen. Also, holen Sie Ihren 64er aus der Ecke !!

1. So lernt man sich kennen

Guten Tag, mein Name ist Rainer Bartel. Ich stehe in den Startlöchern, um Ihnen einen großen Stapel Ideen zu bringen. Vorher muß ich Sie aber unbedingt ein wenig kennenlernen; sonst wäre ja keine Chancengleichheit zwischen uns gegeben.

Der Fragebogen dient übrigens auch Ihrer Selbsterkenntnis. Vielleicht haben Sie sich ja noch nie so recht Gedanken darüber gemacht, was Sie mit Ihrem Gerät eigentlich anfangen wollen. Wenn Sie alle Fragebögen bearbeitet haben, sind Sie schlauer.

Also, frisch ans Werk, den Filzstift in die Hand, das Spiel beginnt.

```
*****
*                                     *
*           F R A G E B O G E N           *
*                                     *
*****
```

```
*****
*                                     *
*   Ihre Hardware-Konfiguration ?           *
*                                     *
*                                     *
*   1.  COMODORE 64                       JA/NEIN *
*   2.  Datasette (Datenrekorder)         JA/NEIN *
*   3.  Ein Diskettenlaufwerk VC 1541     JA/NEIN *
*   4.  Eine zweite (3.,usw.) Floppy      JA/NEIN *
*   5.  Monochromer-Monitor (Datensichtgerät)JA/NEIN *
```

- * 6. Farb-Monitor JA/NEIN *
- * 7. Matrix oder Typenrad-Drucker JA/NEIN *
- * 8. Dazu ein Plotter JA/NEIN *
- * *
- *****

Jetzt können Sie schon die erste Auswertung vornehmen. Keine Angst, dafür brauchen Sie Ihren 64er nicht einzuschalten.

KEINMAL JA = Was dieses Buch betrifft: Sie sollten es wirklich von vorne bis hinten durchlesen. Erstens werden Sie dabei eine Menge über die phantastischen Fähigkeiten, aber auch über die Grenzen des COMMODORE 64 erfahren; zweitens finden Sie einen Haufen Anregungen für den zukünftigen Einsatz Ihres Gerätes. Übrigens, den Rest des Fragebogens dürfen Sie überschlagen.

EINMAL JA = Merkwürdig, entweder Sie haben irgendein Zusatzgerät billig erstanden, das jetzt auf den Computer wartet, oder Ihr 64er steht noch mutterseelenallein rum. Meine Empfehlung: erst erweitern, dann weiterlesen.

ZWEI- oder DREIMAL JA = Sie besitzen bereits eine Ausrüstung, mit der man einige interessante Dinge machen kann. Auch für Sie ist in diesem Buch was drin. Mein Tip allerdings: Lesen Sie zuerst die Kapitel 1.2 und folgende, vermutlich werden Sie dann eine Erweiterung Ihrer Anlage ernsthaft ins Auge fassen. Bis Sie das erledigt haben, gilt für Sie: Alle Ideen, die im Inhaltsverzeichnis mit einem "*" gekennzeichnet sind, können Sie schon jetzt ausprobieren.

MEHR ALS DREIMAL JA = Sehr gut, das wichtigste haben Sie beieinander. Beantworten Sie jetzt bitte die Fragen

des zweiten Fragebogenteils. Dann sprechen wir uns wieder. Jedenfalls: dies Buch wird Ihnen weiterhelfen.

```
*****
*
* Ihre Software-Bibliothek ?
*
* 1. Nur Module JA/NEIN *
* 2. Fertig-Programme auf Kassette JA/NEIN *
* 3. Selbstgeschriebene Programme JA/NEIN *
* 4. Textverarbeitungs-Software JA/NEIN *
* 5. Datenverwaltungs-Software JA/NEIN *
* 6. Buchführungs-Software JA/NEIN *
* 7. Fakturierungs-Software o.ä. JA/NEIN *
* 8. Graphik- bzw. Synthesizer-Software JA/NEIN *
*
*****
```

Auswertung: wenn Sie innerhalb diese Blockes gar kein JA erzielt haben, stimmt etwas nicht; entweder Sie besitzen doch noch keinen C64, oder Sie benutzen ihn nur als Ausstellungsstück. Aber, Spaß beiseite ...

EINMAL JA = Nun haben Sie einen Computer und beginnen sich langsam zu fragen, wofür das "Ding" eigentlich gut ist. Der Kauf dieses Buches ist ein Schritt in die richtige Richtung. Lesen Sie doch mal das Kapitel 1.4 !

ZWEIMAL JA = In Ihrer Software-Bibliothek steht ja schon das eine oder andere Fertigprogramm. Sie sollten das ausbauen. Welcher Art Ihre neue Software sein sollte, entnehmen Sie bitte den Kapiteln 2.1 bis 2.5.

DREI oder MEHR JA = Ganz schön professionell, Ihre Software-Auswahl. Vielleicht fehlen Ihnen nur noch ein paar Anregungen für neue Anwendungen. Bitte, bedienen

Sie sich !

Gut, jetzt sind wir ja schon fast per Du, was unser gemeinsames Interessensgebiet angeht. Und: möglicherweise haben Sie sich auch an meinen Sprachstil gewöhnt. Wenn nicht, nun, dann müssen Sie diese Schreibe noch rund 200 Seiten lang ertragen.

1.1 Das kann der C64 wirklich

Sicher wäre es langweilig, an dieser Stelle die technischen Daten des C64 aufzulisten, die entnehmen Sie bitte dem Handbuch. Aber es wird Sie sicher interessieren, was diese Merkmale im täglichen Gebrauch bedeuten. Zu diesem Zweck müssen wir doch einen kurzen Blick ins Innere des Computers tun (* ACHTUNG ! Technik-Experten ! Bitte sofort ans Ende des Kapitels weiterblättern und Anmerkung lesen). Wie der Name schon sagt, umfaßt der Speicher Ihres COMMODORE insgesamt 64 KB (Kilobyte), von denen aber ein ganz schön großer Teil für die interne Arbeit benötigt wird. Sie sehen ja, wenn Sie das Gerät einschalten, daß es sich mit der berühmten Formulierung

```
"64K RAM SYSTEM 38911 BASIC BYTES FREE"
```

meldet. Für BASIC-Programme stehen also rund 39KBytes zu Diensten.

Konstruieren wir mal den rein theoretischen Fall, alle Speicherstellen wären mit jeweils einem Buchstaben besetzt, dann könnten Sie im freien Speicher des C64 rund 15 Seiten Text unterbringen. Zugegeben, nicht viel; aber dafür ist unser Beispiel auch völlig unrealistisch. Denn, erstens könnten Sie mit dem gespeicherten Text überhaupt nichts anfangen, weil kein Platz für ein Programm da ist, das den Text verwaltet und zweitens gibt es ja externe Speichermedien, ohne die der Computer so gut wie nutzlos ist. Darüber später mehr.

Vielleicht werden Sie sich fragen, was denn eigentlich in den ca. 25.000 besetzten Speicherstellen steckt. Nun, dort haben die cleveren Konstrukteure und

Programmierer alles das untergebracht, was die tollen Fähigkeiten des 64er ausmacht.

Da ist zunächst der sogenannte BASIC-Interpreter. Wie der Name schon sagt, wird damit etwas interpretiert, also übersetzt. Dieser Teil des Systems hat immer dann gut zu tun, wenn Programme laufen, die in BASIC (Beginners All-Purpose Symbolic Interpretation Code = Anfängers Allzweck Symbol Sprache) geschrieben sind. Diese Sprache ist zwar der menschlichen - besonders der englischen - recht gut angepaßt; nur das Hirn des Computers, der zentrale Prozessor, versteht kein Wort BASIC. Genaugenommen versteht er gar keine Buchstaben oder gar Wörter, sondern nur Ziffern und davon auch nur zwei: die 0 und die 1. Was bleibt den Computer-Konstrukteuren also übrig, als jeden Befehl, den die CPU (Central Process Unit = Zentral-Prozessor) ausführen soll, mit diesen zwei Ziffern auszudrücken.

Aber keine Angst, wenn Sie vor Ihrer Tastatur sitzen, dürfen Sie mehr tun, als nur die Tasten 0 oder 1 zu drücken. Da hat man vorgesorgt. Nicht mal die Ingenieure, die den C64 zusammengesetzt haben, mußten auf diese Sprachstufe hinabsteigen. Die mühsame Arbeit wurde schon bei der Entwicklung des zentralen Mikroprozessors erledigt. Der ist in der Lage, Befehle, die in Maschinensprache - also dem Code, den die Maschine versteht - zu befolgen.

Später, wenn Sie schon viel, viel mehr vom Computern verstehen und Sie dazu Lust haben, können Sie lernen, Ihren 64er in Maschinensprache zu programmieren. Sie werden sich wundern, mit welcher unglaublicher Geschwindigkeit dann Ihre Programme ablaufen ! Aber zurück zum BASIC-Interpreter. Denn der ist nun dafür verantwortlich, der CPU mitzuteilen, was Sie eigentlich von ihr wollen, wenn Sie einen bestimmten BASIC-Befehl verwenden.

Der Interpreter belegt ca. 8KByte, etwa gleich viel wie das Betriebssystem des C64. Was ist denn das nun wieder, werden Sie möglicherweise fragen. Die Antwort ist simpel: ganz ohne Betriebssystem läuft die Chose nicht, ohne diese segensreiche Erfindung könnte der Computer weder feststellen, welche Taste gedrückt wurde oder welche Daten von der Diskette gelesen wurden, noch könnte er Nachrichten auf dem Bildschirm oder dem Drucker ausgeben. Das Betriebssystem kümmert sich außerdem um die Farbe.

Die ganz großen Pluspunkte des C64 liegen sicher auf den Gebieten Grafik und Musik. Diese Fähigkeiten hier zu schildern, würde den Rahmen des Buches sprengen; bei der einen oder anderen Programmidee werden wir allerdings darauf zurückkommen. In diesem Bereich lassen sich aber auch gewissen Grenzen des Komforts feststellen: will man alles ausnutzen, was die Maschine bietet, muß man wahre PEEK-und-POKE-Orgien feiern. Da greift mancher doch lieber zu Fertig-Software, die auch reichlich angeboten wird.

Die Speicherkapazität des COMMODORE 64 haben wir ja schon angesprochen. Ein Wort noch dazu: für die allermeisten Anwendungen reicht sie aus. Natürlich dürfen Sie nicht dem Irrglauben verfallen, mit diesem Wunderding, ein paar Diskettenlaufwerken, einem guten Drucker und der üblichen Software könnten Sie einen mittleren Betrieb mit - sagen wir mal - einhundert Angestellten komplett verwalten. Dem ist nicht so. Dagegen kann sich beispielsweise ein selbstständiger Malermeister mit dem 64er von einer ganzen Menge nervtötender und zeitraubender Schreib- und Rechenarbeit befreien. Und das mit einem Investitionsvolumen, das höchstens 25% der Kosten für einen neuen Lieferwagen entspricht.

Für den Hausgebrauch reicht der interne Speicher allemal aus, zumindest, wenn ein externer

Massenspeicher wie die Floppy 1541 hinzukommt. Dann bereitet das Anlegen einer Schallplatten-Datei von z.B. 500 Titeln keine Schwierigkeit.

Wenn man gewissen Meinungsumfragen unter Homecomputer-Besitzern Glauben schenken will, so nutzt immer noch ein großer Teil der User (=Benutzer) ihr Gerät für Spiele. Parallel zur allgemeinen Leistungssteigerung der erschwinglichen Computer geht eine gewaltige Verbesserung der Qualität der Spiele einher; und das nicht nur bei Ton und Farbe, sondern auch was den Inhalt der Computer-Games angeht. Wo ganz früher mehr als ein monochromes Ping-Pong oder später ein stupides Weltraum-Peng-Peng nicht drin war, entstehen heute Fantasy-Spiele, die so schön und intelligent gemacht sind, daß der geneigte Spieler wirklich ganz in eine märchenhafte Welt mit kaum lösbaren Rätseln hinabsteigen kann. Gerade für solch anregendes Freizeitvergnügen bietet der C64 beste Voraussetzungen. Neben den bereits erwähnten Vorzügen bei Sound und Grafik, hilft die große Geschwindigkeit der Maschine mit.

Wagen Sie doch den Schritt vom eher passiven Spieler zum aktiven Spiel-Designer ! Vielleicht finden Sie so zu einem ganz neuen, kreativen Hobby. Erste Anregungen dazu finden Sie in den letzten Kapiteln.

***) ANMERKUNG:**

Wenn Sie schon eine Ahnung davon haben, was sich in den Chips des Computers abspielt, lesen Sie bitte ganz vorsichtig weiter. Die Beschreibung der Innereien ist so ungenau wie unvollständig. Ich meine nur, daß keinem Einsteiger damit gedient ist, wenn er weiß, was es mit der RAM-ROM-Überlappung oder gar der speziellen Archtiktur der CPU auf sich hat. Wer sich ernsthaft mit

dem Gerät auseinandersetzt, wird sich früher oder später dafür interessieren. Dann kann er immer noch zum Standardwerk "64 intern" greifen.

1.2 Speichern tut not

Welche Datenmengen Sie speichern wollen, hängt natürlich in erster Linie von der Art Ihres Programmes ab. Eine begrenzte Zahl von Anwendungen kommt mit dem internen Speicher des C64 für Daten und Programm aus. Aber was machen Sie, wenn Sie beispielsweise Ihre Dia-Sammlung mit 200 Magazinen verwalten wollen ? Nicht schlimm, werden Sie sagen, dann benutze ich einfach meine Datensette als Datenspeicher. Und damit haben Sie auch recht.

Eines ist also klar: ein externes Speichermedium muß sein. Und nicht nur zum Festhalten von Daten, sondern auch zum Aufbewahren Ihrer selbstgeschriebenen Programme. Denn, wenn Sie Ihren 64er ausschalten, ist alles, was im Arbeits-Speicher war, unwiederbringlich verloren. In aller Regel benutzen Sie auf Ihrem Daten-Rekorder besondere Daten-Kassetten, die im Gegensatz zu den bekannten MusiCassetten nur 5, 10 oder 20 Minuten Laufzeit pro Seite haben. Es empfiehlt sich nach Möglichkeit nur ein Programm (inkl. Daten) pro Seite zu speichern. Der Grund dafür ist leicht zu verstehen. Datenkassetten können nur sequentiell, d.h. hübsch der Reihe nach, gelesen werden, so daß das komplette Magnetband nach dem Gewünschten abgesucht werden muß. Wenn z.B. die gesuchte Datei ganz am Ende steht, wird alles, was davor gespeichert ist, abgesucht.

Nur durch sehr ausgeklügelten Einsatz der besonderen C64-Eigenheiten läßt sich dieser Mangel abmildern. Von vorne herein bequemer ist natürlich die Arbeit mit der Floppy.

Bequemlichkeit ist nicht alles. Noch wertvoller ist

vielleicht die große Speicherkapazität einer Diskette. Auf der COMMODORE-Floppy 1541 stehen Ihnen 170Kbyte zur Verfügung. Erinnern Sie sich an unser Speicherplatz-Beispiel aus dem vorigen Kapitel ? Unsere Rechnung dort: 38KB entsprechen ca. 15 Seiten Text. Nach Adam Riese passen also fast fünfmal soviel Seiten auf eine Diskette. Der Text dieses Buches wurde z.B. auf nur vier Scheiben eingefangen.

Ein nicht zu unterschätzendes Argument für die Floppy ist zweifellos die unvergleichliche Zugriffs-Geschwindigkeit. Die ist deshalb so groß, weil man bei der Diskette - im Gegensatz zur Datenkassette - direkt auf die gespeicherten Informationen zugreifen kann. Das gilt zunächst nur für Programme. Es sind nämlich alle Daten auf 35 Spuren, die wiederum in mehrere Sektoren aufgeteilt sind, gespeichert. Jeder Sektor auf jeder Spur hat eine eigene Kennung, die als Name im Directory (=Inhaltsverzeichnis der Diskette) erscheint. Diese Kennung sorgt dafür, daß bei der Suche nach einem bestimmten Programm nicht die ganze Diskette abgesucht werden muß.

Das sind also drei ganz gewichtige Gründe, sich ein Diskettenlaufwerk (später auch ein zweites !) zuzulegen. Für professionelle Nutzungen des C64 ist so ein Speicher eigentlich sogar unerläßlich. Und schließlich ist die VC-1541 mit einem Preis von um die DM 700 (Stand Mai 84) auch noch erschwinglich.

- BEQUEMLICHKEIT
- KAPAZITÄT
- GESCHWINDIGKEIT

sind die großen Vorteile des Diskettenlaufwerkes. Sollten diese Argumente Sie noch nicht überzeugt haben, die Ideen in diesem Buch werden es sicher tun !

1.3 Auch ein 64er braucht Druck

"Was du schwarz auf weiß besitzt, kannst du getrost nach Hause tragen." Dieser schlaue Spruch hat auch beim Computern seine Gültigkeit. Wenn Sie erstmal den Schritt zum Textverarbeitungs- und/oder Datenbank-Programm gewagt haben, wird die Frage nach dem Drucker akut. Denn: was nützt die schönste Adressenliste, wenn sie nur auf der Mattscheibe existiert ? Und wenn es gar ans Schreiben geht, dann muß so ein Ausgabegerät einfach her. Was das alles wieder kostet, werden Sie einwenden. Aber dazu fällt mir eine recht einfache Rechnung ein.

Nehmen wir an, Sie besitzen einen C64 plus externem Speicher. Nehmen wir weiter an, Ihre alte Schreibmaschine ist reif fürs Museum und eine elektrische soll angeschafft werden; also alles Voraussetzungen, die einigermaßen realistisch sind. Folgende Alternativen stehen zur Auswahl: eine moderne elektronische Schreibmaschine, Preis ca. 900 bis 1.000 DM ODER ein zum 64er passender Drucker, den Sie - mit passablen Fähigkeiten für rund DM 700 erstehen können. Als Dreingabe erhalten Sie ein Medium, das Listings und Inhaltsverzeichnisse (von Disketten oder Kassetten) druckt und sogar ein wenig Grafik darstellen kann.

Ist die Qual der Wahl bei Diskettenlaufwerken für den C64 gering, so ist sie bei den Druckern um so größer. Rechnet man nur die gängigeren Geräte, bleiben doch immerhin rund 10 Drucker mit recht unterschiedlichen Eigenschaften im Rennen.

- SCHRIFTQUALITÄT
- GESCHWINDIGKEIT
- GRAFIK-FÄHIGKEIT

Das sind die wichtigsten Kriterien beim Druckerkauf. Außerdem spielen die von Ihnen vorrangig gewünschten Anwendungen eine Rolle.

Gehen wir doch die Varianten im einzelnen durch:

D-84, der billigste. Eigentlich nur zum Ausdrucken von Listings mit maximal 40 Zeichen pro Zeile gedacht. Wer will, kann jedoch auch mehr auf den schmalen Papierstreifen zaubern. Preis: ca. DM 400;

VC-1526, der universelle. Kam als DER Drucker zum C64 auf den Markt und mußte nach vielen Schwierigkeiten, die bei den Besitzern kaum Freude aufkommen ließen, mit einem neuen Betriebssystem ausgerüstet werden. Seitdem ist er empfehlenswert, da ein sehr gutes Preis-Leistungsverhältnis erzielt wurde. Mit dem VC-1526 kann man Briefe schreiben, die sich sehen lassen können. Preis: unter DM 1.000;

GP 700 A, der farbige. Dieser SEIKO-Drucker bringt Ihnen die Möglichkeit, den Bildschirminhalt in Farbe auszudrucken. Mit dem entsprechenden Interface druckt der 700 A den kompletten C64-Zeichensatz. Preis: ca. DM 1.800 inkl. Interface;

EPSON FX-80, der schönste. Dieser Drucker hat nicht nur das (zugegeben: Geschmackssache !) beste Design, sondern auch eine für Matrixdrucker ungewöhnlich schöne Schrift. Dazu ist er schnell und leise. Leider kann man die - sage und schreibe - 63 Schriftarten vom C64 aus nur mühsam nutzen. Aber auch das kann eine besondere Herausforderung für Freaks sein. Preis: unter DM 2.000 inkl. Interface;

VC-1520, der Plotter. So ein Gerät hat weder Drucknadeln, noch Typenrad; er schreibt mit vier verschiedenfarbigen Stiften, die per Software ausgewählt werden können. Der Vorteil: Linien einer Grafik oder eines Diagramms geraten nicht mehr zu einer Aneinanderreihung von Punkten, sondern als wirklich durchgezogene Linien. Wer die Grafik-Möglichkeiten des

C64 ausnutzen will, sollte über einen VC-1520 (vielleicht als Zweitgerät) nachdenken. Preis: unter DM 600;

MS 15/VC, der Briefeschreiber. Kurz gesagt: ein erschwinglicher Typenraddrucker, der zum 64er paßt. Empfehlenswert für semi-professionelle Textverarbeiter. Preis: rund DM 1.700.

Außerdem gibt es noch Drucker mit so speziellen Eigenheiten, daß sie hier nicht besprochen werden können.

1.4. Wenig Software - wenig Nutzen

Jeder Homecomputer-Besitzer braucht Software. Das wissen wir bereits. Doch, wie kommt man an gute Programme, mit denen die ganz individuellen Aufgaben bewältigt werden können? Grundsätzlich gibt es drei Möglichkeiten:

- 1) Programme selbst entwickeln
- 2) Programmlistings abtippen
- 3) Fertig-Software kaufen

Man kann nun nicht behaupten, die eine oder andere Möglichkeit wäre besser, billiger oder komfortabler, da eine solche Einschätzung von den persönlichen Bedürfnissen abhängt. Zu dem sind die drei Bereiche nicht scharf voneinander abgegrenzt. Betrachten wir die Punkte im einzelnen.

Programme selbst entwickeln

Die unterste Stufe ist das Schreiben von kurzen Testprogrammen, die den Gebrauch von BASIC üben. Hier werden Sie normalerweise nicht mehr als 10 bis 20 Programmzeilen eingeben und sich dann daran freuen, daß der Computer macht, was Sie wollen.

Doch recht bald werden Ihre Ansprüche steigen. Wenn Sie dann die Motivation verspüren, beim Selbstprogrammieren zu bleiben, sind Sie auf dem besten Wege zum BASIC-Programmierer. Sicher, niemand wird völlig aus der hohlen Hand Superprogramme entwickeln, Anregungen aus Fachbüchern (siehe Literaturverzeichnis), aus einschlägigen Computer-Magazinen oder von Hobbykollegen aus diversen User Clubs gehören schon dazu; aber viele, heute recht bekannte Programme entstanden in den Wohnzimmern von

Amateur-Programmierern. Allerdings besteht die Gefahr, daß das Programmieren zum Selbstzweck wird und die ursprünglichen Anwendungsideen in den Hintergrund geraten. Dafür haben Sie dann den Schlüssel zum Wunderland der Chips in der Hand.

Programmlistings abtippen

Grauzone zwischen Selbstprogrammieren und Abtippen ist das Anpassen von Listings für andere Computer an Ihren C64. Die Anforderungen an Ihre Programmierkenntnisse sind aus zwei Gründen kaum geringer als im ersten Fall. Erstens ist BASIC eine Sprache mit vielen Dialekten. Die Lokalisierung von PRINT-Befehlen auf dem Bildschirm ist z.B. bei allen gängigen Homecomputern sehr verschieden. Wollen Sie also ein schickes Programm, das für den ATARI geschrieben ist, auf den COMMODORE übersetzen, müssen Sie es Zeile für Zeile nach solchen PRINT-Anweisungen durchkämmen. Im zweiten Fall wird es ganz schlimm: wenn PEEKs und POKEs auftauchen ist, es sogar höchst kompliziert, Programme, die auf dem VC 20 laufen, für den 64er nutzbar zu machen. Trotzdem können sich sehr wohl die Mühen der Anpassung lohnen. Oft haben in diesen Programmen gewiefte Amateure simple, aber wirkungsvolle Ideen untergebracht, die für alle Gleichgesinnten von Nutzen sind. Warum soll jeder Computer-Fan das Rad zum zweiten Mal erfinden ? Abkupfern ist in diesem speziellen Fall wirklich keine Schande.

Reine Programmsammlungen tauchen in verschiedenster Form auf. Will man darin gefundene Programme nicht direkt übernehmen, bleibt doch zumindest die Chance, aus den Erfahrungen anderer zu lernen. Aber Vorsicht ! Je nachdem, wie unseriös der jeweilige Herausgeber ist, muß man damit rechnen, daß die Listings mit Fehlern gespickt sind, die einem wirklich alles vermiesen können. Gerade Veröffentlichungen in den zahlreichen Computer-

Zeitschriften, eigentlich die wichtigste Quelle fürs Abtippen, sind mit Vorsicht zu geniessen. Es kann durchaus vorkommen, daß sich die Korrekturen eines Programms über etliche Ausgaben hinziehen. Nur wenn man treu und brav alle nötigen Hefte gekauft und die Änderungen nachvollzogen hat, darf man sich am funktionierenden Programm freuen.

Wer sich schon mal ein Wochenende mit dem Debugging (=Fehlersuche und -bereinigung) solcher Listings um die Ohren geschlagen hat, wird geneigt sein, beim nächsten Mal ausgetesteter Software aus einem einschlägig bekannten Haus zu vertrauen. Womit wir beim nächsten Punkt wären ...

Fertig-Software kaufen

Wer die Wahl hat, hat auch die Qual. Und die Auswahl bei Fertigprogrammen ist groß. Gerade für den C64 wird eine schier unübersehbare Zahl verschiedenster Programme angeboten. Selbst wenn man von den Spiel-Programmen absieht, bleibt noch genug übrig. Es gibt Textverarbeitungs-Programme auf Diskette, Datenbanken auf Kasette, Graphik-Tools (=Grafik-Hilfsprogramme) auf Steckmoduln, usw., usw. Wie soll man da den Überblick kriegen ?

Versuchen Sie es einmal mit der folgenden Check-Liste:

```

*****
*
*          CHECK-LISTE für den SOFTWARE-KAUF
*
*****
*
*
* A) FRAGEN
*
* 1) Für welchen Aufgabenbereich wird die Software
*     benötigt ? (siehe Kapitel 2.1 bis 2.6)
* 2) Welcher Art soll das Speichermedium sein ?
*     - Steckmodul
*     - Kassette
*     - Diskette
* 3) Wie teuer darf die Software maximal sein ?
*
*****
*
* B) CHECK-POINTS
*
* 1) Software-Tests in den Fachzeitschriften studieren
* 2) Software-Anzeigen beachten
* 3) Interessante Angebote aussuchen
* 4) Beim Händler Informationen über die ausgesuchten
*     Software-Pakete einholen
*     - mündlich im Computerladen oder
*     - telefonisch beim Versandhandel
* 5) Anwender der ausgesuchten Software interviewen
* 6) Handbücher dieser Software-Pakete lesen
* 7) Beim Händler Software am Computer vorführen lassen
* 8) Preisvergleiche anstellen
* 9) Falls möglich, Software-Paket zum Test nach Hause
*     ausleihen
*10) Falls nötig, die Punkte 1),2),4) und 5) wiederholen
*11) Software endgültig auswählen
*12) Service der Anbieter testen
*     - können defekte bzw. versehentlich gelöschte

```

```

*      Programm-Disketten kostenlos oder gegen Schutz-   *
*      Gebühr umgetauscht werden ?                         *
*      - kann man sich bei Schwierigkeiten, die während  *
*      des Betriebs auftauchen, telefonisch Hilfe geben  *
*      lassen ?                                           *
*      - wird ein Software-Paket kostenlos (oder gegen   *
*      geringe Kosten) gegen eine aktualisierte Fassung *
*      ausgetauscht ?                                     *
*13) Anbieter mit dem besten Preis-Leistungs-Verhältnis *
*      auswählen                                         *
*14) Software kaufen                                     *
*                                                                 *
*****

```

Sie werden vermutlich die Liste nicht stur von Anfang bis Ende durchgehen, zumal einige Punkte nur aus Sicherheitsgründen wiederholt werden. Ein Wort in eigener Sache: aus dem gleichen Haus wie dieses Buch stammen einige Software-Pakete, die alle Punkte der Check-Liste erfüllen. Die DATA-BECKER-Produkte DATAMAT, TEXTOMAT, Supergraphik 64 und andere werden bei den Programm-Ideen verständlicherweise öfter auftauchen. Sie sollten diese Software in Ihre engere Auswahl nehmen.

Ansonsten werden Sie bei der Software-Auswahl sicher die gleiche Sorgfalt walten lassen, wie Sie es beim Kauf Ihres COMMODORE getan haben; bedenken Sie, daß z.B. eine komfortable Benutzerführung (Begriffserklärung später !) die Arbeit am Gerät genauso erleichtert, wie eine ergonomisch (=körpergerecht) geformte Tastatur !

1.5 Fernseher oder Monitor ?

Das hört sich an wie eine Gretchen-Frage, ist es aber nicht. Hier geht es nicht um Glauben oder Meinen, sondern eher um den Umsatz Ihres Apothekers mit Augetropfen. Sicher, solange Sie es nicht ernst meinen mit dem Computern, wird es der vorhandene Fernseher auch tun. Aber schauen wir uns zunächst einmal an, was es zum Gucken am Computer alles gibt.

- Schwarz-Weiß-Fernseher
- Farbfernseher mit Videoeingang
- Monochromer Monitor (oft ohne Tonwiedergabe)
- Farbmonitor
- Farbmonitor mit Tonwiedergabe

Noch ganz kurz zum Arbeiten mit einem üblichen Fernseher: der größte Nachteil dieser Geräte für Ihre Zwecke liegt darin begründet, daß das sichtbare Bild ganz anders erzeugt wird als bei Datensichtgeräten. Dadurch ist beispielsweise eine befriedigende Auflösung gar nicht zu erreichen. Darunter leidet dann auch die Bildruhe, mit drei Worten: das Bild flimmert. Physikalisch gesehen werden beim normalen Fernseher abwechselnd Bilder abgestrahlt, die die gewünschte visuelle Information enthalten, und solche, die schwarz sind. Ihre Augen werden so mit einem dauernden, bewußt nicht wahrzunehmenden Helligkeitswechsel konfrontiert. Überreizung und Müdigkeit sind die Folgen.

Ganz beheben läßt sich mit der verfügbaren Kathodenstrahlröhren-Technologie dieser Mangel nicht, aber wesentlich mildern. Inzwischen kann man sich auch kaum noch mit dem Argument "Monitore sind so teuer" herausreden. Gute monochrome Datensichtgeräte gibt es schon unter DM 300. Natürlich können Sie auch das 10fache investieren, z.B. für Super-Farbmonitor mit HiFi-Tonwiedergabe, wenn Sie wollen

...

Schauen wir uns eine Auswahl von Geräten an, die zum COMMODORE 64 passen:

SANYO DM 2112, der preiswerte. Die Daten: 12" Bildschirm; 15MHz Taktfrequenz; wahlweise grüne oder (als SANYO 2212) bernsteinfarbene Darstellung. Dieser Zwerg ist sogar für die Darstellung von 80 Zeichen pro Zeile geeignet und damit das preiswerteste Gerät für Leute, die vorwiegend Textverarbeitung betreiben. Preis: um DM 300.

TECO TM-1265, der brillante. Die Daten: 12" Bildschirm; 20MHz; Grüne Darstellung, entspiegelt. Dieser Monitor bietet ein tolles, brillantes Bild und ist deshalb für alle zu empfehlen, die Ihren C64 für Textverarbeitung und Datenverwaltung einsetzen. Das Spielen am TM-1265 ist weniger lustig, weil Farbe und Ton fehlen. Preis: ca. DM 400.

PHILIPS V 7001/00, der lautstarke. 12" Mattscheibe. Spezielle Monitortechnik, daher besonders flimmerfrei. Der Clou: nicht nur allgemein Tonwiedergabe, sondern ein spezieller Kanal für Sprachwiedergabe. Empfehlenswert für alle, die außer auf Farbe auf nichts verzichten wollen, was den Monitor zum wichtigen Bestandteil der Anlage macht. Preis: ebenfalls rund DM 400.

SANYO COLOR CD 3185, der farbige. 36 cm großer Bildschirm, entspiegelt, umschaltbar auf monochrome Grün-Darstellung. Mit allen notwendigen Video- und Audioeingängen ausgestattet, eignet sich dieser SANYO nicht nur für alle Computerzwecke, sondern sogar als Monitor für Ihre Video-Anlage. Mit Zusatzkabel direkt an den 64er anschließbar. Preis: unter DM 1.000.

COMMODORE 1702, der angepaßte. Gehört einfach dazu, weil

er alle Eigenschaften hat, die man braucht, wenn man alle Fähigkeiten des C64 nutzen und optisch und akustisch genießen will. 14" Bildschirm, gute Tonwiedergabe. Preis: rund DM 900.

Zugegeben, eine kleine Auswahl; aber wenn Sie sich einmal darüber informieren, wie groß das Monitor-Angebot ist, werden Sie die Beschränkung verstehen. Welcher für Sie der Richtige ist, hängt stark davon ab, welche Anwendungen bei Ihnen im Vordergrund stehen. Die Programm-Ideen enthalten auch Hinweise darauf, welches Equipment (=Ausrüstung) jeweils am besten geeignet ist.

1.6 Nützliche Kleinigkeiten

Fangen wir beim Bekanntesten an:

Joysticks und Paddles

Es ist wohl nicht übertrieben, wenn man behauptet, daß jeder, der schon einmal die Tastatur eines Computers bedient hat, auch weiß, wie man mit einem Joystick umgeht. Sparen wir uns also langatmige Erklärungen über Aussehen und Funktion dieser Steuerknüppel.

Für uns sind sie sowieso nur dann interessant, wenn wir sie für etwas anderes als Spiele verwenden wollen. Und dann sind Paddles manchmal sogar wesentlich besser geeignet. Mit diesen Zubehör-Teilen lassen sich beispielsweise Menue-Punkte auf dem Bildschirm ansteuern und anwählen.

Die Preise für Joysticks liegen zwischen DM 40 und weit über DM 100, Paddles sind meistens billiger.

Lightpen und Grafik-Tablett

Der Lightpen (auf gut Deutsch: Lichtgriffel) ist ein vielseitig nutzbares Eingabemedium. Mit dem Stift in der Hand kann man - je nach Software-Unterstützung - direkt auf dem Bildschirm malen und zeichnen oder wie mit Joysticks Menuepunkte anwählen. Gerade für Grafik-Programme wäre der Lightpen eine tolle Sache, wenn, ja wenn es einen gäbe, der

zum C64 paßt. Zur Zeit bekommt man nur solche Stifte, die nicht software-unterstützt und damit fast unbrauchbar sind. Das Gesagte gilt im Großen und Ganzen auch für ein Grafik-Tablett. Einziger Unterschied: es wird nicht auf die Mattscheibe gezeichnet, sondern auf eine besonders präparierte Fläche. Zum C64 passend ist das KOALA-PAD (wird mit einem Steckmodul geliefert), das die grafischen Möglichkeiten des Computers voll ausnützt. Preis: etwa DM 300.

Pflege und Schutz

Haben Sie eine Vorstellung davon, was passiert, wenn ein Staubkorn auf einer Diskette mit ins Laufwerk kommt ? Nein ? Glauben Sie mir, das kann, muß aber nicht ins Unglück führen. Man sollte jedenfalls alles dazu tun, daß die bösen, kleinen Staubkörner ihr Zerstörungswerk weder in der Floppy, noch im Drucker oder sonstwo anrichten können. Für alle genannten Geräte gibt es Staubschutzhüllen, die man sich schleunigst zulegen sollte. Und die Disketten lagern wir in einer staubdichten (wenn Sie wollen, abschließbaren) Disketten-Box. So, alles dicht ?

Pflege im engeren Sinne des Wortes brauchen nur zwei Peripherie-Geräte: Floppy und Drucker. Und wieder sind Staub und Schmutz die Schuldigen.

Der Schreib-Lese-Kopf des Diskettenlaufwerks muß ab und zu gereinigt werden, denn an ihm bleibt alles hängen. Ein verschmutzter Kopf kann sich beim Datenlesen oder -speichern schon mal vertun. Und so Ihre mehrtägige Arbeit vernichten. Also lieber vorbeugen. Besondere Reinigungsdisketten machen diesen Job.

Beim Drucker ist der Staub sozusagen hausgemacht. Besonders

in der Perforation von Endlospapier sitzen feine Fasern, die vom Druck-Kopf abgeklopft werden und irgendwo ins Gerät fallen. Wohin merken Sie erst, wenn die interne Elektronik des Druckers ausfällt. Zur Druckerpflege gibt es keine besonderen Mittel. Ein Tip: im Fotogeschäft bekommt man Druckluft aus der Spraydose, die bläst den Staub weg. Aber bitte nicht im Bereich der Elektronik verwenden, das können die Fachleute besser. Zu denen bringen Sie vernünftigerweise jedes Gerät, das nicht einwandfrei funktioniert.

2. Grundrezepte

Haben Sie schon mal ein gutes Kochbuch in der Hand gehabt ? Dann ist Ihnen sicher aufgefallen, daß im Anhang oft eine Liste sogenannter Grundrezepte erscheint und daß die verführerischen Rezepte im Hauptteil meistens nichts anderes sind als mehr oder weniger modifizierte Grundrezepte. Nun, meiner Meinung nach verhält es sich beim Computern ähnlich, wie beim Kochen: aus drei bis vier Grundanwendungen lassen sich durch kleine Änderungen und geschicktes Mixen tausend Programme zusammenbrauen. Deshalb habe ich dieses Kapitel "Grundrezepte" genannt.

- TEXTVERARBEITUNG
- DATENVERWALTUNG
- RECHNEN und KALKULIEREN
- SPIELEN

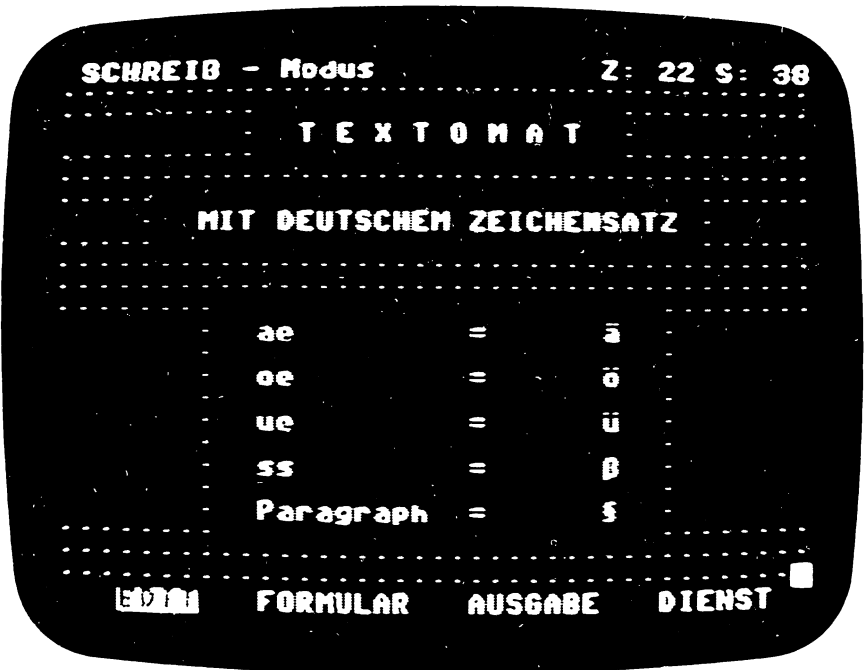
Dabei kann der Punkt "Spielen" eigentlich nicht als eigenständiges Anwendungsgebiet bezeichnet werden, weil die berühmt-berüchtigten Computergames nicht mehr sind als eine Mischung der ersten drei Punkte plus Ton und Grafik.

Bei der Einteilung der Programm-Ideen habe ich mich an das obige Schema weitgehend gehalten. Nur das Kapitel "Musizieren, Malen und Entwerfen" fällt etwas aus dem Rahmen, da dort Ideen beschrieben sind, die mehrere Grundanwendungen in vermischter Form nutzen.

Wenn Sie nun Ihre Siebensachen - C64, Floppy 1541, Monitor und Drucker - beisammen haben, das eine oder andere Fertigergericht (Textverarbeitungs-, Datenverwaltungs- und andere Software) im Vorratsschrank steht, nehmen Sie dies Kochbuch und probieren Sie die Leckerbissen. Bleibt mir nur noch, Ihnen einen "Guten Appetit" zu wünschen !

TEXTOMAT

Ein außergewöhnliches Textverarbeitungsprogramm: 80 Zeichen pro Zeile durch horizontales Scrolling, Ausdruck bis zu 255 Zeichen, Textlänge bis zu 24000 Zeichen im Speicher, Verketteten von Texten, umfangreiche Textbausteinverarbeitung und Formatierungsmöglichkeit, Formularsteuerung, Anpassung an fast jeden Drucker, Diskettenverwaltung, umfangreicher Befehlssatz und ca. 30 Steuerzeichen, Schnittstelle zu DATAMAT zur Erstellung von Rundschreiben mit individueller Anrede. TEXTOMAT ist komplett in Assembler geschrieben und dadurch extrem schnell. Menuesteuerung, deutsche Benutzerführung, natürlich deutscher Zeichensatz auf dem Schirm und ausführliches, 75-seitiges Handbuch machen gerade für den Anfänger die Arbeit mit TEXTOMAT zum Kinderspiel und das zum sagenhaften Preis von nur DM 99,-.



2.1 Textverarbeitung

"Das ist ein weites Feld", hätte mein alter Deutschlehrer zu diesem Thema sicher angemerkt. Wie bei allen weiten Feldern kriegt man zunächst leichte Orientierungsschwierigkeiten beim Erforschen des Begriffs "Textverarbeitung". Das fängt schon bei der Begriffsbestimmung an. Was ist eigentlich ein Text ? Ganz einfach: ein Text ist eine geordnete Aneinanderreihung von Wörtern zu einem gewissen Zweck. Diese Erklärung scheint ein wenig abstrakt, muß aber so sein. Präziser wird es, wenn wir damit anfangen, den Zweck näher zu erläutern.

- BRIEFE
- DOKTOR-, DIPLOM- und ähnliche ARBEITEN
- ROMANE, ERZÄHLUNGEN, GEDICHTE usw.
- NOTIZEN

Jede dieser Anwendungen hat ihre eigenen Probleme. Eines ist ihnen jedoch allen gemeinsam: Fehler.

Blenden wir ein paar Jahre zurück, genauer gesagt ins Jahr 1 vor KT. (das Jahr 0 dieser Zeitrechnung ist das Jahr, in dem die Korrektur-Taste an elektrischen Schreibmaschinen erfunden wurde !). Die Schreibmaschinen-Technologie hatte schon eine Reihe von Fortschritten gemacht. Die Sehenscheidenentzündung als Berufskrankheit der Sekretärin schien nach der Einführung der elektrischen Schreibmaschine so gut wie ausgerottet, der Kugelkopf trat seinen Siegeszug durch die Büros der Welt an, der Weltrekord im Schnellschreiben erreichte ungeahnte Höhen und der TIPP-EX-Verbrauch wuchs. Denn: wer schneller schrieb, machte auch schneller Fehler. Die kleinen weißen Fläschchen, die gerade bei weiblichen Benutzern so anheimelnde Erinnerungen an Nagellack auslösten (was bisweilen zu weißen Fingernägeln

führte), waren von keinem Schreibtisch mehr wegzudenken.

Leider hatte dies Wundermittel die unangenehme Eigenschaft, im Laufe der Zeit einzutrocknen. Also, vor Gebrauch schütteln und, falls nötig, Verdüner einfüllen. Das Schütteln von TIPP-EX-Fläschchen wurde zur typischen Handbewegung aller Schreibmaschinentäter. Die Zeit, die durch den Kugelkopf gewonnen war, ging fürs Schütteln wieder drauf. Gut, ganz so dramatisch wars nicht, aber fast ...

Aber dann, dann schlug die Erfindung des Mister IBM wie eine Bombe ein: jene kleine, orangefarbene Taste, die das weiße Fläschchen fast verdrängen sollte. Hatte man in einer Zeile einen Fehler entdeckt, fuhr man den Schreibkopf auf den entdeckten Fehler, drückte "Orange", anschließend den korrekten Buchstaben und der Fehler war aus der Zeile getilgt. Welch ein Fortschritt !

Eine Verbesserung schien undenkbar. Dabei gab es sie schon. Große Unternehmen hatten Jahre zuvor bereits riesige - wie sie damals genannt wurden - Elektronen-Gehirne angeschafft, die, von magischen Männern und Frauen in weißen Kitteln liebevoll umsorgt, rechneten, was das Zeug hielt. Die Ausrede "Unser Computer ist ausgefallen" wurde zum geflügelten Wort bei jeder Art von Schlamperei. Clevere Organisations-Experten kamen bald auf den Gedanken, daß diese Riesenmaschinen doch zu mehr gut sein müßten, als nur mit Zahlen um sich zu werfen. Das Zeitalter der elektronischen Textverarbeitung brach an. Überspringen wir nun die Jahre, in denen Textverarbeitungs-Software nur für fünf-stellige DM-Beträge zu haben war.

Die entscheidende Frage war die: Wie wäre es, wenn der über die Tastatur eingegebene Text statt auf dem Papier auf dem Bildschirm dargestellt würde ? Es stellte sich heraus, daß das ganz prima wäre. Denn die Elektronen, die auf die Leuchtschicht der Bildröhre abgefeuert Buchstaben ergaben, hatten eine erfreuliche Eigenschaft, die den schwarz auf weiß gehämmerten Zeichen abging: sie ließen sich in

wahnwitziger Geschwindigkeit verändern. Die schlauen Erfinder waren sich einig, so sollte es sein: ein Text wird auf den Bildschirm getippt, anschließend werden die Fehler ausgemerzt und dann gehts ab aufs Papier.

Soweit zur Fehlererkennung und -bekämpfung. Bei all diesem Nachdenken fielen ihnen noch einige Dinge ein, die für uns heutzutage das Textverarbeiten mit dem Computer so angenehm machen.

- Formatieren
- Einfügen von Textblöcken
- Finden und Ersetzen bestimmter Textpassagen
- Speichern von Text

Formatieren

An geeigneter Stelle im Arbeitsablauf beschäftigt sich der Texter damit, wie das Geschriebene eigentlich später auf Papier aussehen soll. "Wie," werden Sie fragen, "sieht man das nicht beim Schreiben ?" Die Antwort ist nicht ganz einfach. In dieser Frage gibt es zwei Philosophien. Die eine besagt, daß das Textverarbeitungs-Programm in der Lage sein muß, während des Prozesses der Textherstellung alles so auf den Bildschirm zu bringen, wie es später auf dem Papier aussieht. Das hat aber nur einen Sinn, wenn der Monitor in der Lage ist, alles das darzustellen, was auch der Drucker drucken kann. Um es vorweg zu nehmen: solche Programme sind toll, aber leider für den C64 nicht geeignet, denn der kann auf dem Bildschirm z.B. weder fett schreiben, noch können normalerweise hoch- oder tiefgestellte Zeichen dargestellt werden. Vergessen wir diesen Ansatz also einstweilen.

Die zweite Philosophie, die bei beinahe allen

Textverarbeitungs-Paketen, die für den 64er angeboten werden, zutrifft, besagt, daß das Formatieren, also das Aufteilen eines Textes zum anschließenden Druck, in einem separaten Arbeitsgang geschehen soll. Einen ganz gewichtigen Vorzug hat dieses System. Der läßt sich am verständlichsten an einem Beispiel erklären:

Sie haben mit einem Textverarbeitungs-Programm einen Brief geschrieben. Beim Texten haben Sie weder auf Zeilenlänge noch auf Worttrennungen geachtet. Ihr Text ist fertig und Sie wollen ihn ausdrucken. Der Drucker weiß aber gar nicht, wie er das anstellen soll. Also werden Sie aufgefordert, ein Formular anzulegen. Diese Anweisung enthält folgende Angaben

- linker Rand
- rechter Rand
- oberer Rand
- unterer Rand
- Blocksatz Ja/Nein
- Zentrieren Ja/Nein
- Zeilenabstand
- Zeichenabstand
- Zeichensatz-Auswahl

Nachdem Sie dem Drucker die gewünschten Angaben gemacht haben (die auch im Computer gespeichert sind), können Sie sich den so formatierten Text auf dem Bildschirm zeigen lassen. Wenn das Gezeigte nicht Ihren Vorstellungen entspricht, brauchen Sie bloß das Formular zu ändern. Dies Spiel können Sie solange treiben, bis Sie zufrieden sind.

Nebenbei: wo meine Beschreibungen ins Detail gehen, beziehen sie sich auf das Textverarbeitungs-Programm TEXTOMAT aus dem Hause DATA BECKER (zum Preise von DM 99), mit dem übrigens auch der Text dieses Buches geschrieben, bearbeitet, formatiert und auf dem EPSON-DRUCKER LQ 1500 gedruckt wurde ! Die für den C64 geeigneten Textverarbeitungs-Pakete

ähneln sich - außer im Preis. Alles über TEXTOMAT gesagte läßt sich im großen und ganzen auch auf andere Programme übertragen.

Textblöcke bearbeiten

Was ein Textblock ist, können Sie selber definieren. Ein beliebiges Stück Text wird durch Ihre Markierung zum Block. Mit einem solchen Textblock können Sie eigentlich alles machen, was Sie wollen.

- Löschen
- Speichern
- Kopieren
- Verschieben

Was diese Fähigkeit in der Praxis bedeutet, läßt sich am besten an einem Beispiel verdeutlichen. Nehmen wir an, Sie benutzen Ihr Textverarbeitungs-Programm zum Schreiben von Angeboten. Sie können sich vorstellen, daß eine ganze Reihe von Floskeln in jedem Brief immer wieder gebraucht werden. Z.B. der Absatz:

"Dieses Angebot mit den genannten Preisen, Lieferbedingungen und Rabatten halten wir 14 Tage aufrecht. Nach Ablauf dieser Frist fordern Sie bitte ein überarbeitetes Angebot an."

Probieren Sie selber aus, wie lange Sie brauchen, diesen Absatz zu tippen; bei mir waren es rund 20 Sekunden. Das bedeutet bei 100 Briefen 2.000 Sekunden = ca. 30 Minuten. Aber halt, Sie haben recht, die gewonnene Zeit brauche ich dafür, den wiederverwendbaren Textblock von der Diskette einzulesen. Es lohnt sich also nicht, kurze Blöcke zu speichern. Wie wäre es denn, wenn man den ganzen Brief abspeichern würde? Gut, aber alle konkreten Stückzahlen und Preise sind z.B. jedesmal verschieden. Was tun?

Sie schreiben ein Angebot, daß vom Text her recht durchschnittlich ist. Diesen Brief entwerfen Sie wie gewohnt, bearbeiten ihn und übergeben das so entstandene Werk dem Drucker. Jetzt lassen Sie den Text ruhig auf dem Bildschirm und beginnen mit Ihren Blockoperationen: alle Stellen, die nur diesem einen Angebot zugehörig sind, löschen Sie (auch das Datum !). Den Rest erklären Sie zum Block und speichern ihn ab. Wenn Sie nun wieder ein Angebot zu schreiben haben, lesen Sie den so entstandenen Block ein und verpassen ihm an den richtigen Stellen die passenden Angaben. Fertig ist das Angebot. Zeitersparnis in diesem Fall: ca. 70 Minuten bei 100 Briefen.

Auch bei der Arbeit an umfangreichen Manuskripten werden Sie Blockoperationen zu schätzen wissen. Mir ging es jedenfalls beim Schreiben dieses Buches so: oft gefiel mir eine bestimmte Textpassage so gut, daß ich sie auf Verdacht abspeicherte. Und siehe da, der Rest des Kapitels ging mangels Qualität in den Elektronen-Mülleimer, aber mein Textblock überlebte alles und ist heute auf diesen Seiten zu bewundern. Man kann sagen: was man magnetisch auf der Floppy hat, kann man getrost wiederverwenden.

Übrigens müssen diese Blockoperationen nicht immer über den externen Speicher laufen. Sie können auch innerhalb eines Textes, der nur auf dem Bildschirm existiert, Blöcke schieben, kopieren und löschen. Probieren Sie einmal aus, was mit dieser Methode alles geht, ich habe noch lange nicht alle Möglichkeiten durch !

Finden und Ersetzen

Das ist wirklich ein tolles Ding ! Sie können einen ellenlangen Text nach einer bestimmten Zeichenfolge absuchen und jedes Mal, wenn das Gesuchte auftaucht, den Zusammenhang übersehen.

Auch wieder ein Beispiel: Sie arbeiten an einem hochwissenschaftlichen, historischen Werk über die Binnenschifffahrt in Südosteuropa, kann ja sein. Nun spielt eine gewisse, sprichwörtliche Gesellschaft eine herausragende Rolle: die Donaudampfschiffahrtsgesellschaft. Dieses Bandwurmwort wird insgesamt über vierzig Mal genannt. Schonen Sie Ihre Nerven und Fingerspitzen; überall, wo das ominöse Ding erscheinen soll, tippen Sie das Kürzel "DD" ins Manuskript. Nach dem Ihre Abhandlung in der Rohform fertig ist, wählen Sie die Funktion "Finden und Ersetzen" an. Sie suchen nach der Zeichenfolge "DD" und weisen das Programm an, wann immer die Abkürzung gefunden wurde, diese durch die ausgeschriebene Variante "Donaudampfschiffahrtsgesellschaft" zu ersetzen. So einfach ist das. Überall, wo in diesem Text das Wort "Computer" stand, findet sich im Manuskript das Zeichen "c", wenn Sie mögen, können Sie nachzählen, wie oft ...

Eine weitere Anwendung ist die Rechtschreib-Prüfung. Wenn Sie wüßten, wie schwer ich mich mit der Ortografy, Entschuldigung, ich meine mit dem richtigen Buchstabieren von Fremdwörtern tue. Meine Methode: beim Lesen des Manuskripts erscheint mir die Schreibweise des Wortes Kathodenstrahlröhre verdächtig. Also kommt das 19-Buchstaben-Wort auf den Steckbrief, und zwar in der Form "Kat*", da kann ich nicht falsch liegen. Das Programm soll, falls ich im Einzelfall zustimme, diese Zeichenfolge durch die richtige (duden-geprüfte) Schreibweise ersetzen. Natürlich wird mir nicht nur die bewußte Röhre angeboten, sondern auch so uninteressante Dinge, wie "Katze", "Kater", "Katastrophe" und "Katarakt", aber auch die Kathodenstrahlröhre kann berichtigt werden. Auch hier gilt: Probieren geht über Studieren !

Text speichern

Es bedarf wohl keiner besonderen Erwähnung, daß jeder Text, der auch nur in dem Verdacht steht, irgendwann einmal gebraucht zu werden, auch abgespeichert wird. Und eins dürfte klar sein: nur mit einem (besser noch: mit zweien) Diskettenlaufwerk als Speichermedium ist Textverarbeitung im beschriebenen Sinne wirklich machbar.

Zu diesem Thema nur ein paar allgemeine Tips:

- Speichern Sie Ihre Texte nicht erst, wenn alles fertig ist. Gerade bei längeren Ergüssen empfiehlt es sich, Blöcke von etwa 80 bis 100 Zeilen Länge zu begrenzen und abzuspeichern. Schon mancher hat im Müdigkeitsschwahn morgens um halb drei den Stecker rausgezogen ohne den im Arbeitsspeicher liegenden Text abzuspeichern.
- Geben Sie Ihren Textteilchen sinnvolle Namen !
Eine Möglichkeit: Kürzel des Gesamttitels plus "text" plus Nummer des Kapitels. Beispiel: Ihr Werk heißt "Die elektronische Schaltung im Wandel der Zeiten", Kapitel 7.1.1 "Buddhismus und Transistoren". Das ergibt als Blocktitel "eswztext7.1.1". Schreiben Sie als erstes ein - wenn auch vorläufiges - Inhaltsverzeichnis, das Sie von Zeit zu Zeit revidieren.
- Speichern Sie grundsätzlich doppelt !!!
Eine Diskette ist zwar ein recht zuverlässiges Speichermedium, aber eben auf magnetischer Basis. Wissen Sie, was passieren kann, wenn Sie versehentlich eine Schere auf eine Floppy Disk gelegt haben ? Nein ? Scheren haben die unangenehme Eigenschaft leicht magnetisch zu sein und diese Magnetfelder bringen Ihre schönen gespeicherten Informationen erheblich durcheinander. Ganz leicht kann die Arbeit mehrer Wochen futsch sein. Dieses Back-Up (=Doppeltes Speichern zur Sicherheit) können Sie täglich oder aber nur dann durchführen, wenn eine Diskette voll ist. Mit zwei Laufwerken ist das eine ganz leichte Sache, mit

einem müssen Sie schon Disk-Jockey spielen: Datendiskette rein, in den Arbeitsspeicher einlesen, Back-up-Diskette rein, Text darauf speichern, Text aus Arbeitsspeicher löschen, Datendiskette rein, usw. usw. Wenn Ihnen das zu umständlich ist, gewöhnen Sie es sich wenigstens an, den gerade geschriebenen Text auszudrucken, wenn dann was mit der Diskette ist, haben Sie Ihr Werk wenigstens auf Papier.

2.2 Rechnen, Planen, Kalkulieren

Wo immer es um Zahlen geht, hat der Computer ein Heimspiel. Das ist sein eigentliches Element: Addieren, Subtrahieren, Multiplizieren, Dividieren, Sortieren, statistisch Auswerten, Verknüpfen usw. geht im Inneren der Maschine rasend schnell. Aber, was gehört alles dazu, sollen die Ergebnisse dem Benutzer genießbar serviert werden. An dieser Stelle setzen die Kalkulations-Programme an.

Um gleich Verwirrung zu schaffen: gerade auf diesem Gebiet ist das Fachchinesisch besonders komplex. Wir wollen die wirklich professionellen Software-Pakete, die inzwischen für ziemlich viel Geld auch auf den 64er zu bringen sind, ausklammern. Wer einen mittleren Betrieb finanztechnisch per Computer verwalten will, wird wohl kaum auf den C64 zurückgreifen.

Was können Kalkulations-Programme ?

- Tabellen erstellen
- Innerhalb der Tabellen rechnen
- Tabellen auswerten
- Statistiken erstellen
- Statistiken darstellen
- Diagramme entwerfen
- Prognosen erstellen (begrenzt!)

Daraus lassen sich drei, aufeinander aufbauende Grundfunktionen destillieren.

- TABELLEN
- STATISTIK
- DIAGRAMME

Wenn alles so läuft, wie ich mir es denke, werden Sie im ersten Abschnitt die grundsätzlichen Dinge über Tabellenkalkulation erfahren und anschließend, was Sie mit den Ergebnissen alles machen können. Nur eins vorweg: Aussagen über management-orientierte Planungs-Software und ähnlich komplexe Dinge, bleiben ausgeklammert. Wer darüber mehr wissen möchte, bediene sich der zahlreich vorliegenden Publikationen dazu.

Tabellen

Leider läßt sich der Begriff "Tabelle" nur sehr abstrakt erklären. Also: eine Tabelle ist ein mehrdimensionales Feld mit definierter Verknüpfung der Felder. Ein Beispiel wird uns auf die Sprünge helfen. Eine Multiplikations-Tabelle (im Volksmund Ein-mal-eins genannt) hat im einfachsten Fall zwei Dimensionen; in vertikaler Richtung die Zahlen von - sagen wir mal - 1 bis 4 und in horizontaler Richtung die Zahlen von 1 bis 12. Die Verknüpfung der beiden Dimensionen sei die Multiplikation. Es entsteht folgendes Bild:

```
*****
*-----1---2---3---4---5---6---7---8---9--10--11--12*
*****
*---*
*-1-*
*---*
*-2-*
*---*
*-3-*
*---*
*-4-*
*---*
*****
```

Man kann sich jetzt ein Schachbrett aus $12 * 4 = 48$ Felder denken. In jedem der Felder erscheint auf Anweisung die vorgegebene Verknüpfung; von links oben nach rechts unten: 1,2,3,...,40,44,48. Schon ist die Tabelle fertig. Wollen Sie mal sehen ?

```

*****
*-----1---2---3---4---5---6---7---8---9--10--11--12*
*****
*---*
*-1-* 1  2  3  4  5  6  7  8  9 10 11 12*
*---*
*-2-* 2  4  6  8 10 12 14 16 18 20 22 24*
*---*
*-3-* 3  6  9 12 15 18 21 24 27 30 33 36*
*---*
*-4-* 4  8 12 16 20 24 28 32 36 40 44 48*
*---*
*****

```

Betrachten Sie diese Tabelle als Ausgangsmaterial. Sie könnten nämlich noch die verschiedensten Dinge damit anstellen:

- zeilenweise addieren
- spaltenweise addieren
- Ausschnitte definieren
- Ausschnitte addieren
- usw.

Natürlich braucht es nicht bei der Addition zu bleiben. Ein Beispiel: Sie lassen jede Zeile aufaddieren und anschließend berechnen und anzeigen, wieviel Prozent der Gesamtsumme eine Zeilensumme ausmacht.

Oder Sie gehen in die dritte Dimension. Abstrakt ausgedrückt hatte Ihr Koordinaten-System bisher eine x-Richtung (= horizontal) und eine y-Richtung (= vertikal), die im Ursprung (= gemeinsamer Nullpunkt) in einem Winkel von 90 Grad aufeinander stehen. Der Schritt in die dritte Dimension

bedeutet nun, daß eine z-Achse hinzugefügt wird, die wiederum mit 90 Grad auf dem Ursprung steht. Es entsteht ein imaginärer Würfel mit den Kanten x-Achse, y-Achse und z-Achse. Entsprechend der Einteilung dieser drei Achsen ist der Würfel in lauter kleine Würfelchen eingeteilt, die nun jeweils ein Ergebnis vorgegebener Operationen enthalten. Schwierig, nicht ?

Ein ganz konkretes Beispiel macht es vielleicht deutlich. Nehmen Sie unsere Tabelle, sie ist zweidimensional und begrenzt durch $x=12$ und $y=4$. Wir ersetzen diese Werte (z.B. x bleibt gleich; y erhält die Werte 5 bis 10) und bekommen eine neue Tabelle, sagen wir: ein zweites Blatt. Noch einmal setzen wir neue Werte ein, Tabelle Nr. 3 ist fertig. Dies Spiel wiederholen wir noch ein paar mal und haben schließlich einen Haufen Blätter auf unserem gedachten Schreibtisch. Die nehmen wir, bringen Sie zum Buchbinder, der Sie schön zusammenleimt und mit einem Einband versieht. Was wir dann in der Hand halten, ist eine dreidimensionale Tabelle, wobei z gleich der Seitenzahl ist. Der Vorteil gegenüber einer real existierenden Tabellensammlung ist, daß über die Seiten hinweg Verknüpfungen durchgeführt werden können.

Und nun zur vierten Dimension. Ich höre Sie schon stöhnen, aber im Computer ist die vierte Dimension ein Kinderspiel. Auch unser Bild von eben funktioniert noch. Stellen Sie sich vor, Sie haben sich auf oben beschriebene Weise nicht nur ein, sondern ganz viele Bücher machen lassen. Alles klar ? Vielleicht wirken die Programm-Ideen im Kapitel 4. ein wenig erhellend.

Statistik

Jetzt kommt uns zugute, daß wir im vorhergehenden Abschnitt so ins Detail gegangen sind. Denn Tabellen sind gute Voraussetzungen für Statistik. Und Statistik ist im einfachsten Falle nichts anderes als das Aufsummieren von Spalten und Zeilen. Wohl gemerkt, im einfachsten Fall.

Definieren wir den Begriff "Statistik" durchaus unwissenschaftlich, aber brauchbar für unsere Zwecke. Statistik ist, wenn man Ordnung in einen Wust von Daten bringt. Ganz toll ist, wenn es gelingt, nach dem Ordnen Rückschlüsse auf das Verknüpfungssystem dieser Daten zu machen. Ein Beispiel: wir sammeln im Bekanntenkreis Geburtsdaten. Von den Bekannten, deren Kindern, Omas und Opas, Nachbarn usw. Dazu lassen wir uns noch jeweils das Lieblingsessen nennen. Alle noch ungeordneten Informationen legen wir in einer geeigneten Tabelle ab. Nun kann das Spiel beginnen. Wieviel Prozent der Befragten gehören zum Sternbild "Skorpion" ? Wieviele sind "Stier" ? Wieviel Prozent der Auskunftspersonen geben als Leibgericht "Königsberger Klopse" an ? Oder: wieviel Prozent der "Wassermänner und -frauen" essen am liebsten Gurkensalat ? Anschließend versuchen wir die alles entscheidende Frage zu beantworten: Gibt es einen signifikanten (=typischen) Zusammenhang zwischen Sternzeichen und Lieblingsessen ? So wird man zum Statistiker.

"Alles Schaumschlägerei !" werden Sie einwenden. Okay, bei diesem Beispiel haben Sie ja recht. Aber ersetzen Sie einmal Sternzeichen durch Alter und Leibgericht durch Hobby, erweitern Sie den Kreis der Befragten. Als Inhaber eines Bastelgeschäfts würde Ihnen sicher etwas einfallen, um das Ergebnis in Umsatz zu übersetzen.

Die logische Folge der Statistik ist die Prognose. Wie der

Name schon sagt (griechisch/römisch: pro = vor, vorher und gnosis = Glaube, Meinung) verlassen wir das Land der harten Fakten. Besonders im Falle, daß wir über keinen Großrechner vom Schlage der Geräte besitzen, die im Fernsehen die Wahlergebnisse immer so schön falsch vorhersagen. Trotzdem, eine Beschäftigung ist das Thema wert. Nur nicht hier, mangels Platz. Also nur ein paar Stichworte:

- Daten sammeln und erfassen
- Daten ordnen
- Daten statistisch erfassen
- Statistik = Diagnose
- Aus der Diagnose eine Prognose entwickeln
- Das Eintreffen der Prognose an neuen Daten überprüfen

Diagramme

Nach all den schwierigen, abstrakten Dingen, die ich Ihnen bisher zugemutet habe, nun etwas Einfaches. Das Diagramm ist, wie auch immer es aussehen mag, eine grafische Darstellung von Daten, Statistik und Prognose. Als Besitzer der Floppy 1541 haben Sie sich vielleicht auch kaputtgelacht, als beim Abspielen der Demodiskette ein Balkendiagramm über Umsatzzahlen von Computerfirmen vorgeführt wurde. Ich habe mich jedenfalls nicht gewundert, daß der gelbe COMMODORE-Balken unter allerlei Geräuschen den A-Balken (A wie APPLE oder ATARI ?) und die anderen überholte. Solche Balkengrafiken lassen sich jedenfalls ohne besondere Software selber programmieren und sind oft das beste Mittel, Ergebnisse anschaulich zu machen.

Komischerweise müssen die Werbeleute der Computerhersteller der Ansicht sein, sogenannte Kuchengrafiken seien besonders eindrucksvoll. Kaum eine Anzeige für den neuesten Personal-Computer kommt ohne solche Diagramme aus. Ist aber auch schön: die ganze Torte stellt 100% dar, die Stücke haben einen ihrem Anteil entsprechende Größe. Ohne

Grafik-Erweiterung sollten Sie sich nicht unbedingt darauf versteifen, Ihr Haushaltsbudget per Tortengrafik darzustellen. Die notwendigen PEEKs und POKEs könnten Ihnen den Appetit verderben.

Immer wieder gern gesehen sind Kurven-Grafiken. Sie haben den Nachteil, daß sie bei der Darstellung von mehr als ca. vier oder fünf Linien zugleich unübersichtlich werden. Kurven sind jedoch allgemein am besten dazu geeignet, Veränderungen von Werten in der Abhängigkeit von der Zeit darzustellen.

Wenn Sie Anregungen für die Gestaltung Ihrer Diagramme suchen, möchte ich Sie aufs Fernsehen verweisen. Da wird seit einiger Zeit häufig gemalt (siehe die Bemerkung zur Wahlanalyse oben) und viele dieser Diagramme kann man ganz gut auf dem 64er nachempfinden.

Diagramme sind der Showteil einer trockenen Datensendung. Mit schicken Grafiken kann man oft einen guten Eindruck hinterlassen und aus ziemlich aussageschwachen Daten noch etwas fürs Auge machen. Sie wissen ja: "Theres no business like showbusiness"

2.3 Daten speichern und sortieren

Was sind Daten ? "Da stellen wir uns ganz dumm", lautet das geflügelte Wort, das, aus dem berühmten Film "Die Feuerzangenbowle" stammend, in solchen Fällen zur Anwendung gelangt. Beherzigen wir den Rat und tun so, als ob wir nicht bis drei zählen könnten.

Wieso spricht man immer von den Daten, also in der Mehrzahl ? Wie heißt eigentlich die Einzahl ? Ich will es verraten: Datum. Aha, da haben wir eine Spur, die zur Auflösung der Begriffsverwirrung führen könnte. Das Datum kennen wir, Freitag, der 13. April 1984, 1. Mai oder 10.11.52 sind uns geläufige Ausdrücke. Und richtig, wir haben die Prototypen der Daten gefunden.

Wichtigste Eigenschaft unserer Beispiele: sie unterscheiden sich. Erstens im Wert (klar, alles verschiedenen Tage) und zweitens im Format. Kämen wir auf die Idee, einem programmierten Terminkalender die Daten derart chaotisch einzugeben, würde uns der Computer, spätestens bei der Ausgabe, vermutlich aber schon bei der Eingabe, was husten und sich mit einem mürrischen "REDO FROM START"-Error (= "Fang-noch-mal-an-Fehler) verabschieden. Sehr gute Programme treiben der CPU diese Flausen aus, indem sie den Computer für alle denkbaren Eingabe-Formate wappnen. Wir lernen trotzdem: Daten, die weiter verarbeitet werden sollen, müssen in aller Regel einem definierten Format entsprechen.

Ansonsten gibt es bei der Datenverwaltung nicht viele Spielregeln zu beachten. Außer einer eminent wichtigen: Versuche nie mehr Daten einzugeben als der Arbeitsspeicher schlucken kann ! Im schlimmsten Fall führt so etwas zum Datenmüll; es ist fast alles noch da, aber wie ...

Ach so, was sind denn nun Daten ? Definieren wir es mal salopp: alle Tatsachen, die sich computergerecht durch

alphanumerische Zeichen (=Ziffern und Buchstaben) ausdrücken lassen. Also z.B. Namen, Adressen, Titel, Preise, Daten (im Sinne von Tag/Monat/Jahr), laufende Nummern, Schuhgrößen, Haarfarben, Hubraum, Speicherkapazität, offene Rechnungen, Notizen, Termine, Kalorien und und ...

Sie sehen, Spiel ohne Grenzen. Nun müssen diese Daten noch irgendwie Form annehmen und das nicht nur beim Eingabeformat. Das Mittel, das Ordnung in die Sache bringt, heißt Datei (oder englisch: File). Ein wirkliches Universalwort ! Wo mehr als zwei Daten versammelt sind, wird sofort ein File aufgemacht. Und das muß so sein. Nur mit geeigneten Dateien ist es möglich, die gespeicherten Informationen auch wieder rauszukriegen. Der Möglichkeiten, Dateien anzulegen sind viele. Erklären lassen sich die verschiedenen Arten besser am konkreten Fall, als sozusagen auf dem Trockenen. Nehmen Sie das Gesagte einfach als gegeben an.

Einfache Dateien, die durch ein- oder mehrdimensionale Felder (=Arrays) gebildet werden, lassen sich prima selber anlegen und verarbeiten. Dazu braucht man nur recht einfache Programme, die man mit BASIC-Grundkenntnissen selber basteln kann. Ein gutes Beispiel finden Sie in dem Buch "64 für Einsteiger" von Norbert Szczepanowski, das ebenfalls im Hause DATA BECKER erschienen ist.

Die grundsätzlichen Arbeiten an einer Datei sind folgende

- Datei einrichten
- Daten pflegen
- Datei sortieren
- Datei auswerten

Wenn auch diese Tätigkeiten bei selbstgeschriebenen Programmen - mehr oder weniger - obligatorisch sind, erklären lassen Sie sich am besten an einem Software-Paket,

wie dem DATAMAT (ein DATA BECKER Produkt für DM 99).

Datei einrichten

Die erste Frage, die Sie sich stellen müssen ist, "Welche Daten will ich speichern und bearbeiten?". Die zweite, "Welche Informationen bilden einen Datensatz?". Nehmen wir wieder ein Beispiel: Sie möchten die Geburtstage Ihrer ungewöhnlich umfangreichen Verwandtschaft in den Griff kriegen. Also nennen Sie Ihre Datei

*** GEBURTSTAGE ***

und beschließen nach sorgfältigem Nachdenken folgende Informationen in jeden Datensatz aufzunehmen:

- Verwandtschaftsgrad (Onkel, Tante, Sohn ...)
- Vorname
- Nachname
- Strasse/Nr.
- PLZ
- Wohnort
- Telefon
- Geburtsdatum
- Geschenkideen
- Beliebtheitsgrad
(Erbonkel, böse Schwiegermutter, Patenkind etc.)

Sie überprüfen die Liste, ob Sie auch nichts vergessen haben und werfen einen Blick auf die Reihenfolge. Die scheint noch nicht optimal zu sein. Aber dazu später. Nun entwerfen Sie eine Eingabemaske, d.h., Sie fahren mit dem Cursor (kennen Sie: das blinkende Quadrat, das die Stelle markiert, an der das nächste Zeichenentsteht). Vielleicht ziehen Sie erstmal einen schicken Rahmen, setzen den Namen der Datei rein und - verfallen in dumpfes Grübeln. "Wo", so fragen Sie sich,

"sollen die einzelnen Informationen auf meiner Karteikarte eingefügt werden ?" Ich mache Ihnen einen Vorschlag: sehen Sie sich mal die Eingabemaske an, die ich für Sie entworfen habe.

```
*****
*
*          *** GEBURTSTAGE ***          *
*
*****
* 1) Datum   : 21.11.1950                *
*
* 4) Vorname : Annita                    *
*           _____                  *
*
* 5) Name    : Fietsch                   *
*           _____                  *
*
* 4) Adresse : Schwege                    Nr.99 *
*           _____                  *
*
* 6) PLZ/Ort : 4578.Bohmte II            *
*           _____                  *
*
* 8) Telefon : _____                *
*
* 9) Verwandt: Cousine                   *
*           _____                  *
*
*10) Beliebt : Alte Liebe                *
*           _____                  *
*
*11) Geschenk: Rote Rosen                 *
*           _____                  *
*
*****
```

Das Programm verlangt jetzt nach einem Namen für die Maske und für die mit dieser Maske entstehende Datei. Es ist meistens sinnvoll, den gleichen Namen zu verwenden, aber nicht immer. Sie könnten ja die gleiche Maske für zwei Dateien verwenden (in unserem Fall ist das unwahrscheinlich, es sei denn, Sie wechseln Ihre Verwandtschaft). Nachdem die

Maske abgespeichert ist, fragt DATAMAT danach, welches Feld Ihrer Eingabe-Maske Indexfeld sein soll, d.h. nach welchem Informationsstückchen später in Ihrer Datei rumgesucht werden kann. Bei dieser Entscheidung sollten Sie drei Dinge beachten:

- Je kürzer das Indexfeld, desto mehr Datensätze passen in die Datei
- Je weiter vorne das Indexfeld in der Maske steht, desto schneller kann gesucht werden
- Wählen Sie ein Indexfeld, das garantiert bei jedem Datensatz eine Information erhält

Die beiden ersten Tips betrachten Sie bitte nur als Faustregeln.

Auf der Mattscheibe erfahren Sie, wieviele Datensätze Ihre Datei enthalten darf und werden gefragt, wieviele Sie wirklich hineinpacken wollen. Seien Sie ruhig ein bißchen knauserig. Geben Sie am besten die Anzahl ein, die Sie wirklich brauchen und nicht, weil Sie 657 Datensätze aufnehmen könnten, gleich 650. Sooo groß wird Ihre Verwandtschaft doch nicht sein.

Datei pflegen

Ihre Geburtstagsdatei ist nun fertig zur weiteren Bearbeitung. Das heißt Sie können nun beginnen, Daten einzugeben. Damit verbringen Sie jetzt die nächsten anderthalb Stunden. Und ich kann endlich eine Pause machen.

So, fertig mit Eingeben ? Ja ? Dann kann ich ja wieder etwas erklären. Sie haben jetzt Gelegenheit, in Ihrer Datei zu stochern und zu wühlen und sich alles raussuchen lassen, was Sie interessiert. Das geht, wenn Sie den Menue-Punkt "Suchen" angewählt haben. Wenn Sie das Indexfeld benutzen, geht es am schnellsten. Das Indexfeld in der

Geburtstags-Datei ist sinnigerweise der Monat. Wieso der Monat ? Ich habe mir gedacht, daß die simpelste Verwendung der Datei die ist, daß man sich am Letzten eines Monats zeigen läßt, wer im nächsten Monat Geburtstag hat. Heute ist der 30. Mai. Also geben Sie ins Indexfeld die Zahl 06 für Juni. Wenn Sie die Suche gestartet haben, erscheint in Sekundenschnelle der erste Verwandte, der im Juni Geburtstag hat auf dem Schirm. Gucken Sie sich das in aller Ruhe an.

Das Programm will nun wissen, ob es weitersuchen soll. Soll es, befehlen Sie, denn vermutlich haben noch andere Onkels oder Tanten im Juni Geburtstag. Und schon erscheint die nächste Karte auf dem Schirm usw. bis alle Junikinder dran waren. So geht das Suchen.

Und wie geht Ändern und Löschen ? Wenn Sie einen Datensatz der gelöscht oder geändert werden soll, auf dem Monitor haben, wählen Sie die Menue-Punkte "Ändern" bzw. "Löschen". Beim Ändern können Sie dann genauso arbeiten, wie beim Eingeben; beim Löschen, nun das probieren Sie mit Ihrem eigenen DATAMAT aus.

Datei sortieren

Jetzt wirds spannend. Hier beginnt der Computer seine Vorteile voll auszuspielen. Was wir bisher mit unserer Datei angestellt haben, hätten wir auch zu Fuß erledigen können: Anlegen, Ausfüllen und einmal Sortieren. Okay, aber nun stellen Sie sich vor, Sie wollten aus Ihrem Geburtstags-Karteikasten alle Karten aussuchen, die mit Monat November bezeichnet sind, die den Vermerk "Beschenken" tragen und als Geschenkidee "Krawatten". "Kleinigkeit!", werden Sie sagen. Fangen Sie doch schon mal an, ich denk mir was Neues aus.

Ach ja, jetzt hätte ich gerne alle Junikinder, die Tanten sind. Wie, noch nicht fertig? Richtig, Sie müssen ja erst alle beim ersten Mal rausgefischten Karten an die richtige Stelle bringen. Sehen Sie, das muß der Computer nicht machen.

"Was solls?", werden Sie sagen, "Das brauch ich aber selten." Bei einer Geburtstags-Datei vielleicht, aber bei einer 1.000 Namen umfassenden Kundenkartei?

Außerdem kann jede dieser Selektionen (=Auswahlen) als "Pointerdatei" gespeichert und bei Bedarf wieder hervorgeholt werden. Und: wenn Sie bei der Eingabe von neuen Datensätzen alles richtig machen, können Sie die schnell und bequem auch in Ihren Pointerdateien aufnehmen.

Zum gleichen Komplex zählt das allgemeine Sortieren einer Datei. Das kann mit dem Computer mehrstufig passieren.

- Sortiere nach Monaten
- Innerhalb des Monats nach Tagen-

- Innerhalb des Datums nach Vornamen
- Ende

Sie könnten nun innerhalb des Vornamens nach Verwandtschaftsgrad sortieren lassen. Aber weil der Computer nach dem Alphabet vorgeht, käme möglicherweise folgendes dabei heraus: Datum: 21.11.; Vorname: Annita

- 1) Cousine Annita
- 2) Mutter Annita
- 3) Nichte Annita
- 4) Oma Annita
- 5) Schwester Annita
- 6) Tante Annita
- 7) Urgroßmutter Annita

Quatsch, nicht wahr ? Trotzdem werden Sie Dateien haben, bei denen es sehr nützlich ist, die Datensätze nach allen Eingabefeldern zu sortieren. Wir kommen noch darauf zurück.

Datei auswerten

Diesen Punkt handeln wir jetzt aber ganz kurz ab. Eine Datei auszuwerten bedeutet eigentlich nur, daß alle oder nur ausgewählte Datensätze in einer definierten Form ausgedruckt werden.

Sie können sich beim Format einer solchen Liste ganz auf Ihren Drucker verlassen, also beispielsweise die 80-Zeichen-Breite ausnutzen, Steuerzeichen senden um Breit-, Schmal- oder Schönschrift zu befehlen usw.

Auch welche Felder wo gedruckt werden, bleibt Ihnen überlassen. Überschriften lassen sich einfügen. Für unsere

Geburtstags-Datei wäre folgendes Ausgabeformat sinnvoll:

Geburtstag:(tt).(mm).(jj)
Vorname/Nachname
Adresse/PLZ/Ort
Telefon
Geschenkidee

Bedenken Sie: mit diesem Format wird für jeden Monat eine neue Liste ausgedruckt ! Sie können sich so eine Liste immer für den folgenden Monat erstellen oder fürs ganze Jahr. Das hat den Nachteil, daß Veränderungen jedesmal zu einem komplett neuen Ausdruck führen.

Mit dem gleichen Programmteil lassen sich auch die Etiketten herstellen, mit denen die Geschenkpäckchen versehen werden.

So weit, so gut. "Ja, der DATAMAT kann das ", werden Sie sagen,"aber andere Datenverwaltungsprogramme ?" Ich kann Sie beruhigen. Die meisten Datenverwaltungs-Pakete für den C64 haben ähnliche Eigenschaften und bieten die gleichen Möglichkeiten. Sie unterscheiden sich eigentlich nur durch den Bedienungskomfort, die Arbeitsgeschwindigkeit und - den Preis. Wenn Sie sich in etwa nach der Check-Liste für den Software-Kauf (Kapitel 1.4) richten, werden Sie schon das Richtige für sich herauspicken.

2.4 Musizieren

Ganz ehrlich: ich liebe Musik. In jeder Form, von Vivaldi bis Bruckner, von Elvis bis zu den Beatles, ob Rock, ob Reggae oder Funk, ich höre mir alles gerne an und bin trotzdem unmusikalisch. Mehr als "Hänschen klein" auf der Blockflöte war mir nicht vergönnt. Und nun soll ich über das Musizieren mit dem COMMODORE 64 schreiben. Nur Mut, ich fange an:

"Die Basisadresse des SID ist 54272(Dez)=D400(Hex), dabei kann das Lo-Byte Werte von (0..255), das Hi-Byte von (0..255) annehmen. Beachten Sie, daß der Anschlag für die Stimme 1 im Register 19 hart (0*16) oder weich (15*16) eingestellt werden kann."

Sie hörten aus der Reihe: "Der COMMODORE 64 als Synthesizer" die erste Folge: "Grundlagen für Anfänger". So ungefähr springt das C64-Handbuch mit uns armen Laien um, wenn wir es wagen, das Musizieren mit unserem Computer lernen zu wollen. Das scheint ja ein ganz geheimer Geheimbund zu sein, dessen Mitglieder das schon können.

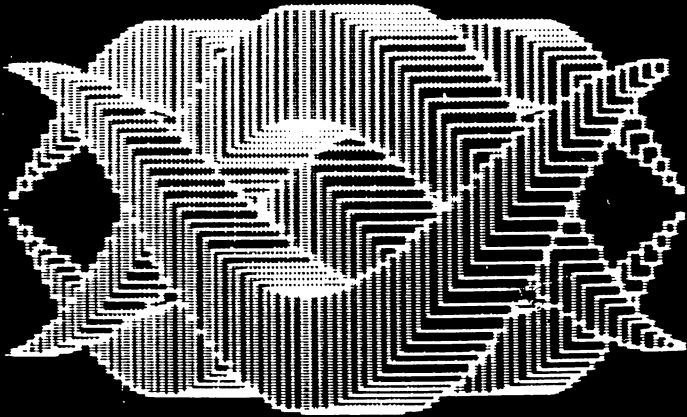
Egal, lassen Sie sich nicht abschrecken. Es gibt phantastische Hilfsmittel, um dem 64er wohlklingende Geräusche zu entlocken. Die Grafikerweiterung SUPERGRAFIK enthält z.B. recht einfache Befehle auch zum Erzeugen von Musik; das Software-Paket SYNTHIMAT befaßt sich ausschließlich damit und wird inzwischen sogar von professionellen Musikern benutzt. Jedem der in diese Materie tiefer eindringen möchte, empfehle ich das C64 MUSIKBUCH (ein DATA BECKER Produkt).

Ich verabschiede mich einstweilen, um mir noch einmal die Musik-Demos der Floppy-1541-Testdiskette zu Gemüte zu führen.

SUPERGRAPHIK 64

Die neueste Version unserer beliebten SUPERGRAPHIK enthält jetzt über 30(!) Befehle zur Ausnutzung der fantastischen Möglichkeiten, die der 64 mit hochauflösender Graphik und Farbe bietet. Mit SUPERGRAPHIK 64 können Sie Punkte, Linien und Kreise ziehen. SPRITES definieren und manipulieren, Farben setzen, komplette Graphikbildschirme auf Diskette abspeichern bzw. laden und vieles andere mehr. Ergänzt wurde die SUPERGRAPHIK 64 zusätzlich um SUPERSOUND, eine neue Befehls-erweiterung zur Nutzung der hervorragenden Sound-möglichkeiten des 64 und der Farb-Hardcopy auf dem neuen SEIKO GP 700 A. Mit SUPERGRAPHIK 64 machen Sie mehr aus Ihrem 64er, und das für nur DM 99,-.

Dies ist ein kleines Beispiel der
grossen Leistungsfähigkeit von
SUPERGRAFIK 64



2.5 Malen, Zeichnen und Entwerfen

Was die Bemerkung über die Umgangsformen des C64-Handbuchs mit Anfängern angeht, möchte ich mich nicht wiederholen, aber beim Thema "Grafik" auf das oben Gesagte verweisen.

Vielleicht haben Sie schon davon gelesen, daß Ihr Computer über ungeahnte Möglichkeiten im Bereich Grafik verfügt; daß beispielsweise im HIRES-Modus (=high resolution = Hohe Auflösung) 320 mal 200 Punkte einzeln gesetzt werden können, das sind, wenn mein Taschenrechner recht hat, 64.000 Stück. Da könnte man tolle Bilder malen, wenn man könnte. So bleibt es bei zwei Empfehlungen: beschaffen Sie sich entweder eine Grafikerweiterung (z.B. SUPERGRAFIK, s.o.) oder ein gutes Buch zum Thema (z.B. das C64 GRAFIKBUCH von DATA BECKER) oder beides. Es könnte sein, daß Sie ganz plötzlich ein neues Hobby haben.

Weil mir das Malen und Zeichnen mehr liegt als das Musikmachen, will ich mich noch nicht so schnell verabschieden, wie im vorigen Kapitel. Es gibt nämlich ein paar Dinge, die mit Grafik zu tun haben, aber trotzdem kein Studium voraussetzen.

Sie können nämlich sogar im normalen Groß/Grafik-Modus (=die Betriebsart, bei der nur Großbuchstaben, aber auch die Grafikzeichen dargestellt werden) ganz hübsche Bilder herstellen. Dazu dienen die Zeichen, die auf der Vorderseite einiger Tasten zu sehen sind. Am besten probieren Sie das aus. Wenn ein wenig Übung dazu kommt, stehen Ihnen der Bildschirm-Speicher (wo gespeichert wird, was in jedem der 1.000 Bildpunkte steht) und der Farbspeicher (wo steht, welche Farbe jeder der 1.000 Punkte hat) zum fleissigen Poken offen.

Das Prinzip ist einfach. Man stelle sich vor, diese Speicher

wären nichts anderes als Setzkästen mit jeweils 1.000 Fächern (haben Sie auch einen an der Wand, in dem Kram verstaubt ?). Wenn sich nun der Computer daran macht ein Bild auf die Mattscheibe zu zaubern, geht er ganz systematisch vor: er schaut zunächst in den Setzkasten Numero 1 (der für die Zeichen) ins Fach Numero 1, nimmt das gefundene Zeichen mit zum Kasten Numero 2. Dieser hat in jedem Fach ein kleines Farbtöpfchen. Der Computer taucht das mitgebrachte Zeichen in den passenden Farbtopf und klebt es (von innen) an die Glasscheibe des Monitors (jetzt wirds aber zu albern !). Diesen Vorgang wiederholt die Maschine 1.000 Mal, bis das Bild fertig ist.

Normalerweise bestimmen Sie den Inhalt der Setzkastenfächer durch PRINT-Anweisungen in Ihrem BASIC-Programm. Sie können das aber auch schneller erledigen; mit POKE(Fachnummer,Zeichencode) werden die gewünschten Symbole in der gewünschten Farbe direkt in die Fächer geworfen. Das ist schon ein Fortschritt ! Um es Ihnen nicht zu einfach zu machen, sind die Fächer nicht von 1 bis 1.000 numeriert, sondern von 1.024 bis 2.023 (für die Zeichen) bzw. von 55.296 bis 56.295. Aber es ist eine leichte Übung, das umzurechnen.

Eine Frage bewegt dann immer noch den Grafik-Fan. "Wie, um Himmels Willen, komme ich an die hochauflösende Grafik ?" Wie in vielen anderen Dingen, ist das ein Verpackungsproblem. Verpackt man nämlich die Informationen zu jedem Bildpunkt in einem Byte (=8 Bit, Sie wissen schon ...), können eben nur 1.000 Punkte definiert werden, mehr Platz ist einfach nicht da. Das Dumme ist, daß ein ganz einfacher Punkt genauso ein Byte zum Transport braucht, wie ein kompliziertes Zeichen, weil alle Zeichen durch ein 8x8-Kästchen-Muster dargestellt werden. Etwa so:

```
01234567
*****. 1
...**... 2
...**... 3
...**... 4
...**... 5
...**... 6
...**... 7
..... 8
```

Jedes Sternchen steht für eine 1, der Punkt für die 0. So kann jedes Zeichen des Zeichensatzes dargestellt werden, also auch die Grafik-Symbole.

Im HIRES-Modus braucht man keine Zeichen, also stellt man die Gleichung auf 1 Punkt = 1 Bit. Noch eine Rechnung: 8 Bit = 1 Byte; 1024 Byte = 1 Kilobyte. 64.000 Bit = ? KB; richtig etwas weniger als 8 Kilobyte. Im normalen Groß/Grafik-Modus sind je 1.000 Byte als Bild- bzw. Farbspeicher belegt und das wäre für HIRES zu wenig. Also werden flugs die Speicher verschoben, anders genutzt und schon funktioniert die Sache. Übrigens, wenn Sie einmal Probleme mit der Grafik haben, der Schuldige heißt VIC 6569, von Beruf Mikroprozessor und mit dem Chef Ihres C64 eng verwandt.

Hinweise auf ganz konkrete Nutzung der geschilderten Tatbestände finden Sie in dem Buch 64 INTERN aus dem gleichen Hause wie dieses Werk.

In den Kapiteln 6.3ff werden Sie Anregungen finden, wie Sie die Grafik-Fähigkeiten Ihres Geräts für etwas anderes als schlichtes Bildermalen verwenden können. Nur eins vorab: Stellen Sie sich vor, Sie könnten ein beliebig gekrümmte Linie mit diversen Haken und Schnörkeln direkt auf den Schirm zeichnen und anschließend erfahren, wie lang diese Linie ist; fallen Ihnen da nicht schon ganz tolle Anwendungen ein ?

2.6 Selbst programmieren

Vielleicht sträubt sich alles in Ihnen dagegen, sich mit diesem Thema zu befassen. Kann ich verstehen, im Vertrauen, ich selber bevorzuge die Arbeit mit Fertig-Software, besonders dann, wenn ich z.B. Texte verarbeiten oder Daten verwalten will und nicht kopfüber in den BASIC-Interpreter stürzen möchte.

Neulich ist mir jedoch etwas Merkwürdiges passiert. Ich blätterte nichtsahnend in einer Computerzeitschrift, als mir plötzlich ein Artikel ins Auge sprang, der mit einem einzigen Wort überschrieben war:

*** F O R T H ***

Das klang nach "force" (=Kraft, Stärke) oder nach "fourth" (=der, die, das Vierte) und machte mich neugierig. Mein Erstaunen war groß, als ich merkte, daß der Artikel von einer Programmiersprache handelte. Trotzdem vertiefte ich mich in die Beschreibung und kam bald zu dem Schluß "Das wär was für mich !". So, und nun will ich Ihnen die Spannung erhalten und von etwas ganz anderem berichten. Es gibt folgende Programmiersprachen, die auf dem 64er laufen

- BASIC (eingebaut)
- PASCAL
- LOGO
- FORTH
- ASSEMBLER

Natürlich gibt es noch viel, viel mehr Programmiersprachen (COBOL, FORTRAN, C, Lisp, ADA, um nur die geläufigsten zu nennen) aber nur die genannten sind auf die eine oder andere

Weise auf dem C64 zu verwenden. Über BASIC wollen wir nicht viel sprechen, Informationen dazu finden Sie sowieso in etlichen Kapiteln.

PASCAL

PASCAL ist von Hause aus eine Lehrsprache, d.h. sie ist besonders geeignet, Programmieren zu lernen. Sie sollten nicht der Illusion unterliegen, das Programmieren bestände ausschließlich aus dem Eintippen von Programmzeilen, im Gegenteil, der weitaus umfangreichere Teil der Arbeit ist für vorbereitende Tätigkeiten anzusetzen.

- Programmidee
- Programmwurf
- Ablaufplan/Flußdiagramm
- Entwerfen von
 Bildschirmmasken, Dateiaufbau, Variablenlisten
- Ausgabeverfahren
- Programmdokumentation

sind die Marksteine einer Programmentwicklung vor dem Eintippen. PASCAL unterstützt solch strukturiertes Vorgehen dadurch, daß ein Teil der gezeigten Aufgaben in PASCAL Bestandteil der Programme ist.

Strukturiertes Programmieren ist überhaupt das Schlüsselwort für PASCAL. Jedes Programm ist durch die von der Sprache erzwungene Struktur jederzeit durchschaubar. Nur ein Beispiel: PASCAL kommt ohne den in BASIC-Programmen beliebten Befehl GOTO aus, der in BASIC-Programmen bisweilen zu wüsten Verirrungen führt.

Darüberhinaus sind PASCAL-Programme ziemlich schnell und dadurch eher als BASIC-Programme für professionellen Einsatz geeignet.

LOGO

So pfiffig der Name, so pfiffig die Sprache. Wer von LOGO schon mal gehört hat, verbindet diese Sprache automatisch mit dem Thema "Kinder & Computer". Das hat seinen guten Grund. LOGO ist eine Sprache mit wenigen Befehlen, niemand muß sich mit kaum verständlichen Kürzeln plagen (wie in BASIC: CLR, LEN, SYS o.ä.), sondern kann mit (wenn er englisch kann) einfachen Wörtern den Computer steuern. Nachteil: LOGO-Programme sind unverhältnismäßig lang.

Ähnlich ist übrigens etwas, das sich "Turtle-Graphics" (=Schildkröten-Zeichnen) nennt. Als eigene Sprache läßt sich TG nicht bezeichnen. Die "Turtle" ist nichts anderes als ein Cursor, der direkt auf den Bildschirm zeichnet und schrittweise gesteuert wird. LOGO und TG müßten eigentlich gut zum 64er passen, ich habe es allerdings noch nicht ausprobiert.

FORTH

"Endlich", werden Sie jetzt denken und wenn Sie den Abschnitt gelesen haben, sind Sie ganz enttäuscht, weil Sie garnicht so begeistert von FORTH sind, wie ich. Lassen wir es darauf ankommen.

FORTH ist

- leicht lernbar
- schnell und
- universell anwendbar

kann man ohne Übertreibung behaupten. Der Clou ist das Wesen der Befehls-Bibliothek von FORTH. Sie ist sehr schmal. Das bedeutet, FORTH kann sich im Speicher des Computers sehr

klein machen. Weil man mit wenig Befehlen wenig anfangen kann oder ellenlange Programme braucht, haben die Erfinder von FORTH zu einem einfachen Trick gegriffen. Der Befehlsvorrat von FORTH kann vom Benutzer jederzeit für seine programmspezifischen Zwecke erweitert werden.

FORTH kennt keinen Befehl zum Potenzieren, also muß erst mit bekannten Befehlen erklärt werden, was Potenzieren ist. FORTH kann multiplizieren, also definiert man zu Beginn des Programms das Potenzieren als mehrfaches Multiplizieren. So einfach ist das. Soll in einem Programm nur gerechnet werden, braucht man praktisch kein Ausgabebefehl, wird ein Textprogramm benötigt, muß das Potenzieren nicht definiert werden. Ein FORTH-Programm muß nie in einer großen Bibliothek suchen, wenn ein Befehl eingegeben wird.

Es lohnt sich sicher, FORTH genauer unter die Lupe zu nehmen. FORTH ist in vielen Anwendungen bis zu zehn-, zwanzig-, dreißigmal schneller als ein ähnliches BASIC-Programm.

Assembler -----

Genauso schwierig wie das Programmieren in Assembler, ist die Begriffsbestimmung. Sie haben sicher schon gehört oder gelesen, daß jemand in Maschinensprache programmiert hat und haben den leichten Verdacht, das könnte was mit Assembler zu tun haben. So ist es. Nur ist der Ausdruck "in Maschinensprache" eigentlich korrekter; denn laut Microcomputer-Lexikon ist "Assembler" die Bezeichnung für ein in Maschinensprache geschriebenes Programm. Wer weiß, was ein anderes Lexikon dazu sagt ? Bleiben wir bei dieser Begriffsbestimmung.

Maschinensprache gibt es nicht als solche. MS ist immer auf einen bestimmten Mikroprozessor bezogen, es gibt also 6502-MS, Z80-MS oder 68000-MS. Die MS enthält Befehle, die der

Prozessor direkt versteht, die Ihnen gegenüber aber an Anschaulichkeit zu wünschen übrig lassen. Das ist die eine Klippe. Die andere ist, daß Sie Maschinensprache nur anwenden können, wenn Sie sozusagen jedes Eckchen im Prozessor kennengelernt haben.

Programme in Maschinensprache sind zwangsläufig die schnellsten, die es überhaupt geben kann. Jeder Befehl, den Sie verfüttern, wird postwendend ausgeführt.

Wenn Sie sich aufs Selber-Programmieren stürzen, sollten Sie immer auch mit einem Auge auf Maschinensprache schielen. Schon beim Programmieren in BASIC können Sie kleine Maschinenroutinen (= Unterprogramme in Maschinensprache) einbauen. Die Beschäftigung mit PEEK und POKE bringt Sie ebenfalls auf die richtige Fährte. Außerdem gibt es gute Bücher zum Thema (MASCHINENSPRACHEBUCH zum C64 herausgegeben von DATA BECKER) !

Computer-Benutzer, die sich Ihre Anwendungsprogramme - in welcher Sprache auch immer - selber schreiben, stehen eigentlich immer vor der Frage, ob sich der zeitliche Aufwand für die Programmentwicklung lohnt. Natürlich: mit fortschreitender Erfahrung bekommt man Routine (und entwickelt Routinen, also Programmscheibchen, die immer wieder in neue Programme eingebaut werden) und der Zeitaufwand sinkt. Doch bis dahin braucht es ein paar Wochen/Monate/Jahre (Nichtzutreffendes Streichen !!). Wollen Sie Ihren COMMODORE 64 vom ersten Tag voll einspannen, sollten Sie zweispännig fahren: Fertig-Software fürs alltägliche Benutzen und selbstentwickelte Programme zum Spaß, zum Lernen und späteren Nutzen.

3. Texten, Drucken und Versenden

Längere Vorreden zum ersten Block Programm-Ideen können wir uns schenken. In Zweifelsfall blättern Sie bitte zum Kapitel 2.1 zurück.

Nur über das Thema "Monitor" möchte ich ein paar Sätze verlieren. Wenn Sie mit Ihrem 64er vorwiegend Textbearbeitung treiben wollen, empfehle ich Ihnen, unbedingt zu einem monochromen Monitor zu greifen. Besonders bei den gängigen Farben Grün bzw. Amber (=bernsteinfarbig oder einfach gesagt: orange) auf Schwarz vermeiden Sie Augenbeschwerden. Die Schärfe ist besser und Sie werden durch Farbenzauber nicht vom Eigentlichen angelenkt.

Einige Text-Programme bieten die Möglichkeit, immer neue Farbkombinationen einzustellen. Ich stelle z.B. meinen TEXTOMAT fast täglich anders ein; momentan schreibe ich Gelb auf Weinrot. Probieren Sie einfach aus, was Ihnen am besten gefällt.

Ein Wort noch zu den Angaben über die Zutaten: es werden immer die Mindest-Anforderungen genannt. Gerade bei der Textverarbeitung lassen sich generelle Empfehlungen zum Drucker nicht machen. Es gilt: kein Drucker druckt so schön, daß ein anderer nicht noch schöner druckt. Besonders die Frage "Typenraddrucker oder nicht ?" ist im wesentlichen ein Problem des Geldbeutels.

3.1 Gedichte vom Computer

```
*****
*      MAN NEHME      :      *
*****

*****
*      HARDWARE      :      *
*****
*
*      COMMODORE 64      *
*      Datasette      *
*      Monochromer Monitor      *
*
*****

*****
*      SOFTWARE      :      *
*****
*
*      keine      *
*
*****

*****
*      VORKENNTNISSE      *
*****
*
*      Commodore-BASIC-Kenntnisse      *
*
*****
```

Also, mit künstlicher Intelligenz (=Artificial Intelligence; ein Ausdruck mit Zukunft !) will ich Sie nicht behelligen.

Damit wäre der C64 auch überfordert. Bei der hier vorgeführten Programm-Idee handelt es sich oberflächlich betrachtet um eine harmlose Spielerei. Ernsthafter Hintergrund ist das Einüben vom Umgang mit Strings (=alphanumerische Zeichenketten; in BASIC das, was innerhalb von " " steht).

Den Begriff "Gedicht" schränken wir ein auf solche, die sich am Ende reimen. Ausgangsidee ist, daß in den meisten Fällen Wörter, die sich reimen, in ihrem hinteren Teil gleiche Zeichenfolgen haben. In der ersten Version kümmern wir uns nur um solche Reime.

Auf gut COMMODORE-BASIC ist der hintere Teil eines Wortes der

```
RIGHT$(A$,X)
```

wobei A\$ das ganze Wort bezeichnet und X die Anzahl der gemeinten Zeichen; also:

die letzten X Zeichen des Wortes A\$.

Probieren wir: A\$="Tassen", für einen Reim interessant ist das Stück "assen". Das sind die letzten 5 Zeichen von A\$. Den Reim finden wir ganz leicht. Vor den RIGHT\$(A\$,5) setzen wir alle Buchstaben des Alphabets und suchen uns die heraus, die einen Sinn machen. Das Programm-Listing sieht so aus.

```
100 PRINT CHR$(147)
110 A$="TASSEN"
120 PRINT A$
130 B$=RIGHT$(A$,5)
140 FOR Z=65 TO 90
150 :   C$=CHR$(Z)
160 :   D$=C$+B$
170 :   PRINT D$
180 NEXT Z
```

Allzu simpel, mögen Sie denken, aber in diesem 9-Zeilen-Programm steckt die Grundidee für ein vollelektronisches Reimlexikon. Auffallend ist beim Ablauf des Programms, daß ziemlich sinnlose Wortgebilde entstehen. Also fügen wir spaßeshalber eine Abfrage ein. Dazu noch ein paar kleine Verbesserungen. Es entsteht:

```
090 NR=1
100 PRINT CHR$(147)
110 AW$="TASSEN"
120 PRINT AW$
130 HI$=RIGHT$(AW$,5)
140 FOR Z=65 TO 90
150 :   C$=CHR$(Z)
160 :   VO$=C$+HI$
170 :   PRINT VO$
180 :   INPUT "SINNVOLL ";IN$
190 :   IF ASC(IN$)=74 THEN GOSUB 300
200 NEXT Z
210 PRINT CHR$(147)
220 FOR Z=1 TO NR
230 :   PRINT RW$(Z)
240 NEXT Z
299 END
300 REM UNTERPROGRAMM
310 RW$(NR)=VO$
320 NR=NR+1
330 RETURN
```

Erklärung

110 - 130 : Der Teil von A\$, der sich reimen soll,

wird abgetrennt

140 - 200 : In einer Schleife werden die Buchstaben
des Alphabets vorangestellt und gefragt, ob
die so entstandene Folge sinnvoll ist;
falls ja, wird zum Unterprogramm verzweigt

210 - 240 : Die als sinnvoll ausgewählten
299 : Programmende

300 - 330 : Im Unterprogramm werden die als sinnvoll
ausgesuchten Kombinationen in ein Feld
aufgenommen

Das sieht schon etwas besser aus. Ist aber noch nicht befriedigend. Es können beispielsweise solche Wörter, die mit mehreren Konsonanten beginnen weder mit sinnvollem Resultat eingegeben werden, noch werden derartige Wörter angeboten. Probieren Sie mal "Schnabel".

Das Problem läßt sich leider nur mit DATA-Zeilen lösen, die alle sinnvollen Konsonanten-Kombinationen enthalten. Im Deutschen sind das:

BL, BR, CH, DR, FL, FR, GL, GR, KL, KN, KR,
PF, PL, PR, SP, ST, TR, SCH, STR, SCHN, SCHW

Tests ergeben, daß auch QU dazu gehört. Wir haben mit dieser Auswahl das Programm schon ganz schön verbessert. Noch ein paar bedienungsfreundliche Elemente gehören dahin und schon sieht unser ehemals so bescheidenes Listing ganz eindrucksvoll aus:

```

10 rem *****
20 rem *      *
30 rem *   reim   *
40 rem *      *
50 rem *****
70 rem
80 nr=1 : rem zaehlvariable reimwoerter
90 a =1 : rem variable fuer aw# zustand
100 rem
200 print ""
210 input "ausgangswort"; aw#
220 :      rem aw# analysieren

```

```

230 gosub 1000
240 gosub 2000
250 l=len(aw$)
260 hi$=right$(aw$,l-a)
280 rem
290 rem
300 rem *****
310 rem * reimvorschlaege *
320 rem *****
330 rem
340 for z=65 to 90
350 :   c$=chr$(z)
360 :   gosub 3000 : rem vorzeichen
370 :   if a<>0 then goto 420
420 next z
430 rem
440 rem
450 for z=1 to 22
460 :   read do$
470 :   if do$="sc" then c$="sch"
480 :   if do$<>"sc" then c$=do$
490 :   gosub 3000 : rem vorzeichen
500 next z
510 goto 6000
520 end
600 if nr=11 then 700
610 rem *****
620 rem * reimwoerter liste *
630 rem *****
640 rem
650 rem
660 rw$(nr)=vo$
670 nr=nr+1
690 return
810 data b1,br,ch,dr,f1,fr
820 data g1,gr,k1,kn,kr',pf
830 data p1,pr,qu,sc,sp,st,tr
840 data str,schn,schw
850 data 65,69,73,79,85,89
1000 rem
1010 rem *****
1020 rem * u-pr:doppelkonsonant *
1030 rem *****
1040 rem
1050 rem

```

```

1060 for z=1 to 22
1070 :   read do$ : d=len(do$)
1080 :   if left$(aw$,d)=do$ then a=d
1090 next z
1110 return
2000 rem
2010 rem *****
2020 rem * u-pr:vokal am anfang *
2030 rem *****
2040 rem
2050 co=asc(aw$)
2060 for v=22 to 27
2070 :   read vo
2080 :   if co=vo then a=0
2090 next v
2100 restore
2110 return
3000 rem
3010 rem *****
3020 rem * unterprogramm : aussuchen *
3030 rem *****
3040 rem
3050 rem
3060 vo$=c$+hi$ : if vo$=aw$ then 3110
3070 print vo$,
3080 input "sinvoll";in$ : rem abfrage
3090 if asc(in$)=74 then gosub 600
3100 if asc(in$)=133 then goto 6000
3110 return
6000 rem
6010 rem *****
6020 rem * reimwoerter ausgeben *
6030 rem *****
6040 rem
6050 rem
6060 print ""
6070 print "ihr ausgangswort",aw$
6080 print : print
6090 print "meine vorschlaege : "
6100 print : print : print ;
6110 for z=1 to 10
6120 :   print rw$(z)
6130 next z
6140 end

```

Die Erklärungen:

- 0010 - 0090 : Programm-Titel;
Variablen werden vorgestellt
- 0100 - 0290 : Eingabe des Ausgangswortes
- 0300 - 0600 : Reimvorschläge werden gemacht
- 0610 - 0690 : Die Reimwörterliste
- 0810 - 0850 : DATA-Zeilen mit Mehrfach-Konsonanten
und Vokalen
- 1000 - 1110 : Unterprogramm :
Mehrfachkonsonant am Ausgangswortbeginn ?
- 2000 - 2110 : Unterprogramm :
Vokal am Ausgangswortbeginn ?
- 3000 - 3110 : Unterprogramm :
Sinnvolle Vorschläge auswählen
- 6000 - 6180 : Ausgabe der Reimwörterliste
auf dem Bildschirm
- 6190 : Programmende

Sie sehen: das Programm ist weder gut strukturiert, noch fertig. Es sollte doch Z.B. möglich sein zu verhindern, daß Wortmonstren wie "IIGEL" überhaupt angeboten werden. Eine feine Verbesserung wäre auch ein Programmmodul, daß eingegebene Wörter so auseinandernimmt, daß nur die reimrelevanten Silben übrigbleiben: Ausgangswort "STIEFELSCHRANK", reimrelevanter Teil "-ANK". Aber das überlasse ich alles Ihnen. Viel Spaß.

3.2 Einladung zur Party

```
*****  
*      MAN NEHME      :      *  
*****
```

```
*****  
*      HARDWARE      :      *  
*****  
*      *      *  
*      COMMODORE 64      *  
*      Datasette, besser Floppy 1541      *  
*      Monochromer Monitor      *  
*      Matrix-Drucker      *  
*      *      *  
*****
```

```
*****  
*      SOFTWARE      :      *  
*****  
*      *      *  
*      keine      *  
*      *      *  
*****
```

```
*****  
*      VORKENNTNISSE      *  
*****  
*      *      *  
*      Gute Kenntnisse in      *  
*      COMMODORE-BASIC      *  
*      *      *  
*****
```

"Hurra, wir feiern eine Riesenparty, eine Fete, wie sie im Buche steht !!"

"Und wen laden wir ein ?"

"Warte mal: Ernst und Renate, Holger und Eva, die Butzmanns mit Sohn ..."

"Halt, das machen wir aber mal systematisch. Hol mal nen Zettel."

"Sag mal, wo wohnt Erwin jetzt eigentlich ?"

Später:

"Puh, das sind ja über hundert Leute, wenn die alle einladen wollen, gibts aber ne Menge Schreibkram."

Halt !!

Wozu haben Sie denn einen COMMODORE 64 im Haus ? Das Schreiben von vielen, gleichlautenden Briefen ist doch geradezu die klassische Anwendung für einen Homecomputer.

Eine Frage: Haben Sie schon Ihr privates Adressbüchlein im Computer ? Wenn ja, können Sie den folgenden Teil überschlagen.

3.2.1 Eine einfache Adressen-Datei

Vorab: das Programm ist primitiv und unkomfortabel. Nachdem gleichen Prinzip, aber wesentlich bedienungsfreundlicher, dazu noch prima strukturiert und erklärt, arbeitet ein Adressdatei-Programm, daß in dem DATA BECKER Buch "64 für Einsteiger" beschrieben ist. Module wie "Adressen löschen" und "Adressen ändern" sind dort abzugucken. In unserem Progrämmchen verzichten wir darauf.

Die 1. Frage: Wieviel Adressen sollen erfaßt werden ?

Die 2. Frage: Wieviele Felder enthält eine Adresse ?

Wir wählen als Anzahl der Datensätze 100. Ein Datensatz, wie wir ihn in unserem Beispiel brauchen muß folgende Felder enthalten:

- Vorname
- Nachname
- Strasse/Nr.
- PLZ/Ort
- mehr nicht

Macht 4 Felder pro Datensatz. Einen Datensatz nennen wir im Programm DS\$. Alles zusammen ergibt die Programmzeile:

```
100 DIM D$(100,4)
```

Als erstes schreiben wir uns eine Eingaberoutine:

```
100 DIMD$(100,4)
200 REM EINGABE
210 Z=1
220 INPUT "VORNAME(N)"; D$(Z,1)
230 INPUT "NACHNAME(N); D$(Z,2)
240 INPUT "STRASSE/NR.; D$(Z,3)
250 INPUT "PLZ/ORT ; D$(Z,4)
260 Z=Z+1
270 PRINT : INPUT "MEHR EINGABEN"; A$
280 IF A$="J" THEN 220
290 END
```

Kurze Erklärung: Z zählt die eingegebenen Datensätze und wird nach jeder Eingabe um 1 erhöht. Die Eingabe der Felder 1 - 4 ist in der Länge auf 255 Zeichen begrenzt, je kürzer Sie sich fassen, desto schneller geht es später beim Einlesen. Ein Gag am Rande, der in vielen Adressprogrammen fehlt: Sie können im Feld 1 mehrere Vornamen eingeben und so alle Familienmitglieder persönlich ansprechen; im Feld 2 sind ebenfalls mehrer Namen erlaubt, es können ja auch zwei

Leute mit verschiedenen Vornamen in einer Wohnung leben. Das "END" in Zeile 290 ist nur vorläufig, hier kommt gleich der Block "Daten speichern" hin.

Das Programm-Modul "Daten speichern" ist für den Gebrauch einer Datasette geschrieben, die notwendigen Änderungen beim Betrieb mit Floppy liefere ich nach.

```
290 REM SPEICHERN
300 OPEN 1,1,1,"ADR.DATEI"
310 FOR N=1 TO Z
320 :   FOR M=1 TO 4
330 :       PRINT(NR.)1,D$(N,M)
340 :   NEXT M
350 NEXT N
360 CLOSE 1
```

Der OPEN-Befehl sollte Ihnen nicht unbekannt sein. Falls doch THEN GOTO "Einsteigerbuch". Die Schleife N zählt die Datensätze, die M-Schleife die Felder. (Das (NR.) am PRINT-Befehl steht für das amerikanische Nummern-Symbol, das meinem Textprogramm nicht gleichzeitig mit deutschen Umlauten zu entlocken ist.) Für die Floppy sieht der OPEN-Befehl so aus:

```
300 OPEN 1,8,2,"ADR.DATEI,S,W"
```

Gut, ich würde sagen, fangen Sie an mit dem Eingeben Ihrer Adressen; ich gehe mir frischen Kaffee holen. Stop, vergessen Sie nicht beim Speichern auf Datasette das Band zurückzuspulen !

Wollen Sie Ihre Daten wiederhaben, brauchen Sie eine Routine, die sich nur in zwei Dingen vom Eingabeprogramm unterscheidet: der OPEN-Befehl muß heißen:

```
400 OPEN 1,1,0,"ADR.DATEI"
```

Und wo in Zeile 330 PRINT(NR.) steht gehört jetzt

```
430 :          INPUT(NR.),D$(N,M)
```

hin. Probieren Sie es aus. Sie sehen, Sie sehen nichts. Die Adressen sind zwar geladen, aber niemand hat Ihnen gesagt, daß Sie auf dem Bildschirm erscheinen sollen. Dies Anliegen teilen wir dem Computer in einer kleinen Schleife mit.

```
490 REM AUSGABE/BILDSCHIRM
500 FOR N=1 TO Z
510 :   FOR M=1 TO 4
520 :       PRINT D$(N,M)
530 :   NEXT M
540 :   PRINT : PRINT
550 NEXT N
```

Sie merken schon, man macht dauernd das gleiche. Die N/M-Schleifen für Speichern, Laden und Ausgeben sind identisch, bis auf das Befehlswort für die Ausführung. Außerdem sind die OPEN-Befehle anders. Wenn wir bedenken, daß beim Programmteil "Daten auf Drucker ausgeben" die Schleiferei wieder losgeht, sind wir für Verbesserungen motiviert.

Wie wäre es, wenn die einzelnen Programm-Module nicht einfach nacheinander ablaufen würden ? Schön wäre das. Also entwerfen wir ein MENUE.

```
90 REM MENUE
100 PRINT "MENUE"
110 PRINT : PRINT : PRINT
120 PRINT "ADRESSEN EINGEBEN   = 1"
130 PRINT "ADRESSEN SPEICHERN = 2"
140 PRINT "ADRESSEN LADEN     = 3"
150 PRINT "ADRESSEN DRUCKEN   = 4"
160 PRINT "PROGRAMM BEENDEN   = 5"
170 PRINT : PRINT : PRINT
```

180 INPUT "IHRE WAHL";W

So ein Menue macht immer ordentlich was her und ist leicht zu programmieren. Einige Kommentare dazu: "ADRESSEN DRUCKEN" bedeutet auch Ausgabe am Bildschirm. einen Punkt "PROGRAMM BEENDEN" nehme ich immer dann in ein Programm auf, wenn ich erreichen will, daß ein Notausgang bleibt, bei dessen Betreten keine Daten oder gar das Programm verloren gehen.

Ohne viel darüber zu sprechen, habe ich jetzt noch ein paar Verschönerungen angebracht und ganz alleine das Unterprogramm in Zeile 500 geschrieben. Das war Ihnen doch recht ? Das folgende Listing enthält das lauffähige Programm für den Floppy-Betrieb.

Wenn Sie mit der Datasette arbeiten, verwenden Sie die oben beschriebenen Programm-Module fürs Speichern und Laden. Sie sehen, daß das Einlesen der Daten von der Diskette nicht ganz so einfach ist. Es muß die interne Statusvariable "st" abgefragt werden. Sie zeigt das Ende der Datei an. Würde man einfach die normale Schleife (N,M) verwenden, wüßte das Floppy-Betriebssystem nicht, wo das Ende der Datei ist und bliebe irgendwann ratlos stehen. Dabei wird noch nicht einmal ein Fehler angezeigt.

Das beschriebene Verfahren ist also vorzuziehen. Es ist außerdem universell für alle Arten sequentieller Dateien anwendbar.

Jetzt aber endlich das Listing:

```

10 rem * * * * *
20 rem *   adress - datei   *
30 rem * * * * *
40 rem
50 dimds$(100,4)
60 poke 53281,08 : poke 53280,08
70 poke 53272,23 : print chr$(158)
80 rem
90 rem
100 print "" : print "MENUE" : print
110 print : print
120 print "Adressen eingeben = 1"
130 print "Adressen speichern = 2"
140 print "Adressen laden = 3"
150 print "Adressen ausgeben = 4"
160 print "Programm beenden = 5"
170 print : print
180 input "ihre wahl 3";w
190 on w gosub 200,300,600,400,700
195 goto 100
200 rem eingabe
210 print ""
220 z=z+1 : print "Nummer :      ";z
230 input"Vorname(n)      ";d$(z,1)
240 input"Nachname(n)    ";d$(z,2)
250 input"Strasse/Nr.    ";d$(z,3)
260 input"Plz/Ort       ";d$(z,4)
270 print : print : input"Mehr Eingaben";a$
280 if a$="j" then 220
300 print""
310 open 1,8,2,"s:adr.datei,s,w"
340 for n=1 to z
350 :   for m=1 to 4
360 :     print#1,d$(n,m)   f
370 :   next m
380 next n
390 close 1
395 return
400 print ""
410 input"Drucker oder Bildschirm (d/b) ";a$
420 if a$="d" then open 4,4,2 : cmd 4
430 print ""

```

```

440 for n=1 to z
450 :   for m=1 to 4
460 :     print d$(n,m)
470 :   next m
480 :   print : print
490 next n
500 if a$<>"d" then 520
510 close 4
520 get a$ : if a$="" then 520
530 goto 100
600 open 1,8,2,"adr.datei,s,r"
610 n=1
620 input#1,d$(n,1),d$(n,2),d$(n,3),d$(n,4)
630 if st<>64 then n=n+1 : goto 620
640 close1
650 z=n
660 return
700 rem ende
710 print ""
720 print "e n d e"
730 for i=1 to 1000 : next i
740 end

```

Noch einige wenige Kommentare:

Die Farbe des Bildschirms wird bekanntlich mit den beiden POKEs in Zeile 60 gesetzt, das POKE in der nächsten Zeile bewirkt, daß auf Groß/Klein-Schrift umgeschaltet wird.

Anstelle der Warteschleife in Zeile 730 kann auch eine Sicherheitsabfrage eingebaut werden:

```

720 input "Soll wirklich beendet werden (j/n) ";in$
725 if g$="n" then 100
730 end

```

Wenn die Frage verneint wird, gehts zurück ins Menue.

So, jetzt gucken wir uns eine dazu passende Textverarbeitung an.

3.2.2 Ein ganz einfaches Textprogramm

das kurz und bündig, unkomfortabel und einfach ist; mit einem Wort: simpel. Daher auch der schöne Name für das Programm:

*** SIMPEL-TEXT 64 ***

Mit SIMPEL-TEXT können Sie Texte schreiben, die maximal 10 Seiten zu höchstens je 20 Zeilen umfassen. Ihre Texte können Sie korrigieren, anschauen, auf Kassette oder Diskette speichern bzw. laden, sowie das Ganze drucken. SIMPEL-TEXT enthält eine Schnittstelle (=Verbindung) zu dem oben vorgestellten Adress-Programm. SIMPEL-TEXT ist modular aufgebaut. Dadurch haben Sie die Chance mit einiger Tüftelei noch etwas daraus zu machen. Verbesserte Versionen nehme ich gerne entgegen.

```
10 rem * * * * * programm beginn * * * * *
20 poke 53280,0 : poke 53281,0
30 poke 53272,23 : print chr$(30 )
35 dim t$(200)
40 print chr$(147);
50 print " * * * * * "
60 print "*"
70 print "*      simpel - text 64      *"
80 print "*"
90 print " * * * * * "
99 rem *****
100 rem * menu *
101 rem *****
110 print : print : print
120 print "Text eingeben      = 1"
140 print "Text korrigieren = 2"
150 print "Text speichern   = 3"
160 print "Text laden       = 4"
170 print "Text zeigen      = 5"
180 print "Text drucken    = 6"
190 print : print "Ende              = 7"
200 print : print : print
210 input "Ihre Wahl      1";w$
220 w=val(w$) : if w>0 and w<8 then 240
230 print "" : goto 220
240 on w gosub 300,600,800,1000,1200,1300,9000
250 goto 040
299 rem *****
300 rem * eingeben *
301 rem *****
```

```

310 clr : print chr$(147);
320 print "";
330 input "Name (max.15 Zeichen) ";tn$
340 if len(tn$)>15 then print "" : goto 330
350 input "maximale Zeilenlaenge 40";zm
360 if zm<10 or zm>40 then print "" : goto 350
370 print "Eine Seite hat maximal 20 Zeilen "
380 input "maximale Seitenzahl 10";sm
390 if sm<1 or sm>10 then print "" : goto 380
400 bl=1 : zl=1 : sp=1
420 print chr$(147); : print "Seite: ";spc(7);"Zeile: ";spc(8);"Spalte:"
430 print "#####" : print;
440 print "F1 = ""Neue Zeile/Neue Seite ";"F2 = ""Abbrechen";
450 p1=1030 : p2=1043 : p3=1058
460 poke p1,(int(bl/10))+48 : poke p1+1,(bl-(int(bl/10)*10))+48
470 poke p2,(int(zl/10))+48 : poke p2+1,(zl-(int(zl/10)*10))+48
480 poke p3,(int(sp/10))+48 : poke p3+1,(sp-(int(sp/10)*10))+48
490 print chr$(95);chr$(157);
495 get in$ : if in$="" then 495
500 if in$=chr$(137) then 590
505 if in$<>chr$(20) then 525
510 print chr$(32);chr$(157); : sp=sp-1
515 print chr$(157);chr$(32);chr$(157);
520 t$(zl)=left$(t$(zl),len(t$(zl))-1) : goto 450
525 if in$=chr$(133) then 560
530 if in$=chr$(13) then 550
535 t$(zl)=t$(zl)+in$
540 print in$;
545 sp=sp+1 : if sp<zm then 450
550 print chr$(13); : t$(zl)=t$(zl)+chr$(13)
555 zl=zl+1 : sp=1 : if zl<20 then 450
560 print : print : print " *** Seite voll ***"
570 get g$ : if g$="" then 570
580 bl=bl+1 : if bl<sm then 450
590 goto 40
599 rem *****
600 rem * korrigieren *
601 rem *****
610 print chr$(147);"";
620 print "";
630 input "Welche Seite 1";in$
640 bl=val(in$) : if bl>20 then print "" : goto 630
645 print chr$(147); : print "Text: ";tn$; : print " Seite: ";in$
650 zl=((bl-1)*20)+1
655 print t$(zl)
660 if t$(zl)<>"" then zl=zl+1
670 if t$(zl)="" or zl>20 then 690
680 goto 655
685 print : print
690 input "Welche Zeile ";in$

```

```

695 in=val(in$)
700 if in=0 then 790
705 z1=((b1-1)*20)+in : k$=""
710 poke 214,14 : sys 58640
720 print t$(z1) : print
725 print chr$(95);chr$(157);
730 get in$ : if in$="" then 730
735 if in$=chr$(137) then 785
740 if in$<>chr$(20) then 760
750 print chr$(32);chr$(157);
755 goto 725
760 if in$=chr$(133) then 780
765 k$=k$+in$
770 print in$;
775 goto 725
780 t$(z1)=k$
785 print
790 input "Weitere Korrekturen (j/n)   n";in$ : if in$="j" then 600
795 return
799 rem *****
800 rem * speichern *
801 rem *****
805 u$=""
810 print chr$(147);
820 input "Laufwerk aufnahmebereit   n";in$
830 if in$="n" then print "" : goto 820
840 print " ";tn$;" wird gespeichert "
850 open 1,8,2,u$+tn$+"s,w"
860 for n=1 to z1
870 :   print#1,t$(n)
880 next n
890 close1
900 rem
910 rem
915 open 15,8,15
920 input#15,ff,fb$,sp,se
925 close15
930 if ff=63 then u$="S:" : goto 850
940 print "Speichern/Laden beendet" : for i=1 to 1000 : next i
950 return
999 rem *****
1000 rem * laden *
1001 rem *****
1010 clr : print chr$(147);
1020 input "Name (max.15 Zeichen) ";tn$
1040 print " ";tn$;" wird geladen "
1050 open 1,8,2,tn$+"s,r"
1060 z1=1
1070 input#1,t$(z1)
1080 if st<>64 then z1=z1+1 : goto 1070
1090 close 1

```

```

1100 print "Speichern/Laden beendet" : for i=1 to 1000 : next i
1110 goto 40
1199 rem *****
1200 rem * zeigen *
1201 rem *****
1210 print chr$(147);"";
1220 input "Welche Seite 1";in$
1225 bl=val(in$) : if bl>20 then print "" : goto 1220
1230 print chr$(147); : print "Text: ";tn$; : print " Seite: ";in$
1235 print
1240 z1=((bl-1)*20)+1
1250 print t$(z1)
1255 if t$(z1)<>" " then z1=z1+1
1260 if t$(z1)="" or z1>20 then 1280
1270 goto 1250
1280 input "Weitere Seiten (j/n) n";in$ : if in$="j" then 1200
1290 return
1299 rem *****
1300 rem * drucken *
1301 rem *****
1310 print chr$(147);
1320 print "Es wird der aktuelle Text mit dem Namen"
1330 print " ";tn$;" ";" gedruckt."
1340 print
1350 input "Drucker fertig n";in$
1360 if in$="n" then print "" : goto 1350
1370 open4,4,2
1380 z1=1
1390 if t$(z1)="" then 1440
1410 print#4,t$(z1)
1420 z1=z1+1 : goto 1390
1430 print#4
1440 close4
1450 return
8999 rem *****
9000 rem * ende *
9001 rem *****
9010 print ""
9020 print "*** E N D E ***"
9030 print : print : print
9040 input "Text gespeichert n";in$
9050 if in$="n" then 800
9060 end
9999 rem * * * * * programm ende * * * * *
50000 open15,8,15
50100 input#15,ff,fb$,sp,se
50200 close15
50300 printff;fb$;sp;se

ready.

```

3.3 Diplomarbeit - professionell gestaltet

```
*****
*      MAN NEHME      :      *
*****

*****
*      HARDWARE      :      *
*****
*
*      COMMODORE 64      *
*      Floppy 1541      *
*      Monochromer Monitor      *
*
*****

*****
*      SOFTWARE      :      *
*****
*
*      Textverarbeitung (z.B. TEXTOMAT) *
*
*****

*****
*      VORKENNTNISSE      *
*****
*
*      Erfahrung im Umgang mit der      *
*      C64-Tastatur und der Floppy      *
*
*****
```

Um gar keine Mißverständnisse aufkommen zu lassen: ersetzen Sie "Diplomarbeit" nach Belieben durch die Bezeichnung für

die schriftliche Arbeit, mit der Sie zu tun haben. Das muß ja kein wissenschaftlicher Erguß oder ein Referat für den Oberstufenunterricht sein. Auch der Reisebericht von der letzten Kegeltour kann umfangreich genug sein für den Einsatz eines (halb-)professionellen Textverarbeitungs-Programmes. Aus eigener Erfahrung kann ich nur prophezeien: Sie werden sich so sehr an das Arbeiten mit Text-Software gewöhnen, daß Sie sich bald nicht mehr vorstellen können, wie Sie sowas eigentlich früher gemacht haben.

Dieses Kapitel dient nicht dazu, von A bis Z in allen Einzelheiten zu erklären, wie die Handgriffe bei der Arbeit mit einem Textverarbeitungsprogramm ablaufen. Also keine Sätze a la "nehmen Sie die Programmdiskette in die linke Hand und öffnen Sie mit der rechten vorsichtig den Laufwerkschacht"; das entnehmen Sie dem Handbuch Ihres Programmes. Vielmehr möchte ich ein paar Tips und Tricks verraten, wie man einerseits denkbare Probleme überwindet und sich andererseits die Arbeit erleichtern kann. Wo es ins Detail geht, ist TEXTOMAT (von DATA BECKER) gemeint, denn mit dem Programm habe ich dieses Buch geschrieben, da kenne ich mich am besten aus. Die meisten Hinweise sind jedoch auf die gängigsten Text-Systeme für den 64er übertragbar.

Eingewöhnungsphase

Wenn Sie zum ersten Mal die Programmdiskette ins Laufwerk gelegt haben, werden Sie - sofern Sie das Handbuch gründlich studiert haben - über die verschiedenen Fragen, die Sie zu beantworten haben, nicht sehr überrascht sein und alles leicht bewältigen. Sehen Sie zu, daß Sie schnell in den Schreibmodus kommen und los gehts. Tippen Sie einfach drauf los.

Bevor Sie ernsthaft zu schreiben beginnen, sollten Sie mit den Funktionen des Programms vertraut sein. Testen Sie:

- Einzelne Zeichen korrigieren
- Ganze Zeilen löschen
- Neue Zeilen einfügen
- Im Text herumfahren
- Steuerzeichen benutzen

Halt, Steuerzeichen ist ein guter Übergang zum nächsten Übungs-Block. Die Steuerzeichen dienen dazu, das spätere Aussehen des Textes zu bestimmen. Wenn Sie TEXTOMAT-User sind, können Sie nun das Formatieren trainieren. Einen geschriebenen und formatierten Text müssen Sie nicht jedesmal ausdrucken, die Funktion "Zeigen" erlaubt, das Geschriebene so auf den Monitor zu bringen, wie es später auch auf Papier wird. Die wichtigsten Angaben zum Formatieren kommen ins "Formular". Sie wissen ja schon aus dem Kapitel 2.1, welche Entscheidungen da auf Sie zukommen. Mein Tip dazu: probieren Sie die verschiedensten Kombinationen aus. Jedes Formular können Sie separat abspeichern und wieder verwenden.

Schwierigkeiten kann es zum Beispiel geben, wenn Sie häufig mit Einrückungen arbeiten. Zeilen, die mit dem Absatzzeichen beendet wurden, eine größere Anzahl Leerzeichen enthalten und länger sind als die im Formular festgelegte Zeilenlänge, können im Modus "Zeigen" dazu führen, daß der Textspeicher völlig durcheinandergerät. Statt der gewünschten Zeilen sehen Sie dann eine merkwürdig verstümmelte Zeile. Mir widerfuhr dies Schicksal einige Male mit den schicken Tabellen in diesem Buch. Wenn dann das Zeigen nicht rechtzeitig angehalten wird, stürzt das Programm ab. Sehr ärgerlich. Als mir das zum ersten Male passierte, wußte ich überhaupt nicht, was los war. Erst beim dritten oder vierten Crash (=Programm-Zusammenbruch) wurde mir der Zusammenhang klar. Provozieren Sie diesen Fall ruhig einmal, dann sind Sie auf solche Dinge vorbereitet.

Als nächstes befassen Sie sich mit dem Drucker. Vielleicht haben Sie sich das Gerät eigens für die bevorstehende Diplom- (oder was-auch-immer-) Arbeit angeschafft, dann ist es sogar am einfachsten, die Handhabung des Druckers mit einem Textverarbeitungsprogramm zu lernen. ABER, aber nur wenn Sie in der glücklichen Lage sind, ein Textsystem zu besitzen, bei dem Sie nicht - gleich welchen Drucker Sie haben - eine Tabelle anlegen müssen. Der TEXTOMAT ist softwaremäßig komplett vorbereitet für den COMMODORE Drucker VC 1525 (bzw. den GP 100 VC), falls Sie so einen Ihr eigen nennen, brauchen Sie sich um nichts mehr zu kümmern. Auch beim VC 1526 und den kompatiblen haben Sie nicht viel Arbeit: Sie müssen nur die im Programm vorbereitete Tabelle abspeichern (siehe im Handbuch den Punkt "Dienstprogramme"). Für alle anderen Drucker gilt: jedem Zeichen des COMMODORE-Zeichensatzes muß der entsprechende Zeichencode Ihres Druckers zugeordnet werden. Zum Glück ist das eine einmalige Arbeit.

Zu dieser Einstellarbeit gehört auch das Ausfüllen einer Liste mit Steuerzeichen, das Festlegen der Sekundäradresse des Druckers und falls nötig das Anpassen einer RS 232-Schnittstelle (wenn Sie eine am 64er haben, wissen Sie schon, was gemeint ist !).

Wenn Sie das alles ausprobiert haben, kommen Sie zur

Vorbereitungsphase

Große Arbeiten müssen gut vorbereitet sein, das wissen Sie sicher genauso gut wie ich. Aber auch die eigentliche Textarbeit kann man nicht aus dem Ärmel schütteln und gegenüber der Arbeit mit Papier und Bleistift bzw. Schreibmaschine gibt es bei der elektronischen Textverarbeitung Unterschiede. Ich will Ihnen da auch keine Vorschriften machen. Grundsätzlich kenne ich zwei Arten von Textsoftware-Anwendern

- diejenigen, die die gesamte Schreibearbeit am Terminal erledigen
- und andere, die den Text konventionell per Hand schreiben und erst die endgültige Fassung mit dem Computer gestalten

Beide Auffassungen haben ihre Vor- und Nachteile: die Letztgenannten sparen sich die teilweise sehr anstrengende Tätigkeit vor der Mattscheibe; die anderen kommen mit viel weniger Papier aus und können sich jederzeit ein Bild davon machen (lassen), wie der Text später aussieht. Zu denen würde auch ich mich zählen. Nicht nur deshalb soll diese Methode hier im Vordergrund stehen. Es ist nämlich auch so, daß nur so die ganze Bandbreite der Möglichkeiten ausgenutzt wird.

Bevor Sie ans Schreiben gehen, haben Sie ja eigentlich Ihr komplettes Werk im Kopf, stimmt's ? Hoffentlich auch eine klare Gliederung, die kann Ihnen sehr hilfreich sein. Also schreiben Sie als erstes ein vorläufiges Inhaltsverzeichnis. Besonders clever ist es, dies schon in der endgültigen Form zu tun; also mit der richtigen Kapitelnummerierung, der Aufteilung des Blattes usw. Dieses Inhaltsverzeichnis wächst mit der Arbeit. Alle Änderungen von Kapitelüberschriften und Verschiebungen von Teilen werden immer umgehend ins Inhaltsverzeichnis übertragen. Im Idealfall haben Sie am Ende des Textes das Inhaltsverzeichnis druckfertig. Außerdem bietet so ein Verzeichnis eine gute Struktur für die Aufteilung und Benennung der Textdateien auf den Disketten.

Sie stellen sich doch sicher auch Ihre Stichwörter und Literaturstellen auf Karteikarten zusammen ? Haben Sie schon daran gedacht, das mit dem Computer zu machen ? Blättern Sie doch mal zum Kapitel 5.11 ! Sogar eine "Fußnoten-Datei" ist mehr als ein Bündel Notizen. TEXTOMAT z.B. hat eine Schnittstelle zu sequentiellen Dateien. Sie können Ihre Anmerkungen aus einer Datei direkt in den Text übernehmen.

Noch ein Hinweis: bleiben Sie soweit möglich bei Ihrer gewohnten Arbeitsweise, nutzen Sie den Computer als Werkzeug; sonst beschäftigen Sie sich am Ende mehr mit dem Gerät als mit Ihrer Arbeit.

Schreib- und Arbeitsphase

Wie Sie schreiben, welchen Rhythmus Sie dabei haben, ob Sie lieber nachts oder in Badehose schreiben, welches Radioprogramm Sie bei der Arbeit hören, bleibt Ihnen überlassen (ich höre meistens SWF3); aber was das Abspeichern angeht, würde ich furchtbar gerne täglich hinter Ihnen stehen und aufpassen, daß Sie auch oft genug Geschriebenes auf Eisenoxid bannen.

Ich spreche aus eigener Erfahrung: eines schlechten Tage hatte mich die Muse geküßt und meine Finger flitzten nur so über die Tasten, der Text hatte Stil, Klasse und (genug des Eigenlobs) ... jedenfalls saß ich von mittags zwölf bis gegen zwei Uhr morgens am Terminal und hatte nur die ersten 10 bis 12 Seiten schon abgespeichert. Und es kam, wie es kommen mußte: nachtblind, übermüdet und nicht mehr ganz beieinander setzte ich den letzten Punkt unter das Werk, betrachtete meine Leistung noch einmal unter der Funktion "Zeigen" und - schaltete, in dem festen Glauben alles wäre auf Diskette verewigt, den Computer AUS. Alles weg, Verzweiflung, Wut über den Computer und und. Danach habe ich mir drei Dinge angewöhnt:

- Textteile von jeweils rund 100 Zeilen werden abgespeichert;
- Textstücke von je 200 - 300 Zeilen ergeben, wenn es mit der Kapiteleinteilung ungefähr hinkommt eine Textdatei
 - am Ende einer Arbeitssitzung wird alles bis dahin gespeicherte auf eine zweite (Back-Up-)Diskette überspielt. Die Back-Up-Diskette bekommt nach jedem

Abspeichern einen Schreibschutz (das sind diese Klebestreifen, die über die Kerbe rechts-oben geklebt werden);

- alle Handgriffe während des Schreibens werden in der gleichen Reihenfolge ausgeführt.

Diese Angewohnheiten will ich noch kurz erläutern.

Die erstgenannte Speichermethode arbeitet überlappend, d.h., daß zuerst die ersten 100 Zeilen gespeichert werden, dann die nächsten 100 mit den ersten 100 zusammen unter dem gleichen Namen usw., da kann kaum noch was verloren gehen. Der Vorschlag zur Länge von Dateien ergibt sich aus zwei Umständen: erstens ergeben die genannten 300 Zeilen ungefähr 100 Blocks auf der Diskette, also rund 6 Dateien pro Diskette, das ist ein recht brauchbares Maß. Zum Back-Up, dem Sicherheits-Zusatz-Abspeichern, steht schon genug in Kapitel 2.1. Und zur festgelegten Reihenfolge der Tätigkeiten ist nur zu sagen, daß das eine Methode ist, Fehlbedienungen auszuschalten. Bei mir geht es so:

- Eröffnungszereemonie bis zum Schreibmodus
- Text schreiben, in gleichbleibenden Abständen speichern, weiter schreiben
- Speichern, Zeigen lassen, evtl. korrigieren und
- das gleiche nochmal bis es stimmt
- endgültig abspeichern
- ausdrucken
- Inhalt der Diskette kontrollieren
- wenn alles in Ordnung ist Back-Up durchführen
- Schreibschutz an Back-Up-Diskette anbringen
- Programm-, Text- und Back-Up-Diskette verstauen
- erst wenn das alles geschehen ist
- Feierabend !

"Ganz schön umständlich" werden Sie denken, aber was sich hier anhört, wie stundenlanges Diskettengeschlebe geht in Wirklichkeit ziemlich flott und, wenn man es sich angewöhnt hat, auch ohne großes Nachdenken. Jedenfalls ist es bequemer als einen langen Text rekonstruieren zu müssen. Ist Ihnen

übrigens aufgefallen, daß der bearbeitete Text bis zum Abschalten im Arbeitsspeicher verbleibt. Das ist ein wichtiger Sicherheitsfaktor.

Endphase

Der Text steht, Sie sind ganz zufrieden und wollen nun das Werk ausdrucken. Zwei Dinge sind nun bei den verschiedenen Textprogrammen sehr unterschiedlich: das Paginieren (=automatische Seitennummerierung) und das Einfügen von Fußnoten und Anmerkungen. Die Seitenzahl kann bei einigen Programmen über Drucker-Steuerzeichen durchgeführt werden oder aber zu Fuß durch das nachträgliche Eintragen der Seitenzahlen in die Textdateien (sehr frustrierend !!). Auch die Anmerkungen können von Hand eingesetzt werden, genialer aber ist die Methode, eine Fußnoten-Datei einzulesen. Beim TEXTOMAT geht das über das Steuerzeichen zum Verketteten von Dateien. Mit dieser Funktion können Texte, die sich auf der gleichen Diskette befinden müssen, nachgeladen und beim Drucken an den Haupttext angehängt werden. ACHTUNG : wollen Sie die Fußnoten auf dem Bildschirm sehen, müssen Sie darauf achten, daß die Gesamtlänge nicht den Platz im Arbeitsspeicher überschreitet.

Wenn Sie Ihren kompletten Text fix und fertig ausgedruckt (in mehreren Exemplaren !!) haben, können Sie die Früchte Ihrer Arbeit geniessen. Stop ! Eines sollten Sie unbedingt noch tun: um Ihr Werk der Nachwelt zu erhalten, sollte es nicht nur auf flüchtigem Papier existieren, sondern auch auf Diskette. Beginnen Sie also ein letztes Mal das ganz GROSSE ABSPEICHERN. Und zwar in folgender Reihenfolge:

- 1) Die Drucker-Parameter
- 2) Die Formulare in der Reihenfolge ihres Auftritts
- 3) Die Textdateien von 1. bis Ende

Das kann je nach Länge der Arbeit auch noch einige Stunden dauern, aber es ist der beste Weg, den Text so zu sichern, daß Sie noch nach Jahren, wenn alle gedruckten Versionen längst zu Staub geworden sind, das Werk korrekt zu rekonstruieren.

So, nun wünsche ich Ihnen eine gute Inspiration, frohes Schaffen und gute Noten ! Wenn Sie noch Fragen haben, schreiben Sie mir; für jede Kritik (und jedes Lob auch) und alle Anregungen bin ich dankbar.

3.4 Individuelle Werbebriefe

```
*****  
*      MAN NEHME      :      *  
*****
```

```
*****  
*      HARDWARE      :      *  
*****  
*      *      *  
*      COMMODORE 64      *  
*      Floppy 1541      *  
*      Monochromer Monitor      *  
*      Korrespondenz-Matrixdrucker      *  
*      *      *  
*****
```

```
*****  
*      SOFTWARE      :      *  
*****  
*      *      *  
*      Textverarbeitungsprogramm      *  
*      Datenverwaltungsprogramm      *  
*      *      *  
*****
```

```
*****  
*      VORKENNTNISSE      *  
*****  
*      *      *  
*      Kenntnisse im Umgang mit      *  
*      Tastatur und Floppy      *  
*      *      *  
*****
```

Schon wieder sollen Sie eine Rolle spielen, was Ihnen sicher nicht schwerfällt, wenn Sie dieses Kapitel als erstes aufgeschlagen haben: Sie sind ab sofort ein Spielwarenhändler, spezialisiert auf Modelleisenbahnen.

Übrigens: haben Sie wirklich mit diesem Kapitel begonnen ? Dann blättern Sie doch bitte zunächst zum Kapitel 2.1 zurück, Sie sparen mir und sich einige Wiederholungen.

In Ihrer neuen Rolle erfüllt die direkte Korrespondenz mit Ihrem festen Kundenstamm eine wichtige Funktion. Oft sind Sie durch fleissiges Recherchieren auf einen Posten ganz seltener Loks oder Oldtimer-Modelle gestossen. Sie wissen, einige Ihrer Kunden sind ganz scharf auf diese Stücke. Sie brauchen bloß noch die Interessenten anzuschreiben und schon ist Ihr Angebot verkauft. Mit konventionellen Methoden gibt es nur zwei Möglichkeiten:

- Jeder Kunde wird einzeln angeschrieben
(möglicherweise jeweils mit einem eigenen Briefftext)
- Ein Angebotsbrief wird geschrieben, kopiert und die Adressen von Hand eingefügt

Die erste Möglichkeit hat den Vorteil, daß Sie dem Adressaten genau das anbieten können, was er sucht. Dafür müssen Sie aber auch ziemlich viel tippen (lassen). Die zweite Variante hat einen gewissen Geschwindigkeitsvorteil. Wie schön wäre es, könnte man die Vorteile der beiden Methoden kombinieren ! Man kann !

Voraussetzung ist eine topfite Kunden-Datei. Darüber wollen wir uns zuerst informieren.

3.4.1 Eine multifunktionale Kunden-Datei

Mit dem DATA-BECKER DATAMAT entsteht diese Datei so.
 Zunächst die Eingabemaske (wie das genau geht, entnehmen Sie
 bitte dem Kapitel 2.3). Ich würde für unser Beispiel
 folgende Aufteilung vorschlagen:

Lukas-Spielwaren	Kundendatei
(1) Name	: .Schulze
(2) Vorname	: .Hans-Walter
(3) Anrede	: .Herr . .r .
(5) Strasse/Nr.:	.Bahnstr.11
(6) PLZ	: .4005.
(7) Ort	: .Meerbusch
(8) Land	: . . .
Kundenmerkmale	
(9) letzte Bestellung am	.13.02.84.
(12) letztes Angebot am	.29.06.83.
(15) offene Rechnung Nr.	. / .
(17) spezialisiert auf :	.Vorkriegs- .
	.Maerklin HO-Lokomotiven
(18) bietet an	: .Vorkriegs HO- .
	.Modelle aller Art

Vielleicht weckt das Feld 4) Ihre Neugier. Die Lösung ist
 einfach. Falls der Kunde ein Herr ist, erscheint hier ein
 "r", das bei Briefen die Floskel "Sehr geehrte"
 vervollständigt. Das Feld 8) wird nur ausgefüllt, wenn es
 sich um einen ausländischen Kunden handelt.

Der ganze Abschnitt "Kundenmerkmale" ist natürlich

branchenspezifisch. Für den Eisenbahn-Händler ist ja auch interessant, welcher Kunde eventuell gesuchte Stücke anbietet. Das Datum der letzten Bestellung bzw. des letzten Angebots kann später als Auswahlkriterium verwendet werden. Nun geht DATAMAT daran, die Datei auszurechnen. Die gezeigte Maske umfaßt 208 Zeichen, Indexfeld ist die 1), der Nachname, summarum: 432 Datensätze sind das Maximum für unsere Kundendatei. Damit kommen wir nicht weit (bei angenommenen 2000 Kunden), also legen wir mit der gleichen Maske 5 Dateien zu je 400 Sätzen an. Die Aufteilung der Kunden auf die verschiedenen Dateien wollen wir der Einfachheit halber nach dem Alphabet vornehmen:

- A bis F;
- G bis L;
- M bis R;
- S, St und Sch;
- und der Rest T bis Z.

Im alltäglichen Betrieb wird sich diese Aufteilung möglicherweise als unpraktisch erweisen. Es kann sein, daß eine Aufteilung nach den Interessensgebieten für Ihre Zwecke sinnvoll ist, aber das läßt sich ändern.

Das Auswählen von Kunden nach deren besonderen Wünschen werden wir sowieso noch erledigen. Jetzt gehts erstmal ans Dateneingeben. Um eine erste Liste zu erhalten, haben wir zwei Möglichkeiten: erstens können wir von jedem eingegebenen Datensatz mit der Taste "F8" eine Hardcopy (=Ausdruck des Bildschirminhalts) herstellen lassen oder wir können über den Punkt "Auswerten" eine Gesamt-Liste mit neuer Aufteilung anordnen.

Nehmen wir an, als Basisliste reicht eine Aufstellung der Namen und Adressen der Kunden. Dann legen wir die Auswertung so an, daß nur die gewünschten Felder erscheinen und diese zweizeilig pro Datensatz mit 60 Zeichen pro Zeile ausgedruckt werden:

aus ganz verschiedenen Branchen. Was meinen Sie, wieviele Textteile sich in Ihren Geschäftsbriefen wiederholen. Ich kenne Sekretärinnen, die kennen die Briefftexte, die ihnen ihr Chef diktieren will, bevor er den Mund aufgemacht hat. "Ich möchte aber individuelle Briefe an meine Kunden schicken", werden Sie protestieren. Aber genau das kann ein vernünftiges Textverarbeitungsprogramm leisten.

Bleiben wir bei unserem Rollenspiel und schauen wir uns einen ziemlich typischen Angebotsbrief des Spielwarenhändlers Lukas an:

Hans-Walter Schulze
Bahnstr.11

4005 Meerbusch

Sehr geehrter Herr Schulze,

bei unserem Besuch der internationalen Tauschbörse für Modelleisenbahnen in Luzern ist es uns gelungen, die umfangreiche Sammlung eines schweizer Hobbyisten zu ersteigern.

Unter den zahlreichen Stücken der Sammlung befinden sich auch einige Vorkriegs-Maerklin-Lokomotiven und -Waggons, die Sie sicher interessieren werden. Es sind dies im Einzelnen:

- 1 Stück XZY, Serien-Nummer 4711
- 2 Stück ZYX, Serien-Nummer 1510 und 8623
- 15 Stück verschiedene Güterwaggons
- 11 Stück 2.Klasse-Wagen Baureihe 009

Alle Stücke befinden sich in hervorragendem Originalzustand.

Aus Ihren Anfragen entnehmen wir, daß die angebotenen Modelle ganz besonders gut zu Ihrer Sammlung passen würden.

Falls Sie am Erwerb der angebotenen Stücke interessiert sind, fordern Sie bitte unverbindlich die Fotos der genannten Modelle an.

Unser Angebot können wir leider nur bis zum 15. Juli aufrechterhalten, da uns eine Anfrage zum Erwerb der kompletten Sammlung vorliegt. Über Ihre Antwort würden wir uns sehr freuen und verbleiben

mit freundlichen Grüßen

Ihre LUKAS-Spielwaren

(Unterschrift)

Düsseldorf, den 1. Juli 1984

Zerlegen wir den Brief in seine Einzelteile. Da ist zunächst die Anschrift, die kann jederzeit aus der Kunden-Datei in Briefe übernommen werden. der erste Abschnitt ist zu speziell, um ihn abzuspeichern. Mit den nächsten drei Absätzen ist es genauso. Erst der fünfte Teil des Briefes hat die Chance in vielen anderen Briefen wieder aufzutauchen. Aber zum Abspeichern erscheint er mir zu kurz; Sie mögen anders darüber denken. Aus dem letzten Absatz kann auch nicht viel noch einmal zum Zuge kommen.

Fazit: dieser Brief ist für weitere Aufgaben ungeeignet. Halt, nicht ganz. Stellen Sie sich vor, die erwähnte

Sammlung enthält auch ein paar H0-Oldtimer-Automodelle, für die sich andere Kunden sehr interessieren. Um diese Leute zu informieren, nehmen Sie einfach den Brief, löschen den Absatz 3) ganz und ersetzen die Formulierung über die Vorkriegs-Maerklin durch einen Satz über die Oldtimer, dann noch die Liste und die Angaben über den Zustand der Stücke angepasst und schon steht der zweite Brief. Sehen Sie mal:

Andreas Nimzick
Daimlerstr. 88

7000 Stuttgart 1

Sehr geehrter Herr Nimzick,

bei unserem Besuch der internationalen Tauschbörse für Modelleisenbahnen in Luzern ist es uns gelungen, die umfangreiche Sammlung eines schweizer Hobbyisten zu ersteigern.

Unter den zahlreichen Stücken der Sammlung befinden sich auch einige ! H0-Automodelle aus den Jahren 37 bis 40 ! , die Sie sicher interessieren werden. Es sind dies ! ! :

! Verschiedene PKW, Busse und Lastwagen zum Teil mit historischen Werbeaufschriften und als besonderer Leckerbissen ein VW-Käfer-Modell Baujahr 1940 der Firma Pirsche, Cannstatt. !

Alle Stücke befinden sich in ! gutem Zustand und bedürfen nur geringer Restaurierung. !

Aus Ihren Anfragen entnehmen wir, daß die angebotenen Modelle ganz besonders gut zu Ihrer Sammlung passen würden.

Falls Sie usw. usw.

Den Rest können wir uns schenken. Aber, ist Ihnen aufgefallen, wie gering die Änderungen gegenüber dem ursprünglichen Brief sind ? Und trotzdem ist der zweite genauso individuell wie der erste (die Änderungen sind durch das !-Zeichen angedeutet).

Wir haben hier den klassischen Fall des - wie ich es nennen würde - variablen Formbriefes vor uns; ein Briefmuster wird individuell dem Adressaten angepasst. Der andere Fall, in dem Textverarbeitungs-Software zum Zuge kommt, ist, neben dem reinen Standardbrief, den wir hier nicht besonders besprechen wollen, die Floskel-Datei.

Auch Sie verwenden Floskeln und ständig wiederkehrende Formulierungen; teilweise sind Sie sogar dazu gezwungen (wo es beispielsweise um Haftungsfragen oder andere gesetzlich vorgeschriebene Hinweise geht). Die Frage, die sich immer stellt ist "sind diese Textblöcke umfangreich genug, um Sie als eigene Dateien abzuspeichern ?" Nun, da müssen wir unser Rollenspiel verlassen. Was für den Modelleisenbahn-Händler Lukas sinnvoll ist, kann bei Ihnen völlig nebensächlich sein. Sie müssen es schon selber wissen, welche Textteile so oft vorkommen, daß sich ein Speichern lohnt. Überlegen Sie sich nur eins: Sie bauen einen Serienbrief, auch wenn Sie ihn individuell anpassen wollen, ja nur am Anfang der Arbeit zusammen. Da kann es sich lohnen auch kleinere Textsplitter einer wohlsortierten Floskel-Auswahl zu entnehmen.

3.4.3. Text und Daten zusammenbringen

Bleiben wir bei dem Beispiel mit dem Info-Brief über die ersteigerte Sammlung. Der erste Schritt ist, den Standardbrief zu entwerfen. Dabei sind zwei Strategien denkbar: entweder Sie schreiben einen Basistext mit Freistellen für die individuelle Ansprache oder einen Brief, der auf einen bestimmten Kundenkreis bezogen ist und später für die anderen Interessenten angepasst wird. Wie Sie es machen, ist Geschmackssache; mancher mag nicht gern ins Blaue hinein schreiben.

Nun suchen Sie sich die Kunden heraus, für die das Angebot überhaupt in Frage kommt. Das können Sie über die vorbereiteten Pointerdateien tun oder, falls Ihre Kundenkartei nicht zu groß ist, von Hand aus der Gesamt-Datei.

Die erste Methode setzt voraus, daß Ihre Pointerdateien leicht zu identifizieren sind, aus dem Namen "Pl" für so eine Datei läßt sich schwer erkennen, für was sich die darin enthaltenden Kunden interessieren, während die Bezeichnung "altmaerklinh0" schon aufschlußreicher ist.

So, und nun kommt der Drucker zum Zug. Wie Name und Anschrift in den Brief eingesetzt werden, ist im TEXTOMAT-Handbuch erklärt. Sie gehen nun einfach die ausgewählten Pointerdateien durch und suchen (bzw. ändern um) die dazu passenden Texte aus.

Der Rest ist Falten, Eintüten und Verschicken. Übrigens: verwenden Sie Fensterumschläge, dann vermeiden Sie Fehladressierungen, die beim Adressieren mit Etiketten schon mal vorkommen. Bei derart individuellen Briefen sind solche Fehler ziemlich peinlich.

3.4.4 Fazit

Der Spielwarenhändler Lukas hat einen großen Bedarf an individuellen Werbebriefen, Sie vielleicht nicht. Aber möglicherweise versenden Sie oft gleichlautende Briefe an sehr viele Adressaten. Dann lohnt sich der Einsatz von C64, Text- und Datenverwaltungs-Programm natürlich auch. Einige ergänzende Tips noch:

- erstellen Sie Ihre Formulare (Rechnung, Mahnung, Angebot) per Textverarbeitung;
- pflegen Sie Ihre Standardbrief-Bibliothek mit einem Datenverwaltungs-Programm.
- Schaffen Sie sich eine ständig aktualisierte Hardcopy der Standardbrief-Bibliothek zum Ausschuchen, mit Angaben darüber, wo der Text gespeichert ist.
- speichern Sie auch Briefe, die Ihnen auf den ersten Blick zu speziell erscheinen, vielleicht können Sie Teile daraus später doch verwenden

Das Kapitel ist sowieso schon viel länger geworden, als vorgesehen, also lasse ich einige zusätzliche Anmerkungen unter den Tisch fallen. Sie werden im alltäglichen Einsatz sicher Erfahrungen machen, die aus dem Rahmen des hier geschilderten fallen.

4. Tabellen erstellen, Rechnen und Planen

Aus dem Kapitel 2.2 kennen Sie ja die besonderen Vorzüge des Computers auf diesem Gebiet: Schnelligkeit, Bequemlichkeit und Genauigkeit. Wie bei allen Dingen, die die Maschine besonders gut kann, bleibt die Frage "Wie nutze ich das am besten aus ?" Dieses Kapitel gibt Ihnen 6 Antworten auf diese Frage.

Wie nicht anders zu erwarten, wird das liebe Geld eine Hauptrolle spielen. Es ist einfach so: bei Finanzen die Übersicht zu behalten, ist eine ausgesprochene Talentfrage. Ich habe dies Talent nicht, dafür besitze ich einen C64. Das ist ein guter Ausgleich.

Bei Problemen, wie der in Kapitel 4.3 beschriebenen Baukosten-Berechnung, kann sich Ihr Gerät sogar wirklich bezahlt machen. Stellen Sie sich vor, mit diesem Programm suchen Sie eine Finanzierung aus, die Ihnen über die Jahre 5.000 DM spart ? Wäre das nicht prima ?

Auch die allseits beliebten Computer-Haushaltsbücher sind echte Sparhelfer. Wer genau weiß, wohin die Groschen rollen, weiß auch bald, wo er sparen kann. Ähnliches gilt für die Autokosten. Berechnen Sie anhand von Testberichten die ungefähren Kilometerkosten für alle Modelle, die Sie für den Kauf ins Auge gefaßt haben, und Sie schaffen sich eine gute Entscheidungshilfe.

Eins noch: alle Programme sind als Anregungen gemeint, Sie sollten, wenn Sie es sich zutrauen, daran experimentieren und manipulieren, bis Sie das haben, was Ihren ganz persönlichen Bedürfnissen entspricht.

4.1 Haushaltsbuch - Tips für den eigenen Entwurf

Ich weiß, Sie werden die Hände über dem Kopf zusammenschlagen, "Schon wieder ein Haushaltsbuch für den 64er !" Es gibt auch wirklich keine Ausrede, außer, daß es ganz unterschiedliche Ansätze für ein elektronisches Haushaltsbuch gibt. Die meisten Programme, die Sie irgendwo abtippen können, tun so, als sei ein stinknormaler bundesdeutscher Haushalt zu verwalten wie ein mittlerer Handwerksbetrieb; da ist von Konten, vom Buchen und Saldieren die Rede, daß einem ganz schwindlig werden kann. Und richtig, die Mehrzahl dieser Programme basiert auf Kontoführungs-Software. Wie gesagt, das ist der eine Weg. Wir wollen uns den anderen ansehen.

Der einfache Weg zum finanziellen Durchblick

Darum geht es ! Die häusliche Buchführung dient dazu, von jedem Groschen zu wissen, woher er kommt und wohin er geht. Und das ist einfach, aber individuell verschieden. Sie finden nun eine Check-Liste, die Ihnen hilft, ein Haushaltsbuch zu entwerfen, daß Ihren Bedürfnissen entspricht.

```
*****
*
*          CHECK-LISTE für den HAUSHALTSBUCH-ENTWURF          *
*
*****
*
*  A) FRAGEN                                                    *
*
*  1) Anzahl der festen Einnahmen ?      (bis 5 / über 5) *
*  2) Anzahl der festen Ausgaben ?      (bis 9 / über 9) *
*  3) Anzahl der Rubriken für          *
```

- * wechselnde Einnahmen ? (bis 10 / über 10) *
- * 4) Anzahl der Rubriken für *
- * wechselnde Ausgaben ? (bis 10 / über 10) *
- * 5) Sollen die einzelnen Rubriken für *
- * das ganze Jahr addiert werden ? (NEIN / JA) *
- * 6) Sollen mehrmals im Monat Beträge *
- * eingegeben werden ? (NEIN / JA) *
- * 7) Werden Statistiken benötigt *
- * - für die Monatssummen ? (NEIN / JA) *
- * - für die Rubriken ? (NEIN / JA) *
- * 8) Sollen die Statistiken grafisch *
- * dargestellt werden ? (NEIN / JA) *
- * 9) Sollen Bankkonten automatisch *
- * überwacht werden ? (NEIN / JA) *
- * *
- *****

Bevor Sie zu der eigentlichen Checkliste gehen, werten Sie bitte erst die Fragen aus. Bei den Fragen 1) bis 4) zählt eine Entscheidung für den kleineren Wert als NEIN.

9 MAL JA : Sie sind hier falsch ! Was Sie brauchen ist kein Haushaltsbuch sondern eine Mini-FiBu. Für Ihre Zwecke eignet sich ein entsprechendes Software-Paket am besten.

4 BIS 5 MAL JA : Wenn sich die Anzahl Ihrer JA-Antworten hauptsächlich durch den Mengenbedarf ergibt, sollten Sie dabei bleiben. Hinweise darauf, wie Sie ein sehr einfaches Haushaltsbuch-Programm für viele Rubriken selbst entwickeln können, finden Sie in diesem Kapitel. Haben Sie dagegen die Fragen 5) bis 9) mit JA beantwortet, sollten Sie nach Bearbeitung der CHECK-POINTS zum Kapitel 4.1.2 weiterblättern.

WENIGER ALS 4 MAL JA : Sie sind bescheiden ! Ihr Programm wird entsprechend einfach zu schreiben sein. Sehen Sie sich die CHECK-POINTS an.

Die folgende Check-Liste bezieht sich auf die Idee, ein Haushaltsbuch als relative Datei aufzubauen. Dieser Ansatz scheint mir der universellste zu sein. Ein komplettes Listing zum Abtippen finden Sie im "Großen Floppy-Buch" von Englisch/Szczepanowski (ein DATA BECKER Buch). Das Programm ist vollständig dokumentiert und recht verständlich erklärt. "Wozu dann noch die Check-Liste?" werden Sie sagen. Die Antwort: um Ihnen die Möglichkeit zu geben, das Haushaltsbuch individuell aufzubauen. Beispielsweise sollen Sie sich dafür entscheiden können, auf allen Firlefanzen zu verzichten und ein reines Eingabe/Ausgabe-Modul als Programm zu verwenden, oder aber ein Haushaltsbuch mit allen Schikanen zu entwickeln (Statistik, Grafik, Prognose etc.).

Die Punkte 1),2) und 5) sind - gleich in welcher Form - obligatorisch für jedes Haushaltsbuch-Programm, der Punkt 3) erhöht den Durchblick und der Punkt 4) macht das Ergebnis greifbar. Wenn Sie sich für die Grundversion entscheiden, brauchen Sie noch nicht einmal ein "Menue" im eigentlichen Sinne, die Reihenfolge der Programmteile kann zwangsweise geregelt sein. Wie Sie das nun alles zusammen bauen können, erfahren Sie im nächsten Kapitel.

```

*****
*
*
* B) CHECK-POINTS
*
*
* 1) Rubriken anlegen
*   - Anzahl festlegen ( maximal ca. 40 )
*   - Namen festlegen ( je max. 20 Zeichen )
*   - Max. Länge der Beträge definieren
*     (z.B. größtmöglicher Betrag ist
*       DM 100.000,00 = Feldlänge 9 Zeichen
*   - Ausgaben- bzw. Eingaberubriken festlegen
*   - Rubriken nummerieren
*   - eine Liste der Rubriken mit
*     ihren Nummern wird erstellt
*
* 2) Bildschirm Ein-/Ausgabe
*   - Rubriken-Nummer eingeben
*   - Rubriken-Name wird ausgegeben
*   - die Beträge für alle 12 Monate werden ausgegeben
*   - Monat eingeben
*   - Betrag (positiv oder negativ) eingeben
*   - Veränderter Betrag wird angezeigt
*   - alle Rubriken für einen Monat werden ausgegeben
*   - die Jahressumme einer Rubrik wird ausgegeben
*   - die Monatssumme aller Rubriken wird ausgegeben
*   - die Jahressumme aller Monate wird ausgegeben
*
* 3) Statistik/Grafik-Modul
*   - die o.g. Ausgaben werden grafisch dargestellt
*     - als Kurven-Diagramm
*     - als Balken-Diagramm
*   - die Beträge werden als prozentuale Anteile
*     der jeweiligen Gesamtsumme bearbeitet und
*     grafisch dargestellt
*     - als Torten-Grafik
*     - als Säulen-Diagramm
*   - aus Kurven-Diagrammen werden zukünftige Werte
*     als Prognose extrapoliert
*
*

```

```
* 4) Drucker-Ausgabe *
*   - wie Bildschirm Ein-/Ausgabe *
*   - zusätzlich Ausgabe aller Rubriken, aller *
*     Monate, aller Summen in einem Druckvorgang *
* *
* 5) Speichern/Einlesen *
* *
* 6) Menue *
* *
*****
```

4.1.1 Und so sieht es in der Praxis aus

```

*****
*   MAN NEHME   :                               *
*****

*****
*   HARDWARE   :                               *
*****
*                                                     *
*   COMMODORE 64                               *
*   Floppy 1541                               *
*   Monochromer Monitor                       *
*                                                     *
*****

*****
*   SOFTWARE   :                               *
*****
*                                                     *
*   keine                                           *
*                                                     *
*****

*****
*   VORKENNTNISSE :                           *
*****
*                                                     *
*   je mehr, je besser                           *
*                                                     *
*****

```

Die Drohung steht im Raum: ich werde Ihnen jetzt zu erklären versuchen, was eine relative Datei ist. Um mit einem dummen Scherz zu beginnen: eine relative Datei ist etwas anderes

als eine sequentielle. Der Begriff ist Ihnen in diesem Buch ja schon das eine oder andere Mal über den Weg gelaufen, Sie wissen, "sequentiell" heißt soviel wie "hintereinander". Sehen Sie sich einmal folgendes Beispiel einer sequentiellen Datei an:

*** OBST-PREISLISTE ***

1) Äpfel	1,95 / kg
2) Birnen	3,10 / kg
3) Erdbeeren	8,85 / kg
4) Feigen	12,50 / kg
5) Grapefruit	4,00 / kg
6) Himbeeren	11,05 / kg
7) Ingwer	5,60 / kg
8) Johannisbeeren	7,00 / kg
9) Kirschen	9,95 / kg
10) Mandarinen	1,25 / kg
11) Nektarinen	4,20 / kg
12) Orangen	3,70 / kg
13) Pflaumen	6,88 / kg
14) Quitten	10,05 / kg
15) Stachelbeeren	2,40 / kg
16) Trauben	10,10 / kg
17) Walnüsse	2,95 / kg

Wenn diese Liste (mit einem kleinen Programm) erstellt worden ist, kommen wir zum dramatischen Teil: dem Speichern auf Diskette. Die Laufnummern dienen nur der Bildschirmausgabe, werden also nicht gespeichert. Wollen Sie mal sehen, wie die Liste auf Diskette aussieht ?

```
Äpfel*1,95*Birnen*3,10 Erdbeeren*8,85*Feigen*12,50*
Grapefruit*4,00*Himbeeren*11,05*Ingwer*5,60*Johannisbeeren
*7,00*Kirschen*9,95*Mandarinen*1,25*Nektarinen*4,20
*Orangen*3,70*Pflaumen*6,88*Quitten*10,05*usw.usw.usw.
(wobei der "*" für das RETURN am Ende jeder Eingabe steht)
```

Die Datei "Obst-Preisliste" ist sequentiell aufgebaut. Nun

wollen Sie nicht nur speichern, sondern auch Daten wieder auf den Bildschirm holen. Ganz einfach. Im Programm ist der Befehl enthalten, den Namen der Frucht in Spalte 4, den Preis in Spalte 20 zu drucken. Gesagt, getan; die abgebildete Liste erscheint auf der Mattscheibe. Jetzt wollen Sie die Preise ändern. Und zwar für Trauben. Sie schalten auf Eingabemodus, fahren den Cursor zur Spalte 20 und geben den neuen Preis ein und speichern wie gewohnt ab, fertig. Ganz einfach.

Jetzt stellen Sie sich folgende Situation vor: die Preisliste umfaßt nicht nur 17, sondern 1.000 Artikel. Die passen alle gar nicht auf den Bildschirm, eventuell noch nicht einmal in den Arbeitsspeicher. Also muß die Datei scheinchenweise in den Speicher geholt werden, dann wird jeder Datensatz mit dem Suchwort verglichen, dann das nächste Stück Datei herangeschafft, gesucht und so weiter, und so weiter.

Unglücklicherweise steht "Trauben" im 999. Datensatz. Die Sucherei kann also dauern.

"Könnte man nicht dem Computer irgendwie sagen, das er nur von X bis Y suchen muß?" Wenn Ihnen diese Idee gerade gekommen ist, sind Sie auf der richtigen Spur. Das ist das Prinzip einer relativen Datei. Jeder Datensatz hat eine feste Position innerhalb der Datei, die durch den RELATIVEN (daher der Name!) Abstand zum Dateibeginn ausgedrückt wird. Ihr Einwand: "Wenn aber die Datensätze unterschiedlich lang sind, wie soll das denn erfaßt werden?" Richtig. Deshalb dürfen die Datensätze in einer relativen Datei einfach nicht verschieden lang sein. Im Notfall werden zu kurze Datensätze (bzw. Felder) mit Leerzeichen aufgefüllt und so auf die Norm gebracht.

Die Obst-Preisliste sähe auf der Diskette als relative Datei ungefähr so aus:

```
*01*Äpfel           * 1,95*02*Johannisbeeren * 7,00*
```

Die Datensätze und die Felder innerhalb eines Datensatzes haben alle die gleiche Anzahl Zeichen und sind durchnummeriert. Jeder Record (= anderes Wort für Datensatz) kann nun über seine Recordnummer aufgerufen werden. Dadurch, daß alle Records gleichlang sind, ist die Recordnummer gleich dem Abstand des Datensatzes vom Anfang der Datei. Um es ganz deutlich zu machen noch eine Demonstration:

Nehmen wir an, die Diskette wäre ein Regalbrett. Auf dem Brett ist alle 10 Zentimeter ein Strich gezogen, die so entstandenen Fächer sind durchnummeriert. .

/01-----/02-----/03-----/04-----/05----/

In diese Fächer werden Bananen gelegt. Sequentielle Bananen sind unterschiedlich lang

Banane 1 = /+++++++/
Banane 2 = /+++++/
Banane 3 = /+++++++/
Banane 4 = /+++++++/
usw.

Nur die Banane 2 paßt in EIN Fach, alle anderen sind zu lang. Der Computer soll jetzt die Banane 3 suchen. Da man ihm vorher gesagt hat, jede Banane läge im Fach mit ihrer Bananen-Nummer (=es handelte sich um eine relative Datei), greift er ins Fach 3 und, sehen Sie selbst, was er hervorholt:

Inhalt von Fach 3 = *lllllll* (ca. 8 cm Banane)

Das mag bei Bananen nicht tragisch sein, aber bei Beträgen einer Haushaltsberechnung schon (8 cm Datensatz sähen z.B. 5,00 Juli D5 oder ähnlich aus).

Die Grundbedingung für eine relative Datei ist also klar, die Vorteile auch. Wie also installiert der geneigte Anwender eine derartige Datei mit seinen Mitteln ? Radio

Eriwan antwortet : Im Prinzip gar nicht. Das BASIC 2.0, daß auf dem 64er implementiert (=eingebaut) ist, kennt keine Befehle zur Verwaltung relativer Dateien. Und es geht trotzdem. Zwar kompliziert, aber immerhin.

Im oben bereits erwähnten "Großen Floppy-Buch" wird genau gezeigt und erklärt, wie Sie eine relative Datei aufbauen und verwalten. Ich will mich hier nur auf die harten Fakten einlassen und zeigen wie es praktisch funktioniert.

Zunächst müssen wir uns überlegen, wie ein einzelner Record unserer Haushaltsbuch-Datei (aber sofort nenne ich sie HAHABU !) beschaffen sein muß. Entscheiden Sie zwischen zwei Varianten:

- ein Record = ein Monat
das ergibt 13 Records
(je 1 pro Monat + 1 für die Jahressumme)
- ein Record = eine Rubrik

Für die Zugriffsgeschwindigkeit spielt die Wahl keine Rolle, für den Bedienungskomfort ebenfalls nicht, sie ist also Geschmackssache. Nehmen wir für unser Beispiel die zweite Variante. Ein Record wird also folgendermaßen aufgebaut.

- | | |
|---------------------|--------------|
| 1. Name der Rubrik | (20 Zeichen) |
| 2. Januar-Betrag | (10 Zeichen) |
| 3. Februar- Betrag | (10 Zeichen) |
| 4. März-Betrag | (10 Zeichen) |
| 5. April-Betrag | (10 Zeichen) |
| ... | |
| 11. Oktober-Betrag | (10 Zeichen) |
| 12. November-Betrag | (10 Zeichen) |
| 13. Dezember-Betrag | (10 Zeichen) |
| 14. Jahressumme | (12 Zeichen) |

Das ergibt als Länge für einen Record 153 (20 + 12*10 + 12 + 1 für RETURN). Die Anzahl der Records ist gleich der Anzahl der Rubriken, nehmen wir so aus der hohlen Hand an, es seien

36. Die relative Datei wird mit dem Befehl:

```
OPEN 1,8,2,"HAHABU,L,"+CHR$(153)
```

eröffnet. Das L teilt dem Betriebssystem der Floppy (dem DOS = Disk Operating System) mit, daß HAHABU als relative Datei eröffnet wird, CHR\$(153) zeigt die Recordlänge an. Um alle Records zum Beschreiben freizugeben, wird der letzte Record mit CHR\$(255) belegt. Es ist nämlich so, daß Sie ohne diesen Vorgang bei jedem neuen Eingeben eines Satzes dessen Record immer erst freigegeben müßten.

Beim HAHABU werden die Records nicht freiweg von 1 - 36 geschrieben, sondern in beliebiger Reihenfolge. Damit aber der Januar-Betrag für Miete nicht beim März-Gehalt landet, muß erst positioniert werden, d.h. ein Zeiger auf das richtige Feld im richtigen Datensatz weisen. Das geht mit dem folgenden Befehl:

```
PRINT(nr)2,"P"+CHR$(2)+CHR$(0)+CHR$(20)+CHR$(1)
```

Ich will das der Reihe nach erklären. Zunächst das (nr) entspricht dem amerikanischen Nummerzeichen (Doppelkreuz); "P" gibt den Befehl zum Positionieren; der Wert des 1. CHR\$ ist der Kanal, der zur Übermittlung geöffnet wurde (hier die 2, weil die 1 ja für die Datei gebraucht wird); die nächsten beiden CHR\$ enthalten die Recordnummer. Sie wird aufgeteilt in zwei CHR\$, weil ein CHR\$-Wert maximal 255 betragen, eine relative Datei aber bis zu 65535 Records enthalten kann. Wir bleiben unter 255, da interessiert uns der erste CHR\$ nicht, der bleibt 0. Im zweiten ist die 20 angegeben, d.h., wir wollen was mit dem Record Nr. 20 anstellen. Im letzten CHR\$ steht die Nummer des Zeichens innerhalb des Records, daß wir beschreiben oder lesen wollen. Hier sollte tunlichst immer eine 1 eingegeben werden, da ein Record nur komplett verarbeitet werden kann, wenn man bei seinem ersten Zeichen anfängt. Machen wir aus dem theoretischen Erklären ein praktisches Programm:

```
100 NR=36: REM NR = RECORDNUMMER
110 OPEN 1,8,2,"HAHABU,L,"+CHR$(153)
120 OPEN 2,8,15 : REM BEFEHLSKANAL FUER POSITIONIERUNG
130 PRINT(nr)2,"P"+CHR$(2)+CHR$(0)+CHR$(36)+CHR$(1)
140 PRINT(nr)1,CHR$(255)
150 CLOSE 1 : CLOSE 2
```

Die Datei ist eröffnet, 36 Records zu je 153 Zeichen freigegeben. Das Spiel beginnt.

Und mit diesem Wissen will ich Sie jetzt alleine lassen. Wahrscheinlich sind Sie, wenn Sie das Kapitel wirklich bis hier hin gelesen haben ,in der Lage ein Programm zu schreiben, daß Ihren Bedürfnissen entspricht. Falls nicht, möchte ich Sie noch einmal auf das "Große Floppy-Buch" verweisen. Und wenn Ihr HAHABU dann läuft, werden Sie sich wundern, wie schnell und bequem es geht.

4.1.2 Haushaltsbuch als Software-Paket

```
*****
*      MAN NEHME      :      *
*****

*****
*      HARDWARE      :      *
*****
*
*      COMMODORE 64      *
*      Floppy 1541      *
*      Beliebiger Bildschirm      *
*
*****

*****
*      SOFTWARE      :      *
*****
*
*      Haushaltsbuch-Software      *
*
*****

*****
*      VORKENNTNISSE      *
*****
*
*      keine      *
*
*****
```

Ein Problem: das hier beschriebene Programm "FINANZGENIE"
(von DATA BECKER) gab es noch nicht, als ich es beschrieb

(Stand Juni 84). Was mir vorlag, war ein Prototyp. Deshalb will ich nicht versuchen, eine Bedienungsanleitung zu liefern, sondern eine Beschreibung der Möglichkeiten.

"FINANZGENIE" basiert auf einem ganz anderen Ansatz als die meisten Programme, die angeblich dazu geeignet sind, einen Haushalt zu verwalten. Man könnte sagen, es arbeitet termin-orientiert. Das bedeutet, daß alle Daten anhand ihrer Fälligkeit identifiziert werden können.

Das kann das "FINANZGENIE":

- Einnahmen/Ausgaben verwalten;
- Bankkonten aller Art überwachen;
- Kredite und Sparverträge überwachen
- und berechnen;
- Terminkalender für einmalige
- und wiederkehrende Termine führen;
- Monatsabschlüsse erstellen;
- Prognosen berechnen.
- Das alles kann gespeichert,
- geladen
- und gedruckt werden.

Vielleicht fürchten Sie, daß ein Abendkursus nötig wäre, um dies alles zu beherrschen. Mitnichten ! Alle Funktionen sind menuegesteuert. Sie wählen ganz bequem mit den Cursortasten an, was erledigt werden soll und das Programm begibt sich an die Arbeit. Ein wenig Gewöhnungszeit braucht es schon, das ist der Preis für die Vielfalt. Selbst der Prototyp bot schon so viel, daß ich nicht alle Fähigkeiten ausschöpfen konnte.

Besonders faszinierend fand ich die Zins- und Tilgungsberechnungen. Geben Sie die Darlehenssumme ein, fragt das Programm nach dem effektiven Jahreszins, der Laufzeit und dem Turnus der Zahlungen. In Null-Komma-Nichts wirft der Rechner die Ratenhöhe aus. Auch umgekehrt

funktioniert es. Eingabe: Ratenhöhe und Anzahl der Raten;
Ausgabe: Kreditsumme. Wenn Sie nur einmal Angebote
verschiedener Banken so verglichen und sich das günstigste
ausgesucht haben, hat sich die ganze Computer-Anlage samt
Software schon bezahlt gemacht.

Ich hoffe, Sie sind recht neugierig geworden und wollen das
"FINANZGENIE" schon bald im Hause haben. Achten Sie auf die
Veröffentlichungen in der einschlägigen Presse !

4.2 Autokosten im Griff

```
*****
*      MAN NEHME      :      *
*****

*****
*      HARDWARE      :      *
*****
*
*      COMMODORE 64      *
*      Floppy 1541      *
*      Beliebiger Monitor *
*
*****

*****
*      SOFTWARE      :      *
*****
*
*      keine      *
*
*****

*****
*      VORKENNTNISSE      *
*****
*
*      Gute bis sehr gute      *
*      BASIC-Kenntnisse      *
*
*****
```

Gut, daß ich nicht so chaotisch autofahre, wie ich BASIC-Programme schreibe, mein Konto in Flensburg wäre

immens und meine Versicherungsprämie auch. Zum Glück kann man mit konfusen Listings niemanden über den Haufen fahren, höchstens verärgern.

Kürzlich übergab ich, im Rahmen eines Softwaretausches, einem Hobbykollegen mit stolzgeschwellter Brust mein selbstfabriziertes Programm "Autokosten", worauf er - nach etlichen erfolglosen Versuchen das Ding ans Laufen zu kriegen - jeden Kontakt mit mir abbrach. Schade, er war ganz nett. Aber, wie sollte er ahnen, daß das Programm auf die Eingabe von "0" allergisch reagierte und grundsätzlich mit "DIVISION BY ZERO ERROR" antwortete und das diese Klippe nur mit der Eingabe von "0.001" statt "0" zu umschiffen war.

Nun, ich habe aus dem Vorfall meine Konsequenzen gezogen. Erstens habe ich mir das Buch "64 für Profis" angeschaut und, zweitens, das Gelesene umgesetzt. Das Ergebnis sehen Sie hier.

Die Programmidee

Das Programm "Autokosten" dient zum Erfassen und Speichern sämtlicher Kosten eines Privat-Kfz. Außerdem sollen monatlich und jährlich die Gesamtkosten berechnet werden, sowie die Kosten pro Kilometer. Die so gewonnenen Daten sollen in einer sequentiellen Datei gespeichert werden.

Die Problemanalyse

Was soll das Programm leisten ?

- 1) Daten erfassen
- 2) Daten berechnen
- 3) Daten ausgeben
- 4) Daten speichern

Der Programmablauf ergibt sich daraus:

- 1) Daten einlesen
- 2) Monatszahl eingeben
- 3) Monatsdaten ausgeben
- 4) Rubriknummer eingeben
- 5) Betrag eingeben
- 6) Neuen Betrag berechnen
- 7) Neue Monatsdaten ausgeben
- 8) Jahresbeträge berechnen
- 9) Daten speichern
- 10) Programm beenden

Ein Benutzer-Menue erübrigt sich also

Die sequentielle Datei ist folgendermaßen aufgebaut:

- 13 Datensätze (für 12 Monate und das Jahr)
- zu je 9 Feldern

- 1) Tankkosten
- 2) Reparaturen
- 3) Pflege/Wartung
- 4) Erstattungen
- 5) Gefahrene Strecke
- 6) Steuern
- 7) Versicherung
- 8) Gesamtkosten
- 9) Kilometer-Kosten

Die Bildschirmmaske

sieht so aus:

```
-----  
00 *****  
01 * *  
02 *          AUTOKOSTEN          *  
03 * *  
04 *****
```

```

06 Monatsdaten Ausgabe fuer (Monatsname)
07
08
09 (1) Tankkosten           = DM .....
10 (2) Reparaturen         = DM .....
11 (3) Pflege/Wartung      = DM .....
12 (4) Erstattungen       = DM .....
13 (5) Gefahrene Strecke = Km .....
14
15 (6) Steuern            = DM .....
16 (7) Versicherung       = DM .....
17
18 (8) Gesamtkosten      = DM .....
19 (9) Kilometer-Kosten Pf/Km .....
20
21
22 (( EINGABEZEILE ))
23
24
-----

```

Der Datei-Entwurf

```

DATEINAME       : dat.autokosten
PROGRAMMNAME    : autokosten
DATEIART        : seq.
DATENSÄTZE     : 13
FELDER          : je 9
DATENSATZLÄNGE : 65 Zeichen

```

Die Felder:

NR.	INHALT	ART	LÄNGE	NAME
1	Tankkosten	N	8	ak(m,1)
2	Reparaturen	N	8	ak(m,2)
3	Pflege/Wartung	N	8	ak(m,3)

4	Erstattungen	N	8	ak(m,4)
5	Gefahrene Strecke	N	6	ak(m,5)
6	Steuern	N	7	ak(m,6)
7	Versicherung	N	7	ak(m,7)
8	Gesamtkosten	N	8	ak(m,8)
9	Kilometer-Kosten	N	5	ak(m,9)

Notwendige Rechenformeln

1. Steuern und Versicherung

Frage:

"Steuerbetrag = DM"? und "Zahlungsweise (12,6,4,1) "?

Rechenformel:

Steuerbetrag = is

Zahlungsweise = zw%

Monatsanteil ak(m,6) = is/zw%

Jahresbetrag ak(13,6) = 12*(is/zw%)

Für die Versicherung geht es genauso.

2. Gesamtkosten und Kilometer-Kosten

Eingabebetrag = b

Gefahrene Strecke = ak(m,5)

Gesamtkosten ak(m,8) = ak(m,8)+b

Kilometer-Kosten ak(m,9) = (ak(m,8)/ak(m,5))*100

Für die Jahressumme gilt entsprechendes.

Außerdem brauchen wir noch Routinen zum Runden der DM-Beträge und zum Umformen der Werte in Dezimalform mit zwei Nachkommastellen.

Das Listing

Das Licht im Saal geht aus, der Vorhang hebt sich: das Programm ***** AUTOKOSTEN ***** ; bitteschön. Schauen Sie sich die vielen Zeilen in aller Ruhe (mit einem Glas Wein am Kamin, vielleicht ?) an, die Kommentare kommen anschließend.

```
1 rem *****
2 rem *** autokosten ***          ***
3 rem *** aus dem ***   version 3.10 ***
4 rem *** data becker***   stand 28.5   ***
5 rem *** ideen-buch ***          ***
6 rem *****
7 rem
8 rem
9 print chr$(147)                : rem bildschirm loeschen
10 poke 53272,23                 : rem gross/klein schrift
15 poke 53280,5 : poke 53281,5   : rem bildschirm farbe = gruen
20 print chr$(149)               : rem schrift farbe = braun
25 dim ak(13,9)                 : rem datenfeld bearbeitung
30 gosub 11000                  : rem eingabezeile
35 rem
40 input "Ist dies die erste Bearbeitung ";w#
45 if w#<>"j" and w#<>"J" and w#<>"n" and w#<>"N" then 30
50 if w#="j" or w#="J" then 50000
55 rem
60 dim d$(13,9)                 : rem datenfeld speicherung
65 u#="5:"                      : rem datei ueberschreiben
70 print chr$(147)              : rem bildschirm loeschen
80 print chr$(149)              : rem schrift farbe = braun
90 rem
95 rem
100 rem ++++++++ programmstart ++++++++
110 rem
120 gosub 30000                 : rem daten einlesen
130 gosub 10000                 : rem ein/ausgabe-maske
140 gosub 11000                 : rem eingabezeile setzen
145 rem
150 input "Welcher Monat (0=Ende)      ";m
155 rem
```

```

160 if m=0 then 340                : rem daten abspeichern
170 if m>13 then 140              : rem neue eingabe
180 gosub 12000                   : rem monatsnamen klartext
190 gosub 20000                   : rem monatsdaten ausgeben
200 if m=13 then gosub 22000     : rem jahreseingaben
210 if m=13 then 140             : rem monatseingabe
220 gosub 11000                   : rem eingabezeile
225 rem
230 input "Welche Rubrik (0 bis 5 / 0=Ende) ";r
235 rem
240 if r>5 then 220               : rem neue eingabe
250 if r=0 then 140              : rem monatseingabe
260 cs=1026+40*cz(r)             : rem rubriken-zeiger
270 poke cs+34,188               : rem zeiger setzen
275 rem
280 if r<5 then input "Betrag eingeben      = DM ";b
290 if r=5 then input "Kilometer eingeben  = Km ";b
295 rem
300 gosub 21000                   : rem monatssummen berechnen
310 gosub 20000                   : rem monatsdaten ausgeben
320 poke cs+34,32                : rem rubriken-zeiger loeschen
330 goto 220                      : rem rubriken-eingabe
340 gosub 31000                   : rem daten abspeichern
350 gosub 11000                   : rem eingabezeile
355 rem
360 input "Soll das Programm beendet werden ";w#
370 if w#<>"j" and w#<>"J" and w#<>"n" and w#<>"N" then 450
380 if w#="n" or w#="N" then 100
385 rem
390 end
400 rem ++++++ prog amende ++++++
500 rem
505 rem
9000 rem ++++++ unterprogramme ++++++
9005 rem
9010 rem
10000 rem ++++++ ein/ausgabe maske ++++++
10005 rem
10010 print chr$(147);
10020 print "*****";
10030 print "*";
10040 print "*          AUTOKOSTEN          *";
10050 print "*";
10060 print "*****";
10070 print
10080 print "Monatsdaten-Ausgabe fuer          ";
10090 print : print
10100 print "(1) Tankkosten          = DM ..... "; : cz(1)=9
10110 print "(2) Reparaturen          = DM ..... "; : cz(2)=10
10120 print "(3) Pflege/Wartung       = DM ..... "; : cz(3)=11

```

```

10130 print "(4) Erstattungen      = DM .....    "; : cz(4)=12
10140 print "(5) Gefahrene Strecke = km .....    "; : cz(5)=13
10150 print "(6) Steuern            = DM .....    "; : cz(6)=15
10160 print "(7) Versicherung       = DM .....    "; : cz(7)=16
10170 print "(8) Gesamtkosten     = DM .....    "; : cz(8)=18
10180 print "(9) Kilometer-Kosten Pf/Km .....    "; : cz(9)=19
10300 return
10990 rem
10995 rem
11000 rem ++++++ eingabezeile ++++++
11005 rem eingabezeile setzen
11010 poke 214,22 : poke 211,0 : sys 58640
11020 return
11040 return
11990 rem
11995 rem
12000 rem ++++++ monatsnamen klar text ++++++
12005 rem
12010 data "Januar","Februar","Maerz"
12020 data "April","Mai","Juni","Juli"
12030 data "August","September"
12040 data "Oktober","November"
12050 data "Dezember","das Jahr 84"
12060 rem
12070 for n=1 to 13
12080 :   read n#
12090 :   if n=m then 12120
12100 next n
12120 mo#=n# : restore
12130 poke 214,6 : poke 211,26 : sys 58640
12140 print "      ";
12150 poke 214,6 : poke 211,26 : sys 58640
12160 print mo#;
12170 return
12990 rem
12995 rem
20000 rem ++++++ monatsdaten ausgeben ++++++
20005 ff#="      " : rem leerstring
20010 for r=1 to 9
20020 :   ak=ak(m,r)
20040 :   if r=5 then dz#=str$(ak) : dz%=len(dz#) : goto 20060
20050 :   gosub 40000 : rem rundung & dezimalformat
20060 :   poke 214,cz(r) : poke 211,28 : sys 58640
20070 :   print ff#
20080 :   poke 214,cz(r) : poke 211,36-dz% : sys 58640
20100 :   print dz#
20110 next r
20120 return
20990 rem
20995 rem

```

```

21000 rem ++++++ monatsdaten berechnen ++++++
21005 if r=4 then b=b*(-1) : rem erstattungen
21010 ak(m,r)=ak(m,r)+b
21015 if r=5 then 21030 : rem gefahrene strecke
21020 ak(m,8)=ak(m,8)+b : rem gesamtkosten
21030 gk=ak(m,8) : km=ak(m,5) : kk=gk/km
21040 ak(m,9)=int(kk*100)/10 : rem kilometerkosten
21050 ak(13,r)=ak(13,r)+b : rem jahressummen
21055 if r=5 then 21070
21060 ak(13,8)=ak(13,8)+b
21070 gk=ak(13,8) : km=ak(13,5) : kk=gk/km
21080 ak(13,9)=int(kk*100)/10
21090 return
21990 rem
21995 rem
22000 rem ++++++ jahreseingaben ++++++
22005 rem
22010 gosub 11000 : rem eingabezeile
22020 input "Steuern oder Versicherung eingeben";w#
22025 rem
22030 if w#<>"j" and w#<>"J" and w#<>"n" and w#<>"N" then 22010
22040 if w#="n" or w#="N" then return
22100 rem steuern eingeben
22105 print " ";
22110 gosub 11000 : rem eingabezeile
22120 input "Steuerbetrag = DM ";is
22130 gosub 11000 : rem eingabezeile
22140 input "Zahlungsweise (12,6,4,1) = ";zw%
22150 ms=is/zw% : rem monatsanteil
22160 for m=1 to 12
22170 : ak(m,6)=ms
22175 ak(m,8)=ak(m,8)+ms
22180 next m
22190 ak(13,6)=12*ms : rem jahressteuern
22200 rem versicherung eingeben
22205 print " "
22210 gosub 11000 : rem eingabezeile
22220 input "Versicherungsbeitrag = DM ";iv
22230 gosub 11000 : rem eingabezeile
22240 input "Zahlungsweise (12,6,4,1) = ";zw%
22250 mv=iv/zw% : rem monatsanteil
22260 for m=1 to 12
22270 : ak(m,7)=mv
22275 ak(m,8)=ak(m,8)+mv
22280 next m
22290 ak(13,7)=12*mv : rem jahresversicherung
22300 return
22990 rem
22995 rem

```

```

30000 rem ++++++ daten einlesen ++++++
30005 print "EINEN MOMENT BITTE"
30010 print : print "DATEI WIRD GELADEN"
30015 open 1,9,2,"dat.autokosten,s,r"
30020 m=1
30030 for r=1 to 9
30040 :   input#1,d$(m,r)
30050 next r
30060 if st<>64 then m=m+1 : goto 30030
30070 close 1
30080 for m=1 to 13
30090 :   for r=1 to 9
30100 :     ak(m,r) = val(d$(m,r))
30130 :   next r
30140 next m
30150 print chr$(147);
30160 return
30990 rem
30995 rem
31000 rem ++++++ daten speichern ++++++
31005 print chr$(147) : print "EINEN MOMENT BITTE"
31010 print : print "DATEI WIRD GESPEICHERT"
31015 open 1,9,2,u$+"dat.autokosten,s,w"
31020 for m=1 to 13
31030 :   for r=1 to 9
31035 :     pr$=str$(ak(m,r))
31040 :     print#1,pr$
31050 :   next r
31060 next m
31070 close 1
31080 print chr$(147);
31090 return
31990 rem
31995 rem
40000 rem ++++++ runden & dezimaldarstellung ++++++
40005 rem : rem beispiel ak=1.2345
40010 d1 = int(ak*1000) : rem d1=1234
40020 d2 = d1/10 : rem d2=123.4
40030 d3 = int(d2) : rem d3=123
40040 d4% = (d2-d3)*10 : rem d4=4
40050 :   if d4%>=5 then d5%=1 : rem aufrunden
40060 :   if d4% <5 then d5%=0 : rem abrunden
40070 d3 = d3+d5% : rem d3=123
40080 ak = d3/100 : rem ak=1.23
40090 rem dezimalformat
40100 dz$= str$(ak) : dz%=len(dz$)
40110 for p=1 to dz%
40120 :   p$=mid$(dz$,p,1) : rem dezimalpunkt suchen
40130 :   if p$="." then 40160

```

```

40140 next p
40150 dz$=dz$+".00"           : rem dz$ enthaelt keinen input#unkt
40160 if p=dz%-2 then 40180   : rem dz$ hat 2 nachkomma-stellen
40170 if p=dz%-1 then dz$=dz$+"0"
40180 dz%=len(dz$)
40190 return
50000 rem ++++++++ datei initialisieren ++++++++
50005 rem
50010 for m=1 to 13
50020 :   for r=1 to 9
50030 :       ak(m,r)=0
50035 :       ak(m,5)=1
50040 :   next r
50050 next m
50060 gosub 31000
50070 open2,8,15
50080 input#2,ff,fb$,sp,se   : rem fehlerkanal auslesen
50090 close 2
50100 print chr$(147)
50110 print chr$(144);ff;fb$;sp;se
50120 if ff=0 then 40        : rem alles o.k.
50130 if ff<> 63 then end    : rem file exists
50140 print chr$(147);      : rem bildschirm loeschen
50150 print "Es existiert bereits eine Datei"
50155 rem
50160 print "mit dem Namen AUTOKOSTEN          "
50170 print : print "Wenn diese geloescht werden soll      "
50180 print "bitte "+" F1 - Taste "+" druecken"
50190 get g$ : if g$="" then 50190
50195 rem
50200 if g$= "" then u$="$:" : print chr$(147) : gosub 31000
50210 get g$ : if g$="" then 50210
50220 if u$="$:" then 60
50230 end

```

Die Kommentare

Zeilen 00001 bis 00095: Das Programm stellt sich vor und macht sich schön. Außerdem will es wissen, ob es das erste Mal ist. Falls ja, werden evtl. existierende Dateien überschrieben.

Zeilen 00100 bis 00390: Die Hauptsache. Von hier aus wird der gesamte Ablauf gesteuert.

Zeilen 10000 bis 10995: Die Maske. Universell wie sie ist, wird sie sowohl zur Ein-, wie auch zur Ausgabe verwendet. Die Variable `cz(n)` steuert den Cursor bei der Ein- und Ausgabe.

Zeilen 11000 bis 11995: Die Eingabezeile. Wenn dies Unterprogramm aufgerufen wird, springt der Cursor in Zeile 22; dort wird die verlangte Eingabe eingeschrieben. Danach gehts zurück.

Zeilen 12000 bis 12995: Die Monatsnamen. Ein kleiner Luxus. Die Monatsnummer wird in Klartext übersetzt. An der richtigen Stelle der Maske erscheint der Monatsname.

Zeilen 20000 bis 20995: Die Monatsdaten werden hier aus der Tabelle geholt, gerundet, in Dezimalform mit zwei Stellen hinter dem Komma gebracht und in die Maske eingefügt.

Zeilen 21000 bis 21995: Die Berechnung. Das ist das Programm, das aus den notwendigen Rechenformeln entstanden ist.

Zeilen 22000 bis 22995: Die Jahreseingaben. Steuer- und Versicherungsbeträge werden abgefragt und in Monatsanteile zerlegt.

Zeilen 30000 bis 30990: Das Laden der Daten. Wie üblich bei Benutzung eines Diskettenlaufwerkes. In der zweiten Hälfte des Unterprogramms werden die eingelesenen Daten in die Arbeitstabelle geschrieben.

Zeilen 31000 bis 31990: Das Speichern der Daten. Auch das sollte uns nicht mehr unbekannt sein.

Zeilen 40000 bis 40190: Das Auf- und Abrunden. Die Routine zum Runden der Beträge ist nur deshalb so umständlich, weil damit der Vorgang durchschaubar wird. Normalerweise geht das mit der Formel für kaufmännisches Runden. Im zweiten Teil des Unterprogramms werden Beträge, die entweder volle DM sind, oder nur eine Stelle hinterm Komma haben, in die normale DM-Form gebracht.

Zeilen 50000 bis 50230: werden nur angesprungen, wenn man zum ersten Mal das Programm betritt. Die Datei wird vorbereitet. An dieser Stelle könnten noch verschiedene Dienstprogramme zum Formatieren, Directory-Lesen und ähnlichem eingebaut werden.

Ein Schlußwort: das Programm ist so geschrieben, das es (fast) beliebig gekürzt werden kann. Natürlich können Sie auch noch mehr Unterprogramme einbauen. Interessante Tools (=Programmier-Werkzeuge) finden Sie in allen erwähnten DATA-BECKER-Büchern.

4.3 Wie hoch wird die monatliche Belastung ?

```
*****  
*      MAN NEHME      :      *  
*****
```

```
*****  
*      HARDWARE      :      *  
*****
```

```
*      *      *  
*      COMMODORE 64      *  
*      Floppy 1541      *  
*      Monochromer Monitor      *  
*      Matrixdrucker      *  
*      *      *  
*****
```

```
*****  
*      SOFTWARE      :      *  
*****
```

```
*      *      *  
*      keine      *  
*      *      *  
*****
```

```
*****  
*      VORKENNTNISSE      *  
*****
```

```
*      *      *  
*      Gute bis sehr gute      *  
*      BASIC-Kenntnisse      *  
*      *      *  
*****
```

Jetzt wollen wir einen Teil der Leser aus dem Raum schicken. Sie haben 6 Richtige im Lotto ? Raus. Ihr amerikanischer Erbonkel hat Ihnen ein dickes Paket COMMODORE-Aktien vermachst ? Raus. Sie wollen sich ganz bestimmt nie ein Haus bauen ? Raus.

Gehören Sie zu den Verbliebenen, darf ich davon ausgehen, daß Sie erstens zukünftige/r Bauherr/in und zweitens weder reich noch arm sind. Mit anderen Worten: Sie werden Ihre vier Wände finanzieren lassen müssen. Eine Frage quält Sie sicher "Kann ich mir das leisten ?", "Wie hoch wird denn die monatliche Belastung ?"

Diese Berechnung ist eigentlich nicht sehr kompliziert, man braucht nur einen großen Block Karopapier, Bleistift und viel Zeit. Selbst bei geflissentlicher Benutzung eines Taschenrechners dauert es einige Stunden, bis die Zahlen für die je 12 Monate der nächsten, sagen wir mal, 25 Jahre vorliegen. Ihr C64 kann das viel schneller.

Nun werden Sie schnell zwei, drei Seiten weiterblättern um das Listing zu suchen. Aber ich muß Sie enttäuschen: kein Listing. Ich habe mir nämlich gedacht, daß Sie inzwischen soweit sind, ein solches Programm ganz alleine zu entwickeln. Gerade das Kapitel 4.2 enthält ja vieles, was Sie dazu brauchen.

Keine Angst, so ganz allein will ich Sie nicht lassen. Gehen wir doch einfach genauso vor, wie beim Programm "Autokosten".

Die Programmidee

Das Programm "Baukosten" soll das Berechnen der monatlichen Belastung durch Zinsen und Tilgung der Baudarlehen errechnen. Steuervorteile bleiben unberücksichtigt. Das Ergebnis wird auf Bildschirm und Drucker ausgegeben und kann

auf Diskette abgespeichert werden.

Die Problemanalyse

Was soll das Programm leisten ?

- 1) Daten erfassen
- 2) Daten berechnen
- 3) Daten ausgeben

Das Programm läuft etwa so ab:

- 1) Eingabe des Darlehens-Betrages
- 2) Eingabe eines eventuellen Guthabens (Beim Bausparen)
- 3) Eingabe des Zinssatzes
- 4) Eingabe des Beginns der Laufzeit
- 5) Eingabe des Endes der Laufzeit
- 6) Eingabe der anfänglichen Tilgungsrate
- 7) Berechnen der monatlichen Zinsen
- 8) Berechnen der monatlichen Belastung
- 9) Berechnen der Schulden am Monatsanfang
- 10) Berechnen der Schulden am Monatsende
- 11) Berechnen der Gesamttilgung pro Jahr
- 12) Berechnen der Gesamtzinsen pro Jahr
- 13) Berechnen der Annuität (=Zinsen + Tilgung pro Jahr)
- 14) Berechnen der Schulden am Jahresanfang
- 15) Berechnen der Schulden am Jahresende
- 16) Ausgeben der Daten aus 8) und 11) bis 15)
- 17) Drucken der Angaben aus 16)
- 18) Abspeichern der Daten

Berechnungen, die abgespeichert wurden, sollen - falls gewünscht - mit geänderten Parametern wieder durchgeführt werden. Besonders wegen dieser Option bekommt das Programm ein kleines Menue.

MENUE Baukosten

- (1) Berechnung einlesen

- (2) Parameter ändern
- (3) Neue Berechnung durchführen
- (4) Ergebnis drucken
- (5) Ergebnis speichern
- (6) Programm beenden

Bildschirmmasken

Davon brauchen wir im Programm "Baukosten" mehrere. Für die eigentlichen Daten müssen zwei entworfen werden.

Maske für die Daten-Eingabe

```
*-----*
*          BAUKOSTEN - BERECHNUNG          *
*-----*
```

EINGABE

```
Darlehen                = DM .....
Guthaben                = DM .....
Zinsen                  = %      ...
Tilgungs-Rate           = %      ...
```

```
Laufzeit Beginn Monat .. Jahr ....
```

```
Laufzeit Ende   Monat .. Jahr ....
```

Maske für die Daten-Ausgabe

```
*-----*
*          BAUKOSTEN - BERECHNUNG          *
*-----*
```

AUSGABE der ERGEBNISSE

```

*------(1)------(2)-*
Jahr           =      ....      ....

Anfangs-Schuld. = .....,..  .....,..
Annuitaet      = .....,..  .....,..
Monatsbelast.  = .....,..  .....,..
Zins/Jahr      = .....,..  .....,..
Tilgung/Jahr   = .....,..  .....,..
End-Schulden   = .....,..  .....,..

*-----*

```

Bei der Ausgabemaske habe ich mir das so vorgestellt, daß in der Zeile "Jahr" unter der (1) eine Jahreszahl eingegeben wird, danach füllt das Programm die Maske für das eingegebene Jahr und das folgende aus. Auf beliebigen Tastendruck werden die Werte der nächsten zwei Jahre ausgegeben usw.. Leider hat der 64er ja nur die Möglichkeit, 40 Zeichen pro Zeile auf die Mattscheibe zu bringen, sonst könnte man aus den Zeilen der obigen Maske Spalten machen und die Werte aller Jahre untereinander ausgeben.

Fürs Einlesen und Speichern gibt es auch ein Formular:

```

*-----*
*      BAUKOSTEN - BERECHNUNG      *
*-----*

```

(einlesen/speichern)

Name der Berechnung : (nd).....

Moment bitte

(nd)..... wird (geladen/gespeichert)

Die Klammern stehen für variable Texte, also die Texte, die je nach Anwahl aus dem Menue eingefügt werden. (nd) ist der Name, den Sie der Datei geben bzw. gegeben haben.

Variablenliste und Datei-Entwurf

NAME	DIMENSION	ART	LÄNGE	INHALT
db	DM 0 - 999999,99	N	max. 9	Darlehens-Betrag
gh	kleiner als db	N	max. 9	Guthaben
zs	% 0,01 - 19,99	N	max. 5	Zinssatz
aj	1900 - 2099	N	max. 4	Anfangsjahr
am	1 - 12	N	max. 2	Anfangsmonat
ej	1900 - 2099	N	max. 4	Endjahr
em	1 - 12	N	max. 2	Endmonat
mb		N	max. 8	monatl. Belastung
ms		N	max. 4	monatl. Zinssatz
tr		N	max. 4	anfängl. Tilgungsrate
x	50	N		max. Anzahl der Jahre
jz(x)		N	max. 9	Jahres-Zinsbelastung
jt(x)		N	max. 9	Jahres-Tilgung
an(x)		N	max. 9	Annuitaet
ja(x)		N	max. 9	Anfangs-Schulden
je(x)		N	max. 9	End-Schulden
y	12	N		Monate
mz(y)		N	max. 8	monatl. Zinsbetrag
mt(y)		N	max. 8	monatl. Tilgungsbetr.
ma(y)		N	max. 9	monatl. Anfangs-Schuld.
me(y)		N	max. 9	monatl. End-Schulden

nd\$		A	max.15	Name einer Datei
dd\$(1)	"DATEI EINLESEN"	A	14	Masken-Titel
dd\$(2)	"DATEI SPEICHERN"	A	15	Masken-Titel
er	5	N		Eingabe-Rubriken
ce(5)		N	1	Rubriken-Zeile
ar	7	N		Ausgabe-Rubriken
ca(7)		N	1	Rubriken-Zeile

Um den "REDIM.D ARRAY ERROR" auszuschließen, wird die Datei grundsätzlich mit 50 Datenfeldern dimensioniert. Sollte Ihre Bank das Darlehen länger als 50 Jahre laufen lassen, können Sie das "x" immer noch vergrößern.

Notwendige Rechenformeln

brauchen wir in Hülle und Fülle. Zunächst die notwendigen Umrechnungen der Eingabewerte. Dazu noch eine Anmerkung: theoretisch wäre es schön, wenn man eine konstante Monats-Belastung und die anfängliche Tilgungsrate frei wählen und eingeben könnte. Einfacher ist, wenn nur einer der beiden Werte eingegeben wird und der andere errechnet wird. Wir wollen es so halten, daß die Tilgungsrate gewählt wird, die monatliche Belastung variabel ist. Jetzt die Formeln:

```

zs = zs/100
tr = mt/100
ms = zs/12
mb = db*(zs+tr)/12
ja(aj) = db-gh
jjz(aj) = ja(aj)*zs
jt(aj) = 12*mb-jz(aj)
an(aj) = jz(aj)+jt(aj)
ma(am) = ja(aj)

```

```

mz(am) = ma(am)*ms
mt(am) = mb-mz(am)
me(am) = ma(am)-mt(am)
ma(am+1) = me(am)
mz(am+1) = ma(am+1)*ms
...usw.usw. bis 12
je(aj) = me(12)
jz(aj) = mz(am)+mz(am+1)+...+mz(12)
jt(aj) = mt(am)+mt(am+1)+...+mt(12)
je(ej) = 0
me(em) = 0

```

Das wars auch schon. Vielleicht wundern Sie sich darüber, daß zwei verschiedene Berechnungen für den Jahres-Zinsbetrag bzw. den Jahres-Tilgungsbetrag aufgeführt sind. Der Grund: Anfangs- und Endjahr der Laufzeit haben - darlehentechnisch gesehen - weniger als 12 Monate und werden daher anders berechnet, als ganze Jahre. Sie können natürlich so tun, als ob jedes Jahr eine andere Anzahl Monate hat, die bei der Berechnung angegeben wird. Dann benötigen Sie nur die Formel, die die Monatswerte aufaddiert.

So, jetzt dürfen Sie den Computer anschalten und mit dem Codieren beginnen. Falls Sie wider Erwarten Schwierigkeiten bekommen, greifen Sie zu Büchern, die Ihnen weiterhelfen können (z.B. die C64-Bücher aus dem Hause DATA BECKER).

4.4 Erster, Zweiter, Dritter

```
*****
*      MAN NEHME      :      *
*****

*****
*      HARDWARE      :      *
*****
*
*      COMMODORE 64      *
*      Floppy 1541      *
*      Monochromer Monitor      *
*
*****

*****
*      SOFTWARE      :      *
*****
*
*      DATAMAT      *
*
*****

*****
*      VORKENNTNISSE      *
*****
*
*      keine      *
*
*****
```

Merkwürdig: ein unerwartet hoher Anteil der - als "Stubenhocker" verschrienen - Computerfans treibt als zweites und oft auch erstes Hobby aktiv Sport. Surfen, Fahrradfahren und Laufen scheinen die Favoriten zu sein. In

Sportvereinen sind Computer-Besitzer inzwischen gefragt, bietet doch ein Verein reiche Betätigungsfelder für Statistik aller Arten: Mitgliederkartei, Beitragswesen, Rundbriefe gehören zum alltäglichen Treiben eines Computer-Sportlers. Den Höhepunkt im Leben eines solchen Hobby-Programmierers ist DAS Sportfest des Vereins, oft ein Volkslauf oder Marathon und am liebsten mit internationaler Beteiligung. Da kann der Mensch mit dem Computer so richtig zeigen, was in ihm steckt.

Gewitzte 64er-User im Teenager-Alter finden übrigens im Vereinsleben einen willkommenen Nebenverdienst. Die Vorstände geben neuerdings ihren Verwaltungskram zwecks Erledigung außer Haus. Auf die Gefahr hin, diese Einkommensquelle zum Versiegen zu bringen, will ich doch wenigstens für den Fall "Sportfest-Auswertung" eine sehr bequeme Problemlösung anbieten.

Der Fall:

Der TuS 89 Waldhausen (300 Mitglieder; Turnen, Leichtathletik, Fußball und Handball) liegt seit Jahren voll im Trend; schon bevor die Jogging-Welle über den Atlantik schwappte, lief man in Waldhausen. Somit hat der Volks-Marathon Waldhausen Tradition. Dieses Jahr feiert man ein kleines Jubiläum: zum 10. Mal nehmen Männer, Frauen, Jungen und Mädchen aller Altersklassen 42 Kilometer Strasse unter die Schuhsohlen. Ehrgeiz der Waldhausener war schon immer die große Teilnehmerzahl, diesmal soll außerdem das Feld mit prominenten Marathonläufern und -läuferinnen verfeinert werden. Schließlich hat das Fernsehen sein Erscheinen angekündigt und wird den Namen des Vereins und der Stadt berühmt machen.

Die Zahlen:

- rund 1.200 Teilnehmer

- davon etwa 40% Frauen und Mädchen
- verteilt auf 5 Altersgruppen

Das Problem:

Nicht zuletzt wegen des Fernsehens, aber auch um dem Ganzen einen möglichst professionellen Anstrich zu geben, haben sich die Veranstalter geschworen, daß maximal eine Stunde nach dem Eintreffen des letzten Läufers das Gesamtergebnis vorliegen soll. Dazu soll jeder Teilnehmer eine Urkunde bekommen, die seine Position im Gesamtklassement und in seiner Altersgruppe, sowie die Laufzeit enthält. Das ergibt nach Adam Riese:

- 13 Ergebnislisten und
- über 1.000 Urkunden

Die Lösung

zeichnet sich ab, als das TuS-Mitglied Erwin Taste, C64-Besitzer und Schüler der 12. Klasse des örtlichen Gymnasiums, in einer der Vorbereitungssitzungen behauptet, mit seinem Computer wäre das alles überhaupt kein Problem.

Alle Teilnehmer würden in einer Datei gespeichert. Indexfeld (also: Sortier-Kriterium) sei beim Erstellen der Starterliste die Startnummer, die in der Reihenfolge der Anmeldungen zugeteilt würde. Diese STARTER-LISTE diene

- den Teilnehmern zum Überprüfen ihrer Daten
- den Zeitnehmern und Auswertern zum Erfassen der Laufzeiten

... sagte Erwin Taste und keiner der Anwesenden verstand ein

Wort. Vor lauter Ehrfurcht erteilten die Funktionäre Erwin das Mandat eines stellvertretenden Vorsitzenden des Organisations-Komitees und beschloßen, ihm in allen Fragen der Datenverarbeitung freie Hand zu lassen. So begann Erwin Tasse mit den nötigen Vorarbeiten. Als erstes entwarf er

Die Maske.

Die enthält zwangsläufig folgende Felder:

- Startnummer (als Eingabeparameter)
- Name
- Vorname
- männlich/weiblich
- Altersklasse
- und: Laufzeit

Ansonsten braucht die Maske keinen irgendwie gearteten Schnickschnack, da sie eh nur auf dem Monitor benutzt wird.

sportfest

start-nr. : 00012

name : Kaminski

vorname : Edeltraud

m/w : w

klasse : Senioren

ergebnis : .. h .. min .. sec

So belegt die Maske 55 Zeichen. Es passen also theoretisch fast 1600 Datensätze auf eine Diskette. Warum unbescheiden sein ? Die benötigten 1200 werden also ausgewählt. Erwin nennt die Maske "sportfest", weil er sich denkt, daß sie ihm auch bei anderen Veranstaltungen dienlich sein könnte.

Erste Aufgabe der Maske ist, wie gesagt, das Erfassen der Anmeldungen. Der TuS Waldhausen hat lange für den Lauf geworben und auch erreicht, daß sich über 80% der Teilnehmer schriftlich angemeldet haben. Das ist sehr praktisch, denn so können alle Daten in Ruhe eingegeben und geprüft werden. Bei dieser ersten Eingabe wird das Feld Ergebnis mit ausgefüllt (was nur beim ersten Datensatz eingetippt werden muß, hinterher wird es einfach mit "RETURN" übernommen).

Der erste Nachmittag am Terminal überzeugt unseren Freund davon, daß das Daten-Eingeben eine so blöde Arbeit ist, daß andere Vereinsmitglieder eigentlich helfen könnten. Bald findet sich ein freundlicher Helfer und nach drei Sitzungen zu je 1,5 Stunden sind 1.000 Teilnehmer erfasst.

Die Pointerdateien:

Wie aber sollen die 13 Ergebnislisten entstehen ? Ganz einfach: aus der Gesamtliste werden im Menüpunkt "Datei sortieren" Listen herausgefiltert, die nur noch Teilnehmerdaten enthalten, die bestimmte Kriterien erfüllen. Diese Pointerdateien gibt es für

- 1) alle Teilnehmer (=Gesamtliste);
- 2) alle weiblichen bzw.
- 3) männlichen Teilnehmer;
- 4) für die männliche und
- 5) die weibliche B-Jugend
- 6) die A-Jugend der Knaben,

- 7) sowie der Mädels;
- 8) für die Altersklasse 20 - 30 Jahre der Herren
- 9) und der Damen;
- 10) die der 30 - 40jährigen Männer und
- 11) 30 - 40jährigen Frauen;
- 12) dazu die männliche Senioren und
- 13) die entsprechenden Damen

Diese Pointerdateien werden unter sinnfälligen Namen abgespeichert; z.B. "startweibajug" oder "startweibsen". Die merkwürdigen Kürzel ergeben sich aus der Tatsache, daß die Namen maximal 15 Zeichen haben dürfen. Dazu wird noch eine Gesamtliste hergestellt, die alphabetisch sortiert ist. Erwin braucht zum Herstellen der Pointerdateien gut 3 Stunden, bekommt aber für diese Zeit-Investition die Mittel in die Hand, die Auswertung des Rennens in Minutenschnelle auswerfen zu können.

Der Listen-Ausdruck

muß ebenfalls vor Beginn des Rennens vorbereitet werden. Unter dem Menüpunkt "Datei auswerten" macht sich Erwin Taste Gedanken über das Aussehen der Starter- und der Ergebnislisten. Er kommt zu dem Schluß, daß eine Auswertung reicht. Dabei werden die Teilnehmerdaten in eine Druckzeile verlegt. Das sieht dann so aus:

 volksmarathon 84 tus waldhausen

nr.	vorname/name		klasse	ergebnis
00001	Christa	Vahlensiek	w 30 - 40
00002	Heliodoro	Salazar	m 30 - 40
00003	Addi	Pumann	m 20 - 30
00004	Erika	Pumann	w 20 - 30
00005	Sabine	Pumann	w B-Jugend
00006	Karl-August	Norpolth	m Senioren
00007	Norbert	Stampfer	m A-Jugend
00008	Corinna	Pietsch	w B-Jugend
00009	Rainer	Bartel	m 30 - 40
00010	Patrick	Schiefelbein	m B-Jugend
00011	Paul	Schiefelbein	m Senioren
00012	Edeltraud	Kaminski	w Senioren
00013	Petra	Petel	w 20 - 30

Sie sehen, die Starterliste ist nach Startnummern sortiert und in der Spalte "Ergebnis" sind nur Platzhalter gedruckt, hier können die Zeitnehmer später die Laufzeiten eintragen.

Nun kommt der Drucker zum Zuge. Mit Hilfe der Auswertung wird gedruckt, was das Zeug hält: mehrere Exemplare der Gesamtliste und der verschiedenen Listen aller Altersklassen. Es können selbstverständlich beliebig viele Listen gedruckt werden; zum Aushängen, zum Verteilen an Presse und Fernsehen, fürs Programmheft etc.; für den eigentlichen Ablauf sind nur die Gesamtlisten relevant.

Die Zeitnahme

Wir kommen nun zum Problem der großen Zahl. Man stelle sich vor, die 1.200 Teilnehmer-Ergebnisse müßten sozusagen "sequentiell", also nacheinander erfaßt werden, das System bräche zusammen (ob mit oder ohne Computer). Deshalb geht man normalerweise hin und stellt ganze Kompanien von Zeitnehmern auf, die jeweils für ein bestimmtes Segment zuständig sind.

Bei Volksläufen, die ohne Computer ausgewertet werden, kann es passieren, daß die Teilnehmer nicht mehr nachkommen, weil zuviele Teilnehmer gleichzeitig über die Ziellinie stolpern. Deshalb werden bei üblichen Veranstaltungen besondere Zusatzzeitnehmer dazusetzt, die im Krisenfall tätig werden.

Erwin Tasse hat sich etwas besonderes dazu einfallen lassen. Seine Teilnehmer sind mit (geliehenen) Taschenrechnern ausgerüstet, die solche kleinen Papierstreifen drucken können. Damit wird nicht gerechnet, sondern nur Startnummer und Laufzeit erfaßt. Die Papierstreifen werden von Boten zum Computer getragen und dort weiter verwendet.

Nachdem alle Läufer, die nicht aufgegeben haben, durchs Ziel gekommen sind und ihre Zeiten erfaßt wurden, gehts an die

Ergebnislisten.

Um den Index zu sichern, wird nun im DATAMAT-Menue der Punkt "Programm beenden" gewählt. Dann gehts wieder zum "Datei sortieren". Sortier-Kriterium Nr.1 wird nun die erzielte Zeit. Es entsteht, zunächst nur im Rechner, eine Liste, die mit dem/der schnellsten Läufer/in beginnt und mit dem/der Langsamsten endet.

Mit der alten, hoffentlich gespeicherten Auswertungs-Datei

werden Resultats-Listen hergestellt - genau wie anfangs die Starter-Listen. Das dauert ein bißchen, ist aber immer noch schneller als jede Methode, die ohne Computer arbeitet.

So, alles klar. Die Listen sind im Prinzip fertig, sie müssen nur noch gedruckt werden. Zeitspanne zwischen dem Eintrudeln des letzten Läufers und dem veröffentlichen der ersten Ergebnisse: rund 45 Minuten.

Die Urkunden

Wir kommen zu dem Punkt, der am meisten Handarbeit verlangt. Das liegt daran, daß mit DATAMAT nicht automatisch die Platz-Ziffer berechnet werden kann. Aber wenn so schöne Vordrucke da sind (siehe nächste Seite), ist auch das kein Problem. Geschrieben werden die Urkunden mit dem Textverarbeitungs-Programm TEXTOMAT, das eine Schnittstelle zum DATAMAT hat, d.h. TEXTOMAT liest DATAMAT-Dateien und setzt die Daten an die richtige Stelle im Text. Natürlich gehört der Text auf ein Urkunden-Formular mit Vereinswappen und allem Drum und Dran. Die Waldhausener haben sich solche Urkunden auf Endlospapier drucken lassen, das ist schön und praktisch zugleich.

P.S.:

Der 10. Volksmarathon war ein voller Erfolg. Das Wetter war herrlich, eine gewaltige Zuschauermenge säumte die Strecke und das Fernsehen brachte in der Sportschau einen fünfminütigen Bericht.

Alles hatte zu allseitiger Zufriedenheit geklappt. Die einzige Panne auf dem Computersektor gab es, als Erwin Taste versehentlich vergaß, nach der Eingabe von Teilnehmer-Daten den Programmpunkt "Beenden" zu durchlaufen; die Index-Datei war futsch und mußte am nächsten Tag wieder hergestellt werden. Aber sonst, alles prima. Die Ergebnisliste lag gut 1 Stunde nach Ende des Rennens vor und die ca. 1.050 Läufer, die das Ziel erreicht hatten konnten ihre persönliche

Urkunde schon drei Tage später per Post in Empfang nehmen. Und beim nächsten Mal wird Erwin Tasse mit zwei Computern, vier Floppys und zwei Druckern (ein Typenrad-Drucker für die Urkunden !) arbeiten, da kann das Ausdrucken der Urkunden gleichzeitig mit dem Drucken der Listen begonnen werden.

4.5 Nebenbei ein Taschenrechner

```
*****
*      MAN NEHME      :      *
*****

*****
*      HARDWARE      :      *
*****
*
*      COMMODORE 64      *
*      Speichermedium      *
*      Sichtgerät      *
*
*****

*****
*      SOFTWARE      :      *
*****
*
*      keine      *
*
*****

*****
*      VORKENNTNISSE      *
*****
*
*      Commodore-BASIC-Kenntnisse      *
*
*****
```

Liegt neben Ihrem C64 auch immer ein Schreibblock zum Rechnen oder gar ein Taschenrechner? Eigentlich ärgerlich, da steht die teure Anlage mit schier unerschöpflichen Rechenkapazitäten vor Ihrer Nase, aber "cosinus(90-(36*1024)/49)" müssen Sie zu Fuß berechnen. Ärger kann kreativ sein, und viele waren in dieser Frage schon kreativ.

Viele Wege führen nach Rom. Und viele Wege führen auch zum Software-Taschenrechner auf dem 64er. Die Ansätze sind durchaus unterschiedlich, das Problem ist immer dasselbe: beliebige Rechenoperationen sollen mit beliebigen Werten komfortabel ein- und ausgegeben werden.

Die Lösungswege

1. Die Tastatur simuliert einen Taschenrechner.
2. Auf dem Bildschirm wird eine Tastatur dargestellt, die Eingabe erfolgt über die Cursortasten.
3. Eine Unterprogramm in Maschinsprache kann, während das BASIC-Programm läuft, aufgerufen werden.

1. Tastatur-Taschenrechner

Dieser Taschenrechner beherrscht die vier Grundrechenarten, kann Ergebnisse runden und in einen Konstantenspeicher übernehmen. Einzelne Eingaben und der Speicher können gelöscht werden.

```
10000 rem *** c64 als taschenrechner ***
10010 poke 53281,0 : poke 53280,0
10020 poke 53272,23 : print chr$(150)
10030 u$="+++++---- TASCHENRECHNER +---+---+---+---"
10040 print chr$(147);
10050 print u$ : print
10060 print "Auf wieviele Stellen nach dem Komma soll";
10070 print "gerundet werden ( 1 - 8 ; 9=unbegrenzt )";
10080 input in$ : in=val(in$) : if in>0 and in<10 then 10100
10090 goto 10040
10100 print chr$(147); : print u$ : print
10110 sm$=""
10120 get t$ : if t$="" then 10120
10130 t=asc(t$)
10135 if t=43 or t=45 or t=42 or t=47 then 10310
10140 if t=chr$(136) and sm$="" then sm=0 : k=0 : goto 10100
10150 if t=chr$(133) then k=sm : gosub 10500
10155 if t=chr$(134) then t$=str$(k):goto10300
10160 if t=chr$(135) then gosub 10500
10170 if t=46 then 10300
10180 if t <48 or t >57 then 10110
10300 print t$; : sm$=sm$+t$ : goto 10120
10310 op=val(sm$) : print chr$(0); : print tab(10)"=";
10320 if t=43 then sm=sm+op
10330 if t=45 then sm=sm-op
10340 if t=42 then sm=sm*op
10350 if t=47 then sm=sm/op
10360 if in<>9 then gosub 10400
10370 print sm
10380 goto 10110
10400 sm=(int(sm*10^in+0.5))/10^in
10410 return
10500 print tab(20)"Konstante=";k
10510 return
```

Beachten Sie bitte: Operand (1) eingeben, "+" drücken, Operand (2) eingeben, dann die Operation (+;-;* oder /) und schon erscheint das Ergebnis auf der Mattscheibe. Wenn Sie danach die F1-Taste drücken, wird das Ergebnis in den Konstanten-Speicher übernommen, der mit der F3-Taste gelesen und mit F7 gelöscht werden kann. Fehlerhafte Eingaben können ebenfalls mit der Funktionstaste 7 auf Null gesetzt werden.

2. Bildschirm-Cursor-Taschenrechner

Das Programm entnehmen Sie bitte dem - auch sonst für Selbstprogrammierer sehr - empfehlenswerten DATA-BECKER-Buch "64 für Profis" (Seite 51 bis 92). Daß für so eine leichte Aufgabe so viele Seiten (über 40 !!) gebraucht werden, liegt daran, daß bis ins Detail erklärt und vorgeführt wird, wie das Taschenrechner-Programm entwickelt und das Listing übersichtlich aufgebaut wurde.

Ein paar Einzelheiten zum Appetit-Anregen:

Auf dem Monitor erscheint die graphische Darstellung eines Taschenrechners, mit den Cursor-Tasten werden (a la "Maus") die Tastenfelder angewählt, das Ergebnis erscheint in dem simulierten Display. Sehen Sie mal, wie professionell das aussieht !

<div style="border: 1px dashed black; width: 100%; height: 100%; position: relative;"> <div style="position: absolute; top: 5px; left: 5px; border-bottom: 1px dashed black; width: 95%;"></div> <div style="position: absolute; top: 5px; right: 5px; border-right: 1px dashed black; width: 5%;"></div> <div style="position: absolute; bottom: 5px; left: 5px; border-top: 1px dashed black; width: 95%;"></div> <div style="position: absolute; bottom: 5px; right: 5px; border-left: 1px dashed black; width: 5%;"></div> </div>	T= ALL CLEAR C= CLEAR E= EXPONENT S= ENDE																				
DATA BECKER CBM 64																					
<table border="0" style="width: 100%; border-collapse: collapse;"> <tr> <td style="border: 1px dashed black; padding: 2px 5px;">17</td> <td style="border: 1px dashed black; padding: 2px 5px;">18</td> <td style="border: 1px dashed black; padding: 2px 5px;">19</td> <td style="border: 1px dashed black; padding: 2px 5px;">1C</td> <td style="border: 1px dashed black; padding: 2px 5px;">1T</td> </tr> <tr> <td style="border: 1px dashed black; padding: 2px 5px;">14</td> <td style="border: 1px dashed black; padding: 2px 5px;">15</td> <td style="border: 1px dashed black; padding: 2px 5px;">16</td> <td style="border: 1px dashed black; padding: 2px 5px;">1X</td> <td style="border: 1px dashed black; padding: 2px 5px;">1:</td> </tr> <tr> <td style="border: 1px dashed black; padding: 2px 5px;">11</td> <td style="border: 1px dashed black; padding: 2px 5px;">12</td> <td style="border: 1px dashed black; padding: 2px 5px;">13</td> <td style="border: 1px dashed black; padding: 2px 5px;">1+</td> <td style="border: 1px dashed black; padding: 2px 5px;">1-</td> </tr> <tr> <td style="border: 1px dashed black; padding: 2px 5px;">10</td> <td style="border: 1px dashed black; padding: 2px 5px;">1.</td> <td style="border: 1px dashed black; padding: 2px 5px;">1E</td> <td style="border: 1px dashed black; padding: 2px 5px;">1=</td> <td style="border: 1px dashed black; padding: 2px 5px;">1S</td> </tr> </table>	17	18	19	1C	1T	14	15	16	1X	1:	11	12	13	1+	1-	10	1.	1E	1=	1S	
17	18	19	1C	1T																	
14	15	16	1X	1:																	
11	12	13	1+	1-																	
10	1.	1E	1=	1S																	

3. Maschinenroutinen-Taschenrechner

Eine kleine Überraschung ! Der 64er hat doch einen eingebauten Taschenrechner ! "Aber wo ?" werden Sie fragen. Die Antwort: im BASIC-Interpreter steckt eine Routine mit dem Namen "FRMEVL", die genau die Aufgaben eines Taschenrechners erfüllt. Natürlich können Sie nicht einfach GOTO \$AD9E eingeben, Sie müssen die Betriebssystem-Routine in ein Maschinensprache-Programm einbinden. Außerdem brauchen Sie noch die Adressen der verschiedenen Routinen für die Rechenoperationen. Wie das geht, sagen Ihnen zwei weitere DATA-BECKER-Bücher: "64 intern", das allerdings schon einige Vorkenntnisse bis hin zum Betriebssystem voraussetzt, und das Buch "Maschinensprache", das Ihnen helfen kann, selbst in Assembler zu programmieren.

4. Sonstiges zum Thema Taschenrechner

In DATA BECKERs Grosser 64er Programmsammlung finden Sie auf Seite 119 eine sehr interessante Lösung: aus Ihren Eingaben schreibt sich der Computer selbst eine BASIC-Zeile, die die gewünschte mathematische Operation enthält. Eine echte Alternative zu der oben beschriebenen Lösung 1.

Wenn Sie weniger das Bedürfnis haben, die vier Grundrechenarten zu strapazieren (nach dem Motto: bis 10.000 rechne ich alles im Kopf - herzlichen Glückwunsch !!), sondern häufig kompliziertere Operationen durchführen wollen/müssen, empfehle ich Ihnen den zu Unrecht ziemlich unbekanntem BASIC-Befehl DEF FN x(y) unter die Lupe zu nehmen. Wenn es auch auf den ersten Blick so aussieht, als ob man damit nicht viel ausrichten kann, werden Sie staunen, was passiert, wenn Sie mehrere (bis zu 10) definierte Funktionen ineinander schachteln. Fakultäts-Berechnung, Umrechnungen Sinus auf Grad und viele andere Probleme lassen sich so lösen. Wie gesagt, wenn Sie so etwas oft brauchen, lohnt es sich sogar, eine Formel-Bibliothek anzulegen, die jeweils in Ihre BASIC-Programme aufgenommen wird.

5. Daten aller Arten

Sie können Daten aller Arten in Formblättern eintragen, diese nummerieren, abheften und zig Ordner damit füllen. Sie können dasselbe aber auch mit Ihrem COMMODORE 64 erledigen.

Aus dem Kapitel 2.3 wissen Sie, was mit Datenverwaltungs-Software alles möglich ist. Konkrete Beispiele werden dieses Wissen vertiefen. Und, damit Sie gleich darauf vorbereitet sind, die folgenden Programm-Ideen basieren fast ausschließlich auf der Anwendung des Software-Pakets DATAMAT von DATA BECKER. Ich habe einfach kein vergleichbares Produkt für einen ähnlichen Preis gefunden. Sicher können Sie auch andere Datenverwaltungs-Programme anschaffen, aber warum einen Rolls Royce kaufen, wenn ein VW-Bus den größeren Kofferraum hat ?

Alle beschriebenen Anwendungen bewegen sich in Quantitäten, die den Bedürfnissen eines durchschnittlichen Haushalts entsprechen; übersteigt Ihr Mengenbedarf diese Grenze, werden Sie früher oder später zu anderer Hardware und, beinahe zwangsläufig, zu aufwendigerer Software greifen müssen. Parallel dazu steigen dann auch die Kosten.

Der DATAMAT ist für 99,- zu haben und braucht an Hardware Floppy 1541 und Drucker. Da diese Voraussetzungen für die meisten Programm-Ideen gelten, werden Sie auf den gewohnten Block "MAN NEHME ..." weitgehend verzichten müssen.

Sicherlich ist auch für Sie ein Vorschlag dabei, auf den Sie unbewußt schon gewartet haben. Blättern Sie das Kapitel einfach durch, wer weiß, wo Sie hängenbleiben.

5.1 Rezeptkartei für Hobbyköche

Ich weiß nicht, ob es für oder gegen mich spricht, aber eine der ersten Ideen für eine Anwendung des DATAMAT war, mein persönliches Kochbuch zur Datei zu machen. Sie müssen wissen, das meine Kochkünste weithin berühmt und meine selbstgemachten Nudeln einfach sensationell sind. Also ging ich frisch ans Werk und entwarf zunächst die Eingabemaske. Aber, oh Schreck, das Programm antwortete "Datensatz zu lang". Gut, ein paar Zeilen zu kürzen, ist keine Affäre.

```
-----  
*                REZEPT - KARTEI                *  
-----  
  
(01)  REZEPT : spaghetti bologn. |  
      ZUTATEN      |  
  
(02)  Rindgehacktes.suppengemuesel  
(03)  Bruehe.rotwein.tomaten.    |  
(04)  kreme fraiche.zwiebeln     |  
(05)  Knoblauch.spaghetti.oel    |  
(06)  Salz.pfeffer.basilikum     |  
  
(07)  ZUBEREITUNG : fleisch an-  |  
  
(08)  braten.zwiebeln & knoblauch|  
(09)  und alle gemuese.tomaten dal  
(10)  zu.mit fluesigkeit angiesl  
  
(11)  FUER 041 PERSONEN
```

(250 Zeichen/Datensatz; 150 Datensätze; Index: Feld 1)

Sehen Sie selbst, so kann man natürlich kein Rezept beschreiben, das Limit von 253 Zeichen pro Datensatz ist einfach zu niedrig dazu. Die Enttäuschung ist groß. Was tun ? Wir müssen einen ganz anderen Ansatz wählen. Den will ich Ihnen nun schrittweise erklären.

Ein Rezept besteht normalerweise aus zwei Teilen: die Liste der Zutaten und der Beschreibung der Zubereitung. Wie lassen sich diese zwei Bestandteile auseinanderdividieren. Ein Beispiel:

Ihr Gatte hat von einem Umtrunk in der Firma ein paar Lieblingskollegen mitgebracht. Alle sind hungrig und wissen, daß es bei Ihnen immer was Leckeres zu beißen gibt. Ihre Ehre als Köchin steht auf dem Spiel. Im Kühlschrank liegen zwei einsame Eier rum, ein bißchen Butter und ein Stück Mortadella. Ansonsten gibt es im Vorratsschrank noch ein Döschen Spargel, eins mit Ananas und Toastbrot.

Sie gehen schnell zum C64, laden DATAMAT und die Datei "zutaten". Eine selektierte Liste aller Rezepte, die man mit dem Vorhandenen herstellen kann, erscheint. "Ich renn doch nicht jedesmal zum Computer, wenn Gäste kommen ", werden Sie einwenden. Brauchen Sie auch nicht. Die eben beschriebene Arbeit können Sie natürlich vorher erledigen. Aber darauf kommen wir noch zurück.

Aus der Liste suchen Sie sich ein Rezept aus und schreiben sich den Namen auf. Nun laden Sie die Datei "zubereitung" und lassen sich die notwendigen Angaben ausdrucken.

Es ist selbstverständlich blödsinnig, erst in solchen Situationen ans Gerät zu treten. Schauen wir uns doch mal an, wie man sich diese Rezeptkartei anlegen kann.

 Datei "zutaten"

Das Material können Sie mit folgender Maske ganz gut erfassen :

```

-----
*          REZEpte    (1)  ZUTATEN          *
-----

(01) NAME    Spaghetti bolognese    |
(02) FUER    041 PERSONEN

          ZUTATEN                      MENGE

(03) Spaghetti/uel    |    11000gr |
(05) Rinder gehacktes    |    1 200gr |
(07) Zwiebeln/knoblauch |    ka.2/1 |
(09) Dose tomaten    |    1 1 |
(11) Mohre/poree/seller|    1 |
(13) Bruehe/rotwein    |    1 3/4 |

          GEWUERZE

(15) SALZ ja PFEFFER sch SENF nein
(18) PAPRIKA n MUSKAT ja
(20) SONST    nein    |
(21) KRAEUTER basilikum/petersilil
(22)          evtl.oregano    |
  
```

(251 Zeichen/Datensatz; 100 Datensätze; Index: Feld 1)

Die Eingabe-Felder für die Gewürze sind sehr platzsparend. Müßten sie im Klartext eingegeben werden, bräuchte man ca. 20 Zeichen mehr im Datensatz und wieder wäre das Limit

erreicht. Ändern Sie die Maske nach Ihrem Geschmack (im doppelten Sinne des Wortes).

Datei "zubereitung"

In dieser Datei speichern Sie die eigentlichen Rezepte. Die Eingabe-Maske :

```
-----  
*          REZEPTE (2) ZUBEREITUNG          *  
-----  
  
(01) NAME Spaghetti                          |  
(02) FUER 04 PERSONEN  
  
      ZUBEREITUNG  
  
(03) Viel wasser sprudelnd |  
(04) aufkochen lassen. salz |  
(05) dazu und schuss oel. |  
(06) kal dente kochen. |  
(07) abgiessen und abspuelen. |  
(08) in etwas butter schwenkel  
(09) h. ; |  
  
      ZEITEN  
  
(10) | | FUER TEIL | |  
(11) | | | |  
(12) | | | |  
(13) B-91 GESAMT
```

(231 Zeichen/Datensatz; 100 Datensätze; Index: Feld 1)

Auch hier gibt es wieder Platz-Probleme. Sie dürfen sich selber Gedanken darüber machen, wie Sie mit diesem Mangel fertigwerden.

Aussuchen und Drucken

Sie haben ja hoffentlich Ihre Rezepte schon eingegeben, denn - Sie ahnen es schon - jetzt kommt der Nutzen. Zuvor: sortieren müssen wir nichts, es gibt ja auch keine Felder, nach denen zu sortieren sich lohnt. Aber selektieren (=auswählen) lohnt sich. Sie können z.B. eine "eier"-Datei einrichten. Sie enthält alle Gerichte, bei denen Eier eine Hauptrolle spielen. Sie können nach Bestandteilen des Rezeptnamens auswählen, z.B. Grundrezepte oder Salate, Suppen usw..

Alle diese Pointerdateien (wenn Sie nicht mehr wissen, was das ist, schauen Sie bitte ins Kapitel 2.3) können Sie nun in beliebiger Reihenfolge ausdrucken, mit einer anderen Auswertung (auch das ein Begriff aus dem Kapitel 2.3) erstellen Sie Ihre Inhaltsverzeichnisse.

Hier gilt, wie bei allen anderen Datenverwaltungs-Problemen: Probieren geht über studieren.

5.2 Ordnung in der Tiefkühltruhe

Ordnung ist das halbe Leben. Aber eben nur die eine Hälfte, sagen manche. Ich finde, die haben recht. Man kann natürlich alles ordentlich aufbewahren oder anordnen, aber so ein kleines bißchen kreatives Chaos hat noch keinem geschadet. Bei der alltäglichen Arbeit ist es ein wenig anders.

Nehmen wir doch mal eine Tiefkühltruhe. "Willi, wo haste denn die Koteletts in der Truhe hingetan?" - "Unten rechts, Else!" und Else wühlt in den kalten Paketen, kriegt klamme Finger und hat schon gar keinen Hunger auf Koteletts mehr. Nun sollte in einer Tiefkühltruhe eigentlich von Hause aus eine gewisse Ordnung herrschen, aber Sie wissen ja, wie das ist: der Geist ist willig, ...

Nicht sehr angenehm, die Sucherei. Abhilfe bringt folgendes kleines DATAMAT-Beispiel.

Denken wir uns einen durchschnittlichen Tiefkühltruhen-Besitzer aus: es müßte sich um einen 5-Personen-Haushalt auf dem Lande handeln, der Haushaltsvorstand ist mittleren Alters und scheut sich nicht, auch mal ein Schwein selbst zu schlachten. Außerdem hat unser Muster-Haushalt einen recht ansehnlichen Garten, der ganz schöne Ernten abwirft. Die Truhe ist meistens voll.

Zwei Situationen sind typisch: das geschlachtete Schwein und der Monat Mai (mit Erdbeer- und Spargelernte). Da fällt viel Zeug zum Einfrieren an. Bisher ging das so: Gefriergut vorbereiten, in geeignete Behälter füllen, Etikett schreiben, draufkleben und ab in die Kiste. Was auf den Etiketten steht, enthält auch die Eingabemaske, die ich Ihnen vorab schon mal vorstellen möchte.

```

=====
=
=          TIEFKÜHL - TRUHE          =
=
=====
=
= (1) Nummer   : 40030 F             =
=
= (2) Inhalt   : 4Spargel           F =
= (3)          : 4in Stangen        F =
=
= (4) Menge    : 4010 F   Mass : 4Kilo F =
=
= (6) Eingelagert am : 401 H05 H84 F =
=
= (9) Haltbar bis   : 401 H11 H84 F =
=
=      Platz in der Truhe : 45A   F   =
=
=
=
=====

```

Nun sind das halbe Schwein, 10 Kilo Erdbeeren und etliche Kilo Spargel haltbar untergebracht. Vier Monate später ergibt sich dann der oben beschriebene Dialog zwischen Else und Willi.

Mal angenommen, Willi hat einen 64er, dann ist alles ganz anders. Erstens wurden die Etiketten vom Computer hergestellt und zweitens kann Willi mit seiner Tiefkühlkost-Datei noch andere tolle Sachen machen.

Unser Kandidat legt sich beispielsweise eine neu sortierte Datei des Truhen-Inhalts an: sortiert nach dem Verfallsdatum. Die wird monatsweise in Listen ausgedruckt und in der Küche aufbewahrt. Else weiß nun immer, was als nächstes aufgegessen werden muß.

Wie geht das ? So:

Voraussetzung:

- Datei "tiefkühlkost"
- Index abgespeichert (siehe DATAMAT-Handbuch) !!

Ablauf:

- DATAMAT laden
- "Datei sortieren" anwählen
- Dateiname eingeben
- Maskenname eingeben
- Punkt "sortieren" in der Kommandozeile anwählen
- "01" in das Feld Haltbarkeit/Jahr eingeben
- "02" in das Feld Haltbarkeit/Monat eingeben
- "03" in das Feld Haltbarkeit/Tag eingeben
- Fl drücken
- Punkt "ausführen" in der Kommandozeile anwählen
- Sortierte Datei benennen
- Abspeichern
- "Datei auswerten" anwählen
- Dateiname eingeben
- Maskenname eingeben
- Frage "Soll eine bestehende ..." mit "n" beantworten
- Zeilenlänge "80" eingeben
- Felder schieben bzw. löschen nach Wunsch
- Überschriftenzeile eingeben
- Wenn fertig, Fl drücken; bestätigen
- Frage "Soll eine bestehende Pointerdatei ..." mit "j" beantworten
- Name der Pointerdatei eingeben
- Dru en nach Anweisung

Sie erhalten eine Liste, die, nach Verfallsdatum geordnet, den Inhalt der Truhe zeigt.

TIEFKUEHL - TRUHE

15	11	84	002 Kilo	Spargel	03a
21	09	84	001 Kilo	Erdbeeren	03b
28	09	84	008 Stueck	Kotilletts	01a
01	06	84	004 Port.	Sauerbraten	01a

Ach so, in der Maske taucht bei mir das Feld "Platz" auf. Ich will noch kurz erklären, was es damit auf sich hat. Ich stelle mir vor, daß man die Aufteilung der Päckchen, Dosen und Beutel auf die einzelnen Körbe der Tiefkühltruhe hier in Koordinaten angeben könnte. Da meine "Truhe" nur das Gefrierfach eines Kühlschranks ist, konnte ich das nicht ausprobieren. Vielleicht testen Sie das Prinzip für mich. Berichten Sie mir doch über Ihre Erfahrungen !

5.3 Gartenpflege-Kalender

Garten und Computer - das paßt nicht zusammen, meine ich wirklich. Es gibt allerdings Fälle, in denen die zwei Komponenten vereinbar sind. Weil das wenige Fälle sind, will ich das Kapitel recht kurz halten.

Normalerweise hat der Pflanzenfreund im Gefühl, wann, wo und wie er sein Gemüse hinsetzt und welche Pflege er ihm angedeihen läßt. Nun soll es aber Leute geben, die zwar einen Garten besitzen, ihn auch lieben, aber trotzdem nicht wissen, wie es richtig geht. Hier kann DATAMAT helfen.

Ihre Datei heißt z.B. "garten". Ein Datensatz enthält den Namen der jeweiligen Pflanze und alle notwendigen Pflanz- und Pflegedaten, die Sie einschlägigen Gartenbüchern entnehmen. Sortieren Sie die Datei nach dem Datum und Sie haben den Gartenpflege-Kalender. So einfach ist das.

Für Besitzer großer Gärten bietet sich eine Datei an, in die eingetragen wird, wann welches Beet gedüngt, gepflegt oder sonst was wurde. Das schafft Überblick. Übrigens: große Gärtnereien arbeiten auf diese Weise schon lange mit Computern. Was denen recht ist, sollte Ihnen billig sein.

Landwirte gehören zu der Branche, die bisher noch recht wenig mit elektronischen Hilfsmitteln arbeitet. Aber das wird sich ändern. Sogenannte Nebenerwerbs-Landwirte z.B. können heute schon Nutzen aus Homecomputern ziehen: Erntemengen, Preise, Kosten usw. usw. lassen sich mit geeigneter Software prima erfassen und bearbeiten.

Eine Idee, die zum Teil ins Kapitel 6. ff passen würde, hätte ich noch anzubieten. Wissen Sie, was "Fruchtfolge" bedeutet? Dann wissen Sie auch, daß, wenn man diese Methode im Garten erfolgreich anwenden will, es unbedingt nötig ist,

über Jahre (wenn nicht Jahrzehnte) zu speichern, was auf welchem Beet angepflanzt wurde. Zeichnen Sie sich ein Schema Ihres Gartens und tragen Sie die Bepflanzung ein. Auch zur Planung eines neu anzulegenden Gartens eignet sich Computer-Grafik.

Nun will ich Sie mit diesen Anregungen allein lassen. Ich denke, Sie werden schon eigene Ideen aus dem Gelesenen destillieren.

5.4 Lagerliste für Handwerksbetriebe

Mein Schwager ist selbständiger, allein arbeitender Malermeister. Wenn es die Zeit erlaubt, helfe ich ihm bisweilen als Handlanger. Eines Tages, wir wollten gerade etwas aus dem Materiallager holen, klagte er, daß er nie genau wüßte, was denn nun im Lager rumstehe. "Ja führst du denn keine Lagerliste ?" fragte ich. "Natürlich, aber wie soll ich denn zum Beispiel Farbreste erfassen oder halbe Tapetenrollen und so was; da würde ich nur noch Listen schreiben."

So kam ich auf die Idee, den DATAMAT für eine ganz einfache Lagerverwaltung zu mißbrauchen. Wenn mein Schwager dann eines Tages sein Geschäft vergrößert und die ersten Gesellen einstellt, wird er sowieso zu professioneller Software greifen müssen. Allerdings bleibt ihm dann der EDV-Schock erspart.

Der Bestand wird wöchentlich erfaßt bzw. bereinigt. Die Lagerliste, die dabei entsteht, dient sowohl der Dokumentation vorhandener Materialien, als auch als Formular zum Eintragen. Dieser Rhythmus entspricht auch dem, der ohne Computer eingehalten wurde: samstags wird das Lager aufgeräumt.

Wohl gemerkt: die Datei umfaßt nur solches Material, das nicht vollständig verbraucht und somit Kunden nicht insgesamt berechnet wird. Ich will Ihnen ein Stück Liste zeigen.

```
* * * * *  
      lagerliste reste  
* * * * *
```

Rauhfaser

3 1/2 Roll	2 Rollen	Rest von
21 02 84	01 07 84	Baustelle
21.00	12.00	Internat

Etwas anderes für vergeßliche Handwerker:

Angenommen Sie sind Installateur und sollen bei einem Kunden einen Wasseranschluß in den Garten legen. Wissen Sie auf Anhieb, welches Material und welche Werkzeuge (außer den üblichen) Sie mitnehmen müssen ? Nein ? Dann schreiben Sie sich einfach für alle denkbaren Fälle sogenannte Check-Listen, die Sie vor der Fahrt zum Kunden Punkt für Punkt abhaken. Z.B. so:

```
+++++
+
+   AUFTRAG : Rauhfaser streichen      +
+
+   MATERIAL: Wandfarbe                () +
+           Abdeckfolie                () +
+           Abdeckpapier               () +
+           Kleister                   () +
+
+   WERKZEUG: Große Rolle              () +
+           Kleine Rolle                () +
+           Eckenpinsel                 () +
+           Schwamm                     () +
+           Farbeimer (2)               () +
+           Wassereimer                 () +
+           Kleiner Eimer               () +
+           (für Kleister)              +
+
+++++
```

Ein schlichtes Beispiel, zugegeben, aber bei schwierigeren Dingen hätten vermutlich meine Fachkenntnisse nicht ausgereicht.

Fazit: mit dem COMMODORE 64 und geeigneter Software kann der

selbstständige Handwerker sich an den Umgang mit dem Computer gewöhnen. Wenn der Laden läuft, muß ein Personal-Computer ins Haus. Der macht die Buchhaltung, die Personalverwaltung, das Rechnungs- und Mahnwesen, hilft beim Angebot-Erstellen und allen dazugehörigen Berechnungen und schreibt Netz- und Ablaufpläne. Dann kann sich die Frau Meisterin ganz dem Betriebsklima widmen ...

Zu diesem Thema möchte ich Ihnen das Buch "Computer fürs Geschäft" (von DATA BECKER) ganz besonders ans Herz legen.

5.5 Ein ganz persönliches Gesundheitsarchiv

Gesundheit - das ist ein ernstes Thema. Ich werde meine dummen Scherze auf ein Minimum reduzieren. Krank werden wir alle einmal. Oft ist es nur ein Schnupfen, manchmal eine schwerere Sache und - hoffentlich nie - etwas Unheilbares. Nehmen Sie als Beispiel einen Diabetiker: sein Leiden ist, mit den heute bekannten Methoden nicht zu heilen, es kann im besten Falle ein Zustand relativen Wohlbefindens erreicht werden. Diabetiker leben mit der ständigen Kontrolle ihrer Körperfunktionen. Inzwischen können Zuckerkrankte eine ganzes Stück Therapie selber leisten.

Wußten Sie, daß Homecomputer enorme Hilfe zur Selbsthilfe darstellen können ? Nicht umsonst gewann ein siebzehnjähriger Schüler beim 84er "Jugend forscht" einen Sonderpreis für ein Diabetiker-Softwarepaket.

Mit diesen Programmen ist es möglich, die gemessenen Blutzucker-Werte, Kreislauf und Puls nicht nur zu erfassen, sondern auch zu berechnen und zu bewerten. Mit den Eingaben über die Einhaltung vorgeschriebener Diät erstellt das Software-Paket eine Beurteilung des Tagesverhaltens. Wenn am Abend eine "0" herauskommt, weiß der Kranke, er hat sich richtig verhalten, eine "12" mahnt zu besserer Beachtung der Regeln. Das Programm enthält außerdem eine Informations-Datei, aus der der Anwender alle notwendigen, zum Teil mit seinen Werten aufbereitete, Informationen über die Krankheit.

In Kürze wird diese Diabetes-Software auch auf den gängigen Homecomputern, wie dem C64, laufen können. Als Betroffener sollten Sie sich (über den Hausarzt oder die Krankenkassen) näher informieren.

Eine bekanntere Computer-Anwendung im Gesundheits-Sektor

sind die Programme zur Risikoberechnung bzw. zum Berechnen der Lebenserwartung. Wenn Sie ein derartiges Programm irgendwo zum Abtippen finden, genießen Sie es mit Vorsicht ! Bei den Rechen- und Speicherkapazitäten von Homecomputern läßt sich keine ernsthafte Aussage aus Angaben über EB-, Trink-, Rauch- und sonstigen Lebensgewohnheiten machen. Einziger Nutzeffekt solcher Programme: wenn Ihre Lebenserwartung so niedrig ist, daß Sie das Flattern kriegen, werden Sie vielleicht das Rauchen einstellen.

Apropos Rauchen: das Bundesministerium für Gesundheit und Familie verschickt auf Anforderung eine Broschüre, die eine Anleitung bietet, wie man sich das Rauchen abgewöhnen kann. Die Methode ist so einfach, wie computer-geeignet. Der Kandidat erfaßt seine Rauchgewohnheiten (eine Zigarette nach dem Essen, ganz viele im Streß usw.) und trainiert sich darin, das Rauchen in den zigaretten-trächtigen Situationen zunächst einzuschränken und schließlich aufzugeben. Die Methode ist bewährt und ein Anreiz, sie in ein Programm umzusetzen.

Das Verhältnis zur Medizin ist durch allerlei Skandale und Verunsicherungen bei vielen leicht getrübt. Mancher besinnt sich auf die natürlichen Heilmethoden und Omas Hausmittelchen. Eine positive Entwicklung, wie ich finde. Inhalieren von Kamillendampf ist eben weniger riskant als Tablettenschlucken. Einen Nachteil hat die "Volks-Medizin": sie beruht weitestgehend auf Erfahrung und ist in ihrer Wirkung von Mensch zu Mensch völlig verschieden.

Für Menschen, die Hausmittel gezielt einsetzen wollen, ist besonders wichtig, die Wirkung von kalten und warmen Wickeln, von verschiedensten Tees und Kräutern im jeweiligen Krankheitsfall sorgfältig zu registrieren. DATAMAT hilft Ihnen dabei.

Sie erkennen ja inzwischen eine DATAMAT-Maske, wenn Sie sie sehen. Diese ist so einfach, daß sich größere Erklärungen erübrigen. Beim Gesundheitsarchiv ist es ähnlich wie bei den Rezepten: es ist Unsinn, den 64er anzuschalten, wenn Sie die Grippe kommen spüren. Arbeiten Sie vor. Tragen Sie Symptome ein und die möglichen Therapien. Drucken Sie sich daraus Ihr Gesundheitslexikon. Das ist sinnvoller.

Ein Hinweis: zur Vorsorge gehören auch Sport und Ernährung. Lesen Sie doch auch in den Kapiteln 5.8 und 5.9 nach, wie Ihnen der 64er helfen kann.

```

#####
$
$      Gesundheits-Archiv      $
$-----$
$ - 1- Krankheit      -Bluterguss      + $
$      Symptome      $
$ - 2- Fieber          -nein+ Puls  -===+ $
$ - 4- Blutdruck      -nor+ zu -mal+ $
$ - 6- Schmerzen      -      '      + $
$ - 7- Uebelkeit      -n + Ja/Nein $
$ - 8- Durchfall      -n + Ja/Nein $
$ - 9- Erbrechen      -n + Ja/Nein $
$ -10- Bewusstlos     -n + Ja/Nein $
$ -11- Sonstige       -Je nach Bereich + $
$ -12-                -sehr schmerzhaft + $
$      Therapie      $
$ -13- -Packung mit Essigsaurer Ion+ $
$ -14- Herde. Kuehlen. Schonen !    + $
#####

```

5.6 Egal, was Sie sammeln ...

Ich hoffe, wir sind uns einig, wenn ich behaupte, das Sammeln, gleich welcher Dinge, läßt sich rational nicht begründen. Es ist ja wohl auch ein menschlicher Urtrieb: Gegenstände suchen, zusammentragen, ordnen und katalogisieren. Wenn Sie also kein Jäger sind, so sind Sie vielleicht doch ein Sammler. Was Sie sammeln, ist gleich, die dazu gehörenden Tätigkeiten sind davon völlig unabhängig.

Viele Homecomputer-Besitzer sind über ihr Sammel-Hobby zum Gerät gekommen. Und so wundert es nicht, daß in den Computer-Magazinen oft Listings auftauchen, die Anwendungen zum Thema Sammeln beschreiben.

Ganz grundsätzlich betrachtet gibt es zwei Möglichkeiten des Sammelns:

- mit dem Ziel etwas vollständig zu besitzen und
- mit dem Anspruch die schönsten (besten, seltensten etc.) Stücke eines Gebietes zu besitzen

Zur ersten Kategorie zählen beispielsweise Briefmarken- und Münzsammlungen; zur zweiten Bibliotheken und Schallplattensammlungen. Die Probleme der Katalogisierung und Verwaltung sind deshalb unterschiedlich. Ich will an zwei Beispielen zeigen, wie Sie mit DATAMAT an Ihrer Sammlung arbeiten können.

Briefmarken-Sammlung

Um es möglichst einfach zu machen, wählen wir ein eng

zehnkampf olympia 1960 ne ja
republik brunelia

0.10b£	+9 02	8.50
0.50b£	+9 02	8.50
1.00b£	+9 02	11.00
2.00b£	00	

zehnkampf olympia 1960 ne ne
republik brunelia

0.10b£	+7 01	1.30
0.50b£	+7 01	1.00
1.00b£	00	
2.00b£	00	

Noch etwas. Viele Philatelisten betrachten Ihre Sammlung nicht nur als Hobby, sondern auch als veritable Wertanlage. Wenn Sie die Spalte "Preis" in Ihrer Datei auf dem letzten Stand (nach Katalog oder Angebot/Nachfrage) halten, haben Sie die Möglichkeit, innerhalb der Listen Summen zu ziehen. Unten steht dann der Wert Ihrer Marken.

Schallplatten-Sammlung

Wenn sich die Zahl Ihrer Platten der 500er-Grenze nähert, stehen Sie vor einer wichtigen Entscheidung: "Soll ich einfach weiter kaufen und auf jede Katalogisierung verzichten ?" oder "Soll ich nun beginnen, etwas Systematik in die Sache zu kriegen ?". Haben Sie sich für die erste Alternative entschieden, dürfen Sie getrost dies Kapitel verlassen. Im anderen Fall will ich Ihnen gerne ein paar Tips geben.

Wieso die Zahl 500 ? Ganz einfach: ungefähr ab hier wird es sinnlos, noch mit der Verwaltung der Sammlung anzufangen, die Arbeit der Eingabe wird einfach zu groß und damit zu langweilig. Außerdem lassen sich fünfhundert Platten noch in einigermaßen sinnfällige Dateien aufteilen. Diese Einteilung sollten Sie sowie so treffen. Ein Rechenexempel aus dem Bereich "Popmusik": 100 Platten mit je ca. 12 Titeln sind 100 Datensätze zu je 12 Feldern. Ein Feld enthält ca. 15 Zeichen. Also enthält eine Eingabemaske (12*15=) 180 Zeichen + 20 Zeichen für den Titel des Albums und 20 Zeichen für den/die Interpreten. Dazu 4 Zeichen für eine Laufnummer und weitere 20 für nähere Angaben über Erscheinungsjahr, Plattenfirma etc. Summasumarum: 244 Zeichen.

Eine mit dieser Maske erstellte Datei hat, vorausgesetzt die Diskette enthält keine weiteren Dateien, maximal 700 Sätze. Es ist also zweckmäßig, die komplette Sammlung auf mehrere Dateien zu verteilen. Etwa so:

- Pop
- Jazz
- Klassik

Oder noch feiner gesiebt:

- Hardrock
- Reggae
- Blues
- Folk
- Swing
- Modern Jazz
- Barock-Musik
- Opern
- Musicals
- Filmmusiken
- usw.

Einen gewichtigen Nachteil hat das Verfahren. Titel, die in

verschiedenen Dateien erfaßt wurden, lassen sich nicht verknüpfen. Wenn Sie beispielsweise alle Versionen von "Slow Boat to China" wissen möchten, können Sie nur alle Dateien durchgehen, die in Frage kommen, und zwar nacheinander. Das ist nicht so schön. Vermeiden läßt sich dieser Effekt dadurch, daß Sie Mehrfachnennungen zulassen. Wenn Sie - sagen wir mal - Bob-Dylan-Scheiben sowohl bei "Folk" als auch bei "Pop" eintragen. Ihrer Phantasie sind keine Grenzen gesetzt.

Die oben beschriebene Maske :

```

+-----+
|                   DISCO-LIST POP                   |
+-----+
| NO.           :.001.                                |
| ALBUM         :.Rubber Soul                        .  |
| INTERPRET     :.Beatles                            .  |
| LABEL        :.EMI          . NO. :.12/789.        |
| SEITE 1              SEITE 2                      |
| .Michelle           . .Norwegian Wood            .  |
| .Run for your li.   .Drive my car                 .  |
| .Nowhere man       . .Gigl                         .  |
| .In my life        . .                             .  |
| .                  . .                             .  |
| .                  . .                             .  |
+-----+

```

Wie immer sind Modifikationen Ihrerseits nicht nur möglich, sondern auch wünschenswert !

Es gibt Plattensammler, die aus Ihrem Hobby einen kleinen Nebenverdienst ziehen; sie bieten sich als mobile DiskJockeys für diverse Festivitäten an. Nun gehört zum Plattenauflegen mehr als nur eine ruhige Hand, die Atmosphäre der Party muß berücksichtigt, Plattenwünsche des Publikums erfüllt werden und Pausen dürfen nicht entstehen.

Ich will Ihnen ein Geheimnis verraten: Profis haben Standardlisten, das sind Aufzeichnungen über die Reihenfolge von Titeln, die bewährt sind oder im heimischen Studio ausprobiert wurden. Mit DATAMAT kann sich auch der Hobby-DJ solcher Werkzeuge bedienen. Ich muß wohl kaum noch im Detail beschreiben, wie solche Aufzeichnungen hergestellt werden, oder ?

Die eben genannten Profis führen auch noch andere Listen. Das sind - besonders bei den mobilen Diskotheken - Auswahlen aus dem Gesamtrepertoire, die man braucht, will man nicht alle zehntausend Scheiben mitschleppen.

Zum Thema Disco liessen sich noch viele Anwendungen anführen, von der computer-gesteuerten Lichtorgel bis zur elektronisch berechneten und designten Eigenbau-Lautsprecherbox, vom 64er als Mischpult bis zum Synthesi-Effekt zum Verfremden von Ansagen. Wer sich zum Profi-DJ mausern will, wird sich das passende schon herausuchen.

5.7 Fotohobby - Erfahrungsdaten sammeln

Fotografieren wird erst schön, wenn man alles selber macht. Wie soll man einem Urlaubs-Pocketkamera-Fotografen vermitteln, was für ein tolles Gefühl es ist, wenn man in der Dunkelkammer dabei ist, wenn im Entwickler langsam das Bild entsteht ? Oder wie viel Kreativität dazugehört, aus einer durchschnittlichen Aufnahme einen eindrucksvollen Abzug zu machen ? Ich weiß, ab hier sind wir unter uns.

Haben Sie auch schon mal vor der Frage gestanden, wie Sie einen wie 39 DIN belichteten Ilford HP5 richtig entwickeln müssen ? Ach, schon probiert ! Wieviele verschiedenen Varianten haben Sie rausgefunden ? Und haben Sie das alles schön notiert ?

Bei mir stecken diese Erfahrungsdaten in einer Diskette. Eine DATAMAT-Datei mit Namen "negadaten" enthält Angaben über Chemikalien, Temperaturen und Entwicklungszeiten von nicht standardmäßig bearbeiteten Filmen. In der Datei "posidaten" stehen die entsprechenden Daten zum Positivprozeß. Beide Dateien passen auf eine Diskette und können jederzeit auf den neuesten Stand gebracht werden.

```

=====
=== NEGA - DATEN ===
=====
(1) Filmtyp      : Ilford HP5
(2) DIN/ASA     : 36 DIN
(3) Kontrast    : HI (HI/LO)
(4) Entwickler  : Perceptol
(5) Zeit        : 390 sec
(6) Temperatur  : 27 Grad
(7) Hinweise   : Oft rollen !!
                  Temperatur genau
                  einhalten !!
=====
```

Der 64er kommt außerdem als Archivhelfer zum Zuge. 1 Film =
1 Datei; 1 Aufnahme = 1 Datensatz :

```
=====
=                               =
=          FOTO ARCHIV          =
=                               =
=====
=                               =
= Nummer -00129-                =
=                               =
= Datum (1) -17.08- Datum (2) -20.08- =
=                               =
= Filmtyp -TriX - -27DIN-      =
=                               =
= Aufnahmen:                    =
=                               =
= (01)-(10) -Bildschirm-Fotos für - =
=                               -DATA BECKER - =
=                               =
= (11)-(20) -Produkt-Fotos diverses - =
=                               -Computer,-Zubehör - =
=                               =
= (21)--38- -wie 11-20,        - =
=                               - =
=                               =
= Bemerkung: -Kunden-Auftrag 124/84 - =
=                               - =
=====
```

Jede Aufnahme kann jederzeit nachvollzogen werden. Auch eine Art Erfahrungsdaten-Liste ! Was altmodische Hobbyisten in dicken Notizbüchern und Aktenordnern festhalten, haben Sie auf drei bis vier handlichen Disketten. Wenn das kein Fortschritt ist ...

Ein Hinweis für Bastler. Die Möglichkeit den C64 direkt zum

Steuern von Vergrößerer bzw. Entwicklungstrommel einzusetzen sind, soweit ich es recherchiert habe, noch nicht sehr weit ausgelotet. Sie sollten die entsprechenden Publikationen (für Computer UND Foto) in dieser Hinsicht verfolgen.

5.8 Jogging

Die Nächte am Computer machen krank: das Rückgrat verbiegt sich, die Augen sind entzündet und der Kreislauf schlafft ab (oder dreht durch). Der Mensch ist halt ein Steppentier. Der Körper braucht Bewegung, besonders, wenn er stundenlang in sitzender Haltung vor sich hin dämmern mußte.

Das Zauberwort heißt Jogging. Gut, daß diese sportliche Betätigung den Ruch der Modeerscheinung inzwischen losgeworden ist, es sind schon andere gute Errungenschaften von ihrem Medien-Image gekillt worden.

Also, rein in den Sweat-Anzug, die Laufschuhe untergeschnallt und raus in den Park ! Am besten morgens vor, oder abends nach dem Berufsverkehrs-Mief. Das macht Spaß und hält gesund.

"Und was hat mein 64er damit zu tun ?" Nun, diesmal spielt die Kiste die Rolle des Antreibers. Mit dem Computer entwerfen Sie einen wunderschönen Trainingsplan, der neben dem Bett oder im Bad (über der Waage) hängt und Sie an Ihr gesundes Vorhaben erinnert. Gleichzeitig dient dieser Plan als Formular zum Erfassen Ihrer täglichen Laufleistung.

Wir machen uns für diese Sache die Fähigkeit des DATAMAT zunutze, Listen ausdrucken zu können. Was soll ich noch viele Worte machen, sehen Sie selbst:

J O G G I N G - P L A N E R

Montag

07.30 Uhr Ort Rheinpark Jogging
19.00 Uhr Ort † Turnhalle TSV

Dienstag

07.30 Uhr Ort Rheinpark Jogging
. Uhr Ort †

Mittwoch

07.30 Uhr Ort Rheinpark Jogging
21.00 Uhr Ort † Tennis-Center

Donnerstag

07.30 Uhr Ort Rheinpark Jogging
21.30 Uhr Ort † Tennis-Center

Freitag

. Uhr Ort
20.00 Uhr Ort † Knubbel-Eck

Samstag

10.00 Uhr Ort Ost-Park Mit den Kindern
17.00 Uhr Ort †

Sonntag

13.00 Uhr Ort Radfahren
. Uhr Ort †

5.9 FdH - Diätplan elektronisch

```
*****
*      MAN NEHME      :      *
*****

*****
*      HARDWARE      :      *
*****
*
*      COMMODORE 64      *
*      Datasette oder Floppy      *
*      Monochromer Monitor      *
*
*****

*****
*      SOFTWARE      :      *
*****
*
*      keine      *
*
*****

*****
*      VORKENNTNISSE      *
*****
*
*      Commodore-BASIC-Kenntnisse      *
*
*****
```

Ach, was gibt es alles für Methoden zur Erhalt
schlanken Linie: Brigitte-, Atkins-, Kartoffel-, Ei-

Reisdiät, Obsttag und Entwässerungskur, 1000-Kalorien- und Nulldiät; alles sei bewährt und garantiert nicht gesundheitsschädlich, sagen die Verfechter jeder Methode.

Alle sind sich jedoch einig: der Verlust von Bierbauch und Speckrücken läßt sich nur über eine Kombination von richtiger Ernährung und ausreichender körperlicher Bewegung erreichen. Zum zweiten Teil haben wir unseren 64er ja bereits gebraucht. Kommen wir nun zum Ernährungs-Programm.

Vor mir liegt eine schmale, schmucklose Broschüre mit dem niedlichen Titel "Kleine Nährwert Tabelle" (Deutsche Gesellschaft für Ernährung e.V.; für DM 5,80 im Buchhandel erhältlich). Das Bändchen enthält eine vollständige Darstellung der Nährstoff-Zusammensetzung (fast) aller denkbaren Speisen, also nicht nur Kalorien, oder KiloJoule, wie es heute heißt, sondern auch die jeweiligen Anteile an Vitaminen, Mineralstoffen, Eiweiß, Fett und Kohlehydraten.

Diese tolle Tabelle sollte Grundlage jeder Überlegung zu einem FdH-Programm sein. Ein wenig Theorie vorweg.

Es geistern oft obskure Zahlen über die notwendige Kalorienzufuhr pro Tag umher; mal sollen es 2000 sein, mal nur 1400 und bisweilen sind 3000 Kilokalorien pro Tag richtig. Alles Quatsch ! Der tägliche Bedarf muß individuell berechnet werden. In unserem Programm wird dies automatisch erledigt. Nur ein Anhaltspunkt: ein durchschnittlicher Büroangestellter sollte nicht mehr als ca. 2600 kcal zu sich nehmen, vorausgesetzt, er hat Normalgewicht. In Gläsern Bier ausgedrückt sind das nicht mehr als 20 Stück (0,2 l) ! Oder 4 Tüten Kartoffel-Chips, bzw. 300 gr Walnüsse. Sie können natürlich Ihre Energie auch in Form von 25 Bechern Frucht-Yoghurt tanken, wenn Sie mögen.

Das folgende Programm ist sozusagen nur die erste Stufe der Diät-Rakete. Es berechnet aus den Eingaben das Normalgewicht und, falls Übergewicht diagnostiziert wurde, die Kalorienzahl, die das Abnehmen bewirken könnte.

Das ganze habe ich in eine - hoffentlich - spaßige Form gebracht; bei derart heiklen Sachen muß der Computer eben zeigen, daß er auch nur ein Mensch ist.

Die zweite Stufe wäre ein Programm, das die gewonnenen Diät-Daten weiter verarbeiten kann. Also z.B. aus einer Liste von Lieblingsspeisen und -getränken die heraussucht, die den Diätplan einhalten.

Ich habe an dieser Stelle ein konkretes Listing dafür weggelassen, weil wir uns ja nicht schon wieder mit den Problemen verschiedener Dateiformen herumschlagen wollen. Davon nimmt man nämlich überhaupt nicht ab.

Übrigens: das Programm "FdH" eignet sich ganz hervorragend als Party-Spaß. Sie wissen ja, für die Leute, die immer fragen "Was machst Du eigentlich mit dem Computer?". Die mögen so etwas.

```

100 rem *****
110 rem * *
120 rem *     eff de ha *
130 rem * *
140 rem * ein spass mit ernstem *
150 rem *     hintergrund *
160 rem *     aus dem data becker *
170 rem *     ideenbuch (c) 1984 *
180 rem * *
190 rem *****
200 rem
210 poke 53280,7
220 poke 53281,7
230 poke 53272,23
240 print chr$(149)
250 print chr$(147); : print "";
259 rem *****
260 rem *** programm beginn ***
261 rem *****
300 print "HALLO" : print
310 print "Mein Name ist EFF DE HA."
320 print "Nennen Sie mich Effi !" : print
330 print "Waren Sie schon bei mir in"
340 input "Behandlung (j/n) ";in$
350 if in$="j" or in$="J" then 11000
360 goto 10000
999 rem *****
1000 rem *** unterprogramm tastenabfrage ***
1001 rem *****
1010 poke 214,23 : sys 58640
1020 print "Bitte irgendeine >Taste< druecken "
1030 get t$: if t$="" then 1030
1040 print chr$(147); : print "";
1050 return
1099 rem *****
1100 rem *** unterprogramm farbe ***
1101 rem *****
1110 poke 53280,12
1120 poke 53281,15
1130 print chr$(151)
1140 input "Besser (j/n) ";in$
1150 if in$="j" then return
1160 print : print "Immer nur meckern ! Na gut ..."
1170 poke 53280,7
1180 poke 53281,7
1190 print chr$(149)
1200 return
1299 rem *****
1300 rem *** unterprogramm namensabfrage ***
1301 rem *****
1310 input "Z.B. den Vornamen ";in$
1320 na$=in$ : an$(1)="Du" : an$(2)="Dein" : an$(3)="Dir" : an$(4)="r"
1330 print : input "Maennlein oder Weiblein (m/w) ";in$
1340 if in$="w" then an$(4)="r"
1350 return

```

```

1399 rem *****
1400 rem *** unterprogramm dutzen ***
1401 rem *****
1410 rem
1420 input "Nett. Sollen wir uns dutzen (j/n) ";in$
1430 if in$="j" then return
1440 print : print "Schade. Dann brauche ich Ihren"
1450 input "Nachnamen ";in$
1460 an$(1)="Sie" : an$(2)="Ihr" : an$(3)="Ihnen"
1470 na$=na$+" "+in$
1480 return
1499 rem *****
1500 rem *** unterprogramm alter ***
1501 rem *****
1510 print an$(2); : input " Alter in Jahren ";in$
1520 in=val(in$) : if in>10 and in<100 then 1550
1530 if in<=10 then print "Komm' doch in 2 Jahren noch mal wieder !"
1540 if in>=100 then print "Wer so alt ist, darf auch dick sein !"
1545 gosub 1000 : goto 1510
1550 al%=in : return
1599 rem *****
1600 rem *** unterprogramm gewicht ***
1601 rem *****
1610 print an$(2); : input " Gewicht in Kilo ";in$
1620 in=val(in$) : if in>20 and in<200 then 1650
1630 if in<=20 then print "Bitte nicht mogeln !"
1640 if in>=200 then print "Das glaube ich ";an$(3);" einfach nicht."
1645 gosub 1000 : goto 1710
1650 gw%=in : return
1699 rem *****
1700 rem *** unterprogramm groesse ***
1701 rem *****
1710 print "Und wie ist ";an$(2)+"e"; : input " Laenge ";in$
1720 in=val(in$) : if in>20 and in<240 then 1770
1730 if in<=20 then print "Auch Zwerge haben klein angefangen."
1740 if in<=20 then print "Aber sooo klein ? Nicht schummeln !"
1750 if in>=240 then print "Auf solche Riesen bin ich leider nicht"
1760 if in>=240 then print "vorbereitet."
1765 gosub 1000 : goto 1810
1770 gr%=in : return
1799 rem *****
1800 rem *** unterprogramm uebergewicht berechnen ***
1801 rem *****
1805 if an$(4)="r" then bo%=0 : if an$(4)=" " then bo%=10
1810 ng%=(gr%-(100-bo%))+int(al%/20) : ig%=ng%-(ng%/20) : ug%=gw%-ng%
1820 if ug%<=0 then 1870
1830 if ug%>0 then ug$=" Ueber"
1840 if ug%>0 and ug%<3 then mu$="Das kriegen wir schnell runter."
1850 if ug%>3 and ug%<10 then mu$="Dies Gewicht geht ans Herz. Abnehmen !"
1860 if ug%>=10 then mu$="Das ist lebensgefuehrlich."
1870 if ug%<=0 then ug$=" Unter"
1880 if ug%<=0 then mu$="Dies ist kein Mastprogramm." : ug%=ug%*(-1)
1890 return

```

```

1999 rem *****
2000 rem *** koerperliche belastung ***
2001 rem *****
2010 print : print " 1 = Stubenhocker"
2020 print " 2 = Handwerker oder Hausfrau"
2030 print " 3 = Echter Malocher"
2040 print " 4 = Holzfaeller oder Steineklopfen"
2050 print " 5 = Fussball-Profi"
2060 print : input "Selbsteinschaetzung ";in$: in=val(in$)
2070 if in<0 or in>5 then print "" : goto 2060
2080 if in=5 then print "Profi und Uebergewicht ? "
2090 if in=5 then print "Wenn das der Trainer erfahrt ..." : in=3
2095 kb%=in : return
2099 rem *****
2100 rem *** unterprogramm sport ***
2101 rem *****
2110 print : print " 1 = Bloss kein Sport !!!"
2120 print " 2 = Hoechstens Kegeln und Bierstemmen";
2130 print " 3 = Radfahren, Schwimmen aber wenig"
2140 print " 4 = Klar ! Mindestens 3 Stunden/Woche";
2150 print " 5 = Schwerathlet"
2160 print : input "Na ??????";in$
2170 in=val(in$) : if in<1 or in>5 then print "" : goto 2160
2180 sp%=in : return
2199 rem *****
2200 rem *** unterprogramm kalorien berechnen ***
2201 rem *****
2210 kb%=(kb%-1)*400 : sp%=(sp%-2)*150 :en%=2400+kb%+sp%
2220 if a1%>15 then en%=en%+200
2230 if a1%>15 and a1%<19 then en%=en%+200
2240 di%=100*ug% : di%=en%-di%
2250 if di%<1000 then di%=1000
2260 if en%-di%<200 then km#="Auf zum Idealgewicht."
2270 if di%<1200 then km#="Was sein muss, muss sein." : return
2280 if di%>1200 and di%<1500 then km#="Nur Mut !" : return
2290 if di%>1500 and di%<1800 then km#="Kein Problem. Oder ?" : return
2295 if di%>1800 then km#="Leben wie Gott in Frankreich" : return
2299 rem *****
3000 rem *** unterprogramm labelle ***
3001 rem *****
3010 pd$(1)=str$(a1%) : pd$(2)=str$(qw%) : pd$(3)=str$(qr%)
3020 pd$(4)=str$(uq%) : pd$(5)=ug#+ "gewicht" : pd$(6)=str$(kb%)
3030 pd$(7)=str$(sp%) : pd$(8)=str$(en%) : pd$(9)=str$(di%) : pd$(10)=str$(ng%)
3040 return
3099 rem *****
3100 rem *** unterprogramm tabelle zeigen ***
3101 rem *****
3110 gosub 1000
3120 print : print "Eff+De-Ha tabelle fuer ";na$ : print
3130 print ">>>"+km#+"<<<": print
3140 for n=1 to 10
3150 : read n$

```

```

3160 : print n#;pd$(n)
3170 next
3180 restore : return
3190 data "Alter          = ","Gewicht      = ","Groesse       = "
3191 data "= ","","Zuschlag 1  = ","Zuschlag 2  = "
3192 data "Bedarf        = ","Diaet        = ","Normalgewicht = "
3199 rem *****
3200 rem *** unterprogramm tabelle speichern ***
3201 rem *****
3210 ff$=" "
3220 if len(na$)>15 then dn#=left$(na$,15)
3230 if len(na$)<=15 then dn#=na#+left$(ff$,15-len(na$))
3240 print ">"+"ff$+"<"
3250 print "+dn$ : print "Unter diesem Namen wird die Tabelle"
3260 print "gespeichert. Bitte ein wenig Geduld !"
3270 open 1,8,2,u#+dn$+"s,w"
3280 for n=1 to 10 : print#1,pd$(n) : next
3290 close 1 : return
3299 rem *****
3300 rem *** unterprogramm tabelle einlesen ***
3301 rem *****
3310 ff$=" "
3320 print "Wie war noch gleich der Name ?"
3330 input na$
3340 print : print "Jetzt muss ich aber erstmal die Tabelle"
3350 print "in meinen Speicher holen."
3360 if len(na$)>15 then dn#=left$(na$,15)
3370 if len(na$)<=15 then dn#=na#+left$(ff$,15-len(na$))
3380 open 1,8,2,dn$+"s,r"
3390 for n=1 to 10
3400 : input#1,pd$(n)
3410 next n
3420 al%=val(pd$(1)) : gw%=val(pd$(2)) : gr%=val(pd$(3))
3430 kb%=val(pd$(6)) : sp%=val(pd$(7)) : en%=0 : di%=0
3440 close 1 : gosub 3500
3450 if ff=0 then return
3460 if ff=62 then 3320
3499 rem *****
3500 rem *** fehlerkanal ***
3501 rem *****
3510 open15,8,15
3520 input#15,ff,fb$,sp,se
3530 close15
3540 if ff<>0 then print ff;fb$;sp;se : gosub 1000 : return
3560 if ff=0 then return
9999 rem *****
10000 rem *** hauptprogramm (1) ***
10001 rem *****
10010 rem
10020 u$=""
10030 print : print "Sie wollen also abnehmen. Gut. Da will"
10040 print "ich Ihnen gerne helfen. Ich brauche "

```

```

10050 print "allerdings ein paar Daten."
10060 print "(Typische Computer, nicht ?)"
10070 gosub 1000 : gosub 1300 : gosub 1000 : gosub 1400 : gosub 1000
10080 print "Uebrigens, ist ";an$(3);" das Bild zu" : input "bunt (j/n) ";in$
10090 if in$="j" then gosub 1100
10100 gosub 1000 : print "Zur Sache. Ich brauche " : gosub 1500
10110 gosub 1000 : print "Und nun das Uebliche..." : gosub 1600
10120 gosub 1000 : gosub 1700 : gosub 1800 : gosub 1000
10130 if an$(1)="Du" then ha$="hast"
10140 if an$(1)="Sie" then ha$="haben"
10150 print "Also, ";an$(1);" ";ha$;" ";ug%;" kg";ug$+"gewicht"
10160 print : print an$(2);" Normalgewicht waeren ";ng%;" kilo"
10170 print "Beziehungsweise ";ig%;" kilo Idealgewicht."
10180 print : print "Liebe"+an$(4)+" "+na$;," ich meine : "
10190 print au$
10200 if mu$="Dies ist kein Mastprogramm." then 10490
10210 gosub 1000 : print "Jeder Mensch lebt anders. Und ist unter-";
10220 print "schieidlich belastet."
10230 print "Wie ist es bei ";an$(3);" ?" : gosub 2000
10240 gosub 1000 : print "Interessant. Ich weiss, Sport ist Nord"
10250 print "aber gesund."
10260 print "Auch hier brauche ich ";an$(2)+"e";" Antwort."
10270 gosub 2100 : gosub 2200
10280 gosub 1000 : print "Nochmal : "
10290 print : print ug$+"gewicht = ";ug%;" kg"
10300 print " Normalgewicht = ";ng%;" kg"
10310 print " Idealgewicht = ";ig%;" kg"
10320 print : print "Wenn ";an$(2);" normaler taeglicher"
10330 print "Energiebedarf etwa ";en%;" kilokalorien"
10340 print "bedaeft, werde ich in anbetracht der "
10350 print "Umstaende empfohlen, nicht mehr als"
10360 print d%;" Kal/Tag aufzunehmen."
10370 print "Mein Kommentar:"
10380 print : print ">>>4m$+<<<"
10390 gosub 1000 : gosub 2000
10400 input "Soll ich nochmal die Tabelle zeigen ";in$
10410 if in$="n" then 10430
10420 gosub 3100
10430 print : input "Soll ich die Tabelle drucken ";in$
10440 if in$="n" then 10450
10450 open4,4,2 : cmd 4 : gosub 3120 : print#4 : close 4
10460 input "Und/oder auf Diskette speichern ";in$
10470 if in$="n" then 10490
10480 gosub 3200
10490 gosub 1000 : print "Na, das war's."
10500 print "Beehren Sie mich bald wieder !!!!!!!!!!"
10510 rem
10520 gosub 1000 : print "***E N D E***" : end
10999 rem *****
11000 rem *** hauptprogramm (2) ***
11001 rem *****

```

```

11010 gosub 1000 : gosub 3300 : gosub 3100
11020 if left$(dn$,4)<>"Herr" and left$(dn$,4)<>"Frau" then gosub 11700
11030 if left$(dn$,4)="Herr" or left$(dn$,4)="Frau" then gosub 12000
11040 print an$(2)+"e";" Daten."
11050 gosub 1000 : input "Hat sich was geaendert (j/n) ";in$
11060 if in$="n" then 10430
11070 print : print " 1 = Alter"
11080 print " 2 = Gewicht"
11090 print " 3 = Groesse"
11100 print " 4 = Belastung"
11110 print " 5 = Sport"
11190 print : input "Was denn ";in$ : in=val(in$)
11200 if in>0 and in<6 then 11220
11210 print "Daneben getippt. Bitte nochmal." : gosub 1000 : goto 11070
11220 on in goto 11230,11270,11340,11380,11430
11230 al%=0 : gosub 1000 : print "Herzlichen Glueckwunsch zum Geburtstag !";
11240 gosub 1500 : gosub 1000 : input "Hat sich noch was geaendert (j/n) ";in$
11250 if in$="n" then 11480
11260 print chr$(147); : print ""; : goto 11070 : gosub 1000
11270 qw%=0 : input "Zugenommen (j/n) ";in$
11280 if in$="j" then print : print "Diaet nicht beachtet, was ?"
11290 if in$="n" then print : print "Na prima, es geht doch !!!"
11300 print "Das neue Gewicht will ich aber wissen" : gosub 1600
11310 gosub 1000 : input "Und sonst ? Weitere Neuigkeiten ";in$
11320 if in$="n" then 11480
11330 print chr$(147); : print ""; : goto 11070
11340 gr%=0 : print "Wohl im Maien-Regen gestanden." : gosub 1700 : gosub 1000
11350 print "Ist denn noch etwas an ";an$(3); : input "anders geworden ";in$
11360 if in$="n" then 11480
11370 print chr$(147); : print ""; : goto 11070
11380 kb%=0 : print "Na ja, ein neuer Job koennte mir auch"
11390 print "gefallen. Was ist es denn ?" : gosub 2000
11400 gosub 1000 : input "Aber sonst ist alles beim Alten ";in$
11410 if in$="j" then 11480
11420 print chr$(147); : print ""; : goto 11070
11430 sp%=0 : print "Na wollen mal sehen, ob's besser"
11440 print "geworden ist." : gosub 2100
11450 gosub 1000 : input "Noch was neues (j/n) ";in$
11460 if in$="n" then 11480
11470 print chr$(147); : print ""; : goto 11070
11480 gosub 1000 : print "Wollen mal sehen, was diesmal rauskommt."
11490 gosub 1800 : gosub 2200 : u$="S;" : goto 10420 : end
11900 an$(1)="Du" : an$(2)="Dein" : an$(3)="Dir"
11910 return
12000 an$(1)="Sie" : an$(2)="hr" : an$(3)="hnen"
12010 return

```

ready.

5.10 Ein intelligentes Wörterbuch

```
*****
*      MAN  NEHME      :      *
*****

*****
*      HARDWARE      :      *
*****
*
*      COMMODORE 64      *
*      Floppy 1541      *
*      Monochromer Monitor      *
*
*****

*****
*      SOFTWARE      :      *
*****
*
*      keine      *
*
*****

*****
*      VORKENNTNISSE      *
*****
*
*      Gute BASIC-Kenntnisse      *
*      Erfahrung im Umgang mit      *
*      relativen Dateien      *
*
*****
```

Von zwei Dingen will ich hier einfach ausgehen: erstens, die Kids (=Computerfans unter 18 Jahren) haben eh mehr Ahnung vom Programmieren als die Alten und zweitens, Spiele sind out, Education ist in.

Kann sein, daß ich falsch liege, aber ich will trotzdem der computerisierten Nachhilfe das Wort reden. Beim folgenden Programm dreht es sich um ein selbstlernendes Wörterbuch. Was ist das ?

Ein Beispiel aus dem Bereich Deutsch - Englisch: Sie geben das schöne deutsche Wort "Klobürste" ein und wollen vom Computer wissen, was das auf Englisch heißt. Der antwortet "Kenn ich nicht. Was heißt das denn ?" Nach der Eingabe "toilet brush" nimmt das Programm den neuen Begriff automatisch in das Verzeichnis auf.

Was das Programm noch leisten soll:

- Vokabeln abfragen
- Vokabeln speichern

Das Beispiel ist etwas außergewöhnlich gewählt. Wohl wenige Schüler werden ein Fremdwörter-Lexikon brauchen. Zur Entschuldigung will ich nur anführen, daß das Beispiel den Vorteil hat, kleinere Wortmengen verwalten zu müssen. Und so ganz abseitig ist diese Anwendung auch nicht; man denke nur an die ähnliche Problematik eines Fachwörterbuches. Im übrigen kann das vorliegende Listing ohne große Schwierigkeiten für alle ähnlichen Fälle umgestrickt werden.

Der größte, leider nicht vermeidbare Nachteil ist das Mengengerüst: rechnet man für einen Datensatz bei Vokabeln je 20 Zeichen für das deutsche wie das englische Wort, so passen nicht mehr als ca. 3000 Begriffe in eine Datei (sprich: auf eine Diskette). Das ist ziemlich wenig. Ein englischer Grundwortschatz beträgt nach Expertenmeinung zwischen 5.000 und 12.000

Wörtern. Wird das Programm weniger als Übersetzungshilfe denn als Lernmittel gebraucht, ist es nicht weiter schlimm; es werden einfach mehrere Dateien angelegt: eine für Verben, eine für Substantive, für Idioms usw.

Wer noch kein Programm benutzt hat, daß eine relative Datei verwendet, sollte beim Abändern des vorliegenden unbedingt ein gutes Buch zu Rate ziehen, z.B. das "Große Floppy-Buch" (von DATA BECKER) oder etwas ähnliches.

So und nun das Listing mit dem nötigen Zubehör.

```

1 rem * * * * *
2 rem      fremdwoerter - lexikon
3 rem *
4 rem      aus dem data becker ideenbuch
5 rem *
6 rem      version 3.      1.6.84
7 rem * * * * *
8 rem
9 rem
10 poke 53272,23      : rem gross/klein
10 poke 53280,12      : rem rahmenfarbe mittelgrau
30 poke 53281,12      : rem hintergrundfarbe hellgrau
40 print chr$(151)    : rem schriftfarbe dunkelgrau
50 rem
60 ff$=""
70 fr$=ff$+"          "
80 cr$=chr$(13)       : rem >>return<<
90 dim ix$(1500)      : rem indextabelle
99 goto 50000
10000 rem * * * ueberschrift * * *
10001 rem
10010 print chr$(147);
10020 print "***-----***";
10030 print "***          ***";
10040 print "***      FREMDWOERTER LEXIKON      ***";
10050 print "***          ***";
10060 print "***-----***";
10070 print : print
10080 return
11000 rem * * * eingabe/kommando zeile * * *
11001 rem
11010 cz=23 : cs=0
11020 poke 214,cz : poke 211,cs : sys 58640
11030 return
12000 rem * * * tasten abfragen * * *
12001 rem
12010 gosub 11000
12020 print "          >Taste druecken<          ";
12030 get g$ : if g$="" then 12030
12040 print "" : print ff$;
12050 return
20000 rem * * * fehlerkanal auslesen * * *
20001 rem
20010 open 15,8,15
20020 input#15,ff,fb$,sp,se
20030 close 15
20040 fm$=str$(ff)+" "+fb$+" "+str$(sp)+" "+str$(se)
20050 return
21000 rem * * * diskette formatieren * * *
21001 rem
21010 gosub 11000 : print ff$; : gosub 11000
21020 print "          Diskette wird formatiert          ";
21030 open 15,8,15
21040 print#15,"n:lexikon,11"
21050 close 15

```

```

21060 gosub 20000
21070 if ff=0 then 22000
21080 gosub 11000
21090 print "      Fehlermeldung : ";fm#;
21100 gosub 12000
21110 gosub 11000
21120 print "Neue Diskette nehmen.Vorgang wiederholen";
21130 gosub 12000
21140 goto 21000
22000 rem * * * relative datei einrichten * * *
22001 rem
22010 gosub 10000
22020 poke 214,9 : poke 211,0 : sys 58640
22030 input "      Anzahl der Fremdwoerter ? 1500";in#
22040 in=val(in#) : if in>1500 then 22020
22050 print : print "      Die Datei heisst >lexikon<      "
22060 print "      Ein Datensatz hat max. >85< Zeichen"
22070 gosub 11000
22080 print "      Datei wird eingerichtet      ";
22090 rn%=in : hb=int(rn%/256) : lb=rn%-hb*256
22100 open 1,8,2,"lexikon,1,"+chr$(86)
22110 open 2,8,15,"p"+chr$(2)+chr$(1b)+chr$(hb)+chr$(1)
22120 print#1,chr$(255)
22130 close 2 : close 1
22140 gosub 20000
22150 if ff=0 or ff=50 then return
22160 gosub 11000
22170 print "      Fehlermeldung : ";fm#;
22180 gosub 12000
22190 gosub 11000
22200 print "Dateien loeschen. Vorgang wiederholen ";
22210 gosub 12000
22220 open 14,8,15
22230 print#14,"s:lexikon"
22240 print#14,"v"
22250 close 14
22260 gosub 20000
22270 if ff<>1 then 21000
22280 return
23000 rem * * * index-datei laden * * *
23001 rem
23010 gosub 10000
23020 gosub 11000
23030 print "      Index-Datei wird geladen      ";
23040 open 3,8,2,"index,s,r"
23050 x=1
23060 input#3,ix$(x)
23070 if st<>64 then x=x+1 : goto 23060
23080 close 3
23090 gosub 20000
23100 if ff<>0 then 23120
23110 return
23120 gosub 11000
23130 print "      Fehlermeldung : ";fm#
23140 gosub 12000

```

```

23150 gosub 11000
23160 print "Bedeutung nachschlagen. Neu beginnen."
23170 gosub 12000
23180 print chr$(147)
23190 end
24000 rem * * * index-datei speichern * * *
24001 rem
24010 gosub 10000
24020 gosub 11000
24030 print "      Index-Datei wird gespeichert      "
24040 open 3,8,2,"9:index,s,w"
24050 for x=1 to rn%
24060 :   print#3,ix$(x)
24070 next x
24080 close 3
24090 gosub 20000
24100 if ff<>0 then 24120
24110 return
24120 gosub 11000
24130 print "      Fehlermeldung : ";fm$
24140 gosub 12000
24150 gosub 11000
24160 print "Bedeutung nachschlagen. Neu beginnen."
24170 gosub 12000
24180 print chr$(147)
24190 end
25000 rem * * * datei-parameter ausgeben * * *
25001 rem
25010 gosub 23000
25020 gosub 11000
25030 print ff$
25040 gosub 10000
25060 print "          *** DATEI PARAMETER ***          "
25070 print : print
25080 print "Die Datei >lexikon< enthaelt ";x;" Saetze."
25090 gosub 12000
25100 return
30000 rem * * * disketten/datei menue * * *
30001 rem
30010 gosub 10000
30020 print "          *** VORBEREITUNGEN ***          "
30030 print
30040 input "      (1) Datendiskette eingelegt  J";in$
30050 if in$<>"n" and in$<>"N" then 30080
30060 print "          > Datendiskette einlegen <"
30070 gosub 12000 : poke 214,10 : poke 211,0 : sys 58640
30080 input "      (2) Diskette formatiert      J";in$
30090 if in$<>"n" and in$<>"N" then 30125
30100 gosub 21000
30110 return
30120 gosub 12000
30125 poke 214,13 : poke 211,0 : sys 58640
30130 input "      (3) Datei lexikon existiert  J";in$
30140 if in$<>"n" and in$<>"N" then 30160
30150 gosub 22000

```

```

30160 gosub 23000
30170 return
40000 rem * * * lexikon modus * * *
40001 rem
40005 gosub 25000
40010 open 1,8,2,"lexikon,1,"+chr$(86)
40015 open 2,8,15
40020 gosub 10000
40030 print "          *** LEXIKON MODUS ***"
40040 gosub 11000
40050 print "          Fremdwort eingeben          "
40060 poke 214,10 : poke 211,0 : sys 58640
40070 input fw$
40080 fw=len(fw$) : if fw>20 then 40060
40090 if fw=20 then 40110
40100 fw$=fw$+left$(fr$,20-fw)
40110 for i=1 to x
40120 :   if fw$=left$(ix$(i),20) then 40160
40130 next i
40140 fl=1 : rn%=x+1 : goto 45110
40150 fl=0 : goto 40280
40160 ix=len(ix$(i))
40170 i$=right$(ix$(i),ix-20)
40180 rn%=val(i$)
40190 hb=int(rn%/256) : lb=rn%-hb*256
40200 print#2,"p"+chr$(2)+chr$(lb)+chr$(hb)+chr$(22)
40210 input#1,ue$
40220 for j=1 to 63
40230 :   j$=mid$(ue$,1,j)
40240 :   if j$=" " then ue$=left$(ue$,j) : goto 40260
40250 next j
40260 poke 214,12 : poke 211,0 : sys 58640
40270 print ue$
40280 gosub 11000 : print ff$; : gosub 11000
40290 input "          Weitere Fragen          J";in$
40300 if in$="n" or in$="N" then 40320
40310 goto 40050
40320 close 2 : close 1
40330 return
45000 rem * * * lexikon erweitern * * *
45001 rem
45005 gosub 25000
45010 rn%=x+1 : fl=0
45015 open 1,8,2,"lexikon,1,"+chr$(86)
45020 open 2,8,15
45025 gosub 10000
45030 print "          *** LEXIKON ERWEITERN *** Nr.:";rn$
45040 gosub 11000
45050 print "          Fremdwort eingeben          "
45060 poke 214,10 : poke 211,0 : sys 58640
45070 input fw$
45080 fw=len(fw$) : if fw>20 then 45060
45090 if fw=20 then 45110
45100 fw$=fw$+left$(fr$,20-fw)
45110 gosub 11000

```

```

45120 print "          Uebersetzung eingeben          "
45130 poke 214,12 : poke 211,0 : sys 58640
45140 input ue$
45150 ue=len(ue$) : if ue>64 then 45130
45160 if ue=64 then 45180
45170 ue$=ue$+left$(fr$,64-ue)
45180 ix$(rn%)=fw$+str$(rn%)
45190 le$=fw$+cr$+ue$
45200 hb=int(rn%/256) : lb=rn%-hb*256
45210 print#2,"p"+chr$(2)+chr$(lb)+chr$(hb)+chr$(1)
45220 print#1,le$
45230 if fl=1 then 40150
45240 gosub 11000 : print ff$ : gosub 11000
45250 input "          Weitere Eintraege  J";in$
45260 if in$="N" or in$="n" then 45300
45270 poke 214,7 : poke 211,35 : sys 58640
45280 rn%=rn%+1 : print rn%
45290 goto 45040
45300 close 2 : close 1
45310 return
49000 rem * * * programm beenden * * *
49001 rem
49010 gosub 10000
49020 gosub 24000
49030 gosub 11000
49040 input "          Programm beenden          N";in$
49050 if in$="j" or in$="J" then 49070
49060 goto 50000
49070 print chr$(147)
49080 poke 214,10 : poke 211,12 : sys 58640
49090 print "***  E N D E  ***"
49100 end
50000 rem * * * hauptprogramm * * *
50001 rem
50010 gosub 10000
50020 print "          *** MENUE ***"
50030 print
50040 print "          (1) Diskette/Datei vorbereiten          ";
50050 print "          (2) Lexikon Modus                          ";
50060 print "          (3) Lexikon erweitern                          ";
50070 print "          -----";
50080 print "          (0) Programm beenden                          "
50090 gosub 11000
50100 input "          >Bitte waehlen<          1";in$
50110 in=val(in$) : if in>3 then 50090
50120 if in=0 then 49000
50130 on in gosub 30000,40000,45000
50140 goto 50000
50150 rem
50160 rem * * * * *
50170 rem *          programm ende          *
50180 rem * * * * *

```

ready.

Das Hauptproblem ist die Wahl der Schlüssel- bzw. Suchbegriffe. Sie müssen einerseits kurz genug sein (wegen der Suchzeiten) und andererseits so aussagekräftig, daß man nicht in die Verlegenheit kommt, mangels Verständnis Dateien wegwerfen zu müssen. (Suchwort: "elschaltl8.99" z.B. ist recht kryptisch ...)

Nicht ganz leicht ist auch, die nötigen Informationen im vorhandenen Platz unterzubringen. Am besten geht es, wenn mit einer kleinen Datei experimentiert wird oder eine fiktive Versuchs-Datei angelegt wird.

Wer auf den althergebrachten Karteikasten nicht verzichten kann, druckt die einzelnen Datensätze auf ausreichend große Etiketten. Die werden dann auf normale Karten geklebt.

Noch ein Tip zum Schluß: die Dateien sollten

- a) regelmäßig aktualisiert und
- b) auf eine zweite Diskette kopiert werden.

Tja, dann wünsche ich Ihnen fröhliches Lernen und eine "EINS" in allen Fächern !

6. Musizieren, Malen und Entwerfen

"Wo man singt, da laß Dich ruhig nieder ..." Und was macht man da, wo gesynthesiziert wird ? Mitmachen. Der COMMODORE 64 kann nämlich ein tolles Musikinstrument sein; mit Betonung auf "kann". Wenn Sie nicht ein ganz gewiefter BASIC- oder gar Maschinensprache-Programmierer sind, dann versuchen Sie bloß nicht, anhand des C64-Handbuchs dem Gerät Töne zu entlocken. Ich hab es jedenfalls schnell aufgegeben.

Und dann fiel mir das Programm "SYNTHIMAT" in die Hände. Seitdem mache ich fast täglich Hausmusik. Auch meine eigenen Digital-Aufnahmen habe ich schon. Was, das glauben Sie nicht ? Blättern Sie mal zum Kapitel 6.2 !

Ohne Hilfsmittel mit dem 64er zu malen, ist alles andere als lustig, es ist frustrierend. Ich frage mich immer, warum haben die schlaunen Hardware-Ingenieure überhaupt so kreative Chips ins Gerät gesetzt, wenn die Software-Menschen nicht in der Lage oder willens waren, die implementierten Fähigkeiten von BASIC aus nutzen zu lassen ?

Andererseits denke ich, daß nur dieser Mangel dazu geführt hat, daß sich tausende und abertausende pfiffige Freaks sich Gedanken gemacht haben, wie man leichter ans Eingemachte des C64 kommen kann. Einer dieser hellen Köpfe gehört Axel Plenge; er schrieb das Programm "SUPERGRAPHIK 64" für DATA BECKER. Das ist eine sogenannte Befehlserweiterung, denn mit SUPERGRAPHIK im Speicher schreiben Sie ganz normale BASIC-Programme, nur mit dem gewaltigen Vorteil, hochauflösende Grafik einbauen zu können. Wer BASIC-Programme schreiben kann, kann auch mit SUPERGRAPHIK arbeiten. Und wer selber Video-Spiele entwickeln will, kommt ohne dies Programm kaum zum Ziel.

Mein Lieblings-Spielzeug ist momentan das "KOALA PAD". Schon so manche (halbe) Nacht habe ich mit diesem Grafik-Tablett

zugebracht. Malen, Zeichnen, Speichern und wieder Hervorholen, alles das kann das KOALA PAD. Einen Fehler hat das Ding, die tollen Bilder konnte man nicht drucken. Leider. Aber: vor kurzem hat Axel Plenge seine "SUPERGRAPHIK" überarbeitet und mit einigen neuen Eigenschaften versehen.

Neben solch prima Utilities (=Dienstprogramme) wie "directory", "merge", "renumber" usw. hat er ein Modul eingebaut, daß es tatsächlich zuwege bringt, Bilder, die mit dem KOALA PAD gemalt wurden, in SUPERGRAPHIK weiterzubearbeiten. Zunächst bin ich nur darauf gekommen, die Bilder über SG auszudrucken. Aber nach ein paar Tagen (besser: Nächten) ist mir soviel eingefallen, daß vier Ideen des folgenden Kapitels davon profitierten.

6.1 Kleine Notenschule

```

*****
*      MAN NEHME      :      *
*****

*****
*      HARDWARE      :      *
*****
*
*      COMMODORE 64   *
*      Floppy 1541   *
*      Beliebiger Fernseher (mit Ton) *
*
*****

*****
*      SOFTWARE      :      *
*****
*
*      SUPERGRAPHIK 64 *
*
*****

*****
*      VORKENNTNISSE *
*****
*
*      Commodore-BASIC-Kenntnisse *
*
*****

```

Im Handbuch zum 64er ist auf Seite 147 das nette kleine Programm "Piano" abgedruckt (nebenbei: auf der Demo-Diskette

zur Floppy ist es sogar zweimal vorhanden). Tippen Sie es ruhig ab, es ist der einzige Weg für einen Einsteiger, dem Computer problemlos Töne zu entlocken. So, und dann klimpern Sie ruhig drauf los, solange es Ihnen Freude macht.

Ich prophezeihe Ihnen, mehr als eine Stunde halten Sie es nicht aus. Spätestens dann wirds langweilig. Wie kann man trotzdem ein wenig Nutzen und Spaß aus diesem Angebot schlagen ? Ich würde vorschlagen, wir bauen das Programm zu einer kleinen Notenschule aus.

Prinzipiell kein Problem: auf dem Bildschirm wird ein Stück Notenblatt dargestellt, mit Eingaben wie "c" oder "gis" usw. werden die entsprechenden Notensymbole in die Linien geschrieben. Wenn die Eingabe beendet ist, kann der Song gespeichert werden

So einfach, dachte ich, könnte die "Kleine Notenschule" sein. Als es aber nicht so funktionierte, wie erhofft, habe ich eine Radikalkur angewandt. Bitte schön: ein völlig neues Musikprogramm namens "64 Musik Maschine" !

Weil ich mit der Befehlserweiterung SUPERGRAPHIK gearbeitet habe, kann ich einen Programmteil "Sound" anbieten. Damit ist das Ausprobieren der Tonvarianten ganz einfach. Das Programm erklärt sich weitestgehend selbst. Testen Sie es. Viel Spaß und - vielleicht lernen Sie tatsächlich die Notensprache damit.


```

40110 NEXT X
40120 FOR Z=1 TO 13
40130 : READ N# : NO$(Z)=N#
40140 NEXT Z
40150 FOR X=25 TO 265 STEP 40
40160 : Z=(X+15)/40
40170 : TEXTE NO$(Z),X,75,0
40180 NEXT X
40190 FOR X=50 TO 250 STEP 40
40200 : Z=((X-10)/40)+7
40210 : TEXTC NO$(Z),X,55,0
40220 NEXT X
40230 RETURN
41000 REM *** NOTEN SCHREIBEN ***
41001 REM
41010 N=NO(T)
41020 IF T=1 OR T=3 OR T=5 OR T=6 OR T=8 OR T=10 OR T=12 THEN M=1
41030 IF T=11 THEN M=3
41040 IF T=2 OR T=4 OR T=7 OR T=9 THEN M=2
41050 IF T<=5 THEN PF(1)=((T+1)*20)-5
41060 IF T>5 THEN PF(1)=((T+2)*20)-5
41070 TEXTE "↑",PF(2),90,0
41080 TEXTC "↑",PF(1),90,0
41090 Y=166-3*(N-2)
41100 TEXTC NT$(M),X,Y,0
41110 X=X+10 : IF X>300 THEN GOSUB 42000
41120 PF(2)=PF(1)
41130 RETURN
42000 REM *** NOTENBLATT LOESCHEN ***
42001 REM
42010 REM
42020 GMODE 0,1
42030 FILL 0,110 TO 319,170
42040 FOR Z=4 TO 0 STEP -1
42050 PLOT 0,160-(6*Z) TO 319,160-(6*Z)
42060 NEXT Z
42070 X=0
42080 RETURN
43000 REM *** SOUND/SONG SPEICHERN ***
43001 REM
43005 REM FL=10 ODER FL=ZAHL DER NOTEN EINES SONGS
43010 PRINT CHR$(147); : PRINT : PRINT : PRINT
43020 PRINT "   ███ SOUND/SONGS SPEICHERN ███" : PRINT : PRINT
43030 INPUT "WIE HEISST DER SOUND/SONG ";SN#
43040 IF LEN (SN#)>15 THEN SN#=LEFT$(SN#,15)
43050 PRINT : PRINT "DER SOUND/SONG MIT DEM NAMEN "; "█"+SN#+"█"
43060 PRINT "   WIRD GESPEICHERT !"
43070 OPEN 1,8,2,"S:"+SN#+"S,W"
43080 FOR Z=1 TO FL
43090 : PRINT#1,S(Z)
43100 NEXT Z
43110 CLOSE 1
43120 PRINT CHR$(147); : PRINT "████████████████████SPEICHERN BEENDET"
43130 FOR I=1 TO 1000 : NEXT
43140 RETURN
44000 REM *** SOUND/SONG LADEN ***
44001 REM

```


6.2 Hausgemachte Hits

```
*****
*      MAN NEHME      :      *
*****

*****
*      HARDWARE      :      *
*****
*
*      COMMODORE 64      *
*      Floppy 1541      *
*      Farb-Monitor      *
*
*****

*****
*      SOFTWARE      :      *
*****
*
*      SYNTHIMAT 64      *
*
*****

*****
*      VORKENNTNISSE      *
*****
*
*      Synthesizer-Grundkenntnisse      *
*
*****
```

Eigentlich stehe ich nicht sehr auf Synthi-Sounds. Mir ist ein anständiges Schlagzeug mit einer fetzigen Rockgitarre

allemaal lieber. Allerdings, was Können aus dem Synthesizer für Töne locken ...

Auch Songs verschiedener aktueller Hit-Gruppen sind ohne Synthesizer nur halb so schön. Ist ja auch egal, wichtig ist doch, daß man mit elektronischen Instrumenten Musik machen kann, ohne vorher das Konservatorium besucht zu haben. Und da stehe ich drauf ! Selbermachen ohne Fachidiot zu werden.

Der COMMODORE 64 ist, so wie er aus der Styropor-Verpackung kommt, ein veritables Musikinstrument. Ich will einmal die herausragenden Sundeigenschaften des C64 tabellarisch vorführen:

- 3 einzeln ansprechbare Oszillatoren
- 4 Wellenformen pro Stimme
- alle Wellenformen untereinander mischbar
- Filter aller Arten
- Hüllkurvengenerator
- 2 Ringmodulatoren
- A/D-Wandler

Wer sich ein bißchen auf diesem Gebiet auskennt, wird einsehen, daß das Möglichkeiten bietet, die vor 10, 12 Jahren der legendäre MOOG-Synthesizer nicht in dieser Auswahl hatte. Und das war immerhin ein Gerät von annähernd Schlafzimmerschrank-Größe mit zig Kabeln und Steckern. Der MOOG war im eigentlichen Sinne nicht einmal programmierbar, denn Veränderungen der Register konnte nur über das Umstecken von Strippen erreicht werden. Mit anderen Worten: was 1970 Emerson, Lake & Palmer produzierten, ist heute im 64er (zumindest versteckt) vorhanden.

Aber eben versteckt. Es geht leider nicht so, wie es das folgende Programm vorschlägt:

```
10 rem synthesizer
20 sound$(1)="oboe"
```

```

30 sound$(2)="percussion"
40 rhythm$(1)="walzer"
50 tune sound$(1),sound$(2),rhythm$(1)
60 play rhythm$(1),sound$(2)
70 melody sound$(1)
80 input "tempo ";in$
90 tempo$=in$
100 get key
110 goto 100

```

Nach "run" ginge der Walzer dann los; das Schlagzeug spielt denn Rhythmus, die Oboe die Melodie. Das Ganze wird automatisch digital auf Diskette gespeichert.

Na ja, vielleicht kommt sowas ja noch. Bis dahin begnügen wir uns mit dem SYNTHIMAT 64 (von DATA BECKER). Was heißt hier "begnügen" ? Dieses Programm ist doch wirklich phantastisch genug !

Sein Erfinder, der 20jährige Student Thomas Dachsel, schreibt im Bedienungshandbuch: "Es ist schon sehr kompliziert, ein gutes Register zu programmieren, und selbst 256 Register würden nie ausreichen, alle Möglichkeiten dieses Programms und des Sound-Chips auszuschöpfen." Dem ist nichts hinzu zu fügen.

Diese Äußerung zeigt aber auch, wie groß die Variationsbreite Ihres Musik-Computers ist. Ich will es Ihnen hier auch nicht im Detail beschreiben, der Platz reicht einfach nicht.

Wenn es Ihnen jetzt noch nicht in den Fingern juckt, dann schauen Sie doch ins "Musikbuch zum Commodore 64" (vom Autor des SYNTHIMAT, Thomas Dachsel), Sie werden danach ganz bestimmt handeln ...

6.3 CAD für Handarbeit

```
*****  
*      MAN NEHME      :      *  
*****
```

```
*****  
*      HARDWARE      :      *  
*****  
*      *      *  
*      COMMODORE 64      *  
*      Floppy 1541      *  
*      Monochromer Monitor      *  
*      Koala Pad      *  
*      beliebiger Drucker      *  
*      *      *  
*****
```

```
*****  
*      SOFTWARE      :      *  
*****  
*      *      *  
*      SUPERGRAPHIK      *  
*      *      *  
*****
```

```
*****  
*      VORKENNTNISSE      *  
*****  
*      *      *  
*      Commodore-BASIC-Kenntnisse      *  
*      *      *  
*****
```

CAD, das ist wieder so ein Fachbegriff, den kaum jemand richtig zu übersetzen weiß. CAD steht für

Computer Aided Design = Computer-Unterstütztes Entwerfen

Ingenieure, Architekten, Erfinder und Grafiker arbeiten längst mit CAD. Aber deren Systeme bewegen sich in Preis-Kategorien, die für den Hobbyanwender außerhalb jeglicher Wünsche liegen. Bis vor kurzem, konnten sich selbst sonst prognosesichere Experten nicht vorstellen, daß es so etwas wie CAD-Software für Personal- oder gar Homecomputer geben könnte.

Und jetzt gibt es das schon. Sogenannte Low-Cost-Systems (Niedrig-Preis-Systeme) laufen z.B. auf dem IBM PC, dem Viktor SIRIUS I und den größeren Apple- und Commodore-Geräten. Aber low cost heißt immer noch um die DM 8.000 - 10.000.

Für den 64er gab es bislang nichts CAD-ähnliches. Und, um Ihnen gleich jede Illusion zu nehmen, auch heute existieren noch keine CAD-Software-Pakete für den C64. Wenn Sie also die Grafik-Möglichkeiten Ihres Computers zum ernsthaften Entwerfen nutzen möchten, sind Sie auf Problemlösungen angewiesen, die CAD nur simulieren. Ich habe mich mit solchen Lösungen für Sie beschäftigt.

Sie sind inzwischen sicher schon an meine merkwürdigen Beispielfälle gewöhnt, diesmal ist die Anwendungsidee höchstens auf den ersten Blick abseitig. Handarbeit - Nähen, Stricken, Häkeln - im landläufigen Sinne wird oft nur als Abendbeschäftigung für alte Omas gesehen. Aber haben Sie schon mal daran gedacht, aus diesem Hobby einen kleinen Nebenverdienst zu machen ?

Exklusive Strickwaren für Boutiquen entstehen zum ganz großen Teil auf den Strickmaschinen von Amateurrinnen, wobei dieser Begriff etwas zu klein ist. Ich kenne jedenfalls Damen, die Ihre Modelle beinahe professionell planen,

entwerfen und ausführen.

Der schwierigste Teil des Arbeitsablaufes ist das Umsetzen von Ideen in Material, lies: Wolle. Nicht immer ist das Gewünschte auch machbar. Viele Faktoren müssen bedacht und schließlich auch berechnet werden. Setzen Sie doch den 64er ein !

Stellen wir zunächst eine Liste der notwendigen Berechnungen auf:

- Konfektionsgröße - detaillierte Maße
- Konfektionsgrößen umrechnen
- Anzahl der aufzunehmenden Maschen
- Materialbedarf
- Reihenberechnungen für Muster

Ausgangspunkt für diese Berechnungen ist der Entwurf. Meine Überlegung war: Wenn man den Entwurf am Computer anfertigt, hat man gleichzeitig die Möglichkeit, Meßpunkte für die Berechnungen zu erfassen.

Nun, mit welchem Hilfsmittel kann überhaupt am Computer gezeichnet werden ?

- 1) Von einem BASIC-Programm aus mit Hilfe der Cursortasten
- 2) desgleichen mit Joysticks
- 3) mit einem Lichtgriffel und entsprechender Software
- 4) mit einem Grafiktablett
- 5) wie 1) und 2) aber mit einer Grafik-Erweiterung

Die Variante 3) scheidet aus dem einfachen Grund aus, daß ich keinen Lightpen mit brauchbarer Software finden konnte. Für die Möglichkeiten 1) und 2) spricht der geringe Hardware-Aufwand, dagegen das problematische Ansprechen der hochauflösenden Grafik von BASIC aus.

Ich habe mich, wie Sie schon aus dem Man-nehme-Block sehen

konnten, für eine Kombination von Grafiktablett (hier das KOALA PAD) plus der Befehlserweiterung SUPERGRAPHIK entschieden.

So und ab sofort will ich weniger von Computer- als viel mehr von Handarbeits-Technik sprechen.

Wie gewohnt, gehen wir die Sache Schritt für Schritt durch. und der erste Schritt ist der Rohentwurf. Sie nehmen das KOALA PAD auf den Schoß und beginnen freihändig einen schicken Pullover mit allen Mustern und Farben zu zeichnen. Das mag zunächst wie eine Kinderzeichnung anmuten, aber bedenken Sie, das ist nur die Ausgangsstufe. Wenn Sie sich schon ein wenig ans KOALA PAD gewöhnt haben, kennen Sie die vielen Möglichkeiten: verschiedene Strichdicken, Farben, feste Grafik-Zeichen und anderes.

Vermutlich ist das eine Beschäftigung, die Spaß macht, aber vom Entwerfen ablenkt. Also, Disziplin ! Nehmen wir an, Ihr Entwurf wäre fertig und gut gelungen. Dann müssen Sie ihn erstmal speichern. Auch das geht mit dem KP sehr einfach. Alles sicher ?

Okay, bitte alle Geräte abschalten, das KP-Steckmodul wird entfernt. Den Stecker des Tablettts können Sie im Control-Port lassen.

Nun kommt die "SUPERGRAPHIK" zum Einsatz. Falls Ihre Version noch ohne KOALA PAD-Unterstützung ist, müssen Sie sich beeilen und die alte Diskette gegen die neue Version umtauschen (Umtausch-Frist ist begrenzt, bitte bei DATA BECKER nachfragen !!).

Sie laden Ihren Pullover-Entwurf ein. Er müßte nun auf dem Bildschirm zu sehen sein. Mit dem KOALA PAD waehlen Sie nun Meßpunkte an. Für jeden Meßpunkt geben Sie die Maschenzahl ein und zwar bezogen auf die Mittelinie des Entwurfes. Sie wissen ja, bei den Anleitungen in Handarbeits-Zeitschriften wird auch immer nur eine Hälfte gezeigt. Nachdem alle

Meßpunkte bestimmt sind, zeichnet die SUPERGRAPHIK automatisch den bereinigten Entwurf.

Mit der eingebauten Hardcopy-Routine druckt Ihr Drucker nun einen Eins-zu-Eins- Schnittmusterbogen aus. Den können Sie als Gedankenstütze benutzen. Natürlich besteht auch die Möglichkeit, bestimmte Ausschnitte des Entwurfes auf dem Monitor zu betrachten; z.B. wenn Sie noch einmal genau wissen möchten, wieviel Maschen Sie am Halsausschnitt abnehmen müssen.

Ein Thema für sich ist die "Maschenprobe". Normalerweise geht die fleissige Handarbeiterin mit viel Intuition und Experimentieren daran. Aber, vorausgesetzt ich habe wirklich kapiert, wie diese ominöse Probe geht, eigentlich wäre auch das ein Job für den Computer.

Also, meine Damen, jetzt zeigen Sie Ihren Männern mal, welche Super-Programme Sie schreiben. Oder, falls Sie des BASIC nicht mächtig sein sollten, treiben Sie Ihren Partner zu programmtechnischen Hochleistungen an.

Mit einem Listing muß ich Sie diesmal verschonen: die ersten Entwürfe, mit denen ich experimentiert habe, waren schon so umfangreich, daß wohlmöglich ein eigenes Buch daraus geworden wäre.

Liebe Leserinnen, schließen wir ein Abkommen: wenn Sie der Meinung sind, dieses CAD für Strickarbeiten könnte Sie interessieren, schreiben Sie mir. Bei ausreichender Resonanz wird aus dieser Idee eventuell ein DATA-BECKER-Softwarepaket mit dem Namen STRICKOMAT. Wer weiß ...

6.4 Von Hamburg nach München

```

*****
*      MAN NEHME      :      *
*****

*****
*      HARDWARE      :      *
*****
*
*      COMMODORE 64      *
*      Floppy 1541      *
*      Beliebiger Monitor      *
*
*****

*****
*      SOFTWARE      :      *
*****
*
*      SUPERGRAPHIK 64      *
*
*****

*****
*      VORKENNTNISSE      *
*****
*
*      Gute BASIC-Kenntnisse      *
*      Etwas Übung im Umgang mit      *
*      SUPERGRAPHIK      *
*
*****

```

Wahrscheinlich sind Sie schon sehr gespannt, was es denn mit dieser Idee auf sich hat. Ich will es verraten: dies ist der Versuch, ein Modell für das Benutzen von Tabellen mit grafischen Hilfsmitteln. Dazu bietet sich ja das Problem Fahrtstrecken-Optimierung geradezu an.

Bevor wir uns auf Listings, PEEKs und POKEs stürzen, sollen Sie erstmal erfahren, wie das Ganze funktionieren soll. Dazu ein Beispiel:

Sie sind Außendienstler eines Großhändlers für Computer-Zubehör; Disketten, Staubschutzhauben etc. Ihr Vertriebsgebiet umfaßt die Postleitzahlen-Räume 4 und 5, Sie betreuen rund 200 Kunden. Wollen Sie bei denen nicht in Vergessenheit geraten, muß jeder Händler Sie möglichst einmal im Quartal zu sehen kriegen. Instinktiv (oder: aufgrund langjähriger Erfahrung) haben Sie sich fünf Touren zu je 4 Tagen zusammengestellt, die Sie regelmäßig abklappern.

Von Osnabrück im Norden bis Koblenz im Süden, von Aachen im Westen bis Paderborn im Osten geht Ihr Revier. Eines Tages keimt in Ihnen der Verdacht auf, daß Ihre Tour nicht optimal ist. Es dämmert Ihnen: "Ich fahre viel zu viele Kilometer!". Nun hat sicher jeder, der mit Computern zu tun hat, einen 64er im Stall (Oder ?) und damit wollen Sie dem Problem zu Leibe rücken.

Der Lösungsweg ist so simpel wie mühsam:

- Liste aller anzufahrenden Orte aufstellen
- Alle Entfernungen zwischen allen Orten auflisten
- Rechnen, rechnen und nochmal rechnen

Es geht also darum, den Vorgang so komfortabel wie möglich zu gestalten. Wie wäre es denn folgendermaßen ?

- 1) Auf dem Bildschirm erscheint die schematische Karte des Gebietes, die Orte werden je nach Anzahl der dort

ansässigen Kunden als verschieden große Kreise dargestellt. Als Kennung dient die PLZ.

- 2) Mit dem Stift wählen Sie (über das KOALA PAD) den Ausgangsort Ihrer Tour an
- 3) Anschließend zeigen Sie auf den Ort, den Sie als zweiten besuchen wollen.
- 4) Beide Kreise werden durch eine Linie miteinander verbunden. Gleichzeitig wird aus der Tabelle die Entfernung herausgesucht und gezeigt.
- 5) Sie verbinden Ort Nr.2 mit Ort Nr.3, die Linie entsteht und diese Entfernung wird addiert.
r09
- 6) Wenn Sie die Tour am Bildschirm fertig haben, verfügen Sie über die Gesamtfahrstrecke und können sich außerdem noch die Reihenfolge der Orte ausgeben lassen.

Nach Lust und Laune entwerfen Sie auf die gleiche Weise verschiedene Alternative-Routen. Drucken Sie alle Vorschläge und Sie können sich die günstigste Lösung aussuchen. Phantastisch, nicht wahr ?

"Puh, was wird das für ein Wahnsinns-Programm ..." höre ich Sie stöhnen. Halb so schlimm, kann ich dazu nur sagen; wir operieren im wesentlichen mit Programm-Moduln, die bei Ideen in diesem Buch schon benutzt wurden.

Erste Voraussetzung

```
-----  
4 Eine Liste der Orte. Etwa so: *  
4 *  
4PLZ ORT KUNDEN *  
4 *  
44000 Düsseldorf 14 *  
44005 Meerbusch 2 *  
44200 Duisburg 8 *
```

4500 Osnabrück	6	*
45300 Bonn	4	*
45000 Köln	13	*
44508 Bohmte	1	*
4usw.		*
4usw.		*
4		*

Zweite Voraussetzung

Die Tabelle mit den Entfernungen zwischen den Orten. Dabei hilft der Computer selbstverständlich. Das Programm gibt zwei Orte aus :

VON Düsseldorf BIS Bonn = ?

Sie lesen aus einer Tabelle den Wert und übertragen ihn. Das Programm spielt alle Kombinationen durch und speichert alles in einer Tabelle.

Dritte Voraussetzung

Sie müssen dem Rechner beim Zeichnen helfen. Und zwar fragt das Programm:

- Welcher Ort liegt am weitesten nördlich ?
- Welcher Ort südlich ?
- Welcher östlich ?
- Welcher westlich ?

Diese vier Orte erscheinen als erste auf der Mattscheibe. Nun geht die Fragerei weiter. Zu jedem Ort aus der Liste werden Sie nach der relativen Position zu den vier Eckpunkten gefragt.

Nach und nach entsteht so eine schematische Karte, wie Sie auf der folgenden Seite abgebildet ist.

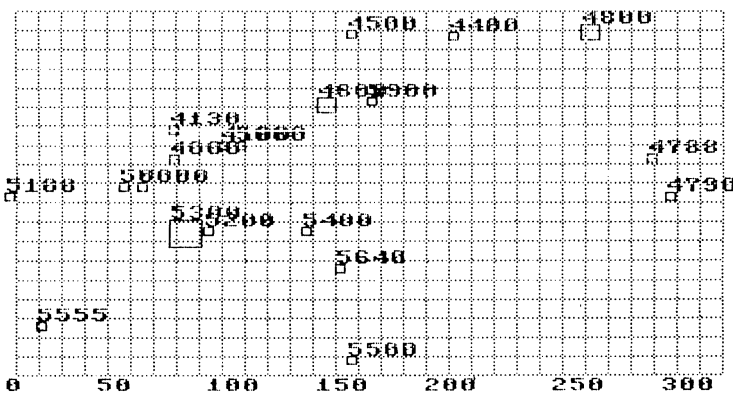
Vierte Voraussetzung

Vor jedem Programmlauf muß SUPERGRAPHIK geladen werden und das KOALA PAD angeschlossen sein. Wichtig ist auch, daß Sie sich vorher mit dem Funktionieren dieser beiden Hilfsmittel - zumindest durch intensive Lektüre des Handbuches - vertraut gemacht haben.

Die Programme

Ja, Sie haben richtig gelesen, es werden mehrere Listings sein. Um genau zu sein: 2. Das dritte, das Sie benötigen müssen Sie wieder einmal selber schreiben. Damit Sie auch was davon haben.

Das erste heißt sinnigerweise Programm 1 (Tabellen). Dies Programm für sich alleine ist schon ziemlich nützlich. Sie können sich Ihre Liste der Orte damit erstellen, speichern und ausgeben. Der zweite Teil hilft Ihnen, zu einer Entfernungstabelle zu kommen. Wenn auch solche Tabellen in manchem Taschenkalender enthalten sind, ist unser Beispiel besser, weil individuell anwendbar. Außerdem brauchen wir sie als Voraussetzung für das Graphik-Programm.



```

10000 rem * * * * *
10001 rem *
10002 rem * fahrtstrecken-optimierung *
10003 rem *
10004 rem *   aus dem data becker   *
10005 rem *   ideen-buch           *
10006 rem *
10007 rem * * * * *
10008 rem *
10009 rem *   programm 1 : tabellen *
10010 rem *
10011 rem * * * * *
10012 rem
10013 poke 53280,11 : poke 53281,11
10014 poke 53272,23 : print chr$(158)
10015 rem
10020 dim ol$(20,4) : rem ortsliste
10030 dim ef$(20,20) : rem entfernungs-tabelle
10040 rem
10050 goto 20000
11000 rem *** orts-liste erstellen ***
11001 rem
11010 z=1
11020 print chr$(147);"";
11030 print "   *** Orts-Liste erstellen ***"
11040 print : print
11050 print "   (1) Datensatz Nr.: ";z
11055 input "   bitte bestaetigen   ";ol$(z,1)
11060 input "   (2) PLZ                       ";ol$(z,2)
11070 input "   (3) Ort                          ";ol$(z,3)
11080 input "   (4) Kundenzahl                  ";ol$(z,4)
11090 z=z+1
11100 if z>20 then return
11110 goto 11020
11299 return
11300 rem *** orts-liste speichern ***
11301 rem
11310 print chr$(147);"";
11320 print "   *** Orts-Liste wird gespeichert ***"
11330 open 1,8,2,"$:ortsliste,s,w"
11340 for z=1 to 20
11350 :   print#1,ol$(z,1),ol$(z,2),ol$(z,3),ol$(z,4)
11360 next z
11370 close 1
11380 print chr$(147);"";
11390 print "   *** Orts-Liste ist gespeichert ***"
11400 for i=1 to 1000 : next : return
11499 return
11500 rem *** orts-liste einlesen ***
11501 rem
11510 print chr$(147);"";
11520 print "   *** Orts-Liste wird eingelesen ***"
11530 open 1,8,2,"ortsliste,s,r"
11540 for z=1 to 20
11550 :   for n=1 to 4
11560 :     input#1,ol$(z,n)
11570 :   next n

```

```

11580 next z
11590 close 1
11600 print chr$(147);"";
11610 print "    *** Orts-Liste ist eingelesen ***"
11620 for i=1 to 1000 : next : return
11799 return
11800 rem *** orts-liste ausgeben ***
11801 rem
11810 print chr$(147);"";
11820 print "    *** Orts-Liste ausgeben ***"
11830 print : print
11840 input "Auf Drucker (d) oder Bildschirm (b) ";w$
11850 if w$<>"d" and w$<>"b" then 11800
11860 if w$="b" then 11890
11870 open 4,4,2
11880 cmd 4
11890 for z=1 to 20
11900 :   for n=2 to 4
11910 :       print ol$(z,n)
11920 :   next n
11930 :
11940 next z
11950 if w$="b" then 11980
11960 rem
11970 close 4
11980 print "Zurueck ins Menu mit beliebiger Taste  "
11990 get q$ : if q$="" then 11990
11999 return
12000 rem *** entfernungs-tabelle erstellen ***
12001 rem
12010 print chr$(147);"";
12020 print "    *** Entfernungs-Tabelle erstellen ***"
12030 print
12040 for z=1 to 20
12050 :   for n=1 to 20
12060 :       o$(1)=ol$(z,3) : o$(2)=ol$(n,3)
12070 :       print "VON ";o$(1);" BIS ";o$(2);
12080 :       input " = Km ";ef$(z,n)
12090 :   next n
12100 next z
12110 return
12120 return
12299 return
12300 rem *** entfernungs-tabelle speichern ***
12301 rem
12310 print chr$(147);"";
12320 print "* Entfernungs-Tabelle wird gespeichert *"
12340 open 1,8,2,"$:entfernungen,s,w"
12350 for z=1 to 20
12360 :   for n=1 to 20
12370 :       print#1,ef$(z,n)
12380 :   next n
12390 next z
12400 print chr$(147);"";
12410 print "* Entfernungs-Tabelle ist gespeichert *"
12420 for i=1 to 1000 : next : return
12499 return

```

```

12500 rem *** entfernungs-tabelle einlesen ***
12501 rem
12510 print chr$(147);"";
12520 print "* Entfernungs-Tabelle wird eingelesen *"
12530 open 1,8,2,"entfernungen,s,r"
12540 for z=1 to 20
12550 :   for n=1 to 20
12560 :     input#1,ef$(z,n)
12570 :   next n
12580 next z
12590 print chr$(147);"";
12600 print "* Entfernungs-Tabelle ist eingelesen *"
12610 for i=1 to 1000 : next : return
12799 return
12800 rem *** entfernungen abfragen ***
12801 rem
12810 print chr$(147);"";
12820 print "      *** Entfernungen abfragen ***"
12830 print : print
12840 input "Entfernungs-Tabelle ausdrucken (j/n) ";w$
12850 if w$="j" then 13100
12860 print chr$(147);"";
12870 input "      VON ";o$(1)
12880 if len(o$(1))=4 then n=2
12890 if len(o$(1))>4 then n=3
12900 input "      BIS ";o$(2)
12910 if len(o$(2))=4 then n=2
12920 if len(o$(2))>4 then n=3
12930 m=1
12940 for z=1 to 20
12950 :   if ol$(z,n)=o$(m) then d(m)=z
12960 next z
12970 m=m+1 : if m<3 then 12940
12980 er$=ef$(d(1),d(2))
12990 print "      = ";er$;" Km"
12999 return
13000 print
13010 input "      Weitere Fragen j";w$
13020 if w$="n" then return
13030 goto 12860
13100 rem *** entfernungs-tabelle drucken ***
13101 rem
13110 open 4,4,2
13120 for z=1 to 20
13130 :   for n=1 to 20
13135 :     if n=z then 13170
13140 :     print#4,"VON ";ol$(z,2),ol$(z,3)
13150 :     print#4,"BIS ";ol$(n,2),ol$(n,3)
13160 :     print#4," = ";ef$(z,n)
13170 :   next n
13180 next z
13190 print#4
13200 close 4
13210 return
18540 input "      Alles gespeichert n";w$
19500 rem *** programm beenden ***
19501 rem

```

```

19510 print chr$(147);"";
19520 print "      ***   E N D E   ***"
19530 print : print
19540 input "          Alles gespeichert      n";w$
19550 if w$="n" then 20110
19560 end
20000 rem *** menue ***
20001 rem
20010 print chr$(147);"";
20020 input "Existiert eine Orts-Liste (j/n) ";in$
20030 if in$="n" then w=1 : goto 20350
20040 if in$="j" then w=3 : goto 20350
20050 if in$<>"n" and in$<>"j" then 20010
20060 print
20070 input "Existiert eine Entfernungstabelle ";in$
20080 if in$="n" then w=5 : goto 20350
20090 if in$="j" then w=7 : goto 20350
20100 if in$<>"n" and in$<>"j" then 20070
20110 print chr$(147);
20120 print " * * * * * ";
20130 print
20140 print "          FAHRTSTRECKEN-OPTIMIERUNG          ";
20150 print
20160 print "          Programm 1 : Tabellen          ";
20170 print
20180 print " * * * * * ";
20190 print : print
20200 print "          (1) Orts-Liste erstellen"
20210 print "          (2) Orts-Liste speichern"
20220 print "          (3) Orts-Liste einlesen"
20230 print "          (4) Orts-Liste ausgeben"
20240 print
20250 print "          (5) Entfernungstabelle erstellen"
20260 print "          (6) Entfernungstabelle speichern"
20270 print "          (7) Entfernungstabelle einlesen"
20280 print "          (8) Entfernungen abfragen"
20290 print
20300 print "          (9) Programm beenden"
20310 print
20320 input "          Bitte waehlen          ";w$
20330 w=val(w$) : if w>9 then 20000
20340 if w=9 then 19500
20350 on w gosub 11000,11300,11500,11800,12000,12300,12500,12800
20360 goto 20110
21000 poke768,61
21010 open15,8,15,"i":close15
21020 poke768,139
21030 if st<>128 then goto21100
21040 end
21100 open15,8,15
21110 input#15,ff,fb$,sp,se
21120 close15
21130 printff;fb$;sp;se
21140 end

```

Ich glaube, Kommentare erübrigen sich. Alle Teile sind irgendwann irgendwo in diesem Buch bereits aufgetaucht. Wenn Sie nur mit diesem Programm verwenden wollen, ohne Graphik-Modul, bieten sich zahlreiche Verbesserungen an; Routinen zum Ändern der Listen; komfortablere Ausgabe etc. Lassen Sie mal Ihre Phantasie walten.

In punkto Graphik sind Sie bisher vielleicht etwas zu kurz gekommen. Es sei hier nachgeholt. Das Programm 2 (Graphik) dient als zweite Voraussetzung für das projekt Fahrtstrecken-Optimierung.

```

9000 REM PROGRAMM BEGINN
9010 PRINT CHR$(147)
9020 PRINT CHR$(158)
9030 PRINT "BITTE WARTEN. GRAPHIK WIRD ERZEUGT !!!"
9040 DIM O$(20,3)
10000 REM KOORDINATEN-GRAPHIK
10010 GMODE 0,5:GCLEAR
10020 FCOL=11 : BCOL=11 : PCOL=7
10030 FOR X=5 TO 315 STEP 10
10040 :   PLOTFC X,0 TO X,190
10050 NEXT X
10060 FOR Y=0 TO 190 STEP 10
10070 :   PLOTFC 5,Y TO 315,Y
10080 NEXT Y
10090 K$="0    50    100    150    200    250    300"
10100 TEXTC K$,0,199,0
10110 GMODE 2,5:GCLEAR
10120 FCOL=11 : BCOL=11 : PCOL=7
10130 TEXTC K$,0,199,0
10140 GMODE 0,0
20000 REM ORTE EINZEICHNEN
20010 ST$=".1.1.2.2.3..3.0.0."
20020 H$(1)="NOERDLICHSTEN" : O$(1,2)=STR$(150) : O$(1,3)=STR$(10)
20030 H$(2)="SUEDLICHSTEN " : O$(2,2)=STR$(150) : O$(2,3)=STR$(180)
20040 H$(3)="WESTLICHSTEN " : O$(3,2)=STR$(0) : O$(3,3)=STR$(95)
20050 H$(4)="OESTLICHSTEN " : O$(4,2)=STR$(290) : O$(4,3)=STR$(95)
20060 RUT=0
21000 REM ORTE ABFRAGEN
21010 N=1
21020 PRINT CHR$(147)
21030 PRINT "      "
21040 PRINT "BITTE PLZ DER ";H$(N);"
21050 INPUT "STADT EINGEBEN ";P$: O$(N,1)=P$
21060 INPUT "ANZAHL DER KUNDEN (1 - 25) ";K
21070 GMODE 0,5
21080 SIZE=INT(K/5)+1
21090 PAINT ST$ FROM VAL(O$(N,2)),VAL(O$(N,3))
21100 TEXTC P$,VAL(O$(N,2)),VAL(O$(N,3)),0
21110 GMODE 2,5
21120 SIZE=INT(K/5)+1
21130 PAINT ST$ FROM VAL(O$(N,2)), VAL(O$(N,3))
21140 TEXTC P$,VAL(O$(N,2)),VAL(O$(N,3)),0
21150 GMODE 0,0
21160 N=N+1 : IF N<5 THEN 21020
21170 GMODE0,1
21180 FOR I=1 TO 5000 : NEXT
21190 GMODE2,1
21200 FOR I=1 TO 5000 : NEXT
22000 REM UEBRIGE ORTE EINTRAGEN
22010 GMODE 0,0
22020 PRINT CHR$(147);
22030 PRINT "      "

```

```

22040 REM
22050 PRINT "BITTE DIE ENTFERNUNGEN ZWISCHEN DEN "
22060 PRINT "EBEN MARKIERTEN ORTEN EINGEBEN !!"
22070 PRINT : PRINT
22080 PRINT "ZWISCHEN "O$(3,1)" UND "O$(4,1) : INPUT EE(1)
22090 PRINT
22100 PRINT "ZWISCHEN "O$(1,1)" UND "O$(2,1) : INPUT EE(2)
22110 PRINT
22120 ME(1)=(EE(1)/290) : REM X-RICHTUNG
22130 ME(2)=(EE(2)/170) : REM Y-RICHTUNG
22140 PRINT "DER ABSTAND ZWISCHEN ZWEI LINIEN IN"
22150 PRINT "X-RICHTUNG ENTSPRICHT ";ME(2);" KM."
22160 PRINT "DER ABSTAND ZWISCHEN ZWEI LINIEN IN"
22170 PRINT "Y-RICHTUNG ENTSPRICHT ";ME(1);" KM."
22180 FOR I=1 TO 5000 : NEXT
22190 FOR N=5 TO 20
22200 PRINT CHR$(147)
22210 PRINT " "
22220 PRINT "DIE POSITION WIRD AUS DER LAGE ZUM "
22230 PRINT H$(1);" ORT UND ZUM ";H$(3)
22240 PRINT "BESTIMMT."
22250 PRINT : PRINT " ENTFERNUNG (1) = MAX. ";EE(1);" KM"
22255 PRINT " ENTFERNUNG (2) = MAX. ";EE(2);" KM"
22260 PRINT "BITTE DIE PLZ DES ";N;".TEN ORTES"
22270 INPUT "EINGEBEN ";P$: O$(N,1)=P$
22280 INPUT "ZAHL DER KUNDEN ";K
22290 PRINT "ENTFERNUNG ZU ";O$(3,1)
22300 INPUT OE(1)
22310 PRINT "ENTFERNUNG ZU ";O$(1,1)
22320 INPUT OE(2)
22330 IF OE(1)<EE(1) AND OE(2)<EE(2) THEN 22360
22340 PRINT "DAS KANN NICHT SEIN ! BITTE KORRIGIEREN"
22350 FOR I=1 TO 1000 : NEXT : GOTO 22200
22360 X=INT(OE(1)/ME(1)) : Y=INT(OE(2)/ME(2))
22370 O$(N,2)=STR$(X) : O$(N,3)=STR$(Y)
22380 GMODE 0,5
22390 SIZE=INT(K/5)+1
22400 PAINT S1$ FROM VAL(O$(N,2)),VAL(O$(N,3))
22410 TEXTC P$,VAL(O$(N,2)),VAL(O$(N,3)),0
22420 GMODE 2,1
22430 SIZE=INT(K/5)+1
22440 PAINT S2$ FROM VAL(O$(N,2)),VAL(O$(N,3))
22450 TEXTC P$,VAL(O$(N,2)),VAL(O$(N,3)),0
22460 FOR I=1 TO 3000 : NEXT : GMODE 0,0
22470 NEXT N
22480 GSAVE1,"1.SCHEMAKARTE",8
22490 GSAVE2,"2.SCHEMAKARTE",8
22500 END

```

READY.

Wahrscheinlich haben Sie das Programm gleich abgetippt und gestartet. Und ? Haben Sie was gesehen ? Wenn nicht, dann haben Sie vergessen vorher SUPERGRAPHIK zu laden. Oder besitzen diese Befehlserweiterung nicht ?

Einige Erläuterungen: die Schemakarte wird auf beiden Graphik-Seiten gezeichnet; und zwar einmal mit, einmal ohne Koordinatensystem. Je nachdem, wie Sie weiter arbeiten wollen, können Sie sich für eine der Darstellungen entscheiden. Beide werden jedenfalls am Ende gespeichert.

Die Karte dürfen Sie selbstverständlich nicht für bare Münze nehmen, sie ist nicht maßstabsgetreu. Wenn Sie einmal verschiedene Karten entworfen haben, werden Sie feststellen, daß die Darstellung für unsere Zwecke völlig ausreicht.

So, das dritte Programm befaßt sich mit dem Zusammenstellen der Touren. Ein Tip noch: lassen Sie graphische Darstellung und rechnerisches Erarbeiten der Touren separat voneinander ablaufen. Und - studieren Sie das SUPERGRAPHIK-Handbuch !

Schlußbemerkung

Die Programme können beinahe unendlich variiert werden:

- ganz Deutschland
- Europa
- weltweit
- aber auch Stadtfahrtstrecken

Egal wie weit es reicht, als komfortable Entfernungstabelle ist es immer zu benutzen.

Wie immer: experimentieren Sie, bauen Sie aus, vereinfachen Sie was und wie Sie wollen; jedes Programm ist nur so gut, wie seine Anpassung an die Bedürfnisse des Anwenders !

6.5 Laufschrift und bewegte Bilder

```

*****
*      MAN NEHME      :      *
*****

*****
*      HARDWARE      :      *
*****
*
*      COMMODORE 64   *
*      Floppy 1541   *
*      Farbmonitor   *
*
*****

*****
*      SOFTWARE      :      *
*****
*
*      SUPERGRAPHIK  *
*
*****

*****
*      VORKENNTNISSE      *
*****
*
*      Commodore-BASIC-Kenntnisse *
*
*****

```

Vor nicht allzu langer Zeit sah man in vielen Schaufenstern kleine schwarze Kästen mit einem Display an der Vorderseite,

auf dem eine Laufschrift zu sehen war. Die Dinger sind inzwischen etwas aus der Mode oder durch Video-Monitore ersetzt.

Optische Werbeträger sind immer ein besonderer Lockvogel. Leider wird kaum ein Einzelhändler in der Lage sein, eigene Videos herstellen zu lassen und muß auf das zurückgreifen, was Firmen an Werbemitteln anbieten. Oft besteht jedoch Bedarf an aktueller Information über eine visuelle Anzeige.

Mit dem 64er kann jeder solche aktuelle Schaufenster-Werbung gestalten. Die Graphik-Erweiterung "SUPERGRAPHIK" unterstützt solche Do-it-yourself-Versuche. Sogar bewegte Bilder sind - in gewissem Umfang - möglich.

In einer Frage will ich Ihnen drei Problemlösungen anbieten: Wie lasse ich eine Laufschrift über den Monitor flitzen ?

1. Variante

Sie bewegen sich im ganz normalen LGR-Modus (also 24 Zeilen zu je 40 Zeichen). In einem oder mehreren Strings legen Sie den gewünschten Text ab, der auf dem Bildschirm ausgegeben wird. Mit dem Befehl "gmove" setzen Sie Ihre Info-Zeile in Bewegung. Es lohnt sich, verschiedene Varianten durchzuspielen; z.B. Mehrere Textzeilen, die nacheinander in Bewegung geraten. Feste Ober- und Unterzeilen, dazwischen bewegter Text usw.. Dazu ein kleines Programm (läuft natürlich nur, wenn vorher SUPERGRAPHIK geladen wurde !):

```
100 a$="SONDERANGEBOTE  !!!!!"
105 print chr$(147)
110 print ""a$
120 gmove1,8,12
130 goto120
```

2. Variante

Mit diesem Programm haben Sie die Möglichkeit, Ihre Texte in Graphiken hineinzuschreiben. Also malen Sie ein Bild, lassen Platz für die Textzeile und lassen diese durch das Gemälde laufen. Der Schlüssel hierzu ist der Befehl "text". testen Sie folgendes Listing:

```
100 input a$
110 print chr$(147)
120 gmode0,1:gclear
130 texta$,0,100,1
140 gmove1,10,12
145 for i=1to50 :next
150 goto 140
```

3. Variante

Dies ist die schwierigste, aber auch vielseitigste Alternative. Zunächst müssen Sie alle benötigten Zeichen als Strings für den Befehl "paint" definieren. Das ist recht schwierig, weil Sie sozusagen im Dunklen tapen; das Ergebnis Ihrer Zahlenkolonnen sehen Sie erst im eigentlichen Programm. Als nächstes werden Größe, Farbe und eventuelle Verzerrung bestimmt, dann kommt der Text auf den Bildschirm. Mit "gmove" bringen Sie das Ganze in Bewegung. Vorteil gegenüber den anderen Varianten: Die Schrift kann beliebig groß sein und läuft auch über vorhandene Grafik. Im Beispiel ist ein "A" definiert worden.

```
100 a$=".1.1.1.2.2.2.2.2.0.0.0.3.:"
110 gmode0,1:gclear
120 size=5:rot=0:painta$from20,20
130 for i=1to2000:next
140 gmode0,0:gclear
150 printchr$(147)
160 list
```

4. Variante

Die habe ich, zugegebenermaßen, noch nicht ausprobiert. Sie beruht darauf, daß die notwendigen Zeichen als Sprites definiert werden; viel Arbeit, ist das ! Aber die Möglichkeiten, damit zu gestalten sind einfach unglaublich. Ein Beispiel: Sie wollen Ihren Namen auf dem Bildschirm effektiv in Szene setzen. Ich nenne Sie mal "Paul". Vier Sprites werden definiert. Zu Beginn der Show setzen wir Sie in die vier Ecken der Mattscheibe. Und dann, bewegen sich die Lettern, wie von Geisterhand gesteuert über das Display und fügen sich zum Namen "Paul" zusammen. Nach einer Weile beginnen die Buchstaben-Sprites sich zu überlappen und bilden schließlich ein neues Symbol (ohne Bedeutung). Können Sie sich das vorstellen ? Nein, dann müssen Sie es halt ausprobieren !

6.6 Geo-Spiele

```

*****
*      MAN NEHME      :      *
*****

*****
*      HARDWARE      :      *
*****
*
*      COMMODORE 64      *
*      Floppy 1541      *
*      Farbmonitor      *
*      KOALA PAD      *
*
*****

*****
*      SOFTWARE      :      *
*****
*
*      SUPERGRAPHIK      *
*
*****

*****
*      VORKENNTNISSE      *
*****
*
*      Gute bis sehr gute      *
*      BASIC-Kenntnisse      *
*
*****

```

Unverhofft kommt oft, sagt man landläufig. Hier paßt der weise Spruch. Wer schon nicht mehr damit gerechnet hatte, daß was zum Spielen im Ideenbuch vorkommt, darf sich freuen.

Die Freude wird allerdings kurz. Denn es folgt ein kurzer, aber nützlicher Tip:

Kennen Sie "RISIKO" ? Das ist ein Stratgiespiel mit durchaus kriegerischem Charakter. Wollen Sie dieses oder ähnliche Spiele auf den 64er übertragen ? Das ist nicht so schwer.

Was die Regeln und Berechnungen angeht, werden Sie leider alleine klarkommen müssen. Wobei ich Ihnen helfen kann, ist beim Spielfeld. Lassen Sie Ihrer Phantasie völlig freien Lauf. Mit dem KOALA PAD zeichnen Sie die tollsten Landkarten und Spielfelder. Die so entstandenen Bilder können Sie in Ihre BASIC-Programme übertragen. Innerhalb der Programme können Sie mit selbstdefinierten Sprites beliebig auf diesen Bildern herumfahren und die jeweilige Position berechnen lassen.

Das einziges Problem, zu dem mir noch keine Lösung eingefallen ist, besteht darin, daß Texte nur einmal an eine bestimmte Stelle geschrieben werden können, wenn Sie versuchen, ein zweites Mal in der gleichen Zeile zu schreiben, werden Sie feststellen, daß das Bild an dieser Stelle zerstört wird.

Vielleicht kommt ja noch ein Experte auf den richtigen Dreh, vielleicht aber gibt es keine Lösung dieser Frage. Na, ich lasse mich überraschen.

Ein Wort noch zum Thema Schach mit 64er. Die angebotenen Schachprogramme genügen den Ansprüchen eines Gelegenheitsspielers, reichen aber kaum an die Spielstärke von Schachcomputern heran. Testen Sie ruhig selber. Was allerdings ungeheuer spannend und kreativ sein kann, ist das Entwickeln eigener Schach-Software. Soweit ich orientiert bin, besteht sogar ein Club der Schachprogramm-Entwickler, dessen Mitglieder auf ganz gewöhnlichen Homecomputern tolle Programme laufen lassen. Nur, mit BASIC kommen Sie bei diesem Thema nicht weit. Erst das Programmieren in Maschinensprache führt auf den richtigen Weg.

ACHTUNG AUTOREN

SIE haben ein gutes Programm oder ein Manuskript zu einem interessanten Buch geschrieben oder würden dies gerne tun

SIE würden das Ergebnis Ihrer Arbeit gerne in größeren Stückzahlen vermarktet sehen

SIE suchen dafür den leistungsfähigen Verlag und Vertriebspartner Ihres Vertrauens

WIR besitzen große Erfahrung in der professionellen Vermarktung von Software und Literatur, nicht nur in Deutschland, sondern auch weltweit

WIR suchen weitere Autoren, mit denen wir gemeinsame Erfolge erringen können

WIR haben in den letzten 12 Monaten mit dem Verkauf von 200.000 Büchern und 50.000 Programmen unsere Leistungsfähigkeit unter Beweis gestellt

SIE und **WIR** sollten zusammenarbeiten.

Bitte, senden Sie eine Beschreibung Ihres Programms oder Ihres Buchprojekts an Dr. Achim Becker c/o DATA BECKER, oder fordern Sie einfach unsere unverbindlichen „Informationen für Autoren“ an.



IHR GROSSER PARTNER FÜR KLEINE COMPUTER

DATA BECKER

Merowingerstraße 30 · 4000 Düsseldorf · Telefon (02 11) 31 00 10 · im Hause AUTO BECKER

Alles, was Sie schon immer über Ihren COMMODORE wissen wollten!



DAS
MASCHINENSPRACHE-
BUCH ZUM
COMMODORE 64
1984, ca. 200 Seiten,
DM 39,-

Eine leicht verständliche Einführung in die Programmierung in Maschinensprache für alle, denen die Programmierung des Commodore 64 in BASIC nicht mehr ausreicht. Sie lernen Aufbau, Arbeitsweise und Register des 6510-Mikroprozessors kennen und werden über Bit, Bytes, Daten und Adressen aufgeklärt. Anhand von Beispielen lernen Sie alle Befehle des 6510 kennen und anwenden. Dabei werden die Analogien zu BASIC Ihnen beim Verständnis helfen. Ein weiteres Kapitel beschäftigt sich mit der Eingabe von Maschinenprogrammen. Dort erfahren Sie auch alles über Monitor-Programme sowie über Assembler. Zum einfachen und komfortablen Erstellen Ihrer eigenen Maschinensprache enthält das Buch einen kompletten Assembler, damit Sie gleich von Anfang an komfortabel und effektiv programmieren können. Weiterhin finden Sie dort einen Disassembler, mit dem Sie sich Ihre Maschinenprogramme oder die Routinen des BASIC-Interpreters und des Betriebssystemes ansehen können. Ein besonderer Clou ist ein in BASIC geschriebener Einzelschrittssimulator, mit dem Sie Ihre Programme schrittweise ausführen können. Dabei werden Sie nach jedem Schritt über Registerinhalte und Flags informiert und können den logischen Ablauf Ihres Programms verfolgen. Eine unschätzbare Hilfe besonders für den Anfänger. Als Beispielprogramm finden Sie ausführlich beschriebene Routinen zur Grafikprogrammierung und für BASIC-Erweiterungen. Natürlich sind alle Beispiele und Programme auf Ihren Commodore 64 zugeschnitten.



DAS TRAININGSBUCH
ZUM SIMON's BASIC,
1984, ca. 300 Seiten,
DM 49,-

SIMON's BASIC ist ein Hit — wenn man es richtig nutzen kann. Deshalb gibt es jetzt zu dieser vielseitigen Befehlsenerweiterung unser umfangreiches Trainingsbuch. Auf über 300 Seiten erklärt es Ihnen detailliert den Umgang mit den über 100 Befehlen des SIMON's BASIC. Alle Befehle werden ausführlich dargestellt, auch die, die nicht im Handbuch stehen! Natürlich zeigen wir auch die Masken des SIMON's BASIC und geben wichtige Hinweise, wie man diese umgeht. Zum Buch gehören zahlreiche Beispielprogramme und viele interessante Programmiertricks. Alle Befehle sind entsprechend ihren Anwendungen in entsprechende Kapitel geordnet: Einführung in das CBM BASIC 2.0 — Programmierhilfen — Fehlerbehandlung — Programmschutz — Programmstruktur — Variablen — Zahlenbehandlung — Eingabekontrolle — Ein/Ausgabe Peripheriebefehle — Graphik — Zeichensatzerstellung — Sprites — Musik — Steuernde Peripherie — SIMON's BASIC und die Verträglichkeit mit anderen Erweiterungen und Programmen. Dazu ein umfangreicher Anhang. Nach jedem Kapitel finden Sie Testaufgaben zum optimalen Selbststudium und zur Lernerfolgskontrolle. DAS TRAININGSBUCH ZUM SIMON's BASIC sollte jeder Anwender dieser universellen Befehlsenerweiterung unbedingt haben.

DATA BECKER BÜCHER



DAS GROSSE
FLOPPY-BUCH,
1983, ca. 320 Seiten,
DM 49,-

Darauf haben Sie gewartet: Endlich ein Buch, das Ihnen ausführlich und verständlich die Arbeit mit der Floppy VC-1541 erklärt. DAS GROSSE FLOPPY BUCH ist für Anfänger, Fortgeschrittene und Profis gleichermaßen interessant. Sein Inhalt reicht von der Programmspeicherung bis zum DOS-Zugriff, von der sequentiellen Datenspeicherung bis zum Direktzugriff, von der technischen Beschreibung bis zum ausführlich dokumentierten DOS-Listing, von den Systembefehlen bis zur detaillierten Beschreibung der Programme der Test/Demodiskette. Exakt beschriebene Beispiel- und Hilfsprogramme ergänzen dieses neue Superbuch. Aus dem Inhalt: Der erste Kontakt — Das Speichern von Programmen — Die Floppy-Systembefehle — Sequentielle Datenspeicherung — Relative Datenspeicherung — Die Fehlermeldungen und Ihre Ursachen — Der Direktzugriff — Der Zugriff auf das DOS — Technik der Floppy und der Diskette — DOS-Listing der VC-1541 — Dienstprogramme — BASIC-Erweiterungen und Programme zur komfortablen Benutzung der VC-1541 - Overlaytechnik — Diskmonitor — IEC-Bus und serieller Bus — Gemeinsamkeiten mit den großen CBM-Floppies und Unterschiede gegenüber der VC-1541. Mit dem GROSSEN FLOPPY-BUCH meistern Sie auch Ihre Floppy.



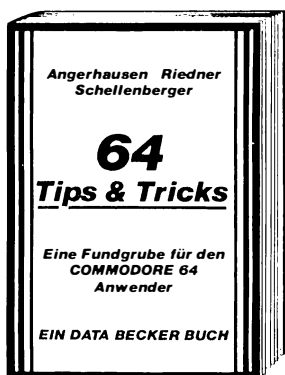
64 FÜR PROFIS
1983, ca. 280 Seiten,
DM 49,-

Wer besser und leichter in BASIC programmieren möchte, der braucht dieses Buch. 64 FÜR PROFIS zeigt, wie man erfolgreich Anwendungsprobleme in BASIC löst und verrät die Erfolgsgeheimnisse der Programmierprofis. Vom Programm-entwurf über Menüsteuerung, Maskenaufbau, Parameterisierung, Datenzugriff und Druckausgabe bis hin zur guten Dokumentation wird anschaulich mit Beispielen dargestellt, wie Profiprogrammierung vor sich geht. Besonders stolz sind wir auf die völlig neuartige Datenzugriffsmethode QUISAM, die in diesem Buch zum ersten Mal vorgestellt wird. QUISAM erlaubt eine beliebige Datenzusatzlänge, die dynamisch mit der Eingabe der Daten wächst. Eine lauffertige Literaturstellenverwaltung veranschaulicht die Arbeitsweise von QUISAM. Neben diesem Programm finden Sie auch noch weitere Programme zur Lager- und Adressenverwaltung, Textverarbeitung und einen Reportgenerator. Alle diese Programme sind mit Variablenliste versehen und ausführlich beschrieben. Damit sind diese für Ihre Erweiterungen offen und können von Ihnen an Ihre speziellen Bedürfnisse angepaßt werden. Mit 64 FÜR PROFIS steigen Sie in die Welt der Programmierprofis ein.

MIT DATA BECKER BÜCHERN MACHEN

SIE MEHR AUS IHREM COMMODORE

DATA BECKER BÜCHER



64 TIPS & TRICKS,
2. Auflage,
1983, ca. 290 Seiten,
DM 49,-

64 Tips & Tricks ist eine hochinteressante Sammlung von Anregungen zur fortgeschrittenen Programmierung des COMMODORE 64, Poke's und anderen nützlichen Routinen, interessanten Programmen sowie aktuellen Programmertips und -tricks. Aus dem Inhalt: 3D-Graphik in BASIC — Farbige Balkengraphik — Definition eines eigenen Zeichensatzes — Die Tastaturbelegung und ihre Änderung — Dateneingabe mit Komfort — Simulation der Maus mit einem Joystick — BASIC für Fortgeschrittene — Ihr COMMODORE 64 spricht Deutsch — CP/M auf dem COMMODORE 64 — Druckeranschluß über den USER-Port — Datenübertragung von und zu anderen Rechnern — Der Expansionsport — Synthesizer in Stereo — Retten einer nicht ordnungsgemäß geschlossenen Datei — Erzeugen einer BASIC-Zeile in BASIC — Der Kassettenpuffer als Datenspeicher — Sortieren von Stringfeldern — Multitasking auf dem COMMODORE 64 — POKE's und die Zeropage — GOTO, GOSUB und RESTORE mit berechneten Zeilennummern INSTR und STRING-Funktion — Repeatfunktion für alle Tasten — und vieles andere mehr. Alle Maschinenprogramme mit BASIC-Ladeprogrammen. 64 TIPS & TRICKS ist eine echte Fundgrube für jeden COMMODORE 64 Anwender.



64 INTERN,
3. Auflage,
1983, ca. 320 Seiten,
DM 69,-

64 INTERN ist unser großes Buch zum COMMODORE 64. Ideal für alle, die sich näher mit Programmierung, Technik und Betriebssystem von COMMODORE's Supermaschine auseinandersetzen wollen. Detailliert werden Architektur und technische Möglichkeiten des C-64 beschrieben, vom Prozessor 6510 über Video-Controller 6569, Sound Controller 6581, Ein-/Ausgabesteuerung mit CIA 6526 bis hin zum seriellen IEC-Bus, USER-Port und EXPANSION-PORT. Parallel dazu wird die Programmierung der hochauflösenden Graphik und der Sprites erklärt, die Sonderprogrammierung und die Programmierung des C-64 in Maschinensprache. Mit einem ausführlich dokumentierten ROM-Listing wird das Betriebssystem zerlegt. Wertvolle Hinweise zum Umsetzen von Programmen bringt ein Vergleich zwischen VC-20, C-64 und CBM 8000. Zahlreiche lauffertige Beispielprogramme und Routinen, z. B. Graphik-Aid, HARDCOPY, RENEN und PRINT USING runden das Buch ab. Der Clou von 64 INTERN sind neben den zahlreichen Schaltbildern und Blockdiagrammen 2 Original COMMODORE-Schaltpläne zum Ausklappen, die zusätzlich sehr ausführlich beschrieben und dokumentiert sind. Dieses Buch sollte jeder COMMODORE 64-Anwender und Interessent haben.

MIT DATA BECKER BÜCHERN MACHEN

SIE MEHR AUS IHREM COMMODORE



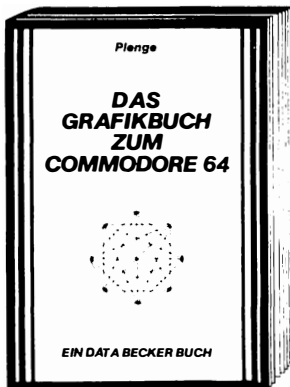
Der COMMODORE 64 ist ein Musikgenie. Mit diesem Buch lernen Sie alles über seine musikalischen Fähigkeiten und wie Sie diese programmtechnisch nutzen. Der Inhalt des Musikbuches reicht von einer Einführung in die Computermusik über die Erklärung der Hardware-Grundlagen und die Programmierung in BASIC bis hin zur fortgeschrittenen Musikprogrammierung in Maschinensprache. Zahlreiche Beispielprogramme, komplette Songs und nützliche Routinen ergänzen den leicht verständlichen Text. Geschrieben wurde das Buch von Thomas Dachselt, dem Autor der weltbekanntesten Musikprogramme SYNTHIMAT und SYNTHESOUND. Erschließen Sie sich die Welt des Sounds und der Computermusik mit dem Musikbuch zum COMMODORE 64.

DAS MUSIKBUCH ZUM
COMMODORE 64
1984, ca. 200 Seiten, DM 39,—



So etwas haben Sie gesucht: Umfassendes Nachschlagewerk zum COMMODORE 64 und seiner Programmierung. Allgemeines Computerlexikon mit Fachwissen von A-Z und Fachwörterbuch mit Übersetzungen wichtiger englischer Fachbegriffe - das DATA BECKER LEXIKON ZUM COMMODORE 64 stellt praktisch drei Bücher in einem dar. Es enthält eine unglaubliche Vielfalt an Informationen und dient so zugleich als kompetentes Nachschlagewerk und als unentbehrliches Arbeitsmittel. Viele Abbildungen und Beispiele ergänzen den Text. Ein Muß für jeden COMMODORE 64 Anwender.

DAS DATA BECKER
LEXIKON ZUM
COMMODORE 64
1984, ca. 350 Seiten, DM 49,—



Graphik ist eine der Hauptstärken des COMMODORE 64. Mit diesem leicht verständlich geschriebenen, aber sehr umfassenden neuen Buch lernen Sie, wie Sie die graphischen Fähigkeiten für Ihre eigenen Programme optimal nutzen. Der Inhalt reicht von den Grundlagen der Graphikprogrammierung über das Erzeugen einfacher Figuren, die Arbeit mit Sprites, Zeichensatzprogrammierung, Hardcopy und IRQ-Handhabung bis hin zur Funktionendarstellung, Laufschrift, Statistik, 3-D, CAD, den Geheimnissen der Actionsspiele und Lightpenanwendungen. Zahlreiche Beispielprogramme, nützliche Routinen und komplette Anwendungen ergänzen diese Buch, das die faszinierende Welt der Computergraphik jedermann zugänglich macht. Geschrieben wurde das Graphikbuch von dem bekannten SUPERGRAPHIK-Autor Axel Plenge.

DAS GRAFIKBUCH
ZUM COMMODORE 64 1984, ca. 250 Seiten, DM 39,—



Das sollte Ihr erstes Buch zum COMMODORE 64 sein: 64 FÜR EINSTEIGER ist eine sehr leicht verständliche Einführung in Handhabung, Einsatz, Ausbaumöglichkeiten und Programmierung des COMMODORE 64, die keinerlei Vorkenntnisse voraussetzt. Sie reicht vom Anschluß des Geräts über die Erklärung der einzelnen Tasten und Funktionen sowie die Peripheriegeräte und ihre Bedienung bis zum ersten Befehl. Schritt für Schritt führt das Buch Sie in die Programmiersprache BASIC ein, wobei Sie nach und nach eine komplette Adressverwaltung erstellen, die Sie anschließend nutzen können. Zahlreiche Abbildungen und Bildschirmfotos ergänzen den Text. Viele Anwendungsbeispiele geben nützliche Anregungen zum sinnvollen Einsatz des COMMODORE 64. Das Buch ist sowohl als Einführung als auch als Orientierung vor dem 64er Kauf gut geeignet.

64 FÜR EINSTEIGER
1984, ca. 200 Seiten, DM 29,—



Diese neue, umfangreiche Programmsammlung hat es in sich. Über 50 Spitzenprogramme für den COMMODORE 64 aus den unterschiedlichsten Bereichen, von attraktiven Superspielen („Senso“, „Pengo“, „Seeschlacht“, „Poison Square“, „Memory“) über Graphik- und Soundprogramme („Fourier 64“, „Akustograph“, „Funktionsplotter“) sowie Utilities („SORT“, „Renummer“, „Disk Init“, „Menue“) bis hin zu kompletten Anwendungsprogrammen wie „Videothek“, „File Manager“ und einer komfortablen Haushaltsbuchführung, in der fast professionell gebucht wird. Der Hit sind zu jedem Programm aktuelle Programmtips und Tricks der einzelnen Autoren zum Selbermachen. Also — nicht nur abtippen, sondern auch dabei lernen und wichtige Anregungen für die eigene Programmierung sammeln.

DATA BECKER'S GROSSE 64er PROGRAMMSAMMLUNG
1984, ca. 250 Seiten, DM 49,—



Achtung Hobbyelektroniker: Dieses neue Buch enthält nicht nur alles über die Ausbaumöglichkeiten des COMMODORE 64 und wie man ihn über seine Schnittstellen mit dem „Rest der Welt“ verbindet, sondern auch umfassende Informationen über die vielfältigen Einsatzmöglichkeiten des COMMODORE 64 von der Lichtorgel über Motorsteuerung, Spannungs- und Temperaturmessung bis zur programmierbaren Stromversorgung, und wie man diese verwirklicht. Umfassende Darstellung der Grundlagen. Als Clou enthält das Buch zehn komplette Schaltungen zum Selberbauen, vom Eprommer über Eprom-Karte, Logic Analyzer, Frequenzzähler, Hardware-Tracer, Pulsmeßgerät, Klatschschalter und Digital-Voltmeter bis zur preiswerten Spracheingabe-Sprachausgabe. Jeweils komplett mit Schaltplan, Layout und Softwarelisting. Auch für Profis interessant.

DER COMMODORE 64 UND
DER REST DER WELT 1984, ca. 220 Seiten, DM 49,—

DATA BECKER BÜCHER



VC-20 INTERN,
2. Auflage,
1983, ca. 230 Seiten,
DM 49,-

Die bereits 3. Auflage von VC-20 INTERN ist wieder erheblich erweitert worden. Das Buch beschäftigt sich ausführlich mit der Technik und dem Betriebssystem des VC-20. Dazu gehört natürlich zuerst einmal ein komplettes und ausführlich dokumentiertes ROM-Listing. Dazu gehört auch die Belegung der ZEROPAGE, dem wichtigsten Speicherbereich für den 6502-Processor, eine übersichtliche Auflistung der Adressen aller Betriebssystemroutinen, ihrer Bedeutung und ihrer Übergabeparameter. Dies ermöglicht dem Programmierer endlich, den VC-20 von Maschinensprache aus sinnvoll einzusetzen. Denn warum Routinen, die bereits vorhanden sind, noch einmal schreiben? Für Einsteiger befindet sich eine Einführung in die Programmierung in Maschinensprache am Anfang des Buches. Diese behandelt die Themen Maschinensprache-monitor, Disassembler und Assembler. Außerdem wird die Verbindung von BASIC- und Maschinenspracheprogrammen besprochen. Doch nicht nur die Software, auch die Hardware wird ausführlich beschrieben. Detailliert werden alle wichtigen IC's im VC-20 in Ihrer Arbeitsweise beschrieben. Neben einem übersichtlichen Blockschaltbild enthält VC-20 INTERN noch als besonderen Clou drei Original COMMODORE Schaltpläne zum Ausklappen. Dieses Buch braucht jeder, der sich näher mit Technik und Maschinenspracheprogrammierung des VC-20 auseinandersetzen möchte.



VC-20 TIPS & TRICKS,
2. Auflage,
1983, ca. 230 Seiten,
DM 49,-

Die überarbeitete und erheblich erweiterte 2. Auflage von VC-20 TIPS & TRICKS enthält endlich eine genaue Erklärung des Speicheraufbaus des VC-20 in allen möglichen Erweiterungsstufen. Vergessen Sie alles, was Ihnen über Grafik und Sound auf dem VC-20 bisher geliefert wurde. In diesem Buch finden Sie von den Grundlagen angefangen über die programmierbaren Zeichen, über eine Grafikerweiterung in Maschinensprache, über luxuriöse Programme wie z. B. Grafikeditor, Funktionenplotter oder 3D-Grafiken bis hin zur Supererweiterung zum VC-20 alles Wissenswerte und zum Arbeiten Notwendige für eine gute Grafik. Ebenso ausführlich wird die Tonerzeugung behandelt. Neben der Erläuterung der Grundlagen und einem Soundeditor sind natürlich auch fertige Programme, z. B. der VC-20 als Synthesizer oder Schlagzeug abgedruckt. Auf über 100 Seiten finden Sie viele nützliche Tricks, POKE-Befehle, BASIC-Erweiterungen fix und fertig zum Eintippen. Dazu gehören z. B. Unnew, automatische Zeilennummerierung, die Programmierung des User-Port, ein Diskmenue, Hardcopy und so weiter. Bringen Sie Ihrem VC-20 völlig neue Befehle bei! Abgerundet wird das Buch durch Anwenderprogramme zum Eintippen, wie Dateiverwaltung, Textverarbeitung und Spiele. VC-20 TIPS & TRICKS braucht jeder VC-20 Besitzer, der mehr über seinen Computer wissen möchte.

DAS STEHT DRIN:

Fast alles, was man mit dem COMMODORE 64 machen kann, ist in diesem Buch ausführlich beschrieben. Es ist nicht nur interessant geschrieben wie ein Roman, sondern enthält neben nützlichen Programmlistings vor allem viele, viele Anwendungsmöglichkeiten des C 64. Dabei wurde besonderer Wert darauf gelegt, daß das Buch auch für Laien leicht verständlich ist. Eine Auswahl aus der Themenvielfalt:

- Gedichte vom Computer
- Einladung zur Party
- Diplomarbeit - professionell gestaltet
- individuelle Werbebriefe
- Autokosten im Griff
- Baukostenberechnung
- Taschenrechner
- Rezeptkartei
- Lagerliste
- persönliches Gesundheitsarchiv
- Diätplan elektronisch
- intelligentes Wörterbuch
- kleine Notenschule
- CAD für Handarbeit
- Routenoptimierung
- Schaufensterwerbung
- Strategiespiele

UND GESCHRIEBEN HAT DIESES BUCH:

Rainer Bartel (32) ist Journalist. Er kam über die Schachcomputer zum Homecomputer. Er interessiert sich weniger für Technik, als vielmehr für die praktischen Einsatzmöglichkeiten des Homecomputers im Alltag.