

# ERSTAUNLICHE COMPUTERWELT

Gerhard Saeltzer



## Dank

Ein Buch ist stets das Ergebnis gemeinsamer Arbeit. Mitgeholfen haben meine Frau Christine durch viel Verständnis und Herr Prof. Dr. Siegfried Pilz mit Hinweisen. Allen danke ich.

Gerhard Saeltzer



Das ist Tom. Er besucht die Arbeitsgemeinschaft CC (wie Computer-Club). Er kennt sich schon ganz gut aus in der Welt des Computers und wird dich auf wichtige Dinge hinweisen.



GERHARD SAELTZER

# ERSTAUNLICHE COMPUTERWELT

---

ILLUSTRATIONEN  
VON WOLFGANG PARSCHAU

DER KINDERBUCHVERLAG BERLIN

## **Lieber Leser!**

*Dieses Buch mit seinen vielen Bildern handelt von einer der wichtigsten, großartigsten und folgenreichsten Erfindungen des Menschen – dem Computer oder Rechenautomaten. Manchmal werden große Computer auch als Datenverarbeitungsanlagen bezeichnet.*

*Computer sind schon heute unentbehrliche und leistungsfähige Helfer des Menschen auf allen wichtigen Gebieten. Ihre Anwendung prägt in der Zukunft noch mehr als bisher die Wissenschaft, die Technik, die Arbeit, die Freizeit – unser Leben insgesamt. Immer mehr Menschen werden täglich mit Computern Umgang haben.*

*In diesem Buch lernst Du viel Interessantes kennen. Auf die Fragen*

*Wie sehen die verschiedenen Computertypen aus?*

*Was können Computer, und was können sie nicht?*

*Machen Computer auch Dummheiten oder Späße?*

*Wie wird ein Computer angewandt?*

*Was passiert im Innern eines Computers?*

*Welche Auswirkungen hat ihr Gebrauch?*

*Welche Berufe haben mit Computern zu tun?*

*Was können künftige Computer und Roboter?*

*sowie auf viele andere Fragen wird es Dir eine Antwort geben. Du erhältst auch Gelegenheit, erworbene Kenntnisse im Wissenstest zu überprüfen. Zum Basteln und Knobeln hält es viele Tips für Dich bereit.*

*Am Schluß des Buches erwartet Dich noch eine Überraschung. Falls Du keinen Computer hast, kannst Du Dir einen Computer aus Karton basteln, mit diesem Modell spielen und interessante Versuche anstellen.*

*Viel Spaß beim Eindringen in die erstaunliche Welt des Computers, in dieses Gebiet mit großer Zukunft, wünscht Dir*

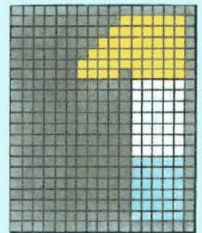
*Gerhard Saeltzer*



# 1. Eine unterhaltsame Spielerei

---

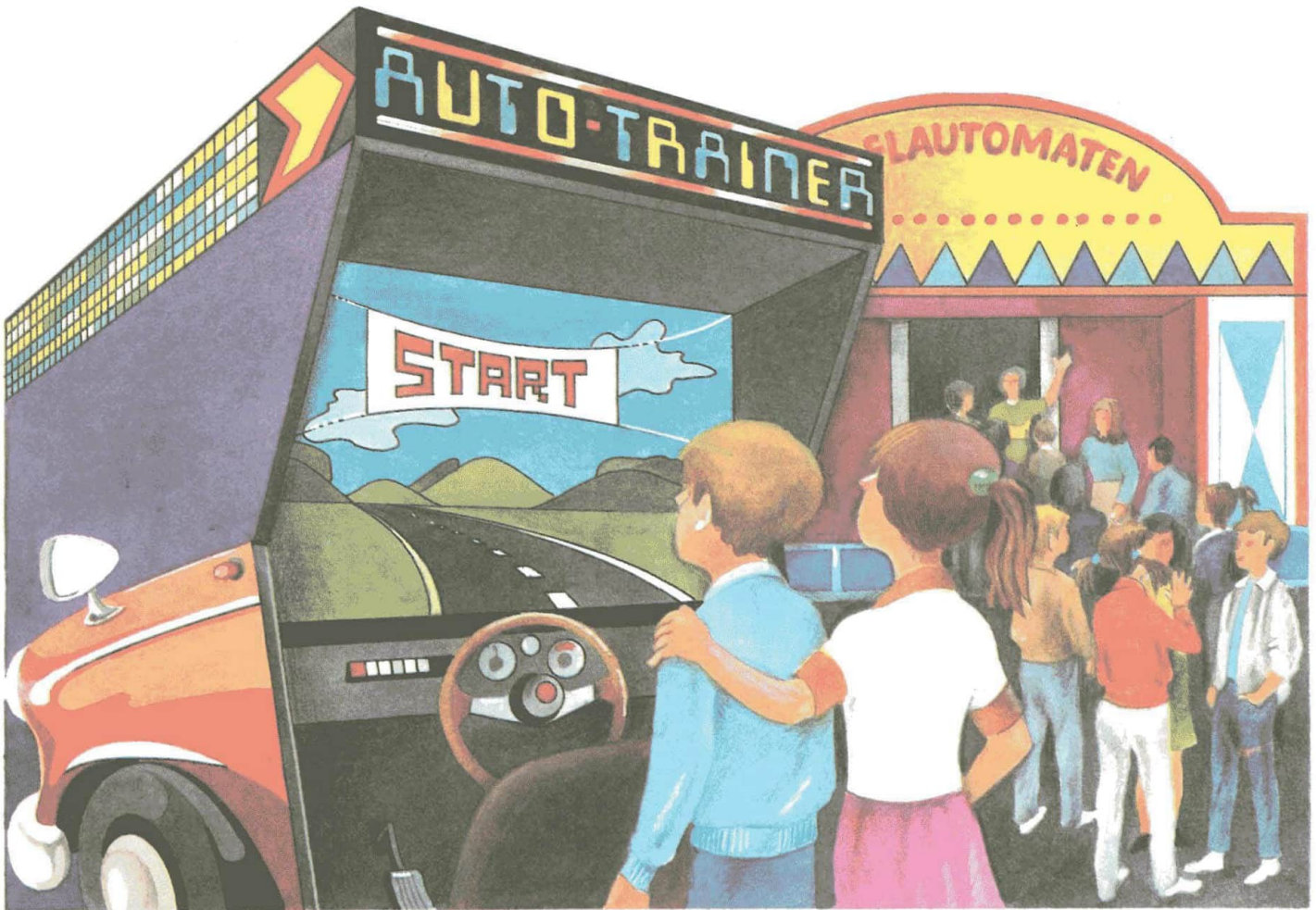
Einem Computer kann man im Alltag in vielerlei Gestalt begegnen. Vielleicht auch bei einem Spielchen...



## Wieder ist Frühlingsfest!

Kinder, Jugendliche und Erwachsene strömen zu dem vergnüglichen Rummel. Sie lassen sich in Karussells, in Berg-und-Tal-Bahnen herumwirbeln, erleben die aufregende Fahrt mit dem Riesenrad, versuchen ihr Geschick und ihr Glück in Spiel- und Losbuden. Laute Musik und der Duft von Bratwürsten ziehen über den Platz...

Immer dicht umlagert ist der Wagen, auf dem in flackernder Lampenschrift steht: Spielautomaten. Wir





müssen warten und uns langstrecken, wenn wir etwas sehen wollen. Wer noch klein ist, hat es schwer.

Endlich gibt unser Vordermann den Blick frei. Da steht er nun vor uns, der Automat. Oben ein eingebautes Farbfernsehgerät, vorn ein Lenkrad wie in einem Auto. Über den Bildschirm huschen „lebende“ Bilder.

Ein neues Spiel beginnt. Wir werfen eine Münze ein. Auf dem Bildschirm erscheint das farbige Titelbild des Spiels:



Jetzt sehen wir den Wagen. Er schaut aus wie eine Weltraumrakete auf Rädern.

Dann wird der Gang angezeigt: erster, zweiter, dritter..., bis wir das Lenkrad bewegen. Wir fahren mit Gang acht.

Die erste Tour ist eine Übungsfahrt. Das Raketenauto startet, seine Geschwindigkeit nimmt ständig zu, bis es gleichmäßig schnell die Straße entlangsaust. Der Motor heult. Bewegen wir das Lenkrad, so folgt der Wagen in die gewünschte Richtung. Es ist ein Simulator genanntes Gerät, das alle von uns verursachten Fahrsituationen im Bild nachahmt.

Plötzlich tauchen Hindernisse auf. Die Straße wird schmaler. Bäume, Häuser, ein parkendes Auto, ein müder Radfahrer, ein langsam dahinrollender Lastzug. Alles kommt auf uns zu. Wir müssen vorbeisteuern. Gelingt es nicht, gibt es einen Zusammenstoß: Es



ertönt ein lauter Knall, auf dem Bildschirm blitzt und funkt es. Dann erscheint ein Reparaturmännchen und repariert am Straßenrand unseren defekten Rennwagen.

Nach dieser Unterbrechung geht die Jagd weiter – durch Rechtskurven, Linkskurven, S-Kurven, weiter über Brücken, durch Tunnel. Dann die Zielgerade.

Die Wertungstafel für die Übungstour erscheint am Bildschirm:



Wir wissen Bescheid. Nun beginnt die „heiße“ Wertungsfahrt...

Wer erzeugt die bewegten Bilder einer Autofahrt in der Spielmaschine? Wer ist der unsichtbare Spielpartner auch in anderen Spielautomaten, zum Beispiel dem Schachautomaten? Wer oder was ermöglicht solche unterhaltsamen Spiele?

Vor Jahrhunderten gab es auch schon interessante Spielmaschinen. Aber in manch einer saß versteckt ein Mensch. Heute ist der Spielpartner ein kleines technisches Wunderwerk: der Computer.

Spielautomaten sind eine spezielle Anwendung des Computers. Weitere Anwendungen des Computers werden wir jetzt kennenlernen.



## 2. Zwanzig Begegnungen mit Computern

---

Computer sind vielfältig einsetzbar. Sie wurden dem Menschen auf vielen Gebieten unentbehrliche Helfer. Zweimal zehn Beispiele zeigen das.





## Computer im Alltag

### In der Schule und bei den Mathe-Hausaufgaben

können wir schwierige Berechnungen rasch und genau mit dem Taschenrechner ausführen. Er ist der Zwerg unter den Computern.

### In großen Kaufhallen

berechnet an der Kasse lautlos ein Computer, was wir zu bezahlen haben und welches Wechselgeld wir zurückerhalten.

### Bei internationalen Sportwettkämpfen

sind auch Computer dabei. Sie werten die Ergebnisse aus und zeigen sie uns geordnet an großen Leuchtafeln oder in der Fernsehsendung.

### In der Großstadt

werden Ampelanlagen an wichtigen Straßenkreuzungen von Computern gesteuert, die untereinander verbunden sind. Motorräder, PKW, Omnibusse, LKW kommen schneller durch das Straßennetz ans Ziel. Die Computer erzeugen eine „grüne Welle“.

### Auf den Bahnhöfen

großer Städte rechnet uns der Fahrkartenautomat rasch den Preis für die gewünschte Fahrkarte aus, und wenn wir die Bedienungsanleitung beachten und ausreichend Münzen zur Hand haben, können wir ihm eine gültige Fahrkarte entnehmen.

### Im Flugbüro

gibt ein Computer Auskunft, ob in der nächsten Maschine nach Moskau oder Budapest noch Plätze frei sind.

### In einigen Krankenhäusern

werden Frischoperierte und Schwerverletzte, die ohne Bewußtsein sind, sehr genau von Meßgeräten und einem Computer überwacht. Weichen zum Beispiel



der Herzschlag oder die Körpertemperatur von den Normalwerten ab, alarmiert der Computer die Schwester oder den Arzt.

### **Beim Fotografieren**

muß die Kamera richtig eingestellt werden, wenn klare Bilder entstehen sollen. In einem elektronischen Fotoapparat erledigt das automatisch ein winziger Computer für uns.

### **Beim Backen**

in einem elektronischen Küchenherd sorgt ein Computer dafür, daß der Kuchen in einer bestimmten Zeit gut geraten ist.

### **Als Hobby**

wird hier eine Computeranwendung gebaut, die pünktlich weckt, rechtzeitig das Radio und das Licht an- oder ausschaltet und die Fische im Aquarium füttert. Diese Anlage kann auch die Signale und Weichen der elektrischen Modelleisenbahn steuern, Spielpartner beim Knobeln und Rätseln sein und an Einkäufe und Geburtstage erinnern.

Sind das nicht erstaunliche Fähigkeiten, über die ein Computer verfügen kann?

Setzen wir unsere kurze Reise durch die Welt der Computeranwendungen fort!

## **Weitere Begegnungen mit dem Computer**

### **In modernen Industriebetrieben**

verrichten Maschinen schwere, gefährliche und eintönige Arbeiten. Man nennt sie Industrieroboter. Die exakten Bewegungen ihrer Arme und Greifer werden von eingebauten Computern berechnet und bestimmt.



### **Um Häuser und Industriehallen**

bauen zu können, müssen viele komplizierte Berechnungen vorgenommen und Zeichnungen angefertigt werden. Spezielle Computer übernehmen immer wiederkehrende Arbeiten.

### **In Büros und Zeitungsredaktionen**

werden Computer zum Schreiben von Briefen, Berichten und Artikeln benutzt. Mit ihrer Hilfe kann man den Text auch ändern, aufbewahren, in eine gewünschte Form bringen und drucken.

### **In Rechenzentren**

stehen große Computeranlagen. Sie führen für Betriebe und Verwaltungen umfangreiche Berechnungen aus, drucken Stapel von Unterlagen und leiten Ergebnisse schnell an Rechenzentren in anderen Städten weiter.

### **In Forschungslabors**

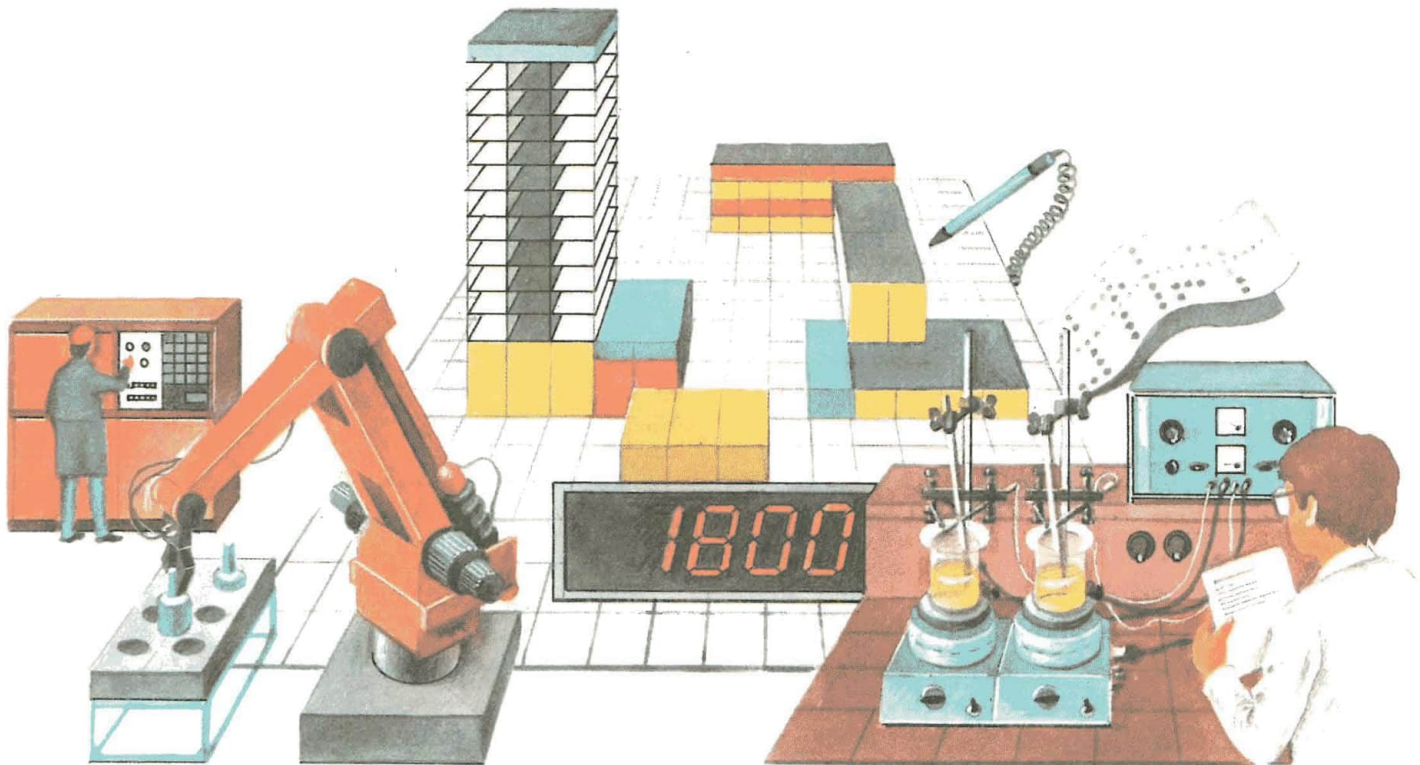
werden mit Hilfe von Computern komplizierte Versuchsreihen ausgewertet. Computer berechnen mathematische Formeln und sagen künftige Entwicklungen voraus, zum Beispiel, wie lange die Kohlevorkommen ausreichen oder wie sich die Verunreinigung der Luft verändern wird.

### **Die Wasserstände und die Sauberkeit des Wassers**

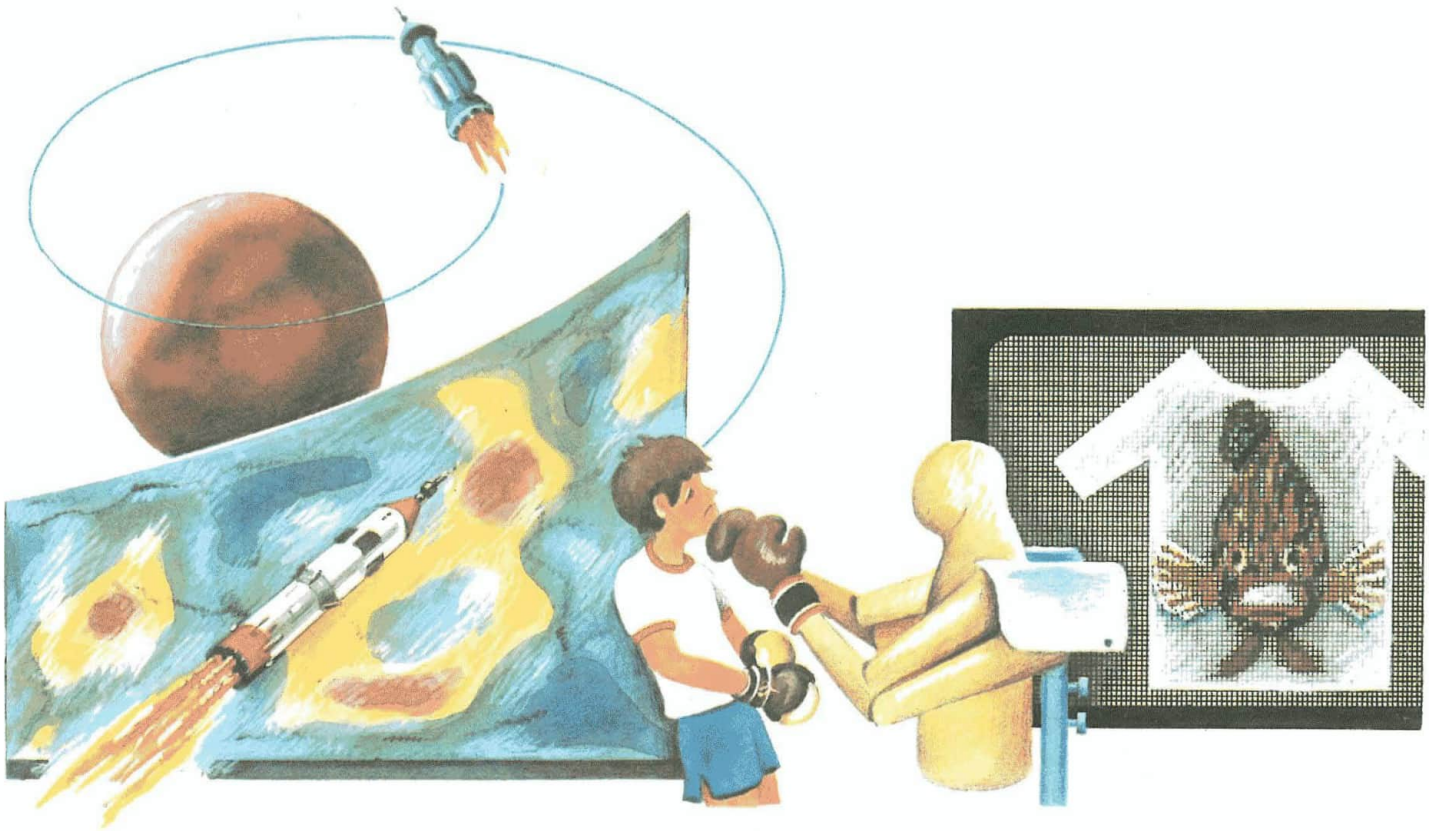
einiger wichtiger Flüsse und Talsperren lassen sich von vielen kleinen, untereinander verbundenen Computern überwachen. Gefährliche Situationen für Mensch und Umwelt können so frühzeitig erkannt und Schäden vermieden werden.

### **Der Start und die Flugbahn von Trägerraketen und Raumschiffen**

werden mit Computern exakt vorausberechnet. Com-







puter an Bord und im Flugleitzentrum gewährleisten, daß die Flugkörper trotz ihrer enorm hohen Geschwindigkeiten das Ziel genau erreichen. Sie reagieren sehr viel schneller als ein Mensch.

### **Neue schöne und mehrfarbige Strickmuster**

lassen sich mit einem Computer entwerfen. Die Ergebnisse werden Strickautomaten übermittelt, die das zuvor entworfene Muster von allein stricken.

### **Beim Boxtraining**

kann ein Roboter der unermüdliche Boxpartner sein. Er erteilt stundenlang unterschiedliche Schläge. Schlagart und -stärke werden aus der Ferne vom Trainer reguliert. Alle Bewegungen der Boxmaschine bestimmt ein eingebauter Computer.

### **Schüler, Studenten, Arbeiter und Fachleute**

lernen am und mit dem Rechner. Der Computer erklärt neuen Lehrstoff und stellt dazu Aufgaben. Er

prüft die Antworten und merkt sich auf Wunsch die Ergebnisse; auch Zensuren kann er erteilen.

Vielfältig werden Computer benutzt, um dem Menschen die Arbeit zu erleichtern. Auf immer mehr Gebieten werden sie eingesetzt. Computer multiplizieren die Fähigkeit des Menschen, die Welt zu erkennen, die Umwelt nach seinen Wünschen zu verändern und das Leben angenehmer und interessanter zu gestalten.

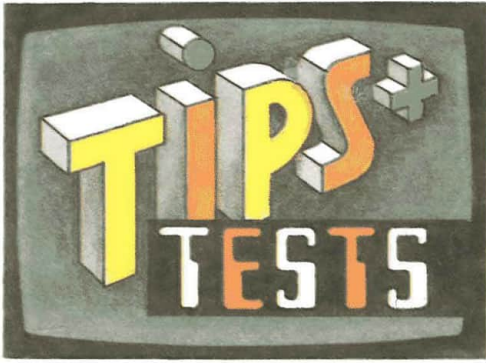
Computer vermögen verwirrend viel. Doch sie sind keine unverständlichen Wunderwerke, sondern Werkzeuge des Menschen. Wichtige Eigenschaften der Computer werden wir in diesem Buch kennenlernen.

### **Wir merken uns:**

Computer sind sehr vielseitig verwendbar.

Computer vervielfachen die Fähigkeit des Menschen, denken, rechnen und sich erinnern zu können. Sie werden ein immer wichtigeres Werkzeug.





## Tips und Tests für Pfiffige

- 2.1. Ermittle weitere Anwendungen des Computers! Sammle Zeitungsartikel über Computeranwendungen! Welche Arbeiten verrichtet der Computer?
- 2.2. Bilde Gruppen von Anwendungen des Computers (zum Beispiel Wissenschaft, Technik, Lernen)!
- 2.3. Was geschähe bei einer dir bekannten Anwendung, wenn es keine Computer mehr gäbe?
- 2.4. Kennst du Anwendungen des Computers, wo es auf Sekunden oder den Bruchteil einer Sekunde ankommt?
- 2.5. Nenne Anwendungen des Computers, wo es nur auf Stunden oder Tage ankommt!
- 2.6. Sind die Anwendungsmöglichkeiten des Computers zu zählen?
- 2.7. Der Bleistift ist ein Werkzeug zum Schreiben. Der Computer ist ein Werkzeug zum ...?

Die Lösungen zu ausgewählten Aufgaben aus „Tips und Tests für Pfiffige“ findest du im Teil A des Anhangs.

# 3. Was Computer alles können

---

Wir lernen wichtige Eigenschaften des Computers kennen und erfahren, was Daten sind.



## Erforschen, was er verarbeitet

In einen Geldwechselautomaten stecken wir Münzen, und selbsttätig gibt er uns das Wechselgeld richtig zurück. Ein Softeisautomat verarbeitet selbständig Milch, Zucker und andere Zutaten zu Softeis. Was aber verarbeitet der Automat Computer? Wir wollen diese Frage beantworten, indem wir ein Beispiel genauer untersuchen.

Wir rechnen  $12 \times 23$  mit dem Taschenrechner. Dazu müssen wir die Zahlen 12 und 23 eingeben, aber auch die Tasten für  $\times$  (mal) und  $=$  (ist gleich) drücken. Daraus errechnet der Taschenrechner das Ergebnis 276 und zeigt es uns an. Der Computer hat also die Zahlen 12 und 23 sowie die Zeichen  $\times$  und  $=$  in das Ergebnis 276 umgewandelt, er hat die Ziffern und anderen Zeichen verarbeitet.

Computer haben viel mehr Tasten als ein Taschenrechner, mit denen unterschiedliche Zeichen eingetippt, eingegeben werden können, zum Beispiel alle Buchstaben des Alphabets, Satzzeichen, wie Punkt, Komma, Strich, und andere Zeichen. Computer können also weit mehr, als nur mit Zahlen rechnen! Beim Computerspiel verarbeiten sie unsere Spielzüge zu ihrem Gegenzug.

Diese Daten kann man nicht anfassen wie einen Bleistift, ein Buch, ein Messer, einen Stuhl oder ein Fahrrad – doch man kann sie sehen beziehungsweise lesen, eingeben oder aufschreiben.

### Wir merken uns:

Zeichen aller Art werden Daten genannt.

Daten können in den Computer eingegeben, von ihm verarbeitet und ausgegeben werden.

Computer können Daten verarbeiten.



|   |  |
|---|--|
| A B c e<br>Tom<br>Liebe Susi<br>Buchstaben und Text           | Im Jahr 2000<br>3,50 Mark<br>8 km<br>Zahlen und Text |
| 012<br>1990<br>+3,50<br>Zahlen                                | $1+1=2$<br>$123 \times 77 = ?$<br>$x=a+b$<br>Formeln |
| . , ; :<br>! ? - ( )<br># \$ % & @<br>Satz- und Sonderzeichen | + - × ÷<br>■ ▢ ▣ ●<br>♠ ♥ ♦ ♪<br>Grafikzeichen       |



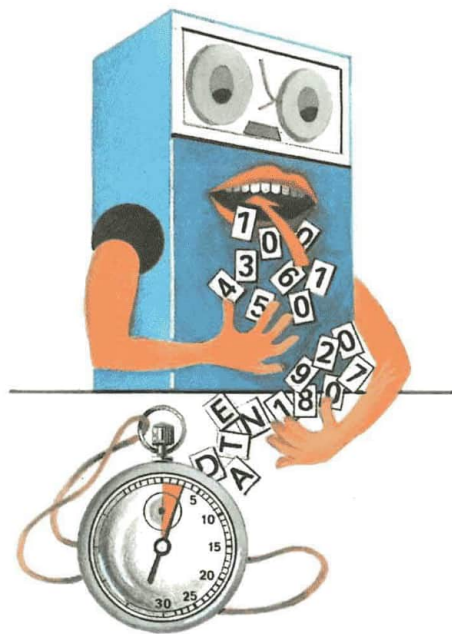
Computer zur automatischen Verarbeitung von Daten werden auch Datenverarbeitungsmaschinen genannt. Große Computer heißen Datenverarbeitungsanlagen.

Die Fähigkeit, Daten verarbeiten zu können, ist eine Grundlage für die vielseitige Verwendbarkeit von Computern.

## Wie schnell arbeiten Computer?

Die Antwort auf diese Frage hängt vom Typ des Computers ab. Es gibt heute Computer in Fingergröße, die in einer Sekunde 50 000 Zahlen addieren. Mancher moderne Computer, der groß ist wie ein Reisekoffer, addiert in einer Sekunde Millionen von Zahlen fehlerfrei.

Das sind erstaunliche Leistungen! Probieren wir einmal aus, wieviel Zeit wir brauchen, um mit Papier und Bleistift die Zahlen 12345 und 6789 zu addieren. 10 Sekunden? Wenn das so ist, benötigen wir 100 000 Sekunden, um 20 000 Zahlen zu addieren. Das sind



Eine Frau war in einem Monat mit 20 Arbeitstagen einen Tag krank. Der Computer berechnete ihr ein negatives, ein Minusgehalt, weil sie 81 Arbeitstage gefehlt habe.

Dieser ärgerliche Fehler war schwierig zu finden. Beim Eingeben der Krankheitstage war versehentlich statt der Taste 1 die darunterliegende Taste für den Buchstaben Q gedrückt worden. Wenn du Kapitel 12 gelesen hast, wirst du verstehen, wie ein solch falsches Ergebnis zustande kommt.

### Wir merken uns:

Computer führen exakt unsere Aufträge aus, die wir ihnen übertragen. Als Datenverarbeitungsmaschinen verstehen sie nichts von dem, was sie machen.

## Was uns am Computer gefällt

### Computer

- verarbeiten Daten sehr schnell und genau
- leisten viel für wenig Geld
- sind anspruchslos und arbeitswillig
- arbeiten tagelang ohne Pause, ohne Urlaub oder Ferien
- sind ruhig und unaufdringlich
- können sich mit anderen, weit entfernten Computern „unterhalten“ und Daten untereinander austauschen
- sind (fast) immer einsatzbereit.

Natürlich können Computer – wie alle technischen Geräte – kaputtgehen. Sie machen dann verrückte Sachen oder arbeiten überhaupt nicht mehr. Dann wird es Zeit, sie zu reparieren. Das ist bei modernen Computern jedoch selten erforderlich.

Insgesamt ist der Computer ein brauchbarer Diener und fleißiger Helfer beim Verarbeiten von Daten. Oft arbeitet er am besten, wenn er rund um die Uhr beschäftigt wird.



## Tips und Tests für Pfiffige

- 3.1. Was sind Daten – *a* Buchstaben, *b* Ziffern und Zahlen, *c* Satzzeichen und Sonderzeichen (wie , ; “ ? ! §), *d* Strichmarkierungen (zum Beispiel zur Warenauszeichnung)?

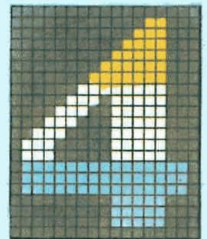


- 3.2. Was können Computer?  
3.3. Erkundige dich, wie schnell verschiedene Computer arbeiten können!  
3.4. Sammle Mitteilungen in der Presse über Fehler, die bei Computeranwendungen passiert sind!  
3.5. Was gefällt uns am Computer?  
3.6. Was ärgert uns manchmal am Computer?  
3.7. Nenne eine mögliche Ursache, warum ein Computer nicht arbeitet!  
3.8. Wieviel Ferien oder Urlaub benötigt ein Computer im Jahr?

## 4. Wir bauen ein Modell und benutzen es

---

Die Welt des Computers und seiner Anwendungen ist sehr vielfältig und dadurch verwirrend. Wir wollen in sie eindringen und schaffen uns deshalb einfache, übersichtliche Modelle. Dann benutzen wir sie, um die Computerwelt besser zu verstehen.





# Die Entdeckung von EvA

Wir beobachten jetzt die Anwendung eines Computers und versuchen, dafür ein einfaches Modell zu finden. Benutzt werden ein kinderleichtes Rechenbeispiel und der uns vertraute Taschenrechner – der Winzling unter den Computern. Tom wird uns helfen.

Das ist die Aufgabe:  $12345 + 6789 = ?$

Sie soll von Tom mit einem Taschenrechner gelöst werden. Für den Taschenrechner ist die Aufgabe eine Folge von Daten, die er verarbeiten soll.

Wir zeichnen den Vorgang „verarbeiten“ in ein Kästchen.

Das Eingeben der Aufgabe mit ihren Daten in den Taschenrechner stellen wir durch einen Pfeil dar.

Die Bildteile „Daten“ und „verarbeiten“ setzen wir nebeneinander, um zu zeigen, daß die Daten auch verarbeitet werden sollen.

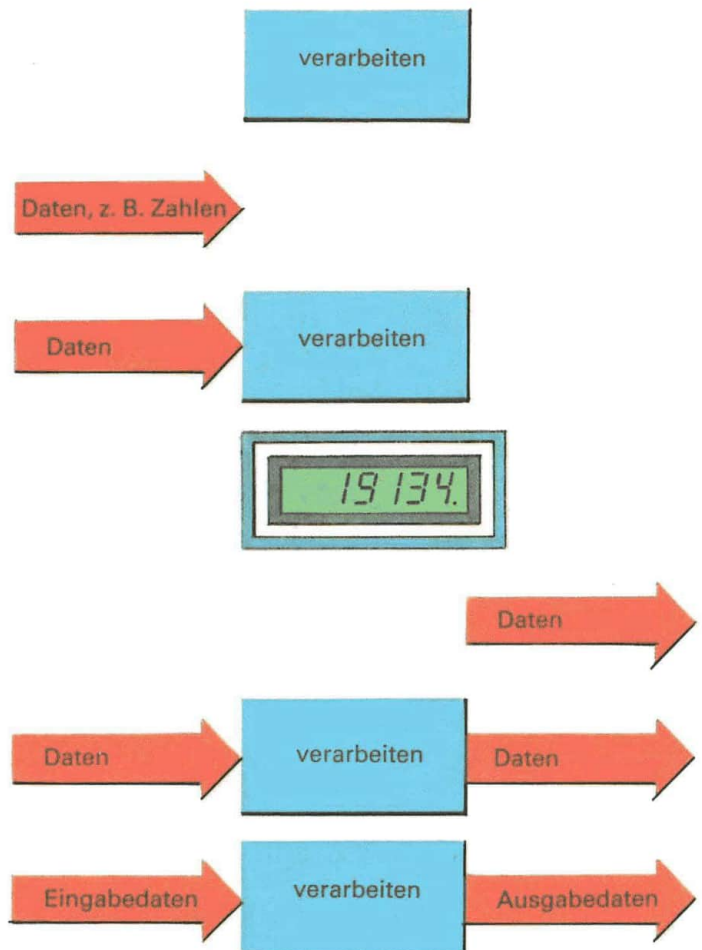
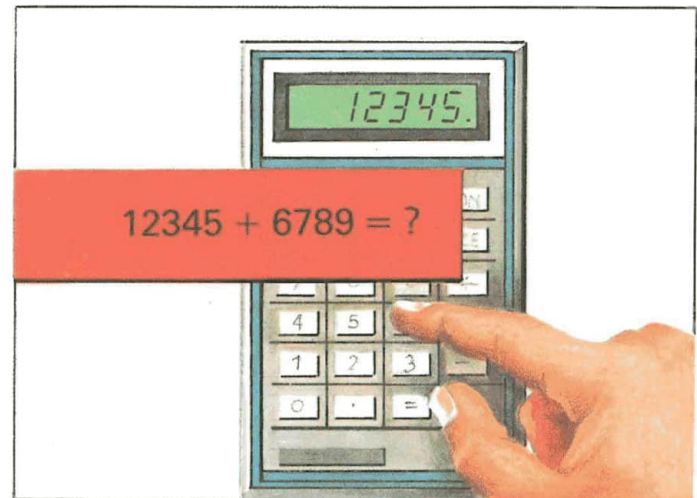
Hat Tom mit dem Taschenrechner die Aufgabe ausgerechnet, so erhalten wir das Ergebnis – eine Zahl.

Das Ergebnis sind wiederum Daten. Auch sie stellen wir in Form eines Datenpfeils dar.

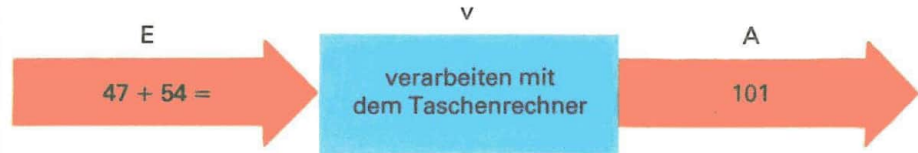
Diesen Datenpfeil setzen wir rechts an das Kästchen „verarbeiten“.

Es ist günstig, die Daten, die in das Kästchen „verarbeiten“ hineinführen, von denen zu unterscheiden, die aus ihm herausführen. Wir verwenden daher die Bezeichnungen „Eingabedaten“ und „Ausgabedaten“.

Damit haben wir uns ein kleines Modell geschaffen. Wir bezeichnen es kurz mit den Anfangsbuchstaben: EvA.



Die Aufgabe: Berechne das Ergebnis von  $47 + 54$ !



Noch genauer als zeitliche Schrittfolge dargestellt:

| betätigte Tasten |     | angezeigte Werte |
|------------------|-----|------------------|
| Einschalten      | OFF | 0.               |
|                  | 4   | 4.               |
|                  | 7   | 47.              |
|                  | +   | 47.              |
|                  | 5   | 5.               |
|                  | 4   | 54.              |
|                  | =   | 101.             |
| Ablesen          |     |                  |
| Ausschalten      | ON  |                  |

Zeitrichtung ↓

Das EvA-Modell,  
einmal ganz genau am Beispiel  
einer Addition mit dem  
Taschenrechner ausprobiert.



Diese Schrittfolge – nur kürzer geschrieben:

| Aufgabe     | eingegabene Werte | angezeigte Werte |
|-------------|-------------------|------------------|
| $47 + 54 =$ | 4 7 + 5 4 =       | 101.             |

Zeitablauf →

## Wir probieren EvA aus

Mit Modellen muß man etwas anfangen können, sonst taugen sie nichts. Tom zeigt uns kleine Beispiele für die Anwendung des EvA-Modells. Auch andere Vorgänge aus dem Alltag lassen sich mit dem EvA-Modell klar darstellen.

### Tom löst eine Fahrkarte nach Dresden

Er benutzt dazu einen Fahrkartenautomaten. Tom gibt Reiseziel und anderes sowie Münzen ein, erhält den Fahrpreis angezeigt und die Fahrkarte ausgedruckt.



Eingabedaten

Reiseziel Dresden, Münzen u. a.

verarbeiten

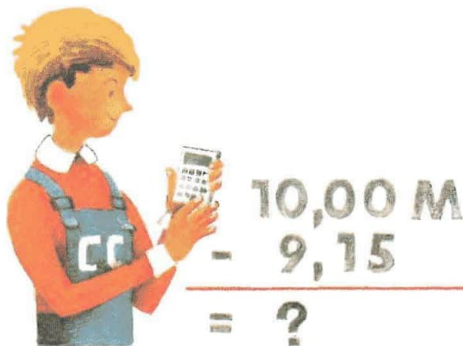
Fahrkartenautomat  
verarbeitet Eingabedaten

Ausgabedaten

Fahrpreis, Fahrkarte

### Tom überrechnet seine Geldausgaben

Er hat 10 Mark Taschengeld erhalten und davon 9,15 Mark ausgegeben. Wieviel verbleibt Tom?



Eingabedaten

$10,00 - 9,15 = ?$

verarbeiten

berechnen mit Taschenrechner

Ausgabedaten

Ergebnis = 0,85

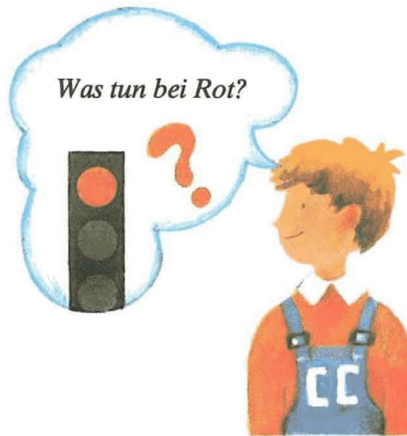


## Tom und das Ampelrot

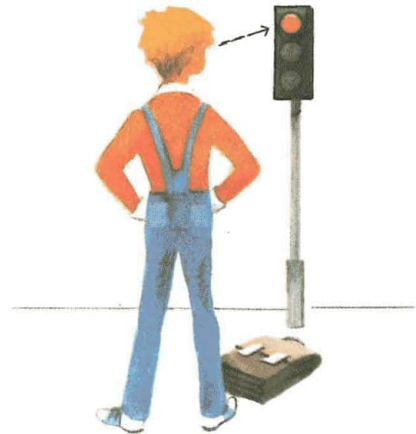
Tom nähert sich der Ampel und sieht Rot



Er überlegt und beschließt zu warten



Tom bleibt stehen



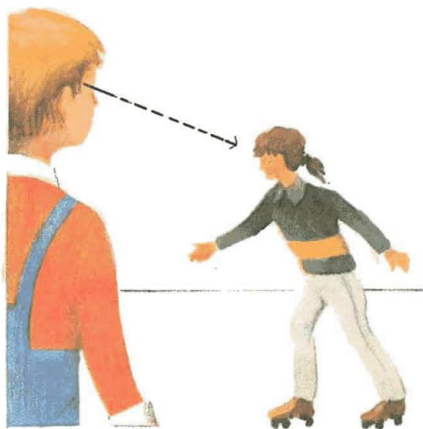
Tom kommt, Ampel ist rot

überlegen

Tom bleibt stehen

## Tom trifft ein hübsches Mädchen

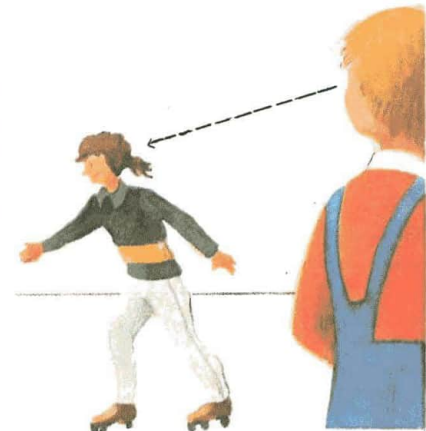
Tom sieht ein Mädchen



Er verarbeitet den ersten Eindruck und überlegt, ob er noch



Tom dreht sich nach dem Mädchen um



Tom

Mädchen

überlegen

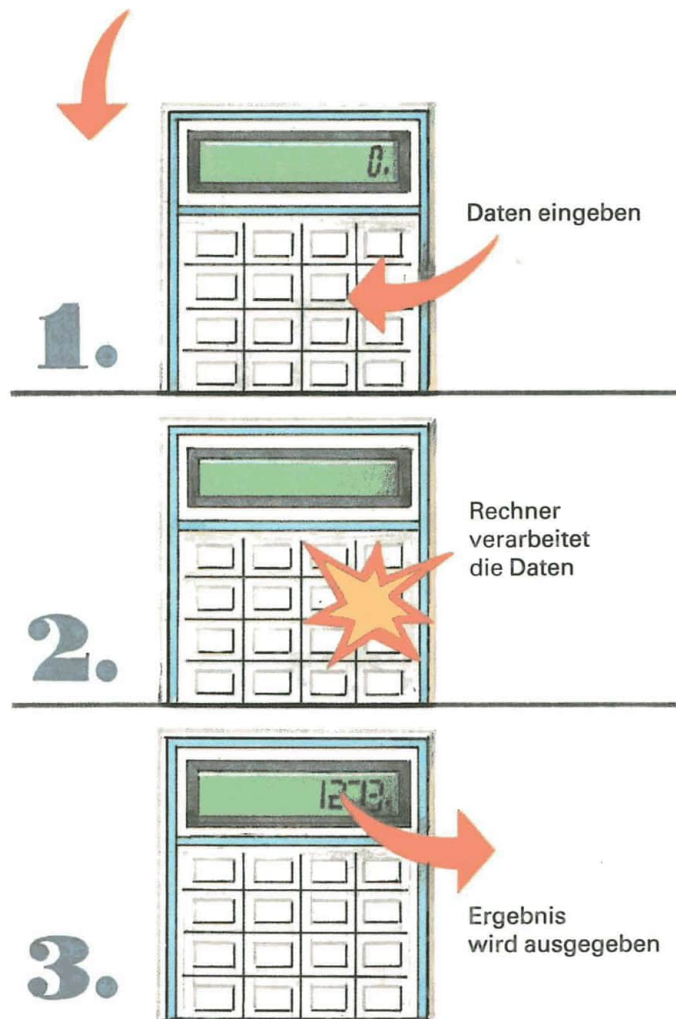
Tom dreht sich nach dem Mädchen um

## Wir entdecken Neues mit EvA

Mit dem Modell „Eingabedaten – verarbeiten – Ausgabedaten“ können wir noch etwas ausfindig machen, was uns tiefer in die Computerwelt eindringen läßt.

Betrachten wir das Kästchen „verarbeiten“ einmal genauer, zum Beispiel beim Addieren von zwei Zahlen mit dem Taschenrechner! Dieses Verarbeiten besteht aus drei Schritten. Sie sind rechts im Bild in Form dreier Kästchen dargestellt. Die Kästchen sind mit Strichpfeilen verbunden, welche die Reihenfolge der Schritte angeben.

### Die Arbeitsschritte beim Benutzen eines Taschenrechners



*Das ist ein Modell für die Benutzung des Computers. Die Kästchen geben Tätigkeiten an.*



Damit haben wir ein weiteres wichtiges Modell der Arbeitsweise eines Computers entdeckt.

Das Modell mit den Schritten „eingeben – verarbeiten – ausgeben“ unterscheidet sich wesentlich von dem EvA-Modell: Es werden *drei* Tätigkeiten angegeben. Beim EvA-Modell ist es nur *eine* Tätigkeit. Das neue Modell kennt keine Pfeile für Daten, sondern nur Pfeile für die Reihenfolge der *Tätigkeiten*.

Unsere zuvor entdeckten Modelle berücksichtigen eine wichtige Eigenschaft des Computers noch nicht. Wenn wir beispielsweise Daten in den Taschenrechner eingeben, so verschwinden diese nicht, sobald wir die Finger von den Tasten nehmen. Der Taschenrechner „merkt“ sie sich, er speichert sie. Das gilt auch für die Ergebnisse, welche er anzeigt – bis wir neue Daten eingeben oder den Rechner abschalten. Die Fähigkeit zum Speichern müssen wir unserem Modell hinzufügen.

Wir haben nun die vier Tätigkeiten eines Computers, die er bei einer Anwendung an Daten ausführt, kennengelernt:

- Daten einlesen, die ihm von außen eingegeben werden
- Daten verarbeiten
- Daten speichern
- Daten ausgeben.

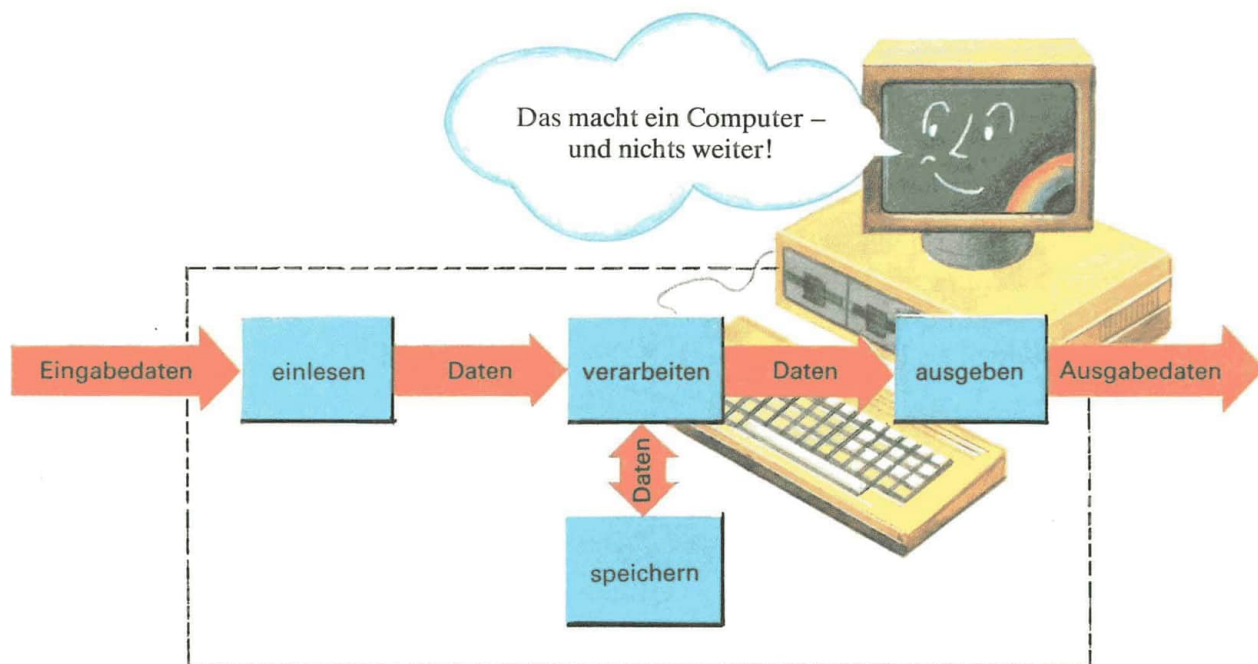
#### Wir merken uns:

Computer können Daten einlesen, verarbeiten, speichern und ausgeben (kurz: *evsa*).

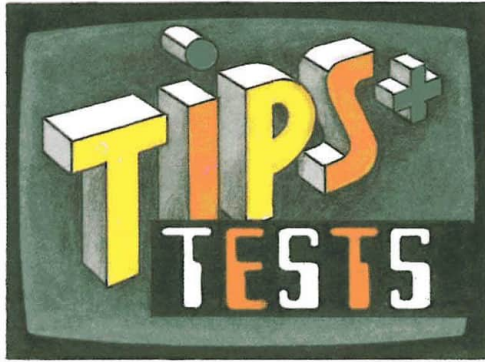
Stellen wir die Tätigkeiten in Form von Kästchen dar und die Daten als dicke Pfeile, so ergibt sich das untenstehende Bild.

Am Kasten „speichern“ sehen wir einen Doppelpfeil. Beim Speichern soll der Computer sich nicht nur etwas merken, sondern auch später wieder daran erinnern. Das entspricht dem hin- und wegführenden Datenpfeil.

Wir prägen uns das letzte Bild ein. Es ist ein entscheidender Schlüssel zur vielgesichtigen Welt des Computers und seiner Anwendung.







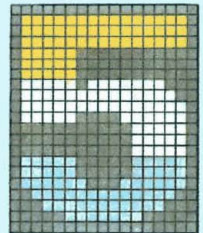
## Tips und Tests für Pfiffige

- 4.1. Zeichne das EvA-Modell für die Arbeit eines Computers und erkläre es an einem Beispiel!
- 4.2. Erkläre Alltagsvorgänge mit dem EvA-Modell!
- 4.3. Was wird im EvA-Modell als Kästchen und was als Pfeil dargestellt?
- 4.4. Woran erkennt man, daß der Taschenrechner Daten speichert?
- 4.5. Welche der angegebenen Tätigkeiten führt ein Computer aus – *a* fragen, *b* einlesen, *c* verarbeiten, *d* speichern, *e* ausgeben, *f* antworten?
- 4.6. Zeichne die vier Tätigkeiten eines Computers und dazugehörige Daten!

## 5. Besichtigung einer Ausstellung von Computern

---

Wir wollen uns jetzt gemeinsam mit Tom wichtige Computertypen anschauen und Geräte für verschiedene Aufgaben kennenlernen.



## Klein, größer, am größten...

Computer gibt es in vielfältiger Gestalt. Computerzwerge, wie Taschenrechner, kennen wir schon. Betrachten wir nun vier größere Vertreter der Computerfamilie!

Mit einem Heim-, Klein- oder Lerncomputer kann man Spiele machen, schwierigste mathematische Aufgaben lösen, Aufsätze und Briefe auf den Bildschirm

Das ist ein Bildschirm, wie wir ihn vom Fernseher her kennen. Er zeigt uns Daten und Ergebnisse an.

*Das ist ein Klein-, Heim- oder Lerncomputer. Er besteht aus einem gewöhnlichen Fernseher, einem Kassettenrecorder, einer Tastatur (mit eigentlichem Computer) und Verbindungskabeln.*



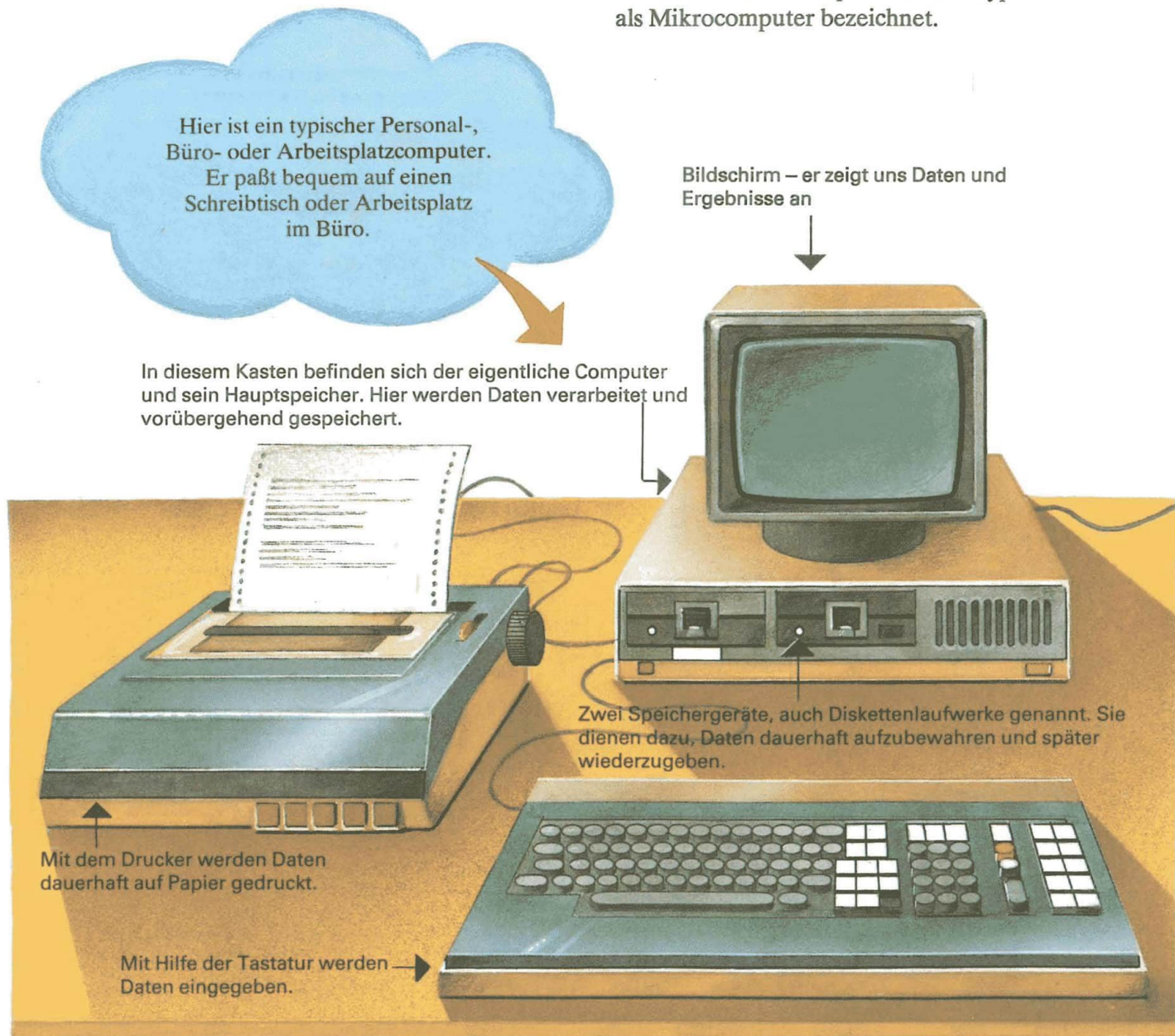


schreiben und speichern, Neues lernen und Wissen testen. Auch Modelleisenbahnen oder Kleinstroboter lassen sich mit seiner Hilfe steuern.

Er ist der billigste vollwertige Computertyp.

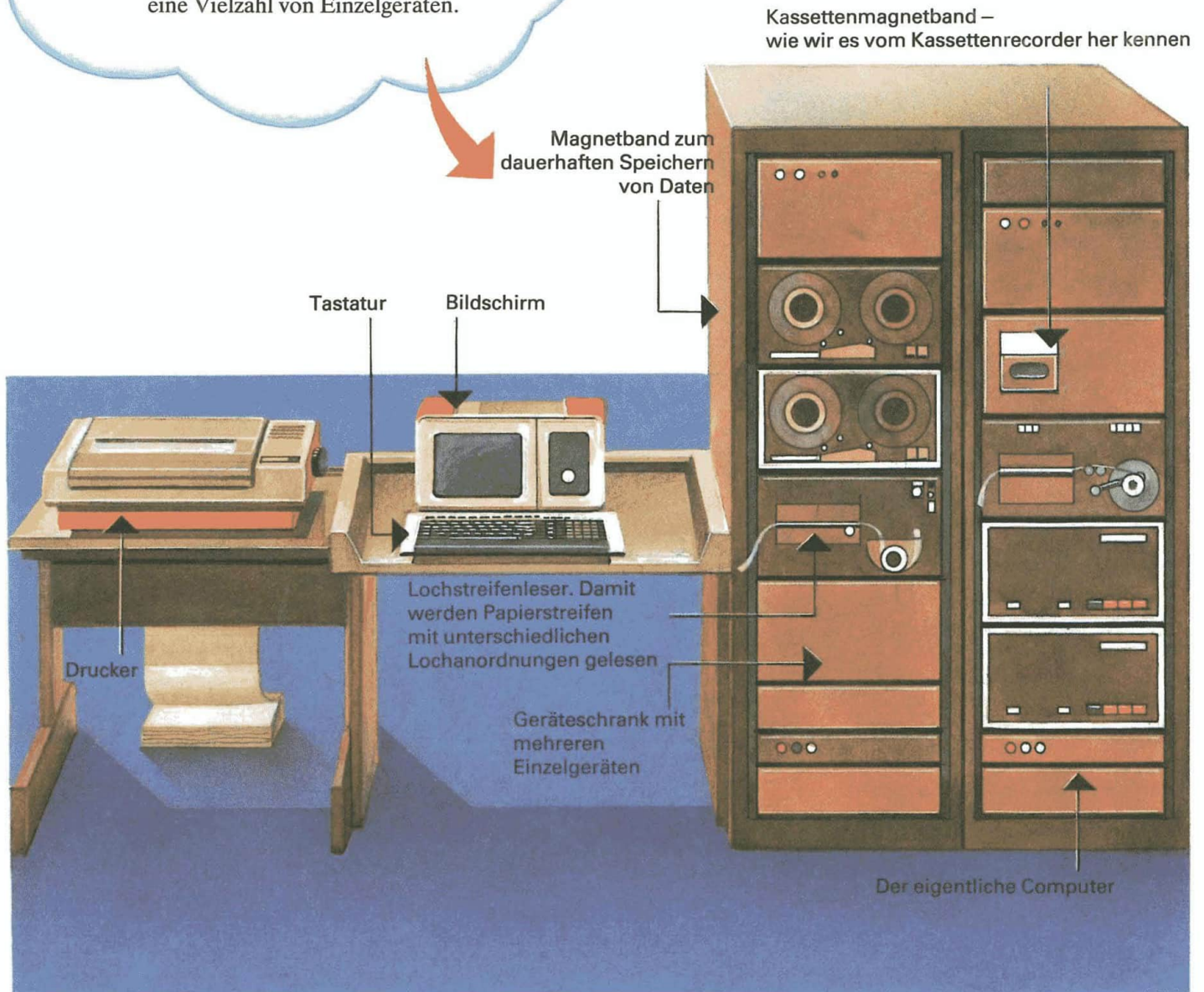
Mit einem Personalcomputer, auch Büro- oder Arbeitsplatzcomputer genannt, ist manche der im Kapitel 2 genannten Computeranwendungen möglich, sofern die Aufgaben nicht zu „groß“ sind. Später, im Kapitel 9, werden einzelne Geräte genauer betrachtet.

Personalcomputer sind leistungsfähiger als Heim-, Klein- oder Lerncomputer. Beide Typen werden auch als Mikrocomputer bezeichnet.



Ein Minicomputer braucht viel mehr Platz als ein Personalcomputer, und er leistet auch mehr. Fast alle Aufgaben von Kapitel 2 vermag er für uns zu erledigen. Ein Minicomputer ist so leistungsstark, daß mehrere Kollegen ihn mit Arbeit „füttern“ müssen.

So sieht ein  
Minicomputer aus.  
Zu ihm gehört  
eine Vielzahl von Einzelgeräten.





Dieser Minicomputer ist zusätzlich mit Geräten zur Ein- und Ausgabe technischer Zeichnungen ausgerüstet.

*Dieser Minicomputer  
ist besonders interessant  
für alle, die viel  
zeichnen müssen.*

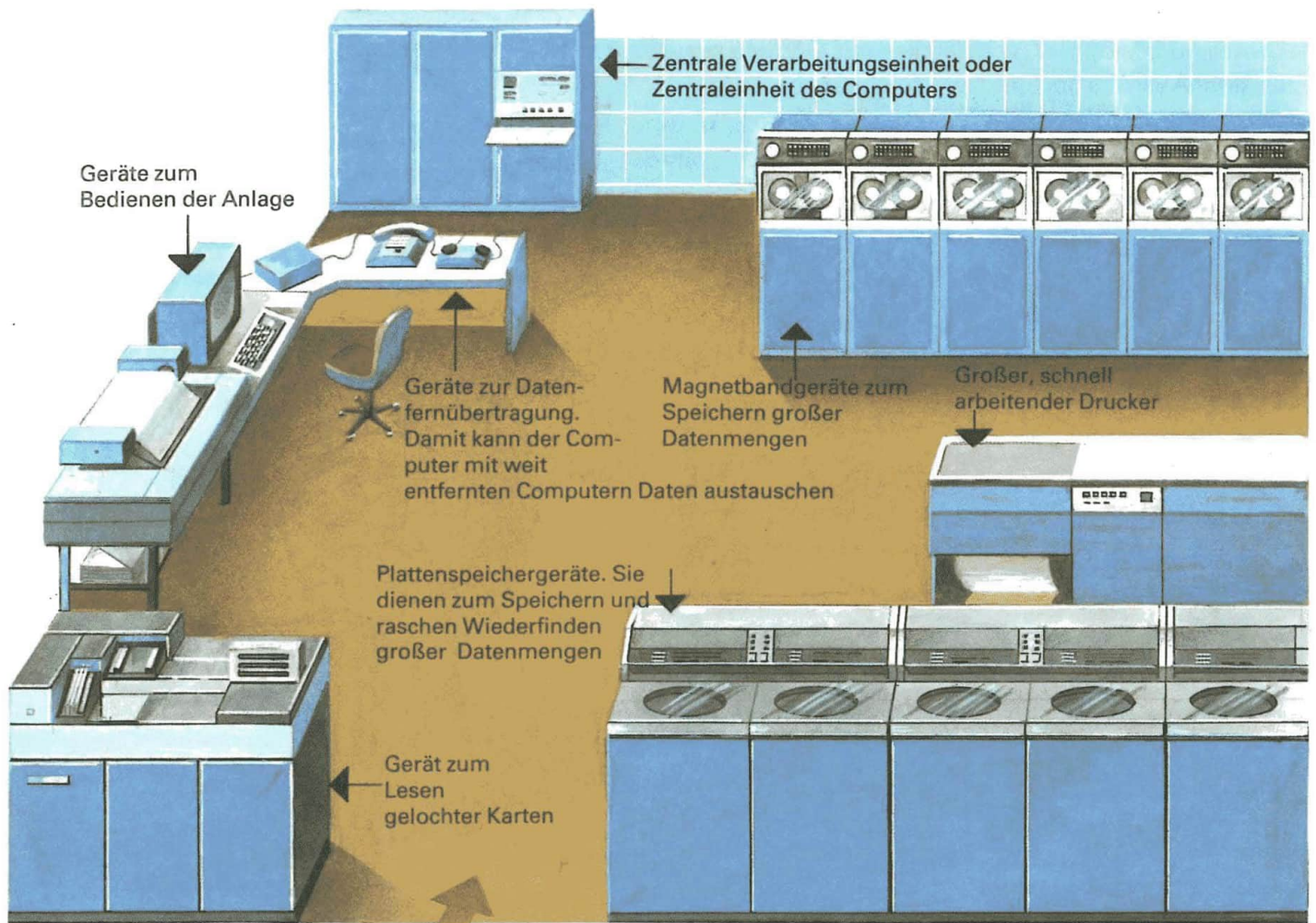


Den Bildschirm  
kennen wir schon

Mit diesem Gerät lassen sich  
Punkte und Zeichnungen in  
den Computer eingeben. Es  
heißt Digitalisiergerät oder  
Digitizer (sprich: digiteiser)

Automatisches Zeichengerät,  
Plotter genannt





Ein großes Zimmer oder einen ganzen Saal füllt dieser große Computer, auch Großrechner oder Datenverarbeitungsanlage genannt. Er ist der leistungsfähigste und teuerste Computertyp, und er arbeitet einwandfrei nur in klimatisierten Räumen. Die Verbindungskabel zwischen den Geräten sind im Fußboden unsichtbar verlegt.

Ein solcher Großcomputer bewältigt selbst die umfangreichsten und schwierigsten Aufgaben. Er wird oft als Datenverarbeitungsanlage bezeichnet. Wir finden ihn in Datenverarbeitungs- oder Rechenzentren. Viele Fachleute nutzen ihn. Manche von ihnen sind über große Entfernungen mit der Anlage verbunden.

Einige Geräte, die wir schon beim Mikro- und beim Minicomputer kennengelernt haben, finden wir hier wieder, teilweise größer und leistungsfähiger. Vor zwanzig Jahren gab es ausschließlich diesen aufwendigen Computertyp, nur arbeitete er viel langsamer.

Die vorgestellten Vertreter der Computerfamilie gibt es in sehr vielen Ausführungen. Alle enthalten den eigentlichen Computer, der Daten verarbeitet und speichert (die Verarbeitungs- oder Zentraleinheit, auch als Prozessor bezeichnet) sowie Zusatzgeräte zum Eingeben, Speichern, Ausgeben und Übertragen von Daten (Peripherie genannt). Die Verarbeitungseinheit ist mit den Peripheriegeräten stets über Kabel verbunden. Welche Geräte benötigt werden, hängt von den Aufgaben ab, die der Computer lösen soll.

#### **Wir merken uns:**

Man unterscheidet nach Größe und Leistungsfähigkeit Mikrocomputer, Minicomputer, Großcomputer und Einbaupcomputer.

Mikrocomputer führen meist mehrere zehntausend Additionen in der Sekunde aus, Großcomputer viele Millionen. Minicomputer liegen hinsichtlich der Leistungsfähigkeit zwischen diesen beiden Computertypen. Die Abgrenzung ist nicht eindeutig. Neue Computertypen arbeiten schneller als ihre Vorgänger.

Moderne Mikrocomputer leisten heute so viel wie früher Minicomputer, moderne Minicomputer erreichen die Leistung von älteren Großcomputern.



In diesem unscheinbaren Blechkasten mit einem Lämpchen verbirgt sich ein leistungsfähiger moderner Mikrocomputer. Er wird in Maschinen eingeschoben oder versteckt eingebaut und deshalb oft Einbaupcomputer genannt. Immer mehr moderne Maschinen und Geräte enthalten solche „eingebetteten“ Computer.



## Speziell und originell

Jetzt sehen wir uns noch Beispiele für Geräte mit eingebauten Computern an. Die Computer sind äußerlich nicht mehr sichtbar, sie sind eingebettet. Ihr Vorhandensein nehmen wir nur an der geschickten Arbeitsweise der Geräte wahr.

Der Einbaucomputer dieser großen Schreibmaschine speichert die eingegebenen Texte, hilft beim Ändern, druckt mehrfach aus und vermag noch vieles andere.

Dieser Lerncomputer in Kleinstformat stellt kleine Rechenaufgaben und prüft selbständig, ob das eingegebene Ergebnis richtig ist.

Geräte mit eingebautem  
Computer

Eine elektronische Schreibmaschine



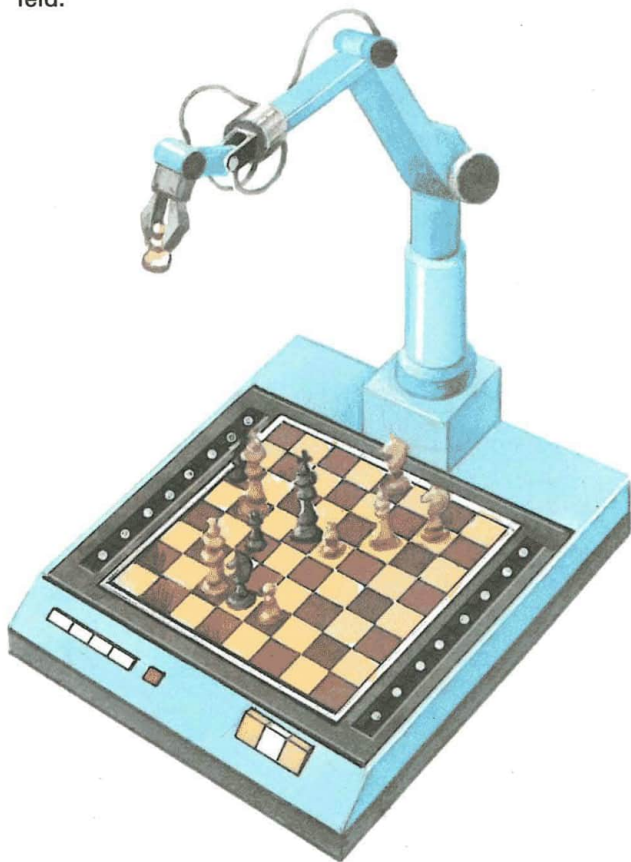
Ein Lerncomputer





Hier ist ein origineller Schachcomputer zu sehen, ausgerüstet mit einem einfachen Gelenkarmroboter. Dessen Greifer setzt die Figuren und nimmt sie auch vom Feld. Die Armbewegungen werden – ebenso wie der gesamte Spielverlauf – von einem eingebauten Computer gesteuert. Wenn der Computer das Spiel verloren hat, fängt er an zu zittern und zu jammern. (Der Bau eines solchen Roboters ist eine anspruchsvolle und lehrreiche Knobel- und Bastelaufgabe für Arbeitsgemeinschaften und Computer-Clubs. Man kann natürlich auch einfachere Gelenkarmformen für den Roboter verwenden.)

Ein Schachcomputer mit Roboterbedienung. Der Roboterarm mit Greifer bewegt die Figuren auf dem Spielfeld.

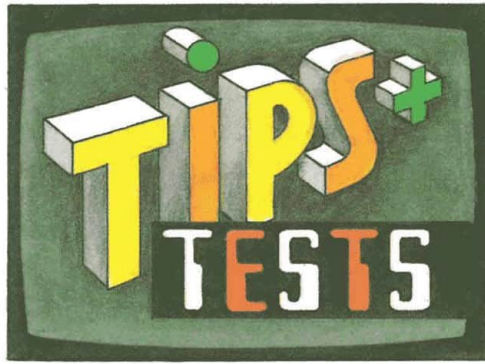


Bei den letzten drei Geräten sind – wie beim Spielautomaten am Anfang des Buches – Computer fest eingebaut und erfüllen einen bestimmten Zweck. Diese Geräte sind daher nicht so universell verwendbar wie die zuvor gezeigten Mikro-, Mini- oder Großcomputer. Mit einem Schachcomputer lassen sich keine Rechenaufgaben lösen oder Briefe schreiben. In Spezialcomputern sind Wissen und Können schon nutzbar eingebaut. Universalcomputer müssen erst programmiert werden, bevor sie nützliche Aufgaben für uns erledigen können. Wie das gemacht wird, erfährst du im Kapitel 10.

#### Wir merken uns:

Nach ihren Verwendungsmöglichkeiten unterscheiden wir Universalcomputer und Spezialcomputer.

Einen Universal- und Spezialcomputer von unerreichbarer Leistungsfähigkeit haben wir bisher vergessen: unser Gehirn. Es verfügt über Fähigkeiten, Daten zu verarbeiten und zu speichern, und vollbringt bewundernswerte schöpferische Leistungen, die keiner der modernsten Computer erreicht.



## Tips und Tests für Pfiffige

- 5.1. Welche Vertreter der Computerfamilie kennst du?
- 5.2. Werden die Computertypen nach  $a$  Größe,  $b$  Leistung,  $c$  Gewicht unterschieden?
- 5.3. Nenne Geräte, die zu jedem Computertyp gehören! Unterscheide nach Geräten für die Eingabe von Daten, für die Verarbeitung, Speicherung und die Ausgabe von Daten!
- 5.4. Gibt es sehr viele Benutzer für  $a$  einen Personalcomputer oder für  $b$  eine Datenverarbeitungsanlage in einem Rechenzentrum?
- 5.5. Ist ein Spielcomputer  $a$  ein Universalcomputer oder  $b$  ein Spezialcomputer?
- 5.6. Ist ein Personalcomputer  $a$  ein Universalcomputer oder  $b$  ein Spezialcomputer?
- 5.7. Wie entwickelt sich die Computertechnik weiter? Sind neue Geräte im allgemeinen  $a$  so leistungsfähig wie ältere oder  $b$  leistungsfähiger?
- 5.8. Sammle Bilder von Computern aus Zeitschriften und Prospekten! Ordne sie nach Computertypen!

## 6. Drei Schlüssel zur Computerwelt

---

Auf den ersten Blick ist die Welt des Computers erstaunlich und geheimnisvoll. Doch ihre wichtigsten Geheimnisse sind leicht zu verstehen. Damit haben wir die Schlüssel zur Computerwelt.





## Ein uralter Traum des Menschen – heute Wirklichkeit

Der Wunsch nach einer Maschine, die selbständig, ohne Hilfe des Menschen für uns arbeitet, die Hoffnung auf einen dienstbaren Geist, der wie ein Heinzelmännchen alle unsere Aufträge erfüllt, sind uralte. Technisch verwirklicht haben Menschen diese Sehnsüchte in Automaten.

### Wir merken uns:

Automaten sind Maschinen, die selbständig für uns arbeiten.

Automaten benötigen die Hilfe des Menschen nur zum Einschalten, Einstellen und zum Abschalten.

Heute trägt fast jeder einen kleinen Automaten bei sich – die Uhr. Sie arbeitet selbständig und zeigt uns die Uhrzeit an. Andere Beispiele aus dem Alltag sind der Waschautomat, der selbständig unsere Wäsche wäscht, der Aufzug in Hochhäusern, der uns selbständig in die gewünschte Etage befördert.

Das Grimmsche Märchen „Der süße Brei“ handelt

von einem Töpfchen, das selbständig süßen Brei kocht, also von einem Breikochautomaten. Die Mutter kennt sich nicht so gut wie die Tochter mit dem Automaten aus. Sie weiß ihn zwar „einzuschalten“, aber das Zauberwort zum „Abschalten“ hat sie vergessen. Da gibt es natürlich viel zuviel Brei und mächtigen Ärger. Ärger gibt es mit allen Automaten, die man nicht richtig zu bedienen weiß.

Die Idee einer selbständig arbeitenden Maschine, des Automaten, ist auch im Computer verwirklicht. Das ist ein erster wichtiger Schlüssel zum Verständnis des Computers.

## Am Steuer

Ein anderer Schlüssel zur Computerwelt ist das *Steuern* beziehungsweise die *Steuerung*. Vorgänge, die von Menschen gesteuert werden, begegnen uns im Alltag überall. Der Kraftfahrer steuert das Auto, der Pilot das Flugzeug, der Steuermann das Schiff zum Reiseziel.

### Wir merken uns:

Steuern ist das zielgerichtete Beeinflussen von Vorgängen.

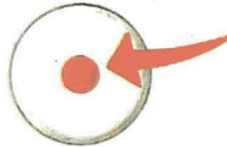
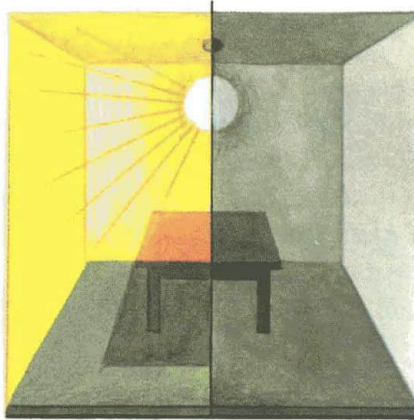
Arbeitet eine Steuerung selbständig, ohne unsere Mitwirkung, so spricht man von einer automatischen Steuerung. Im Alltag und in den Betrieben lassen sich auch dafür Beispiele entdecken, zum Beispiel die automatische Steuerung von Geräten, Maschinen, Robotern, Fertigungsstraßen (siehe auch Beispiele in Kapitel 2).

Computer können nicht nur Vorgänge automatisch steuern, sondern auch die Abläufe im Innern der Computer selbst werden automatisch gesteuert. Genauereres darüber erfährst du im Kapitel 13.

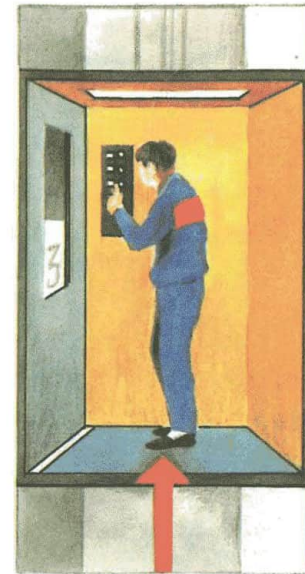
Und wer befindet sich unsichtbar am Steuer eines Computers? Dieses wichtige Geheimnis wird jetzt gelüftet.



Licht an- oder ausschalten (Helligkeit beeinflussen)

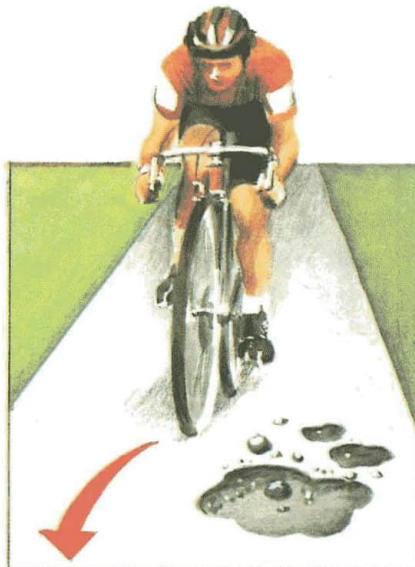


Gewünschte Etage im Aufzug einstellen (Bewegung des Aufzugs beeinflussen)



*Hier wird gesteuert,  
also zielgerichtet  
beeinflusst.*

Den Lenker bewegen (Fahrtrichtung beeinflussen)



Das Boot am Schwimmer vorbei-  
steuern (Fahrtrichtung beeinflussen)



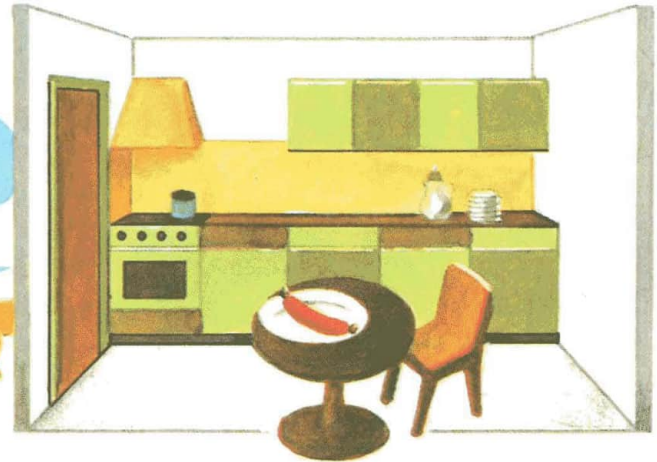


Ein Algorithmus sagt,  
wie ein Problem gelöst  
wird.

Rex hat ein Problem



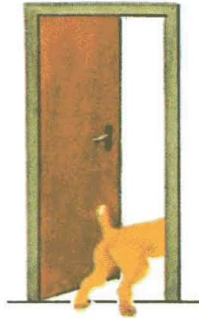
Ich will diese Wurst.  
Doch die Tür ist zu.



Er löst es mit einem Algorithmus in 2 Schritten



1. Schritt  
Klinke nach unten drücken



2. Schritt  
Tür aufdrücken



Jetzt sitzt Rex glücklich unter dem Tisch,  
weil er das Problem gelöst hat.



## Immer Schritt für Schritt

Stell dir folgende Situation vor: Dein Hündchen Rex sitzt vor der Küchentür und schnuppert. Durch das Schlüsselloch zieht der Duft von Knackwurst, die auf dem Küchentisch liegt. Doch die Tür ist zugeklinkt. Rex hat ein Problem: Wie komme ich durch die Tür in die Küche zur Wurst?

Wir kennen natürlich seit unseren ersten Lebensjahren die Lösung dieses Problems, auch wenn wir nicht mehr darüber nachdenken. Sie besteht aus zwei Tätigkeiten, die nacheinander auszuführen sind:

1. Türklinke betätigen.
2. Tür aufdrücken oder aufziehen (je nach Standort).

Eine solche Folge von Tätigkeiten heißt *Algorithmus*. Hier besteht der Algorithmus aus nur zwei aufeinanderfolgenden Schritten. Es kann sein, daß Rex diesen Algorithmus beherrscht und anwendet und ... schwupp, weg ist die Wurst und Rex mit ihr.

Wer schreiben gelernt hat, beherrscht auch diesen Algorithmus mit drei Schritten.

| Aufgabe | Schreibe das Wort REX mit seinen 3 Buchstaben!   |
|---------|--|
| Lösung  | <ol style="list-style-type: none"><li>1. Schreibe R!</li><li>2. Schreibe (rechts daneben) E!</li><li>3. Schreibe (rechts daneben) X!</li></ol> |

Diesen Algorithmus lernt jedes Kind im 1. Schuljahr.

Wenn du jetzt deiner Freundin telefonisch berichten willst, was Rex ausgefressen hat, so begegnet dir in der Telefonzelle wieder ein Algorithmus. Er schreibt vor, wie der Münzfernsprecher bedient werden muß:

1. Handapparat abnehmen.
2. Freizeichen abwarten.
3. 20-Pfennig-Stück einwerfen und Stillstand der Münze abwarten.
4. Rufnummer wählen.

Dieser Algorithmus besteht aus vier Schritten, die nacheinander auszuführen sind. Sie lösen das Problem, wie man mit dem Münzfernsprecher jemanden anrufen kann.

Eine bekannte Regel, um ein Gesicht zu zeichnen, lautet: Punkt, Punkt, Komma, Strich, fertig ist das Mondgesicht.

Das ist ebenfalls ein Algorithmus, nur stark verkürzt. Ausführlicher müßten seine fünf Schritte so lauten:

1. Male einen Punkt.
2. Male in gleicher Höhe daneben noch einen Punkt.
3. Setze darunter in die Mitte ein Komma.
4. Setze darunter einen waagerechten Strich.
5. Zeichne um das Ganze einen Kreis.

Diese fünf Schritte lösen das Problem, ein Mondgesicht zu zeichnen. So ähnlich sehen auch die Arbeitsvorschriften für Zeichenautomaten (Plotter) aus, die beliebige Bilder zeichnen können.

Algorithmen begegnen uns im Alltag in vielfältiger Weise. Wir erkennen sie oft nicht auf den ersten Blick (wie den Algorithmus für das Türöffnen) und entdecken sie erst, wenn wir die Arbeitsvorgänge genauer beobachten und darüber nachdenken.

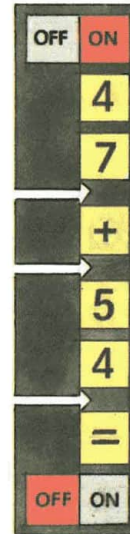
Die meisten Tätigkeiten, Denk- und Arbeitsvorgänge, die zu einem Ergebnis führen, können durch aufeinanderfolgende Handlungsschritte – also durch Algorithmen – beschrieben werden, natürlich auch der Lösungsweg mathematischer Aufgaben.

Schrittfolge beim  
Addieren der Größen A und B  
mit dem Taschenrechner.  
Die Aufgabe lautet:  
 $47 + 54 = ?$



## 1. Lösungsvorschriften genau als Schritt- und Tastenfolge

Einschalten



Blickkontrolle

Blickkontrolle

Blickkontrolle

Ergebnis  
ablesen

Ausschalten

Durch Blickkontrolle  
überprüfen wir, ob  
das Eingegebene  
so ist, wie wir es  
wünschen, denn es  
wird danach nicht  
mehr angezeigt.  
Haben wir uns vertippt,  
wird das Ergebnis  
falsch. Alle Computer  
haben Tasten  
zur Korrektur  
falscher Eingaben

Nehmen wir zum Beispiel die einfache mathematische Aufgabe, drei beliebige Zahlengrößen – wir bezeichnen sie als A, B und C – mit dem Taschenrechner zu addieren. Die Lösung dieser Aufgabe oder dieses Problems besteht in der Schrittfolge:

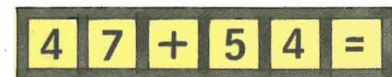
1. Taschenrechner einschalten.
2. Zahl A eintippen.
3. Taste + drücken.
4. Zahl B eintippen.
5. Taste + drücken.
6. Zahl C eintippen.
7. Taste = drücken.
8. Ergebnis ablesen.
9. Taschenrechner ausschalten.

Und hier ein Beispiel für nur zwei Zahlen:

## 2. ... vereinfacht nur als Tastenfolge untereinander



nebeneinander



Auch das sind alles Ablauf-  
pläne/Algorithmen!



## Erstaunlich – aber keine Hexerei

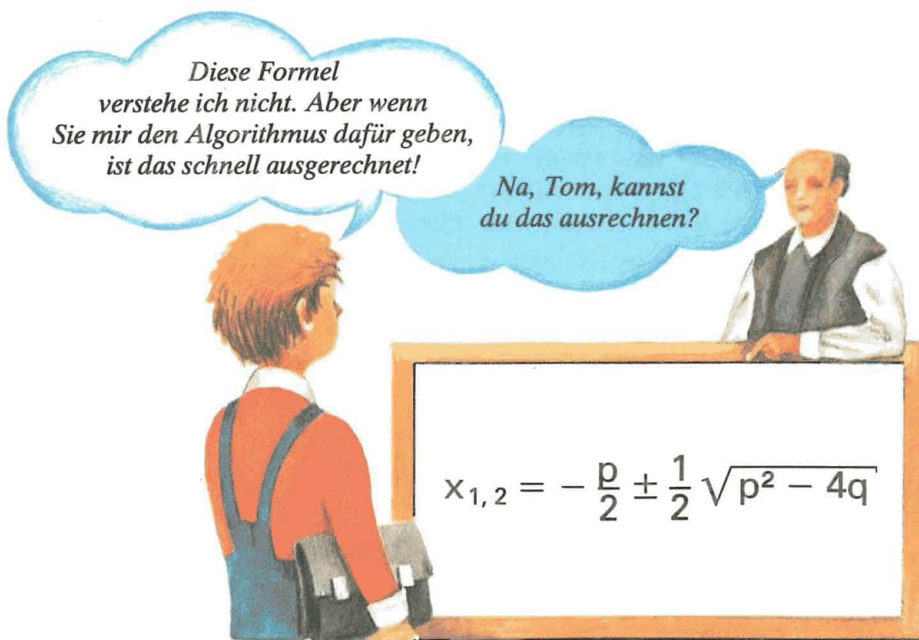
Der Algorithmus ist ein weiterer Schlüssel jeder Anwendung von Automaten und Robotern. Algorithmen beschreiben den Lösungsweg von Aufgaben oder Problemen und erzielen ein bestimmtes (gewünschtes) Ergebnis. Sie haben einen Anfang und erreichen ein Ende. Sie lösen Aufgaben, indem sie eine genaue Aufeinanderfolge von kleinen, nachvollziehbaren Handlungsschritten vorschreiben. Auch die *Reihenfolge* der Schritte ist wichtig. Eine unverschlossene Tür mit eingeklinktem Schloß kann im allgemeinen nicht geöffnet werden, indem erst Schritt 2 (dagegendrücken) und dann Schritt 1 (Türklinke betätigen) ausgeführt wird. Und wir können mit einem Taschenrechner nicht rechnen, wenn er nicht eingeschaltet wurde, und so weiter.

Eine Eigenschaft der Algorithmen macht sie uns so sehr wichtig: Sie sind Arbeitsvorschriften, um auf gleiche Weise Aufgaben zu lösen, die immer wiederkehren – zum Beispiel die Tür öffnen, mit dem Münzfernsprecher anrufen, Zahlen addieren und vieles andere.

### Wir merken uns:

Ein Algorithmus ist eine Arbeitsvorschrift. Sie beschreibt, wie bestimmte Aufgaben gelöst werden und wir so zum Ergebnis kommen. Ein Algorithmus schreibt genau vor, welche Einzelschritte dazu in welcher Reihenfolge auszuführen sind.

Algorithmen haben noch eine erstaunliche Eigenschaft: Wer einen Algorithmus anwendet, braucht von dem Problem gar nichts zu verstehen; nach einer begrenzten Anzahl von Einzelschritten gelangt er dennoch zur Lösung. Einen Münzfernsprecher vermag jemand zu bedienen, der keinerlei Kenntnisse über die Arbeitsweise des Gerätes hat. Mit dem Algorithmus „Tür öffnen“ kann jemand eine Tür öffnen, der nichts von der Arbeitsweise eines Schlosses versteht. Wer eine Lösungsvorschrift für eine schwierige mathematische Aufgabe hat, kann diese damit lösen, auch wenn er nichts von der Mathematikaufgabe begreift. Das klingt fast nach einem Zaubertrick, der – wie wir nun wissen – eine handfeste Grundlage hat: den Algorithmus.





# Der versteckte Steuermann

Computer begreifen von dem, was sie ausführen, überhaupt nichts. Übergibt man ihnen jedoch einen Algorithmus, dessen Einzelschritte sie vollziehen können, so vermögen sie ein schwieriges Problem richtig zu lösen und nützliche Dinge zu verrichten – völlig ohne Verstand! Algorithmen, die vom Computer ausgeführt werden, heißen *Programme*.

## Wir merken uns:

Ein Programm sagt einem Computer, wie er bestimmte Aufgaben schrittweise lösen soll. Es ist eine Arbeitsvorschrift für einen Computer.

Bei einem Computer heißen die einzelnen Schritte *Anweisungen*.

## Wir merken uns:

Ein Programm ist eine Kette (oder Folge) von Anweisungen. Nach Aufnahme in einen Computer bewirkt es, daß dieser eine bestimmte Aufgabe löst oder ein bestimmtes Ergebnis erzielt.

Genauer über Programme, Algorithmen, ihre Bausteine und darüber, wie man programmiert, erfährst du im Kapitel 10.

Algorithmen für einen Computer müssen natürlich klarer und eindeutiger sein als Algorithmen für Menschen. Vieles, was für uns selbstverständlich ist, weiß der Computer nicht; man muß ihm alles genau mitteilen. Ein Kochrezept für einen Pudding ist für uns (meistens) verständlich, aber ein Computer könnte damit nichts anfangen. Computerprogramme müssen ebenso genau beschrieben sein wie die Tastenfolge beim Ausrechnen einer Aufgabe mit dem Taschenrechner.

Ein Fehlgriff beim Eingeben, und das Ergebnis ist unbrauchbar. Es ist daher kein Wunder, daß lange Computerprogramme Fehler enthalten. Kostproben

dafür haben wir schon im Kapitel 3 kennengelernt. Wenn die 106jährige Oma zur Einschulung aufgefordert wird, so ist nicht der Computer fehlerhaft, sondern der benutzte Algorithmus beziehungsweise das verwendete Programm. Sie schreiben dem Computer Unsinniges vor.

Ebenso wie es im alltäglichen Leben Tätigkeiten gibt, die zu keinem Ergebnis führen, gibt es „Algorithmen“, die kein brauchbares Ergebnis liefern. Ein solch „lausiger“ Algorithmus ist beispielsweise:

1. Addiere den Wert der Größe A!
2. Wiederhole Schritt 1!

Solche „Algorithmen“ kommen – genau nach Vorschrift – nie zu einem Ende! Und wenn die Größe A den Wert Null hat, ist der Unfug noch größer. Ein solcher „Algorithmus“ beschäftigt den, welcher ihn ausführen muß, völlig, und er bewirkt nichts. – Algorithmen und Programme können Fehler enthalten ...

## Wir merken uns:

Algorithmen und Programme müssen sorgfältig geprüft werden, ob sie wirklich brauchbar sind. Denn Computer sind programmgesteuerte Automaten.

Computer können Programme ausführen und damit Aufgaben für uns lösen. Computer arbeiten Algorithmen automatisch ab. Das ist ein entscheidender Schlüssel zum Verständnis der Computerwelt.

Wer ist nun der gesuchte versteckte Steuermann des Computers?

Das eingegebene Programm!

Was kann ein Computer ohne Programm für uns erledigen?

Soviel wie ein Auto ohne Fahrer, wie ein Flugzeug ohne Pilot – einfach nichts!



## Tips und Tests für Pfffige

- 6.1. Was ist ein Automat?
- 6.2. Nenne Beispiele für Automaten im Alltag!
- 6.3. Was ist eine Steuerung?
- 6.4. Suche Beispiele für Steuerungen aus dem Alltag und beschreibe sie! Welches Ziel soll durch das Steuern erreicht werden?
- 6.5. Sind Kochrezepte Algorithmen?
- 6.6. Teste ein Kochrezept, zum Beispiel das für einen Pudding! Führt der Algorithmus zum gewünschten Ergebnis? Erfinde einen Algorithmus für einen Puddingkochautomaten!
- 6.7. Susis Briefe bestehen aus drei Zeilen. Nenne einen Algorithmus, nach dem diese Briefe geschrieben werden können.
- 6.8. Mathematische Hausaufgaben, wie Textaufgaben, müssen in folgenden Schritten gelöst werden:
  1. Aufgabe genau verstehen.
  2. Passenden Lösungsweg suchen.
  3. Lösungsweg auf die Aufgabe anwenden.
  4. Ergebnis überprüfen.
 Schüler führen oft nicht den Schritt 1 aus. Was ist das Ergebnis?
- 6.9. Beschreibe deinen Weg zur Schule! Ist das ein Algorithmus? Kann dich damit ein programm-
 

gesteuerter Transportroboter täglich selbständig in die Schule fahren?

6.10. Wenn du eine häufig vorkommende schwierige Aufgabe erfolgreich gelöst hast, schreibe den Lösungsweg (Algorithmus) für später in ein Merkheft! Lege so eine Rezept-, Trick- oder Algorithmensammlung an! Welchen Vorteil hat sie für dich oder andere?
- 6.11. Prüfe einen Algorithmus, den du erfunden hast, mit einem Freund! Er soll den Algorithmus genau nachspielen! Vergleiche das erwartete mit dem tatsächlichen Ergebnis! Korrigiere den Algorithmus, wenn das Ergebnis falsch ist!
- 6.12. Veranstatte mit deinen Freunden einen Erfinderswettbewerb für Algorithmen, die nützliche Alltagsaufgaben erledigen helfen, zum Beispiel für einen Haushalts- oder Schulreinigungsroboter HAUSROB! Wer findet bei dieser Algorithmenolympiade den besten, klarsten, kürzesten, originellsten Algorithmus? Diskutiere das Ergebnis mit einem Erwachsenen, der von Technik etwas versteht.
- 6.13. Es läßt sich alles automatisieren. Ist dieser Satz *a* richtig oder *b* falsch? Begründe deine Antwort!
- 6.14. Bastle dir einen ALSCHI (einen Algorithmen-schieber) oder einen KACO (einen Kartoncomputer)! Damit lassen sich Algorithmen besser abarbeiten und überprüfen. Benötigt werden Karton, Klebstoff, Bleistift, Schere, rechter Winkel. KACO besteht – wie ein ordentlicher Computer – aus einem Programm und einem Computergehäuse. Wie diese beiden Teile beschaffen sind und montiert werden, ist auf der nächsten Seite dargestellt.
 

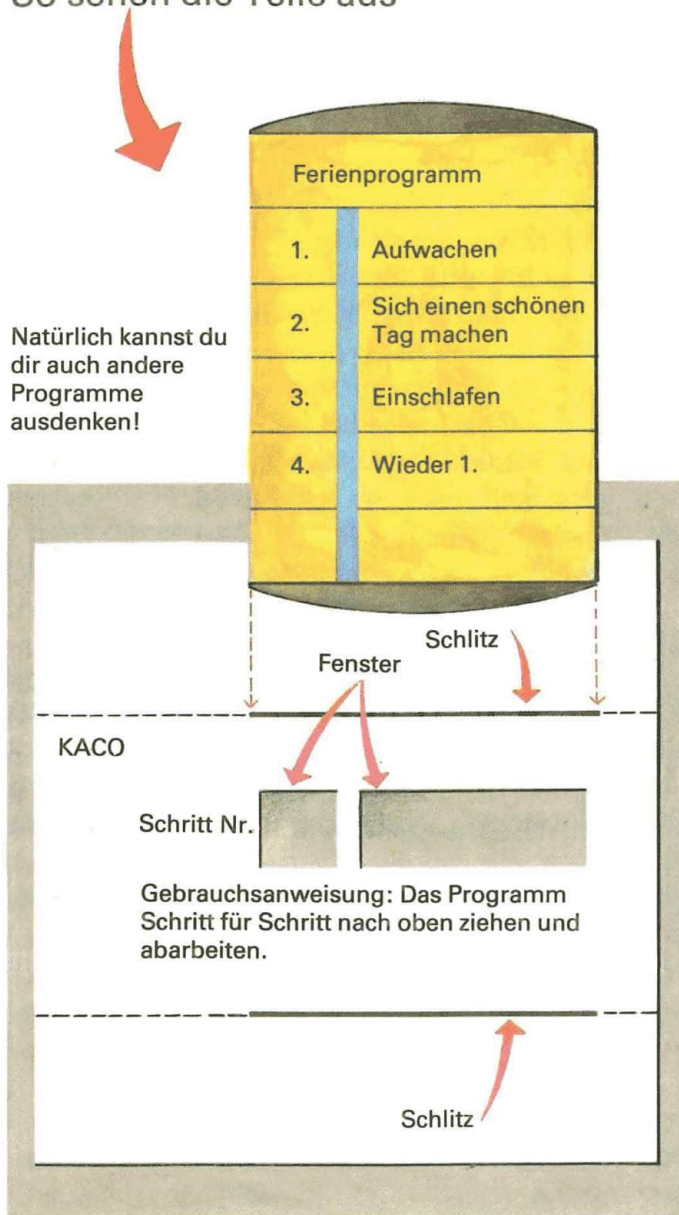
Diesen – sehr einfachen – Kartoncomputer wirst du noch in einer stark verbesserten Ausführung kennenlernen.

6.15. Ein Klassenfest war ein großer Erfolg. Das nächste Fest soll ebenso gut gelingen. Was ist zu tun?

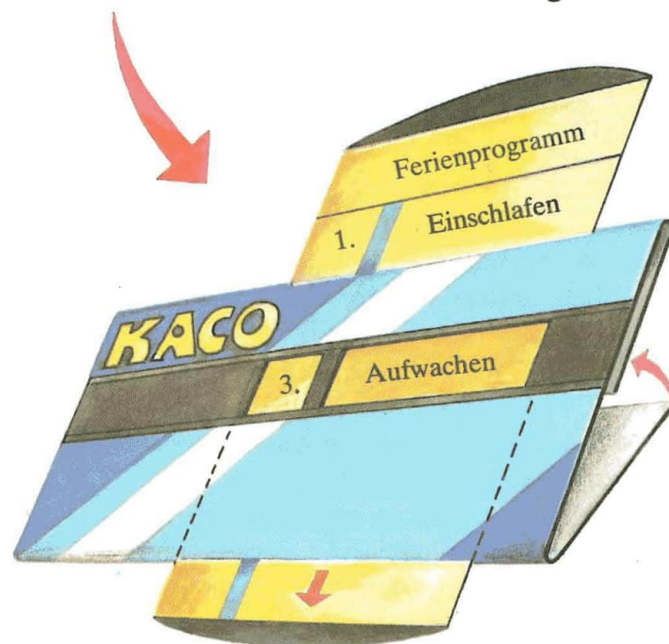


## KACO Bastelanleitung

So sehen die Teile aus



So werden die Teile zusammengesetzt

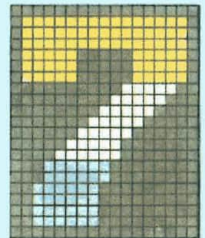




## 7. Geheimnisse des Computergedächtnisses

---

Soll ein Computer Daten verarbeiten, muß er sie in einem Speicher aufbewahren und von dort wieder entnehmen. Das soll uns jetzt beschäftigen. Wir werden einen weiteren wichtigen Schlüssel zur Computerwelt finden und Fachbegriffe wie Bit und Byte (sprich: beit) kennenlernen.





## Der kleinste Speicherplatz und das kleinste Datum

Computer können Daten – zum Beispiel Zahlen, Texte, Muster oder Bilder – speichern. Was ist eigentlich der aller kleinste Platz zum Speichern von Daten? Was ist das aller kleinste Datum (= Singular von Daten), das sich darin ablegen läßt?

Schauen wir uns im Alltag um! Was tun wir, wenn wir uns irgend etwas merken wollen und keinen Merktzettel zur Hand haben? Jungs schlingen als Gedächtnisstütze zuweilen einen Knoten ins Taschentuch. Mädchen malen sich manchmal mit dem Füller ein Kreuz auf den Daumnagel. Sie schauen eben öfter auf die Fingernägel, und die Jungs greifen häufiger in ihre Hosentasche und werden so erinnert. Wichtig ist, daß man weiß, was der Knoten oder das Kreuz bedeuten.

Was haben diese Beispiele gemeinsam?



Es gibt jeweils nur zwei zusammenhängende Situationen (Fachleute sagen dazu *Zustände*). Diese können eingestellt werden und bleiben erhalten; die Einstellung wird gespeichert. Der Knoten bleibt im Taschentuch, bis er gelöst wird. Das Kreuz auf dem Daumnagel bleibt „gespeichert“, bis es gelöscht wird. Die Tür bleibt geschlossen, bis jemand sie öffnet.

Das kleinste „Fach“, in dem sich einer von zwei möglichen Zuständen oder eine Entscheidung speichern lässt, heißt *Bit*.

### Wir merken uns:

Der kleinste Speicherplatz ist ein Bit. In ihm kann man einen von zwei Zuständen speichern.

Es hat sich international eingebürgert, die beiden möglichen Zustände mit 0 und 1 zu bezeichnen. (Diese beiden Ziffern der „Zweierzahlen“ sind nicht mit der 0 und der 1 des Zehnersystems – mit den Ziffern 0 bis 9 – zu verwechseln.)

### Wir merken uns:

In einem Bit kann entweder 0 oder 1 gespeichert werden.

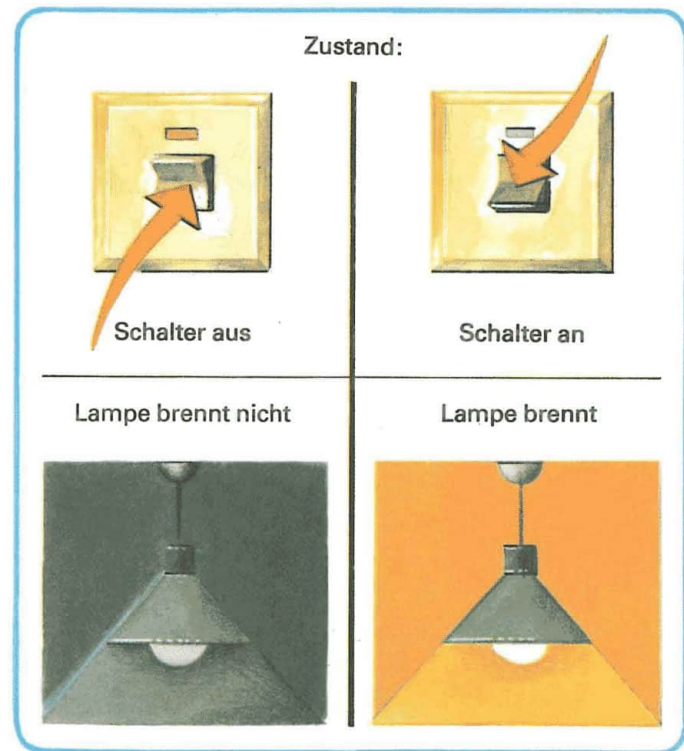
Damit ist noch nichts darüber gesagt, welcher Wert nun gespeichert ist. Die Entscheidung, ob der Wert 0 oder 1 ausgewählt und eingetragen wird, ist die kleinste Mitteilung. Sie wird *bit* genannt. In 1 Bit lässt sich also 1 bit eintragen. Auf die Frage „Scheint die Sonne?“ kann mit Ja oder Nein geantwortet werden. Jede Antwort ist eine Auswahl von beiden Möglichkeiten, die kleinste denkbare Mitteilung von 1 bit. Um diese Antwort zu speichern, benötigt man einen Speicherplatz von 1 Bit.

Eine Computertaste kann betätigt sein oder nicht (zwei Zustände!). Drücken wir diese Taste oder betätigen wir sie nicht, so ist das ebenfalls eine Ja/Nein-Entscheidung, eine Mitteilung von 1 bit. Um sich diese Tastenstellung zu „merken“, benötigt der Computer

einen Speicher von 1 Bit. Solche Ja/Nein-Entscheidungen und Mitteilungen von 1 bit sind übrigens der kleinste Schritt in der Anweisungsfolge eines Algorithmus.

Bit-Speicher lassen sich mit Hilfe der modernen Technik billig und in großer Anzahl herstellen. Sie arbeiten schnell und sicher.

Diesen technischen Speicher und seine Wirkungen kennt jeder:



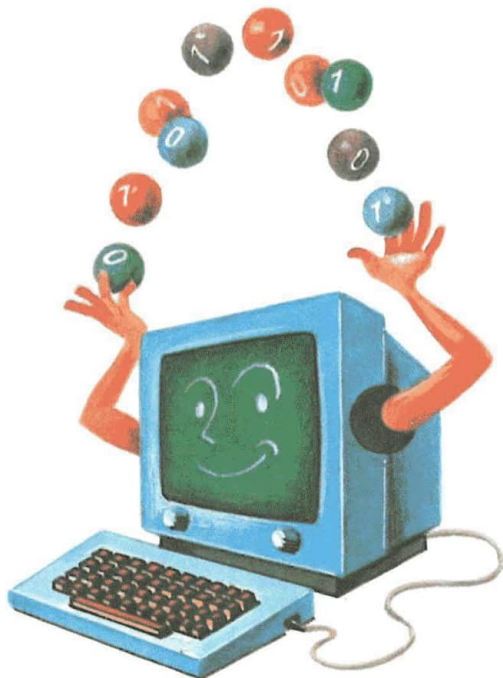


### Wir merken uns:

Computer speichern bits in Bits und rechnen mit ihnen, also nur mit den Werten 0 und 1.

Um in der Schule mit Zahlen rechnen zu können, müssen wir die zehn Ziffern von 0 bis 9 beherrschen. Computer arbeiten viel primitiver, indem sie nur die Werte 0 und 1 verwenden. Doch damit läßt sich wesentlich einfacher rechnen als mit den Ziffern des Zehnersystems. Wie das funktioniert, wird im Kapitel 13 behandelt.

Computer können eigentlich nichts weiter, als blitzschnell und ganz exakt mit Nullen und Einsen jonglieren.



## Bits, Bits und nochmals Bits

Um Daten verarbeiten zu können, benötigt der Computer ein Programm. Wo befindet sich eigentlich das Programm, welches die Arbeit eines Computers steuert?

Nicht nur die Daten, auch das aus einer sehr großen Anzahl von Bits bestehende Programm befindet sich im Speicher eines Computers. So, wie man beliebige Zahlen in einen Computer einspeichern kann, lassen sich auch beliebige Programme in den Speicher eines Computers laden, sofern in ihm genügend Speicherplatz in Bits vorhanden ist. Daten *und* Programme werden als Folgen von 0 und 1 gespeichert. Nachdem man einen speicherprogrammierten Rechner mit einem (neuen) Programm geladen hat, kann er wie ein völlig anderer Automat arbeiten.

### Wir merken uns:

Computer sind speicherprogrammierte oder speicher-gesteuerte Automaten.

Das ist ein weiterer Schlüssel zum Verständnis des Computers und dessen vielfältiger Anwendungsmöglichkeiten – natürlich des Universalcomputers. Spezialcomputer haben ein fest eingebautes spezielles Programm; es kann vom Nutzer nicht ohne weiteres verändert werden.

Auch Taschenrechner enthalten fest eingespeicherte Programme, die durch das Drücken einer Verarbeitungstaste, zum Beispiel + oder –, gestartet werden.

Für die Lösung praktischer Probleme der Datenverarbeitung benötigt man allerdings wesentlich mehr als nur die zwei Werte eines Bits. Es ist daher vorteilhaft, mehrere nebeneinanderliegende Bits zusammenzufassen und als ein geschlossenes Ganzes zu behandeln. Das geschieht in Gruppen von 8 Bits, den *Bytes*. (Bytes [sprich: beits] = Plural von Byte.)

### **Wir merken uns:**

Eine Gruppe von 8 aufeinanderfolgenden Bits wird Byte genannt.

1 Byte besteht also aus 8 Bits. Um 1 Byte durch Kreuze auf Fingernägeln zu speichern, benötigt man 8 Finger. Und man braucht 8 nebeneinanderliegende Taschentücher, um 1 Byte zu speichern (wenn man nur 1 Knoten je Taschentuch zuläßt). Die Elektronik benötigt zum Speichern eines Bytes einen Platz, der viel kleiner ist als die Dicke eines Kinderhaares. Später, im Kapitel 13, werden wir auch noch einen Blick in diese Computerwelt des Allerkleinsten werfen.

Moderne Computer verarbeiten zur gleichen Zeit (parallel) nicht nur jeweils 1 Bit, sondern gleichzeitig ganze Bit-Ketten. Taschenrechner arbeiten zum Beispiel mit 4 Bits, Kleincomputer mit 8 oder 16 Bits, Minirechner und Großcomputer mit 32 oder mehr Bits. Nach dieser Verarbeitungsbreite spricht man auch zum Beispiel von einem 16-Bit-Computer. Dieser Computer verarbeitet gleichzeitig 16 Bits.

### **Wir merken uns:**

Computer werden auch nach ihrer Verarbeitungsbreite (in Bits) unterschieden.

Je größer die Verarbeitungsbreite, desto mehr Bits werden gleichzeitig bewegt und desto schneller arbeiten diese Computer.

## Gedächtnismeister in Bytes

Einige Bits und Bytes reichen jedoch nicht aus, um Daten und Programme für praktische Aufgaben zu speichern. Welche Größe haben die Speicher in Computern? Um diese Frage beantworten zu können, muß man wissen: Die Speichergröße der Zentraleinheit von Computern wird in Kilobyte, abgekürzt KByte – auch

KB – ( $1 \text{ KByte} = 1024 \text{ Bytes} = 8192 \text{ Bits}$ ) gemessen, die der Speicher von Zusatzgeräten in Megabyte, abgekürzt MByte – auch MB – ( $1 \text{ MByte} = 1024 \text{ KBytes} = 1\,048\,576 \text{ Bytes} = 8\,388\,608 \text{ Bits}$ ). Die Computerfachleute benutzen den Vorsatz „Kilo“ als Abkürzung für 1024 und nicht – wie sonst üblich – als Abkürzung für 1000 Einheiten.

Moderne leistungsfähige Personalcomputer haben beispielsweise häufig einen Speicher von 256 KByte in der Zentraleinheit und einen Zusatzspeicher von 20 MByte. Das sind wirklich erstaunliche Leistungen der Technik! Unternehmen wir hierzu ein kleines Gedankenexperiment: Den Speicher von 1 Bit wollen wir darstellen mit einem Schüler, der entweder eine Tafel mit einer 0 oder eine Tafel mit einer 1 hochhält, je nach Inhalt des Bits. Nehmen wir an, ein Schüler braucht für sich 25 Zentimeter Platz. Acht Schüler in einer Reihe können 1 Byte anzeigen. Diese Schülerreihe ist 2 Meter lang. Die Länge der Schülerkette für verschiedene Speichergrößen zeigt das Bild auf der folgenden Seite. Solch leistungsfähige Speicher finden im Gehäuse eines Tischcomputers unauffällig Platz.

Die Speicherkapazität des menschlichen Gehirns soll im Bereich  $10^3 \dots 10^{13}$  MByte liegen ( $10^3$  bedeutet eine 1 mit 3 Nullen = 1000,  $10^{13}$  eine 1 mit 13 Nullen = 10 000 000 000 000).



Diese  
Speicher-  
größe ...

wird veranschaulicht durch

1 Bit



1 Schüler, der entweder die Tafel „0“ oder die Tafel „1“  
hochhält  
25 cm, diesen Platz braucht 1 Schüler

1 Byte



8 Schüler  
Diesen Platz braucht die  
Schülerkette

1 KByte



$\frac{1}{2}$  Wegstunde  
2,048 km



256 KByte



rund 10 Stunden Fahrt mit dem Personenzug  
524,288 km



1 MByte



rund 2,5 Flugstunden  
2097,152 km



20 MByte



rund 1mal um die Erde  
41 943,04 km





## Wie findet man Daten im Speicher wieder?

Computer arbeiten sehr schnell. Die Daten müssen also rasch abgelegt und zügig herbeigeschafft werden, da ist keine Zeit für langes Suchen.

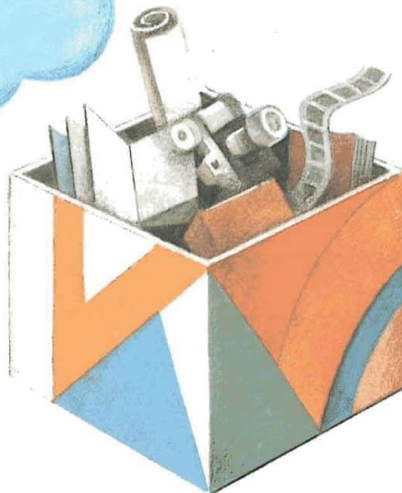
Kinder sammeln gern alles mögliche und speichern es: Ansichtskarten, Briefmarken, Steine, Blätter, Reiseandenken.

Ordnung halten vermeidet Suchen und spart Zeit. In einem Schrank mit vielen Fächern (oder Mappen), die sorgfältig beschriftet sind, findet man wesentlich schneller etwas als in einem ungeordneten Haufen. In eindeutig beschrifteten Fächern (ähnlich Briefkästen) den *Inhalt* ablegen und anhand der Fachbeschriftungen schnell das Richtige wiederfinden – dieses Prinzip wird auch im Computer benutzt. Im Speicher des Computers nehmen Fächer die Daten auf, zum Beispiel



### So kann aufbewahrt und gespeichert werden

Ein ungeordneter Haufen in einer Kiste – meine Sammlung



Meine Sammlung – geordnet in einem Schrank mit vielen kleinen Fächern



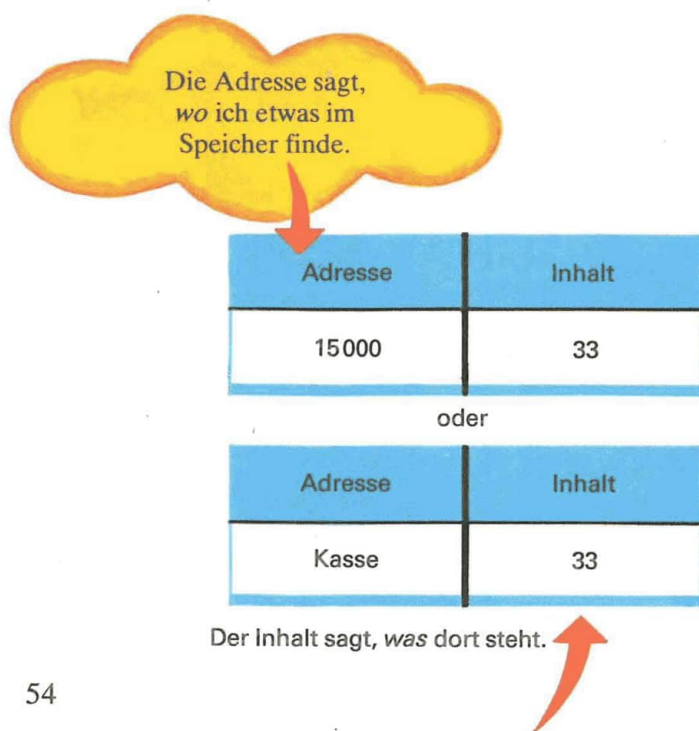
1 Byte oder mehrere Bytes mit Buchstaben oder Ziffern. Die Fächer sind allesamt unterschiedlich und eindeutig bezeichnet, zum Beispiel mit einer fortlaufenden Zahl oder einem Namen. Das ist die *Adresse* der gespeicherten Daten.

### Wir merken uns:

Die Adresse gibt den Ort an, wo bestimmte Daten zu finden sind.

Daten werden an bestimmten Adressen im Speicher abgelegt und mit Hilfe dieser Adressen rasch wiedergefunden.

Sehr sorgfältig muß die Adresse eines Speicherplatzes von dem Inhalt der gespeicherten Daten unterschieden werden. Stellen wir uns vor, an der Adresse 15 000 ist der Betrag noch vorhandenen Taschengeldes gespeichert: 33 Pfennig. Der Inhalt des Speicherplatzes mit der Adresse 15 000 ist also die Zahl 33. Es ist natürlich völlig falsch, anzunehmen, daß unter der Adresse 15 000 oder 33 die Zahl 15 000 – also 15 000 Pfennig = 150 Mark – steht.



### Wir merken uns:

Adresse und Inhalt eines Speicherplatzes sind zweierlei. Sie müssen sorgfältig unterschieden werden.

Auch im Alltag darf man die Adresse nicht mit den Menschen verwechseln, die dort wohnen.

Um keinen Fehler zu machen, schreiben wir Adresse und Inhalt getrennt nebeneinander, wie es das Bild zeigt.

Was im Speicher eines Computers steht, läßt sich auf langen Papierstreifen notieren, wo neben der Adresse immer der Inhalt steht. Das Kartonmodell eines solchen Computerspeichers wird später beim Demonstrationscomputer in Teil E des Buchanhangs genutzt.

## Von Schnelligkeit und Fassungsvermögen

Es gibt Computertypen unterschiedlicher Schnelligkeit in der Verarbeitung von Daten. Muß es da nicht auch verschiedene Speichertypen geben, weil die zu verarbeitenden Daten unterschiedlich schnell gespeichert werden müssen?

Alle Computertypen benötigen einen Speicher, aus dem Ausgangsdaten schnell gelesen und in den Ergebnisse schnell geschrieben werden: den Hauptspeicher. Dieser ist sowohl beim Taschenrechner als auch bei anderen Computertypen in der Zentraleinheit untergebracht. Für komplizierte Anwendungen braucht man außer dem Hauptspeicher noch Zusatzspeicher. Diese befinden sich oft in Geräten außerhalb des Computergehäuses. (Wir erinnern uns an die Bilder der Computerausstellung!) Zusatzspeicher haben meist ein großes Fassungsvermögen an Bytes, arbeiten jedoch nicht so schnell wie der Hauptspeicher. Die kleinsten und schnellsten Speicher im Hauptspeicher jedes Computers heißen *Register*.

Genauer über Speichertypen zeigt die folgende Tabelle.



Diese Angaben beziehen sich auf eine Textseite mit rund 2000 Zeichen

| Speichertyp    | Speicher                        | Fassungsvermögen<br>(Gesamtinhalt)    | Schnelligkeit beim<br>Speichern oder<br>Herauslesen |
|----------------|---------------------------------|---------------------------------------|---|
| Hauptspeicher  | Register                        | einige Zeilen<br>(einige Bytes)       | kleiner als eine<br>Millionstelsekunde              |
|                | Hauptspeicher                   | viele Seiten<br>(viele Bytes)         | Millionstelsekunde                                  |
| Zusatzspeicher | Hilfs- oder Massen-<br>speicher | Tausende Seiten<br>(einige MByte)     | Bruchteil einer<br>Sekunde                          |
|                | Archivspeicher                  | viele tausend Seiten<br>(viele MByte) | Sekunden/Minuten                                    |

Diesen werden wir uns noch genauer ansehen!

Wer die Tabelle aufmerksam studiert hat, wird einen wichtigen allgemeinen Zusammenhang erkannt haben: Je schneller ein Speicher ist, desto kleiner ist sein Fassungsvermögen in Bytes.

Das ist eine Faustregel. Sie gilt nur für Speicher von annähernd gleichem Baujahr, weil die technische Entwicklung immer schneller arbeitende Speicher hervorbringt.

Warum so viele Speichertypen? Wäre es nicht günstiger, wir benutzen nur die schnellsten?

Diese Frage können wir selbst beantworten. Auf welche Weise gelangen wir täglich in die Schule? Entweder wir laufen zu Fuß, auf Schusters Rappen, oder wir setzen uns auf ein Fahrrad. Das ist billig. Mancher benutzt den Bus, die Straßenbahn oder den Zug. Das kostet schon etwas mehr. Ein schnelleres Fahrzeug wäre ein Auto, ein Taxi. Und noch viel

schneller kämen wir – eine geeignete Start- und Landefläche vorausgesetzt – mit dem Hubschrauber zur Schule. Doch das wäre ein zu teurer Spaß!

Ähnlich ist es bei den Computern. Leistungsstärkere Speicher sind teurer als leistungsschwächere. Und nicht für alle Aufgaben sind sehr schnelle Speicher nötig, zum Beispiel für das Archivieren wichtiger Daten.

Vor Tausenden von Jahren nutzten unsere Vorfahren Steine oder Tontafeln, um Daten ihres Lebens, ihres Handels und Wandels sowie ihrer Beobachtungen einzuritzen. Wir können diese Zeugnisse in Geschichtsmuseen betrachten.

Viele Daten unseres Zusammenlebens sind heute in Computerspeichern abgelegt. Computerspeicher werden damit zu Zeugen der jüngsten Geschichte. Sie stellen eine Art elektronische Chronik dar.





## Tips und Tests für Pfiffige

- 7.1. Wie heißt der kleinste Speicherplatz, und was läßt sich mit ihm speichern?
- 7.2. Computer arbeiten zweiwertig (das Fremdwort dafür ist binär). Erkläre, was das bedeutet!
- 7.3. Ein Computer hat 77 Zahlen gespeichert und diese addiert. Wie viele Adressen muß das Programm verarbeiten: *a* eine, *b* beliebig viele, *c* 77?
- 7.4. Erläutere den Satz: Ein Computer ist ein Automat, der von einem gespeicherten Programm gesteuert wird!
- 7.5. Wieviel nebeneinanderliegende Bits ergeben 1 Byte – *a* 1, *b* 2, *c* 4, *d* 8, *e* 16?
- 7.6. Ein 16-Bit-Personalcomputer verarbeitet wieviel Bytes gleichzeitig?
- 7.7. Sind 1 KByte *a* 128, *b* 1000, *c* 1024, *d* 1025 Bytes?
- 7.8. Auf dem Speicherplatz 1234 eines Hauptspeichers steht die Zahl 2345. Ist die Adresse *a* 2345 oder *b* 1234, der Inhalt *c* 1234 oder *d* 2345?
- 7.9. Benötigt ein Computer für seine Arbeit unbedingt einen *a* Zusatzspeicher, *b* Hauptspeicher, *c* Haupt- und Zusatzspeicher?
- 7.10. Welcher Zusammenhang besteht zwischen der Schnelligkeit von Speichern und ihrem Fassungsvermögen (bei gleichem Stand der Technik)?

7.11. Wie könnte ein zweiwertiges Rechenbrett (ein binärer Abakus) aussehen?

7.12. Bastle ein Speichermodell, das aus einem Papierstreifen sowie einem Papier„gehäuse“ mit einem Adreß- und einem Inhaltsfenster besteht!

In den Fenstern zeigen sich jeweils eine Adresse und der dazugehörige Speicherinhalt. Der Papierstreifen soll verschiebbar sein (vergleiche Basteltip 6.14.).



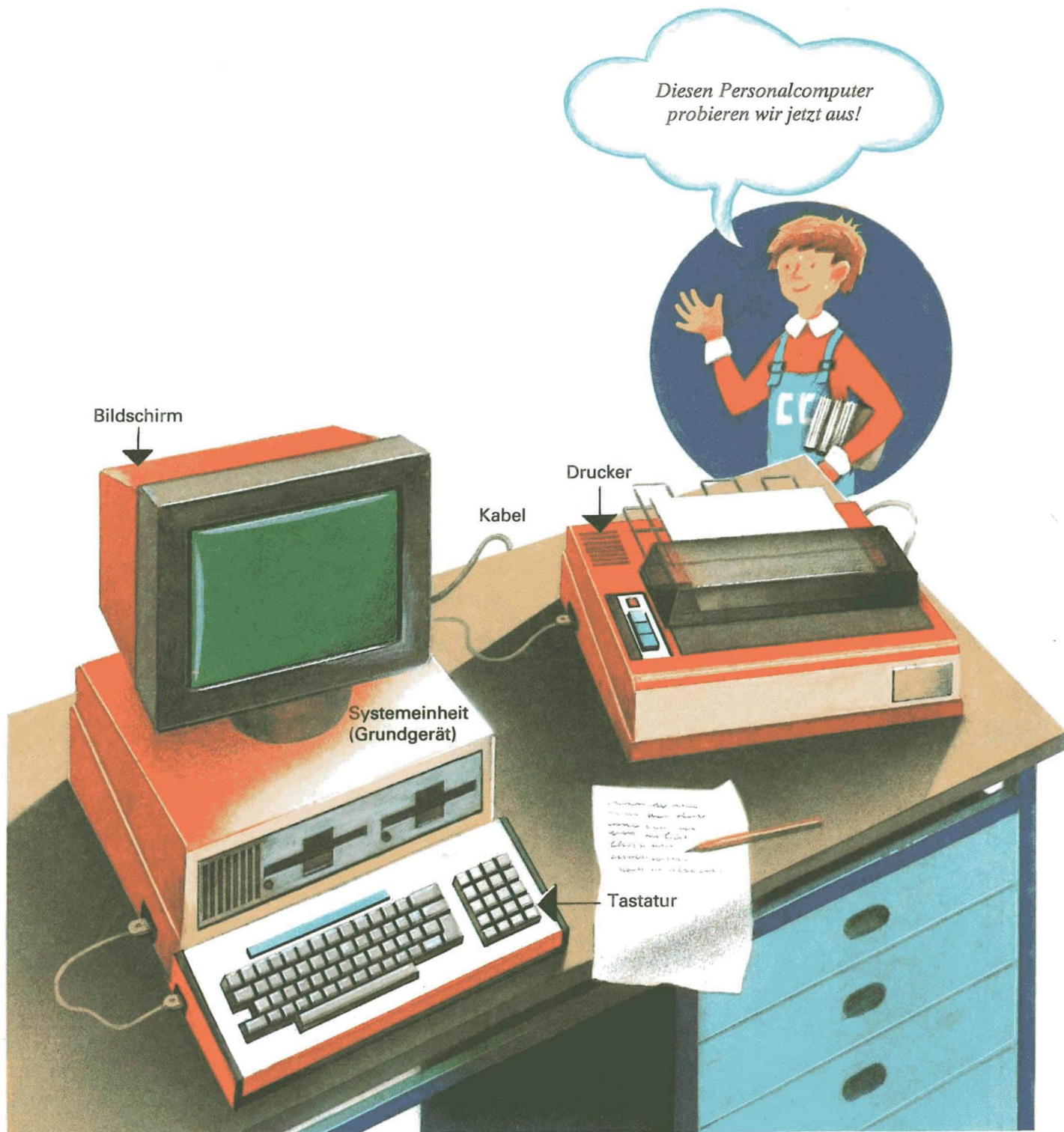
Zweiwertiges  
Rechenbrett  
für 1 Byte

## 8. Wir probieren einen Computer aus

---

Ein Computer kann vieles. Wie veranlassen wir ihn, für uns eine bestimmte Aufgabe zu lösen, beispielsweise einen Brief an Susi zu schreiben, eine komplizierte Hausaufgabe zu lösen oder ein schönes Rätselspiel mit uns zu spielen? Wie wird ein Computer gestartet? Wie ist mit ihm umzugehen?







## Einen Personalcomputer starten

Am besten, wir probieren das gleich einmal an einem „richtigen“ Computer aus. Für unseren Versuch verwenden wir einen modernen und leistungsfähigen Personalcomputer (kurz: PC), wie er in Büros, Betrieben, Hochschulen, Krankenhäusern und in anderen Einrichtungen, jetzt auch in Schulen benutzt wird.

Computer sind hochwertige und empfindliche Geräte, wie Fernsehapparate. Wir müssen mit ihnen sehr sorgsam umgehen. Deshalb halten wir uns genau an ihre Bedienungsvorschrift.

Zunächst verbinden wir den Computer über den Netzstecker mit dem elektrischen Stromnetz. Dann betätigen wir eine (oft verdeckte) Taste und schalten ihn so ein.

Plötzlich kommt Leben in das tote Gerät. Auf der Tastatur beginnen rote Lämpchen zu flackern. Ein leises Surren ertönt. Nach einer Weile wird auch der Bildschirm „lebendig“. In rascher Folge zeigt das Gerät Zahlen, Buchstaben und Wörter an. Am Drucker leuchtet ein rotes Lämpchen, und aus dem Gehäuse dringt ein leises Brummen. Der Computer schreibt noch einige Zeilen auf den Bildschirm, dann verändert sich die Anzeige nicht mehr.

Das Bild rechts oben zeigt, wie der Bildschirm nun aussieht.

Neben der Pfeilspitze blinkt ein kleiner Strich. Das bedeutet: Ich, der Personalcomputer, bin genügend warm geworden und bereit, jetzt für dich zu arbeiten. Ich weiß nur nicht, was ich tun soll. Sag mir, welche Aufgabe ich für dich erledigen soll. Du mußt mir deinen Auftrag mit den Tasten meiner Tastatur mitteilen.

Die mit der Tastatur eingegebenen Zeichen werden an die Stelle auf dem Bildschirm geschrieben, wo der blinkende Strich gerade steht. Dieses auffällige Zeichen – auf manchen Computerbildschirmen ist es auch ein flackerndes helles Rechteck – wird *Cursor* genannt.



Hell-dunkel blinkender Strich, der Cursor. Manchmal sind es auch andere auffällige Zeichen, zum Beispiel ein blinkendes kleines Rechteck.

## Wir erteilen einen Auftrag

Der Computer soll einen Brief an unsere Freundin Susi schreiben, schön säuberlich, wie eine Schreibmaschine es vermag. Wir versuchen das einfach einmal. Mit den Tasten geben wir das Wort SUSI ein und warten ab, was geschieht.

Wird der PC einen Brief schreiben, kaputtgehen oder sich selbst abschalten? Wird er anfangen, auf Computerart zu schimpfen oder einen unangenehmen Ton von sich zu geben?



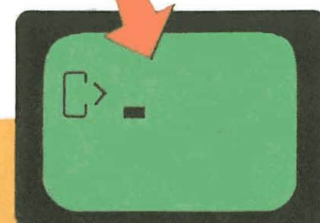
*Buchstabenweise  
erteilen wir den Auftrag  
SUSI.*

Das passiert  
am Bildschirm  
an der Tastatur

Schritt

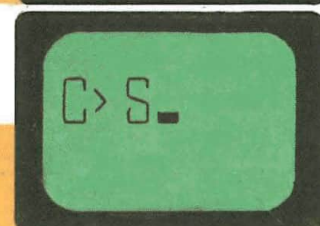
1

Das bekannte Ausgangsbild



2

Umschalttaste für Großbuchstaben betätigen und S eingeben

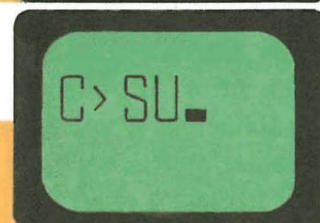


3

Der Buchstabe S erscheint auf dem Bildschirm

4

Buchstabe U eingeben

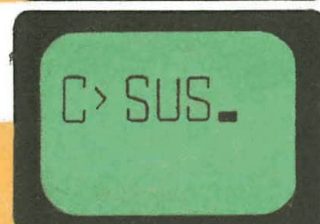


5

Der Buchstabe U erscheint rechts neben dem S

6

Buchstabe S eingeben

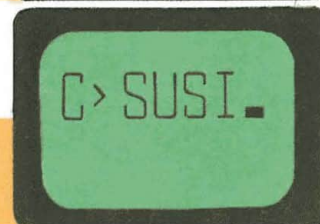
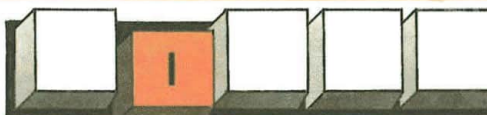


7

Der Buchstabe S erscheint rechts neben dem U

8

Buchstabe I eingeben



9

Der Buchstabe I erscheint rechts neben dem S

10

Zum Abschluß die „Eingabe-Ende“-Taste drücken.



Der Cursor zeigt immer auf die Stelle, an der das nächste Zeichen eingegeben wird.



Es geschieht nichts. Am Bildschirm steht jedoch das Wort SUSI. Der Computer wartet auf weitere Zeichen, denn er weiß nicht, daß der Auftrag fertig eingegeben ist. Was ist zu tun? Wie teilen wir dem Computer mit, daß er den Auftrag SUSI auszuführen hat? Für diesen Zweck gibt es eine besondere Taste. Fachleute nennen sie ENTER-, NEWLINE- (sprich: njulein) oder RETURN- (sprich: ritörn) Taste (das hängt vom Typ der Tastatur ab) oder einfach Endetaste. Manchmal ist sie mit ET beschriftet. Diese Taste drücken wir. (Siehe Schritt 10 des linken Bildes!)

Damit haben wir dem Computer unseren Auftrag mitgeteilt. Jetzt muß er den Brief für uns schreiben, und wir können uns ausruhen ...

## Das ist die Antwort des Computers

Blitzschnell erscheint nach dem Drücken der ENTER-Taste in der nächsten Zeile am Bildschirm die Antwort des Computers (Schritt 11) auf unseren Auftrag. Bei manchen Geräten erscheinen solche *Ausschriften* in einer Fremdsprache, oft in englisch.

Wie ist die Ausschrift „Falsches Kommando...“ zu verstehen? Wir haben etwas falsch gemacht. Der Computer versuchte, den Auftrag SUSI auszuführen. Er „schaute“ in seinem Speicher nach, ob er dort die genauen Arbeitsvorschriften zum Erledigen des Auftrags SUSI findet. Das Kommando SUSI ist ihm fremd. Er weiß nicht, was zu tun ist. Er wartet deshalb wieder. Der Cursor blinkt und zeigt die Bereitschaft zur Entgegennahme des nächsten Auftrags oder Kommandos.

Ist der Computer nicht ein aufmerksamer, geduldiger und genauer Partner?



Schritt

11

Der Computer antwortet auf den Auftrag SUSI ... und erwartet einen neuen Auftrag.





## Wir probieren trotzdem weiter...

Manche Kinder und Erwachsene vertragen es nicht, wenn ihnen gesagt wird, daß sie etwas falsch gemacht haben, und verlassen entrüstet den Computer. Zu ihnen zählen wir nicht. Wir wollen den Computer weiter ausprobieren und beginnen mit einem erneuten Versuch. Jetzt soll er den Auftrag H wie Hausaufgaben für uns erledigen.

Die erneute Ausschrift „Falsches Kommando...“ verwundert uns nicht mehr, denn einen Auftrag oder das Kommando H kennt der Computer gleichfalls nicht. Er kann somit auch unsere Hausaufgaben nicht erledigen.

Bei unseren Versuchen hat der Computer nicht folgsam das gemacht, was wir wünschten. Doch wir haben Wichtiges über sein Verhalten gelernt, auf das wir uns beim Umgang mit ihm einstellen müssen: Was wir dem Computer nicht beigebracht haben, das kann er auch nicht.

## Ein Andenken drucken lassen

Einen Computer zu benutzen, ist teuer, und die Zeit, mit ihm arbeiten zu können, knapp. Wir unternehmen deshalb den letzten Versuch.

Was jetzt auf dem Bildschirm steht, möchten wir schwarz auf weiß gedruckt mitnehmen. Bei vielen Computern ist das kein Problem. Wir schalten den Drucker ein und erteilen durch Tastendruck den entsprechenden Auftrag.

Das Ergebnis liegt nun ausgedruckt vor uns. Wir können es nach Hause tragen.

Insgesamt sind Computer eigenwillige Partner, auf die wir uns einrichten müssen, wollen wir sie beherrschen. Leider gibt es sehr viele Typen, die sich in wichtigen Kleinigkeiten (zum Beispiel Tastaturen) unterscheiden. Wollen wir beispielsweise eine Hartkopie, einen ausgedruckten Bildschirmtext, erhalten, so müssen wir bei verschiedenen Computertypen unterschiedliche Handgriffe ausführen.

Unser nächster Versuch



1.

Wir geben den Auftrag H ein (drücken die Tasten H und ENTER)

2.

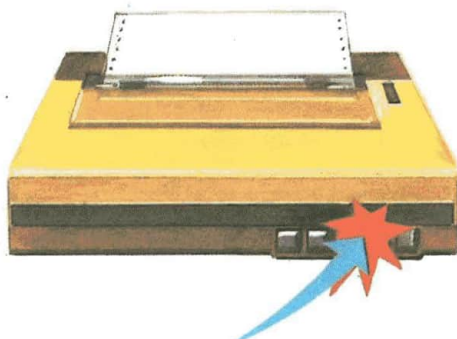
Blitzschnell antwortet der Computer und erwartet einen neuen Auftrag



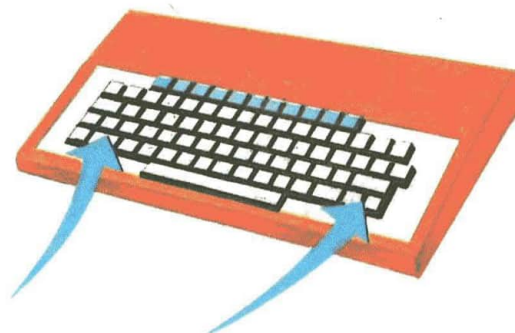


So wird eine  
„harte“ Kopie des  
Bildschirms angefertigt.

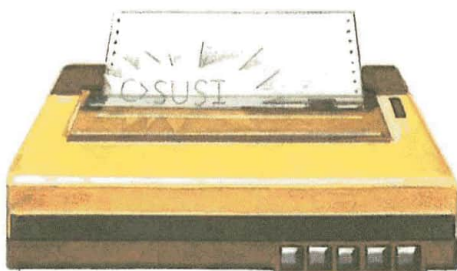
- 1.** Den Drucker einschalten



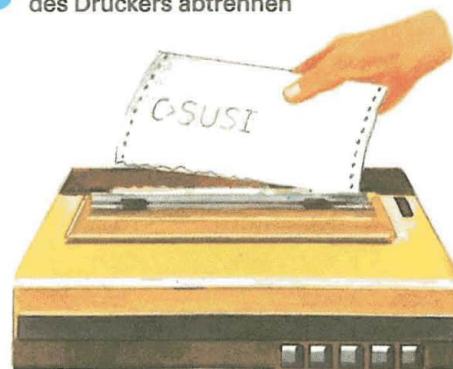
- 2.** Die Tasten  $\uparrow$  und PRN auf der Tastatur gleichzeitig drücken



- 3.** Einen Moment warten, bis der Drucker fertig gedruckt hat



- 4.** Das bedruckte Papier z. B. an der Abreißkante des Druckers abtrennen



### Wir merken uns:

Ein Computer versteht nur Anweisungen und Aufträge, die er schon kennt, also gespeichert hat, oder die wir ihm eingeben.

Zum Umgang mit einem Computer gehört viel Wissen, Sorgfalt und Geduld – aber auch die nötige Fingerfertigkeit.

Einen Computer probieren wir am besten mit einem Fachmann aus, der diesen Computertyp gut kennt.



## Tips und Tests für Pfiffige

- 8.1. Welche Anweisungen und Aufträge „versteht“ ein Computer?
- 8.2. Viele Computer kennen zum Beispiel die Anweisung CLS. Sie veranlaßt, daß alles vom Bildschirm gelöscht, „wegradiert“, wird. Was passiert, wenn wir nur die drei Buchstaben CLS eingeben?
- 8.3. Was macht ein Computer, der CLS „versteht“, wenn wir nur die Tasten C und L und dann die ENTER-Taste drücken?
- 8.4. Ein Computer führt nicht die Anweisungen aus, die er ausführen soll. Woran kann das liegen?
- 8.5. Wie zeigt ein Computer uns am Bildschirm, daß er auf eine Eingabe wartet?
- 8.6. Wird ein Computer ungeduldig, wenn er keinen Auftrag von uns hat?
- 8.7. Wie muß man einen Computer behandeln?



## 9. Zwanzig Zusatzgeräte – genauer angeschaut

---

Im Kapitel 5 haben wir komplette Computer betrachtet. Sie bestehen aus der zentralen Verarbeitungseinheit, kurz Zentraleinheit genannt. An diese sind unterschiedliche Zusatzgeräte, die *Peripherie*, angeschlossen, mit deren Hilfe man Daten eingeben, speichern und ausgeben kann. Einige Peripheriegeräte wollen wir uns jetzt genauer anschauen. Welche Geräte zusätzlich benötigt werden, hängt von dem Typ und den Aufgaben des Computers ab.



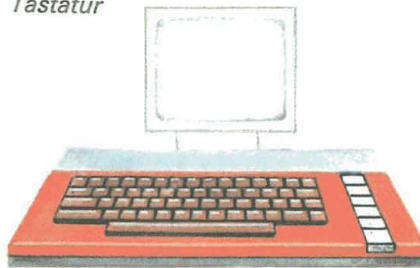
## Damit wir ihn „füttern“ können

Computer müssen von Menschen mit Daten und Programmen versorgt werden, sonst „wissen“ sie nicht, was sie tun sollen. Das geschieht mit besonderen Geräten, den Eingabegeräten. Über solche Geräte hat der Benutzer direkten Kontakt mit dem Computer.

*Einige Geräte  
zum Eingeben von Daten*



*Tastatur*



Zur Eingabe von Daten werden die gewünschten Tasten nacheinander mit dem Finger gedrückt. Es gibt Tasten für Buchstaben, Sonderzeichen, Zahlen, Satzzeichen und anderes.

*Steuerknüppel*



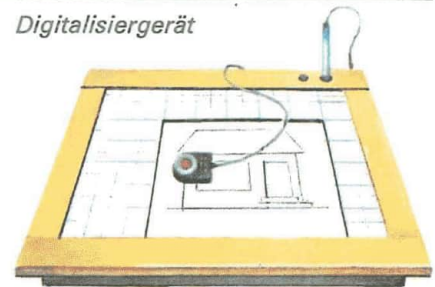
Der Steuerknüppel wird ähnlich wie die Gangschaltung in einem Auto betätigt. Er dient meist dazu, Zeichen oder Bildchen auf einem Bildschirm zu bewegen.

*Lesepistole für Klerschrift*



Die Lesepistole wird in der Hand gehalten und über eine Textzeile geführt, die in einer besonderen Druckschrift geschrieben wurde. Auf diese Weise lässt sich die Zeile in den Computer einlesen.

*Digitalisiergerät*



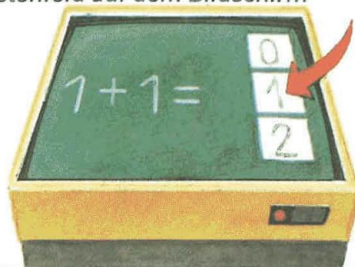
Mit der „Lesemaus“ können Bildpunkte von technischen Zeichnungen oder Tabellen „abgetastet“ und dem Computer übermittelt werden. Sie hat eine Lupe und einige Tasten.

### Lichtgriffel für den Bildschirm



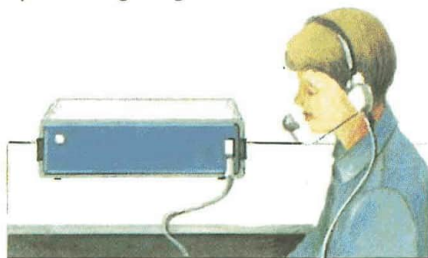
Mit dem Lichtgriffel kann direkt auf den Bildschirm gezeichnet werden. So lassen sich auch Zeichnungen in den Computer eingeben.

### Tastenfeld auf dem Bildschirm



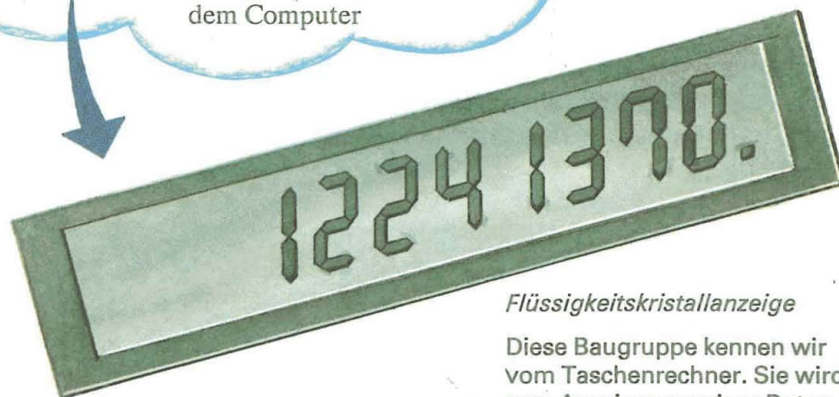
Auf dem Bildschirm sind Tasten abgebildet. Sie werden betätigt, indem man sie mit dem Finger berührt. Diese einfache Art der Eingabe ist technisch jedoch ziemlich aufwendig.

### Spracheingabegerät



Dieses Gerät erkennt einige gesprochene Wörter. Es kann ähnlich wie eine Tastatur benutzt werden, benötigt aber zum Eingeben keine Berührung mit dem Finger, sondern unsere Stimme.

Geräte für die rasche Anzeige von Daten, zum Beispiel beim „Gespräch“ mit dem Computer



### Flüssigkeitskristallanzeige

Diese Baugruppe kennen wir vom Taschenrechner. Sie wird zum Anzeigen weniger Daten benutzt.

### Schwarzweiß-Bildschirm



Er ist das wichtigste moderne Ausgabegerät für Computer, das uns Daten und Ergebnisse anzeigt. Weil Grün für die Augen angenehm ist, schreiben viele moderne Bildschirme hell auf dunkelgrünen Grund.

### Farb-Bildschirm



Damit können farbige Schriften und Bilder – wie bei einem Farbfernseher – ausgegeben werden. Natürlich sind diese Geräte teurer als Schwarzweiß-Bildschirme.

Bildschirme haben den Vorteil, daß ihre Bilder schnell wechseln können. Man benutzt sie vor allem für „Gespräche“ (Dialoge) mit dem Computer. Nach dem Abschalten der Geräte verlöschen die angezeigten Daten und Bilder.



## Wie beim Fernsehen

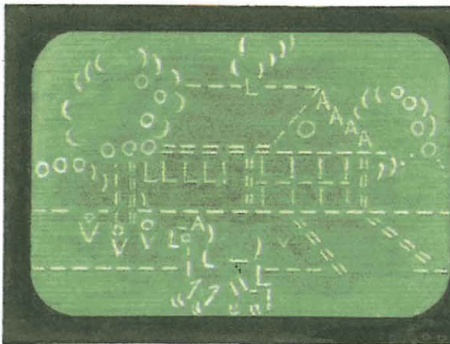
Computer arbeiten schnell, und sie sollen uns auch schnell ihre Ergebnisse anzeigen. Dazu dienen diese Geräte.

Auf Bildschirmen sind die Bilder und die Schriftzeichen aus kleinen Lichtpunkten, den Pixels, zusammengesetzt. Diese können hell auf dunklem Grund oder umgekehrt, dunkel auf hellem Grund, erscheinen. Aus farbigen Pünktchen oder Flächen entstehen farbige Bilder. Die Bildqualität hängt von Art und An-

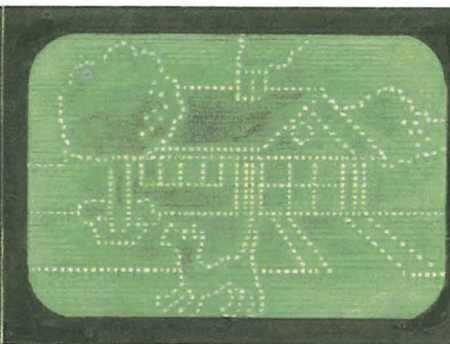
zahl der benutzten Bildelemente ab. Für wirklichkeitsnahe Bilder werden teure Bildschirme mit sehr vielen Pixels benötigt.

Manche Computer verfügen über kleine Lautsprecher, wie sie in Taschenradios enthalten sind. Damit vermögen sie unterschiedliche Töne und Geräusche zu erzeugen. Es gibt auch Ausgabegeräte, die einige Worte oder Sätze „sprechen“. Diese kommen aus einem Lautsprecher.

Computer können nicht nur mit dem Menschen Daten austauschen, sondern auch mit nahen oder weit entfernten Computern, Maschinen, Automaten oder



Das Bild wird aus Buchstaben und Satzzeichen zusammengesetzt



Die Bildelemente sind kleine rechteckige Flächen



Das Bild besteht aus vielen kleinen Pünktchen (Pixels), die man einzeln erkennt



Im Unterschied zu allen anderen Bildern sieht dieses ganz „normal“ aus. Es besteht aus sehr vielen Pünktchen, die man jedoch nicht mehr voneinander unterscheiden kann

Robotern in Verbindung treten und Daten senden und empfangen. Die erforderlichen Geräte sind meist unauffällig mit im Computergehäuse oder in anderen Geräten an Übertragungskabeln untergebracht.

Will man Bilder dauerhaft festhalten, das heißt schwarz oder farbig auf Papier drucken, braucht man zusätzliche Geräte.

## Schwarz auf weiß nach Hause tragen

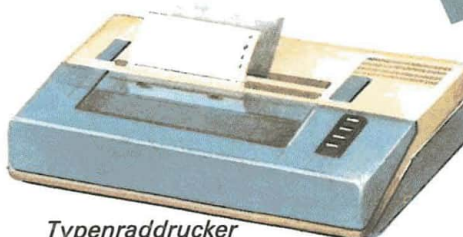
Die Daten auf der Anzeige des Taschenrechners oder dem Computerbildschirm verlöschen, sobald die Geräte abgeschaltet werden. Es sind nur flüchtige Bilder.

Geräte für das dauerhafte Festhalten von Daten auf Papier



**Nadeldrucker**

Er arbeitet verhältnismäßig geräuscharm. Die Schriftzeichen sind aus kleinen Punkten zusammengesetzt, die von feinen Drucknadeln erzeugt werden. Nadeldrucker können meist mehr als eine ganze Zeile in der Sekunde ausgeben. Das Druckbild wirkt grobkörnig.

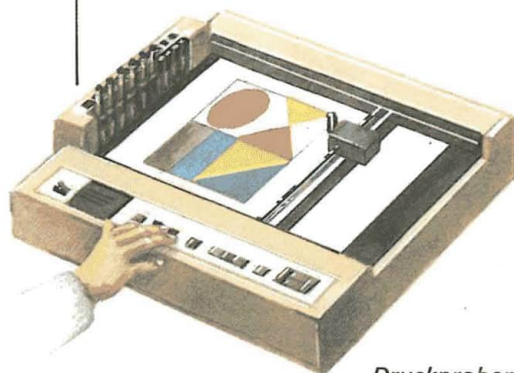


**Typenraddrucker**

Er liefert gestochen scharfe Druckbilder – wie gute Schreibmaschinen. Für eine Druckzeile benötigt er mehr als eine Sekunde. Typenraddrucker sind teurer als Nadeldrucker.

**Plotter**

Das ist ein zeichnender Roboter. Er zeichnet automatisch mit einem Stift, der über das Papier hin und her gefahren wird. Plotter arbeiten sehr sorgfältig und genau, dafür entsprechend langsam. Es gibt auch Geräte, die in mehreren Farben zeichnen.



*Druckproben befinden sich auf Seite 70 oben*



**Schnelldrucker**

Er ist groß, arbeitet schnell und geräuschvoll. Manche Geräte verfügen nur über Großbuchstaben. Sie drucken mit einem Schlag eine ganze Zeile, schaffen oft 20 und mehr Zeilen in der Sekunde. Schnelldrucker werden nur an große Computer angeschlossen.



Typenraddrucker

Erstaunliche Werke  
vollbringt der Mensch.

Nadeldrucker

Erstaunliche Werke  
vollbringt der Mensch.

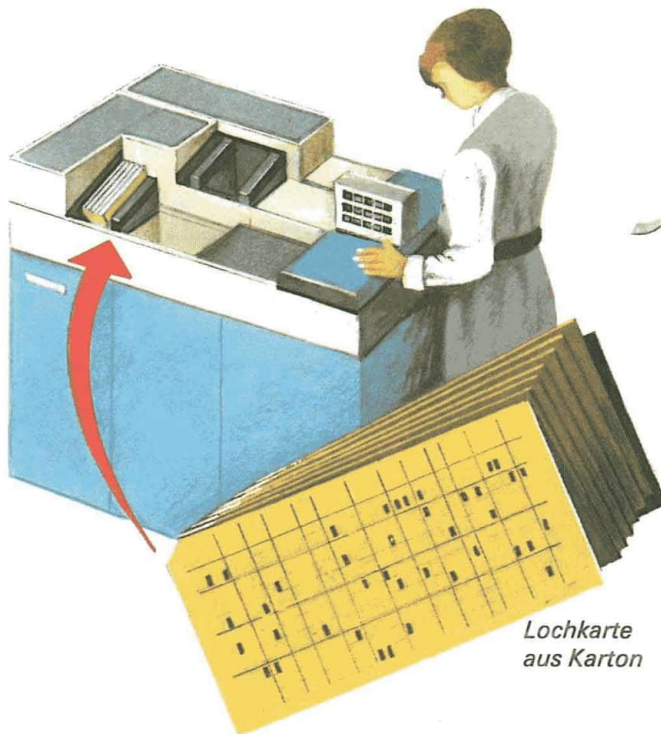
Schnelldrucker

ERSTAUNLICHE WERKE  
VOLLBRINGT DER MENSCH

Plotter

Schrift Plotter

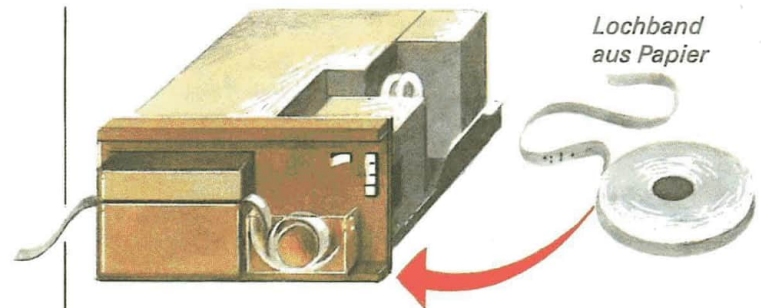
Einige Speichergeräte  
und ihre Datenträger



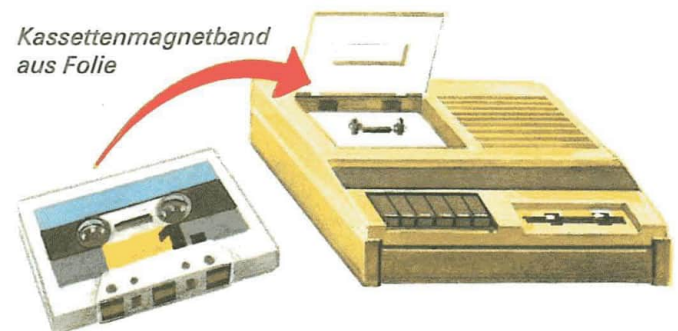
Die Lochkarte ist die Großmutter unter den Datenträgern. Sie war früher der wichtigste Datenträger. Die Daten werden als bestimmte Muster aus kleinen rechteckigen Löchern in einen besonderen Karton gestanzt. Lochkartengeräte findet man nur noch an großen Computern.

## Damit nichts vergessen wird

Der Arbeitsspeicher eines Computers „vergißt“ alle Daten, sobald er abgeschaltet wird. Programme und Daten für jeden Computerlauf neu einzugeben, ist mühsam und für lange Programme zu aufwendig. Man hat daher verschiedene Geräte zum dauerhaften Aufbewahren von Daten und Programmen entwickelt. Wollen wir Daten, zum Beispiel unseren Namen, auf herkömmliche Weise dauerhaft aufschreiben, benöti-



Das Lochband ist mit einem Muster aus runden Löchern versehen. Lange Lochbänder wickelt man auf Rollen. Geräte der Lochbandtechnik sind zuweilen auch an älteren kleinen Computern zu finden.



Das Kassettenmagnetbandgerät kennen wir vom Kassettenrecorder. Es ist nicht sehr teuer, nicht sehr schnell, aber schneller als ein Lochbandgerät. Wir finden es häufig bei Heim- und Lerncomputern



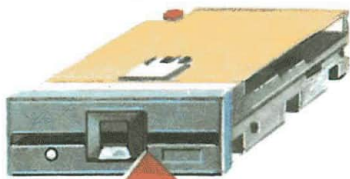
gen wir ein Blatt Papier, das die Daten tragen soll, Hand, Stift, Augen, um schreiben und wieder lesen zu können. Auch technische Speicher bedürfen eines Datenträgers und eines Gerätes zum Schreiben und zum Lesen. (Bei bewegten Datenträgern heißen solche Geräte Laufwerke.)

Manche Datenträger können nicht ausgewechselt werden, sie sind fest eingebaut, beispielsweise bei Festplattenspeichern. Im Unterschied zu Wechselplattenspeichern lassen sich die Plattenstapel nicht austauschen.

### Wir merken uns:

Um die richtigen Geräte für eine Computeranwendung zusammenstellen zu können, benötigt man große Sachkenntnis.

Es muß bekannt sein: Welche Aufgaben hat der Computer zu lösen? Welche Programme und Daten sollen das erledigen? Welche Geräteeigenschaften werden dafür benötigt?

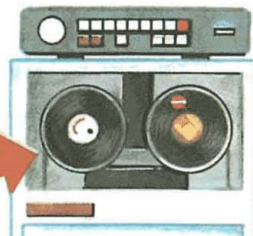


Diskette  
aus Plast  
(mit Kartonhülle)

Die Diskette ist eine dünne Plastscheibe (ähnlich einer kleinen Schallplatte), die geschützt in einer Kartonhülle verbleibt. Diese wird in den Schlitz des Laufwerkes geschoben. Es arbeitet relativ schnell und ist teurer als ein Kassettenmagnetbandgerät. Disketten und ihre Laufwerke finden wir als Massenspeicher häufig bei Mini- und Personalcomputern. Es gibt große Disketten (Standarddisketten) und kleine Disketten (Minidisketten).



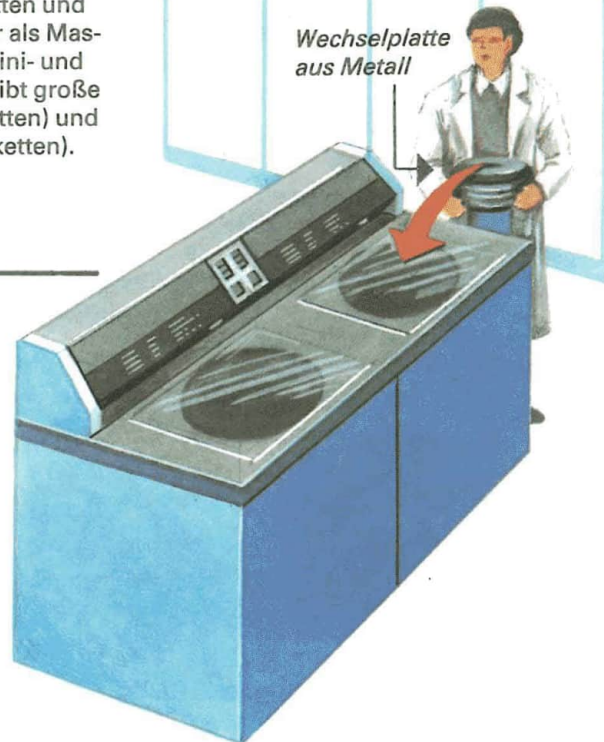
Magnetband  
aus Folie



Das Magnetband wird häufig zum längeren Aufbewahren (Archivieren) von Daten benutzt. Die Laufwerke für Magnetbänder sind teure, oft schrankgroße Geräte. Große Computer benötigen mehrere von ihnen. Das Laufwerk arbeitet viel schneller als ein Kassettenmagnetbandgerät. Das Suchen von Daten auf den Magnetspulen dauert allerdings oft Minuten.

Die Wechselplatte ist ein Speicher für große Datenmengen (Massenspeicher). Sie wird bei Klein- und Großcomputern eingesetzt. Im Bruchteil einer Sekunde werden auf der Wechselplatte beliebige Daten wiedergefunden.

Wechselplatte  
aus Metall



In herkömmlichen Büros werden noch die meisten Arbeiten mit dem Datenträger und Datenspeicher Papier ausgeführt. Kennzeichen solcher Büros sind Mappen voller Papier und Schränke voller Ordner.

In Büros mit moderner Computertechnik lassen sich Daten und Berichte papierlos auf geringstem Raum speichern. So finden zum Beispiel 70 bis 390 Schreibmaschinenseiten Text auf einer Minidiskette Platz. Fachleute sprechen deshalb von einem automatisierten, papierlosen (oder wenigstens papierarmen) Büro.

Mit solcher Technik kann zugleich wertvoller Rohstoff – das Holz, aus dem man Papier herstellt – eingespart werden.

## Speziell und originell

Technischen Ideen und Erfindungen sind (fast) keine Grenzen gesetzt. Das zeigen diese letzten vier Beispiele.

### *Rollkugel*

Die Kugel in dem Kästchen wird mit der Handfläche berührt und gerollt. Dadurch bewegen sich auf



dem Bildschirm Zeichen oder Bildchen. Die Rollkugel ist ein spezielles Eingabegerät.

### *Tastatur*

Diese Tastatur enthält in ihrem Gehäuse eine komplette Zentraleinheit. Solche Geräte sind das Kernstück von Heim-, Lern- oder Bildungscomputern.



### *Akustikkoppler*

Mit diesem Gerät können über das Telefon Daten an einen anderen Computer übertragen oder von



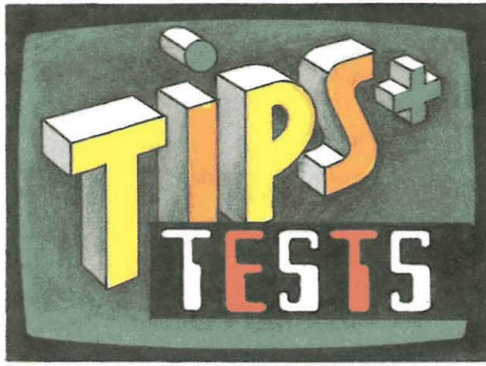
diesem empfangen werden. Die Computerzeichen werden dabei in verschiedenartige Töne umgewandelt. Das Gerät arbeitet langsam.

### *Tragbarer Personalcomputer*

Oben befindet sich ein ganz flacher Bildschirm. Zum Tragen des Computers wird der Bildschirm auf die Tastatur geklappt.







## Tips und Tests für Pffiffige

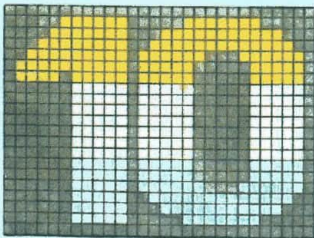
- 9.1. Benötigt ein Computer Zusatzgeräte?
- 9.2. Sammle Bilder von Peripheriegeräten! Du findest sie in Zeitschriften und Prospekten. Notiere dazu den Computertyp und wofür das Gerät benutzt wird!
- 9.3. Nenne Zusatzgeräte für Computer!
- 9.4. Benötigt man für einen Klein-, Heim- oder Lerncomputer unbedingt einen zusätzlichen Speicher?
- 9.5. Es ist kalt. Tino trägt Fausthandschuhe. Kann er damit eine Computertastatur zügig bedienen? Begründe deine Antwort.
- 9.6. Viele Computertastaturen – auch die Tastaturen der Schreibmaschinen – haben mindestens 26 Tasten für Buchstaben. Kann man mit diesen Tasten 52 Klein- und Großbuchstaben eingeben?
- 9.7. Was ist erforderlich, um die richtigen Geräte für eine Computeranwendung zusammenstellen zu können?
- 9.8. Auf Seite 135 findest du eine verkleinert dargestellte Tastatur eines leistungsstarken Personalcomputers. Du kannst sie vergrößert auf einen Karton zeichnen. Dann eignet sie sich zum Üben. Die Tasten lassen sich auch mit kreisrunden Scheiben aus Moosgummi versehen, die beschriftet werden. (Moosgummi wird zum Abdichten von Fenstern und Türen benutzt.)
- 9.9. Geschicklichkeitsspiel: Wer ist Tippmeister? (Wir verwenden die Übungstastatur von Tip 9.8. und eine Stoppuhr.)  
Veranstaltet einen Wettbewerb und ermittelt:  
Wer tippt am schnellsten?  
Wer hat sich am wenigsten vertippt?  
Wer tippt am schnellsten den folgenden Satz und macht dabei die wenigsten Fehler? „Übung macht den Computermeister und den Olympiasieger.“
- 9.10. Wer entwirft die schönsten Bilder für eine Computergrafik am Bildschirm? Die Bilder sind *a* aus Buchstaben, *b* aus Satzzeichen oder *c* aus kleinen dunklen oder hellen Rechtecken gleicher Größe zusammenzusetzen. Vielleicht gelingt dir das Porträt eines Mitschülers oder sogar ein Selbstporträt?



# 10. Wie programmiert man einen Computer?

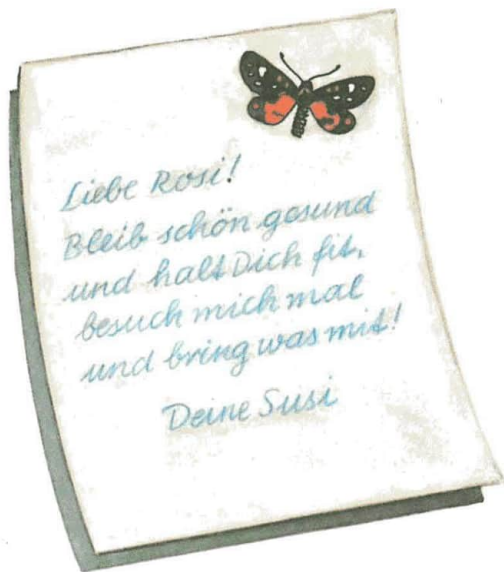
---

Ohne Programm kann der Computer für uns nicht nützlich werden. Allein das Programm schreibt ihm vor, was er tun soll. Doch wie erarbeitet man ein Programm für den Computer, und wie sorgt man dafür, daß er eine Aufgabe für uns erledigt? Das wollen wir jetzt ergründen.



## Briefe schreiben – o welche Automatenlust!

Susi schreibt gern Briefe und Karten an ihre vielen Freunde und Freundinnen, an ihre Bekannten und Verwandten. Am liebsten schreibt sie einen kurzen Brief mit einem kleinen Vers, zum Beispiel diesem:



Susi schreibt öfter solche Briefe. Tom hört davon und erklärt: „So etwas erledigt ein moderner Mensch mit einem Computer. Der schreibt auf einen Bildschirm.“

Susi sagt darauf: „Du Angeber!“

„Abwarten“, meint Tom.

Am nächsten Tag zeigt Tom Susi, wie man mit Hilfe des Computers einen Brief schreibt. Das Bild gibt es wieder.

Die heutigen Computer verstehen unsere Alltagssprache nicht. Wenn wir ihnen einen Algorithmus mitteilen wollen, müssen wir eine sehr vereinfachte, aber eindeutige Sprache – eine *Programmiersprache*, wie beispielsweise BASIC (sprich: beesik), – verwenden, die der Computer „versteht“.

### Wir merken uns:

Algorithmen für Computer werden in einer Programmiersprache geschrieben.

Eine Programmiersprache ist eine einfache künstliche Sprache, mit der einem Computer mitgeteilt wird, wie er eine Aufgabe lösen soll.



Das Programm, welches uns das Bild zeigt, besteht aus vier untereinanderstehenden Programmzeilen oder Anweisungen. Das sind kleine Arbeitsvorschriften für

den Computer. Jede Zeile beginnt mit einer Zahl – der Zeilennummer. Dann folgt das Wort PRINT (englisch: drucke) und in Anführungszeichen der auszugebende Text einer Briefzeile.

Das Programm beeindruckt Susi nicht, sie sagt zu Tom nur: „Angeber!“

Nun führt Tom die ungläubige Susi an einen Kleincomputer. Hier zeigt er ihr, daß das Programm tatsächlich den Brief auf den Bildschirm schreibt. Und das geschieht auf diese Weise:

1. Den Computer so einstellen, daß er die Sprache BASIC „versteht“ (zum Beispiel das Wort BASIC eingeben).
2. Sorgfältig und genau das gesamte Programm eingeben.
3. Das eingegebene Programm starten (zum Beispiel das Wort RUN [sprich: rann; englisch: laufe] eintippen).

Nicht vergessen: Nach jedem Wort und jeder Programmzeile die ENTER-Taste drücken!

Rasch erscheint der Brief, Zeile für Zeile, auf dem Bildschirm.

Susi bekommt große Augen. Aber sie ist noch nicht zufrieden. Auf dem Bildschirm stehen vor dem Brief Wörter, die nicht zum Brief gehören.

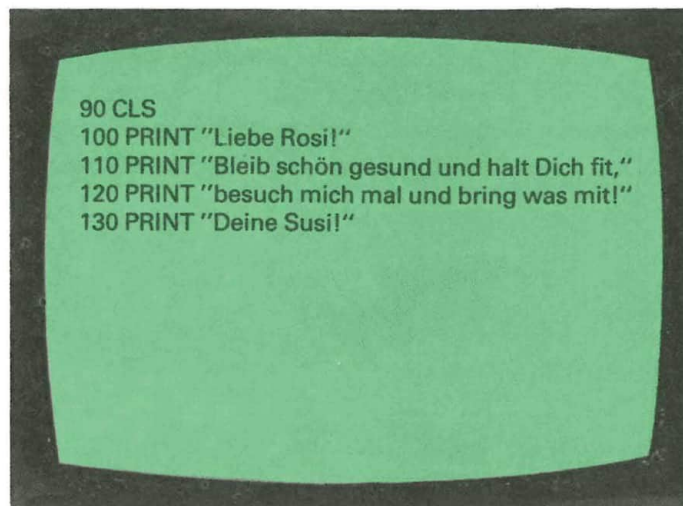
„Das ist kein Problem“, sagt Tom. „Wir brauchen nur die Anweisung zum Bildschirmlöschen CLS vor das Programm zu setzen.“

Tom tippt die Anweisung 90 CLS ein. (90 ist die Zeilennummer; dorthin, an die richtige Stelle im Programm, soll der Computer die Anweisung einfügen.) Dann schließt Tom die Zeile mit der Zeilenende-Taste (je nach Computertyp ENTER, ET, RETURN oder NEWLINE) ab. Nun startet er das Programm, indem er die Taste RUN drückt (oder RUN und ENTER eingibt).

Blitzschnell wird der Bildschirm gelöscht, und der Brief erscheint Zeile für Zeile.

Ist der Computer nicht ein folgsamer Geselle?

Unser kleines Programm mit seinen fünf Zeilen sieht nun so aus:



Der Computer schreibt das Programm sofort auf den Bildschirm, wenn die LIST-Taste gedrückt (oder die Anweisung LIST eingegeben und die ENTER-Taste betätigt) wird. Wir sehen, daß die Zeile 90 vom Computer an die gewünschte Stelle vor den anderen Anweisungen eingefügt wurde. Das kleine Spielprogramm vermag schon allerhand. Auf Tastendruck veranlaßt es den Computer, einen Brief zu schreiben. Susi und Tom haben einen briefschreibenden Automaten! Er verdient eine Extrabezeichnung. Nennen wir ihn BRIEFOMAT.

## Nun wird der BRIEFOMAT verbessert

Susi meint: „Das ist zwar schön und gut für die paar Briefe, die ich im Jahr an Rosi schicke. Doch ich schreibe auch Briefe an Grit, Kerstin, Barbara, Maudi und viele andere. Und das kann dein Programm nicht!“

„Auch dieses Problem zu lösen ist mit der Program-



miersprache BASIC ein Kinderspiel. Ich bastle das Programm einfach etwas um. Ich baue eine Anweisung ein, mit der du den Namen des Briefempfängers eingibst. Eine weitere Anweisung schreibt diesen Namen dann in die erste Briefzeile, in die Anrede.“




So sieht das Programm für den verbesserten BRIEFOMATEN aus.

Der Name des Briefempfängers wird vom Computer erfragt und unsere Antwort in der ersten Briefzeile mit ausgegeben.

```

70 PRINT "Brief an wen (Namen eingeben)"
80 INPUT N$
90 CLS
100 PRINT "Liebe"; N$; "!"
110 PRINT "Bleib schön gesund und halt Dich fit,"
120 PRINT "besuch mich mal und bring was mit!"
130 PRINT "Deine Susi"
140 END

```

 Neue Anweisungen  
 Veränderte Anweisung  
 Anweisung, die dem Computer das Programmende anzeigt

Das Programm enthält die veränderte Zeile 100 und zusätzlich die Zeilen 70, 80 und 140. Die Zeile 70 fordert den Benutzer auf, den Namen des Briefempfängers einzugeben.

Die Zeilen 80 und 100 schauen wir uns genauer an!

Mit der Zeile 80 INPUT N\$ können eingegebene Zeichen von der Tastatur an das Programm übergeben werden. (INPUT [englisch] = gib ein. \$ ist das internationale Währungszeichen; es wird üblicherweise als Abkürzung für Geldwährungen, zum Bei-

spiel Mark, Rubel, Dollar, benutzt. Manche Tastaturen, Bildschirme und Drucker haben dieses Zeichen nicht, man benutzt dafür \$.) Die so eingegebenen Zeichen werden unter der Adresse mit dem Namen N\$ gespeichert. Solche „Speicherfächer“ mit einem bestimmten Namen für verschiedene Daten heißen *Variable*.

Die Anweisung 100 PRINT "Liebe"; N\$; "!" setzt neben das Wort "Liebe" den eingegebenen Namen, der unter dem Variable-Namen N\$ gespeichert ist. Und daran wird noch – wie wir es möchten – das Ausrufezeichen geschrieben.

Der BRIEFOMAT arbeitet natürlich auch mit anderen Namen, zum Beispiel mit Butterblume, Lokomotive oder Fee, die auf diese Weise zu einem Besuch bei Susi eingeladen werden. Der Computer versteht ja nichts von dem, was er tut.

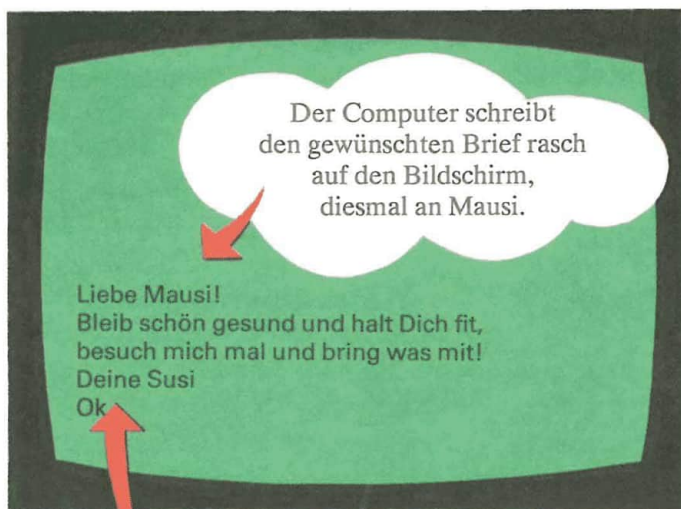
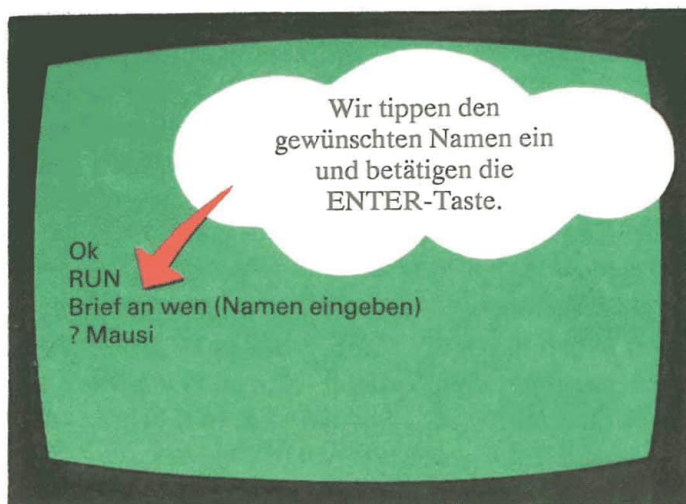
So läuft das Programm auf dem Computer, nachdem es gestartet wurde.

```

Ok
RUN
Brief an wen (Namen eingeben)
?

```

Nach dem Eingeben von RUN und dem Betätigen der ENTER-Taste fragt uns der Computer nach dem Namen und erwartet an der Stelle, wo das ? steht, unsere Eingabe von der Tastatur.



Diese beiden Buchstaben – das sogenannte Bereitschaftszeichen – bedeuten: Ich, der Computer, habe meinen Auftrag in der Programmiersprache BASIC wunschgemäß erledigt und erwarte neue Befehle.  
Der Brief erscheint als neues Bildschirmbild.

Schneller, als man lesen kann, schreibt der Computer den Brief auf den Bildschirm.

Susi schweigt. Sie hat rote Wangen bekommen. Ihr schwirrt der Gedanke durch den Kopf, daß sie mit dem BRIEFOMATEN an alle Tiere im Zoo, an alle Kinder der Stadt oder gar des ganzen Landes solche Briefchen schreiben könnte...

Der Computer und BASIC und das Beherrschen dieser Programmiersprache machen es möglich!

## Mathematische Formeln – mit BASIC kein Problem

Mit dem Computer wie mit einer „intelligenten“ Schreibmaschine zu arbeiten, ist sicher ganz nützlich. Aber wie schreibt man ein Programm, das eine kleine mathematische Aufgabe löst? Schau dir das nächste Beispiel an!

Rechenaufgaben teilt man in der Programmiersprache BASIC dem Computer einfach dadurch mit, daß man die Berechnungsformel eingibt. Das muß in einer Form geschehen, die der Computer ausführen kann. So ist zum Beispiel für den Mal-Punkt ein Mal-Stern einzugeben. Also statt  $0,47 \cdot 23 \cdot 2$  ist dem Computer  $0.47 * 23 * 2$  mitzuteilen. In den Programmiersprachen ist es auch üblich, statt des Kommas (beispielsweise in 0,47) einen Dezimalpunkt zu schreiben (also 0.47).

Natürlich läßt sich die mathematische Aufgabe auch mit einem Taschenrechner lösen. Nur müssen wir hierbei die Formel jedesmal neu eingeben. Im BASIC-Programm kann diese Formel gespeichert und nach Wunsch beliebig oft geladen und abgearbeitet werden.

Wir haben einen ersten Eindruck erhalten, wie kleinste Aufgaben für einen Computer mit einer Programmiersprache programmiert werden. Fassen wir unsere Erfahrungen zusammen. Was ist nun programmieren?



*Eine kleine mathematische Aufgabe*

Auf einem Kinderfest soll es Bowle für alle Kinder geben. Ein Glas kostet 47 Pfennig oder 0,47 Mark. 23 Kinder nehmen an diesem Fest teil, jedes Kind wird 2 Glas Bowle trinken. Was kostet die Bowle für alle Kinder?

Dieses BASIC-Programm von 1 Anweisung löst die Aufgabe

```
PRINT "Die Bowle für 23 Kinder kostet";
      0.47 * 23 * 2; "M"
```

Das ist die Berechnungsformel für das Ergebnis



Und das wird nach dem Eingeben und Ausführen der Programmzeile am Bildschirm angezeigt:

Die Bowle für 23 Kinder kostet 21.62 M  
Ok

Diese Mitteilung kennen wir schon.

### Wir merken uns:

Programmieren ist das Übersetzen einer zu lösenden Aufgabe in eine Folge von Anweisungen oder Befehlen, die ein Computer ausführen kann. Arbeitet der Computer diese Folge von Anweisungen (den Algorithmus) ab, wird die Aufgabe für uns gelöst.

## Auf jeden Punkt kommt es an

Computer führen die Anweisungen, die wir ihnen eingeben, ganz genau aus. Werden Befehle oder Daten falsch eingegeben, so weist uns der Computer mit einer Fehlermeldung darauf hin. Es kann aber auch passieren, daß er bei Daten und Befehlen anderes ausführt, als wir beabsichtigen.

Zu den ersten und beeindruckenden Erlebnissen eines Anfängers im Programmieren zählt, wie erstaunlich viele kleinere und größere Fehler er gemacht hat, ehe sein Programm wunschgemäß gelaufen ist. Schon beim Eingeben eines kleinen Programms kommt es zu Tippfehlern, die jedoch leicht erkannt und korrigiert werden können.

Für das Korrigieren gibt es besondere Tasten und Hilfsprogramme. Andere Fehler in Programmen – sie werden im Englischen anschaulich als Wanzen oder Flöhe bezeichnet – sind (auch in kleinen Programmen) nur schwer zu finden.

Mittlere Programme haben oft Zehntausende Anweisungen, sehr große Programme einige Millionen. In allen größeren Programmen können noch Programmfehler versteckt sein. Vielleicht enthalten sie einige Kombinationen von Eingabedaten, die fehlerhaft sind, nicht überprüft wurden und nun zu unerwünschten Ergebnissen führen. Es ist daher (leider) normal, daß größere Programme manchmal seltsame Dinge anstellen. Kleine Kostproben davon haben wir schon kennengelernt.

Wurden Programme jedoch schon längere Zeit von vielen Anwendern benutzt und erkannte Fehler vom



Hersteller immer korrigiert, so enthalten sie selten ernsthafte Fehler; man kann ihnen demzufolge Vertrauen schenken.

#### Wir merken uns:

Neu geschriebene Programme müssen vor ihrer Anwendung sehr sorgfältig geprüft werden, ob sie ihren Zweck auch erfüllen.

Mit welcher Aufmerksamkeit und Sorgfalt Programme ausgearbeitet werden müssen, zeigt einer der bisher teuersten Tippfehler in der Geschichte der Computeranwendung. Im Juli 1962 startete in einem US-amerikanischen Raketenversuchsgelände die erste Rakete mit einer automatisch arbeitenden Planetensonde. Schon wenige Minuten nach dem Start kam die Rakete gefährlich von ihrem Kurs ab und mußte durch ein Kommando über Funk zerstört werden. Ursache war ein Fehler im Programm des Computers, das die Flugbahn vorausberechnete. Statt der durch ein Komma getrennten Zahlen 1,3 stand dort 1.3.

Punkt und Komma hatten in der benutzten Programmiersprache völlig verschiedene Wirkungen. Mit dem Punkt lieferte das Programm unerwünschte und damit falsche Ergebnisse. Der Programmfehler verursachte einen Verlust von vielen Millionen Dollar.

#### Wir merken uns:

Programme müssen mit größter Sorgfalt geschrieben und geprüft werden. Es kommt dabei auf jedes Zeichen an!

Fehlern durch sorgfältiges und systematisches Arbeiten beim Programmieren vorzubeugen, laufend die Arbeitsergebnisse zu überprüfen – das ist besser, als später die Fehlerfolgen und Schäden zu beseitigen.

## Große Probleme beherrschen – aber wie?

Wie lassen sich schwierige und umfangreiche Probleme automatisieren? Wie bringt man sie auf den Computer?

Ein hoher Berg läßt sich nicht mit einem Satz erklimmen, ein Haus nicht mit einem Handgriff erbauen.

Einen hohen Gipfel kann man in Etappen besteigen, ein Haus Stein für Stein oder Platte für Platte bauen. Man muß große Probleme oder Aufgaben in kleine zerlegen und diese Schritt für Schritt bewältigen. So arbeitet man auch große Programme für Computer aus.

Große Programme lassen sich aus kleinen Programmen zusammensetzen und alle Programme aus kleinsten Programmbausteinen zusammenfügen. Drei wichtige kleine Bausteine werden wir jetzt kennenlernen.

In allen bisher vorgestellten Programmen und Algorithmen gab es zwischen Beginn und Ende nur Anweisungen, die stets *nacheinander* ausgeführt wurden und die man daher als *Folge* bezeichnet. Beim Abarbeiten einer Folge wird jede ihrer Anweisungen gleich oft ausgeführt.

Es gibt aber auch Anweisungen und Vorgänge, die nicht immer ausgeführt werden. Das Mädchen muß sich beispielsweise entscheiden, ob es seine Münze in den Eisautomaten steckt oder nicht.

Der Junge muß sich entscheiden, ob er sich nach dem Mädchen umdreht oder nicht. Der Wanderer muß wählen, welchen Weg er nimmt. Der Algorithmusbaustein für solche Entscheidungen heißt *Auswahl*. Der Algorithmus läuft einen von zwei Wegen, nicht beide gleichzeitig.

Anschaulich wird das in Ablaufplänen gezeichnet. Sie bestehen aus Kästchen, die mit immer nur einem Pfeil verbunden sind. Die Kästchen enthalten die Anweisungen oder Teilschritte; die Pfeile zeigen die

# Algorithmen- und Programmbausteine

Eis kaufen

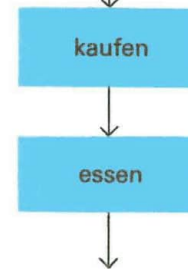


Eis essen

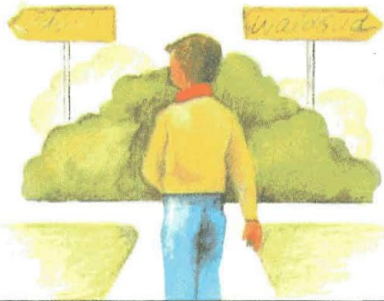


So werden die Vorgänge als Ablaufplan gezeichnet.

Folge



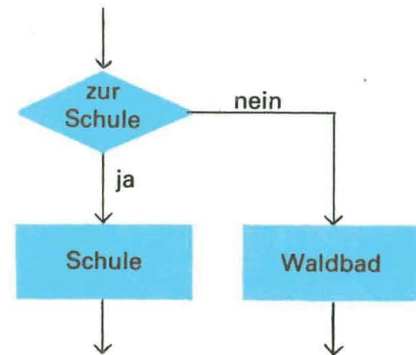
Auf einen Wegweiser treffen



Dort den Weg nach rechts wählen



Auswahl



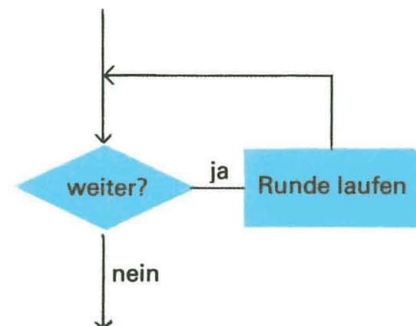
Runden im Stadion laufen



Eine weitere Runde laufen

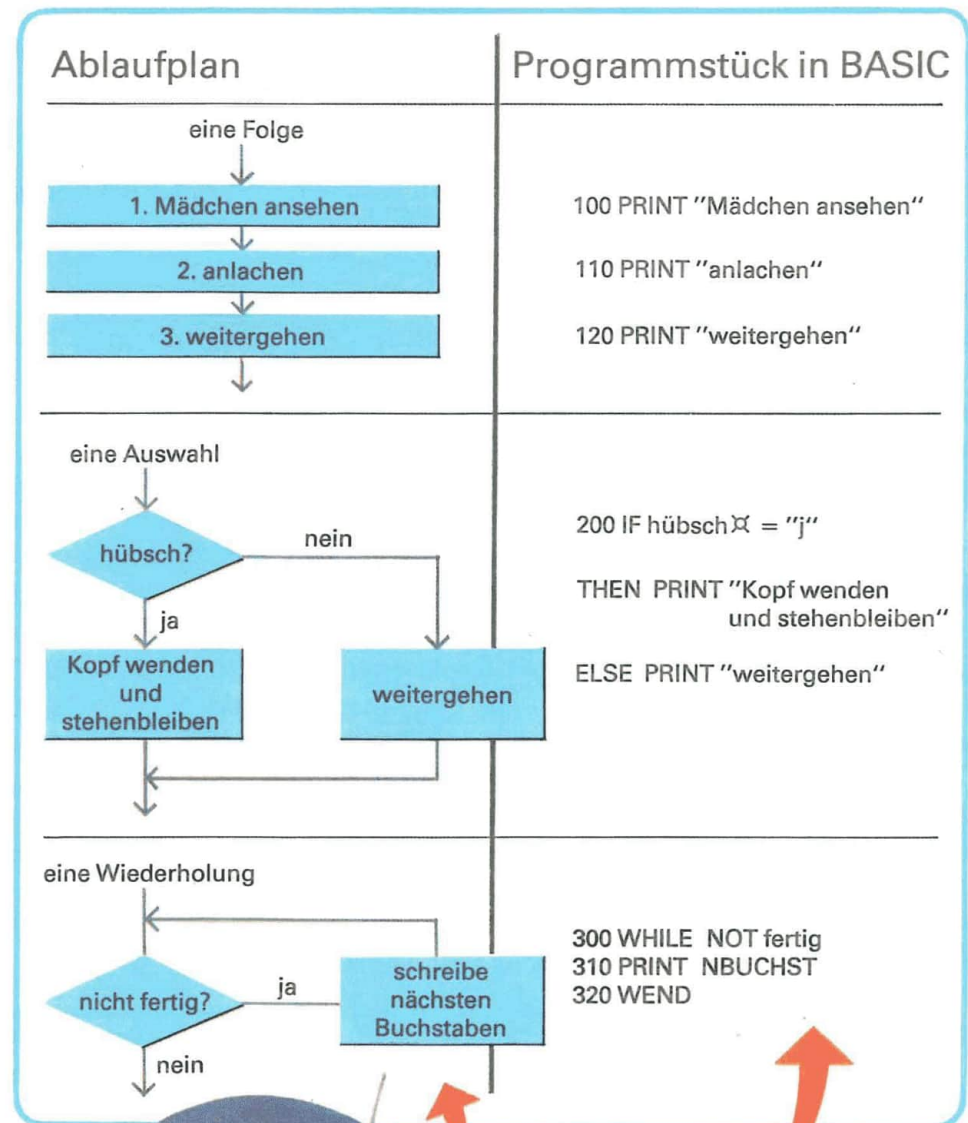
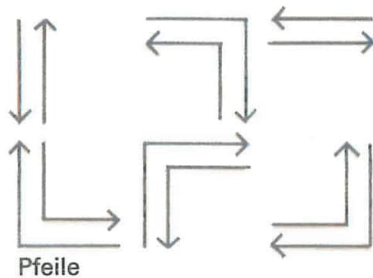


Wiederholung





Das sind die Bauteile  
eines Ablaufplanes



*Dieser Ablauf  
wird so programmiert.*

Richtung an, in der sie aufeinanderfolgen. Bei einer Auswahl verzweigt sich der Algorithmus in zwei Richtungen, es gehen zwei Pfeile von einem Kästchen aus. Doch es darf nur in eine Richtung gegangen werden. Welche Richtung gewählt wird, hängt von einer Bedingung ab: der Antwort auf die Frage im Verzweigungskästchen.

Der dritte Programmbaustein ist die *Wiederholung*. Ein Beispiel dafür: Das Mädchen kauft sich für ihr Taschengeld nicht nur ein Eis, sondern mehrere – solange das Geld reicht oder der Appetit anhält. Der Junge dreht sich nicht nur einmal in seinem Leben nach einem Mädchen um, sondern vielleicht nach allen Mädchen, die einen Pferdeschwanz tragen und nicht größer sind als er selbst. Die Sportler laufen im Stadion ihre Runden, solange sie nicht die letzte Runde gelaufen sind. Auch die Wiederholung kann man anschaulich als Ablaufbild mit Kästchen und Pfeilen zeichnen. Das Programm durchläuft entlang der Pfeile wiederholt dieselben Kästchen, solange die Frage mit ja beantwortet wird.

Diesen Algorithmus habt ihr schon in der 1. Klasse gelernt:

#### *Aufgabe*

Schreibe ein Wort, das aus mehreren Buchstaben besteht!

*Lösung* (Algorithmus mit einer Wiederholung)

1. Schreibe den ersten Buchstaben!
2. Ist noch ein Buchstabe des Wortes zu schreiben (ist das Wort noch nicht zu Ende geschrieben)?
3. Wurde 2. mit ja beantwortet, dann schreibe den nächsten Buchstaben und gehe zu 2.!
4. Fertig.

So wird das Wort vom ersten bis zum letzten Buchstaben geschrieben.

In all diesen Beispielen werden Anweisungsfolgen so lange wiederholt, wie eine Bedingung erfüllt ist.

In Programmiersprachen gibt es Anweisungen für die Auswahl beziehungsweise für die Wiederholung mit englischen Bezeichnungen. Sie werden als

IF *b* THEN *A*

ELSE *B* (sprich: iff... Bän... äß...)

beziehungsweise

WHILE (sprich: weil) *b*

*A*

WEND

notiert. „*b*“ ist eine Bedingung, „*A*, *B*“ sind Anweisungen. Diese Anweisungen bedeuten:

IF *b* THEN *A*

Wenn die Bedingung *b* erfüllt ist, dann führe die Anweisung *A* aus. Ist sie nicht erfüllt, führe *B* aus.

ELSE *B*

WHILE *b*

Solange die Bedingung *b* erfüllt ist, führe die Anweisung *A* aus. Ist sie nicht mehr erfüllt, setze nach WEND fort.

*A*

WEND

Mit solchen Anweisungen wird der Ablauf der Programme bei ihrer Abarbeitung gesteuert.

#### **Wir merken uns:**

Folge, Auswahl und Wiederholung sind Bausteine für Algorithmen und Programme. Sie steuern den Ablauf der Verarbeitung.

Große Programme werden nicht sofort niedergeschrieben, sondern schritt- oder etappenweise ausgearbeitet. Beispielsweise zeichnet man erst Ablaufpläne des Algorithmus und prüft sie sorgsam, bevor man eine Programmzeile schreibt.

#### **Wir merken uns:**

Große Programme werden aus vielen Programmteilen aufgebaut und in Etappen fertiggestellt.

So entsteht die Lebensgeschichte großer Programme (Fachleute sprechen vom Lebenszyklus).



Wenn man im ersten Schritt des Programmierens nicht genau festlegt, was das Programm leisten soll, wird das Programm auch nicht das leisten, was man von ihm erwartet. Von der exakten Aufgabenstellung hängt – wie überall – alles weitere ab!

Das Entwickeln eines Programms von 100 000 Zeilen Umfang ist ein ähnlich schwieriges Unterfangen wie das Bauen einer Maschine aus 100 000 Teilen.

#### **Wir merken uns:**

Große, komplizierte Programme sind schwierig auszu-  
arbeiten. Man kann die Aufgabe aber teilen und  
Schritt für Schritt lösen.

## Der Computer hilft beim Programmieren

Das Anfertigen großer Programme bereitet viel Mühe und verlangt große Sorgfalt und Konzentration. Die Fachleute versuchen deshalb, für diese Arbeit immer mehr den Computer zu nutzen, um die Arbeit zu erleichtern und zu beschleunigen.

#### **Wir merken uns:**

Computer werden immer mehr benutzt, um das Pro-  
grammieren zu erleichtern.

Dafür verwendet man spezielle Werkzeuge in Gestalt von Programmen. Sie helfen beim Schreiben und Prüfen von Programmen. Eine Programmiersprache, wie BASIC, ist ein solches Hilfsmittel.

#### **Wir merken uns:**

Programmiersprachen erleichtern das Schreiben und  
Prüfen von Programmen.

Natürlich kann man dem Computer ein Programm Bit für Bit eingeben (so hat man anfangs Computer programmiert). Mit einer Programmiersprache geht

es wesentlich einfacher und schneller. BASIC ist auf Mikrocomputern weltweit verbreitet und läßt sich leicht erlernen. Sehr viele Programme sind in BASIC geschrieben. Diese Sprache gibt es jedoch in vielen Ausführungen, die sich mehr oder weniger unterscheiden; Fachleute sprechen von Dialekten. Deshalb läuft leider nicht jedes BASIC-Programm auf jedem Computer, der BASIC „versteht“. Neben BASIC gibt es eine ganze Reihe anderer, höherer Programmiersprachen, die moderner sind, zum Beispiel PASCAL.

Erfahrene Programmierer benutzen beim Schreiben, Ändern, Speichern, Drucken und Prüfen von Programmen sowie der dazugehörigen Unterlagen einen Satz zusammengehöriger Hilfsprogramme. Man nennt ihn „Werkzeugkasten“ oder „Programmierungsumgebung“. – So helfen Programme auf dem Computer dem Programmierer beim Programmieren!

## Große Programme werden in folgenden Schritten ausgearbeitet:

- 1** Was soll das Programm leisten? Was wird eingegeben, was soll herauskommen?
- 2** Wie soll das erreicht werden und das Programm aufgebaut sein? (Gefragt wird nach dem Bauplan für die Daten und die Algorithmen.)
- 3** Wie sehen das Programm und die Daten für den benutzten Computer im einzelnen aus? (Der Bauplan wird ausgeführt.)
- 4** Leistet das Programm genau das, was es soll? (Das Programm wird vor seiner Verwendung „auf Herz und Nieren“ geprüft und zur Nutzung freigegeben.)
- 5** Das fertige Programm wird benutzt. (Läuft das Programm einwandfrei oder sind noch Fehler zu korrigieren oder Verbesserungen anzubringen?)



Programme taugen nur etwas, wenn alle diese Schritte sorgfältig ausgeführt und die Zwischenergebnisse überprüft worden sind!





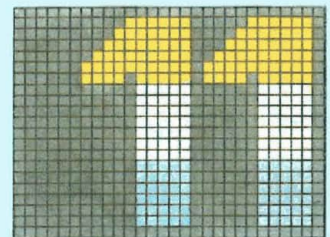
## Tips und Tests für Pfiffige

- 10.1. Im Anhang D findest du ein komplettes BASIC-Programm mit Erläuterungen und einem Ablaufplan. Lies alles sorgfältig durch und versuche, die Arbeitsweise des REZEPTOMATEN zu verstehen!
- 10.2. „Verstehen“ die heutigen Computer unsere Alltagssprache?
- 10.3. Wozu dienen Programmiersprachen?
- 10.4. Was ist Programmieren?
- 10.5. Worauf kommt es beim Programmieren an: auf *a* Sorgfalt, *b* Geduld, *c* jedes Zeichen?
- 10.6. Wie werden große Programme ausgearbeitet?
- 10.7. Werden durch Programme Daten oder Algorithmen festgelegt?
- 10.8. Nenne die wichtigsten Algorithmenbausteine und erlaüttere sie an Alltagsbeispielen.
- 10.9. Wer ist der wichtigste Gehilfe beim Programmieren eines Computers?
- 10.10. Mußt du beim Schreiben eines Wortes, also von aneinandergesetzten Buchstaben, *a* eine Folge oder *b* eine Auswahl oder *c* eine Wiederholung ausführen?
- 10.11. Sollten soeben fertiggestellte Programme als *a* fehlerfrei oder *b* fehlerhaft angesehen werden?

# 11. Bewährtes verwenden, nicht neu erfinden!

---

Hört man Computerfachleuten zu, so vernimmt man immer wieder die Wörter Hardware (sprich: hardwär) und Software (sprich: softwär). Und sie reden davon, daß gute Software gebraucht wird. Was die Experten damit meinen, wollen wir jetzt verstehen lernen.





## Harte und weiche Computerware

Damit ein Computer für uns Aufgaben erledigen kann, benötigen wir erst einmal die notwendigen Geräte und vielleicht Zusatzgeräte. Das sind alles harte Gegenstände, „harte Waren“. Sie werden mit dem englischen Wort *Hardware* benannt.

Nützlich wird die Hardware aber erst durch die auf Datenträgern gespeicherten Programme, die dazugehörigen Beschreibungen und Handbücher für die Anwender (die Anwenderdokumentation). Sie sind im Unterschied zur harten Ware der Geräte eine „weiche Ware“ – *Software*. Dieses Wort stammt ebenfalls aus dem Englischen.

Ein Taschenrechner (Hardware) nutzt uns nichts, wenn wir keine Bedienungsanleitungen und passenden Rechenvorschriften (Software) zur Hand haben. An-

dererseits können wir die beste Rechenvorschrift nicht verwenden, wenn keine Computertechnik zur Verfügung steht. Jede Computeranwendung steht auf diesen beiden Beinen: Hardware und Software.

### Wir merken uns:

Jede Computeranwendung setzt sowohl Hardware als auch Software voraus.

Zur Hardware gehören die Geräte und Zusatzgeräte, zur Software die Programme auf Datenträgern sowie ihre Beschreibungen und Bedienungsanleitungen.

Und was stellt dieses Buch über Computer dar? Ist es Software? Es enthält die Beschreibungen kleiner Programme (beispielsweise die im Anhang D), aber keine Programme auf einem Datenträger, wie Diskette oder Lochband. Damit enthält das Buch Teile von Software, aber es ist keine Software.

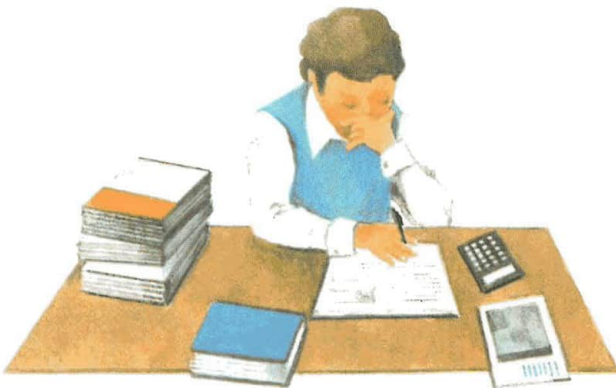


Wo man mehr über Software erfährt  
(Immer genau auf den richtigen  
Computertyp, benutzte Systemsoftware,  
verständliche Beschrei-  
bungen und Handbücher achten!)

**Software-Kataloge**  
enthalten Angaben über fertige Software.  
Ein Anruf bei Fachleuten klärt oft viele Fragen.



**Fachzeitschriften**  
berichten über neue Hardware und Software.



Bei unseren Versuchen mit dem Personalcomputer im Kapitel 8 haben wir Programme benutzt, die mit der Gerätetechnik geliefert werden und für deren wirksame Anwendung notwendig sind. Man bezeichnet sie als Systemprogramme beziehungsweise *Systemsoftware*. Im vorangegangenen Kapitel haben wir selbst kleine Programme geschrieben, die für uns als Anwender nützlich sein können. Das sind die Anwenderprogramme beziehungsweise es ist die *Anwendersoftware*.

### Wir merken uns:

Für die Anwendung eines Computers benötigen wir Systemsoftware und Anwendersoftware.

Beide Arten von Programmen sind für die Anwendung des Computers so unentbehrlich wie Benzin und Öl zum Fahren mit dem Auto.

## Weiche Ware – hartes Brot

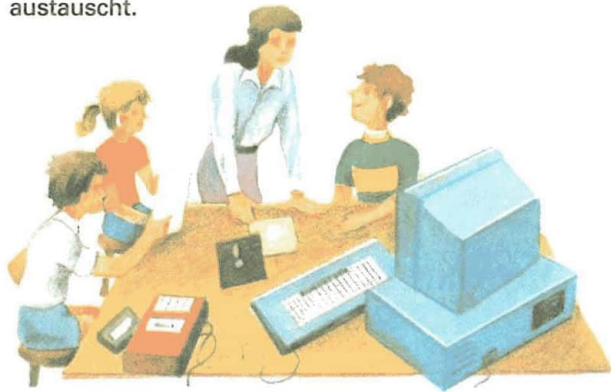
Kleine Probleme lassen sich in einer guten Programmiersprache schnell programmieren, so daß sie der Rechner für uns erledigen kann. Das haben wir mit BASIC schon ausprobiert. Aber der Computer soll auch schwierige und komplizierte Arbeiten ausführen, die Programme mit vielen tausend Anweisungen erfordern.

Lassen sich solch große Programme ebenfalls schnell und leicht schreiben?

Erfahrene Fachleute im Programmieren schreiben in einem Jahr angestrengter Arbeit nur einige tausend anwendbare Anweisungen eines Computerprogramms (mit allen notwendigen Erläuterungen und Bedienungsanleitungen, wie sie zu einer ordentlichen Software gehören). Mikrocomputer verarbeiten mehrere ...zigtausend Anweisungen in nur einer Sekunde! Was also ein Programmierer in einem Jahr mühsam



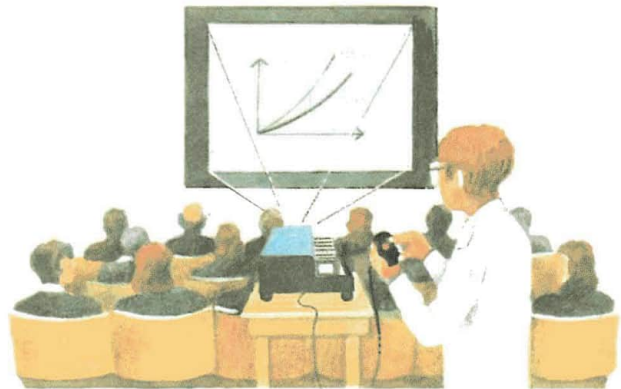
**Interessen- und Nutzergemeinschaften sowie Computer-Clubs**  
sind Treffpunkte gleichgesinnter Computerfreunde, wo man diskutiert, Erfahrungen und Programme austauscht.



**Informations- und Beratungseinrichtungen**  
geben Auskunft über nachnutzbare Software auf bestimmten Anwendungsgebieten.



**Fachtagungen**  
bieten den Teilnehmern Gelegenheit, das Neueste direkt von den Fachleuten zu erfahren und zu diskutieren.



**Messen und Ausstellungen**  
bieten Gelegenheit, neueste Computer und Computeranwendungen kennenzulernen und Software angeboten zu bekommen.





ausknobelt, erprobt und beschreibt, arbeitet der Mikrocomputer in Sekundenschnelle ab. Es ist deshalb ein hoffnungsloses Unterfangen, den „gefräßigen“ Computer nur mit eigenen Programmen „füttern“ zu wollen. Hinzu kommt bei den neuen großen Programmen der Ärger mit den Fehlern, deren Anzahl hoch ist, wenn Anfänger die Programme schreiben.

Kein Wunder, daß Software teuer ist. Vielen Computeranwendern kostet sie heute mehr als die Hardware, denn gute Software muß mühsam hergestellt werden, und sie ist schwierig zu beschaffen.

### **Wir merken uns:**

Die Software einer Computeranwendung kostet heute mehr als die Hardware.

Computer arbeiten viel zu schnell, um sie mit selbstgeschriebenen Programmen auslasten zu können. Was ist zu tun? Wo ist der Ausweg? Wie kommt man zu geeigneter Software?

## **Das Fahrrad nicht noch einmal erfinden!**

Jeder Computeranwender benötigt, wie wir bereits wissen, Systemsoftware. Der Computerhersteller liefert sie zusammen mit den Geräten. Die von ihm angebotene Anwendersoftware reicht für die speziellen Wünsche des Anwenders leider nicht aus. Für die Ausarbeitung eigener guter Software fehlen vielen Anwendern die Möglichkeiten und die Erfahrungen. Andererseits gibt es für weitverbreitete Computertypen reichlich Software; doch diese ist bei den Anwendern verteilt und daher nicht leicht zu finden.

Man muß sich also nach vorhandener Software gründlich umschauchen, sie gut auswählen und Brauchbares nachnutzen. Bewährtes soll man verwenden, das

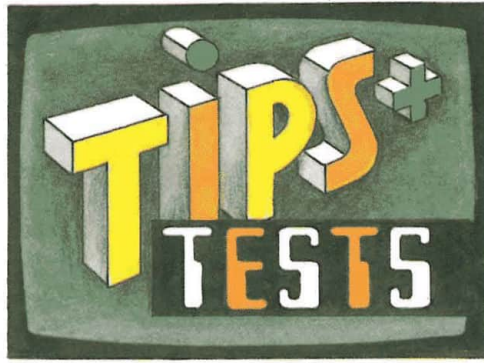
bereits Vorhandene nicht neu erfinden, auch wenn das Herstellen eigener Software vielleicht mehr Spaß macht, als die Produkte anderer zu benutzen. Zu Recht würde man sich über einen Menschen lustig machen, der das Fahrrad oder das Einmaleins noch einmal erfände...

Wer Software sucht, ist gut beraten, sich ständig zu informieren. Was es gestern noch nicht gab, kann heute schon verfügbar sein. Was gestern gut war, kann heute von Besserem überholt sein.

Wer Software trotzdem selbst ausarbeiten möchte, sollte an später denken und die Programme gut machen und sorgfältig beschreiben, daß auch andere damit einmal arbeiten können. Sorgfältiges Arbeiten erhöht die Aussichten, durch Tausch oder Verkauf zur gewünschten Software anderer zu gelangen.

Weltweit sind Forscher und Ingenieure bemüht, Vorschriften, Werkzeuge und Anleitungen zu schaffen, die helfen, Software schneller und besser auszuarbeiten, als das heute möglich ist.

Wer nicht zurückbleiben will, muß sich ständig auf dem laufenden halten, sich Neues zu eigen machen. Lernen und sich weiterbilden sind Tagesaufgaben.



## Tips und Tests für Pfiffige

- 11.1. Welche zwei Arten von Software gibt es?
- 11.2. Setzt jede Computeranwendung *a* Hardware oder *b* Software voraus?
- 11.3. Ist ein Programm, das auf einem Computer laufen kann, für das es aber keine Beschreibung und Bedienungsanleitung gibt, Software?
- 11.4. Warum gelingt es nicht, einen Computer mit selbstgeschriebenen Programmen auszulasten?
- 11.5. Wie kommt man zu Software?
- 11.6. Wenn du dich gern mit Computern beschäftigst, so such dir Partner! Gemeinsam macht es mehr Spaß, und du kommst schneller voran. Beteilige dich an Wettbewerben, Leistungsvergleichen, kleinen Messen!
- 11.7. Bringe in Erfahrung, ob es in deiner Nähe einen Computer-Club gibt und womit er sich beschäftigt!
- 11.8. Erfrage, in welcher Zeitschrift Software für dein Interessengebiet angeboten wird!
- 11.9. Stelle selbst einen Softwarekatalog für dich und andere zusammen! (Für einen Taschenrechner kann das eine Sammlung von Rechenvorschriften zur Lösung schwieriger wiederkehrender Aufgaben sein.)

## 12. Alles mit 0 und 1 beschreiben

---

Die uns vertraute Schriftsprache unterscheidet sich sehr von der Sprache des Computers mit ihren Nullen und Einsen. Doch der Computer nimmt Zeichen unserer Schrift entgegen und gibt uns bekannte Zeichen aus. Wie ist das möglich? Diese Frage wird jetzt geklärt.



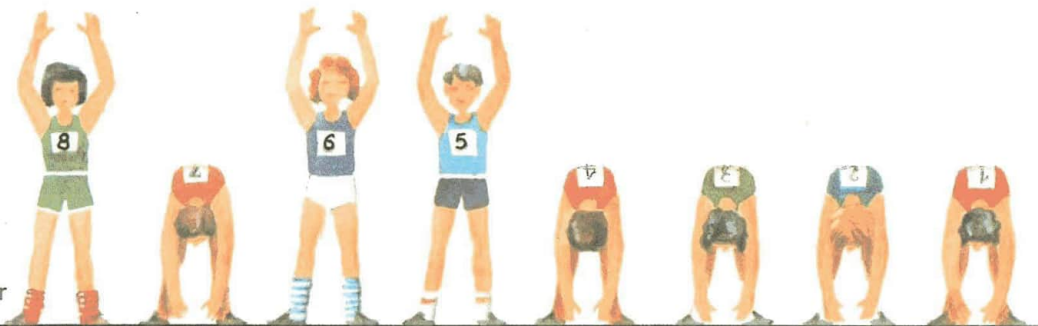


# Ablauf der Sportübung

**1.** Start



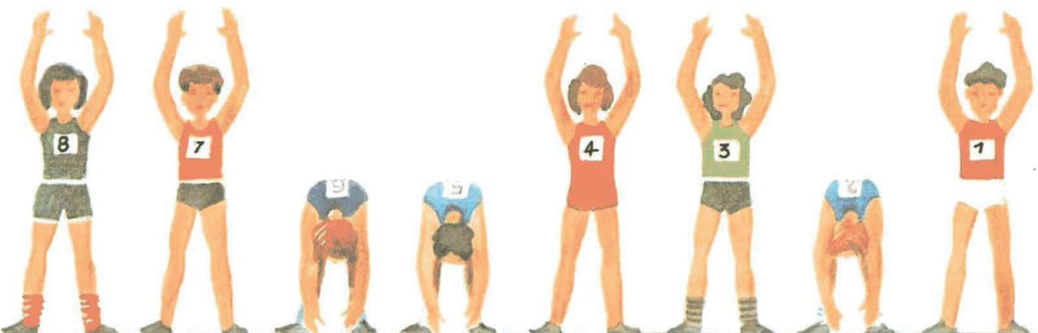
**2.** 2 Minuten später



**3.** 4 Minuten später



**4.** Ende



# Sportbilder für den Computer übersetzt

Computer arbeiten in ihrem Innern – wie wir bereits wissen – nur mit den Werten 0 und 1; sie arbeiten *binär*. Soll ein Computer Zahlen, Zeichen oder Bilder verarbeiten, so müssen sie erst in eine binäre Maschinensprache übersetzt werden. Diesen Vorgang nennt man verschlüsseln oder *kodieren* (geschrieben auch: codieren). Benutzt wird dabei eine Vorschrift, die Schlüssel oder *Kode* (geschrieben auch: Code) genannt wird.

## Wir merken uns: Computer benutzen Binärkodes.

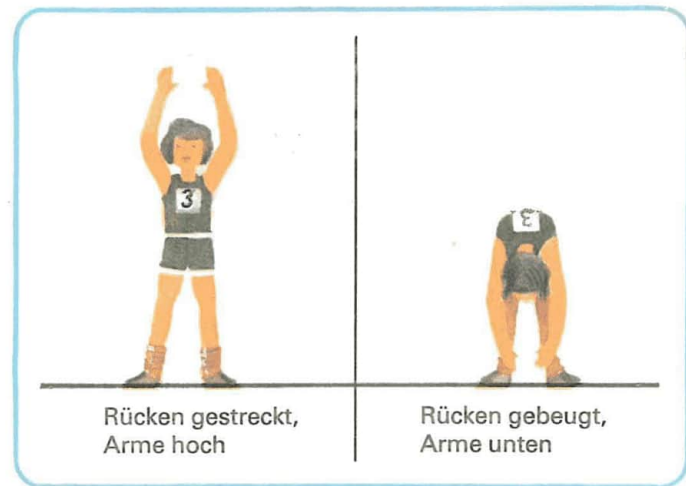
Wie man einen Alltagsvorgang binär kodiert, zeigt uns ein Beispiel.

Acht Kinder haben sich beim Sport in einer Reihe aufgestellt. Alle tragen Trainingsanzüge; an diesen sind Startnummern befestigt.

Die erste Sportübung beginnt: strecken – beugen – strecken – beugen und so weiter. Das Bild links zeigt die Gruppe in vier aufeinanderfolgenden Situationen.

Wie kann man diese Situationen für einen Computer verschlüsseln?

Wir wollen die Aufgabe in zwei Schritten lösen. Zunächst untersuchen wir das Verhalten des Sportlers mit der Startnummer 3. Danach beschreiben wir das Verhalten der ganzen Gruppe.



Auf dem Bild oben sehen wir den Sportler mit der Startnummer 3 in zwei unterschiedlichen Stellungen.

Auch die restlichen Kinder nehmen keine anderen Stellungen bei den Übungen ein.

Jede Haltung der Startnummer 3 können wir dem Computer mitteilen mit dem Satz: Sportler 3 hält den Rücken gestreckt und die Arme hoch. Oder: Sportler 3 hält den Rücken gebeugt und die Arme unten.

Diese Beschreibung ist umständlich, viel zu lang und verursacht beim Eingeben in den Computer reichlich Mühe. Wir suchen deshalb eine äußerst knappe und computergerechte Form der Mitteilung.

Da jeder Sportler nur zwei Haltungen einnimmt, können wir diese mit den Werten 0 und 1 eines Bits kodieren.

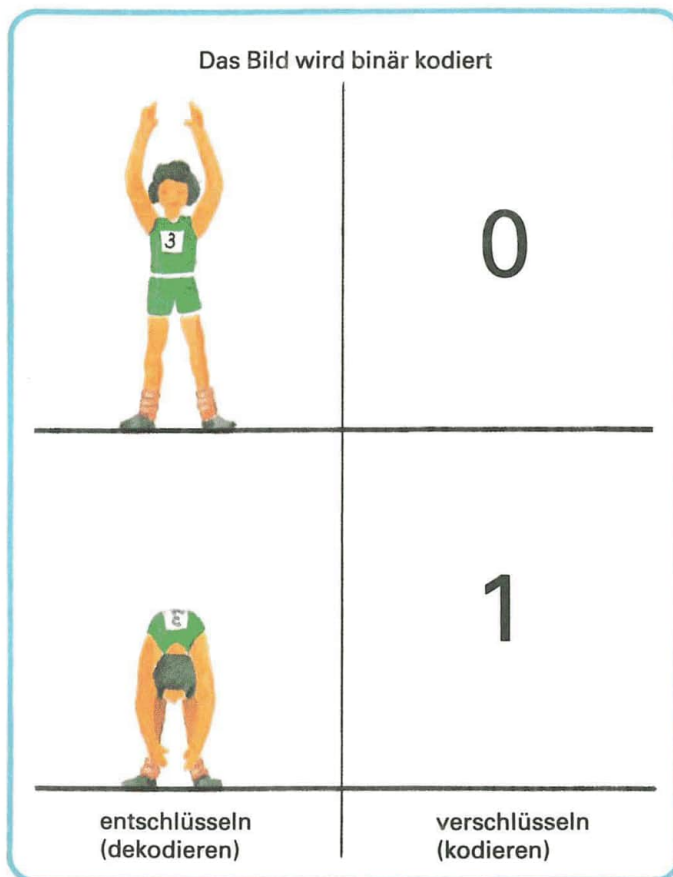
Die Haltungen des Sportlers 3 in den vier aufeinanderfolgenden Situationen des Gruppenbildes werden nach unserem Kode in die binäre Ziffernfolge 0110 kodiert.

In einem zweiten Schritt können wir das Verhalten aller 8 Kinder computergerecht kodieren. Wieviel Bits benötigen wir dafür? Natürlich 8 Bits. Das ist – wie wir wissen – 1 Byte. Wir brauchen also für die 4 Situationen insgesamt 4 Bytes.

Überprüfe das Bild, indem du es in die Haltungen zurückübersetzt, dekodierst, und mit dem Ausgangsbild auf Seite 94 vergleichst!

In die zwei Werte 0 und 1 eines Bits lassen sich sehr viele Alltagssituationen kodieren und nach Kenntnis der Vorschrift zum Verschlüsseln, des Kodes, wieder dekodieren.

Nach dieser Vorschrift (Kode) übersetzen wir das Bild vom Sportunterricht in 0 und 1. Und wir können umgekehrt aus der Binärzahl in das Bild zurückübersetzen (dekodieren).



Das Bild vom Ablauf der Sportübung, binär kodiert, sieht so aus.

| Situation       | Sportler Nummer | Bit-Nummer<br>← |
|-----------------|-----------------|-----------------|
|                 | 8 7 6 5 4 3 2 1 |                 |
| 1. Start        | 0 0 0 0 0 0 0 0 | 4 Bytes         |
| 2. 2 min später | 0 1 0 0 1 1 1 1 |                 |
| 3. 4 min später | 0 1 0 1 0 1 0 1 |                 |
| 4. Ende         | 0 0 1 1 0 0 1 0 |                 |

Ist das nicht herrlich knapp formuliert?



Jeweils zwei  
zusammenhängende Situationen,  
die sich mit 0 und 1  
verschlüsseln lassen.

Noch ein anderes Beispiel: Kodieren wir den „Hund Rex ohne Wurst“ mit 0 und den „Hund Rex mit Wurst“ mit 1. Jetzt geben wir dem hungrigen Hund (ohne Wurst) eine Wurst. Wie ist das in die binäre Sprache des Computers zu übersetzen, zu kodieren?

Aus der Eintragung in das Bit für Rex, eine 0, wird eine 1 gemacht! So einfach (und platzsparend) geht das! Solche Veränderungen von Bits sind die Grundlage für alle Rechenoperationen eines Computers. Darüber werden wir im nächsten Abschnitt noch Genaueres erfahren.

### Wir merken uns:

Ein Binärkode legt eindeutig fest, wie Bilder oder Zeichen in 0 oder 1 verschlüsselt (kodiert) und entschlüsselt (dekodiert) werden.

Für unsere Sportgruppe haben wir einen Binärkode erfunden. Andere Codes, die wir nicht zu erfinden brauchen, lernen wir jetzt kennen.

|                        |       |                    |
|------------------------|-------|--------------------|
| freut sich             | Kind  | weint              |
|                        |       |                    |
| allein                 | Kind  | nicht allein       |
|                        |       |                    |
| steht am Schwimmbecken | Kind  | schwimmt im Wasser |
|                        |       |                    |
| frei                   | Stuhl | besetzt            |
|                        |       |                    |
| Antwort                |       |                    |
|                        |       |                    |

## Ziffern und Buchstaben binär verschlüsseln

So werden ganze Dezimalzahlen binär verschlüsselt.

| Der Dezimalzahl | entspricht die Binärzahl | entspricht das An-aus-Lampenmuster |
|-----------------|--------------------------|------------------------------------|
| 0               | 0000                     |                                    |
| 1               | 0001                     |                                    |
| 2               | 0010                     |                                    |
| 3               | 0011                     |                                    |
| 4               | 0100                     |                                    |
| 5               | 0101                     |                                    |
| 6               | 0110                     |                                    |
| 7               | 0111                     |                                    |
| 8               | 1000                     |                                    |
| 9               | 1001                     |                                    |
| 10              | 1010                     |                                    |
| 11              | 1011                     |                                    |
| 12              | 1100                     |                                    |
| 13              | 1101                     |                                    |
| 14              | 1110                     |                                    |
| 15              | 1111                     |                                    |
| 16              | ?                        | ?                                  |

aus (Binärzahl 0)

an (Binärzahl 1)

Soll ein Computer Ziffern oder andere Zeichen verarbeiten, so müssen diese bei der Eingabe kodiert werden. Bei der Ausgabe dekodiert der Computer die Ergebnisse und ihre Anzeige aus der binären Darstellung in lesbare Zeichen.

Ein Beispiel für das Kodieren: Anne ist 9 Jahre, ihr Bruder Frank 10 Jahre alt. Wie werden solche Altersangaben in binäre Zahlen, also mit 0 und 1, verschlüsselt?

Aus der nebenstehenden Kodiertabelle können wir ablesen: Der Dezimalzahl 9 entspricht die Binärzahl 1001, der Dezimalzahl 10 die Binärzahl 1010.

In der Computersprache müßten wir also sagen: Anne ist binär 1001 Jahre, ihr Bruder binär 1010 Jahre

| Mit dieser Anzahl von Bits | lassen sich diese Werte verschlüsseln  |
|----------------------------|--|
|                            | So wird diese Zahl errechnet           |
| 1                          | $2 = 2 = 2^1$                          |
| 2                          | $4 = 2 \cdot 2 = 2^2$                  |
| 3                          | $8 = 2 \cdot 2 \cdot 2 = 2^3$          |
| 4                          | $16 = 2 \cdot 2 \cdot 2 \cdot 2 = 2^4$ |
| 5                          | 32 usw. $= 2^5$                        |
| 6                          | 64 $= 2^6$                             |
| 7                          | 128 $= 2^7$                            |
| 8                          | 256 $= 2^8$                            |
| .                          | .                                      |
| .                          | .                                      |
| .                          | .                                      |
| 16                         | 65536 $= 2^{16}$                       |

$2^x$  (sprich: zwei hoch x) ist eine verschlüsselte Schreibweise für ein Produkt von x Zweien



alt. Nirgends auf der Welt lebt ein Mensch, der 1010 Jahre (dezimal!) alt ist. Man kann sich also untereinander und mit dem Computer nur richtig verständigen, wenn man den benutzten Kode (hier binär) mit angibt! Sonst gibt es Mißverständnisse.

Um die dezimale Zahl 9 aufzuschreiben, benötigen wir nur 1 Dezimalstelle. Für die dezimale 10 aber schon 2 Stellen, also für die zehn Ziffern 0 bis 9 immer eine Dezimalstelle.

Und wie ist es bei der Bildung von Binärzahlen nach dem obigen Kode?

Mit jeweils einer zusätzlichen Bitstelle läßt sich die doppelte Anzahl von Ziffern verschlüsseln. Im alltäglich benutzten Zehnersystem der Dezimalzahlen ist mit jeder zusätzlichen Dezimalstelle die zehnfache Anzahl möglicher Ziffern darstellbar.

Die Zahl 16 läßt sich nicht mit 4 Bits verschlüsseln, da nach dem Kode auf Seite 98 die 16 Zahlen von 0 bis 15 schon 4 Bits belegen. Wieviel Binärstellen oder Bits benötigen wir, um die Dezimalzahl 16 zu kodieren?

Niemand würde es Spaß machen, heute einen Computer zu verwenden, der mit 0 oder 1 gefüttert werden muß. Bei den ersten Computern war es so. Die Mühe des Kodierens von Zahlen bei der Eingabe und des Dekodierens bei der Ausgabe der Ergebnisse nimmt uns bekanntlich der Computer ab. Auch der Taschenrechner tut es automatisch in Blitzesschnelle.

Wollen wir den Computer auch das Alphabet „lehren“, so müssen wir ihm einen anderen Binärkode „beibringen“. Wir könnten uns beispielsweise eine Kodetabelle ausdenken: links untereinander die Buchstaben und rechts daneben jeweils eine andere Folge von Nullen und Einsen.

Dieser neue Kode hätte freilich einen großen Nachteil: Niemand außer wir selbst verstünde ihn, kein anderer Computer könnte ihn verarbeiten, Daten ließen sich nicht mit anderen Menschen oder einer anderen Maschine austauschen.

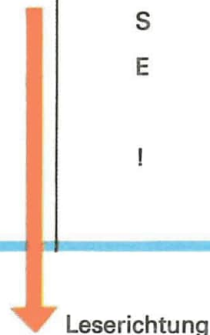
Wir verzichten deshalb lieber auf einen eigenen Kode und verwenden einen weltweit bekannten: den

7-Bit-Kode (von Computerfachleuten auch als ISO-7-Bit-Kode oder KOI-7-Bit-Kode oder ASCII-Kode bezeichnet). Wer wissen möchte, wie dieser Kode aussieht, findet ihn in Teil C des Anhangs.

Unser Ruf „1-2-3-Klasse!“ würde in der binären Zeichensprache wie aussehen?



| Zeichen Nr. | Zeichen | Binärzahl |
|-------------|---------|-----------|
| 1           | 1       | 0110001   |
| 2           | –       | 0101101   |
| 3           | 2       | 0110010   |
| 4           | –       | 0101101   |
| 5           | 3       | 0110011   |
| 6           | –       | 0101101   |
| 7           | K       | 1001011   |
| 8           | L       | 1001100   |
| 9           | A       | 1000001   |
| 10          | S       | 1010011   |
| 11          | S       | 1010011   |
| 12          | E       | 1000101   |
| 13          |         | 0100000   |
| 14          | !       | 0100001   |





Diese Folge von 14 Zeichen können wir auch als An-aus-Muster von Lampen zeichnen oder als Dezimalzahl kodieren.

So sieht der Ruf  
„1-2-3-Klasse!“  
im 7-Bit-Kode als Dezimalzahl  
und als Lampenmuster  
aus.

| Zeichen | Dezimalzahl | An-aus-Lampenmuster |
|---------|-------------|---------------------|
| 1       | 49          |                     |
| -       | 45          |                     |
| 2       | 50          |                     |
| +       | 45          |                     |
| 3       | 51          |                     |
| -       | 45          |                     |
| K       | 75          |                     |
| L       | 76          |                     |
| A       | 65          |                     |
| S       | 83          |                     |
| S       | 83          |                     |
| E       | 69          |                     |
|         | 32          |                     |
| !       | 33          |                     |

aus

an

### Wir merken uns:

Für Buchstaben und Zahlen gibt es international vereinbarte Binärkodes. Diese ermöglichen den Austausch von Daten und Programmen zwischen Computern und über Ländergrenzen hinweg.

Solche Vereinbarungen, sie heißen auch *Standards*, sind ein schönes Beispiel sachlicher und friedlicher Zusammenarbeit von Wissenschaftlern und Technikern in vielen Ländern der Erde. Sie fördern den weltweiten Austausch von Arbeitsergebnissen und den Handel.

Der 7-Bit-Kode wird auch verwendet, um Daten von dem Eingabegerät, zum Beispiel einer Tastatur, über eine Verbindungsleitung zum Computer zu übertragen und das Ergebnis an einem Ausgabegerät, zum Beispiel einem Bildschirm oder einem Drucker, auszugeben.

Die notwendigen Kodierungen erledigt der Computer für uns unmerklich, mühelos und nebenbei, wie wir es schon vom Taschenrechner gewöhnt sind.

## Und weiter verschlüsselt

Nicht nur die Daten sind im Computer binär verschlüsselt, sondern auch die Programme, also Folgen von Anweisungen an den Computer. Die Programme sind als 0-1-Muster im Speicher des Computers abgelegt. Jeder Computertyp hat seine eigene Kodiervorschrift für Anweisungen.

### Wir merken uns:

Nicht nur die Daten, auch die Anweisungen eines Programms werden im Computer binär kodiert und gespeichert.

In diesem Kapitel haben wir erfahren, daß man mit Hilfe von Codes Zeichen, Wörter und Mitteilungen ganz anders schreiben kann. Es entstehen gewisser-



Beispiel: Personalcomputer

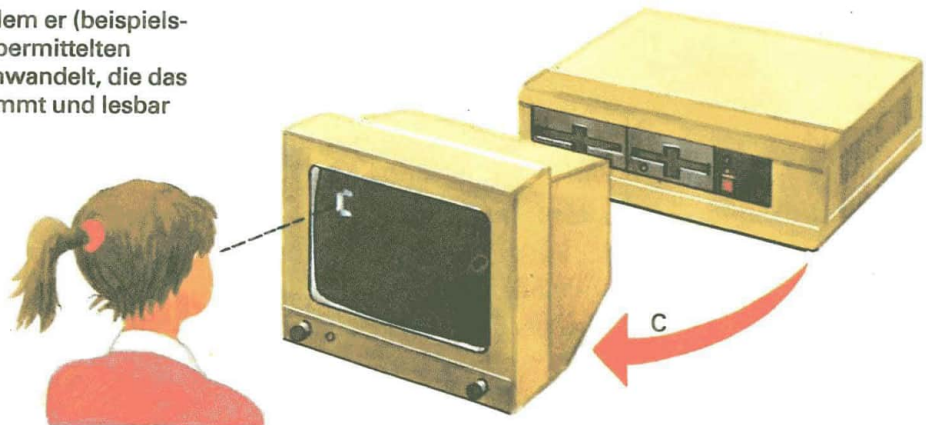
1.

Der Buchstabe C wird über die Tastatur eingegeben und der entsprechende Binärkode (1000011) über das Kabel zur Verarbeitungseinheit weitergeleitet.



2.

Der Computer reagiert, indem er (beispielsweise zur Kontrolle) den übermittelten Binärkode in eine Form umwandelt, die das Ausgabegerät entgegennimmt und lesbar ausgibt.



maßen Geheimschriften für alle, die den Kode nicht kennen.

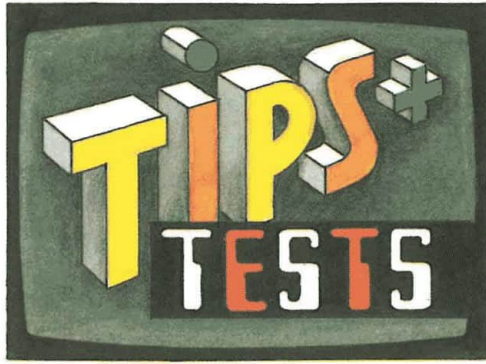
Andere, nichtbinäre Kodes werden benutzt, um im Alltag Namen und Bezeichnungen, zum Beispiel von Personen, Städten, Bezirken, Betrieben, Erzeugnissen oder Waren, abgekürzt und unverwechselbar mit Zahlen zu kennzeichnen. Dazu gehören beispielsweise die Personenkennzahl, die Postleitzahl, die Bezirksnummer, die Betriebs- und die Erzeugnisnummer, die Diagnosenummer einer Krankheit.

### **Wir merken uns:**

Mit Kodes kann man Personen und Sachen eindeutig kennzeichnen (und so Zeit sparen) oder auch Daten geheimhalten.

Zur automatischen Datenverarbeitung gehören viele Kodes. Sie sind den Computerfachleuten in Fleisch und Blut übergegangen. Ständig erfinden sie neue. Die vielen Abkürzungen und Kodes machen die Sprache der Computerfachleute für Laien allerdings schwer verständlich.





## Tips und Tests für Pfiffige

- 12.1. Was heißt binär kodieren und binär dekodieren? Was ist ein Binärkode?
- 12.2. Der Kater Leo fängt zwei Tage hintereinander eine Maus und danach sogar zwei. Verschlüsselse dieses Fangverhalten binär!
- 12.3. Das ist eine Reihe von Schülern einer Klasse.

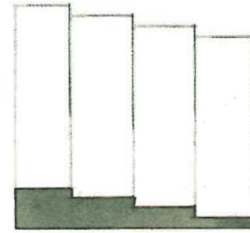


Die Lehrerin stellt eine Frage. Jetzt sieht das Bild so aus:

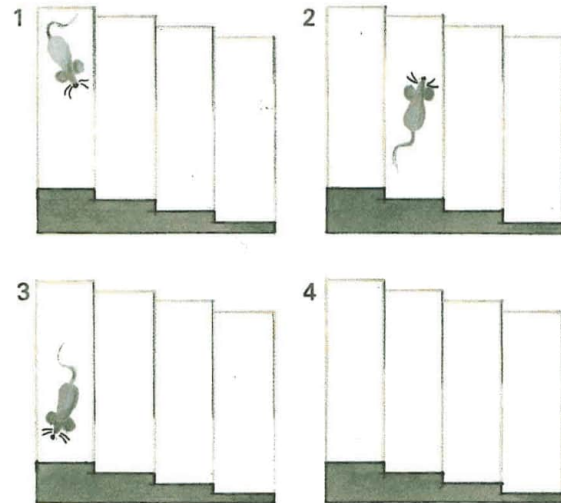


Kodiere die beiden Situationen binär!

- 12.4. Das ist eine Kellertreppe mit vier Stufen, von oben gesehen.



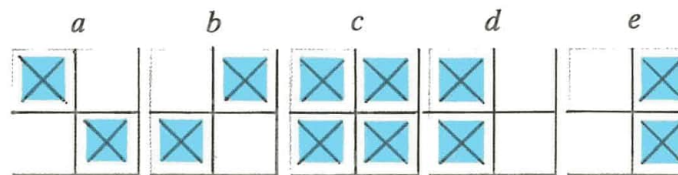
Peter schaut viermal in den Keller und sieht manchmal eine Maus auf einer der Stufen.



Kodiere binär, was Peter gesehen hat!

- 12.5. Was der Kater Leo (Aufgabe 12.2.) nunmehr in vier Wochen an Mäusen gefangen hat, gibt folgende Binärzahl an: 001000111010. Wieviel Mäuse sind das? (Benutze den Kode, der in der Antwort 12.2. verwendet wird.)
- 12.6. Wieviel Bits benötigt man, um die Zahlen 0 bis 9 beziehungsweise die Großbuchstaben A bis Z zu kodieren?
- 12.7. Mit den 5 Fingern einer Hand wird gewöhnlich von 1 bis 5 gezählt. Auch von 0 bis 5 ist möglich (0 als kein Finger gestreckt). Bis zu welcher Zahl könnte man bei binärer Kodierung (Finger gestreckt oder gebeugt) zählen?

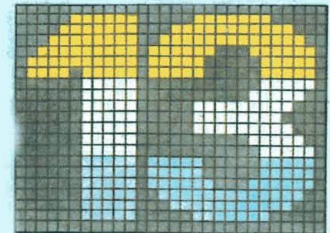
- 12.8. Der Schüler Ottokar schreibt einen Bericht über seine Schule. In diesem Bericht kommen die Worte „meine Schule“ 20mal vor. Wieviel Schreibarbeit (in Anzahl von Buchstaben) hätte er sich erspart, wenn er statt „meine Schule“ abgekürzt „mS“ geschrieben hätte?
- 12.9. Verschlüssele die Strickmuster *a* bis *e* im Bild unten in Binärzahlen (jeweils von links nach rechts und dann von oben nach unten)!
- 12.10. Vom Sportstadion kommt eine Meldung im 7-Bit-Kode:  
1010100 1001111 1010010 0100001  
Was bedeutet sie?
- 12.11. Verschlüssele deinen Vornamen und dein Geburtsjahr mit Hilfe der Tabelle in Teil C des Anhangs!
- 12.12. Am polizeilichen Kennzeichen läßt sich erkennen, wo ein Auto angemeldet wurde. Ermittle diesen Kode und nutze ihn!
- 12.13. Wie werden im Computer Daten und Programme gespeichert?
- 12.14. Erfinde einen Geheimkode nach folgendem Rezept: Gib dem 1. Zeichen in der Kodetabelle des Anhangs, Teil C, den Binärkode für das 2. Zeichen, dem 2. Zeichen den Kode für das 3. und so weiter, dem letzten Zeichen den Kode des 1. Zeichens. Probiere diesen Kode zum Verschlüsseln und Entschlüsseln (zum Beispiel von Namen)!
- 12.15. Bastle dir einen Kodeschieber aus Pappe, mit dem du Zeichen Binärziffern zuordnen kannst!
- 12.16. Versuche mit deinen Freunden folgendes Kodierspiel mit 3, 5, 7, ... Teilnehmern! Zuerst einigen sich alle auf einen bestimmten Kode (zum Beispiel den ISO-7-Bit-Kode in Teil C des Anhangs), dann nimmt jeder einen Zettel und einen Stift. Nun setzen sich alle in eine Reihe oder rund um einen Tisch. Der erste Teilnehmer wählt irgendeinen kurzen Ausspruch (oder ein Wort), schreibt ihn (oder es) auf den Zettel und flüstert ihn (oder es) seinem Nachbarn zu. Dieser kodiert das Gehörte, schreibt das Ergebnis auf den Zettel und sagt es seinem nächsten Nachbarn. Dieser muß den Text dekodieren, das Ergebnis auf seinen Zettel schreiben, es weiterflüstern und so fort. Am Schluß lesen der erste und der letzte Spielteilnehmer vor, was auf ihren Zetteln steht. Stimmt beides überein, haben alle den Kode richtig benutzt. Stimmt es nicht überein, beginnt das Detektivspiel: Wer war der Täter? Anhand der Zettel läßt sich ermitteln, wer falsch kodiert oder dekodiert hat.
- 12.17. Veranstaltet einen Kodierwettbewerb: Wer macht die wenigsten Fehler beim Kodieren eines Satzes, wie zum Beispiel diesem: Welcher Esel hat den Elefant' auf den Hof gesandt? Verleiht dem Sieger den Titel eines Kodiergroßmeisters.



# 13. Im Computer passiert Erstaunliches

---

Wir werfen jetzt einen neugierigen Blick in das seltsame technische Innenleben eines Computers. Wie führt ein Computer automatisch ein Programm aus? Wie verarbeitet er Bits zu anderen Bits? Was ist zu erkennen, wenn wir einen modernen Computer immer mehr auseinandernehmen und immer tiefer in ihn hineinschauen? Wie sieht sein Inneres aus?





## Vier Schulklassen und ein Dirigent spielen Computer



## Vier Schulklassen probieren ein neues Spiel

Stell dir vor: Eines Tages beschließen vier Schulklassen, gemeinsam Computer zu spielen. Die eine Klasse spielt die Dateneingabe, die andere die Datenausgabe. Die dritte sorgt für das Speichern von Daten, zum Beispiel auf Zetteln in Regalfächern. Die vierte Klasse muß Daten verarbeiten, also rechnen, vergleichen, transportieren, sortieren und anderes. Reichen diese vier Tätigkeiten aus, um Programme automatisch ganz genau abzuarbeiten?

Man muß kein Computerkonstrukteur sein, um zu erkennen: So klappt es nicht. Die Schüler würden hin und her rennen, keiner wüßte genau, was er als nächstes zu tun hätte. Es fehlte ein guter Organisator, ein Direktor oder, sagen wir, ein Dirigent, der dafür sorgt, daß ein Zusammenspiel gelingt.

Die „Eingabeklasse“ nimmt die richtigen Daten von außen entgegen. Dann schreibt die „Speicherklasse“ diese Daten sorgfältig auf Zettel und bewahrt sie auf.

Die „Verarbeitungs-klasse“ erhält danach genaue Anweisungen, von welchen Zetteln sie Daten lesen und verarbeiten und auf welche Zettel sie die Ergebnisse notieren soll. Die „Ausgabeklasse“ muß die Ergebnisse aus der „Speicherklasse“ rechtzeitig erreicht bekommen.

Der Dirigent des ganzen Spiels muß entsprechend dem vorliegenden Arbeitsprogramm genau sagen, was jede Klasse als nächstes zu erledigen hat. Auch muß er den richtigen Takt angeben, an den sich alle halten müssen.

Damit die Arbeit ganz gleichmäßig und auch schnell abläuft, wird sich der Dirigent nach einem *Taktgeber*, zum Beispiel dem Sekundenzeiger einer Uhr, richten.

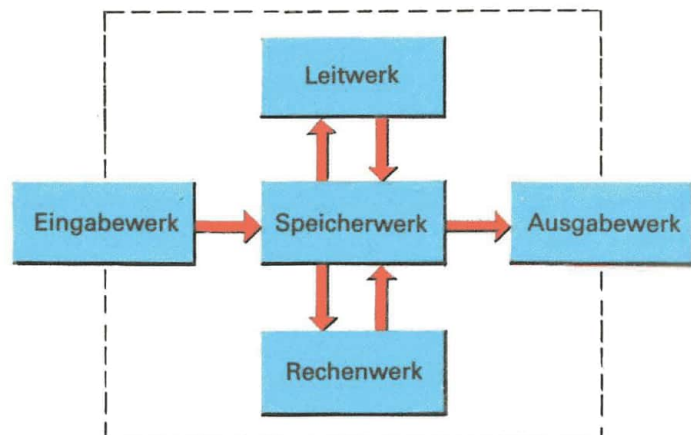
In einem wirklichen Computer führt die Tätigkeiten des Dirigenten ein zusätzliches Computerteil aus, das Steuer- oder *Leitwerk*.

### Wir merken uns:

Das Leitwerk eines Computers sorgt für die automatische Abarbeitung der eingegebenen und gespeicherten Programme.

Mit dem Einschalten des Computers nimmt das Leitwerk seine Arbeit auf und setzt sie pausenlos fort bis zum Abschalten oder bis zu einer Störung. Wer verstehen will, wie ein moderner Computer Programme automatisch abarbeitet, muß die Arbeitsweise des Leitwerks verstanden haben.

Das sind die wichtigsten Teile eines Computers zum Abarbeiten von Programmen

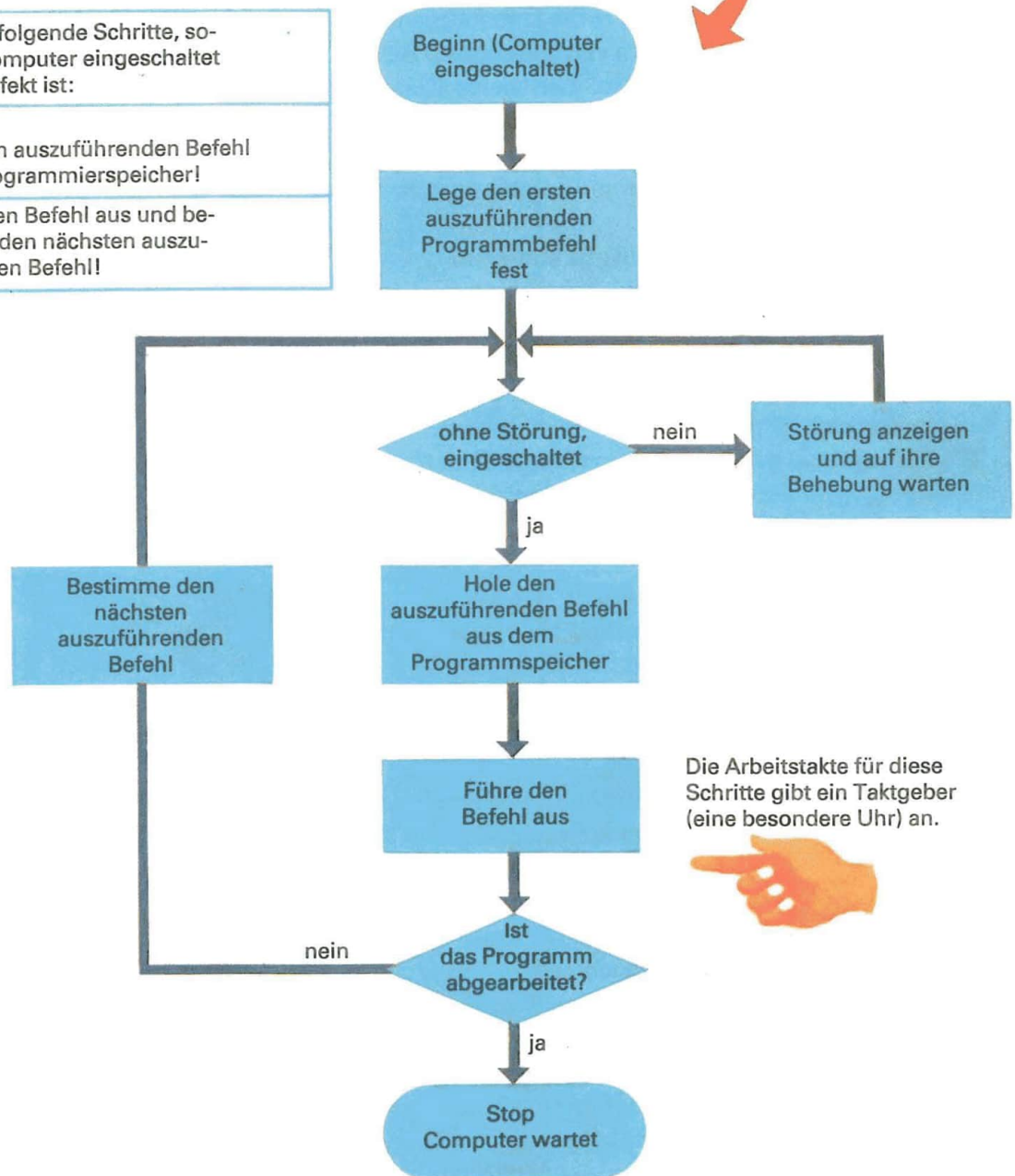


So steuert das Leitwerk  
automatisch den Programmablauf  
im Computer.

Wiederhole folgende Schritte, so-  
lange der Computer eingeschaltet  
und nicht defekt ist:

- 1.** Hole den auszuführenden Befehl  
vom Programmierspeicher!
- 2.** Führe den Befehl aus und be-  
stimme den nächsten auszu-  
führenden Befehl!

Oder etwas genauer:



Die Arbeitstakte für diese  
Schritte gibt ein Taktgeber  
(eine besondere Uhr) an.





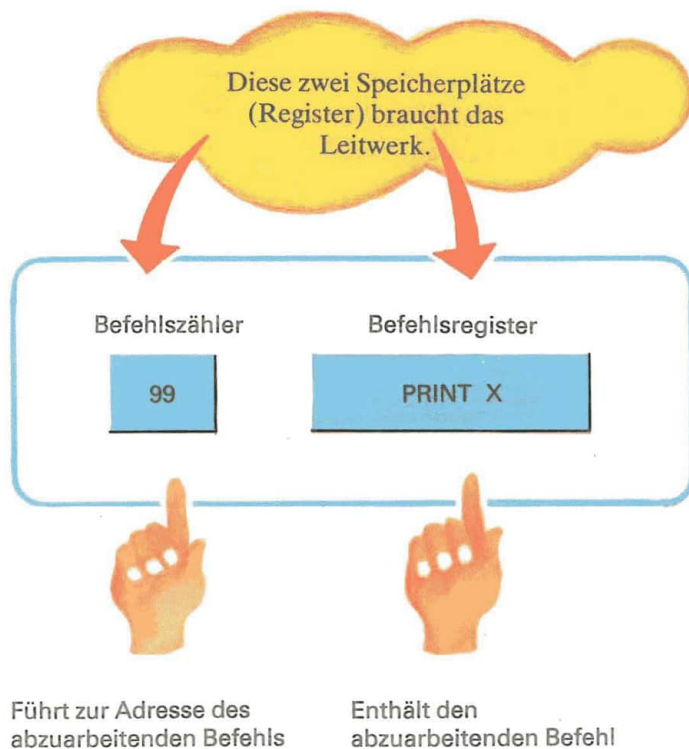
## Dem Dirigenten zugeschaut

Was muß das Leitwerk tun? Welchen Algorithmus muß es beim Steuern verwenden, um die Programmbefehle im Speicher abzuarbeiten? (Versuch einmal selbst, dieses Rätsel zu lösen!)

Antwort gibt das Bild links; wir prägen es uns ein. Der gezeigte einfache Algorithmus (Bildteil links oben) garantiert, daß der Computer pausenlos fleißig die Programme abarbeitet.

Wer das Buch bisher aufmerksam gelesen hat, wird mit der Antwort im Bildteil links oben aber nicht voll zufrieden sein. Jeder Programmbefehl steht an einer bestimmten Adresse im Speicher und benötigt einen bestimmten Speicherplatz. Wir können die Arbeit des Leitwerks genauer beschreiben (rechter Bildteil).

Auf welchem speziellen Speicherplatz der nächste



auszuführende Befehl steht, verrät uns der *Befehlszähler*. Wenn der Computer seine Arbeit aufnimmt, wird automatisch die Adresse des ersten auszuführenden Befehls eingestellt (oft ist das eine 0). Der Satz (teil) „bestimme den nächsten auszuführenden Befehl“ aus dem Bild kann nun genauer geschrieben werden, nämlich als „berechne den neuen Stand des Befehlszählers und trage ihn in den Befehlszähler ein“.

## Sprünge verschiedener Art

Wie ermittelt das Leitwerk den neuen Stand des Befehlszählers der auszuführenden Programmbefehle? Wir müssen das begreifen, um die Arbeitsweise des Leitwerks mit dem Demonstrationscomputer (Anhang E) nachspielen zu können.

Die Berechnung erfolgt unterschiedlich für die bekannten Bausteine der Ablaufsteuerung von Programmen: die Folge, die Auswahl und die Wiederholung.

Bei einem Programm, das aus einer Folge besteht, wird einfach immer die Adresse des folgenden Befehls, beispielsweise 0, 1, 2, 3, 4, 5 und so weiter, in den Befehlszähler eingetragen. Der Computer marschiert gewissermaßen geradeaus durch das Programm, von einer Nummer des Befehls zur nächsthöheren.

Bei den anderen Bausteinen passieren (manchmal) Sprünge. In Programmstücken, die aus einer Auswahl beziehungsweise einer Wiederholung bestehen, springt der Befehlszähler zuweilen zu einem höheren beziehungsweise niedrigeren Wert. Anweisungen werden so übergangen (bei einer Auswahl), oder es wird auf frühere Anweisungen zurückgesprungen (bei einer Wiederholung). Beispielsweise können bei einer Auswahl im Befehlszähler nacheinander die Adressen 1, 4 stehen (2 und 3 werden ausgelassen), bei einer Schleife kann nach dem Zählerstand 3 eine 1 folgen.

Die beiden Programmbeispiele für den „Eßroboter“ enthalten Schleifen. In dem Eßprogramm A wird nach dem 3. Befehl wieder auf Befehl 2 zurückgesprungen.

## GOTO A und IF b THEN A in Programmierbeispielen

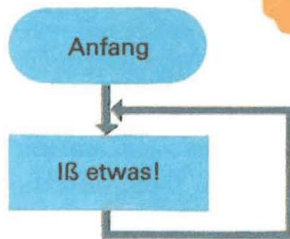
### 1. Das Eßprogramm eines Roboters, der nie satt wird

... als Arbeitsvorschrift

1. Beginne!
2. Iß etwas!
3. Setze fort bei 2.! (GOTO 2!)

unbedingter Sprung

... als Bild



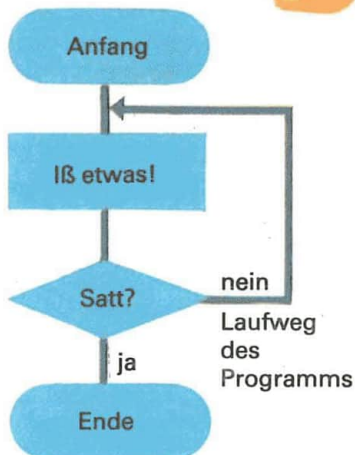
### 2. Das Eßprogramm eines Roboters, der satt wird

... als Arbeitsvorschrift

1. Beginne!
2. Iß etwas!
3. Bist du satt?  
Wenn nein, gehe zu 2.!  
Wenn ja, gehe zu 4.!
- (IF [falls] nicht satt, THEN [dann] 2.)
4. Ende

bedingter Sprung

... als Bild



Das ist ein *unbedingter Sprung*, in Programmiersprachen meist als GOTO geschrieben (sprich: go tu; englisch: gehe zu).

Welche Adressen stehen bei der Abarbeitung im Befehlszähler? 1, 2, 3, 2, 3, 2, 3... Das Programm ist in eine endlose Schleife geraten, aus der es allein nicht wieder herauskommt. Das Programm verlangt es so. Der Roboter hört nie auf zu essen und ißt alles, was er erreichen kann. Er arbeitet jetzt so wie der Breikochautomat in dem Grimmschen Märchen, der mit dem Kochen nicht aufhört.

Im Eßprogramm A des Roboters wird jedoch ein Ende erreicht, indem eine Bedingung geprüft wird (ob der Roboter satt ist). Wird diese bejaht (der Roboter ist satt), kommt der Algorithmus zum Ende. Wird jedoch diese Bedingung verneint (der Roboter ist noch nicht satt), geht der Algorithmus zurück zum Eßbefehl. Das ist eine spezielle Auswahl, ein *bedingter Sprung*, in Programmiersprachen oft geschrieben als IF b THEN A (sprich: iff ... Bän). Das sind englische Wörter; sie bedeuten hier: Wenn die Bedingung b erfüllt ist, dann führe die Anweisung A aus. Ist die Bedingung nicht erfüllt, dann tue nichts.

Ist der Eßroboter gemäß Program B beispielsweise nach dem zweiten Essen satt, so sind die aufeinanderfolgenden Befehlszählerstände 1, 2, 3, 2, 3, 4. Hier wurde – wie in dem Grimmschen Märchen – endlich doch die Bedingung erfüllt, welche die Arbeit des Automaten beendet.

An Großcomputern älterer Bauart wurden die Befehlszählerstände durch Lämpchen angezeigt. Die Aufeinanderfolge der Sprünge des Befehlszählers beim laufenden Programm konnte so sichtbar gemacht werden.

#### Wir merken uns:

Mit unbedingten und bedingten Sprunganweisungen wird der feste Programmablauf von einer Adresse zur nächsthöheren verändert. Die Sprungbedingungen hängen meist von Zwischenergebnissen ab.



Die im Programm festgelegte Ablaufsteuerung eines Programms und ihre Ausführung durch das Leitwerk sind ein weiterer wichtiger Schlüssel, um die Arbeitsweise eines Computers genau zu verstehen. Ihre Erfindung und spätere technische Verwirklichung waren ein Meilenstein in der Geschichte des Computers.

Wenn du alles verstanden hast, bist du in der Lage, den Demonstrationscomputer aus Karton in Teil E des Anhangs zum Computerspielen, für Programmknobeleien und zum Programmprüfen zu verwenden.

## Bits ein bißchen verändern

Wie kann man mit Bits rechnen? Wie kann man Binärzahlen mit Hilfe technischer Geräte addieren, multiplizieren, vergleichen und so weiter? Wie lassen sich Operationen mit Bits ausführen?

Es gibt keine einfachen technischen Geräte, die das ohne weiteres können. Aber es gibt einen Ausweg. Komplizierteste mathematische Berechnungen lassen sich aus den einfachen Rechenoperationen „plus“, „minus“, „multipliziert mit“ und „dividiert durch“ aufbauen. Ebenso komplizierte binäre Operationen aus einfachen.

### Wir merken uns:

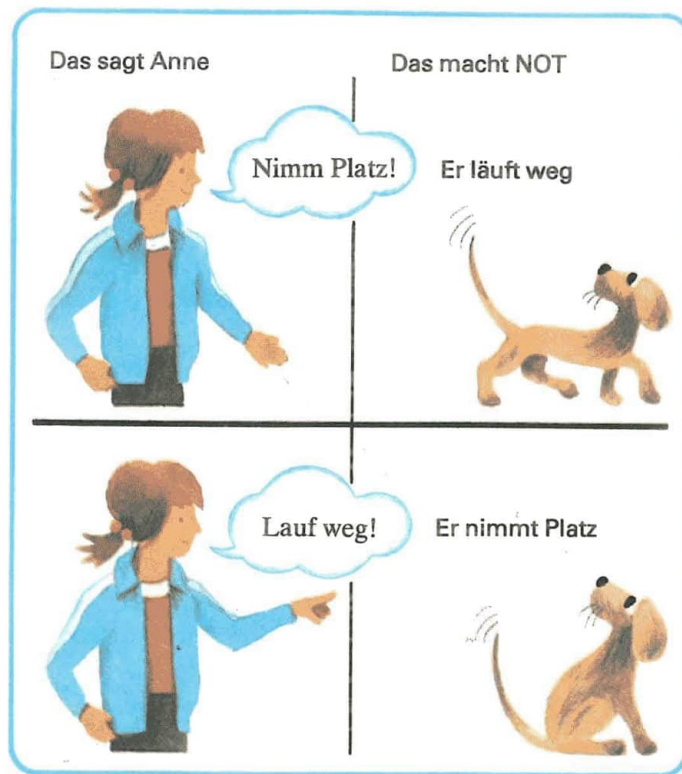
Komplizierte Operationen mit binären Zahlen lassen sich aus einfachsten binären Operationen aufbauen, die sich mit technischen Bausteinen verwirklichen lassen.

(Diese Bausteine werden in der Technik auch als Bauglieder oder Bauelemente bezeichnet.)

Solche einfachen binären Operationen sind sogar leichter zu verstehen als das Einmaleins, das wir in der ersten Klasse mühsam gelernt haben.

## NOT, der unfolgsame kleine Hund

Anne hat einen kleinen, ganz jungen Hund mit dem seltsamen Namen NOT (sprich: nott). Er macht stets genau das Gegenteil von dem, was sie ihm sagt.



Verschlüsseln wir „Platznehmen“ mit 0 und „Weglaufen“ mit 1, so können wir das letzte Bild in eine Tabelle übersetzen.

| Eingang   | Ausgang   |
|-----------|-----------|
| Anne sagt | NOT macht |
| 0         | 1         |
| 1         | 0         |



Verstehen wir Annes Befehl als binären Eingang und das, was ihr Hündchen macht, als binären Ausgang, so können wir die Eingabe, die Verarbeitung und die Ausgabe als Bild zeichnen (so wie wir es von unserem EvA-Modell aus Kapitel 4 kennen).



Damit haben wir schon die einfachste Operation oder den einfachsten Verarbeitungsschritt eines Bits kennengelernt: die Nichtoperation (englisch: not = nicht). Sie verwandelt einen binären Eingang in einen Ausgang mit entgegengesetzten Werten, wie es die Tabelle auf Seite 111 zeigt. Einfacher geht's wirklich nicht!

Solche Nichtoperationen können technisch leicht in Bauelementen verwirklicht werden.

Jetzt lernen wir noch zwei kompliziertere Operationen kennen, die *zwei* Eingänge zu *einem* Ausgang verarbeiten. Damit werden wir bald in der Lage sein, Bits zu addieren.

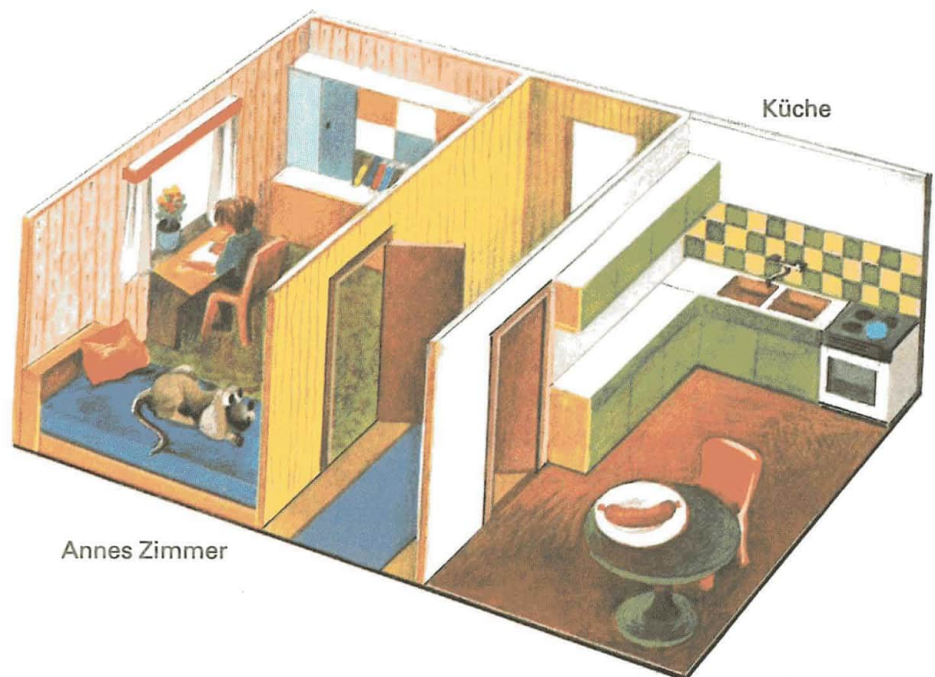
## Der Fall Bocki

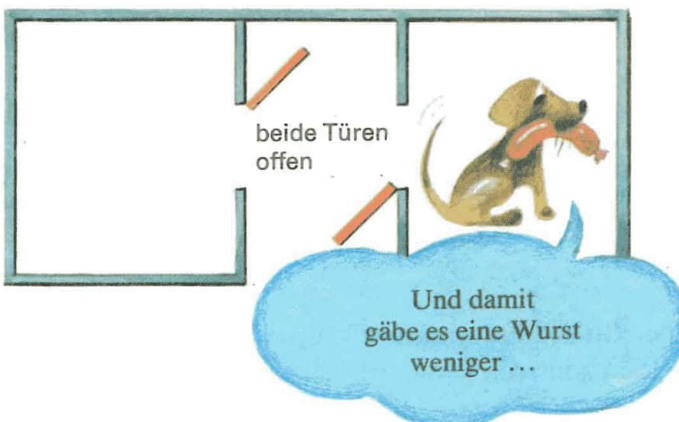
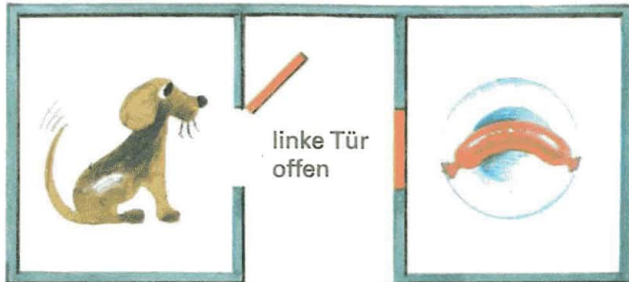
Annes Freundin Sabine besitzt ebenfalls einen Hund; keine Bockwurst ist vor ihm sicher. Er wird deshalb Bocki gerufen.

Sabine hat eine Bockwurst gekauft und auf den Küchentisch gelegt. Sie hat Bocki in ihr Zimmer gebracht. Zwischen ihrem Zimmer und der Küche befinden sich zwei Türen.

Kann Bocki die Wurst holen und auffressen? Untersuchen wir die Situation genau!

Ob Bocki an die Bockwurst gelangen kann, hängt von der Stellung der beiden Türen ab. Wir zeichnen alle Möglichkeiten auf.





Bocki kann die Wurst fressen, wenn sowohl die linke als auch die rechte Tür offen ist. Verschlüsseln wir die Stellung jeder Tür so:

Tür zu 0

Tür auf 1

und das „Schicksal“ der Wurst wie folgt:

Hund nicht in Küche (Wurst nicht in Gefahr) 0

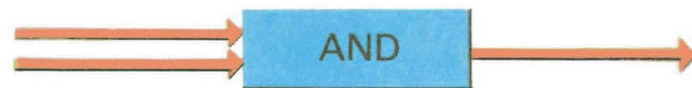
Hund in Küche (Wurst in Gefahr) 1

so können wir die vier Situationen auch tabellenförmig erfassen:

| Eingang   |            | Ausgang         |
|-----------|------------|-----------------|
| linke Tür | rechte Tür | Wurst in Gefahr |
| 0         | 0          | 0               |
| 1         | 0          | 0               |
| 0         | 1          | 0               |
| 1         | 1          | 1               |

Die Tabelle beschreibt eine binäre Operation, welche aus zwei Eingängen (den beiden Türstellungen) einen Ausgang (ob die Wurst in Gefahr ist) errechnet. Diese Operation wird mit AND bezeichnet (englisch: and [sprich: änd] = und). Das Ergebnis einer AND-Operation hat den Wert 1, wenn beide Eingänge 1 sind, sonst ist der Ergebniswert 0.

Ein Baustein, der diese Operation ausführt, wird so gezeichnet:



Auch diese Operation ist einfacher als die Addition einer Binärstelle!

Das Ergebnis unserer Untersuchung im Fall Bocki lautet: Mit einer AND-Operation berechnen wir das Schicksal der Wurst aus der Stellung der Türen.

Jetzt werden wir eine weitere binäre Operation kennenlernen. Damit verfügen wir dann über genügend Bausteine, um Binärziffern addieren zu können.



# Nur einer darf aufs Rad!

Auf Vaters Rennrad hat nur ein Fahrer Platz. Manchmal will Anne mit dem Rad fahren, ein andermal ihr Bruder Hannes. Manchmal wollen beide aufsteigen.



Da sie sich jedoch nie einigen können, wer das Rad erhält, bekommt es dann keiner von beiden.  
 Wir verschlüsseln:  
 Kind will nicht radfahren 0  
 Kind will radfahren 1  
 Alle möglichen Ausgangssituationen oder Eingänge schreiben wir auf.

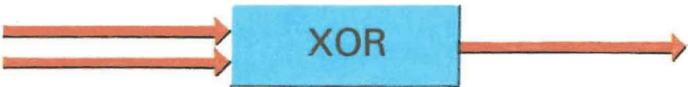
| Anne | Hannes |
|------|--------|
| 0    | 0      |
| 0    | 1      |
| 1    | 0      |
| 1    | 1      |

Ob das Rad besetzt wird oder nicht, verschlüsseln wir so:  
 Rad besetzt 1  
 Rad nicht besetzt 0.

Die Wertetabelle der Eingänge (links unten) können wir damit um die Ergebnisspalte ergänzen.

| Eingänge |        | Ausgang |
|----------|--------|---------|
| Anne     | Hannes | Rennrad |
| 0        | 0      | 0       |
| 0        | 1      | 1       |
| 1        | 0      | 1       |
| 1        | 1      | 0       |

Die letzte Zeile besagt: Das Rad kann nicht von zwei Personen besetzt werden.  
 Fachleute nennen diese Operation, die zwei Eingangsbits in ein Ausgangsbit verwandeln, *exklusives ODER*, abgekürzt *XOR*, aber auch *Antivalenz*. Ein solcher Baustein wird so gezeichnet:



Auch diese binäre (oder logische) Operation ist immer noch einfacher als die Addition von zwei Bits!

## Wir konstruieren einen Bit-Addierer

Der Bit-Addierer eines Computers, der zwei Binärstellen addieren kann, muß sich nach einer Vorschrift oder Regel richten, die folgende Tabelle zeigt:



Addieren von zwei Bits  
als Formel

als Tabelle

0 + 0 = 0  
0 + 1 = 1  
1 + 0 = 1  
1 + 1 = 10

| 1. Summand<br>A | 2. Summand<br>B | Summe |
|-----------------|-----------------|-------|
| 0               | 0               | 0     |
| 0               | 1               | 1     |
| 1               | 0               | 1     |
| 1               | 1               | 10    |

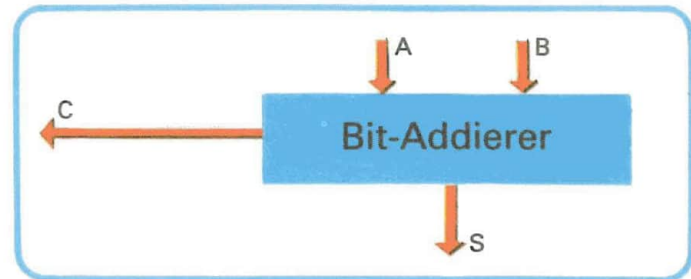
Übertrag  
C

Probleme entstehen bei der binären Addition von 1 + 1. Es ergibt sich binär 10 (oder dezimal 2). Das erfordert ein zusätzliches Bit, das *Übertragungsbit*. (Genau wie beim Addieren von zwei einstelligen Dezimalzahlen, zum Beispiel 5 + 5 = 10, entsteht ein Übertrag zur nächsten Stelle, wenn das Ergebnis größer als 9 ist. Das Ergebnis ist dann zwei Stellen lang.) Wir müssen also die rechte Ergebnisspalte in zwei Spalten (oder Ausgänge) aufteilen, da wir nur bitweise rechnen.

Addition von zwei Bits  
(A und B)  
mit Übertrag C

| 1. Summand<br>A | 2. Summand<br>B | Übertrag<br>C | Summe<br>S |
|-----------------|-----------------|---------------|------------|
| 0               | 0               | 0             | 0          |
| 0               | 1               | 0             | 1          |
| 1               | 0               | 0             | 1          |
| 1               | 1               | 1             | 0          |

Damit haben wir die tabellarische Beschreibung eines Bit-Addierers mit zwei Eingängen (den Summanden A und B) sowie zwei Ausgängen (dem Übertrag C und der Summe S). Wir zeichnen ihn so:



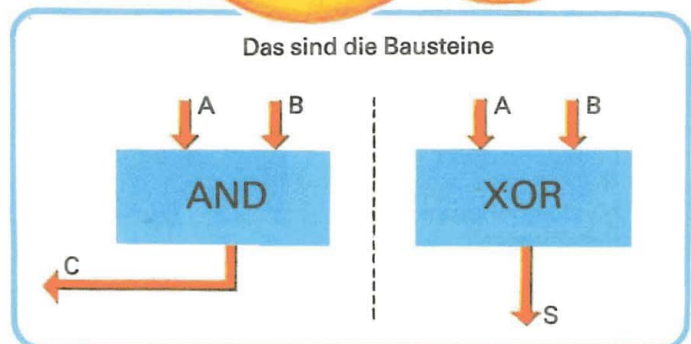
Fachleute nennen diesen Baustein auch Halb-addierer.

Durch welche der uns bekannten Operationen werden die beiden Ausgänge aus den beiden Eingängen berechnet? Versuche das zu entdecken!

Die Lösung zeigen die nächsten Bilder. Vergleiche sie mit deinem Resultat! Man nennt eine solche Verbindung von Bausteinen oder Bauelementen eine Schaltung.

Der Bit-Addierer  
und seine Bausteine

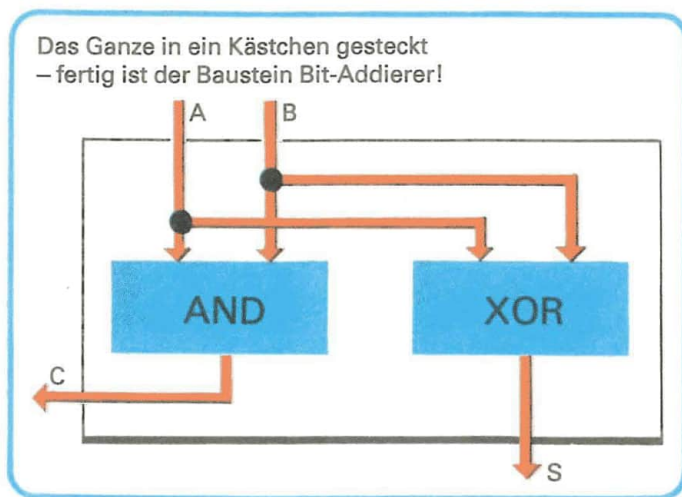
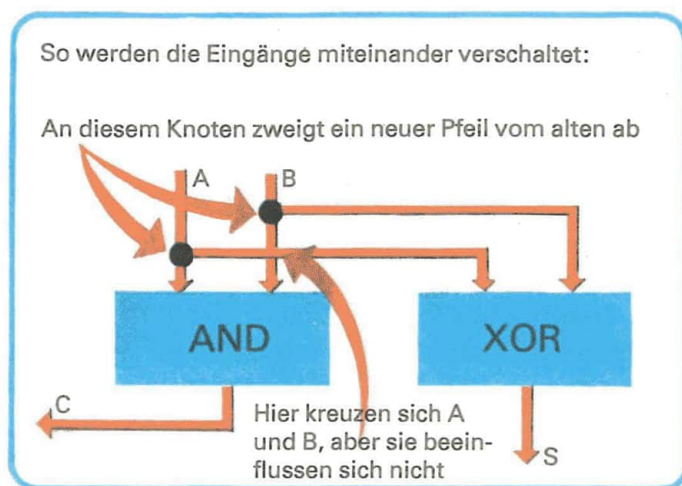
Das sind die Bausteine



Auf diese Weise werden komplizierte Operationen in einfachere binäre zerlegt und aus einfachen Operationen komplizierte zusammengebaut; die entsprechenden Bauelemente werden verschaltet (verbunden).

### Wir merken uns:

Die Befehle, die ein Computer auszuführen vermag, werden aus kleinsten binären Operationen, wie NOT, AND, XOR, zusammengesetzt.



In der Schaltalgebra, der Wissenschaft von binären Schaltungen, werden solche Aufgaben mathematisch behandelt.

## Von modernen Bauelementen eines Computers

Bisher kennen wir Schaltungen nur als Skizzen auf dem Papier. Wie aber wird ein Gerät gebaut, das die gewünschten Operationen (Funktionen) der Schaltung ausführt? Man kann es mit mechanischen Teilen wie Rädern und Stangen, mit Schaltern und Hebeln oder flüssigkeitsdurchströmten Röhren bauen. Derartige „Rechner“ sind jedoch viel zu langsam und sehr störanfällig. Deshalb benutzt man *elektronische Bauelemente*.

Elektronische Bauelemente leiten ihren Namen von den Elektronen her. Das sind kleinste (für uns nicht ohne weiteres sichtbare) Bestandteile der Materie. Wenn sie sich bewegen, fließt elektrischer Strom. Metalle, wie Kupfer und Aluminium, leiten elektrischen Strom besonders gut. Porzellan und Kunststoffe leiten ihn nicht. Es gibt jedoch auch *Halbleiter*. In ihnen können sich die Elektronen nur unter bestimmten Bedingungen bewegen. Daher eignen sich diese elektronischen Bauelemente gut zum Verändern binärer Werte, zum Schalten. Solche elektronischen Schalter haben entscheidende Vorteile: Sie arbeiten blitzschnell und fehlerfrei, sie verschleiß und ermüden nicht wie mechanische Geräte.

Um beispielsweise einen gebrauchsfähigen, sehr einfachen Computer zu bauen, benötigt man über 10 000 elektronische Bauelemente. Sie alle müssen mit Leitungsbahnen richtig verbunden sein.

Die ersten elektronischen Computer wurden vor einem halben Jahrhundert gebaut. Darunter waren Ungetüme von Maschinen; manche von ihnen nahmen eine Fläche ein, so groß wie ein Häuserblock. Um sie in Gang zu halten, waren ein eigenes Kraftwerk und



ein Kühlhaus erforderlich. Nur etwa eine Stunde funktionierten sie. Heute leisten preiswerte Computer, die man in die Tasche stecken und mit Batterien oder Solarzellen (sie erzeugen aus dem Licht der Sonne elektrischen Strom) betreiben kann, mehr als diese Maschinengiganten aus der Babyzeit des Computers. Wie ist das möglich?

## Von elektronischen Winzlingen mit großer Leistung

In den letzten zwanzig Jahren ist es Wissenschaftlern, Technikern und Technologen der Industrie gelungen, die elektronischen Bauelemente und ihre Verbindungsleitungen mikroskopisch klein zu machen und eine Vielzahl auf aller kleinstem Raum zusammenzufassen. Hochleistungsfähige elektronische Geräte in der Größe eines Fingernagels (und kleiner), bestehend aus vielen tausend Bauelementen, wurden geschaffen. Diese Winzlinge verbrauchen wenig Energie, halten lange, arbeiten nahezu fehlerfrei und sind sehr billig, weil sie automatisch und in großen Stückzahlen hergestellt werden.

Ein derartiges modernes elektronisches Bauelement heißt *integrierter Schaltkreis* oder *Chip* (englisch: chip –sprich: tschip: = Plättchen, Splitter). Chips sind das bisher wichtigste Ergebnis der Elektronik des mikroskopisch Kleinen – der Mikroelektronik. Aus Kleinstbauteilen aufgebaute Computer heißen Mikrorechner oder Mikrocomputer. Auch der Taschenrechner oder die Quarzuhr enthalten einen Chip.

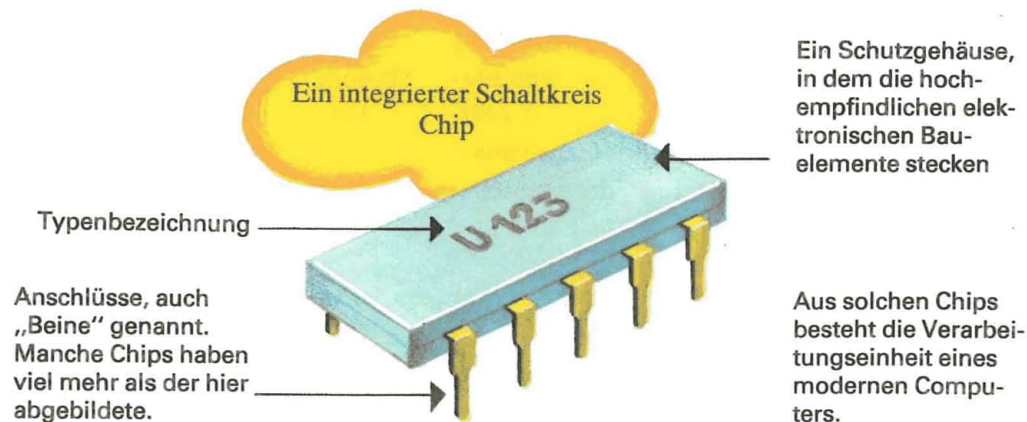
### Wir merken uns:

Moderne Computer werden aus integrierten Schaltkreisen (Chips) aufgebaut.

Integrierte Schaltkreise finden wir heute in allen modernen elektronischen Geräten, auch im Radio, Fernsehgerät oder Kassettenrecorder, im Waschautomaten und Telefon.

Es gelingt bereits, in einem integrierten Schaltkreis von Fingernagelgröße 1 000 000 einfachste elektronische Bauelemente zum Speichern oder Verarbeiten von Bits unterzubringen! Das sind wirklich erstaunliche Leistungen von Wissenschaftlern, Ingenieuren, Technikern und Facharbeitern.

Heute leisten Chips für den gleichen Preis vieltausendmal mehr als elektronische Bauelemente vor





25 Jahren und benötigen weit weniger Platz und Energie.

Welche gewaltige Umwälzung die Mikroelektronik in nur einem halben Menschenleben brachte, zeigt folgendes Beispiel: Hätte sich die Technik zur Herstellung von Mopeds in den letzten 25 Jahren ebenso rasch entwickelt wie die mikroelektronische Computertechnik, dann sähe das heutige Moped so aus:

- Es führe ebensoschnell (ungefähr 60 Kilometer in der Stunde).
- Es kostete 1 (in Worten: eine) Mark.
- Es paßte in einen kleinen Briefumschlag.
- Es brauchte für eine Strecke, die rund um den Erdball reicht, einen Kanister Kraftstoff.

### Wir merken uns:

Die Fortschritte bei der Herstellung mikroelektronischer Geräte sind eine der Ursachen für den massenhaften Einsatz von Computern.

## Wir schauen immer tiefer hinein

Jetzt wollen wir in Gedanken einen modernen Mikrocomputer auseinandernehmen und nachschauen, was alles in ihm steckt.

Den Netzstecker haben wir gezogen. Nun öffnen wir den Computerkasten und staunen erst einmal, wie wenig sich darinnen befindet. Der große Leerraum gestattet einen guten Abzug der Wärme, die während der Betriebszeiten des Computers entsteht. Dann entdecken wir einige Kunststoffplatten, hand- oder heftgroß. Auf ihnen sind verschiedene Bauteile, die Chips, befestigt. Die Anschlüsse, die „Beine“ der Bauteile,

sind mit dünnen Metallstreifen, den Leiterzügen, verbunden, die auf der Kunststoffplatte festkleben. Das Ganze ist eine *Leiterplatte* oder *Platine*.

Auf der Leiterplatte befinden sich Chips zum Speichern (Speicherchips) und zum Verarbeiten (Prozessorchips). Wir nehmen gedanklich einen Chip von der Leiterplatte, entfernen das Schutzgehäuse, um in sein elektronisches Gehirn zu schauen. In der Mitte sehen wir den eigentlichen integrierten Schaltkreis, an den feinste Drähte angeschlossen sind, die zu den Anschlüssen am Gehäuseboden führen.

Um in dem Chip noch mehr zu erkennen, müssen wir ihn sauber auftrennen und vergrößern, beispielsweise mit Hilfe eines Mikroskops. Bei geringer Vergrößerung sehen wir ein sehr dichtes verwinkeltes Netzgewebe von Strichen, wie Straßen in einem Stadtplan. Ihre Anordnung ist verschieden.

Wenn der Computer arbeitet, bewegen sich nach dem Takt einer unsichtbaren Uhr blitzschnell Elektronen in diesem Staßennetz. Techniker sprechen von Stromstößen oder *Impulsen*. (Ein Impuls bedeutet beispielsweise eine 1, kein Impuls eine 0.)

Übrigens kann man die Arbeit der Computer hörbar machen. Betätigt man einen Taschenrechner älterer Bauart dicht neben einem Radio, so vernimmt man ein Störgeräusch aus dem Apparat. Dieses macht die Impulsfolgen im Taschenrechner hörbar.

Zerteilen wir den Chip weiter und vergrößern ihn noch stärker, so erkennen wir eine geometrisch geordnete und geschichtete Landschaft mit Gräben, Rinnen, Flächen, Hochebenen. Jedes dieser Gebilde hat eine besondere Aufgabe beim Durchlauf der Impulse.

Das rasche Entwicklungstempo in der Mikroelektronik und der Mikrorechentechnik zu mehr Leistung auf weniger Platz hält unvermindert an.

## Vom Äußeren des Computers bis in das Innerste eines Chips (mit Größenvergleich)

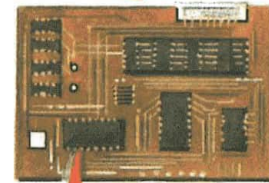
Mikrocomputer im Gehäuse

Etwa so lang wie ein Arm



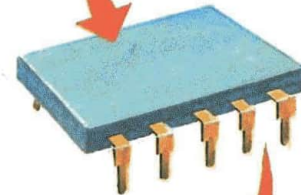
Leiterplatte aus dem Gehäuse

Etwa so lang wie ein Finger



Verarbeitungs- oder Prozessorchip von der Leiterplatte

Etwa so groß wie ein Fingerglied



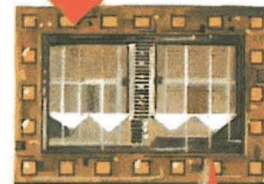
Prozessorchip ohne Schutzgehäuse

Etwa so groß wie ein Fingernagel



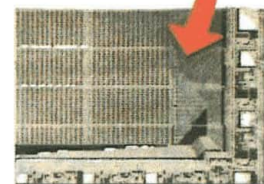
Vergrößertes Inneres des Chips

Etwa so groß wie eine Fingerkuppe

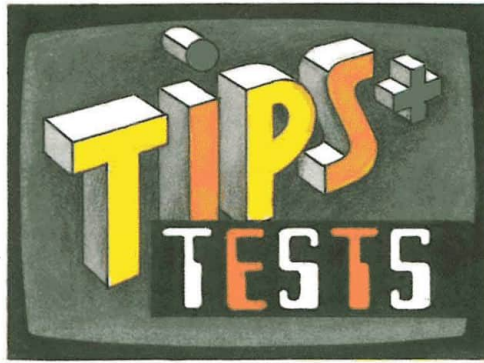


Vieltausendfach vergrößertes Teil des Chipinneren

Etwa so breit wie ein Kinderhaar







## Tips und Tests für Pfiffige

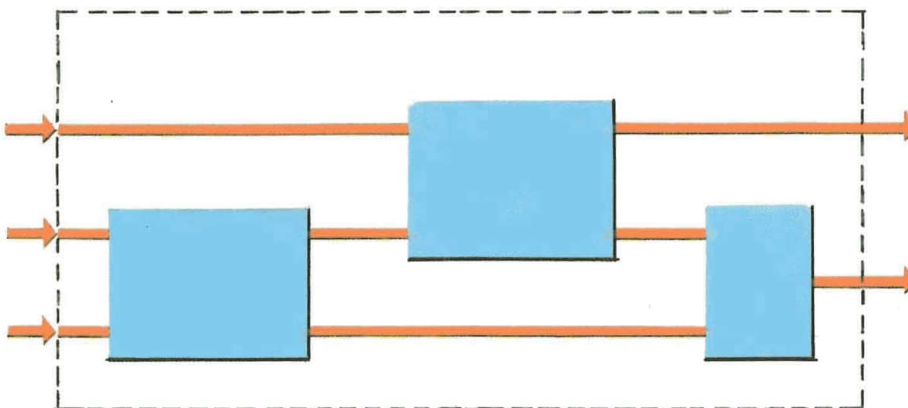
- 13.1. Welches Teil eines Computers sorgt für die automatische Ausarbeitung der Programme?
- 13.2. Schreibe den Algorithmus auf, den das Leitwerk ausführt!
- 13.3. Wozu dient der Befehlszähler?
- 13.4. Mit welcher Programmanweisung kann der Programmablauf von Zwischenergebnissen abhängig gemacht werden?
- 13.5. Ist die Umwandlung der Werte eines Bits in die gleichen Werte  $a$  eine Operation,  $b$  keine Operation?

- 13.6. Bei einer NOT-Operation werden aus welchen Eingangswerten welche Ausgangswerte?
- 13.7. Um ein Eis zu essen, muß es Eis geben, und du mußt das Geld dafür haben. Welche binäre Operation verknüpft hier „es muß Eis geben“ und „du mußt das Geld dafür haben“?
- 13.8. Aus welchen beiden Binäroperationen kann man einen Halbaddierer aufbauen?
- 13.9. Wofür werden binäre Operationen benötigt?
- 13.10. Eine schwierige Knobelaufgabe:

Ein Volladdierer oder Volladder verwirklicht die vollständige Addition zweier binärer Eingänge. Er wird zum Addieren von Zahlen mit mehreren Bits benötigt. Der Volladdierer hat drei Eingänge (die Summanden A, B und den Übertrag  $C_v$  von der vorhergehenden Stelle) sowie zwei Ausgänge (die Summe S und den Übertrag  $C_n$  zur nächsten Stelle).

Der Volladdierer wird aus zwei Bit-Addierern und einem OR-(ODER)-Glieder aufgebaut (siehe Bild unten). Wie sind die Ein- und die Ausgänge der drei Bauelemente miteinander zu verbinden? Zeichne die Lösung!

- 13.11. Was ist ein Chip?
- 13.12. Was ermöglicht unter anderem den massenhaften Einsatz von Computern?



| OR-Glied |   |         |
|----------|---|---------|
| Eingänge |   | Ausgang |
| 0        | 0 | 0       |
| 0        | 1 | 1       |
| 1        | 0 | 1       |
| 1        | 1 | 1       |



# 14. Von Berufen rund um den Computer

---

Immer mehr Menschen arbeiten in der Datenverarbeitung und üben Berufe aus, die mit dem Computer zu tun haben. Welche Berufe sind das? Sind es Berufe mit Zukunft?



## Computerspezialisten – hauptberuflich

Neue Technik, die viel benutzt wird, bringt neue Berufe mit sich. Auch in der gut hundertjährigen Geschichte des Autos entstanden neue Berufe, die es früher, zur Zeit der Pferdedroschke, noch nicht gab. Denken wir nur an den Berufskraftfahrer, den Kfz-Schlosser, den Tankwart oder an die vielen Berufe in der Automobilindustrie.

Ähnlich verhält es sich mit dem Computer, der neue Arbeitsaufgaben, neue Betriebe und neue Berufe mit sich brachte, auch alte Berufe wandeln und in Zukunft weitere neue Berufe entstehen lassen wird.

### Wir merken uns:

Mit dem Computer entstanden und entstehen neue Berufe, alte werden gewandelt.

In der Welt des Computers gibt es so gut wie keine Tätigkeiten, die man einfach anlernen kann. Die Ausübung aller Berufe rund um den Computer erfordert eine gründliche Ausbildung als Facharbeiter, Techniker, Ingenieur oder Diplomingenieur, sie verlangt viel Wissen, Geschick und fachliches Können.

Geeignet für solche Berufe ist nicht jeder. Bestimmte gesundheitliche und charakterliche Eigenschaften sowie geistige Fähigkeiten müssen vorhanden sein. Wer einen Beruf fürs Leben wählen möchte, muß prüfen, ob dieser später wirklich gebraucht wird und ob er selbst die nötigen Voraussetzungen mitbringt.

## Computerspezialist – so nebenbei

Software und die dazugehörige Hardware dienen dazu, Arbeitsprozesse in Industrie- und Landwirtschaftsbetrieben, in Krankenhäusern, Instituten und Labors, in Büros, Zeitungsredaktionen, Bibliotheken und an-

deren Einrichtungen zu automatisieren, um Zeit, Material und Geld zu sparen. Will man einen Computer mit seinen großen Möglichkeiten wirklich nutzbringend verwenden, reichen Kenntnisse über die Hardware und die Software allein nicht aus. Es werden unbedingt auch gute Fachkenntnisse auf dem Anwendungsgebiet benötigt. Beispielsweise kann niemand eine ordentliche Computersteuerung für eine Werkzeugmaschine bauen, der nicht ganz genau über diese Maschine Bescheid weiß. Deshalb müssen Computerspezialisten mit den Fachleuten der Anwendungsgebiete gut zusammenarbeiten.

Immer häufiger eignen sich, umgekehrt, auch Nicht-Rechentechniker Wissen und Fertigkeiten im Umgang mit dem Computer an; sie werden Computerspezialisten – so nebenbei. Wie auch viele Autobesitzer ihr eigener Kraftfahrer und auch Autoschlosser sind.

Außer den Hardware- und Softwarespezialisten gibt es Fachleute, für die der Computer wichtigstes Werkzeug geworden ist, ohne daß sie Computerspezialisten sind. Mancher Bauingenieur, Architekt, Robotereinrichter, Maschinenbauer, mancher Planer, Buchhalter oder auch Landwirt, Bibliothekar und Arzt zählt schon dazu.

### Wir merken uns:

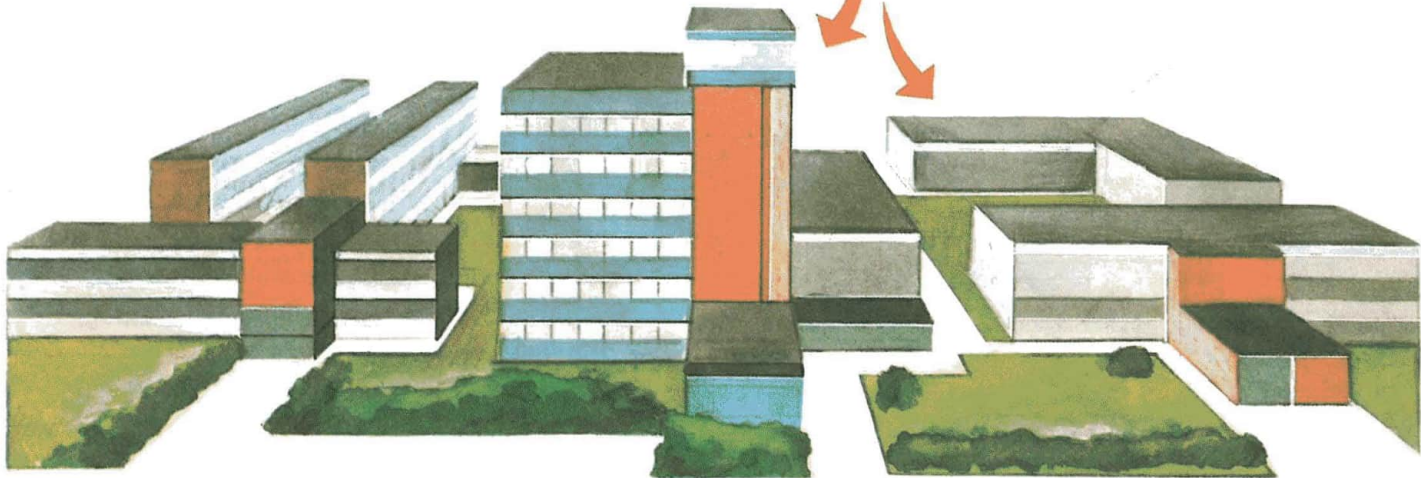
Es gibt immer mehr Berufe, für die der Computer unentbehrlich ist.

## Wie sind die Aussichten?

Neue Technik prägt neue Begriffe. Für die Wissenschaft von der systematischen Verarbeitung von Daten, insbesondere mit Computern, wird der Begriff *Informatik* gebraucht. Man benutzt ihn auch für Berufsbezeichnungen. Gefragt sind Informatik-Fachleute, die auf ihrem Gebiet das Allerneueste wissen und beherrschen. Sie haben gute berufliche Aussichten – wie alle Spezialisten in der Computeranwendung.



## Neue Betriebe und Arbeitsmöglichkeiten, die mit dem Computer entstanden sind



### *Hardware-Hersteller*

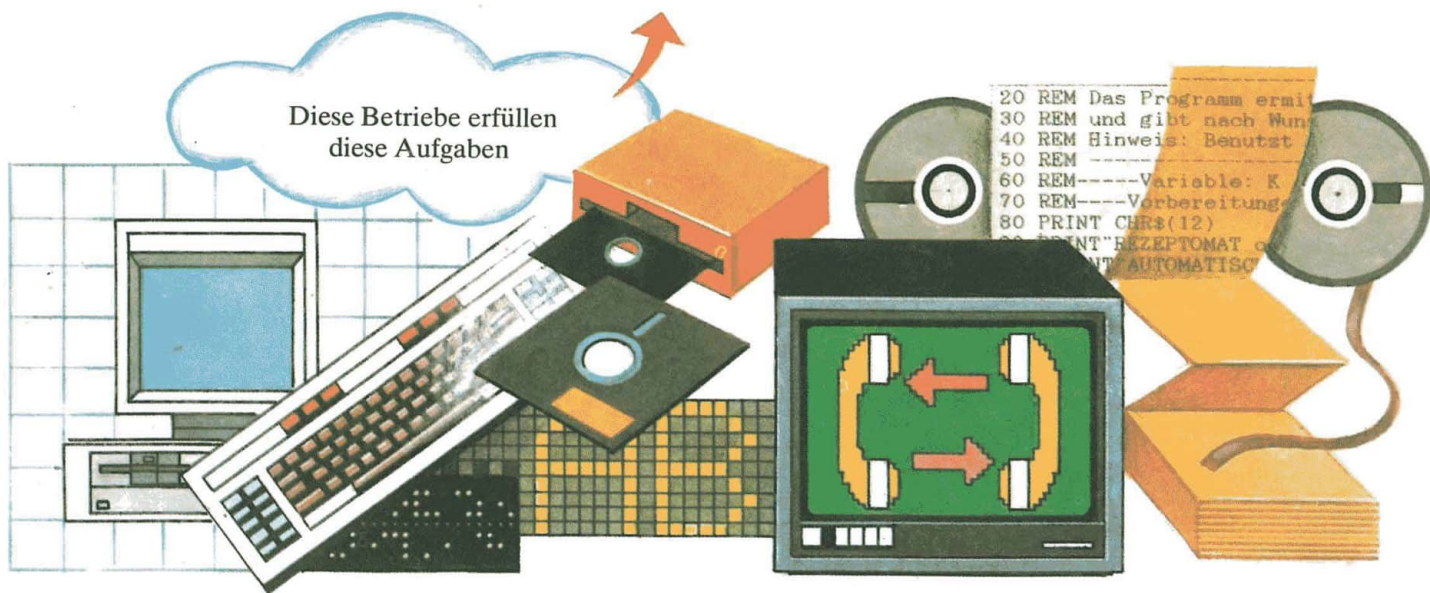
neue Geräte konstruieren (entwickeln)  
und produzieren  
vorhandene Geräte verbessern und  
reparieren  
Geräte verkaufen

### *Software-Hersteller*

vorhandene Software pflegen und  
verbessern  
neue Software ausarbeiten  
Software verkaufen  
Anwender beraten und schulen

### *EDV-Service und Rechenzentrum*

große Computer und Software  
benutzen  
umfangreiche Datenverarbeitungsauf-  
gaben für viele Kunden erledigen  
Daten von Belegen erfassen, die  
Ergebnisse weiterleiten  
Software für Kunden ausarbeiten







*Solche Berufe  
beispielsweise sind in  
diesen Betrieben zu finden.  
Manche Berufsbezeichnungen  
ändern sich oder sind  
von Land zu Land  
unterschiedlich.*

#### *Hardware-Hersteller*

Elektroniker  
Informationstechniker  
Elektroniktechnologe  
Automatisierungstechniker  
Technischer Zeichner



#### *Software-Hersteller*

Programmierer  
Programmierassistent  
Automatisierungstechniker  
EDV-Organisator  
Systemanalytiker  
EDV-Projektant



#### *EDV-Service und Rechenzentrum*

EDV-Bediener (Operateur)  
CAD-Spezialist  
Wartungsingenieur  
Datenerfasser  
Programmierer  
Facharbeiter für Datenerfassung



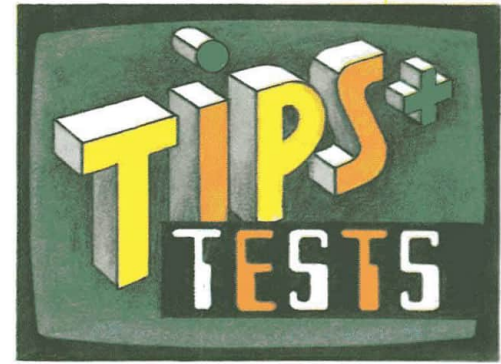
### Wir merken uns:

Viele Berufe auf dem Gebiet der Computertechnik und ihrer Anwendung, der Informatik, haben gute Aussichten, sind Berufe mit Zukunft.

Da sich die Welt des Computers rasch weiterentwickelt, müssen die Informatik-Fachleute stets das Neueste auf ihrem Fachgebiet wissen. Weiterbildung ist für sie Berufspflicht!

Seit Jahrhunderten benutzt der Mensch Papier und Schreibgerät, um Gedanken und Daten festzuhalten, zu speichern. Jetzt wird zunehmend der Computer das unentbehrliche Werkzeug beim Speichern und Verarbeiten von Daten. Mit einem Computer kann man jedoch nicht umgehen wie mit Schreibgerät und Papier. Umlernen wird für viele Menschen zur Notwendigkeit.

Immer mehr Menschen müssen es verstehen, den Computer als Hilfsmittel für ihre Aufgaben zu nutzen. Eines Tages wird das Beherrschen einer Computersprache ebenso zur Allgemeinbildung gehören wie der sichere Umgang mit der Muttersprache.



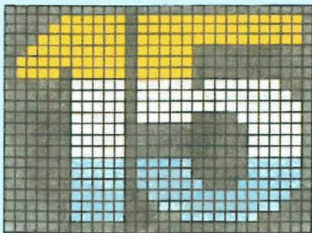
## Tips und Tests für Pfiffige

- 14.1. Nenne Betriebe, in denen hauptsächlich mit dem Computer gearbeitet wird!
- 14.2. Was geschieht bei einem Hardware-Hersteller, einem Software-Hersteller und in einem Rechenzentrum?
- 14.3. Erfordern Berufe auf dem Gebiet des Computers *a* keine Ausbildung, *b* eine Facharbeiter-, *c* eine Fachschul-, *d* eine Hochschulausbildung?
- 14.4. Auf dem nebenstehenden Bild sind viele Berufe und Tätigkeiten aus der Computerwelt aufgezählt. Erkundige dich, welche Arbeitsaufgaben diese Berufe zu lösen haben! Werte auch Tageszeiten aus!
- 14.5. Was muß bei der Wahl eines Berufes auf dem Gebiet der Computertechnik und deren Anwendung beachtet werden?
- 14.6. Besuche ein Berufsberatungszentrum oder Berufsberatungskabinett und erkundige dich, welche Berufe es auf dem Gebiet des Computers gibt und ob sie gefragt sind!
- 14.7. Erkundige dich nach Facharbeiterberufen auf dem Gebiet der Datenverarbeitung und Informatik! Wie sehen die entsprechenden Berufsbilder aus, was sind die Voraussetzungen für die Aufnahme einer Lehre in diesen Berufen?
- 14.8. Auf welchem Gebiet gibt es viele Berufe mit Zukunft?

# 15. Ein Blick in die Zukunft des Computers

---

Wie wird die technische Entwicklung des Computers weitergehen? Was wird es Neues in der Anwendung geben? Von erstaunlichen Dingen, einer neuen, „intelligenten“ Computergeneration soll jetzt die Rede sein.





## Immer besser, schneller, kleiner

Alle zwei bis drei Jahre werden international neue, leistungsfähigere Computertypen angeboten. Entscheidende Grundlage dafür sind immer bessere, leistungsfähigere integrierte Schaltkreise oder Chips. Den Fachleuten in der Mikroelektronik gelingt es, die Anzahl der in einem Chip untergebrachten Schalt- und Speicherelemente ständig zu erhöhen. Schon längst finden komplette Mikrorechner mit Speicher in einem einzigen Chip Platz; man nennt sie 1-Chip-Mikrorechner. Solche Kleinstrechner in Daumengröße finden zunehmend Anwendung in den verschiedensten Geräten und Maschinen, in Fahrzeugen und Robotern. Sie verleihen diesen neue, erstaunliche Eigenschaften.

Sorgenkind bleibt vorerst die Dateneingabe. Doch auch hier sind verbesserte Geräte zur Eingabe von Bildern, Handschriften und der gesprochenen Sprache in Vorbereitung. Wenn es einst eine leistungsfähige und preiswerte Spracheingabe gibt, wird man zum Beispiel eine „hörende“ Schreibmaschine bauen können. Man diktiert in ein Mikrofon, und ein wenig später hat man das Gesprochene schwarz auf weiß gedruckt vorliegen.



Das ist Tom, schon etwas älter und mit Bart. Er diktiert gerade einer neuen „hörenden“ Schreibmaschine.

## Die fünfte Generation

Die heutigen Computer haben eine Reihe von Eigenschaften, welche die Fachleute und alle anderen Benutzer in aller Welt stören:

- Computer stellen manchmal dumme und unverständige Dinge an.
- Computer hören und sehen nur wenig und verstehen nichts.
- Computer lassen sich nur mühsam programmieren.
- Computern kann man Wissen von Fachleuten nicht direkt beibringen.
- Computer sind keine Gesprächspartner, man kann sich mit ihnen nicht vernünftig unterhalten.

So ganz zufrieden kann man mit dem „Kollegen Computer“ der gegenwärtigen Generation wirklich noch nicht sein.

Wissenschaftler und Techniker in aller Welt arbeiten daher an einer neuen, der fünften Generation von Computern, die diese Mängel nicht mehr aufweisen soll. Sie wollen die neue Generation mit einer sogenannten *Künstlichen Intelligenz* (abgekürzt: KI) ausstatten. Künftige Computer sollen Bilder und die natürlich gesprochene Sprache erkennen, menschliches Wissen entgegennehmen, speichern und für Schlußfolgerungen nutzbringend auswerten können, Probleme selbständig lösen – ohne daß der Mensch ihnen die Lösung genau vorprogrammiert.

Nicht wie bisher mehr Daten, sondern Wissen sollen sie verarbeiten.

An die Stelle der früheren Datenverarbeitung tritt die Wissensverarbeitung. Man will Chips benutzen, die viele Millionen Bauelemente enthalten. Die Leistungsfähigkeit und Anwendbarkeit der Computer steigt damit enorm.

Mit ungeheuren Anstrengungen und großem Einsatz wird international daran gearbeitet, eine kühne technische Vision zu verwirklichen: Computer und Roboter mit Eigenschaften, die wir beim Menschen als intelligent bezeichnen.

## Ein interessanter Test

Kann man einer Maschine überhaupt menschliche Eigenschaften, wie klug, verständig, rasch verstehend, also intelligent, zubilligen? Darüber gibt es unterschiedliche Auffassungen.

Von dem Mathematiker Turing, einem der Väter der KI, stammt hierzu ein interessanter Versuch. Stell dir folgendes Spiel vor: Hinter zwei Vorhängen sitzen jeweils ein Junge und ein Mädchen. Vor diesen Vorhängen befindet sich ein Dritter, ein Spielpartner. Dieser stellt Fragen und soll allein anhand der schriftlichen Antworten der unsichtbaren Mitspieler herausfinden, hinter welchem Vorhang das Mädchen sitzt.

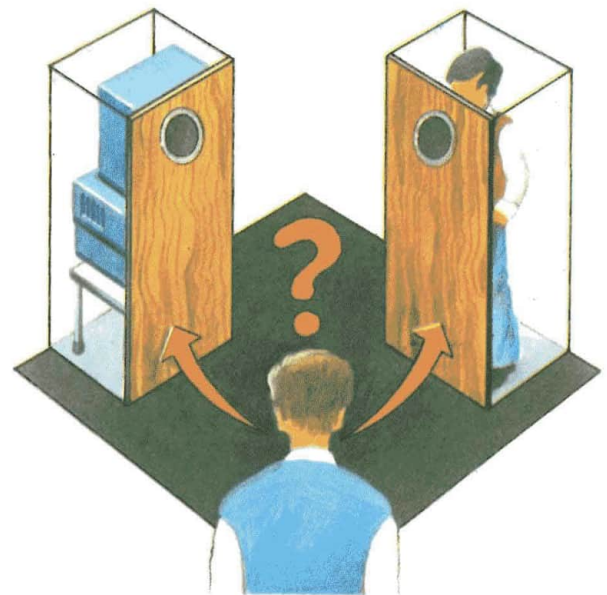
Turing verändert dieses Spiel. Er setzt einen Menschen und einen Computer unsichtbar hinter die Vorhänge oder in jeweils eine Kabine. Die Fragen und die Antworten werden schriftlich (zum Beispiel mit Fernschreiber) ausgetauscht. Kann der Prüfer nach dem Test nicht richtig feststellen, hinter welchem Vorhang oder in welcher Kabine sich der Computer befindet, dann müßte man den versteckten Computer als „intelligent“ bezeichnen.

Kurze Geschichte über eine merkwürdige Begegnung mit einem Computer der Gegenwart und einem (intelligenten) Computer der Zukunft, der fünften Generation

Nehmen wir an, du triffst den Computer und stellst die Frage: „Kannst du mir sagen, wie spät es ist?“ Was wird er antworten?



Der Gegenwartscomputer: „ERROR“ (Fehler).  
Der Zukunftscomputer, wenn ihm auch Witz beigebracht wurde: „Ja.“  
Der Zukunftscomputer, wenn er keinen Spaß macht: „Es ist 10 Uhr 23 Minuten und 11 Sekunden mitteleuropäischer Zeit.“





# Ratschläge und Auskünfte – elektronisch

Schon heute haben erste wissende und ratgebende Computer auf Spezialgebieten weltweit Aufsehen erregt. Fachleute nennen sie *Expertensysteme* (Experte = Fachmann). Sie enthalten Computerprogramme, welche die Fähigkeiten von Experten simulieren. Wie kluge, hochspezialisierte Fachleute vermögen sie auf unvorbereitete Fragen verständliche und nützliche Antworten zu geben. Aus dem Wissen, das diese Computer gespeichert haben, ziehen sie selbständig Schlüsse und zeigen auch, wie sie die Lösung fanden. Manche stellen medizinische Diagnosen, andere erkennen Defekte in Autos oder Computern und schlagen Reparaturen vor. Wieder andere helfen, bestimmte Geräte und Maschinen zu konstruieren, stellen Geräte für Computeranwendungen zusammen, spüren Bodenschätze auf oder sagen Erdbeben voraus. Das Frage-und-Antwort-Spiel vollzieht sich in Schriftform unter Verwendung von Tastaturen und Bildschirmen, später vielleicht einmal in Sprachform über Mikrofon und Lautsprecher oder über Telefon.

Expertensysteme sind erste Wissensverarbeitungsautomaten.

Wie läßt sich Wissen darstellen und in einem Computer speichern, damit es wieder verarbeitet werden kann? Wie kann man beispielsweise einem Computer erklären, daß Schokolade gut schmeckt?

Das muß anders geschehen als bei einfachen Daten, deren Darstellung wir kennen. Das Bild zeigt eine Möglichkeit.

Expertensystemen geht es in gewisser Beziehung wie Menschen: Bei der „Geburt“ (Herstellung) wissen sie überhaupt noch nichts. Ein Lehrer (ein Wissensingenieur) muß ihnen das notwendige Fachwissen aufbereiten und eingeben. Danach sind sie verständig. Der Unterricht geht sehr rasch vonstatten und kann nach Stunden oder Tagen abgeschlossen sein. Doch sehr

## Der Satz „Schokolade schmeckt gut“

als Daten

1

und als Wissen

23

1

Daten

„Schokolade schmeckt gut“

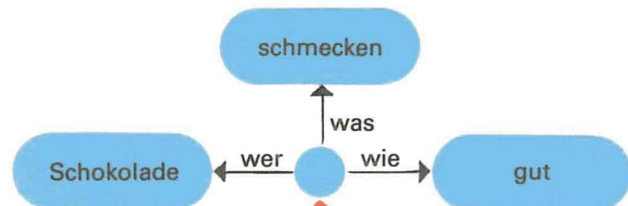


Eine Folge von Zeichen

2

Wissen

dargestellt als ein Netz von Bedeutungen und Zusammenhängen



Hier beginnt das Netz für den Satz (der Wurzelknoten)

3

Wissen

dargestellt und gespeichert als Menge von Folgen mit 3 Werten (Tripeln):

(wer, Ø, Schokolade) (was, Ø, schmecken) (wie, Ø, gut)



Jedes dieser drei Klammerpaare entspricht einem Pfeil im oben dargestellten Netz.



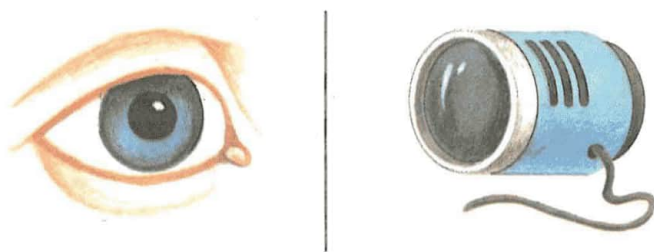
viel Zeit erfordert die Aufbereitung des Fachwissens.

Ohne den Lehrer Mensch sind auch die intelligenten Expertensysteme nur Nichtskönner, die viel kosten. Bringt ein Wissensingenieur einem Expertensystem versehentlich bei, daß  $1 + 1 = 3$  ist, so wird es nicht einmal  $1 + 1$  richtig berechnen können.

Auch Computer mit Künstlicher Intelligenz können nur so viel, wie der Mensch in sie eingebaut oder eingespeichert hat. Deshalb ist der Begriff Künstliche Intelligenz eigentlich nicht treffend. Der Mensch bleibt der alleinige Beherrscher dieser Technik, sie ist ihm Werkzeug und Diener.

## Neuartige Roboter

Ein Mädchen sucht in einer Spielkiste nach seiner Puppe, findet sie und nimmt sie an sich. An dieser scheinbar einfachen Leistung eines Kleinkindes scheitern die heutigen Industrieroboter. Auch stellen sie manchmal Unfug an, weil sie nichts sehen und nichts wissen. Sie greifen zum Beispiel nach Werkstücken, die nicht bereitliegen, oder verspritzen Lack, obwohl das zu lackierende Teil noch fehlt. Ein mit elektronischem Auge, Künstlicher Intelligenz und geeigneten Greifern ausgerüsteter künftiger Roboter wird solche Fehlleistungen nicht aufweisen.



Das Bild oben zeigt ein elektronisches Auge in Gestalt einer kleinen Fernsehkamera. Es wird heute schon benutzt, um die Arbeit von Automaten zu beobachten oder um Einblick in Gefahrenzonen zu nehmen.

Intelligente Industrieroboter werden sehen, Werkstücke erkennen und auf Störungen vernünftig reagieren, aus ihren bisherigen Handlungen lernen und sich auf neue Situationen einstellen können. Sie werden den Ablauf ihrer Arbeit selbständig verbessern und vielleicht ihnen zugerufene Anweisungen befolgen.

Solche intelligenten Roboter vermögen, mehr noch als ihre Vorgänger, dem Menschen bei der Arbeit zu helfen. Immer mehr Computer und Roboter kommen in den Fabriken zum Einsatz. Sie werden das Antlitz künftiger *automatisierter Fabriken* bestimmen und in noch höherem Maße als bisher die unterschiedlichsten Arbeitsprozesse steuern, organisieren und zusammenfassen helfen.

Und daheim? Halten eines Tages Roboter auch in den Wohnungen Einzug? Werden Automaten die Räume säubern, Wäsche waschen und bügeln, Kartoffeln schälen und Gemüse putzen, Einkäufe erledigen und anderes tun?

Heute sind Roboter noch sehr teuer. Ihre Anschaffung lohnt sich nur, wenn man sie ausreichend beschäftigen kann. Und: sie eignen sich nur für spezielle Aufgaben.

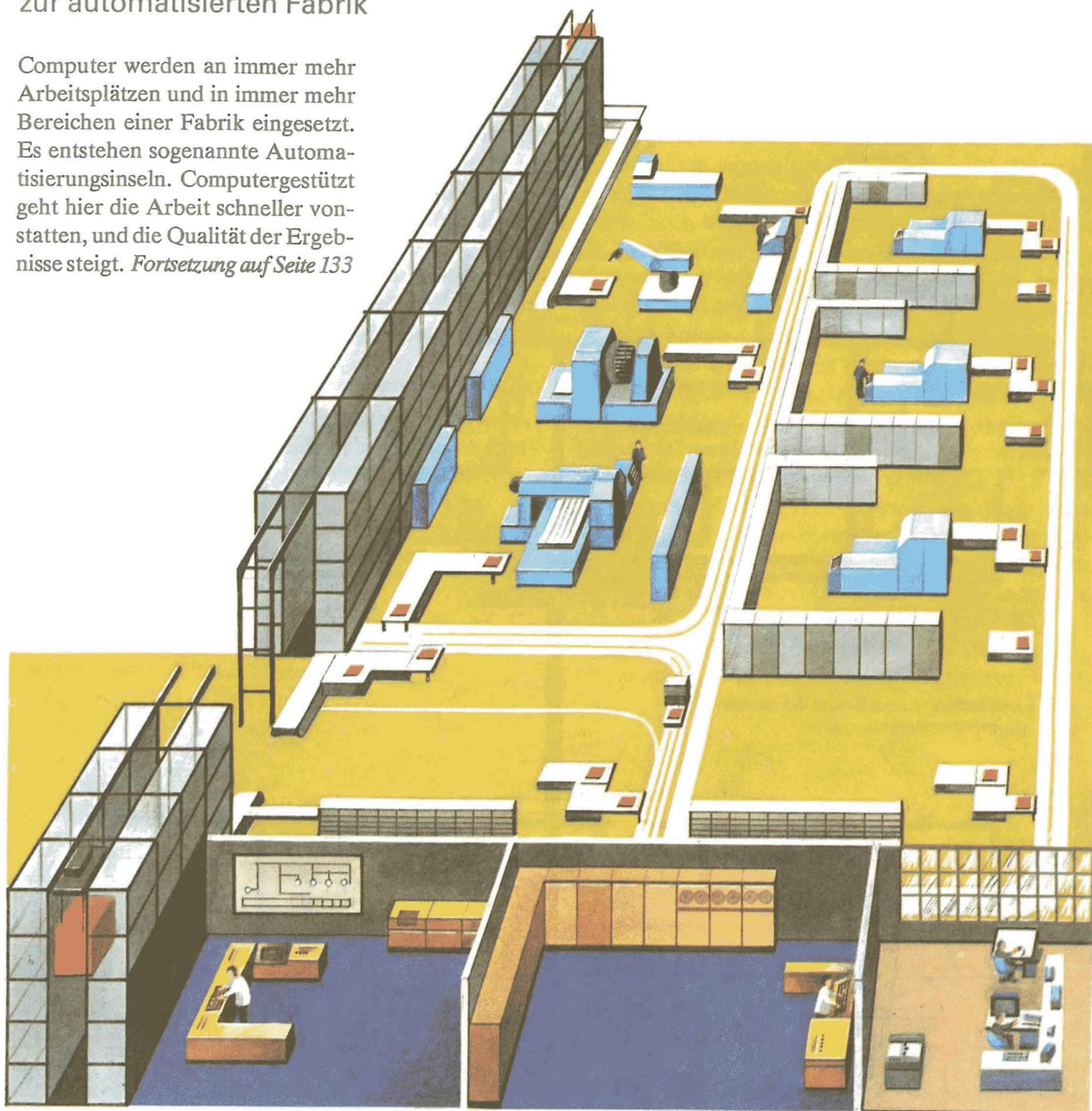
Für ausgewählte Hausarbeiten aber wird es Roboter einmal geben. Erste Geräte für Reinigungsarbeiten in Großbetrieben sind schon bekannt. Auch ein Roboter, der an einer Orgel Musikstücke nach Wahl spielt, war auf einer Ausstellung schon zu sehen.

Ganz gewiß ist es notwendig, über den Bau einfacher Roboter (ohne Künstliche Intelligenz) nachzudenken, die alten, kranken und behinderten Menschen bestimmte Handgriffe abnehmen, die diesen schwerfallen.

Doch wofür Computer – mit oder ohne Künstliche Intelligenz – auch immer eingesetzt werden, sie bleiben Werkzeug ihres Schöpfers: des denkenden, wissenden Menschen.

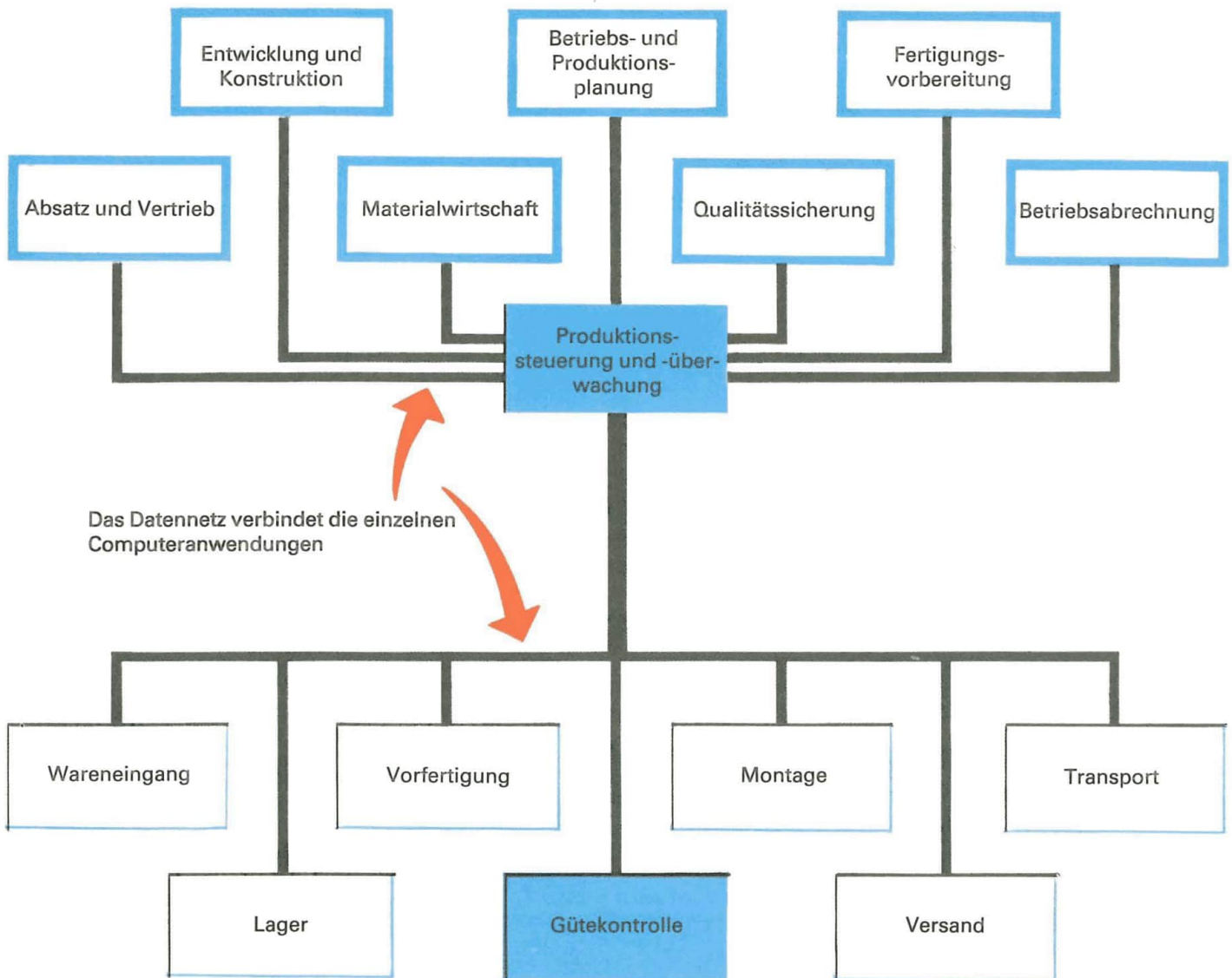
## Auf dem Weg zur automatisierten Fabrik

Computer werden an immer mehr Arbeitsplätzen und in immer mehr Bereichen einer Fabrik eingesetzt. Es entstehen sogenannte Automatisierungsinself. Computergestützt geht hier die Arbeit schneller vonstatten, und die Qualität der Ergebnisse steigt. *Fortsetzung auf Seite 133*





Ein Datennetz verbindet die einzelnen Computeranwendungen





In einem Arbeitskollektiv werden bessere Ergebnisse erzielt, wenn an der gemeinsamen Aufgabe alle enger zusammenwirken. So auch bei den Computeranwendungen im Betrieb: Die Arbeit läuft noch besser, wenn die in den verschiedenen Bereichen eingesetzten Computer untereinander zusammenarbeiten, wenn sie mit elektronischer Geschwindigkeit Daten austauschen und sich übermitteln. Dazu dienen Leitungen (wie beim Telefon oder auch Lichtleiter aus Glasfasern) und wiederum Computer. Man spricht von Datenübertragungsnetzen oder kurz Datennetzen.

Mit Computern, Datennetzen und entsprechender Software wachsen die Automatisierungsinself zu einem noch leistungsfähigeren Organismus zusammen: der automatisierten Fabrik.

So bringt zunehmende „Intelligenz“ der Computeranwendungen die betrieblichen Prozesse enger zusammen, integriert sie. Man spricht daher von computerintegrierter Fertigung (englische Abkürzung: CIM). Der Mensch wird mit dieser Technik in die Lage versetzt, betriebliche Prozesse noch wirksamer zu beherrschen. Mehr, bessere und neue Waren können in kürzerer Zeit erzeugt werden. Das bedeutet Einsparungen an Zeit, Platz und Geld. Und für die Menschen am Arbeitsplatz wird die berufliche Tätigkeit oft anspruchsvoller und interessanter.

## Zum Abschied

Computer ermöglichen es dem Menschen, schneller, billiger und erfolgreicher zu arbeiten und zu lernen. Sie können helfen, sein Leben reicher, interessanter und lebenswerter zu gestalten. Unbelastet von langweiligen und ermüdenden Arbeiten, die ihm Maschinen abnehmen, wird der Mensch seine schöpferischen Fähigkeiten noch besser entwickeln und einsetzen können als gegenwärtig. Eine großartige menschenwürdige Zukunft wird sich auftun – sofern diese Automaten für das Wohl der Menschen und für den Frieden eingesetzt werden.

Computer und Roboter können manche Wünsche und Verlangen des Menschen erfüllen. Ihre Anwendung stellt jedoch auch Forderungen an ihn selbst. Eine geordnete, sorgfältige und planvolle, gewissermaßen programmierte Arbeitsweise ist für die wirksame Nutzung dieser neuen Technik notwendig. Kluge Menschen mit großem Wissen und fachlichem Können, mit hohem Verantwortungsbewußtsein werden gebraucht, Menschen, die auch zu Umstellungen und zum Umdenken im Interesse des Fortschritts bereit sind.

Lernen und Weiterlernen wird im Zeitalter der Computer und Roboter zur Pflicht für jedermann.



## Tips und Tests für Pfiffige

- 15.1. Sammle Zeitungsausschnitte über neue Computer- und Robotertechnik und werte sie aus!
- 15.2. Sammle Zeitungsausschnitte über Computer und Roboter mit Künstlicher Intelligenz! Gruppier die Ausschnitte nach Anwendungsgebieten!
- 15.3. Welches Ziel hat die Künstliche Intelligenz?
- 15.4. Warum sind Computer der neuen, fünften Generation notwendig?
- 15.5. Kann mit Künstlicher Intelligenz *a* die Leistungsfähigkeit oder *b* die Anwendungsbreite von Computern bedeutend gesteigert werden?
- 15.6. Intelligente Computer speichern und verarbeiten nicht mehr nur Daten, sondern auch ...?
- 15.7. Erarbeite mit deinen Freunden einen Vorschlag für einen elektronischen Ratgeber (ein Expertensystem) für Kinder! Unterscheidet nach Wünschen, die er unbedingt erfüllen soll, und solchen, auf deren Verwirklichung ihr verzichten könntet!
- 15.8. Denke dir einen „intelligenten“ Computer aus, der dir und deinen Freunden Unterricht in deinem Lieblingsfach erteilt. Was müßte er können?
- 15.9. Nehmen wir an, du mußt in einer Fremdsprache 10 000 Vokabeln lernen. Wie kann dir ein Computer dabei helfen?
- 15.10. Entwickle mit deinen Freunden einen Vorschlag für einen „intelligenten“ Haushaltsroboter und beschreibe diesen! Er soll im Haushalt häufig anfallende Arbeiten erledigen. (Aber verlangt von ihm nicht zuviel, sonst wird er zu groß und zu teuer!)
- 15.11. Untersuche mit deinen Freunden, welche Hausarbeiten Kranken, Verletzten, Körperbehinderten oder älteren Menschen besonders schwerfallen und welche dieser Arbeiten sehr häufig erledigt werden müssen! Veranstaltet in eurer Schule einen Knobel- und Erfinderswettbewerb um den besten Vorschlag für ein technisches Hilfsmittel, vielleicht auch einen Hilfsroboter. Erkundigt euch, ob es solche Hilfsmittel schon gibt! Vergleicht eure Lösung gegebenenfalls mit diesen! Diskutiert euren Vorschlag mit Kranken, Körperbehinderten oder alten Menschen, auch mit erfahrenen Technikern. Vergebt den Timur-Technik-Preis!
- 15.12. Denkende Computer und Roboter sind *a* nichts Neues oder *b* etwas völlig Neues, und sie bewirken *c* enorme Veränderungen oder *d* keinerlei Veränderungen?
- 15.13. Lohnt es sich, einen „intelligenten“ Roboter zu bauen, der Fliegen fängt?
- 15.14. Wissen Computer mit Künstlicher Intelligenz *a* nichts, *b* alles, *c* einiges?
- 15.15. Was wird in der Tagespresse über automatisierte Betriebe im In- und Ausland berichtet?
- 15.16. Was können „intelligente“ Computer und Roboter?
- 15.17. Computer können vieles. Wer entscheidet, wofür dieses Können benutzt wird – *a* der Computer, *b* der Mensch?
- 15.18. Ist der Satz „Der Computer verändert unser Leben“ *a* richtig oder *b* falsch?



## Lieber Leser!

Jetzt hast du schon manches aus der Welt des Computers und seiner Anwendung gelernt.

Schmerzt oder raucht schon vor Anstrengung dein Kopf?

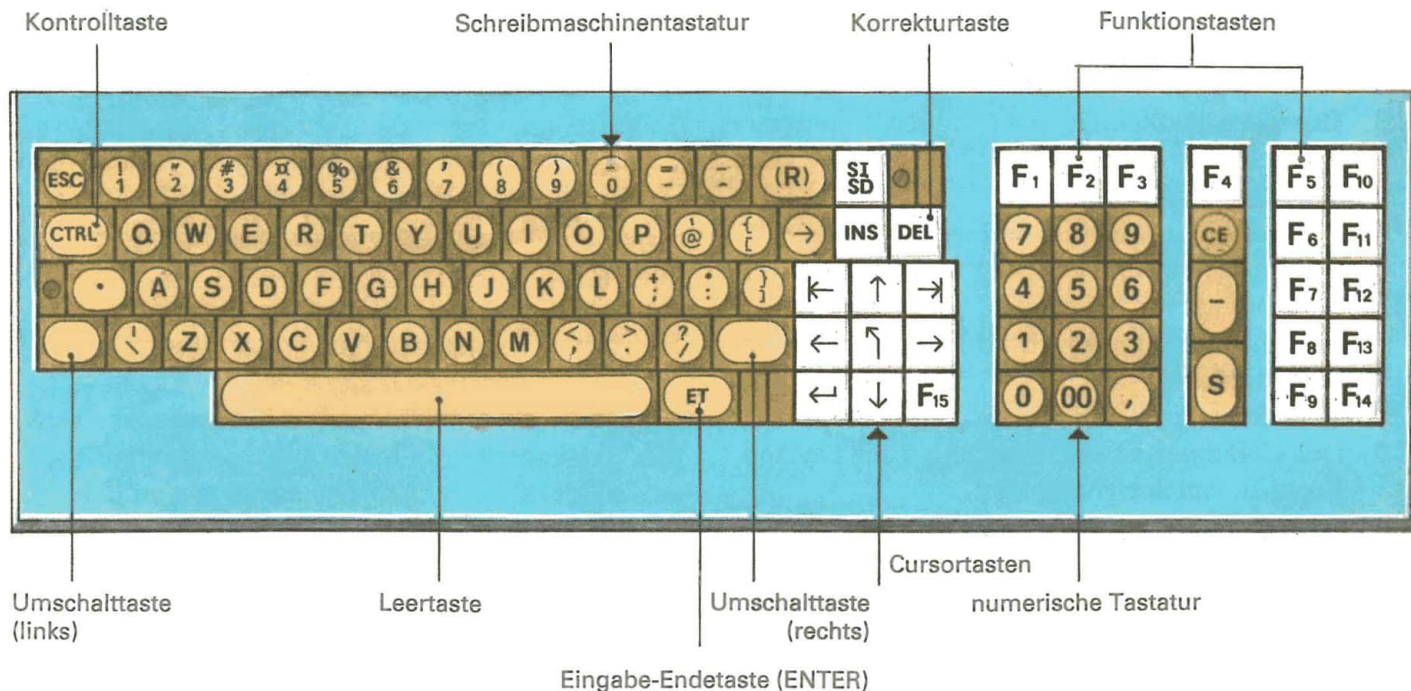
Tröste dich: Immer mehr Menschen müssen heute den Computer beherrschen lernen, auch ältere Menschen, denen das Lernen nicht mehr so leichtfällt wie dir.

Einen Computer beherrscht, wer viele Einzelheiten bis auf den i-Punkt genau weiß. Wenn du Lust hast, Weiteres zu erfahren, neue Computerbegriffe kennenzulernen, so vertiefe dich jetzt in den Anhang. Dort fin-

dest du auch die Bastelanleitung für den Demonstrationscomputer aus Karton. Du kannst außerdem die Tastatur eines 8-Bit-Personalcomputers erkunden und vergrößert nachbauen. Dann vermagst du an diesem Tastaturmodell zu üben und Daten oder Befehle einzugeben.

Falls du dich weiter in die Welt des Computers vertiefst, so unternimmst du etwas Wichtiges für dich und uns alle: Du bereitest dich auf eine Zukunft vor, in der Grundkenntnisse über den Computer und seinen Gebrauch zur Allgemeinbildung gehören werden, wie heute das Schreiben und Lesen.

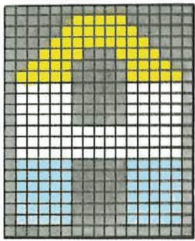
Gerhard Saeltzer



Unterschiedliche Computertypen haben unterschiedliche Tastaturen. Die Anordnung der Tasten für die Ziffern und die Buchstaben ist jedoch meist gleich. Kleine Abweichungen hängen von der Sprache ab. So gibt es im Deutschen den Buchstaben ä, im Englischen oder Französischen aber nicht. Die kyrillische Tastatur hat natürlich eine andere Belegung der Tasten.



# Anhang



## Einige Antworten auf die Fragen des Wissenstests

Viele Antworten auf die Testfragen gehen aus dem Text der jeweiligen Kapitel hervor. Deshalb werden hier nur ausgewählte Fragen beantwortet. Sie sind nach den Kapiteln des Buches geordnet.

- 2.3. Menschen müßten die Arbeiten des Computers übernehmen, oder die Arbeiten könnten nicht mehr ausgeführt werden.
- 2.4. Überwachung Schwerkranker, Roboterbewegungen, Triebwerksteuerung, Ampelsteuerung.
- 2.5. Berechnung von Gebäuden, technische Zeichnungen, Planung und Abrechnung in Betrieben und Banken.
- 2.6. Nein. Ständig entstehen neue Anwendungen.
- 2.7. Der Computer ist ein Werkzeug zum Denken, Rechnen und Speichern.
- 3.1.  $a$ ,  $b$  und  $c$ ; auch  $d$ , wenn der Computer diese Strichmarkierungen lesen kann.
- 3.7. Er ist nicht eingeschaltet; er ist defekt; er hat nichts zu tun.
- 3.8. Ein Computer benötigt keinerlei Arbeitspausen, Ferien oder Urlaub. Oft arbeitet er am besten, wenn er rund um die Uhr beschäftigt ist.
- 4.3. Tätigkeiten werden als Kästchen, Daten als Pfeile gezeichnet.
- 4.4. Eingegebene Zahlen sind auch dann in der Anzeige zu lesen, wenn wir die Finger von den Tasten genommen haben. – Rechenergebnisse bleiben sichtbar, auch wenn der Taschenrechner nicht mehr rechnet, aber noch eingeschaltet ist.
- 4.5. Zumeist die Tätigkeiten  $b$  bis  $e$ .
- 5.2.  $a$ ,  $b$  und  $c$  sind richtig. Sehr leistungsfähige Computer sind auch groß und schwer.
- 5.4.  $b$
- 5.5.  $b$
- 5.6.  $a$
- 5.7.  $b$
- 6.5. Sie sollten es – zumindest für Anfänger – sein.
- 6.7. Drucke drei Zeilen!  
Oder:
  - 1. Drucke Zeile 1!
  - 2. Drucke Zeile 2!
  - 3. Drucke Zeile 3!
- 6.8. Das Ergebnis wird falsch.
- 6.9. Es ist ein Algorithmus für einen Menschen, für eine automatische Steuerung ist er zu ungenau: Ihm fehlen zum Beispiel Vorgaben für die Geschwindigkeit, die Fahrtrichtung oder für Ausweichmanöver.
- 6.13. Der Satz ist falsch ( $b$ ), denn nicht für alle Vorgänge sind Algorithmen vorhanden oder exi-

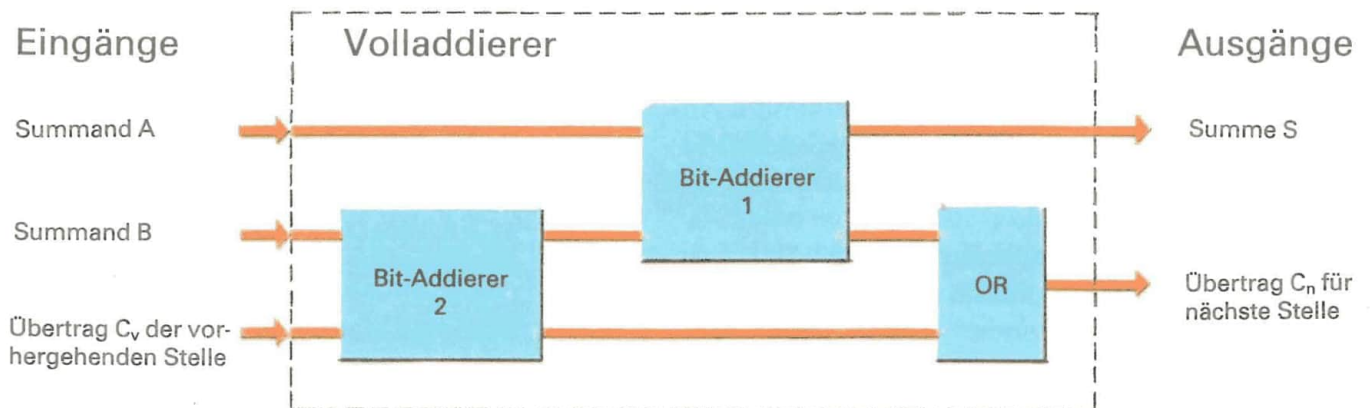
stieren Automaten, welche die Algorithmen in vertretbarer Zeit ausführen können.

- 6.15. Der benutzte Algorithmus für die Vorbereitung ist aufzuschreiben und für das nächste Fest wiederzuverwenden.
- 7.1. Ein Bit, in das eine Ja/Nein-Entscheidung (bit) gespeichert werden kann.
- 7.2. Sie arbeiten mit den beiden Werten (Zuständen) 0 und 1.
- 7.3. *c*
- 7.5. *d*
- 7.6. 2 Bytes.
- 7.7. *c*
- 7.8. *b* und *d*.
- 7.9. *b*
- 7.10. Je größer ihre Speicherkapazität (Fassungsvermögen), um so langsamer arbeiten sie.
- 7.11. In jeder Reihe darf nur eine Kugel sein. Diese kann zwei Zustände (links oder rechts), 0 oder 1, annehmen.
- 8.2. Es passiert nichts. Der Computer wartet auf das Ende der Eingabe. Das müssen wir ihm noch mitteilen, indem wir die Endetaste (ET-, ENTER-, RETURN- oder NEWLINE-Taste) drücken. Dann führt er das Kommando aus.
- 8.3. Er antwortet mit einer Meldung, daß er dieses Kommando nicht kennt. Es kann aber auch sein, daß er ein Kommando CL ausführt, sofern er dieses gespeichert hat.
- 8.4. Entweder haben wir einen Fehler beim Eingeben gemacht, oder der Computer kennt unsere Anweisung nicht, das heißt, die entsprechenden Programme oder Daten befinden sich nicht im Speicher.
- 8.6. Ein Computer kennt keine Ungeduld. Er wartet, bis er einen Auftrag erhält, auch wenn es Stunden oder Tage dauern würde.
- 8.7. Man muß ihn sorgsam behandeln und sich genau an die Bedienungsvorschriften halten. Im Zwei-

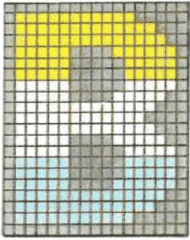
felsfall soll man einen erfahrenen Fachmann fragen.

- 9.1. Diese Frage kann nicht eindeutig beantwortet werden. Die Antwort ist vom Verwendungszweck des Computers abhängig.
- 9.4. Nein, sofern man bereit ist, alle Daten und Programme bei jeder Verwendung neu einzugeben.
- 9.5. Nein. Die Tasten sind zu klein. Es gelingt nicht, auch nur eine einzige Taste zu betätigen, ohne gleichzeitig andere mitzudrücken.
- 9.6. Man benutzt eine zusätzliche Taste, die Umschalttaste, welche von Kleinbuchstaben auf Großbuchstaben um- und zurückschaltet. Es werden also insgesamt 27 Tasten benötigt.
- 10.2. Nein.
- 10.5. *a*, *b* und *c*.
- 10.7. Beides.
- 10.9. Der Computer.
- 10.10. *a* ist immer richtig. – Aber in dem Wort „Auspu“ ist die Folge „Auspu“, die Wiederholung „fff“ und die Folgen „lam“ und „me“ zu schreiben. Doch man kann das Schreiben eines Wortes stets auch als die Wiederholung des Schreibens eines Buchstabens verstehen. Mit dieser Begründung wäre *c* richtig. Unterschiedliche Algorithmen können durchaus das gleiche leisten!
- 10.11. *b*
- 11.1. Systemsoftware und Anwendersoftware.
- 11.2. *a* und *b*.
- 11.3. Nein. Zur Software gehören neben dem Programm auch die Beschreibung und die Bedienungsanleitung (Dokumentation). Ohne diese ist ein Programm durch einen Interessenten nicht anwendbar, also wertlos.
- 12.2. Wird eine Maus je Tag mit 0 und werden zwei Mäuse mit 1 kodiert, so lautet das Fangverhalten 001.

- 12.3. Werden die Schüler mit erhobenem Arm mit 1 und die mit gesenktem Arm mit 0 kodiert, so ist das Ergebnis 0000, 1001.
- 12.4. Verschlüsseln wir eine Stufe mit einer Maus mit 1 und eine Stufe ohne Maus mit 0, so ergeben sich die Binärzahlen 1000, 0100, 1000, 0000.
- 12.5. 17 Mäuse.
- 12.6. 4 beziehungsweise 5.
- 12.7. Von 0 bis 31 (das sind  $2^5 = 32$  verschiedene Werte).
- 12.8. Ottokar hätte sich das Schreiben von 180 Buchstaben erspart.
- 12.9. Wird ein Kreuz mit 0 und kein Kreuz mit 1 verschlüsselt, so ergeben sich  $a$  0110,  $b$  1001,  $c$  0000,  $d$  0101,  $e$  1010.
- 12.10. TOR!
- 13.4. Mit bedingten Sprüngen (IF  $b$  THEN  $A$ ).
- 13.5.  $a$  – es ist eine Gleichheitsoperation.
- 13.7. AND.
- 13.9. Um komplizierte Operationen (zum Beispiel Rechenoperationen, wie Addition und Subtraktion) aufzubauen.
- 13.10. Lösung siehe Bild unten!
- 14.3. Sowohl  $b$  als auch  $c$  oder  $d$  können zutreffen.
- 15.5.  $a$  und  $b$  (Fachleute erwarten eine neue industrielle Revolution).
- 15.6. Wissen.
- 15.12.  $b$  und  $c$ .
- 15.13. Nein. Es gibt wirksamere und billigere Bekämpfungsmittel, zum Beispiel Sprays und Fliegenklatschen.
- 15.14. Anfangs wissen sie  $a$  nichts. Wurden sie von einem Wissensingenieur „belehrt“, wissen sie  $c$  einiges.
- 15.16. Nur das, was der Mensch in sie eingebaut oder eingegeben hat.
- 15.17.  $b$
- 15.18.  $b$ ; Computer sind Werkzeuge des Menschen. Sie können selbst *nichts* verändern. Der *Mensch* verändert die Welt und bedient sich dabei des Computers wie anderer Werkzeuge oder Hilfsmittel.







## Kleines Wörterbuch für Neugierige

Computerfachleute verwenden viele Fremdwörter und Abkürzungen, die andere Menschen zumeist nicht kennen. Einigen dieser Begriffe bist du in diesem Buch bereits begegnet.

Dieses kleine Wörterbuch will dich mit weiteren wichtigen Fachbegriffen und deren Bedeutungen bekannt machen. Sie sind alphabetisch geordnet. Stammt ein Wort aus dem Englischen, so ist es mit der Abkürzung „engl.“ gekennzeichnet; stammt es aus dem Russischen, so steht neben diesem Wort „russ.“. Befindet sich vor einem Begriff ein Pfeil (→), so verweist er dich auf ein Stichwort, wo du Weiteres nachlesen kannst.

**AI** ist die Abkürzung für Artificial Intelligence (engl.) = Künstliche Intelligenz.

**Bus** heißt die Sammelleitung im Mikrocomputer. Sie besteht aus vielen nebeneinanderliegenden elektrischen Leitungen. Die Geräte der → Peripherie eines Computers werden an den Bus angeschlossen.

**CAD** ist die Abkürzung für Computer Aided Design (engl.) = rechnergestützte Konstruktion. So bezeichnet man eine Anwendung des Computers, mit deren Hilfe man Maschinen, Geräte und Erzeugnisse konstruieren, berechnen und ihre Produktion vorbereiten kann.

**CAL** ist die Abkürzung für Computer Aided Learning (engl.) = rechnergestütztes Lernen; die Nutzung des Computers zum Lernen und Üben.

**CAM** ist die Abkürzung für Computer Aided Manufacturing (engl.) = rechnergestützte Fertigung; sie bezeichnet die Anwendung des Computers zur Steuerung in der Produktion von Erzeugnissen.

**CIM** ist die Abkürzung für Computer Integrated Manufacturing (engl.). Das bedeutet: Mittels Computer werden alle rechnergestützten Arbeitsprozesse von der Planung bis zur Fertigung von Erzeugnissen zusammengefaßt, organisiert und gesteuert.

**COBOL** (engl.) ist eine ältere, weltweit verbreitete höhere Programmiersprache. Sie wird vor allem für Aufgaben der Planung und der Abrechnung in Betrieben und Büros benutzt.

**Compiler** heißt ein Hilfsprogramm, mit dem Programme, die in einer höheren Programmiersprache, wie → FORTRAN, → COBOL oder → PASCAL, geschrieben sind, in die ausführbare → Maschinensprache des Computers übersetzt werden. Dann erst vermag der Computer die Programme abzuarbeiten.

**Computergestützt** bedeutet rechnergestützt, mit Hilfe des Computers.

**Computergrafik** heißt die Anwendung des Computers, bei der Bilder und Zeichnungen in den Computer eingegeben, verarbeitet und auf dem Bildschirm oder auf Papier ausgegeben werden.

**CPU** ist die Abkürzung für Central Processing Unit (engl.) = das steuernde, rechnende und (kurzzeitig) speichernde Zentrum eines Computers, die Zentraleinheit.

**Cursor** (engl.) heißt ein auffälliges (zum Beispiel blinkendes) Zeichen am Bildschirm, welches anzeigt, wo das nächste auf der Tastatur eingegebene Zeichen stehen wird.

**Datei** ist die Bezeichnung für Daten, die für einen bestimmten Zweck geordnet und gespeichert sind. Beispiel: Schülerdatei einer Stadt, geordnet nach Schulen, Namen und Geburtsjahr der Schüler.

**Datenbank** heißt die geordnete Speicherung umfangreicher Daten eines ganzen Anwendungsgebietes. Beispiele: Arbeitspläne, Zeichnungen, Namen, Adressen. Die Datenbank umfaßt viele → Dateien, die vom → Datenbankbetriebssystem verwaltet werden.

**Datenbankbetriebssystem** heißt ein Satz von Programmen, mit denen Daten in einer umfangreichen → Datenbank gespeichert, wiedergefunden und geändert werden können.

**Dialog** heißt das unmittelbare „Zwiesgespräch“ mit dem Computer. Beispiele: Der Mensch gibt Fragen ein, der Computer antwortet mit Ausschriften. Oder: Der Computer fragt, und der Nutzer antwortet.

**Editor** ist ein Hilfsprogramm, mit dem Programme und Texte in den Computer eingegeben, geändert und ausgegeben werden. Beispiel: Ausmerzen von Tippfehlern.

**FILE** (engl.) ist eine → Datei.

**FORTRAN** (engl.) ist eine weltweit verbreitete höhere Programmiersprache. Sie wird für wissenschaftliche, technische und mathematische Aufgaben benutzt. FORTRAN ist die älteste höhere Programmiersprache. Es gibt auch moderne Fassungen, zum Beispiel FORTRAN 77.

**II** ist die Abkürzung für iskusstwennyj intellekt (russ.) = Künstliche Intelligenz.

**Instruktion** ist ein Befehl oder eine Anweisung für einen Computer.

**Interface** (engl.) = Schnittstelle; so wird eine auftrennbare Verbindung bezeichnet, über die zwei oder mehr Computerbausteine miteinander verbunden sind, zum Beispiel die → CPU mit der → Peripherie. (Das Interface zum elektrischen Energienetz sind die Steckdose und der Stecker.)

**Interpreter** ist ein Hilfsprogramm, mit dem Anweisungen eines Programms in einer höheren Programmiersprache, wie BASIC, sofort ausgeführt werden. Mit Interpretern lassen sich Computerprogramme schnell ausarbeiten.

**LISP** ist die Abkürzung für List Processing (engl.) und bezeichnet die erste Programmiersprache der Künstlichen Intelligenz. Es gibt bereits Computer, die ausschließlich LISP-Programme verarbeiten, die sogenannten LISP-Maschinen.

**LOGO** ist eine von der Künstlichen Intelligenz und → LISP beeinflusste neue höhere Programmiersprache. Sie kann schon von achtjährigen Schülern erlernt werden. LOGO benutzt Elemente der → Computergrafik. Als Ausgabegerät wird eine sogenannte Schildkröte (oder ein Igel) benutzt. Das ist ein kleiner gesteuerter zeichnender → Roboter, der auf Papier entlangfährt und dabei mit einem Stift seinen Weg aufzeichnet. Die Ausgabe erfolgt auch auf dem Bildschirm.

**Maschinensprache** (oder Maschinenkode) ist die einzige Sprache, die der Computer ohne Hilfsprogramm direkt versteht und abarbeiten kann. Für das Programmieren ist sie viel zu umständlich. Alle Befehle und Daten müssen als Bitmuster verschlüsselt werden.

**Menü**, es bietet dem Anwender eines Computers am Bildschirm mehrere Möglichkeiten, wie das Programm fortgesetzt werden kann, zur Auswahl an (vergleichbar dem Speisenangebot auf einer Speisekarte), von denen dann eine durch Tastendruck auszuwählen ist. Menüs werden zur Steuerung komplizierter Programme benutzt.



**Operationssystem** (oder Betriebssystem) heißt ein Satz von Programmen, welcher die Reihenfolge und die Abarbeitung von anderen Programmen im Computer steuert. Operationssysteme erleichtern sehr den Umgang mit dem Computer und eine wirksame Nutzung seiner Programmenteile. Sie werden deshalb von allen Computeranwendern unbedingt benötigt.

**PASCAL** ist eine moderne höhere Programmiersprache. Sie wurde nach dem französischen Mathematiker Blaise Pascal (1623–1662) benannt.

**Peripherie**, zu ihr zählen alle Geräte, die an eine → CPU angeschlossen sind und dazu dienen, Daten einzugeben, zu speichern oder auszugeben: Tastaturen, Bildschirme, Drucker, Laufwerke für Magnetbandkassetten oder für Disketten.

**Plotter** heißt ein Ausgabegerät der → Peripherie, mit dem technische Zeichnungen auf Papier gezeichnet oder auf Folie gebracht werden. Plotter benötigt man für → CAD und die → Computergrafik.

**PROLOG** (Programmieren in Logik) ist eine junge Programmiersprache der Künstlichen Intelligenz. Es gibt bereits Computer, die ausschließlich PROLOG-Programme verarbeiten, sogenannte PROLOG-Maschinen.

**RAM** ist die Abkürzung für Random Access Memory (engl.) und bezeichnet einen Schreib-Lese-Speicher, ein Speicherbauteil eines Computers, in dem Daten sowohl eingeschrieben als auch wiedergelesen werden können. Beim Abschalten des Computers gehen alle gespeicherten Daten verloren.

**Roboter** nennt man einen computergesteuerten Hand-

habe- oder Transportautomaten. Er führt selbständig Arbeiten aus, die der Mensch sonst mit Hand und Arm oder den Beinen verrichten müßte.

**ROM** ist die Abkürzung für Read Only Memory (engl.) und bezeichnet einen Festwertspeicher, ein Speicherbauteil, aus dem die Daten oder Programme nur gelesen werden können. Beim Abschalten des Computers bleiben alle gespeicherten Daten erhalten (im Unterschied zum → RAM).

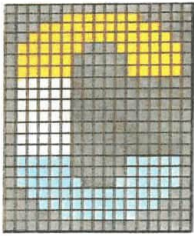
**Simulation** ist die Modellierung und Untersuchung von wirklichen oder erdachten Vorgängen auf einem Computer. Sie wird auch zum Lernen und Üben benutzt. (Die Simulation einer Autofahrt mit einem erdachten Raketenauto wurde am Anfang des Buches beschrieben.)

**System**, es faßt eine Menge von Teilen zusammen, zwischen denen Zusammenhänge oder Verbindungen bestehen. So setzt sich ein Computersystem aus Computerbausteinen zusammen, die über → Interface und Leitungen miteinander verbunden sind.

**Textverarbeitung** heißt die Anwendung des Computers zum Eingeben, Speichern, Ändern und Drucken von Texten aller Art (Briefe, Aufsätze, Pläne, Zeitungsartikel, Bücher). Die Textverarbeitung automatisiert Arbeiten, die in Büros und Redaktionen früher mit Schreibmaschinen oder -geräten verrichtet wurden.

**Z3** ist der Name des ersten funktionstüchtigen programmgesteuerten Rechenautomaten der Welt. Er wurde 1941 in Berlin in Betrieb genommen, arbeitete binär und benötigte für eine Multiplikation 3 Sekunden. Erfinder und Erbauer war der deutsche Bauingenieur Konrad Zuse (geboren 1910). Während eines Bombenangriffs im zweiten Weltkrieg wurde der Z3 zerstört.





## Ein weltbekannter Zeichenkode

Kodes werden benutzt, um die Zeichen unserer Schriftsprache in die Bit-Sprache des Computers zu verschlüsseln oder – umgekehrt – zu entschlüsseln.

Einen Ausschnitt aus dem weltweit bekannten ISO-7-Bit-Kode zeigt dir die folgende Tabelle. (Der Kode wird auch als KOI-7-Bit-Kode oder ASCII-Kode bezeichnet.)

In der Tabelle stehen jeweils links bekannte Zeichen unserer Schriftsprache (zuerst Ziffern, danach Buchstaben und Sonderzeichen), rechts daneben findest du ihre Verschlüsselung in eine Dezimalzahl, dann in eine Binärzahl und in ein Hell-dunkel-Muster. Mit 7 Lampen, die entweder brennen oder nicht brennen, kann das 7-Bit-Muster angezeigt werden.

| ... werden mit dem ISO-7-Bit-Kode verschlüsselt in  |                      |                    |                        |  |
|---|----------------------|--------------------|------------------------|--|
|   | die<br>Dezimalzahlen | die<br>Binärzahlen | die Hell-dunkel-Muster |  |
| <div>Diese Zeichen<br/>unserer Schriftsprache</div> | 0                    | 48                 | 0110000                |  |
|   | 1                    | 49                 | 0110001                |  |
|   | 2                    | 50                 | 0110010                |  |
|   | 3                    | 51                 | 0110011                |  |
|   | 4                    | 52                 | 0110100                |  |
|   | 5                    | 53                 | 0110101                |  |
|   | 6                    | 54                 | 0110110                |  |
|   | 7                    | 55                 | 0110111                |  |
|   | 8                    | 56                 | 0111000                |  |
|   | 9                    | 57                 | 0111001                |  |

Diese Zeichen  
unserer Schriftsprache

... werden mit dem ISO-7-Bit-Kode verschlüsselt in

|   | die<br>Dezimalzahlen | die<br>Binärzahlen | die Hell-dunkel-Muster |
|---|----------------------|--------------------|------------------------|
| A | 65                   | 1000001            |                        |
| B | 66                   | 1000010            |                        |
| C | 67                   | 1000011            |                        |
| D | 68                   | 1000100            |                        |
| E | 69                   | 1000101            |                        |
| F | 70                   | 1000110            |                        |
| G | 71                   | 1000111            |                        |
| H | 72                   | 1001000            |                        |
| I | 73                   | 1001001            |                        |
| J | 74                   | 1001010            |                        |
| K | 75                   | 1001011            |                        |
| L | 76                   | 1001100            |                        |
| M | 77                   | 1001101            |                        |
| N | 78                   | 1001110            |                        |
| O | 79                   | 1001111            |                        |
| P | 80                   | 1010000            |                        |
| Q | 81                   | 1010001            |                        |
| R | 82                   | 1010010            |                        |
| S | 83                   | 1010011            |                        |
| T | 84                   | 1010100            |                        |
| U | 85                   | 1010101            |                        |
| V | 86                   | 1010110            |                        |
| W | 87                   | 1010111            |                        |
| X | 88                   | 1011000            |                        |
| Y | 89                   | 1011001            |                        |
| Z | 90                   | 1011010            |                        |



|  |                       | ... werden mit dem ISO-7-Bit-Kode verschlüsselt in |                    |                           |
|--|-----------------------|--|--------------------|---------------------------|
|  |                       | die<br>Dezimalzahlen                               | die<br>Binärzahlen | die<br>Hell-dunkel-Muster |
| Sonderzeichen für Texte  |                       |  |                    |                           |
| Auch kein Schriftzeichen (Abstand) ist für den Computer ein wichtiges Zeichen! |                       |  |                    |                           |
|  | Leerzeichen           | 32   | 0100000            |                           |
|  | , (Komma)             | 44   | 0101100            |                           |
|  | . (Punkt)             | 46   | 0101110            |                           |
|  | : (Doppelpunkt)       | 58   | 0111010            |                           |
|  | ; (Semikolon)         | 59   | 0111011            |                           |
|  | " (Anführungszeichen) | 34   | 0100010            |                           |
|  | ! (Ausrufezeichen)    | 33   | 0100001            |                           |
|  | ? (Fragezeichen)      | 63   | 0111111            |                           |
| Sonderzeichen für die Mathematik   |                       |  |                    |                           |
|  | + (plus)              | 43   | 0101011            |                           |
|  | - (Strich, minus)     | 45   | 0101101            |                           |
|  | × (Stern, mal)        | 42   | 0101010            |                           |
|  | / (geteilt durch)     | 47   | 0101111            |                           |
|  | = (ist gleich)        | 61   | 0111101            |                           |
|  | < (kleiner als)       | 60   | 0111100            |                           |
|  | > (größer als)        | 62   | 0111110            |                           |

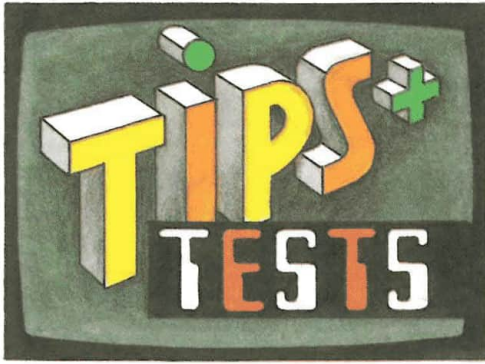


hell



dunkel





## Zusätzliche Tips und Tests für Pfiffige (mit Lösungen)

AUFGABE: Schreibe drei Leerzeichen dezimal im benutzten Kode!

LÖSUNG: 32 32 32

AUFGABE: Verschlüssele  $1+1=?$  in den angegebenen Zeichenkode (als Dezimalzahl)!

LÖSUNG: 49 43 49 61 63

AUFGABE: Kodiere  $1+1=?$  in den angegebenen Zeichenkode (als Binärzahl)!

LÖSUNG: 0110001 0101011 0110001 0111101  
0111111

AUFGABE: Die Zeichenfolge  $1+1=?$  soll im angegebenen Zeichenkode mit  $5 \cdot 7 = 35$  nebeneinanderstehenden Lampen angezeigt werden. Die Binärziffer 1 soll als „X“, die Ziffer 0 als „-“ erscheinen. Zeichne das Muster!

LÖSUNG: -XX---X  
-X-X-XX  
-XX---X  
-XXXX-X  
-XXXXXX

AUFGABE: Kodiere  $1 + 1 = ?$  in den benutzten Zeichenkode (als Dezimalzahl)! (Beachte die Leerräume zwischen den Zeichen!)

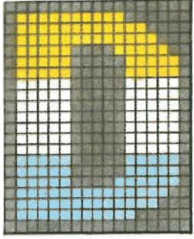
LÖSUNG: 49 32 43 32 49 32 61 32 63

AUFGABE: Verschlüssele die Zeichen „1.Klasse“ in den benutzten Zeichenkode (als Dezimalzahl)!

LÖSUNG: 49 46 75 76 65 83 83 69

AUFGABE: Auf dem Schulhof erhielt Susi folgende geheime Nachricht zugesteckt: 73 67 72 32 76 73 69 66 69 32 68 73 67 72 (im KOI-7-Bit-Kode dezimal verschlüsselt). Entschlüssele die Nachricht!

LÖSUNG: ICH LIEBE DICH

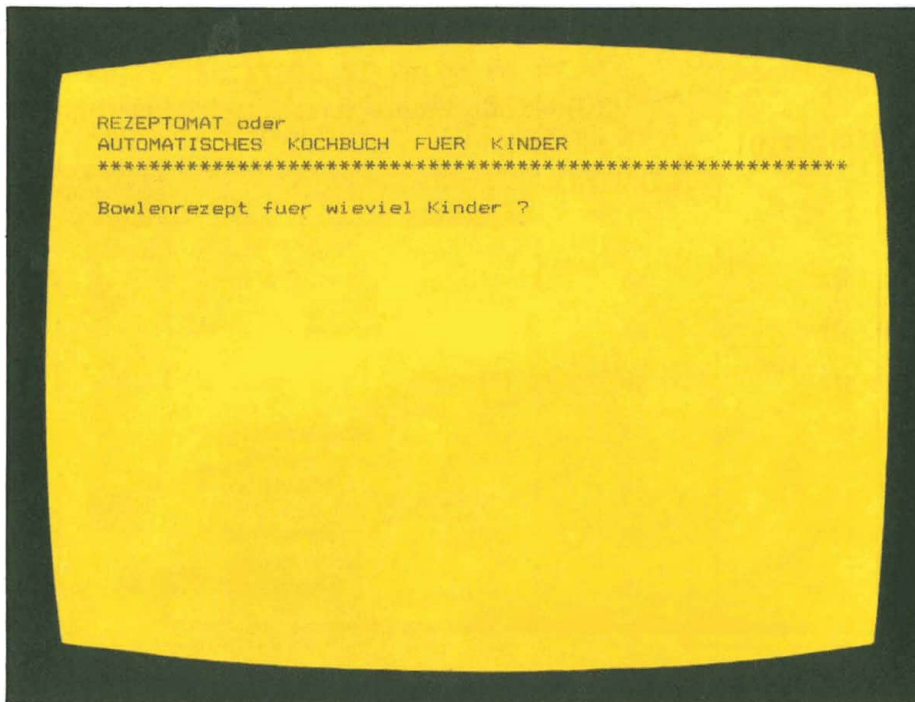


## Ein fertiges Computerprogramm

Was passiert am Bildschirm, wenn ein fertiges Programm abläuft?

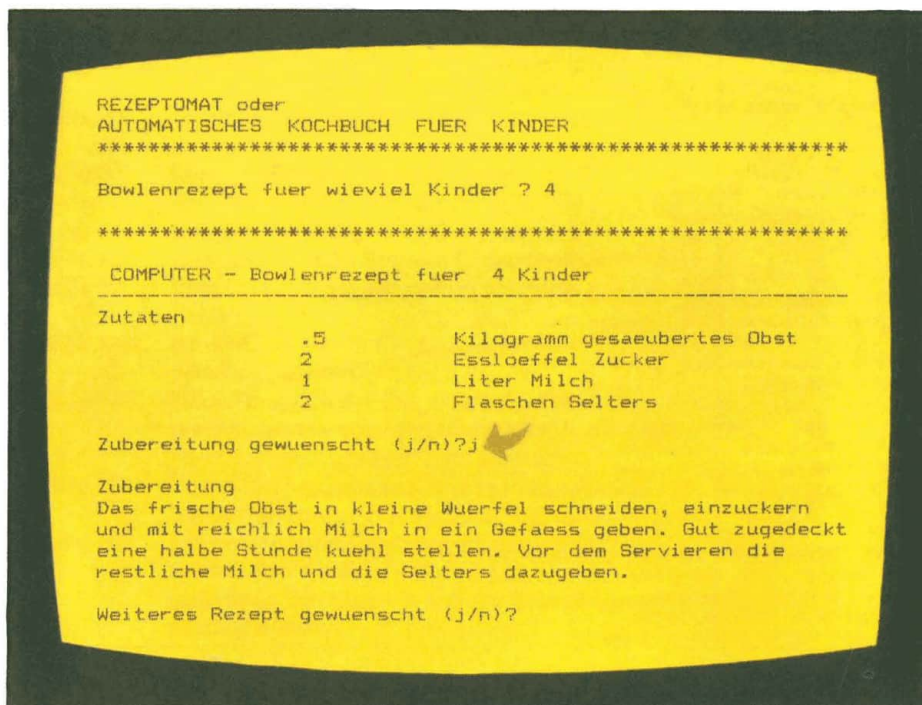
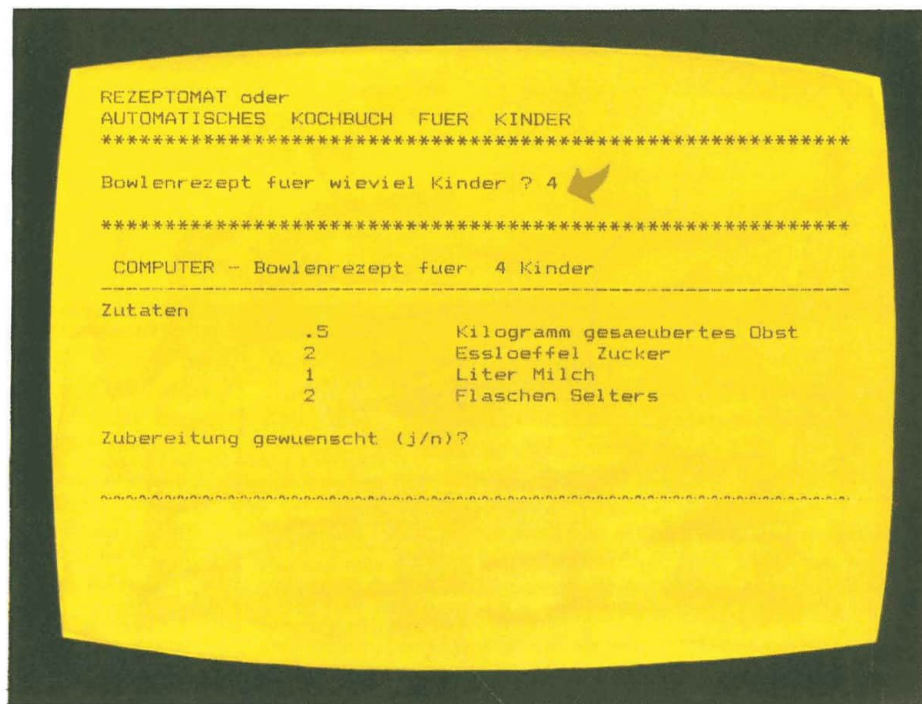
Das Programm auf der Seite 148 kann auf einem Computer, der die Programmiersprache BASIC versteht, laufen. Zuvor muß es jedoch in den Arbeitsspeicher geladen werden. Steht das Programm mit

dem Namen REZEPT schon auf einer Diskette oder einer Bandkassette, so genügt eine Anweisung wie BASIC REZEPT. Andernfalls muß es fehlerfrei und genau so wie auf Seite 148 abgedruckt mit der Tastatur eingegeben werden und kann dann mit RUN gestartet werden.



Das erscheint auf dem bisher leeren Bildschirm ganz oben.

Wir wünschen das Rezept für 4 Kinder, geben deshalb eine 4 ein und betätigen die ENTER-Taste. Rasch wird auf den Bildschirm weitergeschrieben.



Wir möchten wissen, wie die Kinderbowle zubereitet wird, und tippen das j (= ja) ein. Sofort schreibt der Computer weiter auf den Bildschirm (der Titel verschwindet, er „rollt“ nach oben).

Fortsetzung auf Seite 150



Das ist ein fertiges  
Computerprogramm, geschrieben  
in der Programmiersprache  
BASIC.

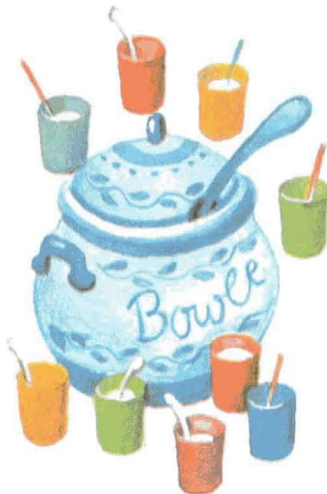
Zeilennummer


Schlüsselwort jeder Anweisung

```

10 REM -----R e z e p t o m a t-----
20 REM Das Programm ermittelt die Zutaten fuer eine Kinderbowle
30 REM und gibt nach Wunsch am Bildschirm aus, wie sie zubereitet wird.
40 REM Hinweis: Benutzt wird die STRING$ - Funktion und OR-Operation
50 REM -----
60 REM-----Variable: K - Anzahl Kinder, Z$,W$ - Antworten des Nutzers
70 REM-----Vorbereitungen
80 PRINT CHR$(12)          'Bildschirmloeschen wie mit CLS
90 PRINT"REZEPTOMAT oder"
100 PRINT"AUTOMATISCHES KOCHBUCH FUEER KINDER"
110 PRINT STRING$(66,"*")
120 PRINT
130 REM---- Eingabe der Personen und Pruefung
140 INPUT "Bowlenrezept fuer wieviel Kinder ";K
150 IF K>0 THEN 190
160 PRINT "Bitte Zahl groesser 0 eingeben !"
170 GOTO 140
180 REM----Rezept Titel
190 PRINT
200 PRINT STRING$(66,"*")
210 PRINT
220 PRINT" COMPUTER - Bowlenrezept fuer "; K ;"Kinder"
230 PRINT STRING$(66,"-")
240 REM
250 REM----Zutaten berechnen
260 PRINT"Zutaten"
270 PRINT ,K/8," Kilogramm gesaeubertes Obst"
280 PRINT ,K/2," Essloeffel Zucker"
290 PRINT ,K/4," Liter Milch"
300 PRINT ,K/2," Flaschen Selters"
310 PRINT
320 REM----Abfrage der Zubereitung
330 INPUT "Zubereitung gewuenscht (j/n)";Z$
340 IF Z$<>"j" THEN 440
350 REM----Ausgabe der Zubereitung
360 PRINT
370 PRINT"Zubereitung"
380 PRINT"Das frische Obst in kleine Wuerfel schneiden, einzuckern"
390 PRINT"und mit reichlich Milch in ein Gefaess geben. Gut zugedeckt"
400 PRINT"eine halbe Stunde kuehl stellen. Vor dem Servieren die"
410 PRINT"restliche Milch und die Selters dazugeben."
420 PRINT
430 REM---- Fortsetzungsabfrage
440 INPUT "Weiteres Rezept gewuenscht (j/n)";W$
450 IF (W$="j") OR (W$="J") THEN 80
460 REM----Schlussausgabe
470 PRINT STRING$(66,"*")
480 PRINT" REZEPTOMAT - Ende"

```





So ist das Programm  
zu verstehen.

Alle Zeilen mit dem Schlüsselwort REM beachtet der Computer nicht (hinter REM stehen Erläuterungen für den Programmierer). In Zeilen mit PRINT wird dem Computer befohlen, den nachfolgenden Text auf den Bildschirm zu schreiben. Gelangt der Computer zu einer Zeile mit INPUT, so soll er warten, bis wir Daten von der Tastatur eingetippt haben. In Zeilen mit IF soll der Computer eine Auswahl treffen: Er prüft zunächst einen Wert und setzt das Programm – abhängig vom Ergebnis der Prüfung – mit einer anderen Zeile fort, die nach dem Wort THEN steht.

Zeilen 10 bis 50 Der Programmname wird genannt und dem Leser gesagt, was das Programm leistet. Auch das Herstellungsdatum und der Autor des Programms werden eingetragen. Auf benutzte Besonderheiten der BASIC-Sprache wird hingewiesen.

Zeile 60 Die benutzten Variablen (Speicherplätze) werden erläutert.

Zeile 70 ist eine Überschrift für die nachfolgenden Zeilen 80 bis 120.

Zeile 80 Lösche alles auf dem Bildschirm.

Zeilen 90 bis 100 Gib Text auf dem Bildschirm aus, der das Programm ankündigt und kurz erläutert.

Zeile 110 Eine Zeile mit 66 Sternen ist anzuzeigen.

Zeile 120 Eine Leerzeile ist auf dem Bildschirm auszugeben.

Zeile 130 ist die Überschrift für den nächsten Programmabschnitt (Zeilen 140 bis 170).

Zeile 140 Nach Ausgabe der Frage soll der Computer warten, bis wir die Anzahl der Kinder eingeben und die Endetaste betätigt haben.

Zeile 150 Wenn die Anzahl der Kinder größer als Null ist, hat der Computer mit der Zeile 190 fortzusetzen.

Zeile 160 Ein Fehler wurde festgestellt (eine negative Anzahl von Kindern wurde versehentlich eingegeben). Ein Hinweis soll am Bildschirm angezeigt werden.

Zeile 170 Der Computer soll in Zeile 140 fortsetzen.

Zeile 180 Wieder eine Erläuterung zum nachfolgenden Programmteil (Zeilen 190 bis 240).

Zeile 220 Der Titel für das Rezept ist auf dem Bildschirm auszugeben.

Zeile 230 Ein Strich wird hier erzeugt und ist am Bildschirm auszugeben (vgl. Zeile 110).

Zeile 250 Eine Erläuterung zum nachfolgenden Programmteil.

Zeile 260 Die Überschrift für die Zutaten wird angezeigt.

Zeilen 270 bis 300 Aus der Anzahl K der Kinder sind die notwendigen Zutaten zu berechnen und am Bildschirm anzuzeigen.

Zeile 310 Eine Leerzeile ist auszugeben.

Zeile 320 Erläutert die nächsten beiden Zeilen.

Zeile 330 Der Computer soll warten, bis wir auf der Tastatur die Antwort (j von ja oder n von nein) eingegeben haben.

Zeile 340 Wenn die Antwort nicht j lautet, soll der Computer zu Zeile 440 gehen.

Zeile 350 Wieder eine Überschrift für den folgenden Programmabschnitt.

Zeilen 360 bis 420 Die Zubereitung ist am Bildschirm anzuzeigen.

Zeile 430 Erläutert die nächsten beiden Programmzeilen.

Zeile 440 Der Computer soll fragen, ob er ein weiteres Rezept ausgeben soll, und wartet, bis wir die Antwort (j oder n) eingegeben haben.

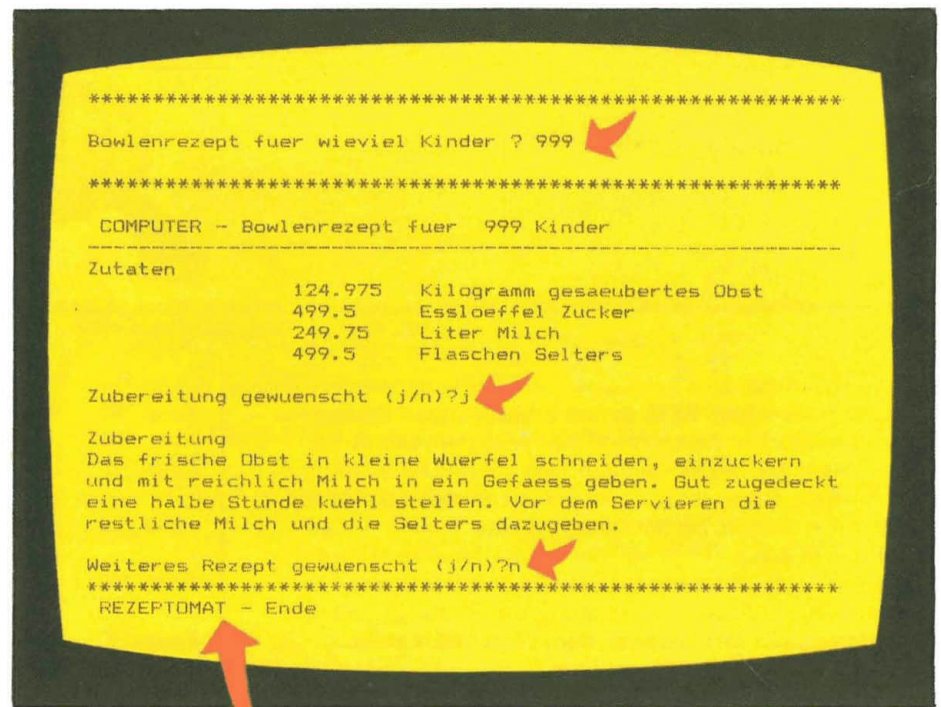
Zeile 450 Wenn die Antwort j oder J lautet, ist das Programm bei Zeile 80 fortzusetzen.

Zeile 460 Die letzten beiden Programmzeilen werden erklärt.

Zeile 470 Wieder sind – wie in der Zeile 110 – eine Reihe Sterne zu erzeugen und anzuzeigen.

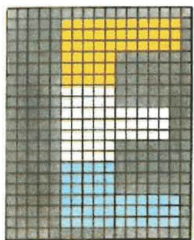
Zeile 480 Der Computer soll dem Nutzer die erfolgreiche Abarbeitung des Programms auf dem Bildschirm mit der Ausschrift „REZEPTOMAT – Ende“ mitteilen.

Nun wissen wir Bescheid: Doch welche Zutaten benötigen wir für ein großes Kinderfest mit 999 Kindern? Für den REZEPTOMATEN ist das kein Problem. Wir müssen die letzte Frage auf dem Bildschirm mit j beantworten, dann die Zahl 999, j und n eingeben und natürlich die ENTER-Taste drücken. Im Nu steht am Bildschirm das Ergebnis. Die Vorbereitungen für das Fest können beginnen ...



Dieses Zeichen sagt uns: Der Computer hat das Programm abgearbeitet und erwartet einen neuen Auftrag. Bekommt er keinen, wartet er, bis er abgeschaltet wird.





## Unser Demonstrationscomputer FIX

Vielleicht hast du durch das Buch Lust bekommen, einmal einen Computer auszuprobieren oder mit ihm zu spielen. Computer sind allerdings nicht billig und nicht überall zu haben. Wie kann dir geholfen werden?

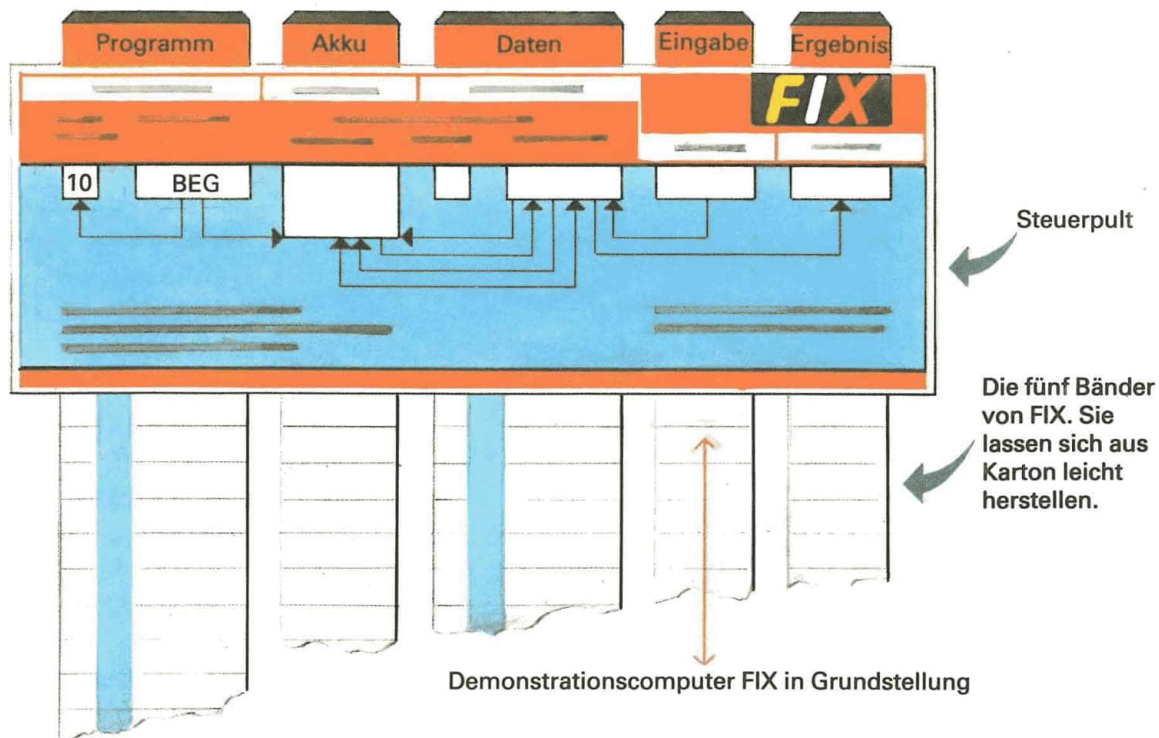
Versuche es einmal mit dem Demonstrationscomputer FIX. Du kannst ihn ohne große Mühe selbst basteln. Für FIX lassen sich kleine Programme schreiben, die du mit ihm ausprobieren kannst. Dabei lernst du viel über die Arbeitsweise eines Computers. Du

kannst auch Freunde, Freundinnen oder deine Eltern einladen, mit dir Computer zu spielen. Gemeinsam macht's mehr Spaß!

FIX arbeitet nur mit Karton und Bleistift oder Kugelschreiber. Wichtig ist, mit Geduld und Sorgfalt Schritt für Schritt vorzugehen. Gedankensprünge sind nicht erlaubt.

Der dem Buch beigelegte Bastelbogen enthält das Steuerpult und verschiedene Bänder.

Erläuterungen und wichtige Tips folgen jetzt.



## BASTELANLEITUNG

FIX besteht aus dem Steuerpult sowie zwei breiten und drei schmalen Streifen – den Bändern.

So wird FIX zusammengebaut:

1. Alle Teile sorgfältig ausschneiden.
2. Prüfen, ob die ausgeschnittenen Bänder in die vorgestanzten Schlitzte passen (vergleiche mit dem Bild auf Seite 151!). Eventuell nacharbeiten.
3. Die Laufstege A an der gestrichelten Linie nach der unbedruckten Seite des Steuerpultes knicken.
4. Die Rückwand B an der gestrichelten Linie gleichfalls nach der unbedruckten Seite knicken.
5. Nun die Laufstege A an der unteren Seite mit Klebstoff dünn bestreichen.
6. Die Stege A so auf die Rückseite des Steuerpultes C kleben, daß sie genau parallel zu den Seitenkanten des Steuerpultes verlaufen (zwischen den Stegen sollen sich die Bänder bewegen lassen).
7. Die angeklebten Stege auf der anderen Seite dünn mit Klebstoff bestreichen.

8. Die Rückwand B auf die Stege kleben und alles eine Weile pressen, zum Beispiel mit alten Büchern.
9. Alle Bänder in der Reihenfolge und Richtung, wie es das Bild zeigt, ins Steuerpult einschieben.
10. Letzte Prüfung des Werkes: Jedes Band läuft durch den vorgestanzten Schlitz unten in das Steuerpult und kommt oben zwischen den jetzt unsichtbaren Laufstegen A wieder heraus – sofern du sorgfältig montiert hast.
11. Nun etwas zurücktreten, das fertige Werk bewundern und dann ausprobieren!

### FIX besteht wie jeder Computer aus:

- dem Leitwerk
- dem Programmspeicher
- dem Rechenwerk
- dem Datenspeicher
- der Eingabe
- der Ausgabe.

Alle Teile befinden sich auf dem Steuerpult.





## GEBRAUCHSANLEITUNG

So wird FIX zum Start vorbereitet:

Alle Bänder in Grundstellung bringen (siehe Bild auf Seite 151). Damit ist zugleich die Startadresse des Programms im Befehlszähler eingestellt, und der erste Befehl muß im Befehlsregister stehen: BEG (mit diesem Befehl fängt jedes Programm für FIX an).

### So bringen wir FIX

**mit einem fertigen Programm zum Laufen:**

1. Die Anweisung im Befehlsregister lesen und verstehen! Ist dir etwas unklar, dann schau auf die Pfeile am Steuerpult oder in die Befehlsliste auf den Seiten 155/156.
2. Bringe die Daten, welche für die Anweisung gebraucht werden, herbei! Zum Beispiel: Beim Befehl L x muß im Datenspeicher nach der Größe mit dem Namen x gesucht werden und das Datenband auf diese Adresse eingestellt werden.
3. Führe den Befehl mit den Daten aus!
4. Ermittle die Nummer des nächsten Befehls und verschiebe das Programmband so, daß dieser Befehl im Befehlsregister steht!
5. Setze bei 1. fort bis zum Befehl END! Hier ist der Programmablauf zu Ende.

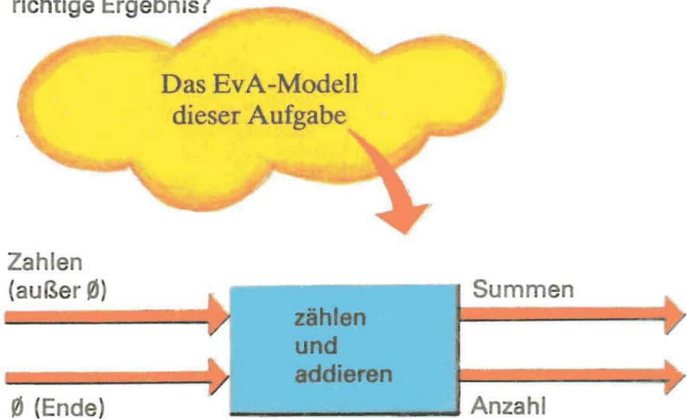
### Weitere Hinweise

- Die genaue Beschreibung der Befehle und dessen, was auszuführen ist, befindet sich auf Seite 155/156.
- Den Inhalt einer Adresse im Datenspeicher findest du, indem du das Datenband so verschiebst, bis die Adresse im Adreßfeld steht.
- Beim GO-Befehl (einem unbedingten Sprung) wird das Programmband so verschoben, daß die neue Adresse im Befehlszähler steht. Das gilt auch für die bedingten Sprünge (TH- oder EL-Befehl), wenn die Sprungbedingung (gleich beziehungsweise ungleich) erfüllt ist. Näheres steht auf Seite 155.

Um dir das Kennenlernen von FIX zu erleichtern, lie-

### Die Programmieraufgabe

Beliebig viele Zahlen sind zu zählen und zu addieren. Bei Eingabe einer Null ist das Zählen und Addieren zu beenden und das Ergebnis auszugeben. Schreibe ein Programm für FIX! Spiele mit FIX den Programmablauf nach! Liefert er das richtige Ergebnis?



gen eine Programmieraufgabe, ihr Lösungsweg (Seite 154) und das fertige Programmband (Seite 156) schon bereit.

Außerdem findest du ein kurzes Eingabeband mit den Zahlen 100, 70 und 50 zum Ausprobieren. Was vorgegeben war, ist auf allen Bändern blau, was mit FIX herauskommt, rot eingetragen. Doch halt! Zuerst mußt du die Programmiersprache kennenlernen.

### DIE PROGRAMMIERSPRACHE FÜR FIX

Damit FIX seine Aufgaben ausführt, mußt du ihn programmieren. Dafür benutzt du eine ganz einfache, erfundene Programmiersprache, die noch viel leichter als BASIC ist. Ein Programm in dieser Sprache besteht aus untereinander geschriebenen Befehlen:

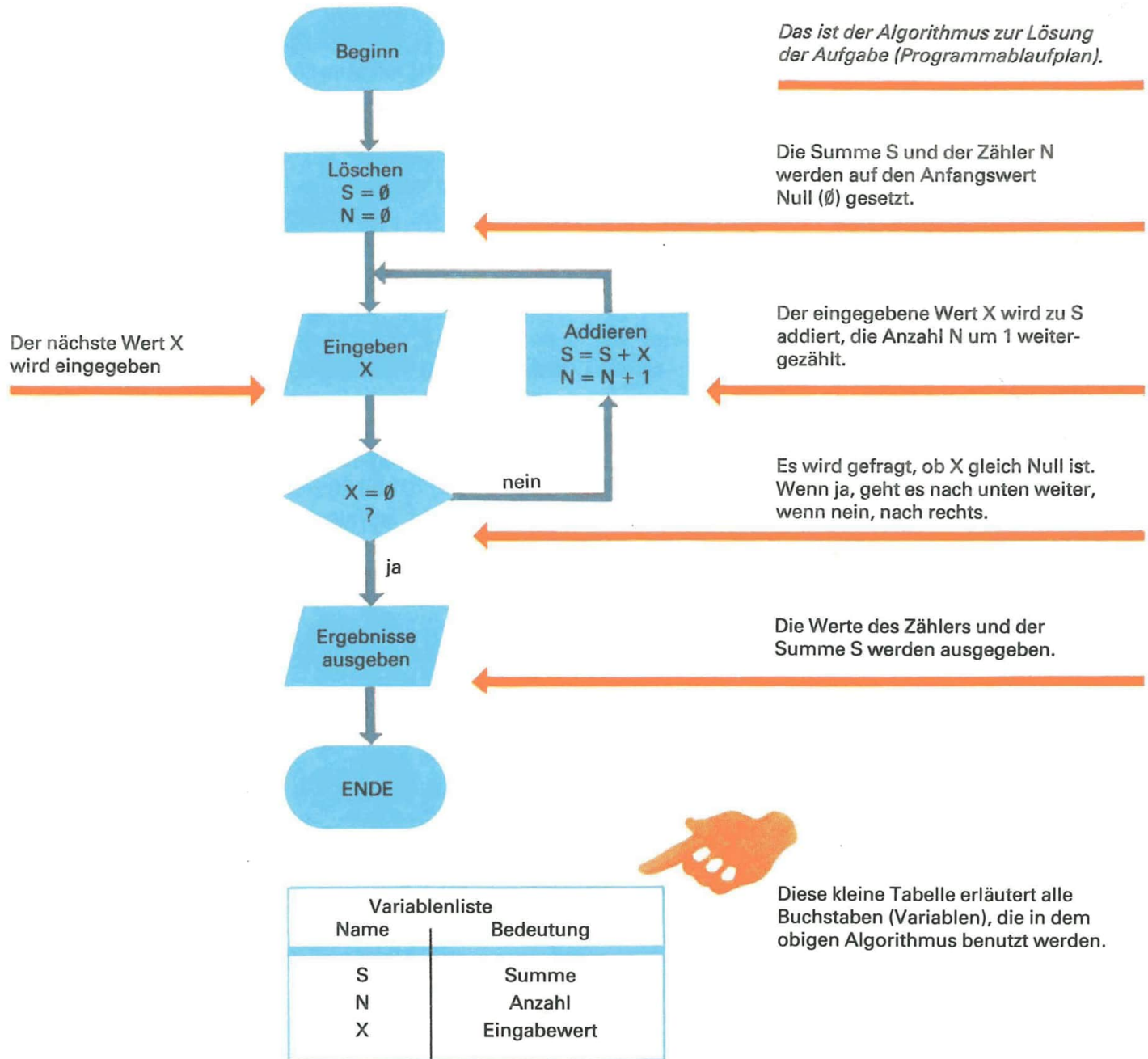
1. Befehl
2. Befehl
3. Befehl usw.

Solche Befehlsfolgen können auf das Programmband geschrieben werden. Jeder Befehl besteht aus:

|                  |  |   |
|------------------|--|---|
| Nummer           | vereinbartes Schlüsselwort des Befehls | einer Zahl, einem Buchstaben oder einem Namen |
| oder             | (1 bis 3 Großbuchstaben)               |   |
| Name des Befehls |  |   |



## Der Lösungsweg für die Aufgabe



Zum Beispiel: 13 L x

In der Zeile mit der Nummer 13 ist der Befehl L mit x auszuführen. Der Inhalt von Adresse x ist aus dem Datenspeicher zu lesen und in den Akkumulator zu schreiben. (Der Akkumulator, abgekürzt Akku, ist ein Hilfsspeicher zum Rechnen.)

Wie alle Befehle von FIX zu notieren sind und was sie bedeuten, findest du in der nachfolgenden Befehlsliste.

Vom Leitwerk werden die Befehle in ihre Bedeutung übersetzt. Du mußt sie ausführen. Dabei kannst du die Befehlsliste verwenden. Je nach Anweisung werden verschiedene Computerteile benutzt. Bei dem Befehl 13 L x suchst du auf dem Datenband die Adresse x, liest den Inhalt dieser Adresse und schreibst ihn in das Fenster des Akkumulators. Danach kann der nächste Befehl ausgeführt werden und so weiter.

Außer nach dem GO-, TH- und dem EL-Befehl wird stets der nächste Befehl ausgeführt. Dazu ziehst du das Programmband einfach ein Stück nach oben, bis der nächste Befehl im Befehlsregister erscheint. Aber immer nur Schritt für Schritt vorgehen und keinen Befehl überspringen – sofern das nicht ein Sprungbefehl verlangt.

## DIE BEFEHLSLISTE VON FIX

Jetzt schauen wir uns an, wie die Anweisungen geschrieben werden und was sie bedeuten.

### Befehle zum Transportieren von Daten

- L a Lies den Inhalt von Adresse a aus dem Datenspeicher und bringe ihn in den Akkumulator!
- S a Schreibe den Inhalt des Akkumulators in den Datenspeicher an die Adresse a!
- D w Bringe den Wert w direkt in den Akkumulator!

### Befehle zur Steuerung (und Änderung) des Programmablaufs (unbedingte und bedingte Sprünge)

- GO b Gehe weiter zu dem Befehl mit der Adresse b auf dem Befehlsband! Der Ablauf springt an eine andere Stelle des Programms.
- IF x Prüfe, ob der Wert von x gleich dem Wert im Akkumulator ist.
- TH b Ergab sich beim letzten IF *Gleichheit*, so gehe zu dem Befehl des Programmbandes mit der Nummer b! (Programmband so verschieben, bis Befehlszähler den Wert b anzeigt.)  
Ergab sich keine Gleichheit, so tue nichts.  
Der Ablauf des Programms springt also bedingt an eine andere Stelle.
- EL b Ergab sich beim letzten IF *Ungleichheit*, so gehe zu dem Befehl mit der Nummer b.  
Der Ablauf des Programms springt wie bei TH bedingt an eine andere Stelle des Programms.

(Ist keiner dieser Befehle auszuführen, wird das Programm mit dem nächsten Befehl des Programmbandes fortgesetzt.)

### Befehle zum Ein- oder Ausgeben von Daten

- E a Eingabe des nächsten Wortes vom Eingabeband in den Datenspeicher an die Adresse a.
- A a Ausgabe des nächsten Wortes von der Adresse a des Datenspeichers auf das Ausgabeband.

## Befehle zum Rechnen

(Taschenrechner dürfen verwendet werden.)

Die Ergebnisse aller folgenden Operationen werden in den Akkumulator eingetragen.

- + a Addiere zum Inhalt des Akkumulators den Inhalt der Adresse a des Datenspeichers!
- − a Subtrahiere vom Inhalt des Akkumulators den Inhalt der Adresse a des Datenspeichers!
- × a Multipliziere den Inhalt des Akkumulators mit dem Inhalt der Adresse a des Datenspeichers!

- / a Dividiere den Inhalt des Akkumulators durch den Inhalt der Adresse a des Datenspeichers!

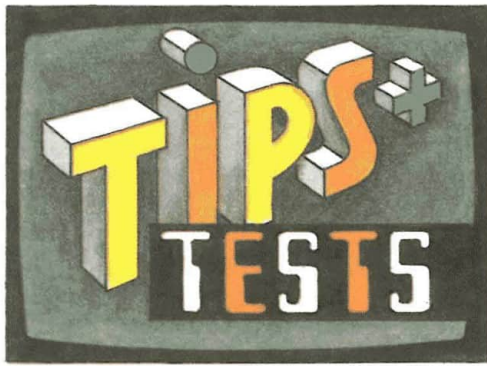
## Sonstige Befehle

- BEG zeigt uns den Beginn des Programms an.
- END zeigt uns das Ende des Programms an.
- REM Damit beginnt eine Erläuterung zum Programm, die keinen Einfluß auf die Abarbeitung und das Ergebnis hat, uns aber das Programm verständlicher macht.

| Programm-Band |                   | Akkumulator-Band | Daten-Band |                 | Eingabe-Band | Ergebnis-Band |
|---------------|-------------------|------------------|------------|-----------------|--------------|---------------|
| Adresse       | Inhalt            | Werte            | Adresse    | Inhalt          | Werte        | Werte         |
| 1             | BEG               | 0, 0, 100, 100   | EINS       | 1               | 100          | (N) 3         |
| 2             | REM Beispiel      | 0, 1, 0, 70, 70  | S          | 0, 100, 70, 220 | 70           | (S) 220       |
| 3             | REM SN-Löschen    | 1, 2, 0, 50, 220 | N          | 0, 1, 2, 3      | 50           |               |
| 4             | D 0               | 2, 3, 0          | X          | 100, 70, 50     | 0            |               |
| 5             | SS                |                  |            |                 |              |               |
| 6             | SN                |                  |            |                 |              |               |
| 7             | REM Input Eingabe |                  |            |                 |              |               |
| 8             | EX                |                  |            |                 |              |               |
| 9             | D 0               |                  |            |                 |              |               |
| 10            | JFX               |                  |            |                 |              |               |
| 11            | TH 21             |                  |            |                 |              |               |
| 12            | REM summieren     |                  |            |                 |              |               |
| 13            | LX                |                  |            |                 |              |               |
| 14            | +S                |                  |            |                 |              |               |
| 15            | SS                |                  |            |                 |              |               |
| 16            | LN                |                  |            |                 |              |               |
| 17            | +EINS             |                  |            |                 |              |               |
| 18            | SN                |                  |            |                 |              |               |
| 19            | GOB               |                  |            |                 |              |               |
| 20            | REM Ergebnisse    |                  |            |                 |              |               |
| 21            | AN                |                  |            |                 |              |               |
| 22            | AS                |                  |            |                 |              |               |
| 23            | END               |                  |            |                 |              |               |

Vor dem Programmablauf werden die Bänder vorbereitet; alle Anweisungen und Werte werden *blau* eingetragen. Beim Abspielen des Programms entstehen neue Werte; sie werden *rot* eingetragen. Um nicht die Ziffer 0 (Null) mit dem Buchstaben O (wie in Otto) zu verwechseln, wurden alle Nullen durchgestrichen (Ø).





## Tips zum Knobeln, Ausprobieren und Basteln

1. Probiere FIX aus, bis du auf genau die gleichen Ergebnisse kommst, die auf dem Akkumulator- und dem Ergebnisband des Programmbeispiels stehen!
2. Verändere den Programmablauf, indem du das Eingabeband änderst! (Verschiebe zum Beispiel die Null auf dem Eingabeband auf die zweite oder erste Stelle, verlängere das Eingabeband, indem du weitere Werte daraufschreibst.) Spiel den neuen Ablauf nach!
3. Verändere das Programm, indem du Befehle wegläßt (unerlaubt überspringst)! Was passiert?
4. Verändere das Programm, indem du Befehle vertauschst! Was geschieht?
5. Verändere das Programm, indem du neue Befehle hinzufügst! Was passiert?
6. Erweitere das Programm, indem du am Schluß auch den Mittelwert ( $S$  dividiert durch  $N$ , also  $S/N$ ) berechnest und ausgibst!
7. Schneide dir aus großkariertem Papier eigene Bänder! Suche dir eine Aufgabe zum Verarbeiten

eines Textes und schreibe ein Programm dafür. Prüfe deinen Lösungsweg und das Programm gemeinsam mit Freunden, Bekannten oder Eltern!

8. Zeige Ausdauer und Zähigkeit, wenn ein Programm nicht auf Anhieb richtig läuft! Tröste dich mit dem Wort erfahrener Programmierer: Ein Programm, das auf Anhieb läuft, ist sicher falsch!
  9. Erfinde selbst Befehle, die du vielleicht auf Taschenrechnern gesehen hast oder die dir helfen, kürzere Programme zu schreiben und das Programmieren zu erleichtern. Probiere diese Befehle in Algorithmen und Spielprogrammen aus, die du erfindest!
  10. Du willst den Demonstrationscomputer FIX für eine Arbeitsgemeinschaft basteln. Dieser FIX muß viel größer sein als der bereits vorhandene. Die Bänder sollen aus Hartpappe gefertigt werden. Wie muß FIX für diesen Zweck aussehen, damit die Bänder nicht von allein durchrutschen, sondern in der Einstellung verbleiben?
  11. Du kannst FIX auch verbessern. Beim Programmablauf muß man sich stets merken, wie die Prüfung in einem IF-Befehl ausging. Wie läßt sich das abstellen? (Hinweis: Baue in das Steuerpult zusätzlich einen Bit-Speicher ein, der das Prüfungsergebnis immer anzeigt. Es kann auch ein weiteres Band sein.)
  12. Hat sich der Leistungsstand eines Schülers, der Klasse, der Schule, aller Schulen verbessert oder verschlechtert oder blieb er unverändert? Wir können die Fragen beantworten, indem wir den Mittelwert der Zensuren ausrechnen und vergleichen. Dieser wird errechnet aus der Summe aller Zensuren, dividiert durch ihre Anzahl. Versuche diese Aufgabe mit FIX für deine Zensuren zu lösen! (Hinweis: Vergleiche 6.! Fertige dir das Eingabeband an.)
- Wer FIX richtig zum „Laufen“ bringt, ist schon fast ein Programmierer. Er wird sehr schnell eine höhere Programmiersprache wie BASIC erlernen und die schematische Arbeitsweise eines Computers verstehen.

# Inhalt

|  |     |
|--|-----|
| Vorwort  | 4   |
| 1. Eine unterhaltsame Spielerei                    | 5   |
| 2. Zwanzig Begegnungen mit Computern               | 9   |
| 3. Was Computer alles können                       | 15  |
| 4. Wir bauen ein Modell und benutzen es            | 19  |
| 5. Besichtigung einer Ausstellung von Computern    | 27  |
| 6. Drei Schlüssel zur Computerwelt                 | 37  |
| 7. Geheimnisse des Computergedächtnisses           | 47  |
| 8. Wir probieren einen Computer aus                | 57  |
| 9. Zwanzig Zusatzgeräte – genauer angeschaut       | 65  |
| 10. Wie programmiert man einen Computer?           | 74  |
| 11. Bewährtes verwenden, nicht neu erfinden!       | 87  |
| 12. Alles mit 0 und 1 beschreiben                  | 93  |
| 13. Im Computer passiert Erstaunliches             | 105 |
| 14. Von Berufen rund um den Computer               | 121 |
| 15. Ein Blick in die Zukunft des Computers         | 126 |
| Nachwort   | 135 |
| Anhang   | 136 |
| A Einige Antworten auf die Fragen des Wissenstests | 136 |
| B Kleines Wörterbuch für Neugierige                | 139 |
| C Ein weltbekannter Zeichenkode                    | 142 |
| D Ein fertiges Computerprogramm                    | 146 |
| E Unser Demonstrationscomputer FIX                 | 151 |

ISBN 3-358-01088-0



1. Auflage 1988

© DER KINDERBUCHVERLAG BERLIN – DDR 1988

Lizenz-Nr. 304-270/130/88

Gesamtherstellung: Grafischer Großbetrieb Sachsensdruck Plauen

LSV 7820

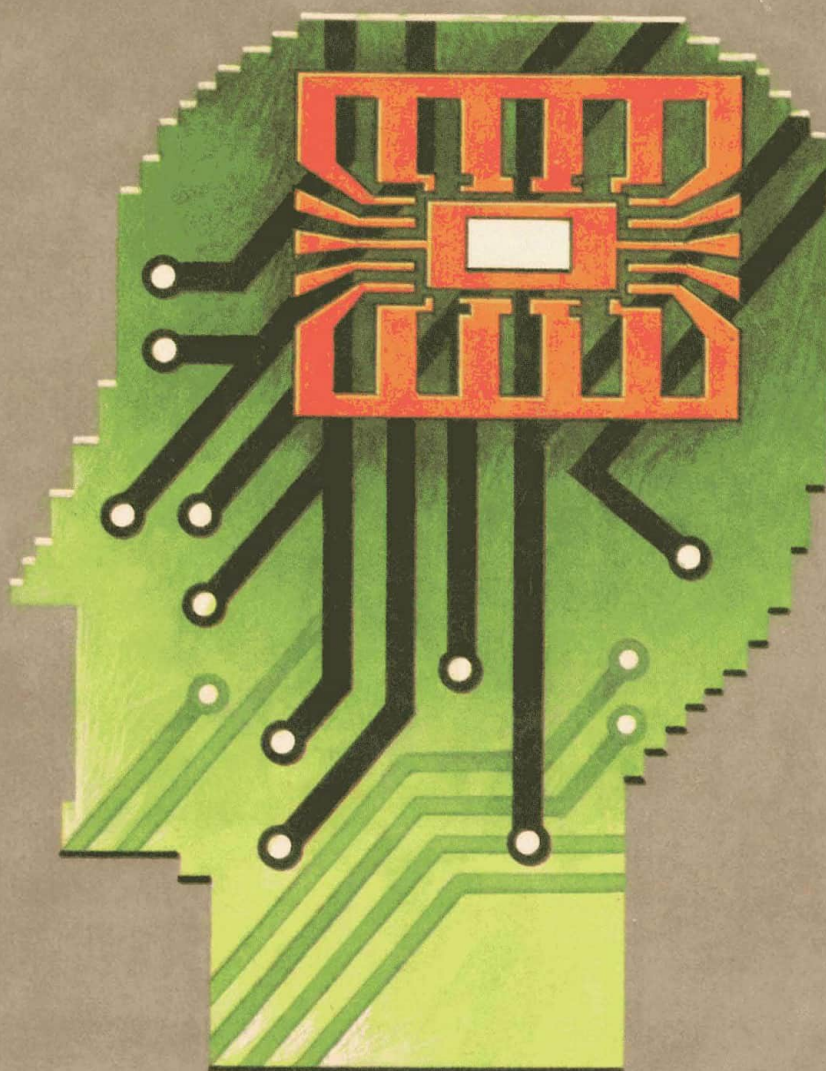
Für Leser von 12 Jahren an

Bestell-Nr. 632 752 7

01960



Dieses Buch lädt dich ein in die erstaunliche Welt des Computers. Auf unterhaltsame Art lernst du sie Schritt für Schritt kennen. Du erfährst, wie man mit einem Computer umgeht und ihn programmiert, wie er funktioniert, was er leisten kann und was er nicht vermag. Deine neuen Kenntnisse kannst du in Wissenstests überprüfen. Für interessante Knobeleien bekommst du vielerlei Tips, und ein Minilexikon erklärt dir wichtige Begriffe, die mit dem Computer zu tun haben. Am Ende hält das Buch noch eine Überraschung für dich bereit: einen Computer zum Ausschneiden. Du kannst ihn programmieren und nutzen.



ISBN 3-358-01088-0