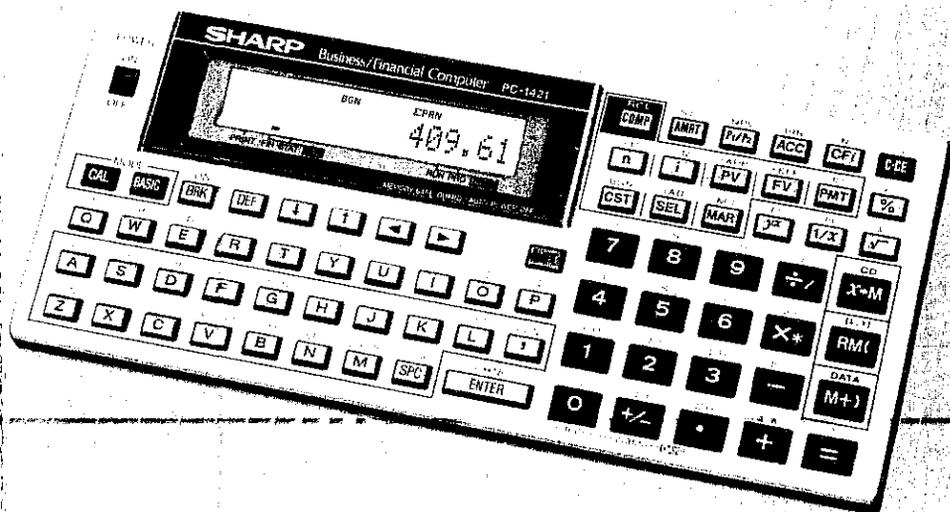


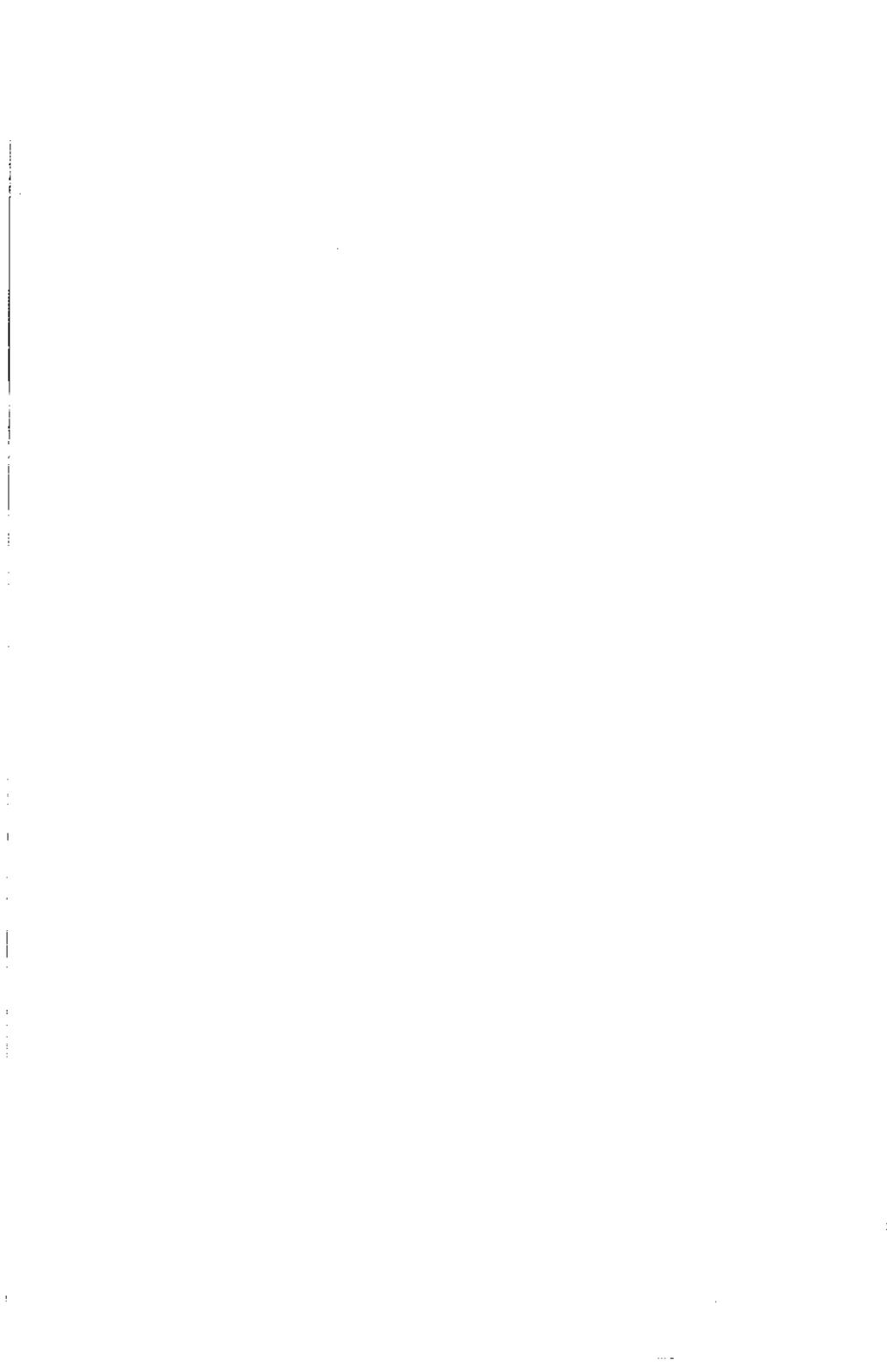
SHARP®

TASCHENCOMPUTER

MODELL **PC-1421**

BEDIENUNGSANLEITUNG





INHALTSVERZEICHNIS

	Seite
EINLEITUNG	4
KAPITEL 1 WIE MAN DIESES HANDBUCH BENUTZT	5
KAPITEL 2 VORSTELLUNG DES PC-1421	7
Beschreibung des Systems	7
Schalter und Tasten	8
Modi	9
Schalter ALL RESET	12
Austauschen der Batterien	14
KAPITEL 3 DER PC-1421 ALS TASCHE NRECHNER	17
Einschalten	17
Ausschalten	17
Automatische Abschaltung	17
Rechnen im CAL-Modus	18
Lesen des Displays	20
Grundrechenarten	21
Höhere Rechenarten	23
Finanzielle Berechnungen	25
Statistische Berechnungen	32
1. Statistische Berechnungen mit einer Variablen	35
2. Statistische Berechnungen mit zwei Variablen und lineare Regression	36
Berechnungsbereich	39
Manuelles Rechnen im BASIC-Modus	41
Wie manuell gerechnet wird	41
Abrufen nach der Eingabe	42
Fehlermeldungen	46
Verkettung von Rechenoperationen	47
Negative Zahlen	48
Komplexere Rechenoperationen und Klammerung	49
Der Gebrauch von Variablen in Rechenoperationen	50
Rechenoperationen mit mehreren Rechenschritten	51
Wissenschaftliche Notation	52
Grenzen	53
Letztes-Ergebnis-Funktion	54
Wissenschaftliche Berechnungen im BASIC-Modus	56
Direkte Berechnung	62

	Seite
Priorität bei manuellen Berechnungen	62
KAPITEL 4 BASIC-BEGRIFFE UND -AUSDRÜCKE.	63
Stringkonstanten.	63
Variable.	64
Vorgegebene Variable.	65
Einfache Variable	66
Feldvariable	67
Variable der Form A ()	70
Ausdrücke	72
Numerische Operatoren	72
String-Ausdrücke.	72
Verhältnis-Ausdrücke.	73
Logische Ausdrücke.	74
Klammerung und Vorrang der Operatoren	76
RUN-Modus.	76
Funktionen	76
KAPITEL 5 PROGRAMMIEREN MIT DEM PC-1421	78
Programme.	78
BASIC-Statements.	78
Zeilennummern.	78
BASIC-Befehle	79
BASIC-Kommandos.	79
Modi.	79
Erstes Programmieren des PC-1421	80
Beispiel 1 — Eingeben und Ausführen eines Programms.	80
Beispiel 2 — Editieren eines Programms	81
Beispiel 3 — Der Gebrauch von Variablen beim Programmieren.	83
Beispiel 4 — Komplexere Programme.	85
Speichern von Programmen	87
KAPITEL 6 VERBINDUNG DER FINANZ-FUNKTIONEN MIT BASIC	88
Für BASIC verwendbare Tasten	88
Statements.	88
Finanzfunktionen	89
KAPITEL 7 VEREINFACHUNGEN	105
Die DEF-Taste und benannte Programme.	105
Schablonen	106
KAPITEL 8 DER GEBRAUCH DES CE-126P DRUCKER/ CASSETTENRECORDER-INTERFACE.	107

	Seite
Der Gebrauch des Druckers	107
Der Gebrauch des Cassettenrecorder-Interface	109
Anschluß des CE-126P an einen Cassettenrecorder	109
Cassettenrecorder	109
Bedienung des Interface und des Cassettenrecorders	111
Hinweise zu Cassetten	115
KAPITEL 9 BASIC-REFERENZTEIL	116
Kommandos	119
Befehle	129
Funktionen	174
Pseudovariablen	174
Numerische Funktionen	175
String-Funktionen	182
KAPITEL 10 PANNENHILFE	184
Bedienung des Geräts	184
Fehlersuche in BASIC	185
KAPITEL 11 DIE INSTANDHALTUNG DES PC-1421	187
ANHÄNGE	
Anhang A: Fehlermeldungen	188
Anhang B: Zeichencode-Tabelle	190
Anhang C: Formatieren der Datenausgabe	192
Anhang D: Bewertung von Ausdrücken und Vorrang der Operatoren	197
Anhang E: Tastenfunktionen im BASIC-Modus	199
Anhang F: Technische Daten	209
PROGRAMMBEISPIELE	211
STICHWORTVERZEICHNIS	283

EINLEITUNG

Willkommen in der Welt der **SHARP-Besitzer!**

Nur wenige Industriezweige der Welt können heute mit dem schnellen Wachstum und der technischen Entwicklung im Bereich der Personalcomputer Schritt halten. Computer, die noch vor wenigen Jahren einen ganzen Saal gefüllt hätten, deren Programmierung einen akademischen Abschluß erforderte und die viele Tausend Dollar kosteten, passen heute in Ihre Handfläche, sind einfach zu programmieren und kosten so wenig, daß sie für praktisch jedermann erschwinglich sind.

Ihr neuer **SHARP PC-1421** macht Ihnen die neuesten Errungenschaften der Computer-Revolution zugänglich. Er verfügt über eine Vielzahl fortschrittlichster Funktionen:

- * **FINANZCOMPUTER** — Vor dem **PC-1421** mußte man zwischen Anwendungen im Finanzbereich (einschließlich Statistik) und normalen Berechnungen unterscheiden. Jetzt nicht mehr! Der **PC-1421** ist gleichzeitig ein Finanzcomputer und ein Taschencomputer mit Finanz-Funktionen plus BASIC-Kommandotasten für einfaches Programmieren.
- * **SPEICHERSCHUTZ** — der **PC-1421** erhält gespeicherte Programme und Variable auch im ausgeschalteten Zustand.
- * **Batteriebetrieb** — Sie können den Computer überall und jederzeit verwenden.
- * **AUTOMATISCHE ABSCHALTUNG** — eine gewisse Zeit nach der letzten Tastenbetätigung wird das Gerät automatisch ausgeschaltet, um Batteriespannung zu sparen.
- * Eine erweiterte BASIC-Version, die formatierte Ausgabe, zweidimensionale Felder, variable Stringlängen und viele andere fortschrittliche Merkmale aufweist.
- * Als Option: Ein Drucker/Cassettenrecorder-Interface (Modell CE-126P) für Ausdruck von Programmen und Daten auf dem Drucker und zum Anschließen eines Cassettenrecorders, um Programme und Daten auf Cassetten zu speichern.

Wir gratulieren Ihnen zum Eintritt in eine aufregende und faszinierende neue Welt. Wir sind sicher, daß Sie diesen Kauf niemals bereuen werden. Der **SHARP PC-1421** ist ein leistungsfähiges Hilfsmittel, das Sie bei Ihren mathematischen, wissenschaftlichen, finanziellen, geschäftlichen und persönlichen Aufgaben wirkungsvoll unterstützt. Mit dem **SHARP PC-1421** legen Sie jetzt den Grundstein für die Lösung Ihrer Probleme von morgen.

KAPITEL 1 WIE MAN DIESES HANDBUCH BENUTZT

Dieses Handbuch soll Sie mit den Fähigkeiten und Funktionen Ihres **PC-1421** vertraut machen und Ihnen außerdem als wertvolles Nachschlagewerk dienen. Ob Sie nun Anfänger sind oder schon über Erfahrungen mit Computer verfügen, Sie sollten die Kapitel 2 bis 6 lesen und durcharbeiten, um den **PC-1421** gründlich kennenzulernen.

- * Kapitel 2 beschreibt die äußere Erscheinung des **PC-1421**.
- * Kapitel 3 erklärt den Gebrauch des **PC-1421** als Finanzrechner.
- * Kapitel 4 erläutert einige Ausdrücke und Begriffe, die für das Programmieren in BASIC von Wichtigkeit sind, und erörtert die spezielle Bedeutung dieser Begriffe für die Arbeit mit dem **PC-1421**.
- * Kapitel 5 macht Sie mit dem Programmieren in BASIC auf dem **PC-1421** bekannt und erklärt Eingeben, Korrigieren und Ausführen von Programmen.
- * Kapitel 6 demonstriert den Gebrauch der finanziellen Funktionen im BASIC-Modus.
- * Kapitel 7 diskutiert einige Vereinfachungen, die das Arbeiten mit dem Computer erleichtern, so daß es noch mehr Spaß macht.

Erfahrene BASIC-Programmierer sollten dann noch Kapitel 9 durchlesen, um die spezifischen Züge der BASIC-Implementierung für den **PC-1421** kennenzulernen. Da sich alle BASIC-Dialekte geringfügig unterscheiden, empfiehlt es sich, dieses Kapitel vor dem Programmieren auf dem **PC-1421** mindestens einmal durchzulesen.

Kapitel 9 ist ein Nachschlageteil, der alle Befehle, Kommandos und Funktionen des BASIC in übersichtlicher alphabetischer Anordnung enthält.

Falls Sie noch über keine Erfahrung mit Programmieren in BASIC verfügen, sollten Sie sich hierfür ein spezielles Lehrbuch zulegen oder an einem BASIC-Kurs teilnehmen, bevor Sie diese Kapitel durcharbeiten. Dieses Handbuch ist nicht als Lehrbuch für Programmieren konzipiert.

Der verbleibende Teil des Handbuchs besteht aus:

- * Kapitel 8 – Grundlegende Informationen über das als Option erhältliche CE-126P Drucker/Cassettenrecorder-Interface.
- * Kapitel 10 – Ein Leitfaden, der Ihnen helfen soll, Probleme bei der Bedienung und Programmierung zu lösen.
- * Kapitel 11 – Pflege und Wartung Ihres neuen Computers.

Wie man dieses Handbuch benutzt

Ein umfangreicher Anhang am Ende dieses Handbuchs stellt Ihnen nützliche Tabellen und spezielle Informationen bezüglich Gebrauch und Funktionsweise des PC-1421 zur Verfügung.

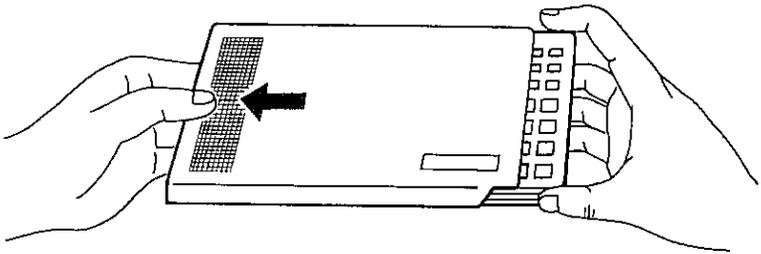
Gebrauch des Schutzdeckels

Dieser Schutzdeckel sollte bei Nichtgebrauch des Computers über die Tastatur angebracht werden.

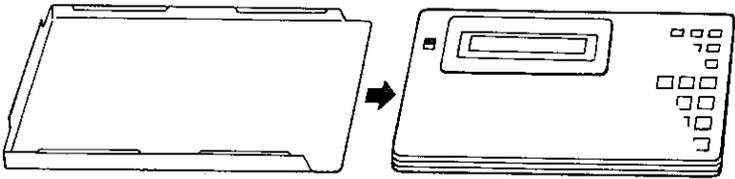
- Gebrauch des Computers.

Den Schutzdeckel wie unten gezeigt vom Computer abnehmen.

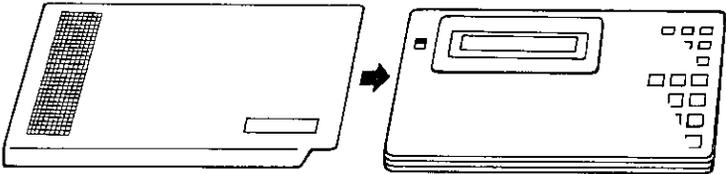
Schritt ①



Schritt ②



- Nichtgebrauch des Computers.

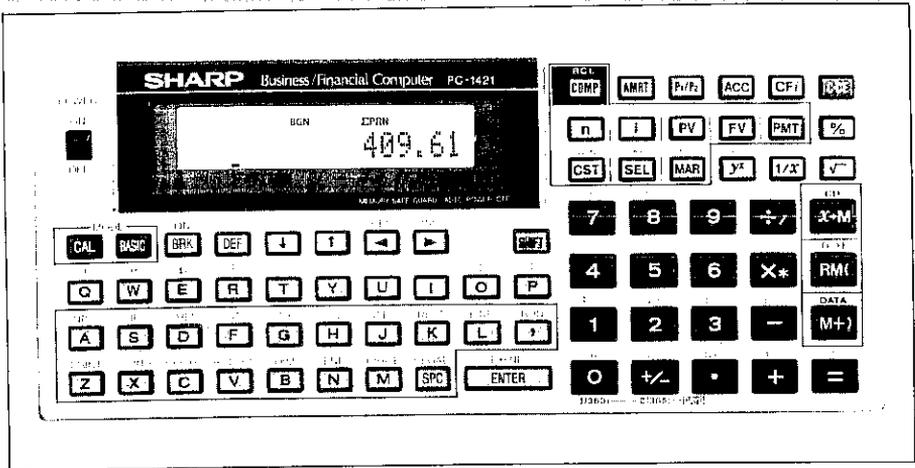


KAPITEL 2 VORSTELLUNG DES PC-1421

Beschreibung des Systems

Das System **SHARP PC-1421** besteht aus:

- * 76-Zeichen-Tastatur
- * 16-Zeichen-Display
- * Leistungsfähiges BASIC in 40 KB ROM
- * 8-Bit CMOS Prozessor
- * 4,2 KB RAM
- * Option: CE-126P Drucker/Cassettenrecorder-Interface



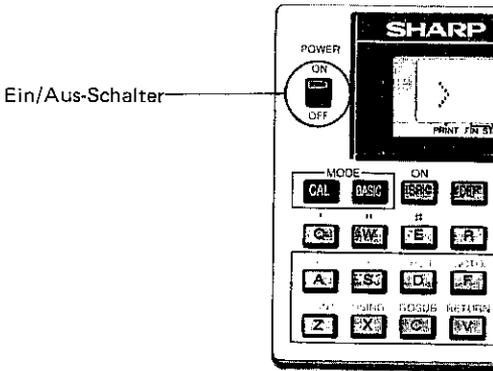
Schalter und Tasten

Der PC-1421 verfügt über 76 Tasten und einen Schiebeschalter. Jede Tastenfunktion ist durch verschiedene Zeichen, Zahlen oder Symbole gekennzeichnet, die sich auf oder über den Tasten befinden.

(1) Einschalten

Zuerst wird der Computer eingeschaltet.

Dafür wird der Ein/Aus-Schalter (POWER) links oben auf dem Computer auf die Position ON gestellt.



Modi

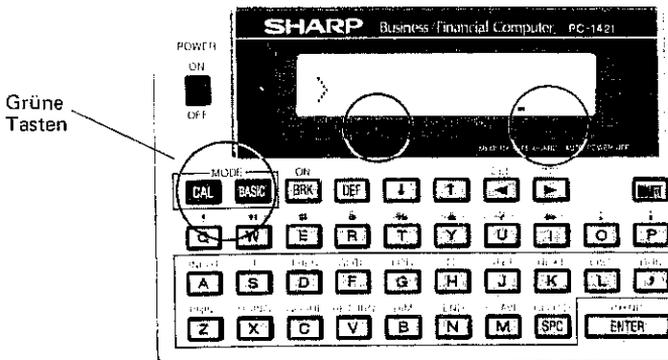
Der **PC-1421** kann grundsätzlich in zwei verschiedenen Modi arbeiten, **CAL-Modus** und **BASIC-Modus**. Der **CAL-Modus** ist in drei weitere Modi und der **BASIC-Modus** in zwei weitere Modi unterteilt.

- **CAL-Modus** (Taste **CAL** drücken)
 1. Normale Berechnungen (Keine Modus-Anzeige auf dem Display.)
 2. Finanzielle Berechnungen
(Über **FIN** leuchtet ein Strich (—) auf dem Display.)
 3. Statistische Berechnungen
(Über **STAT** leuchtet ein Strich (—) auf dem Display.)
- **BASIC-Modus** (Taste **BASIC** drücken)
 1. Programmierung
(Über **PRO** leuchtet ein Strich (—) auf dem Display.)
 2. Programm-Ausführung
(Über **RUN** leuchtet ein Strich (—) auf dem Display.)

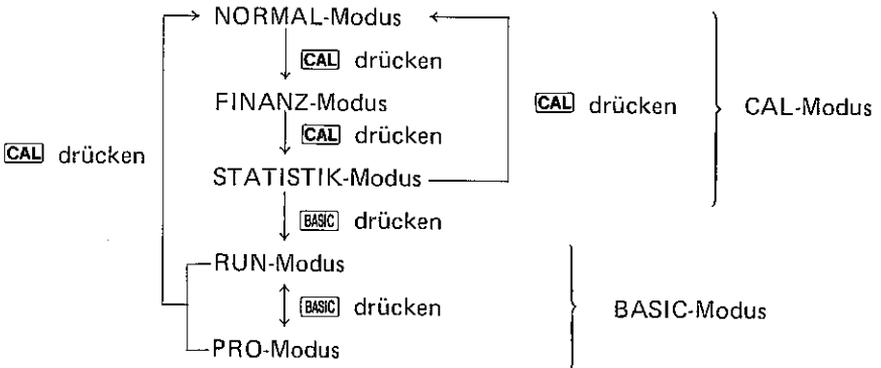
Der Rechen-Modus wird mit der Taste **CAL** und der **BASIC-Modus** mit der Taste **BASIC** umgeschaltet.

Der Computer schaltet zurück auf den **CAL-Modus**, wenn die Taste **CAL** im **BASIC-Modus** gedrückt wird.

Für die entgegengesetzte Umschaltung die Taste **BASIC** drücken.



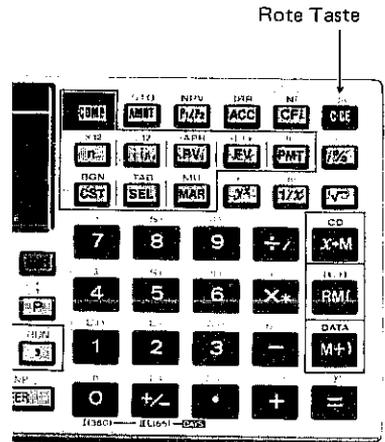
Umschalten der Modi



1. CAL-Modus

Wir wollen jetzt die Tasten betätigen. Setzen Sie zuerst den Modus auf CAL. Im CAL-Modus können die rechts abgebildeten Tasten und Funktionen für Berechnungen verwendet werden.

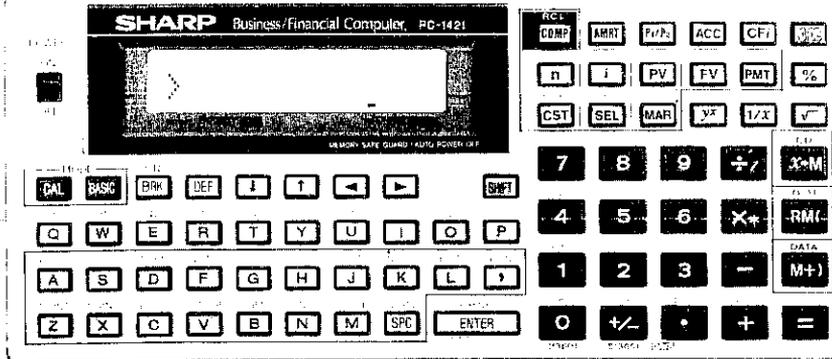
	Anzeige
1 2	→ 12.
+	→ 12.
3	→ 3.
=	→ 15.



2. BASIC-Modus

Wechseln Sie mit der Taste **BASIC** auf den RUN- oder PRO-Modus und betätigen die folgenden Tasten, während Sie auf das Display schauen:

Im RUN- und PRO-Modus können die unten abgebildeten Tasten für Berechnungen verwendet werden.



Beispiel:

PRINT Z	USING X	GOSUB C	→ ZXC _		
Σxy 1	Σy 2	Σx² •	Σy² 3	→ ZXC12.3 _	
CA C-CE			→ >		
INPUT A	≡	x̄ 4	x[^] +	Sx 5	→ A = 4 + 5 _

↑
Cursor

Durch Drücken einer alphabetischen oder numerischen Taste wird das auf der Taste angegebene Zeichen eingegeben. Zum Eingeben der Zeichen oder Symbole, die in brauner Farbe über den Tasten angegeben sind, muß die gelbe Taste **SHIFT** gedrückt gehalten und dann die jeweilige Taste gedrückt werden.

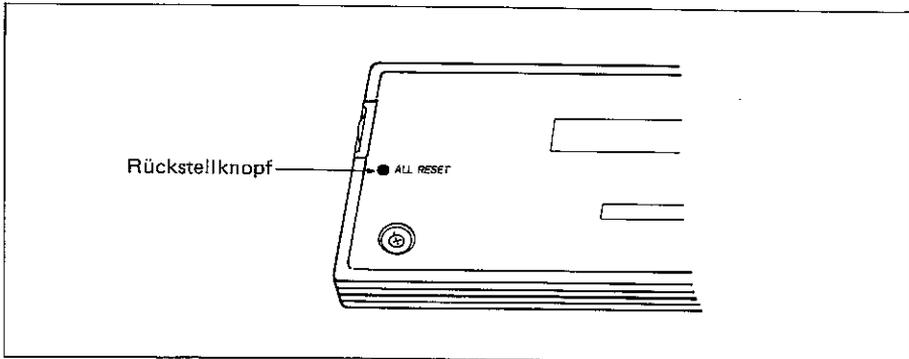
CA C-CE	SHIFT	PRINT Z	→ PRINT _	
SHIFT	W	RM(√	→ PRINT "(√ _

Die Taste **SHIFT** dient zum Eingeben der Zeichen oder Symbole, die in braun über Tasten mit zwei oder drei Funktionen angegeben sind. Durch wiederholtes Drücken der Taste **SHIFT** erscheint und verschwindet oben auf dem Display das

Vorstellung

Symbol SHIFT abwechselnd. Das Symbol SHIFT bedeutet, daß die Taste **SHIFT** betätigt wurde und die über den Tasten in braun angegebenen Zeichen eingegeben werden können.

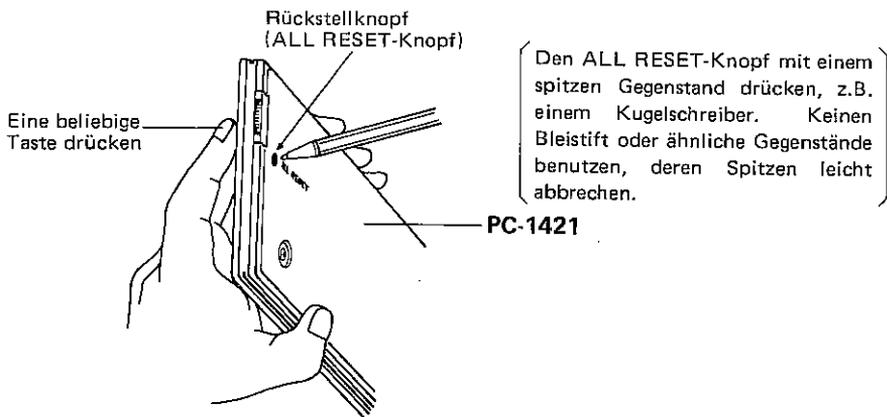
Schalter ALL RESET



Rückstellknopf: Dieser Knopf dient zum Initialisieren des Computers, wenn ein Problem nicht mit C•CE oder CA beseitigt werden kann.

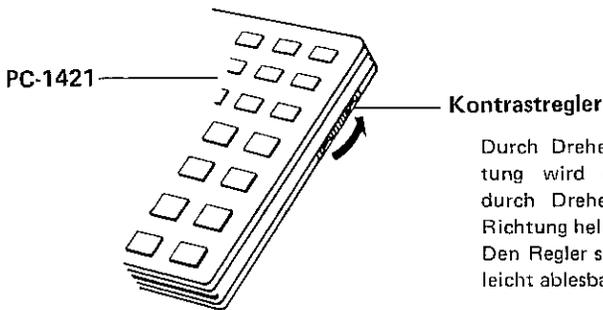
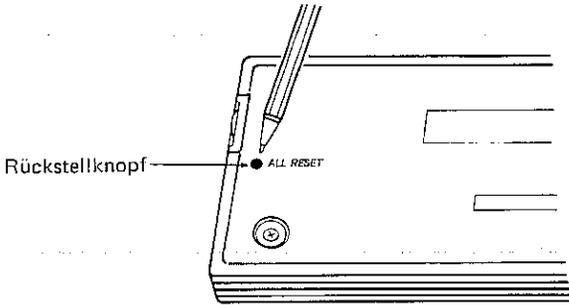
HINWEIS

- ① Zum Initialisieren des **PC-1421** eine beliebige Taste der Tastatur gedrückt halten und gleichzeitig den Rückstellknopf an der Geräterückseite drücken. Dadurch werden Programme und Variable nicht gelöscht.



- ② Wenn nach Durchführung der obigen Bedienung auf Betätigen von Tasten keine Reaktion erfolgt, den Rückstellknopf drücken, ohne gleichzeitig eine Taste gedrückt zu halten.

Durch diese Bedienung werden die Programme und Daten gelöscht, sie sollte daher nur durchgeführt werden, wenn es absolut erforderlich ist.



Durch Drehen des Reglers in Pfeilrichtung wird das Display dunkler und durch Drehen in die entgegengesetzte Richtung heller.

Den Regler so einstellen, daß das Display leicht ablesbar ist.

AUSTAUSCHEN DER BATTERIEN

Der **PC-1421** arbeitet ausschließlich mit Lithiumzellen.

Durch Beachten der folgenden Hinweise können Sie sich viele Probleme ersparen:

- Immer beide Batterien gleichzeitig austauschen.
- Niemals eine neue und eine schon gebrauchte Batterie zusammen verwenden.
- Benutzen Sie nur Lithiumzellen (Typ CR-2032), zwei Zellen sind erforderlich.

EINSETZEN DER BATTERIEN

Wenn das Display bei maximal eingestelltem Kontrast schwach leuchtet und schwer ablesbar ist, sind die Batterien verbraucht und sollten umgehend erneuert werden.

- (1) Den Schiebeschalter auf **OFF** stellen, um den Computer auszuschalten.
- (2) Mit einem kleinen Schraubenzieher die Schrauben auf dem rückwärtigen Gehäusedeckel entfernen. (Abb. 1)

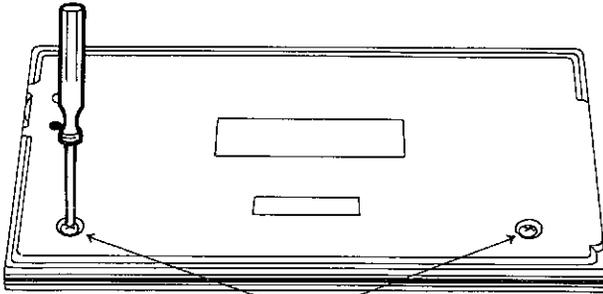


Abb. 1

Schrauben

- (3) Den Batteriefachdeckel in Pfeilrichtung schieben und abnehmen, siehe Abb. 2.

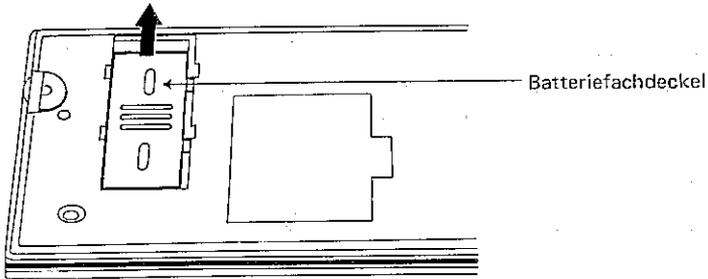


Abb. 2

- (4) Beide Batterien erneuern. (Abb. 3)

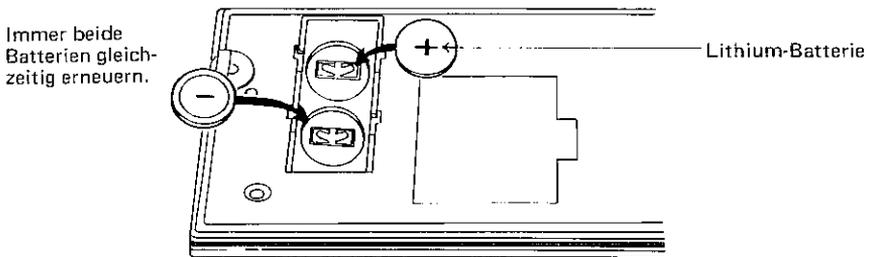


Abb. 3

- (5) Den Batteriefachdeckel entgegen der Pfeilrichtung in Abb. 2 wieder anbringen.
 (6) Die Klauen des Gehäusedeckels in die entsprechenden Aussparungen des Geräts einsetzen. (Abb. 4)

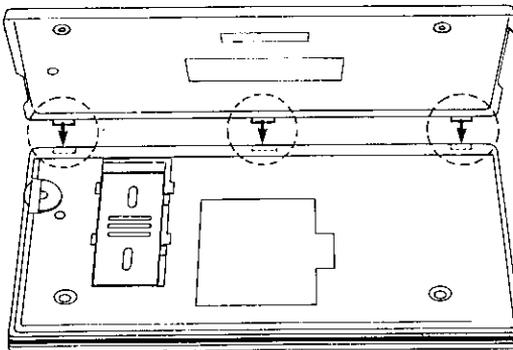


Abb. 4

Vorstellung

- (7) Die Schrauben wieder einschrauben, dabei den Gehäusedeckel leicht andrücken.
- (8) Zum Einschalten des Computers den Schiebeschalter auf ON stellen und den RESET-Knopf drücken, um den Computer zu initialisieren.
Das Display sollte wie folgt aussehen:



Wenn das Display leer bleibt oder eine andere Anzeige als "0." erscheint, die Batterien herausnehmen, erneut einsetzen und dann das Display noch einmal kontrollieren.

HINWEIS:

Leere Batterien, die im Computer verbleiben, können den Computer durch Auslaufen der Batteriesäure beschädigen. Daher leere Batterien umgehend erneuern!

ACHTUNG: Die Batterien außerhalb der Reichweite von Kindern aufbewahren.

KAPITEL 3 DER PC-1421 ALS TASCHENRECHNER

Nachdem Sie jetzt mit den Bestandteilen des **SHARP PC-1421** und deren Anordnung vertraut sind, wollen wir Sie mit den faszinierenden Fähigkeiten Ihres neuen Computers bekannt machen.

Da der **PC-1421** über alle gebräuchlichen mathematischen Funktionen verfügt und außerdem in **BASIC** programmierbar ist (für komplexere Rechenoperationen), wird er gemeinhin als "schlauer" Rechner bezeichnet. Damit macht er Sie natürlich zu einem "schlauem" Benutzer.

(Überprüfen Sie, daß die Batterien korrekt eingesetzt sind, bevor Sie mit dem **PC-1421** arbeiten.)

Einschalten

Zum Einschalten des **PC-1421** wird der Schiebeschalter nach oben geschoben.

Wenn über **RUN** oder **PRO** der Strich (—) leuchtet, durch Drücken der Taste **CAL** auf den **CAL**-Modus schalten.

Ausschalten

Zum Ausschalten des **PC-1421** den Schiebeschalter auf **OFF** stellen.

Durch Ausschalten des Geräts wird das Display gelöscht.

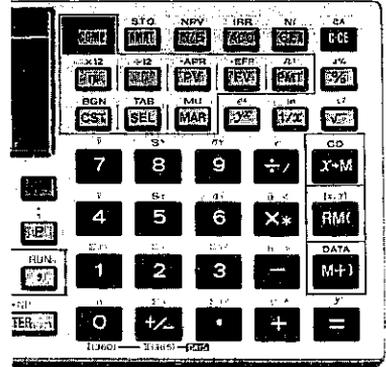
Automatische Abschaltung

Um Batterieleistung zu sparen, schaltet der **PC-1421** sich automatisch aus, wenn etwa 10 Minuten keine Taste betätigt wurde. (Hinweis: Während der Ausführung eines Programms schaltet sich der **PC-1421** nicht automatisch aus.)

Nach einer automatischen Abschaltung wird der Computer durch Drücken der Taste **ON BRK**, die sich rechts neben den grünen Tasten befindet, wieder eingeschaltet. Bitte beachten Sie, daß nach Drücken der Taste **ON BRK** der **CAL**-Modus gewählt sein kann.

Rechnen im CAL-Modus

Im CAL-Modus können die rechts abgebildeten Tasten und Funktionen für Berechnungen verwendet werden.



Wir wollen jetzt einige einfache Berechnungen durchführen. Schauen Sie auf das Display und drücken Sie die folgenden Tasten:

Eingabe

Σxy Σy^2
1 2 3

$x' \wedge$
+

σx Sx \bar{x}
6 5 4

y'
=

↑

Die Gleich-Taste
 wie bei einem normalen
 Taschenrechner verwenden.

Anzeige

123.

123.

654.

777.

(123 + 654 = 777)

* $\frac{CA}{C-CE}$ (Löschtaste) (rote Taste)

Wird diese Taste unmittelbar nach der Eingabe numerischer Daten oder nach Abrufen des Speicherinhalts gedrückt, werden die Daten gelöscht. In jedem anderen Fall werden durch Betätigung der Taste $\frac{CA}{C-CE}$ die eingegebenen Operatoren und/oder numerischen Daten gelöscht. Mit der Taste $\frac{CA}{C-CE}$ kann der Speicherinhalt nicht gelöscht werden.

Eingabe		Anzeige	Eingabe		Anzeige		
123	$+$	456	6	\times	2	$+$	
		456.					12.
	CA C-CE	123.		CA C-CE			0.
789	$=$	912.	6	\div	2	$+$	
		(123 + 789 = 912)					3.
			5	$=$			8.

Die Taste **CA**
C-CE kann auch zum Löschen von Fehlermeldungen benutzt werden.

Eingabe	Anzeige
5 \div 0 $=$	0. ^E ← Fehlermeldung
CA C-CE	0.

* **TAB** (Spezifizierung der Dezimalstellen)

Diese Taste dient zum Spezifizieren der Dezimalstellen, wenn sie zusammen mit den numerischen Datentasten verwendet wird.

	Eingabe	Anzeige
(1) Spezifizierung von 2 Dezimalstellen.	CA C-CE SHIFT TAB SEL 2	0.00
	5 \div 8 $=$	0.63
(2) Spezifizierung von 5 Dezimalstellen.	SHIFT TAB SEL 5	0.62500
(3) Löschen der TAB-Daten.	SHIFT TAB SEL .	0.625

Taschenrechner

+/- : Diese Taste dient zum Eingeben von negativen numerischen Daten (oder zum Umkehren des Vorzeichens von negativ auf positiv).

Eingabe

Anzeige

1.23 +/-

-1.23

+/-

1.23

Lesen des Displays

Dieser Abschnitt beschreibt die Anzeigeformate und Symbole, die im CAL-Modus verwendet werden.

1,234,567,898.

Festpunkt-Anzeigeformat
(normales format)

-1.234567898E-99

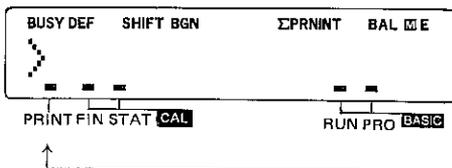
Gleitpunkt-Anzeigeformat

Mantisse (12 Stellen) Exponent (4 Stellen)

Das Display des **PC-1421** hat 16 Stellen. Im CAL-Modus werden die Ergebnisse normalerweise mit Festpunkt angezeigt. Ist das Ergebnis kleiner als 0.00000001 oder größer als 999999999 (größer als -0.00000001 oder kleiner als -999999999), wird es in Gleitpunkt-Darstellung angezeigt. Bei der Gleitpunkt-Darstellung wird die Mantisse mit 12 signifikanten Stellen und der Exponent mit 4 signifikanten Stellen angezeigt (einschließlich Dezimalpunkt, Vorzeichen und Symbol).

Anzeige-Symbole

Die folgenden Symbole und Zustand-Zeichen können auf dem Display erscheinen.



Siehe "Der Gebrauch des Druckers" in KAPITEL 8, DER GEBRAUCH DES CE-126P DRUCKER/ CASSETTENRECORDER-INTERFACE.

Die oben abgebildeten Symbole und Zeichen, die auf dem Display des PC-1421 erscheinen können, haben die folgenden Bedeutungen:

- SHIFT:** Dieses Wort leuchtet, wenn die Taste **SHIFT** betätigt wird, und bedeutet, daß die in braun über den Tasten angegebenen Funktionen gewählt werden können.
- BGN:** Dieses Wort leuchtet, wenn die Tasten **SHIFT** **BGN** **CST** gedrückt werden, und bedeutet, daß eine Beginn-Funktion gewählt wurde.
- ΣPRN:** Bedeutet, daß die angezeigte Zahl das angesammelte Kapital für die Periode darstellt.
- ΣINT:** Bedeutet, daß die angezeigte Zahl die aufgelaufenen Zinsen für die Periode darstellt.
- PRN:** Bedeutet, daß die angezeigte Zahl ein Kapital darstellt.
- INT:** Bedeutet, daß die angezeigte Zahl Zinsen darstellt.
- BAL:** Bedeutet, daß die angezeigte Zahl ein Restkapital (Bestand) ist.
- M :** Dieses Symbol leuchtet, wenn ein andere Zahl als Null im Rechen-
speicher vorhanden ist, und zeigt an, daß der Speicher belegt ist.
- E :** Dieses Symbol leuchtet, wenn ein Fehler aufgetreten ist. Der Fehler kann mit der Taste **C-CE** gelöscht werden.
- STAT:** Durch Betätigung der Taste **CAL** kann der Strich (-) über STAT links unten auf dem Display aktiviert werden. STAT steht für Statistik und bedeutet, daß der Computer im Statistik-Berechnungsmodus ist.
- FIN:** Durch Betätigung der Taste **CAL** kann der Strich (-) über FIN aktiviert werden. FIN bedeutet Finanz-Berechnungsmodus.

Grundrechenarten

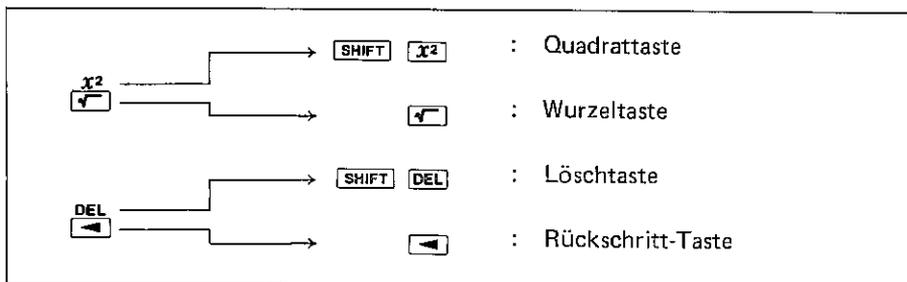
Dieser Abschnitt beschreibt die Grundrechenarten im CAL-Modus für den PC-1421. Vor den Berechnungen muß der Computer richtig eingestellt werden. Zuerst die Taste **CAL** drücken, um den Computer für normale Rechenoperationen einzustellen. Danach **CA** **C-CE** drücken und überprüfen, daß die folgende Anzeige auf dem Display erscheint.



Wenn nicht, die folgenden Punkte durchlesen und die erforderlichen Maßnahmen durchführen:

- ① Wenn mehr als eine Null angezeigt wird (z.B. 0.00):
Die Anzahl der Nachkommastellen (Dezimalstellen) wurde spezifiziert.
Durch Drücken von **SHIFT** **TAB** **◻** wird diese Spezifizierung gelöscht.
- ② Über STAT oder FIN wird ein Strich (-) angezeigt:
Der Computer befindet sich im Statistik- oder Finanz-Modus. Im normalen Rechenmodus wird dieser Strich nicht angezeigt.
- ③ Das Symbol **M** wird angezeigt:
Im Speicher sind bereits numerische Daten enthalten. Dieses Symbol kann durch Drücken von **CA** **C-CE** **X↔M** gelöscht werden.
- ④ Alle Symbole, die im oberen Bereich des Displays angezeigt werden, können mit der Taste **C-CE** gelöscht werden. Ausgenommen davon sind die über Punkt ④ beschriebenen Anzeigen.

In diesem Handbuch werden die Tastenfunktionen immer wie folgt dargestellt:



Zur Umstellung auf Gleitpunkt-Anzeige die Tasten **SHIFT** **TAB** **◻** drücken.

1. Addition, Subtraktion

Eingabe:

12 **+** 45.6 **-** 32.1 **+** 789 **-** 741 **+** 213 **=**

Ergebnis: 286.5

2. Multiplikation, Division

Eingabe: 841 **X** 586 **÷** .12 **=** Ergebnis: 4106883.333

Multiplikation mit einer Konstanten: Die zuerst eingegebene Zahl ist die Konstante.

Eingabe: 3 **X** 5 **=** Ergebnis: 15

Eingabe: 10 **=** Ergebnis: 30

Division mit einer Konstanten: Die nach dem Divisionszeichen eingegebene Zahl ist die Konstante.

Eingabe: 15 \div 3 $=$

Ergebnis: 5

Eingabe: 30 $=$

Ergebnis: 10

3. Speicher-Berechnungen

Der unabhängig zugängliche Speicher wird mit drei Tasten angesprochen: $\text{X}\rightarrow\text{M}$, RM , $\text{M}\rightarrow$. Vor der Berechnung den Speicher durch Drücken von CA und $\text{C}\rightarrow\text{CE}$ löschen.

Eingabe: 12 $+$ 5 $\text{M}\rightarrow$

Ergebnis: 17

→ Eingabe für Subtraktion: 2 $+$ 5 $=$ +/- $\text{M}\rightarrow$

Ergebnis dieser Gleichung: -7

Zum Abrufen des Speichers RM eingeben: 10

Eingabe: 12 \times 2 $=$ $\text{X}\rightarrow\text{M}$

Ergebnis: 24 (ersetzt die 10 im Speicher)

Eingabe: 8 \div 2 $\text{M}\rightarrow$

Ergebnis: 4 RM : 28

- Hinweise:
- Im STAT-Modus (statistische Berechnungen) können Speicher-Berechnungen nicht durchgeführt werden.
 - Zum Subtrahieren einer Zahl vom Speicherinhalt die Tasten +/- und $\text{M}\rightarrow$ drücken.

Höhere Rechenarten

1. Potenzen

Aufgabe: 20^2

Eingabe: 20 SHIFT x^2

Ergebnis: 400

Aufgabe: 3^3 und 3^4

Eingabe: 3 y^{x} 3 $=$

Ergebnis: 27

Eingabe: 3 y^{x} 4 $=$

Ergebnis: 81

2. Wurzeln

Aufgabe: $\sqrt{25}$

Eingabe: 25 $\sqrt{\quad}$

Ergebnis: 5

3. Kehrwerte

Aufgabe: $1/6 + 1/7$

Eingabe: 6 $\frac{1}{x}$ + 7 $\frac{1}{x}$ =
 Ergebnis: 0.309523809

4. Fakultäten

Aufgabe: 69!

Eingabe: 69 **SHIFT** **n!**

Ergebnis: 1.711224524E 98 ($1.711224524 \times 10^{98}$)

Beachten Sie die Rechengrenzen des Taschenrechners, die im Abschnitt über Fehler behandelt werden.

5. Prozentrechnung

Aufgabe: 45% von 2780 ($2780 \times \frac{45}{100}$)

Eingabe: 2780 **x** 45 **SHIFT** **%**

Ergebnis: 1251

Aufgabe: $\frac{547 - 473}{473} \times 100$

Eingabe: 547 **-** 473 **SHIFT** **%**

Ergebnis: 15.6448203

Dezimalstellen

Mit den Tasten **SHIFT** und **TAB** werden die Dezimalstellen (Nachkommastellen) des Rechenergebnisses spezifiziert. Die Anzahl Stellen nach dem Dezimalpunkt wird durch Drücken einer numerischen Taste (**0** ~ **9**) nach Drücken der Tasten **SHIFT** und **TAB** spezifiziert. Zum Löschen der zugewiesenen Dezimalstellen die Tasten **SHIFT** **TAB** und **.** drücken.

SHIFT **TAB** **0** → Spezifiziert 0 Dezimalstellen.
 (Die erste Dezimalstelle wird gerundet.)

SHIFT **TAB** **1** → Spezifiziert 1 Dezimalstelle.
 ? (Die zweite Dezimalstelle wird gerundet.)

SHIFT **TAB** **9** → Spezifiziert 9 Dezimalstellen.
 (Die 10. Dezimalstelle wird gerundet.)

SHIFT **TAB** **.** → Spezifiziert Gleitpunkt-Darstellung. (Die Dezimalstellen-Spezifizierung wird zurückgestellt.)
 (Die 11. Ziffer wird von der oberen Ziffer gerundet.)

Finanzielle Berechnungen

Die speziellen Business-Programme, über die der **PC-1421** verfügt, können nur im FIN-Modus verwendet werden. Vor den Berechnungen mit **[SHIFT]** **[CA]** löschen.

Finanzielle Berechnungen sind oft lang und kompliziert. Es kann gelegentlich bis zu einer Minute dauern, bis die Lösung angezeigt wird.

Zum Unterbrechen einer finanziellen Berechnung die Taste **[ON]** **[BRK]** (Unterbrechungs-Funktion) drücken.

Der **PC-1421** hat drei Grundtypen für finanzielle Berechnungen:

- A. Zinsen
- B. Amortisation
- C. Barwertanalyse

ZINSEN

A. Mit Zinsrechnung wird die Wertsteigerung von Kapital pro Zeit berechnet. Vier Tasten werden gebraucht:

- n = Anzahl der Laufzeiten
- i = Zinssatz pro Laufzeit
- PV = Gegenwärtiger Kapitalwert
- FV = Zukünftiger Kapitalwert

Die Grundgleichung ist: $FV = PV (1 + i)^n$

BEISPIEL:

Der Finanzdirektor des Unternehmens BNW, Ed Green, möchte \$1.000.000 für 1 Jahr investieren. Bank A bietet 9,3% jährliche Zinsen mit monatlicher Kapitalisierung und Bank B bietet 9,35% jährliche Zinsen mit vierteljährlicher Kapitalisierung. Welche Bank gibt nach einem Jahr den höchsten Betrag zurück?

Eingabe: **[SHIFT]** **[CA]** **[SHIFT]** **[TAB]** **2**

1000000 **[+/-]** **[PV]**

12 **[n]**

9.3 **[SHIFT]** **[÷12]** **[i]**

[COMP] **[FV]**

\$1.097.068,34 (Bank A)

[x↔M]

4 **[n]**

9.35 **[÷]** **4** **[=]** **[i]**

[COMP] **[FV]**

\$1.096.829,73 (Bank B)

[-] **[RM]** **[=]**

-\$238,61 (Bank B – Bank A)

Bank A würde \$238,61 mehr zurückzahlen als Bank B.

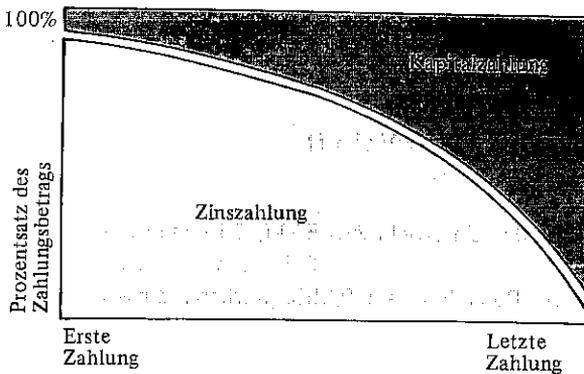
Eine andere Möglichkeit, zwei verschiedene Einlagen zu vergleichen, besteht im Vergleich der effektiven Jahreszinssätze.

Eingabe: **SHIFT CA SHIFT TAB 2**
12 SHIFT ←EFF 9.3 **≡ 9.71**
 (Effektive Jahresverzinsung von Bank A)
4 SHIFT ←EFF 9.35 **≡ 9.68**
 (Effektive Jahresverzinsung von Bank B)

(Ab jetzt wollen wir bei Wert-Berechnungen den gegenwärtigen Wert als negativen Betrag bezeichnen. Zahlungen an Banken oder Lieferanten werden als negative Beträge auf dem Display erscheinen. In vielen der folgenden Beispiele haben wir das negative Vorzeichen ausgelassen.)

AMORTISATION

- B. Amortisation bezieht sich auf die Art Abzahlungskredit, die als direkter Abzug bekannt ist. Die laufenden Zahlungen sind stets gleich. Die gezahlten Zinsen sind bei der ersten Zahlung der größte Anteil und bei der letzten Zahlung der kleinste Anteil. Das zurückgezahlte Kapital folgt dem umgekehrten Muster.



Zusätzlich zu den oben beschriebenen Tasten benutzen wir noch die folgenden:

PMT = Zahlung

AMRT = ergibt den Kapitalanteil und den Zinsanteil für eine bestimmte Zahlungsperiode – Restkapital

ACC (p_1/p_2) = ergibt die Summe von Kapital und Zinsen für eine bestimmte Zeitdauer.

Die Grundgleichung ist: $PV = PMT \frac{1 - (1+i)^{-n}}{i}$

BEISPIEL:

Das Unternehmen BNW plant, in diesem Jahr den Firmensitz nach Greenwich in Connecticut zu verlegen. Der Präsident des Unternehmens, John Brown, möchte in den Südwesten von Connecticut umziehen und dort ein Haus für \$140.000 kaufen. Er macht eine Anzahlung von \$70.000 und erhält eine Hypothek von \$70.000 bei einer jährlichen Verzinsung von 12,5% über 30 Jahre. Welchen Betrag muß er monatlich zahlen?

Eingabe: **SHIFT** **CA** **SHIFT** **TAB** **2**

70000 **PV**

12.5 **SHIFT** **÷12** **i**

30 **SHIFT** **x12** **n**

COMP **PMT**

Ergebnis: **-\$747,08/Monat**

Wie hoch würde die monatliche Belastung bei einer 25-jährigen Hypothek sein?

25 **SHIFT** **x12** **n**

COMP **PMT**

Ergebnis: **-\$763,25/Monat**

Für die 25-jährige Hypothek:

- a. wollen wir die Kapitalzahlung, Zinszahlung und das Restkapital nach der ersten Zahlung berechnen:

Eingabe:

1 **AMRT** Kapital (PRN) = **-\$34,08**

AMRT Zinsen (INT) = **-\$729,17**

AMRT Restkapital (BAL) = **\$69.965,92**

- b. nach der 150. Zahlung:

Eingabe:

150 **AMRT** Kapital (PRN) = **-\$159,62**

AMRT Zinsen (INT) = **-\$603,63**

AMRT Restkapital (BAL) = **\$57.788,56**

- c. Wie hoch wären die Summen der Kapitalzahlungen und Zinszahlungen im zweiten Laufjahr? (die 13. bis 24. Zahlungsperiode)

Eingabe:

13 **P₁/P₂** **24** **P₁/P₂**

ACC Kapital (Σ PRN) = **-\$490,61**

ACC Zinsen (Σ INT) = **-\$8.668,37**

d. Wie hoch wären die während der gesamten Laufzeit gezahlten Kapital- und Zinsbeträge?

Eingabe: 1 $\frac{P_1/P_2}{P_1/P_2}$ 300 $\frac{P_1/P_2}{P_1/P_2}$
 $\frac{ACC}{ACC}$ Kapital (ΣPRN) = $-\$70.000,00$
 $\frac{ACC}{ACC}$ Zinsen (ΣINT) = $-\$158.974,37$

HINWEIS: Bei Zinsberechnungen werden die Werte automatisch gespeichert und können wie folgt abgerufen werden:

Tasten	Bezeichnung
$\frac{SHIFT}{SHIFT}$ $\frac{RCL}{RCL}$ $\frac{i}{i}$	Zinssatz/Laufzeit
$\frac{SHIFT}{SHIFT}$ $\frac{RCL}{RCL}$ $\frac{\pi}{\pi}$	Anzahl der Laufzeiten
$\frac{SHIFT}{SHIFT}$ $\frac{RCL}{RCL}$ $\frac{PV}{PV}$	gegenwärtiger Wert
$\frac{SHIFT}{SHIFT}$ $\frac{RCL}{RCL}$ $\frac{FV}{FV}$	zukünftiger Wert
$\frac{SHIFT}{SHIFT}$ $\frac{RCL}{RCL}$ $\frac{PMT}{PMT}$	Zahlung

BARWERTRECHNUNG

C. Barwert-Analyse* dient zur Bewertung einer Investition, ob sie den Profiterwartungen entspricht. In solchen Bewertungen muß der Zeitwert von Geld berücksichtigt werden, d.h. auf den gegenwärtigen Wert umgerechnet werden. Die Netto-Gegenwartswert-Technik vergleicht alle gegenwärtigen Werte aller zukünftigen Kapitalflüsse mit der Anfangs-Investition. Der interne Gewinn oder Zinsertrag einer Investition kann auch berechnet werden. Die Tasten sind:

i = voraussichtlicher Ertrag

CF_i = zur Eingabe des Kapitalflusses für jede Periode

NPV = berechnet den Netto-Gegenwartswert

IRR = berechnet den internen Gewinn oder Zinsertrag einer Investition

* Eine ausgezeichnete Darstellung dieses Themas und anderer geschäftlicher Anwendungen findet sich in dem Buch "Financial Analysis and Business Decisions on the Pocket Calculator" von Jon M. Smith, John Wiley & Sons, New York, New York, 1976.

BEISPIEL:

Das Unternehmen BNW beratschlagt den Kauf eines Gebäudes für ein Lager für 4 Jahre. Das Gebäude wird \$100.000 kosten, und Mr. Green plant, es nach 4 Jahren für \$120.000 zu verkaufen. Die Gegenwerte für die Einsparungen bei der Miete für die ersten drei Jahre werden als \$7.000, \$8.000 und \$8.000 angesetzt. Entspricht diese Investition dem Ziel von 10% Investitionsgewinn pro Jahr?

Eingabe: **SHIFT** **CA** **SHIFT** **TAB** **2**

10 **i**

100000 **+/-** **CFi**

(Wenn die Analyse eine Investition enthält, muß der Anfangskapitalfluß als negative Zahl eingegeben werden. Wenn keine Investition vorhanden ist, sollte der Anfangskapitalfluß als 0 eingegeben werden.)

7000 **CFi**

2 **SHIFT** **Ni** **8000** **CFi**

120000 **CFi**

SHIFT **NPV** Ergebnis: **947,34** — Das Ergebnis für NPV ist positiv, daher ist der Gewinn größer als 10%.

SHIFT **IRR** Ergebnis: **10,29%** — Tatsächlicher Ertrag für dieses Beispiel.

Hinweise zur Barwert-Analyse: die maximal zulässige Anzahl Kapitalflüsse (ein Kapitalfluß kann, braucht aber nicht, Mehrfach-Kapitalflüsse unter Verwendung der Ni-Funktion enthalten) beträgt 20. Die Ni-Funktion erlaubt die Eingabe von bis zu 99 aufeinanderfolgenden, identischen Kapitalflüssen.

• Überprüfung einer Kapitalfluß-Eingabe

SHIFT **RCL** und unten zugewiesene Kapitalfluß-Nummer drücken.

Für Mehrfach-Kapitalflüsse mit der Ni-Funktion **SHIFT** **RCL** Kapitalfluß-Nummer **SHIFT** **Ni** drücken.

Kapitalfluß

CF₀
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19

Kapitalfluß-Nummer

0
1
2
3
4
5
6
7
8
9
.0
.1
.2
.3
.4
.5
.6
.7
.8
.9

● **Ändern einer Kapitalfluß-Eingabe**

Eingabe: Neuer Betrag **[SHIFT]** **[STO]** Kapitalfluß-Nummer.

● **Ändern von Mehrfach-Kapitalflüssen**

Eingabe: Anzahl der Mehrfach-Kapitalflüsse* **[SHIFT]** **[N/]** neuer Betrag **[SHIFT]** **[STO]** Kapitalfluß-Nummer.

* Beim Ändern des Betrags von Mehrfach-Kapitalflüssen muß die Originalnummer eingegeben werden.

BEISPIEL:

Modus: FIN

Eingabe:

[SHIFT] **[CA]** **[SHIFT]** **[TAB]** 2
10 **[i]**
100000 **[+/-]** **[CF_i]**
7500 **[CF_i]**
4 **[SHIFT]** **[N/]** 8000 **[CF_i]**

Anzeige:

0.00
10.00
-100.000.00 CF₀
7.500.00 CF₁
8.000.00 CF₂

3	SHIFT	Ni	9000	CFi	9.000.00	CF ₃
120000	CFi				120.000.00	CF ₄
SHIFT	NPV				-5.339.28	
SHIFT	IRR				9.11	

(1) Ändern des Kapitalfluß-Betrags (CFi)

Ändern Sie CFi von 7500 in 7000.

Eingabe: 7000 **SHIFT** **STO** 1 7.000.00 7000 in CF₁ speichern

(2) Ändern der Anzahl von Mehrfach-Kapitalflüssen (Ni)

Ändern Sie Ni von CF₂ von 4 in 2.Eingabe: 2 **SHIFT** **Ni** 8000 **SHIFT** **STO** 2 8.000.00 2 in N₂ speichern

(3) Ändern des Betrags von Mehrfach-Kapitalflüssen

Ändern Sie CF₃ von 9000 in 10000 (Ni bleibt gleich)Eingabe: 3 **SHIFT** **Ni** 10000 **SHIFT** **STO** 3 10.000.00
10000 in CF₃ speichern**SHIFT** **NPV** -751.21**SHIFT** **IRR** 9.85

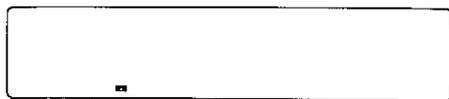
(4) Überprüfung der eingegebenen Daten

CF₁ abrufenEingabe: **SHIFT** **RCL** 1 7.000.00CF₂ abrufenEingabe: **SHIFT** **RCL** 2 8.000.00
SHIFT **RCL** **SHIFT** **Ni** 2.00**HINWEIS: BEGINN** **BGN**

Bei Zahlungen ist die Angabe erforderlich, ob die Zahlungen am Beginn oder Ende der Zahlungsperiode erfolgen. Die Zahlungsperiode-Taste des **PC-1421** ist als **SHIFT** **BGN** bezeichnet. Wenn **BGN** auf dem Display erscheint, werden die Zahlungen berechnet, als ob sie am Beginn der Periode erfolgten. Durch Drücken von **SHIFT** **BGN** wird **BGN** auf dem Display gelöscht, dann werden die Zahlungen berechnet, als erfolgten sie am Ende der Periode.

Statistische Berechnungen

Drücken Sie für statistische Berechnungen die Taste **CAL**, so daß ein Strich (=) über STAT auf dem Display erscheint. Das STAT steht für Statistik und zeigt an, daß der Computer im Modus für statistische Berechnungen ist.

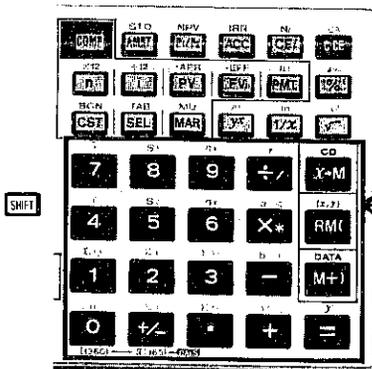


FIN STAT **CAL**



Die Taste **CAL** drücken, so daß hier der Strich angezeigt wird.

Tasten, die hauptsächlich für statistische Berechnungen verwendet werden.



Zwischenergebnisse können erhalten werden, und dann können zusätzliche Daten hinzugefügt werden.

Bei statistischen Berechnungen werden die folgenden statistischen Werte automatisch in den im BASIC-Modus gebrauchten festen Variablen (Speicherplätze) gespeichert. Diese statistischen Werte können im BASIC-Modus verarbeitet werden, weil sie auch dann erhalten bleiben, wenn der Statistik-Modus ausgeschaltet wird. Sie werden gelöscht, wenn der gegenwärtige Statistik-Modus ausgeschaltet und dann wieder eingeschaltet wird.

Speicher	Z	Y	X	W	V	U
Statistische Daten	n	Σx	Σx^2	Σxy	Σy	Σy^2

Zum Löschen der vorherigen statistischen Eingaben und Berechnungen den Statistik-Modus noch einmal ausschalten und wieder einschalten. Sonst werden bei der Durchführung von neuen statistischen Berechnungen falsche Ergebnisse erhalten.

Wenn der Statistik-Modus eingestellt ist, können die folgenden Berechnungen nicht durchgeführt werden:

- * Speicher-Berechnungen

ACHTUNG

Von den im CAL-Modus erhaltenen statistischen Daten werden die folgenden in die Speicherplätze des BASIC-Modus (U bis Z) gespeichert.

Speicher	Z	Y	X	W	V	U
Statistische Daten	n	Σx	Σx^2	Σxy	Σy	Σy^2

Für Berechnungen unter Verwendung dieser statistischen Daten den RUN-Modus benutzen.

Beispiel: Bedienung zur Berechnung der Quadratsummen (S^2) der vier Daten 205, 221, 226 und 220:

$$\begin{aligned}
 S^2 &= \Sigma (x - \bar{x})^2 \\
 &= \Sigma x^2 - n\bar{x}^2 \\
 &= \Sigma x^2 - \frac{1}{n} (\Sigma x)^2
 \end{aligned}$$

- Die Daten im STAT-Modus eingeben.

CAL **CAL** ...

STAT
0.

SHIFT **CA** 205 **DATA** 221 **DATA**

226 **DATA** 220 **DATA**

4.

- Zum RUN-Modus überwechseln und S^2 berechnen.

BASIC

>

X **-** **Y** **X*** **Y** **÷/** **Z**

X - Y * Y / Z _

ENTER

246.

1. Statistische Berechnungen mit einer Variablen

Wir wollen die folgenden statistischen Daten berechnen.

- (1) n : Stichprobenumfang
- (2) Σx : Summe der Stichprobendaten
- (3) Σx^2 : Summe der Quadrate der Stichprobendaten
- (4) \bar{x} : Mittelwert der Stichprobendaten $\bar{x} = \frac{\Sigma x}{n}$
- (5) S_x : Standardabweichung mit Gesamtheitsparameter als "n-1".

$$S_x = \sqrt{\frac{\Sigma x^2 - n\bar{x}^2}{n-1}}$$

(Zur Schätzung der Standardabweichung einer Gesamtheit aus den Stichprobendaten, die aus dieser Gesamtheit stammen.)

- (6) σ_x : Standardabweichung mit Gesamtheitsparameter als "n".

$$\sigma_x = \sqrt{\frac{\Sigma x^2 - n\bar{x}^2}{n}}$$

(Wird verwendet, wenn alle Gesamtheiten als Stichprobendaten dienen oder zum Finden der Standardabweichung einer Gesamtheit, wobei die Stichprobe als Gesamtheit angesehen wird.)

Die Daten für statistische Berechnungen mit einer Variablen werden auf die folgende Weise eingegeben

- (1) Datum
- (2) Datum Häufigkeit (wenn zwei oder mehr Daten gleich sind)

Beispiel:

Berechnen Sie Standardabweichung, Mittelwert und Varianz (S_x)² aus den folgenden Daten:

Vorher den Computer auf den Statistik-Modus stellen.

Wert	35	45	55	65
Häufigkeit	1	1	5	2

Bei der Eingabe der Stichprobendaten erscheinen jeweils die einzelnen Nummern der Daten rechts auf dem Display.

	<u>Eingabe:</u>	<u>Anzeige:</u>
35	<input type="text" value="DATA"/>	1.
45	<input type="text" value="DATA"/>	2.
55 x 5	<input type="text" value="DATA"/>	7.
65 x 2	<input type="text" value="DATA"/>	9.

SHIFT **TAB** **2** drücken (2 Dezimalstellen)

	<u>Eingabe:</u>	<u>Anzeige:</u>
Mittelwert:	SHIFT \bar{x}	53.89
Standardabweichung:	SHIFT Sx	9.28
Mittelwert:	SHIFT x^2	86.11

Datenkorrektur (CD): Die letzte Eingabe oben war falsch und muß in 60 x 2 korrigiert werden.

	<u>Eingabe:</u>	<u>Anzeige:</u>
65	X 2 CD	7.00
60	X 2 DATA	9.00

2. Statistische Berechnungen mit zwei Variablen und lineare Regression

Bei statistischen Berechnungen mit zwei Variablen werden die statistischen Funktionen für x und y und zusätzlich die Summe der Produkte der Stichprobenwerte $\sum xy$ berechnet. Statistische Berechnungen mit zwei Variablen ermöglichen, die beiden Datensätze in Beziehung (Korrelation) zu setzen. Jedes Datenpaar hat einen x - und y -Wert. Aus diesen Datenpaaren kann eine Regressionsgerade berechnet werden. Die Inbeziehungsetzung der beiden Datensätze mit Hilfe der Regressionsgeraden wird als "lineare Regression" bezeichnet. In der linearen Regression gibt es drei wichtige Werte: r , a und b .

Die Geradengleichung ist $y = a + bx$, wobei a der Schnittpunkt zwischen der Geraden und der Y -Achse und b die Steigung der Geraden sind.

Der Korrelationskoeffizient r repräsentiert die Beziehung zwischen den beiden Datensätzen. Bei $r = 1$ hat man eine perfekte Korrelation zwischen zwei Werten (-1 ist eine perfekte negative Korrelation), d.h. wenn man den Wert einer Variablen kennt, kann man den Wert der anderen Variablen mit 100% Sicherheit vorhersagen. Je mehr sich r von 1 unterscheidet, desto unzuverlässiger die Voraussagen. Die folgende Tabelle enthält Definitionen der Werte des Korrelationskoeffizienten:

	<u>Wert von r</u>	<u>Bezeichnung</u>	
Positive Korrelation	{	+0,80 to +1,00	sehr starke Korrelation
		+0,60 to +0,80	starke Korrelation
		+0,40 to +0,60	mäßige Korrelation
		+0,20 to +0,40	schwache Korrelation
		-0,20 to +0,20	keine Korrelation
Negative Korrelation	{	-0,20 to -0,40	schwache Korrelation
		-0,40 to -0,60	mäßige Korrelation
		-0,60 to -0,80	starke Korrelation
		-0,80 to -1,00	sehr starke Korrelation

r Korrelationskoeffizient

$$r = \frac{S_{xy}}{\sqrt{S_{xx} \cdot S_{yy}}}$$

a $a = \bar{y} - b\bar{x}$
 b $b = \frac{S_{xy}}{S_{xx}}$ } Koeffizient der linearen
 Korrelationsgleichung
 $y = a + bx$

$$\left[\begin{array}{l} S_{xx} = \Sigma x^2 - \frac{(\Sigma x)^2}{n} \\ S_{yy} = \Sigma y^2 - \frac{(\Sigma y)^2}{n} \\ S_{xy} = \Sigma xy - \frac{\Sigma x \cdot \Sigma y}{n} \end{array} \right]$$

Beispiel ① : Wenn die Punktzahl eines Schülers in Mathematik bekannt ist, kann man dann die Punktzahl in Englisch vorhersagen?

Die folgende Tabelle enthält die Punktzahlen von fünf zufällig ausgewählten Schülern:

Schüler-Nr. n	Punktzahl in Mathematik x	Punktzahl in Englisch y
1	82	79
2	53	50
3	61	87
4	74	96
5	51	73
6	51	73

Eingabe:	Anzeige:
82 $\boxed{(x,y)}$ 79 \boxed{DATA}	1.
53 $\boxed{(x,y)}$ 50 \boxed{DATA}	2.
61 $\boxed{(x,y)}$ 87 \boxed{DATA}	3.
74 $\boxed{(x,y)}$ 96 \boxed{DATA}	4.
51 $\boxed{(x,y)}$ 73 \boxed{X} 2 \boxed{DATA}	6. (Hinweis: gleiche Stichproben- daten werden wie dargestellt eingegeben)

Drücken: \boxed{SHIFT} \boxed{TAB} $\boxed{2}$

\boxed{SHIFT} \boxed{r} 0.57

\boxed{SHIFT} \boxed{a} 34.26 (y-Achse)

\boxed{SHIFT} \boxed{b} 0.68 (Steigung)

Der Wert $r = 0,57$ bedeutet eine mäßige Korrelation. Die Gleichung für die Regressionsgerade dieser Daten ist $y = 34,26 + 0,68x$.
 Aufgrund dieser Analyse würde ein Schüler mit einer Mathematik-Punktzahl von 90 eine Englisch-Punktzahl von 95 aufweisen.

Beispiel ② : Kann man anhand des Körpergewichts die Lebenserwartung von 65-jährigen Männern vorhersagen? Im Jahre 1950 wurden 10 Männer, alle mit der gleichen Körpergröße von 1,80 m, für das Experiment ausgewählt, ob das Körpergewicht die Lebenserwartung beeinflusst.

Stichprobe	1	2	3	4	5	6	7	8	9	10
Todesalter	72	67	69	85	91	68	77	74	70	82
Gewicht mit 65	185	226	200	169	170	195	175	174	198	172

Eingabe: STAT-Modus, **SHIFT** **CA** **SHIFT** **TAB** **□**

72 **(x,y)** 185 **DATA**

67 **(x,y)** 226 **DATA**

(auf diese Weise alle Daten eingeben)

SHIFT **f**

-0,792926167

Das Ergebnis ist eine relativ starke negative Korrelation. Höheres Gewicht deutet auf eine kürzere Lebenserwartung. Zum Zeichnen der Regressionsgeraden werden a und b verwendet.

SHIFT **a** 321,929125 (Y-Achse)

SHIFT **b** -1,795088908 (Steigung)

Vorhersage der Lebenserwartung eines 1,80-m-Mannes mit einem Gewicht von 190 Pfund im Jahre 1950.

190 **SHIFT** **x'** 73,4945283 Jahre

Wieviel sollte ein Mann 1960 wiegen, um 90 Jahre alt zu werden?

90 **SHIFT** **y'** 160,3712108 Pfund

Und wie schwer muß man sein, um 150 Jahre alt zu werden? Offensichtlich ist diese Frage falsch gestellt und weist darauf hin, daß man lineare Regressionen nicht zu weit extrapolieren darf.

BERECHNUNGSBEREICH

Eingabe und vier (4) arithmetische Berechnungen: 1., 2. Operand und Ergebnis:
 $\pm 1 \times 10^{-99} \sim \pm 9,999999999 \times 10^{99}$ und 0

Funktionen		Dynamischer Bereich
$\ln x$		$1 \times 10^{-99} \leq x \leq 9,999999999 \times 10^{99}$
e^x		$-9,999999999 \times 10^{99} \leq x \leq 230,2585092$
y^x		<ul style="list-style-type: none"> • $y > 0$: $-9,999999999 \times 10^{99} \leq x \log Y \leq 99,99999999$ • $y = 0$: $1 \times 10^{-99} \leq x \leq 9,999999999 \times 10^{99}$ • $y < 0$: x: ganze Zahl oder $\frac{1}{x}$: ungerade Zahl ($x \neq 0$) und $-9,999999999 \times 10^{99} \leq x \log Y \leq 99,99999999$
\sqrt{x}		$0 \leq x \leq 9,999999999 \times 10^{99}$
x^2		$ x \leq 9,999999999 \times 10^{49}$
$1/x$		$ x \leq 9,999999999 \times 10^{99}$ $x \neq 0$
$n!$		$0 \leq n \leq 69$ (n: ganze Zahl)
Statistische Berechnungen	Daten CD	$ x \leq 9,999999999 \times 10^{49}$ $ y \leq 9,999999999 \times 10^{49}$ $ \Sigma x \leq 9,999999999 \times 10^{99}$ $\Sigma x^2 \leq 9,999999999 \times 10^{99}$ $ \Sigma y \leq 9,999999999 \times 10^{99}$ $\Sigma y^2 \leq 9,999999999 \times 10^{99}$ $ \Sigma xy \leq 9,999999999 \times 10^{99}$ $ n \leq 9,999999999 \times 10^{99}$
	\bar{x}	$n \neq 0$
	S_x	$n \neq 1$ $0 \leq \frac{\Sigma x^2 - n\bar{x}^2}{n-1} \leq 9,999999999 \times 10^{99}$
	σ_x	$n \neq 0$ $0 \leq \frac{\Sigma x^2 - n\bar{x}^2}{n} \leq 9,999999999 \times 10^{99}$
	S_y	$n \neq 1$ $0 \leq \frac{\Sigma y^2 - n\bar{y}^2}{n-1} \leq 9,999999999 \times 10^{99}$
	σ_y	$n \neq 0$ $0 \leq \frac{\Sigma y^2 - n\bar{y}^2}{n} \leq 9,999999999 \times 10^{99}$

Funktionen		Dynamischer Bereich
Statistische Berechnungen	r	$n \neq 0$ $0 < (\sum x^2 - n\bar{x}^2) \cdot (\sum y^2 - n\bar{y}^2) \leq 9,999999999 \times 10^{99}$ $ \sum xy - \frac{\sum x \cdot \sum y}{n} \leq 9,999999999 \times 10^{99}$ $ \frac{\sum xy - \frac{\sum x \cdot \sum y}{n}}{\sqrt{(\sum x^2 - n\bar{x}^2) \cdot (\sum y^2 - n\bar{y}^2)}} \leq 9,999999999 \times 10^{99}$
	b	$n \neq 0$ $0 < \sum x^2 - n\bar{x}^2 \leq 9,999999999 \times 10^{99}$ $ \sum xy - \frac{\sum x \cdot \sum y}{n} \leq 9,999999999 \times 10^{99}$ $ \frac{\sum xy - \frac{\sum x \cdot \sum y}{n}}{\sum x^2 - n\bar{x}^2} \leq 9,999999999 \times 10^{99}$
	a	a ist die gleiche Bedingung wie b, und $ \bar{y} - b\bar{x} \leq 9,999999999 \times 10^{99}$
	y'	$ a + bx \leq 9,999999999 \times 10^{99}$
	x'	$ \frac{y - a}{b} \leq 9,999999999 \times 10^{99}$

Hinweis: Im obigen Berechnungsbereich werden die Rechenergebnisse oder Zwischenergebnisse als Null behandelt und angezeigt, wenn ihre Absolutwerte kleiner sind als 1×10^{-99} .

Manuelles Rechnen im BASIC-Modus

Was ist manuelles Rechnen

Der Computer PC-1421 kann grundsätzlich auf zwei verschiedene Weisen benutzt werden. Eine Weise ist, das gesamte Berechnungsverfahren mit allen Schritten als Programm im voraus im Computer zu speichern und später vom Computer automatisch ausführen zu lassen. Die andere Möglichkeit ist, die Aufgabe Schritt für Schritt durch manuelle Betätigung der Tasten zu lösen.

Die letztgenannte Rechenweise wird als "manuelles Rechnen" bezeichnet.

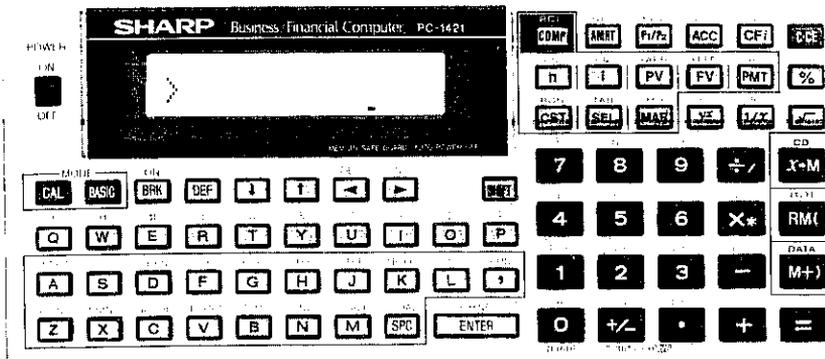
Natürlich werden im CAL-Modus alle Berechnungen manuell ausgeführt. Wir wollen jedoch nur solche Berechnungen als manuelles Rechnen bezeichnen, die im BASIC-Modus (RUN- oder PRO-Modus) manuell ausgeführt werden.

Wie manuell gerechnet wird

Wir wollen manuelles Rechnen im RUN-Modus ausprobieren. Drücken Sie die Taste **BASIC**, um den Computer in den RUN-Modus zu schalten.



Im RUN-Modus sind die in der folgenden Abbildung gezeigten Tasten und Funktionen verwendbar (das gleiche gilt für den PRO-Modus)



Taschenrechner

Bevor wir zu den Beispielen weitergehen, müssen wir noch einige wichtige Punkte behandeln.

Zum Rechnen mit Bleistift und Papier verwenden wir gewöhnlich die Operatoren (Rechenzeichen) $+$, $-$, \times und \div . Für arithmetische Operationen in BASIC benutzen wir \times und \div nicht, sondern $*$ und $/$.

Die Operatoren werden durch Drücken der Taste $\boxed{\times}$ bzw. $\boxed{\div}$ eingegeben. Drücken Sie die Taste $\boxed{\text{ENTER}}$ anstelle der Taste $\boxed{=}$, um das Ergebnis der manuellen Berechnung zu erhalten.

Benutzen Sie für Berechnungen mit dem PC-1421 nicht das Dollarzeichen und das Komma, da diese Zeichen in der Programmiersprache BASIC spezielle Bedeutungen haben. Versuchen Sie sich jetzt an den folgenden, einfachen Beispielen. Nicht vergessen, vor jeder neuen Berechnung mit $\boxed{\text{C}\text{C}\text{E}}$ zu löschen.

Eingabe

$\boxed{5}$ $\boxed{0}$ $\boxed{+}$ $\boxed{5}$ $\boxed{0}$ $\boxed{\text{ENTER}}$

$\boxed{1}$ $\boxed{0}$ $\boxed{0}$ $\boxed{-}$ $\boxed{5}$ $\boxed{0}$ $\boxed{\text{ENTER}}$

$\boxed{6}$ $\boxed{0}$ $\boxed{*}$ $\boxed{1}$ $\boxed{0}$ $\boxed{\text{ENTER}}$

$\boxed{3}$ $\boxed{0}$ $\boxed{0}$ $\boxed{/}$ $\boxed{5}$ $\boxed{\text{ENTER}}$

$\boxed{1}$ $\boxed{0}$ $\boxed{\text{SHIFT}}$ $\boxed{\wedge}$ $\boxed{2}$ $\boxed{\text{ENTER}}$

$\boxed{2}$ $\boxed{*}$ $\boxed{\text{P}}$ $\boxed{\text{I}}$ $\boxed{\text{ENTER}}$

$\boxed{\leftarrow}$ $\boxed{6}$ $\boxed{4}$ $\boxed{\text{ENTER}}$

Anzeige

100.

50.

600.

60.

100.

6.283185307

8.

Abrufen nach der Eingabe

Nachdem der PC-1421 das Rechenergebnis angezeigt hat, kann die letzte Eingabe wiederangezeigt werden. Verwenden Sie zum Abrufen den linken $\boxed{\leftarrow}$ und rechten $\boxed{\rightarrow}$ Pfeil.

Mit dem linken Pfeil $\boxed{\leftarrow}$ steht der Cursor hinter dem **letzten** Zeichen des abgerufenen Ausdrucks.

Mit dem rechten Pfeil $\boxed{\rightarrow}$ steht der Cursor "auf" dem **ersten** Zeichen des abgerufenen Ausdrucks.

Beachten Sie, daß der linke und rechte Pfeil außerdem zum Versetzen des Cursors innerhalb einer Zeile dienen. Diese bedien Tasten sind äußerst hilfreich zum Edietieren (oder Korrigieren) von Eingaben, weil dann nicht der ganze Ausdruck neu eingegeben werden muß.

Die folgenden Beispiele werden Sie mit dem linken und rechten Pfeil vertraut machen. In diesen Beispielen übernehmen Sie die Rolle eines Managers in einem Unternehmen und führen die jeweiligen Berechnungen durch.

Als Personalleiter einer großen Verkaufsabteilung sind Sie für die Planung des jährlichen Verkäufer-Treffens verantwortlich. 300 Teilnehmer werden zur dreitägigen Konferenz erwartet. Für einen Teil der Zeit werden die Verkäufer in kleinen Gruppen tagen. Sie denken, daß sechs Personen eine gute Gruppengröße darstellen. Wieviele Gruppen werden es insgesamt?

Eingabe

Anzeige

Nach nochmaligem Überdenken kommen Sie zur Ansicht, daß Gruppen mit ungeraden Teilnehmerzahlen vielleicht effektiver sind. Rufen Sie die letzte Eingabe mit ab.

Eingabe

Anzeige

Zur Berechnung der neuen Anzahl der Gruppen müssen Sie die vorher eingegebene sechs durch eine ungerade Zahl ersetzen. Fünf erscheint Ihnen sinnvoller als sieben. Da Sie die letzte Eingabe mit abgerufen haben, steht der Cursor an der rechten Seite der Anzeige. Bewegen Sie den Cursor mit um eine Stelle nach links.

Eingabe

Anzeige

Nach Versetzen des Cursors wird er zu einem blinkenden Rechteck . Immer wenn Sie den Cursor "auf" ein Zeichen setzen, blinkt er auf diese Weise.

Geben Sie 5 ein, um die 6 zu ersetzen. Aber beachten Sie immer: Sobald Sie vorhandene Zeichen überschreiben, ist das Original für immer verloren! Sie können Zeichen, die Sie überschrieben haben, nicht wieder abrufen.

Taschenrechner

Eingabe

5

ENTER

Anzeige

300/5_

60.

Sechzig Gruppen erscheinen Ihnen vernünftig, daher belassen Sie es bei der Gruppenstärke von fünf Personen.

Abrufen ist sehr praktisch, um die letzte Eingabe zu überprüfen, vor allem wenn das Ergebnis nicht sinnvoll ist. Nehmen wir an, Sie hätten die folgende Berechnung ausgeführt:

Eingabe

3 **0** **/** **5** **ENTER**

Anzeige

6.

Selbst ein überarbeiteter, müder Manager wie Sie wird bemerken, daß 6 als Anzahl Gruppen nicht stimmen kann, wenn es sich um einige Hundert Teilnehmer handelt! Rufen Sie die letzte Eingabe mit **▶** ab.

Eingabe

▶

Anzeige

30/5

Da Sie die letzte Eingabe mit **▶** abgerufen haben, befindet sich der blinkende Cursor jetzt auf dem ersten Zeichen der Anzeige. Zur Korrektur müssen Sie eine Null einfügen. Bewegen Sie den Cursor mit **▶**, bis er sich auf der Null befindet. Zum Einfügen von Zeichen mit der Taste **▶** müssen Sie den Cursor auf das Zeichen setzen, vor dem Sie ein anderes Zeichen einfügen wollen.

Eingabe

▶

Anzeige

30/5

Mit der Taste **INS** schaffen Sie eine Leerstelle, wo Sie das neue Zeichen einfügen können.

Eingabe

SHIFT **INS**

Anzeige

3 0 / 5

Durch Drücken der Taste **INS** bewegen sich alle Zeichen um eine Stelle nach rechts, und ein Zwischenraum mit Klammern entsteht. Der blinkende Cursor befindet sich jetzt auf dieser Leerstelle und bezeichnet damit die Stelle für die nächste Eingabe. Geben Sie die erforderliche Null ein. Nach der Korrektur sieht das Ergebnis besser aus.

Eingabe

0**ENTER**

Anzeige

300/5**60.**

Nehmen Sie jetzt mal an, Sie hätten des Guten zuviel getan:

Eingabe

3 0 0 / 5 ENTER

Anzeige

600.

Das Ergebnis erscheint viel zu groß. Wenn nur 300 Personen an der Konferenz teilnehmen, wie kommen Sie dann auf 600 "Kleingruppen"? Rufen Sie die Eingabe mit **▶** ab.

Eingabe

▶

Anzeige

3000/5

Der blinkende Cursor befindet sich auf dem ersten Zeichen der Anzeige. Zur Korrektur müssen Sie eine der Nullen löschen. Bewegen Sie den Cursor mit **▶** auf die erste Null (oder eine andere). Zum Löschen eines Zeichens müssen Sie den Cursor auf das zu löschende Zeichen setzen.

Eingabe

▶

Anzeige

3000/5

Mit der Taste **DEL** werden Sie die überflüssige Null los.

Eingabe

SHIFT DEL

Anzeige

300/5

Durch Drücken der Taste **DEL** bewegen sich alle Zeichen um eine Stelle nach links. Das Zeichen, auf dem der Cursor steht, wird gelöscht, und diese Stelle wird besetzt. Der blinkende Cursor bleibt auf der gleichen Stelle, die nächste Eingabe würde hier erscheinen. Da keine weiteren Korrekturen erforderlich sind, können Sie jetzt die Berechnung zu Ende führen.

Eingabe

ENTER

Anzeige

60.

(Hinweis: Wird die Taste **SPC** (Leertaste) gedrückt, wenn der Cursor auf einem Zeichen steht, wird das Zeichen gelöscht, stattdessen entsteht eine Leerstelle. **DEL** löscht das Zeichen, die entstehende Leerstelle wird besetzt.)

Fehlermeldungen

Abrufen der letzten Eingabe ist unerlässlich, wenn eine der gefürchteten Fehlermeldungen (ERROR) angezeigt wird. Stellen Sie sich vor, daß Sie versehentlich die folgende Eingabe gemacht haben:

Eingabe

3 0 0 / / 5 ENTER

Anzeige

ERROR 1

Natürlich sind Sie erst einmal überrascht, wenn diese Meldung erscheint. Mit ERROR 1 will der Computer sagen: "Ich verstehe nicht, was ich machen soll." Um das Problem herauszufinden, müssen Sie die Eingabe mit **◀** oder **▶** abrufen.

Eingabe

◀

Anzeige

300 / / 5

Wenn Sie die Taste **◀** oder **▶** betätigen, bezeichnet der blinkende Cursor die Stelle, die den Computer verwirrt. Und die Verwirrung des Computers ist verständlich, mit zwei Operatoren kann er nicht arbeiten. Korrigieren Sie diesen Fehler mit der Taste **DEL**.

Eingabe

SHIFT DEL ENTER

Anzeige

60.

Falls Sie nach der Anzeige ERROR 1 bemerken, daß Sie ein Zeichen ausgelassen haben, korrigieren Sie mit der Taste **[INS]**.

Solange Sie den PC-1421 als Taschenrechner benutzen, wird die Mehrzahl der auftretenden Fehlermeldungen ERROR 1 (Syntaxfehler) sein. Sämtliche Fehlermeldungen finden Sie im Anhang A.

Verkettung von Rechenoperationen

Mit dem PC-1421 können Sie das Ergebnis einer Berechnung als Bestandteil der nächsten Berechnung verwenden.

Zu Ihren Aufgaben bei der Konferenzplanung gehört auch, eine ausführliche Kostenvorschau auszuarbeiten und zur Genehmigung vorzulegen. Für jeden Teilnehmer sind \$150,00 vorgesehen. Daraus ergibt sich der Gesamtbetrag wie folgt:

Eingabe

Anzeige

[3] [0] [0] [*] [1] [5] [0] [ENTER] 45000.

Von diesem Betrag wollen Sie 15% für die Schlußfeier am letzten Tag mit der Preisverleihung aufwenden. Bei der Verkettung von Rechenoperationen ist es nicht erforderlich, das vorige Ergebnis erneut einzugeben. Sie dürfen es aber auch nicht löschen (d.h. Sie dürfen nicht die Taste **[C-CE]** drücken). Wieviel Geld haben Sie zum Feiern?

Eingabe

Anzeige

[*] [.] [1] [5] 45000.*.15_

Beachten Sie, daß bei der Eingabe der zweiten Berechnung (*.15) der Computer automatisch das Ergebnis der ersten Berechnung links auf dem Display anzeigt und in die neue Berechnung einschließt. Bei Verkettung von Rechenoperationen muß die Eingabe immer mit einem Operator beginnen. Und Sie müssen die Eingabe immer mit **[ENTER]** beenden:

HINWEIS: Bei der Berechnung können die Tasten **[%]** und **[SHIFT] [4%]** nicht verwendet werden. Die Taste **[%]** dient ausschließlich als Zeichen, und die Tasten **[SHIFT] [4%]** funktionieren nicht.

Beispiel: 45000 **[*]** 15 **[%]** → ERROR 1

Eingabe

Anzeige

[ENTER] 6750.

Taschenrechner

Berechnen Sie Ihr Etat weiter. Das Festessen im Hotel wird sich auf \$4000 belaufen:

Eingabe

- **4** **0** **0** **0**

ENTER

Anzeige

6750.-4000_

2750.

Die Saaldekoration wird \$1225 kosten:

Eingabe

- **1** **2** **2** **5** **ENTER**

Anzeige

1525.

Und schließlich müssen Sie für die Musikband und die Lautsprecheranlage \$2200 bereithalten:

Eingabe

- **2** **2** **0** **0** **ENTER**

Anzeige

-675.

Entweder müssen Sie Ihre Pläne zurückschrauben oder die verfügbaren Mittel anders aufteilen!

Negative Zahlen

Da Sie den Abschlußabend wirklich zu etwas Besonderem machen wollen, halten Sie an Ihrem Plan fest, auch wenn er teurer wird als vorgesehen. Sie möchten jetzt natürlich wissen, auf welchen Anteil am Gesamtetat sich dieser Abend belaufen wird. Ändern Sie zuerst das Vorzeichen des zusätzlich benötigten Betrags:

Eingabe

***** **-** **1**

ENTER

Anzeige

-675.*-1_

675.

Addieren Sie dazu den ursprünglich geplanten Betrag:

Eingabe

+ 6 7 5 0 ENTER

Anzeige

7425.

Division durch 45000 ergibt den Prozentsatz dieses neuen Betrags am Gesamtetat:

Eingabe

/ 4 5 0 0 0 ENTER

Anzeige

0.165

Sie entschließen sich, 16,5% des Gesamtetats für die Abschlußfeier einzusetzen.

Komplexere Rechenoperationen und Klammerung

Bei den oben beschriebenen Berechnungen hätten Sie mehrere davon zusammenfassen können. Beispielsweise könnten Sie die beiden folgenden in einer Zeile eingeben:

$$675 + 6750 / 45000$$

Zusammengefaßte Rechenoperationen müssen jedoch sehr sorgfältig eingegeben werden:

$675 + 6750 / 45000$ kann interpretiert werden als

$$\frac{675 + 6750}{45000} \quad \text{oder} \quad 675 + \frac{6750}{45000}$$

Der **PC-1421** verfügt über spezielle Regeln für Bewertung der Ausdrücke und Vorrang der Operatoren (siehe Seite 62). Verwenden Sie Klammern, um die Ausdrücke klarer zu machen und auch wirklich die gewünschte Berechnung durchzuführen:

$$(675 + 6750) / 45000 \quad \text{oder} \quad 675 + (6750 / 45000)$$

Führen Sie die beiden folgenden Beispiele durch, um den Unterschied zu erkennen, den verschiedene Klammerung verursachen kann:

Eingabe

(6 7 5 + 6
 7 5 0) / 4
 5 0 0 0 ENTER

Anzeige

0.165

6 7 5 + (6
 7 5 0 / 4 5 0 0
 0) ENTER

675.15

Der Gebrauch von Variablen in Rechenoperationen

Der **PC-1421** kann bis zu 26 feste **Variable** unter den alphabetischen Zeichen A bis Z speichern. Falls Sie mit Variablen nicht vertraut sind, finden Sie genaue Erläuterungen in Kapitel 4. Variable werden mit Zuweisungs-Statements zugeordnet:

A = 5
 B = -2

Sie können außerdem den Wert einer Variablen (rechts) einer anderen Variablen (links) zuweisen:

C = A + 3
 D = E

Variable können anstelle von Zahlen in jeder Rechenoperation eingesetzt werden.

Die Abschlusfeier ist jetzt fest geplant, aber es bleibt noch genug zu tun, damit die Konferenz reibungslos abläuft. Sie wollen auch den Rest des Etats nach Prozenten aufteilen. Zunächst müssen Sie aber feststellen, über wieviel Geld Sie noch verfügen. Weisen Sie den Restbetrag der Variablen R zu:

Eingabe

R = 45000 - 7425
 - 7425
 ENTER

Anzeige

R = 45000 - 7425 _

37575.

Durch Drücken von **ENTER** führt der **PC-1421** die Rechenoperation durch und zeigt den Wert von R an. Sie können den jeweiligen Wert jeder Variablen anzeigen, indem Sie das alphabetische Zeichen, unter dem die Variable gespeichert ist, eingeben:

Eingabe

R ENTER

Anzeige

37575.

Jetzt können Sie Rechenoperationen mit der neuen Variablen ausführen. Der Wert von R bleibt erhalten, bis ein neuer Wert zugewiesen wird.

Sie möchten 60% des verbleibenden Betrags für Unterkünfte aufwenden:

Eingabe

R * . 6 0

ENTER

Anzeige

R * . 6 0 _

22545.

Außerdem wollen Sie für 25% des verbleibenden Betrags Management-Trainingsseminare durchführen:

Eingabe

R * . 2 5 ENTER

Anzeige

9393.75

Die zugewiesenen Werte der Variablen bleiben erhalten, wenn das Gerät ausgeschaltet wird oder sich automatisch ausschaltet. Die Werte gehen nur in den folgenden Fällen verloren:

- * Wenn Sie der gleichen Variablen einen neuen Wert zuweisen.
- * Wenn Sie CLEAR **ENTER** eingeben (nicht die Löschtaste **C-CE**).
- * Wenn Sie den Computer mit dem ALL RESET Knopf initialisieren.
- * Wenn Sie die Batterien auswechseln.

Für die Zuweisung von Variablen bestehen bestimmte Einschränkungen, und durch bestimmte Programmierverfahren werden die Werte der Variablen geändert. Die Zuweisung wird in Kapitel 4 diskutiert. Über die Verwendung von Variablen in Programmen finden Sie alle erforderlichen Angaben in Kapitel 5.

Rechenoperationen mit mehreren Rechenschritten

Zusätzlich zur Kombination mehrerer Operatoren in einer Berechnung ermöglicht der PC-1421 auch, mehrere Rechenoperationen nacheinander auszuführen, ohne jeweils **ENTER** drücken zu müssen. Dabei müssen die einzelnen Rechenoperationen durch Kommata getrennt werden. Nur das Ergebnis der **letzten** Berechnung wird angezeigt. (Beachten Sie aber, daß die maximale Zeilenlänge, die der

Taschenrechner

Computer akzeptiert, einschließlich **ENTER** 80 Zeichen beträgt.)

Sie würden gerne wissen, welcher Betrag für die Unterkunft verfügbar wäre, wenn Sie bei Ihrem ursprünglichen Plan von 15% für die Abschlußfeier geblieben wären:

Eingabe

R **=** **.** **8** **5** ***** **4** **5**
0 **0** **0** **.** **R** *****
. **6** **0**

Anzeige

.85*45000,R*.60_

Der Computer führt zwar alle Rechenoperationen nacheinander aus, zeigt aber nur das letzte Ergebnis an:

Eingabe

ENTER

Anzeige

22950.

Geben Sie R ein, um den Wert von R in diesen Rechenoperationen herauszufinden:

Eingabe

R **ENTER**

Anzeige

38250.

Fehlermeldungen

Wenn eine manuelle Berechnung zu einem Fehler führt, erscheint auf dem Display eine Fehlermeldung:

ERROR 1

oder

ERROR 2

Der Fehlerzustand kann mit der Taste **C-CE** oder **◀** oder **▶** gelöscht werden. Wird die Taste **◀** oder **▶** zum Löschen des Fehlerzustands gedrückt, erscheint der Teil der Formel, wo der Fehler auftrat, auf dem Display (siehe auch die Beschreibung über Abrufen nach der Eingabe).

Wissenschaftliche Notation

Für den Umgang mit sehr großen und sehr kleinen Zahlen wird oft ein besonderes Format verwendet, das als Exponentialdarstellung oder **wissenschaftliche Notation** bezeichnet wird. In der wissenschaftlichen Notation werden Zahlen in zwei Teile aufgeteilt.

Der erste Teil besteht aus einer normalen Dezimalzahl zwischen 1 und 10. Der

Unter bestimmten Umständen, wenn Zahlen häufig verwendet werden, benutzt der **PC-1421** eine spezielle kompakte Form. In diesen Fällen gelten spezielle Grenzen für die Größe der Zahlen, gewöhnlich entweder 0 bis 65535 oder -32768 bis +32767. Diese Zahlen können in 16 Binär-Bits dargestellt werden. Die Umstände, unter denen diese Formate verwendet werden, sind in Kapitel 8 beschrieben.

Letztes-Ergebnis-Funktion

Bei der Verkettung von Rechenoperationen konnten Sie das Ergebnis der vorigen Berechnung nur als ersten Operanden der folgenden Berechnung verwenden. Schauen Sie sich das folgende Beispiel an.

Eingabe	Anzeige
3 <input type="button" value="+"/> 4 <input type="button" value="ENTER"/>	7.
<input type="button" value="*"/> 5	7.*5 _
<input type="button" value="ENTER"/>	35.

Drücken Sie und dann die Taste oder . Wenn Sie diese Tasten unmittelbar nach Beendigung des obigen Rechenbeispiels betätigen, wird "35." angezeigt. Diese Anzeige ist das Ergebnis Ihrer **letzten** Rechenoperation.

Der **PC-1421** kann sich an das letzte Ergebnis von manuellen Berechnungen "erinnern", es kann mit der Taste oder auf dem Display abgerufen werden.

Bei der oben beschriebenen Verkettung von Rechenoperationen konnten Sie das Ergebnis der vorigen Berechnung nur als ersten Operanden der nächsten Berechnung verwenden. Die Letztes-Ergebnis-Funktion gestattet es jedoch, das Ergebnis der vorigen Berechnung in jede beliebige Position der folgenden Berechnung einzusetzen.

Beispiel: Benutzen Sie das Ergebnis (6,25) der Operation $50/8$ und berechnen Sie damit $12 \times 5 / 6,25 + 24 \times 3 / 6,25 = :$

Eingabe	Anzeige
50 <input type="button" value="/"/> 8 <input type="button" value="ENTER"/>	6.25
	Letztes Ergebnis → ↑

12 \times 5 \div \uparrow

12*5/6.25_

Letztes Ergebnis wird
abgerufen \div 24 \times 3 \div \downarrow

/6.25+24*3/6.25_

Letztes Ergebnis wird
abgerufen

ENTER

21.12

C-CE \downarrow

21.21_

Das letzte Ergebnis wird durch das Ergebnis der gegenwärtigen Berechnung ersetzt, wenn eine manuelle Berechnung mit der Taste **ENTER** durchgeführt wird.

Wie aus diesem Beispiel klar wird, kann das letzte Ergebnis jederzeit und in jede Position abgerufen werden. Es wird jedoch immer wieder durch eines neues letztes Ergebnis der letzten Berechnung ersetzt.

Das letzte Ergebnis wird nicht durch **C-CE** oder durch Tastenbetätigung gelöscht.

- In den folgenden Fällen kann das letzte Ergebnis nicht abgerufen werden: wenn der Computer nicht im RUN-Modus ist, wenn die Programmausführung vorübergehend unterbrochen ist und wenn der Trace-Modus gewählt ist.

Länge der Formeln

Die Länge von Formeln, die Sie in den Computer eingeben können, ist begrenzt. Beim **PC-1421** sind es für eine Rechenformel bis zu 79 Tastenbetätigungen (ohne die Taste **ENTER**). Wenn Sie eine 80. Taste betätigen, blinkt der Cursor () auf diesem Zeichen und zeigt damit an, daß die 80. Tasteneingabe nicht gültig ist.

Manuelles USING

Die Anzahl der anzuzeigenden Stellen und die Exponentialdarstellung der Ergebnisse kann für manuelles Rechnen spezifiziert werden.

USING "####.#" **ENTER**

(5 Ziffern, einschließlich dem Vorzeichen, für den ganzzahligen Teil und 2 Ziffern für den Nachkommateil werden spezifiziert.)

50 * -60 **ENTER** \longrightarrow -3000.00

Die Einstellung mit dem USING-Kommando bleibt gültig, bis eine neue Einstellung vorgenommen wird. (Die Einstellung wird jedoch gelöscht, wenn **SHIFT** **CA** oder USING **ENTER** eingegeben wird oder wenn der Computer ausgeschaltet wird.

• **3-stellige Abgrenzung (USING-Kommando)**

Das USING-Kommando kann spezifizieren, daß das Ergebnis von manuellen Berechnungen oder von ausgeführten Programmen alle 3 Ziffern auf dem Display durch Kommata abgegrenzt wird.

Manuelle Operation

USING " ,#####.##" **ENTER**

40 * 60 **ENTER** → 2.400.00

Wissenschaftliche Berechnungen im BASIC-Modus

Der **PC-1421** verfügt über viele wissenschaftliche Funktionen, die im BASIC-Modus eingesetzt werden können.

Zur Ausführung von wissenschaftlichen Funktionen müssen Sie am Ende der Eingabe **ENTER** drücken, sonst wird die Funktion nicht ausgeführt.

Diese Funktionen sind die folgenden:

Funktion	Definition im PC-1421	Operation	Bemerkungen
Trigonometrische Funktionen	sin	SIN	S I N
	cos	COS	C O S
	tan	TAN	T A N
Inverse trigonometrische Funktionen	\sin^{-1}	ASN	A S N
	\cos^{-1}	ACS	A C S
	\tan^{-1}	ATN	A T N
Hyperbolische Funktionen	sinh	HSN	H S N
	cosh	HCS	H C S
	tanh	HTN	H T N

Funktion	Definition im PC-1421	Operation	Bemerkungen
Inverse hyperbolische Funktionen			
\sinh^{-1}	AHS	A H S	
\cosh^{-1}	AHC	A H C	
\tanh^{-1}	AHT	A H T	
Logarithmische Funktionen			
\ln	LN	L N	$\log_e x$
\log	LOG	L O G	$\log_{10} x$
Exponentialfunktionen			
e^x	EXP	SHIFT e^x oder E X P	$e \approx$ 2,718281828
10^x	TEN	T E N	TEN (x)
Kehrwert			
$\frac{1}{x}$	RCP	1/x	
Quadrat			
x^2	SQU	SHIFT x²	
Quadratwurzel			
$\sqrt{\quad}$	$\sqrt{\quad}$ oder SQR	√	
Kubikwurzel			
$\sqrt[3]{\quad}$	CUR	C U R	CUR (x)
Fakultät			
n!	FACT	SHIFT n!	
Pi			
	PI	P I	$\pi \approx$ 3,141592654
DMS → DEG	DEG	D E G	
DEG → DMS	DMS	D M S	
Potenzen			
y^x	^	SHIFT ^ oder y^x	y^x
x-te Wurzel			
$\sqrt[x]{y}$	ROT	R O T	$y \text{ ROT } x : \sqrt[x]{y}$
Kartesische Koordinaten → Polarkoordinaten			
	POL	P O L	
Polarkoordinaten → Kartesische Koordinaten			
	REC	R E C	

Funktion	Definition im PC-1421	Operation	Bemerkungen
Ganze Zahlen	INT	I N T	INT (x)
Absolutwert x	ABS	A B S	ABS (x)
Signum	SGN	S G N	SGN (x) $x > 0 : 1$ $x = 0 : 0$ $x < 0 : -1$

Die meisten dieser Funktionen werden mit den alphabetischen Tasten eingegeben. Für Kehrwert (RCP), Quadrat (SQU), Quadratwurzel (SQR), Fakultät (FACT) und Potenzierung (^) sind besondere Funktionstasten vorhanden. Beispielsweise können Sie die Quadratwurzel aus 30 ($\sqrt{30}$) durch die Eingabe von $\sqrt{\quad}$ 30 berechnen. Für trigonometrische und inverse trigonometrische Funktionen und für Koordinatenumwandlung muß das gewünschte Winkelmaß vorher spezifiziert werden. Für manuelle Berechnungen kann das Winkelmaß entweder durch **SHIFT** **DRG** oder mit den folgenden Anweisungen spezifiziert werden:

Winkelmaß	Kommando	Beschreibung
Altgrad	DEGREE	Rechter Winkel wird als 90 [°] dargestellt.
Bogenmaß	RADIAN	Rechter Winkel wird als $\pi/2$ [rad] dargestellt.
Neugrad	GRAD	Rechter Winkel wird als 100 [Grad] dargestellt.

Diese Anweisungen dienen zum Spezifizieren des Winkelmaßes in Programmen. Verwenden Sie diese Anweisungen zur Übung in den folgenden Rechenbeispielen:

(Beispiel) $\sin 30^\circ =$

(Operation) DEGREE **ENTER** (Spezifiziert Altgrad als Winkelmaß.)

SIN 30 **ENTER**

0.5

(Beispiel) $\tan \frac{\pi}{4} =$

(Operation) RADIAN **ENTER** (Spezifiziert Bogenmaß als Winkelmaß.)

TAN (PI/4) **ENTER**

1.

(Beispiel) $\cos^{-1}(-0.5) =$ (Operation) DEGREE (Spezifiziert Altgrad als Winkelmaß.)ACS - 0.5

(120°)

(Beispiel) $\log 5 + \ln 5 =$ (Operation) LOG 5 + LN 5 (Beispiel) $e^{2+3} =$ (Operation) EXP (2 + 3) (Beispiel) $\sqrt[3]{4^3 + 5^3} =$ (Operation) CUR (4 ^ 3 + 5 ^ 3)

(Beispiel) Umwandlung von 30 Grad 30 Minuten in Sexagesimal-Darstellung in Dezimal-Darstellung.

(Operation) DEG 30. 30

(30,5 Grad)

(Beispiel) Umwandlung von 30,755 Grad in Dezimal-Darstellung in Sexagesimal-Darstellung.

(Operation) DMS 30.755

(30 Grad 45 Minuten 18 Sekunden)

(Beispiel) Umwandlung von orthogonalen Koordinaten in Polarkoordinaten: Bestimmung der Polarkoordinate (r, θ) für Punkt $(3, 8)$ auf einer orthogonalen Koordinate:(Operation) DEGREE (Spezifiziert Altgrad als Winkelmaß.)POL (3, 8) (r)

(r ≈ 8,5)

Z **ENTER** (θ) 69.44395478
 ($\theta \doteq 69^\circ$)

* Der Wert von θ wird in die Variable Z und der Wert von r in die Variable Y übertragen.

(Beispiel) Umwandlung von Polarkoordinaten in kartesische Koordinaten:
 Bestimmung der kartesischen Koordinate (x, y) für Punkt P $(12, \frac{4}{5} \pi)$
 auf einer Polarkoordinate:

(Operation) RADIAN **ENTER** (Spezifiziert Bogenmaß als Winkelmaß.)
 REC (12, (4/5 * PI))

ENTER (x) -9.708203933
 ($x \doteq -9,7$)

Z **ENTER** (Y) 7.053423028
 ($y \doteq 7,1$)

* Die Werte von y und x werden in die Variablen Z bzw. Y übertragen.

Hinweis: Bei Koordinatenumwandlung wird das Ergebnis in den Variablen Z und Y gespeichert. Dabei werden die vorherigen Inhalte von Z und Y überschrieben.

— Referenz —

Ausdrücke, die aus logischen Operatoren ($=$, $>$, $<$, \geq , \leq , $\langle \rangle$) bestehen, nehmen die in der Tabelle unten angegebenen Werte an:

x und y sind numerische Werte.

$=^*$	1 wenn $x = y$ \emptyset wenn $x \neq y$	\geq	1 wenn $x \geq y$ \emptyset wenn $x < y$
$>$	1 wenn $x > y$ \emptyset wenn $x \leq y$	\leq	1 wenn $x \leq y$ \emptyset wenn $x > y$
$<$	1 wenn $x < y$ \emptyset wenn $x \geq y$	$\langle \rangle$	1 wenn $x \neq y$ \emptyset wenn $x = y$ (“ $\langle \rangle$ “ bedeutet “ \neq “.)

* Wenn beispielsweise “A = numerischer Wert“ oder “B = Formel“ in einem logischen Ausdruck verwendet wird, behandelt der Computer es nicht als logischen Ausdruck, sondern als Zuweisungs-Statement für Variable. Wenn Sie das Gleichheitszeichen ($=$) für einen logischen Ausdruck verwenden, verwenden Sie es in der Form “numerischer Ausdruck = A“ oder “Formel = B“, mit der Ausnahme von Bedingungsdrücken in IF-Statements.

Direkte Berechnung

In den bisher beschriebenen manuellen Berechnungen haben wir immer die Taste **ENTER** verwendet, um die Eingabe zu beenden und das Ergebnis zu erhalten. Es ist jedoch möglich, die Funktionen des **PC-1421** direkt anzusprechen mit der entsprechenden Funktionstaste (ohne Betätigung der Taste **ENTER**), wenn das Argument auf dem Display angezeigt wird.

(Beispiel) Bestimmung von $8!$

(Operation) **C-CE** 8

8 _

SHIFT **7!**

40320.

Diese "direkte" Berechnung ist jedoch nicht möglich für Funktionen, die mehr als ein Argument erfordern (binomische Funktionen) wie Potenzen, x-te Wurzel und Koordinatenumwandlung.

- Nach direkten Berechnungen ist die Abruf-Funktion nicht wirksam. Durch Betätigung der Taste **◀** oder **▶** wird nur der Cursor angezeigt.

Priorität bei manuellen Berechnungen

Im BASIC-Modus können Sie Formeln in genau der gleichen Reihenfolge eingeben, in der sie normalerweise geschrieben werden, einschließlich Klammern und Funktionen. Die Prioritätsfolge bei der Berechnung und Behandlung der Zwischenergebnisse wird vom Computer durchgeführt.

Die interne Prioritätsfolge bei manuellen Berechnungen ist wie folgt:

- 1) Abrufen von Variablen oder π
- 2) Funktionen (sin, cos, usw.)
- 3) Potenzen (\wedge) und x-te Wurzeln (ROT)
- 4) Vorzeichen (+, -)
- 5) Multiplikation und Division (*, /)
- 6) Addition und Subtraktion (+, -)
- 7) Größenvergleich (>, >=, <, <=, <>)
- 8) Logisches AND, OR

- Hinweise:
- * Wird in einer Formel Klammerung verwendet, hat die Operation in Klammern die höhere Priorität.
 - * Zusammengesetzte Funktionen werden von rechts nach links abgearbeitet ($\sin \cos^{-1} 0.6$).
 - * Mehrfach-Potenzen (3^{4^2} oder $3 \wedge 4 \wedge 2$) und x-te Wurzeln werden von rechts nach links abgearbeitet.
 - * Bei den obigen Punkten 3) und 4) hat die letzte Eingabe die höhere Priorität. (Beispiel) $-2 \wedge 4 \rightarrow -(2^4)$
 $3 \wedge -2 \rightarrow 3^{-2}$

KAPITEL 4

BASIC-BEGRIFFE UND -AUSDRÜCKE

In diesem Kapitel werden wir einige Begriffe und Ausdrücke der Programmiersprache BASIC näher betrachten.

Stringkonstanten

Außer Zahlen kann der **SHARP PC-1421** auch Buchstaben und Symbole in vielfacher Weise verarbeiten. Buchstaben, Zahlen und Symbole werden allgemein Zeichen genannt. Der **PC-1421** verfügt über die folgenden Zeichen:

```

1 2 3 4 5 6 7 8 9 0
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
! " # $ % & ( ) * + , - . / : ; < = > ? @ √ π ^

```

In BASIC wird eine Gruppe von Zeichen als **String** bezeichnet. Damit der **PC-1421** in der Lage ist, zwischen Strings und anderen Programmteilen wie Befehlen und Variablennamen zu unterscheiden, müssen Strings in Anführungsstriche (") eingeschlossen werden.

Es folgen einige Beispiele für Stringkonstanten:

```

"HALLO"
"LEBEWOHL"
"SHARP PC-1421"

```

Die folgenden Beispiele werden **nicht** als Stringkonstanten akzeptiert:

```

"COMPUTER      Anführungszeichen am Ende fehlen
"ISN"T"        Innerhalb eines Strings dürfen Anführungszeichen nicht
                benutzt werden.

```

Variable

Computer bestehen aus vielen kleinen Speichereinheiten, die **Bytes** genannt werden. Man kann sich jedes Byte als einzelnes Zeichen vorstellen. Daher würde das Wort "Byte" im Speicher vier Byte belegen, weil es aus vier Zeichen besteht. Durch die Eingabe MEM **[ENTER]** können Sie erfahren, wieviel Byte Speicherplatz verfügbar sind. Die angezeigte Zahl bezeichnet die Anzahl Bytes, die zum Schreiben von Programmen verfügbar sind. Diese Methode arbeitet für Buchstaben ausgezeichnet, ist aber zum Speichern von Zahlen völlig unzureichend. Aus diesem Grund werden Zahlen in codierter Form gespeichert. Der Computer kann dank dieser Codierung auch große Zahlen in nur acht Bytes speichern. Die größte Zahl, die gespeichert werden kann, ist +9.999999999 E+99.

Die kleinste speicherbare Zahl ist +1.E-99. Der Bereich zwischen diesen beiden Zahlen wird für die meisten Anwendungen ausreichen. Sollte jedoch ein Rechenergebnis außerhalb dieses Bereichs liegen, macht der Computer Sie darauf durch Einschalten des Fehler-Indikators und durch Anzeige der Fehlermeldung auf dem Display aufmerksam. Der Fehler-Indikator ist ein kleines E oben rechts auf dem Display. Die Fehlermeldungen finden Sie im Anhang A. Erzeugen Sie jetzt gleich zum Kennenlernen eine Fehlermeldung:

9 **[EXP]** 99 * 9 **[ENTER]**

Drücken Sie die Taste **[C-CE]**, damit der Computer wieder normal arbeitet. Aber wie speichert man jetzt Informationen, werden Sie fragen. Es ist nicht so schwer. Computer arbeiten derart, daß Daten und Datengruppen Namen erhalten. Wir wollen die Zahl 556 im Computer speichern. Sie können dieser Zahl jeden beliebigen Namen geben, zum Beispiel R. Mit dem Statement LET kann man einem Variablennamen einen Wert zuweisen, aber nur in Programmen. Das Kommando LET braucht aber nicht angegeben werden, und wir werden es auch nicht oft benutzen. Machen Sie einfach die Eingabe R = 556 drücken dann **[ENTER]**. Der Computer stellt dadurch eine Beziehung zwischen dem Wert 556 und dem Buchstaben R her. Die Buchstaben, in denen Informationen gespeichert werden, werden **Variable** genannt. Um den Inhalt der Variablen R abzurufen, drücken Sie die drei Tasten **[C-CE]**, R und **[ENTER]**. Daraufhin zeigt der Computer rechts auf dem Display den Wert 556 an. Diese Möglichkeit wird sehr nützlich, wenn man Programme und Formeln eingibt.

Als nächstes wollen wir die Variable R in einer einfachen Formel einsetzen. In dieser Formel repräsentiert die Variable R den Radius eines Kreises, dessen Fläche wir berechnen wollen. Die Formel für die Kreisfläche ist: $A = \pi * R^2$. Machen Sie die folgende Eingabe R **[SHIFT]** **[^]** 2 **[*]** **[P]** **[I]** **[ENTER]** Das Ergebnis ist 971179.3865. Die Technik, Variable in Formeln zu verwenden, wird verständlicher, sobald wir die ersten Programme schreiben.

Bislang haben wir nur numerische Variable diskutiert. Wie aber werden alphabetische Zeichen gespeichert? Das Konzept ist im Grunde gleich. Damit der Computer den Unterschied zwischen den beiden Variablentypen erkennt, wird dem Variablennamen ein \$ hinzugefügt. Speichern Sie als Beispiel das Wort BYTE in der Variablen B\$.

Dabei sagt das \$ hinter dem B dem Computer, daß der Inhalt der Variablen B alphanumerisch bzw. ein Daten-String ist.

Damit dies klarer wird, machen Sie die folgende Eingabe: B $\boxed{\text{SHIFT}}$ $\boxed{\text{R}}$ = $\boxed{\text{SHIFT}}$ $\boxed{\text{W}}$ BYTE $\boxed{\text{SHIFT}}$ $\boxed{\text{W}}$ $\boxed{\text{ENTER}}$. Jetzt ist der Wert BYTE in der Variablen B\$ gespeichert. Geben Sie B $\boxed{\text{SHIFT}}$ $\boxed{\text{R}}$ $\boxed{\text{ENTER}}$ ein, um dies zu überprüfen. Auf dem Display wird BYTE angezeigt. Aber diesmal ist die Anzeige auf der linken Seite des Displays, nicht auf der rechten.

Die Variablen, mit denen der SHARP PC-1421 arbeitet, sind wie folgt aufgegliedert:

Variable	{	Numerische Variable	{	Vorgegebene numerische Variable (A bis Z)
				Einfache numerische Variable (AB, C1 usw.)
	}	Numerische Feld-Variable		
		String-Variable	{	Vorgegebene String-Variable (A\$ bis Z\$)
				Einfache String-Variable (BB\$, C2\$ usw.)
				String-Feld-Variable

Vorgegebene Variable

Der erste Bereich, die vorgegebenen Variablen, werden vom Computer grundsätzlich zum Speichern von Daten benutzt. Man kann ihn sich als reservierten Variablen-Bereich denken. Das bedeutet: egal wieviel Speicherplatz Ihr Programm belegt, Sie haben jederzeit mindestens 26 Variable zum Speichern von Daten zur Verfügung. Diese Daten können zwei Typen sein: numerische Daten oder String-Daten (Buchstaben). Die vorgegebenen Speicherplätze haben eine Länge von acht Byte und können jeweils nur einen Datentyp aufnehmen. Dies wird durch folgendes Beispiel klar:

```
A = 123  $\boxed{\text{ENTER}}$ 
A$  $\boxed{\text{ENTER}}$ 
```

Sie erhalten die Fehlermeldung:

ERROR 9

Diese Meldung bedeutet, daß Sie im Speicherplatz mit dem Namen A numerische Daten gespeichert und danach als String-Daten abzurufen versucht haben. Das aber verwirrt den Computer, so daß er eine Fehlermeldung ausgibt. Drücken Sie die Taste $\boxed{\text{CC}}$, um den Fehlerzustand aufzuheben. Versuchen Sie jetzt das folgende

Beispiel:

A\$ = "ABC" **ENTER**

A **ENTER**

Der Computer ist wieder verwirrt und meldet ERROR 9. Aus der folgenden Tabelle können Sie ersehen, daß die Variable A den gleichen Speicherbereich wie die Variable A\$ belegt, B den gleichen Speicherbereich wie B\$, und so fort für alle Buchstaben des Alphabets.

Gleiche Speicherbereiche:

A = A\$ = A(1) = A\$(1)

B = B\$ = A(2) = A\$(2)

C = C\$ = A(3) = A\$(3)

D = D\$ = A(4) = A\$(4)

E = E\$ = A(5) = A\$(5)

F = F\$ = A(6) = A\$(6)

G = G\$ = A(7) = A\$(7)

H = H\$ = A(8) = A\$(8)

I = I\$ = A(9) = A\$(9)

J = J\$ = A(10) = A\$(10)

K = K\$ = A(11) = A\$(11)

L = L\$ = A(12) = A\$(12)

M = M\$ = A(13) = A\$(13)

N = N\$ = A(14) = A\$(14)

O = O\$ = A(15) = A\$(15)

P = P\$ = A(16) = A\$(16)

Q = Q\$ = A(17) = A\$(17)

R = R\$ = A(18) = A\$(18)

S = S\$ = A(19) = A\$(19)

T = T\$ = A(20) = A\$(20)

U = U\$ = A(21) = A\$(21)

V = V\$ = A(22) = A\$(22)

W = W\$ = A(23) = A\$(23)

X = X\$ = A(24) = A\$(24)

Y = Y\$ = A(25) = A\$(25)

Z = Z\$ = A(26) = A\$(26)

Einfache Variable

Einfache Variable bestehen aus zwei (oder mehr) alphanumerischen Zeichen, beispielsweise AA oder B1. Anders als die festen Variablen haben einfache Variable keinen bestimmten im Speicher reservierten Bereich. Der Speicherbereich für

einfache Variable wird automatisch im Programm- oder Datenbereich bereitgestellt, sobald eine einfache Variable erstmalig benutzt wird.

Da für einfache numerische Variable und einfache String-Variable separate Speicherbereiche definiert sind, können auch Variable mit dem gleichen Namen (wie AB und AB\$) gleichzeitig verwendet werden.

Für Namen einfacher Variablen können zwar alphanumerische Zeichen verwendet werden, das erste Zeichen muß jedoch immer ein alphabetisches Zeichen sein. Werden mehr als zwei Zeichen zur Definition einer Variablen verwendet, sind nur die beiden ersten Zeichen für die Unterscheidung der Variablen von Bedeutung.

- Hinweise:
- Die Namen der Funktionen und BASIC-Befehle dürfen beim **PC-1421** nicht für Variablen-Namen verwendet werden (z.B. PI, IF, TO, ON, SIN usw.).
 - Unter einer einfachen String-Variablen können bis zu 16 Zeichen oder Symbole gespeichert werden.

Feldvariable

Für bestimmte Zwecke ist es nützlich, Zahlen in der Form von organisierten Gruppen zu verwenden, beispielsweise eine Liste der Fußballergebnisse oder eine Steuertabelle. In BASIC werden derartige Gruppen als **Felder** bezeichnet. Ein Feld kann entweder **eindimensional** wie eine Liste oder **zweidimensional** wie eine Tabelle sein. Zur Definition eines Feldes wird der DIM-Befehl (Abkürzung für Dimension) verwendet. Felder müssen vor Gebrauch immer "vereinbart" (definiert) werden. (Die Ein-Wert-Variablen, die wir bislang benutzt haben, erforderten keine Definition.) Das Format des DIM-Befehls für numerische Felder ist:

DIM variablenname (größe)

Dabei bedeuten:

variablenname ist ein Variablenname, der den bereits diskutierten Regeln für die Namen numerischer Variablen entspricht.

größe ist die Anzahl der Speicherplätze und muß zwischen 0 und 255 liegen. Wird die Größe spezifiziert, wird ein Speicherplatz mehr als die spezifizierte Anzahl bereitgestellt.

Beispiele für DIM-Befehle für numerische Felder:

```
DIM X (5)
DIM AA (24)
DIM Q5 (0)
```

Der erste Befehl vereinbart das Feld X mit 6 Speicherplätzen, der zweite das Feld AA mit 25 Speicherplätzen. Der dritte Befehl erzeugt das Feld Q5 mit einem Speicherplatz und ist nicht sehr sinnvoll (zumindest für Zahlen), da genausogut eine einwertige numerische Variable vereinbart werden könnte.

Es ist wichtig zu wissen, daß im SHARP-Rechner die Feldvariable X und die Variable X separat abgespeichert und unterschieden werden. Das erste X bezeichnet eine Serie numerischer Speicherplätze und das zweite X einen einzelnen und separaten Speicherplatz.

Nachdem Sie jetzt wissen, wie Felder vereinbart werden, fragen Sie vielleicht, wie man die einzelnen Speicherplätze erreicht. Die gesamte Gruppe der Speicherplätze (das Feld) hat einen gemeinsamen Namen. Daher verwenden wir zum Ansprechen eines einzelnen Speicherplatzes ("Element" genannt) den Namen und eine Zahl in Klammern. Diese Zahl wird als "Index" bezeichnet. Um beispielsweise die Zahl 8 im fünften Element des vorher vereinbarten Feldes X zu speichern, schreiben wir:

$$X(4) = 8$$

Lassen Sie sich nicht durch den Index 4 verwirren. Die Nummerierung der Feldelemente beginnt mit Null bis zur Obergrenze, die im DIM-Befehl definiert wurde.

Die Vorteile von Feldern sind, daß ein Ausdruck oder ein Variablenname als Index eingesetzt werden kann.

Zur Vereinbarung eines String-Feldes hat der DIM-Befehl eine etwas andere Form:

DIM string-variablenname {größe} * länge

Dabei bedeuten:

string-variablenname ist ein Variablenname, der den vorher diskutierten Regeln für normale String-Variablen entspricht.

größe ist die Anzahl der Speicherplätze und muß im Bereich zwischen 0 und 255 liegen. Die Anzahl der Speicherplätze ist um Eins größer als die spezifizierte Anzahl.

* länge ist fakultativ. Falls angegeben, wird hierdurch die Länge der einzelnen Strings definiert, die das Feld ergeben. Die Länge muß im Bereich zwischen 1 und 80 liegen. Wird die Länge nicht spezifiziert, wird automatisch der Wert 16 vorgegeben.

Beispiele für Vereinbarung von String-Felder:

DIM X\$ (4)

DIM NMS (10) * 10

DIM IN\$ (1) * 80
 DIM R\$ (0) * 26

Im ersten Beispiel wird ein Feld mit fünf Strings mit je 16 Zeichen vereinbart. Der zweite DIM-Befehl definiert das Feld NM mit elf Strings zu je 10 Zeichen. Ein Möglichkeit, Speicherplatz zu sparen, ist die Definition von Stringlängen, die kleiner als der vorgegebene Wert sind. Das dritte Beispiel vereinbart ein Feld mit zwei Elementen mit je 80 Zeichen und das vierte einen einzigen String mit 26 Zeichen.

Neben den Feldern, die wir gerade kennengelernt haben, ermöglicht der PC-1421 die Vereinbarung von "zweidimensionalen" Feldern. Ein eindimensionales Feld kann man sich als Liste von Daten vorstellen, die untereinander in einer einzelnen Spalte angeordnet sind. Ein zweidimensionales Feld ist dann eine Tabelle mit Zeilen und Spalten. Zweidimensionale Felder werden wie folgt spezifiziert:

DIM variablenname (zeilen, spalten)

oder

DIM string-variablenname (zeilen, spalten) * länge

Dabei bedeuten:

zeilen die Anzahl der Zeilen des Feldes. Diese Zahl muß im Bereich 0 bis 255 liegen. Die tatsächliche Anzahl der Zeilen ist eine Zeile mehr als die spezifizierte Anzahl.

spalten ist die Anzahl der Spalten des Feldes. Diese Zahl muß im Bereich 0 bis 255 liegen. Die tatsächliche Anzahl der Spalten ist eine Spalte mehr als die spezifizierte Anzahl.

Die folgende Abbildung erklärt die Speicherplätze, die durch die Definition DIM T (2, 3) bereitgestellt werden, und die Indices (die jetzt aus zwei Zahlen bestehen) zum Ansprechen der einzelnen Speicherplätze.

	Spalte 0	Spalte 1	Spalte 2	Spalte 3
Zeile 0	T (0, 0)	T (0, 1)	T (0, 2)	T (0, 3)
Zeile 1	T (1, 0)	T (1, 1)	T (1, 2)	T (1, 3)
Zeile 2	T (2, 0)	T (2, 1)	T (2, 2)	T (2, 3)

Hinweis: Zweidimensionale Felder verbrauchen sehr viel Speicherplatz. Beispielsweise belegt ein Feld mit 25 Zeilen und 35 Spalten 875 Speicherplätze.

Felder sind zum effektiven Programmieren unerlässlich.

Die folgende Tabelle enthält die Anzahl der Bytes, die zur Definition der einzelnen Variablen benötigt werden, und die Anzahl der Bytes pro Programm-Anweisung.

Variable	Variablenname	Daten	
Numerische Variable	7 Bytes	8 Bytes	
String-Variable	7 Bytes	Feldvariable	Spezifizierte Anzahl
		Einfache Variable (2-Zeichen-Variable)	16 Bytes

* Wird beispielsweise `DIM Z$(2, 3) * 10` spezifiziert, werden 12 Variable mit je 10 Zeichen bereitgestellt. Dafür sind 7 Bytes (Variablenname) + 10 Bytes (Anzahl der Zeichen) x 12 = 127 Bytes erforderlich.

Element	Zeilennummer	Anweisungen und Funktionen	Sonstige, ENTER
Anzahl der verwendeten Bytes	3 Bytes	1 Byte	1 Byte

Variable der Form A ()

Ein Bereich im Speicher des Computers ist für feste Variable reserviert. Dieser Bereich kann auch benutzt werden, um indizierte Variable zu speichern, die die gleiche Form wie Feldvariable haben.

26 feste Variablenamen sind verfügbar, d.h. A bis Z (A\$ bis Z\$). Jeder dieser Namen kann mit den Zahlen 1 bis 26 indiziert werden, x.B. A(1) bis A(26) oder A\$(1) bis A\$(26). Das bedeutet, daß die Variable A(1) anstelle von A, A(2) anstelle von B, A(3) anstelle von C usw. eingesetzt werden kann.

Wurde jedoch bereits ein Feld mit dem Namen A oder A\$ mit einem DIM-Befehl definiert, können indizierte Variable mit dem Namen A nicht mehr verwendet werden. Wenn zum Beispiel das Feld A durch `DIM A(5)` definiert wurde, werden die Speicherplätze A(0) bis A(5) im Programm/Daten-Bereich bereitgestellt. Wird danach die Variable A(2) spezifiziert, bezieht sie sich nicht auf die feste Variable B, sondern auf die Feldvariable A(2), die im Programm/Daten-Bereich definiert ist. Wird A(9) spezifiziert, tritt ein Fehler auf, weil A(9) außerhalb des Bereichs der durch `DIM A(5)` spezifizierten Dimension liegt.

Wurden umgekehrt bereits indizierte Variable der Form A () spezifiziert, ist es nicht möglich, ein Feld A oder A\$ mit dem DIM-Befehl zu vereinbaren, es sei denn, die Definition der indizierten Variablen wird mit dem CLEAR-Befehl gelöscht.

* Verwendung größerer Indices als 26:

Werden für indizierte Variable A () größere Indices als 26 verwendet und A ist nicht durch eine DIM-Anweisung definiert, dann werden die entsprechenden Speicherplätze im Programm/Daten-Bereich für die Variable A () bereitgestellt. Durch die Ausführung von A(35) = 5 werden beispielsweise Speicherplätze für die Variable A(27) bis A(35) im Programm/Daten-Bereich reserviert.

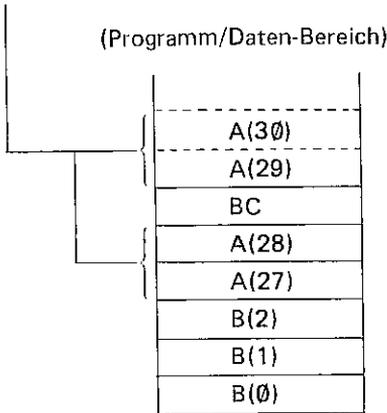
Variable mit höheren Indices als 26 werden als Feldvariable angesehen, unterliegen jedoch den folgenden Einschränkungen:

- (1) Die Speicherplätze für ein Feld mit dem gleichen Namen müssen im Programm/Daten-Bereich aufeinanderfolgend sein, ansonsten entsteht ein Fehler.

```

10 DIM B(2)
20 A(28) = 5
30 BC = 12
40 A(30) = 9
    
```

Bei der Ausführung dieses Programms wird das Feld "A" nicht in zwei aufeinanderfolgenden Segmenten im Programmbereich definiert, so daß in Zeile 40 ein Fehler auftritt.



- (2) Numerische Feldvariable und String-Feldvariable mit dem gleichen Index können nicht gleichzeitig definiert werden. Beispielsweise können A(30) und A\$(30) nicht gleichzeitig definiert werden, weil beide den gleichen Speicherplatz im Programm/Daten-Bereich belegen.

- (3) Zweidimensionale Felder können nicht definiert werden, es ist auch nicht möglich, die Länge der Zeichenstrings für String-Feldvariable zu spezifizieren. Zum Beispiel ist die Länge eines Zeichenstrings, der in der String-Feldvariablen A\$() untergebracht werden kann, auf sieben Zeichen oder weniger begrenzt.
- (4) Variable mit dem Index Null (0) können nicht definiert werden. Wird A(0) oder AS(0) definiert, tritt ein Fehler auf.

Ausdrücke

Ein **Ausdruck** ist eine Zusammenstellung von Variablen, Konstanten und Operatoren, die durch Anwendung von Rechenoperationen auf einen einzigen Wert zusammengefaßt werden kann. Die Berechnungen, die Sie in Kapitel 3 durchgeführt haben, waren Beispiele für Ausdrücke. Ausdrücke sind ein wesentlicher Bestandteil von BASIC-Programmen. Beispielsweise kann ein Ausdruck eine Formel zum Berechnen einer Gleichung sein, ein Test zur Bestimmung des Verhältnisses zwischen zwei Größen oder ein Mittel zur Formatierung von Strings.

Numerische Operatoren

Der **PC-1421** verfügt über fünf **numerische Operatoren**. Dies sind die arithmetischen Rechenzeichen, die Sie beim Kennenlernen des **PC-1421** als Taschenrechner in Kapitel 3 benutzt haben.

- + Addition
- Subtraktion
- * Multiplikation
- / Division
- ^ Potenzierung

Ein **numerischer Ausdruck** wird auf die gleiche Weise zusammengesetzt, wie Sie komplexe Rechenoperationen eingegeben haben. Numerische Ausdrücke können aus jeder sinnvollen Kombination von numerischen Konstanten, numerischen Variablen und diesen numerischen Operatoren bestehen:

$$(A * B) ^ 2$$
$$A (2, 3) + A (3, 4) + 5.0 - C$$
$$(A/B) * (C + D)$$

String-Ausdrücke

String-Ausdrücke sind den numerischen Ausdrücken ähnlich, allerdings gibt es nur einen einzigen String-Operator – die Verkettung (+). Als Symbol wird das normale Pluszeichen verwendet. Steht dieses Symbol zwischen zwei Strings, wird der zweite

String an den ersten angehängt, so daß ein langer String entsteht. Bei komplexerer String-Verkettung und anderen String-Operationen muß beachtet werden, daß der Arbeitsbereich des **PC-1421** für String-Operationen auf nur 79 Zeichen begrenzt ist.

HINWEIS: String-Einheiten und numerische Einheiten können nicht in einem Ausdruck verwendet werden, außer wenn ein der Funktionen benutzt wird, die einen String-Wert in einen numerischen Wert oder umgekehrt umwandeln.

"15" + 10 ist nicht zulässig
 "15" + "10" ergibt "1510", nicht "25"

Verhältnis-Ausdrücke

Ein **Verhältnis-Ausdruck** vergleicht zwei Ausdrücke und bestimmt, ob das angegebene Verhältnis wahr oder falsch ist. Die Verhältnis-Operatoren sind:

- > größer als
- >= größer oder gleich
- = gleich
- <> ungleich
- <= kleiner oder gleich
- < kleiner als

Die folgenden Beispiele sind gültige Verhältnis-Ausdrücke:

A < B
 C(1, 2) >= 5
 D(3) <> 8

Mit A gleich 10, B gleich 12, C(1, 2) gleich 6 und D(3) gleich 9 wären diese Verhältnis-Ausdrücke alle wahr. Auch Zeichen-Strings können in Verhältnis-Ausdrücken verglichen werden. Dabei werden die beiden Strings, beginnend mit dem ersten Zeichen, Zeichen für Zeichen nach ihrem ASCII-Wert verglichen (siehe Anhang B für die ASCII-Werte). Ist ein String kürzer als der andere, wird für die fehlenden Zeichen \emptyset oder NUL eingesetzt. Die folgenden Verhältnis-Ausdrücke sind alle wahr:

"ABCDEF" = "ABCDEF"
 "ABCDEF" <> "ABCDE"
 "ABCDEF" > "ABCDE"

Verhältnis-Ausdrücke haben zwei mögliche Ergebnisse: wahr oder falsch. Im **PC-1421** wird "wahr" durch 1 und "falsch" durch \emptyset dargestellt. In einem logischen Test wird ein Ausdruck mit dem Wert 1 oder höher als "wahr" und ein Ausdruck mit dem Wert \emptyset oder weniger als "falsch" angesehen. Gute Programmierung

erfordert jedoch die Verwendung von expliziten Verhältnis-Ausdrücken, man darf sich nicht auf diese Übereinstimmung verlassen.

Logische Ausdrücke

Logische Ausdrücke sind Verhältnis-Ausdrücke, die die Operatoren AND, OR und NOT benutzen. AND und OR dienen zum Verknüpfen von zwei Verhältnis-Ausdrücken; die folgenden Tabellen zeigen die Werte, die die verknüpften Ausdrücke annehmen:

A AND B

		Wert von A	
		Wahr	Falsch
Wert von B	Wahr	Wahr	Falsch
	Falsch	Falsch	Falsch

A OR B

		Wert von A	
		Wahr	Falsch
Wert von B	Wahr	Wahr	Wahr
	Falsch	Wahr	Falsch

(Hinweis: Die Werte von A und B müssen 0 oder 1 sein.)

- Dezimalzahlen werden wie folgt in binärer Schreibweise mit 16 Bit dargestellt:

DEZIMALE DARSTELLUNG	BINÄRE DARSTELLUNG MIT 16 BIT
32767	0111111111111111
⋮	
3	0000000000000011
2	0000000000000010
1	0000000000000001
0	0000000000000000
-1	1111111111111111
-2	1111111111111110
-3	1111111111111101
⋮	⋮
-32768 →	1000000000000000

Die negative Form (NOT) der Binärzahl 0000000000000001 wird wie folgt erhalten:

NOT 0000000000000001
 (negativ) → 1111111111111110

So wird Bit für Bit 1 zu 0 bzw. 0 zu 1 umgekehrt, was "Invertierung (NOT)" genannt wird.

Die Addition von 1 und NOT 1 ergibt somit das folgende Resultat:

	0000000000000001	(1)
+)	1111111111111110	(NOT 1)
	1111111111111111	(-1)

Alle Bits werden zu 1. Entsprechend der obigen Zahlenliste werden die Bits in dezimaler Schreibweise -1, d.h. $1 + \text{NOT } 1 = -1$.

Die Beziehung zwischen einem numerischen Wert X und seiner negativen Form (NOT X) ist:

$$X + \text{NOT } X = -1$$

Daraus folgt die Gleichung $\text{NOT } X = -X - 1$, d.h. $\text{NOT } X = -(X + 1)$

Aus dieser Gleichung lassen sich die folgenden Ergebnisse ableiten.

- NOT 0 = -1
- NOT -1 = 0
- NOT -2 = 1

Mit diesen Operatoren können mehr als zwei Verhältnis-Ausdrücke verknüpft werden. Dabei sollten Klammern verwendet werden, um den beabsichtigten Vergleich zu verdeutlichen.

- (A < 9) AND (B > 5)
- (A >= 10) AND NOT (A > 20)
- (C = 5) OR (C = 6) OR (C = 7)

Der **PC-1421** behandelt logische Operatoren "bitweise" als logische Funktionen auf der Basis von 16 Bit. (Siehe den Hinweis zu Verhältnis-Ausdrücken und wahr/falsch.) In normalen Operationen ist dies nicht von Bedeutung, weil einfach 0 oder 1 (wahr oder falsch) als Ergebnis der Verhältnis-Ausdrücke nur je ein Bit belegen. Wenn Sie jedoch logische Operatoren auf andere Werte als 0 und 1 anwenden, wird jedes Bit einzeln betrachtet. Beispielsweise ergibt sich mit A = 17 und B = 22 für (A OR B) der Wert 23:

BASIC-Begriffe und -Ausdrücke

17 in binärer Darstellung ist 10001

22 in binärer Darstellung ist 10110

17 OR 22 ergibt 10111 (1 wenn 1 in einer der Zahlen, sonst 0)

10111 in dezimaler Darstellung ist 23.

Sobald Sie mehr Erfahrungen als Programmierer gesammelt haben, werden Sie den Wert dieser Art Operationen erkennen. Anfänger sollte sich jedoch, solange sie noch Anfänger sind, an eindeutigen Wahr/Falsch-Verhältnis-Ausdrücken halten.

Klammerung und Vorrang der Operatoren

Bei der Bearbeitung komplexer Ausdrücke befolgt der PC-1421 vordefinierte Prioritäten, die die Reihenfolge für die Abarbeitung der Operatoren bestimmen. Dies kann sehr wichtig sein:

$5 + 2 * 3$ kann bedeuten

$$5 + 2 = 7$$

oder

$$2 * 3 = 6$$

$$7 * 3 = 21$$

$$6 + 5 = 11$$

Die genauen Regeln für den Vorrang der Operatoren finden Sie in Anhang D.

Wenn Sie sich nicht all diese Regeln merken wollen und um außerdem das Programm übersichtlicher zu gestalten, sollte die Reihenfolge der Abarbeitung durch Klammerung verdeutlicht werden. Dann ergibt sich für das obige Beispiel:

$$(5 + 2) * 3$$

oder

$$5 + (2 * 3)$$

RUN-Modus

Jeder der obigen Ausdrücke kann generell genauso im RUN-Modus wie bei der Programmierung einer BASIC-Anweisung verwendet werden. Im RUN-Modus wird ein Ausdruck sofort ausgerechnet und das Ergebnis angezeigt, z.B.:

Eingabe

Anzeige

(5 > 3) AND (2 < 6) **ENTER**

1.

Die 1 bedeutet, daß der Ausdruck wahr ist.

Funktionen

Funktionen sind spezielle Bestandteile der Programmiersprache BASIC, die einen Wert in einen anderen Wert umwandeln. Funktionen verhalten sich wie Variable, deren Wert durch den Wert anderer Variable oder Ausdrücke bestimmt wird. Die Funktion ABS ergibt den Absolutwert des Arguments:

ABS (-5)	ist	5
ABS (6)	ist	6

Die Funktion LOG berechnet den dekadischen Logarithmus (Logarithmus zur Basis 10) des Arguments.

LOG (100)	ist	2
LOG (1000)	ist	3

Funktionen können an jeder Stelle eingesetzt werden, wo Variable verwendet werden können. Bei vielen Funktionen ist Klammerung nicht erforderlich:

LOG 100	ist gleich wie	LOG (100)
---------	----------------	-----------

Für Funktion, die mehr als ein Argument haben, muß Klammerung verwendet werden. Programme werden viel übersichtlicher, wenn für alle Funktionen Klammerung benutzt wird.

In Kapitel 8 finden Sie eine vollständige Liste der Funktionen, über die der PC-1421 verfügt.

KAPITEL 5

PROGRAMMIEREN MIT DEM PC-1421

Im vorigen Kapitel haben Sie schon einige Begriffe und Ausdrücke der Programmiersprache BASIC kennengelernt. In diesem Kapitel wollen wir diese Elemente benutzen, um Programme für den **PC-1421** zu schreiben. Wir möchten aber noch einmal darauf hinweisen, daß diese Bedienungsanleitung kein Lehrbuch für Programmieren in BASIC ist. Dieses Kapitel will Sie mit der Verwendung von BASIC für den **PC-1421** vertraut machen.

Programme

Ein **Programm** besteht aus einer Reihe von Befehlen für den Computer. Vergessen Sie nicht, daß der **PC-1421** nur eine Maschine ist und daher genau das und nur das ausführt, was Sie ihm sagen. Sie als Programmierer sind für die Richtigkeit und Durchführbarkeit der Befehle verantwortlich.

BASIC-Statements

Der **PC-1421** setzt Programme nach einem bestimmten Format um. Dieses Format wird **Statement** genannt. BASIC-Statements werden immer in der gleichen Form eingegeben. Ein Statement muß mit einer Zeilennummer beginnen:

```
10: PRINT "HELLO"
20: END
30:   :
```

Zeilennummern

Jede Programmzeile muß eine eigene Zeilennummer haben, eine ganze Zahl zwischen 1 und 65279. Die Zeilennummern sind die Bezugspunkte des Computers. Sie sagen dem **PC-1421**, in welcher Reihenfolge ein Programm abzuarbeiten ist. Sie brauchen die Programmzeilen nicht der laufenden Nummer nach einzugeben (obwohl wir es für Anfänger empfehlen, weil es weniger verwirrend ist). Der Computer beginnt die Ausführung eines Programms immer mit der niedrigsten Zeilennummer und arbeitet die Programmzeilen in aufsteigender Reihenfolge der Zeilennummern ab.

Beim Programmieren ist es sinnvoll, bei der Nummerierung der Zeilen Zwischenräume zu lassen (10, 20, 30, . . . , 10, 30, 50 usw.), um später noch zusätzliche Zeilen einfügen zu können.

ACHTUNG: Verwenden Sie nicht die gleichen Zeilennummern in verschiedenen Programmen. Wenn Sie die gleichen Nummern verwenden, werden bei der Eingabe von neuen Zeilen die vorigen Zeilen mit den gleichen Nummern gelöscht.

BASIC-Befehle

Alle BASIC-Statements müssen **Befehle** enthalten. Diese Befehle sagen dem Computer, was er machen soll. Befehle sind immer Programmbestandteil und bewirken keine unmittelbare Operation.

```
10: PRINT "HELLO"
20: END
30: :
```

Einige Statements fordern oder erlauben **Operanden**:

```
10: PRINT "HELLO"
20: END
30: :
```

Operanden geben dem Computer weitere Informationen darüber, welche Daten der Befehl verarbeitet. Bei einigen Befehlen sind Operanden erforderlich, bei anderen sind sie fakultativ. Für bestimmte Befehle dürfen keine Operanden eingesetzt werden. (Kapitel 9 enthält eine vollständige Liste der BASIC-Befehle und ihrer Verwendung für den **PC-1421**.)

BASIC-Kommandos

Kommandos sind Anweisungen für den Computer, die von außerhalb des Programms eingegeben werden. Kommandos bewirken, daß der Computer bestimmte Prozesse mit dem Programm ausführt oder einen bestimmten Modus einstellt, der dann wieder die Programmausführung festlegt.

Anders als Befehle zeigen Kommandos sofortige Wirkung – unmittelbar nach Eingabe eines Kommandos (durch Drücken der Taste **ENTER**) wird das Kommando ausgeführt. Kommandos haben **keine** Zeilennummern:

```
RUN
NEW
RADIAN
```

Einige Befehle können auch als Kommandos verwendet werden. (In Kapitel 9 finden Sie eine vollständige Liste der BASIC-Kommandos und ihrer Benutzung für den **PC-1421**.)

Modi

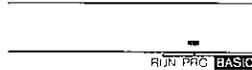
Der **RUN-Modus** dient auch zur Ausführung der Programme. Im **PRO-Modus** werden Programme eingegeben und korrigiert.

Erstes Programmieren des PC-1421

Nachdem Sie den **PC-1421** als Taschenrechner jetzt kennengelernt haben, sind Sie mit der Tastatur weitgehend vertraut. Ab jetzt werden wir bei der Darstellung von Eingaben nicht jede einzelne Tastenbetätigung angeben. Vergessen Sie nicht, für die Zeichen, die oberhalb der Tasten angegeben sind, **SHIFT** zu drücken, und beenden Sie jede Eingabezeile durch Drücken von **ENTER**.

Jetzt können Sie mit dem Programmieren beginnen.

Stellen Sie den Ein/Aus-Schalter auf ON und drücken dann die Taste **BASIC** zweimal. Auf dem Display erscheint die folgende Information.



Die obige Anzeige bedeutet, daß der Computer im PRO-Modus (Programmierung) ist.

(Falls sich der Strich über CAL oder RUN befindet, drücken Sie die Taste **BASIC** ein- bzw. zweimal.)

Geben Sie das Kommando NEW ein.

Eingabe

Anzeige

NEW



Das NEW-Kommando bewirkt, daß alle Programme und Daten im Speicher des **PC-1421** gelöscht werden. Nach Drücken von **ENTER** erscheint das Bereitschafts-symbol. Es zeigt an, daß der Computer jetzt eine Eingabe erwartet.

Beispiel 1 – Eingeben und Ausführen eines Programms

Stellen Sie den **PC-1421** auf den PRO-Modus und geben das folgende Programm ein:

Eingabe

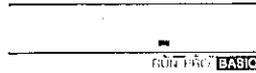
Anzeige

10 PRINT "HELLO"

10:PRINT "HELLO"

Beachten Sie, daß nach Drücken von **ENTER** der **PC-1421** die Eingabe anzeigt und automatisch zwischen der Zeilennummer und dem Befehl einen Doppelpunkt (:) einfügt. Vergewissern Sie sich, daß Sie das Statement in der richtigen Form eingegeben haben.

Drücken Sie jetzt die Taste **[BASIC]** , um den Computer in den RUN-Modus zu schalten.



Eingabe

Anzeige

RUN

HELLO

Da dieses Programm aus nur einer Zeile besteht, stoppt die Ausführung an diesem Punkt. Drücken Sie **[ENTER]** , um das Programm zu verlassen, und geben Sie RUN ein, wenn das Programm noch einmal laufen soll.

Beispiel 2 — Editieren eines Programms

Um die Meldung, die das Programm anzeigt, zu ändern, müssen Sie das Programm "editieren". Bei diesem Programm mit nur einer Zeile könnten Sie genauso gut die Zeile neu eingeben, aber für längere Programme wird Editieren zu einem wichtigen Bestandteil des Programmierens. Wir wollen jetzt das Programm editieren, das Sie soeben geschrieben haben.

Ist der Computer noch im RUN-Modus? Wenn ja, wechseln Sie auf den PRO-Modus.

Zum Editieren müssen Sie das gespeicherte Programm abrufen. Dafür wird der aufwärtsweisende Pfeil (**↑**) benutzt. Wenn das Programm vollständig ausgeführt wurde, wird durch Betätigung der Taste **↑** die letzte Programmzeile abgerufen. Trat während der Ausführung ein Fehler auf oder wurde die Ausführung durch Betätigung der Taste **↑** unterbrochen, ruft **[BRK]** die Zeile ab, in der der Fehler auftrat bzw. die Programmausführung unterbrochen wurde. Um Änderungen im Programm vorzunehmen, gehen Sie mit der Taste **↑** in Richtung Programm-anfang (die vorhergehende Zeile abrufen) und mit der Taste **↓** in Richtung Programmende (die folgende Zeile anzeigen). Wird **↑** oder **↓** gedrückt gehalten, "rollt" das Programm auf dem Display nach oben bzw. unten, d.h. die einzelnen Zeilen laufen nacheinander durch das Display.

Wir haben schon erklärt, wie der Cursor mit dem rechten Pfeil **▶** und dem linken Pfeil **◀** innerhalb einer Zeile bewegt wird. Bringen Sie mit der Taste **▶** den Cursor auf das erste Zeichen, das Sie ändern möchten.

Programmieren

Eingabe

↑

◀◀◀◀◀

Anzeige

10: PRINT "HELLO"

10 PRINT "HELLO"

Beachten Sie, daß der Cursor die Form eines blinkenden Rechtecks hat, um anzuzeigen, daß er sich "auf" einem Zeichen befindet. Machen Sie die folgende Eingabe:

Eingabe

GOOD"!_

Anzeige

10 "GOOD"!_

Vergessen Sie nicht, am Ende der Zeile **ENTER** zu drücken. Schalten Sie durch Betätigung von **BASIC** um in den RUN-Modus.

Eingabe

RUN

Anzeige

ERROR 1 IN 10

Dies ist eine neue Art Fehlermeldung. Nicht nur der Fehlertyp ist angegeben (unser alter Freund der Syntaxfehler), sondern auch die **Zeilennummer**, in der der Fehler auftrat.

Drücken Sie **C-C** und kehren dann zurück in den PRO-Modus. Korrekturen des Programms können nur im PRO-Modus ausgeführt werden. Rufen Sie mit der Taste ↑ die letzte Programmzeile ab.

Eingabe

↑

Anzeige

10: PRINT "GOOD"!

Der blinkende Cursor befindet sich auf dem Problem. In Kapitel 4 haben Sie gelernt, daß bei der Eingabe von String-Konstanten in BASIC alle Zeichen von Anführungszeichen eingeschlossen sein müssen. Löschen Sie das "!" mit der Taste DEL:

Eingabe

DEL

Anzeige

10 PRINT "GOOD" _

Fügen Sie als nächstes das ! an der richtigen Stelle ein. Zum Editieren von Programmen werden DEL und INS auf die gleiche Weise wie zum Korrigieren von Rechenformeln (siehe Kapitel 3) verwendet. Setzen Sie mithilfe der Taste  den Cursor auf das Zeichen, vor dem Sie die Einfügung vornehmen wollen.

Eingabe

Anzeige

◀

10 PRINT "GOOD"

Drücken Sie die Taste INS. Auf der Stelle, wo neue Daten eingegeben werden können, erscheint .

Eingabe

Anzeige

INS

10 PRINT "GOOD"

Geben Sie das ! ein. Die Anzeige sieht dann wie folgt aus:

Eingabe

Anzeige

!

10 PRINT "GOOD!"

Vergessen Sie nicht,  zu drücken, um die Korrektur in das Programm einzugeben.

HINWEIS: Wenn Sie eine ganze Zeile aus dem Programm löschen wollen, geben Sie einfach nur die Zeilennummer ein. Dadurch wird die ursprüngliche Zeile gelöscht.

Beispiel 3 — Der Gebrauch von Variablen beim Programmieren

Sollten Sie mit dem Gebrauch von numerischen und String-Variablen noch nicht vertraut sein, lesen Sie noch einmal die entsprechenden Abschnitte in Kapitel 4.

Die Benutzung von Variablen beim Programmieren erlaubt eine viel bessere Ausnutzung der Fähigkeiten des PC-1421.

Wie Sie sich erinnern, werden numerische feste Variable mit den Buchstaben A bis Z bezeichnet:

A = 5

Für String-Variablen werden Buchstaben, gefolgt von einem Dollarzeichen, verwendet. Sie dürfen aber nicht den gleichen Buchstaben für eine feste numerische

und eine feste String-Variable verwenden! Beispielsweise dürfen Sie im gleichen Programm nicht A und A\$ benutzen.

Beachten Sie, daß feste String-Variable in der Länge 7 Zeichen nicht überschreiten dürfen:

A\$ = "TOTAL"

Die Werte, die Variablen zugewiesen werden, können während der Ausführung des Programms geändert werden. Die Variable nehmen die Werte an, die eingegeben oder berechnet werden. Eine Möglichkeit dafür ist der Befehl INPUT. Im folgenden Programm ändert sich der Wert von A\$ entsprechend der Daten, die als Antwort auf die Aufforderung "WORD?" eingegeben werden. Geben Sie zunächst das Programm ein:

```

10 INPUT "WORD?"; A$
20 B = LEN (A$)
30 PRINT "WORD_ _IS_ _"; B; " _LTRS"
40 END
    
```

↑ bedeutet Leerstelle

Bevor Sie das Programm laufen lassen, werfen Sie noch einen Blick auf einige neue Merkmale. In Zeile 30 wird die maximale Anzeigekapazität von 16 Zeichen des **PC-1421** überschritten. Wenn eine Zeile länger als 16 Zeichen ist (bis zu maximal 79 Zeichen), wird bei Überschreiten der 16 Zeichen die Anzeige nach links bewegt, wobei aber die vorige Eingabe erhalten bleibt. Diese Bewegung nach links wird als horizontales Rollen bezeichnet.

Das zweite neue Element in diesem Programm ist die Verwendung des Statements **END** am Programmende. **END** teilt dem Computer mit, daß das Programm fertig ist. Man sollte jedes Programm mit dem Statement **END** beschließen.

Wenn Ihre Programme umfangreicher werden, möchten Sie sie vor der Ausführung möglicherweise noch einmal durchsehen. Zu diesem Zweck benutzen Sie das Kommando **LIST**. Auf das Kommando **LIST**, das nur im **PRO**-Modus verwendet werden kann, wird der Programmanfang mit der niedrigsten Zeilennummer angezeigt.

Listen Sie Ihr Programm auf:

Eingabe

Anzeige

LIST

10: INPUT "WORD?"

Mit den Tasten  und  können Sie sich im Programm bewegen und es genau ansehen. Zum Überprüfen einer Zeile, die mehr als 16 Zeichen hat, bringen Sie den Cursor zur rechten Seite des Displays, dann erscheinen die anderen Zeichen

auf dem Display. Nach der Überprüfung starten Sie das Programm:

Eingabe

Anzeige

RUN

WORD?_

HELP

WORD IS 4. LTRS

ENTER

>

Damit ist das Programm schon zu Ende. Sie könnten jetzt natürlich durch Eingabe von RUN das Programm laufen lassen. Es würde aber viel mehr Spaß machen, wenn man mehr als ein Wort eingeben könnte. Daher wollen wir das Programm derart ändern, daß es weiterarbeitet, ohne daß jedesmal RUN eingegeben werden muß.

Schalten Sie um in den PRO-Modus und zeigen Sie mit den Pfeilen (oder LIST) Zeile 40 an.

Sie können 40 eingeben, um die gesamte Zeile zu löschen, oder mit ► den Cursor auf das E von END setzen. Ändern Sie Zeile 40 folgendermaßen:

40: GOTO 10

Und jetzt starten Sie das geänderte Programm.

Das Statement GOTO erzeugt eine Schleife im Programm, so daß die gleiche Operation immer wieder wiederholt wird. Da die Schleife nicht begrenzt wurde, wird sie "unendlich" weiterlaufen. Daher müssen Sie zum Stoppen dieses Programms die Taste **BRK** drücken.

Wenn Sie das Programm mit der Taste **BRK** gestoppt haben, können Sie es mit dem Kommando CONT weiterlaufen lassen. CONT bedeutet Fortsetzung. Das Kommando CONT bewirkt, daß die Programmausführung in der Zeile fortgesetzt wird, die bei Drücken der Taste **BRK** gerade ausgeführt wurde.

Beispiel 4 — Komplexere Programme

Obwohl der PC-1421 mit einer Fakultät-Funktion ausgestattet ist, wollen wir in diesem Abschnitt die Berechnung der n-ten Fakultät als Beispiel für komplexere Programmierung erklären.

Das folgende Programm berechnet die N-te Fakultät (N!). Das Programm beginnt mit 1 und berechnet N! bis zur eingegebenen Grenze (LIMIT im Programm). Geben Sie das folgende Programm ein.

Programmieren

```
100 F = 1: WAIT 118
110 INPUT "LIMIT? "; L
120 FOR N = 1 TO L
130 F = F * N
140 PRINT N, F
150 NEXT N
160 END
```

Dieses Programm enthält mehrere Neuerungen. Der Befehl WAIT in Zeile 100 bestimmt die Zeitdauer, für die die Anzeige bleibt, bis das Programm fortgesetzt wird. Die Zahlen und ihre Fakultäten werden angezeigt, nachdem sie berechnet wurden. Die Zeitdauer ihrer Anzeige ist durch den Befehl WAIT auf etwa 2 Sekunden festgesetzt, so daß Sie die Taste **ENTER** nicht drücken brauchen.

Beachten Sie auch, daß die Zeile 100 zwei Statements enthält, die durch einen Doppelpunkt (:) getrennt sind. **Sie können beliebig viele Statements, die jeweils durch Doppelpunkte getrennt sind, in eine Zeile setzen. Die maximale Anzahl Zeichen pro Zeile von 80 einschließlich ENTER darf jedoch nicht überschritten werden.** Derartige Verbundanweisungen machen Programme jedoch schwer lesbar und unübersichtlich. Daher sollten sie nur benutzt werden, wenn die Einzel-Statements sehr einfach sind oder wenn es besondere Gründe gibt, mehrere Statements in einer Zeile zusammenzufassen.

In diesem Programm haben wir mit dem Befehl FOR in Zeile 120 und dem Befehl NEXT in Zeile 150 eine Schleife erzeugt. In Beispiel 3 haben Sie eine "unendliche" Schleife programmiert, die die Statements innerhalb der Schleife immer wieder ausgeführt hat, bis Sie die Taste **BRK** drückten. Bei der FOR/NEXT-Schleife addiert der **PC-1421** jedesmal bei Erreichen des Befehls NEXT 1 zu N. Danach wird überprüft, ob N größer ist als L. Ist N kleiner oder gleich L, kehrt die Ausführung zum Schleifenanfang zurück, und die Statements werden wieder ausgeführt. Ist N größer als L, wird die Ausführung mit der nächsten Zeile fortgesetzt, in unserem Beispiel Zeile 160, wo das Programm stoppt.

In einer FOR/NEXT-Schleife können Sie jede numerische Variable benutzen. Sie brauchen nicht mit 1 anzufangen, und Sie können in jedem Schritt jeden beliebigen Betrag hinzuaddieren. Einzelheiten finden Sie in Kapitel 8.

Wir haben dieses Programm mit Zeilennummern versehen, die mit 100 beginnen. Mehrere Programme können gleichzeitig im Speicher enthalten sein, wenn sie verschiedene Zeilennummern haben. Um dieses Programm, und nicht das Programm, das mit Zeile 10 beginnt, zu starten, machen Sie die folgende Eingabe:

```
RUN 100
```

Zusätzlich zur Ausführung verschiedener Programme durch Eingabe der Anfangs-Zeilenummer können Sie den Programmen auch einen Buchstaben als Programmnamen geben und dann mit der Taste **[DEF]** starten (siehe Kapitel 6).

Sie werden bemerkt haben, daß während der Ausführung eines Programms, wenn keine Anzeige auf dem Display vorhanden ist, die BUSY-Anzeige leuchtet. Lassen Sie das Programm noch einige Male mit verschiedenen L-Werten laufen.

Speichern von Programmen

Programme bleiben im Speicher erhalten, wenn Sie den **PC-1421** ausschalten oder das Gerät sich automatisch ausschaltet. Die Programme bleiben auch erhalten, wenn Sie die Tasten **[BRK]**, **[C-CE]** oder **[CA]** betätigen.

Nur in den folgenden Fällen gehen die gespeicherten Programme verloren:

- * Wenn Sie vor der Programmierung **NEW** eingeben.
- * Wenn Sie den Computer mit dem **ALL RESET** Schalter initialisieren.
- * Wenn Sie ein neues Programm mit den gleichen Zeilennummern wie ein bereits im Speicher vorhandenes Programm eingeben.
- * Wenn Sie die Batterien auswechseln.

Diese kurze Bekanntmachung mit dem Programmieren auf dem **PC-1421** sollte Ihnen die aufregenden Möglichkeiten zeigen, die Ihr neuer Computer bietet.

KAPITEL 6 VERBINDUNG DER FINANZ- FUNKTIONEN MIT BASIC

Der PC-1421 erhält den Inhalt des Finanz-Speichers auch nach Beendigung der Finanz-Berechnungen im FIN-Modus und anschließendem Übergang in den BASIC-Modus. Der Inhalt des Finanz-Speichers kann sowohl in manuellen Berechnungen als auch in Programmen verwendet werden.

Variable, die in Finanz-Berechnungen benutzt wurden, können auch in BASIC-Programmen verwendet werden. Sie können beliebig oft einfach durch Eingabe von Daten benutzt werden.

Für BASIC verwendbare Tasten

Die Finanz-Tasten, die im FIN-Modus benutzt werden, können für den Finanz-Speicher und für die Finanz-Funktionen verwendet werden.

Modus	Anwendung	Speicher & Finanz	Speicher	Statement
Normaler Modus	Kosten, Verkaufspreise, Gewinnspanne, Kalkulationsaufschlag	CST, SEL MAR MU		
Finanz-Modus	Zinseszinssatz	n, i PV, FV PMT		BGNON
	Abschreibungs-berechnungen		PRN, INTE BAL, SPRN SINTE	AMRT ACC BGNOFF
	Kapitalaufwands-berechnungen	NPV IRR	CFi(0) ~ CFi(19) Ni(0) ~ Ni(19)	

- Speicher und Finanz: Hat sowohl die Finanz-Speicherfunktion als auch die Finanz-Funktion.
- Speicher: Hat die Finanz-Speicherfunktion.

Statements

- BGNON und BGNOFF bedeuten Anfang bzw. Ende von Zahlungsfristen.
- AMRT und ACC sind Statements, die für Finanz-Berechnungen benutzt werden. Die Daten werden mit den Finanz-Tasten oder den alphabetischen Tasten eingegeben.

1) Beginn der Zahlungsfrist (BGNON)

Kann manuell oder innerhalb eines Programms eingestellt werden.

• Manuelle Operation

Durch Drücken von BGNON **ENTER** leuchtet das Symbol "BGN" auf dem Display, das anzeigt, daß der Beginn der Zahlungsfrist eingestellt wurde.

Programmbeispiel

```
10: CLEAR
20: BGNON ← Einstellung von Beginn der Zahlungsfrist
30: INPUT n
   :
```

2) Ende der Zahlungsfrist (BGNOFF)

Diese Statement löscht die Einstellung des Beginns der Zahlungsfrist (durch BGNON eingestellt) und stellt das Ende der Zahlungsfrist ein. Dabei erlischt das Symbol "BGN" auf dem Display.

Hinweis: Der Beginn und das Ende von Zahlungsfristen können im BASIC-Modus mit **SHIFT** **BGN** eingestellt werden. Wenn **SHIFT** **BGN** gedrückt werden, erscheinen gleichzeitig das Symbol "BGN" und der Strich (—) über STAT auf dem Display. Der RUN-Modus und der PRO-Modus haben Vorrang, so daß die Berechnungen nicht beeinflußt werden, wenn der Strich über STAT vorhanden ist. Vor der Ausführung von BASIC-Programmen das STAT-Symbol durch Drücken von **SHIFT** **CA** löschen.

Finanzfunktionen

COMP (finanzielle Berechnungen), EFF (effektiver Jahreszinssatz), APR (nominaler Zinssatz), DAYS I, DAYS II (Berechnung der Anzahl Tage)

1) COMP

Diese arbeitet als Finanzfunktion, wenn nachfolgende Eingaben in den Finanz-Rechenspeicher gemacht werden, und führt Finanzberechnungen mit den erforderlichen Speichervariablen durch. Mit dieser Funktion können die Variablen CST, SEL, MAR, MU, PV, FV, PMT, n, i, NPV und IRR benutzt werden.

(1) Gebrauch von COMP als Kommando

Programmbeispiel

```
10 PV = -200: PMT = -50: i = 5.25/12: n = 3 * 12
20 COMP FV
30 PRINT FV
```

(2) Gebrauch von COMP als Funktion

Bei manuellen Berechnungen **COMP** **PV** **ENTER** drücken, um das Ergebnis anzuzeigen.

Gebrauch in einem Programm:

```
10 PV = -200 : PMT = -50 : i = 5.25/12 : n = 3 * 12
20 PRINT COMP FV
```

In diesem Programm befindet sich COMP hinter dem PRINT-Kommando, d.h. COMP wird hier als Funktion und nicht als Kommando verwendet.

(3) MAR oder MU (je nachdem, welche zuletzt benutzt wurde) können bei der Berechnung von COMP CST oder COMP SEL verwendet werden. Nach der Rückstellung kann MAR benutzt werden.

(4) COMP IRR

Für IRR kann es mehrere Antworten geben. Sind mehr als zwei Antworten vorhanden, können die Werte für IRR mit dem benachbarten Wert als Anfangswert erhalten werden. Bestimmung von IRR von einem bestimmten Anfangswert:

```
Beispiel:  CF(0) = -8700
           CF(1) = 10600      NI(0) ~ NI(5) = 1
           CF(2) = -900
           CF(3) = 74100
           CF(4) = -154400
           CF(5) = 79700
           IRR = 5,26, 23,38, 44,55
```

FIN-Modus

Nach Eingabe des Anfangswerts **SHIFT** **IRR** drücken, um die Antwort zu erhalten. Im obigen Beispiel 23 **SHIFT** **IRR** oder 44 **SHIFT** **IRR** drücken.

BASIC-Modus

Wie bei **COMP** **SHIFT** **IRR** (Anfangswert) den Anfangswert nach IRR eingeben. Selbst Gleichungen wie $A * 5/3$ können als Anfangswert benutzt werden. Im Beispiel oben **COMP** **SHIFT** **IRR** (23), **COMP** **SHIFT** **IRR** (44) eingeben.

Zinseszinsberechnung (**i , **n** , **PV** , **FV** , **PMT** , **COMP**)**

Manuelle Berechnung (RUN-Modus)

Beispiel 1: Speichern von Werten.

Eingabe

Anzeige

USING " ,#####.##"

ENTER

>

i **≡** 9.75 **÷** 2

i=9.75/2_

ENTER

4.87

C-CE

>

i **ENTER**

4.87

MDF **i** **ENTER**

4.88

(Die dritte Dezimalstelle wird gerundet)

Beispiel 2: Nach der Methode in Beispiel (1) die Daten speichern und PMT (Zahlung) bestimmen.

$$n = 15 * 2$$

$$i = 9,75/2$$

$$PV = -800000$$

$$FV = 1500000$$

Eingabe

Anzeige

COMP **PMT**

COMP PMT_

ENTER

-179.360.07

C-CE

>

MDF **PMT**

MDF PMT_

ENTER

-179.360.08

(Die dritte Dezimalstelle wird gerundet)

Verbindung der Finanz-Funktionen mit BASIC

Beispiel 3: Programm

PRO-Modus

```
10 ERASE FIN
20 BGNON
30 USING ",#####.##"
40 INPUT "n=" ; n
50 INPUT "i=" ; i
60 INPUT "PV=" ; PV
70 INPUT "FV=" ; FV
80 PRINT MDF COMP PMT
90 END
```

RUN-Modus

Eingabe

Anzeige

RUN

n =

15 x 2

n = 15 * 2

i =

9,75 ÷ 2

PV =

-800000

FV =

15000000

-171,022.72

2) Kosten, Verkaufspreis und Gewinnspanne (Kalkulationsaufschlag)

Berechnet werden CST (Kosten), SEL (Verkaufspreis), MAR (Gewinnspanne) und MU (Kalkulationsaufschlag).

Beispiel 1: Berechnung der Gewinnspanne

RUN-Modus

Eingabe

Anzeige

USING ",####.##"

>

150

CST = 150_

ENTER

150.00

SEL [=] 600 ENTER

600.00

COMP MAR

COMP MAR

ENTER

75.00

MDF MAR ENTER

75.00

Beispiel 2: Programm

PRO-Modus

```

10 ERASE FIN
20 USING ",####.##"
30 INPUT "CST ="; CST
40 INPUT "SEL ="; SEL
50 PRINT MDF COMP MAR
60 END
    
```

RUN-Modus

<u>Eingabe</u>	<u>Anzeige</u>
RUN ENTER	CST =
150 ENTER	SEL =
600 ENTER	75.00

3) Nominaler und effektiver Jahreszinssatz

Jahreszinssatz (APR, EFF)

<u>Eingabe</u>		<u>Anzeige</u>
SHIFT [+APR]	→	APR (Nominaler Jahreszinssatz)
SHIFT [-EFF]	→	EFF (Effektiver Jahreszinssatz)

Durch Eingabe von APR (N, EFF) wird in den nominalen Jahreszinssatz umgerechnet.

Durch Eingabe von EFF (N, APR) wird in den effektiven Jahreszinssatz umgerechnet.

(N: Anzahl der Perioden/Jahr)

Verbindung der Finanz-Funktionen mit BASIC

Beispiel 1: Berechnung des effektiven Jahreszinssatzes, wenn die Zinsen 4 mal jährlich gezahlt werden und der nominale Jahreszinssatz 5,5% beträgt.

Eingabe

USING "###.##" **ENTER**

MDF **SHIFT** **EFF**

(**4** , 5.5)

ENTER

Anzeige

MDF EFF (4, 5.5)

5.61

Beispiel 2: Berechnung des nominalen Jahreszinssatzes.

Programmbeispiel

PRO-Modus

10 USING "#####.##"

20 INPUT "N="; N

30 INPUT "EFF="; E

40 PRINT MDF APR (N, E)

50 END

RUN-Modus

Eingabe

RUN **ENTER**

4 **ENTER**

5.35 **ENTER**

Anzeige

N =

EFF =

5.25 (Ergebnis)

4) Berechnung der Anzahl Tage (DAYSI, DAYSII)

Kann nur durch die Funktionstasten eingegeben werden.

Den Anfangstag und den Endtag eingeben, um die Anzahl Tage zu berechnen.

DAYSI (Berechnet die Anzahl Tage mit 360 Tage/Jahr)

DAYSII (Berechnet die Anzahl Tage mit 365 Tage/Jahr)

Beispiel 1: Manuelle Berechnung (RUN-Modus)

Berechnung der Anzahl Tage zwischen 1. Februar 1980 und 1. Februar 1985 mit 360 Tage/Jahr (DAYSI).

SHIFT **I(360)**

(**2.011980** , **2.011985**)

ENTER

Ergebnis: 1800

Beispiel 2: Programm

PRO-Modus

```
10 INPUT "A="; A, "B="; B
20 C = DAYSII (A, B)
30 PRINT C
40 END
```

RUN-Modus

RUN

<u>Eingabe</u>	<u>Anzeige</u>
A =	2 [.] 011980 <input type="button" value="ENTER"/>
B =	2 [.] 011985 <input type="button" value="ENTER"/>

Ergebnis: 1827

5) Abschreibungs-Berechnung

Geben Sie n (Anzahl der Zeitperioden), i (Zinssatz) und PV (gegenwärtiger Wert einer Investition) ein, um PMT (Zahlung) zu erhalten. Dann können der Zinsanteil (INTE) und der Kapitalanteil (PRN) der Gesamtzahlung und BAL (Bestand oder Restkapital) erhalten werden.

(1) AMRT-Gleichung (Gleichung: spezifiziert die Häufigkeit)

PRN, INTE und BAL werden nach Ausführung der AMRT-Gleichung separat gespeichert.

Beispiel 1: Manuelle Berechnung (RUN-Modus)

```
PV = -70000
i = 12,5 ÷ 12
n = 25 x 12
```

Nachdem PMT mit den obigen Daten berechnet wurde, können die 150. PRN, INTE und BAL auf folgende Weise erhalten werden:

<u>Eingabe</u>	<u>Anzeige</u>
USING ",#####.##" <input type="button" value="ENTER"/>	
<input type="button" value="AMRT"/> 150	AMRT 150_
<input type="button" value="ENTER"/>	
MDF PRN	MDF PRN

ENTER	159.62
MDF INTE ENTER	603.63
MDF BAL ENTER	-57,788.56

Beispiel 2: Programm

PRO-Modus

```

10 ERASE FIN
20 USING ",#####.##"
30 INPUT "PV="; PV
40 INPUT "i="; i
50 INPUT "n="; n
60 COMP PMT
70 INPUT "TIMES="; X
80 AMRT X
90 PRINT MDF PRN
100 PRINT MDF INTE
110 PRINT MDF BAL
120 END
    
```

RUN-Modus

Eingabe

Anzeige

RUN ENTER

-70000

PV = -70000_

ENTER

i =

12.5 ÷ 12 ENTER

n =

25 x 12 ENTER

TIMES =

150 ENTER

159.62

ENTER

603.63

ENTER

-57,788.56

(2) ACC Gleichung 1, Gleichung 2 (Gleichung 1: Startperiode, Gleichung 2: Endperiode)

Die Werte für SPRN und SINTE werden berechnet und separat gespeichert, wenn ACC Gleichung 1, Gleichung 2 ausgeführt wird.

SPRN (akkumuliertes Kapital der Gesamt-Rückzahlung einer beliebigen Periode)

SINTE (akkumulierte Zinsen der Gesamt-Rückzahlung einer beliebigen Periode)

Beispiel 1:

Nach Berechnung von PMT mit den Daten $PV = 70000$, $i = 12,5/12$ und $n = 25 \times 12$ berechnen Sie das akkumulierte Kapital und den akkumulierten Zins vom 13. bis zum 24. mal.

<u>Eingabe</u>	<u>Anzeige</u>
ACC 13, 24	ACC 13, 24
ENTER	>
MDF SPRN	MDF SPRN_
ENTER	490.61
MDF SINTE ENTER	8.668.37

(Gliedern Sie die Zahlen alle 3 Ziffern und stellen 2 Dezimalstellen mit dem USING-Statement ein.)

Beispiel 2: Programm

PRO-Modus

```

10
? Gleich wie das Programm im Beispiel auf Seite 96.
60
70 INPUT "START="; X, "END="; Y
80 ACC X, Y
90 PRINT MDF SPRN
100 PRINT MDF SINTE
110 END

```

RUN-Modus

<u>Eingabe</u>	<u>Anzeige</u>
RUN ENTER	PV =
-70000 ENTER	i =
12.5 ÷ 12 ENTER	n =
25 x 12 ENTER	START =
13 ENTER	END =
24 ENTER	490.61
ENTER	8.668.37

6) Kapitalbewertungsberechnung

Aufgrund des investierten Kapitals und des Gewinns wird durch Berechnung von NPV und IRR bewertet, ob eine Investition den Gewinnerwartungen entspricht.

i : Erwartete Gewinnrate

NPV : Gegenwärtiger Netto-Wert

IRR : Interne Rückflußrate

CFi : Kapitalfluß-Speicher (20 Speicherplätze CFi(0) bis CFi(19))

Ni : Identische Kapitalflüsse (Ni(0) bis Ni(19))

(1) CFi (Gleichung 1) = Gleichung 2

(2) Ni (Gleichung 1) = Gleichung 2

Gleichung 1: gibt die Häufigkeit an

Gleichung 2: tatsächlicher Betrag

In Ni(0) bis Ni(19) können nur ganze Zahlen zwischen 1 und 99 gespeichert werden. Wenn versucht wird, einen anderen Wert zu speichern, entsteht ein ERROR 3. Die Speicherplätze 0 bis 19 sind sowohl für CFi als auch für Ni verfügbar.

Kapitalfluß-Speicher (CFi)

1 FIN-Modus

Durch die Eingabe von **SHIFT** **CA** 1000 **+/-** **CFi** 700 **CFi** 300 **CFi** ... werden Daten, beginnend mit CFi(0), der Reihe nach eingegeben. Ist der Wert von Ni nicht spezifiziert, wird für Ni automatisch 1 eingegeben. (Abb. 1)

Speicher-Nr.	CFi	Ni
0	-10000	1
1	700	1
2	300	1
3	⋮	⋮
⋮	⋮	⋮
19		

(Abb. 1)

Durch die Eingabe **SHIFT CA 10000 +/- (CFi) 300 SHIFT STO** wird 10000 in CFi(0) und 300 in CFi(8) gespeichert. In Ni(0) bis Ni(8) wird 1 und in Ni(9) bis Ni(19) 0 gespeichert.

Speicher-Nr.	CFi	Ni
0	-10000	1
1	0	1
2	0	1
⋮	⋮	⋮
7	0	1
8	300	1
9	0	0
⋮	⋮	⋮
19	0	0

← Bis in den 8. Speicherplatz werden numerische Werte gespeichert.
(Ni(8) = 1, Ni(9) = 0)

(Abb. 2)

Daher werden bis zur Speicher-Nr. der letzten 1 in Ni (bevor Ni gleich 0 wird) numerische Werte in CFi gespeichert.

2 BASIC-Modus

Wie im FIN-Modus wird, wenn 100 in CFi(10) und nichts in Ni eingegeben wird, eine 1 in Ni(10) gespeichert. (Abb. 3)

Verbindung der Finanz-Funktionen mit BASIC

Speicher-Nr.	CFi	Ni
0	0	1
⋮	⋮	⋮
9	0	1
10	100	1
11	0	0
⋮	⋮	⋮
19	0	0

(Abb. 3)

Wenn nach Eingabe von numerischen Werten in CFi(0) bis CFi(10) ein numerischer Wert in CFi(14) eingegeben wird, werden in CFi(11) bis CFi(13) Nullen (0) und in Ni(11) bis Ni(13) Einsen (1) gespeichert. (Abb. 4)

Speicher-Nr.	CFi	Ni
0	XX	XX
⋮	⋮	⋮
10	XX	XX
11	0	1
⋮	⋮	⋮
13	0	1
14	XX	XX
15	0	0
⋮	⋮	⋮
19	0	0

(Abb. 4)

Nachsehen der Werte von Ni der Reihe nach wird die Speicher-Nr. der letzten Daten, die in CFi und Ni gespeichert sind, ergeben.

Beispiele:

(1) Speichern von \$100.000 in CFi(0).

Eingabe	Anzeige
[CFi] (0) [=] 100000	CFI(0) = 100000_
[ENTER]	100000.
[C-CE]	
[CFi] (0)	CFI(0) _
[ENTER]	100000.

(2) Speichern von \$35.000 in CFi(1) und 6, der identische Kapitalfluß, in Ni(1). (Spezifizieren Sie mit USING Aufgliederung der Zahlen nach 3 Ziffern.)

Eingabe	Anzeige
[CFi] (1) = 35000	CFI(1) = 35000_
[ENTER]	35.000.
[Ni] (1) = 6	NI(1) = 6_
[ENTER]	6.
[CFi] (1) [ENTER]	35.000.
[Ni] (1) [ENTER]	6.

(3) Berechnen Sie nach Speichern der unteren Daten NPV und IRR. (Spezifizieren Sie mit USING Aufgliederung nach 3 Ziffern und 2 Dezimalstellen.)

DATEN

i = 10	CFi (0) = -100000
	CFi (1) = 7000
	CFi (2) = 8500
	CFi (3) = 9000
	CFi (4) = 120000

Verbindung der Finanz-Funktionen mit BASIC

Eingabe

Anzeige

MDF **COMP** **SHIFT** **NPV**

MDF COMP NPV_

ENTER

2,111.88

MDF **COMP** **SHIFT** **IRR**

MDF COMP IRR_

ENTER

10.64

- (4) Berechnen Sie den gegenwärtigen Nettowert für das investierte Kapital \$79.000, 13,4% Zinsen und die in der Tabelle enthaltenen Zahlungen.

Jahr	Zahlung	Jahr	Zahlung
1	\$14.000	6	9.100
2	11.000	7	9.000
3	10.000	8	9.000
4	10.000	9	4.500
5	10.000	10	100.000

- RUN-Modus Spezifizieren Sie mit USING „#####.##“

Eingabe

Anzeige

CFi (0) = -79000 **ENTER**

-79,000.00

CFi (1) = 14000 **ENTER**

14,000.00

CFi (2) = 11000 **ENTER**

11,000.00

CFi (3) = 10000 **ENTER**

10,000.00

CFi (4) = 9100 **ENTER**

9,100.00

CFi (5) = 9000 **ENTER**

9,000.00

CFi (6) = 4500 **ENTER**

4,500.00

CFi (7) = 100000 **ENTER**

100,000.00

Ni (3) = 3 **ENTER**

3.00

Ni (5) = 2 **ENTER**

2.00

i = 13.5 **ENTER**

13.50

MDF **COMP** **SHIFT** **NPV** **ENTER**

907.77

• Programm

PRO-Modus

```

10 ERASE FIN
20 USING ",#####.##"
30 FOR N=0 TO 7
40 INPUT "X="; X, "Y="; Y
50 CFI(N) = X : NI(N) = Y
60 NEXT N
70 INPUT "i="; i
80 PRINT MDF COMP NPV
90 END
    
```

RUN-Modus

Eingabe

```

RUN 
-79000 
1 
14000 
1 
11000 
1 
10000 
3 
9100 
1 
9000 
2 
4500 
    
```

Anzeige

X =
Y =
X =
Y =
X =
Y =
X =
Y =
X =
Y =
X =
Y =
X =
Y =

Verbindung der Finanz-Funktionen mit BASIC

1 **ENTER**

100000 **ENTER**

1 **ENTER**

13.5 **ENTER**

X =

Y =

i =

907.77

KAPITEL 7

VEREINFACHUNGEN

Der PC-1421 bietet mehrere Möglichkeiten, die Programmierung zu vereinfachen, indem die Anzahl der Tastenbetätigungen zur Eingabe sich wiederholenden Materials reduziert wird.

Eine dieser Möglichkeiten besteht darin, Befehle und Kommandos abzukürzen (siehe Kapitel 9).

In diesem Kapitel wird die Möglichkeit zur Reduzierung der Anzahl Tastenbetätigungen diskutiert – die Taste **DEF**.

Die DEF-Taste und benannte Programme

Es wird häufig vorkommen, daß Sie gleichzeitig mehrere Programme im PC-1421 speichern möchten. (Vergessen Sie nicht, daß die Programme unterschiedliche Zeilennummern haben müssen.) Normalerweise müssen Sie die Anfangs-Zeilenummer wissen, um ein Programm mit dem Kommando RUN oder GOTO zu starten (siehe Kapitel 9). Aber es gibt auch eine einfachere Möglichkeit! Sie können jedes Programm mit einem Buchstaben benennen und die Programmausführung mit nur zwei Tastenbetätigungen starten. Und so wird's gemacht:

Hinweis: Setzen Sie die Kennbuchstaben, auf den Sie sich später beziehen wollen, in die ersten Zeilen der Programme. Die Kennbuchstaben werden von Anführungszeichen eingeschlossen, ein Doppelpunkt wird nachgesetzt:

```
10: "A": PRINT "FIRST"
20: END
80: "B": PRINT "SECOND"
90: END
```

Jedes der folgenden Zeichen kann benutzt werden: A, S, D, F, G, H, J, K, L, , Z, X, C, V, B, N, M und SPC (Leerstelle). Dies sind die Tasten in den beiden unteren Reihen der alphabetischen Tastatur.

Hinweis: Zur Ausführung eines Programms brauchen Sie nicht RUN 80 oder GOTO 10 einzugeben, sondern nur die Taste **DEF** drücken und einen Kennbuchstaben eingeben. Im obigen Beispiel würde durch Drücken von **DEF** und danach "B" die Anzeige "SECOND" auf dem Display erscheinen.

Wenn DEF zur Ausführung eines Programms verwendet wird, werden Variable und Modus-Einstellung auf die gleiche Weise behandelt wie bei der Verwendung von GOTO. Einzelheiten finden Sie in Kapitel 9.

Schablonen

Zu Ihrem **PC-1421** gehört eine Schablone, die Sie verwenden sollten, um sich an häufig benutzte DEF-Tastenzuweisungen zu erinnern. Nach Benennung der Programme beschriften Sie die Schablone, so daß Sie leicht sehen können, welches Programm mit welcher Taste gestartet wird.

Wenn Sie beispielsweise eine Gruppe Programme haben, die häufig zusammen benutzen, benennen Sie die Programme mit Kennbuchstaben und beschriften Sie die Schablone, so daß Sie jedes der Programme durch zwei Tastenbetätigungen starten können.

Beispiel:

AVER- AGE										

KAPITEL 8 DER GEBRAUCH DES CE-126P DRUCKER/CASSETTENRECORDER INTERFACE

Der als Sonderzubehör erhältliche **CE-126P** Drucker/Cassettenrecorder-Interface erweitert den **PC-1421** um einen Drucker und ermöglicht den Anschluß eines Cassettenrecorders.

Die Merkmale des **CE-126P** sind:

- * Thermaldrucker mit 24 Zeichen Schreibbreite.
- * Handliche Papierzuführung und Abreißvorrichtung.
- * Auf Wunsch simultaner Ausdruck von Berechnungen.
(Nicht im CAL-Modus.)
- * Einfache Steuerung der Anzeige und der Druckerausgabe in BASIC.
- * Eingebautes Cassettenrecorder-Interface mit Fernbedienungsfunktion.
- * Manuelle und programmgesteuerte Kontrolle des Cassettenrecorders zum Speichern von Programmen und Daten.
- * Netzunabhängig durch Batteriebetrieb.

Für den Anschluß des **PC-1421** an den **CE-126P** schauen Sie bitte in die mit dem **CE-126P** mitgelieferte Bedienungsanleitung.

Der Gebrauch des Druckers

Bei manuellen Berechnungen mit dem **PC-1421** können Sie gleichzeitig die Berechnungen auf dem **CE-126P** ausdrucken.

ACHTUNG:

Die Ergebnisse, die mit der direkten Berechnungs-Funktion bei manuellen Berechnungen erhalten werden, können nicht gedruckt werden. Auch die Berechnungen im CAL-Modus können nicht ausgedruckt werden.

Zum Ausdrucken drücken Sie die Taste **SHIFT** und danach die Taste **ENTER** (P ↔ NP) im RUN-Modus.

Unten rechts auf dem Display erscheint über "PRINT" das Strich-Symbol. Wenn Sie danach am Ende der Berechnung **ENTER** drücken, wird der Inhalt des Displays auf eine Zeile gedruckt und das Ergebnis auf die nächste, z.B.:

Eingabe

300/50 **ENTER**

Ausdruck

300/50

6.

Sie können mit dem Statement LPRINT aus BASIC-Programmen auf den Drucker ausgeben (für Einzelheiten siehe Kapitel 9). LPRINT wird auf die gleiche Weise wie PRINT verwendet. Der Unterschied ist, daß bei Ausgabe mit PRINT auf des Display es keine Möglichkeit gibt, die restlichen Zeichen zu sehen, wenn die Ausgabe länger als 16 Zeichen ist. Mit dem Befehl LPRINT werden die folgenden Zeichen auf eine zweite und gegebenenfalls dritte Zeile gedruckt.

Programme, die für die Ausgabe PRINT enthalten, können durch Einfügung des Statements PRINT = LPRINT für Ausgabe auf den Drucker geändert werden (für Einzelheiten siehe Kapitel 9). Alle PRINT-Statements, die auf dieses Statement folgen, werden sich dann wie LPRINT-Statements verhalten. Durch PRINT = PRINT wird dieser Änderung wieder in den normalen Zustand zurückversetzt. Diese Anweisung kann in einem Programm auch in einem IF-Statement enthalten sein, so daß die Art der Ausgabe während der Ausführung des Programms gewählt werden kann.

Sie können Ihre Programme mit dem Kommando LLIST vom Drucker auflisten lassen (für Einzelheiten siehe Kapitel 9). Wird LLIST ohne Zeilennummern eingegeben, werden alle gegenwärtig im Speicher vorhandenen Programme in der Reihenfolge der Zeilennummern ausgedruckt. LLIST kann auch mit einer Anfangs- und Endzeilennummer eingegeben werden, so daß nur dieser Zeilenbereich ausgedruckt wird. Sind Programmzeilen länger als 24 Zeichen, werden sie auf zwei oder mehr Zeilen ausgedruckt. Die zweite und folgende Zeilen werden vier oder sechs Zeilen eingerückt, so daß die Zeilennummer jede einzelne Programmzeile deutlich erkennen läßt. (Zeilennummer 1 bis 999: vier Zeichen Einrückung, über 999: sechs Zeichen Einrückung)

Achtung:

- Sollte aufgrund von Papierstau ein Fehler auftreten (ERROR 8), reißen Sie das Papier ab und ziehen den restlichen Teil des Papiers ganz aus den Drucker. Drücken Sie danach die Taste **C-CE**, um den Fehlerzustand aufzuheben.
- Wird der Drucker starken externen elektrischen Störungen ausgesetzt, kann es vorkommen, daß er willkürlich Zahlen ausdruckt. Stoppen Sie das Drucken in einem solchen Fall durch Drücken der Taste **BRK**. Schalten Sie den **CE-126P** einmal aus und wieder ein und drücken dann die Taste **C-CE**. Durch Drücken von **C-CE** wird der Drucker wieder in den normalen Zustand zurückversetzt.

Wenn während des Ausdrucks ein Papierstau auftritt oder der Drucker starken elektrischen Störungen ausgesetzt ist, arbeitet der Drucker eventuell nicht normal, sondern lediglich das Symbol "BUSY" wird angezeigt. Wenn dies vorkommt, unterbrechen Sie den Druckvorgang durch Betätigung der Taste **BRK**. (Beseitigen Sie gegebenenfalls den Papierstau.) Schalten Sie den **CE-126P** aus und wieder ein und drücken die Taste **C-CE**.

- Schalten Sie den Drucker bei Nichtgebrauch immer aus, um Batterieleistung zu sparen.

Der Gebrauch des Cassettenrecorder-Interface

Mithilfe dieses Cassettenrecorder-Interface können Sie Programme und Daten aus dem Speicher des Computers auf Cassettenband überspielen. (Natürlich benötigen Sie dafür auch einen Cassettenrecorder. Wir bieten speziell für dieses Computersystem einen Cassettenrecorder an.) Und die auf Band gespeicherten Daten und Programme können Sie bei Bedarf wieder in den Computer laden.

Anschluß des CE-126P an einen Cassettenrecorder

Nur drei Anschlüsse sind erforderlich:

1. Den roten Stecker an die Mikrofon-Buchse (MIC) des Cassettenrecorders anschließen.
2. Den grauen Stecker an die Ohrhörer-Buchse (BAR) des Cassettenrecorders anschließen.
3. Den schwarzen Stecker an die Fernbedienungs-Buchse (REM) des Cassettenrecorders anschließen.

Cassettenrecorder

Im folgenden geben wir die minimalen Spezifikationen, die für Cassettenrecorder für den einwandfreien Betrieb mit dem Interface **CE-126P** erforderlich sind.

Punkt	Erfordernisse
1. Recorder-Typ	Jedes Spulen-Tonbandgerät, jeder Cassetten- oder Mikrocassetten-Recorder kann verwendet werden, wenn die nachstehenden erfüllt sind.
2. Eingangsbuchse	Der Recorder sollte einen Mikrofon-Eingang (MIC) als Minibuchse aufweisen. Niemals die AUX-Buchse verwenden.
3. Eingangsimpedanz	Der Eingang sollte ein niederohmiger Eingang sein (200 bis 1000 Ohm).
4. Minimaler Eingangspegel	Unter 3 mV bzw. -50 dB
5. Ausgangsbuchse	Sollte eine Minibuchse mit der Bezeichnung "EXT" (Zusatzlautsprecher), "MINITOR", "EAR" (Ohrhörer) oder dergleichen sein.
6. Ausgangsimpedanz	Sollte unter 10 Ohm sein.
7. Ausgangspegel	Sollte über 1 V sein (praktischer Maximalausgang über 100 mW).
8. Verzerrungen	Sollte im Bereich zwischen 2 und 4 kHz innerhalb von 15% liegen.
9. Gleichlaufschwankungen	0,3% Maximum (WRMS)
10. Sonstiges	Der Recordermotor sollte keine Drehzahl-schwankungen aufweisen.

* Sollten die Ministecker des **CE-126P** nicht mit den Eingangs/Ausgangsbuchsen Ihres Cassettenrecorders übereinstimmen, erhalten Sie im Fachgeschäft spezielle Zwischenstecker.

HINWEIS:

- Bestimmte Cassettenrecorder lassen sich aufgrund unterschiedlicher technischer Daten nicht anschließen. Cassettenrecorder, die nach langjähriger Verwendung Verzerrungen, Zunahme der Störungen und Leistungsabnahme aufweisen, führen eventuell nicht zu zufriedenstellenden Ergebnissen.

Bedienung des Interface und des Cassettenrecorders

Aufnahme (Sichern) auf Cassette

Siehe die Hinweise zu Cassetten

1. Den REMOTE-Schalter des CE-126P ausschalten.
2. Ein Programm oder Daten in Den Computer eingeben.
3. Eine Cassette in den Cassettenrecorder einlegen.
Das Band zu der Stelle spulen, wo das Programm aufgenommen werden soll.
 - Wird eine neue Cassette verwendet, das Band bis zum Anfang des Magnetbandes vorspulen (das Vorlaufband darf sich nicht vor dem Tonkopf befinden).
 - Ist die Cassette bereits teilweise bespielt, einen leeren Bandabschnitt suchen.
4. Den roten Stecker des Interface an die MIC-Buchse des Cassettenrecorders und den schwarzen Stecker an die REM-Buchse anschließen.
5. Den REMOTE-Schalter einschalten.
6. Die Aufnahme- und Wiedergabetaste des Cassettenrecorders gleichzeitig drücken (auf Aufnahme schalten).
7. Die Aufnahmeanweisungen (Statements CSAVE, PRINT#) in den Computer eingeben und die Taste **ENTER** drücken.

Zuerst den Computer auf den RUN- oder PRO-Modus stellen. Dann die folgenden Tasten betätigen: **C S A V E** **SHIFT** **"** dateiname **SHIFT** **"** **ENTER** .

(Zum Überspielen des Inhalts des Datenspeichers auf Band die folgenden Tasten drücken: **P R I N T** **SHIFT** **#** **ENTER** .)

Beispiel: **C S A V E** **SHIFT** **"** **A A** **SHIFT** **"** **ENTER**

Nach Drücken der Taste **ENTER** fängt das Band an zu laufen. Zuerst wird ein Nicht-Signal-Abschnitt von etwa 8 Sekunden hergestellt. (Piepston wird aufgenommen.) Danach werden der Dateiname und der Dateiinhalt aufgenommen.

8. Nach Beendigung der Aufnahme wird das Aufforderungszeichen (>) angezeigt, und der Cassettenrecorder stoppt automatisch. Jetzt befindet sich das Programm auf Band (es ist auch noch im Speicher des Computers vorhanden).
Wenn Daten automatisch während der Programmausführung aufgenommen werden sollen (Statement PRINT#, nicht im manuellen Betrieb), müssen vor Starten des Programms die Schritte 1 bis 6 ausgeführt werden.
Benutzen Sie das Bandzählwerk des Cassettenrecorders, um die auf Band aufgenommenen Programme leichter finden zu können.

Vergleichen des Computer-Speicherinhalts mit der Bandaufnahme

Siehe die Hinweise zu Cassetten.

Nach Überspielen eines Programms auf Cassetten oder Laden von Cassette in den Computer können Sie überprüfen, daß das Programm im Computer und das auf Cassette identisch sind (auf diese Weise können Sie sichergehen, daß sich keine Fehler bei der Übertragung eingeschlichen haben).

1. Den REMOTE-Schalter ausschalten.
2. Die Cassette in den Recorder einlegen und das Band auf die Position unmittelbar vor der zu prüfenden Datei spulen.
3. Den grauen Stecker an die EAR-Buchse und den schwarzen Stecker an die REM-Buchse anschließen.
4. Den REMOTE-Schalter einschalten.
5. Die PLAY-Taste des Recorders drücken.
6. Das Statement CLOAD? eingeben und die Ausführung mit der Taste **ENTER** starten. Dies wird wie folgt gemacht: Den Computer in den RUN- oder PRO-Modus stellen. Die folgenden Tasten betätigen:

C L O A D SHIFT ? SHIFT " A A SHIFT " ENTER

Dann sucht der Computer automatisch nach dem entsprechenden Dateinamen und vergleicht den Inhalt auf Band mit dem Speicherinhalt.

Während des Vergleichs wird ganz rechts auf dem Display das Symbol "*" angezeigt. Nach Beendigung des Vergleichs verschwindet das "*". Während der Suche nach einem Dateinamen wird das Symbol "*" nicht angezeigt, weil der Vergleich noch nicht gestartet hat.

(Das gleiche tritt auf, wenn das erste Programm ohne Dateinamen gelesen wird.)

Wenn die Überprüfung ergibt, daß die Programm identisch sind, wird auf dem Display des Computers das Aufforderungszeichen (>) angezeigt.

Sollten die Programme Unterschiede aufweisen, wird die Ausführung unterbrochen und der Fehler ERROR 8 angezeigt. Wenn dies vorkommt, versuchen Sie zunächst, den Vergleich noch einmal durchzuführen.

Laden von einer Cassette

Siehe die Hinweise zu Cassetten.

Zum Laden oder Überspielen von Programmen und Daten von der Cassette in den Speicher des Computers führen Sie die folgende Bedienung durch.

1. Den REMOTE-Schalter ausschalten.

2. Die Cassette in den Recorder einlegen. Das Band bis unmittelbar vor der zu ladenden Datei spulen.
3. Den grauen Stecker an die EAR-Buchse des Cassettenrecorders und den schwarzen Stecker an die REM-Buchse anschließen.
(Sollte der Cassettenrecorder keine REM-Buchse aufweisen, drücken Sie die PAUSE-Taste, um den Bandlauf vorübergehend zu stoppen.)
4. Den REMOTE-Schalter einschalten.
5. Die PLAY-Taste des Cassettenrecorders drücken (auf Wiedergabe schalten).

Den Lautstärkeregler auf die Mitte oder Maximum einstellen.

Den Klangregler auf maximale Höhen einstellen.

6. Die Überspieldaten (Statements CLOAD, INPUT #) in den Computer eingeben und die Taste **ENTER** drücken.

Zuerst den Computer auf den RUN-Modus stellen. Dann die folgenden Tasten betätigen: **C** **L** **O** **A** **D** **SHIFT** **"** dateiname **SHIFT** **"** **ENTER** . (Zum Laden des Inhalts des Datenspeichers die folgenden Tasten drücken: **I** **N** **P** **U** **T** **SHIFT** **#** **ENTER** .)

Beispiel: **C** **L** **O** **A** **D** **SHIFT** **"** **A** **A** **SHIFT** **"** **ENTER**

Nach dem spezifizierten Dateinamen wird automatisch gesucht, und der Inhalt wird in den Speicher des Computers geladen.

Beim Laden des vorher mit CSAVE gesicherten Programms vom Band in den Speicher des Computers wird das Symbol " * " angezeigt.

(Das gleiche tritt auf, wenn das erste Programm ohne Dateinamen gelesen wird.)

Das Symbol " * " verschwindet, wenn der Ladevorgang beendet ist.

7. Nach dem Laden wird das Aufforderungszeichen (>) angezeigt, und der Cassettenrecorder stoppt automatisch.

Zum Überspielen von Daten (Statement INPUT #) innerhalb eines Programms vor Ausführung des Programms die Schritte 1 bis 5 durchführen.

Hinweise:

- Sollte ein Fehler auftreten (ERROR 8), beginnen Sie noch einmal von vorn. Tritt der Fehler wieder auf, verringern oder erhöhen Sie die Lautstärke.
- Wenn kein Fehler angezeigt wird, das Band aber immer weiterläuft (während der Computer "BUSY" anzeigt), ist die Übertragung nicht in Ordnung. Drücken Sie die Taste **ON** **BRK** zum Stoppen des Bandes und wiederholen Sie die Bedienung.
- Wenn der Fehler immer wieder angezeigt wird oder das Band immer weiterläuft, reinigen und entmagnetisieren Sie den Tonkopf des Recorders.

ACHTUNG:

Nach Überspielen eines Programms auf Band sollte der Inhalt immer mit dem auf Seite 112 beschriebenen Verfahren überprüft werden. Wenn als Ergebnis von wiederholten Überprüfungen ein Fehler auftritt, stellen Sie den Computer in den RUN-Modus und versuchen Sie die folgende Berechnung.

CLEAR
 (3454 – MEM)/120
 und (3455 – MEM)/120

Sollte das Ergebnis einer dieser Berechnungen eine ganze Zahl sein (1, 2, 3, ..., 29), reduzieren Sie das Programm um eine Zeile (kombinieren Sie zwei Zeilen zu einer) oder fügen Sie eine Zeile hinzu (ein Zeile in zwei aufteilen) und speichern es dann noch einmal.

(Beispiel)	10 : A = 0, B = 0	→	10: A = 0, B = 0
	200 : NEXT I		200: NEXT I: END
	210 : END		

Überprüfen Sie danach den gespeicherten Inhalt noch einmal.

Sollte der Fehler immer noch auftreten, nehmen Sie das Programm auf einem anderen Bandabschnitt auf oder verwenden eine andere Cassette. Wenn dies nichts hilft, ist die Ursache beim Cassettenrecorder zu suchen.

Hinweise zu Cassetten

- 1) Verwenden Sie für Zurückladen in den Computer und für Vergleichen immer den Cassettenrecorder, der für die Aufnahme verwendet wurde. Wird ein anderer Cassettenrecorder verwendet, ist Laden oder Vergleichen eventuell nicht möglich.
- 2) Verwenden Sie zum Aufnehmen von Programmen und Daten immer Cassetten höchster Qualität (billigere Cassetten haben eventuell nicht die erforderlichen Eigenschaften für digitale Aufnahme).
- 3) Halten Sie die Tonköpfe und die Bandführungen sauber – lassen Sie vor Aufnehmen, Vergleichen oder Laden eine Reinigungscassette durchlaufen.
- 4) Lautstärke auf Mitte oder Maximum einstellen.
Die Lautstärkeeinstellung kann beim Laden von Band in den Computer sehr wichtig sein, nur geringe Änderungen können den Unterschied zwischen Fehler und fehlerfreiem Betrieb bewirken.
- 5) Überprüfen Sie immer die Verbindungen zwischen Computer und Cassettenrecorder-Interface und die Verbindungen zwischen Interface und Cassettenrecorder.
- 6) Sollten beim Netzbetrieb des **CE-126P** und/oder des Recorders Probleme auftreten, verwenden Sie Batteriebetrieb (manchmal wird das Signal durch "Brummen" aus dem Wechselstromnetz verändert, wodurch die digitale Aufnahme gestört wird).
 - Zum Anschließen des Netzteils an den **CE-126P**, zuerst den **CE-126P** ausschalten und danach das Netzteil anschließen.
- 7) Klangregler auf maximale Höhen stellen.
- 8) Sollen Programme oder Daten auf bereits benutzte Cassetten aufgenommen werden, muß die Cassette zuerst vollkommen gelöscht werden, bevor die Aufnahme ausgeführt wird. (Auf der Cassette dürfen keine Teile der vorigen Aufnahme verbleiben.)

KAPITEL 9

BASIC-REFERENZTEIL

Dieses Kapitel ist in drei Abschnitte aufgeteilt:

- Kommandos:** Anweisungen, die außerhalb von Programmen zum Ändern der Bearbeitungsbedingungen, Durchführen von Dienstprogrammen und zur Steuerung von Programmen verwendet werden.
- Befehle:** Befehlswörter, die beim Programmieren für die BASIC-Statements verwendet werden.
- Funktionen:** Spezielle Operatoren, die in BASIC-Programmen zur Umwandlung von Variablen verwendet werden.

In diesem Kapitel sind die Kommandos und Befehle alphabetisch angeordnet. Für jedes Kommando bzw. jeden Befehl ist eine eigene Seite vorhanden, um das Nachschlagen zu erleichtern. Unten ist der Inhalt jedes der drei Abschnitte angegeben, so daß Sie schnell die Kategorie, zu der ein Operator gehört, finden können. Die Funktionen sind nach drei Kategorien gruppiert und innerhalb der Kategorien alphabetisch angeordnet.

Kommandos

Programmkontrolle

CONT
GOTO*
NEW
RUN

Cassettenrecorderkontrolle

CLOAD
CLOAD?
CSAVE
INPUT #*
PRINT #*

Fehlersuche

LIST
LLIST
TROFF*
TRON*

Variablenkontrolle

CLEAR
DIM*

Winkelmodus

DEGREE*
GRAD*
RADIAN*

Sonstiges

BEEP*
PASS
RANDOM*
USING*
WAIT*

* Diese Kommandos sind gleichzeitig BASIC-Befehle. Ihre Wirkung als Kommando und als Befehl ist gleich, daher werden sie nicht in diesem Kommando-Abschnitt beschrieben, sondern im Befehl-Abschnitt.



Kontrolle und Verzweigung

2END
 FOR ... TO ... STEP
 GOSUB
 GOTO
 IF ... THEN
 NEXT
 ON ... GOSUB
 ON ... GOTO
 RETURN
 STOP

Zuweisung und Vereinbarung

CLEAR
 DIM
 ERASE
 LET

Eingabe und Ausgabe

AREAD
 CSAVE
 DATA
 INPUT
 INPUT #
 LPRINT
 PAUSE
 PRINT
 PRINT #
 READ
 RESTORE
 USING
 WAIT

Sonstiges

BEEP
 DEGREE
 GRAD
 RADIAN
 RANDOM
 REM
 TROFF
 TRON

FunktionenPseudovariablen

INKEY\$
MEM
PI

String-Funktionen

ASC
CHR\$
LEFT\$
LEN
MID\$
RIGHT\$
STR\$
VAL

Numerische Funktionen

ABS
ACS
AHC
AHS
AHT
ASN
ATN
COS
CUR
DEG
DMS
EXP
FAC
HCS
HSN
HTN
INT
LN
LOG
MDF
POL
RCP
REC
RND
ROT
SGN
SIN
SQR
SQU
TAN
TEN

KOMMANDOS

- 1 CLOAD
- 2 CLOAD "dateiname"

Abkürzungen: CLO., CLOA.

Siehe auch: CLOAD?, PASS

Wirkung

Das Kommando CLOAD dient zum Laden eines Programms von Cassette in den Computer. Es kann nur in Verbindung mit dem Drucker/Cassettenrecorder-Interface CE-126P (Sonderzubehör) und einem Cassettenrecorder verwendet werden.

Anwendung

Die erste Form des Kommandos CLOAD löscht alle im Speicher des Computers vorhandene Programme und löscht das erste Programm, das sich hinter der gegenwärtigen Bandposition auf der Cassette befindetet.

Die zweite Form des Kommandos CLOAD löscht den Speicher, sucht auf der Cassette das Programm mit dem Namen "dateiname" und lädt dieses Programm in den Computer.

Die auf Cassette gespeicherten Programme werden geladen, wenn der PC-1421 im PRO- oder im RUN-modus ist.

Beispiele

- CLOAD Lädt das erste Programm auf der Cassette.
CLOAD "PRO3" Sucht das Programm mit dem Namen PRO3 und lädt es.

Hinweise:

1. Wie die angegebene Datei nicht gefunden, sucht der Computer weiter nach dem Dateinamen, auch wenn das Band zu Ende ist. Stoppen Sie in diesem Fall den Suchvorgang durch Drücken der Taste ^{ON}_{BRK}. Dieser Hinweis gilt auch für die später beschriebenen Kommandos CLOAD? und INPUT #.
2. Tritt während der Ausführung des Kommandos CLOAD ein Fehler auf (wird später beschrieben), so ist das im Computer gespeicherte Programm nicht intakt.

Kommandos CLOAD?

1 CLOAD?

2 CLOAD? "dateiname"

Abkürzungen: CLO.?, CLOA.?

Siehe auch: CLOAD, PASS

Wirkung

Das Kommando CLOAD? dient dazu, ein auf Cassette aufgenommenes Programm mit einem Programm im Speicher des Computers zu vergleichen. Dieses Kommando kann nur zusammen mit dem Drucker/Cassettenrecorder-Interface CE-126P (Sonderzubehör) und einem Cassettenrecorder verwendet werden.

Anwendung

Die erste Form des Kommandos CLOAD? vergleicht das Programm im Speicher des Computers mit dem ersten Programm auf der Cassette nach der gegenwärtigen Bandposition.

Die zweite Form des Kommandos CLOAD? sucht auf der Cassette nach dem Programm mit den Namen "dateiname" und vergleicht es dann mit dem Programm im Speicher des Computers.

Beispiele

CLOAD? Vergleicht das erste Programm auf der Cassette mit dem Programm im Computer.

CLOAD? "PRO3" Sucht auf dem Band nach dem Programm PRO3 und vergleicht es dann mit dem Programm im Computer.

1 CONT

Abkürzungen: C., CO., CON.

Siehe auch: RUN, Befehl STOP

Wirkung

Mit dem Kommando CONT wird ein Programm fortgesetzt, das unterbrochen wurde.

Anwendung

Wird ein Programm mit dem Programm-Befehl STOP unterbrochen, kann es mit dem Kommando CONT fortgesetzt werden.

Wird ein Programm durch Betätigung der Taste **BRK** unterbrochen, kann es mit dem Kommando CONT fortgesetzt werden.

Beispiele

CONT Fortsetzung eines unterbrochenen Programms.

- 1 CSAVE
- 2 CSAVE "dateiname"
- 3 CSAVE "passwort"
- 4 CSAVE "dateiname", "passwort"

Abkürzungen: CS., CSA., CSAV.

Siehe auch: CLOAD, CLOAD?, PASS

Wirkung

Das Kommando CSAVE dient zum Sichern (Aufnehmen) eines Programms auf Cassette. Es kann nur mit dem Drucker/Cassettenrecorder-Interface **CE-126P** (Sonderzubehör) verwendet werden.

Anwendung

Die erste Form des Kommandos CSAVE schreibt alle Programme, die sich im Speicher des Computers befinden, ohne Spezifizierung eines Dateinamens auf die Cassette.

Die zweite Form des Kommandos CSAVE schreibt alle Programme im Speicher des Computers unter dem angegebenen Dateinamen auf die Cassette.

Die dritte Form des Kommandos CSAVE schreibt alle Programme im Computerspeicher ohne Zuweisung eines Dateinamens, aber mit einem Passwort auf die Cassette. Programme, die mit einem Passwort geschützt sind, können beliebig geladen werden, können aber nur dann aufgelistet oder verändert werden, wenn das richtige Passwort angegeben wird. (Siehe die Beschreibung des Kommandos PASS.)

Die vierte Form des Kommandos CSAVE schreibt alle Programme im Computerspeicher auf die Cassette und weist dabei den Dateinamen und das Passwort wie angegeben zu.

Die Programme im Speicher werden auf Cassette überspielt, wenn der **PC-1421** im PRO- oder RUN-Modus ist.

Beispiele

CSAVE "PRO3", "SECRET" Die Programme aus dem Computer werden unter dem Namen PRO3 und geschützt durch das Passwort SECRET auf Cassette gesichert.

1 GOTO ausdruck

Abkürzungen: G., GO., GOT.

Siehe auch: RUN

Wirkung

Mit dem Kommando GOTO wird die Ausführung eines Programms gestartet.

Anwendung

Das Kommando GOTO kann anstelle des Kommandos RUN benutzt werden, um die Programmausführung mit der Zeilennummer zu starten, die durch den Ausdruck spezifiziert ist.

GOTO unterscheidet sich in fünf Punkten von RUN:

- 1) Der Wert des WAIT-Intervalls wird nicht zurückgesetzt.
- 2) Das mit dem Statement USING bestimmte Anzeigeformat wird nicht gelöscht.
- 3) Variable und Felder bleiben erhalten.
- 4) Der Status PRINT = LPRINT wird nicht zurückgestellt.
- 5) Der Zeiger für READ wird nicht zurückgestellt.

Die Ausführung eines Programms mit GOTO ist identisch zur Ausführung mit der Taste **DEF**.

Beispiele

GOTO 100

Die Ausführung des Programms beginnt in Zeile 100.

Kommandos LIST

- 1 LIST
- 2 LIST ausdruck

Abkürzungen: L., LI., LIS.

Siehe auch: LLIST

Wirkung

Das Kommando LIST dient zum Anzeigen von Programmen.

Anwendung

Das Kommando LIST kann nur im PRO-Modus verwendet werden. Die erste Form des Kommandos LIST bewirkt die Anzeige des Statements mit der niedrigsten Zeilennummer.

Durch die zweite Form wird das Statement mit der Zeilennummer angezeigt, die gleich oder am wenigsten größer als der Wert des Ausdrucks ist. Mit den Aufwärts- und Abwärts-Pfeilen können die vorgehenden und nachfolgenden Programmzeilen angezeigt werden.

Beispiele

LIST 100

Die Zeile 100 wird angezeigt.

- 1 LLIST
- 2 LLIST ausdruck
- 3 LLIST ausdruck 1, ausdruck 2
- 4 LLIST ausdruck,
- 5 LLIST ausdruck

Abkürzungen: LL., LLI., LLIS.

Siehe auch: LIST

Wirkung

Das Kommando LLIST dient zum Ausdrucken eines Programms auf dem Drucker/Cassettenrecorder-Interface CE-126P (Sonderzubehör).

Anwendung

Das Kommando LLIST kann im PRO- oder im RUN-Modus verwendet werden.

Die erste Form druckt alle Programme aus, die sich im Speicher des Computers befinden.

Die zweite Form druckt nur die Zeile, deren Zeilennummer durch den Ausdruck gegeben ist.

Die dritte Form druckt die Programmzeilen zwischen den Zeilennummern aus, die mit Ausdruck 1 und Ausdruck 2 bestimmt sind. Dabei wird jeweils die Zeilennummer genommen, die gleich oder am wenigsten größer dem Ausdruck ist. Zwischen den beiden Zeilennummern müssen sich mindestens zwei Zeilen befinden.

Die vierte Form druckt das Programm aus, beginnend mit der durch den Ausdruck bestimmten Zeilennummer.

Die fünfte Form druckt alle Programme bis einschließlich der durch den Ausdruck bestimmten Zeilennummer.

Beispiele

LLIST 100, 200 Die Programmzeilen 100 bis 200 werden ausgedruckt.

Kommandos
NEW

1 NEW

Abkürzungen: keine

Wirkung

Mit dem Kommando NEW werden Programme im Speicher gelöscht.

Anwendung

Wird dieses Kommando im PRO-Modus verwendet, so werden allen Programme und Daten gelöscht, die sich im Speicher befinden.

Das Kommando NEW ist im RUN-Modus nicht definiert und bewirkt dort den Fehler ERROR 9.

Beispiele

NEW

Löscht Programme oder Daten.

1 PASS "zeichenstring"

Abkürzungen: PA., PAS.

Siehe auch: CSAVE, CLOAD

Wirkung

Mit dem Kommando PASS werden Passwörter gesetzt und wieder gelöscht.

Anwendung

Passwörter schützen Programme vor Einsehen und Verändern durch andere Personen. Ein Passwort besteht aus einem Zeichenstring mit **nicht mehr als sieben Zeichen**. Diese sieben Zeichen können Buchstaben des Alphabets sein oder aus den folgenden Symbolen bestehen: ! # \$ % & () * + - / , . : ; < = > ? @ $\sqrt{\quad}$ π ^

Sobald das Kommando PASS gegeben wurde, sind die Programme im Speicher geschützt. Ein mit Passwort geschütztes Programm kann nicht im Speicher eingesehen oder verändert werden. Es kann nicht auf Cassette ausgegeben oder mit LIST oder LLIST aufgelistet werden, auch Programmzeilen können nicht hinzugefügt oder gelöscht werden. Befinden sich mehrere Programme im Speicher, wenn das Passwort gesetzt wird, so sind alle Programme im Speicher geschützt. Die einzige Möglichkeit, diesen Schutz zu beseitigen, ist die Eingabe eines zweiten PASS-Kommandos mit dem gleichen Passwort oder die Eingabe von NEW (wodurch die Programme gelöscht werden).

Beispiele

PASS "SECRET" Setzt das Passwort SECRET für alle Programme im Speicher.

Kommandos RUN

- 1 RUN
- 2 RUN zeilennummer

Abkürzungen: R., RU.

Siehe auch: GOTO

Wirkung

Mit dem Kommando RUN wird ein Programm, das sich im Speicher befindet, gestartet.

Anwendung

Die erste Form des Kommandos RUN startet das Programm im Speicher mit der niedrigsten Zeilennummer gestartet.

Mit der zweiten Form des Kommandos RUN wird ein Programm beginnend mit der spezifizierten Zeilennummer gestartet.

RUN unterscheidet sich in fünf Punkten von GOTO:

- 1) Der Wert des WAIT-Intervalls wird zurückgestellt.
- 2) Das mit dem USING-Statement gesetzte Anzeigeformat wird gelöscht.
- 3) Variable und Felder mit Ausnahme der festen Variablen werden gelöscht.
- 4) Der Status PRINT = PRINT wird gesetzt.
- 5) Der READ-Zeiger wird auf das erste DATA-Statement zurückgesetzt.

Die Ausführung eines Programms mit GOTO ist identisch zur Ausführung mit der Taste **DEF**. In allen drei Formen der Programmausführung werden die FOR/NEXT- und GOSUB-Verschachtelungen gelöscht.

Beispiele

RUN 100 Startet das Programm, das mit Zeile 100 beginnt.

B E F E H L E

1 AREAD variablenname

Abkürzungen: A., AR., ARE., AREA.

Siehe auch: Befehl INPUT und Beschreibung der Taste DEF in Kapitel 7

Wirkung

Der Befehl AREAD dient zum Einlesen eines einzigen Werts in ein Programm, das mit der Taste **DEF** gestartet wurde.

Anwendung

Wenn ein Programm ein "Label" hat (mit einem Buchstaben benannt ist), so daß es mit der Taste **DEF** gestartet werden kann, kann mit dem Befehl AREAD ein einziger Startwert eingegeben werden, ohne daß der Befehl INPUT benutzt wird. Der Befehl AREAD muß sich in der ersten Programmzeile nach dem Label befinden. Wenn der Befehl woanders im Programm auftaucht, wird er ignoriert. Eine numerische oder eine String-Variable kann benutzt werden, jedoch kann pro Programm nur eine dieser Variablenarten benutzt werden.

Zum Verwenden des Befehls AREAD geben Sie den gewünschten Wert im RUN-Modus ein, drücken die Taste **DEF** und danach den Buchstaben, der das Programm bezeichnet. Wird eine String-Variable verwendet, ist es nicht erforderlich, den eingegebenen String in Anführungszeichen zu setzen.

Beispiele

```
10 "X": AREAD N
20 PRINT N^2
30 END
```

Durch Eingabe von "7 **DEF** X" erscheint die Anzeige "49".

Hinweise:

1. Wenn am Beginn der Programmausführung das Aufforderungszeichen (">") angezeigt wird, ist die zugewiesene Variable gelöscht.
2. Wird am Beginn der Programmausführung der Inhalt mit dem Befehl PRINT angezeigt, wird das folgende gespeichert:

Befehle
AREAD

- Wird auf dem Display PRINT numerischer ausdruck, numerischer Ausdruck oder PRINT "string", "string" angezeigt, wird der Inhalt auf der rechten Seite des Displays gespeichert.

Beispiel: Wenn das Programm unten ausgeführt wird:
10 "A": PRINT "ABC", "DEFG"
20 "S": AREAD A\$: PRINT A\$
RUN-Modus
DEF A → ABC DEFG
DEF S → DEFG

- Bei der Anzeige PRINT numerischer ausdruck; numerischer ausdruck; numerischer ausdruck... auf dem Display wird der zuerst angezeigte Inhalt (ganz links) gespeichert.
- Wird auf dem Display PRINT "string"; "string"; "string"... angezeigt, wird der zuletzt zugewiesene "string" gespeichert.

1 BEEP ausdruck

Abkürzungen: B., BE., BEE.

Wirkung

Der Befehl BEEP erzeugt einen Ton.

Anwendung

Der Befehl BEEP bewirkt, daß der **PC-1421** einen oder mehrere Töne von 4 kHz ausgibt. Die Anzahl der Töne wird durch den Ausdruck bestimmt, der ein numerischer Ausdruck sein muß (positive Zahl kleiner als 9,999999999E+99). Der Ausdruck wird ausgerechnet, aber nur der ganzzahlige Anteil wird zur Bestimmung der Anzahl der Töne benutzt.

BEEP kann auch als Kommando mit numerischen Literalen und vordefinierten Variablen benutzt werden. In diesem Fall ertönt der Ton unmittelbar nach Drücken der Taste **ENTER**

Beispiele

- | | |
|----------------------|--|
| 10 A = 5 : B\$ = "9" | |
| 20 BEEP 3 | Erzeugt 3 Töne. |
| 30 BEEP A | Erzeugt 5 Töne. |
| 40 BEEP (A+4)/2 | Erzeugt 4 Töne. |
| 50 BEEP B\$ | Dies ist unzulässig und bewirkt die Fehlermeldung ERROR 9. |
| 60 BEEP -4 | Erzeugt keine Töne, bewirkt aber auch keine Fehlermeldung. |

1 CLEAR

Abkürzungen: CL., CLE., CLEA.

Siehe auch: DIM

Wirkung

Mit dem Befehl CLEAR werden alle im Programm verwendeten Variablen gelöscht und alle vor-zugeordneten auf Null oder NUL zurückgesetzt.

Anwendung

Der Befehl CLEAR gewinnt Speicherplatz zurück, der für die Speicherung von Variablen benutzt wurde. Dies ist hilfreich, wenn beispielsweise die Variablen im ersten Teil des Programms im zweiten Teil nicht benötigt werden und der Speicherplatz begrenzt ist. CLEAR kann auch am Anfang eines Programms benutzt werden, wenn mehrere Programme im Speicher vorhanden sind und der Speicherplatz freigemacht werden soll, der durch die vorigen Programme belegt wurde.

CLEAR setzt den Speicherplatz, der durch die Variablen A – Z, A\$ – Z\$ oder A(1) – A(26) (ohne DIM-Vereinbarung) belegt ist, nicht frei, da dieser Speicherplatz fest zugewiesen ist (siehe Kapitel 4). CLEAR stellt numerische Variable auf Null und String-Variable auf NUL zurück.

Beispiele

1Ø A = 5 : DIM C(5)

2Ø CLEAR

Setzt den Speicherplatz frei, der C() zugewiesen wurde, und setzt A auf Null zurück.

1 DATA ausdruck-liste

Dabei ist: ausdruck-liste: ausdruck
 oder: ausdruck, ausdruck-liste

Abkürzungen: DA., DAT.

Siehe auch: READ, RESTORE

Wirkung

Der Befehl DATA stellt die für den Befehl READ benötigten Daten bereit.

Anwendung

Sollen Daten in ein Feld gefüllt werden, empfiehlt es sich, diese Daten in DATA-Befehlen bereitzustellen und in einer FOR/NEXT-Schleife mit dem Befehl READ in das Feld zu lesen. Der erste READ-Befehl liest den ersten Wert des ersten DATA-Statements, der zweite READ-Befehl den zweiten Wert und so fort. Mit jedem READ-Befehl wird ein Zeiger auf den nächsten Wert gesetzt. Die READ-Befehle lesen die Daten der Reihe nach aus den DATA-Statements in der Reihenfolge, wie sie im Programm vorhanden sind, unabhängig davon, wieviele Werte in jedem DATA-Statement vorhanden sind oder wieviele DATA-Statements benutzt werden.

DATA-Statements haben keinen Einfluß auf den Ablauf des Programms und können daher an jeder beliebigen Stelle in das Programm eingefügt werden. Viele Programmierer setzen sie unmittelbar hinter die READ-Befehle, die auf die Daten zugreifen. Es ist möglich, die Werte in einem DATA-Statement mithilfe des Befehls RESTORE ein zweites Mal zu lesen.

Beispiele

10 DIM B(10)	Deklariert ein Feld.
20 WAIT 128	
30 FOR I = 1 TO 10	
40 READ B(I)	Liest die Werte aus dem DATA-Statement in B().
50 PRINT B(I)	B(1) wird 10, B(2) 20, B(3) 30 usw.
60 NEXT I	
70 DATA 10 20 30 40 50 60	
80 DATA 70 80 90 100	
90 END	

Befehle
DEGREE

1 DEGREE

Abkürzungen: DE., DEG., DEGR., DEGRE.

Siehe auch: GRAD und RADIAN

Wirkung

Mit dem Befehl DEGREE wird das Winkelmaß auf Altgrad umgeschaltet.

Anwendung

Der PC-1421 kann drei Winkelmaße verarbeiten - Altgrad, Bogenmaß und Neugrad. Diese Winkelmaße werden für die Spezifizierung der Argumente für die Winkel-funktionen SIN, COS und TAN und für die Ergebnisse der Umkehrfunktionen ASN, ACS und ATN benutzt.

Mit DEGREE werden alle Winkelmaße als Altgrad dargestellt, bis der Befehl GRAD oder RADIAN eingegeben wird. Mit den Funktionen DMS und DEG können Dezimalwinkel in die Form Grad, Minuten, Sekunden und umgekehrt umgewandelt werden. Die Funktionen REC und POL dienen zum Umwandeln von Polarkoor-dinaten in kartesische Koordinaten und umgekehrt.

Beispiele

10 DEGREE

20 X = ASN 1

30 PRINT X

X hat jetzt den Wert 90, d.h. 90° , den Arcussinns von 1.

1 DIM dim-liste

Dabei bedeuten:

<u>dim-liste</u>	:	<u>dimensionierung</u>
	oder :	<u>dimensionierung, dim-liste</u>
<u>dimensionierung</u>	:	<u>numerische dimensionierung</u>
	oder :	<u>string-dimensionierung</u>
<u>numerische dimensionierung</u>	:	<u>numerischer name (größe)</u>
<u>string-dimensionierung</u>	:	<u>string-name (größen)</u>
	oder :	<u>string-name (größen) * len</u>
<u>numerischer name</u>	:	<u>erlaubter numerischer Variablenname</u>
<u>string-name</u>	:	<u>erlaubter String-Variablenname</u>
<u>größen</u>	:	<u>größe</u>
	oder :	<u>größe, grÖße</u>
<u>größe</u>	:	<u>Anzahl der Elemente</u>
<u>len</u>	:	<u>Länge jedes Strings im String-Feld</u>

Abkürzungen: D., DI.

Wirkung

Der Befehl DIM dient zur Reservierung von Speicherplatz für numerische Felder und String-Felder.

Anwendung

Außer für Felder der Form A(), AS(), Zwei-Zeichen() und Zwei-Zeichen\$() muß für jedes andere Feld der erforderliche Speicherplatz mit einem DIM-Befehl reserviert werden.

Die Höchstzahl für Dimensionen eines Feldes ist zwei, jede Dimension darf in der Größe maximal 255 betragen. Zusätzlich zur Anzahl Elemente, die im DIM-Befehl spezifiziert wird, wird ein "null-tes" Element reserviert. Beispielsweise stelle DIM B(3) die Speicherplätze B(0), B(1), B(2) und B(3) bereit. Für zweidimensionale Felder wird je eine zusätzliche "null-te" Zeile und Spalte reserviert.

Für String-Felder muß zusätzlich zur Anzahl der Elemente die Größe jedes String-Elements spezifiziert werden. So wird mit DIM B\$(3) * 12 Speicherplatz für 4 Strings mit jeweils maximal 12 Zeichen reserviert. Wird die Länge nicht spezifiziert, wird ein Wert von 16 Zeichen pro String vorgegeben.

Befehle
DIM

Bei der Vereinbarung eines numerischen Feldes werden alle Werte auf Null und bei einem String-Feld alle Werte auf NUL gesetzt.

Siehe auch die Feld-Variablen auf Seite 67.

Für die DIM-Vereinbarung der Felder A und A\$ siehe den Abschnitt, in dem die Variablen erklärt werden.

Beispiele

- 10 DIM B(10) Reserviert Speicherplatz für ein numerisches Feld mit 11 Elementen.
- 20 DIM C\$(4, 4) * 10 Reserviert Speicherplatz für ein zweidimensionales String-Feld mit 5 Zeilen und 5 Spalten, jeder String kann maximal 10 Zeichen aufnehmen.

1 END

Abkürzungen: E., EN.

Wirkung

Der Befehl END signalisiert das Ende eines Programms.

Anwendung

Wenn gleichzeitig mehrere Programme im Speicher vorhanden sind, muß das Ende der einzelnen Programme bezeichnet werden, damit bei Ausführung eines Programms nicht auch andere Programme gestartet werden. Zu diesem Zweck wird als letztes Statement eines Programms END eingesetzt.

Beispiele

10 PRINT "HALLO"

20 END

30 PRINT "LEBEWOHL"

40 END

Wenn diese Programme im Speicher vorhanden sind, wird durch RUN 10 HALLO ausgedruckt, aber nicht LEBEWOHL. Durch RUN 30 wird LEBEWOHL ausgedruckt.

Befehle
ERASE

- | |
|---|
| <ol style="list-style-type: none">1 ERASE <u>feldname</u> [, <u>feldname</u>,...]2 ERASE FIN |
|---|

Wirkung

1) ERASE

Eine oder mehrere Feldvariable, spezifiziert durch ERASE feldname [, feldname ...], werden gelöscht.

2) ERASE FIN

Der Finanz-Speicher im Benutzerbereich wird gelöscht.

Beispiele

A = 10 **ENTER**

CST = 20 **ENTER**

ERASE FIN **ENTER**

A **ENTER** → 10

CST **ENTER** → 0

- ```
1 FOR numerische variable = ausdruck 1 TO ausdruck 2
2 FOR numerische variable = ausdruck 1 TO ausdruck 2
 STEP ausdruck 3
```

Abkürzungen: F. und FO.; STE.

Siehe auch: NEXT

### Wirkung

Der Befehl FOR wird in Verbindung mit dem Befehl NEXT gebraucht, um eine Serie von Operationen mehrmals zu wiederholen. Die Häufigkeit der Wiederholung wird spezifiziert.

### Anwendung

Die Befehle FOR und NEXT werden immer als Paar verwendet. Sie schließen die Statements ein, die wiederholt werden sollen. Bei der ersten Ausführung diese Statements hat die Schleifenvariable (die numerische Variable, die unmittelbar auf FOR folgt) den Wert von Ausdruck 1.

Wenn die Ausführung den Befehl NEXT erreicht, wird die Schleifenvariable um die Schrittweite (1 oder Wert von Ausdruck 3) erhöht, dann wird dieser Wert mit Ausdruck 2 verglichen. Ist der Wert der Schleifenvariablen kleiner oder gleich Ausdruck 2, wird die eingeschlossene Statement-Gruppe erneut ausgeführt, beginnend mit dem Statement nach FOR. In der ersten Form des Befehls ist die Schrittweite gleich 1, in der zweiten Form ist sie durch den Ausdruck 3 gegeben. Ist der Wert der Schleifenvariablen größer als Ausdruck 2, wird die Programmausführung mit dem Statement unmittelbar nach NEXT fortgesetzt. Da der Vergleich am Ende ausgeführt wird, werden die von einem FOR/NEXT-Paar eingeschlossenen Statements immer mindestens einmal ausgeführt.

Ausdruck 1, Ausdruck 2 und Ausdruck 3 müssen im Bereich  $-9,999999999E99 \sim 9,999999999E99$  liegen. Hat der Ausdruck 3 den Wert Null, wird die FOR/NEXT-Schleife unendlich oft ausgeführt.

Die Schleifenvariable kann innerhalb der eingeschlossenen Statement-Gruppe verwendet werden, beispielsweise als Index für ein Feld. Man muß aber sehr sorgfältig programmieren, wenn der Wert der Schleifenvariablen verändert werden soll.

Programme müssen so geschrieben werden, daß niemals von außerhalb in eine FOR/NEXT-Schleife gesprungen wird.

## Befehle FOR

Gleichermaßen dürfen FOR/NEXT-Schleifen niemals durch einen Sprung verlassen werden, sondern immer über das NEXT-Statement. Dafür muß die Schleifenvariable auf einen größeren Wert als Ausdruck 2 gesetzt werden.

Die Statement-Gruppe, die von einem FOR/NEXT-Paar eingeschlossen ist, kann weiter FOR/NEXT-Schleifen enthalten. Jede FOR/NEXT-Schleifen muß eine eigene Schleifenvariable haben, außerdem muß jedes Paar vollständig geschlossen sein, d.h. wenn in einer FOR/NEXT-Schleife ein FOR-Befehl enthalten ist, muß auch der zugehörige NEXT-Befehl enthalten sein. FOR/NEXT-Schleifen können bis zu fünffach "verschachtelt" werden.

### Beispiele

```
10 FOR I = 1 TO 5
20 PRINT I
30 NEXT I
```

Diese Schleife gibt die Zahlen 1, 2, 3, 4, 5 aus.

```
40 FOR N = 10 TO 0 STEP -1
50 PRINT N
60 NEXT N
```

In dieser Schleife werden die Zahlen 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0 rückwärts gezählt und ausgegeben.

```
70 FOR N = 1 TO 10
80 X = 190 : FOR F = 1 TO N
90 FOR F = 1 TO N
100 X = X * F
110 NEXT F
120 PRINT X
130 NEXT N
```

Diese Schleifen berechnen für die Zahlen 1 bis 10 die Fakultäten und geben die Ergebnisse aus.

**1 GOSUB ausdruck**

Abkürzungen: GOS., GOSU.

Siehe auch: GOTO, ON ... GOSUB, ON ... GOTO, RETURN

**Wirkung**

Der Befehl GOSUB dient zum Anspringen eines BASIC-Unterprogramms.

**Anwendung**

Wenn eine Gruppe von Befehlen mehrmals ausgeführt oder in mehreren Programmen benutzt werden soll, so empfiehlt sich die Verwendung dieser Befehls-Gruppe als Unterprogramm mit den Befehlen GOSUB und RETURN.

Diese Befehls-Gruppe muß sich an einer Stelle im Programm befinden, wo sie während der normalen Ausführung des Programms nicht erreicht wird. Normalerweise wird die Befehls-Gruppe nach dem END-Befehl gesetzt. An den Stellen im Hauptprogramm, wo Unterprogramme ausgeführt werden sollen, wird ein GOSUB-Statement mit einem Ausdruck, der die Anfangs-Zeilenummer des betreffenden Unterprogramms angibt, eingesetzt. Die letzte Zeile jedes Unterprogramms muß RETURN sein. Durch den Befehl GOSUB wird die Programmausführung mit der angegebenen Zeilenummer fortgesetzt, bis ein RETURN erreicht wird. Daraufhin wird die Ausführung mit dem Statement nach GOSUB fortgesetzt.

Auch in Unterprogrammen kann der Befehl GOSUB vorkommen. Auf diese Weise können Unterprogramme bis zu zehnfach "verschachtelt" werden.

Der Ausdruck im GOSUB-Statement darf kein Komma enthalten, beispielsweise darf A(1, 2) nicht eingesetzt werden. Um von einer bestimmten Programmstelle aus wahlweise mehrere Unterprogramme anzuspringen, wird der Befehl ON ... GOSUB verwendet. Daher besteht der Ausdruck im GOSUB-Statement normalerweise nur aus einer bestimmten Zeilenummer. Wird ein numerischer Ausdruck verwendet, muß er eine zulässige Zeilenummer bezeichnen, d.h. zwischen 1 und 65279 liegen, sonst wird die Fehlermeldung ERROR 4 verursacht.

**Beispiele**

```
10 GOSUB 100
20 END
100 PRINT "HALLO"
110 RETURN
```

Das Programm springt von Zeile 10 zum Unterprogramm in Zeile 100, gibt das Wort HALLO aus und kehrt dann zu Zeile 20 des Hauptprogramms zurück.

1 **GOTO** ausdruck

Abkürzungen: G., GO., GOT.

Siehe auch: GOSUB, ON . . . GOSUB, ON . . . GOTO

### **Wirkung**

Durch den Befehl **GOTO** springt die Programmausführung zur spezifizierten Zeilennummer.

### **Anwendung**

Mit dem Befehl **GOTO** springt die Programmausführung von einer Programmstelle zu einer anderen. Anders als **GOSUB** "erinnert" sich **GOTO** nicht an die Stelle, von der aus der Sprung erfolgte.

Der Ausdruck im **GOTO**-Statement darf kein Komma enthalten, beispielsweise darf **A(1, 2)** nicht eingesetzt werden. Um von einer bestimmten Programmstelle aus wahlweise mehrere Unterprogramme anzuspringen, wird der Befehl **ON . . . GOSUB** verwendet. Daher besteht der Ausdruck im **GOTO**-Statement normalerweise nur aus einer bestimmten Zeilennummer. Wird ein numerischer Ausdruck verwendet, muß er eine zulässige Zeilennummer bezeichnen, d.h. zwischen 1 und 65279 liegen, sonst wird die Fehlermeldung **ERROR 4** verursacht.

Gut Konstruierte Programme laufen ohne Sprünge vom Anfang bis zum Ende, abgesehen von Sprüngen zu Unterprogrammen. Daher liegt der grundsätzliche Gebrauch des Befehls **GOTO** im Statement **IF . . . THEN**.

### **Beispiele**

```
10 INPUT A$
20 IF A$ = "Y" THEN GOTO 50
30 PRINT "NO"
40 GOTO 60
50 PRINT "YES"
60 END
```

Dieses Programm druckt **YES**, wenn ein **Y** eingegeben wird, und sonst **NO**.

## 1 GRAD

Abkürzungen: GR., GRA.

Siehe auch: DEGREE und RADIAN

### Wirkung

Der Befehl GRAD dient zum Umschalten des Winkelmaßes auf Neugrad.

### Anwendung

Der PC-1421 hat drei Formen der Winkeldarstellung – Altgrad, Winkelmaß und Neugrad. Diese Formen dienen zur Spezifizierung der Argumente der Funktionen SIN, COS und TAN und Darstellung der Ergebnisse der Umkehrfunktionen ASN, ACS und ATN.

Der Befehl GRAD schaltet das Winkelmaß auf Neugrad um, bis der Befehl DEGREE oder RADIAN eingegeben wird. In Neugrad beträgt ein Vollkreis  $400^g$  ( $400 \text{ Gon}$ ), d.h.  $45^\circ$  ist gleich  $50^g$ .

### Beispiele

10 GRAD

20 X = ASN 1

30 PRINT X

X hat jetzt den Wert 100, d.h.  $100^g$ , den Arcussinus von 1.

## Befehle IF... THEN

- 1 **IF** bedingung **THEN** statement
- 2 **IF** bedingung **statement**

Abkürzungen: keine für IF, T., TH., THE.

### Wirkung

Das Befehlspar IF ... THEN dient dazu, ein Statement auszuführen, wenn eine Bedingung erfüllt ist.

### Anwendung

Im normalen Ablauf eines BASIC-Programms werden die Statements in der Reihenfolge ihres Vorkommens abgearbeitet. Das Befehlspar IF ... THEN ermöglicht es, einen Befehl nur dann auszuführen, wenn eine Bedingung erfüllt ist. Ist die Bedingung des IF-Statements wahr, wird das Statement ausgeführt, und sonst ausgelassen.

Die Bedingung des IF-Statements kann jeder Werhältnis-Ausdruck sein, wie in Kapitel 4 beschrieben. Es ist auch möglich, einen numerischen Ausdruck als Bedingung einzusetzen, obwohl die Bedeutung des Statements dadurch weniger klar wird. Jeder Ausdruck, der Null oder einen negativen Wert ergibt, wird als "falsch" angesehen. Ist der Ausdruck positiv, ist die Bedingung "wahr".

Nach THEN kann jedes beliebige BASIC-Statement folgen, selbst ein weiteres IF ... THEN. Ist es ein LET-Statement, darf der Befehl LET nicht ausgelassen werden. Außer wenn das Statement ein END, GOTO oder ON ... GOTO ist, wird das Statement, das auf dem IF ... THEN-Statement folgt, unabhängig davon, ob die Bedingung wahr ist, ausgeführt.

Die beiden Formen des IF-Statements sind in der Ausführung identisch, die erste Form ist jedoch klarer.

### Beispiele

```
10 INPUT "CONTINUE?",A$
20 IF A$ = "YES" THEN GOTO 10
30 IF A$ = "NO" THEN GOTO 60
40 PRINT "YES OR NO, PLEASE"
50 GOTO 10
60 END
```

Das Programm wiederholt die Frage CONTINUE?, solange YES eingegeben wird. Es stoppt, wenn NO eingegeben wird. Wird weder YES noch NO eingegeben, erscheint die Aufforderung YES OR NO, PLEASE.

## 1 INPUT eingabeliste

Dabei bedeuten:

|                       |       |                                            |
|-----------------------|-------|--------------------------------------------|
| <u>eingabeliste</u>   | :     | <u>eingabegruppe</u>                       |
|                       | oder: | <u>eingabegruppe</u> , <u>eingabeliste</u> |
| <u>eingabegruppe</u>  | :     | <u>variablenliste</u>                      |
|                       | oder: | <u>frage</u> , <u>variablenliste</u>       |
|                       | oder: | <u>frage</u> ; <u>variablenliste</u>       |
| <u>variablenliste</u> | :     | <u>variable</u>                            |
|                       | oder: | <u>variable</u> , <u>variablenliste</u>    |
| <u>frage</u>          | :     | beliebige String-Konstante                 |

Abkürzungen: I., IN., INP., INPU.

Siehe auch: INPUT#, READ

### Wirkung

Der Befehl INPUT dient zur Eingabe einer oder mehrerer Variablen von der Tastatur.

### Anwendung

Um bei jedem Lauf eines Programms verschiedene Werte zu verarbeiten, lassen sich diese Werte am einfachsten mit dem Befehl INPUT von der Tastatur eingeben.

In der einfachsten Form enthält der Befehl INPUT keinen Frage-String, stattdessen wird links auf dem Display das Fragezeichen angezeigt. Daraufhin muß ein Wert eingegeben und die Taste **ENTER** gedrückt werden. Dieser eingegebene Werte wird dere ersten Variablen in der Variablenliste des Befehls zugewiesen. Enthält die Liste noch weitere Variable, werden diese abgefragt, bis alle Variablen der Liste belegt sind.

Wenn das INPUT-Statement eine Frage enthält, ist der Ablauf genau gleich wie oben mit der Ausnahme, daß statt des Fragezeichens der Frage-String links auf dem Display angezeigt wird. Folgt auf dem Frage-String ein Semikolon, wird der Cursor unmittelbar hinter die Frage gesetzt. Wenn auf der Frage ein Komma folgt, wird der Frage-String angezeigt und, sobald eine Taste gedrückt wird, wieder gelöscht. Dann befindet sich das erste Zeichen der Eingabe ganz links auf dem Display.

Ist eine Frage spezifiziert und enthält die darauf folgende Variablenliste mehr als eine Variable, werden die zweite und weitere Variablen mit einem Fragezeichen abgefragt. Enthält die Liste eine zweite Frage, wird sie für die Variable angezeigt, die unmittelbar darauf folgt.

## Befehle INPUT

Wird kein Wert eingegeben, sondern nur die Taste **ENTER** gedrückt, bleibt der Wert der Variablen gleich wie vor dem INPUT-Statement.

### Beispiele

10 INPUT A

Löscht das Display und zeigt an der linken Seite ein Fragezeichen an.

20 INPUT "A=", A

Zeigt "A=" an und wartet auf Eingabedaten.

30 INPUT "A=", A

Zeigt "A=" an.

Wird ein Wert eingegeben, verschwindet "A=", und der eingegebene Werte wird von der linken Seite aus angezeigt.

40 INPUT "X=? "; X, "Y=? ";

Zeigt "X=?" an und wartet auf die erste Eingabe.

Nach Drücken von **ENTER** wird das Display gelöscht, und "Y=?" wird an der linken Seite angezeigt.

- 1 INPUT # variablenliste
- 2 INPUT # "dateiname"; variablenliste

Dabei ist:

variablenliste : variable  
oder: variable, variablenliste

Abkürzungen: I. #, IN. #, INP. #, INPU. #

Siehe auch: INPUT, PRINT #, READ

## Wirkung

Der Befehl INPUT # dient zur Eingabe von Werten von einer Cassette.

## Anwendung und Beispiele

In INPUT # Statements können die folgenden Variablentypen spezifiziert werden:

- (1) Feste Variable — A, B, C, A(7), D\*, A(20)\* usw.
- (2) Einfache Variable — AA, B3, CP\$ usw.
- (3) Feldvariable — S(\*), HP(\*), K\$(\*) usw.

### 1) Übertragung von Daten in feste Variable

Zur Übertragung von Daten von der Cassette in feste Variable müssen die Variablennamen im INPUT # Statement spezifiziert werden.

```
INPUT # "DATA 1"; A, B, X, Y
```

Dieses Statement überträgt Daten von der Cassetten-Datei "DATA 1" in die Variablen A, B, X und Y in dieser Reihenfolge.

Um alle festen Variablen und, falls definiert, erweiterte Variablen (A(27) und weiter) mit Daten von der Cassette zu füllen, muß der ersten Variablen ein Sternchen (\*) angehängt werden.

```
INPUT # "D-2"; D*
```

Dieses Statement überträgt den Inhalt der Cassetten-Datei "D-2" in die Variablen D bis Z und A(27) und weiter.

```
INPUT # A(10)* (ohne DIM-Deklaration)
```

Dieses Statement überträgt die Daten aus der ersten Datei, die auf dem Cassettenband gefunden wird, in die Variablen A(10) und weiter (in J bis Z und A(27) und weiter).

**Befehle**  
**INPUT #**

- Hinweis 1. Wurde bereits mit dem DIM-Statement ein Feld "A" definiert, können keine festen indizierten Variablen der Form A( ) definiert werden. Außerdem werden keine Daten in die Variablen A(27) und weiter übertragen.
- Hinweis 2. Die Übertragung von Daten in die festen Variablen und erweiterten Variablen (A(27) und weiter) wird fortgesetzt, bis das Ende der Cassetten-Datei erreicht wird oder bis der Speicher des Computers voll ist.

**2) Übertragung von Daten in einfache Variable**

Daten in einer Cassetten-Datei können in feste Variable übertragen werden, indem die gewünschten Variablennamen im INPUT # Statement spezifiziert werden.

```
INPUT # "DM-1"; AB, Y1, XY$
```

Dieses Statement überträgt Daten aus der Cassetten-Datei "DM-1" in die einfachen Variablen AB, Y1 und XY\$.

- Hinweis 1. Numerische Daten müssen in numerische einfache Variable und String-Daten in einfache String-Variable übertragen werden. Andere Zuweisungen sind nicht möglich.
- Hinweis 2. Speicherplatz für einfache Variable im Programm-Datenbereich muß vor der Ausführung des Statements INPUT # bereitgestellt werden, weil sonst ein Fehler verursacht wird. Diese Reservierung kann mit Zuweisungen aus geführt werden.

```
AA = 0 Für die Platzreservierung in den Zuwei-
B1$ = "A" sungen entsprechend der Variablen-Typen
INPUT # AA, B1$ numerische Werte oder Strings verwenden.
```

**3) Übertragung von Daten in Feldvariable**

Zur Übertragung von Daten aus einer Cassetten-Datei in Feldvariable muß der Feldname im INPUT # Statement in der Form "Feldname (\*)" spezifiziert werden.

```
5Ø DIM B(5)
6Ø INPUT # "DS-4"; B(*)
```

Dieses Statement überträgt Daten aus der Cassetten-Datei "DS-4" in die Variablen (B(0) bis B(5)) in Feld B.

- Hinweis 1. Numerische Daten müssen in numerische Feldvariable gleicher Länge und String-Daten müssen in String-Feldvariable gleicher Länge übertragen werden. Wird dies nicht beachtet, tritt ein Fehler auf.

Hinweis 2. Vor der Ausführung des Statements INPUT muß Speicherplatz für die Feldvariable bereitgestellt werden, weil sonst ein Fehler verursacht wird. Zur Definition des Feldes das DIM-Statement verwenden.

— ACHTUNG —

Falls die Anzahl der Variablen, die im INPUT # Statement spezifiziert wurde, nicht mit der Menge der auf der Cassette vorhandenen Daten übereinstimmt, passiert das folgende:

- \* Wenn die Datenmenge auf der Cassette größer ist als die Anzahl der spezifizierten Variablen, werden Daten bis zur letzten spezifizierten Variablen übertragen, die danach folgenden Daten auf der Cassette werden ignoriert.
- \* Wenn die Datenmenge auf der Cassette kleiner ist als die Anzahl der spezifizierten Variablen, werden alle Daten von der Cassette bis zum Ende der Datei in die Variablen übertragen, die verbleibenden Variablen behalten ihren vorigen Inhalt.  
In diesem Fall wartet der Computer jedoch auf Datenübertragung von der Cassette. Betätigen Sie die Taste 

|     |
|-----|
| ON  |
| BRK |

, um diesen Zustand zu beenden.
- \* Ist im INPUT # Statement keine Variable spezifiziert, erscheint eine Fehlermeldung (ERROR 1).

## Befehle

### LET

- 1 **LET** variable = ausdruck
- 2 variable = ausdruck

Abkürzungen: LE.

### Wirkung

Mit dem Befehl LET wird einer Variablen ein Wert zugewiesen.

### Anwendung

Der Befehl LET weist der angegebenen Variablen den Wert des Ausdrucks zu. Der Typ des Ausdrucks muß mit dem der Variablen übereinstimmen, d.h. nur numerische Ausdrücke können numerischen Variablen und nur String-Ausdrücke String-Variablen zugewiesen werden. Um einen Typ in den anderen umzuwandeln, muß eine der Funktionen STR\$ bzw. VAL eingesetzt werden.

Der Befehlsname LET kann in allen LET-Statements ausgelassen werden mit Ausnahme derer, die in nach THEN in IF...THEN Statements vorkommen. In diesem einen Fall muß der Befehlsname LET vorhanden sein.

### Beispiele

10 I = 10

Weist I den Wert 10 zu.

20 A = 5\*I

Weist A den Wert 50 zu.

30 X\$ = STR\$ (A)

Weist X\$ den Wert "50" zu.

40 IF I >= 10 THEN LET Y\$ = X\$ + ".00"

Weist Y\$ den Wert "50.00" zu.

- 1 LPRINT druckausdruck
- 2 LPRINT druckausdruck, druckausdruck
- 3 LPRINT druckliste

Dabei bedeuten:

druckliste : druckausdruck  
oder : druckausdruck; druckliste  
druckausdruck : ausdruck  
oder : USING-format: ausdruck

SING-format ist unter USING beschrieben.

Abkürzungen: LP., LPR., LPRI., LPRIN.

Siehe auch: PAUSE, PRINT, USING und WAIT

## Wirkung

Mit dem Befehl LPRINT werden Informationen auf dem Drucker des Drucker/Cassettenrecorder-Interface **CE-126P** (Sonderzubehör) ausgedruckt.

## Anwendung

Mit dem Befehl LPRINT können Eingabeaufforderungen, Berechnungsergebnisse usw. ausgedruckt werden. Die erste Form des LPRINT-Statements druckt einen einzelnen Wert aus. Numerische Ausdrücke werden ganz rechts und String-Ausdrücke ganz links auf dem Papier ausgedruckt.

Die zweite Form des LPRINT-Statements teilt das Papier in zwei Hälften mit je 12 Zeichen auf. In jede Hälfte wird ein Wert nach den obigen Regeln ausgedruckt.

Bei der dritten Form beginnt der Ausdruck immer an der linken Seite, und jeder Wert wird unmittelbar nach dem vorigen ohne Zwischenraum von links nach rechts ausgedruckt.

Es ist möglich, PRINT-Statements als LPRINT-Statements arbeiten zu lassen. Einzelheiten finden Sie unter dem Befehl PRINT.

Enthält ein LPRINT-Statement mehr als 24 Zeichen, werden die ersten 24 Zeichen in einer Zeile ausgedruckt, die nächsten 24 in der nächsten Zeile und so weiter.

Anders als bei PRINT gibt es nach der Ausführung eines LPRINT-Statements keine Unterbrechung oder Warten.

Befehle  
LPRINT

**Beispiele**

```
10 A = 10: B = 20: X$ = "ABCDEF"
20 LPRINT A
30 LPRINT X$
40 LPRINT A, B
50 LPRINT A; B; X$
```

Ausdruck

|              |     |
|--------------|-----|
| ABCDEF       | 10. |
| 10.20.ABCDEF | 20. |

1 NEXT numerische variable

Abkürzungen: N., NE., NEX.

Siehe auch: FOR

### Wirkung

Mit dem Befehl NEXT wird das Ende einer Gruppe von Statements gekennzeichnet, die in einer FOR/NEXT-Schleife wiederholt werden.

### Anwendung

Die Verwendung des Befehls NEXT ist unter FOR beschrieben. Die numerische Variable in einem NEXT Statement muß die gleiche wie im zugehörigen FOR-Befehl sein.

### Beispiele

```
10 FOR I = 1 TO 10 Die Zahlen 1 bis 10 werden ausgegeben.
20 PRINT I
30 NEXT I
```

1 ON ausdruck GOSUB ausdruckliste

Dabei ist:

ausdruckliste : ausdruck  
oder : ausdruck, ausdruckliste

Abkürzungen: O.: GOS., GOSU.

Siehe auch: GOSUB, GOTO, ON...GOTO

## **Wirkung**

Mit dem Befehl ON...GOSUB wird eines von mehreren Unterprogrammen abhängig vom Wert des Kontrollausdrucks ausgeführt.

## **Anwendung**

Bei Ausführung des Befehls ON...GOSUB wird der Ausdruck zwischen ON und GOSUB berechnet und in eine ganze Zahl umgewandelt. Ist der Wert der ganzen Zahl 1, wird das erste Unterprogramm in der Liste wie bei einem normalen GOSUB-Befehl ausgeführt. Ist der Ausdruck gleich 2, wird das zweite Unterprogramm ausgeführt, und so weiter. Nach RETURN des Unterprogramms wird die Programmausführung mit dem Statement fortgesetzt, das auf ON...GOSUB folgt.

Ist der Ausdruck gleich Null, negativ oder größer als die Anzahl der Unterprogramme in der Liste, wird kein Unterprogramm ausgeführt, und die Programmausführung wird mit der nächsten Zeile fortgesetzt.

**HINWEIS:** In den Ausdrücken, die auf GOSUB folgen, dürfen keine Kommata verwendet werden. Der PC-1421 kann nicht zwischen Kommata in Ausdrücken und Kommata zwischen Ausdrücken unterscheiden.

## **Beispiele**

```
10 INPUT A
20 ON A GOSUB 100, 200, 300
30 END
100 PRINT "FIRST"
110 RETURN
200 PRINT "SECOND"
210 RETURN
300 PRINT "THIRD"
310 RETURN
```

Die Eingabe 1 bewirkt die Ausgabe "FIRST", 2 "SECOND" und 3 "THIRD". Bei anderen Eingaben erfolgt keine Ausgabe.

1 ON ausdruck GOTO ausdruckliste

Dabei ist:

ausdruckliste : ausdruck  
oder : ausdruck, ausdruckliste

Abkürzungen: O.; G., GO., GOT.

Siehe auch: GOSUB, GOTO, ON...GOSUB

### Wirkung

Mit dem Befehl ON...GOTO wird die Programmausführung abhängig vom Wert des Kontrollausdrucks an einer anderen Stelle fortgesetzt.

### Anwendung

Bei Ausführung des Befehls ON...GOTO wird der Ausdruck zwischen ON und GOTO berechnet und in eine ganze Zahl umgewandelt. Ist der Wert der ganzen Zahl 1, wird das Programm an der ersten Stelle in der Liste fortgesetzt. Ist der Ausdruck gleich 2, wird es an der zweiten Stelle fortgesetzt, und so weiter.

Ist der Ausdruck gleich Null, negativ oder größer als die Anzahl der Sprungadressen in der Liste, wird die Programmausführung mit der nächsten Zeile fortgesetzt.

HINWEIS: In den Ausdrücken, die auf GOTO folgen, dürfen keine Kommata verwendet werden. Der PC-1421 kann nicht zwischen Kommata in Ausdrücken und Kommata zwischen Ausdrücken unterscheiden.

### Beispiele

```
10 INPUT A
20 ON A GOTO 100, 200, 300
30 GOTO 900
100 PRINT "FIRST"
110 GOTO 900
200 PRINT "SECOND"
210 GOTO 900
300 PRINT "THIRD"
310 GOTO 900
900 END
```

Die Eingabe 1 bewirkt die Ausgabe "FIRST", 2 "SECOND" und 3 "THIRD". Bei anderen Eingaben erfolgt keine Ausgabe.

- 1 PAUSE druckausdruck
- 2 PAUSE druckausdruck, druckausdruck
- 3 PAUSE druckliste

Dabei sind:

|                   |      |                                          |
|-------------------|------|------------------------------------------|
| <u>druckliste</u> | :    | <u>druckausdruck</u>                     |
|                   | oder | <u>druckausdruck</u> ; <u>druckliste</u> |
| druckausdruck     | :    | <u>ausdruck</u>                          |
|                   | oder | <u>USING-format</u> ; <u>ausdruck</u>    |

USING-format ist separat unter USING beschrieben.

Abkürzungen: PAU., PAUS.

Siehe auch: LPRINT, PRINT, USING und WAIT

## Wirkung

Mit dem Befehl PAUSE erfolgt eine zeitlich begrenzte Ausgabe auf dem Display.

## Anwendung

Der Befehl PAUSE dient dazu, Eingabeaufforderungen, Berechnungsergebnisse usw. anzuzeigen. Die Ausführung von PAUSE ist gleich wie beim PRINT-Befehl mit der Ausnahme, daß der PC-1421 nach der Ausführung von PAUSE eine kurze Zeitdauer von etwa 0,85 Sekunden wartet und dann die Programmausführung fortsetzt, ohne auf die ENTER-Taste oder die Dauer des WAIT-Intervalls zu warten.

Die erste Form des PAUSE-Statements zeigt einen einzigen Wert an. Ist der Ausdruck numerisch, wird der Wert ganz rechts auf dem Display angezeigt. Ist es ein String-Ausdruck, erfolgt die Anzeige ganz links auf dem Display.

Bei der zweiten Form des PAUSE-Statements wird das Display in zwei Hälften mit je 8 Zeichen aufgeteilt. Die beiden Werte werden nach den obigen Regeln in den beiden Hälften angezeigt.

Bei der dritten Form beginnt die Ausgabe an der linken Seite, und jeder Wert wird unmittelbar nach dem vorigen Wert ohne Zwischenraum von links nach rechts angezeigt.

PAUSE-Statements werden durch das Statement PRINT = LPRINT (siehe PRINT) nicht beeinflußt.

Es ist zwar möglich, PRINT-Statements mit mehr als 16 Ausgabezeichen zu schreiben, aber nur die ersten 16 Zeichen von links erscheinen auf dem Display. Es gibt keine Möglichkeit, die folgenden Zeichen anzuzeigen.

**Beispiele**

10 A = 10 : B = 20 : X\$ = "ABCDEF"

20 PAUSE A

30 PAUSE X\$

40 PAUSE A, B

50 PAUSE A; B; X\$

Anzeige

|                |
|----------------|
| 10.            |
| ABCDEF         |
| 10. 20.        |
| 10. 20. ABCDEF |

**Befehle**  
**PRINT**

- 1 **PRINT** druckausdruck
- 2 **PRINT** druckausdruck, druckausdruck
- 3 **PRINT** druckliste
- 4 **PRINT** = **LPRINT**
- 5 **PRINT** = **PRINT**

Dabei sind:

|                      |      |                                          |
|----------------------|------|------------------------------------------|
| <u>druckliste</u>    | :    | <u>druckausdruck</u>                     |
|                      | oder | <u>druckausdruck</u> ; <u>druckliste</u> |
| <u>druckausdruck</u> | :    | <u>ausdruck</u>                          |
|                      | oder | <u>USING-format</u> ; <u>ausdruck</u>    |

USING-format ist separat unter USING beschrieben.

Abkürzungen: P., PR., PRI., PRIN.

Siehe auch: LPRINT, PAUSE, USING und WAIT

### **Wirkung**

Mit dem Befehl PRINT werden Ausgaben auf dem Display oder dem Drucker des Drucker/Cassettenrecorder-Interface CE-126P ausgeführt.

### **Anwendung**

Der Befehl PRINT dient dazu, Eingabeaufforderungen, Berechnungsergebnisse usw. anzuzeigen. Die erste Form des PRINT-Statements zeigt einen einzigen Wert an. Ist der Ausdruck numerisch, wird der Wert ganz rechts auf dem Display angezeigt. Ist es ein String-Ausdruck, erfolgt die Anzeige ganz links auf dem Display.

Bei der zweiten Form des PRINT-Statements wird das Display in zwei Hälften mit je 8 Zeichen aufgeteilt. Die beiden Werte werden nach den obigen Regeln in den beiden Hälften angezeigt.

Bei der dritten Form beginnt die Ausgabe an der linken Seite, und jeder Wert wird unmittelbar nach dem vorigen Wert ohne Zwischenraum von links nach rechts angezeigt.

Die vierte und fünfte Form des PRINT-Statements bewirken keine Ausgabe. Die vierte Form bewirkt, daß alle darauf im Programm folgenden PRINT-Statements als LPRINT-Statements behandelt werden. Die fünfte Form stellt diesen Status zurück, so daß PRINT-Statements wieder Ausgabe auf das Display bewirken.

Es ist zwar möglich, PRINT-Statements mit mehr als 16 Ausgabezeichen zu schreiben, aber nur die ersten 16 Zeichen von links erscheinen auf dem Display. Es gibt keine Möglichkeit, die folgenden Zeichen anzuzeigen.

**Beispiele**

```
10 A = 10 : B = 20 : X$ = "ABCDEF"
```

Anzeige

```
20 PRINT A
```

|     |
|-----|
| 10. |
|-----|

```
30 PRINT X$
```

|        |
|--------|
| ABCDEF |
|--------|

```
40 PRINT A, B
```

|     |     |
|-----|-----|
| 10. | 20. |
|-----|-----|

```
50 PRINT A; B; X$
```

|              |
|--------------|
| 10.20.ABCDEF |
|--------------|

**Befehle**  
**PRINT #**

- 1 **PRINT #** variablenliste
- 2 **PRINT #** "dateiname"; variablenliste

Dabei ist:

variablenliste : variable  
oder : variable, variablenliste

Abkürzungen: P. #, PR. #, PRI. #, PRIN. #

Siehe auch: INPUT #, PRINT, READ

## Wirkung

Der Befehl PRINT # dient zum Speichern von Daten auf Cassette.

## Anwendung und Beispiele

Die folgenden Variablentypen können spezifiziert werden:

- (1) Feste Variable — A, B, X, A(26), C\*, A(10)\*, usw.
- (2) Einfache Variable — AA, B2, XY\$ usw.
- (3) Feldvariable — B(\*), CD(\*), N\$(\*) usw.

### 1) Sichern fester Variablen auf Cassette

Der Inhalt fester Variablen: kann auf Cassette gesichert werden, indem die gewünschten Variablennamen (getrennt durch Kommata) im PRINT # Statement spezifiziert werden.

```
PRINT # "DATA 1"; A, B, X, Y
```

Dieses Statement sichert die Variablen A, B, X und Y in die Cassetten-Datei "DATA 1".

Um alle feste Variablen, beginnend mit einer bestimmten, auf Cassette zu sichern, wird die erste Variable mit einem Sternchen \* versehen.

PRINT# "D-2"; D\*      Dieses Statement sichert die festen Variablen D bis Z (und die erweiterten Variablen A(26) und weiter, falls definiert) in die Cassetten-Datei "D-2".

PRINT# E, X\$, A(30)\*      Dieses Statement sichert die festen Variablen E und X\$ und die erweiterten Variablen A(30) und alle weiteren auf die Cassette.

Hinweis:

Die indizierten festen Variablen A(1) bis A(26) können auf die gleiche Weise wie A bis Z (oder A\$ bis Z\$) im PRINT # Statement spezifiziert werden. Wurde jedoch in einem DIM-Statement das Feld A definiert, kann A( ) nicht mehr zur Definition indizierter fester Variablen benutzt werden.

## 2) Sichern einfacher Variablen (Zwei-Zeichen-Variable)

Einfache Variable werden durch Spezifizieren der Variablenamen auf Cassette gesichert.

```
PRINT # "DM-1"; AB, Y1, XY$
```

Dieses Statement sichert den Inhalt der einfachen Variablen AB, Y1 und XY\$ in die Cassetten-Datei "DM-1".

## 3) Sichern von Feld-Variablen

Alle Variablen eines Feldes werden auf Cassette gesichert, indem der Feldname mit einem Sternchen in Klammern (\*) versehen wird.

```
PRINT # "DS-2"; X(*), Y$(*)
```

Dieses Statement sichert alle Elemente (X(0), X(1), ...) des Feldes X und alle Elemente (Y\$(0), Y\$(1), ...) des Feldes Y\$ in die Cassetten-Datei "DS-2".

Hinweis:

Es ist nicht möglich, nur ein Element oder nur bestimmte Elemente eines Feldes auf Cassette zu sichern.

Bei festen Variablen und indizierten festen Variablen ist es möglich, nur bestimmte dieser Variablen auf Cassette zu sichern. Bei Feldern (wie A), die mit dem DIM-Statement definiert wurden, können keine bestimmten Teile davon gesichert werden.

- \* Wird das PRINT # Statement ausgeführt, ohne daß Variablenamen spezifiziert sind, erscheint eine Fehlermeldung (ERROR 1).

## — ACHTUNG —

Die Speicherplätze für erweiterte Variable wie A(27) und weiter, einfache Variable und/oder Feldvariable müssen im Programm/Daten-Bereich bereitgestellt werden, bevor das PRINT # Statement ausgeführt wird. Sonst führt die Ausführung des PRINT # Statements für nicht definierte Variable zu einem Fehler.

**Befehle**  
**RADIAN**

**1 RADIAN**

Abkürzungen: RAD., RADL., RADIA.

Siehe auch: DEGREE und GRAD

**Wirkung**

Mit dem Befehl RADIAN wird die Winkel-Darstellung in Bogenmaß umgeschaltet.

**Anwendung**

Der **PC-1421** hat drei Möglichkeiten zur Darstellung von Winkelwerten: Altgrad, Bogenmaß und Neugrad. Diese Formen werden benutzt, um die Argumente für die Funktionen SIN, COS und TAN zu spezifizieren und die Ergebnisse der Funktionen ASN, ACS und ATN anzuzeigen.

Mit dem Befehl RADIAN werden alle Winkelwerte in Bogenmaß dargestellt, bis einer der Befehle DEGREE oder GRAD eingegeben wird. In dieser Darstellung werden Winkel als Bogenmaß in Abhängigkeit vom Radius wiedergegeben. Dann ist  $360^\circ$  im Bogenmaß  $2\text{ PI}$ , da der Kreisumfang  $2\text{ PI}$  mal dem Radius ist.

**Beispiele**

```
10 RADIAN
20 X = ASN 1
30 PRINT X
```

X hat jetzt den Wert 1,570796327 oder  $\text{PI}/2$ , den Arcussinus von 1.

## 1 RANDOM

Abkürzungen: RA., RAN., RAND., RANDO.

### Wirkung

Mit dem Befehl RANDOM wird eine neue Startzahl für den Zufallszahlen-Generator eingestellt.

### Anwendung

Bei der Erzeugung von Zufallszahlen mit der Funktion RND startet der **PC-1421** mit einer vorgegebenen Zahl. Mit dem Befehl RANDOM wird diese Zahl auf einen neuen, zufällig bestimmten Wert gesetzt.

Die Startzahl für den Zufallszahlen-Generator ist nach dem Einschalten des **PC-1421** immer die gleiche. Daher wiederholt sich auch die Sequenz der mit der RND-Funktion erzeugten Zufallszahlen, wenn nicht die Startzahl geändert wird. Diese Eigenschaft ist sehr hilfreich bei der Erstellung eines Programms, weil das Verhalten des Programms bei jedem Lauf trotz der RND-Funktion immer das gleiche sein wird. Und um echte Zufallszahlen zu erhalten, wird die Startzahl mit dem RANDOM-Befehl zufällig gewählt.

### Beispiele

```
10 RANDOM
20 X = RND 10
```

Wird das Programm in Zeile 20 gestartet, beruht der Wert von X auf der Standard-Startzahl. Wird es in Zeile 10 gestartet, wird eine neue Startzahl eingestellt.

**Befehle**  
**READ**

1 **READ** variablenliste

Dabei ist:

variablenliste : variable  
oder: variable, variablenliste

Abkürzungen: REA.

Siehe auch: DATA, RESTORE

**Wirkung**

Mit dem Befehl READ werden Werte aus DATA-Statements gelesen und Variablen zugewiesen.

**Anwendung**

Um einem Feld Anfangswerte zuzuweisen, ist es praktisch, diese Werte in einem DATA-Statement unterzubringen und mit dem READ-Befehl in einer FOR ... NEXT-Schleife in das Feld zu lesen. Bei der ersten Ausführung von READ wird der erste Wert im ersten DATA-Statement gelesen. Nachfolgende READ-Befehle lesen die Werte in der Reihenfolge ihres Vorkommens im Programm, unabhängig davon, wieviele Werte in jedem DATA-Statement vorhanden sind und wieviele DATA-Statements benutzt werden.

Mit dem RESTORE-Statement ist es möglich, die Werte in DATA-Statements ein zweites Mal zu lesen.

**Beispiele**

10 DIM B (10)

Deklaration eines Feldes

20 WAIT 128

30 FOR I = 1 TO 10

40 READ B(I)

Die Werte aus den DATA-Statements werden in B( ) gelesen — B(1) wird 10, B(2) 20, B(3) 30 usw.

50 PRINT B(I)

60 NEXT I

70 DATA 10, 20, 30, 40, 50, 60

80 DATA 70, 80, 90, 100

90 END

## 1 REM kommentar

Abkürzungen: keine

### Wirkung

Mit dem Befehl REM werden Kommentare in Programme eingefügt.

### Anwendung

In der Regel ist es unerlässlich, erklärende Kommentare in ein Programm einzufügen. Diese Kommentare können Titel, Namen des Programmierers, Datum der letzten Änderung, Anwendungshinweise, Erläuterungen der verwendeten Algorithmen usw. enthalten. Solche Kommentare werden mit dem REM-Statement in das Programm eingefügt.

Der REM-Befehl beeinflusst die Ausführung des Programms nicht und kann überall im Programm eingefügt werden. Alles, was auf das Befehlswort REM in einer Zeile folgt, wird als Kommentar angesehen. Daher muß das REM-Statement das letzte in einer Zeile sein, wenn Verbund-Statements benutzt werden.

### Beispiele

```
10 REM THIS LINE HAS NO EFFECT
```

## Befehle RESTORE

- 1 RESTORE
- 2 RESTORE ausdruck

Abkürzungen: RES., REST., RESTO., RESTOR.

Siehe auch: DATA, READ

### Wirkung

Der Befehl RESTORE dient dazu, Werte in DATA-Statements noch einmal zu lesen oder die Reihenfolge zu ändern, in der diese Werte gelesen werden.

### Anwendung

Bei der normalen Verwendung des READ-Befehls liest der **PC-1421** den ersten Wert in einem DATA-Statement und danach der Reihe nach die folgenden Werte. Die erste Form des RESTORE-Statements setzt den READ-Zeiger auf den ersten Wert des ersten DATA-Statements zurück, so daß alles noch einmal gelesen werden kann. Die zweite Form des RESTORE-Statements setzt den READ-Zeiger auf den ersten Wert des ersten DATA-Statements, dessen Zeilennummer größer als der Wert des Ausdrucks ist.

### Beispiele

```
10 DIM B(10)
20 FOR I = 1 TO 10
30 READ B(I)
40 RESTORE
50 NEXT I
60 DATA 10
```

Deklaration eines Feldes

Jedem Element von B( ) wird der Wert 10 zugewiesen.

## 1 RETURN

Abkürzungen: RE., RET., RETU., RETUR.

Siehe auch: GOSUB, ON . . . GOSUB

### Wirkung

Der Befehl RETURN bildet die letzte Zeile von Unterprogrammen. Von dort wird die Programmausführung mit dem Statement, das auf den GOSUB-Befehl folgt, fortgesetzt.

### Anwendung

Unterprogramme können mehrere RETURN-Statements enthalten. Die Ausführung von Unterprogrammen wird jedoch mit dem ersten Ausführen eines RETURN-Statements beendet. Dann wird die Programmausführung mit dem Statement fortgesetzt, das auf GOSUB oder ON . . . GOSUB, von wo das Unterprogramm aufgerufen wurde, folgt. Wird ein RETURN ohne zugehöriges GOSUB ausgeführt, erfolgt die Fehlermeldung ERROR 5.

### Beispiele

```
10 GOSUB 100
20 END
100 PRINT "HALLO"
110 RETURN
```

Bei der Ausführung dieses Programms wird das Wort "HALLO" einmal ausgegeben.

**Befehle**  
**STOP**

**1 STOP**

Abkürzungen: S., ST., STO.

Siehe auch: END; Kommando CONT

**Wirkung**

Mit dem Befehl **STOP** wird die Ausführung von Programmen für Prüfzwecke unterbrochen.

**Anwendung**

Wird bei der Programmausführung auf den Befehl **STOP** gestoßen, unterbricht der **PC-1421** die Ausführung, und eine Meldung "BREAK IN 200" wird angezeigt, wobei 200 die Nummer der Zeile mit **STOP** ist. Der Befehl **STOP** wird beim Erstellen von Programmen verwendet, um den Ablauf zu überprüfen oder um den Status von Variablen zu kontrollieren. Die Programmausführung kann mit dem Kommando **CONT** fortgesetzt werden.

**Beispiele**

10 STOP

Auf dem Display erscheint die Meldung "BREAK IN 10".

## 1 TROFF

Abkürzungen: TROF.

Siehe auch: TRON

### Wirkung

Mit dem Befehl TROFF wird der Trace-Modus ausgeschaltet.

### Anwendung

Die Ausführung des Befehls TROFF bewirkt, daß Programme wieder auf "normale" Weise ausgeführt werden.

### Beispiele

```
10 TRON
20 FOR I = 1 TO 3
30 NEXT I
40 TROFF
```

Während der Ausführung dieses Programms werden die Zeilennummern 10, 20, 30, 30 und 40 angezeigt.

## 1 TRON

Abkürzungen: TR., TRO.

Siehe auch: TROFF

### Wirkung

Mit dem Befehl TRON wird der Trace-Modus eingeschaltet.

### Anwendung

Der Trace-Modus hilft bei der Fehlersuche in Programmen. Wenn der Trace-Modus eingeschaltet ist, wird die Zeilennummer jedes Statements angezeigt, **nachdem** das Statement ausgeführt wurde. Danach unterbricht der **PC-1421** die Ausführung und wartet mit der Ausführung der nächsten Zeile, bis die Taste mit dem Abwärts-Pfeil gedrückt wird. Durch Drücken des Aufwärts-Pfeils wird das soeben ausgeführte Statement angezeigt. Der Trace-Modus bleibt wirksam, bis er mit dem Befehl TROFF ausgeschaltet wird.

### Beispiele

```
10 TRON
20 FOR I = 1 TO 3
30 NEXT I
40 TROFF
```

Während der Ausführung dieses Programms werden die Zeilennummern 10, 20, 30, 30, 30 und 40 angezeigt.

- 1 USING
- 2 USING "spezifikation"
- 3 USING zeichenvariable

Abkürzungen: U., US., USI., USIN.

Siehe auch: LPRINT, PAUSE, PRINT

Weitere Erklärungen zum Gebrauch von USING finden Sie in Anhang C.

### Wirkung

Mit dem Befehl USING wird die Ausgabe auf Display oder Drucker formatiert.

### Anwendung

Der Befehl USING kann alleinstehend oder in den Statements LPRINT, PAUSE und PRINT vorkommen. Mit dem Befehl USING wird ein Ausgabeformat vereinbart, das für alle folgenden Ausgaben wirksam ist, bis es durch einen anderen USING-Befehl geändert wird.

Die Spezifikation des Befehls USING besteht aus einer Zeichenkette in Anführungsstrichen, die aus einer Kombination der folgenden Zeichen besteht:

- # Rechtsbündiges Feld für numerische Zeichen
  - Dezimalpunkt
- ^ Zur Kennzeichnung, daß Zahlen in wissenschaftlicher Notation dargestellt werden sollen.
- & Linksbündiges Feld für alphanumerische Zeichen.
  - Aufgliederung nach drei Ziffern
 Das USING-Kommando kann spezifizieren, daß das Ergebnis von manuellen Berechnungen oder von Programmen auf dem Display in Gruppen von drei Ziffern aufgegliedert wird, die durch Kommata getrennt werden.

Die Spezifikation "#####" ergibt ein rechtsbündiges Feld für numerische Zeichen mit 3 Ziffern und dem Vorzeichen. In numerischen Feldern muß eine Stelle für das Vorzeichen freigehalten werden, auch wenn die Zahlen immer positiv sind.

USING-Spezifikationen können mehr als ein Feld beschreiben. Beispielsweise werden mit "####&&&" ein numerischen und ein alphanumerisches Feld direkt nebeneinander ausgegeben.

**Befehle**  
**USING**

Enthält der Befehl keine Spezifikation, wie in Form 1, wird die vorher spezifizierte Formatierung ausgeschaltet, und das Standard-Format wird wieder verwendet.

**Beispiele**

10 A = 125 : X\$ = "ABCDEF"

Anzeige

20 PRINT USING "##.##^"; A

1. 25E 02

30 PRINT USING "&&&&&&&"; X\$

ABCDEF

40 PRING USING "#####&&"; A; X\$

125ABC

- Hinweise:
1. Wenn die Gesamtzahl der Stellen, die mit USING für "PRINT ausdruck" spezifiziert wurde, 16 überschreitet, resultiert ERROR 7.
  2. Wenn die Anzahl der Stellen für den ganzzahligen Anteil (einschließlich Vorzeichen und Dezimalpunkt) für die Fest-Dezimalpunkt-Darstellung für "PRINT ausdruck, ausdruck" 8 überschreitet, resultiert ERROR 7.  
Wenn der Zeichenstring des Ausdrucks in der Form "PRINT ausdruck, ausdruck" 8 Stellen überschreitet, wird der überschüssige Teil nicht angezeigt.
  3. Wenn der anzuzeigende Inhalt der Form "PRINT ausdruck, ausdruck" 16 Stellen überschreitet, wird der überschüssige Teil nicht angezeigt.

● **Aufgliederung nach 3 Ziffern**

Programmbeispiel

PRO-Modus

10 USING ",#####.##"

20 PRINT 60 \* 40

30 END

RUN-Modus

RUN **ENTER** → 2,400.00

- 1 WAIT
- 2 WAIT ausdruck

Abkürzungen: W., WA., WAI.

Siehe auch: PAUSE, PRINT

### Wirkung

Mit dem Befehl WAIT wird die Zeitdauer bestimmt, die eine Anzeige auf dem Display bleibt, bevor die Programmausführung fortgesetzt wird.

### Anwendung

Bei der normalen Ausführung unterbricht der PC-1421 die Programmausführung nach PRINT-Befehlen, bis die Taste **ENTER** gedrückt wird. Der WAIT-Befehl bewirkt, daß die Anzeigen für die bestimmte Zeitdauer erscheinen und daß danach die Programmausführung automatisch fortgesetzt wird (ähnlich dem PAUSE-Befehl). Die Zeitdauer wird von dem Ausdruck bestimmt, der nach WAIT steht. Die Zeitdauer kann auf jeden Wert zwischen 0 und 65535 eingestellt werden. Dabei wird die Zeitdauer durch jede Erhöhung des Werts um 1 um etwa 1/59 Sekunde verlängert. WAIT 0 ist zu schnell, um gelesen werden zu können, WAIT 65535 dauert etwa 19 Minuten. WAIT ohne Ausdruck bewirkt eine Rückstellung des PC-1421 auf die normale Arbeitsweise, bei der auf Drücken der Taste **ENTER** gewartet wird.

### Beispiele

10 WAIT 59

Nach PRINT wird etwa 1 Sekunde gewartet.

# FUNKTIONEN

## Pseudovariablen

Pseudovariablen sind eine Gruppe von Funktionen, die kein Argument annehmen und wie feste Variablen verwendet werden.

### 1 INKEY\$

INKEY\$ ist eine String-Pseudovariablen, die den Wert der letzten Taste hat, die gedrückt wurde. **ENTER**, **ESC**, **SHIFT**, **DEF**, **↑**, **↓**, **▶** und **◀** haben alle den Wert NUL. Mit INKEY\$ können Eingaben über die Tastatur gemacht werden, ohne daß am Ende der Eingabe die Taste **ENTER** gedrückt werden muß.

```
10 A$ = INKEY$
20 B = ASC A$
30 IF B = 0 THEN GOTO 10
40 IF B . . .
```

Die Zeilen ab Zeile 40 können eine Tastatur-Testroutine enthalten. (Beispielsweise 40 PRINT A\$.) Bei der ersten Ausführung des Programms ist der Wert von INKEY\$ gleich NUL, weil die zuletzt die Taste **ENTER** gedrückt wurde. Folgt INKEY\$ auf PRINT oder PAUSE, nimmt INKEY\$ den Wert der Anzeige an.

### 1 MEM

MEM ist eine numerische Pseudovariablen, die den Wert der Anzahl der im Programmspeicher verbleibenden Zeichen annimmt. Der verfügbare Programmspeicher ist der Gesamtspeicher minus dem Speicherplatz, der von Programmen und Feldvariablen belegt ist. MEM kann auch als Kommando eingesetzt werden. Unmittelbar nach der Initialisierung hat MEM den Wert 3454.

### 1 PI

PI ist eine numerische Pseudovariablen, die den Wert pi hat. Wie alle anderen Zahlen wird auch der Wert pi mit einer Genauigkeit von 10 Stellen wiedergegeben (3,141592654).

## Numerische Funktionen

Numerische Funktionen sind eine Gruppe mathematischer Operatoren. Einige haben als Argument einen Wert und als Ergebnis einen Wert. Andere numerische Funktionen erfordern zwei Argumente und liefern zwei Werte als Ergebnis. Zu den numerischen Funktionen gehören trigonometrische Funktionen, logarithmische Funktionen und Funktionen, die den ganzzahligen Anteil und das Signum von Zahlen errechnen. In vielen BASIC-Dialekten ist es erforderlich, daß die Argumente von Funktionen in Klammern eingeschlossen werden. Der **PC-1421** benötigt Klammersetzung nur dann, wenn dadurch in komplizierten Ausdrücken das Argument genau angegeben werden muß. Numerische Funktionen mit zwei Argumenten erfordern immer Klammersetzung.

LOG 100 + 100 wird interpretiert als  
(LOG 100) + 100                      und nicht als                      LOG (100 + 100).

### 1 ABS numerischer ausdrück

ABS ist eine numerische Funktion, die den Absolutwert des Arguments berechnet. Der Absolutwert einer Zahl ist der Wert ohne Berücksichtigung des Vorzeichens. ABS -10 ist 10.

### 1 ACS numerischer ausdrück

ACS berechnet den Arcussinus des Arguments. Der Arcussinus ist der Winkel, dessen Cosinus gleich dem Ausdruck ist. Das Ergebnis ist unterschiedlich abhängig davon, ob der **PC-1421** als Winkel-Modus auf Altgrad, Bogenmaß oder Neugrad eingestellt ist. ACS .5 ist im Altgrad-Modus gleich 60.

### 1 AHC numerischer ausdrück

AHC berechnet den hyperbolischen Arcuscossinus des Arguments. AHC 5 ist 2,29243167.

1 **AHS** numerischer ausdrück

AHS berechnet den hyperbolischen Arcussinus des Arguments. AHS 6 ist 2,491779853.

1 **AHT** numerischer ausdrück

AHT berechnet den hyperbolischen Arcustangens des Arguments.

1 **ASN** numerischer ausdrück

ASN berechnet den Arcussinus des Arguments. Der Arcussinus ist der Winkel, des Sinus gleich dem Argument ist. Das Ergebnis ist vom Winkelmaß-Modus des **PC-1421** – Altgrad, Neugrad oder Bogenmaß – abhängig. ASN .5 ist im Altgrad-Modus gleich 30.

1 **ATN** numerischer ausdrück

ATN berechnet den Arcustangens des Arguments. Der Arcustangens ist der Winkel, dessen Tangens gleich dem Ausdruck ist. Das Ergebnis ist abhängig vom Winkelmaß-Modus des **PC-1421** – Altgrad, Bogenmaß oder Neugrad. ATN 1. ist gleich 45 im Altgrad-Modus.

1 **COS** numerischer ausdrück

COS berechnet den Cosinus des Winkel-Arguments. Das Ergebnis ist abhängig vom Winkelmaß-Modus des **PC-1421** – Altgrad, Bogenmaß oder Neugrad. Cos 60 ist gleich 0,5 im Altgrad-Modus.

1 **CUR** numerischer ausdruck

CUR berechnet die Kubikwurzel des Arguments. CUR 8 ist gleich 2.

1 **DEG** numerischer ausdruck

Die Funktion DEG wandelt ein Winkel-Argument im DMS-Format (Grad, Minuten, Sekunden) in die Dezimaldarstellung DEG um. Im DMS-Format sind der ganzzahlige Anteil der Zahl die Grad, die erste und zweite Dezimalstelle die Minuten, die dritte und vierte Dezimalstelle die Sekunden und weitere Stellen die Dezimalsekunden. Beispielsweise wird  $55^{\circ} 10' 44,5''$  als 55.10445 dargestellt. Im DEG-Format ist der ganzzahlige Anteil auch Grad und die Dezimalstellen sind Dezimalgradbruchteile. DEG 55.10445 ist gleich 55.17902778.

1 **DMS** numerischer ausdruck

DMS wandelt ein Winkel-Argument im DEG-Format in das DMS-Format um (Umkehrfunktion zu DEG). DMS 55.17902778 ist gleich 55.10445.

1 **EXP** numerischer ausdruck

EXP berechnet die Potenz der Zahl e (2,718281828 – die Basis des natürlichen Logarithmus). EXP 1 ist gleich 2.718281828. ( **E** **X** **P** 1 **ENTER** drücken.).

1 **FACT** numerischer ausdruck

FACT berechnet die Fakultät des Arguments. FACT 5 ist gleich 120.

**Funktionen**  
**Numerische Funktionen**

1 **HCS** numerischer ausdrück

HCS berechnet den hyperbolischen Cosinus des Arguments. HCS 5 ist gleich 74.20994852.

1 **HSN** numerischer ausdrück

HSN berechnet den hyperbolischen Sinus des Arguments. HSN 4 ist gleich 27.2899172.

1 **HTN** numerischer ausdrück

HTN berechnet den hyperbolischen Tangens des Arguments. HTN 1 ist gleich 0.761594156.

1 **INT** numerischer ausdrück

INT ergibt den ganzzahligen Anteil des Arguments. INT PI ist gleich 3.

1 **LN** numerischer ausdrück

LN berechnet den Logarithmus des Arguments zur Basis e (2,718281828). LN 100 ist gleich 4.605170186.

1 **LOG** numerischer ausdrück

LOG berechnet den Logarithmus des Arguments zur Basis 10. LOG 100 ist gleich 2.

1 **MDF** numerischer ausdrück

Diese Rundungs-Funktion ist praktisch, um beispielsweise ein Rechenergebnis, das in weiteren Berechnungen verwendet werden soll, auf 3 Dezimalstellen zu runden, so daß die Zahl nicht von der Tastatur eingegeben werden muß.

Beispiel

USING "###.###" **ENTER**  
MDF (5/9) \* 9 **ENTER** 5.004

Diese Funktion arbeitet nur dann, wenn die Anzahl der Dezimalstellen mit dem Kommando USING spezifiziert wurde. Sie wird ignoriert, wenn das USING-Kommando freigegeben wird.

MDF kann als gewöhnliche Funktion in BASIC-Programmen benutzt werden.

1 **POL** (numerischer ausdrück, numerischer ausdrück)

POL verwandelt Argumente in kartesischer Koordinaten-Darstellung in Polarkoordinaten.

Das erste Argument bezeichnet den Abstand zur Y-Achse und das zweite den Abstand zur X-Achse. Das Ergebnis, der Abstand und der Winkel in Polarkoordinaten, wird den festen Variablen Y und Z zugewiesen. Das Ergebnis ist abhängig vom Winkelmaß-Modus des PC-1421 – Altgrad, Bogenmaß oder Neugrad. POL (3, 4) ist im Altgrad-Modus gleich (5,53.13010235).

1 **RCP** numerischer ausdrück

RCP berechnet den Kehrwert des Arguments. RCP 5 ist gleich 0.2.

1 **REC** (numerischer ausdrück, numerischer ausdrück)

REC verwandelt Argumente in Polarkoordinaten-Darstellung in kartesische Koordinaten.

Das erste Argument bezeichnet den Abstand und das zweite den Winkel, welcher vom Winkelmaß-Modus des PC-1421 – Altgrad, Bogenmaß oder Neugrad –

**Funktionen**  
**Numerische Funktionen**

abhängig ist. Das Ergebnis, der Abstand zur Y-Achse und der Abstand zur X-Achse, wird in den festen Variablen Y und Z abgelegt. REC (7, 50) ist im Altgrad-Modus gleich (4.499513268, 5.362311102).

1 **RND** numerischer ausdrück

RND erzeugt Zufallszahlen. Ist das Argument kleiner als 1, aber größer oder gleich 0, ist auch die erzeugte Zufallszahl kleiner als 1, aber größer oder gleich 0. Ist das Argument eine ganze Zahl größer oder gleich 1, ist das Ergebnis eine Zufallszahl größer oder gleich 1 und kleiner oder gleich dem Argument. Ist das Argument größer oder gleich 1 und keine ganze Zahl, ist die erzeugte Zufallszahl größer oder gleich 1 und kleiner oder gleich der kleinsten ganzen Zahl, die größer als das Argument ist. (In diesem Fall ist die Erzeugung der Zufallszahl abhängig vom Wert des Dezimalanteils des Arguments.)

| <u>Argument</u> | <u>Untergrenze</u> | <u>Obergrenze</u> |
|-----------------|--------------------|-------------------|
| .5              | 0                  | < 1               |
| 2               | 1                  | 2                 |
| 2.5             | 1                  | 3                 |

Die erzeugte Sequenz der Zufallszahlen ist nach dem Einschalten des **PC-1421** immer gleich, weil die Startzahl immer die gleiche ist. Mit dem Befehl **RANDOM** kann eine zufällige Startzahl erzeugt werden.

1 numerischer ausdrück **ROT** numerischer ausdrück

ROT berechnet die n-te Wurzel des Arguments. 125 ROT 3 ergibt 5. ( $\sqrt[3]{125}$  wird als 125 ROT 3 eingegeben.)

1 **SGN** numerischer ausdrück

SGN ergibt einen Wert abhängig vom Vorzeichen des Arguments. Ist das Argument positiv, ist das Ergebnis 1; ist das Argument gleich Null, ist das Ergebnis 0; ist das Argument negativ, ist das Ergebnis -1. SGN -5 ist gleich -1.

1 **SIN** numerischer ausdrück

SIN berechnet den Sinus des Winkel-Arguments. Das Ergebnis ist abhängig vom Winkelmaß-Modus des PC-1421 – Altgrad, Bogenmaß oder Neugrad. SIN 30 ist gleich .5.

1 **SQR** numerischer ausdrück

SQR berechnet die Quadratwurzel des Arguments. Das Ergebnis ist identisch mit der Anwendung der Taste  auf der Tastatur. SQR 4 ist gleich 2.

1 **SQU** numerischer ausdrück

SQU berechnet das Quadrat des Arguments. SQU 3 ist gleich 9.

1 **TAN** numerischer ausdrück

TAN berechnet den Tangens des Winkel-Arguments. Das Ergebnis ist abhängig vom Winkelmaß-Modus des PC-1421 – Altgrad, Bogenmaß oder Neugrad. TAN 45 ist im Altgrad-Modus gleich 1.

1 **TEN** numerischer ausdrück

TEN berechnet die Potenz von 10 (der Basis des natürlichen Logarithmus), das Argument bildet den Exponenten.  
TEN 3 ist 1000.

## String-Funktionen

String-Funktionen dienen zum Verarbeiten von Strings. Einige Funktionen aus dieser Gruppe erfordern als Argument einen String und haben einen numerischen Wert zum Ergebnis. Bei anderen sind Argument und Ergebnis beide Strings. Dann gibt es String-Funktionen, die aus einem numerischen Wert einen String erstellen oder solche, die ein String-Argument und ein oder zwei numerische Argumente erfordern und einen String zum Ergebnis haben. Viele BASIC-Dialekte erfordern, daß das Argument in Klammern steht. Der **PC-1421** benötigt diese Klammerung nicht, außer er wenn in komplizierten Ausdrücken deutlich gemacht werden muß, welcher Teil des Ausdrucks das Argument bildet. String-Funktionen mit zwei oder drei Argumenten erfordern immer Klammern.

1 **ASC** string-ausdruck

ASC errechnet den ASCII-Code des ersten Zeichens des Arguments. Die ASCII-Zeichencode-Tabelle finden Sie in Anhang B. ASC "A" ist gleich 65.

1 **CHR\$** numerischer ausdruck

CHR\$ ist die Umkehrfunktion von ASC und errechnet zum gegebenen ASCII-Code das entsprechende Zeichen. Die ASCII-Zeichencode-Tabelle finden Sie in Anhang B. CHR\$ 65 ist "A".

1 **LEFT\$** (string-ausdruck, numerischer ausdruck)

LEFT\$ ergibt den linken Teil des String-Ausdrucks im Argument. Die Länge des Teilstrings (Anzahl Zeichen) wird durch den numerischen Ausdruck im Argument bestimmt. LEFT\$ ("ABCDEF", 2) ist gleich "AB".

1 **LEN** string-ausdruck

LEN errechnet die Länge des String-Ausdrucks im Argument (Anzahl Zeichen). LEN "ABCDEF" ist gleich 6.

1 **MID\$** (string-ausdruck, numerischer ausdruck 1, numerischer ausdruck 2)

MID\$ ergibt den mittleren Teil des String-Ausdrucks im Argument. Der erste numerische Ausdruck im Argument bezeichnet das erste Zeichen des String-Ausdrucks, das im Ergebnis enthalten sein soll. Der zweite numerische Ausdruck im Argument gibt die Länge des Ergebnisses (Anzahl Zeichen) an. MID\$ ("ABCDEF", 2, 3) ist gleich "BCD".

1 **RIGHT\$** (string-ausdruck, numerischer ausdruck)

RIGHT\$ ergibt den rechten Teil des String-Ausdrucks im Argument. Die Länge des Ergebnis-Strings (Anzahl Zeichen) wird durch den numerischen Ausdruck im Argument bestimmt. RIGHT\$ ("ABCDEF", 3) ist gleich "DEF".

1 **STR\$** numerischer ausdruck

STR\$ verwandelt den numerischen Ausdruck im Argument in einen String um. Die Umkehrfunktion zu STR\$ ist VAL. STR\$ 1.59 ist gleich "1.59".

1 **VAL** string-ausdruck

VAL ist die Umkehrfunktion zu STR\$ und verwandelt einen String in einen numerischen Ausdruck. Enthält der String-Ausdruck im Argument keinen numerischen Ausdruck, ist das Ergebnis Null. VAL "1.59" ist gleich 1.59.

Hinweis: VAL kann nur numerische Zeichen (0 bis 9), Vorzeichen (+ und -) und das "E" für Exponenten umwandeln. Enthält der String-Ausdruck andere Zeichen, werden alle Zeichen rechts des letzten erlaubten Ausdrucks ignoriert. Leerstellen werden als nicht vorhanden behandelt.

# KAPITEL 10

## PANNENHILFE

In diesem Kapitel finden Sie Hilfestellung, wenn Ihr **SHARP PC-1421** nicht das tut, was Sie von ihm erwarten. Das Kapitel ist in zwei Teile gegliedert – der erste beschäftigt sich mit der allgemeinen Bedienung des Geräts und der zweite mit BASIC-Programmierung. Für jedes Problem bringen wir einige Lösungsvorschläge. Probieren Sie diese der Reihe nach aus, bis Sie das Problem gelöst haben.

### Bedienung des Geräts

#### Wenn:

Sie schalten das Gerät ein, aber auf dem Display erscheint keine Anzeige.

Die Anzeige erscheint zwar, aber auf Betätigung der Tasten erfolgt keine Reaktion.

Sie haben eine Berechnung oder Antwort eingegeben, aber es erfolgt keine Reaktion darauf.

Während der Ausführung eines BASIC-Programms erfolgt eine Anzeige, danach stoppt das Programm.

#### Dann:

1. Steht der Ein/Aus-Schalter auf der Position ON?
2. Überprüfen Sie durch Drücken der Taste **ON** **BRK**, ob sich das Gerät automatisch ausgeschaltet hat.
3. Die Batterien auswechseln.

1. Die Anzeige durch Drücken der Taste **C-CE** löschen.
2. Die Anzeige durch Drücken von **CA** ( **SHIFT** **C-CE** ) löschen.
3. Das Gerät aus- und wieder einschalten.
4. Irgendeine Taste gedrückt halten und den RESET-Schalter drücken.
5. Den RESET-Schalter drücken, ohne eine Taste gedrückt zu halten.

1. Im RUN-Modus **ENTER** und im CAL-Modus **⇐** drücken.

1. **ENTER** drücken.

Sie haben eine Rechnung eingegeben und diese wird im BASIC-Format (Doppelpunkt nach der ersten Zahl) angezeigt.

Auf Betätigung der Tasten erfolgt keinerlei Reaktion.

1. Für Berechnungen vom PRO-Modus in den RUN-Modus umschalten.

1. Irgendeine Taste gedrückt halten und den RESET-Schalter drücken.
2. Wenn auch nach Durchführung des obigen Schritts keine Reaktion auf Betätigung der Tasten erfolgt, drücken Sie den RESET-Schalter, ohne eine Taste gedrückt zu halten. Dadurch werden jedoch alle Programme und Daten gelöscht.

### Fehlersuche in BASIC

Wenn Sie ein neues BASIC-Programm in den Computer eingeben, ist es ganz normal, daß es beim ersten Startversuch **nicht** läuft. Selbst bei einfachen Programmen, von deren Fehlerfreiheit Sie überzeugt sind wie beispielsweise die Programme in dieser Bedienungsanleitung, macht man gewöhnlich mindestens einen Tippfehler. Längere neue Programme enthalten außerdem meistens noch mindestens einen logischen Fehler. Daher wollen wir Ihnen einige grundsätzliche Hinweise geben, wie Sie solche Fehler finden und korrigieren.

Wenn Sie Ihr Programm starten und eine Fehlermeldung erhalten:

1. Gehen Sie zurück in den PRO-Modus und bringen mit den Tasten  und  die Zeile mit dem Fehler zur Anzeige. Der Cursor befindet sich dann an der Stelle, die den **PC-1421** zur Fehlermeldung verursacht hat.
2. Wenn Sie in der Programmzeile, wo der Fehler auftrat, keinen Fehler finden können, können die verwendeten Daten die Ursache sein. Beispielsweise verursacht CHR\$(A) einen Fehler, wenn A den Wert 1 hat, weil CHR\$(1) nicht zulässig ist. Die Werte der Variablen können im RUN- oder PRO-Modus überprüft werden, indem Sie die einzelnen Variablennamen und danach  eingeben.

Wenn Sie Ihr Programm starten, zwar keine Fehlermeldung erhalten, aber das Programm auch nicht das tut, was Sie von ihm erwarten:

3. Überprüfen Sie das Programm Zeile für Zeile mit LIST und den Tasten  und , um herauszufinden, ob Sie das Programm richtig eingegeben haben. Es ist verblüffend, wieviele Fehler bei einem erneuten Durchsehen eines Programms gefunden werden können!

4. Versuchen Sie beim Durchsehen des Programms, die einzelnen Zeilen so anzusehen, als wären Sie der Computer. Führen Sie die Operationen der Zeilen mit Beispielswerten durch, um herauszufinden, ob Sie die erwarteten Ergebnisse erhalten.
5. Fügen Sie zusätzliche PRINT-Statements in das Programm ein, um wichtige Werte und Zwischenwerte auf dem Display anzuzeigen. Auf diese Weise können Sie die Programmteile bestimmen, die korrekt arbeiten, und die Fehlerstelle lokalisieren. Außerdem kann man auf diese Weise feststellen, welche Teile des Programms abgearbeitet wurden. Sie können die Programmausführung auch an kritischen Punkten mit STOP unterbrechen und die Inhalte der Variablen überprüfen.
6. Benutzen Sie TRON und TROFF, entweder als Kommandos oder direkt im Programm, um den Programmablauf über mehrere Zeilen zu verfolgen. Halten Sie das Programm an kritischen Punkten an, um den Inhalt von wichtigen Variablen zu überprüfen. Diese Fehlersuchmethode ist zwar sehr langsam, aber manchmal die einzigste verbleibende Methode.

## KAPITEL 11 DIE INSTANDHALTUNG DES PC-1421

Bitte beachten Sie die folgenden Punkte, um ein störungsfreies Funktionieren des PC-1421 zu gewährleisten:

- \* Behandeln Sie den Computer immer sorgfältig, weil die Flüssigkristallanzeige aus Glas besteht.
- \* Setzen Sie den Computer nicht starken Temperaturschwankungen, Feuchtigkeit oder Staub aus. Im Sommer entstehen in geschlossenen Fahrzeugen, die der direkten Sonnenbestrahlung ausgesetzt sind, sehr hohe Temperaturen. Wird der Computer längere Zeit hohen Temperaturen ausgesetzt, können Beschädigungen die Folge sein.
- \* Verwenden Sie zur Reinigung des Computers nur ein weiches und trockenes Tuch. Benutzen Sie niemals chemische Lösungsmittel, Wasser oder ein feuchtes Tuch.
- \* Soll der Computer längere Zeit nicht benutzt werden, entnehmen Sie die Batterien, um Beschädigungen durch Auslaufen der Batterien zu vermeiden.
- \* Wenn Sie die Hilfe einer Werkstatt benötigen, geben Sie den Computer nur an eine autorisierte **SHARP**-Kundendienststelle.
- \* Wenn der Computer starker statischer Elektrizität oder starken Störsignalen ausgesetzt ist, kann er sich unter Umständen "aufhängen" (d.h. er reagiert nicht mehr auf Tastenbetätigung). Sollte dies vorkommen, halten Sie irgendeine Taste gedrückt und drücken den ALL RESET-Schalter. (Siehe unter Pannenhilfe.)
- \* Bewahren Sie diese Bedienungsanleitung auf, um jederzeit nachschlagen zu können.

# ANHANG A FEHLERMELDUNGEN

Der **PC-1421** verfügt über neun verschiedene Fehlercodes, die im folgenden erklärt werden.

**Fehler-  
nummer**

**Bedeutung**

**1**

Syntaxfehler

- Dieser Fehler bedeutet, daß der **PC-1421** Ihre Eingabe nicht verstehen kann. Überprüfen Sie Dinge wie Semikolons am Ende von PRINT-Statements, falsch geschriebene Befehle und fehlerhaften Gebrauch von Zeichen.

3\* / 2

**2**

Rechenfehler

Hier haben Sie wahrscheinlich einen der folgenden drei Fehler gemacht:

1. Versucht, eine zu große Zahl einzugeben.  
Ein Rechenergebnis größer als 9.999999999E99 erhalten.
2. Versucht, durch Null zu teilen.

5 / 0

3. Versucht, eine unzulässige Rechnung auszuführen.

LN -30 oder ANS 1.5

**3**

Dimensionierungsfehler/Argument-Fehler

- Feldvariable existiert bereits.

Sie haben versucht, ein Feld zu spezifizieren, ohne es vorher zu dimensionieren.

Der Feld-Index übersteigt die im DIM-Statement spezifizierte Feldgröße.

DIM B(256)

- Unzulässiges Funktionsargument. Das bedeutet, Sie haben etwas von Ihrem Computer verlangt, was er nicht kann.  
Beispiel: WAIT-Intervall ist größer als 65535.

WAIT 66000

4 Zeilennummer-Fehler

Hier haben Sie vermutlich einen der folgenden Fehler gemacht:

1. In GOTO, GOSUB, RUN, LIST oder THEN eine nicht existente Zeilennummer eingesetzt.
2. Eine zu große Zeilennummer verwendet. Die höchste zulässige Zeilennummer ist 65279.

5 Verschachtelungs-Fehler

Die Verschachtelung von Unterprogrammen überschreitet 10 Ebenen.

Die Verschachtelung einer FOR-Schleife überschreitet 5 Ebenen.

RETURN ohne GOSUB, NEXT ohne FOR oder READ ohne DATA.

Kapazität des Puffers überschritten.

6 Speicher-Überlauf

Dieser Fehler tritt auf, wenn Sie ein Feld deklarieren, das zu groß für den Speicher ist. Wenn ein Programm zu groß wird, kann dieser Fehler auch entstehen.

7 PRINT USING-Fehler

In einem USING-Statement ist eine unzulässige Format-Spezifikation enthalten.

8 Ein/Ausgabe-Fehler

Dieser Fehler kann nur auftreten, wenn der Drucker und/oder Cassettenrecorder (beides Sonderzubehör) an den PC-1421 angeschlossen ist. Die Fehlermeldung bedeutet, daß in der Kommunikation zwischen dem PC-1421 und dem Ein/Ausgabe-Gerät ein Problem vorhanden ist.

9 Sonstige Fehler

Dieser Fehlercode wird angezeigt, wenn der Computer ein Problem hat, das nicht unter eines der anderen acht Fehlercodes fällt. Eine der häufigsten Ursachen für diese Fehlermeldung ist der Versuch, Daten einer Variablen mit einem bestimmten Namen anzusprechen (z.B. A\$), während die Daten unter einem anderen Namen gespeichert wurden (z.B. A).

## ANHANG B ZEICHENCODE-TABELLE

Die folgende Tabelle enthält die Umwandlungswerte für die Funktionen CHR\$ und ASC. Die Spalten zeigen die ersten vier Hexadezimalzeichen bzw. die ersten vier Binärbits und die Zeilen die folgenden Hexadezimalzeichen bzw. Binärbits. Oben links in jedem Element der Tabelle ist die Dezimalzahl für das Zeichen enthalten. Unten rechts befindet sich das Zeichen selbst. Ist kein Zeichen abgebildet, ist es ein unzulässiges Zeichen für den PC-1421.

Beispielsweise hat das Zeichen "A" den Code 65 in Dezimaldarstellung, 41 in Hexadezimaldarstellung und 01000001 in Binärdarstellung. Das Zeichen "√" wird dezimal als 252, hexadezimal als FC und binär als 11111100 dargestellt.

Erste vier Bits

Der PC-1421 kann die Codes in den dunkeln Bereichen nicht verarbeiten. Wird ein Code aus einem dunklen Bereich eingegeben, resultiert ein Fehler.

Z  
w  
e  
i  
t  
e  
  
v  
i  
e  
r  
  
B  
i  
t  
s

| Hex    | 0    | 1    | 2     | 3    | 4    | 5    | 6    | 7    | 8    | E    | F    |
|--------|------|------|-------|------|------|------|------|------|------|------|------|
| Binary | 0000 | 0001 | 0010  | 0011 | 0100 | 0101 | 0110 | 0111 | 1000 | 1110 | 1111 |
| 0      | 0    | 16   | 32    | 48   | 64   | 80   | 96   | 112  | 128  | 224  | 240  |
| 0000   | NUL  |      | SPACE | 0    | @    | P    |      |      |      |      |      |
| 1      | 1    | 17   | 33    | 49   | 65   | 81   | 97   | 113  | 129  | 225  | 241  |
| 0001   |      |      | !     | 1    | A    | Q    |      |      |      |      |      |
| 2      | 2    | 18   | 34    | 50   | 66   | 82   | 98   | 114  | 130  | 226  | 242  |
| 0010   |      |      | "     | 2    | B    | R    |      |      |      |      |      |
| 3      | 3    | 19   | 35    | 51   | 67   | 83   | 99   | 115  | 131  | 227  | 243  |
| 0011   |      |      | #     | 3    | C    | S    |      |      |      |      |      |
| 4      | 4    | 20   | 36    | 52   | 68   | 84   | 100  | 116  | 132  | 228  | 244  |
| 0100   |      |      | \$    | 4    | D    | T    |      |      |      |      |      |
| 5      | 5    | 21   | 37    | 53   | 69   | 85   | 101  | 117  | 133  | 229  | 245  |
| 0101   |      |      | %     | 5    | E    | U    |      |      |      |      |      |
| 6      | 6    | 22   | 38    | 54   | 70   | 86   | 102  | 118  | 134  | 230  | 246  |
| 0110   |      |      | &     | 6    | F    | V    |      |      |      |      |      |
| 7      | 7    | 23   | 39    | 55   | 71   | 87   | 103  | 119  | 135  | 231  | 247  |
| 0111   |      |      | ,     | 7    | G    | W    |      |      |      |      |      |
| 8      | 8    | 24   | 40    | 56   | 72   | 88   | 104  | 120  | 136  | 232  | 248  |
| 1000   |      |      | (     | 8    | H    | X    |      |      |      |      |      |
| 9      | 9    | 25   | 41    | 57   | 73   | 89   | 105  | 121  | 137  | 233  | 249  |
| 1001   |      |      | )     | 9    | I    | Y    |      |      |      |      |      |
| A      | A    | 26   | 42    | 58   | 74   | 90   | 106  | 122  | 138  | 234  | 250  |
| 1010   |      |      | *     | :    | J    | Z    |      |      |      |      |      |
| B      | B    | 27   | 43    | 59   | 75   | 91   | 107  | 123  | 139  | 235  | 251  |
| 1011   |      |      | +     | ;    | K    | [    |      |      |      |      | π    |
| C      | C    | 28   | 44    | 60   | 76   | 92   | 108  | 124  | 140  | 236  | 252  |
| 1100   |      |      | ,     | <    | L    | ¥    |      |      |      |      | ✓    |
| D      | D    | 29   | 45    | 61   | 77   | 93   | 109  | 125  | 141  | 237  | 253  |
| 1101   |      |      | -     | =    | M    | ]    |      |      |      |      |      |
| E      | E    | 30   | 46    | 62   | 78   | 94   | 110  | 126  | 142  | 238  | 254  |
| 1110   |      |      | .     | >    | N    | ^    |      |      |      |      |      |
| F      | F    | 31   | 47    | 63   | 79   | 95   | 111  | 127  | 143  | 239  | 255  |
| 1111   |      |      | /     | ?    | O    | _    |      |      |      |      |      |

## ANHANG C FORMATIEREN DER DATENAUSGABE

Es ist manchmal sehr hilfreich, das Format der Datenausgabe bestimmen zu können. Beim **PC-1421** erfolgt diese mit dem Befehl **USING**, mit dem folgendes spezifiziert werden kann:

- \* Anzahl der Stellen
- \* Position des Dezimalpunktes
- \* wissenschaftliche Notation
- \* Anzahl der Zeichen in einem String

Die verschiedenen Formate werden mithilfe von "Ausgabemasken" spezifiziert. Diese Masken können String-Konstante oder String-Variable sein:

```
10: USING "####"
20: M$ = "&&&&&&"
30: USING M$
```

Wird der Befehl **USING** ohne Maske benutzt, werden alle vorher spezifizierten Formate aufgehoben.

```
40: USING
```

Das Befehlswort **USING** kann auch in einem **PRINT**-Statement auftreten:

```
50: PRINT USING M$; N
```

Ein durch **USING** spezifizierten Format bleibt gültig, bis ein neuer **USING**-Befehl auftritt.

### Numerische Masken

Eine numerische **USING**-Maske darf nur benutzt werden, um numerische Werte anzuzeigen, also numerische Konstanten oder numerische Variable. Wird eine String-Konstante oder String-Variable ausgegeben, während eine numerische **USING**-Maske gültig ist, wird die Maske ignoriert. Anzuzeigende Werte müssen immer in den Raum passen, der von der Maske bereitgestellt wird. Für das Vorzeichen muß immer eine Stelle reserviert werden, auch wenn ausschließlich positive Werte angezeigt werden. Daher kann eine Maske, die vier Anzeigestellen spezifiziert, für die Ausgabe von höchstens dreistelligen Zahlen verwendet werden.

### Spezifizierung der Stellenzahl

Die gewünschte Anzahl der Stellen wird mit dem Zeichen “#” spezifiziert. Jedes “#” in der Maske reserviert Platz für eine Stelle. Die Ausgabe oder der Ausdruck enthalten immer so viele Zeichen, wie in der Maske vorgesehen sind. Die Zahl erscheint ganz rechts in diesem Bereich, die verbleibenden Stellen links von der Zahl werden mit Leerstellen aufgefüllt. Positive Zahlen haben links immer mindestens eine Leerstelle. Da der PC-1421 nicht mehr als 10 signifikante Stellen erfassen kann, sollten in der numerischen Maske nicht mehr als 11 “#” enthalten. Überschreitet die Gesamtzahl der Stellen des ganzzahligen Teils 11, wird der ganzzahlige Teil im PC-1421 als 11 Stellen angesehen.

**HINWEIS:** In allen Beispielen in diesem Anhang wird der Anfang und das Ende der Anzeige mit einem “|” markiert, um die Größe der Anzeige zu verdeutlichen.

Statement

Anzeige

10: USING “####”

(Den PC-1421 in den RUN-Modus schalten, RUN eingeben und **ENTER** drücken.)

20: PRINT 25

| 25 |

30: PRINT -350

| -350 |

40: PRINT 1000

**ERROR 7 IN 40**

Beachten Sie, daß das letzte Statement einen Fehler verursachte, weil 5 Stellen (4 Ziffern und Vorzeichen) benötigt wurden, aber nur 4 Stellen vorhanden waren.

### Spezifizierung des Dezimalpunktes

Das Zeichen für den Dezimalpunkt “.” kann in numerische Masken aufgenommen werden, um die Position des Dezimalpunktes zu bestimmen. Wenn die Maske mehr Dezimalstellen bereitstellt, als zur Anzeige eines Werts erforderlich sind, werden die übrigen Stellen auf der rechten Seite mit Nullen aufgefüllt. Enthält der anzuzeigende Wert mehr Dezimalstellen als in der Maske vorhanden sind, werden die überschüssigen Dezimalstellen abgeschnitten (nicht gerundet).

**ANHANG C**  
**Formatieren der Datenausgabe**

| <u>Statement</u>    | <u>Anzeige</u> |
|---------------------|----------------|
| 10: USING "####.##" |                |
| 20: PRINT 25        | 25.00          |
| 30: PRINT -350.5    | -350.50        |
| 40: PRINT 2.547     | 2.54           |

**Spezifizierung der wissenschaftlichen Notation**

Ist das Zeichen “^” in der Maske enthalten, erfolgt die Anzeige in wissenschaftlicher Notation. Mit den Zeichen “#” und “.” wird das Format des “charakteristischen” Teils der Zahl spezifiziert, also des Teils, das links von E angezeigt wird. Links vom Dezimalpunkt müssen immer mindestens zwei “#” vorhanden sein, um Platz für das Vorzeichen und eine ganze Zahl zu reservieren. Der Dezimalpunkt kann enthalten sein, muß aber nicht. Rechts vom Dezimalpunkt können sich bis zu 9 “#” befinden. Nach dem “charakteristischen” Teil wird das Exponentialzeichen **E** angezeigt, darauf folgen eine Stelle für das Vorzeichen und zwei Stellen für den Exponenten. Daher wäre die kleinste Maske für wissenschaftliche Notation dargestellt durch “##^” für Anzeigen in der Form “2E 99”. Das größte Format für wissenschaftliche Notation ist “##.#####^” für die Ausgabe von Zahlen wie beispielsweise “-1.234567890E-12”.

| <u>Statement</u>    | <u>Anzeige</u> |
|---------------------|----------------|
| 10: USING "###.##^" |                |
| 20: PRINT 2         | 2.00E00        |
| 30: PRINT -365.278  | -3.65E02       |

### Spezifizierung alphanumerischer Masken

String-Konstante und -Variable werden mit dem Zeichen "&" spezifiziert. Jedes "&" bedeutet ein Ausgabe-Zeichen. Der String wird an der linken Seite des Feldes angezeigt. Ist der String kürzer als das Feld, werden die überschüssigen Stellen mit Leerstellen aufgefüllt. Wenn der String länger als das Feld ist, wird er auf die Länge des Feldes beschnitten.

| <u>Statement</u>      | <u>Anzeige</u> |
|-----------------------|----------------|
| 10: USING "&&&&&&"    |                |
| 20: PRINT "ABC"       | ABC            |
| 30: PRINT "ABCDEFGHI" | ABCDEFI        |

### Gemischte Masken

In den meisten Fällen enthalten USING-Masken entweder nur numerische Formatzeichen oder nur String-Formatzeichen. Für bestimmte Zwecke ist es jedoch günstig, beide Arten Formatzeichen in die USING-Maske aufzunehmen. In solchen Fällen bedeutet jeder Übergang von numerischen zu String-Formatzeichen und umgekehrt die Begrenzung eines Werts. Daher ist die Maske "#####&&&&" eine Spezifikation für die Anzeige zweier Werte — ein numerischer Wert mit 5 Stellen und ein String mit 4 Stellen:

| <u>Statement</u>                     | <u>Anzeige</u> |
|--------------------------------------|----------------|
| 10: PRINT USING "###.##&&"; 25; "CR" | 25.00CR        |
| 20: PRINT -5.789; "DB"               | -5.78DB        |

**Beachten Sie:** Ein USING-Statement wird auf alle folgenden Ausgaben angewendet, bis das Format durch einen anderen USING-Befehl geändert oder aufgehoben wird.

### Hinweise zum Befehl USING

- Beachten Sie beim Formatieren mit USING bitte die folgenden Punkte.
1. Wenn die Gesamtzahl der Stellen in einem USING-Format in PRINT ausdrück 16 überschreitet, wird ERROR 7 verursacht.

**ANHANG C**  
**Formatieren der Datenausgabe**

Beispiel: 10: USING "#####.#####"  
17 Stellen oder mehr

20: PRINT A

Hinweis: Wenn die Gesamtzahl der Stellen, die für den ganzzahligen Teil spezifiziert wurden, 11 einschließlich des Vorzeichens überschreitet, wird der überschüssige Teil ignoriert. Dann wird dieser ganzzahlige Teil als 11-stellig angesehen.

Beispiele:

1) 10: USING "#####.#####."  
Dieses Format verursacht keinen Fehler.

2) 10: USING "#####.####"  
Dieses Format verursacht keinen Fehler.

Ganzzahliger Teil: 11 Stellen, Dezimalpunkt: 1 Stelle und  
Dezimalteil: 4 Stellen

Gesamt: 16 Stellen (weniger als 17 Stellen)

3) 10: USING "#####.#####"  
Dieses Format verursacht ERROR 7.

(Die Gesamtzahl der Stellen ist größer als 16.)

- Im letzten Beispiel muß das USING-Format auf 16 Stellen oder weniger korrigiert werden.

2. Wenn in einem USING-Format der ganzzahlige Teil 8 Stellen überschreitet, verursacht PRINT ausdruck, ausdruck den Fehler ERROR 7.

Beispiel: 10: USING "#####.##"  
9 Stellen oder mehr

20: PRINT A, B

- Das USING-Format so ändern, daß es 8 Stellen oder weniger hat.

3. Wenn der anzuzeigende Inhalt der Form PRINT ausdruck ; ausdruck ; ausdruck ; ... 16 Stellen über schreitet, wird der überschüssige Teil nicht angezeigt.

4. Die Anzeige von Werten, die in Hexadezimalzahlen umgewandelt wurden, führt zu einem Fehler, wenn mit dem USING-Statement die Fließpunktdarstellung (Exponential-Anzeige) bestimmt wurde. (ERROR 7, Formatfehler)

## ANHANG D BEWERTUNG VON AUSDRÜCKEN UND VORRANG DER OPERATOREN

Der SHARP PC-1421 berechnet bei komplexen Ausdrücken die Teile des Ausdrucks in einer Reihenfolge, die durch den Vorrang der Operatoren bestimmt wird. Bei der Eingabe

$$100 / 5 + 45$$

– entweder als Berechnung oder in einem Programme – kann eine der beiden folgenden Darstellungen gemeint sein:

$$\frac{100}{5 + 45} = 2 \quad \text{oder} \quad \frac{100}{5} + 45 = 65$$

Der PC-1421 entscheidet zwischen diesen beiden Möglichkeiten nach den Regeln des Operatoren-Vorrangs. Weil Division eine höhere "Priorität" hat als Addition (siehe unten), wird zuerst die Division und danach die Addition ausgeführt, d.h. die zweite Formel wird berechnet, das Ergebnis ist 65.

### Operatoren-Priorität

Im BASIC-Modus des PC-1421 werden die Operatoren nach den folgenden Prioritätsregeln verarbeitet (angefangen mit der höchsten Priorität):

| Prioritätsebene | Operation                                   |
|-----------------|---------------------------------------------|
| 1.              | Klammerung                                  |
| 2.              | Variable und Pseudovvariable                |
| 3.              | Funktionen                                  |
| 4.              | Potenzierung (^), (ROT)                     |
| 5.              | Vorzeichen (+, -)                           |
| 6.              | Multiplikation und Division (*, /)          |
| 7.              | Addition und Subtraktion (+, -)             |
| 8.              | Verhältnis-Operatoren (<, <=, =, <>, >=, >) |
| 9.              | Logische Operatoren (AND, OR)               |

Sind in einem Ausdruck zwei oder mehr Operatoren der gleichen Prioritätsebene enthalten, werden sie von links nach rechts abgearbeitet. (Potenzierung wird von rechts nach links verarbeitet.) Beispielsweise ist bei  $A + B - C$  das Ergebnis gleich, wenn zuerst die Addition oder zuerst die Subtraktion ausgeführt wird.

## ANHANG D Bewertung von Ausdrücken

Enthält ein Ausdruck mehrfache Klammerung, wird zuerst das innerste Klammersymbol berechnet und das äußerste zuletzt.

Bei Ebene 3 und 4 hat die letzte Eingabe die höhere Priorität.

$$\begin{aligned} \text{Beispiel: } & -2 \wedge 4 \rightarrow -(2^4) \\ & 3 \wedge -2 \rightarrow 3^{-2} \end{aligned}$$

### Beispiel

Wir wollen die Arbeitsweise des **PC-1421** an dem folgenden Ausdruck verfolgen:

$$( (3+5-2) * 6+2) / 10 \wedge \text{LOG } 100$$

Der **PC-1421** berechnet zunächst die innerste Klammer. Da "+" und "-" die gleiche Priorität haben, wird von links nach rechts verarbeitet und die Addition zuerst ausgeführt:

$$( (8-2) * 6+2) / 10 \wedge \text{LOG } 100$$

Danach die Subtraktion:

$$( (6) * 6+2) / 10 \wedge \text{LOG } 100$$

oder:

$$(6 * 6+2) / 10 \wedge \text{LOG } 100$$

In der nächsten Klammer wird zuerst die Multiplikation durchgeführt:

$$(36+2) / 10 \wedge \text{LOG } 100$$

Und danach die Addition:

$$(38) / 10 \wedge \text{LOG } 100$$

oder:

$$38 / 10 \wedge \text{LOG } 100$$

Nach Verarbeitung der Klammern hat jetzt die LOG-Funktion die höchste Priorität und wird als nächstes berechnet:

$$38 / 10 \wedge 2$$

Als nächstes die Potenzierung:

$$38 / 100$$

Und zuletzt die Division:

$$0.38$$

Dies ist der Wert des Ausdrucks.

## ANHANG E TASTEN-FUNKTIONEN IM BASIC-MODUS

ON  
BRK

(ON)

Zum Einschalten des PC-1421, wenn er sich automatisch ausgeschaltet hat.

(BREAK)

- Die Betätigung dieser Taste während der Programmausführung bewirkt ein Unterbrechen des Programms.
- Bei manuellen Operationen, Ein/Ausgabe-Kommandos wie BEEP, CLOAD usw. wird durch Betätigung dieser Taste die Kommando-Ausführung unterbrochen.

SHIFT

- Mit dieser gelben Taste mit der Aufschrift "SHIFT" werden bei Tasten mit Doppelbelegung die zweiten Funktionen, die in braun über den Tasten angegeben sind, angesprochen.

Beispiel: SHIFT ? U → ? wird eingegeben.

C-CE

- Zum Löschen der Eingabe und des Displays. (Freigabe von Fehlern.)

SHIFT CA

- Löscht nicht nur das Display, sondern initialisiert auch den Computer.
  - Initialisierung –
  - Rückstellung des WAIT-Zeitgebers.
  - Rückstellung des Anzeigeformats (USING-Format).
  - Rückstellung des TRON-Status (TROFF).
  - Rückstellung von PRINT = LPRINT.
  - Rückstellung von Fehlern.

0 ~ 9

- Numerische Tasten.

.

- Dezimalpunkt.
- Dient zur Eingabe von Abkürzungen für Kommandos/Befehlen/Funktionen.
- Zur Bestimmung des Dezimalpunktes im USING-Format.

E

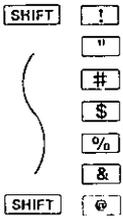
- Zur Kennzeichnung des Exponenten in wissenschaftlicher Notation. (Diese Taste ist der Buchstabe E.)

EXP

- Zur Zuweisung des Exponenten in wissenschaftlicher Notation.

**ANHANG E**  
**Tasten-Funktion im BASIC-Modus**

- /
  - Division-Taste.
  
- \*
  - Multiplikation-Taste.
  - Zur Kennzeichnung von Feldvariablen in INPUT#, PRINT# usw.
  
- +
  - Addition-Taste.
  
- - Subtraktion-Taste.
  
- SHIFT ?
  - Zur Eingabe von CLOAD?.
  
- SHIFT :
  - Zur Trennung von zwei oder mehr Statements in einer Programmzeile.
  
- ,
  - Bezeichnet eine Pause zwischen zwei Gleichungen und zwischen Variablen und Kommentaren.
  
- SHIFT ;
  - Zur Anzeige von zwei oder mehr Werten oder Inhalten zur gleichen Zeit.
  - Schafft eine Pause zwischen Instruktion und Variable.
  
- =
  - Weist in Zuweisungs-Statements den Inhalt (Zahl oder Zeichen) an der rechten Seite der Variablen an der linken Seite zu.
  - Zur Eingabe von logischen Operatoren in IF-Anweisungen.
  
- DEF
  - Wird eine der folgenden 18 Tasten (A, S, D, F, G, H, J, K, L, Z, X, C, V, B, N, M,  $\rightarrow$ , SPC) nach Drücken der Taste DEF gedrückt, so startet das Programm mit dem entsprechenden Kennbuchstaben.
  
- A ~ Z
  - Alphabetische Tasten. Diese Tasten sind die gleichen wie bei einer Schreibmaschine. Auf dem Display des **PC-1421** werden die Zeichen als Großbuchstaben dargestellt.
  
- ( , )
  - Zur Eingabe von Klammern.
  
- SHIFT < }  
SHIFT > }
  - Zur Eingabe von logischen Operatoren in IF-Anweisungen.
  
- SPC
  - Zur Eingabe von Leerstellen.
  
- SHIFT ^
  - Potenzierung-Zeichen.
  - Zur Spezifizierung des Gleitpunkt-Systems (Exponent-System) für numerische Daten in USING-Statements.
  
- √
  - Quadratwurzel-Zeichen.



- Zum Eingeben der entsprechenden Symbole.
- " : • Zum Kennzeichnen und Löschen von Zeichen.  
• Zum Spezifizieren von Kennbuchstaben.
- # : Zur Bestimmung des Anzeigeformats von numerischen Daten in USING-Statements.
- \$ : • Zur Bestimmung von String-Variablen.
- & : • Zur Bestimmung des Anzeigeformats von String-Daten in USING-Statements.  
• Zur Bezeichnung von Hexadezimalzahlen.



- Bewegt den Cursor nach rechts (für eine Stelle einmal drücken, für rasche Vorwärtsbewegung gedrückt halten)
- Zur Ausführung von Wiedergabe-Anweisungen.
- Löscht Fehlermeldungen im manuellen Betrieb.



- Bewegt den Cursor nach links (für eine Stelle einmal drücken, für rasche Rückwärtsbewegung gedrückt halten)
- Zur Ausführung von Wiedergabe-Anweisungen.
- Löscht Fehlermeldungen im manuellen Betrieb.



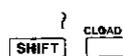
- Fügt eine Leerstelle (das Zeichen " " erscheint) zwischen dem Zeichen, auf dem der Cursor steht, und dem Zeichen davor ein.



- Löscht den Inhalt der Stelle, auf der der Cursor steht.



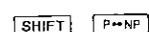
- Zum Voreinstellen von Kommando- und Statement-Tasten. Durch Drücken von **SHIFT** und dann der **alphabetischen** Tasten (einschließlich Komma, Leerstelle und **ENTER**) **unter dem gewünschten Kommando oder Statement wird das so gewählte Kommando oder Statement in den PC-1421 eingegeben.**



- Zum Einstellen des CAL-Modus.



- Zum Einstellen des RUN-Modus, wenn der CAL-Modus eingestellt ist.  
Zum Einstellen des PRO-Modus, wenn der RUN-Modus eingestellt ist.  
Mit jedem Drücken der Taste **BASIC** wird zwischen RUN- und PRO-Modus hin- und hergeschaltet.



- Zum Ein- und Ausschalten des Druck-Modus, wenn ein Drucker (Sonderzubehör) an den **PC-1421** angeschlossen ist.

## ANHANG E

### Tasten-Funktion im BASIC-Modus

|                |       |
|----------------|-------|
| $\sqrt{\quad}$ |       |
| $y^x$          | $1/x$ |
| SHIFT          | $x^2$ |
| SHIFT          | $n!$  |
| SHIFT          | $e^x$ |
| ENTER          |       |

- Zur Eingabe der entsprechenden Funktion.
- Eingabe von Programmzeilen in den Computer.
- Wird beim Schreiben von Programmen benötigt.
- Bewirkt manuelle Berechnung oder direkte Ausführung von Kommando-Statements durch den Computer.
- Zum Fortsetzen der Programmausführung nach Eingabe von Daten, die durch ein INPUT-Statement abgefragt wurde, oder nach Ausführung eines PRINT-Statements.

Die Tasten  und  haben die folgenden Funktionen, abhängig vom Modus und Status des Computers.

| Modus | Status                                                                                    |  |                               |
|-------|-------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------|
| RUN   | Programm wird ausgeführt                                                                  | /                                                                                 | /                                                                                                              |
|       | Programm ist vorübergehend unterbrochen                                                   | Zur Ausführung der nächsten Zeile                                                 | Zur Anzeige von Programmzeilen, die ausgeführt werden oder schon ausgeführt sind, diese Taste gedrückt halten. |
|       | INPUT-Statement wird ausgeführt                                                           |                                                                                   |                                                                                                                |
|       | PRINT-Statement wird ausgeführt                                                           |                                                                                   |                                                                                                                |
|       | Unterbrechung                                                                             |                                                                                   |                                                                                                                |
|       | Fehler während Programmausführung                                                         | /                                                                                 | Diese Taste gedrückt halten, um die Zeile mit dem Fehler anzuzeigen.                                           |
|       | TRON eingeschaltet                                                                        | Für Fehlersuche                                                                   | Zur Anzeige von Programmzeilen, die ausgeführt werden oder schon ausgeführt sind, diese Taste gedrückt halten. |
|       | Sonstiger Status                                                                          | Zur Anzeige eines vorher berechneten Ergebnisses (Letzte-Antwort-Funktion)        | Wie links                                                                                                      |
| PRO   | (Wenn der Modus von RUN auf PRO umgeschaltet wird und keine Programmzeile angezeigt wird) |                                                                                   |                                                                                                                |
|       | Programm ist vorübergehend unterbrochen                                                   | Zur Anzeige der Zeile, in der unterbrochen wurde.                                 | Wie links                                                                                                      |
|       | Fehler                                                                                    | Zur Anzeige der Zeile mit dem Fehler.                                             | Wie links                                                                                                      |
|       | Sonstiger Status                                                                          | Zur Anzeige der ersten Zeile                                                      | Zur Anzeige der letzten Zeile                                                                                  |
|       | (Wenn eine Programmzeile angezeigt wird)                                                  |                                                                                   | Zur Anzeige der nächsten Programmzeile.                                                                        |

## Tasten für finanzielle/statistische Berechnungen

**COMP** : **Berechnungstaste**  
Wird diese Taste unmittelbar vor einer der Tasten **CST** , **SEL** , **MAR** , **SHIFT** **MU** , **i** , **PV** , **FV** , **PMT** oder **PMT** gedrückt, wird die folgende Berechnung ausgeführt:

Normaler Modus

- COMP** **CST** : berechnet Kosten.
- COMP** **SEL** : berechnet Verkaufspreis.
- COMP** **MAR** : berechnet Bruttohandelsspanne.
- COMP** **SHIFT** **MU** : berechnet Kalkulationsaufschlag.

FIN-Modus

- COMP** **n** : berechnet die Anzahl der Perioden.
- COMP** **i** : berechnet den Zinssatz.
- COMP** **PV** : berechnet den Gegenwartswert der Investition.
- COMP** **FV** : berechnet den Zukunftswert der Investition.
- COMP** **PMT** : berechnet die periodische Zahlung.

**SHIFT** **RCL** : **Abruftaste**  
Zum Abrufen der gespeicherten Werte für finanzielle Berechnungen usw.

**AMRT** : **Amortisation-Taste**  
Für Amortisations-Berechnungen. Ergibt Kapital, Zinsen und Restkapital.

**SHIFT** **STO** : **Speichertaste**  
Zum Speichern von Daten in CFi oder Ni beim Ändern der Daten bei Kapitalfluß-Berechnungen.

**P<sub>1</sub>/P<sub>2</sub>** : **P<sub>1</sub>/P<sub>2</sub>-Taste**  
Kennzeichnet die Anfangs- und End-Periode für Kapitalansammlung.

**SHIFT** **NPV** : **Taste für gegenwärtigen Nettowert**  
Vergleicht den gegenwärtigen Wert jedes zukünftigen Kapitalflusses mit der Original-Investition.

**ACC** : **Kapitalansammlung-Taste**  
Dieser Taste muß immer P<sub>1</sub>/P<sub>2</sub> vorangehen. Ergibt das aufgelaufene Kapital und die aufgelaufenen Zinsen für die Periode.

- SHIFT IRR** : **Taste für interne Rückkehrtrate**  
Zur Bestimmung der internen Rückkehrtrate von Kapitalflüssen: Gewinn.
- CFj** : **Kapitalfluß-Eingabetaste**  
Zur Eingabe von bis zu 20 Kapitalfluß-Beträgen (ein Kapitalfluß kann – muß aber nicht – Mehrfach-Kapitalflüsse unter Verwendung der Ni-Funktion enthalten).
- SHIFT Ni** : **Kapitalfluß-Eingabetaste**  
Zur Eingabe der Anzahl von bis zu 99 aufeinanderfolgenden identischen Kapitalflüssen.
- n** : **Taste für Anzahl der Perioden**  
Zur Zuweisung oder Berechnung der Anzahl der Perioden bei finanziellen Berechnungen.
- SHIFT x12** : **x 12 Taste**  
Zur Berechnung der Anzahl der zusammensetzenden Perioden (monatlich) bei finanziellen Berechnungen.
- i** : **Zins-Taste**  
Zur Zuweisung oder Berechnung des Zinssatzes bei finanziellen Berechnungen.
- SHIFT ÷12** : **÷ 12 Taste**  
Zur Berechnung des Zinssatzes pro zusammensetzende Periode (monatlich) bei finanziellen Berechnungen.
- PV** : **Taste für gegenwärtigen Wert**  
Zur Zuweisung oder Berechnung des Anfangswerts einer Investition bei finanziellen Berechnungen.
- SHIFT APR** : **Jahresprozentsatz-Umwandlungstaste**  
Zur Umwandlung der effektiven Jahresrate (EFF) in die Jahresprozentrate (APR).
- FV** : **Taste für zukünftigen Wert**  
Zur Zuweisung oder Berechnung des zukünftigen Werts einer Investition bei finanziellen Berechnungen.
- SHIFT EFF** : **Umwandlungstaste für effektive Jahresrate**  
Zur Umwandlung der Jahresprozentrate (APR) in die effektive Jahresrate (EFF).

## ANHANG E

### Tasten-Funktion im BASIC-Modus

- [PMT]** : **Zahlung-Taste**  
Zur Zuweisung oder Berechnung der periodischen Bezahlung einer Amortisationsanleihe oder von Jahreszahlungen.
- [CST]** : **Kosten-Taste**  
Zur Zuweisung oder Berechnung von Kosten.
- [SHIFT] [BGN]** : **BGN-Taste**  
Bestimmt, ob Berechnungen mit dem Beginn oder dem Ende jedes Monats durchgeführt werden.
- [SEL]** : **Verkaufspreis-Taste**  
Zur Zuweisung oder Berechnung von Verkaufspreisen.
- [SHIFT] [TAB]** : **Tabulator-Taste**  
Mit den Tasten **[SHIFT] [TAB]** wird die Anzahl der Dezimalstellen des Rechenergebnisses bestimmt. Dafür wird eine der numerischen Tasten ( **[0]** ~ **[9]** ) entsprechend der Anzahl Stellen nach dem Dezimalpunkt nach den Tasten **[SHIFT] [TAB]** gedrückt. Für Fließpunkt-Berechnungen die Taste **[.]** nach **[SHIFT] [TAB]** drücken. Der Übertrag wird automatisch gerundet.  
Die Zuweisung der Dezimalstellen bleibt erhalten, auch wenn das Gerät ausgeschaltet wird.  
Zuerst drücken: **[SHIFT] [TAB] [.]**  
Eingeben: **[C-CE] 1.23456789 [=]**  
Anzeige: 1.23456789  
Drücken: **[SHIFT] [TAB] [3]**  
Anzeige: 1.235  
Drücken: **[SHIFT] [TAB] [9]**  
Anzeige: 1.23456789
- [MAR]** : **Handelsspanne-Taste**  
Zur Zuweisung oder Berechnung der Bruttohandelsspanne.
- [SHIFT] [MU]** : **Aufschlag-Taste**  
Zur Zuweisung oder Berechnung des Aufschlags.
- [SHIFT] [I(360)]** : **Tage-I-Taste**  
Berechnet die Anzahl Tage mit 360 Tage/Jahr.
- [SHIFT] [II(365)]** : **Tage-II-Taste**  
Berechnet die Anzahl Tage mit 365 Tage/Jahr.

- SHIFT**  **$n$**  : **STAT-Modus** (Wenn der Rechner auf den statistischen Berechnungs-Modus eingestellt ist.)  
Dient zur Errechnung der Anzahl der eingegebenen Stichprobenwerte.
- SHIFT**  **$\Sigma xy$**  : **STAT-Modus**  
Zur Errechnung der Summe der Produkte der X- und Y-Daten bei statistischen Berechnungen mit zwei Variablen.
- SHIFT**  **$\Sigma y$**  : **STAT-Modus**  
Zur Errechnung der Summe der Y-Daten.
- SHIFT**  **$\Sigma y^2$**  : **STAT-Modus**  
Zur Errechnung der Summe der Quadrate der X-Daten.
- SHIFT**  **$\bar{x}$**  : **STAT-Modus**  
Zur Errechnung des Mittelwerts der X-Daten.
- SHIFT**  **$Sx$**  : **STAT-Modus**  
Zur Errechnung der Standardabweichung ( $Sx$ ) der X-Stichproben-  
daten.
- SHIFT**  **$\sigma x$**  : **STAT-Modus**  
Zur Errechnung der Standardabweichung ( $\sigma x$ ) der X-Populations-  
daten.
- SHIFT**  **$\bar{y}$**  : **STAT-Modus**  
Zur Errechnung des Mittelwerts der Y-Daten.
- SHIFT**  **$Sy$**  : **STAT-Modus**  
Zur Errechnung der Standardabweichung ( $Sy$ ) der Y-Stichproben-  
daten.
- SHIFT**  **$\sigma y$**  : **STAT-Modus**  
Zur Errechnung der Standardabweichung ( $\sigma y$ ) der Y-Populations-  
daten.
- SHIFT**  **$\Sigma x$**  : **STAT-Modus**  
Zur Errechnung der Summe der X-Daten.
- SHIFT**  **$\Sigma x^2$**  : **STAT-Modus**  
Zur Errechnung der Summe der Quadrate der X-Daten.
- SHIFT**  **$a$**  : **STAT-Modus**  
Zur Errechnung der Konstanten a der linearen Regressionsgleichung  
 $y = a + bx$ .

## ANHANG E

### Tasten-Funktion im BASIC-Modus

**SHIFT** **b** : **STAT-Modus**

Zur Errechnung des Koeffizienten B der linearen Regressionsgleichung  $y = a + bx$ .

**SHIFT** **y'** : **STAT-Modus**

Zur Errechnung des Schätzwertes  $y$  aus  $y = a + bx$ .

**SHIFT** **r** : **STAT-Modus**

Zur Errechnung des Korrelationskoeffizienten zwischen den X- und Y-Daten.

**SHIFT** **x'** : **STAT-Modus**

Zur Errechnung des Schätzwertes  $x$  aus  $y = a + bx$ .

**CD** : **STAT-Modus**

Zur Korrektur von fehlerhaften Eingaben.

**(x,y)** : **STAT-Modus**

Zur Unterscheidung der X- und Y-Daten bei statistischen Berechnungen mit zwei Variablen.

**DATA** : **STAT-Modus**

Zur Eingabe von Daten bei statistischen Berechnungen mit einer oder mit zwei Variablen.

## ANHANG F TECHNISCHE DATEN

|                            |                                                                                                                                                                                                                                                                                                                                                                                                   |
|----------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Model:</b>              | PC-1421 Business/Finanz-Computer                                                                                                                                                                                                                                                                                                                                                                  |
| <b>Prozessor:</b>          | 8-Bit-CMOS-CPU                                                                                                                                                                                                                                                                                                                                                                                    |
| <b>Programmiersprache:</b> | BASIC                                                                                                                                                                                                                                                                                                                                                                                             |
| <b>Speicherkapazität:</b>  | System-ROM: 40 KByte                                                                                                                                                                                                                                                                                                                                                                              |
|                            | RAM:                                                                                                                                                                                                                                                                                                                                                                                              |
|                            | System Ca. 600 Byte                                                                                                                                                                                                                                                                                                                                                                               |
|                            | Benutzer                                                                                                                                                                                                                                                                                                                                                                                          |
|                            | Fester Speicherbereich 208 Byte<br>(A ~ Z, A\$ ~ Z\$)                                                                                                                                                                                                                                                                                                                                             |
|                            | Programm/Daten-Bereich 3454 Byte                                                                                                                                                                                                                                                                                                                                                                  |
| <b>Kellerspeicher:</b>     | Unterprogramme: 10 Ebenen                                                                                                                                                                                                                                                                                                                                                                         |
|                            | Funktionen: 16 Ebenen                                                                                                                                                                                                                                                                                                                                                                             |
|                            | FOR/NEXT-Schleifen: 5 Ebenen                                                                                                                                                                                                                                                                                                                                                                      |
|                            | Daten: 8 Ebenen                                                                                                                                                                                                                                                                                                                                                                                   |
| <b>Operatoren:</b>         | Addition, Subtraktion, Multiplikation, Division, trigonometrische und inverse trigonometrische Funktionen, logarithmische und Exponential-Funktionen, Winkelumwandlung, Quadrat und Quadratwurzel, Kubikwurzel, hyperbolische und inverse hyperbolische Funktionen, Koordinatenumwandlung, n-te Wurzel, Signum, Absolutwert, ganzzahliger Betrag, Verhältnis-Operatoren, logische Operatoren usw. |
| <b>Rechengenauigkeit:</b>  | 10 Stellen (Mantisse) + 2 Stellen (Exponent)                                                                                                                                                                                                                                                                                                                                                      |
| <b>Editiermöglichkeit:</b> | Cursor nach links und rechts, Zeile aufwärts und abwärts, Zeicheneinfügung, Löschen von Zeichen                                                                                                                                                                                                                                                                                                   |
| <b>Speicherschutz:</b>     | CMOS, batteriegestützt                                                                                                                                                                                                                                                                                                                                                                            |
| <b>Leuchtanzeige:</b>      | 16-Stellen-Flüssigkristallanzeige mit 5 x 7-Punkt-Zeichen                                                                                                                                                                                                                                                                                                                                         |
| <b>Tastatur:</b>           | 76 Tasten: alphabetisch, numerisch, Sonderzeichen und Funktionen. Numerische Schablone. Definierbare Tastenfunktionen.                                                                                                                                                                                                                                                                            |
| <b>Stromversorgung:</b>    | 6,0 V Gleichstrom: Lithium-Zellen<br>Typ: CR-2032 x 2                                                                                                                                                                                                                                                                                                                                             |

**ANHANG F**  
**Technische Daten**

|                        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Stromverbrauch:</b> | 6,0 V Gleichstrom, 0,03 W<br>Ca. 300 Stunden bei kontinuierlicher Anzeige von 55555555. bei einer Betriebstemperatur von 20°C. Diese Betriebszeit kann abhängig von der Arbeitsmethode usw. geringfügig schwanken. <ul style="list-style-type: none"><li>• Bei einer Stunde Betrieb pro Tag kann die Batterie etwa 4 Monate benutzt werden. Dies gilt für einstündigen Betrieb, wovon 10 Minuten für Berechnungen oder Programmausführung und 50 Minuten für Anzeige genutzt werden.</li></ul> |
| <b>Abmessungen:</b>    | 170(B) x 72(T) x 9,5(H) mm                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| <b>Gewicht:</b>        | Ca. 150 g (mit Batterien und Schutzdeckel)                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| <b>Zubehör:</b>        | Deckel, zwei Lithium-Zellen (eingesetzt), Tastatur-Schablone und Bedienungsanleitung                                                                                                                                                                                                                                                                                                                                                                                                           |
| <b>Sonderzubehör:</b>  | Drucker/Cassettenrecorder-Interface <b>CE-126P</b>                                                                                                                                                                                                                                                                                                                                                                                                                                             |

# INHALTSVERZEICHNIS

|                                | (Seite) |
|--------------------------------|---------|
| DISKONTIERTE ZAHLUNGEN.....    | 212     |
| KAUF ODER LEASING.....         | 217     |
| RENDITE.....                   | 225     |
| EFFEKTIV- UND NOMINALZINS..... | 229     |
| TILGUNGSPLAN.....              | 233     |
| DARLEHENSRÜCKZAHLUNG.....      | 237     |
| ABSCHREIBUNG.....              | 242     |
| RATENKREDIT.....               | 250     |
| T-TEST.....                    | 254     |
| CHI-QUADRAT-TEST.....          | 258     |
| SIMPSON-REGEL.....             | 262     |
| POLYNOMAUSWERTUNG.....         | 266     |
| MATRIZENMULTIPLIKATION.....    | 270     |
| QUADRATISCHE GLEICHUNGEN.....  | 275     |
| WURZEL EINER GLEICHUNG.....    | 279     |

## ÜBERBLICK

Bei Investitionen spielt die Analyse diskontierter Zahlungsströme eine Rolle. Die meistverwendeten Methoden hierzu sind die Bestimmung des Nettobarwertes (NPV) und des internen Zinssatzes (IRR).

## INHALT (Formel)

Ist der Nettobarwert (NPV) kleiner als 0, so liegt die tatsächliche Verzinsung unter der gewünschten Rendite, ist er größer als 0, so liegt sie darüber.

Der interne Zinssatz IRR gibt die tatsächliche Rendite an:

Formeln:      NPV = Nettobarwert  
                    $CF_j$  = Zahlung in Periode j  
                   i     = erwartete (erwünschte) Rendite

$$NPV = CF_0 + \frac{CF_1}{(1+i)^1} + \frac{CF_2}{(1+i)^2} + \dots + \frac{CF_n}{(1+i)^n}$$

$$CF_0 = \sum_{j=1}^k CF_j \left[ 1 - \frac{(1+IRR)^{-n_j}}{IRR} \right] \left[ (1+IRR)^{-\sum_{i < j} n_i} \right]$$

## ANLEITUNG ZUM BETRIEB

1. Das Programm wird mit **DEF** **A** gestartet.
2. Eingabe von n, i,  $CF_j$  (I),  $N_j$  (I)
3. Ausgabe von NPV, IRR

**Bemerkung:** Beachten Sie, daß Zahlungsausgänge negativ einzugeben sind.

## BEISPIEL

Es besteht die Möglichkeit, eine Eigentumswohnung für DM 180.000,- zu kaufen. Ein Investor möchte eine Rendite von mindestens 6% erzielen. Er erwartet nach 6 Jahren einen Wiederverkauf der Wohnung für DM 230.000,- und erwartet folgende Kosten bzw. Einnahmen:

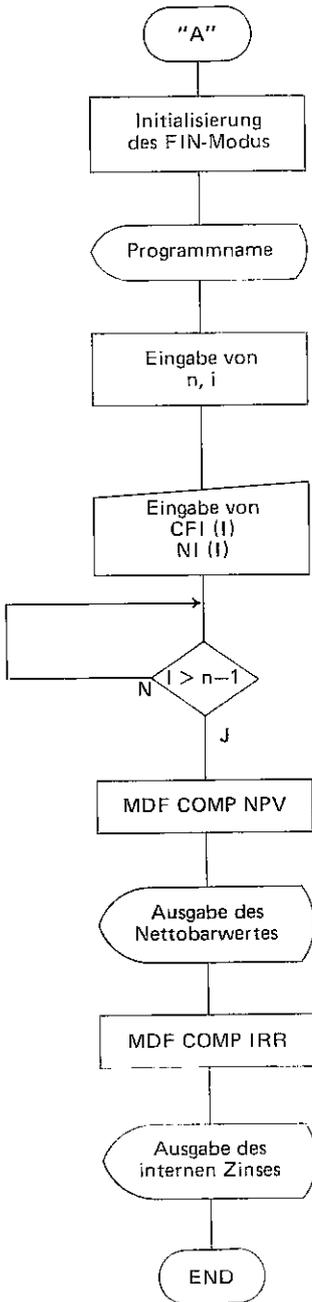
| Jahr | Zahlungen |
|------|-----------|
| 1    | -5.000    |
| 2    | 4.500     |
| 3    | 5.500     |
| 4    | 5.500     |
| 5    | 4.500     |
| 6    | 230.000   |

Der Investor möchte wissen, ob er die erwartete Rendite erzielen wird oder nicht.

# TASTENBETÄTIGUNG

| Schritt-Nr. | Eingabe              | Anzeige         | Bemerkungen                                        |
|-------------|----------------------|-----------------|----------------------------------------------------|
| 1           | <b>DEF</b> A         | NPV/IRR<br>n =  | n eingeben                                         |
| 2           | 6 <b>ENTER</b>       | i =             | i eingeben                                         |
| 3           | 6 <b>ENTER</b>       | CFI (0.) =<br>? | Investitionsbetrag eingeben                        |
| 4           | -180000 <b>ENTER</b> | NI (0.) =<br>?  | Häufigkeit dieses Betrages in einer Folge eingeben |
| 5           | 1 <b>ENTER</b>       | CFI (1.) =<br>? | 1. folgende Zahlung eingeben                       |
| 6           | -5000 <b>ENTER</b>   | NI (1.) =<br>?  | Häufigkeit von CFI(1) eingeben                     |
| 7           | 1 <b>ENTER</b>       | CFI (2.) =<br>? |                                                    |
| 8           | 4500 <b>ENTER</b>    | NI (2.) =<br>?  |                                                    |
| 9           | 1 <b>ENTER</b>       | CFI (3.) =<br>? |                                                    |
| 10          | 5000 <b>ENTER</b>    | NI (3.) =<br>?  | Dieser Zahlungseingang erfolgt 2mal hintereinander |
| 11          | 2 <b>ENTER</b>       | CFI (4.) =<br>? |                                                    |
| 12          | 4500 <b>ENTER</b>    | NI (4.) =<br>?  |                                                    |
| 13          | 1 <b>ENTER</b>       | CFI (5.) =<br>? |                                                    |
| 14          | 230000 <b>ENTER</b>  | NI (5.) =<br>?  |                                                    |
| 15          | 1 <b>ENTER</b>       | NPV = -6233.99  | Ausgabe des Nettobarwertes                         |
| 16          | <b>ENTER</b>         | IRR = 5.37      | Ausgabe des internen Zinssatzes                    |
| 17          | <b>ENTER</b>         | >               | Programmende                                       |

# FLUSSDIAGRAMM



## PROGRAMMLISTING

```

10:"A":CLEAR:ERASE FIN
20:WAIT 100:PRINT " NPV
 / IRR "
30:INPUT " n ="; n," i
 ="; i
40:FOR I=0 TO n -1
50:PRINT " CFI (";I;")=
 "
60:INPUT CFI (I)
70:PRINT " NI (";I;")="
80:INPUT NI (I)
90:NEXT I
100:WAIT :USING "####.#
 #":PRINT " NPV =";
 MDF COMP NPV
110:PRINT " IRR =";MDF
 COMP IRR
120:END

```

## SPEICHERINHALT

|   |   |
|---|---|
| A |   |
| B |   |
| C |   |
| D |   |
| E |   |
| F |   |
| G |   |
| H |   |
| I | ✓ |
| J |   |
| K |   |
| L |   |
| M |   |
| N |   |
| O |   |
| P |   |
| Q |   |
| R |   |
| S |   |
| T |   |
| U |   |
| V |   |
| W |   |
| X |   |
| Y |   |
| Z |   |

## ÜBERBLICK

Der Vergleich der Bruttokonten vor Steuern einer Kauffinanzierung mit denen von Leasing – etwa der Effektivzinssätze – sagt noch nichts über die Vorteilhaftigkeit einer der beiden Alternativen aus. Vielmehr muß die unterschiedliche steuerliche (vor allem gewerbsteuerliche) Behandlung berücksichtigt werden.

## INHALT

Bei gleichbleibenden Annuitäten errechnet sich der Barwert eines Zahlungsflusses gemäß

$$FV = PV (1 + i)^n - PMT \frac{(1 + i)^n - 1}{i}$$

wobei

- FV    den Endwert
- PV    den Barwert
- i     den Zinssatz pro Abrechnungsperiode
- PMT  die Annuität
- n     die Anzahl der Abrechnungsperioden

bezeichnet.

Bei nicht-gleichbleibenden Annuitäten gilt:

$$FV_j = PV_{j+1} (1 + i)^{n_{j+1}} - PMT \frac{(1 + i)^{n_{j+1}} - 1}{i}$$

mit  $FV_0 = FV$ ,  $FV_{j+1} = -PV_{j+1}$

Die Restbuchwerte einer Abschreibung ergeben sich aus:

$$R_j = R_{j-1} - A_j \text{ mit } R_0 = C,$$

wobei C den Anschaffungspreis und  $A_j$  die j-te Abschreibung bezeichne.

Bei einer Kauffinanzierung ist die Körperschafts- und Gewerbeertragssteuer auf den Barwert der Abschreibung zu berechnen und vom Barwert der Finanzierung (100%) abzusetzen.

Ferner ist die Gewerbekapitalsteuer und (bei (teilweiser) Finanzierung durch Eigenkapital) die Vermögenssteuer auf den Barwert der Buchwerte den Finanzierungskosten aufzuschlagen.

Beim Leasing sind die Zahlungen mit dem Nettozins zu diskontieren und um die Gewerbeertrags- und Körperschaftssteuerersparnis zu mindern.

Die so ermittelten Nettokosten von Kauf und Leasing sind zu vergleichen.

## ANLEITUNG ZUM BETRIEB

1. Programm mit **DEF** **A** starten.

2. Eingabe der Steuersätze:

|   |                        |
|---|------------------------|
| K | (Körperschaftssteuer)  |
| G | (Gewerbekapitalsteuer) |
| E | (Gewerbeertragssteuer) |
| V | (Vermögenssteuer)      |

3. Eingabe der Finanzierungsdaten für Kauf:

|      |                                       |
|------|---------------------------------------|
| N    | (Lebensdauer/Nutzungsdauer)           |
| C    | (Eigenkapitalanteil)                  |
| R    | (Eigenkapitalrendite)                 |
| Z    | (Effektivzins der Fremdfinanzierung)  |
| A(I) | (Abschreibung – vorher zu berechnen!) |

4. Eingabe der Leasingdaten:

|             |                              |
|-------------|------------------------------|
| M           | (Dauer des Leasing)          |
| L bzw. L(I) | (Leasingraten)               |
| H bzw. H(I) | (Häufigkeit der Leasingrate) |
| W           | (Restwert)                   |

5. Ausgabe der Nettokosten für Kauf und Leasing.

## BEISPIEL

Es ist folgender Leasingvertrag über 54 Monate

|                   |      |
|-------------------|------|
| 1. bis 12. Monat  | 3,2% |
| 13. bis 24. Monat | 2,5% |
| 25. bis 54. Monat | 2,0% |
| Restwert          | 10%  |

mit folgender Kauffinanzierung zu vergleichen:

|                     |     |
|---------------------|-----|
| Kreditzins          | 12% |
| Eigenkapital        | 20% |
| Eigenkapitalrendite | 10% |

Nutzungsdauer 5 Jahre, lineare Abschreibung, Kauf in der ersten Jahreshälfte.

Es werden folgende Steuersätze zugrundegelegt:

|                      |      |
|----------------------|------|
| Körperschaftssteuer  | 56%  |
| Gewerbeertragssteuer | 15%  |
| Gewerbekapitalsteuer | 0,6% |
| Vermögenssteuer      | 0,7% |

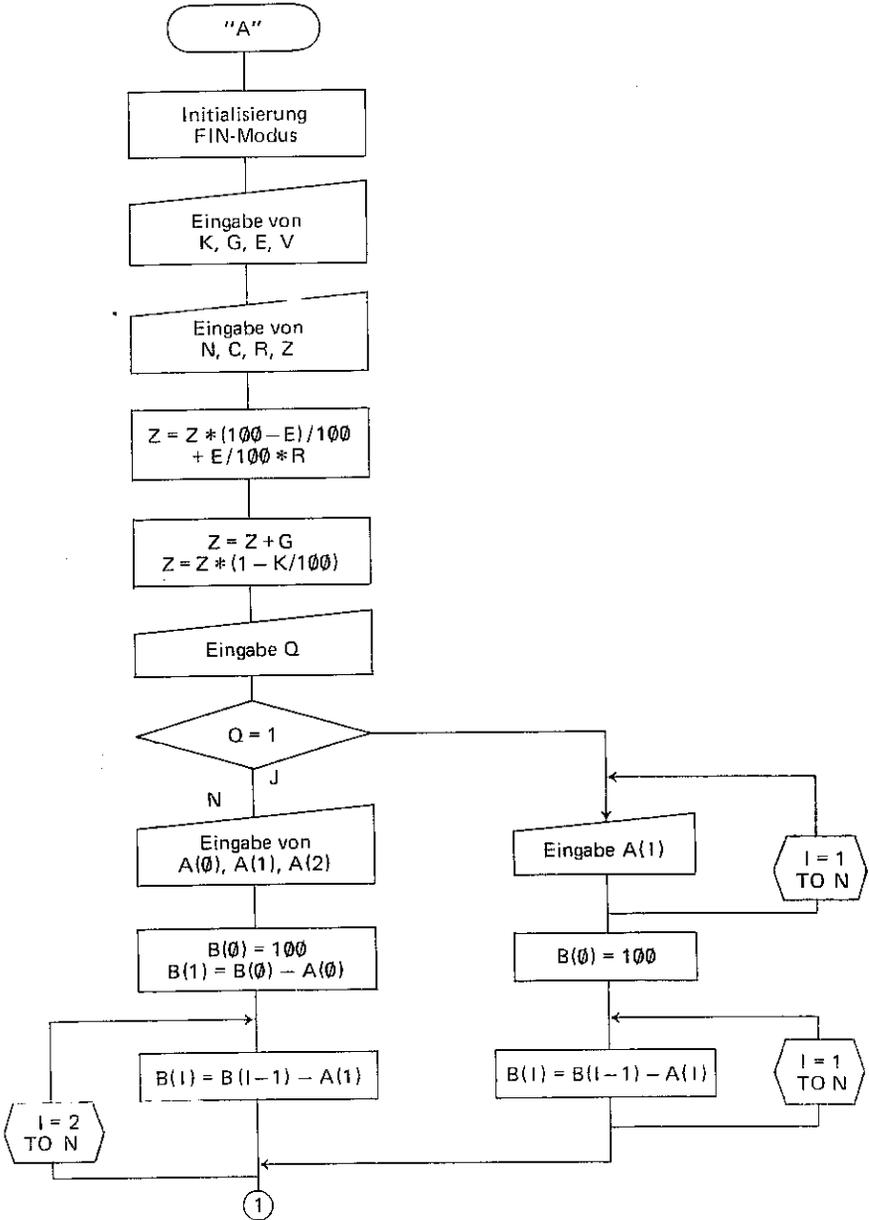
## TASTENBETÄTIGUNG

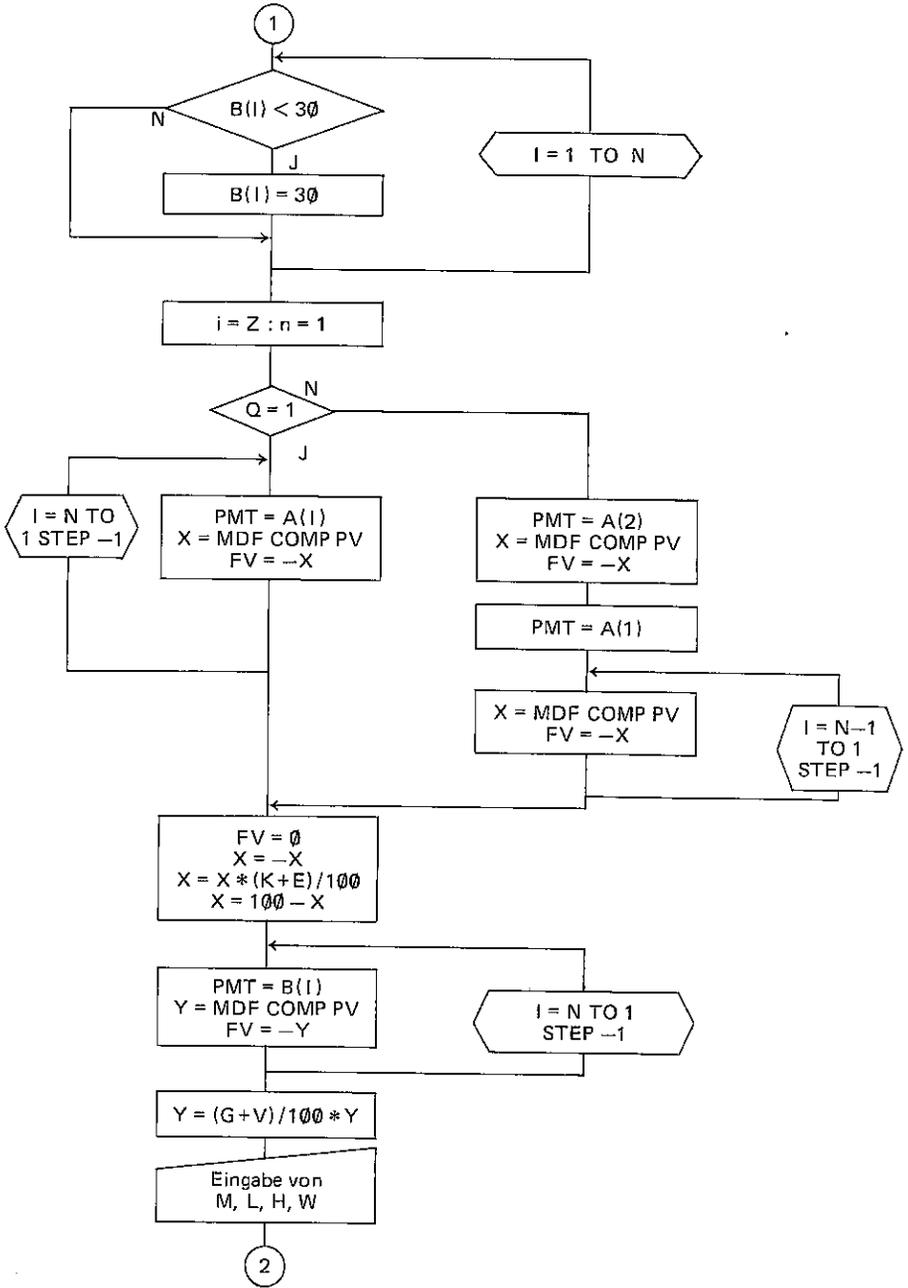
| Schritt-Nr. | Eingabe          | Anzeige                                                    | Bemerkungen                       |
|-------------|------------------|------------------------------------------------------------|-----------------------------------|
| 1           | <b>DEF</b> A     | STEUERSAETZE<br>KOERPERSCHAFT =                            |                                   |
| 2           | 56 <b>ENTER</b>  | GEW. KAP. =                                                |                                   |
| 3           | .6 <b>ENTER</b>  | GEW. ERTR. =                                               |                                   |
| 4           | 15 <b>ENTER</b>  | VERMOEG. =                                                 |                                   |
| 5           | .7 <b>ENTER</b>  | KAUF<br>LEB. DAUER =                                       |                                   |
| 6           | 5 <b>ENTER</b>   | EIG. KAP. =                                                |                                   |
| 7           | 20 <b>ENTER</b>  | RENDITE =                                                  |                                   |
| 8           | 10 <b>ENTER</b>  | ZINS =                                                     |                                   |
| 9           | 12 <b>ENTER</b>  | ABSCHREIBUNG<br>LINEAR = 0<br>NICHT LINEAR = 1<br>(0/1)? _ |                                   |
| 10          | 0 <b>ENTER</b>   | 1. ABSCHR. =                                               | Abschreibung im Kaufjahr eingeben |
| 11          | 20 <b>ENTER</b>  | ABSCHR. =                                                  | Weitere Abschreibungen eingeben   |
| 12          | 20 <b>ENTER</b>  | REST ABSCHR. =                                             | Evtl. Restabschreibung eingeben   |
| 13          | 0 <b>ENTER</b>   | LEASING<br>DAUER =                                         |                                   |
| 14          | 54 <b>ENTER</b>  | RATE =                                                     |                                   |
| 15          | 3.2 <b>ENTER</b> | HAEUFIGKEIT =                                              |                                   |
| 16          | 12 <b>ENTER</b>  | RESTWERT =                                                 |                                   |

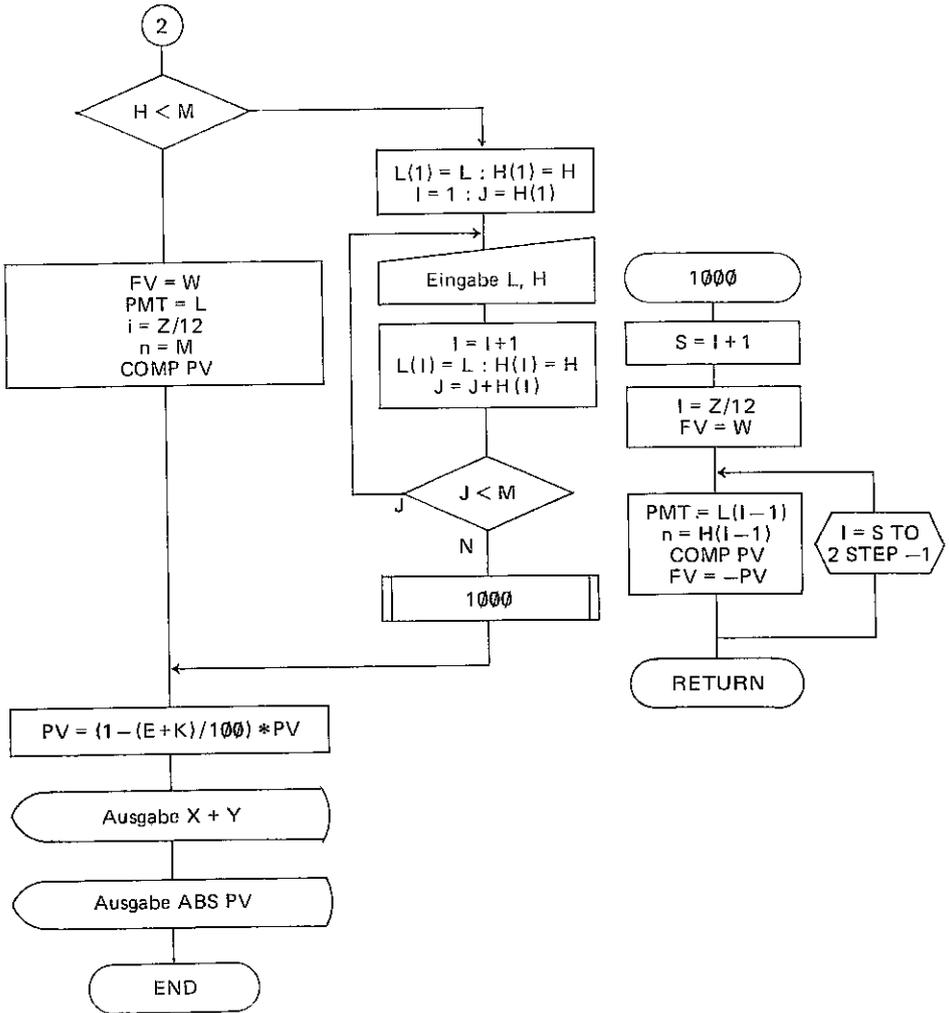
# TASTENBETÄTIGUNG

| Schritt-Nr. | Eingabe                                  | Anzeige          | Bemerkungen             |
|-------------|------------------------------------------|------------------|-------------------------|
| 17          | 10 <input type="button" value="ENTER"/>  | RATE =           |                         |
| 18          | 2.5 <input type="button" value="ENTER"/> | HAEUFIGKEIT =    |                         |
| 19          | 12 <input type="button" value="ENTER"/>  | RATE =           |                         |
| 20          | 2 <input type="button" value="ENTER"/>   | HAEUFIGKEIT =    |                         |
| 21          | 30 <input type="button" value="ENTER"/>  | KAUF<br>47.37    | Nettokosten bei Kauf    |
| 22          | <input type="button" value="ENTER"/>     | LEASING<br>35.67 | Nettokosten bei Leasing |
| 23          | <input type="button" value="ENTER"/>     | >                | Programmende            |

# FLUSSDIAGRAMM







# PROGRAMMLISTING

```

10: "A": CLEAR : WAIT 100
20: ERASE FIN
30: PRINT "STEUERSAETZE"
40: INPUT "KÖRPERSCHAFT
 =": K, "GEW. KAP. =": G, "
 GEW. ERTR. =": E, "VERMO
 EGEN. =": V
50: PRINT "KAUF"
60: INPUT "LEB. DAUER =": N
 , "EIG. KAP. =": C, "REND
 ITE =": R, "ZINS =": Z
70: Z = Z * (100 - E) / 100 + E / 10
 0 * R
80: Z = Z + G: Z = Z * (1 - K / 100)
90: PRINT "ABSCHREIBUNG"
 : PRINT "LINEAR = 0":
 PRINT "NICHT LINEAR =
 1"
100: INPUT "(0/1)?: " I
110: IF I = 0 THEN GOTO 200
120: DIM A(N), B(N)
130: FOR I = 1 TO N: PRINT "
 A(; I;) = "
140: INPUT A(I): NEXT I
150: B(0) = 100
160: FOR I = 1 TO N
170: B(I) = B(I - 1) - A(I)
180: NEXT I
190: GOTO 300
200: DIM A(2), B(N)
210: INPUT "1. ABSCHR. =": A
 (0), "ABSCHR. =": A(1),
 "REST ABSCHR. =": A(2)
220: B(0) = 100: B(1) = B(0) - A
 (0)
230: FOR I = 2 TO N
240: B(I) = B(I - 1) - A(I)
250: NEXT I
300: FOR I = 1 TO N
310: IF B(I) < 30 THEN LET
 B(I) = 30
320: NEXT I
330: I = Z: n = 1
340: IF I = 1 THEN GOTO 400
350: PMT = A(2): X = MDF COMP
 PV : FV = -X
360: PMT = A(1)
370: FOR I = N - 1 TO 1 STEP
 -1
380: X = MDF COMP PV : FV = -
 X
390: NEXT I: GOTO 500
400: FOR I = N TO 1 STEP -1
410: PMT = A(I): X = MDF COMP
 PV : FV = -X
420: NEXT I
500: X = -X: X = X * (K + E) / 100: X
 = 100 - X: FV = 0
510: FOR I = N TO 1 STEP -1
520: PMT = B(I): Y = MDF COMP
 PV : FV = -Y

```

# SPICHERINHALT

|      |                                 |
|------|---------------------------------|
| A(N) | (Abschreibung)                  |
| B(N) | (Restbuchwert)                  |
| C    | (Eigenkapitalanteil)            |
| D    |                                 |
| E    | (Gewerbeertragssteuersatz)      |
| F    |                                 |
| G    | (Gewerkekapitalsteuersatz)      |
| H(M) | (Anzahl gleicher Leasingraten)  |
| I    |                                 |
| J    | $\Sigma H(I)$                   |
| K    | (Körperschaftsteuersatz)        |
| L(M) | (Leasingraten)                  |
| M    | (Leasingdauer)                  |
| N    | (Nutzungsdauer)                 |
| O    |                                 |
| P    |                                 |
| Q    | "Markierung"                    |
| R    | (Eigenkapitalrendite)           |
| S    |                                 |
| T    |                                 |
| U    |                                 |
| V    | (Vermögenssteuersatz)           |
| W    | (Restwert)                      |
| X    | (Nettobarwert der Abschreibung) |
| Y    | (Kapitalsteuern)                |
| Z    | (Zins)                          |

```

530: NEXT I
540: Y = (G + V) / 100 * Y
600: PRINT "LEASING"
700: INPUT "DAUER =": N, "RA
 TE =": L, "HAEFUGIGKEIT =
 ": H, "RESTWERT =": W
710: IF H < M THEN 800
720: FV = W: PMT = L: I = Z / 12
 : n = M
730: COMP PV : GOTO 900
800: DIM L(N), H(N)
810: L(I) = L: H(I) = H: I = I + 1: J =
 H(I)
820: INPUT "RATE =": L, "HAE
 UFIGKEIT =": H
830: I = I + 1: L(I) = L: H(I) = H:
 J = J + H(I)
840: IF J < M THEN 820
850: GOSUB 1000
900: PV = (1 - (E + K) / 100) *
 PV
910: PRINT "KAUF"
920: WAIT : USING "###.##"
930: PRINT X + Y
940: WAIT 100: PRINT "LEAS
 ING"
950: WAIT : PRINT ABS PV
960: END
1000: S = I + 1: FV = W: I = Z / 1
 2
1010: FOR I = S TO 2 STEP
 -1
1020: PMT = L(I): n = H(I)
 -1: COMP PV : FV = -
 PV
1030: NEXT I
1040: RETURN

```

## ÜBERBLICK

In vielen Bereichen (z.B. Immobilienmarkt, Rentenrechnung etc.) ist der Barwert einer Kapitalanlage und ihre Rendite von Interesse. Dieses Programm berechnet entweder den Barwert einer Kapitalanlage bei vorgegebener Rendite oder umgekehrt.

## INHALT (Formel)

|           |     |                                             |
|-----------|-----|---------------------------------------------|
| Bezeichne | PV  | den Barwert                                 |
|           | FV  | den zukünftigen Wert (nach n Jahren)        |
|           | PMT | den jährlich zu erwartenden Gewinn/Annuität |
|           | i   | die jährliche Rendite                       |

Die Berechnung von PV bzw. i erfolgt gemäß der Formel:

$$FV = PV (1+i)^n - PMT \frac{(1+i)^n - 1}{i}$$

## ANLEITUNG ZUM BETRIEB

1. Das Programm wird mit **DEF** A gestartet.
2. Eingabe, ob Barwert oder Rendite berechnet werden soll.
3. Eingabe von FV, PMT, i, n bzw. FV, PMT, PV, n
4. Ausgabe von PV bzw. i

## BEISPIEL

Eine Immobiliengesellschaft möchte Wohneinheiten erwerben, die jährlich einen Gewinn von DM 125.000,- abwerfen. Die Gesellschaft möchte einen Wiederverkaufswert von DM 5.400.000,- nach dieser Zeit.

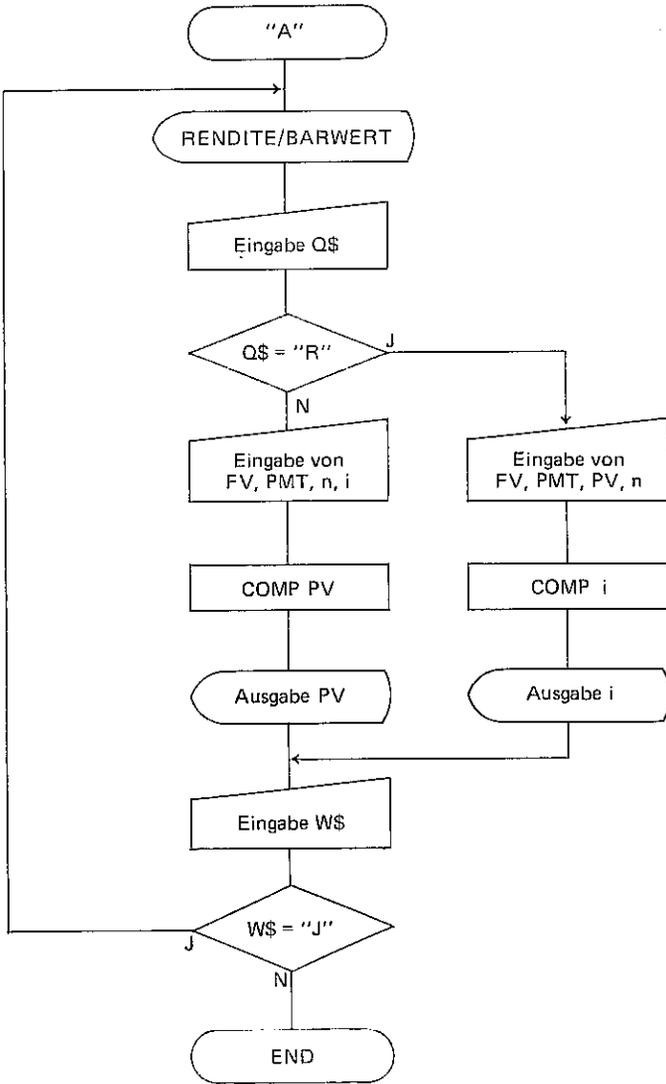
- (1) Es soll der maximale Betrag (Barwert) bestimmt werden, den die Gesellschaft für die Wohnungen zahlen kann, wenn eine jährliche Rendite von 12% erzielt werden soll.
- (2) Wie hoch ist die jährliche Rendite, wenn die Gesellschaft für die Wohnungen DM 4.000.000,- zahlt?

**Bemerkung:** Beachten Sie, daß der Barwert PV von der Gesellschaft zu zahlen ist, also negativ eingegeben werden muß.

# TASTENBETÄTIGUNG

| Schritt-Nr. | Eingabe               | Anzeige                     | Bemerkungen                                             |
|-------------|-----------------------|-----------------------------|---------------------------------------------------------|
| 1           | <b>DEF</b> A          | BARWERT/RENDITE<br>B/R?     | Eingeben, ob Barwert oder Rendite berechnet werden soll |
| 2           | B <b>ENTER</b>        | FV =                        | Zukünftigen Wert eingeben                               |
| 3           | 5400000 <b>ENTER</b>  | PMT =                       | Jährlichen Gewinn eingeben                              |
| 4           | 125000 <b>ENTER</b>   | n =                         | Anzahl der Jahre eingeben                               |
| 5           | 5 <b>ENTER</b>        | i =                         | Rendite eingeben                                        |
| 6           | 12 <b>ENTER</b>       | BARWERT:<br>PV = 3514702.05 |                                                         |
| 7           | <b>ENTER</b>          | WEITER (J/N)?               | J eingeben, wenn weitere Berechnungen erwünscht sind    |
| 8           | J <b>ENTER</b>        | BARWERT/RENDITE<br>B/R?     |                                                         |
| 9           | R <b>ENTER</b>        | FV =                        |                                                         |
| 10          | 5400000 <b>ENTER</b>  | PMT =                       |                                                         |
| 11          | 125000 <b>ENTER</b>   | PV =                        |                                                         |
| 12          | -4000000 <b>ENTER</b> | n =                         |                                                         |
| 13          | 5 <b>ENTER</b>        | RENDITE:<br>i = 8.97        |                                                         |
| 14          | <b>ENTER</b>          | WEITER (J/N)?               |                                                         |
| 15          | N <b>ENTER</b>        | >                           | Programmende                                            |

# FLUSSDIAGRAMM



## PROGRAMMLISTING

```

10:"A":CLEAR :WAIT 100
20:PRINT "BARWERT/RENDI
TE"
30:INPUT "B/R?";Q$
40:IF LEFT$(Q$,1)="R"
GOTO 200
100:INPUT " FV =" ; FV , "
PMT =" ; PMT , " n =" ;
n , " i =" ; i
110:COMP PV
120:PRINT "BARWERT:" :
USING "#####.##"
130:WAIT :PRINT " PV =" ;
PV :USING
140:GOTO 300
200:INPUT " FV =" ; FV , "
PMT =" ; PMT , " PV ="
; PV , " n =" ; n
210:COMP i
220:PRINT "RENDITE:" :
USING "###.##"
230:WAIT :PRINT " i =" ;
i :USING
300:INPUT "WEITER (J/N)?
";W$:WAIT 100
310:IF LEFT$(W$,1)="J"
GOTO 20
320:END

```

## SPEICHERINHALT

|     |   |
|-----|---|
| A   |   |
| B   |   |
| C   |   |
| D   |   |
| E   |   |
| F   |   |
| G   |   |
| H   |   |
| I   |   |
| J   |   |
| K   |   |
| L   |   |
| M   |   |
| N   |   |
| O   |   |
| P   |   |
| Q\$ | ✓ |
| R   |   |
| S   |   |
| T   |   |
| U   |   |
| V   |   |
| W\$ | ✓ |
| X   |   |
| Y   |   |
| Z   |   |

## ÜBERBLICK

Der Nominalzins ist derjenige Zinssatz, zu dem eine Schuld bei jährlicher Verzinsung verzinst wird. Erfolgt die Verzinsung aber tatsächlich unterjährig (z. B. monatlich, wie es bei Dispositionskrediten üblich ist), so treten ab der zweiten Abrechnungsperiode nicht nur Zinsen auf die eigentliche (Rest-)Schuld, sondern auch Zinseszinsen auf. Der tatsächlich zu zahlende Jahreszins, der Effektivzins, berücksichtigt diese Tatsache.

## INHALT (Formel)

APR = Nominalzins

EFF = Effektivzins

n = Zahl der Abrechnungsperioden pro Jahr

$$\text{EFF} = \left(1 + \frac{\text{APR}}{n}\right)^n - 1$$

## ANLEITUNG ZUM BETRIEB

1. Das Programm wird mit **DEF** A gestartet.
2. Eingabe, ob APR oder EFF berechnet werden soll.
3. Eingabe von N, EFF bzw. APR.
4. Ausgabe von APR bzw. EFF.

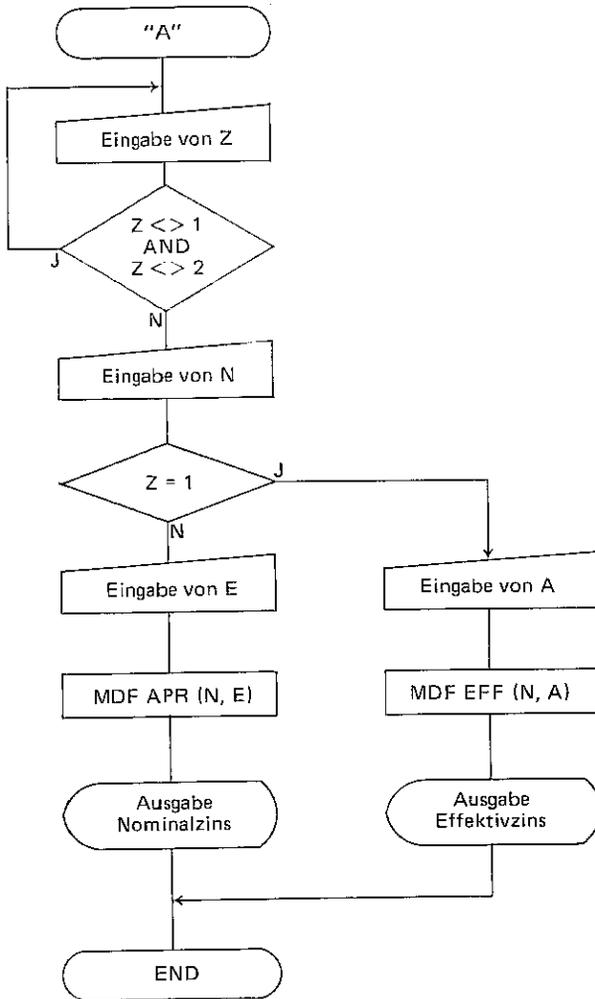
## BEISPIEL

Sie benötigen für die Anschaffung eines Neuwagens ein Darlehen. Ihre Bank bietet Ihnen die gewünschte Summe zu einem effektiven Jahreszins von 11,3% an. Ein wohlhabender Freund bietet Ihnen an, Ihnen den Betrag zu 10% Jahreszinsen bei vierteljährlicher Abrechnung zu leihen. Welches Angebot ist für Sie günstiger?

# TASTENBETÄTIGUNG

| Schritt-Nr. | Eingabe         | Anzeige                 | Bemerkungen                                        |
|-------------|-----------------|-------------------------|----------------------------------------------------|
| 1           | <b>DEF</b> A    | EFF = 1<br>APR = 2<br>? | Soll EFF oder APR berechnet werden?                |
| 2           | 1 <b>ENTER</b>  | n =                     | Anzahl der jährlichen Abrechnungsperioden eingeben |
| 3           | 4 <b>ENTER</b>  | APR =                   | APR eingeben                                       |
| 4           | 10 <b>ENTER</b> | EFF = 10.38             |                                                    |
| 5           | <b>ENTER</b>    | >                       | Programmende                                       |

# FLUSSDIAGRAMM



## PROGRAMMLISTING

```

10:"A":CLEAR :USING "##
 #.##"
20:WAIT 100:PRINT "EFF
 =1"
30:PRINT "APR =2":WAIT
40:INPUT Z
50:IF Z<>1 AND Z<>2
 GOTO 20
60:INPUT " n =" ;N
70:IF Z=1 GOTO 100
80:INPUT "EFF =" ;E
90:PRINT "APR =" ;MDF
 APR (N,E):END
100:INPUT "APR =" ;A
110:PRINT "EFF =" ;MDF
 EFF (N,A)
120:END

```

## SPEICHERINHALT

|   |     |
|---|-----|
| A | APR |
| B |     |
| C |     |
| D |     |
| E | EFF |
| F |     |
| G |     |
| H |     |
| I |     |
| J |     |
| K |     |
| L |     |
| M |     |
| N | n   |
| O |     |
| P |     |
| Q |     |
| R |     |
| S |     |
| T |     |
| U |     |
| V |     |
| W |     |
| X |     |
| Y |     |
| Z | √   |

## ÜBERBLICK

Meist werden bei Darlehen fest Annuitäten vereinbart. Dieses Programm erlaubt für jede Rückzahlungsrate den auf Tilgung und Zinsen entfallenden Anteil zu ermitteln.

## INHALT

Tilgung, Zinsen und Restschuld für die  $j$ -te Zahlungsperiode ergeben sich aus:

$$\begin{aligned}PV_j &= PV_{j-1} + PRN_j && \text{(Restschuld)} \\INT_j &= -PV_{j-1} * i && \text{Zinsanteil der } j\text{-ten Annuität} \\PRN_j &= PMT - INT_j && \text{j-te Tilgungsrate (für } j \geq 1\text{)}\end{aligned}$$

mit  $PV_0 = PV =$  Darlehnsbetrag bei nachschüssiger  
und  $INT_1 = \emptyset$  bei vorschüssiger Zahlungsweise.

Beachten Sie, daß Zahlungen (PMT,  $PRN_j$ ) negative Zahlen sind.

## ANLEITUNG ZUM BETRIEB

1. Das Programm wird mit **DEF** A gestartet.
2. Eingabe von PV, i, n, J
3. Eingabe, ob vor- oder nachschüssige Zahlung
4. Ausgabe von  $PRN_j$ ,  $INT_j$ ,  $PV_j$

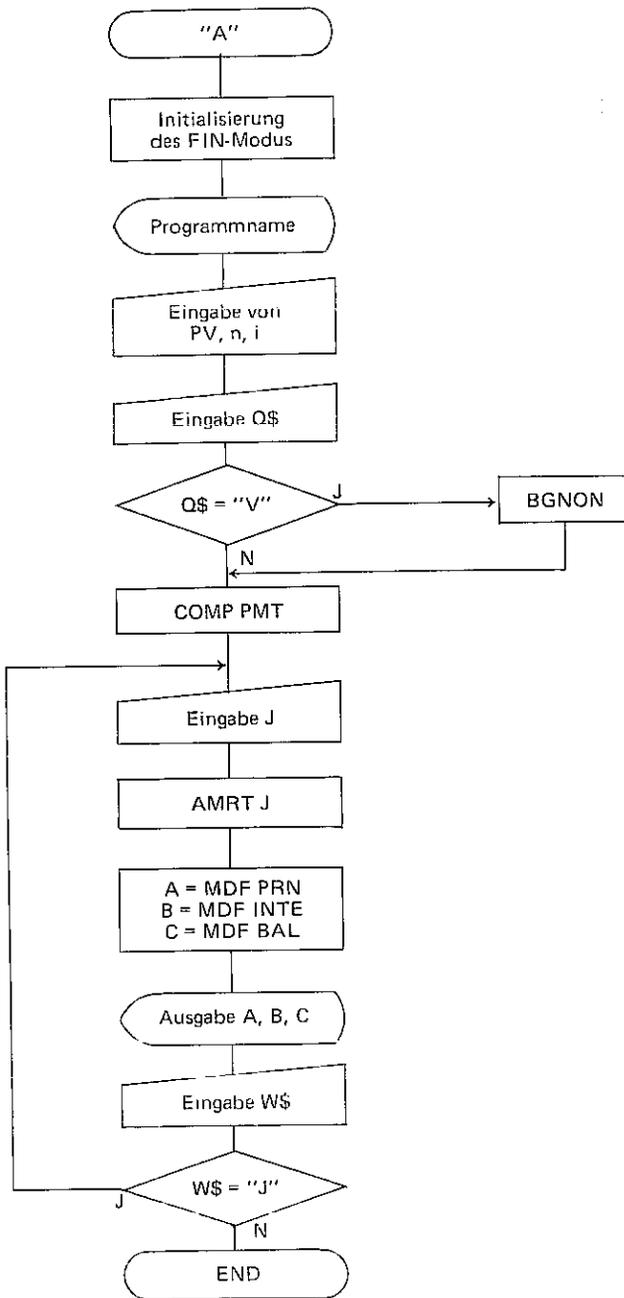
## BEISPIEL

Zur Finanzierung eines Anbaues können Sie eine Hypothek über DM 30.000,- bei einer Laufzeit von 20 Jahren zu 13 3/4% Jahreszins erhalten. Es ist bei vorschüssiger monatlicher Zahlungsweise (Rückzahlungsrate DM 363,46) der jeweilige Tilgungs- und Zinsanteil der ersten drei Raten sowie die jeweils verbleibende Restschuld zu bestimmen.

# TASTENBETÄTIGUNG

| Schritt-Nr. | Eingabe                                     | Anzeige               | Bemerkungen                                         |
|-------------|---------------------------------------------|-----------------------|-----------------------------------------------------|
| 1           | <b>DEF</b> A                                | TILGUNGSPLAN<br>PV =  | Darlehensbetrag eingeben                            |
| 2           | 30000 <b>ENTER</b>                          | i =                   | Jahreszinssatz eingeben                             |
| 3           | 13.75 <b>SHIFT</b> <b>I</b><br><b>ENTER</b> | n =                   | Laufzeit eingeben                                   |
| 4           | 20 <b>SHIFT</b> <b>n</b><br><b>ENTER</b>    | VOR/NACH (V/N)?       | Bei vorschüssiger Zahlungsweise V eingeben, sonst N |
| 5           | V <b>ENTER</b>                              | # RATE =              | Nr. der Rate eingeben                               |
| 6           | 1 <b>ENTER</b>                              | -363.46               | PRN <sub>1</sub>                                    |
| 7           | <b>ENTER</b>                                | 0.00                  | INT <sub>1</sub>                                    |
| 8           | <b>ENTER</b>                                | 29636.54              | PV <sub>1</sub>                                     |
| 9           | <b>ENTER</b>                                | WEITERE RATEN?<br>J/N |                                                     |
| 10          | J <b>ENTER</b>                              | # RATE =              |                                                     |
| 11          | 2 <b>ENTER</b>                              | -23.87                | PRN <sub>2</sub>                                    |
| 12          | <b>ENTER</b>                                | -339.59               | INT <sub>2</sub>                                    |
| 13          | <b>ENTER</b>                                | 29612.67              | PV <sub>2</sub>                                     |
| 14          | <b>ENTER</b>                                | WEITERE RATEN?<br>J/N |                                                     |
| 15          | J <b>ENTER</b>                              | # RATE =              |                                                     |
| 16          | 3 <b>ENTER</b>                              | -24.15                | PRN <sub>3</sub>                                    |
| 17          | <b>ENTER</b>                                | -339.31               | INT <sub>3</sub>                                    |
| 18          | <b>ENTER</b>                                | 29588.53              | PV <sub>3</sub>                                     |
| 19          | <b>ENTER</b>                                | WEITERE RATEN?<br>J/N |                                                     |
| 20          | N <b>ENTER</b>                              | >                     | Programmende                                        |

# FLUSSDIAGRAMM



# PROGRAMMLISTING

```

10:"A":CLEAR :WAIT 100
20:ERASE FIN
30:PRINT "TILGUNGSPLAN"
40:INPUT " PV ="; PV ,"
 i ="; i ," n ="; n
50:INPUT "VOR/NACH(V/N)
 ?";Q$
60:IF LEFT$(Q$,1)="V"
 THEN BGNON
70:COMP PMT
80:INPUT "#RATE=";J
90:AMRT J:A=MDF PRN :B=
 MDF INTE :C=MDF BAL
100:WAIT :USING "#####.
 ##":PRINT A:PRINT B:
 PRINT C
110:WAIT 100:PRINT "WEIT
 ERE RATEN?"
120:INPUT "J/N";W$
130:IF LEFT$(W$,1)="J"
 GOTO 80
140:END

```

# SPEICHERINHALT

|     |                  |
|-----|------------------|
| A   | PRN <sub>j</sub> |
| B   | INT <sub>j</sub> |
| C   | PV <sub>j</sub>  |
| D   |                  |
| E   |                  |
| F   |                  |
| G   |                  |
| H   |                  |
| I   |                  |
| J   | j (# Rate)       |
| K   |                  |
| L   |                  |
| M   |                  |
| N   |                  |
| O   |                  |
| P   |                  |
| Q\$ | ✓                |
| R   |                  |
| S   |                  |
| T   |                  |
| U   |                  |
| V   |                  |
| W\$ | ✓                |
| X   |                  |
| Y   |                  |
| Z   |                  |

## ÜBERBLICK

Für den Darlehensnehmer stellen sich die Fragen,

- wieviele monatliche Rückzahlungsraten er bei vorgegebenem Nominalzins zu leisten hat, wenn er monatlich einen bestimmten Betrag (Annuität) zahlt (zahlen kann);
- wie hoch die monatliche Annuität bei vorgegebenem Nominalzins und vorgegebener Laufzeit ist;
- wie hoch seine Gesamtzahlungen für Darlehen einschließlich Zinsen sind.

Dieses Programm gibt wahlweise eine Antwort auf die Fragen a) und c) oder b) und c).

## INHALT (Formel)

|              |     |                                    |
|--------------|-----|------------------------------------|
| Bezeichnung: | i   | der (monatliche) Nominalzinssatz   |
|              | PMT | die monatliche Annuität            |
|              | PV  | der Darlehensbetrag                |
|              | n   | die Anzahl der zu leistenden Raten |

So gilt:

$$PV = PMT \frac{1 - (1 + i)^{-n}}{i}$$

Die Gesamtschuld ist die Summe der Annuitäten.

## ANLEITUNG ZUM BETRIEB

- Das Programm wird mit **R** **U** **N** **ENTER** gestartet.
- Eingabe, ob PMT oder n berechnet werden soll.
- Eingabe von (1) PV  
PMT  
i (Jahres-) Nominalzins  
oder (2) PV  
n  
i

4. Ausgabe von (1) n

L

Höhe der letzten Rate

S

Summe aller Zahlungen

(2) PMT

S

**Bemerkung:** Zu leistende Zahlungen (PMT, L, S) werden als negative Werte eingegeben bzw. ausgegeben.

## BEISPIEL

Zur Finanzierung eines Anbaus bietet Ihnen ein Freund ein Darlehen in Höhe von DM 30.000, – zu 10,5% Nominalzins an.

- (1) Wenn Sie an jedem Monatsende DM 300, – zurückzahlen und die Verzinsung monatlich erfolgt, wie lange wird die Rückzahlung dauern?
- (2) Wenn sich die Rückzahlung über 20 Jahre (240 Monate) erstrecken soll, wie hoch ist dann die monatlich zu leistende Rate?

## TASTENBETÄTIGUNG

(1)

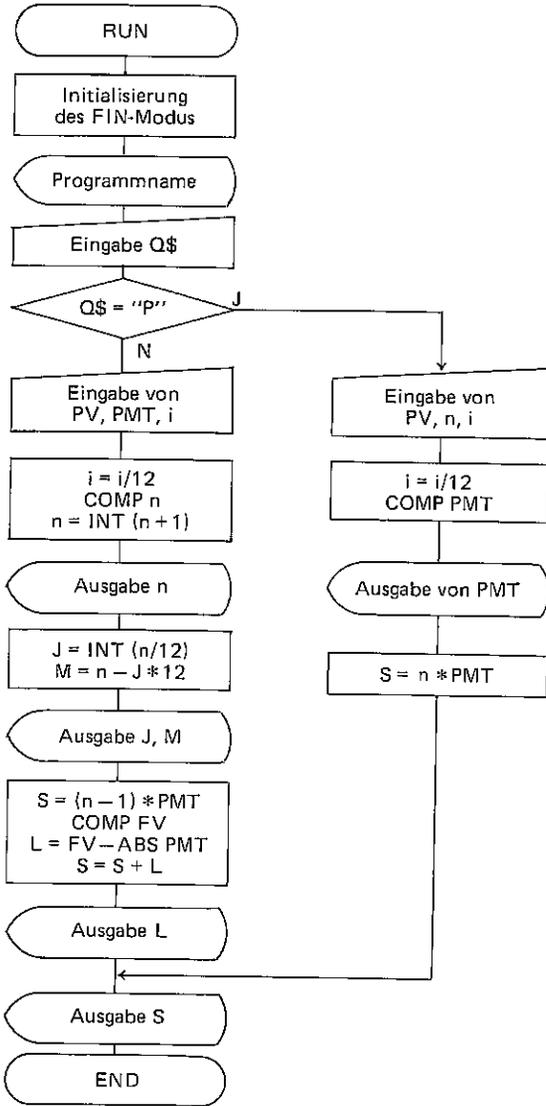
| Schritt-Nr. | Eingabe                                    | Anzeige                   | Bemerkungen                                                  |
|-------------|--------------------------------------------|---------------------------|--------------------------------------------------------------|
| 1           | <b>R</b> <b>U</b> <b>N</b><br><b>ENTER</b> | DARLEHEN<br>PMT/n?<br>P/N | P eingeben, wenn die Annuität berechnet werden soll, sonst N |
| 2           | N <b>ENTER</b>                             | PMT =                     | PMT eingeben                                                 |
| 3           | -300 <b>ENTER</b>                          | PV =                      | Darlehensbetrag eingeben                                     |
| 4           | 30000 <b>ENTER</b>                         | i =                       | Nominalzins eingeben                                         |
| 5           | 10.5 <b>ENTER</b>                          | n = 239.                  | Es sind 239 Raten zu zahlen                                  |
| 6           | <b>ENTER</b>                               | 19.J.; 11.MO.             | 239 Monat sind 19 Jahre und 11 Monate                        |
| 7           | <b>ENTER</b>                               | LETZTE RATE =<br>-206.88  | Die letzte Rate beträgt DM 206,88                            |
| 8           | <b>ENTER</b>                               | SUMME =<br>-71606.88      | Die Summe Ihrer Zahlungen ist DM 71606,88                    |
| 9           | <b>ENTER</b>                               | >                         | Programmende                                                 |

# TASTENBETÄTIGUNG

(2)

| Schritt-Nr. | Eingabe                                                                                                                              | Anzeige                   | Bemerkungen                                                  |
|-------------|--------------------------------------------------------------------------------------------------------------------------------------|---------------------------|--------------------------------------------------------------|
| 1           | <input type="text" value="R"/> <input type="text" value="U"/> <input type="text" value="N"/><br><input type="button" value="ENTER"/> | DARLEHEN<br>PMT/n?<br>P/N | P eingeben, wenn die Annuität berechnet werden soll, sonst N |
| 2           | P <input type="button" value="ENTER"/>                                                                                               | PV =                      |                                                              |
| 3           | 30000 <input type="button" value="ENTER"/>                                                                                           | n =                       | Anzahl der Raten eingeben                                    |
| 4           | 240 <input type="button" value="ENTER"/>                                                                                             | i =                       |                                                              |
| 5           | 10.5 <input type="button" value="ENTER"/>                                                                                            | PMT =<br>-299.52          | Die monatliche Annuität beträgt DM 299,52                    |
| 6           | <input type="button" value="ENTER"/>                                                                                                 | SUMME =<br>-71.884.80     |                                                              |
| 7           | <input type="button" value="ENTER"/>                                                                                                 | >                         | Programmende                                                 |

# FLUSSDIAGRAMM



PROGRAMMLISTING

SPEICHERINHALT

```

10:ERASE FIN :WAIT 100
20:PRINT "DARLEHEN"
30:PRINT " PMT / n ?"
40:INPUT "P/N?:Q$
50:IF LEFT$(Q$,1)="P"
 GOTO 200
60:INPUT " PMT ="; PMT
 ;" PV ="; PV ;" i ="
 ; i
70:i = i /12:COMP n :n
 = INT (n +1)
80:WAIT :PRINT " n =";
 n
100:J= INT (n /12):M= n
 -J*12
110:PRINT J;"J. " ;M;"MO.
 "
120:S=(n -1)* PMT
130:GOTO 500
200:INPUT " PV ="; PV ;"
 n ="; n ;" i =" ; i
210:i = i /12:COMP PMT :
 PMT = INT (PMT *100
)/100:S= n * PMT
220:WAIT :PRINT " PMT ="
 ; PMT
230:GOTO 540
500:COMP FV :L= FV -ABS
 PMT :S=S+L
510:WAIT 100:PRINT "LETZ
 TE RATE="
520:WAIT :USING "#####.
 ##"
530:PRINT L
540:WAIT 100:PRINT "SUMM
 E="
550:USING "#####.##"
560:WAIT :PRINT S

```

|     |                                     |
|-----|-------------------------------------|
| A   |                                     |
| B   |                                     |
| C   |                                     |
| D   |                                     |
| E   |                                     |
| F   |                                     |
| G   |                                     |
| H   |                                     |
| I   |                                     |
| J   | (Anzahl der Ruckzahlungs-<br>jahre) |
| K   |                                     |
| L   | (SchluBzahlung)                     |
| M   | $n - 12 * J$                        |
| N   |                                     |
| O   |                                     |
| P   |                                     |
| Q\$ | ✓                                   |
| R   |                                     |
| S   | $(n - 1) PMT + L$                   |
| T   |                                     |
| U   |                                     |
| V   |                                     |
| W   |                                     |
| X   |                                     |
| Y   |                                     |
| Z   |                                     |

## ÜBERBLICK

Es ist üblich, die Abschreibung auf ein Kalenderjahr zu beziehen. Dabei werden Wirtschaftsgüter, die im ersten Halbjahr angeschafft wurden, im ersten Jahr mit dem vollen Jahresabschreibungsbetrag abgeschrieben und Wirtschaftsgüter, die im zweiten Halbjahr angeschafft wurden, dürfen im ersten Jahr mit dem halben Abschreibungsbetrag abgeschrieben werden (vgl. Einkommenssteuerrichtlinien).

Es sind drei Methoden der Abschreibung üblich: Lineare Abschreibung, degressive Abschreibung und digitale Abschreibung.

Das folgende Programm erlaubt unter Berücksichtigung der o. a. Steuerrichtlinien für jedes Abschreibungsjahr die Abschreibung, den Restbuchwert sowie den noch abschreibbaren Restbetrag für die drei genannten Methoden zu bestimmen.

## INHALT (Formeln)

Bezeichnung: C Anschaffungswert  
 S Restwert (Schrottwert)  
 n Abschreibungsdauer (Lebensdauer)

Dann gilt bei linearer Abschreibung:

$$A = A_j = \frac{C - S}{n} \quad \text{für jede Abschreibungsperiode } (j = \{1, 2, \dots, n\})$$

und

$$A_1 = 0,5 A \quad \text{für die erste Abschreibungsperiode bei Anschaffung im 2. Halbjahr}$$

Der Restbuchwert ist  $R_j = R_{j-1} - A_j$  mit  $R_0 = C$ .

Der abschreibbare Restbetrag ist  $C_j = C_{j-1} - A_j$  mit  $C_0 = C - S$ .

Bei degressiver Abschreibung gilt bei Degressionsfaktor  $f$  (in %):

$$A_j = C_{j-1} \frac{f}{100n} \quad j = \{1, 2, \dots, n\}$$

mit

$$A_1 = C \frac{f}{100n} \cdot 0,5 \quad \text{für die halbe erste Abschreibung}$$

Bei digitaler Abschreibung gilt mit

$$D_k = \frac{(W+1)(W+2F)}{2} \quad \text{für} \quad \begin{array}{l} W = \text{INT}(K) \\ F = K - W \end{array}$$

$$A_j = \frac{(n-j+1)}{D_n} (C-S)$$

und bei halber erster Abschreibung:

$$A_1 = \frac{n}{2D_n} (C-S)$$

$$A_j = \frac{n-0,5-j+2}{D_{n-0,5}} (C-A_1-S) \quad (j \neq 1)$$

## ANLEITUNG ZUM BETRIEB

1. Das Programm mit **DEF** A starten.

2. Eingabe von C, S, N

- Y (Anzahl der Restmonate im Anschaffungsjahr)
- Z (= 1 lineare, =2 degressive, =3 digitale Abschreibung)
- J Abschreibungsperiode
- F (bei degressiver Abschreibung)

3. Ausgabe von  $A_j$ ,  $R_j$ ,  $C_j$

## BEISPIEL

Ein älteres Gebäude wird im Dezember für einen Betrag von DM 125.000,- erstanden. Die verbleibende Nutzungsdauer des Gebäudes wird auf 25 Jahre geschätzt und es wird kein Restwert angesetzt.

Es soll der Abschreibungsbetrag für die ersten drei Jahre, sowie der jeweils verbleibende Restbuchwert bestimmt werden. Dabei sollen alle drei Abschreibungsmethoden verglichen werden, wobei ein Dregressionsmultiplikator von 250% angesetzt wird.

# TASTENBETÄTIGUNG

| Schritt-Nr. | Eingabe             | Anzeige                                         | Bemerkungen                                                        |
|-------------|---------------------|-------------------------------------------------|--------------------------------------------------------------------|
| 1           | <b>DEF</b> A        | NEU =                                           | Neuwert eingeben                                                   |
| 2           | 125000 <b>ENTER</b> | REST =                                          | Restwert eingeben                                                  |
| 3           | 0 <b>ENTER</b>      | DAUER =                                         | Lebensdauer eingeben                                               |
| 4           | 25 <b>ENTER</b>     | 1. JAHR =                                       | Anzahl der Restmonate im 1. Jahr eingeben                          |
| 5           | 1 <b>ENTER</b>      | LINEAR = 1<br>DEGRESSIV = 2<br>DIGITAL = 3<br>? | Eingabe, nach welcher Methode abgeschrieben werden soll            |
| 6           | 1 <b>ENTER</b>      | PERIODE =                                       | Abschreibungsperiode (Jahr) eingeben                               |
| 7           | 1 <b>ENTER</b>      | ABSCH = 2500.00                                 | Abschreibung im 1. Jahr (linear)                                   |
| 8           | <b>ENTER</b>        | BUCH = 122500.00                                | Restbuchwert nach dem 1. Jahr                                      |
| 9           | <b>ENTER</b>        | ABSCHREIBBARER REST = 122500.00                 | Verbleibender abschreibbarer Betrag                                |
| 10          | <b>ENTER</b>        | NEUE RECHN (J/N)                                | "J" eingeben, falls weitere Abschreibungen berechnet werden sollen |
| 11          | J <b>ENTER</b>      | LINEAR = 1<br>DEGRESSIV = 2<br>DIGITAL = 3<br>? |                                                                    |
| 12          | 1 <b>ENTER</b>      | PERIODE =                                       |                                                                    |
| 13          | 2 <b>ENTER</b>      | ABSCH = 5000.00                                 | Abschreibung im 2. Jahr (linear)                                   |
| 14          | <b>ENTER</b>        | BUCH = 112500.00                                |                                                                    |
| 15          | <b>ENTER</b>        | ABSCHREIBBARER REST = 112500.00                 |                                                                    |
| 16          | <b>ENTER</b>        | NEUE RECHN (J/N)                                |                                                                    |
| 17          | J <b>ENTER</b>      | LINEAR = 1<br>DEGRESSIV = 2<br>DIGITAL = 3<br>? |                                                                    |
| 18          | 3 <b>ENTER</b>      | PERIODE =                                       |                                                                    |
| 19          | 1 <b>ENTER</b>      | ABSCH = 4807.69                                 | 1. Abschreibung digital                                            |

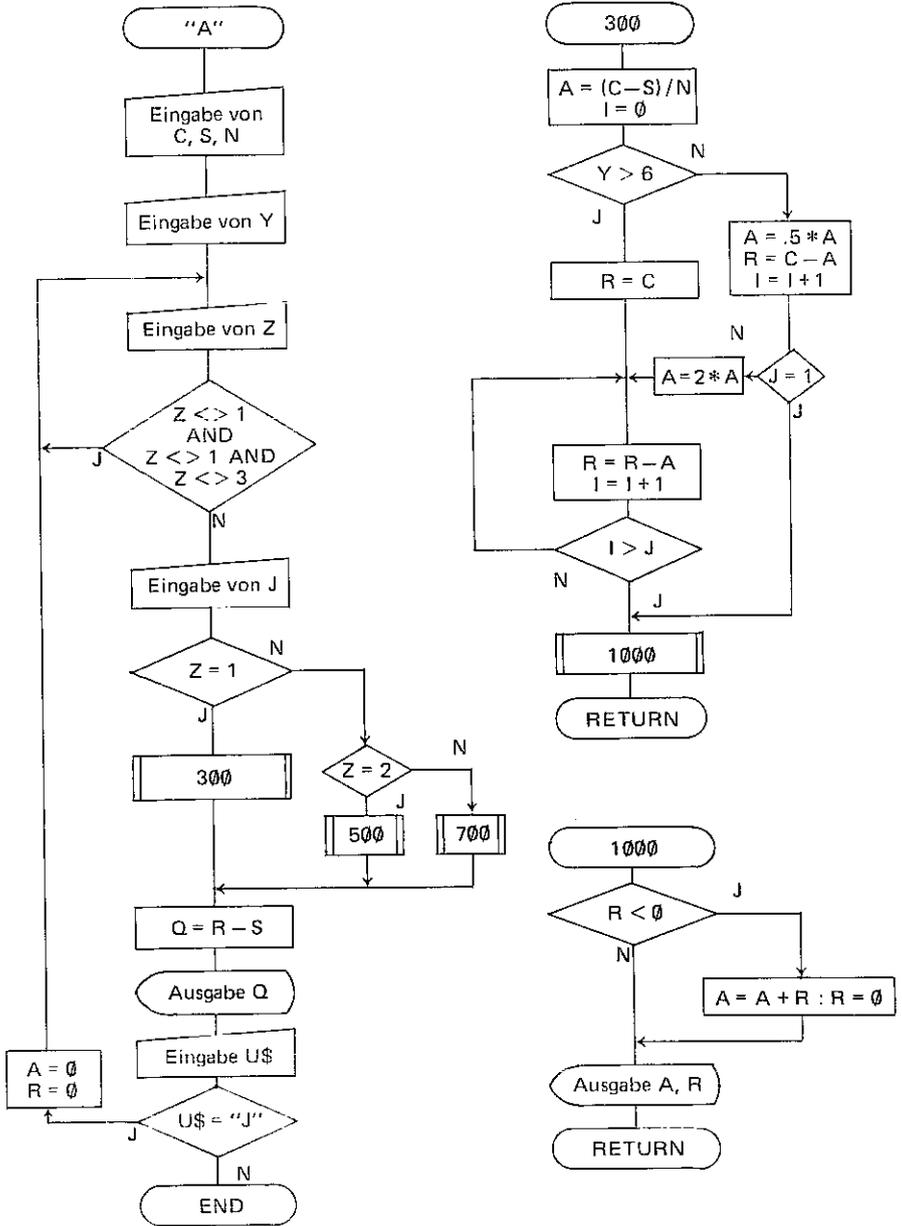
# TASTENBETÄTIGUNG

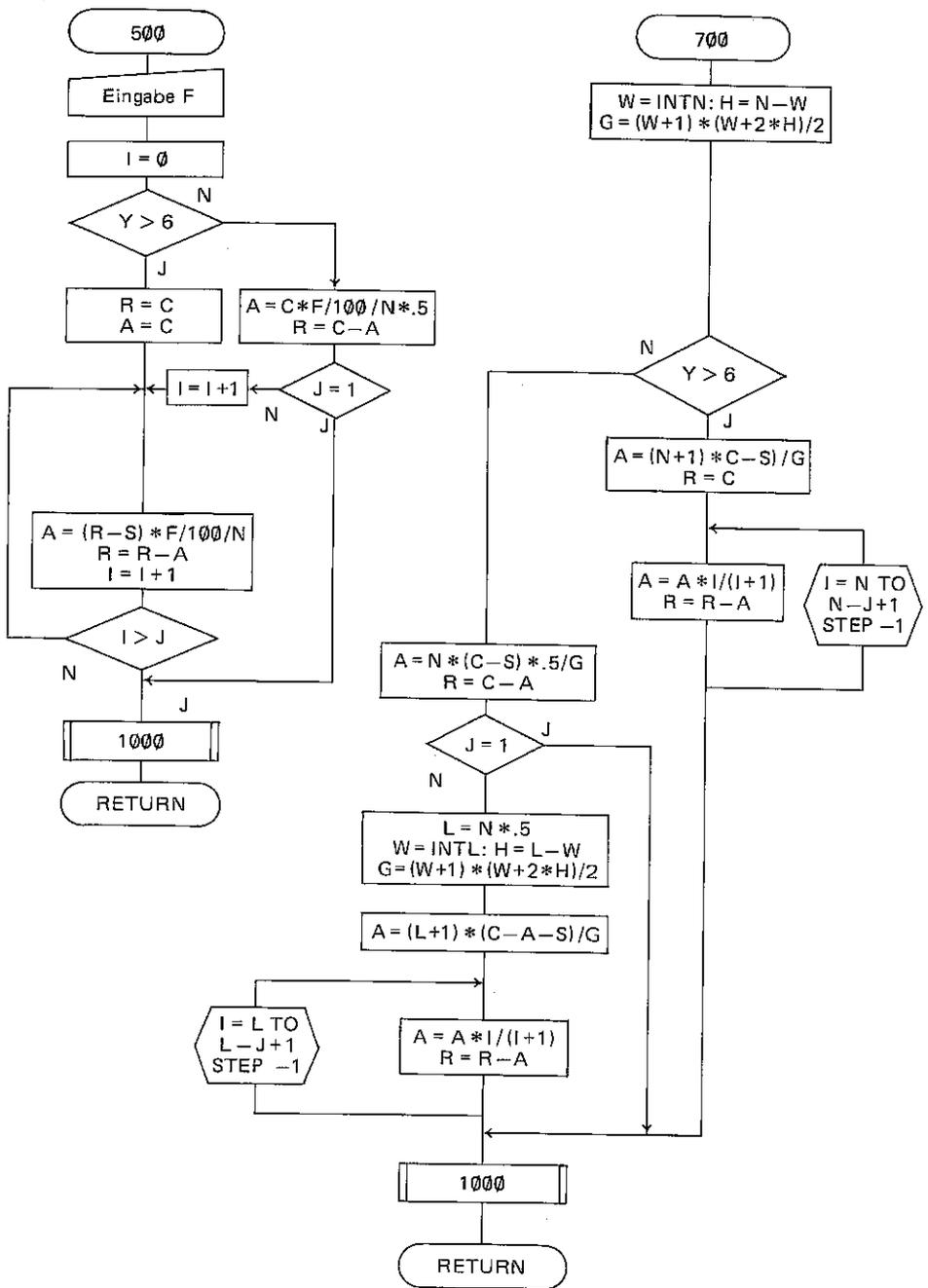
| Schritt-Nr. | Eingabe                                  | Anzeige                                         | Bemerkungen                          |
|-------------|------------------------------------------|-------------------------------------------------|--------------------------------------|
| 20          | <input type="button" value="ENTER"/>     | BUCH = 120192.30                                |                                      |
| 21          | <input type="button" value="ENTER"/>     | ABSCHREIBBARER<br>REST = 120192.30              |                                      |
| 22          | <input type="button" value="ENTER"/>     | NEUE RECHN (J/N)                                |                                      |
| 23          | J <input type="button" value="ENTER"/>   | LINEAR = 1<br>DEGRESSIV = 2<br>DIGITAL = 3<br>? |                                      |
| 24          | 3 <input type="button" value="ENTER"/>   | PERIODE =                                       |                                      |
| 25          | 2 <input type="button" value="ENTER"/>   | ABSCH = 9038.46                                 | 2. Abschreibung (digital)            |
| 26          | <input type="button" value="ENTER"/>     | BUCH = 101730.76                                |                                      |
| 27          | <input type="button" value="ENTER"/>     | ABSCHREIBBARER<br>REST = 101730.76              |                                      |
| 28          | <input type="button" value="ENTER"/>     | NEUE RECHN (J/N)                                |                                      |
| 29          | J <input type="button" value="ENTER"/>   | LINEAR = 1<br>DEGRESSIV = 2<br>DIGITAL = 3<br>? |                                      |
| 30          | 3 <input type="button" value="ENTER"/>   | PERIODE =                                       |                                      |
| 31          | 3 <input type="button" value="ENTER"/>   | ABSCH = 8653.84                                 | 3. Abschreibung (digital)            |
| 32          | <input type="button" value="ENTER"/>     | BUCH = 93076.92                                 |                                      |
| 33          | <input type="button" value="ENTER"/>     | ABSCHREIBBARER<br>REST = 93076.92               |                                      |
| 34          | <input type="button" value="ENTER"/>     | NEUR RECHN (J/N)                                |                                      |
| 35          | J <input type="button" value="ENTER"/>   | LINEAR = 1<br>DEGRESSIV = 2<br>DIGITAL = 3<br>? |                                      |
| 36          | 2 <input type="button" value="ENTER"/>   | PERIODE =                                       |                                      |
| 37          | 1 <input type="button" value="ENTER"/>   | FAKTOR =                                        | Degressionsmultiplikator<br>eingeben |
| 38          | 250 <input type="button" value="ENTER"/> | ABSCH = 6250.00                                 | 1. Abschreibung (degressiv)          |

# TASTENBETÄTIGUNG

| Schritt-Nr. | Eingabe                                  | Anzeige                                         | Bemerkungen                 |
|-------------|------------------------------------------|-------------------------------------------------|-----------------------------|
| 39          | <input type="button" value="ENTER"/>     | BUCH = 1      118750.00                         |                             |
| 40          | <input type="button" value="ENTER"/>     | ABSCHREIBBARER<br>REST =            118750.00   |                             |
| 41          | <input type="button" value="ENTER"/>     | NEUE RECHN (J/N)                                |                             |
| 42          | J <input type="button" value="ENTER"/>   | LINEAR = 1<br>DEGRESSIV = 2<br>DIGITAL = 3<br>? |                             |
| 43          | 2 <input type="button" value="ENTER"/>   | PERIODE =                                       |                             |
| 44          | 2 <input type="button" value="ENTER"/>   | FAKTOR =                                        |                             |
| 45          | 250 <input type="button" value="ENTER"/> | ABSCH =            10687.50                     |                             |
| 46          | <input type="button" value="ENTER"/>     | BUCH =            96187.50                      |                             |
| 47          | <input type="button" value="ENTER"/>     | ABSCHREIBBARER<br>REST =            96187.50    |                             |
| 48          | <input type="button" value="ENTER"/>     | NEUE RECHN (J/N)                                |                             |
| 49          | J <input type="button" value="ENTER"/>   | LINEAR = 1<br>DEGRESSIV = 2<br>DIGITAL = 3<br>? |                             |
| 50          | 2 <input type="button" value="ENTER"/>   | PERIODE =                                       |                             |
| 51          | 3 <input type="button" value="ENTER"/>   | FAKTOR =                                        |                             |
| 52          | 250 <input type="button" value="ENTER"/> | ABSCH =            9618.75                      | 3. Abschreibung (degressiv) |
| 53          | <input type="button" value="ENTER"/>     | BUCH =            86568.75                      |                             |
| 54          | <input type="button" value="ENTER"/>     | ABSCHREIBBARER<br>REST =            86568.75    |                             |
| 55          | <input type="button" value="ENTER"/>     | NEUE RECHN (J/N)                                |                             |
| 56          | N <input type="button" value="ENTER"/>   | >                                               | Programmende                |

# FLUSSDIAGRAMM





# PROGRAMMLISTING

```

10:"A":CLEAR
20:INPUT "NEU=";C,"REST
 =" ;S,"DAUER=";N
30:INPUT "1. JAHR=";Y
40:WAIT 100:PRINT "LINE
 AR=1"
50:PRINT "DEGRESSIV=2":
 PRINT "DIGITAL=3"
60:INPUT Z:WAIT
70:IF Z<>1 AND Z<>2 AND
 Z<>3 GOTO 4
80:INPUT "PERIODE=";J
90:ON Z GOSUB 300,500,7
 00
100:Q=R-S:WAIT 100
110:PRINT "ABSCHREIBBARE
 R"
120:WAIT :PRINT "REST=";
 R:R=0
130:INPUT "NEUE RECHN.(J
 /N)";I$
140:IF LEFT$(I$,1)="J"
 GOTO 40
150:END
300:A=(C-S)/N:I=0
310:IF Y>6 GOTO 400
320:A=.5*A:R=C-A:I=I+1
330:IF J=1 GOTO 450
340:A=2*A:GOTO 420
400:R=C
420:R=R-A:I=I+1
430:IF I>J GOTO 450
440:GOTO 420
450:GOSUB 1000
460:RETURN
500:INPUT "FAKTOR=";F
510:I=0
520:IF Y>6 GOTO 600
530:A=.5*C*F/100/N:R=C-A
540:IF J=1 GOTO 650
550:I=I+1:GOTO 620
600:R=C:A=C
620:A=(R-S)*F/100/N:R=R-
 A:I=I+1
630:IF I>J GOTO 650
640:GOTO 620
650:GOSUB 1000
660:RETURN
700:W= INT N:H=N-W:G=(W+
 1)*(W+2*H)/2
710:IF Y>6 GOTO 900
720:A=.5*N*(C-S)/G
730:R=C-A
740:IF J=1 GOTO 950
750:L=N-.5
760:W= INT L:H=L-W:G=(W+
 1)*(W+2*H)/2
770:A=(L+1)*(C-A-S)/G
780:FOR I=L TO N-J+1
 STEP -1

```

# SPEICHERINHALT

|     |                               |
|-----|-------------------------------|
| A   | A <sub>j</sub> (Abschreibung) |
| B   |                               |
| C   | C (Anschaffungspreis)         |
| D   |                               |
| E   |                               |
| F   | f (Degressionsmultiplikator)  |
| G   | (W+1) (W+2H)/2                |
| H   | n-INT (n)                     |
| I   | ✓                             |
| J   | j (Periode)                   |
| K   |                               |
| L   | n-05                          |
| M   |                               |
| N   | n (Lebensdauer)               |
| O   |                               |
| P   |                               |
| Q   | $C_j = R_j - S$               |
| R   | R (Restbuchwert)              |
| S   | S (Restwert)                  |
| T   |                               |
| U\$ | ✓                             |
| V   |                               |
| W   | INT (n), INT (n-0.5)          |
| X   |                               |
| Y   | Monate des 1. Jahres          |
| Z   | "Markierung"                  |

```

790:A=A*I/(I+1):R=R-A
800:NEXT I
810:GOTO 950
900:A=(N+1)*(C-S)/G:R=C
910:FOR I=N TO N-J+1
 STEP -1
920:A=A*I/(I+1):R=R-A
930:NEXT I
950:GOSUB 1000
960:RETURN
1000:WAIT :USING "####
 ##.##"
1005:IF R<0 THEN LET A=
 A+R:R=0
1010:PRINT "ABSCH=";A
1020:PRINT "BUCH=";R
1030:A=0
1040:RETURN

```

## ÜBERBLICK

Bei einem Ratenkredit ist meistens der Kreditzins in einem monatlichen Prozentsatz vom sog. Bruttoanfangsdarlehen ausgedrückt. D. h., daß der Tilgungsanteil der Annuitätenzahlungen nicht bei der Verzinsung berücksichtigt wird. Oft ist auch das Bruttoanfangsdarlehen höher als der eigentliche Kreditbetrag, da eine Bearbeitungsgebühr mitfinanziert wird. Die eigentliche Auszahlung kann sogar unter 100% des Kreditbetrages liegen (Disagio), wenn z.B. eine Vermittlungsgebühr u. ä. sofort fällig wird. Das folgende Programm berücksichtigt diese Parameter und bestimmt den Effektivzins.

## INHALT (Formel)

PV Nettoanfangsdarlehen (Kreditvolumen)  
 A mitfinanzierte Gebühren u. ä. in % von PV  
 n Anzahl der zu leistenden Raten  
 q monatlicher Zinssatz  
 D Auszahlung (PV-Disagio) in % von PV  
 I Effektivzins

Es gilt dann:

$$PMT = -PV (1 + A/100) (1 + nq)/n$$

$$PV = PMT \frac{1 - (1 + i)^{-n}}{i}$$

falls  $D = 100\%$  ist bzw.

$$PV (D/100) = PMT \frac{1 - (1 + i)^{-n}}{i}$$

falls  $D < 100\%$  ist.

$$I = \left( \left( 1 + \frac{i}{n100} \right)^n - 1 \right) 100$$

## ALEITUNG ZUM BETRIEB

1. Das Programm mit **DEF** A starten.
2. Eingabe von PV, A, n, Q, D.
3. Ausgabe von PMT, Effektivzins I

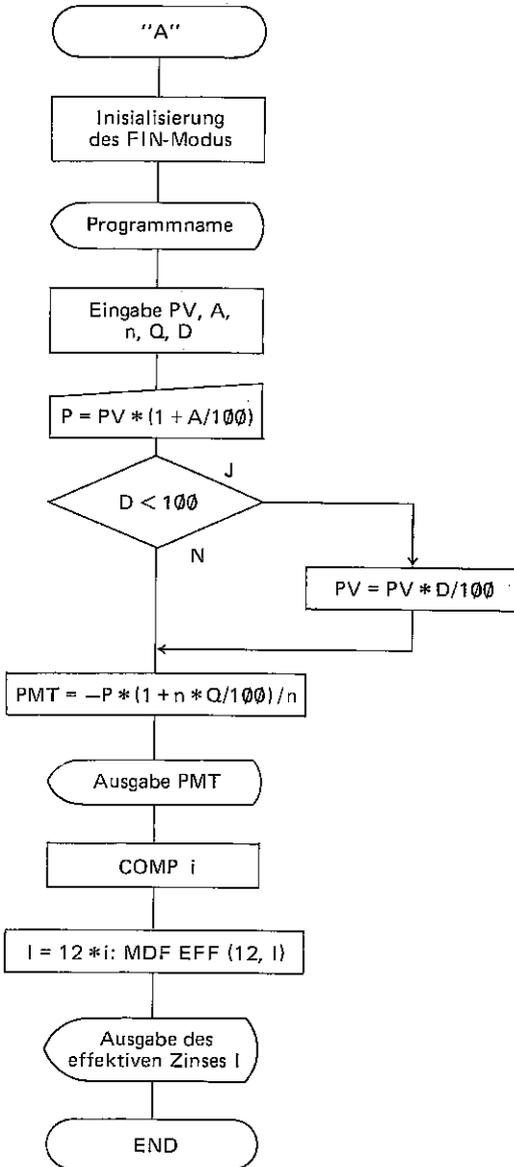
## BEISPIEL

Ein Ratenkredit über DM 10.000,-läuft über 24 Monate bei einem monatlichen Zinssatz von 0,43%. Es ist eine Bearbeitungsgebühr von 2% des Kreditvolumens mitzufinanzieren, sowie eine Kreditvermittlungsgebühr von 2% sofort fällig. Die Raten sind jeweils am Monatsende, beginnend einen Monat nach Auszahlung, zu leisten. Wie hoch ist die monatliche Belastung und der Effektivzins?

## TASTENBETÄTIGUNG

| Schritt-Nr. | Eingabe            | Anzeige             | Bemerkungen                           |
|-------------|--------------------|---------------------|---------------------------------------|
| 1           | <b>DEF</b> A       | RATENKREDIT<br>PV = | Kreditvolumen eingeben                |
| 2           | 10000 <b>ENTER</b> | A =                 | Mitfinanzierte Gebühren in % eingeben |
| 3           | 2 <b>ENTER</b>     | n =                 | Anzahl der Raten eingeben             |
| 4           | 24 <b>ENTER</b>    | Q =                 | Monatlichen Zinssatz eingeben         |
| 5           | .43 <b>ENTER</b>   | AUSZAHLUNG%         | Auszahlung in % von PV eingeben       |
| 6           | 98 <b>ENTER</b>    | PMT = -468.86       | Monatliche Rate                       |
| 7           | <b>ENTER</b>       | EFF = 14.53         | Effektiver Jahreszins                 |
| 8           | <b>ENTER</b>       | >                   | Programmende                          |

# FLUSSDIAGRAMM



## PROGRAMMLISTING

```

10:"A":CLEAR :WAIT 100
20:ERASE FIN
30:PRINT "RATENKREDIT"
40:INPUT " PV ="; PV ,"
 A=";A;" n ="; n ,"Q=
 ";Q
50:INPUT "AUSZAHLUNG:";
 D
60:P = PV *(1+A/100)
70:IF D<100 GOTO 200
80:PMT =-P*(1+ n *Q/100
)/ n
90:WAIT :USING "#####.#
 #"
100:PRINT " PMT ="; PMT
110:COMP i :I=12* i
120:I=MDF EFF (12,I)
130:PRINT "EFF=";I
140:END
200:PV = PV *D/100
210:GOTO 80

```

## SPEICHERINHALT

|   |                                  |
|---|----------------------------------|
| A | A (mitfinanzierte Gebühren in %) |
| B |                                  |
| C |                                  |
| D | D (= PV — Disagio)               |
| E |                                  |
| F |                                  |
| G |                                  |
| H |                                  |
| I | Effektiver Jahreszins            |
| J |                                  |
| K |                                  |
| L |                                  |
| M |                                  |
| N |                                  |
| O |                                  |
| P | PV (1 + A/100)                   |
| Q | q (monatlicher Zinssatz)         |
| R |                                  |
| S |                                  |
| T |                                  |
| U |                                  |
| V |                                  |
| W |                                  |
| X |                                  |
| Y |                                  |
| Z |                                  |

## ÜBERBLICK

Der t-Test liefert die Möglichkeit zu testen, ob ein aus einer Stichprobe gewonnener Mittelwert mit dem erwarteten Mittelwert übereinstimmt. Er ist schon bei kleinen Stichproben anwendbar. Bei vorgegebener Sicherheitswahrscheinlichkeit wird die Schätzung durch Vergleichen mit den Werten der t-Verteilung (Tabellen) angenommen oder verworfen.

## INHALT (Formel)

- $\bar{x}$  Mittelwert der Stichprobe
- $\mu_0$  erwarteter Mittelwert
- n Stichprobenumfang
- S Standardabweichung

$$t = \frac{\bar{x} - \mu_0}{S} \sqrt{n}$$

Wobei gilt:

$$S^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$$

Daten aus dem STAT-Modus werden im BASIC-Modus den folgenden Variablen zugeordnet:

| Variable | Statistische Daten |
|----------|--------------------|
| U        | $\Sigma y^2$       |
| V        | $\Sigma y$         |
| W        | $\Sigma xy$        |
| X        | $\Sigma x^2$       |
| Y        | $\Sigma x$         |
| Z        | n                  |

## ANLEITUNG ZUM BETRIEB

1. Eingabe der Stichprobendaten im STAT-Modus.
2. Umschalten auf RUN-Modus.
3. Starten des Programms mit **DEF** A .
4. Ausgabe des T-Werts.

## BEISPIEL

Es werden 5 Transistoren auf ihre Lebensdauer getestet. Es ergeben sich die Werte (in Stunden) von

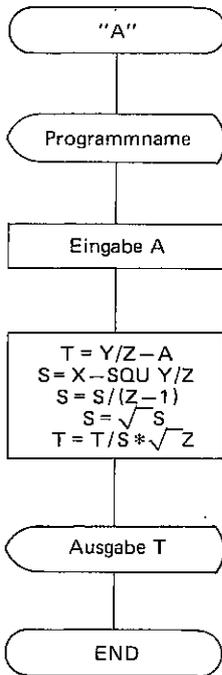
7400, 8000, 8300, 8700, 9900.

Es wird eine mittlere Lebensdauer von 8100 Stunden verlangt.

## TASTENBETÄTIGUNG

| Schritt-Nr. | Eingabe               | Anzeige                | Bemerkungen                            |
|-------------|-----------------------|------------------------|----------------------------------------|
| 1           | <b>CAL</b> <b>CAL</b> | 0.                     | Einschalten des STAT-Modus             |
| 2           | 7400 <b>ENTER</b>     | 1.                     | Eingabe der Daten in den STAT-Speicher |
| 3           | 8000 <b>ENTER</b>     | 2.                     |                                        |
| 4           | 8300 <b>ENTER</b>     | 3.                     |                                        |
| 5           | 8700 <b>ENTER</b>     | 4.                     |                                        |
| 6           | 9900 <b>ENTER</b>     | 5.                     |                                        |
| 7           | <b>ENTER</b>          | >                      | Umschalten auf RUN-Modus               |
| 8           | <b>DEF</b> A          | T-TEST<br>MITTELWERT = | Erwarteten Mittelwert eingeben         |
| 9           | 8100 <b>ENTER</b>     | T = 8.615E-01          | T-Wert                                 |
| 10          | <b>ENTER</b>          | >                      | Programmende                           |

# FLUSSDIAGRAMM



## PROGRAMMLISTING

```

10:"A":WAIT 100
20:PRINT "T-TEST"
30:USING "###.###^":
 WAIT
40:INPUT "MITTELWERT=";
 A
50:T=Y/Z-A:S=X-SQU Y/Z
60:S=S/(Z-1):S=J8
70:T=JZ*T/S
80:PRINT "T=";T
90:END

```

## SPEICHERINHALT

|   |                        |
|---|------------------------|
| A | $\mu_0$                |
| B |                        |
| C |                        |
| D |                        |
| E |                        |
| F |                        |
| G |                        |
| H |                        |
| I |                        |
| J |                        |
| K |                        |
| L |                        |
| M |                        |
| N |                        |
| O |                        |
| P |                        |
| Q |                        |
| R |                        |
| S | S (Standardabweichung) |
| T | t                      |
| U | $\sum y_i^2$           |
| V | $\sum y_i$             |
| W | $\sum x_i y_i$         |
| X | $\sum x_i^2$           |
| Y | $\sum x_i$             |
| Z | n                      |

## ÜBERBLICK

Bei einer Verteilung, bei der  $m$  verschiedene Ausprägungen mit positiver Wahrscheinlichkeit auftreten können, ist die erwartete Verteilung (Hypothesen) gegen die in einer Stichprobe beobachtete zu testen. Der Chi-Quadrat-Test ist dazu eine bewährte Methode.

## INHALT (Formel)

$$x^2 = \sum_{i=1}^n \frac{(n_i - np_i)^2}{np_i}$$

wobei  $n_i$  die beobachtete Häufigkeit in der  $i$ -ten Ausprägung  
 $p_i$  die Wahrscheinlichkeit der  $i$ -ten Ausprägung  
 $n$  der Stichprobenumfang

ist.

## ANLEITUNG ZUM BETRIEB

1. Das Programm wird mit **DEF** A gestartet.
2. Eingabe von  $M$ ,  $N_i$ ,  $Q_i$  ( $Q_i = N * P_i$ )
3. Ausgabe von  $x^2$

## BEISPIEL

In der Bevölkerung ist der Anteil der Frauen 52%. Es werden in drei Betrieben A, B, C, in denen die Beschäftigung unabhängig vom Geschlecht ist, eine Stichprobe durchgeführt. Zu erwarten wären folgende Zahlen:

| Firma | Anzahl der Beschäftigten | Anzahl der weiblichen Beschäftigten | Anzahl der männlichen Beschäftigten |
|-------|--------------------------|-------------------------------------|-------------------------------------|
| A     | 10000                    | 5200                                | 4800                                |
| B     | 5650                     | 2938                                | 2712                                |
| C     | 3290                     | 1711                                | 1579                                |

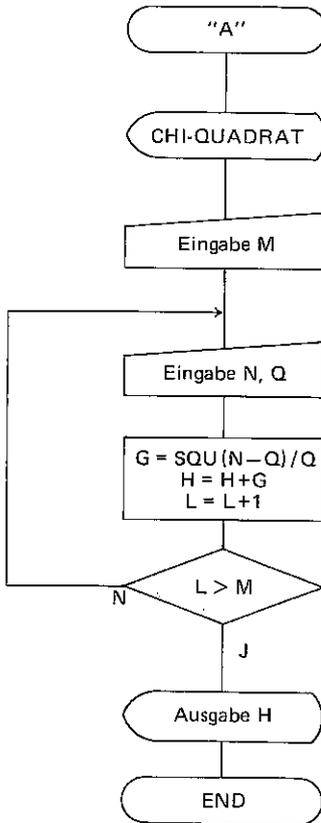
Beobachtet wurde bei der Untersuchung folgende Verteilung:

| Firma | Anzahl der weiblichen Beschäftigten | Anzahl der männlichen Beschäftigten |
|-------|-------------------------------------|-------------------------------------|
| A     | 3100                                | 6900                                |
| B     | 2900                                | 2750                                |
| C     | 1800                                | 1490                                |

## TASTENBETÄTIGUNG

| Schritt-Nr. | Eingabe                                 | Anzeige                   | Bemerkungen                                   |
|-------------|-----------------------------------------|---------------------------|-----------------------------------------------|
| 1           | <input type="text" value="DEF"/> A      | CHI-QUADRAT<br>MERKMALE = | Anzahl der Merkmale eingeben                  |
| 2           | 6 <input type="text" value="ENTER"/>    | N =                       | Häufigkeit des 1. Merkmals eingeben           |
| 3           | 3100 <input type="text" value="ENTER"/> | Q =                       | Erwartete Häufigkeit des 1. Merkmals eingeben |
| 4           | 5200 <input type="text" value="ENTER"/> | N =                       | Häufigkeit des 2. Merkmals eingeben           |
| 5           | 2900 <input type="text" value="ENTER"/> | Q =                       | Erwartete Häufigkeit des 2. Merkmals eingeben |
| 6           | 2938 <input type="text" value="ENTER"/> | N =                       |                                               |
| 7           | 1800 <input type="text" value="ENTER"/> | Q =                       |                                               |
| 8           | 1711 <input type="text" value="ENTER"/> | N =                       |                                               |
| 9           | 6900 <input type="text" value="ENTER"/> | Q =                       |                                               |
| 10          | 4800 <input type="text" value="ENTER"/> | N =                       |                                               |
| 11          | 2750 <input type="text" value="ENTER"/> | Q =                       |                                               |
| 12          | 2712 <input type="text" value="ENTER"/> | N =                       |                                               |
| 13          | 1490 <input type="text" value="ENTER"/> | Q =                       |                                               |
| 14          | 1579 <input type="text" value="ENTER"/> | CHI-QUADRAT =<br>1777.49  |                                               |
| 15          | <input type="text" value="ENTER"/>      | >                         | Programmende                                  |

# FLUSSDIAGRAMM



## PROGRAMMLISTING

```

10:"A":CLEAR :WAIT 100
20:PRINT "CHI-QUADRAT"
30:INPUT "MERKMALE=":M
40:FOR L=1 TO M
50:INPUT "N=":N:"Q=":Q
60:G=SQU (N-Q)/Q
70:H=H+G
80:NEXT L
90:PRINT "CHI-QUADRAT="
100:USING "#####.##":
 WAIT
110:PRINT H
120:END

```

## SPEICHERINHALT

|   |                        |
|---|------------------------|
| A |                        |
| B |                        |
| C |                        |
| D |                        |
| E |                        |
| F |                        |
| G | $(n_i, np_i)^2 / np_i$ |
| H |                        |
| I |                        |
| J |                        |
| K |                        |
| L | √                      |
| M | m                      |
| N | $n_i$                  |
| O |                        |
| P |                        |
| Q | $np_i$                 |
| R |                        |
| S |                        |
| T |                        |
| U |                        |
| V |                        |
| W |                        |
| X |                        |
| Y |                        |
| Z |                        |

## ÜBERBLICK

Die Simpson-Regel ist eine viel benutzte Methode zur approximativen Berechnung des Wertes eines bestimmten Integrals.

## INHALT

Die Fläche unter einer Kurve werde in  $n$  Teile gleicher Breite

$$h = \frac{b - a}{n}$$

zerlegt.

Bezeichnung:  $x_k = a + kh$  ( $k = 0, 1, \dots, n$ )

Dann gilt:

$$\int_a^b f(x) dx = \sum_{k=0}^{n-1} \int_{x_k}^{x_{k+1}} f(x) dx$$

und  $\int_{x_k}^{x_{k+1}} f(x) dx$  wird approximativ vermöge

$$S = \frac{x_{k+1} - x_k}{6} (f(x_k) + 4f(\frac{x_k + x_{k+1}}{2}) + f(x_{k+1}))$$

berechnet.

## ANLEITUNG ZUM BETRIEB

1. Das Programm wird mit     gestartet.
2. Eingabe von A, B, N.
3. Ausgabe von S (approximativer Wert des Integrals).

## BEISPIEL

Berechnung des Integrals

$$\int_1^5 x^3 + 2x^2 - x + 2 dx \quad \text{mit } n = 4$$

**Bemerkung:** Die Funktion wird als FUntprogramm in Zeile 200 geschrieben.  
 Dazu wird im PRO-Modus folgende Unterroutine eingegeben:

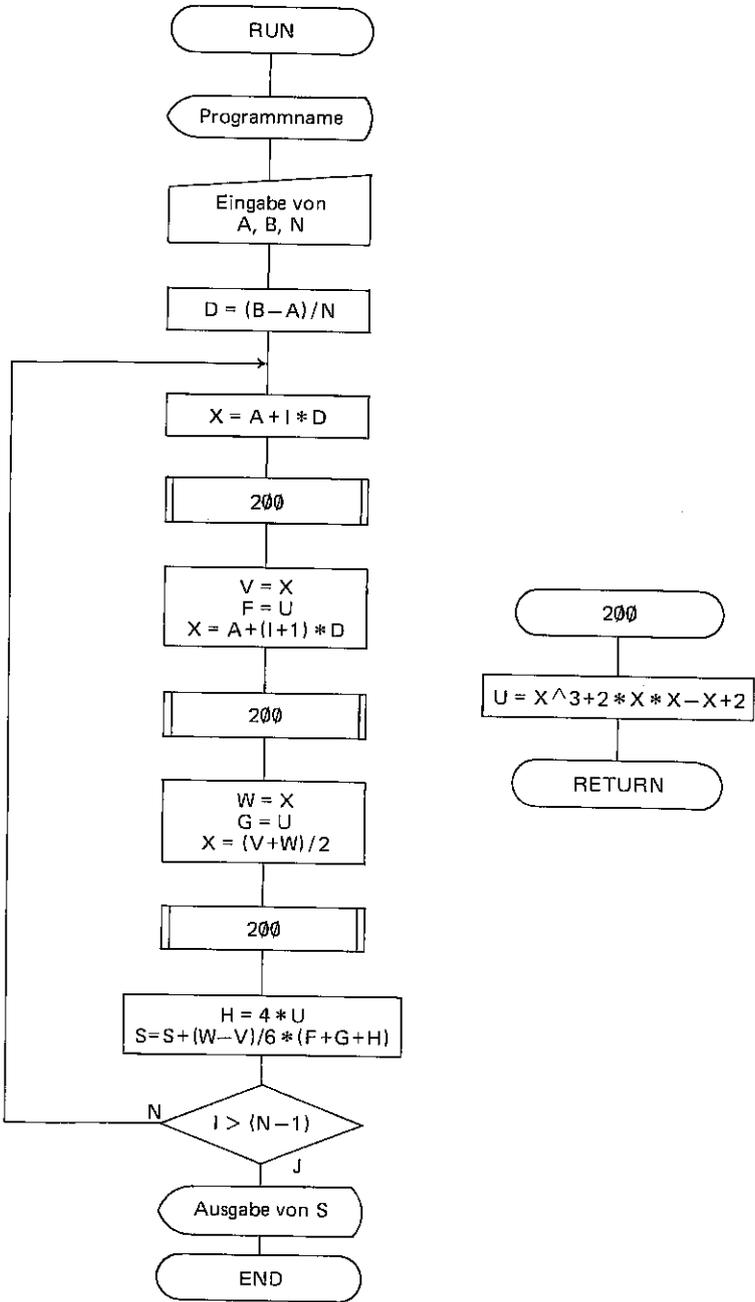
200: U = X ^ 3 + 2 \* X \* X - X + 2    **ENTER**

210: RETURN    **ENTER**

## TASTENBETÄTIGUNG

| Schritt-Nr. | Eingabe                     | Anzeige                   | Bemerkungen                        |
|-------------|-----------------------------|---------------------------|------------------------------------|
| 1           | [R] [U] [N]<br><b>ENTER</b> | SIMPSON REGEL<br>A =      | Anfangspunkt eingeben              |
| 2           | 1 <b>ENTER</b>              | B =                       | Endpunkt eingeben                  |
| 3           | 5 <b>ENTER</b>              | N =                       | Anzahl der Teilintervalle eingeben |
| 4           | 4 <b>ENTER</b>              | INTEGRAL =<br>234.6666667 | Nerecimeter Wert des Integrals     |
| 5           | <b>ENTER</b>                | >                         | Programmende                       |

# FLUSSDIAGRAMM



## PROGRAMMLISTING

```

10: CLEAR : WAIT 100
20: PRINT "SIMPSON REGEL
"
30: INPUT "A="; A, "B="; B,
" N="; N
40: D=(B-A)/N
50: FOR I=0 TO N-1
60: X=A+I*D
70: GOSUB 200
80: V=X; F=U
90: X=A+(I+1)*D
100: GOSUB 200
110: W=X; G=U
120: X=(V+W)/2
130: GOSUB 200
140: H=4*U
150: S=S+(W-V)/6*(F+G+H)
160: NEXT I
170: PRINT "INTEGRAL="
180: WAIT : PRINT S
190: END
200: U=X^3+2*X*X-X+2
210: RETURN

```

## SPEICHERINHALT

|   |                                   |
|---|-----------------------------------|
| A | a (Anfangspunkt)                  |
| B | b (Endpunkt)                      |
| C |                                   |
| D | $(b - a)/n$                       |
| E |                                   |
| F | $f(x_k)$                          |
| G | $f(x_{k+1})$                      |
| H | $4f((x_k + x_{k+1})/2)$           |
| I | $\sqrt{\quad}$                    |
| J |                                   |
| K |                                   |
| L |                                   |
| M |                                   |
| N | n (Anzahl der Teilintervalle)     |
| O |                                   |
| P |                                   |
| Q |                                   |
| R |                                   |
| S | S (approximativer Integralwert)   |
| T |                                   |
| U | $f(x)$                            |
| V | $x_k$                             |
| W | $x_{k+1}$                         |
| X | $x_k, x_{k+1}, (x_k + x_{k+1})/2$ |
| Y |                                   |
| Z |                                   |

## ÜBERBLICK

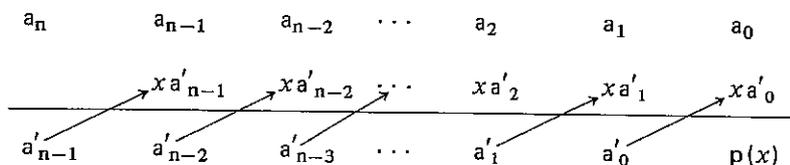
Auch in vielen außermathematischen Fragestellungen (z. B. in Wirtschaftswissenschaften, Physik, Chemie etc.) müssen die Werte von Polynomen berechnet werden. Dieses Programm berechnet die Werte eines Polynoms nach dem Horner-Schema.

## INHALT

Ein Polynom (vom Grade höchstens  $n$ ) hat die Form:

$$p(x) = a_0 + a_1x + a_2x^2 + \dots + a_{n-1}x^{n-1} + a_nx^n$$

Im ersten Teil des Programms werden die Koeffizienten  $a_0, a_1, \dots, a_n$  eingelesen. Der zweite Teil berechnet dann den Wert des Polynoms für einzugebende Zahlen  $x$  nach dem Horner-Schema.



## ANLEITUNG ZUM BETRIEB

1. Das Programm wird mit **DEF** A gestartet.
2. Eingabe der Koeffizienten  $a_0, a_1, \dots, a_n$ .
3. Eingabe von  $x$ .
4. Ausgabe von  $P(x)$ .

## BEISPIEL

Es sollen die Werte des Polynoms

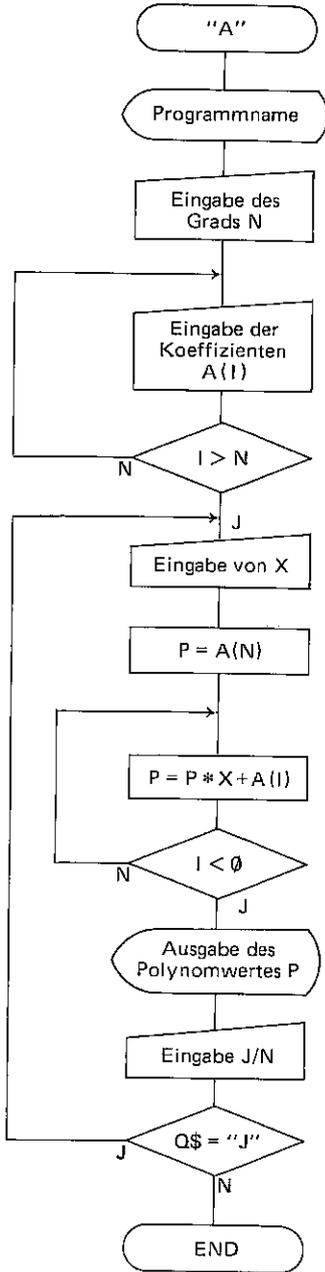
$$p(x) = 2x^4 - 8x^3 + 4x^2 + 1$$

für  $x = 1, 10$  berechnet werden.

# TASTENBETÄTIGUNG

| Schritt-Nr. | Eingabe         | Anzeige               | Bemerkungen                                            |
|-------------|-----------------|-----------------------|--------------------------------------------------------|
| 1           | <b>DEF</b> A    | POLYNOME<br>GRAD =    | Es ist der Grad des Polynoms einzugeben                |
| 2           | 4 <b>ENTER</b>  | A(0.)<br>?            | $a_0$ eingeben                                         |
| 3           | 1 <b>ENTER</b>  | A(1.)<br>?            | $a_1$ eingeben                                         |
| 4           | 0 <b>ENTER</b>  | A(2.)<br>?            | $a_2$ eingeben                                         |
| 5           | 4 <b>ENTER</b>  | A(3.)<br>?            | $a_3$ eingeben                                         |
| 6           | -8 <b>ENTER</b> | A(4.)<br>?            | $a_4$ eingeben                                         |
| 7           | 2 <b>ENTER</b>  | X =                   | x eingeben                                             |
| 8           | 1 <b>ENTER</b>  | P(1.) = -1.           | Berechneter Wert von P(x)                              |
| 9           | <b>ENTER</b>    | WEITERE WERTE?<br>J/N | Wenn weiter Wert berechnet werden sollen, "J" eingeben |
| 10          | J <b>ENTER</b>  | X =                   | Neuen Wert von X eingeben                              |
| 11          | 10 <b>ENTER</b> | P(10) = 12401.        | Berechneter Wert von P(x)                              |
| 12          | <b>ENTER</b>    | WEITERE WERTE?<br>J/N |                                                        |
| 13          | N <b>ENTER</b>  | >                     | Programmende                                           |

# FLUSSDIAGRAMM



# PROGRAMMLISTING

```

10:"A":CLEAR :WAIT 100
20:PRINT "POLYNOME"
30:INPUT "GRAD=";N:DIM
 A(N)
40:FOR I=0 TO N
50:PRINT "A(";I;")=":
 INPUT A(I)
60:NEXT I
70:INPUT "X=";X
80:P=A(N)
90:FOR I=N-1 TO 0 STEP
 -1
100:P=P*X+A(I)
110:NEXT I
120:WAIT :PRINT "P(";X;")
)=";P
130:WAIT 100:PRINT "WEIT
 ERE WERTE?"
140:INPUT "J/N?";Q$
150:IF LEFT$(Q$,1)="J"
 GOTO 70
160:END

```

# SPEICHERINHALT

| A(I) | $a_i$         |
|------|---------------|
| B    |               |
| C    |               |
| D    |               |
| E    |               |
| F    |               |
| G    |               |
| H    |               |
| I    | ✓             |
| J    |               |
| K    |               |
| L    |               |
| M    |               |
| N    | n (Grad P(x)) |
| O    |               |
| P    | P(x)          |
| Q\$  | ✓             |
| R    |               |
| S    |               |
| T    |               |
| U    |               |
| V    |               |
| W    |               |
| X    | x             |
| Y    |               |
| Z    |               |

## ÜBERBLICK

Das Produkt einer Matrix A vom Typ  $m \times l$  und einer Matrix B vom Typ  $l \times n$  ist eine Matrix vom Typ  $m \times n$ . Die Koeffizienten dieses Matrizenprodukts werden berechnet.

## INHALT (Formel)

$$m \left\{ \underbrace{\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1l} \\ a_{21} & a_{22} & \cdots & a_{2l} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & & a_{ml} \end{pmatrix}}_l \underbrace{\begin{pmatrix} b_{11} & \cdots & b_{1n} \\ b_{21} & \cdots & b_{2n} \\ \vdots & & \vdots \\ b_{l1} & & b_{ln} \end{pmatrix}}_n \right. = \begin{pmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & & \vdots \\ c_{m1} & & c_{mn} \end{pmatrix}$$

Das Matrizenprodukt ist definiert vermöge:

$$c_{ij} = \sum_{k=1}^l a_{ik} b_{kj} \qquad \begin{matrix} i = 1, 2, \dots, m \\ j = 1, 2, \dots, n \end{matrix}$$

## ANLEITUNG ZUM BETRIEB

1. Das Programm mit DEF A starten.
2. Eingabe von M, L  
A (I, J)  
N  
B (J, K)
3. Ausgabe von C (I, K)

## BEISPIEL

Es soll das Produkt der Matrizen

$$A = \begin{pmatrix} 0 & 1 & 2 \\ 1 & 3 & 5 \\ 8 & \emptyset & 1 \end{pmatrix} \quad \text{und} \quad B = \begin{pmatrix} 1 & 3 \\ 4 & 5 \\ 6 & \emptyset \end{pmatrix}$$

berechnet werden.

Das Ergebnis ist eine Matrix der Form:

$$C = \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \\ C_{32} & C_{32} \end{pmatrix}$$

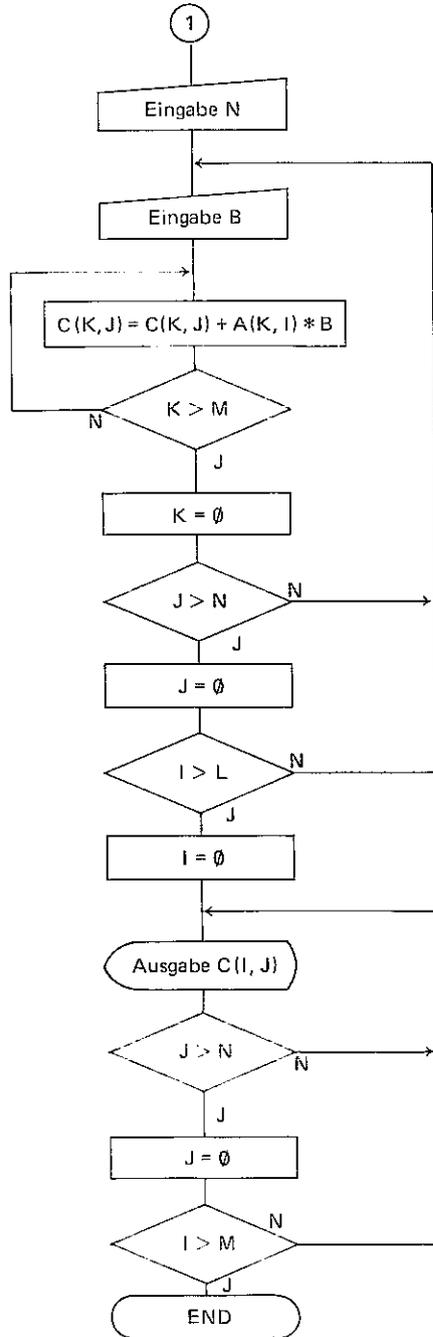
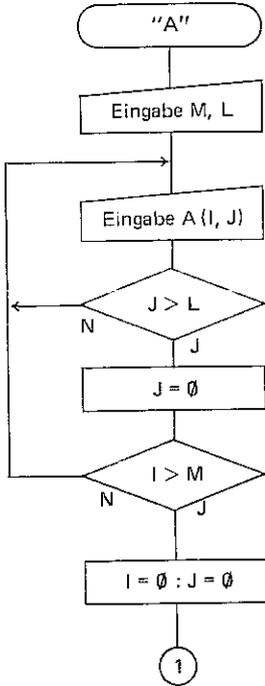
## TASTENBETÄTIGUNG

| Schritt-Nr. | Eingabe        | Anzeige          | Bemerkungen                  |
|-------------|----------------|------------------|------------------------------|
| 1           | <b>DEF</b> A   | M =              |                              |
| 2           | 3 <b>ENTER</b> | L =              |                              |
| 3           | 3 <b>ENTER</b> | A(1., 1.) =<br>? | Eingabe der Koeffizienten    |
| 4           | ∅ <b>ENTER</b> | A(1., 2.) =<br>? |                              |
| 5           | 1 <b>ENTER</b> | A(1., 3.) =<br>? |                              |
| 6           | 2 <b>ENTER</b> | A(2., 1.) =<br>? |                              |
| 7           | 1 <b>ENTER</b> | A(2., 2.) =<br>? |                              |
| 8           | 3 <b>ENTER</b> | A(2., 3.) =<br>? |                              |
| 9           | 5 <b>ENTER</b> | A(3., 1.) =<br>? |                              |
| 10          | 8 <b>ENTER</b> | A(3., 2.) =<br>? |                              |
| 11          | ∅ <b>ENTER</b> | A(3., 3.) =<br>? |                              |
| 12          | 1 <b>ENTER</b> | N =              | Spaltenzahl von B eingeben   |
| 13          | 2 <b>ENTER</b> | B(1., 1.) =<br>? | Koeffizienten von B eingeben |
| 14          | 1 <b>ENTER</b> | B(1., 2.) =<br>? |                              |
| 15          | 3 <b>ENTER</b> | B(2., 1.) =<br>? |                              |
| 16          | 4 <b>ENTER</b> | B(2., 2.) =<br>? |                              |
| 17          | 5 <b>ENTER</b> | B(3., 1.) =<br>? |                              |

# TASTENBETÄTIGUNG

| Schritt-Nr. | Eingabe                                          | Anzeige           | Bemerkungen                     |
|-------------|--------------------------------------------------|-------------------|---------------------------------|
| 18          | 6 <input type="button" value="ENTER"/>           | $B(3., 2.) = ?$   |                                 |
| 19          | $\emptyset$ <input type="button" value="ENTER"/> | $C(1., 1.) = 16.$ | Ausgabe der Koeffizienten von C |
| 20          | <input type="button" value="ENTER"/>             | $C(1., 2.) = 5.$  |                                 |
| 21          | <input type="button" value="ENTER"/>             | $C(2., 1.) = 43.$ |                                 |
| 22          | <input type="button" value="ENTER"/>             | $C(2., 2.) = 18.$ |                                 |
| 23          | <input type="button" value="ENTER"/>             | $C(3., 1.) = 14.$ |                                 |
| 24          | <input type="button" value="ENTER"/>             | $C(3., 2.) = 24.$ |                                 |
| 25          | <input type="button" value="ENTER"/>             | >                 | Programmende                    |

# FLUSSDIAGRAMM



## PROGRAMMLISTING

```

10:"A":CLEAR
20:INPUT "M=";M,"L=";L
30:M=M-1;L=L-1
40:DIM A(M,L)
50:WAIT 100
60:FOR I=0 TO M:FOR J=0
 TO L
70:PRINT "A(";I+1;",";J
 +1;")="
80:INPUT A(I,J):NEXT J:
 NEXT I
90:INPUT "N=";N:N=N-1
100:DIM C(M,N)
110:FOR I=0 TO L:FOR J=0
 TO N
120:PRINT "B(";I+1;",";J
 +1;")="
130:INPUT B
140:FOR K=0 TO M
150:C(K,J)=C(K,J)+A(K,I)
 *B
160:NEXT K:NEXT J:NEXT I
170:WAIT :FOR I=0 TO M:
 FOR J=0 TO N
180:PRINT "C(";I+1;",";J
 +1;")=";C(I,J)
190:NEXT J:NEXT I
200:END

```

## SPEICHERINHALT

|        |                             |
|--------|-----------------------------|
| A(M,L) | Elemente der Matrix A       |
| B      | Elemente der Matrix B       |
| C(M,N) | Elemente der Matrix C       |
| D      |                             |
| E      |                             |
| F      |                             |
| G      |                             |
| H      |                             |
| I      | ✓                           |
| J      | ✓                           |
| K      | ✓                           |
| L      | Spalten von A, Zeilen Von B |
| M      | Zeilen von A                |
| N      | Spalten von B               |
| O      |                             |
| P      |                             |
| Q      |                             |
| R      |                             |
| S      |                             |
| T      |                             |
| U      |                             |
| V      |                             |
| W      |                             |
| X      |                             |
| Y      |                             |
| Z      |                             |

## ÜBERBLICK

Die Lösung einer quadratischen Gleichung kann entweder eine reelle oder eine komplexe Zahl sein. Dieses Programm untersucht mit Hilfe der Diskriminante, ob die Lösung reell oder komplex ist und berechnet die Lösung in beiden Fällen.

## INHALT (Formel)

Mit der Diskriminante

$$D = b^2 - 4ac$$

läßt sich entscheiden, ob die Lösung der Gleichung

$$ax^2 + bx + c = 0 \quad (a \neq 0)$$

reell oder komplex ist.

- (1) Wenn  $D > 0$  ist, dann gibt es zwei verschiedene reelle Lösungen.
- (2) Wenn  $D = 0$  ist, dann gibt es eine reelle Lösung.
- (3) Wenn  $D < 0$  ist, dann gibt es zwei komplexe Lösungen.

Es gelten die folgenden Lösungsformeln:

$$\text{In Fall (1):} \quad x = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

$$\text{In Fall (2):} \quad x = -\frac{b}{2a}$$

$$\text{In Fall (3):} \quad x = \frac{-b + \sqrt{|b^2 - 4ac|} i}{2a}$$

## ANLEITUNG ZUM BETRIEB

1. Das Programm wird mit **DEF** A gestartet.
2. Eingabe: A, B, C
3. Ausgabe: Wurzel der Gleichung gemäß obigen Formeln.  
(Um die zweite Wurzel auszugeben, **ENTER**-Taste drücken.)

## BEISPIEL

$$(1) \quad 2x^2 + 8x - 10 = 0$$

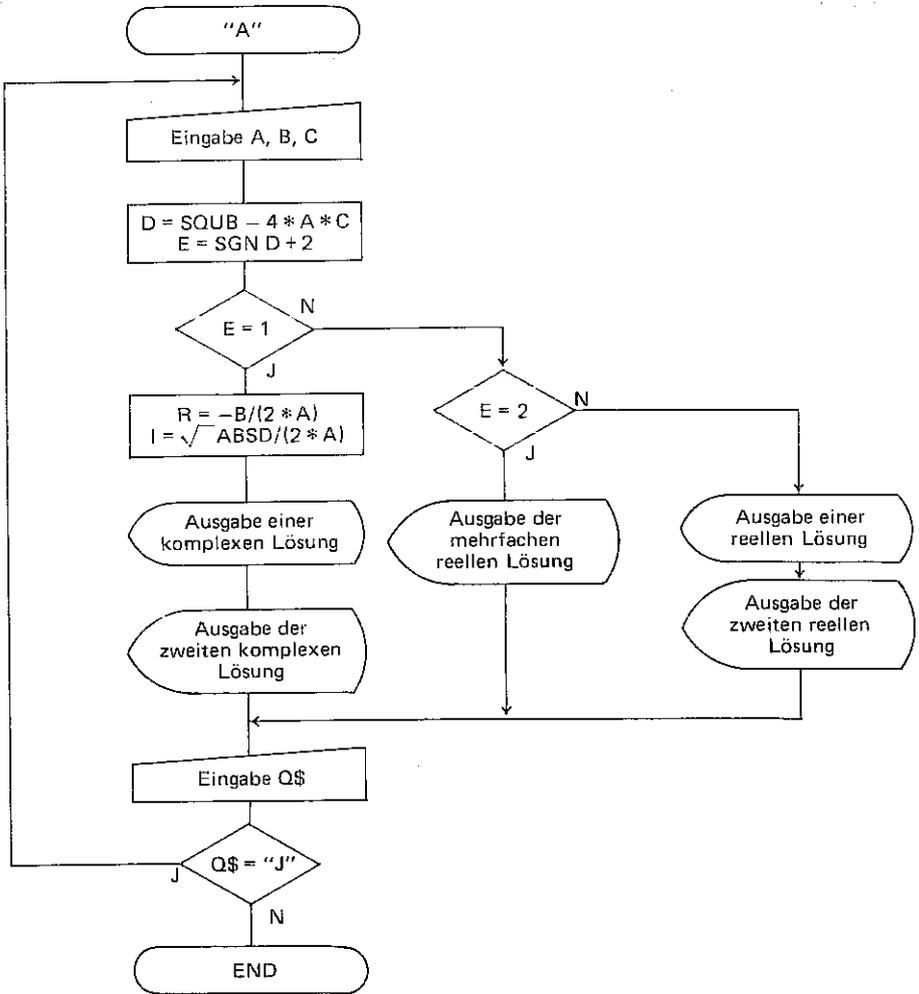
$$(2) \quad x^2 - 2x + 1 = 0$$

$$(3) \quad x^2 - x + 1 = 0$$

# TASTENBETÄTIGUNG

| Schritt-Nr. | Eingabe          | Anzeige                                          | Bemerkungen                             |
|-------------|------------------|--------------------------------------------------|-----------------------------------------|
| 1           | <b>DEF</b> A     | A =                                              | Eingabe von A                           |
| 2           | 2 <b>ENTER</b>   | B =                                              | Eingabe von B                           |
| 3           | 8 <b>ENTER</b>   | C =                                              | Eingabe von C                           |
| 4           | -10 <b>ENTER</b> | 1. WURZEL =                                      | 1. Ausgabe der ersten reellen Wurzel    |
| 5           | <b>ENTER</b>     | 2. WURZEL =                                      | -5. Ausgabe der zweiten reellen Wurzel  |
| 6           | <b>ENTER</b>     | WEITER (J/N) ?                                   | J oder N eingeben                       |
| 7           | J <b>ENTER</b>   | A =                                              | Eingabe von A                           |
| 8           | 1 <b>ENTER</b>   | B =                                              | Eingabe von B                           |
| 9           | -2 <b>ENTER</b>  | C =                                              | Eingabe von C                           |
| 10          | 1 <b>ENTER</b>   | WURZEL =                                         | 1. Ausgabe der Wurzel                   |
| 11          | <b>ENTER</b>     | WEITER (J/N) ?                                   | J oder N eingeben                       |
| 12          | J <b>ENTER</b>   | A =                                              | Eingabe von A                           |
| 13          | 1 <b>ENTER</b>   | B =                                              | Eingabe von B                           |
| 14          | -1 <b>ENTER</b>  | C =                                              | Eingabe von C                           |
| 15          | 1 <b>ENTER</b>   | WURZEL = (RE; IM)<br>1. WURZEL =<br>(0.50; 0.86) | 1. Ausgabe der ersten komplexen Wurzel  |
| 16          | <b>ENTER</b>     | 2. WURZEL =<br>(0.50; -0.86)                     | 2. Ausgabe der zweiten komplexen Wurzel |
| 17          | <b>ENTER</b>     | WEITER (J/N) ?                                   | J oder N eingeben                       |
| 18          | N <b>ENTER</b>   | >                                                | Programmende                            |

# FLUSSDIAGRAMM



## PROGRAMMLISTING

```

10:"A":CLEAR
20:INPUT "A=";A,"B=";B,
 "C=";C
30:D=SQU B-4*A*C:E=SGN
 D+2
40:ON E GOTO 100,200,30
 0
100:REM KOMPLEX
110:R=-B/(2*A):I=√ABS D/
 (2*A)
120:WAIT 100:PRINT "WURZ
 EL=(RE;IM)"
130:PRINT "1.WURZEL=":
 WAIT
140:PRINT USING "###.##"
 ;"(";R;" ";I;")":
 USING
150:I=-I;Z=Z+1
160:IF Z=1 THEN WAIT 100
 :PRINT "2.WURZEL=":
 WAIT :GOTO 140
170:GOTO 400
200:REM EINE REELLE WURZ
 EL
210:WAIT 100:PRINT "WURZ
 EL=":WAIT
220:PRINT -B/(2*A)
230:GOTO 400
300:REM ZWEI REELLE WURZ
 ELN
310:WAIT 100:PRINT "1.WU
 RZEL=":WAIT
320:PRINT (-B+√D)/(2*A)
330:WAIT 100:PRINT "2.WU
 RZEL=":WAIT
340:PRINT (-B-√D)/(2*A)
400:INPUT "WEITER (J/N)?
 ";Q$
410:IF LEFT$(Q$,1)="J"
 GOTO 10
420:END

```

## SPEICHERINHALT

|     |                                      |
|-----|--------------------------------------|
| A   | a (Leitkoeffizient)                  |
| B   | b (Koeffizient des linearen Gliedes) |
| C   | c (Konstanter Term)                  |
| D   | D (Diskriminante)                    |
| E   | √                                    |
| F   |                                      |
| G   |                                      |
| H   |                                      |
| I   | Imaginäranteil der komplexen Wurzel  |
| J   |                                      |
| K   |                                      |
| L   |                                      |
| M   |                                      |
| N   |                                      |
| O   |                                      |
| P   |                                      |
| Q\$ | √                                    |
| R   | Realteil der komplexen Wurzel        |
| S   |                                      |
| T   |                                      |
| U   |                                      |
| V   |                                      |
| W   |                                      |
| X   |                                      |
| Y   |                                      |
| Z   | "Markierung"                         |

## ÜBERBLICK

Die Wurzel einer Gleichung zu ermitteln, ist i. a. sehr zeitraubend. Nachstehend ist ein Verfahren zur Wurzelberechnung nach Newton aufgeführt. Bei dem Verfahren nach Newton variiert der Anfangspunkt automatisch gemäß dem festgelegten Intervall.

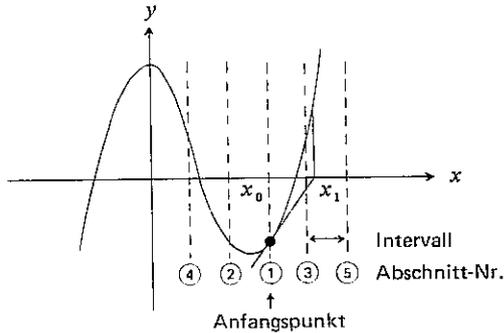
## INHALT (Formel)

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Beträgt der absolute Wert der Differenz zwischen  $x_n$  und  $x_{n+1}$  weniger als  $10^{-8}$ , wird  $x_n$  als Wurzel angezeigt. Der Differentialwert wird wie folgt definiert:

$$f'(x) = \frac{f(x+h) - f(x)}{h} \quad (h : \text{Minutenwert})$$

Um  $10^{-8}$  abzuwandeln, verändert man E-8 in Zeile 340.



## ANLEITUNG ZUM BETRIEB

1. Das Programm wird mit **DEF** A gestartet.
2. Eingabe von Anfangspunkt  
Minutenwert  
Intervall
3. Ausgabe: Wurzelwert (Die **ENTER** -Taste drücken, um die Wurzel im nächsten Intervall zu ermitteln.)

## BEISPIEL

$$x^3 - 2x^2 - x + 2 = 0 \quad (\text{Wurzeln: } -1, 1, 2)$$

Anfangspunkt = 0

Minutenwert =  $10^{-4}$

Intervall = 0.5

Die Funktion wird als Unterprogramm ab Zeile 500 geschrieben. Dazu wird im PRO-Modus eingegeben:

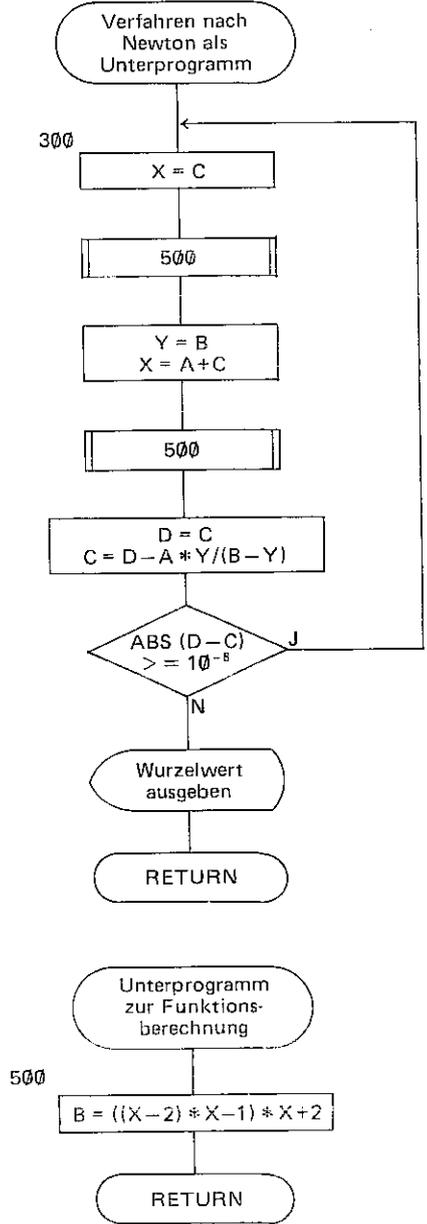
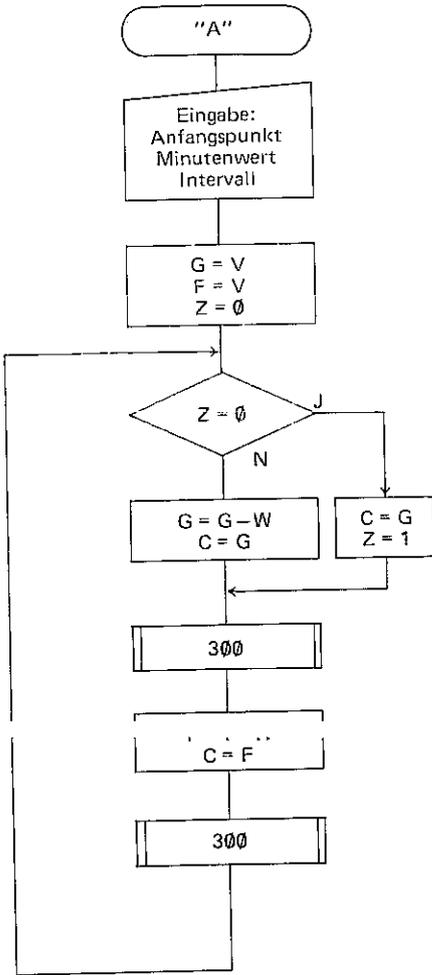
500: B = ((x - 2) \* X - 1) \* X + 2 **ENTER**

510: RETURN **ENTER**

## TASTENBETÄTIGUNG

| Schritt-Nr. | Eingabe            | Anzeige        | Bemerkungen                                                                              |
|-------------|--------------------|----------------|------------------------------------------------------------------------------------------|
| 1           | <b>DEF</b> A       | ANFANGSPUNKT = | Anfangspunkt eingeben                                                                    |
| 2           | 0 <b>ENTER</b>     | MINUTENWERT =  | Minutenwert eingeben                                                                     |
| 3           | .0001 <b>ENTER</b> | INTERVALL =    | Intervall eingeben                                                                       |
| 4           | .5 <b>ENTER</b>    |                | 2.                                                                                       |
| 5           | <b>ENTER</b>       |                | 1.<br>Zum Auffinden der nächsten Wurzel ist nochmals die <b>ENTER</b> -Taste zu drücken. |
| 6           | <b>ENTER</b>       |                | -1.                                                                                      |
| 7           | <b>ENTER</b>       |                | 1.                                                                                       |
| 8           | <b>ENTER</b>       |                | -1.                                                                                      |
| 9           | <b>ENTER</b>       |                | -1.                                                                                      |
| 10          | <b>ENTER</b>       |                | -1.                                                                                      |
| 11          | <b>ENTER</b>       |                | 2.                                                                                       |
| 12          | <b>BRK</b>         | BREAK IN 350   | Programmabbruch                                                                          |

# FLUSSDIAGRAMM



## PROGRAMMLISTING

```

10:"A":INPUT "ANFANGSPU
 NKT=";V
20:INPUT "MINUTENWERT="
 ;A
30:INPUT "INTERVALL=";W
40:G=V:F=V:Z=0
50:IF Z=0 GOTO 70
60:G=G-W:C=G:GOTO 80
70:C=G:Z=1
80:GOSUB 300
90:F=F+W:C=F
100:GOSUB 300
110:GOTO 50
120:END
300:X=C:GOSUB 500
310:Y=B:X=A+C
320:GOSUB 500
330:D=C:C=D-A*Y/(B-Y)
340:IF ABS (D-C)>1E-8
 GOTO 300
350:BEEP 3:PRINT C
360:RETURN
500:B=((X-2)*X-1)*X+2
510:RETURN

```

## SPEICHERINHALT

|   |                   |
|---|-------------------|
| A | h (Minutenwert)   |
| B | f (x)             |
| C | $x_0$             |
| D | f (x + h)         |
| E |                   |
| F | ✓                 |
| G | ✓                 |
| H |                   |
| I |                   |
| J |                   |
| K |                   |
| L |                   |
| M |                   |
| N |                   |
| O |                   |
| P |                   |
| Q |                   |
| R |                   |
| S |                   |
| T |                   |
| U |                   |
| V | Anfangswert       |
| W | Intervall         |
| X | x                 |
| Y | f (x)             |
| Z | Anfangsmarkierung |

## STICHWORTVERZEICHNIS

|                           |     |                      |     |
|---------------------------|-----|----------------------|-----|
| &                         | 63  | BEEP                 | 131 |
| *                         | 72  | Barwertanalyse       | 25  |
| +                         | 72  | Batterien            | 14  |
| -                         | 72  | Befehle              | 79  |
| /                         | 72  | Benannte Programme   | 105 |
| ^                         | 72  | CA-Taste             | 18  |
| √                         | 181 | CAL-Modus            | 9   |
| <                         | 73  | CAL-Taste            | 9   |
| ◀                         | 42  | CE-126P              | 107 |
| <=                        | 73  | CHR\$                | 182 |
| <>                        | 73  | CLEAR                | 132 |
| =                         | 73  | CLOAD                | 119 |
| >                         | 73  | CLOAD?               | 120 |
| ▶                         | 42  | CONT                 | 121 |
| >=                        | 73  | COS                  | 170 |
| ↑                         | 81  | CSAVE                | 122 |
| ↓                         | 81  | CUR                  | 177 |
| A( ) Variable             | 70  | Cassetten            | 111 |
| ABS                       | 175 | Cassettenrecorder    | 109 |
| ACS                       | 175 | Cursor               | 11  |
| AHC                       | 175 | DATA                 | 133 |
| AHS                       | 176 | DEF-Taste            | 105 |
| AHT                       | 176 | DEG                  | 177 |
| ALL RESET                 | 12  | DEGREE               | 134 |
| AND                       | 74  | DEL-Taste            | 82  |
| AREAD                     | 129 | DIM                  | 135 |
| ASC                       | 182 | DMS                  | 177 |
| ACII                      | 182 | Direkte Berechnungen | 62  |
| ASN                       | 176 | Drucker              | 107 |
| ATN                       | 176 | END                  | 137 |
| Amortisation              | 26  | ERASE                | 138 |
| Anzeige                   | 20  | ERASE FIN            | 138 |
| Aufgelaufenes Kapital     | 27  | ENTER-Taste          | 42  |
| Aufgelaufen Zinsen        | 27  | EXP                  | 177 |
| Ausdrücke                 | 72  | Einfache Variable    | 66  |
| Automatische Ausschaltung | 17  | Einschalten          | 17  |
| BASIC-Modus               | 41  | FACT                 | 177 |
| BASIC-Taste               | 41  | FOR...TO...STEP      | 139 |

## Stichwortverzeichnis

|                         |     |                                |     |
|-------------------------|-----|--------------------------------|-----|
| Fehlermeldungen         | 188 | NEW                            | 126 |
| Fehlersuche             | 185 | NEXT                           | 153 |
| Feldvariable            | 67  | NOT                            | 75  |
| Formatierte Ausgabe     | 192 | Netto-Gegenwartswert           | 28  |
| Funktionen              | 174 | Numerische Ausdrücke           | 72  |
| GOSUB                   | 141 | Numerische Funktionen          | 175 |
| GOTO                    | 142 | Numerische Variable            | 65  |
| GRAD                    | 143 | ON (Korrigieren bei manuellen) | 17  |
| Gegenwartswert          | 28  | ON... GOSUB                    | 154 |
| Gewinn                  | 29  | ON... GOTO                     | 155 |
| Grenzen der Zahlen      | 53  | OR                             | 74  |
| HCS                     | 178 | Operatoren                     | 72  |
| HSN                     | 178 | Operatoren-Priorität           | 56  |
| HTN                     | 178 | (BASIC-Modus)                  |     |
| Hypotheken              | 27  | P ↔ NP                         | 107 |
| IF... THEN              | 144 | PASS                           | 127 |
| INKEY\$                 | 174 | PAUSE                          | 156 |
| INPUT                   | 145 | PI                             | 174 |
| INPUT#                  | 147 | PRINT                          | 158 |
| INT                     | 148 | PRINT#                         | 160 |
| INS-Taste               | 44  | PRO-Modus                      | 9   |
| Instandhaltung          | 187 | POL                            | 179 |
| Interne Rückkehrrate    | 28  | Programme                      | 78  |
| Kapital                 | 107 | Priorität (CAL-Modus)          | 10  |
| Kommandos               | 119 | RADIAN                         | 162 |
| LEFT\$                  | 182 | RANDOM                         | 163 |
| LEN                     | 182 | RCP                            | 179 |
| LET                     | 150 | READ                           | 164 |
| LIST                    | 124 | REC                            | 179 |
| LLIST                   | 125 | REM                            | 165 |
| LN                      | 178 | RESET                          | 12  |
| LOG                     | 178 | RESTORE                        | 166 |
| LPRINT                  | 151 | RETURN                         | 167 |
| Letzte-Antwort-Funktion | 54  | RIGHT\$                        | 183 |
| Lineare Regression      | 36  | RND                            | 180 |
| Logische Ausdrücke      | 74  | RUN                            | 128 |
| Löschtaste              | 18  | RUN-Modus                      | 11  |
| MEM                     | 174 | ROT                            | 180 |
| MID\$                   | 183 | SGN                            | 180 |
| Manuelle Berechnung     | 41  | SHIFT-Taste                    | 21  |
| Masken                  | 192 | SIN                            | 181 |

|                                                  |     |                            |     |
|--------------------------------------------------|-----|----------------------------|-----|
| SQR                                              | 181 | String-Variable            | 65  |
| SQU                                              | 181 | TAN                        | 181 |
| STOP                                             | 168 | TEN                        | 181 |
| STR\$                                            | 183 | TRON                       | 170 |
| Schablone                                        | 106 | TROFF                      | 169 |
| Schutzdeckel                                     | 6   | USING                      | 171 |
| Speicherschutz                                   | 86  | Unterprogramme             | 141 |
| Statements                                       | 32  | VAL                        | 183 |
| Statistische Berechnungen                        | 32  | Variable                   | 64  |
| Statistische Berechnungen mit<br>einer Variablen | 35  | Verhältnis-Ausdrücke       | 73  |
| Statistische Berechnungen mit<br>zwei Variablen  | 36  | WAIT                       | 173 |
| Störungssuche                                    | 184 | Wissenschaftliche Notation | 52  |
| String-Ausdrücke                                 | 72  | Zeilennummern              | 78  |
| String-Funktionen                                | 182 | Zinssatz                   | 25  |
|                                                  |     | Zusammensetzende Periode   | 31  |









# SHARP CORPORATION

OSAKA, JAPAN