

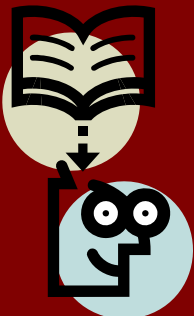
Manfred Becker

über  
120  
Seiten!

# Casio Basic Library

© 29.08.2009

Alle Listings zum  
Scientific Library 116  
des Taschenrechners  
**CASIO FX-850P/880P**



eBook

## Copyright

Alle in diesem Tutorial enthaltenen Programme und Verfahren wurden nach bestem Wissen erstellt und mit großer Sorgfalt getestet. Dennoch sind Fehler nicht ganz auszuschließen. Aus diesem Grund ist das im vorliegenden Tutorial enthaltene Programm-Material mit keiner Verpflichtung oder Garantie irgendeiner Art verbunden. Der Autor übernimmt infolgedessen keine Verantwortung und wird keine daraus folgende oder sonstige Haftung übernehmen, die auf irgendeine Art aus der Benutzung dieses Programm-Materials oder Teilen davon entsteht.

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt auch ohne besondere Kennzeichnung nicht zur Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürften.

Dieses Werk ist urheberrechtlich geschützt. Alle Rechte, auch die der Übersetzung, des Nachdrucks und der Vervielfältigung des Tutorials oder Teilen daraus, vorbehalten.

Autor:

Dipl.-Ing.(FH) Manfred Becker

*Manfred Becker*

[mani.becker@web.de](mailto:mani.becker@web.de)

<http://manib.ma.funpic.de/>

## Vorwort



Ich hatte schon immer viel Freude dabei anderen zu helfen. Gerade bei Programmierfragen helfen mein fundiertes Wissen und meine große Erfahrung dabei, andere Hilfestellung zu geben. Bei einigen Foren habe ich das bereits erfolgreich praktiziert.

Allerdings habe ich festgestellt, dass meine Hinweise nur sehr wenigen Nutzern zugute kamen. Oft wurde ein und dieselbe Frage kurze Zeit später von einem Anderen erneut gestellt.

Besser ist es, so finde ich, ein ordentliches Tutorial anbieten zu können, in dem die Standardfragen abgehandelt werden. Nun genügt als Forenantwort der einfache Link in das entsprechende Kapitel.

Aus diesem Grund ist dieses Tutorial entstanden. Als Thema habe ich die **Casio Scientific Library 116** herausgegriffen. Die Basic-Listings zu dieser beeindruckenden Bibliothek sind nicht nur für Programmieranfänger von Interesse.

Ich hoffe, es hilft recht vielen Programmierern, und Alle die es werden wollen, bei ihrer täglichen Arbeit.

## In eigener Sache

Wenn Sie Vorschläge oder Fragen zum Tutorial haben, erreichen Sie mich per E-Mail:

[mani.becker@web.de](mailto:mani.becker@web.de)

oder im Forum auf meiner Homepage:

<http://manib.ma.funpic.de/>

Hier finden Sie auch die aktuelle Version und die zugehörigen BASIC-Quelltexte.

Aber auch wenn Sie keinen solchen Taschenrechner besitzen und dennoch wissen wollen wie man programmiert, finden Sie auf meiner Seite viele weitere sehr interessante Artikel. Schauen Sie doch einfach einmal vorbei. Ich freue mich über Ihren Besuch...

Manfred Becker  
Aug. 2009



# Inhaltsverzeichnis

<b>1</b>	<b>Einführung .....</b>	<b>9</b>
1.1	Casio FX-850P / FX-880P .....	9
1.2	Spezifikation Casio FX-850P / FX-880P .....	10
1.3	Preise und Zubehör für Casio Pocket Computer .....	11
<b>2</b>	<b>Voraussetzungen .....</b>	<b>12</b>
2.1	Der Taschenrechner Casio FX-850P/FX-880P .....	12
2.1.1	CAL-Modus .....	13
2.1.2	Formelspeicher-Funktion .....	14
2.1.3	Datenbank-Funktion .....	15
2.1.4	Eingebaute Wissenschaftliche Bibliothek .....	16
2.1.5	BASIC-Modus .....	17
2.2	Die CASIO-Bedienungsanleitung .....	18
2.3	Interesse und Zeit .....	19
<b>3</b>	<b>Basic-Listings der Scientific Library 116.....</b>	<b>20</b>
3.1	TRANSLIB .....	20
3.2	Menü- und Startprogramm .....	22
3.2.1	K0630 Menü- und Startprogramm für die LIB-Programme .....	22
3.3	Mathematik .....	25
3.3.1	1000 Speicher-Berechnung .....	25
3.3.2	5010 Primfaktoren-Analyse .....	26
3.3.3	5020 Grösster gemeinsamer Teiler, Kleinstes gemeinsames Vielfaches .....	26
3.3.4	5040 Gleichungssystem (Gauss-Elimination) .....	27
3.3.5	5050 Quadratische Gleichung .....	28
3.3.6	5060 Kubische Gleichung .....	29
3.3.7	5080 Numerische Lösung einer Gleichung (Newton-Methode) .....	30
3.3.8	5090 Numerische Lösung einer Gleichung (Halbierungs-Methode) .....	31
3.3.9	5100 Matrixoperationen .....	32
3.3.10	5200 Numerische Integration (Romberg-Methode) .....	34
3.3.11	5220 Gewöhnliche Differenzialgleichung (Runge-Kutta-Methode) .....	35
3.3.12	5230 Lagrangsche Interpolation .....	36
3.3.13	5250 Gamma-Funktion $\Gamma(x)$ .....	36
3.3.14	5260 Besselsche Funktion $J_n(x)$ .....	37
3.3.15	5270 Besselsche Funktion $Y_n(x)$ .....	38
3.3.16	5280 Modifizierte Besselsche Funktion $I_n(x)$ .....	39
3.3.17	5290 Modifizierte Besselsche Funktion $K_n(x)$ .....	40
3.3.18	5300 Komplexe Zahlen .....	41
3.3.19	5350 Binär-Dezimal-Hexadezimal .....	42
3.3.20	5510 Gerade durch zwei Punkte .....	44
3.3.21	5520 Schnittwinkel von zwei Geraden .....	44
3.3.22	5530 Abstand zwischen Punkt und Gerade .....	44
3.3.23	5540 Drehbewegung .....	45
3.3.24	5550 Kreis durch drei Punkte .....	45
3.3.25	5560 Länge von Tangenten von einem Punkt zu einem Kreis .....	46
3.3.26	5570 Tangentialgleichung .....	47
3.3.27	5600 Dreiecksfläche .....	48
3.3.28	5605 Fläche eines Trapezoides .....	48
3.3.29	5610 Fläche eines Parallelogramms .....	49
3.3.30	5615 Fläche eines Kreises .....	49
3.3.31	5620 Fläche eines Sektors .....	49
3.3.32	5625 Fläche eines Segments .....	50
3.3.33	5630 Fläche einer Ellipse .....	50
3.3.34	5635 Fläche eines Polygons .....	50
3.3.35	5650 Flächeninhalt einer Kugel .....	51
3.3.36	5655 Flächeninhalt einer Kugelzone .....	51
3.3.37	5660 Flächeninhalt eines Kugelsektors .....	51

3.3.38	5665 Oberfläche eines Kreiszylinders.....	52
3.3.39	5670 Oberfläche eines Kreiskegels.....	52
3.3.40	5675 Flächeninhalt eines Kreiskegelstumpfes.....	52
3.3.41	5700 Rauminhalt einer Kugel.....	53
3.3.42	5705 Rauminhalt einer Kugelzone.....	53
3.3.43	5710 Rauminhalt eines Kugelsektors.....	53
3.3.44	5715 Rauminhalt eines Kreiszylinders.....	54
3.3.45	5720 Rauminhalt eines Kreiskegels.....	54
3.3.46	5725 Rauminhalt eines Kreiskegelstumpfes.....	54
3.3.47	5730 Rauminhalt eines Keils.....	55
3.3.48	5735 Rauminhalt einer Pyramide.....	55
3.3.49	5740 Rauminhalt eines Pyramidenstumpfes.....	55
3.3.50	5745 Rauminhalt eines Ellipsoiden.....	56
3.3.51	5750 Einbeschriebener Kreis und umbeschriebener Kreis eines Polygons.....	56
3.3.52	5760 Regelmäßiger Polyeder.....	57
3.4	<i>Wissenschaftliche Formeln</i> .....	58
3.4.1	5800 Zerlegung in Faktoren.....	58
3.4.2	5810 Trigonometrische Funktionen.....	59
3.4.3	5820 Differenziale.....	60
3.4.4	5830 Integrationen.....	61
3.4.5	5840 Laplace-Transformation.....	62
3.4.6	5900 Tabelle des Periodischen Systems.....	63
3.4.7	5910 Wissenschaftliche Konstanten.....	65
3.4.8	5920 Konstante der elektrolytischen Dissoziation.....	66
3.4.9	5930 Bewegung und Energie.....	67
3.4.10	5932 Wellenbewegung.....	68
3.4.11	5934 Wechselstrom und Gleichstromkreise.....	69
3.4.12	5936 Elektrische und magnetische Felder.....	70
3.4.13	5938 Thermodynamik und Anderes.....	71
3.4.14	5950 Metrische Umwandlungen für Länge.....	72
3.4.15	5960 Metrische Umwandlungen für Flächen.....	73
3.4.16	5970 Metrische Umwandlungen für Rauminhalt.....	74
3.4.17	5980 Metrische Umwandlungen für Gewicht.....	75
3.5	<i>Statistik</i> .....	76
3.5.1	6210 Obere Wahrscheinlichkeitsintegrale (Normalverteilung).....	76
3.5.2	6220 Obere Wahrscheinlichkeitsintegrale ( $x^2$ Verteilung).....	76
3.5.3	6230 Obere Wahrscheinlichkeitsintegrale (t Verteilung).....	76
3.5.4	6240 Obere Wahrscheinlichkeitsintegrale (F Verteilung).....	77
3.5.5	6310 Obere Summenhäufigkeit (Binomial-Verteilung).....	77
3.5.6	6320 Obere Summenhäufigkeit (Poisson-Verteilung).....	77
3.5.7	6330 Obere Summenhäufigkeit (Hypergeometrische Verteilung).....	78
3.5.8	6410 Prozentpunkt (Normalverteilung).....	78
3.5.9	6420 Prozentpunkt ( $x^2$ Verteilung).....	78
3.5.10	6430 Prozentpunkt (t Verteilung).....	79
3.5.11	6440 Prozentpunkt (F Verteilung).....	79
3.5.12	6450 Normale Zufallszahlen.....	79
3.5.13	6460 Exponentielle Zufallszahlen.....	80
3.5.14	6500 Statistische Berechnungen mit einer Variablen.....	80
3.5.15	6510 Lineare Regression ( $y = a + bx$ ).....	81
3.5.16	6520 Logarithmische Regression ( $y = a + b \ln x$ ).....	81
3.5.17	6530 Exponentielle Regression ( $y = ab^x$ ).....	81
3.5.18	6540 Potenz-Regression ( $y = ax^b$ ).....	81
3.5.19	6610 Mittelwert-Intervallschätzung (für bekannte Varianz).....	82
3.5.20	6620 Mittelwert-Intervallschätzung (für unbekannte Varianz).....	82
3.5.21	6630 Varianz-Intervallschätzung.....	83
3.5.22	6640 Standardabweichung-Intervallschätzung.....	83
3.5.23	6650 Varianzquotient-Intervallschätzung.....	84
3.5.24	6660 Mittelwertdifferenz-Intervallschätzung.....	85
3.5.25	6670 Verhältnis-Intervallschätzung.....	85
3.5.26	6680 Verhältnisdifferenz-Intervallschätzung.....	86
3.5.27	6710 Grundgesamtheits-Mittelwert-Test (zweiseitig): für bekannte Varianz.....	86
3.5.28	6711 Grundgesamtheits-Mittelwert-Test (rechtsseitig): für bekannte Varianz.....	86
3.5.29	6712 Grundgesamtheits-Mittelwert-Test (linksseitig): für bekannte Varianz.....	86
3.5.30	6720 Grundgesamtheits-Mittelwert-Test (zweiseitig): für unbekannte Varianz.....	87
3.5.31	6721 Grundgesamtheits-Mittelwert-Test (rechtsseitig): für unbekannte Varianz.....	87
3.5.32	6722 Grundgesamtheits-Mittelwert-Test (linksseitig): für unbekannte Varianz.....	87
3.5.33	6730 Grundgesamtheits-Varianz-Test (zweiseitig).....	87

3.5.34	6731 Grundgesamtheits-Varianz-Test (rechtsseitig) .....	87
3.5.35	6732 Grundgesamtheits-Varianz-Test (linksseitig) .....	87
3.5.36	6740 Varianzquotient-Test (zweiseitig) .....	88
3.5.37	6741 Varianzquotient-Test (rechtsseitig) .....	88
3.5.38	6742 Varianzquotient-Test (linksseitig) .....	88
3.5.39	6750 Mittelwertdifferenz-Test (zweiseitig) .....	88
3.5.40	6751 Mittelwertdifferenz-Test (rechtsseitig) .....	88
3.5.41	6752 Mittelwertdifferenz-Test (linksseitig) .....	88
3.5.42	6760 Quotiententest (zweiseitig) .....	88
3.5.43	6761 Quotiententest (rechtsseitig) .....	89
3.5.44	6762 Quotiententest (linksseitig) .....	89
3.5.45	6770 Quotientendifferenztest (zweiseitig) .....	89
3.5.46	6771 Quotientendifferenztest (rechtsseitig) .....	89
3.5.47	6772 Quotientendifferenztest (linksseitig) .....	89
<b>3.6</b>	<b>Statistik Unterprogramme .....</b>	<b>90</b>
3.6.1	M6510 Unterprogramm der Statistik-Programme 6510, 6520, 6530, 6540 .....	90
3.6.2	M6710 Unterprogramm der Statistik-Programme 6710, 6711, 6712 .....	92
3.6.3	M6720 Unterprogramm der Statistik-Programme 6720, 6721, 6722 .....	93
3.6.4	M6730 Unterprogramm der Statistik-Programme 6730, 6731, 6732 .....	94
3.6.5	M6740 Unterprogramm der Statistik-Programme 6740, 6741, 6742 .....	95
3.6.6	M6750 Unterprogramm der Statistik-Programme 6750, 6751, 6752 .....	96
3.6.7	M6760 Unterprogramm der Statistik-Programme 6760, 6761, 6762 .....	97
3.6.8	M6770 Unterprogramm der Statistik-Programme 6770, 6771, 6772 .....	98
3.6.9	S6020 Unterprogramm der Statistik-Programme 6610, 6620, 6630, 6640, 6650, 6660, M6710, M6720, M6730, M6740, M6750 .....	99
3.6.10	S6030 Unterprogramm der Statistik-Programme 6650, 6660, M6740, M6750 .....	100
3.6.11	S6210 Unterprogramm der Statistik-Programme 6210, S6220 .....	101
3.6.12	S6220 Unterprogramm der Statistik-Programme 6220, S6420 .....	101
3.6.13	S6230 Unterprogramm der Statistik-Programme 6230, S6430 .....	101
3.6.14	S6240 Unterprogramm der Statistik-Programme 6240, S6440 .....	102
3.6.15	S6310 Unterprogramm der Statistik-Programme 6310 .....	102
3.6.16	S6320 Unterprogramm der Statistik-Programme 6320 .....	102
3.6.17	S6330 Unterprogramm der Statistik-Programme 6330 .....	103
3.6.18	S6410 Unterprogramm der Statistik-Programme 6410, 6610, 6670, 6680, M6710, M6760, M6770, S6420, S6430, 103 .....	103
3.6.19	S6420 Unterprogramm der Statistik-Programme 6420, 6630, 6640, M6730, S6440 .....	104
3.6.20	S6430 Unterprogramm der Statistik-Programme 6430, 6620, 6660, M6720, M6750, S6440 .....	104
3.6.21	S6440 Unterprogramm der Statistik-Programme 6440, 6650, M6740 .....	105
<b>3.7</b>	<b>Undokumentierte Funktionen .....</b>	<b>106</b>
3.7.1	0400 Speicher- und Schnittstellen-Testprogramm .....	106
3.7.2	0800 Download-Programm .....	107
3.7.3	0900 Konvertier-Programm .....	108
<b>4</b>	<b>Anhang .....</b>	<b>110</b>
4.1	Befehls-Übersicht .....	110
4.2	Übersicht der Wissenschaftlichen Bibliothek .....	114
4.3	Index .....	118





# 1 Einführung

## 1.1 Casio FX-850P / FX-880P

Meinen Casio FX-850P habe ich inzwischen schon sehr lange (seit 1991). Ich habe ihn damals für mein Studium gekauft. Allerdings musste ich feststellen, dass er dafür nur zum Teil geeignet war. Er musste deshalb in der Mathe-Vorlesung einem HP 48SX weichen. Der HP war für Integral-Rechnungen und für Gleichungslösungen einfach unschlagbar.



Abbildung 1

Dennoch blieb ich dem Casio treu, nicht zuletzt deswegen, weil er sehr unkompliziert ist. Die Basic-Programmierung ist einfach genial, und ich konnte ihn in manch anderen Vorlesungen verwenden.

Ein weiteres Phänomen ist die große Beliebtheit dieses Rechners, und das obwohl er längst ein Museumsstück sein könnte. Er wird zwar schon länger nicht mehr hergestellt, aber nichts desto trotz steht er (und sein Bruder FX-880P) auch heute noch hoch im Kurs. Bei eBay Auktionen werden Preise bis zu 200.- Euro geboten!

Aufgrund des Alters findet man aber heute leider kaum noch Zubehör oder Dokumentation zu dem Rechner. Bücher sind ebenfalls schwer zu bekommen. Auch im Internet werden viele der Webseiten, die Themen zum Casio anbieten, kaum noch gepflegt. Klar, es gibt Ausnahmen, doch wenn man auf der Suche nach Informationen ist, hat man es heutzutage schwer.

Das hat mich dazu bewogen meine alten Daten zu diesem Rechner auszugraben, und auf meiner Homepage online zu stellen. Ich hoffe, die Artikel machen ihnen auch heute noch viel Freude.

Ein Thema hatte aber bis jetzt gefehlt: **Alle Basic-Listings der Scientific Library 116!**

Das will ich nun mit diesem Tutorial nachholen.

In diesem Sinne wünsche ich Ihnen beim Lesen dieser Lektüre genauso viel Spaß, wie ich beim Schreiben hatte...

## 1.2 Spezifikation Casio FX-850P / FX-880P

Hier zunächst einmal die Technischen Daten des Casio FX-850P / FX-880P.

- **Anzeige**  
Punktmatrix-Flüssigkristallanzeige mit 32 Spalten x 2 Zeilen (5x7 Punkte)  
Symbol-Anzeigen und eine 5-stellige 7-Segment Anzeige  
LCD Ansteuerungs-IC: 2 x HD66100F  
Regelbarer Kontrast
- **Prozessor**  
CPU: VLSI mit 1,228MHz. Hitachi HD62002A01  
(FX-860P, FX-880P: HD62002A03)
- **Speicher**  
8 KB RAM (FX-860P: 24 KB, FX-880P: 32 KB)  
Interner Einbauschacht für Speichererweiterungskarte RP-8 bzw. RP-33  
Maximale Speicheraufrüstung insgesamt 40 KB (FX-880P: 64 KB)
- **Spannungsversorgung**  
2x 3V CR2032 Lithium Batterie für Haupt-Spannungsversorgung  
1x 3V CR1220 Lithium Batterie für Speicherschutz-Spannungsversorgung  
Leistungsaufnahme 0.04 W  
Ausschaltautomatik nach ca. 6 Minuten
- **Batterie-Lebensdauer**  
ca. 90 Stunden bei Dauerbetrieb  
ca. 150 Stunden bei Daueranzeige von 5555555555  
ca. 4.5 Monate bei täglich einstündiger Benutzung
- **Programmiersprache**  
BASIC, 10 Programmbereiche (P0 bis P9)
- **Eingebauter Lautsprecher**
- **Anschlussstecker für Peripheriegeräte**  
Unterstützt RS232 und Centronics  
(Spezielles Übertragungskabel oder Interfaceeinheit FA-6 notwendig )
- **Abmessungen**  
11.6 x 193 x 78 (HxBxT) mm
- **Gewicht**  
197g mit Batterien
- **Zubehör**  
Etui, Referenzkarte und Bedienungsanleitung

Der Taschenrechner hat einen BASIC Interpreter, eine MEMO Funktion, eine Bibliothek mit 116 wissenschaftlichen Funktionen, und eine Schnittstelle, die es erlaubt zu anderen Geräten Verbindung aufzunehmen. Der Speicher kann insgesamt auf 40 KB aufgerüstet werden, wenn man die Speichererweiterungskarte RP-33 einsetzt (8 KB + 32 KB).

Später wurde der Casio FX-880P gebaut, der dem FX-850P gleicht, außer dass bei ihm standardmäßig 32 KB interner Speicher eingebaut waren. Somit kann der FX-880P auf insgesamt 64 KB aufgerüstet werden.

Die Bedienungsanleitung können Sie im Internet herunterladen:

- <http://www.silrun.info>
- <http://www.usersmanualguide.com/casio/calculators/fx-850p>

### 1.3 Preise und Zubehör für Casio Pocket Computer

Folgende CASIO Pocket Computer nebst Zubehör kann (bzw. konnte) man erwerben.

Pocket Computer:

• FX-730P	8 KB RAM	159,00 DM
• FX-795P	16 KB RAM	229,00 DM
• FX-850P	8 KB RAM	249,00 DM
• FX-880P	32 KB-RAM	299,00 DM

Zubehör:

• FA-6	RS-232 + Kassetteninterface	299,00 DM
• SB-7	Kassettenrecorder Kabel	39,90 DM
• PK-7	Kabel FP-100, SB-43, FA-6	79,90 DM
• RP-8	8 KB-RAM Erweiterung	49,90 DM
• RP-33	32 KB-RAM Erweiterung	99,90 DM
• FP-40	Plotter 40/80 Zeichen	249,00 DM
• SB-43	RS-232 Centronicsbox an FP-40	89,90 DM
• SB-51	Centronicskabel FA-80 -> FP-40	149,00 DM

Wenn Sie heutzutage ein Übertragungskabel suchen, müssen Sie schon viel Glück haben, eines zu ergattern. Selbst bei eBay werden nur sehr selten solche Kabel angeboten. Und diese Kabel benötigen dann fast immer eine Serielle Schnittstelle. Doch diese ist bei neuen Rechnern und Notebooks durch eine USB-Schnittstelle ersetzt.

Aus diesem Grund habe ich ein USB-Interface entwickelt, welches Sie über meine Homepage beziehen können!



Abbildung 2

• USB-Interface	39,90 €
-----------------	---------

<http://manib.ma.funpic.de/>

## 2 Voraussetzungen

Um mit diesem Tutorial erfolgreich arbeiten zu können, sollten Sie folgende Dinge haben:

- Den Taschenrechner Casio FX-850P/FX-880P
- Die dazugehörige CASIO-Bedienungsanleitung
- Dieses Tutorial, Interesse und Zeit

### 2.1 Der Taschenrechner Casio FX-850P/FX-880P

Ja, das ist er. Ich gebe zu, ich habe schon immer ein Fabel für alles Technische gehabt. Insbesondere Taschenrechner haben es mir angetan. Aber dieser Casio ist etwas ganz besonderes.



Abbildung 3

Er ist viel mehr wie ein normaler Taschenrechner. Er hat fünf Betriebsarten, ein sehr gutes, ausführliches Handbuch, und ein schönes, schlankes Design. Das Alu-Gehäuse, das große Display und die QWERTY-Tastatur verleihen dem Rechner etwas Erhabenes. Dann die abgesetzte Nummerntastatur mit den typischen Taschenrechner-Funktionen an der Seite rundet den positiven Eindruck ab. Die Tastatur (Plastiktasten) ist sehr gut zu bedienen. Sie hat zwar keinen so guten Druckpunkt, wie etwa die von einem HP 48SX Rechner, aber ist immer noch besser wie die Gummi-Tastatur von einem SHARP PC-E500(S).

Ehrlich gesagt, gibt es nur drei Dinge, die man an ihm bemängeln könnte:

- Er besitzt leider kein grafisches Display, sondern „nur“ eine 2x32 Zeichen Anzeige.
- In seiner Grundausstattung hat der FX-850P lediglich 8 kByte RAM (der FX-880P hat übrigens standardmäßig 32kByte)
- Er wird nicht mehr hergestellt

Aber das mit der Grafik ist nicht ganz so schlimm, und den Speicher kann man nachträglich aufrüsten. Schade nur, dass er nicht mehr hergestellt wird. Lediglich bei Flohmärkten oder bei eBay kann man das gute Stück manchmal noch ergattern.

Aber schauen wir uns die fünf Betriebsarten (Betriebsmodi oder kurz Modi) an.

Die Betriebsarten des Casio FX-850P/FX-880P:

- CAL-Modus
- Formelspeicher-Funktion
- Datenbank-Funktion
- Eingebaute Wissenschaftliche Bibliothek
- BASIC-Modus

### 2.1.1 CAL-Modus

Zunächst einmal ist da der CAL-Modus, mit dem man ihn wie ein ganz normaler wissenschaftlicher Taschenrechner bedienen kann. Von großem Vorteil ist natürlich das für einen Taschenrechner ungewöhnliche Display, mit dem 2 Zeilen zu je 32 Zeichen dargestellt werden können. Über diesen zwei Zeilen befinden sich die Symbolanzeigen, welche die verschiedenen Betriebsmodi anzeigen.

Schön ist auch, dass über dem Ziffernblock die Tasten für die wissenschaftlichen Funktionen angeordnet sind. Das ist Standard bei vielen Wissenschaftlichen Taschenrechnern, und man muss sich nicht umgewöhnen.

Nach dem Einschalten befindet sich der Rechner automatisch im CAL-Modus, in dem arithmetische Berechnungen, Berechnungen mit Funktionen, Ausführung der wissenschaftlichen Bibliothek, Formelspeicher-Berechnungen, Programmausführung und Datenabruf durchgeführt werden können.



Die Bedienung ist wie bei normalen Taschenrechnern. Allerdings wird das Ergebnis in der zweiten Zeile angezeigt. Dadurch hat man Berechnung und Ergebnis immer gleichzeitig auf dem Display, was sicherlich von Vorteil ist.

Anstelle der Taste [=] muss die Taste [EXE] verwendet werden.



Ein weiterer Vorteil dieses Taschenrechners ist die Verwendung von Variablen im CAL-Modus. Was Variable sind werde ich später noch im Detail erklären. Hier nur soviel: Einer Variablen kann man Werte zuweisen, die der Rechner sich merkt, und erst beim Ausschalten wieder vergisst. Die Variable wird über ihren Variablennamen, angesprochen. Dieser Variablenname kann frei gewählt werden. Er muss allerdings mit einem Buchstaben beginnen, und darf keine Sonderzeichen enthalten (später mehr).

Z.B. wird hier der Variablen X der Wert 4 zugewiesen.



Danach kann eine Berechnung mit dieser Variablen durchgeführt werden.



**Hinweis**

Die Bedienung und Berechnungsfunktionen sind in der CASIO-Bedienungsanleitung, Kapitel 3 ab Seite 15 ausführlich beschrieben.

## 2.1.2 Formelspeicher-Funktion

Ein echtes Highlight ist wohl die Formelspeicher-Funktion. Damit lassen sich Berechnungen, die sich mehrfach wiederholen, ganz einfach durchführen. Ideal also, wenn es darum geht Wertetabellen zu einer Formel zu erstellen.

Folgende drei Tasten spielen dabei eine Rolle:

- [IN] Speichert die angezeigte Formel
- [OUT] Zeigt die gespeicherte Formel an
- [CALC] Führt die Berechnung der Formel aus

Zunächst einmal muss die gewünschte Formel eingegeben werden.

Nach der Eingabe muss die Taste [IN] gedrückt werden.

```
CAPS S CAL BASIC DEGRADGRA MEMO IN EDIT LIB 8 8 8 8 DEFM PRT TR STOP
Y=0.5*X^3-2*X^2+3*X-4
-
```

Wir nehmen einmal an, dass die Wertetabelle der Funktion von 0 bis 9 zu berechnen wäre. Nun kann mit der Taste [CALC] die Berechnung der Formel gestartet werden.

Der Rechner fordert zur Eingabe der Variablen X auf.

```
CAPS S CAL BASIC DEGRADGRA MEMO IN EDIT LIB 8 8 8 8 DEFM PRT TR STOP
X?_
```

Nach der Eingabe der 0 und drücken der Taste [EXE], wird das Ergebnis angezeigt.

```
CAPS S CAL BASIC DEGRADGRA MEMO IN EDIT LIB 8 8 8 8 DEFM PRT TR STOP
X?0
Y=-4
```

Nach erneuten drücken der [EXE] Taste wird die nächste Eingabe erwartet.

```
CAPS S CAL BASIC DEGRADGRA MEMO IN EDIT LIB 8 8 8 8 DEFM PRT TR STOP
X?1
Y=-2.5
```

Dieser Vorgang kann beliebig oft wiederholt werden.

```
CAPS S CAL BASIC DEGRADGRA MEMO IN EDIT LIB 8 8 8 8 DEFM PRT TR STOP
X?2
Y=-2
```

Die Taste [BRK] unterbricht die Berechnung. Später kann man mit der Taste [CALC] wieder weitermachen.

```
CAPS S CAL BASIC DEGRADGRA MEMO IN EDIT LIB 8 8 8 8 DEFM PRT TR STOP
X?3
Y= 0.5
```

Wertetabelle zu:  $f(x) = 0.5x^3 - 2x^2 + 3x - 4$

x	0	1	2	3	4	5	6	7	8	9
f(x)	-4	-2.5	-2	0.5	8	23.5	50	90.5	148	225.5



**Hinweis**

Die Formelspeicher-Funktion ist in der CASIO-Bedienungsanleitung, Kapitel 4 ab Seite 33 beschrieben.

### 2.1.3 Datenbank-Funktion

Die Datenbank-Funktion ist ein weiteres Feature, das dieser Taschenrechner bietet. Damit kann man alles Mögliche eingeben und verwalten. Es kann als Notizbuch, Adressspeicher, Telefonliste, Formelspeicher, (Spickzettel;) u.v.m eingesetzt werden.

Die eingegebenen Daten können später erweitert, editiert oder wieder gelöscht werden. Sehr interessant ist die Tatsache, dass man auf diese Daten auch von einem Basic-Programm zugreifen kann. Außerdem lassen sich abgelegt Formeln ganz einfach mit der Formelspeicher-Funktion verwenden.

Die Dateneingabe erfolgt im MEMO-IN-Modus. Dieser wird durch die Betätigung der Tasten [MODE] [9] aktiviert.

Im MEMO-IN-Modus lassen sich Daten ganz einfach eingeben.



Die gespeicherten Daten können nun angezeigt werden. Dazu ist die Taste [MEMO] zu drücken. Hierbei kann man mit den Cursor-Tasten zwischen den einzelnen Datensätzen wechseln.

Sollen weitere Daten aufgenommen oder vorhandene Daten editiert werden, muss der MEMO-IN-Modus aktiviert werden. Auch das Löschen von Datensätzen kann hierbei durchgeführt werden. Verlassen wird der MEMO-IN-Modus indem man einen anderen Modus aktiviert. Z.B. den CAL-Modus mit [MODE] [0].

Mann kann auch nach einem Begriff suchen. Dazu tippt man den gesuchten Begriff im CAL-Modus ein, und drückt dann die [MEMO] Taste. Sollte der Begriff in den Daten vorkommen, wird der entsprechende Datensatz sofort angezeigt. Mit jedem folgenden Drücken von [MEMO] wird das nächste Datenelement angezeigt, das mit dem Suchbegriff übereinstimmt.

Die Datenbank-Daten können aber auch in einem Basic-Programm verwendet werden. Dazu stehen folgende Basic-Befehle zur Verfügung:

READ#	Liest Daten aus der Datenbank
RESTORE#	Positioniert den Lesezeiger
WRITE#	Schreibt Daten in die Datenbank



#### Hinweis

Die Datenbank-Funktion ist in der CASIO-Bedienungsanleitung, Kapitel 5 ab Seite 37 ausführlich beschrieben.

## 2.1.4 Eingebaute Wissenschaftliche Bibliothek

Bei diesem Taschenrechner kann auf eine Bibliothek von insgesamt 116 Programmen zurückgegriffen werden. Diese Programme behandeln mathematische, statische sowie physikalische und wissenschaftliche Themen. Am Ende dieses Tutorials habe ich die Übersicht aller 116 Programme angefügt.

Die Programme sind im Handbuch vorbildlich beschrieben. Jedes der Programme hat eine vierstellige Bibliotheksnummer. Der Aufruf erfolgt dementsprechend durch die Eingabe dieser Nummer gefolgt von der [LIB] Taste.

z.B. Gerade durch zwei Punkte:  
5510 [LIB]

```
CAPS S CAL BASIC DEGRADGRA MEMO IN EDIT LIB 8 8 8 8 DEFM PRT TR STOP
y=ax+b ←- (x1,y1),(x2,y2)
x1= 0 ?_
```

Aber noch Interessanter wird das Ganze, wenn man im Handbuch liest, dass alle Bibliotheksprogramme in BASIC programmiert sind! Und es kommt noch besser, denn die Programmlistings kann man durch einen Trick in den Basic-Programmspeicher laden. Dort kann man sich dann das Programmlisting anschauen und somit direkt von den Casio-Profis lernen.

Der Trick, wie sollte es auch anders sein, wird über ein Basic Programm realisiert. Dieses Basic-Programm (TRANSLIB) finden Sie im Kapitel 3.1. Ich habe mir für Sie die Mühe gemacht, alle 116 Bibliotheksprogramme auszulesen. Das Ergebnis finden Sie hier in diesem Tutorial ab Kapitel 3.

Hier sehen Sie z.B. das Listing zum oben gezeigten Bibliotheksprogramm 5510:

### Listing 1 5510 Gerade durch zwei Punkte

```
5 ONERRORGOTO500
10 MODE8:DIM:ERASEa:DIMa(5):a$="x1y1x2y2ab":c$=CHR$(5)
20 CLS:PRINT"y=ax+b ←- (x1,y1),(x2,y2)";
30 FORi=0 TO3
40 LOCATE0,1:PRINTc$:MID$(a$,1+2*i,2)="":a(i);"?":INPUT@42:a(i):LOCATE0,0
50 NEXT:LOCATE0,1:PRINTc$:
70 IFA(0)=a(2) THENIFA(1)=a(3) THEN100 ELSEPRINTc$:"x
=":a(0);:w$=INPUT$(1,@):GOTO30
72 IFA(1)=a(3) THENPRINTc$:"y =" :a(1);:w$=INPUT$(1,@):GOTO30
75 a(4)=(a(3)-a(1))/(a(2)-a(0)):a(5)=a(1)-a(4)*a(0)
80 FORi=4 TO5:PRINTc$:MID$(a$,i+5,1):" =":a(i);
82 w=ASC(INPUT$(1,@)):IFw=13 THEN90
84 IFw=30 ANDi>4 THENi=i-2:GOTO90
85 IFw=31 ANDi<5 THEN90
88 GOTO82
90 NEXT:GOTO30
100 LOCATE0,1:PRINTc$:"not found":w$=INPUT$(1,@):GOTO20
500 IFERR=1 THENCLS:ONERRORGOTO0
510 IFERR=40 THENRESUME40
520 RESUME100
```



#### Hinweis

Die Eingebaute Wissenschaftliche Bibliothek ist in der CASIO-Bedienungsanleitung, Kapitel 11 ab Seite 180 ausführlich beschrieben.



## 2.1.5 BASIC-Modus

Der Basic-Modus ist das Beste am Casio Rechner. Im Gegensatz zu „normalen“ Taschenrechnern, bei denen die Funktionen fest vorgegeben und unveränderbar sind, kann man hier über die Basic-Programmierung die Funktionen beliebig erweitern und anpassen. Die Vorstellung einem Rechner sagen zu können, was er tun soll, hat mich fasziniert seit ich das erste Mal<sup>1</sup> einen Sinclair ZX-81 in den Fingern hatte. Und diese Faszination hat mich bis heute nicht wieder losgelassen.

Beim Casio dient der BASIC-Modus für die Erstellung und Ausführung der Programme. Dieser Modus wird durch das Drücken der Tasten [MODE] [1] aktiviert.

```

CAPS S CAL BASIC DEGRADGRA MEMO IN EDIT LIB 8 8 8 8 DEFM PRT TR STOP
P 0 1 2 3 4 5 6 7 8 9      3536B
Ready P0

```

Es gibt 10 Programmbereiche (P0 bis P9) in denen jeweils ein eigenes Basic-Programm abgelegt werden kann. Dieser Programmbereich kann im Basic-Modus über die Tastenfolge [SHIFT] gefolgt von [0] bis [9] gewechselt werden. Programme bleiben nach der Eingabe erhalten. Man muss also nicht Sichern, wie man das z.B. von einem Editor-Programm auf einem PC gewöhnt ist.

Ein Basic-Programm besteht meist aus mehreren Programmzeilen, der Taschenrechner hat aber nur 2 Zeilen zur Anzeige zur Verfügung. Aus diesem Grund gibt es einen speziellen Edit-Modus, bei dem über die Cursor-Tasten durch das Programmlisting gescrollt werden kann.

Außerdem lassen sich im Basic-Modus manuelle Befehle direkt eingeben (wie z.B. LIST, EDIT, PRINT oder RUN). Eine Basic-Programmzeile dagegen beginnt immer mit einer Zeilennummer (1 bis 65535), gefolgt von einer oder mehreren Basic-Anweisungen.

Der Casio hat eine Vielzahl von Funktionen und Befehle, welche als Anweisungen verwendet werden können. Vor allen bei den Mathematischen Funktionen bietet das Casio Basic eine sehr große Auswahl.

Erstellte Programme werden mit dem Befehl RUN gestartet. Ein laufendes Programm kann mit der Taste [BRK] unterbrochen werden. Wenn ein Programm eines Programmbereichs gelöscht werden soll, verwendet man den Befehl NEW. Sollen alle Programme in allen Programmbereichen gelöscht werden, verwendet man NEW ALL. Einzelne Zeilen eines Programms werden durch die Eingabe der Zeilennummer, gefolgt von [EXE] gelöscht.



**Hinweis**

Die Basic-Programmierung ist in der CASIO-Bedienungsanleitung, Kapitel 6 ab Seite 46 ausführlich beschrieben.  
Die Basic-Befehls-Referenz finden Sie im Kapitel 10, ab Seite 91.

<sup>1</sup> Das war 1983, während meiner Ausbildungszeit.

## 2.2 Die CASIO-Bedienungsanleitung

Ich gehe davon aus, dass Sie im Besitz der CASIO-Bedienungsanleitung sind, da ich ab und zu auf dortige Kapitel verweise. Falls nicht, rate ich Ihnen die CASIO-Bedienungsanleitung im Internet<sup>2</sup> herunter zu laden. Meiner Meinung nach ist die CASIO-Bedienungsanleitung ein sehr gutes Nachschlagewerk. Es ist in folgende 11 Kapitel unterteilt:

1. Konfiguration des Geräts.....	1
2. Grundbedienung.....	11
3. Berechnungsfunktionen.....	15
4. Formelspeicher-Funktion.....	33
5. Datenbank-Funktion.....	37
6. Basic-Programmierung.....	45
7. Peripheriegeräte.....	71
8. Kompatibilität mit der Serie PB-100.....	80
9. Grundlegendes zum Umgang mit Dateien.....	87
10. Befehls-Referenz.....	91
11. Wissenschaftliche Bibliothek.....	180

Wie Sie sehen, wird alles Notwendige behandelt. Schade nur, dass bei der Befehlsreferenz nicht einfach nur nach dem Alphabet sortiert wurde. Das würde das Auffinden wesentlich vereinfachen. So muss man sich zunächst im Klaren sein, unter welcher Kategorie man suchen soll:

1. Manuelle Befehle.....	92
CLEAR, EDIT, FRE, LIST, LIST ALL, NEW, NEW ALL, PASS, RUN, TROFF, TRON, VARLIST	
2. Grundbefehle.....	100
BEEP, CLS, DATA, DEFSEG, DIM, END, ERASE, ERL, ERR, FOR ~ NEXT, GOSUB, GOTO, IF ~ THEN ~ ELSE, IF ~ GOTO ~ ELSE, INKEY\$, INPUT, INPUT\$, LET, LOCATE, ON ERROR GOTO, ON GOSUB, ON GOTO, PEEK, POKE, PRINT, READ, REM ( ' ), RESTORE, RESUME, RETURN, SET, STOP, TAB	
3. Numerische Funktionen.....	133
ABS, ACS, ANGLE, ASN, ATN, COS, CUR, EXP, FACT, FIX, FRAC, HYP ACS, HYP ASN, HYP ATN, HYP COS, HYP SIN, HYP TAN, INT, LN, LOG, NCR, NPR, PI, POL, RAN#, REC, ROUND, SGN, SIN, SQR, TAN	
4. Zeichen-Funktionen.....	150
&H, ASC, CHR\$, DEG, DMS\$, HEX\$, LEFT\$, LEN, MID\$, RIGHT\$, STR\$, VAL, VALF	
5. Eingabe/Ausgabe-Befehle.....	162
CLOSE, EOF, INPUT#, INPUT\$, LLIST, LOAD, LOAD ALL, LPRINT, OPEN, PRINT#, SAVE, SAVE ALL, VERIFY	
6. Datenbank-Befehle.....	174
LIST#, LLIST#, LOAD#, NEW#, READ#, RESTORE#, SAVE#, WRITE#	

Am Ende dieses Tutorials habe ich deshalb die alphabetisch geordnete Befehls-Übersicht angefügt, incl. Verweis auf die Seite im Casio Handbuch.

<sup>2</sup> <http://www.silrun.info/>

## 2.3 *Interesse und Zeit*

Eine wesentliche Voraussetzung für das erfolgreiche Bearbeiten dieses Tutorials ist natürlich ihr Interesse an diesem Thema. Gleichzeitig müssen Sie auch einiges an Zeit investieren. Um das Programmieren zu erlernen ist es leider nicht damit abgetan dieses Tutorial von vorne bis hinten durchzulesen. Viel wichtiger ist es, die Theorie in die Praxis umzusetzen. Und für dieses Erarbeiten benötigt man viel Zeit.

Gerade zu Beginn wird es häufig vorkommen, dass etwas nicht so funktioniert, wie Sie es sich vorgestellt haben. Sie werden manchmal Stunden damit verbringen den Quellcode wieder und wieder durchzuforschen, auf der Suche nach dem Fehler - um am Ende feststellen zu müssen, dass es wieder einmal ein kleiner Tippfehler war. Lassen Sie sich davon nicht entmutigen, denn auch so etwas gehört zum Lernen dazu.

Geübte Leser werden dieses Tutorial an einem Tag durchgelesen haben. Ich gehe aber davon aus, dass Sie erst nach einem Monat soweit sind, eigene Programme in BASIC zu schreiben. Ich verspreche Ihnen aber, die Zeit bis dahin wird spannend, interessant und abwechslungsreich werden.

BASIC-Programmierung ist der ideale Einstieg in die Welt der Programmierung. Gerade wenn Sie noch gar keine Erfahrung mit irgendeiner Programmiersprache haben, kann ich ihnen BASIC wärmstens empfehlen.

Aber warum gerade BASIC, zu einer Zeit, in der Sprachen wie Visual C++, C#.NET, Delphi oder Java gibt (um nur einige zu nennen). Ist BASIC nicht längst veraltet? Macht es nicht mehr Sinn gleich in eine aktuelle Hochsprache einzusteigen, und so was Antiquarisches wie dieses BASIC einfach links liegen zu lassen?

Nein, denn gerade beim Erlernen einer Programmiersprache finde ich es sehr wichtig, wenn man sich dabei auf das Wesentliche konzentrieren kann. Und bei BASIC ist das der Fall, denn Sie müssen sich nicht um andere Dinge kümmern. Später werden Sie es mit ihrem Wissen viel leichter haben, auch andere Programmiersprachen zu lernen.

In diesem Sinne, wünsche ich Ihnen mit den folgenden Kapiteln viel Spaß!

### 3 Basic-Listings der Scientific Library 116

Hier finden Sie alle Basic-Quelltexte der Scientific Library 116 des Casio FX-850P/880P.

Bei einigen Statistik-Programmen werden Unterprogramme aufgerufen. Diese sind im Quelltext als Link ausgeführt. Dadurch können Sie auch zu dem Quelltext der dieser Unterprogramme springen.

#### 3.1 TRANSLIB

Sie fragen sich sicher, woher ich die Quellcodes der Library Programme habe. Nun, diese befinden sich im ROM des Taschenrechners. Man muss sie lediglich in den Programmbereich des Rechners laden, damit sie sichtbar werden. Dazu ist das Program TRANSLIB notwendig.

Dieses Programm überträgt ein beliebiges Casio Library-Programm in den Programmbereich P0 ihres Casio FX-850P/FX-880P.

Stellen Sie vor dem Übertragen sicher, dass genügend Speicherplatz auf Ihrem Casio-Rechner vorhanden ist (Der Basic-Befehl NEW löscht ein Programm im aktuellen Programmbereich. Falls ihr alle Programme löschen wollen, verwendet den Basic-Befehl NEW ALL. Denkt aber vorher an Ihre Datensicherung!

Nun kopieren Sie das Programm TRANSLIB1 nach P1 und das Programm TRANSL2 nach P2. Da das Library-Programm in P0 erzeugt wird, sollten Sie diesen Programmbereich unbedingt löschen (falls noch nicht geschehen). Starten Sie das Programm im Programmbereich P1, und geben Sie die gewünschte LIBRARY-Nummer ein. Sobald Sie eine Nummer eingegeben haben, wird das Library- Programm im Programmbereich P0 erstellt. Dort können Sie es listen, starten oder abspeichern (aber nicht editieren!).

#### Listing 2 TRANLIB 1

```

10 REM Programm:TRANS-LIB -> P1
11 REM Aufgabe :Hauptprogramm
20 CLEAR: CLS: PRINT "Dieses Hilfsprogramm uebertraestein Library-Pros. nach P0."
30 INPUT "Bitte geben Sie die Library- Programm-Nr. ein :";NR#
40 DEFSEG=0: X=PEEK1715+PEEK1716*256-15
50 A=PEEK1718+PEEK1719*256+PEEK1720*65536
60 B=PEEK1724+PEEK1725*256+PEEK1726*65536
70 FOR I=A TO B-15 STEP 15
80 DEFSEG=I/16: C=I-INT(I/16)*16+6
90 FOR K=1 TO LEN(NR#): IF ASC(MID$(NR#,K)=PEEK(K+C) THEN NEXT K: GOTO 110
100 NEXT I: PRINT "Gesuchtes Library-Programm nichtgefunden. Bitte neu waehlen!"
: GOTO 10
110 REM Adressen ermitteln
120 DEFSEG=0: IF PEEK(X+2)=0 THEN A0=PEEK X: A1=PEEK(X+1): A3=PEEK(X+3)
: A4=PEEK(X+4)
130 FOR K=0 TO 5: DEFSEG=I/16: N=PEEK(K+I-INT(I/16)*16): DEFSEG=0: POKE X+K,N
: NEXT K
140 PRINT "Library-Programm-Nr.:";NR#;: PRINT " ist nun in P0 gespeichert!"

```

Das Programm TRANSLIB2 im Programmbereich P2 macht den Programmbereich P0 wieder normal nutzbar. Jetzt erst kann das Programm auf einen PC übertragen werden.

**Listing 3**

## TRANLIB 2

```
10 REM Programm:TRANS-LIB -> P2
11 REM Aufgabe :P0 loeschen
20 INPUT "P0 loeschen? [J/N] ";Q$
30 IF Q$<>"J" OR X=0 THEN END
40 POKE X+0,A0
41 POKE X+1,A1
42 POKE X+2,0
43 POKE X+3,A3
44 POKE X+4,A4
45 POKE X+5,0
46 X=0
```

Mit diesem Programm habe ich damals (1994) alle Library Programme heruntergeladen! In dem folgenden Kapitel können Sie das Ergebnis bewundern.

Hinweis: Dieses Programm TRANSLIB hatte ich damals aus einem Buch zum Casio entnommen und etwas an meine Bedürfnisse angepasst. Leider weiss ich heute den genauen Titel nicht mehr.

## 3.2 Menü- und Startprogramm

### 3.2.1 K0630 Menü- und Startprogramm für die LIB-Programme

#### LIB K0630 Menü- und Startprogramm für die LIB-Programme

```

10 CLS:GOTO10000
1000 DATA"Memory calculations" [MC],[MR],[M-],[M+]",,1000,5010
5010 DATA"Prime factors" Base = a * b * c * ...",,1000,5020
5020 DATA"G.C.M. & L.C.M.",,5010,5040
5040 DATA"ax1+bx2+cx3+...=y",,5020,5050
5050 DATA"ax2+bx+c=0",,5040,5060
5060 DATA"ax3+bx2+cx+d=0",,5050,5080
5080 DATA"Numerical solution f(x)=0" Newton's method",,5060,5090
5090 DATA"Numerical solution f(x)=0" Method of bisection",,5080,5100
5100 DATA"Matrix operations",,5090,5200
5200 DATA"∫f(x)dx [a,b]" Romberg's method",,5100,5220
5220 DATA"dy/dx=f(x,y(x))" Runse-Kutta method",,5200,5230
5230 DATA"Lagrange's interpolation",,5220,5250
5250 DATA"Gamma function Γ(x)",82FE8280C0,5230,5260
5260 DATA"Bessel function Jn(x)",,5250,5270
5270 DATA"Bessel function Yn(x)",,5260,5280
5280 DATA"Modified Bessel function In(x)",,5270,5290
5290 DATA"Modified Bessel function Kn(x)",,5280,5300
5300 DATA"Complex number a+bi",,5290,5350
5350 DATA"Binary-Decimal-Hexadecimal",,5300,5510
5510 DATA"y=ax+b ← (x1,y1),(x2,y2)",,5350,5520
5520 DATA"Angle(θ) ← y=ax+b,y=cx+d",,5510,5530
5530 DATA"Distance ← y=ax+b,(x1,y1)",,5520,5540
5540 DATA"(X,Y) ← (x,y),angle(θ)",,5530,5550
5550 DATA"Circle (x-a)2+(y-b)2=r2" ← (x1,y1),(x2,y2),(x3,y3)",,5540,5560
5560 DATA"Length(y=cx+d)" ← (x-a)2+(y-b)2=r2,(x1,y1)",,5550,5570
5570 DATA"y=cx+d,y=ex+f" ← (x-a)2+(y-b)2=r2,(x1,y1)",,5560,5600
5600 DATA"Area (triangle)",,5570,5605
5605 DATA"Area (trapezoid)",,5600,5610
5610 DATA"Area (parallelogram)",,5605,5615
5615 DATA"Area (circle)",,5610,5620
5620 DATA"Area (sector)",,5615,5625
5625 DATA"Area (segment)",,5620,5630
5630 DATA"Area (ellipse)",,5625,5635
5635 DATA"Area (polygon)",,5630,5650
5650 DATA"Surface (sphere)",,5635,5655
5655 DATA"Surface (zone of sphere)",,5650,5660
5660 DATA"Surface (spherical sector)",,5655,5665
5665 DATA"Surface (circular cylinder)",,5660,5670
5670 DATA"Surface (circular cone)",,5665,5675
5675 DATA"Surface (frustum of circular cone)",,5670,5700
5700 DATA"Volume (sphere)",,5675,5705
5705 DATA"Volume (zone of sphere)",,5700,5710
5710 DATA"Volume (spherical sector)",,5705,5715
5715 DATA"Volume (circular cylinder)",,5710,5720
5720 DATA"Volume (circular cone)",,5715,5725
5725 DATA"Volume (frustum of circular cone)",,5720,5730
5730 DATA"Volume (wedge)",,5725,5735
5735 DATA"Volume (pyramid)",,5730,5740
5740 DATA"Volume (frustum of pyramid)",,5735,5745
5745 DATA"Volume (ellipsoid)",,5740,5750
5750 DATA"Polygon A,n → r,R,l",,5745,5760
5760 DATA"Polyhedron (a,r,R,S,U)",,5750,5800
5800 DATA"Factorization" a2+2ab+b2=(a+b)2",,5760,5810
5810 DATA"Trigonometric function" sin2θ+cos2θ=1",,5800,5820
5820 DATA"Differential" y=ax2+bx+c → y'=2ax+b",,5810,5830
5830 DATA"Integration" ∫ x dx = 1/2·x2+C",,5820,5840
5840 DATA"Laplace transformation" F(p)=1/p → f(t)=1",,5830,5900
5900 DATA"Periodic table" H Hydrogen 1(1-1a) 1.00794",,5840,5910
5910 DATA"Scientific constant" e=9.80665 [m·s-2]",,5900,5920
5920 DATA"Electrolytic dissociation" HCOOH ↔ HCOO- + H+",,5910,5930
5930 DATA"Motion" Ek=1/2·mv2",,5920,5932

```

## LIB KO630 Menü- und Startprogramm für die LIB-Programme

```

5932 DATA"Waves
5934 DATA"AC & DC circuits
5936 DATA"Electric & Magnetic fields
5938 DATA"Thermodynamics and others
5950 DATA"Metric conversion (length)
5960 DATA"Metric conversion (area)
5970 DATA"Metric conversion (volume)
5980 DATA"Metric conversion (weight)
6210 DATA"Upper probability integrals
6220 DATA"Upper probability integrals
6230 DATA"Upper probability integrals
6240 DATA"Upper probability integrals
6310 DATA"Upper cumulative
6320 DATA"Upper cumulative
P(x, m)", ,424C300806, 6310, 6330
6330 DATA"Upper cumulative
6410 DATA"Percentage points
6420 DATA"Percentage points
6430 DATA"Percentage points
6440 DATA"Percentage points
6450 DATA"Normal random numbers", ,6440, 6460
6460 DATA"Exponential random numbers", ,6450, 6500
6500 DATA"Statistics [x]", ,6460, 6510
6510 DATA"Regression analysis
6520 DATA"Regression analysis
6530 DATA"Regression analysis
6540 DATA"Regression analysis
6610 DATA"Estimation a<μ<b
6620 DATA"Estimation a<μ<b
6630 DATA"Estimation a<σ<b
6640 DATA"Estimation a<σ<b
6650 DATA"Estimation a<σ2/σ2<b
6660 DATA"Estimation a<μ2-μ<b
6670 DATA"Estimation a<P<b
6680 DATA"Estimation a<P2-P<b
6710 DATA"Test N(μ, σ2) σ2: known
6711 DATA"Test N(μ, σ2) σ2: known
6712 DATA"Test N(μ, σ2) σ2: known
6720 DATA"Test N(μ, σ2)
6721 DATA"Test N(μ, σ2)
6722 DATA"Test N(μ, σ2)
6730 DATA"Test N(μ, σ2)
6731 DATA"Test N(μ, σ2)
6732 DATA"Test N(μ, σ2)
6740 DATA"Test N(μ2, σ2), N(μ2, σ2)
6741 DATA"Test N(μ2, σ2), N(μ2, σ2)
6742 DATA"Test N(μ2, σ2), N(μ2, σ2)
6750 DATA"Test N(μ2, σ2), N(μ2, σ2)
6751 DATA"Test N(μ2, σ2), N(μ2, σ2)
6752 DATA"Test N(μ2, σ2), N(μ2, σ2)
6760 DATA"Test B(1, P)
6761 DATA"Test B(1, P)
6762 DATA"Test B(1, P)
6770 DATA"Test B(1, P2), B(1, P2)
6771 DATA"Test B(1, P2), B(1, P2)
6772 DATA"Test B(1, P2), B(1, P2)
10000 ONERRORGOTO10400
10010 MODE8: DIM: x=1000: y=6772
10020
DEFSEG=0: a=PEEK(&H743)*256+PEEK(&H742): r$=HEX$(PEEK(&H744))+ "000": b=VAL("&H"+r$)
10030 DEFSEG=b: r$="": FOR i=a TO a+3: r$=r$+CHR$(PEEK(i+7)): NEXT i: r=VAL(r$)
10060 DEFCHR$(254)="3E02040830"
10070 DEFCHR$(253)="2E2A3A0000"
10080 DEFCHR$(252)="123E020000"
10110 RESTORE(r)
10120 READm$, n$, b, c
10130 IF n$<>" " THEN DEFCHR$(255)=n$
10140 CLS: PRINTMID$(STR$(r), 2): " ": m$:
10150 z=ASC(INPUT$(1, a))
f=fo*(v-v0)/(v-v0)", ,5930, 5934
R=R0+R0 , 1/R=1/R0+1/R0", ,5932, 5936
Q=CU , C=eo*S/d", ,5934, 5938
W=JQ , J=4.19 [J/cal]", ,5936, 5950
1[cm] -> 0.01[m]", ,5938, 5960
1[cm2] -> 0.0001[m2]", ,5950, 5970
1[cm3] -> 0.000001[m3]", ,5960, 5980
1[g] -> 0.001[kg]", ,5970, 6210
N(0, 12)", ,5980, 6220
X2(x2, m)", ,6210, 6230
t(x, m)", ,6220, 6240
F(x, m, m)", ,6230, 6310
frequencies B(x, n, P)", ,6240, 6320
frequencies
frequencies H(x, n, M, N)", ,6320, 6410
N(0, 12)", ,6330, 6420
X2(x2, m)", ,6410, 6430
t(x, m)", ,6420, 6440
F(x, m, m)", ,6430, 6450
[y=a+bx]", ,6500, 6520
[y=a+b*lnx]", ,6510, 6530
[y=ab*x]", ,6520, 6540
[y=ax^b]", ,6530, 6610
N(μ, σ2) σ2: known", ,6540, 6620
N(μ, σ2)", ,6610, 6630
N(μ, σ2)", ,6620, 6640
N(μ, σ2)", ,6630, 6650
N(μ2, σ2), N(μ2, σ2)", ,6640, 6660
N(μ2, σ2), N(μ2, σ2)", ,6650, 6670
B(1, P)", ,6660, 6680
B(1, P2), B(1, P2)", ,6670, 6710
Ho: μ=μ0 Ho: μ≠μ0", ,6680, 6711
Ho: μ=μ0 Ho: μ>μ0", ,6710, 6712
Ho: μ=μ0 Ho: μ<μ0", ,6711, 6720
Ho: μ=μ0 Ho: μ≠μ0", ,6712, 6721
Ho: μ=μ0 Ho: μ>μ0", ,6720, 6722
Ho: μ=μ0 Ho: μ<μ0", ,6721, 6730
Ho: σ2=σ02 Ho: σ2≠σ02", ,6722, 6731
Ho: σ2=σ02 Ho: σ2>σ02", ,6730, 6732
Ho: σ2=σ02 Ho: σ2<σ02", ,6731, 6740
Ho: σ2=σ02 Ho: σ2≠σ02", ,6732, 6741
Ho: σ2=σ02 Ho: σ2>σ02", ,6740, 6742
Ho: σ2=σ02 Ho: σ2<σ02", ,6741, 6750
Ho: μ=μ0 Ho: μ≠μ0", ,6742, 6751
Ho: μ=μ0 Ho: μ>μ0", ,6750, 6752
Ho: μ=μ0 Ho: μ<μ0", ,6751, 6760
Ho: P=P0 Ho: P≠P0", ,6752, 6761
Ho: P=P0 Ho: P>P0", ,6760, 6762
Ho: P=P0 Ho: P<P0", ,6761, 6770
Ho: P=P02 Ho: P≠P02", ,6762, 6771
Ho: P=P02 Ho: P>P02", ,6770, 6772
Ho: P=P02 Ho: P<P02", ,6771, 6772

```

**LIB K0630** Menü- und Startprogramm für die LIB-Programme

```
10160 IFz=29 ANDr<>x THENa=&HF7EA:r=x:GOTO10110
10170 IFz=28 ANDr<>y THENa=&HFEA7:r=y:GOTO10110
10180 IFz=30 ANDr<>b THENa=a-15:r=b:GOTO10110
10190 IF(z=31 ORz=218) ANDr<>c THENa=a+15:r=c:GOTO10110
10200 IFz=13 ORz=252 THEN10300
10210 GOTO10150
10300 CLS:DEFSEG=0:a$=HEX$(a):b=VAL("&H"+LEFT$(a$,2)):c=VAL("&H"+RIGHT$(a$,2))
:POKE&H742,c:POKE&H743,b:r$="LIB0:"+MID$(STR$(r),2):GOTOr$
10400 IFERR=1 THENONERRORGOTO0
10410 r=x:RESUME10110
```



### 3.3 Mathematik

#### 3.3.1 1000 Speicher-Berechnung

##### LIB 1000 Speicher-Berechnung

```

5  ONERRORGOTO500
10  MODE8:DIM:a,a#:DIMa(3),a$(1):DEFSEG=0
100 CLS:PRINT" MC[↑] MR[↓] M-[←] M+[→]";
110 LOCATE0,1:PRINTCHR$(5);a(2):LOCATE16,1:PRINTa(3):LOCATE0,0:LOCATE0,1
120 a$(0)=INPUT$(1,0):a(0)=ASC(a$(0)):IFA(0)=46 OR(a(0)>47 ANDa(0)<58) THEN200
130 IFA(0)=42 ORa(0)=43 ORa(0)=45 ORa(0)=47 ORa(0)=94
THENa$(0)=STR$(a(2))+a$(0):GOTO200
140 IFA(0)<14 ORa(0)>127 THEN110
150 IFA(0)=40 ORa(0)=41 THEN200
160 IFA(0)>27 ANDa(0)<32 THEN300
200 LOCATE0,0:LOCATE0,1:PRINTCHR$(5);a$(0);
210 a(0)=ASC(INPUT$(1)):a(1)=PEEK(206):IFA(0)>32 ANDa(0)<127 ANDa(1)=0
THENPRINTCHR$(a(0)):GOTO210
220 IFA(0)=8 THENPRINTCHR$(8):GOTO210
230 IFA(0)=31 ANDa(1)=0 THENGOSUB350:PRINTa$(1):GOTO210
240 IFA(0)=13 ORa(0)=28 ORa(0)=29 THEN400
250 IFA(0)=30 THENa(3)=0:GOTO210
260 IFA(0)=12 THENa$(0)="":GOTO200
270 IFA(0)=9 ORa(0)=14 ORa(0)=15 THENPRINTCHR$(a(0));
280 GOTO210
300 IFA(0)=28 THENa(3)=a(3)+a(2):GOTO110
310 IFA(0)=29 THENa(3)=a(3)-a(2):GOTO110
320 IFA(0)=30 THENa(3)=0:GOTO110
330 IFA(0)=31 THENGOSUB350:a$(0)=a$(1):GOTO200
350 a$(1)=STR$(a(3)):IFA(3)>=0 THENa$(1)=MID$(a$(1),2)
360 RETURN
400 MODE33(32,1024,223):IFCALC$="" THEN110
405 a(2)=VALF(CALC$):IFA(0)=13 THEN110
410 IFA(0)=28 THENa(3)=a(3)+a(2):GOTO110
420 IFA(0)=29 THENa(3)=a(3)-a(2):GOTO110
500 IFERR=1 THENCLS:ONERRORGOTO0
510 LOCATE0,0:LOCATE0,1:PRINTCHR$(5);" Error":w#=INPUT$(1,0)
520 IFERL=300 ORERL=310 THENRESUME110
530 LOCATE0,1:PRINTCHR$(5):CALC$:RESUME210

```

## 3.3.2 5010 Primfaktoren-Analyse

## LIB 5010 Primfaktoren-Analyse

```

5 ONERRORGOTO1000
10 MODE8:DIM:c#=CHR$(5)
20 CLS:PRINT"Prime factors (2≤Base<10¹⁰)":a$="*"
40 LOCATE0,1:PRINTc#;"Base ?":INPUT@11;x:LOCATE0,0
45 IFx<2 ORx>=1e10 ORFRACx<>0 THEN40
50 LOCATE0,1:PRINTc#;x;" = .....":
60 u=0:v=0:n=1:t$="":a=x
70 GOTO130
100 u=2:GOSUB500
110 a=a/2
120 IFa-1=0 THEN240
130 IFFRAC(a/2)=0 THEN100
140 b=3
150 c=SQRa+1
160 IFb>=c THEN230
170 IFFRAC(a/b)=0 THEN190
180 b=b+2:GOTO160
190 IFa/b*b-a=0 THEN210
200 GOTO180
210 u=b:GOSUB500
220 a=a/b:GOTO150
230 u=a:GOSUB500
240 a$="":GOSUB530
250 PRINTc#;x;"=";t$;:z#=INPUT$(1,@):GOTO20
500 IFv=u THENn=n+1:GOTO550
520 IFv=0 THEN550
530 IFn>1 THENt$=t$+STR$(v)+"^"+MID$(STR$(n),2)+a$ ELSEt$=t$+STR$(v)+a$
540 n=1
550 v=u:RETURN
1000 IFERR=1 THENCLS:ONERRORGOTO0
1010 IFERL=40 THENRESUME40
1050 LOCATE0,1:PRINTc#;"not found":z#=INPUT$(1,@):RESUME20

```

## 3.3.3 5020 Grösster gemeinsamer Teiler, Kleinstes gemeinsames Vielfaches

## LIB 5020 Grösster gemeinsamer Teiler, Kleinstes gemeinsames Vielfaches

```

5 ONERRORGOTO1000
10 MODE8:DIM:c#=CHR$(5):a=1:b=1
20 CLS:PRINT"G.C.M. & L.C.M. (1≤a,b<10¹⁰)":
30 s$="a":z=a:GOSUB700:a=z
40 s$="b":z=b:GOSUB700:b=z:CLS
100 IFb>a THENf=b:a=a ELSEf=a:a=b
110 h=f-INT(f/a)*a:IFh THENf=a:a=h:GOTO110
200 l=a/a*b:PRINT"G.C.M.=";a:TAB(0);"L.C.M.=";l;:z#=INPUT$(1,@):GOTO20
700 LOCATE0,1:PRINTc#;s#;"=";z;"?":INPUT@11;z:LOCATE0,0:z=ABSz
730 IFz>=1e10 ORz<1 ORFRACz<>0 THEN700 ELSERETURN
1000 IFERR=1 THENCLS:ONERRORGOTO0
1010 IFERL=700 THENRESUME700
1050 LOCATE0,1:PRINTc#;"not found":z#=INPUT$(1,@):RESUME20

```

## 3.3.4 5040 Gleichungssystem (Gauss-Elimination)

## LIB 5040 Gleichungssystem (Gauss-Elimination)

```

5  ONERRORGOTO2000:DEFSEG=0
10  MODE0:DIM:ERASEa:c#=CHR$(5):n=2:m#="ax1+bx2+cx3+dx4+ex5+fx6+ax7"
30  CLS:PRINT"ax1+bx2+cx3+...=y      (2≤n≤7)";
40  LOCATE0,1:PRINTc#:"n=":n:"?":INPUT@2:n:LOCATE0,0:IFn<2 ORn>7 THEN40
60  m=n-1:DIMa(m,n):P#=LEFT$(m#,4*m+3)+"=y":CLS:PRINTP#;
100 FORi=0 TOm:FORj=0 TOn
110  s#=MID$(STR$(i+1),2)+" "+MID$(P#,j*4+1,1):z=a(i,j)
120  GOSUB1950:IFw=0 THENa(i,j)=z:GOTO140
130  IFj=0 THENIFi=0 THEN30 ELSEi=i-1:j=n-1 ELSEj=j-2
140  NEXT:NEXT
160  LOCATE0,1:PRINTc#:"x1 = .....":GOTO830
310 FORi=0 TOm:z=b(i):IFz<>0 ANDABSz>=1e-90 THENz=ROUND(z,LOG(ABSz)-7)
315 IFz<>0 ANDABSz<1e-90 THENz=z*1e10:z=ROUND(z,LOG(ABSz)-7):z=z/1e10
320  PRINTc#:"x":MID$(STR$(i+1),2):" =":z;
330  w=ASC(INPUT$(1,@)):IFw=29 ORw=28 THEN30
360  IFw<>13 ANDw<>31 ANDw<>30 THEN330
370  IF(w=31 ANDi=m) OR(w=30 ANDi=0) THEN330
390  IFw=30 THENi=i-2
400  NEXT:GOTO30
830  ERASEb,c,e:y=2*n-1:x=m:DIMc(x,y),e(x),b(x)
840  FORi=0 TOm:FORj=0 TOm:c(i,j)=a(i,j):NEXT:NEXT
850  FORi=0 TOm:FORj=n TOy:IFi=j-n THENc(i,j)=1
860  NEXT:NEXT
870  FORi=0 TOx:e(i)=i:NEXT
880  FORk=0 TOx:w=ABS(c(k,k)):a=k:b=k
890  FORi=k TOx:FORj=k TOx:IFw<ABS(c(i,j)) THENw=ABS(c(i,j)):a=i:b=j
900  NEXT:NEXT
920  IFw<1E-10 THENGOSUB1800:GOTO30
940  IFk<>a THENFORj=k TOy:w=c(k,j):c(k,j)=c(a,j):c(a,j)=w:NEXT
950  IFk<>b THENFORi=0
TOx:w=c(i,k):c(i,k)=c(i,b):c(i,b)=w:NEXT:w=e(k):e(k)=e(b):e(b)=w
960  w=c(k,k):c=k+1:FORj=c TOy:c(k,j)=c(k,j)/w:NEXT
970  FORi=0 TOx:IFi<>k THENw=c(i,k):FORj=c TOy:c(i,j)=c(i,j)-w*c(k,j):NEXT
980  NEXT:NEXT
1040 FORj=n TOy:FORi=0 TOx:w=e(i):c(w,x)=c(i,j):NEXT
1050 FORi=0 TOx:c(i,j)=c(i,x):NEXT:NEXT
1270 FORi=0 TOm:w=0:FORk=0 TOm:w=w+c(i,k+n)*a(k,n):NEXT:b(i)=w:NEXT:GOTO310
1800 LOCATE0,1:PRINTc#:"not found":z#=INPUT$(1,@):RETURN
1950 LOCATE0,1:PRINTc#:"s#=":z:"?":INPUT@43:z:LOCATE0,0:w=PEEK(&H65B)-13:RETURN
2000 IFERR=1 THENCLS:ONERRORGOTO0
2020 IFERR=40 THENRESUME40
2030 IFERR=1950 THENRESUME1950
2040 IFERR=60 THENERASEa:RESUME60
2080 GOSUB1800:RESUME30

```

## 3.3.5 5050 Quadratische Gleichung

## LIB 5050 Quadratische Gleichung

```

5  ONERRORGOTO1000
10  MODE0: DIM: c#=CHR$(5): a=1: b=0: c=0
20  CLS: PRINT "ax²+bx+c=0";
30  s#="a": z=a: GOSUB700: a=z: IF z=0 THEN 30
40  s#="b": z=b: GOSUB700: b=z
50  s#="c": z=c: GOSUB700: c=z
60  LOCATE0,1: PRINT c#: ". . . . .";
100 d=b*b-4*a*c
110 CLS: IF d THEN 130
120 PRINT "a(x-α)²=0": PRINT: GOTO200
130 PRINT "a(x-α)(x-β)=0": PRINT
140 ONSGND+2 GOTO400,300
200 x=-b/(2*a): GOSUB510: PRINT c#: "α =": x:
210 GOSUB550: IF w=13 THEN 20 ELSE 210
300 IF b<0 THEN z=-b+SQ R d ELSE z=-b-SQ R d
305 x=z/2/a: y=2*c/z: GOSUB500
310 PRINT c#: "α =": x:
330 GOSUB550: IF w<>13 AND w<>31 THEN 330
340 PRINT c#: "β =": y:
360 GOSUB550: IF w=30 THEN 310 ELSE IF w=13 THEN 20 ELSE 360
400 x=-b/(2*a): y=SQ R(-d)/(2*a): GOSUB500: x#=STR$(x)+" ": y#=MID$(STR$(y),2): IF x=0
THEN x#=""
410 PRINT c#: "α =": x#: IF x THEN PRINT "+";
420 IF y<-1 THEN PRINT " ": y#: "i": ELSE PRINT " i";
430 GOSUB550: IF w<>13 AND w<>31 THEN 430
440 PRINT c#: "β =": x#: "-": IF x THEN PRINT " ";
450 IF y<-1 THEN PRINT y#: "i": ELSE PRINT "i";
460 GOSUB550: IF w=30 THEN 410 ELSE IF w=13 THEN 20 ELSE 460
500 IF y<>0 AND ABS y>=1e-90 THEN y=ROUND(y, LOG(ABS y)-7)
505 IF y<>0 AND ABS y<1e-90 THEN y=y*1e10: y=ROUND(y, LOG(ABS y)-7): y=y/1e10
510 IF x<>0 AND ABS x>=1e-90 THEN x=ROUND(x, LOG(ABS x)-7)
515 IF x<>0 AND ABS x<1e-90 THEN x=x*1e10: x=ROUND(x, LOG(ABS x)-7): x=x/1e10
520 RETURN
550 LOCATE0,0: LOCATE0,1: w=ASC(INPUT$(1,@)): RETURN
700 LOCATE0,1: PRINT c#: s#: "=": z: "?": INPUT@42: z: LOCATE0,0: RETURN
1000 IF ERR=1 THEN CLS: ONERRORGOTO0
1010 IF ERR=700 THEN RESUME 700
1050 LOCATE0,1: PRINT c#: "not found": z#=INPUT$(1,@): RESUME 20

```

## 3.3.6 5060 Kubische Gleichung

LIB 5060

Kubische Gleichung

```

5  ONERRORGOTO1000
10  MODE0: DIM: c#=CHR$(5): ANGLE0: a=1: b=0: c=0: d=0
20  CLS: PRINT "ax³+bx²+cx+d=0";
30  s#="a": z=a: GOSUB700: a=z: IF a=0 THEN30
40  s#="b": z=b: GOSUB700: b=z
50  s#="c": z=c: GOSUB700: c=z
60  s#="d": z=d: GOSUB700: d=z
70  LOCATE0,1: PRINTc#:".....";
100 u=b/a/3: p=c/a/3-u*u: q=2*u*u*u-u*c/a+d/a
102 IFABS p<u*u*1e-10 THENp=0
104 IFABS q<u*u*u*1e-10 THENq=0
110 t=q*q+4*p*p*p: IFABSt<q*q*5e-10 THENt=0
115 r=(q+SQ R(ABSt))/2: s=(q-SQ R(ABSt))/2: r=SGNr*CUR(ABSr): s=SGNs*CUR(ABSs)
120 v=r+s
130 ONSGnt+2 GOTO400,,300
140 x=-v-u: IF p THEN200
175 GOSUB520: PRINT"a(x-α)³=0": PRINT
180 PRINTc#:"α =":x;
190 GOSUB550: IFw=13 THEN20 ELSE190
200 y=v/2-u
205 GOSUB510: PRINT"a(x-α)(x-β)²=0": PRINT
210 PRINTc#:"α =":x;
230 GOSUB550: IFw<>13 ANDw<>31 THEN230
240 PRINTc#:"β =":y;
260 GOSUB550: IFw=30 THEN210 ELSEIFw=13 THEN20 ELSE260
300 x=-v-u: y=v/2-u: z=ABS(SQ R3/2*(r-s))
302 GOSUB500: PRINT"a(x-α)(x-β)(x-γ)=0": PRINT
305 y#=STR$(y)+" ": z#=MID$(STR$(z),2): IFy=0 THENy#=""
310 PRINTc#:"α =":x;
330 GOSUB550: IFw<>13 ANDw<>31 THEN330
340 PRINTc#:"β =":y#: IFy THENPRINT"+";
350 IFz=1 THENPRINT" ": z#:"i": ELSEPRINT" i";
360 GOSUB550: IFw=30 THEN310 ELSEIFw<>13 ANDw<>31 THEN360
370 PRINTc#:"r =":y#: "-": IFy THENPRINT" ";
380 IFz=1 THENPRINTz#:"i": ELSEPRINT"i";
390 GOSUB550: IFw=30 THEN340 ELSEIFw=13 THEN20 ELSE390
400 h=ACS(q/2/SQ R(ABS(-p*p*p)))/3: i=-2*SQ R(ABS(-p)): x=i*COSH-u: y=i*COS(h+120)-
u: z=i*COS(h+240)-u
405 GOSUB500: PRINT"a(x-α)(x-β)(x-γ)=0": PRINT
410 PRINTc#:"α =":x;
430 GOSUB550: IFw<>13 ANDw<>31 THEN430
440 PRINTc#:"β =":y;
460 GOSUB550: IFw=30 THEN410 ELSEIFw<>13 ANDw<>31 THEN460
470 PRINTc#:"r =":z;
490 GOSUB550: IFw=30 THEN440 ELSEIFw=13 THEN20 ELSE490
500 IFz<>0 ANDABSz>=1e-90 THENz=ROUND(z,LOG(ABSz)-7)
502 IFz<>0 ANDABSz<1e-90 THENz=z*1e10: z=ROUND(z,LOG(ABSz)-7): z=z/1e10
505 IFABSz<ABSu*1e-10 THENz=0
510 IFy<>0 ANDABSy>=1e-90 THENy=ROUND(y,LOG(ABSy)-7)
512 IFy<>0 ANDABSy<1e-90 THENy=y*1e10: y=ROUND(y,LOG(ABSy)-7): y=y/1e10
515 IFABSy<ABSu*1e-10 THENy=0
520 IFx<>0 ANDABSx>=1e-90 THENx=ROUND(x,LOG(ABSx)-7)
522 IFx<>0 ANDABSx<1e-90 THENx=x*1e10: x=ROUND(x,LOG(ABSx)-7): x=x/1e10
525 IFABSx<ABSu*1e-10 THENx=0
530 CLS: RETURN
550 LOCATE0,0: LOCATE0,1:w=ASC(INPUT$(1,@)): RETURN
700 LOCATE0,1: PRINTc#;s#:"=";z#:"?": INPUT@42: z: LOCATE0,0: RETURN
1000 IFERR=1 THENCLS: ONERRORGOTO0
1010 IFERR=700 THENRESUME700
1050 LOCATE0,1: PRINTc#:"not found": z#=INPUT$(1,@): RESUME20

```

## 3.3.7 5080 Numerische Lösung einer Gleichung (Newton-Methode)

## LIB 5080 Numerische Lösung einer Gleichung (Newton-Methode)

```

5  ONERRORGOTO1000
10  DEFCHR$(255)="0000FE000":MODE8:DIM:PRINTCHR$(15):
30  f$="":c$=CHR$(5):s=0:h=1e-5:e=1e-7:m=20:ANGLE1
50  CLS:PRINT"Newton's method  f(x)=0":TAB(0):"1:f(x),x0      2:h,ε,loop":
60  k=ASC(INPUT$(1,0)):IFk=13 ORk=49 THEN90 ELSEIFk=50 THEN150
80  GOTO60
90  CLS:PRINT"Define function":
100 LOCATE0,1:PRINTc$:"f(x)
?":f$=:LOCATE0,0:LOCATE6,1:INPUT@100:f$=:LOCATE0,0:z=VALF(f$)
110 IFLEN(f$)<25 THENs$=f$ ELSEs$=LEFT$(f$,21)+"..."
120 m$="f(x) = "+s$+s$="x0":z=s:GOSUB800:s=z:GOTO250
150 m$="f'(x)=(f(x+h)-f(x))/h (h>0)":s$="h":z=h:GOSUB800:h=z:IFh<0 THEN150
170 m$="Err: eXn+1-Xn<ε (ε>0)":s$="e":z=e:GOSUB800:e=z:IFE<1e-90 THEN170
200 m$="Max loop (n>0)":s$="n":z=m:GOSUB800:m=z:IFm<0 ORFRACm<>0 THEN200 ELSE50
250 CLS:PRINTm$:TAB(0):"x = .....":t=s
310 FORi=1 TOM
320 x=t:s=VALF(f$):x=t+h:f=VALF(f$):x=t-s*h/(f-s)
350 IFABS(t-x)<=e THEN500
360 t=x:NEXT:GOSUB450
390 PRINTc$:"loop =":m:TAB(11):"Xn
=:STR$(x):LOCATE0,0:LOCATE1,1:z$=INPUT$(1,0):GOTO50
450 LOCATE0,1:PRINTc$:"not found":z$=INPUT$(1,0):RETURN
500 x=ROUND(x,INTLOG(ABSε)-1):PRINTc$:"x =":x:z$=INPUT$(1,0):GOTO50
800 CLS:PRINTm$:
810 LOCATE0,1:PRINTc$:"s=":z$="?":INPUT@10:z=:LOCATE0,0:RETURN
1000 IFERR=1 THENCLS:ONERRORGOTO0
1010 IFERL=100 THENIFERR=2 THENRESUME100 ELSERESUMENEXT
1015 IFERL=810 THENRESUME810
1020 GOSUB450:RESUME50

```

## 3.3.8 5090 Numerische Lösung einer Gleichung (Halbierungs-Methode)

## LIB 5090

Numerische Lösung einer Gleichung (Halbierungs-Methode)

```

5 ONERRORGOTO1000
10 DEFCHR$(255)="0000FE0000":MODE8:DIM:PRINTCHR$(15);
30 f$="":c$=CHR$(5):s=0:t=0:e=1e-7:m=30:ANGLE1
50 CLS:PRINT"Method of bisection f(x)=0      1:f(x),x0,x1      2:s,loop":
60 k=ASC(INPUT$(1,@)):IFk=13 ORk=49 THEN90 ELSEIFk=50 THEN170
80 GOTO60
90 CLS:PRINT"Define function":
100 LOCATE0,1:PRINTc$:"f(x)
?" : f$ : LOCATE0,0:LOCATE6,1: INPUT@100: f$: LOCATE0,0: z=VALF(f$)
110 IFLEN(f$)<25 THENs$=f$ ELSEs$=LEFT$(f$,21)+"..."
120 m$="f(x) = "+s$:s$="x0":z=s:GOSUB800:s=z
130 s$="x1":z=t:GOSUB800:t=z:GOTO250
170 m$="Err:  eXn+1-Xn<e (e>0)":s$="e":z=e:GOSUB800:e=z:IFE<1e-90 THEN170
200 m$="Max loop (n>0)":s$="n":z=m:GOSUB800:m=z:IFm<0 ORFRACm<>0 THEN200 ELSE50
250 CLS:PRINTm$:TAB(0):"x = .....":u=s:v=t
270 x=u:s=VALF(f$):IFs=0 THEN500
280 x=v:h=VALF(f$):IFh=0 THEN500
310 FORi=1 TOm:x=(u+v)/2:f=VALF(f$):IFf=0 THEN500
330 IFSGNs=SGNh ANDSGNh=SGNf THENGOSUB450:GOTO50
340 IFSGN(f*s)=1 THENu=x:s=f ELSEv=x:h=f
350 IFABS(u-v)<=e THEN500
370 NEXT:GOSUB450
390 PRINTc$:"loop =":m:TAB(11):":Xn
=" : STR$(x) : LOCATE0,0:LOCATE1,1:z#=INPUT$(1,@):GOTO50
450 LOCATE0,1:PRINTc$:"not found":z#=INPUT$(1,@):RETURN
500 x=ROUND(x,INTLOG(ABSs)-1):PRINTc$:"x =":x:z#=INPUT$(1,@):GOTO50
800 CLS:PRINTm$:
810 LOCATE0,1:PRINTc$:s$="":z$="?": INPUT@10:z:LOCATE0,0:RETURN
1000 IFERR=1 THENCLS:ONERRORGOTO0
1010 IFERL=100 THENIFERR=2 THENRESUME100 ELSERESUMENEXT
1015 IFERL=810 THENRESUME810
1020 GOSUB450:RESUME50

```

## 3.3.9 5100 Matrixoperationen

## LIB 5100 Matrixoperationen

```

5  ONERRORGOTO2000:DEFSEG=0
10  CLS:MODE8:DIM:ERASEa,b,d:DIMa(1,1),b(1,1),d(1,1):m=1:n=1:o=1:p=1:q=1:r=1
   :c#=CHR$(5):e=1
40  GOSUB460:GOSUB470:GOSUB420
50  LOCATE0,0:PRINTc#:"Matrix  A";m#;"x";B";n#;
60  LOCATE28,1:z=ASC(INPUT$(1)):IFz>96 THENz=z-32
100 ONz-41 GOTO1250,1300,,1350,1500,,,,,,,,,,,,,,,,,120,130,700,800
   ,,,,800,,1400,1100,1150,,300,,,1200
110 GOTO60
120 x=m:GOTO500
130 x=o:GOTO600
300 LOCATE0,0:FORi=0 TOm:FORj=0 TOm:z=a(i,j):IFz<>0 ANDABSz>=1e-90
   THENz=ROUND(z,LOG(ABSz)-7)
310 IFz<>0 ANDABSz<1e-90 THENz=z*1e10:z=ROUND(z,LOG(ABSz)-7):z=z/1e10
320 PRINTc#:"a(";MID$(STR$(i+1),2);", ";MID$(STR$(j+1),2);") =";z;
330 z=ASC(INPUT$(1,0)):IFz=28 ORz=29 THEN50
340 IFz>96 THENz=z-32
350 ONz-41 GOTO1250,1300,,1350,1500,,,,,,,,,,,,,,,,,120,130,700,800
   ,,,,800,,1400,1100,1150,,300,,,1200
360 IFz<>13 ANDz<>31 ANDz<>30 THEN330
370 IF(z=31 ANDi+j=n+m) OR(z=30 ANDi+j=0) THEN330
390 IFz=30 THENIFj=0 THENi=i-1:j=n-1 ELSEj=j-2
400 NEXT: NEXT:GOTO50
420 CLS:PRINT"PRINT">A,B,D,I,T,K,+,-,*,M,L,C,P ?":RETURN
440 FORi=0 TOm:FORj=0 TOm:a(i,j)=c(i,j):NEXT: NEXT:RETURN
460 n#="(" +MID$(STR$(o+1),2)+", "+MID$(STR$(p+1),2)+)":RETURN
470 m#="(" +MID$(STR$(m+1),2)+", "+MID$(STR$(n+1),2)+)":RETURN
480 m#="(" +MID$(STR$(i+1),2)+", "+MID$(STR$(j+1),2)+)":RETURN
500 i=x:j=n:GOSUB480:P#="A(m,n) = A"+m#;s#="n":z=x+1:GOSUB1900:IFw THEN40
ELSEx=z-1
510 i=x:j=n:GOSUB480:P#="A(m,n) = A"+m#;s#="n":z=n+1:GOSUB1900:IFw THEN500
520 m=x:n=z-1
530 DIMa(m,n)
540 FORi=0 TOm:FORj=0 TOm
550 P#="a(" +MID$(STR$(i+1),2)+", "+MID$(STR$(j+1),2)+)"
560 z=a(i,j):GOSUB1850:IFw=13 THENa(i,j)=z:GOTO580
570 IFj=0 THENIFi=0 THEN510 ELSEi=i-1:j=n-1 ELSEj=j-2
580 NEXT: NEXT:GOTO40
600 i=x:j=p:GOSUB480:P#="B(m,n) = B"+m#;s#="n":z=x+1:GOSUB1900:IFw THEN40
ELSEx=z-1
610 i=x:j=p:GOSUB480:P#="B(m,n) = B"+m#;s#="n":z=p+1:GOSUB1900:IFw THEN600
620 o=x:p=z-1
630 DIMb(o,p)
640 FORi=0 TOo:FORj=0 TOp
650 P#="b(" +MID$(STR$(i+1),2)+", "+MID$(STR$(j+1),2)+)"
660 z=b(i,j):GOSUB1850:IFw=13 THENb(i,j)=z:GOTO680
670 IFj=0 THENIFi=0 THEN610 ELSEi=i-1:j=n-1 ELSEj=j-2
680 NEXT: NEXT:GOTO40
700 LOCATE0,0:PRINTc#:"Change  A";m#;" ↔ B";n#;:IFm=o ANDn=p THEN750
710 ERASEc:DIMc(m,n):FORi=0 TOm:FORj=0 TOm:c(i,j)=a(i,j):NEXT: NEXT
720 ERASEa:DIMa(o,p):FORi=0 TOo:FORj=0 TOp:a(i,j)=b(i,j):NEXT: NEXT
730 ERASEb:DIMb(m,n):FORi=0 TOm:FORj=0 TOm:b(i,j)=c(i,j):NEXT: NEXT
740 x=m:y=n:m=o:n=p:o=x:p=y:z#=m#:m#=n#:n#=z#:GOTO50
750 FORi=0 TOm:FORj=0 TOm:w=a(i,j):a(i,j)=b(i,j):b(i,j)=w: NEXT: NEXT:GOTO50
800 LOCATE0,0:IFm<>n THENGOSUB1830:GOTO50
810 IFz=68 THENPRINTc#:"Determinant A = ....."; ELSEPRINTc#:"Inverse A → A";
820 t=1E-10:d=1
830 ERASEc,e:y=2*(m+1)-1:x=m:DIMc(x,y),e(x)
840 FORi=0 TOm:FORj=0 TOm:c(i,j)=a(i,j):NEXT: NEXT
850 FORi=0 TOm:FORj=n+1 TOy:IFI=j-n-1 THENc(i,j)=1
860 NEXT: NEXT
870 FORi=0 TOx:e(i)=i: NEXT
880 FORk=0 TOx:w=ABS(c(k,k)):a=k:b=k
890 FORi=k TOx:FORj=k TOx:IFw<ABS(c(i,j)) THENw=ABS(c(i,j)):a=i:b=j
900 NEXT: NEXT

```



## LIB 5100 Matrixoperationen

```

920 IFw<t THENIFz=68 THENd=0:GOTO1000 ELSEGOSUB1830:GOTO50
940 IFk<>a THENd=-d:FORj=k TOy:w=c(k,j):c(k,j)=c(a,j):c(a,j)=w:NEXT
950 IFk<>b THENd=-d:FORi=0 TOx:w=c(i,k):c(i,k)=c(i,b):c(i,b)=w:NEXT
:w=e(k):e(k)=e(b):e(b)=w
960 w=c(k,k):d=d*w:c=k+1:FORj=c TOy:c(k,j)=c(k,j)/w:NEXT
970 FORi=0 TOx:IFI<>k THENw=c(i,k):FORj=c TOy:c(i,j)=c(i,j)-w*c(k,j):NEXT
980 NEXT:NEXT
1000 IFz=68 THENPRINTc#:"Determinant A =":STR$(d):z#=INPUT$(1,0):GOTO50
1040 w=x+1:FORj=w TOy:FORi=0 TOx:w=e(i):c(w,x)=c(i,j):NEXT
1050 FORi=0 TOx:c(i,j)=c(i,x):NEXT:NEXT
1070 FORi=0 TOx:FORj=0 TOx:a(i,j)=c(i,j+x+1):NEXT:NEXT:GOTO300
1100 m=q:n=r:GOSUB470:LOCATE0,0:PRINTc#:"Load A ← M":m#
1110 ERASEa:DIMa(m,n):FORi=0 TOm:FORj=0 Ton:a(i,j)=d(i,j):NEXT:NEXT:GOTO50
1150 q=m:r=n:GOSUB470:LOCATE0,0:PRINTc#:"Memory A → M":m#
1160 ERASEd:DIMd(q,r):FORi=0 TOm:FORj=0 Ton:d(i,j)=a(i,j):NEXT:NEXT:GOTO50
1200 w=m:n=n:w:GOSUB470:LOCATE0,0:PRINTc#:"Transpose A → A":m#
1210 ERASEc:DIMc(m,n):FORi=0 Ton:FORj=0 Tom:c(j,i)=a(i,j):NEXT:NEXT
1220 ERASEa:DIMa(m,n):GOSUB440:GOTO300
1250 LOCATE0,0:IFn<>o THENGOSUB1830:GOTO50
1260 i=m:j=p:GOSUB480:PRINTc#:"A*B → A":m#
1270 ERASEc:DIMc(m,p):FORi=0 TOm:FORj=0 TOp:w=0:FORk=0 Ton:w=w+a(i,k)*b(k,j):NEXT
:c(i,j)=w:NEXT:NEXT:IFm<>o OR n<>p THENn=p:ERASEa:DIMa(m,n)
1280 GOSUB440:GOTO300
1300 LOCATE0,0:IFm<>o ORn<>p THENGOSUB1830:GOTO50
1310 PRINTc#:"A+B → A":m#
1320 ERASEc:DIMc(m,n):FORi=0 TOm:FORj=0 Ton:c(i,j)=a(i,j)+b(i,j):NEXT:NEXT
:GOSUB440:GOTO300
1350 LOCATE0,0:IFm<>o ORn<>p THENGOSUB1830:GOTO50
1360 PRINTc#:"A-B → A":m#
1370 ERASEc:DIMc(m,n):FORi=0 TOm:FORj=0 Ton:c(i,j)=a(i,j)-b(i,j):NEXT:NEXT
:GOSUB440:GOTO300
1400 F#="k*A"+m#+ " → A":z=e:GOSUB1950:GOSUB420:IFw THEN50 ELSEe=z
1410 LOCATE0,0:PRINTF#
1420 ERASEc:DIMc(m,n):FORi=0 TOm:FORj=0 Ton:c(i,j)=e*a(i,j):NEXT:NEXT
:GOSUB440:GOTO300
1500 DEFCHR$(254)="0000F00000":CLS:GOSUB1600:i=2
1510 PRINT:ONi GOSUB1600,1610,1620,1630,1640,1650,1660,1670,1680,1690
,1700,1710,1720
1520 z=ASC(INPUT$(1,0)):IFz=28 ORz=29 THEN1580
1530 IFz<>13 ANDz<>31 ANDz<>30 THEN1520
1540 IF(z=31 ANDi=13) OR(z=30 ANDi=1) THEN1520
1550 IFz=30 THENi=i-2
1570 i=i+1:IFI<14 THEN1510
1580 GOSUB420:GOTO50
1600 PRINT" A : input A(m,n)":RETURN
1610 PRINT" B : input B(m,n)":RETURN
1620 PRINT" D : determinant A":RETURN
1630 PRINT" I : inverse A^-1 → A":RETURN
1640 PRINT" T : transposition A^t → A":RETURN
1650 PRINT" K : scalar-cal k*A → A":RETURN
1660 PRINT" + : addition A+B → A":RETURN
1670 PRINT" - : subtraction A-B → A":RETURN
1680 PRINT" * : multiplication A*B → A":RETURN
1690 PRINT" M : memory A → M":RETURN
1700 PRINT" L : load A ← M":RETURN
1710 PRINT" C : change A ↔ B":RETURN
1720 PRINT" P : print A(m,n)":RETURN
1830 LOCATE0,0:PRINTc#:"not found":z#=INPUT$(1,0):RETURN
1850 CLS:PRINTF#:"=":z:
1860 LOCATE0,1:PRINTc#:"?":INPUT@16:z:LOCATE0,0:w=PEEK(&H65B):RETURN
1900 CLS:PRINTF#:
1910 LOCATE0,1:PRINTc#:"?":INPUT@3:z:LOCATE0,0:w=PEEK(&H65B)-13
:IFw THENRETURN
1920 IFz<1 ORz>10 ORFRAC(z)<>0 THEN1910 ELSERETURN
1950 CLS:PRINTF#:
1960 LOCATE0,1:PRINTc#:"k=":z:"?":INPUT@12:z:LOCATE0,0:w=PEEK(&H65B)-13:RETURN
2000 IFERR=1 THENCLS:ONERRORGOTO0
2010 IFERR=1860 THENRESUME1860
2020 IFERR=1910 THENRESUME1910

```

**LIB 5100** Matrixoperationen

```

2030 IFERL=1960 THENRESUME1960
2040 IFERL=530 THENERASEa:RESUME530
2050 IFERL=630 THENERASEb:RESUME630
2080 GOSUB1830:RESUME40

```

**3.3.10 5200 Numerische Integration (Romberg-Methode)****LIB 5200** Numerische Integration (Romberg-Methode)

```

5 ONERRORGOTO1000
10 DEFCHR$(255)="0000FE0000":MODE8:DIM:PRINTCHR$(15):
30 f$="":c#=CHR$(5):a=0:b=0:e=1e-7:m=8:ANGLE1:ERASEa:DIMa(m)
50 CLS:PRINT"Romberg's method ∫f(x)dx [a,b] 1:f(x),[a,b] 2:s,loop":
60 k=ASC(INPUT$(1,0)):IFk=13ORk=49THEN90ELSEIFk=50THEN170
80 GOTO60
90 CLS:PRINT"Define function":
100 LOCATE0,1:PRINTc$:"f(x) ?":f$::LOCATE0,0:LOCATE6,1:INPUT@100:f$:LOCATE0,0
:z=VALF(f$)
110 IFLEN(f$)<21THENa#=f$ ELSEa#=LEFT$(f$,17)+"..."
120 m$="∫ "+a#+ " dx [a,b]":s$="a":z=a:GOSUB800:a=z
130 s$="b":z=b:GOSUB800:b=z:GOTO250
170 m$="Err: |An+1-An|<ε (ε>0)":s$="ε":z=e:GOSUB800:e=z:IFE<1e-90 THEN170
200 m$="Max loop 2^n (n>0)":s$="n":z=m:GOSUB800:m=z
:IFm<<0ORFRACm<>0THEN200ELSEERASEa:DIMa(m):GOTO50
250 CLS:PRINTm$:TAB(0):"∫f(x)dx = .....":
260 h=b-a:x=a:s=VALF(f$):x=b:f=VALF(f$):a(0)=(s+f)*h/2
300 FORi=1 TOm:h=h/2
310 s=0:FORj=1TO2^i-1STEP2:x=a+j*h:s=s+VALF(f$):NEXT
320 a(i)=a(i-1)/2+h*s
330 n=1:FORk=i-1TO0STEP-1
340 a(k)=a(k+1)+(a(k+1)-a(k))/(4^n-1)
350 IFABS(a(k+1)-a(k)) <e AND(i>3ORi=m)THENx=a(k):GOTO500
360 n=n+1:NEXT:NEXT:GOSUB450
390 PRINTc$:"n =":m:TAB(11):":An =":STR$(a(m-1)):LOCATE0,0:LOCATE1,1
:z#=INPUT$(1,0):GOTO50
450 LOCATE0,1:PRINTc$:"not found":z#=INPUT$(1,0):RETURN
500 x=ROUND(x,INTLOG(ABSε)-1):PRINTc$:"∫f(x)dx =":x:z#=INPUT$(1,0):GOTO50
800 CLS:PRINTm$:
810 LOCATE0,1:PRINTc$:"s$="":z$="?":INPUT@10:z:LOCATE0,0:RETURN
1000 IFERR=1 THENCLS:ONERRORGOTO0
1010 IFERL=100 THENIFERR=2 THENRESUME100 ELSERESUMENEXT
1015 IFERL=810 THENRESUME810
1020 GOSUB450:RESUME50

```

## 3.3.11 5220 Gewöhnliche Differenzialgleichung (Runge-Kutta-Methode)

LIB 5220

Gewöhnliche Differenzialgleichung (Runge-Kutta-Methode)

```

5  ONERRORGOTO1000
10  DEFCHR$(254)="060A12227E":MODE8:DIM:PRINTCHR$(15);
30  f$="":c$=CHR$(5):a=0:b=0:h=1:ANGLE1
90  CLS:PRINT"Define function dy/dx";
100 LOCATE0,1:PRINTc$:"?";f$=:LOCATE0,0:LOCATE1,1:INPUT@100:f$
:LOCATE0,0:z=VALF(f$)
110 IFLEN(f$)<17 THENa$=f$ ELSEa$=LEFT$(f$,20)+"..."
120 m$="dy/dx = "+a$:s$="x0":z=a:GOSUB800:a=z
140 s$="y0":z=b:GOSUB800:b=z
150 m$="Step-size  =h  (<h>0)":s$="eh":z=h:GOSUB800:h=z:IFh<0 THEN150
250 m$="dy/dx = "+a$:n=50
270 ERASEa,b:DIMa(n),b(n):k=32:i=0:j=0:f=a:a=b:GOTO355
300 IFk>31 THENCLS:PRINTm$:TAB(0);".....": ELSEPRINTc$:".....":
305 f=a(i-1):a=b(i-1):j=i
310 x=f:y=a:p=VALF(f$)*h
320 x=f+h/2:y=a+p/2:q=VALF(f$)*h
330 y=a+q/2:r=VALF(f$)*h
340 x=f+h:y=a+r:s=VALF(f$)*h
350 f=f+h:a=a+(p+2*q+2*r+s)/6
355 a(i)=f:b(i)=a
360 y$=""+STR$(a)
:IFSGNf<0 THENz$="y("&STR$(f)&") " ELSEz$="y("&MID$(STR$(f),2)&") "
365 l=LEN(z$)+LEN(y$):IFl>31 THENCLS
:PRINTz$:TAB(0);y$: ELSEIFk>31 THENCLS:PRINTm$:TAB(0);z$:y$: ELSEPRINTc$:"z$:y$";
370 z=ASC(INPUT$(1,@)):k=1:IFz=29 ORz=28 THEN90
375 IFz=31 ANDi=n THEN370
380 IFz=13 ORz=31 THENIFi=n THEN100 ELSEi=i+1:f=a(i):a=b(i)
:IFj>=i THEN360 ELSE300
390 IFz=30 ANDi>0 THENi=i-1:f=a(i):a=b(i):GOTO360
400 GOTO370
450 LOCATE0,1:PRINTc$:"not found":z$=INPUT$(1,@):RETURN
800 CLS:PRINTm$:
810 LOCATE0,1:PRINTc$:s$:"=";z:"?":INPUT@10;z:LOCATE0,0:RETURN
1000 IFERR=1 THENCLS:ONERRORGOTO0
1010 IFERR=100 THENIFERR=2 THENRESUME100 ELSERESUMENEXT
1020 IFERR=810 THENRESUME810
1030 GOSUB450:RESUME90

```

## 3.3.12 5230 Lagrangsche Interpolation

## LIB 5230 Lagrangsche Interpolation

```

5  ONERRORGOTO1000:DEFSEG=0
10  MODE0:DIM:c#=CHR$(5):n=2:x=0:ERASEa,b:m#="Lagrange's interpolation"
20  CLS:PRINTm#:TAB(0):"1:x      2:set data":
30  k=VAL(INPUT$(1,@)):ONk GOTO200,60
40  GOTO30
60  CLS:PRINT"Number of data":
70  s#="n":z=n:GOSUB500:n=z:IFw THEN20
75  IFn<2 ORn>200 ORFRACn<>0 THEN 70
80  DIMa(n),b(n):LOCATE15,0:PRINT"=":n:
110 FORi=1 TO n
120  s#="x"+MID$(STR$(i),2):z=a(i):GOSUB500
:IFw THENIFi=1 THEN60 ELSEi=i-1 ELSEa(i)=z
130  s#="y"+MID$(STR$(i),2):z=b(i):GOSUB500:IFw THEN120 ELSEb(i)=z
140  NEXT:GOTO20
200  LOCATE0,1:PRINTc#:"x?":INPUT@10:z#:LOCATE0,0:IFz#="" THEN20 ELSEx=VALF(z#)
220  c=0:FORi=1 TO n
230  P=1:FORj=1 TO n
240  IFi<j THENP=P*(x-a(j))/(a(i)-a(j))
250  NEXT
260  c=c+P*b(i):NEXT
280  LOCATE12,1:PRINT"y=":STR$(c):z#=INPUT$(1,@):GOTO200
500  LOCATE0,1:PRINTc#:s#:"=":z#:"?":INPUT@10:z:LOCATE0,0:w=PEEK(&H65B)-13:RETURN
1000 IFERR=1 THENCLS:ONERRORGOTO0
1010 IFERL=500 THENRESUME500
1020 IFERL=200 THENRESUME200
1030 IFERL=80 THENERASEa,b:RESUME80
1050 LOCATE0,1:PRINTc#:"not found":z#=INPUT$(1,@):RESUME20

```

3.3.13 5250 Gamma-Funktion  $\Gamma(x)$ LIB 5250 Gamma-Funktion  $\Gamma(x)$ 

```

5  ONERRORGOTO1000
10  MODE0:DIM:ERASEa:DIMa(0):c#=CHR$(5):DEFCHR$(255)="82FE8280C0"
20  RESTORE400:FORi=0 TO8:READa(i):NEXT:x=1
30  CLS:PRINT"Gamma function (0<x≤70)":
50  LOCATE0,1:PRINTc#:"x =":x:"?":INPUT@6:x:LOCATE0,0:IFx<0 ORx>70 THEN50
70  x#=MID$(STR$(x),2)
80  LOCATE0,1:PRINTc#:"Γ("x#") = .....":
100 IFx<1 THENy=x:GOSUB300:f=f/y:GOTO200
110 IFx<5 THENy=x-INTx:GOSUB300:FORi=1TOx-1:f=f*(y+i):NEXT:GOTO200
150 f=EXP((x-.5)*LNx-x+.5*LN(2*PI)+1/12/x-1/360/x^3+1/1260/x^5-1/1680/x^7)
200 IF f THENf=ROUND(f,LOG(ABSf)-7)
210 PRINTc#:"Γ("x#") =":f:z#=INPUT$(1,@):GOTO50
300 f=a(0):FORi=1 TO8:f=f*y+a(i):NEXT:RETURN
400 DATA .035868343,-.193527818,.482199394,-.756704078,.918206857
:,-.897056937,.988205891,-.577191652,1
1000 IFERR=1 THENCLS:ONERRORGOTO0
1010 IFERL=50 THENRESUME50
1020 LOCATE0,1:PRINTc#:"not found":z#=INPUT$(1,@):RESUME30

```

3.3.14 5260 Besselsche Funktion  $J_n(x)$ LIB 5260 Besselsche Funktion  $J_n(x)$ 

```

5  ONERRORGOTO500
10  MODE8: DIM: c#=CHR$(5)
20  CLS: PRINT "Jn(x)          (0≤n≤9, 0≤x≤30)";
30  LOCATE0,1: PRINT c#: "n?      :x?          :J=";
40  LOCATE2,1: INPUT@3;n#: LOCATE0,0:n=VALF(LEFT$(n#,3)):u=ABS n
: IF n<0 OR u>=10 OR FRAC u<>0 THEN 400
60  LOCATE8,1: INPUT@6;x#: LOCATE0,0:x=VALF(LEFT$(x#,6)):v=ABS x: IF x<0 OR v>30 THEN 410
80  LOCATE17,1: PRINT "      ....": GOSUB 200
100 IF n<0 THEN r=r*(-1)^u
110 IF n<>0 AND ABS r>=1e-90 THEN r=ROUND(n, LOG(ABS r)-7)
115 IF n<>0 AND ABS r<1e-90 THEN r=r*1e10: r=ROUND(n, LOG(ABS r)-7): r=r/1e10
120 LOCATE18,1: PRINT "          ": LOCATE17,1: PRINT r: z#=INPUT$(1,@): GOTO 30
200 IF v+u=0 THEN r=1: RETURN
210 IF v=0 THEN r=0: RETURN
220 k=INT(1+3*v^(1/12)+9*v^(1/3)+(ABS(u-v)+ABS(u+v))/2)
230 s=0: b=1.5: c=0
240 a=2*k/x*b-c: k=k-1
250 IF k=u THEN r=a
260 IF k=0 THEN 290
270 IF (k MOD 2)=0 THEN s=s+a
280 c=b: b=a: GOTO 240
290 s=2*s+a: r=r/s
300 RETURN
400 LOCATE2,1: PRINT "          ": GOTO 40
410 LOCATE8,1: PRINT "          ": GOTO 60
500 IF ERR=1 THEN CLS: ONERRORGOTO 0
510 IF ERR=40 THEN RESUME 400
520 IF ERR=60 THEN RESUME 410
530 LOCATE17,1: PRINT "not found": z#=INPUT$(1,@): RESUME 20

```

3.3.15 5270 Besselsche Funktion  $Y_n(x)$ LIB 5270 Besselsche Funktion  $Y_n(x)$ 

```

5  ONERRORGOTO500
10  MODE8: DIM: c#=CHR$(5)
20  CLS: PRINT "Yn(x)          (0≤n≤9, 0<x≤30)";
30  LOCATE0,1: PRINT c#: "n?   :x?       :Y=";
40  LOCATE2,1: INPUT@3;n#: LOCATE0,0:n=VALF(LEFT$(n#,3)):u=ABS n
: IF n<0 OR u>=10 OR FRAC u<>0 THEN 400
60  LOCATE8,1: INPUT@6;x#: LOCATE0,0:x=VALF(LEFT$(x#,6)):v=ABS x
: IF x<0 OR v>30 THEN 410
80  LOCATE17,1: PRINT " .....": GOSUB220
100 IF n<0 THEN r=r*(-1)^u
110 IF n<0 AND ABS r>=1e-90 THEN r=ROUND(r, LOG(ABS r)-7)
115 IF n<0 AND ABS r<1e-90 THEN r=r*1e10: r=ROUND(r, LOG(ABS r)-7): r=r/1e10
120 LOCATE18,1: PRINT "          ": LOCATE17,1: PRINT r: z#=INPUT$(1,@): GOTO30
220 k=INT(1+3*v^(1/12)+9*v^(1/3)+(ABS(u-v)+ABS(u+v))/2)
230 s=0: b=1.5: c=0: j=0
240 a=2*k/x*b-c: k=k-1
250 IF k=1 THEN d=a
260 IF k=0 THEN 290
270 IF (k MOD 2)=0 THEN s=s+a: j=j+2*a*(-1)^(k/2-1)/k
280 c=b: b=a: GOTO240
290 s=2*s+a: j=j/s
300 j=(a*(.577215664901+LN(v/2))/s+2*j)*2/PI
310 IF n=0 THEN r=j: RETURN
320 y=(d/s*j-2/PI/v)/a*s
330 IF u=1 THEN r=y: RETURN
340 FOR i=2 TO u: e=2*(i-1)*y/x-j: j=y: y=e: NEXT
350 r=e: RETURN
400 LOCATE2,1: PRINT "          ": GOTO40
410 LOCATE8,1: PRINT "          ": GOTO60
500 IF ERR=1 THEN CLS: ONERRORGOTO0
510 IF ERR=40 THEN RESUME400
520 IF ERR=60 THEN RESUME410
530 LOCATE17,1: PRINT "not found": z#=INPUT$(1,@): RESUME20

```

## 3.3.16 5280 Modifizierte Besselsche Funktion In(x)

## LIB 5280 Modifizierte Besselsche Funktion In(x)

```

5  ONERRORGOTO500
10  MODE8: DIM: c#=CHR$(5)
20  CLS: PRINT "In(x)          (0≤n≤9, 0≤x≤10)";
30  LOCATE0,1: PRINT c#: "n?   :x?   :I=";
40  LOCATE2,1: INPUT@3;n#: LOCATE0,0:n=VALF(LEFT$(n#,3)):u=ABS n
: IF n<0 OR u>=10 OR FRAC u<>0 THEN 400
60  LOCATE8,1: INPUT@6;x#: LOCATE0,0:x=VALF(LEFT$(x#,6)):v=ABS x
: IF x<0 OR v>10 THEN 410
80  LOCATE17,1: PRINT " .....": GOSUB200
100 IF x<0 THEN r=r*(-1)^u
110 IF r<>0 AND ABS r>=1e-90 THEN r=ROUND(r, LOG(ABS r)-7)
115 IF r<>0 AND ABS r<1e-90 THEN r=r*1e10: r=ROUND(r, LOG(ABS r)-7): r=r/1e10
120 LOCATE18,1: PRINT "      ": LOCATE17,1: PRINT r: z#=INPUT$(1,@): GOTO30
200 IF v+u=0 THEN r=1: RETURN
210 IF v=0 THEN r=0: RETURN
220 k=30+(ABS(u-v)+ABS(u+v))/2
230 IF v<2 THEN k=k-20
240 c=0: b=1E-99: s=0
250 FOR k=INT(k-1) TO 0 STEP-1
260 a=2*(k+1)*b/v+c
270 IF k=u THEN d=a
280 s=s+a: c=b: b=a
290 NEXT
300 s=2*s-a: r=d/s*EXP(v)
310 RETURN
400 LOCATE2,1: PRINT "      ": GOTO40
410 LOCATE8,1: PRINT "      ": GOTO60
500 IF ERR=1 THEN CLS: ONERRORGOTO0
510 IF ERR=40 THEN RESUME400
520 IF ERR=60 THEN RESUME410
530 LOCATE17,1: PRINT "not found": z#=INPUT$(1,@): RESUME20

```

3.3.17 5290 Modifizierte Besselsche Funktion  $K_n(x)$ LIB 5290 Modifizierte Besselsche Funktion  $K_n(x)$ 

```

5  ONERRORGOTO500
10  MODE8: DIM: c#=CHR$(5)
20  CLS: PRINT "Kn(x)          (0≤n≤9, 0<x≤10)";
30  LOCATE0,1: PRINT c#: "n?      :x?          :K=";
40  LOCATE2,1: INPUT@3;n#: LOCATE0,0:n=VALF(LEFT$(n#,3)):u=ABS n
: IF n<0 OR u>=10 OR FRAC u<>0 THEN 450
60  LOCATE8,1: INPUT@6;x#: LOCATE0,0:x=VALF(LEFT$(x#,6)):v=ABS x
: IF x<0 OR v>10 THEN 460
80  LOCATE17,1: PRINT " .....": GOSUB200
100 IF x<0 THEN r=r*(-1)^u
110 IF r<>0 AND ABS r>=1E-90 THEN r=ROUND(r,LOG(ABS r)-7)
115 IF r<>0 AND ABS r<1E-90 THEN r=r*1E10:r=ROUND(r,LOG(ABS r)-7):r=r/1E10
120 LOCATE18,1: PRINT "      ": LOCATE17,1: PRINT r: z#=INPUT$(1,@): GOTO30
200 IF v>2 THEN 400
210 c=0: b=1E-99: s=0
220 FOR k=11 TO 0 STEP -1: a=2*(k+1)*b/v+c: s=s+a: c=b: b=a: NEXT
230 s=2*s-a: d=a/s*EXP(v)
240 w=v/2: e=-LN(w)*d-.5772157+.4227842*w^2+.2306976*w^4+.0348859*w^6+.002627*w^8
+.0001075*w^10+.0000074*w^12
250 IF n=0 THEN r=e: RETURN
260 f=(c/s)*EXP(v): g=(1/v-f*e)/d
270 IF u=1 THEN r=g: RETURN
300 a=e: b=g
310 FOR i=1 TO u-1: c=2*i*b/v+a: a=b: b=c: NEXT
320 r=c: RETURN
400 w=2/v: e=(1.2533141-.0783236*w+.0218957*w^2-.0106245*w^3+.0058787*w^4
-.0025154*w^5+.0005321*w^6)*EXP(-v)/SQ R v
410 IF n=0 THEN r=e: RETURN
420 g=(1.2533141+.2349862*w-.0365562*w^2+.0150427*w^3-.0078035*w^4+.0032561*w^5
-.0006825*w^6)*EXP(-v)/SQ R v
430 IF u=1 THEN r=g: RETURN
440 GOTO300
450 LOCATE2,1: PRINT "      ": GOTO40
460 LOCATE8,1: PRINT "      ": GOTO60
500 IF ERR=1 THEN CLS: ONERRORGOTO0
510 IF ERR=40 THEN RESUME450
520 IF ERR=60 THEN RESUME460
530 LOCATE17,1: PRINT "not found": z#=INPUT$(1,@): RESUME20

```



## 3.3.18 5300 Komplexe Zahlen

## LIB 5300 Komplexe Zahlen

```

5  ONERRORGOTO1000
10  MODE0: DIM: DEFCHR$(255)="0000FE000": e=0: f=0: c=0: d=0: g=0: h=0: l$=CHR$(5)
: GOSUB580
20  a=e: b=f
30  x=a: GOSUB530: a=x: x=b: GOSUB530: b=x
50  IF b<0 THEN a$="-" ELSE a$="+"
60  LOCATE0,0: c$=STR$(a): IF a THEN d$=STR$(ABS b) ELSE d$=STR$(b)
70  IF ABS b=1 THEN d$=MID$(d$,1,1)
80  PRINT l$: IF b=0 THEN PRINT c$: ELSE IF a=0 THEN PRINT d$: "i": ELSE PRINT c$: a$: d$: "i":
90  LOCATE26,1: z=ASC(INPUT$(1)): IF z>96 THEN z=z-32
100 ON z-41 GOTO260,260,,260,600,260,,,,,,,,,,,,,220,,380,,,,420,,480
,,,400,360,,,,,500,,,,,,,,,,,,,450
110 GOTO90
220 CLS: PRINT "Complex number  A(a+bi)":
230 e$="a": x=a: GOSUB520: a=x: e$="b": x=b: GOSUB520: b=x: GOSUB580: GOTO50
260 CLS: PRINT "Complex number  B(c+di)":
270 e$="c": x=c: GOSUB520: c=x: e$="d": x=d: GOSUB520: d=x
290 GOSUB580: ON z-41 GOTO310,320,,330,,340
310 e=a*c-b*d: f=b*c+a*d: GOTO20
320 e=a+c: f=b+d: GOTO20
330 e=a-c: f=b-d: GOTO20
340 e=(a*c+b*d)/(c*d+d*d): f=(b*c-a*d)/(c*d+d*d): GOTO20
360 s=a: h=b: GOTO30
380 e=a: f=b: a=c: b=d: c=e: d=f: GOTO30
400 a=s: b=h: GOTO30
420 e=SQR(a*a+b*b): f=ACS(a/e): IF b THEN f=f*SGN(b)
430 CLS: PRINT "r  =": TAB(0): "θ  =": f: a$=INPUT$(1,0): GOSUB580: GOTO50
450 e=a*a-b*b: f=2*a*b: GOTO20
480 e=a/(a*a+b*b): f=-b/(a*a+b*b): GOTO20
500 IF b=0 THEN IF a<0 THEN f=SQR(-a): e=0: GOTO20 ELSE e=SQR a: f=0: GOTO20
510 IF b<>0 THEN x=SQR(a*a+b*b): e=SQR((a+x)/2): f=b/(2*e): GOTO20
520 LOCATE0,1: PRINT l$: e$: "=": x: "?"": INPUT@14: x: LOCATE0,0
530 IF x<>0 AND ABS x>=1e-90 THEN x=ROUND(x,LOG(ABS x)-8)
540 IF x<>0 AND ABS x<1e-90 THEN x=x*1e10: x=ROUND(x,LOG(ABS x)-8): x=x/1e10
550 RETURN
580 CLS: LOCATE0,1: PRINT ">A,G,I,S,^,+,-,*,/,M,L,C ?": RETURN
600 CLS: GOSUB710: i=2
610 PRINT: ON i GOSUB710,715,720,730,740,750,760,770,780,800,810,820
620 z=ASC(INPUT$(1,0)): IF z=28 OR z=29 THEN 690
630 IF z<>13 AND z<>30 AND z<>31 THEN 620
640 IF (z=30 AND i=1) OR (z=31 AND i=12) THEN 620
650 IF z=30 THEN i=i-2
680 i=i+1: IF i<13 THEN 610
690 GOSUB580: GOTO50
710 PRINT " A : input A": RETURN
715 PRINT " G : Gauss  r:=A: θ=argument(A)": RETURN
720 PRINT " I : inverse      1/A → A": RETURN
730 PRINT " S : square root   √A → A": RETURN
740 PRINT " ^ : square        A² → A": RETURN
750 PRINT " + : addition      A+B → A": RETURN
760 PRINT " - : subtraction   A-B → A": RETURN
770 PRINT " * : multiplication A*B → A": RETURN
780 PRINT " / : division      A/B → A": RETURN
800 PRINT " M : memory       A → M": RETURN
810 PRINT " L : load         A ← M": RETURN
820 PRINT " C : change     A ↔ B": RETURN
1000 IF ERR=1 THEN CLS: ON ERROR GOTO 0
1010 IF ERR=520 THEN RESUME 520
1030 LOCATE0,0: PRINT l$: "not found": a$=INPUT$(1,0): RESUME 690

```

## 3.3.19 5350 Binär-Dezimal-Hexadezimal

## LIB 5350 Binär-Dezimal-Hexadezimal

```

5  ONERRORGOTO2000
10  CLS:MODE8:DIM:a=1:x=0:y=0:c#=CHR$(5):P=2^31:q=2^32:r=0:x#=""
0":ERASEa#,b#:DIMa$(5),b$(15)
40  RESTORE70
50  FORi=0 TO5:READa$(i):NEXT
60  FORi=0 TO15:READb$(i):NEXT
70  DATABIN,DEC,HEX,AND,OR,XOR
80  DATA000,0001,0010,0011,0100,0101,0110,0111,1000,1001,1010,1011,1100,1101
,1110,1111
110  CLS:GOSUB300
120  z=LEN(x#):LOCATE0,0:IFz<17 THENPRINTc#:"[";a$(a);"] ";x#:ELSEPRINTc#;x#;
130  IFR>16 THENGOSUB300
140  r=z:LOCATE31,1
150  f=ASC(INPUT$(1)):IFf>96 THENf=f-32
160  IFf=73 THENGOSUB440:GOSUB290:GOTO330
170  IFf=66 ANDa<>0 THENa=0:GOTO330
180  IFf=68 ANDa<>1 THENa=1:GOTO330
190  IFf=72 ANDa<>2 THENa=2:GOTO330
200  IFf=42 ORf=43 ORf=45 ORf=47 THENGOSUB460:GOSUB290:GOTO800
210  IFf=65 ORf=79 ORf=88 THENGOSUB480:GOSUB290:GOTO900
220  IFf=76 ORf=82 THENGOSUB290:GOTO1040
230  IFf=78 ORf=67 THENGOSUB290:GOTO1100
240  IFf=46 THEN1200
250  GOTO150
270  LOCATE0,0:PRINTc#:"not found":z#=INPUT$(1,0):RETURN
290  LOCATE0,0:PRINTc#;
300  LOCATE0,1:PRINT">I,B,D,H,+,-,*,/,A,O,X,N,C,L,R?":RETURN
330  ONa+1 GOSUB360,350,360:GOTO120
350  x#=STR$(x):RETURN
360  IFx<0 THENt=x+a ELSEt=x
370  s=INT(t/65536):t=t-s*65536:x#=HEX$(s)+HEX$(t):IFA=2THENRETURN
380  y#="" FORi=1 TO8:y#=y#+b$(VAL("&H"+MID$(x#,i,1))):NEXT:x#=y#:RETURN
440  m#="Input data x (,B,D,H) ["+a$(a)+"] x ?":GOSUB550:x=y:RETURN
460  IFA>0 THENm#="["+a$(a)+"] " ELSEm#=""
465  m#=m#+x#+ " "+CHR$(f)+ " ":GOSUB550:RETURN
480  IFA>0 THENm#="["+a$(a)+"] " ELSEm#=""
485  m#=m#+x#+ " "+a$(SGN(f-79)+4)+ " ":GOSUB550:RETURN
550  CLS:PRINTm#;:INPUT@34;z#:LOCATE0,0:IFz#="" THENy=x:RETURN
560  l=LEN(z#):IFl>34 THENz#=LEFT$(z#,34)
570  d=a:c=1:IFl<2 THEN630
580  a#=MID$(z#,l-1,1)
590  IFA#=", " ORa#="/" THENc=l-2:d=ASC(RIGHT$(z#,1)):z#=LEFT$(z#,c) ELSE630
600  IFd>96 THENd=d-32
610  IFd=66 ORD=68 ORD=72 THENd=SGN(d-68)+1 ELSE550
630  y=0:OND+1 GOTO670,710,750
650  IFy<-P ORy>q-1 THEN550 ELSEIFY>=P THENy=y-q
660  RETURN
670  FORi=1TOc:z=ASC(MID$(z#,i,1))-48:IFz=0ORz=1THENy=y*2+z ELSE550
680  NEXT:GOTO650
710  e=0:w=0:FORi=1TOc:z=ASC(MID$(z#,i,1))
720  IF(z>47 ANDz<58) ORz=45 ORz=43 ORz=69 THEN730 ELSE550
730  IFw=0 THENy=INT(VALF(z#)):w=1:IFE=1 THEN550
740  NEXT:GOTO650
750  FORi=1 TOc:z=ASC(MID$(z#,i,1)):IFz>96 THENz=z-32
760  z=z-48:IFz<0 ORz>9 THENz=z-7:IFz<10 ORz>15 THEN550
770  y=y*16+z:NEXT:GOTO650
800  ONf-41 GOSUB840,850,,860,,870
810  IFw>=P ORw<=P THENGOSUB270:GOTO120 ELSEx=w:GOTO330
840  w=x*y:RETURN
850  w=x+y:RETURN
860  w=x-y:RETURN
870  w=FIX(x/y):RETURN
900  IFx<0 THENx=x+a
910  IFy<0 THENy=y+a
920  u=x-INT(x/65536)*65536:v=y-INT(y/65536)*65536:GOSUB950:z=u

```

## LIB 5350 Binär-Dezimal-Hexadezimal

```

930 u=INT(x/65536):v=INT(y/65536):GOSUB950:x=z+u*65536:IFx>=P THENx=x-a
940 GOTO330
950 IFu>32767 THENu=u-65536
960 IFv>32767 THENv=v-65536
970 IFf=65 THENu=u ANDv ELSEIFf=79 THENu=u ORv ELSEu=u XORv
980 IFu<0 THENu=u+65536
990 RETURN
1040 IFx<0 THENx=x+a
1050 IFf=76 THENx=x*2 ELSEx=INT(x/2)
1060 IFx>=a THENx=x-a
1070 IFx>=P THENx=x-a
1080 GOTO330
1100 IFf=78 THENx=-x-1 ELSEIFx>=P THENx=-x
1110 GOTO330
1200 CLS:GOSUB1300:i=2
1210 PRINT:ONi GOSUB1300,1310,1320,1330,1340,1350,1360,1370,1380,1390,1400,1410
,1420,1430,1440
1220 z=ASC(INPUT$(1,a)):IFz=28 ORz=29 THEN110
1230 IFz<>13 ANDz<>30 ANDz<>31 THEN1220
1240 IF(z=30 ANDi=1) OR(z=31 ANDi=15) THEN1220
1250 IFz=30 THENi=i-2
1270 i=i+1:IFI<16 THEN1210
1280 GOTO110
1300 PRINT" I : input data":RETURN
1310 PRINT" B : binary mode":RETURN
1320 PRINT" D : decimal mode":RETURN
1330 PRINT" H : hexadecimal mode":RETURN
1340 PRINT" + : addition":RETURN
1350 PRINT" - : subtraction":RETURN
1360 PRINT" * : multiplication":RETURN
1370 PRINT" / : division":RETURN
1380 PRINT" A : AND":RETURN
1390 PRINT" O : OR":RETURN
1400 PRINT" X : XOR":RETURN
1410 PRINT" N : NOT":RETURN
1420 PRINT" C : 2's complement":RETURN
1430 PRINT" L : bit shift left":RETURN
1440 PRINT" R : bit shift right":RETURN
2000 IFERR=1 THENCLS:ONERRORGOTO0
2010 IFERR=>840 ANDERR=<870 THENGOSUB270:RESUME120
2020 IFERR=730 THENe=1:RESUMENEXT
2030 GOSUB270:RESUME110

```

## 3.3.20 5510 Gerade durch zwei Punkte

## LIB 5510 Gerade durch zwei Punkte

```

5 ONERRORGOTO500
10 MODE8:DIM:ERASEa:DIMa(5):a$="x1y1x2y2ab":c$=CHR$(5)
20 CLS:PRINT"y=ax+b ← (x1,y1),(x2,y2)";
30 FORi=0 TO3
40 LOCATE0,1:PRINTc$:MID$(a$,1+2*i,2);"=";a(i);"?":INPUT@42:a(i):LOCATE0,0
50 NEXT:LOCATE0,1:PRINTc$:
70 IFA(0)=a(2) THENIFA(1)=a(3) THEN100 ELSEPRINTc$:"x ="a(0);
:w$=INPUT$(1,@):GOTO30
72 IFA(1)=a(3) THENPRINTc$:"y ="a(1);:w$=INPUT$(1,@):GOTO30
75 a(4)=(a(3)-a(1))/(a(2)-a(0)):a(5)=a(1)-a(4)*a(0)
80 FORi=4 TO5:PRINTc$:MID$(a$,i+5,1);"="a(i);
82 w=ASC(INPUT$(1,@)):IFw=13 THEN90
84 IFw=30 ANDi>4 THENi=i-2:GOTO90
85 IFw=31 ANDi<5 THEN90
88 GOTO82
90 NEXT:GOTO30
100 LOCATE0,1:PRINTc$:"not found":w$=INPUT$(1,@):GOTO20
500 IFERR=1 THENCLS:ONERRORGOTO0
510 IFERL=40 THENRESUME40
520 RESUME100

```

## 3.3.21 5520 Schnittwinkel von zwei Geraden

## LIB 5520 Schnittwinkel von zwei Geraden

```

5 ONERRORGOTO500
10 MODE8:DIM:a=0:c=0:c$=CHR$(5)
20 CLS:PRINT"Ansle(θ) ← y=ax+b,y=cx+d";
30 s$="a":z=a:GOSUB100:a=z
40 s$="c":z=c:GOSUB100:c=z:LOCATE0,1
60 IFA=c THENPRINTc$:"Parallel"; ELSEIFA*c=-1 THENPRINTc$:"Right angle";
ELSEPRINTc$:"θ ="ATN((c-a)/(1+c*a));
70 w$=INPUT$(1,@):GOTO30
100 LOCATE0,1:PRINTc$:s$="="z;"?":INPUT@42;z:LOCATE0,0:RETURN
500 IFERR=1 THENCLS:ONERRORGOTO0
510 IFERL=100 THENRESUME100
520 LOCATE0,1:PRINTc$:"not found":z$=INPUT$(1,@):RESUME20

```

## 3.3.22 5530 Abstand zwischen Punkt und Gerade

## LIB 5530 Abstand zwischen Punkt und Gerade

```

5 ONERRORGOTO500
10 MODE8:DIM:ERASEa:DIMa(3):a$="a b x1y1":c$=CHR$(5)
20 CLS:PRINT"Distance ← y=ax+b,(x1,y1)";
30 FORi=0 TO3
40 LOCATE0,1:PRINTc$:MID$(a$,1+i*2,(SGN(i-1.5)+1)/2+1);"="a(i);"?":
:INPUT@42:a(i):LOCATE0,0
50 NEXT
60 LOCATE0,1:PRINTc$:"d ="ABS(a(0)*a(2)-a(3)+a(1))/SQR(a(0)^2+1);
:w$=INPUT$(1,@):GOTO30
500 IFERR=1 THENCLS:ONERRORGOTO0
510 IFERL=40 THENRESUME40
520 LOCATE0,1:PRINTc$:"not found":w$=INPUT$(1,@):RESUME20

```

## 3.3.23 5540 Drehbewegung

## LIB 5540 Drehbewegung

```

5 ONERRORGOTO500
10 MODE0: DIM: ERASE a: DIM a(1): c$=CHR$(5): x=0: y=0: a=0: m$="XY"
20 CLS: PRINT "(X,Y) ← (x,y), angle(θ)":
30 s$="x": z=x: GOSUB100: x=z
40 s$="y": z=y: GOSUB100: y=z
50 s$="θ": z=a: GOSUB100: a=z
60 a(0)=x*COSa-y*SINa: a(1)=x*SINa+y*COSa: LOCATE0,1
70 FOR i=0 TO 1: PRINT c$: MID$(m$, i+1, 1): " =": a(i):
75 w=ASC(INPUT$(1,0)): IF w=13 THEN 90
80 IF w=30 AND i>0 THEN i=i-2: GOTO 90
82 IF w=31 AND i<1 THEN 90
88 GOTO 75
90 NEXT: GOTO 30
100 LOCATE0,1: PRINT c$: s$: " =": z: "?": INPUT@42: z: LOCATE0,0: RETURN
500 IFERR=1 THEN CLS: ONERRORGOTO0
510 IFERL=100 THEN RESUME 100
520 LOCATE0,1: PRINT c$: "not found": w$=INPUT$(1,0): RESUME 20

```

## 3.3.24 5550 Kreis durch drei Punkte

## LIB 5550 Kreis durch drei Punkte

```

5 ONERRORGOTO500
10 MODE0: DIM: ERASE a: DIM a(8): l$="abr": m$="x1y1x2y2x3y3": c$=CHR$(5)
20 CLS: PRINT "Circle (x1,y1), (x2,y2), (x3,y3)":
30 FOR k=0 TO 5
40 LOCATE0,1: PRINT c$: MID$(m$, 1+2*k, 2): " =": a(k): "?": INPUT@42: a(k): LOCATE0,0
50 NEXT: LOCATE0,1: PRINT c$:
60 a=a(0)-a(2): b=a(0)+a(2): c=a(1)-a(3): d=a(1)+a(3): e=(a*b+c*d)/2
70 f=a(2)-a(4): g=a(2)+a(4): h=a(3)-a(5): i=a(3)+a(5): j=(f*g+h*i)/2
80 IF a*h-c*f<>0 THEN a(7)=(a*j-e*f)/(a*h-c*f) ELSE 160
90 IF a<>0 THEN a(6)=(e-a(7)*c)/a ELSE IF f<>0 THEN a(6)=(j-a(7)*h)/f ELSE 160
100 a(8)=SQR((a(0)-a(6))^2+(a(1)-a(7))^2)
110 CLS: PRINT "Circle (x-a)^2+(y-b)^2=r^2": PRINT
120 FOR k=1 TO 3: PRINT c$: MID$(l$, k, 1): " =": a(k+5):
125 w=ASC(INPUT$(1,0)): IF w=13 THEN 140
130 IF w=30 AND k>1 THEN k=k-2: GOTO 140
132 IF w=31 AND k<3 THEN 140
138 GOTO 125
140 NEXT: GOTO 20
160 LOCATE0,1: PRINT c$: "not found": w$=INPUT$(1,0): GOTO 20
500 IFERR=1 THEN CLS: ONERRORGOTO0
510 IFERL=40 THEN RESUME 40
520 RESUME 160

```

## 3.3.25 5560 Länge von Tangenten von einem Punkt zu einem Kreis

**LIB 5560** Länge von Tangenten von einem Punkt zu einem Kreis

```

5 ONERRORGOTO500
10 MODE0:DIM:ERASEa:DIMa(4):a$="a b r x1y1":c#=CHR$(5)
20 CLS:PRINT"(x-a)²+(y-b)²=r²,(x1,y1)";
30 FORi=0 TO4
40 LOCATE0,1:PRINTc#:MID$(a$,1+2*i,(SGN(i-2.5)+1)/2+1);"=";a(i);"?";
:INPUT@42:a(i):LOCATE0,0
45 IFi=2 AND a(2)<0 THEN40
50 NEXT:LOCATE0,1:PRINTc#
60 w=(a(3)-a(0))²+(a(4)-a(1))²-a(2)²:IFw<0 THEN100 ELSEPRINTc#
:"1:length =":SQ R(w):w#=INPUT$(1,@):GOTO30
100 LOCATE0,1:PRINTc#:"not found":w#=INPUT$(1,@):GOTO20
500 IFERR=1 THENCLS:ONERRORGOTO0
510 IFERL=40 THENRESUME40
520 RESUME100
5 ONERRORGOTO500
10 MODE0:DIM:ERASEa:DIMa(4):a$="a b r x1y1":c#=CHR$(5)
20 CLS:PRINT"(x-a)²+(y-b)²=r²,(x1,y1)";
30 FORi=0 TO4
40 LOCATE0,1:PRINTc#:MID$(a$,1+2*i,(SGN(i-2.5)+1)/2+1);"=";a(i);"?";
:INPUT@42:a(i):LOCATE0,0
45 IFi=2 AND a(2)<0 THEN40
50 NEXT:LOCATE0,1:PRINTc#
60 w=(a(3)-a(0))²+(a(4)-a(1))²-a(2)²:IFw<0 THEN100 ELSEPRINTc#
:"1:length =":SQ R(w):w#=INPUT$(1,@):GOTO30
100 LOCATE0,1:PRINTc#:"not found":w#=INPUT$(1,@):GOTO20
500 IFERR=1 THENCLS:ONERRORGOTO0
510 IFERL=40 THENRESUME40
520 RESUME100

```

## 3.3.26 5570 Tangentialgleichung

## LIB 5570 Tangentialgleichung

```

5  ONERRORGOTO500
10  MODE8: DIM: ERASE a, b, c#: DIM a(4), b(7), c$(7): a$="a b r x1y1": c#=CHR$(5)
20  CLS: PRINT "(x-a)^2+(y-b)^2=r^2, (x1,y1)":
30  FOR i=0 TO4
40  LOCATE0,1: PRINT c#: MID$(a$, i*2+1, (SGN(i-2.5)+1)/2+1): "="; a(i): "?":
: INPUT@42: a(i): LOCATE0,0
50  IF i=2 AND a(2)<=0 THEN40
60  IF i>2 THEN IF ABS(a(i-3)-a(i))/ABS(a(2))>1e5 THEN30
70  NEXT: a=a(0): b=a(1): r=a(2): x=a(3): y=a(4): LOCATE0,1: PRINT c#: ".....":
75  IF (x-a)^2+(y-b)^2<r^2 THEN GOTO300
80  IF x=a THEN100 ELSE c=(y-b)/(a-x): d=r^2/(a-x): e=c^2+1
: f=c*(c*b+d)+b: g=(c*b+d)^2+b^2-r^2: l=f^2-e*g
90  IF l<0 THEN100 ELSE i=(f+SQ R l)/e: k=(f-SQ R l)/e: h=c*(i-b)-d+a: j=c*(k-b)-
d+a: GOTO200
100 IF y=b THEN300 ELSE c=(x-a)/(b-y): d=r^2/(b-y): e=c^2+1
: f=c*(c*a+d)+a: g=(c*a+d)^2+a^2-r^2: l=f^2-e*g
110 IF l<0 THEN300 ELSE h=(f+SQ R l)/e: j=(f-SQ R l)/e: i=c*(h-a)-d+b: k=c*(j-a)-d+b
200 IF ABS(h-j)+ABS(i-k)<0 THEN210
201 P$="(x1,y1)=(x2,y2)=(x3,y3), ": n=0: c$(0)="d"
202 IF b=y THEN P$=P$+"x=d": b(0)=x: GOTO400
203 IF x=a THEN P$=P$+"y=d": b(0)=y: GOTO400
204 n=1: c$(0)="c": c$(1)="d": P$=P$+"y=cx+d": b(0)=(x-a)/(b-y)
: b(1)=y-(x-a)/(b-y)*x: GOTO400
210 P$="(x2,y2), ": n=2: c$(0)="x2": c$(1)="y2": c$(2)="d"
220 b(0)=h: b(1)=i
230 IF x=h THEN P$=P$+"x=d": b(2)=x: GOTO250
232 IF y=i THEN P$=P$+"y=d": b(2)=y: GOTO250
234 n=3: P$=P$+"y=cx+d": c$(2)="c": c$(3)="d": b(2)=(i-y)/(h-x): b(3)=(y-i)/(h-x)*x+y
250 P$=P$+" : (x3,y3), "
260 n=n+1: c$(n)="x3": b(n)=j
270 n=n+1: c$(n)="y3": b(n)=k
275 n=n+1: c$(n)="f"
280 IF x=j THEN P$=P$+"x=f": b(n)=x: GOTO400
282 IF y=k THEN P$=P$+"y=f": b(n)=y: GOTO400
284 P$=P$+"y=cx+f": c$(n)="e": b(n)=(k-y)/(j-x): n=n+1: c$(n)="f"
: b(n)=(y-k)/(j-x)*x+y: GOTO400
300 LOCATE0,1: PRINT c#: "not found": w#=INPUT$(1,@): GOTO20
400 CLS: PRINT P$: PRINT: FOR z=0 TO n
410 PRINT c#: c$(z): " =": b(z):
420 w=ASC(INPUT$(1,@)): IF w=13 THEN480
430 IF w=30 AND z>0 THEN z=z-2: GOTO480
440 IF w=31 AND z<n THEN480
470 GOTO420
480 NEXT: GOTO20
500 IF ERR=1 THEN CLS: ONERRORGOTO0
510 IF ERR=40 THEN RESUME40
550 RESUME300

```

## 3.3.27 5600 Dreiecksfläche

## LIB 5600 Dreiecksfläche

```

5 ONERRORGOTO900
20 MODE0: DIM: a=0: b=0: c=0: h=0: d=0: c#=CHR$(5)
30 CLS: PRINT "Area (Triangle)": TAB(0): "1: ah/2 2: ab·sinθ/2 3: f(a,b,c)":
: LOCATE0,0: LOCATE0,1
40 w=ASC(INPUT$(1,0))-48: IF w<1 OR w>3 THEN40
50 ON w GOTO100,200,300
100 CLS: PRINT "ah/2 a:base h:height":
110 s#="a": z=a: GOSUB500: a=z
120 s#="h": z=h: GOSUB500: h=z
130 z=a*h/2: GOTO400
200 CLS: PRINT "ab·sinθ/2 a:b:side":
210 s#="a": z=a: GOSUB500: a=z
220 s#="b": z=b: GOSUB500: b=z
230 s#="θ": z=d: GOSUB500: d=z: IF SIND<0 THEN230
240 z=a*b*SIND/2: GOTO400
300 CLS: PRINT "f(s(s-a)(s-b)(s-c)), s=(a+b+c)/2":
320 s#="a": z=a: GOSUB500: a=z
330 s#="b": z=b: GOSUB500: b=z
340 s#="c": z=c: GOSUB500: c=z
350 s=(a+b+c)/2: z=SQR(s*(s-a)*(s-b)*(s-c))
400 IF z<0 THEN450
410 LOCATE0,1: PRINT c#: "Area =": z: z#=INPUT$(1,0): GOTO30
450 LOCATE0,1: PRINT c#: "not found": z#=INPUT$(1,0): GOTO30
500 LOCATE0,1: PRINT c#: s#: "=": z: "?" : INPUT@42: z: LOCATE0,0: IF z<0 THEN500 ELSE RETURN
900 IFERR=1 THEN CLS: ONERRORGOTO0
910 IFERR=500 THEN RESUME500
920 RESUME450

```

## 3.3.28 5605 Fläche eines Trapezoides

## LIB 5605 Fläche eines Trapezoides

```

5 ONERRORGOTO900
20 MODE0: DIM: a=0: b=0: h=0: c#=CHR$(5)
30 CLS: PRINT "(a+b)h/2 a:b:base h:height":
40 s#="a": z=a: GOSUB100: a=z
50 s#="b": z=b: GOSUB100: b=z
60 s#="h": z=h: GOSUB100: h=z
70 LOCATE0,1: PRINT c#: "Area =": (a+b)*h/2: z#=INPUT$(1,0): GOTO40
100 LOCATE0,1: PRINT c#: s#: "=": z: "?" : INPUT@42: z: LOCATE0,0: IF z<0 THEN100 ELSE RETURN
900 IFERR=1 THEN CLS: ONERRORGOTO0
910 IFERR=100 THEN RESUME100
920 LOCATE0,1: PRINT c#: "not found": z#=INPUT$(1,0): RESUME30

```



## 3.3.29 5610 Fläche eines Parallelogramms

## LIB 5610 Fläche eines Parallelogramms

```

5 ONERRORGOTO900
20 MODE0: DIM: a=0: b=0: c=0: h=0: c#=CHR$(5)
30 CLS: PRINT "Area (Parallelogram)"; TAB(0); "1: ah      2: ab·sinθ";
40 w=ASC(INPUT$(1,0))-48: IF w<1 OR w>2 THEN40
50 ON w GOTO100,200
100 CLS: PRINT "ah      a:base  h:height";
110 s$="a": z=a: GOSUB500: a=z
120 s$="h": z=h: GOSUB500: h=z
130 z=a*h: GOTO400
200 CLS: PRINT "ab·sinθ      a:b:side";
210 s$="a": z=a: GOSUB500: a=z
220 s$="b": z=b: GOSUB500: b=z
230 s$="θ": z=c: GOSUB500: c=z: IF SIN c<0 THEN230
240 z=a*b*SIN c
400 LOCATE0,1: PRINT c$: "Area ="; z: z#=INPUT$(1,0): GOTO30
500 LOCATE0,1: PRINT c$: s$: "="; z: "?": INPUT@42: z: LOCATE0,0: IF z<0 THEN500 ELSE RETURN
900 IFERR=1 THEN CLS: ONERRORGOTO0
910 IFERL=500 THEN RESUME500
920 LOCATE0,1: PRINT c$: "not found": z#=INPUT$(1,0): RESUME30

```

## 3.3.30 5615 Fläche eines Kreises

## LIB 5615 Fläche eines Kreises

```

5 ONERRORGOTO100
20 MODE0: DIM: r=0: c#=CHR$(5)
30 CLS: PRINT "πr²      r:radius";
50 LOCATE0,1: PRINT c$: "r="; r: "?": INPUT@42: r: LOCATE0,0: IF r<0 THEN50
70 LOCATE0,1: PRINT c$: "Area ="; PI*r*r: z#=INPUT$(1,0): GOTO50
100 IFERR=1 THEN CLS: ONERRORGOTO0
110 IFERL=50 THEN RESUME50
120 LOCATE0,1: PRINT c$: "not found": z#=INPUT$(1,0): RESUME30

```

## 3.3.31 5620 Fläche eines Sektors

## LIB 5620 Fläche eines Sektors

```

5 ONERRORGOTO900
20 MODE0: DIM: l=0: r=0: a=0: c#=CHR$(5): ANGLE0
30 CLS: PRINT "Area (sector)"; TAB(0); "1: lr/2      2: πr²θ/360";
40 w=ASC(INPUT$(1,0))-48: IF w<1 OR w>2 THEN40
50 ON w GOTO100,200
100 CLS: PRINT "lr/2  1:circular arc  r:radius";
110 s$="l": z=l: GOSUB500: l=z
120 s$="r": z=r: GOSUB500: r=z
130 z=l*r/2: GOTO400
200 CLS: PRINT "πr²θ/360      r:radius  θ:DEG";
220 s$="r": z=r: GOSUB500: r=z
230 s$="θ": z=a: GOSUB500: a=z: IF a>360 THEN230
240 z=PI*a*r^2/360
400 LOCATE0,1: PRINT c$: "Area ="; z: z#=INPUT$(1,0): GOTO30
500 LOCATE0,1: PRINT c$: s$: "="; z: "?": INPUT@42: z: LOCATE0,0: IF z<0 THEN500 ELSE RETURN
900 IFERR=1 THEN CLS: ONERRORGOTO0
910 IFERL=500 THEN RESUME500
920 LOCATE0,1: PRINT c$: "not found": z#=INPUT$(1,0): RESUME30

```

## 3.3.32 5625 Fläche eines Segments

## LIB 5625 Fläche eines Segments

```

5 ONERRORGOTO900
10 ANGLE1:MODE8:DIM:1=0:r=0:c#=CHR$(5):n#="(1-r^2*sin(1/r))/2"
50 m#="n#+ " l:arc":s#="l":z=1:GOSUB500:l=z
80 m#="n#+ " r:radius":s#="r":z=r:GOSUB500:IFz<1/2/PI THEN80 ELSEr=z
100 LOCATE0,1:PRINTc#:"Area =":(1*r-r^2*SIN(1/r))/2:z#=INPUT$(1,0):GOTO50
300 LOCATE0,1:PRINTc#:"not found":z#=INPUT$(1,0):GOTO50
500 CLS:PRINTm#:
510 LOCATE0,1:PRINTc#:s#:"=":z:"?":INPUT@42;z:LOCATE0,0
:IFz<0 THEN500 ELSERETURN
900 IFERR=1 THENCLS:ONERRORGOTO0
910 IFERL=510 THENRESUME510
920 RESUME300

```

## 3.3.33 5630 Fläche einer Ellipse

## LIB 5630 Fläche einer Ellipse

```

5 ONERRORGOTO900
20 MODE8:DIM:a=0:b=0:c#=CHR$(5)
30 CLS:PRINT"πab a:b:radius":
40 s#="a":z=a:GOSUB500:a=z
50 s#="b":z=b:GOSUB500:b=z
60 LOCATE0,1:PRINTc#:"Area =":PI*a*b:z#=INPUT$(1,0):GOTO40
500 LOCATE0,1:PRINTc#:s#:"=":z:"?":INPUT@42;z:LOCATE0,0:IFz<0 THEN500 ELSERETURN
900 IFERR=1 THENCLS:ONERRORGOTO0
910 IFERL=500 THENRESUME500
920 LOCATE0,1:PRINTc#:"not found":z#=INPUT$(1,0):RESUME30

```

## 3.3.34 5635 Fläche eines Polygons

## LIB 5635 Fläche eines Polygons

```

5 ONERRORGOTO900
10 MODE8:DIM:ERASEa,a#:DIMa(2),a#(2):a#(0)="r:inside":a#(1)="R:outside"
:a#(2)="l:side"
20 ANGLE0:n=0:c#=CHR$(5)
30 CLS:PRINT"Area (polygon)":TAB(0):"1:n,r→A 2:n,R→A 3:n,l→A":
31 f=ASC(INPUT$(1,0))-48:IFf<1 ORf>3 THEN31
32 m#="Polygon "
35 s#="n:number":z=n:GOSUB500:n=z:IFn<3 ORn>=1e7 ORFRACn<>0 THEN35
36 m#="m#+(n"+MID$(STR$(n),2)+)" "
37 s#="a#(f-1):z=a(f-1):GOSUB500:a(f-1)=z
40 ONf GOTO41,42,43
41 s=n*a(0)^2*TAN(180/n):GOTO50
42 s=n*a(1)^2*SIN(360/n)/2:GOTO50
43 s=n*a(2)^2/TAN(180/n)/4
50 LOCATE0,1:PRINTc#:"Area =":ABSs:z#=INPUT$(1,0):GOTO30
500 CLS:PRINTm#:s#:
510 LOCATE0,1:PRINTc#:LEFT$(s#,1):"=":z:"?":INPUT@42;z:LOCATE0,0
:IFz<0 THEN510 ELSERETURN
900 IFERR=1 THENCLS:ONERRORGOTO0
910 IFERL=510 THENRESUME510
920 LOCATE0,1:PRINTc#:"not found":z#=INPUT$(1,0):RESUME30

```

## 3.3.35 5650 Flächeninhalt einer Kugel

## LIB 5650 Flächeninhalt einer Kugel

```

5 ONERRORGOTO900
20 MODE8: DIM: r=0: c#=CHR$(5)
30 CLS: PRINT "4πr²          r:radius";
40 LOCATE0,1: PRINT c#: "r="; r; "?": INPUT@42: r: LOCATE0,0: IF r<0 THEN40
50 LOCATE0,1: PRINT c#: "Surface =" ; 4*PI*r^2; : z#=INPUT$(1,@): GOTO40
900 IFERR=1 THENCLS: ONERRORGOTO0
910 IFERL=40 THENRESUME40
920 LOCATE0,1: PRINT c#: "not found": z#=INPUT$(1,@): RESUME30

```

## 3.3.36 5655 Flächeninhalt einer Kugelzone

## LIB 5655 Flächeninhalt einer Kugelzone

```

5 ONERRORGOTO900
20 MODE8: DIM: a=0: b=0: r=0: h=0: c#=CHR$(5)
30 CLS: PRINT "2πrh+π(a²+b²)  h:height";
50 s#="h": z=h: GOSUB500: h=z
51 CLS: PRINT "2πrh+π(a²+b²)  a:b:r:radius";
52 s#="a": z=a: GOSUB500: a=z
53 s#="b": z=b: GOSUB500: b=z
54 s#="r": z=r: GOSUB500: r=z
60 LOCATE0,1: PRINT c#: "Surface =" ; PI*(2*r*h+a^2+b^2); : z#=INPUT$(1,@): GOTO30
500 LOCATE0,1: PRINT c#: s#: "="; z; "?": INPUT@42: z: LOCATE0,0: IF z<0 THEN500 ELSE RETURN
900 IFERR=1 THENCLS: ONERRORGOTO0
910 IFERL=500 THENRESUME500
920 LOCATE0,1: PRINT c#: "not found": z#=INPUT$(1,@): RESUME30

```

## 3.3.37 5660 Flächeninhalt eines Kugelsektors

## LIB 5660 Flächeninhalt eines Kugelsektors

```

5 ONERRORGOTO900
20 MODE8: DIM: r=0: h=0: c#=CHR$(5)
30 CLS: PRINT "2πrh+πar, a=√(h(2r-h))  r:radius";
40 s#="r": z=r: GOSUB500: r=z
45 CLS: PRINT "2πrh+πar, a=√(h(2r-h))  h:height";
50 s#="h": z=h: GOSUB500: h=z
60 a=h*(2*r-h): IF a<0 THEN300 ELSE a=SQ R a
70 LOCATE0,1: PRINT c#: "Surface =" ; PI*r*(2*h+a); : z#=INPUT$(1,@): GOTO30
300 LOCATE0,1: PRINT c#: "not found": z#=INPUT$(1,@): GOTO30
500 LOCATE0,1: PRINT c#: s#: "="; z; "?": INPUT@42: z: LOCATE0,0: IF z<0 THEN500 ELSE RETURN
900 IFERR=1 THENCLS: ONERRORGOTO0
910 IFERL=500 THENRESUME500
920 RESUME300

```

## 3.3.38 5665 Oberfläche eines Kreiszyllinders

## LIB 5665 Oberfläche eines Kreiszyllinders

```

5 ONERRORGOTO900
20 MODE8: DIM: r=0: h=0: c#=CHR$(5)
30 CLS: PRINT "2πrh+2πr²    r:radius  h:height";
40 s#="r": z=r: GOSUB500: r=z
50 s#="h": z=h: GOSUB500: h=z
70 LOCATE0,1: PRINT c#: "Surface ="; 2*PI*r*(r+h);: z#=INPUT$(1,0): GOTO40
500 LOCATE0,1: PRINT c#: s#: "="; z: "?";: INPUT@42; z: LOCATE0,0: IF z<0 THEN500 ELSE RETURN
900 IFERR=1 THEN CLS: ONERRORGOTO0
910 IFERR=500 THEN RESUME500
920 LOCATE0,1: PRINT c#: "not found";: z#=INPUT$(1,0): RESUME30

```

## 3.3.39 5670 Oberfläche eines Kreiskegels

## LIB 5670 Oberfläche eines Kreiskegels

```

5 ONERRORGOTO900
20 MODE8: DIM: r=0: h=0: c#=CHR$(5)
30 CLS: PRINT "πr√(r²+h²)+πr²    r:radius";
40 s#="r": z=r: GOSUB500: r=z
45 CLS: PRINT "πr√(r²+h²)+πr²    h:height";
50 s#="h": z=h: GOSUB500: h=z
70 LOCATE0,1: PRINT c#: "Surface ="; PI*r*(SQR(r*r+h*h)+r);: z#=INPUT$(1,0): GOTO30
500 LOCATE0,1: PRINT c#: s#: "="; z: "?";: INPUT@42; z: LOCATE0,0: IF z<0 THEN500 ELSE RETURN
900 IFERR=1 THEN CLS: ONERRORGOTO0
910 IFERR=500 THEN RESUME500
920 LOCATE0,1: PRINT c#: "not found";: z#=INPUT$(1,0): RESUME30

```

## 3.3.40 5675 Flächeninhalt eines Kreiskegelstumpfes

## LIB 5675 Flächeninhalt eines Kreiskegelstumpfes

```

5 ONERRORGOTO900
20 MODE8: DIM: r=0: s=0: h=0: c#=CHR$(5)
30 CLS: PRINT "π(R+r)√(h²+(R-r)²)+π(R²+r²)";
40 s#="r": z=r: GOSUB500: r=z
50 s#="R": z=s: GOSUB500: s=z
60 s#="h": z=h: GOSUB500: h=z
70 LOCATE0,1: PRINT c#: "Surface ="; PI*(r+s)*SQR(h*h+(s-r)^2)+PI*(r^2+s^2);
: z#=INPUT$(1,0): GOTO40
500 LOCATE0,1: PRINT c#: s#: "="; z: "?";: INPUT@42; z: LOCATE0,0: IF z<0 THEN500 ELSE RETURN
900 IFERR=1 THEN CLS: ONERRORGOTO0
910 IFERR=500 THEN RESUME500
920 LOCATE0,1: PRINT c#: "not found";: z#=INPUT$(1,0): RESUME30

```

## 3.3.41 5700 Rauminhalt einer Kugel

## LIB 5700 Rauminhalt einer Kugel

```

5 ONERRORGOTO900
20 MODE8: DIM: r=0: c#=CHR$(5)
30 CLS: PRINT "4πr³/3          r:radius";
40 LOCATE0,1: PRINT c#; "r="; r; "?";: INPUT@42: r: LOCATE0,0: IFR<0 THEN40
50 LOCATE0,1: PRINT c#; "Volume ="; 4*PI*r³/3;: z#=INPUT$(1,@): GOTO40
900 IFERR=1 THENCLS: ONERRORGOTO0
910 IFERL=40 THENRESUME40
930 LOCATE0,1: PRINT c#; "not found";: z#=INPUT$(1,@): RESUME30

```

## 3.3.42 5705 Rauminhalt einer Kugelzone

## LIB 5705 Rauminhalt einer Kugelzone

```

5 ONERRORGOTO900
20 MODE8: DIM: a=0: b=0: h=0: c#=CHR$(5)
30 CLS: PRINT "πh(3a²+3b²+h²)/6      a:b:radius";
40 s#="a": z=a: GOSUB500: a=z
41 s#="b": z=b: GOSUB500: b=z
42 CLS: PRINT "πh(3a²+3b²+h²)/6      h:heisht";
43 s#="h": z=h: GOSUB500: h=z
50 LOCATE0,1: PRINT c#; "Volume ="; PI*h*(3*a²+3*b²+h²)/6;: z#=INPUT$(1,@): GOTO30
500 LOCATE0,1: PRINT c#; s#; "="; z; "?";: INPUT@42: z: LOCATE0,0: IFz<0 THEN500 ELSERETURN
900 IFERR=1 THENCLS: ONERRORGOTO0
910 IFERL=500 THENRESUME500
930 LOCATE0,1: PRINT c#; "not found";: z#=INPUT$(1,@): RESUME30

```

## 3.3.43 5710 Rauminhalt eines Kugelsektors

## LIB 5710 Rauminhalt eines Kugelsektors

```

5 ONERRORGOTO900
20 MODE8: DIM: r=0: h=0: c#=CHR$(5)
30 CLS: PRINT "2πr²h/3          r:radius h:heisht";
40 s#="r": z=r: GOSUB500: r=z
41 s#="h": z=h: GOSUB500: h=z
50 LOCATE0,1: PRINT c#; "Volume ="; 2*PI*r²*h/3;: z#=INPUT$(1,@): GOTO40
500 LOCATE0,1: PRINT c#; s#; "="; z; "?";: INPUT@42: z: LOCATE0,0: IFz<0 THEN500 ELSERETURN
900 IFERR=1 THENCLS: ONERRORGOTO0
910 IFERL=500 THENRESUME500
930 LOCATE0,1: PRINT c#; "not found";: z#=INPUT$(1,@): RESUME30

```

## 3.3.44 5715 Rauminhalt eines Kreiszylinders

## LIB 5715 Rauminhalt eines Kreiszylinders

```

5 ONERRORGOTO900
20 MODE8:DIM:r=0:h=0:c#=CHR$(5)
30 CLS:PRINT"πr²h      r:radius  h:height";
40 s#="r":z=r:GOSUB500:r=z
41 s#="h":z=h:GOSUB500:h=z
50 LOCATE0,1:PRINTc#:"Volume =";PI*r^2*h;:z#=INPUT$(1,0):GOTO40
500 LOCATE0,1:PRINTc#;s#;"=";z;"?";:INPUT@42;z:LOCATE0,0:IFz<0 THEN500 ELSERETURN
900 IFERR=1 THENCLS:ONERRORGOTO0
910 IFERL=500 THENRESUME500
930 LOCATE0,1:PRINTc#;"not found";:z#=INPUT$(1,0):RESUME30

```

## 3.3.45 5720 Rauminhalt eines Kreiskegels

## LIB 5720 Rauminhalt eines Kreiskegels

```

5 ONERRORGOTO900
20 MODE8:DIM:r=0:h=0:c#=CHR$(5)
30 CLS:PRINT"πr²h/3      r:radius  h:height";
40 s#="r":z=r:GOSUB500:r=z
41 s#="h":z=h:GOSUB500:h=z
50 LOCATE0,1:PRINTc#:"Volume =";PI*r^2*h/3;:z#=INPUT$(1,0):GOTO40
500 LOCATE0,1:PRINTc#;s#;"=";z;"?";:INPUT@42;z:LOCATE0,0:IFz<0 THEN500 ELSERETURN
900 IFERR=1 THENCLS:ONERRORGOTO0
910 IFERL=500 THENRESUME500
930 LOCATE0,1:PRINTc#;"not found";:z#=INPUT$(1,0):RESUME30

```

## 3.3.46 5725 Rauminhalt eines Kreiskegelstumpfes

## LIB 5725 Rauminhalt eines Kreiskegelstumpfes

```

5 ONERRORGOTO900
20 MODE8:DIM:r=0:s=0:h=0:c#=CHR$(5)
30 CLS:PRINT"πh(r²+rR+R²)/3      r:R:radius";
40 s#="r":z=r:GOSUB500:r=z
41 s#="R":z=s:GOSUB500:s=z
42 CLS:PRINT"πh(r²+rR+R²)/3      h:height";
43 s#="h":z=h:GOSUB500:h=z
50 LOCATE0,1:PRINTc#:"Volume =";PI*h*(s^2+s*r+r^2)/3;:z#=INPUT$(1,0):GOTO30
500 LOCATE0,1:PRINTc#;s#;"=";z;"?";:INPUT@42;z:LOCATE0,0:IFz<0 THEN500 ELSERETURN
900 IFERR=1 THENCLS:ONERRORGOTO0
910 IFERL=500 THENRESUME500
930 LOCATE0,1:PRINTc#;"not found";:z#=INPUT$(1,0):RESUME30

```

## 3.3.47 5730 Rauminhalt eines Keils

## LIB 5730 Rauminhalt eines Keils

```

5 ONERRORGOTO900
20 MODE8: DIM: a=0: b=0: c=0: h=0: c#=CHR$(5)
30 CLS: PRINT "bh(2a+c)/6 a:b:c:edae h:height";
40 s#="a": z=a: GOSUB500: a=z
41 s#="b": z=b: GOSUB500: b=z
42 s#="c": z=c: GOSUB500: c=z
43 s#="h": z=h: GOSUB500: h=z
50 LOCATE0,1: PRINTc#: "Volume ="; b*h*(2*a+c)/6; : z#=INPUT$(1,0): GOTO40
500 LOCATE0,1: PRINTc#: s#: "="; z: "?"; INPUT@42: z: LOCATE0,0: IF z<0 THEN500 ELSE RETURN
900 IFERR=1 THEN CLS: ONERRORGOTO0
910 IFERR=500 THEN RESUME500
930 LOCATE0,1: PRINTc#: "not found"; : z#=INPUT$(1,0): RESUME30

```

## 3.3.48 5735 Rauminhalt einer Pyramide

## LIB 5735 Rauminhalt einer Pyramide

```

5 ONERRORGOTO900
20 MODE8: DIM: a=0: b=0: h=0: c#=CHR$(5)
30 CLS: PRINT "abh/3 a:b:edae h:height";
40 s#="a": z=a: GOSUB500: a=z
41 s#="b": z=b: GOSUB500: b=z
42 s#="h": z=h: GOSUB500: h=z
50 LOCATE0,1: PRINTc#: "Volume ="; a*b*h/3; : z#=INPUT$(1,0): GOTO40
500 LOCATE0,1: PRINTc#: s#: "="; z: "?"; INPUT@42: z: LOCATE0,0: IF z<0 THEN500 ELSE RETURN
900 IFERR=1 THEN CLS: ONERRORGOTO0
910 IFERR=500 THEN RESUME500
920 LOCATE0,1: PRINTc#: "not found"; : z#=INPUT$(1,0): RESUME30

```

## 3.3.49 5740 Rauminhalt eines Pyramidenstumpfes

## LIB 5740 Rauminhalt eines Pyramidenstumpfes

```

5 ONERRORGOTO900
20 MODE8: DIM: a=0: b=0: c=0: d=0: h=0: c#=CHR$(5)
30 CLS: PRINT "h(ab+cd+sqrt(abcd))/3 a:b:c:d:edae";
40 s#="a": z=a: GOSUB500: a=z
41 s#="b": z=b: GOSUB500: b=z
42 s#="c": z=c: GOSUB500: c=z
43 s#="d": z=d: GOSUB500: d=z
44 CLS: PRINT "h(ab+cd+sqrt(abcd))/3 h:height";
45 s#="h": z=h: GOSUB500: h=z
50 w=a*b*c*d: LOCATE0,1: PRINTc#: "Volume ="; h*(a*b+c*d+SQ R w)/3;
: z#=INPUT$(1,0): GOTO30
500 LOCATE0,1: PRINTc#: s#: "="; z: "?"; INPUT@42: z: LOCATE0,0: IF z<0 THEN500 ELSE RETURN
900 IFERR=1 THEN CLS: ONERRORGOTO0
910 IFERR=500 THEN RESUME500
930 LOCATE0,1: PRINTc#: "not found"; : z#=INPUT$(1,0): RESUME30

```

## 3.3.50 5745 Rauminhalt eines Ellipsoiden

## LIB 5745 Rauminhalt eines Ellipsoiden

```

5  ONERRORGOTO900
20  MODE0: DIM: a=0: b=0: c=0: c#=CHR$(5)
30  CLS: PRINT "4πabc/3      a:b:c:radius":
40  s#="a": z=a: GOSUB500: a=z
41  s#="b": z=b: GOSUB500: b=z
42  s#="c": z=c: GOSUB500: c=z
50  LOCATE0,1: PRINT c#: "Volume =": 4*PI*a*b*c/3: z#=INPUT$(1,0): GOTO40
500  LOCATE0,1: PRINT c#: s#: "=": z: "?": INPUT@42: z: LOCATE0,0: IF z<0 THEN500 ELSE RETURN
900  IFERR=1 THEN CLS: ONERRORGOTO0
910  IFERR=500 THEN RESUME500
930  LOCATE0,1: PRINT c#: "not found": z#=INPUT$(1,0): RESUME30

```

3.3.51 5750 Einbeschriebener Kreis und umbeschriebener Kreis  
einers PolygonsLIB 5750 Einbeschriebener Kreis und umbeschriebener Kreis eines  
Polygons

```

5  ONERRORGOTO900
10  MODE0: DIM: ERASE a, a#: DIM a(2), a#(2): a#(0)="n:inside": a#(1)="R:outside"
: a#(2)="l:side"
20  s=0: n=3: ANGLE0: c#=CHR$(5): m#="Polygon (n,R,l)  "
40  s#="A:area": z=s: GOSUB500: s=z
41  s#="n:number": z=n: GOSUB500: n=z: IF n<3 OR FRAC n<>0 THEN41
45  CLS: PRINT m#: PRINT
50  a(0)=SQ R(s/n/TAN(180/n))
51  a(1)=SQ R(2*s/n/SIN(360/n))
52  a(2)=SQ R(ABS(4*s/n*TAN(180/n)))
100  FOR i=0 TO2: PRINT c#: a#(i) " =": a(i):
110  w=ASC(INPUT$(1,0)): IF w=13 THEN170
120  IF w=30 AND i>0 THEN i=i-2: GOTO170
130  IF w=31 AND i<2 THEN170
160  GOTO110
170  NEXT: GOTO40
500  CLS: PRINT m#: s#:
510  LOCATE0,1: PRINT c#: LEFT$(s#,1): "=": z: "?": INPUT@42: z: LOCATE0,0
: IF z<0 THEN510 ELSE RETURN
900  IFERR=1 THEN CLS: ONERRORGOTO0
910  IFERR=510 THEN RESUME510
920  LOCATE0,1: PRINT c#: "not found": z#=INPUT$(1,0): RESUME40

```



## 3.3.52 5760 Regelmäßiger Polyeder

## LIB 5760 Regelmäßiger Polyeder

```

5 ONERRORGOTO900
10 MODE8:DIM:a,a#:DIMa(4),a$(4):c#=CHR$(5)
15 RESTORE800:FORi=0 TO4:READa$(i):NEXT
20 CLS:PRINT"Select number of face":TAB(0):"1:4f 2:6f 3:8f 4:12f 5:20f":
30 f=ASC(INPUT$(1,0))-48:IFf<1 OR f>5 THEN30
40 CLS:PRINT"Select input data":TAB(0):"1:a 2:r 3:R 4:S 5:U":
50 k=ASC(INPUT$(1,0))-48:IFk<1 OR k>5 THEN50
60 RESTORE(800+f*10):READw$,n$,e$,t$,b,c,x,y:f=f-1:k=k-1
90 CLS:PRINTw$:"hedron":TAB(18):a$(k):
100 LOCATE0,1:PRINTc$:LEFT$(a$(k),1):"=":a(k):"?":INPUT@42:a(k):LOCATE0,0
105 CLS:PRINTw$:"hedron":TAB(18):n$:",":e$:"e,":t$:"v":PRINT
110 ONk+1 GOTO200,111,112,113,114
111 a(0)=a(1)/x:GOTO200
112 a(0)=a(2)/y:GOTO200
113 a(0)=SQR(ABS(a(3)/b):GOTO200
114 a(0)=CUR(a(4)/c)
115 IFA(k)<0 THEN100
200 a(1)=a(0)*x:a(2)=a(0)*y:a(3)=b*a(0)^2:a(4)=c*a(0)^3:m=SGN(-k)+1:n=SGN(4-k)+3
300 FORi=m TOn:IFI=k THEN380
310 PRINTc$:a$(i):"=":a(i):
320 w=ASC(INPUT$(1,0)):IFw=13 THEN380
330 IFw=30 ANDi>m THENIFI=k+1 THENi=i-3:GOTO380 ELSEi=i-2:GOTO380
340 IFw=31 ANDi<n THEN380
370 GOTO320
380 NEXT:GOTO20
800 DATAa:edse,r:inside,R:outside,S:surface,U:volume
810 DATA4(tetra,3,6,4,1.732050807568,.1178511302,.2041241452319,.6123724357
820 DATA6(hexa,4,12,8,6,1,.5,.866025403784
830 DATA8(octa,3,12,6,3,4.64101615136,.471404520791,.4082482904638,.7071067811865
840 DATA12(dodeca,5,30,20,20,6.4572880706,7.663118960622,1.113516364411
,1.401258538443
850 DATA20(icosa,3,30,12,8,6.6025403784,2.181694990624,.755761314076
,.9510565162947
900 IFERR=1 THENCLS:ONERRORGOTO0
910 IFERL=100 THENRESUME100
920 LOCATE0,1:PRINTc$:"not found":z#=INPUT$(1,0):RESUME20

```

## 3.4 Wissenschaftliche Formeln

### 3.4.1 5800 Zerlegung in Faktoren

#### LIB 5800 Zerlegung in Faktoren

```

5  GOTO10010
10  DATAa2-b2, (a+b)(a-b), 10, 20
20  DATAa3±b3, (a±b)(a2∓ab+b2), 10, 30
30  DATAa4-b4, (a-b)(a+b)(a2+b2), 20, 40
40  DATAa4+b4, (a2+√2ab+b2)(a2-√2ab+b2), 30, 50
50  DATAa2±2ab+b2, (a±b)2, 40, 60
60  DATAa3±3a2b+3ab2±b3, (a±b)3, 50, 70
70  DATA(a±b)2∓4ab, (a∓b)2, 60, 80
80  DATAa2+b2+c2+2bc+2ca+2ab, (a+b+c)2, 70, 90
90  DATAa4+a2b2+b4, (a2+ab+b2)(a2-ab+b2), 80, 100
100 DATAa3+b3+c3-3abc, (a+b+c)(a2+b2+c2-bc-ca-ab), 90, 110
110 DATA(ac-bd)2+(ad+bc)2, (a2+b2)(c2+d2), 100, 120
120 DATA(ac+bd)2+(ad-bc)2, (a2+b2)(c2+d2), 110, 130
130 DATA(ac+bd)2-(ad+bc)2, (a2-b2)(c2-d2), 120, 140
140 DATA(ac-bd)2-(ad-bc)2, (a2-b2)(c2-d2), 130, 150
150 DATAa2(b-c)+b2(c-a)+c2(a-b), -(b-c)(c-a)(a-b), 140, 160
160 DATA(b-c)3+(c-a)3+(a-b)3, 3(b-c)(c-a)(a-b), 150, 170
170 DATAa4+b4+c4-2b2c2-2c2a2-2a2b2, (a+b+c)(b-c-a)(c-a-b)(a-b-c), 160, 180
180 DATAx2+(a+b)x+ab, (x+a)(x+b), 170, 190
190 DATAx3+(a+b+c)x2+(bc+ca+ab)x+abc, (x+a)(x+b)(x+c), 180, 200
200 DATAa2-b2-c2-2bc, (a+b+c)(a-b-c), 190, 210
210 DATA(a+b+c)(bc+ca+ab)-abc, (b+c)(c+a)(a+b), 200, 220
220 DATA(a+b+c)3-(a3+b3+c3), 3(b+c)(c+a)(a+b), 210, 230
230 DATAa3(b-c)+b3(c-a)+c3(a-b), -(b-c)(c-a)(a-b)(a+b+c), 220, 230
10010 MODE8: DIM: x=10: y=230: r=x
10020 DEFCHR$(255)="A6AAAA9280"
10110 RESTORE(r)
10120 READm$, n$, b, c
10130 l$="["+MID$(STR$(r/10), 2)+" ]": l=32-LEN(l$)
10140 CLS: PRINTm$; TAB(1); l$; "="; n$;
10150 z=ASC(INPUT$(1, @))
10170 IFz=29 ANDr<>x THENr=x: GOTO10110
10180 IFz=28 ANDr<>y THENr=y: GOTO10110
10190 IFz=30 ANDr<>b THENr=b: GOTO10110
10200 IF(z=31 ORz=13) ANDr<>c THENr=c: GOTO10110
10280 GOTO10150

```

## 3.4.2 5810 Trigonometrische Funktionen

## LIB 5810 Trigonometrische Funktionen

```

5  GOTO10010
10 DATA sin²θ+cos²θ,1,10,20
20 DATA 1+tan²θ,sec²θ,10,30
30 DATA 1+cot²θ,cosec²θ,20,40
40 DATA sin(α±β),sinα·cosβ±cosα·sinβ,30,50
50 DATA cos(α±β),cosα·cosβ∓sinα·sinβ,40,60
60 DATA tan(α±β),(tanα±tanβ)/(1∓tanα·tanβ),50,70
70 DATA cot(α±β),(cotα·cotβ±1)/(cotβ±cotα),60,80
80 DATA sin2θ,2sinθ·cosθ,70,90
90 DATA cos2θ,cos²θ-sin²θ,80,100
100 DATA cos2θ,1-2sin²θ,90,110
110 DATA cos2θ,2cos²θ-1,100,120
120 DATA tan2θ,2tanθ/(1-tan²θ),110,130
130 DATA sin(θ/2),±√((1-cosθ)/2),120,140
140 DATA cos(θ/2),±√((1+cosθ)/2),130,150
150 DATA tan(θ/2),±√((1-cosθ)/(1+cosθ)),140,160
160 DATA tan(θ/2),(1-cosθ)/sinθ,150,170
170 DATA tan(θ/2),sinθ/(1+cosθ),160,180
180 DATA tan(θ/2),cosecθ-cotθ,170,190
190 DATA cot(θ/2),±√((1+cosθ)/(1-cosθ)),180,200
200 DATA cot(θ/2),sinθ/(1-cosθ),190,210
210 DATA cot(θ/2),(1+cosθ)/sinθ,200,220
220 DATA cot(θ/2),cosecθ+cotθ,210,230
230 DATA sin3θ,3sinθ-4sin³θ,220,240
240 DATA cos3θ,4cos³θ-3cosθ,230,250
250 DATA tan3θ,(3tanθ-tan³θ)/(1-3tan²θ),240,260
260 DATA 2sinα·cosβ,sin(α+β)+sin(α-β),250,270
270 DATA 2cosα·sinβ,sin(α+β)-sin(α-β),260,280
280 DATA 2cosα·cosβ,cos(α+β)+cos(α-β),270,290
290 DATA 2sinα·sinβ,-(cos(α+β)-cos(α-β)),280,300
300 DATA sinα+sinβ,2sin((α+β)/2)·cos((α-β)/2),290,310
310 DATA sinα-sinβ,2cos((α+β)/2)·sin((α-β)/2),300,320
320 DATA cosα+cosβ,2cos((α+β)/2)·cos((α-β)/2),310,330
330 DATA cosα-cosβ,-2sin((α+β)/2)·sin((α-β)/2),320,340
340 DATA tan(45°±(θ/2)),secθ±tanθ,330,350
350 DATA tan(45°±(θ/2)),(1±sinθ)/cosθ,340,360
360 DATA tan(45°±(θ/2)),cot(45°∓(θ/2)),350,370
370 DATA tan(45°+θ),(1+tanθ)/(1-tanθ),360,380
380 DATA cot(45°-θ),(1+cotθ)/(1-cotθ),370,380
10010 MODE8: DIM: x=10: y=380: r=x
10110 RESTORE(r)
10120 READm#,n#,b,c
10130 l$="["+MID$(STR$(r/10),2)+"]":l=32-LEN(l$)
10140 CLS: PRINTm#: TAB(1): l$:"=":n#;
10150 z=ASC(INPUT$(1,q))
10170 IFz=29 ANDr<>x THENr=x:GOTO10110
10180 IFz=28 ANDr<>y THENr=y:GOTO10110
10190 IFz=30 ANDr<>b THENr=b:GOTO10110
10200 IF(z=31 ORz=13) ANDr<>c THENr=c:GOTO10110
10280 GOTO10150

```

## 3.4.3 5820 Differenziale

## LIB 5820 Differenziale

```

5  GOTO10010
10  DATA 0, 10, 20
20  DATA x^n, nx^{n-1}, 10, 30
30  DATA x, 1, 20, 40
40  DATA 1/x, -1/x^2, 30, 50
50  DATA Γ^m, 1/(2Γ^m), 40, 60
60  DATA a^x, a^x loga, 50, 70
70  DATA e^x, e^x, 60, 80
80  DATA e^{nx}, ne^{nx}, 70, 90
90  DATA logx, 1/x, 80, 100
100 DATA x^x, x^x(logx+1), 90, 110
110 DATA sinx, cosx, 100, 120
120 DATA cosx, -sinx, 110, 130
130 DATA tanx, sec^2 x, 120, 140
140 DATA cotx, -cosec^2 x, 130, 150
150 DATA secx, secx·tanx, 140, 160
160 DATA cosecx, -cosecx·cotx, 150, 170
170 DATA sinax, a·cosax, 160, 180
180 DATA cosax, -a·sinax, 170, 190
190 DATA tanax, a·sec^2 ax, 180, 200
200 DATA cotax, -a·cosec^2 ax, 190, 210
210 DATA sin^2 x, 1/√(1-x^2) (≡y≡π/2), 200, 220
220 DATA cos^2 x, -1/√(1-x^2) (0<y<π), 210, 230
230 DATA tan^2 x, 1/(1+x^2) (≡y≡π/2), 220, 240
240 DATA cot^2 x, -1/(1+x^2) (≡y≡π/2), 230, 250
250 DATA sec^2 x, "1/(x√(x^2-1)) (0<y<π, x^2>1)", 240, 260
260 DATA cosec^2 x, "1/(x√(x^2-1)) (≡y≡π/2, x^2>1)", 250, 270
270 DATA sinhx, coshx, 260, 280
280 DATA coshx, sinhx, 270, 290
290 DATA tanhx, sech^2 x, 280, 300
300 DATA cothx, -cosech^2 x, 290, 310
310 DATA sechx, -sechx·tanhx, 300, 320
320 DATA cosechx, -cosechx·cothx, 310, 330
330 DATA sinhr^2 x, 1/√(1+x^2), 320, 340
340 DATA coshr^2 x, "1/√(x^2-1) (y>0, x^2>1)", 330, 350
350 DATA tanhr^2 x, 1/(1-x^2) (x^2<1), 340, 360
360 DATA cothr^2 x, 1/(1-x^2) (x^2>1), 350, 370
370 DATA sechr^2 x, -1/(x√(1-x^2)) (0<x<1), 360, 380
380 DATA cosechr^2 x, -1/(x√(x^2+1)), 370, 380
10010 MODE8: DIM: x=10: y=380: r=x
10020 DEFCHR$(255)="A2A49C92A2"
10030 DEFCHR$(254)="0000FE0000"
10110 RESTORE(r)
10120 READ m$, n$, b, c
10130 l$="[" + MID$(STR$(r/10), 2) + "]: l=32-LEN(l$)
10140 CLS: PRINT "y =" : m$: TAB(1): l$: "y": CHR$(131): "=" : n$:
10150 z=ASC(INPUT$(1, @))
10170 IF z=29 AND r<>x THEN r=x: GOTO10110
10180 IF z=28 AND r<>y THEN r=y: GOTO10110
10190 IF z=30 AND r<>b THEN r=b: GOTO10110
10200 IF (z=31 OR z=13) AND r<>c THEN r=c: GOTO10110
10280 GOTO10150

```

## 3.4.4 5830 Integrationen

## LIB 5830 Integrationen

```

5 GOTO10010
10 DATA x+C, 10, 20
20 DATA x^n, x^{n+1}/(n+1)+C (n+1≠0), 10, 30
30 DATA 1/x, log|x|+C, 20, 40
40 DATA 1/(x±a), log|x±a|+C, 30, 50
50 DATA e^x, e^x+C, 40, 60
60 DATA e^{ax}, e^{ax}/a+C, 50, 70
70 DATA a^x, "a^x/lna+C (a>0, a≠1)", 60, 80
80 DATA a^{nx}, "a^{nx}/(n·lna)+C (a>0, a≠1)", 70, 90
90 DATA log x, x(log x-1)+C, 80, 100
100 DATA x e^{ax}, e^{ax}/a - (nx-1)/a^2 + C, 90, 110
110 DATA sin x, -cos x+C, 100, 120
120 DATA sin ax, -1/a·cos ax+C, 110, 130
130 DATA cos x, sin x+C, 120, 140
140 DATA cos ax, 1/a·sin ax+C, 130, 150
150 DATA tan x, -log|cos x|+C, 140, 160
160 DATA cot x, log|sin x|+C, 150, 170
170 DATA sin^2 x, x/2 - sin 2x/4 + C, 160, 180
180 DATA cos^2 x, x/2 + sin 2x/4 + C, 170, 190
190 DATA sec^2 ax, 1/a·tan ax+C, 180, 200
200 DATA cosec^2 ax, -1/a·cot ax+C, 190, 210
210 DATA 1/sin x, log|tan(x/2)|+C, 200, 220
220 DATA 1/cos x, log|tan((π/4)+(x/2))|+C, 210, 230
230 DATA e^{nx} sin bx, e^{nx} (n·sin bx - b·cos bx) / (n^2 + b^2) + C, 220, 240
240 DATA e^{nx} cos bx, e^{nx} (n·cos bx + b·sin bx) / (n^2 + b^2) + C, 230, 250
250 DATA sin^{-1} x, x sin^{-1} x + √(1-x^2) + C, 240, 260
260 DATA cos^{-1} x, x cos^{-1} x - √(1-x^2) + C, 250, 270
270 DATA sinh x, cosh x+C, 260, 280
280 DATA cosh x, sinh x+C, 270, 290
290 DATA tanh x, log|cosh x|+C, 280, 300
300 DATA 1/√(a^2-x^2), sin^{-1}(x/a)+C (|x|<a), 290, 310
310 DATA 1/√(a^2+x^2), 1/a·tan^{-1}(x/a)+C, 300, 320
320 DATA 1/√(x^2±a^2), log|x+√(x^2±a^2)|+C, 310, 330
330 DATA √(a^2-x^2), 1/2·(x√(a^2-x^2)+a^2 sin^{-1}(x/a))+C, 320, 340
340 DATA 1/√(x^2-a^2), 1/2a·log|(x-a)/√(x^2-a^2)|+C (x>a), 330, 340
10010 MODE8: DIM: x=10: y=340: r=x
10020 DEFCHR$(255)="0000FF0000"
10110 RESTORE(r)
10120 READ m$, n$, b, c: IF m$="" THEN m$=CHR$(8)
10130 l$="["+MID$(STR$(r/10), 2)+"]": l=32-LEN(l$)
10140 CLS: PRINT CHR$(129); " "; m$; " dx"; TAB(1); l$; "="; n$:
10150 z=ASC(INPUT$(1,0))
10170 IF z=29 AND r<>x THEN r=x: GOTO 10110
10180 IF z=28 AND r<>y THEN r=y: GOTO 10110
10190 IF z=30 AND r<>b THEN r=b: GOTO 10110
10200 IF (z=31 OR z=13) AND r<>c THEN r=c: GOTO 10110
10280 GOTO 10150

```

## 3.4.5 5840 Laplace-Transformation

## LIB 5840 Laplace-Transformation

```

5  GOTO10010
10 DATA 1/P, 1, 10, 20
20 DATA 1/P^2, t, 10, 30
30 DATA 1/P^n, "t^n-1/(n-1)!      (n=1,2,3,...)", 20, 40
40 DATA 1/(P±m), e^±mt, 30, 50
50 DATA 1/(P(P+m)), 1/m*(1-e^(-mt)), 40, 60
60 DATA 1/(P^2(P+m)), 1/m^2*(e^(-mt)+mt-1), 50, 70
70 DATA a/(P^2+a^2), sinat, 60, 80
80 DATA P/(P^2+a^2), cosat, 70, 90
90 DATA 1/(P^2+a^2), 1/a*sinat, 80, 100
100 DATA a/(P^2-a^2), sinhat, 90, 110
110 DATA P/(P^2-a^2), coshat, 100, 120
120 DATA 1/(P^2-a^2), 1/a*sinhat, 110, 130
130 DATA 1/(P(P^2+a^2)), 1/a^2*(1-cosat), 120, 140
140 DATA 1/(P^2(P^2+a^2)), 1/a^3*(at-sinat), 130, 150
150 DATA 1/((P+m)(P+n)), 1/(n-m)*(e^(-mt)-e^(-nt)), 140, 160
160 DATA P/((P+m)(P+n)), 1/(m-n)*(me^(-mt)-ne^(-nt)), 150, 170
170 DATA 1/(P+m)^2, te^(-mt), 160, 180
180 DATA 1/(P+m)^n, "1/(n-1)!*t^(n-1)*e^(-mt) (n=1,2,3,...)", 170, 190
190 DATA P/(P+m)^2, e^(-mt)*(1-mt), 180, 200
200 DATA 1/(P(P+m)^2), 1/m^2*(1-(1+mt)e^(-mt)), 190, 210
210 DATA 1/(P^2(P+m)^2), t/m^2-2/m^3+2/m^3*e^(-mt)+t/m^2*e^(-mt), 200, 220
220 DATA (P+n)/(P+m)^2, ((n-m)t+1)e^(-mt), 210, 230
230 DATA 1/(P^2+a^2)^2, 1/2a^3*(sinat-at*cosat), 220, 240
240 DATA P/(P^2+a^2)^2, t/2a*sinat, 230, 250
250 DATA P^2/(P^2+a^2)^2, 1/2a*(sinat+at*cosat), 240, 260
260 DATA (P^2-a^2)/(P^2+a^2)^2, t*cosat, 250, 270
270 DATA 1/((P+m)^2+n^2), 1/n*e^(-mt)*sinnt, 260, 280
280 DATA (P+m)/((P+m)^2+n^2), e^(-mt)*cosnt, 270, 290
290 DATA 1/(P^4-a^4), 1/2a^3*(sinhat-sinat), 280, 300
300 DATA P/(P^4-a^4), 1/2a^2*(coshat-cosat), 290, 310
310 DATA P^2/(P^4-a^4), 1/2a*(sinhat+sinat), 300, 320
320 DATA P^3/(P^4-a^4), 1/2*(coshat+cosat), 310, 330
330 DATA P/(P^4+4a^4), 1/2a^2*sinat*sinhat, 320, 340
340 DATA 4a^3/(P^4+4a^4), sinat*coshat-cosat*sinhat, 330, 350
350 DATA 1/P*((P-m)/(P+m)), -1+2e^(-mt), 340, 360
360 DATA 1/P^2*((P-m)/(P+m)), 2/m-t-2/m*e^(-mt), 350, 360
10010 MODE8: DIM: x=10: y=360: r=x
10020 DEFCHR$(255)="7840384038"
10030 DEFCHR$(254)="40F8480000"
10040 DEFCHR$(253)="90B8900000"
10050 DEFCHR$(252)="7850502000"
10110 RESTORE(r)
10120 READm$, n$, b, c
10130 l$="[ "+MID$(STR$(r/10), 2)+"]": l=32-LEN(l$)
10140 CLS: PRINT"F(p)="; m$: TAB(1); l$: n$:
10150 z=ASC(INPUT$(1, 0))
10170 IF z=29 AND r<>x THEN r=x: GOTO10110
10180 IF z=28 AND r<>y THEN r=y: GOTO10110
10190 IF z=30 AND r<>b THEN r=b: GOTO10110
10200 IF (z=31 OR z=13) AND r<>c THEN r=c: GOTO10110
10280 GOTO10150

```

## 3.4.6 5900 Tabelle des Periodischen Systems

## LIB 5900 Tabelle des Periodischen Systems

10 **GOTO310**  
 100 **DATAAc** Actinium 89(7-3a) 227.0278  
 101 **DATAAg** Silver 47(5-1b) 107.8682  
 102 **DATAAl** Aluminium 13(3-3b) 26.98154  
 103 **DATAAm** Americium 95(7-3a) (243)  
 104 **DATAAr** Arsen 75(4-5b) 74.9216  
 105 **DATAAs** Arsenic 33(4-5b) 74.9216  
 106 **DATAAt** Astatine 85(6-7b) (210)  
 107 **DATAAu** Gold 79(6-1b) 196.9665  
 108 **DATAB** Boron 5(2-3b) 10.81  
 109 **DATABa** Barium 56(6-2a) 137.33  
 110 **DATABe** Beryllium 4(2-2a) 9.01218  
 111 **DATABi** Bismuth 83(6-5b) 208.9804  
 112 **DATABk** Berkelium 97(7-3a) (247)  
 113 **DATABr** Bromine 35(4-7b) 79.904  
 114 **DATAc** Carbon 6(2-4b) 12.011  
 115 **DATAca** Calcium 20(4-2a) 40.08  
 116 **DATAcd** Cadmium 48(5-2b) 112.41  
 117 **DATAce** Cerium 58(6-3a) 140.12  
 118 **DATAcf** Californium 98(7-3a) (251)  
 119 **DATAcl** Chlorine 17(3-7b) 35.453  
 120 **DATAcm** Curium 96(7-3a) (247)  
 121 **DATAco** Cobalt 27(4-8) 58.9332  
 122 **DATAcr** Chromium 24(4-6a) 51.996  
 123 **DATAcs** Cesium 55(6-1a) 132.9054  
 124 **DATAcu** Copper 29(4-1b) 63.546  
 125 **DATAcy** Dysprosium 66(6-3a) 162.50  
 126 **DATAer** Erbium 68(6-3a) 167.26  
 127 **DATAes** Einsteinium 99(7-3a) (252)  
 128 **DATAeu** Europium 63(6-3a) 151.96  
 129 **DATAf** Fluorine 9(2-7b) 18.998403  
 130 **DATAfe** Iron 26(4-8) 55.847  
 131 **DATAfm** Fermium 100(7-3a) (257)  
 132 **DATAfr** Francium 87(7-1a) (223)  
 133 **DATAga** Gallium 31(4-3b) 69.72  
 134 **DATAgd** Gadolinium 64(6-3a) 157.25  
 135 **DATAge** Germanium 32(4-4b) 72.59  
 136 **DATAh** Hydrogen 1(1-1a) 1.00794  
 137 **DATAhe** Helium 2(1-0) 4.00260  
 138 **DATAhf** Hafnium 72(6-4a) 178.49  
 139 **DATAhg** Mercury 80(6-2b) 200.59  
 140 **DATAho** Holmium 67(6-3a) 164.9304  
 141 **DATAi** Iodine 53(5-7b) 126.9045  
 142 **DATAin** Indium 49(5-3b) 114.82  
 143 **DATAir** Iridium 77(6-8) 192.22  
 144 **DATAk** Potassium 19(4-1a) 39.0983  
 145 **DATAkr** Krypton 36(4-0) 83.80  
 146 **DATAla** Lanthanum 57(6-3a) 138.9055  
 147 **DATAli** Lithium 3(2-1a) 6.941  
 148 **DATAlr** Lawrencium 103(7-3a) (260)  
 149 **DATAlu** Lutetium 71(6-3a) 174.967  
 150 **DATAmd** Mendeleevium 101(7-3a) (258)  
 151 **DATAmg** Magnesium 12(3-2a) 24.305  
 152 **DATAmn** Manganese 25(4-7a) 54.9380  
 153 **DATAmo** Molybdenum 42(5-6a) 95.94  
 154 **DATAn** Nitrogen 7(2-5b) 14.0067  
 155 **DATAna** Sodium 11(3-1a) 22.98977  
 156 **DATAnb** Niobium 41(5-5a) 92.9064  
 157 **DATAnd** Neodymium 60(6-3a) 144.24  
 158 **DATAne** Neon 10(2-0) 20.179  
 159 **DATAni** Nickel 28(4-8) 58.69  
 160 **DATAno** Nobelium 102(7-3a) (259)  
 161 **DATAnp** Neptunium 93(7-3a) 237.0482  
 162 **DATAo** Oxygen 8(2-6b) 15.9994  
 163 **DATAos** Osmium 76(6-8) 190.2

## LIB 5900 Tabelle des Periodischen Systems

```

164 DATAP Phosphorus 15(3-5b) 30.97376
165 DATAPa Protactinium91(7-3a)231.0359
166 DATAPb Lead 82(6-4b) 207.2
167 DATAPd Palladium 46(5-8) 106.42
168 DATAPm Promethium 61(6-3a) (145)
169 DATAPo Polonium 84(6-6b) (209)
170 DATAPr Praseodymium59(6-3a)140.9077
171 DATAPt Platinum 78(6-8) 195.08
172 DATAPu Plutonium 94(7-3a) (244)
173 DATARa Radium 88(7-2a) 226.0254
174 DATARb Rubidium 37(5-1a) 85.4678
175 DATARe Rhenium 75(6-7a) 186.207
176 DATARh Rhodium 45(5-8) 102.9055
177 DATARn Radon 86(6-0) (222)
178 DATARu Ruthenium 44(5-8) 101.07
179 DATAS Sulfur 16(3-6b) 32.06
180 DATASb Antimony 51(5-5b) 121.75
181 DATASc Scandium 21(4-3a) 44.9559
182 DATASe Selenium 34(4-6b) 78.96
183 DATASi Silicon 14(3-4b) 28.0855
184 DATASm Samarium 62(6-3a) 150.36
185 DATASn Tin 50(5-4b) 118.69
186 DATASr Strontium 38(5-2a) 87.62
187 DATATa Tantalum 73(6-5a) 180.9479
188 DATATb Terbium 65(6-3a) 158.9254
189 DATATc Technetium 43(5-7a) (98)
190 DATATe Tellurium 52(5-6b) 127.60
191 DATATh Thorium 90(7-3a) 232.0381
192 DATATi Titanium 22(4-4a) 47.88
193 DATATl Thallium 81(6-3b) 204.383
194 DATATm Thulium 69(6-3a) 168.9342
195 DATAU Uranium 92(7-3a) 238.0289
196 DATAV Vanadium 23(4-5a) 50.9415
197 DATAW Tungsten 74(6-6a) 183.85
198 DATAx Xenon 54(5-0) 131.29
199 DATAY Yttrium 39(5-3a) 88.9059
200 DATAyb Ytterbium 70(6-3a) 173.04
201 DATAZn Zinc 30(4-2b) 65.38
202 DATAZr Zirconium 40(5-4a) 91.22
203 DATA#
310 DIM:MODE8:ERASEa#:DIMa$(17)
320 a$(0)=" 1a 2a 3a 4a 5a 6a 7a 8 8 8"
330 a$(1)=" 1b 2b 3b 4b 5b 6b 7b 0"
340 a$(2)="H"
350 a$(3)=" He"
360 a$(4)="Li Be"
370 a$(5)=" B C N O F Ne"
380 a$(6)="Na Mg"
390 a$(7)=" Al Si P S Cl Ar"
400 a$(8)="K Ca Sc Ti V Cr Mn Fe Co Ni"
410 a$(9)="Cu Zn Ga Ge As Se Br Kr"
420 a$(10)="Rb Sr Y Zr Nb Mo Tc Ru Rh Pd"
430 a$(11)="As Cd In Sn Sb Te I Xe"
440 a$(12)="Cs Ba ** Hf Ta W Re Os Ir Pt"
450 a$(13)="Au Hg Tl Pb Bi Po At Rn"
460 a$(14)="Fr Ra **"
470 a$(15)="
471 a$(16)="Lanthanoid: La Ce Pr Nd Pm Sm Eu Gd Tb Dy Ho Er Tm Yb Lu"
472 a$(17)="Actinoid:Ac Th Pa U Np Pu Am Cm Bk Cf Es Fm Md No Lr"
490 a=2
500 CLS:IFa<16 THENPRINTa$(a MOD 2);TAB(0);MID$(STR$(INT(a/2)),2);":":a$(a);
ELSEPRINTRIGHT$(STR$(a),1)":":a$(a);
510 k=ASC(INPUT$(1,a)):IFk=13 THEN620
520 IFk=28 ANDa<15 ANDFRAC(a/2)=0 THENa=a+1:GOTO500
525 IFk=29 ANDa>2 ANDFRAC(a/2)<>0 THENa=a-1:GOTO500
530 IFk=30 ANDa>3 THENa=a-SGN(17-a)-1:GOTO500
535 IFk=31 ANDa<17 THENa=a+(SGN(14.5-a)+1)/2+1:GOTO500
540 GOTO510
620 CLS:PRINT"Input the symbol :";TAB(0);"Push [EXE] then return";

```



## LIB 5900 Tabelle des Periodischen Systems

```

630 LOCATE19,0:PRINT " ";:LOCATE19,0:INPUTa3:d#
:IFd#="" THEN500 ELSEe=LEN(d#):IFe=3 THEN630
635 d=ASC(d#):IFd>96 THENd=d-32
636 c=ASC(MID$(d#,2)):IFc<91 THENc=c+32
640 IFd<65 ORd>90 THEN630 ELSEd#=CHR$(d)+CHR$(c)
650 b=100+14*INT(d/67)+15*INT(d/70)+17*INT(d/76)+16*INT(d/79)+17*INT(d/83)
+16*INT(d/85)
660 RESTORE(b)
665 READc#:e#=LEFT$(c#,2):IFASC(e#)>d THEN630
670 IFe#<>d# THEN665 ELSELOCATE0,1:PRINTCHR$(5):c#
690 GOTO630

```

## 3.4.7 5910 Wissenschaftliche Konstanten

## LIB 5910 Wissenschaftliche Konstanten

```

5 GOTO10010
10 DATAF ,9.648456,×104 [C·mol-1],×103 [emu·mol-1],E4,E3,10,20
20 DATAG ,6.6720,×10-11 [m3·s-2·kg-1],×10-8 [cm3·s-2·g-1],E-11,E-8,10,30
30 DATAh ,6.022045,×1023 [mol-1],×1023 [mol-1],E23,E23,20,40
40 DATAR ,8.31441, " [J·mol-1·K-1]",×101 [erg·mol-1·K-1],,E7,30,50
50 DATAR2 ,1.097373177,×101 [m1],×105 [cm1],E7,E5,40,60
60 DATAUm ,22.41383,×10-3 [m3·mol-1],×103 [cm3·mol-1],E-3,E3,50,70
70 DATAa0 ,5.291706,×10-11 [m],×10-9 [cm],E-11,E-9,60,80
80 DATAc ,299792458, " [m·s-1]",×108 [cm·s-1],,E2,70,90
90 DATAe ,1.6021892,×10-19 [C],×10-20 [emu],E-19,E-20,80,100
100 DATAπ ,9.80665, " [m·s-2]",×108 [cm·s-2],,E2,90,110
110 DATAh ,6.626176,×10-34 [J·s],×10-27 [erg·s],E-34,E-27,100,120
120 DATAk ,1.380662,×10-23 [J·K-1],×10-16 [erg·K-1],E-23,E-16,110,130
130 DATAme ,9.109534,×10-31 [kg],×10-28 [g],E-31,E-28,120,140
140 DATAmn ,1.6749543,×10-27 [kg],×10-24 [g],E-27,E-24,130,150
150 DATAmp ,1.6726485,×10-27 [kg],×10-24 [g],E-27,E-24,140,160
160 DATAu ,1.6605655,×10-27 [kg],×10-24 [g],E-27,E-24,150,170
170 DATAe0 ,8.854187818,×10-12 [F·m-1],,E-12,,160,180
180 DATAμ0 ,12.5663706144,×10-7 [H·m-1],,E-7,,170,190
190 DATAμb ,9.274078,×10-24 [J·T-1],×10-21 [erg·G-1],E-24,E-21,180,200
200 DATAμe ,9.284832,×10-24 [J·T-1],×10-21 [erg·G-1],E-24,E-21,190,210
210 DATAμp ,1.4106171,×10-26 [J·T-1],×10-23 [erg·G-1],E-26,E-23,200,220
220 DATAσ ,5.67032,×10-8 [W·m-2·K-4],×10-5 [erg·s-1·cm-2·K-4],E-8,E-5,210,220
10010 u=0:MODE8:DIM:x=10:y=220:d#=CHR$(5):r=x:DEFCHR$(255)="060A040A0D"
10050 CLS:LOCATE0,1:PRINT">constant : key A-Z ?";
10110 RESTORE(r)
10120 READm#,n#,s#,t#,x#,y#,b,c:l#=["+MID$(STR$(r/10),2)+"]
:IFr<100 THENl#=" "+l#
10130 IFu=1 THENu#=t# ELSEu#=s#
10140 d=0:LOCATE0,0:PRINTd#:m#;" =":n#:u#,:LOCATE27,1:PRINTl#
10150 LOCATE30,1:z=ASC(INPUT$(1,0))
10165 IFz>96 THENz=z-32
10170 IFz=28 ANDu<>1 THENu=1:GOTO10130
10180 IFz=29 ANDu<>0 THENu=0:GOTO10130
10190 IFz=30 ANDr<>b THENr=b:GOTO10110
10200 IF(z=31 ORz=13) ANDr<>c THENr=c:GOTO10110
10220 IFz>64 ANDz<91 THENGOSUB10310:d=1
10230 IF(z=28 ORz=29) ANDd=1 THEN10140
10280 GOTO10150
10310 LOCATE0,0:PRINT[";CHR$(z):"];
10320 IFu=1 THENv#=n#+y# ELSEv#=n#+x#
10330 a=VAL(v#)
10350 ON z-64 GOSUB10401,10402,10403,10404,10405,10406,10407,10408,10409,10410
,10411,10412,10413,10414,10415,10416,10417,10418,10419,10420,10421,10422,10423
,10424,10425,10426
10401 A=a:RETURN
10402 B=a:RETURN
10403 C=a:RETURN
10404 D=a:RETURN

```

## LIB 5910 Wissenschaftliche Konstanten

```

10405 E=a: RETURN
10406 F=a: RETURN
10407 G=a: RETURN
10408 H=a: RETURN
10409 I=a: RETURN
10410 J=a: RETURN
10411 K=a: RETURN
10412 L=a: RETURN
10413 M=a: RETURN
10414 N=a: RETURN
10415 O=a: RETURN
10416 P=a: RETURN
10417 Q=a: RETURN
10418 R=a: RETURN
10419 S=a: RETURN
10420 T=a: RETURN
10421 U=a: RETURN
10422 V=a: RETURN
10423 W=a: RETURN
10424 X=a: RETURN
10425 Y=a: RETURN
10426 Z=a: RETURN

```

## 3.4.8 5920 Konstante der elektrolytischen Dissoziation

## LIB 5920 Konstante der elektrolytischen Dissoziation

```

5 GOTO10010
10 DATAHCOOH, HCOO- + H+, 10, 20, 1.77×10-4, a,
20 DATAH2COOH, CH2COO- + H+, 10, 30, 1.75×10-5, a,
30 DATAH2C=COOH, C2H2COO- + H+, 20, 40, 6.31×10-5, a, 2
40 DATAH2CO2, H+ + HCO2-, 30, 50, 4.45×10-1, a,
50 DATAH2OH, C2H2O- + H+, 40, 60, 1.00×10-10, a, 2
60 DATANH2 + H2O, NH2+ + OH-, 50, 70, 1.78×10-5, b, 1
70 DATAH2N + H2O, C2H2NH+ + OH-, 60, 80, 1.5×10-9, b,
80 DATAH2NH2 + H2O, C2H2NH2+ + OH-, 70, 80, 3.8×10-10, b, 2
10010 MODE8: DIM: x=10: y=80: r=x
10020 DEFCHR$(252)="2E2A3A0000"
10030 DEFCHR$(253)="2A2A3E0000"
10035 DEFCHR$(254)="3A2A2E0000"
10040 DIM: ERASEa#: DIMa#(2): RESTORE10080
10050 FORi=1 TO2: READa#(i): NEXT
10080 DATA38083E0000, 3E2A2E0000
10110 RESTORE(r)
10120 READm#, n#, b, c, d#, e#, d: CLS
10125 IFd THENDEFCHR$(255)=a#(d)
10130 l$="[ "+MID$(STR$(r/10), 2)+" ]"
10140 CLS: PRINTm#: " ↔ "; n#: TAB(0); "K": e#: " = "; d#: " [mol/l]"; TAB(28): l#:
10150 z=ASC(INPUT$(1, @))
10170 IFz=29ANDr<>x THENr=x: GOTO10110
10180 IFz=28ANDr<>y THENr=y: GOTO10110
10190 IFz=30ANDr<>b THENr=b: GOTO10110
10200 IF(z=31 ORz=13) ANDr<>c THENr=c: GOTO10110
10280 GOTO10150

```

## 3.4.9 5930 Bewegung und Energie

## LIB 5930 Bewegung und Energie

```

5  GOTO10010
10 DATAUniformly accelerated motion,"v=vt+at , a=av/at , s=vt+at²/2",10,20,
20 DATANewton's equation of motion,F=ma,10,30,
30 DATACircular motion,T=2πr/v=2π/ω=1/f,20,40,
40 DATACircular motion,"ω=2π/T=2πf=v/r , F=mrv²=mv²/r",30,50,
50 DATASimple harmonic oscillation,"x=r·sinωt , v=rω·cosωt , a=-ω²x",40,60,
60 DATAHooke's law,F=-kx,50,70,
70 DATASpring,"a=F/m=-k/m·x , T=2π√(m/k)",60,80,
80 DATASimple pendulum,"a=F/m=-g/l·x , T=2π√(l/g)",70,90,
90 DATAPotential energy,Ep=mgh,80,100,
100 DATAElastic energy (spring),Ee=1/2·kx²,90,110,
110 DATAKinetic energy,Ek=1/2·mv²,100,120,
120 DATACoefficient of friction,F=μN,110,130,
130 DATAWork,W=Fs,120,140,
140 DATAKepler's law (3'rd),T²/r³=Constant,130,150,
150 DATAUniversal gravitation,"F=G·Mm/r² , G=6.7*10⁻¹¹ [N·m²/kg²]",140,160,
160 DATAPotential energy (Planet),Up=-G·Mm/r ,150,170,
170 DATAKinetic energy (Planet),Ek=1/2·mv²,160,180,
180 DATAMoment of inertia,"I=mr² , E=1/2·Iω²",170,190,
190 DATAAngular momentum,J=Iω,180,200,
200 DATAConservation of momentum,mvᵢ+Mvᵢ=mvᵢ+Mvᵢ,190,200,
10010 MODE8: DIM: x=10: y=200: r=x
10020 DEFCHR$(255)="202048F808"
10030 DEFCHR$(254)="3E020C223E"
10040 DEFCHR$(253)="2E2A3A0000"
10050 DEFCHR$(252)="123E020000"
10110 RESTORE(r)
10120 CLS: READm$,n$,b,c,d
10130 l$="[ "+MID$(STR$(r/10),2)+"]": l=32-LEN(l$)
10140 PRINTm$: TAB(l): l$: n$: : LOCATE0,0: LOCATE1,1
10150 z=ASC(INPUT$(1,0))
10170 IFz=29 ANDr<>x THENr=x: GOTO10110
10180 IFz=28 ANDr<>y THENr=y: GOTO10110
10190 IFz=30 ANDr<>b THENr=b: GOTO10110
10200 IF(z=31 ORz=13) ANDr<>c THENr=c: GOTO10110
10280 GOTO10150

```

## 3.4.10 5932 Wellenbewegung

## LIB 5932 Wellenbewegung

```

5 GOTO10010
10 DATA Wave, "v=λ/T=fλ , y=a·sin2π(t/T-x/λ)", 10, 20,
20 DATA Wave of strings, v=√(F/μ), 10, 30, 1, 3E40484830
30 DATA Interference, "l=λ=(2n+1)λ/2 , l=λ=nλ", 20, 40,
40 DATA Stationary wave, "l=nλ/2 , l=(2n-1)λ/4 (n≠0)", 30, 50,
50 DATA Refraction, n=sinθ/sinφ=vλ/vλ=μλ/μλ, 40, 60,
60 DATA Natural frequency (wave), f=(1/2l)·√(T/μ), 50, 70, 1, 3E40484830
70 DATA Velocity of sound, v=331.5+0.61T, 60, 80,
80 DATA Doppler effect, f=f₀·(v-v₀)/(v-v₀), 70, 90,
90 DATA Beat, f=f₁-f₂ (f₁>f₂), 80, 100,
100 DATA Reflectivity, Rₐ=((n₁²-n₂²)/(n₁²+n₂²))², 90, 110,
110 DATA Critical angle, sinθ=n₂/n₁, 100, 120,
120 DATA de Broglie wave, λ=h/mv, 110, 130,
130 DATA Quantum condition, 2πr=nh/mv=nλ, 120, 140,
140 DATA Photoelectron, 1/2·mv²=hν-W, 130, 150, 1, 3E02040830
150 DATA Frequency condition, hν=Em-En (m>n), 140, 160,
160 DATA Light wave, "λ=c/ν , c=2.998·10⁸ [m/s]", 150, 160, 1, 3E02040830
10010 MODE8: DIM: x=10: y=160: r=x
10020 DEFCHR$(255)="424C300806"
10030 DEFCHR$(253)="2E2A3A0000"
10050 DEFCHR$(252)="123E020000"
10110 RESTORE(r)
10120 CLS: READ m$, n$, b, c, d: IF d=1 THEN READ d$: DEFCHR$(254)=d$
10130 l$="["+MID$(STR$(r/10), 2)+"]": l=32-LEN(l$)
10140 PRINT m$: TAB(1): l$: n$:
10150 z=ASC(INPUT$(1,0))
10170 IF z=29 AND r<>x THEN r=x: GOTO 10110
10180 IF z=28 AND r<>y THEN r=y: GOTO 10110
10190 IF z=30 AND r<>b THEN r=b: GOTO 10110
10200 IF (z=31 OR z=13) AND r<>c THEN r=c: GOTO 10110
10280 GOTO 10150

```

## 3.4.11 5934 Wechselstrom und Gleichstromkreise

## LIB 5934 Wechselstrom und Gleichstromkreise

```

5 GOTO10010
10 DATAOhm's law,"U=IR (I=Q/t , R=ρ·l/S)",10,20,1,3E40484830
20 DATAResistance,"R=R1+R2 , 1/R=1/R1+1/R2",10,30,
30 DATADC circuit,U=E-IR,20,40,
40 DATADC Power & Joule heat,"P=IU=I2R , W=IUt=Pt",30,50,
50 DATAConductance,G=1/R=I/U,40,60,
60 DATAKirchhoff's law,"ΣI=0 , ΣU=0",50,70,
70 DATAWheatstone bridge,R1R2=R3R4,60,80,1,2A2A3E0000
80 DATAAC instantaneous value,"U=U0·sinωt , I=I0·sinωt",70,90,
90 DATAAC effective value,"I=I0/√2 , U=U0/√2",80,100,
100 DATAAC Power,P=UI=U0I0/2,90,110,
110 DATAPower factor,P=VI·cosφ,100,120,
120 DATATransformer,"I1U1=I2U2 , N1/N2=U1/U2",110,130,
130 DATAReactance,"XL=ωL=2πfL , XC=1/ωC=1/2πfC",120,140,
140 DATAImpedance,"Z=√(R2+(-L-1/ωC)2) , U0=ZI0",130,150,
150 DATANatural frequency (circuit),f0=1/2π√(LC),140,160,
160 DATAElectric oscillation,1/2·Q2/C+1/2·LI2=Constant,150,160,
10010 MODE8:DIM:x=10:y=160:r=x
10020 DEFCHR$(255)="3E020C223E"
10040 DEFCHR$(253)="2E2A3A0000"
10050 DEFCHR$(252)="123E020000"
10110 RESTORE(r)
10120 CLS:READm$,n$,b,c,d:IFd=1 THENREADd$:DEFCHR$(254)=d$
10130 l$="[ "+MID$(STR$(r/10),2)+" ]":l=32-LEN(l$)
10140 PRINTm$:TAB(1):l$:n$:
10150 z=ASC(INPUT$(1,0))
10170 IFz=29 ANDr<>x THENr=x:GOTO10110
10180 IFz=28 ANDr<>y THENr=y:GOTO10110
10190 IFz=30 ANDr<>b THENr=b:GOTO10110
10200 IF(z=31 ORz=13) ANDr<>c THENr=c:GOTO10110
10280 GOTO10150

```

## 3.4.12 5936 Elektrische und magnetische Felder

## LIB 5936 Elektrische und magnetische Felder

```

5 GOTO10010
10 DATA Coulomb's law (electric f.), "F=k0·Q1Q2/r2 , k0=9·109 [N·m2/C2]", 10, 20,
20 DATA Electric field, "E=U/d , F=Q·E , W=Q·U", 10, 30,
30 DATA Capacitance, "Q=CU , C=ε0·S/d", 20, 40,
40 DATA Capacitance, "C=C1+C2 , 1/C=1/C1+1/C2", 30, 50,
50 DATA Dielectric constant, "D=ε0E , C=εC0", 40, 60,
60 DATA Electrostatic energy, U=1/2·Q·U=1/2·C·U2, 50, 70,
70 DATA Electrons in electric field, "a=Q·E/m , 1/2·mv2=eU", 60, 80,
80 DATA Coulomb's law (magnetic f.), "F=k0·m1m2/r2 , k0=107/(4π)2", 70, 90,
90 DATA Magnetic field, "H=I/2πr , H=I/2r , H=nI", 80, 100,
100 DATA Magnetic field, F=μ0 IH=IB, 90, 110,
110 DATA Magnetic flux density, B=μ0I/2πr=μ0H, 100, 120,
120 DATA Lorentz force, "F=Q·vB , r=mv/QB", 110, 130,
130 DATA Electrons in magnetic field, "1/2·mv2=Q2B2r2/2m , v=r·Q·B/m", 120, 140,
140 DATA Faraday's law of induction, U=-n·ΔΦ/Δt, 130, 150,
150 DATA Electromagnetic induction, "U=E1=vB1 , I=vB1/R", 140, 160,
160 DATA Mutual induction, U2=-M·ΔI1/Δt, 150, 170,
170 DATA Self-induction, U1'=-L·ΔI1/Δt, 160, 170,
10010 MODE8: DIM: x=10: y=170: r=x
10020 DEFCHR$(254)="3E020C223E"
10040 DEFCHR$(253)="2E2A3A0000"
10050 DEFCHR$(252)="123E020000"
10110 RESTORE(r)
10120 CLS: READ m$, n$, b, c, d
10130 l$=["+MID$(STR$(r/10), 2)+"] : l=32-LEN(l$)
10140 PRINT m$: TAB(l): l$: n$: : LOCATE 0, 0: LOCATE l, 1
10150 z=ASC(INPUT$(1, 0))
10170 IF z=29 AND r<>x THEN r=x: GOTO 10110
10180 IF z=28 AND r<>y THEN r=y: GOTO 10110
10190 IF z=30 AND r<>b THEN r=b: GOTO 10110
10200 IF (z=31 OR z=13) AND r<>c THEN r=c: GOTO 10110
10280 GOTO 10150

```

## 3.4.13 5938 Thermodynamik und Anderes

## LIB 5938 Thermodynamik und Anderes

```

5 GOTO10010
10 DATAAbsolute temperature,T[K]=t[°C]+273.15,10,20,
20 DATAHeat capacity,Q=CT=mcT,10,30,
30 DATAMechanical equivalent of heat,"W=JQ , J=4.19 [J/cal]",20,40,
40 DATABoyle's law,PV=Constant (T=constant),30,50,
50 DATAVolume & Temperature,U=U0(1+T/273),40,60,
60 DATACharle's law,U/U0=T/T0,50,70,
70 DATAEquation of state,"PV=nRT , R=8.31 [J/K]",60,80,
80 DATALaw of partial pressures,P=P0+P0+P0+...,70,90,1,2A2A3E0000
90 DATAPressure,P=1/3·nm²,80,100,1,B08C828CB0
100 DATAInternal energy,U=1/2·m²N=3/2·nRT,90,110,1,B08C828CB0
110 DATASpecific heat,"Cv=ΔU/ΔT=3R/2 , Cp=ΔU/ΔT+R=5R/2",100,120,
120 DATAHalf life,N=N0(1/2)^(x/τ) (x=1/T),110,130,1,40F8480000
130 DATAMass-energy relation ,E=mc²,120,130,
10010 MODE8:DIM:x=10:y=130:r=x
10040 DEFCHR$(253)="2E2A3A0000"
10050 DEFCHR$(252)="123E020000"
10110 RESTORE(r)
10120 CLS:READm$,n$,b,c,d:IFd=1 THENREADd$:DEFCHR$(254)=d$
10130 l$="["+MID$(STR$(r/10),2)+"]":l=32-LEN(l$)
10140 PRINTm$;TAB(l);l$;n$;
10150 z=ASC(INPUT$(1,0))
10170 IFz=29 ANDr<>x THENr=x:GOTO10110
10180 IFz=28 ANDr<>y THENr=y:GOTO10110
10190 IFz=30 ANDr<>b THENr=b:GOTO10110
10200 IF(z=31 ORz=13) ANDr<>c THENr=c:GOTO10110
10280 GOTO10150

```

## 3.4.14 5950 Metrische Umwandlungen für Länge

## LIB 5950 Metrische Umwandlungen für Länge

```

5  GOTO10010
10 DATAx[cm],0.01x[m],X[m]=0.01*x[cm],10,20
20 DATAx[cm],0.393701x[in],X[in]=.393701*x[cm],10,30
30 DATAx[cm],0.0328084x[ft],X[ft]=.0328084*x[cm],20,40
40 DATAx[cm],0.0109361x[yd],X[yd]=.0109361*x[cm],30,50
50 DATAx[cm],0.0000621371x[mile],X[mile]=6.21371E-6*x[cm],40,60
60 DATAx[m],100x[cm],X[cm]=100*x[m],50,70
70 DATAx[m],39.3701x[in],X[in]=39.3701*x[m],60,80
80 DATAx[m],3.28084x[ft],X[ft]=3.28084*x[m],70,90
90 DATAx[m],1.09361x[yd],X[yd]=1.09361*x[m],80,100
100 DATAx[m],0.000621371x[mile],X[mile]=.000621371*x[m],90,110
110 DATAx[in],2.54x[cm],X[cm]=2.54*x[in],100,120
120 DATAx[in],0.0254x[m],X[m]=.0254*x[in],110,130
130 DATAx[in],0.0833333x[ft],X[ft]=.0833333*x[in],120,140
140 DATAx[in],0.0277778x[yd],X[yd]=.0277778*x[in],130,150
150 DATAx[in],0.000157828x[mile],X[mile]=1.57828E-5*x[in],140,160
160 DATAx[ft],30.48x[cm],X[cm]=30.48*x[ft],150,170
170 DATAx[ft],0.3048x[m],X[m]=.3048*x[ft],160,180
180 DATAx[ft],12x[in],X[in]=12*x[ft],170,190
190 DATAx[ft],0.333333x[yd],X[yd]=.333333*x[ft],180,200
200 DATAx[ft],0.000189394x[mile],X[mile]=1.89394E-4*x[ft],190,210
210 DATAx[yd],91.44x[cm],X[cm]=91.44*x[yd],200,220
220 DATAx[yd],0.9144x[m],X[m]=.9144*x[yd],210,230
230 DATAx[yd],36x[in],X[in]=36*x[yd],220,240
240 DATAx[yd],3x[ft],X[ft]=3*x[yd],230,250
250 DATAx[yd],0.000568182x[mile],X[mile]=5.68182E-4*x[yd],240,260
260 DATAx[mile],160934.4x[cm],X[cm]=160934.4*x[mile],250,270
270 DATAx[mile],1609.344x[m],X[m]=1609.344*x[mile],260,280
280 DATAx[mile],63360x[in],X[in]=63360*x[mile],270,290
290 DATAx[mile],5280x[ft],X[ft]=5280*x[mile],280,300
300 DATAx[mile],1760x[yd],X[yd]=1760*x[mile],290,300
10010 MODE8: DIM: x=10: y=300: r=x
10050 F$="Metric conversion (length)"
10110 RESTORE(r)
10120 READm#,n#,s#,b,c
10130 l$="["+MID$(STR$(r/10),2)+"]": l=32-LEN(l$)
10140 CLS: PRINTF$: TAB(l): l$:m$: " → ":n$:
10150 z=ASC(INPUT$(1,0))
10170 IFz=29 ANDr<>x THENr=x: GOTO10110
10180 IFz=28 ANDr<>y THENr=y: GOTO10110
10190 IFz=30 ANDr<>b THENr=b: GOTO10110
10200 IFz=31 ANDr<>c THENr=c: GOTO10110
10210 IFz=13 THENCALCf$=s#: CALCJMP
10280 GOTO10150

```



## 3.4.15 5960 Metrische Umwandlungen für Flächen

## LIB 5960 Metrische Umwandlungen für Flächen

```

5 GOTO10010
10 DATAx[m²],0.01x[a],X[a]=0.01*x[m²],10,20
20 DATAx[m²],0.000247105x[acre],X[acre]=2.47105E-4*x[m²],10,30
30 DATAx[m²],0.000000386102x[mile²],X[mile²]=3.86102E-7*x[m²],20,40
40 DATAx[a],100x[m²],X[m²]=100*x[a],30,50
50 DATAx[a],0.0247105x[acre],X[acre]=0.0247105*x[a],40,60
60 DATAx[a],0.0000386102x[mile²],X[mile²]=3.86102E-5*x[a],50,70
70 DATAx[acre],4046.86x[m²],X[m²]=4046.86*x[acre],60,80
80 DATAx[acre],40.4686x[a],X[a]=40.4686*x[acre],70,90
90 DATAx[acre],0.0015625x[mile²],X[mile²]=.0015625*x[acre],80,100
100 DATAx[mile²],2589990x[m²],X[m²]=2589990*x[mile²],90,110
110 DATAx[mile²],25899.9x[a],X[a]=25899.9*x[mile²],100,120
120 DATAx[mile²],640x[acre],X[acre]=640*x[mile²],110,120
10010 MODE8:DIM:x=10:y=120:r=x
10050 P$="Metric conversion (area)"
10110 RESTORE(r)
10120 READm$,n$,s$,b,c
10130 l$="["+MID$(STR$(r/10),2)+"]":l=32-LEN(l$)
10140 CLS:PRINTP$:TAB(1):l$:m$:" → ":n$:
10150 z=ASC(INPUT$(1,0))
10170 IFz=29 ANDr<>x THENr=x:GOTO10110
10180 IFz=28 ANDr<>y THENr=y:GOTO10110
10190 IFz=30 ANDr<>b THENr=b:GOTO10110
10200 IFz=31 ANDr<>c THENr=c:GOTO10110
10210 IFz=13 THENCALC$=s$:CALCJMP
10280 GOTO10150

```

## 3.4.16 5970 Metrische Umwandlungen für Rauminhalt

## LIB 5970 Metrische Umwandlungen für Rauminhalt

```

5 GOTO10010
10 DATAx[cm³],0.000001x[cm³],X[cm³]=1e-6*x[cm³],10,20
20 DATAx[cm³],0.0610237x[in³],X[in³]=.0610237*x[cm³],10,30
30 DATAx[cm³],0.0000353147x[ft³],X[ft³]=3.53147E-5*x[cm³],20,40
40 DATAx[cm³],0.001x[l],X[l]=.001*x[cm³],30,50
50 DATAx[cm³],0.000264172x[gal(US)],X[gal(US)]=2.64172E-4*x[cm³],40,60
60 DATAx[cm³],0.000219968x[gal(UK)],X[gal(UK)]=2.19968E-4*x[cm³],50,70
70 DATAx[m³],1000000x[cm³],X[cm³]=1E6*x[m³],60,80
80 DATAx[m³],61023.7x[in³],X[in³]=61023.7*x[m³],70,90
90 DATAx[m³],35.3147x[ft³],X[ft³]=35.3147*x[m³],80,100
100 DATAx[m³],1000x[l],x[l]=1E3*x[m³],90,110
110 DATAx[m³],264.172x[gal(US)],X[gal(US)]=264.172*x[m³],100,120
120 DATAx[m³],219.968x[gal(UK)],X[gal(UK)]=219.968*x[m³],110,130
130 DATAx[in³],16.3871x[cm³],X[cm³]=16.3871*x[in³],120,140
140 DATAx[in³],0.0000163871x[m³],X[m³]=1.63871E-5*x[in³],130,150
150 DATAx[in³],0.000578704x[ft³],X[ft³]=.000578704*x[in³],140,160
160 DATAx[in³],0.0163871x[l],X[l]=.0163871*x[in³],150,170
170 DATAx[in³],0.00432900x[gal(US)],X[gal(US)]=.004329*x[in³],160,180
180 DATAx[in³],0.00360464x[gal(UK)],X[gal(UK)]=.00360464*x[in³],170,190
190 DATAx[ft³],28316.8x[cm³],X[cm³]=28316.8*x[ft³],180,200
200 DATAx[ft³],0.0283168x[m³],X[m³]=.0283168*x[ft³],190,210
210 DATAx[ft³],1728x[in³],X[in³]=1728*x[ft³],200,220
220 DATAx[ft³],28.3168x[l],X[l]=28.3168*x[ft³],210,230
230 DATAx[ft³],7.48052x[gal(US)],X[gal(US)]=7.48052*x[ft³],220,240
240 DATAx[ft³],6.22882x[gal(UK)],X[gal(UK)]=6.22882*x[ft³],230,250
250 DATAx[l],1000x[cm³],X[cm³]=1E3*x[l],240,260
260 DATAx[l],0.001x[m³],X[m³]=.001*x[l],250,270
270 DATAx[l],61.0237x[in³],X[in³]=61.0237*x[l],260,280
280 DATAx[l],0.0353147x[ft³],X[ft³]=.0353147*x[l],270,290
290 DATAx[l],0.264172x[gal(US)],X[gal(US)]=.264172*x[l],280,300
300 DATAx[l],0.219968x[gal(UK)],X[gal(UK)]=.219968*x[l],290,300
310 DATAx[gal(US)],3785.41x[cm³],X[cm³]=3785.41*x[gal(US)],300,320
320 DATAx[gal(US)],0.00378541x[m³],X[m³]=.00378541*x[gal(US)],310,330
330 DATAx[gal(US)],231x[in³],X[in³]=231*x[gal(US)],320,340
340 DATAx[gal(US)],0.133681x[ft³],X[ft³]=.133681*x[gal(US)],330,350
350 DATAx[gal(US)],3.78541x[l],X[l]=3.78541*x[gal(US)],340,360
360 DATAx[gal(US)],0.832672x[gal(UK)],X[gal(UK)]=0.832672*x[gal(US)],350,370
370 DATAx[gal(UK)],4546.09x[cm³],X[cm³]=4546.09*x[gal(UK)],360,380
380 DATAx[gal(UK)],0.00454609x[m³],X[m³]=.00454609*x[gal(UK)],370,390
390 DATAx[gal(UK)],277.42x[in³],X[in³]=277.42*x[gal(UK)],380,400
400 DATAx[gal(UK)],0.160544x[ft³],X[ft³]=.160544*x[gal(UK)],390,410
410 DATAx[gal(UK)],4.54609x[l],X[l]=4.54609*x[gal(UK)],400,420
420 DATAx[gal(UK)],1.20095x[gal(US)],X[gal(US)]=1.20095*x[gal(UK)],410,420
10010 MODE8:DIM:x=10:y=300:r=x
10050 F$="Metric conversion (volume)"
10110 RESTORE(r)
10120 READm$,n$,s$,b,c
10130 l$="["+MID$(STR$(r/10),2)+"]":l=32-LEN(l$)
10140 CLS:PRINTF$:TAB(1):l$:m$:" -> ":n$::LOCATE0,0:LOCATE0,1
10150 z=ASC(INPUT$(1,0))
10170 IFz=29 ANDr<>z THENr=x:GOTO10110
10180 IFz=28 ANDr<>y THENr=y:GOTO10110
10190 IFz=30 ANDr<>b THENr=b:GOTO10110
10200 IFz=31 ANDr<>c THENr=c:GOTO10110
10210 IFz=13 THENCALCf=s$:CALCJMP
10280 GOTO10150

```

## 3.4.17 5980 Metrische Umwandlungen für Gewicht

## LIB 5980 Metrische Umwandlungen für Gewicht

```

5 GOTO10010
10 DATAx[a],0.001x[kg],X[kg]=.001*x[a],10,20
20 DATAx[a],0.0352740x[oz],X[oz]=.035274*x[a],10,30
30 DATAx[a],0.00220462x[lb],X[lb]=.00220462*x[a],20,40
40 DATAx[kg],1000x[a],X[a]=1E3*x[kg],30,50
50 DATAx[kg],35.2740x[oz],X[oz]=35.274*x[kg],40,60
60 DATAx[kg],2.20462x[lb],X[lb]=2.20462*x[kg],50,70
70 DATAx[oz],28.3495x[a],X[a]=28.3495*x[oz],60,80
80 DATAx[oz],0.0283495x[kg],X[kg]=.0283495*x[oz],70,90
90 DATAx[oz],0.0625x[lb],X[lb]=.0625*x[oz],80,100
100 DATAx[lb],453.59237x[a],X[a]=453.59237*x[lb],90,110
110 DATAx[lb],0.45359237x[kg],X[kg]=.45359237*x[lb],100,120
120 DATAx[lb],16x[oz],X[oz]=16*x[lb],110,120
10010 MODE8:DIM:x=10:y=120:r=x
10050 P$="Metric conversion (weight)"
10110 RESTORE(r)
10120 READm$,n$,s$,b,c
10130 l$="["+MID$(STR$(r/10),2)+"]":l=32-LEN(l$)
10140 CLS:PRINTP$:TAB(1):l$:m$:" → ":n$:
10150 z=ASC(INPUT$(1,0))
10170 IFz=29 ANDr<>x THENr=x:GOTO10110
10180 IFz=28 ANDr<>y THENr=y:GOTO10110
10190 IFz=30 ANDr<>b THENr=b:GOTO10110
10200 IFz=31 ANDr<>c THENr=c:GOTO10110
10210 IFz=13 THENCALC#=s$:CALCJMP
10280 GOTO10150

```

## 3.5 Statistik

### 3.5.1 6210 Obere Wahrscheinlichkeitsintegrale (Normalverteilung)

#### LIB 6210 Obere Wahrscheinlichkeitsintegrale (Normalverteilung)

```

10 MODE8: DIM: GOSUB500: c#=CHR$(5): x=0
20 CLS: PRINT "Upper probability  N(0,1^2)";
30 LOCATE0,1: PRINT c#: "x="; x; "?": INPUT@10: x: LOCATE0,0
60 LOCATE0,1: PRINT c#: ".....";
70 sr=0: GOSUB"LIB0:S6210": GOSUB500: IF sr THEN*
75 IF P THEN P=ROUND(P, LOG(ABS P)-6)
80 PRINT c#: "P="; P; : z#=INPUT$(1,0): GOTO30
500 ONERROR GOTO510: RETURN
510 IFERR=1 THEN CLS: ONERROR GOTO0
520 IFERR=30 THEN RESUME30
530 LOCATE0,1: PRINT c#: "not found": : z#=INPUT$(1,0): RESUME20

```

### 3.5.2 6220 Obere Wahrscheinlichkeitsintegrale (x^2 Verteilung)

#### LIB 6220 Obere Wahrscheinlichkeitsintegrale (x^2 Verteilung)

```

10 MODE8: DIM: GOSUB500: c#=CHR$(5): n=1: x=0: DEFCHR$(254)="3E02040830"
20 CLS: PRINT "Upper probability  X^2(x^2, n)";
30 z=n: s#="n": GOSUB400: n=z: IF n<0 OR FRAC n<>0 THEN30
40 z=x: s#="x^2": GOSUB400: x=z
60 LOCATE0,1: PRINT c#: ".....";
70 sr=0: GOSUB"LIB0:S6220": GOSUB500: IF sr THEN*
75 IF P THEN P=ROUND(P, LOG(ABS P)-6)
80 PRINT c#: "P="; P; : z#=INPUT$(1,0): GOTO30
400 LOCATE0,1: PRINT c#: s#: "="; z; "?": INPUT@10: z: LOCATE0,0: RETURN
500 ONERROR GOTO510: RETURN
510 IFERR=1 THEN CLS: ONERROR GOTO0
520 IFERR=400 THEN RESUME400
530 LOCATE0,1: PRINT c#: "not found": : z#=INPUT$(1,0): RESUME20

```

### 3.5.3 6230 Obere Wahrscheinlichkeitsintegrale (t Verteilung)

#### LIB 6230 Obere Wahrscheinlichkeitsintegrale (t Verteilung)

```

10 MODE8: DIM: GOSUB500: c#=CHR$(5): n=1: x=0: DEFCHR$(254)="3E02040830"
20 CLS: PRINT "Upper probability  t(x, n)";
30 z=n: s#="n": GOSUB400: n=z: IF n<0 OR FRAC n<>0 THEN30
40 z=x: s#="x": GOSUB400: x=z
60 LOCATE0,1: PRINT c#: ".....";
70 sr=0: GOSUB"LIB0:S6230": GOSUB500: IF sr THEN*
75 IF P THEN P=ROUND(P, LOG(ABS P)-6)
80 PRINT c#: "P="; P; : z#=INPUT$(1,0): GOTO30
400 LOCATE0,1: PRINT c#: s#: "="; z; "?": INPUT@10: z: LOCATE0,0: RETURN
500 ONERROR GOTO510: RETURN
510 IFERR=1 THEN CLS: ONERROR GOTO0
520 IFERR=400 THEN RESUME400
530 LOCATE0,1: PRINT c#: "not found": : z#=INPUT$(1,0): RESUME20

```

## 3.5.4 6240 Obere Wahrscheinlichkeitsintegrale (F Verteilung)

**LIB 6240** Obere Wahrscheinlichkeitsintegrale (F Verteilung)

```

10 MODE8: DIM: GOSUB500: c#=CHR$(5): n=1: m=1: x=0: DEFCHR$(254)="3E02040830"
: DEFCHR$(252)="123E020000": DEFCHR$(253)="2E2A3A0000"
20 CLS: PRINT "Upper probability F(x, m, m)":
30 z=n: s#="n": GOSUB400: n=z: IF n<0 OR FRAC n<>0 THEN 30
40 z=m: s#="m": GOSUB400: m=z: IF m<0 OR FRAC m<>0 THEN 40
50 z=x: s#="x": GOSUB400: x=z
60 LOCATE0, 1: PRINT c#: ".....":
70 sr=0: GOSUB"LIB0: S6240": GOSUB500: IF sr THEN*
75 IF s THEN s=ROUND(s, LOG(ABS s)-6)
80 PRINT c#: "P=": s: z#=INPUT$(1, @): GOTO 30
400 LOCATE0, 1: PRINT c#: s#: "=": z: "?": INPUT@10: z: LOCATE0, 0: RETURN
500 ON ERROR GOTO 510: RETURN
510 IF ERR=1 THEN CLS: ON ERROR GOTO 0
520 IF ERR=400 THEN RESUME 400
530 LOCATE0, 1: PRINT c#: "not found": z#=INPUT$(1, @): RESUME 20

```

## 3.5.5 6310 Obere Summenhäufigkeit (Binominal-Verteilung)

**LIB 6310** Obere Summenhäufigkeit (Binominal-Verteilung)

```

10 MODE8: DIM: GOSUB500: c#=CHR$(5): x=0: n=2: P=0
20 CLS: PRINT "Cumulative frequency B(x, n, P)":
30 z=n: s#="n": GOSUB400: n=z: IF n<0 OR FRAC n<>0 THEN 30
40 z=P: s#="P": GOSUB400: P=z: IF P<0 OR P>1 THEN 40
50 z=x: s#="x": GOSUB400: x=z: IF x<0 OR x>n OR FRAC x<>0 THEN 50
60 LOCATE0, 1: PRINT c#: ".....":
70 sr=0: GOSUB"LIB0: S6310": GOSUB500: IF sr THEN*
75 IF s THEN s=ROUND(s, LOG(ABS s)-6)
80 PRINT c#: "P=": s: z#=INPUT$(1, @): GOTO 30
400 LOCATE0, 1: PRINT c#: s#: "=": z: "?": INPUT@10: z: LOCATE0, 0: RETURN
500 ON ERROR GOTO 510: RETURN
510 IF ERR=1 THEN CLS: ON ERROR GOTO 0
520 IF ERR=400 THEN RESUME 400
530 LOCATE0, 1: PRINT c#: "not found": z#=INPUT$(1, @): RESUME 20

```

## 3.5.6 6320 Obere Summenhäufigkeit (Poisson-Verteilung)

**LIB 6320** Obere Summenhäufigkeit (Poisson-Verteilung)

```

10 MODE8: DIM: GOSUB500: c#=CHR$(5): k=0: x=0: DEFCHR$(254)="424C300806"
20 CLS: PRINT "Cumulative frequency P(x, k)":
30 z=k: s#="k": GOSUB400: k=z: IF k<0 THEN 30
40 z=x: s#="x": GOSUB400: x=z: IF x<0 OR FRAC x<>0 THEN 40
60 LOCATE0, 1: PRINT c#: ".....":
70 sr=0: GOSUB"LIB0: S6320": GOSUB500: IF sr THEN*
75 IF s THEN s=ROUND(s, LOG(ABS s)-6)
80 PRINT c#: "P=": s: z#=INPUT$(1, @): GOTO 30
400 LOCATE0, 1: PRINT c#: s#: "=": z: "?": INPUT@10: z: LOCATE0, 0: RETURN
500 ON ERROR GOTO 510: RETURN
510 IF ERR=1 THEN CLS: ON ERROR GOTO 0
520 IF ERR=400 THEN RESUME 400
530 LOCATE0, 1: PRINT c#: "not found": z#=INPUT$(1, @): RESUME 20

```

## 3.5.7 6330 Obere Summenhäufigkeit (Hypergeometrische Verteilung)

## LIB 6330 Obere Summenhäufigkeit (Hypergeometrische Verteilung)

```

10 MODE8: DIM: GOSUB500: c#=CHR$(5): m=2: k=1: n=1: x=0
20 CLS: PRINT "Cumulative frequency H(x,n,M,N)";
30 z=m: s#="N": GOSUB400: m=z: IF m<2 THEN30
40 z=k: s#="M": GOSUB400: k=z: IF k>m THEN40
50 z=n: s#="n": GOSUB400: n=z: IF n>m-k THEN50
60 z=x: s#="x": GOSUB400: x=z: IF x>n OR x>k THEN60
65 LOCATE0,1: PRINT c#: ".....";
70 sr=0: GOSUB"LIB0:S6330": GOSUB500: IF sr THEN*
75 IF s THEN s=ROUND(s, LOG(ABS s)-6)
80 PRINT c#: "P="; s; : z#=INPUT$(1,0): GOTO30
400 LOCATE0,1: PRINT c#: s#: "="; z: "?"; : INPUT@10: z: LOCATE0,0
: IF z<0 OR FRAC z<>0 THEN400 ELSE RETURN
500 ON ERROR GOTO510: RETURN
510 IF ERR=1 THEN CLS: ON ERROR GOTO0
520 IF ERR=400 THEN RESUME400
530 LOCATE0,1: PRINT c#: "not found"; : z#=INPUT$(1,0): RESUME20

```

## 3.5.8 6410 Prozentpunkt (Normalverteilung)

## LIB 6410 Prozentpunkt (Normalverteilung)

```

10 MODE8: DIM: GOSUB500: c#=CHR$(5): P=0
20 CLS: PRINT "Percentage points N(0,1^2)";
30 LOCATE0,1: PRINT c#: "P="; P: "?"; : INPUT@10: P: LOCATE0,0: IF P=<0 OR P=>1 THEN30
60 LOCATE0,1: PRINT c#: ".....";
70 sr=0: GOSUB"LIB0:S6410": GOSUB500: IF sr THEN*
75 IF x THEN x=ROUND(x, LOG(ABS x)-6)
80 PRINT c#: "x="; x; : z#=INPUT$(1,0): GOTO30
500 ON ERROR GOTO510: RETURN
510 IF ERR=1 THEN CLS: ON ERROR GOTO0
520 IF ERR=30 THEN RESUME30
530 LOCATE0,1: PRINT c#: "not found"; : z#=INPUT$(1,0): RESUME20

```

## 3.5.9 6420 Prozentpunkt (x^2 Verteilung)

## LIB 6420 Prozentpunkt (x^2 Verteilung)

```

10 MODE8: DIM: GOSUB500: c#=CHR$(5): n=1: P=0: DEFCHR$(254)="3E02040030"
20 CLS: PRINT "Percentage points X^2(x^2,0)";
30 z=n: s#="n": GOSUB400: n=z: IF n=<0 OR FRAC n<>0 THEN30
40 z=P: s#="P": GOSUB400: P=z: IF P=<0 OR P>1 THEN40
60 LOCATE0,1: PRINT c#: ".....";
70 sr=0: GOSUB"LIB0:S6420": GOSUB500: IF sr THEN*
75 IF x THEN x=ROUND(x, LOG(ABS x)-6)
80 PRINT c#: "x^2="; x; : z#=INPUT$(1,0): GOTO30
400 LOCATE0,1: PRINT c#: s#: "="; z: "?"; : INPUT@10: z: LOCATE0,0: RETURN
500 ON ERROR GOTO510: RETURN
510 IF ERR=1 THEN CLS: ON ERROR GOTO0
520 IF ERR=400 THEN RESUME400
530 LOCATE0,1: PRINT c#: "not found"; : z#=INPUT$(1,0): RESUME20

```

## 3.5.10 6430 Prozentpunkt (t Verteilung)

## LIB 6430 Prozentpunkt (t Verteilung)

```

10 MODE8: DIM: GOSUB500: c#=CHR$(5): n=1: P=0: DEFCHR$(254)="3E02040830"
20 CLS: PRINT "Percentase points  t(x, n)";
30 z=n: s#="n": GOSUB400: n=z: IF n=<0 OR FRACn<>0 THEN 30
40 z=P: s#="P": GOSUB400: P=z: IF P=<0 OR P>=1 THEN 40
60 LOCATE0, 1: PRINT c#: ".....";
70 sr=0: GOSUB"LIB0: S6430": GOSUB500: IF sr THEN*
75 IF x THEN x=ROUND(x, LOG(ABSx)-6)
80 PRINT c#: "x="; x; : z#=INPUT$(1, @): GOTO 30
400 LOCATE0, 1: PRINT c#: s#: "="; z: "?": INPUT@10: z: LOCATE0, 0: RETURN
500 ONERROR GOTO 510: RETURN
510 IFERR=1 THEN CLS: ONERROR GOTO 0
520 IFERL=400 THEN RESUME 400
530 LOCATE0, 1: PRINT c#: "not found": : z#=INPUT$(1, @): RESUME 20

```

## 3.5.11 6440 Prozentpunkt (F Verteilung)

## LIB 6440 Prozentpunkt (F Verteilung)

```

10 MODE8: DIM: GOSUB500: c#=CHR$(5): n=1: m=1: P=0: DEFCHR$(254)="3E02040830"
: DEFCHR$(252)="123E020000": DEFCHR$(253)="2E2A3A0000"
20 CLS: PRINT "Percentase points  F(x, m, n)";
30 z=n: s#="n": GOSUB400: n=z: IF n=<0 OR FRACn<>0 THEN 30
40 z=m: s#="m": GOSUB400: m=z: IF m=<0 OR FRACm<>0 THEN 40
50 z=P: s#="P": GOSUB400: P=z: IF P=<0 OR P>=1 THEN 50
60 LOCATE0, 1: PRINT c#: ".....";
70 sr=0: GOSUB"LIB0: S6440": GOSUB500: IF sr THEN*
75 IF x THEN x=ROUND(x, LOG(ABSx)-6)
80 PRINT c#: "x="; x; : z#=INPUT$(1, @): GOTO 30
400 LOCATE0, 1: PRINT c#: s#: "="; z: "?": INPUT@10: z: LOCATE0, 0: RETURN
500 ONERROR GOTO 510: RETURN
510 IFERR=1 THEN CLS: ONERROR GOTO 0
520 IFERL=400 THEN RESUME 400
530 LOCATE0, 1: PRINT c#: "not found": : z#=INPUT$(1, @): RESUME 20

```

## 3.5.12 6450 Normale Zufallszahlen

## LIB 6450 Normale Zufallszahlen

```

10 MODE8: DIM: CLS: ANGLE 1
20 x=RAN#(1): IF x=0 THEN 20
25 y=RAN#(1): IF y=0 THEN 25
30 x=SQR(-2*LNx): y=2*PI*y
40 z=x*COSy: GOSUB60
50 z=x*SINy: GOSUB60: GOTO 20
60 PRINTTAB(0); z; : z#=INPUT$(1, @): RETURN

```

## 3.5.13 6460 Exponentielle Zufallszahlen

## LIB 6460 Exponentielle Zufallszahlen

```

5 ONERRORGOTO100
10 MODE8: DIM: DEFCHR$(254)="424C300806": k=1
20 CLS: PRINT"E(=,t)";
30 LOCATE0,1: PRINTCHR$(5); "="; k: "?": INPUT@10; k: LOCATE0,0: LOCATE0,1
40 x=RAN#(1): IFx=0 THEN40
50 PRINTTAB(0); -LNx/k; z#=INPUT$(1,@): GOTO40
100 IFERR=1 THENCLS: ONERRORGOTO0
110 RESUME20

```

## 3.5.14 6500 Statistische Berechnungen mit einer Variablen

## LIB 6500 Statistische Berechnungen mit einer Variablen

```

5 ONERRORGOTO1000
10 MODE8: DIM: ERASE a, a#: DIMa(5), a$(1): a$(1)="Input": a$(0)="Delete": c#=CHR$(5)
20 CLS: PRINT"Statistics [x]"; TAB(0); ">In,Del,Clear,List,T-score,P ?";
30 a=ASC(INPUT$(1)): IFa>96 THENa=a-32
40 ONa-66 GOTO300,100,...,100,...,500,...,800,...,400
50 GOTO30
100 CLS: PRINTa$(SGN(a-68)); " data (x)"; TAB(21); "[EXE]:menu"; z=SGN(a-69)
110 LOCATE0,1: PRINTc#; "x?";
120 LOCATE2,1: INPUT@12; z#: LOCATE0,0: IFz#="" THEN20 ELSEx=VALF(z#)
150 r=sa+z: s=sb+x*z: u=sd+x*x*z
160 sa=r: sb=s: sd=u: GOTO110
200 c=6
210 sa=sb/sa
220 sm=sd-sa*sb: IFsm<0 THENsm=0
250 si=SQ R(sm/sa)
260 sk=SQ R(sm/(sa-1))
280 RETURN
300 PRINTc#; "clear data (Y/N) ?";
310 a#=INPUT$(1,@): IFa#<>"Y" ANDa#<>">" THEN20
320 sa=0: sb=0: sd=0: GOTO20
400 GOSUB200: IFc<6 ORsi=0 THEN20
430 LOCATE0,1: PRINTc#; "x?": INPUT@10; z#: LOCATE0,0: IFz#="" THEN20 ELSEs=VALF(z#)
440 LOCATE12,1: PRINT" T=": ROUND((s-sa)/si*10+50,-2); z#=INPUT$(1,@): GOTO430
500 CLS: GOSUB200: GOSUB580: i=1: GOSUB600: i=2
510 GOSUB600
520 z=ASC(INPUT$(1,@)): IFz=28 ORz=29 THEN20
530 IFz<>13 ANDz<>30 ANDz<>31 THEN520
540 IF(z=30 ANDi=1) OR(z=31 ANDi=c) THEN520
550 IFz=30 THENi=i-2
560 i=i+1: IFi<c THEN510 ELSE20
580 DEFCHR$(254)="0008003E00"
590 a(0)=sa: a(1)=sb: a(2)=sd: a(3)=sa: a(4)=si: a(5)=sk: RETURN
600 RESTORE(600+i*10): READa#, b#: PRINTTAB(0); a#: TAB(7); " ": b#: TAB(13); " ": a(i-1): RETURN
610 DATACNT, n
620 DATASUMX, Σx
630 DATASUMX2, Σx²
640 DATAMEANX, Σx/n
650 DATASDXN, x̄n
660 DATASDX, x̄n̄
800 LOCATE0,1: PRINTc#; ".....": GOSUB200: GOSUB590: RESTORE 610
810 FORi=1 TOc: READa#, b#: LPRINTa#: TAB(10); " ": a(i-1): NEXT: GOTO20
1000 IFERR=1 THENCLS: ONERRORGOTO0
1010 IFERR=120 THENRESUME120
1030 IFERR=430 THENRESUME430
1050 IFERR=210 THENc=3: RESUME280
1060 IFERR=260 THENc=5: RESUME280

```



**LIB 6500** Statistische Berechnungen mit einer Variablen

```
1080 IFERL=150 THENLOCATE0,1:PRINTc$;"data over";:z$=INPUT$(1,0):RESUME20
1090 LOCATE0,1:PRINTc$;"not found";:a$=INPUT$(1,0):RESUME20
```

**3.5.15 6510 Lineare Regression ( $y = a + bx$ )****LIB 6510** Lineare Regression ( $y = a + bx$ )

```
10 S9=0:GOTO"LIB0:M6510"
```

**3.5.16 6520 Logarithmische Regression ( $y = a + b \ln x$ )****LIB 6520** Logarithmische Regression ( $y = a + b \ln x$ )

```
10 S9=1:GOTO"LIB0:M6510"
```

**3.5.17 6530 Exponentielle Regression ( $y = ab^x$ )****LIB 6530** Exponentielle Regression ( $y = ab^x$ )

```
10 S9=2:GOTO"LIB0:M6510"
```

**3.5.18 6540 Potenz-Regression ( $y = ax^b$ )****LIB 6540** Potenz-Regression ( $y = ax^b$ )

```
10 S9=3:GOTO"LIB0:M6510"
```

## 3.5.19 6610 Mittelwert-Intervallschätzung (für bekannte Varianz)

## LIB 6610 Mittelwert-Intervallschätzung (für bekannte Varianz)

```

10 MODE8: DIM: GOSUB900: c#=CHR$(5): DEFCHR$(254)="A2A49C92A2": t#="N( $\mu$ ,  $\sigma^2$ ) a< $\mu$ <b
    $\sigma^2$ : known"
20 CLS: PRINT t#: TAB(0): "input new data (Y/N) ?": a#=INPUT$(1, @)
25 IF a#<>"Y" AND a#<>"y" THEN 40
30 s#="x": GOSUB"LIB0:S6020": GOSUB900
40 t=0: f=95: r=sa: IF r THEN s=sa ELSE s=0
45 CLS: PRINT t#:
50 z=r: s#="n": GOSUB500: r=z
60 z=s: s#="s": GOSUB500: s=z
80 z=t: s#="s": GOSUB500: t=z
90 CLS: PRINT "Confidence level (1- $\alpha$ )[%]":
95 z=f: s#="1- $\alpha$ ": GOSUB500: f=z: IF f<0 OR f>100 THEN 95
97 CLS: PRINT "N( $\mu$ ,  $\sigma^2$ )   "; f: "%": TAB(0): ".....":
100 p=(1-f/100)/2: sr=0: GOSUB"LIB0:S6410": GOSUB900: IF sr THEN*
110 z=s-x*t/SQ R r: GOSUB400: a=z
120 z=s+x*t/SQ R r: GOSUB400: b=z
130 PRINT c#: a: "<  $\mu$  <": b: a#=INPUT$(1, @): GOTO45
400 IF z THEN z=ROUND(z, LOG(ABS z)-5)
410 RETURN
500 LOCATE0, 1: PRINT c#: s#: "=": z: "?": INPUT@10: z: LOCATE0, 0: RETURN
900 ON ERROR GOTO 910: RETURN
910 IF ERR=1 THEN CLS: ON ERROR GOTO 0
920 IF ERR=500 THEN RESUME 500
930 LOCATE0, 1: PRINT c#: "not found": z#=INPUT$(1, @): RESUME 20

```

## 3.5.20 6620 Mittelwert-Intervallschätzung (für unbekannte Varianz)

## LIB 6620 Mittelwert-Intervallschätzung (für unbekannte Varianz)

```

10 MODE8: DIM: GOSUB900: c#=CHR$(5): DEFCHR$(254)="A2A49C92A2": t#="N( $\mu$ ,  $\sigma^2$ ) a< $\mu$ <b"
20 CLS: PRINT t#: TAB(0): "input new data (Y/N) ?": a#=INPUT$(1, @)
25 IF a#<>"Y" AND a#<>"y" THEN 40
30 s#="x": GOSUB"LIB0:S6020": GOSUB900
40 f=95: r=sa: IF r THEN s=sa: t=s*k^2 ELSE s=0: t=0
45 CLS: PRINT t#:
50 z=r: s#="n": GOSUB500: r=z
60 z=s: s#="s": GOSUB500: s=z
80 z=t: s#="U": GOSUB500: t=z
90 CLS: PRINT "Confidence level (1- $\alpha$ )[%]":
95 z=f: s#="1- $\alpha$ ": GOSUB500: f=z: IF f<0 OR f>100 THEN 95
97 CLS: PRINT "N( $\mu$ ,  $\sigma^2$ )   "; f: "%": TAB(0): ".....":
100 n=r-1: p=(1-f/100)/2: sr=0: GOSUB"LIB0:S6430": GOSUB900: IF sr THEN*
110 z=s-x*SQ R (t/r): GOSUB400: a=z
120 z=s+x*SQ R (t/r): GOSUB400: b=z
130 PRINT c#: a: "<  $\mu$  <": b: a#=INPUT$(1, @): GOTO45
400 IF z THEN z=ROUND(z, LOG(ABS z)-5)
410 RETURN
500 LOCATE0, 1: PRINT c#: s#: "=": z: "?": INPUT@10: z: LOCATE0, 0: RETURN
900 ON ERROR GOTO 910: RETURN
910 IF ERR=1 THEN CLS: ON ERROR GOTO 0
920 IF ERR=500 THEN RESUME 500
930 LOCATE0, 1: PRINT c#: "not found": z#=INPUT$(1, @): RESUME 20

```

## 3.5.21 6630 Varianz-Intervallschätzung

## LIB 6630 Varianz-Intervallschätzung

```

10 MODE8: DIM: GOSUB900: c#=CHR$(5): t#="N( $\mu$ ,  $\sigma^2$ ) a< $\sigma^2$ <b"
20 CLS: PRINT t#: TAB(0): "input new data (Y/N) ?": a#=INPUT$(1, @)
25 IF a#<>"Y" AND a#<>"y" THEN 40
30 s#="x": GOSUB"LIB0:S6020": GOSUB900
40 f=95: s=sa: IF s THEN r=sm ELSE r=0
45 CLS: PRINT t#:
50 z=s: s#="n": GOSUB500: s=z
60 z=r: s#="S": GOSUB500: r=z
90 CLS: PRINT "Confidence level (1- $\alpha$ )[%]":
95 z=f: s#="1- $\alpha$ ": GOSUB500: f=z: IF f<0 OR f>100 THEN 95
97 CLS: PRINT "N( $\mu$ ,  $\sigma^2$ )   ": f: "%": TAB(0): ".....":
100 n=s-1: F=(1-f/100)/2: sr=0: GOSUB"LIB0:S6420": GOSUB900: IF sr THEN*
105 g=x: F=1-F: GOSUB"LIB0:S6420": GOSUB900: IF sr THEN*
110 z=r/g: GOSUB400: a=z
120 z=r/x: GOSUB400: b=z
130 PRINT c#: a: "<  $\sigma^2$  <": b: a#=INPUT$(1, @): GOTO 45
400 IF z THEN z=ROUND(z, LOG(ABS z)-5)
410 RETURN
500 LOCATE0, 1: PRINT c#: s#: "=": z: "?": INPUT@10: z: LOCATE0, 0: RETURN
900 ONERROR GOTO 910: RETURN
910 IFERR=1 THEN CLS: ONERROR GOTO 0
920 IFERL=500 THEN RESUME 500
930 LOCATE0, 1: PRINT c#: "not found": z#=INPUT$(1, @): RESUME 20

```

## 3.5.22 6640 Standardabweichung-Intervallschätzung

## LIB 6640 Standardabweichung-Intervallschätzung

```

10 MODE8: DIM: GOSUB900: c#=CHR$(5): t#="N( $\mu$ ,  $\sigma$ ) a< $\sigma$ <b"
20 CLS: PRINT t#: TAB(0): "input new data (Y/N) ?": a#=INPUT$(1, @)
25 IF a#<>"Y" AND a#<>"y" THEN 40
30 s#="x": GOSUB"LIB0:S6020": GOSUB900
40 f=95: s=sa: IF s THEN r=sm ELSE r=0
45 CLS: PRINT t#:
50 z=s: s#="n": GOSUB500: s=z
60 z=r: s#="S": GOSUB500: r=z
90 CLS: PRINT "Confidence level (1- $\alpha$ )[%]":
95 z=f: s#="1- $\alpha$ ": GOSUB500: f=z: IF f<0 OR f>100 THEN 95
97 CLS: PRINT "N( $\mu$ ,  $\sigma$ )   ": f: "%": TAB(0): ".....":
100 n=s-1: F=(1-f/100)/2: sr=0: GOSUB"LIB0:S6420": GOSUB900: IF sr THEN*
105 g=x: F=1-F: GOSUB"LIB0:S6420": GOSUB900: IF sr THEN*
110 z=SQR(r/g): GOSUB400: a=z
120 z=SQR(r/x): GOSUB400: b=z
130 PRINT c#: a: "<  $\sigma$  <": b: a#=INPUT$(1, @): GOTO 45
400 IF z THEN z=ROUND(z, LOG(ABS z)-5)
410 RETURN
500 LOCATE0, 1: PRINT c#: s#: "=": z: "?": INPUT@10: z: LOCATE0, 0: RETURN
900 ONERROR GOTO 910: RETURN
910 IFERR=1 THEN CLS: ONERROR GOTO 0
920 IFERL=500 THEN RESUME 500
930 LOCATE0, 1: PRINT c#: "not found": z#=INPUT$(1, @): RESUME 20

```

## 3.5.23 6650 Varianzquotient-Intervallschätzung

## LIB 6650 Varianzquotient-Intervallschätzung

```

10 MODE8: DIM: GOSUB900: c#=CHR#(5): DEFCHR#(252)="123E020000"
: DEFCHR#(253)="2E2A3A0000": t#="N(μ̄, σ̄²), N(μ̄, σ̄²) a<σ̄²/σ̄²<b"
20 CLS: PRINT t#: TAB(0): "input new data x# (Y/N) ?": a#=INPUT$(1,0)
22 IF a#<>"Y" AND a#<>"y" THEN 30
25 s#="x#": GOSUB"LIB0:S6020": GOSUB900
30 CLS: PRINT t#: TAB(0): "input new data x# (Y/N) ?": a#=INPUT$(1,0)
32 IF a#<>"Y" AND a#<>"y" THEN 36
35 s#="x#": GOSUB"LIB0:S6030": GOSUB900
36 s=s# IF s>1 THEN r=sk^2 ELSE r=0
37 f=95: u=sz: IF u>1 THEN t=s1^2 ELSE t=0
45 CLS: PRINT t#:
50 z=s: s#="n#": GOSUB500: s=z
60 z=r: s#="U#": GOSUB500: r=z
70 z=u: s#="n#": GOSUB500: u=z
80 z=t: s#="U#": GOSUB500: t=z
90 CLS: PRINT "Confidence level (1-α) [%]";
95 z=f: s#="1-α": GOSUB500: f=z: IF f<0 OR f>100 THEN 95
97 CLS: PRINT "N(μ̄, σ̄²), N(μ̄, σ̄²) "; f: "%": TAB(0): ".....";
100 n=s-1: m=u-1: F=(1-f/100)/2: sr=0: GOSUB"LIB0:S6440": GOSUB900: IF sr THEN*
105 s=x: n=m: m=s-1: GOSUB"LIB0:S6440": GOSUB900: IF sr THEN*
110 z=t/r/x: GOSUB400: a=z
120 z=t/r*s: GOSUB400: b=z
130 PRINT c#: a: "< σ̄²/σ̄² <": b: a#=INPUT$(1,0): GOTO45
400 IF z THEN z=ROUND(z, LOG(ABS z)-5)
410 RETURN
500 LOCATE0, 1: PRINT c#: s#: "=": z: "?": INPUT@10: z: LOCATE0, 0: RETURN
900 ONERROR GOTO 910: RETURN
910 IF ERR=1 THEN CLS: ONERROR GOTO 0
920 IF ERR=500 THEN RESUME 500
930 LOCATE0, 1: PRINT c#: "not found": z#=INPUT$(1,0): RESUME 20

```

## 3.5.24 6660 Mittelwertdifferenz-Intervallschätzung

## LIB 6660 Mittelwertdifferenz-Intervallschätzung

```

10 MODE8: DIM: GOSUB900: c#=CHR$(5): DEFCHR$(252)="123E020000"
: DEFCHR$(253)="2E2A3A0000": DEFCHR$(254)="A2A49C92A2": t#="N(μ̄, σ²), N(μ̄, σ²) a<μ̄-
μ̄<b"
20 CLS: PRINT t#: TAB(0): "input new data x# (Y/N) ?": a#=INPUT$(1, @)
22 IF a#<>"Y" AND a#<>"y" THEN 30
25 s#="x#": GOSUB"LIB0: S6020": GOSUB900
30 CLS: PRINT t#: TAB(0): "input new data x# (Y/N) ?": a#=INPUT$(1, @)
32 IF a#<>"Y" AND a#<>"y" THEN 36
35 s#="x#": GOSUB"LIB0: S6030": GOSUB900
36 r=sa: IF r>1 THEN s=sa: t=sm ELSE s=0: t=0
37 f=95: u=sz: IF u>1 THEN v=sh: w=sn ELSE v=0: w=0
45 CLS: PRINT t#:
50 z=r: s#="n#": GOSUB500: r=z
55 z=s: s#="m#": GOSUB500: s=z
60 z=t: s#="s#": GOSUB500: t=z
70 z=u: s#="h#": GOSUB500: u=z
75 z=v: s#="sh#": GOSUB500: v=z
80 z=w: s#="sn#": GOSUB500: w=z
90 CLS: PRINT "Confidence level (1-α) [%]":
95 z=f: s#="1-α": GOSUB500: f=z: IF f<0 OR f>100 THEN 95
97 CLS: PRINT "N(μ̄, σ²), N(μ̄, σ²)   ": f: "%": TAB(0): ".....":
100 n=r+u-2: e=(1-f/100)/2: sr=0: GOSUB"LIB0: S6430": GOSUB900: IF sr THEN*
110 z=s-v-x*SQR((1/r+1/u)*((t+w)/n)): GOSUB400: a=z
120 z=s+v+x*SQR((1/r+1/u)*((t+w)/n)): GOSUB400: b=z
130 PRINT c#: a: "< μ̄-μ̄ <": b: a#=INPUT$(1, @): GOTO45
400 IF z THEN z=ROUND(z, LOG(ABS z)-5)
410 RETURN
500 LOCATE0, 1: PRINT c#: s#: "=": z: "?": INPUT@10: z: LOCATE0, 0: RETURN
900 ONERROR GOTO 910: RETURN
910 IF ERR=1 THEN CLS: ONERROR GOTO 0
920 IF ERR=500 THEN RESUME 500
930 LOCATE0, 1: PRINT c#: "not found": z#=INPUT$(1, @): RESUME 20

```

## 3.5.25 6670 Verhältnis-Intervallschätzung

## LIB 6670 Verhältnis-Intervallschätzung

```

10 MODE8: DIM: GOSUB900: c#=CHR$(5): f=95: r=0: s=0
45 CLS: PRINT "B(1, p) a<p<b":
50 z=r: s#="n": GOSUB500: r=z
60 z=s: s#="Σx": GOSUB500: s=z
90 CLS: PRINT "Confidence level (1-α) [%]":
95 z=f: s#="1-α": GOSUB500: f=z: IF f<0 OR f>100 THEN 95
97 CLS: PRINT "B(1, p)   ": f: "%": TAB(0): ".....":
100 p=(1-f/100)/2: sr=0: GOSUB"LIB0: S6410": GOSUB900: IF sr THEN*
110 e=s/r: z=e-x*SQR(((1-e)*e)/r): GOSUB400: a=z
120 z=e+x*SQR(((1-e)*e)/r): GOSUB400: b=z
130 PRINT c#: a: "< p <": b: a#=INPUT$(1, @): GOTO45
140 GOTO 50
400 IF z THEN z=ROUND(z, LOG(ABS z)-5)
410 RETURN
500 LOCATE0, 1: PRINT c#: s#: "=": z: "?": INPUT@10: z: LOCATE0, 0: RETURN
900 ONERROR GOTO 910: RETURN
910 IF ERR=1 THEN CLS: ONERROR GOTO 0
920 IF ERR=500 THEN RESUME 500
930 LOCATE0, 1: PRINT c#: "not found": z#=INPUT$(1, @): RESUME 45

```

## 3.5.26 6680 Verhältnisdifferenz-Intervallschätzung

**LIB 6680** Verhältnisdifferenz-Intervallschätzung

```

10 MODE8: DIM: GOSUB900: c#=CHR$(5): f=95: DEFCHR$(252)="123E020000"
: DEFCHR$(253)="2E2A3A0000": r=0: s=0: t=0: u=0
45 CLS: PRINT "B(1,P#),B(1,P#) a<P#-P#<b";
50 z=r: s#="n#": GOSUB500: r=z
60 z=s: s#="Σx#": GOSUB500: s=z
70 z=t: s#="n#": GOSUB500: t=z
80 z=u: s#="Σx#": GOSUB500: u=z
90 CLS: PRINT "Confidence level (1-α) [%]";
95 z=f: s#="1-α": GOSUB500: f=z: IF f<0 OR f>100 THEN95
97 CLS: PRINT "B(1,P#),B(1,P#) "; f: "%"; TAB(0); ".....";
100 P#=(1-f/100)/2: sr=0: GOSUB"LIB0:S6410": GOSUB900: IF sr THEN*
110 j=s/r: k=u/t: z=j-k-x*SQ R(((1-j)*j)/r+((1-k)*k)/t): GOSUB400: a=z
120 z=j-k+x*SQ R(((1-j)*j)/r+((1-k)*k)/t): GOSUB400: b=z
130 PRINT c#: a: "< P#-P# <": b: a#=INPUT$(1, @): GOTO45
400 IF z THEN z=ROUND(z, LOG(ABS z)-5)
410 RETURN
500 LOCATE0, 1: PRINT c#: s#: "="; z: "?": INPUT@10: z: LOCATE0, 0: RETURN
900 ONERROR GOTO910: RETURN
910 IF ERR=1 THEN CLS: ONERROR GOTO0
920 IF ERR=500 THEN RESUME500
930 LOCATE0, 1: PRINT c#: "not found": z#=INPUT$(1, @): RESUME45

```

## 3.5.27 6710 Grundgesamtheits-Mittelwert-Test (zweiseitig): für bekannte Varianz

**LIB 6710** Grundgesamtheits-Mittelwert-Test (zweiseitig): für bekannte Varianz

```

10 s# = 226: GOSUB "LIB0: M6710"

```

## 3.5.28 6711 Grundgesamtheits-Mittelwert-Test (rechtsseitig): für bekannte Varianz

**LIB 6711** Grundgesamtheits-Mittelwert-Test (rechtsseitig): für bekannte Varianz

```

10 s# = 62: GOSUB "LIB0: M6710"

```

## 3.5.29 6712 Grundgesamtheits-Mittelwert-Test (linksseitig): für bekannte Varianz

**LIB 6712** Grundgesamtheits-Mittelwert-Test (linksseitig): für bekannte Varianz

```

10 s# = 60: GOSUB "LIB0: M6710"

```

**3.5.30 6720 Grundgesamtheits-Mittelwert-Test (zweiseitig): für unbekannte Varianz**

<b>LIB 6720</b>	Grundgesamtheits-Mittelwert-Test (zweiseitig): für unbekannte Varianz
-----------------	---

10 S9=226:GOSUB"[LIB0:M6720](#)"

**3.5.31 6721 Grundgesamtheits-Mittelwert-Test (rechtsseitig): für unbekannte Varianz**

<b>LIB 6721</b>	Grundgesamtheits-Mittelwert-Test (rechtsseitig): für unbekannte Varianz
-----------------	---

10 S9=62:GOSUB"[LIB0:M6720](#)"

**3.5.32 6722 Grundgesamtheits-Mittelwert-Test (linksseitig): für unbekannte Varianz**

<b>LIB 6722</b>	Grundgesamtheits-Mittelwert-Test (linksseitig): für unbekannte Varianz
-----------------	--

10 S9=60:GOSUB"[LIB0:M6720](#)"

**3.5.33 6730 Grundgesamtheits-Varianz-Test (zweiseitig)**

<b>LIB 6730</b>	Grundgesamtheits-Varianz-Test (zweiseitig)
-----------------	--

10 S9=226:GOSUB"[LIB0:M6730](#)"

**3.5.34 6731 Grundgesamtheits-Varianz-Test (rechtsseitig)**

<b>LIB 6731</b>	Grundgesamtheits-Varianz-Test (rechtsseitig)
-----------------	--

10 S9=62:GOSUB"[LIB0:M6730](#)"

**3.5.35 6732 Grundgesamtheits-Varianz-Test (linksseitig)**

<b>LIB 6732</b>	Grundgesamtheits-Varianz-Test (linksseitig)
-----------------	---

10 S9=60:GOSUB"[LIB0:M6730](#)"

**3.5.36 6740 Varianzquotient-Test (zweiseitig)****LIB 6740** Varianzquotient-Test (zweiseitig)10 `SA=226:GOSUB"LIB0:M6740"`**3.5.37 6741 Varianzquotient-Test (rechtsseitig)****LIB 6741** Varianzquotient-Test (rechtsseitig)10 `SA=62:GOSUB"LIB0:M6740"`**3.5.38 6742 Varianzquotient-Test (linksseitig)****LIB 6742** Varianzquotient-Test (linksseitig)10 `SA=60:GOSUB"LIB0:M6740"`**3.5.39 6750 Mittelwertdifferenz-Test (zweiseitig)****LIB 6750** Mittelwertdifferenz-Test (zweiseitig)10 `SA=226:GOSUB"LIB0:M6750"`**3.5.40 6751 Mittelwertdifferenz-Test (rechtsseitig)****LIB 6751** Mittelwertdifferenz-Test (rechtsseitig)10 `SA=62:GOSUB"LIB0:M6750"`**3.5.41 6752 Mittelwertdifferenz-Test (linksseitig)****LIB 6752** Mittelwertdifferenz-Test (linksseitig)10 `SA=60:GOSUB"LIB0:M6750"`**3.5.42 6760 Quotiententest (zweiseitig)****LIB 6760** Quotiententest (zweiseitig)10 `SA=226:GOSUB"LIB0:M6760"`



**3.5.43 6761 Quotiententest (rechtsseitig)****LIB 6761** Quotiententest (rechtsseitig)10 `S9=62:GOSUB"LIB0:M6760"`**3.5.44 6762 Quotiententest (linksseitig)****LIB 6762** Quotiententest (linksseitig)10 `S9=60:GOSUB"LIB0:M6760"`**3.5.45 6770 Quotientendifferenztest (zweiseitig)****LIB 6770** Quotientendifferenztest (zweiseitig)10 `S9=226:GOSUB"LIB0:M6770"`**3.5.46 6771 Quotientendifferenztest (rechtsseitig)****LIB 6771** Quotientendifferenztest (rechtsseitig)10 `S9=62:GOSUB"LIB0:M6770"`**3.5.47 6772 Quotientendifferenztest (linksseitig)****LIB 6772** Quotientendifferenztest (linksseitig)10 `S9=60:GOSUB"LIB0:M6770"`

## 3.6 Statistik Unterprogramme

### 3.6.1 M6510 Unterprogramm der Statistik-Programme 6510, 6520, 6530, 6540

#### LIB M6510 Unterprogramm der Statistik-Programme [6510](#), [6520](#), [6530](#), [6540](#)

```

5  ONERRORGOTO1000:RESTORE :MODE0:DIM:ERASEa,a#:DIMa(14),a#(5):c#=CHR#(5)
: IFsq<0ORtz>3ORFRACsq<>0THENSq=0
12 IFtz<0ORTz>3ORFRACtz<>0THENTz=0
14 FORi=0TO5:READa#(i):NEXT:d#="" :y?"
15 DATAy=a+bx,y=a+blnx,y=ab^x,y=ax^b,Delete,Input
20 l=21+ABS(SGN(sq-1))*2:CLS:PRINT"Reession
analysis":TAB(1):"[";a#(sq):"]":TAB(0):">In,Del,Clear,List,exX,exY,P ?":
30 IFINKEY#="" THENa=ASC(INPUT#(1)):IFA>96THENA=a-32
40 IFA=186THENSq=(sq+1)MOD4:GOTO20
50 ONa-66GOTO300,100,.,.,100
: IF(sq ANDtz)=0THENONa-75GOTO500,.,.,800,.,.,.,400,450
60 GOTO30
100 CLS:PRINTa#(SGN(a-68)+4):" data (x,y)":TAB(21):"[EXE]:menu":z=SGN(a-69)
110 LOCATE0,1:PRINTc#:"x?":d#:
120 LOCATE2,1:INPUT@12;z#:LOCATE0,0:IFz#="" ORASC(z#)=0THEN20ELSEx=VALF(z#):j=0
125 IFx=<0THENIF(1ANDsq)=1THEN120ELSEtz=tz OR1ELSEIFx>0THENj=LNx:q=ss+j*z
:m=su+j*k*z
130 r=sa+z:s=sb+x*z:u=sd+x*x*z
140 LOCATE18,1:INPUT@12;z#:LOCATE0,0:IFz#="" THEN20ELSEy=VALF(z#)
145 IFy=<0THENIFsq=1THEN140ELSEtz=tz OR2ELSE
IFY>0THENk=LNy:h=st+k*z:n=sv+k*k*z:p=sx+x*k*z:q=sy+j*k*z
160 t=sc+y*z:v=se+y*y*z:w=sf+x*y*z:o=sw+j*y*z:sa=r:sb=s:sc=t:sd=u:se=v:sf=w:ss=q
:st=h:su=m:sv=n:sw=o:sx=p:sy=q:GOTO110
200 c=15:GOSUB350:ONSq GOSUB360,370,380
210 sa=a(1)/sa:sh=a(2)/sa
220 sm=a(3)-sa*a(1):IFsm<0THENsm=0
230 sn=a(4)-sh*a(2):IFsn<0THENsn=0
240 so=a(5)-a(1)*sh:sp=sh*a(3)-sa*a(5):si=SQ R(sm/sa):sj=SQ R(sn/sa)
260 sk=SQ R(sm/(sa-1)):sl=SQ R(sn/(sa-1))
270 P=sp/sm:q=so/sm:r=so/SQ R(sm*sn):IFsq=1THENP=EXPp:IFsq=2THENq=EXPq
280 RETURN
300 PRINTc#:"clear data (Y/N) ?":a#=INPUT#(1,@)
310 IFA#="Y"ORa#="y"THENSa=0:sb=0:sc=0:sd=0:se=0:sf=0:ss=0:st=0:su=0:sv=0:sw=0
:sx=0:sy=0:tz=0
320 GOTO20
350 a(1)=sb:a(2)=sc:a(3)=sd:a(4)=se:a(5)=sf:RETURN
360 a(1)=ss:a(3)=su:a(5)=sw:RETURN
370 a(2)=st:a(4)=sv:a(5)=sx:RETURN
380 a(1)=ss:a(2)=st:a(3)=su:a(4)=sv:a(5)=sy:RETURN
400 CLS:GOSUB200:IFc<15ORso=0THEN20
410 DEFCHR#(255)="12548C4A12":CLS:PRINT"Estimation of x [";a#(sq):"]":
420 LOCATE0,1:PRINTc#:"y?":INPUT@10;z#:LOCATE0,0:IFz#="" THEN20ELSEs=VALF(z#)
430 ONSq+1GOSUB950,960,970,980:LOCATE12,1:PRINT":=":STR#(a):z#=INPUT#(1,@)
:GOTO420
450 CLS:GOSUB200:IFc<15ORsm=0THEN20
460 DEFCHR#(255)="124A844810":CLS:PRINT"Estimation of y [";a#(sq):"]":
470 LOCATE0,1:PRINTc#:"x?":INPUT@10;z#:LOCATE0,0:IFz#="" THEN20ELSEs=VALF(z#)
480 ONSq+1GOSUB900,910,920,930:LOCATE12,1:PRINT":=":STR#(a):z#=INPUT#(1,@)
:GOTO470
500 CLS:GOSUB200:GOSUB580:i=1:GOSUB600:i=2
510 GOSUB600
520 z=ASC(INPUT#(1,@)):IFz=28ORz=29THEN20
530 IFz<>13ANDz<>30ANDz<>31THEN520
540 IF(z=30ANDi=1)OR(z=31ANDi=c)THEN520
550 IFz=30THENi=i-2
560 i=i+1:IFI=c THEN510ELSE20
580 DEFCHR#(254)="0008003E00":DEFCHR#(253)="FE003E203E"

```

**LIB M6510** Unterprogramm der Statistik-Programme  
[6510](#), [6520](#), [6530](#), [6540](#)

```

590 a(0)=sa:a(6)=sq:a(7)=sh:a(8)=si:a(9)=sj:a(10)=sk:a(11)=sl:a(12)=p
:a(13)=q:a(14)=r:RETURN
600 GOSUB605:READa#:a#,b#:PRINTTAB(0):a#:TAB(7):":":b#:TAB(13):":":a(i-1):RETURN
605 IFi=1ORi=15THENj=0:GOTO609
606 IFi=6THENj=sq:GOTO609
607 IFi>12THENj=SGN(sq):GOTO609
608 k=i MOD2:IF(i<6ANDk=1)OR(i>6ANDk=0)THENj=INT(sq/2)ELSEj=(sq MOD2)
609 RESTORE(600+i*10+j):RETURN
610 DATACNT,CNT,n
620 DATASUMX,SUMX,Σx
621 DATASUMLNX,SUM=X,Σx²
630 DATASUMY,SUMY,Σy
631 DATASUMLNY,SUM=Y,Σy²
640 DATASUMX2,SUMX2,Σx²
641 DATASUMLNX2,SUM=X2,Σx²²
650 DATASUMY2,SUMY2,Σy²
651 DATASUMLNY2,SUM=Y2,Σy²²
660 DATASUMXY,SUMXY,Σxy
661 DATASUMLNXY,SUM=XY,Σxy²
662 DATASUMXLNY,SUM=X*Y,Σx*y
663 DATASUMLNXLNY,SUM=X*Y,Σx*y²
670 DATAMEANX,MEANX,Σx/n
671 DATAMEANLNX,MEAN=X,Σx²/n
680 DATAMEANY,MEANY,Σy/n
681 DATAMEANLNY,MEAN=Y,Σy²/n
690 DATASDXN,SDXN,xσn
691 DATASDLNXN,SD=XN,xσn
700 DATASDYN,SDYN,yσn
701 DATASDLNYN,SD=YN,yσn
710 DATASDX,SDX,xσn#
711 DATASDLNX,SD=X,xσn#
720 DATASDY,SDY,yσn#
721 DATASDLNY,SD=Y,yσn#
730 DATALRA,LRA,a
731 DATARA,RA,a
740 DATALRB,LRB,b
741 DATARB,RB,b
750 DATACOR,COR,r
800 LOCATE0,1:PRINTc#:". . . . .":GOSUB200:GOSUB590
810 FORi=1TOc:GOSUB605:READa#:LPRINTa#:TAB(10):":":a(i-1):NEXT:GOTO20
900 a=p+q*s:RETURN
910 a=p+q*LnS:RETURN
920 a=p*q^s:RETURN
930 a=p*s^q:RETURN
950 a=(s-p)/q:RETURN
960 a=EXP((s-p)/q):RETURN
970 a=(LnS-LNp)/LNq:RETURN
980 a=EXP((LnS-LNp)/q):RETURN
1000 IFERR=1THENCLS:ONERRORGOTO0
1010 IFERL=120THENRESUME120
1020 IFERL=140THENRESUME140
1030 IFERL=420THENRESUME420
1040 IFERL=470THENRESUME470
1050 IFERL=210THENc=6:RESUME280
1060 IFERL=260THENc=10:RESUME280
1070 IFERL=270THENc=12:RESUME280
1080 IF(ERL=130ORERL=150)ANDERR=13THENLOCATE0,1:PRINTc#:"data
over":z#=INPUT$(1,0):RESUME20
1090 LOCATE0,1:PRINTc#:"not found":a#=INPUT$(1,0):RESUME20

```

## 3.6.2 M6710 Unterprogramm der Statistik-Programme 6710, 6711, 6712

**LIB M6710** Unterprogramm der Statistik-Programme  
[6710](#), [6711](#), [6712](#)

```

5  IFsq<>60 ANDsq<>62 THENsq=226
10  MODE8: DIM: GOSUB900: ERASEb$: DIMb$(1): b$(0)="Reject": b$(1)="Accept"
: t$="Test   H0: μ=μ0   H1: μ+CHR$(sq)+"μ1": DEFCHR$(254)="A2A49C92A2"
: c$=CHR$(5): DEFCHR$(252)="123E020000"
20  CLS: PRINTt$: TAB(0): "input new data (Y/N) ?": a$=INPUT$(1)
25  IFa$<>"Y" ANDa$<>"y" THEN40
30  s$="x": GOSUB"LIB0:S6020": GOSUB900
40  r=0: s=0: f=5: t=sa: IFt THENu=sa ELSEu=0
45  CLS: PRINTt$:
50  z=r: s$="μ0": GOSUB500: r=z
60  z=s: s$="σ": GOSUB500: s=z
70  z=t: s$="n": GOSUB500: t=z
80  z=u: s$="α": GOSUB500: u=z
90  CLS: PRINT"Significance level α[%]":
95  z=f: s$="α": GOSUB500: f=z: IFf<0 ORf>100 THEN95
97  CLS: PRINTt$: TAB(0): ".....":
100  P=f/100: IFsq=226 THENP=P/2
105  sr=0: GOSUB"LIB0:S6410": GOSUB900: IFsr THEN*
110  b=1: z=(u-r)/(s/SQ Rt): GOSUB400: a=z: z=x: GOSUB400: x=z
120  IFsq=226 THENa=ABSa
121  IF(sq=62 ORsq=226) ANDa>x THENb=0
122  IFsq=60 THENx=-x: IFa<x THENb=0
130  z=SGN(sq-60): PRINTc$: a: CHR$(60+z*2+b*(164-z)): x: "":
: b$(b): a$=INPUT$(1,@): GOTO45
400  IFz THENz=ROUND(z, LOG(ABSz)-5)
410  RETURN
500  LOCATE0, 1: PRINTc$: s$: "=": z: "?": INPUT@10: z: LOCATE0, 0: RETURN
900  ONERRORGOTO910: RETURN
910  IFERR=1 THENCLS: ONERRORGOTO0
920  IFERL=500 THENRESUME500
930  LOCATE0, 1: PRINTc$: "not found": z$=INPUT$(1,@): RESUME20

```

## 3.6.3 M6720 Unterprogramm der Statistik-Programme 6720, 6721, 6722

**LIB M6720** Unterprogramm der Statistik-Programme  
[6720](#), [6721](#), [6722](#)

```

5  IFs9<>60 ANDs9<>62 THENs9=226
10  MODE8: DIM: GOSUB900: ERASEb$: DIMb$(1): b$(0)="Reject": b$(1)="Accept"
: t$="Test   H0: μ=μ0   H1: μ+CHR$(s9)+"μ1": DEFCHR$(254)="A2A49C92A2"
: c$=CHR$(5): DEFCHR$(252)="123E020000"
20  CLS: PRINTt$: TAB(0): "input new data (Y/N) ?": a$=INPUT$(1)
25  IFa$<>"Y" ANDa$<>"y" THEN40
30  s$="x": GOSUB"LIB0:S6020": GOSUB900
40  r=0: f=5: s=sa: IFs>1 THENt=s9: u=sk^2 ELSEt=0: u=0
45  CLS: PRINTt$:
50  z=r: s$="μ0": GOSUB500: r=z
60  z=s: s$="n": GOSUB500: s=z
70  z=t: s$="σ": GOSUB500: t=z
80  z=u: s$="U": GOSUB500: u=z
90  CLS: PRINT"Significance level α[%]":
95  z=f: s$="α": GOSUB500: f=z: IFf<0 ORf>100 THEN95
97  CLS: PRINTt$: TAB(0): ".....":
100 n=s-1: P=f/100: IFs9=226 THENP=P/2
105 sr=0: GOSUB"LIB0:S6430": GOSUB900: IFsr THEN*
110 b=1: z=(t-r)/SQ R(u/s): GOSUB400: a=z: z=x: GOSUB400: x=z
120 IFs9=226 THENa=ABSa
121 IF(s9=62 ORs9=226) ANDa>x THENb=0
122 IFs9=60 THENx=-x: IFa<x THENb=0
130 z=SGN(s9-60): PRINTc$: a: CHR$(60+z*2+b*(164-z)): x: " ": b$(b):
: a$=INPUT$(1,0): GOTO45
400 IFz THENz=ROUND(z, LOG(ABSz)-5)
410 RETURN
500 LOCATE0, 1: PRINTc$: s$: "=": z: "?": INPUT@10: z: LOCATE0, 0: RETURN
900 ONERRORGOTO910: RETURN
910 IFERR=1 THENCLS: ONERRORGOTO0
920 IFERR=500 THENRESUME500
930 LOCATE0, 1: PRINTc$: "not found": z$=INPUT$(1,0): RESUME20

```

## 3.6.4 M6730 Unterprogramm der Statistik-Programme 6730, 6731, 6732

**LIB M6730** Unterprogramm der Statistik-Programme  
[6730](#), [6731](#), [6732](#)

```

5  IF S9<>60 AND S9<>62 THEN S9=226
10  MODE8: DIM: GOSUB900: ERASE b$: DIM b$(1): b$(0)="Reject": b$(1)="Accept"
: t$="Test      H0: σ²=σ0²   H1: σ² "+CHR$(S9)+"σ0²"
: c$=CHR$(5): DEFCHR$(252)="123E020000"
20  CLS: PRINT t$: TAB(0): "input new data (Y/N) ?": a$=INPUT$(1)
25  IF a$<>"Y" AND a$<>"y" THEN 40
30  s$="x": GOSUB"LIB0:S6020": GOSUB900
40  r=0: f=5: s=s0: IF s>1 THEN t=sm ELSE t=0
45  CLS: PRINT t$:
50  z=r: s$="σ0²": GOSUB500: r=z
60  z=s: s$="n": GOSUB500: s=z
70  z=t: s$="S": GOSUB500: t=z
90  CLS: PRINT "Significance level α[%]":
95  z=f: s$="α": GOSUB500: f=z: IF f<0 OR f>100 THEN 95
97  CLS: PRINT t$: TAB(0): ".....":
100 n=s-1: P=f/100: IF S9=226 THEN P=P/2 ELSE IF S9=60 THEN P=1-P
102 a=0: sr=0: GOSUB"LIB0:S6420": GOSUB900: IF sr THEN*
104 IF S9=226 THEN a=x: P=1-P: GOSUB"LIB0:S6420": GOSUB900: IF sr THEN* ELSE b=x: x=a: a=b
110 b=1: z=t/r: GOSUB400: a=z: z=x: GOSUB400: x=z: z=a: GOSUB400: a=z
120 IF (S9=226 OR S9=62) AND a>x THEN b=0
122 IF S9=60 AND a<x THEN b=0
123 IF S9=226 AND a<a THEN c=0 ELSE c=1
130 z=SGN(S9-60): PRINT c$: a: CHR$(60+z*2+b*(164-z)): x:
: IF S9<63 THEN PRINT " ": b$(b): GOTO150
140 PRINT TAB(0): a: CHR$(60+c*164): a: " ": b$(b AND c):
150 a$=INPUT$(1, @): GOTO45
400 IF z THEN z=ROUND(z, LOG(ABS z)-5)
410 RETURN
500 LOCATE0, 1: PRINT c$: s$: "=": z: "?": INPUT@10: z: LOCATE0, 0: RETURN
900 ON ERROR GOTO 910: RETURN
910 IF ERR=1 THEN CLS: ON ERROR GOTO 0
920 IF ERR=500 THEN RESUME 500
930 LOCATE0, 1: PRINT c$: "not found": z$=INPUT$(1, @): RESUME 20

```

## 3.6.5 M6740 Unterprogramm der Statistik-Programme 6740, 6741, 6742

**LIB M6740** Unterprogramm der Statistik-Programme  
[6740](#), [6741](#), [6742](#)

```

5  IFs9<>60 ANDs9<>62 THENs9=226
10  MODE8: DIM: GOSUB900: ERASEb$: DIMb$(1): b$(0)="Reject": b$(1)="Accept"
: t$="Test      H0: σ²=σ²     H1: σ²≠ " +CHR$(s9)+"σ² "
: c$=CHR$(5): DEFCHR$(252)="123E020000": DEFCHR$(253)="2E2A3A0000"
20  CLS:PRINTt$:TAB(0):"input new data x̄ (Y/N) ?":a$=INPUT$(1)
22  IFa$<>"Y" ANDa$<>"y" THEN30
25  s$="x": GOSUB"LIB0:S6020": GOSUB900
30  CLS:PRINTt$:TAB(0):"input new data x̄ (Y/N) ?":a$=INPUT$(1)
32  IFa$<>"Y" ANDa$<>"y" THEN36
35  s$="x": GOSUB"LIB0:S6030": GOSUB900
36  s=sa: IFs>1 THENr=sk^2 ELSEr=0
37  f=5: u=sz: IFu>1 THENt=s1^2 ELSEt=0
45  CLS:PRINTt$:
50  z=s: s$="n": GOSUB500: s=z
60  z=r: s$="U": GOSUB500: r=z
70  z=u: s$="n": GOSUB500: u=z
80  z=t: s$="U": GOSUB500: t=z
90  CLS:PRINT"Significance level α[%]":
95  z=f: s$="α": GOSUB500: f=z: IFf<0 ORf>100 THEN95
97  CLS:PRINTt$:TAB(0):".....":
100 n=s-1: m=u-1: P=f/100: IFs9=226 ANDr<t THENn=u-1: m=s-1
102 IFs9=226 THENP=P/2 ELSEIFs9=60 THENP=1-P
105 sr=0: GOSUB"LIB0:S6440": GOSUB900: IFsr THEN*
107 IFs9=226 ANDr<t THENz=t/r ELSEz=r/t
110 b=1: GOSUB400: a=z: z=x: GOSUB400: x=z
120 IF(s9=226 ORs9=62) ANDa>x THENb=0
122 IFs9=60 ANDa<x THENb=0
130 z=SGN(s9-60): PRINTc$: a: CHR$(60+z*2+b*(164-z)): x: " ": b$(b):
: a$=INPUT$(1,0): GOTO45
400 IFz THENz=ROUND(z, LOG(ABSz)-5)
410 RETURN
500 LOCATE0,1: PRINTc$: s$: "="; z: "?": INPUT@10: z: LOCATE0,0: RETURN
900 ONERRORGOTO910: RETURN
910 IFERR=1 THENCLS: ONERRORGOTO0
920 IFERL=500 THENRESUME500
930 LOCATE0,1: PRINTc$: "not found": z$=INPUT$(1,0): RESUME20

```

## 3.6.6 M6750 Unterprogramm der Statistik-Programme 6750, 6751, 6752

**LIB M6750** Unterprogramm der Statistik-Programme  
[6750](#), [6751](#), [6752](#)

```

5  IFsq<>60 ANDsq<>62 THENsq=226
10  MODE8: DIM: GOSUB900: ERASEb$: DIMb$(1): b$(0)="Reject": b$(1)="Accept"
: t$="Test      H0: μ=μ0   H1: μ≠" + CHR$(sq) + "μ0": DEFCHR$(254)="A2A49C92A2"
: c$=CHR$(5): DEFCHR$(252)="123E020000": DEFCHR$(253)="2E2A3A0000"
20  CLS: PRINTt$: TAB(0): "input new data x̄ (Y/N) ?": a$=INPUT$(1)
22  IFa$<>"Y" ANDa$<>"y" THEN30
25  s$="x̄": GOSUB"LIB0: S6020": GOSUB900
30  CLS: PRINTt$: TAB(0): "input new data x̄ (Y/N) ?": a$=INPUT$(1)
32  IFa$<>"Y" ANDa$<>"y" THEN36
35  s$="x̄": GOSUB"LIB0: S6030": GOSUB900
36  r=sa: IFr>1 THENs=sa: t=sm ELSEs=0: t=0
37  f=5: u=sz: IFu>1 THENv=sh: w=sn ELSEv=0: w=0
45  CLS: PRINTt$:
50  z=r: s$="h̄": GOSUB500: r=z
55  z=s: s$="s̄": GOSUB500: s=z
60  z=t: s$="S̄": GOSUB500: t=z
70  z=u: s$="h̄": GOSUB500: u=z
75  z=v: s$="s̄": GOSUB500: v=z
80  z=w: s$="S̄": GOSUB500: w=z
90  CLS: PRINT"Significance level α[%]":
95  z=f: s$="α": GOSUB500: f=z: IFf<0 ORf>100 THEN95
97  CLS: PRINTt$: TAB(0): ".....":
100 n=r+u-2: p=f/100: IFsq=226 THENp=p/2
105 sr=0: GOSUB"LIB0: S6430": GOSUB900: IFsr THEN*
107 z=(s-v): IFsq=226 THENz=ABSz
110 b=1: z=z/SQR((1/r+1/u)*((t+w)/n)): GOSUB400: a=z: z=x: GOSUB400: x=z
120 IF(sq=226 ORsq=62) ANDa>x THENb=0
122 IFsq=60 THENx=-x: IFa<x THENb=0
130 z=SGN(sq-60): PRINTc$: a: CHR$(60+z*2+b*(164-z)): x: " ": b$(b):
: a$=INPUT$(1, @): GOTO45
400 IFz THENz=ROUND(z, LOG(ABSz)-5)
410 RETURN
500 LOCATE0, 1: PRINTc$: s$: "=": z: "?": INPUT@10: z: LOCATE0, 0: RETURN
900 ONERRORGOTO910: RETURN
910 IFERR=1 THENCLS: ONERRORGOTO0
920 IFERR=500 THENRESUME500
930 LOCATE0, 1: PRINTc$: "not found": z$=INPUT$(1, @): RESUME20

```



## 3.6.7 M6760 Unterprogramm der Statistik-Programme 6760, 6761, 6762

**LIB M6760** Unterprogramm der Statistik-Programme  
[6760](#), [6761](#), [6762](#)

```

5  IF S9<>60 AND S9<>62 THEN S9=226
10  MODE8: DIM: GOSUB900: ERASE b#: DIM b#(1): b#(0)="Reject": b#(1)="Accept"
: t#="Test   H0: P=P0   H1: P+CHR$(S9)+"P0": f=5
: c#=CHR$(5): r=0: s=0: t=0: DEFCHR$(252)="123E020000"
45  CLS: PRINT t#:
50  z=r: s#="P0": GOSUB500: r=z: IF r>=1 OR r<=0 THEN 50
60  z=s: s#="n": GOSUB500: s=z
70  z=t: s#="Σx": GOSUB500: t=z
90  CLS: PRINT "Significance level α[%]":
95  z=f: s#="α": GOSUB500: f=z: IF f<0 OR f>100 THEN 95
97  CLS: PRINT t#: TAB(0): ".....":
100 P=f/100: IF S9=226 THEN P=P/2
105 sr=0: GOSUB"LIB0: S6410": GOSUB900: IF sr THEN*
107 z=t-s*r: IF S9=226 THEN z=ABS z
110 b=1: z=z/SQR(s*r*(1-r)): GOSUB400: a=z: z=x: GOSUB400: x=z
120 IF (S9=226 OR S9=62) AND a>x THEN b=0
122 IF S9=60 THEN x=-x: IF a<x THEN b=0
130 z=SGN(S9-60): PRINT c#: a: CHR$(60+z*2+b*(164-z)): x: " : "; b#(b):
: a#=INPUT$(1, @): GOTO 45
400 IF z THEN z=ROUND(z, LOG(ABS z)-5)
410 RETURN
500 LOCATE 0, 1: PRINT c#: s#: " = "; z: "?": INPUT @10: z: LOCATE 0, 0: RETURN
900 ON ERROR GOTO 910: RETURN
910 IF ERR=1 THEN CLS: ON ERROR GOTO 0
920 IF ERR=500 THEN RESUME 500
930 LOCATE 0, 1: PRINT c#: "not found": z#=INPUT$(1, @): RESUME 45

```

## 3.6.8 M6770 Unterprogramm der Statistik-Programme 6770, 6771, 6772

<b>LIB M6770</b>	Unterprogramm der Statistik-Programme <a href="#">6770</a> , <a href="#">6771</a> , <a href="#">6772</a>
------------------	---

```

5  IF S9<>60 AND S9<>62 THEN S9=226
10  MODE8: DIM: GOSUB900: ERASE b#: DIM b#(1): b#(0)="Reject": b#(1)="Accept"
: t#="Test      H0: P=P0   H1: P≠"+CHR$(S9)+"P=": f=5
: c#=CHR$(5): DEFCHR$(252)="123E020000": DEFCHR$(253)="2E2A3A0000": r=0: s=0: t=0: u=0
45  CLS: PRINT t#:
50  z=r: s#="n#": GOSUB500: r=z
60  z=s: s#="Σx#": GOSUB500: s=z
70  z=t: s#="n#": GOSUB500: t=z
80  z=u: s#="Σx#": GOSUB500: u=z
90  CLS: PRINT "Significance level α[%]":
95  z=f: s#="α": GOSUB500: f=z: IF f<0 OR f>100 THEN 95
97  CLS: PRINT t#: TAB(0): ".....":
100 P=f/100: IF S9=226 THEN P=P/2
105 sr=0: GOSUB"LIB0: S6410": GOSUB900: IF sr THEN*
107 z=s/r-u/t: IF S9=226 THEN z=ABS z
110 b=1: v=(s+u)/(r+t): z=z/SQR(v*(1-v)*(1/r+1/t)): GOSUB400: a=z: z=x: GOSUB400: x=z
120 IF (S9=226 OR S9=62) AND a>x THEN b=0
122 IF S9=60 THEN x=-x: IF a<x THEN b=0
130 z=SGN(S9-60): PRINT c#: a: CHR$(60+z*2+b*(164-z)): x: " ": b#(b):
: a#=INPUT$(1, @): GOTO45
400 IF z THEN z=ROUND(z, LOG(ABS z)-5)
410 RETURN
500 LOCATE0, 1: PRINT c#: s#: "=": z: "?": INPUT@10: z: LOCATE0, 0: RETURN
900 ONERROR GOTO910: RETURN
910 IF ERR=1 THEN CLS: ONERROR GOTO0
920 IF ERR=500 THEN RESUME500
930 LOCATE0, 1: PRINT c#: "not found": z#=INPUT$(1, @): RESUME45

```

## 3.6.9 S6020 Unterprogramm der Statistik-Programme 6610, 6620, 6630, 6640, 6650, 6660, M6710, M6720, M6730, M6740, M6750

<b>LIB S6020</b>	Unterprogramm der Statistik-Programme <a href="#">6610</a> , <a href="#">6620</a> , <a href="#">6630</a> , <a href="#">6640</a> , <a href="#">6650</a> , <a href="#">6660</a> , <a href="#">M6710</a> , <a href="#">M6720</a> , <a href="#">M6730</a> , <a href="#">M6740</a> , <a href="#">M6750</a>
------------------	---

```

5  ONERRORGOTO1000
10  MODE8: DIM: ERASE a, a#: DIM a(5), a#(1): a#(1)="Input": a#(0)="Delete": c#=CHR$(5)
20  CLS: PRINT "Input data (<"; s#: ">"; TAB(0); ">Input,Delete,Clear,List,End ?";
30  a=ASC(INPUT$(1)): IF a>96 THEN a=a-32
40  ON a-66 GOTO 300,100,200,,100,,500
50  GOTO 30
100 CLS: PRINT a#(SGN(a-68)); " data (<"; s#: ">"; TAB(21); "[EXE]:menu"; : z=SGN(a-69)
110 LOCATE 0,1: PRINT c#: s#: "?";
120 LOCATE LEN(s#)+1,1: INPUT@12: z#: LOCATE 0,0: IF z#="" THEN 20 ELSE x=VALF(z#)
150 r=sa+z: s=sb+x*z: u=sd+x*x*z
160 sa=r: sb=s: sd=u: GOTO 110
200 c=6
210 sa=sb/sa
220 sm=sd-sa*sb: IF sm<0 THEN sm=0
250 si=SQR(sm/sa)
260 sk=SQR(sm/(sa-1))
280 RETURN
300 PRINT c#: "clear data (Y/N) ?";
310 a#=INPUT$(1,0): IF a#<>"Y" AND a#<>"y" THEN 20
320 sa=0: sb=0: sd=0: GOTO 20
500 CLS: GOSUB 200: GOSUB 580: i=1: GOSUB 600: i=2
510 GOSUB 600
520 z=ASC(INPUT$(1,0)): IF z=28 OR z=29 THEN 20
530 IF z<>13 AND z<>30 AND z<>31 THEN 520
540 IF (z=30 AND i=1) OR (z=31 AND i=c) THEN 520
550 IF z=30 THEN i=i-2
560 i=i+1: IF i<c THEN 510 ELSE 20
580 DEFCHR$(255)="0000003E00"
590 a(0)=sa: a(1)=sb: a(2)=sd: a(3)=sa: a(4)=si: a(5)=sk: RETURN
600 RESTORE(600+i*10): READ a#, b#: PRINT TAB(0); a#: TAB(7); " "; b#: TAB(13); "="
: a(i-1): RETURN
610 DATACNT, n
620 DATASUMX, Σx
630 DATASUMX2, Σx²
640 DATAMEANX, Σx/n
650 DATASDXN, x̄σn
660 DATASDX, x̄σn̄
1000 IFERR=1 THEN CLS: ONERRORGOTO 0
1010 IFERR=120 THEN RESUME 120
1050 IFERR=210 THEN c=3: RESUME 280
1060 IFERR=260 THEN c=5: RESUME 280
1080 IFERR=150 THEN LOCATE 0,1: PRINT c#: "data over"; : z#=INPUT$(1,0): RESUME 20
1090 LOCATE 0,1: PRINT c#: "not found"; : a#=INPUT$(1,0): RESUME 20

```

3.6.10 S6030 Unterprogramm der Statistik-Programme 6650, 6660,  
M6740, M6750

<b>LIB S6030</b>	Unterprogramm der Statistik-Programme <a href="#">6650</a> , <a href="#">6660</a> , <a href="#">M6740</a> , <a href="#">M6750</a>
------------------	--

```

5 ONERRORGOTO1000
10 MODE8:DIM:ERASEa,a#:DIMa(5),a$(1):a$(1)="Input":a$(0)="Delete":c#=CHR$(5):sf=0
20 CLS:PRINT"Input data (<";s#;">";TAB(0);"Input,Delete,Clear,List,End ?";
30 a=ASC(INPUT$(1)):IFA>96 THENa=a-32
40 ONa-66 GOTO300,100,200,,100,,500
50 GOTO30
100 CLS:PRINTa$(SGN(a-68));" data (<";s#;">";TAB(21);"[EXE]menu";:z=SGN(a-69)
110 LOCATE0,1:PRINTc#:s#;"?";
120 LOCATELEN(s#)+1,1:INPUT@12;z#:LOCATE0,0:IFz#="" THEN20 ELSEy=VALF(z#)
150 r=sz+z:s=sc+y*z:u=se+y*y*z
160 sz=r:sc=s:se=u:GOTO110
200 c=6
210 sh=sc/sz
220 sn=se-sh*sc:IFsn<0 THENsn=0
250 sj=SQR(sn/sz)
260 sl=SQR(sn/(sz-1))
280 RETURN
300 PRINTc#:"clear data (Y/N) ?";
310 a#=INPUT$(1,@):IFA#<"Y" ANDa#<"y" THEN20
320 sz=0:sc=0:se=0:GOTO20
500 CLS:GOSUB200:GOSUB500:i=1:GOSUB600:i=2
510 GOSUB600
520 z=ASC(INPUT$(1,@)):IFz=28 ORz=29 THEN20
530 IFz<>13 ANDz<>30 ANDz<>31 THEN520
540 IF(z=30 ANDi=1) OR(z=31 ANDi=c) THEN520
550 IFz=30 THENi=i-2
560 i=i+1:IFI=<c THEN510 ELSE20
580 DEFCHR$(255)="0000003E00"
590 a(0)=sz:a(1)=sc:a(2)=se:a(3)=sh:a(4)=sj:a(5)=sl:RETURN
600 RESTORE(600+i*10):READa#,b#:PRINTTAB(0);a#:TAB(7);":":b#
:TAB(13);"=";a(i-1):RETURN
610 DATACNT,n
620 DATASUMX, $\Sigma x$ 
630 DATASUMX2, $\Sigma x^2$ 
640 DATAMEANX, $\Sigma x/n$ 
650 DATASDXN, $x\sigma_n$ 
660 DATASDX, $x\sigma_n$ 
1000 IFERR=1 THENCLS:ONERRORGOTO0
1010 IFERL=120 THENRESUME120
1050 IFERL=210 THENc=3:RESUME280
1060 IFERL=260 THENc=5:RESUME280
1080 IFERL=150 THENLOCATE0,1:PRINTc#:"data over":z#=INPUT$(1,@):RESUME20
1090 LOCATE0,1:PRINTc#:"not found":a#=INPUT$(1,@):RESUME20

```

## 3.6.11 S6210 Unterprogramm der Statistik-Programme 6210, S6220

**LIB S6210** Unterprogramm der Statistik-Programme  
[6210](#), [S6220](#)

```

5 ONERRORGOTO500
10 y=ABSx:P=0
15 IFy>100 THEN60
20 IFy<1.9 THEN50
30 FORi=18 TO1 STEP-1:P=i/(y+P):NEXT
40 P=EXP(-.5*y*y)/SQ R(2*PI)/(y+P):GOTO60
50 P=(1+y*(.049867347+y*(.0211410061+y*(.0032776263+y*(.0000380036
+y*(.0000488906+y*(.000005383))))))^-16/2
60 IFx<0 THENP=1-P
80 y=P:RETURN
500 IFERR=1 THENCLS:ONERRORGOTO0
510 sr=1:RESUME80

```

## 3.6.12 S6220 Unterprogramm der Statistik-Programme 6220, S6420

**LIB S6220** Unterprogramm der Statistik-Programme  
[6220](#), [S6420](#)

```

10 GOSUB400:IFx<0 THENP=1:GOTO80
20 IFn>100 THENz=x:x=((x/n)^(1/3)-(1-
2/9/n))/SQ R(2/9/n):GOSUB"LIB0:S6210":GOSUB400:P=y:x=z:GOTO80
30 IFx>400 THENP=0:GOTO80
40 a=EXP(-x/2):P=a:y=2
50 IFn MOD 2<>0 THENz=x:x=SQ R(x):GOSUB"LIB0:S6210"
:GOSUB400:P=2*y:a=SQ R(2/PI)*a/x:x=z:y=1
60 FORi=y TO n-2 STEP2:a=a*x/i:P=P+a:NEXT
80 y=P:RETURN
400 ONERRORGOTO500:RETURN
500 IFERR=1 THENCLS:ONERRORGOTO0
510 sr=1:RESUME80

```

## 3.6.13 S6230 Unterprogramm der Statistik-Programme 6230, S6430

**LIB S6230** Unterprogramm der Statistik-Programme  
[6230](#), [S6430](#)

```

5 ONERRORGOTO500
10 ANGLE1:y=ATN(x/SQ Rn):z=COSy^2
20 IFn MOD 2=0 THENa=SINy/2:b=.5 ELSEb=.5+y/PI:IFn=1 THENa=0 ELSEa=SINy*COSy/PI
30 y=1:FORi=n-2 TO2 STEP-2:y=1+(i-1)/i*z*y:NEXT
40 P=1-(b+a*y):IFP<0 THENP=0
80 y=P:RETURN
500 IFERR=1 THENCLS:ONERRORGOTO0
510 sr=1:RESUME80

```

## 3.6.14 S6240 Unterprogramm der Statistik-Programme 6240, S6440

**LIB S6240** Unterprogramm der Statistik-Programme  
[6240](#), [S6440](#)

```

5 ONERRORGOTO500
10 ANGLE1: IFx=<0 THENP=1:GOTO80
20 IFm MOD 2=0 THENz=m/(m+n*x):GOSUB100:P=1-P:GOTO80
30 IFn MOD 2=0 THENz=n*x/(m+n*x):i=m:m=n:n=i:GOSUB100:i=m:m=n:n=i:GOTO80
40 y=ATNSQR(n*x/m)
50 GOSUB200:P=P+a: IFP<0 THENP=0
80 y=P:RETURN
100 a=1:FORi=m-2 TO2 STEP-2:a=1+(n+i-2)/i*z*a:NEXT
110 P=(1-z)^(n/2)*a:RETURN
200 z=(SINy)^2: IFn=1 THENa=0 ELSEa=1
210 FORi=n-2 TO3 STEP-2:a=1+(m+i-2)/i*z*a:NEXT
220 b=PI:FORi=2 TOm-1 STEP2:b=b*(i-1)/i:NEXT
230 P=2/b*SINy*(COSy)^m*a
250 z=(COSy)^2: IFm=1 THENa=0 ELSEa=1
260 FORi=m-2 TO3 STEP-2:a=1+(i-1)/i*z*a:NEXT
270 a=1-2*y/PI-2/PI*SINy*COSy*a:RETURN
500 IFERR=1 THENCLS:ONERRORGOTO0
510 sr=1:RESUME80

```

## 3.6.15 S6310 Unterprogramm der Statistik-Programme 6310

**LIB S6310** Unterprogramm der Statistik-Programme  
[6310](#)

```

5 ONERRORGOTO500
10 IFP=1 THENIFn=x THENs=1: f=1:GOTO80 ELSEs=0: f=0:GOTO80
20 a=1-P: f=a^n: s=f: z=P/a: y=n+1
30 FORi=1 TOx: f=f*z*(y-i)/i: s=s+f:NEXT
80 s=1-s+f:RETURN
500 IFERR=1 THENCLS:ONERRORGOTO0
510 sr=1:RESUME80

```

## 3.6.16 S6320 Unterprogramm der Statistik-Programme 6320

**LIB S6320** Unterprogramm der Statistik-Programme  
[6320](#)

```

5 ONERRORGOTO500
10 f=EXP(-k):s=f
20 FORi=1 TOx: f=f*k/i: s=s+f:NEXT
80 s=1-s+f:RETURN
500 IFERR=1 THENCLS:ONERRORGOTO0
510 sr=1:RESUME80

```

## 3.6.17 S6330 Unterprogramm der Statistik-Programme 6330

**LIB S6330** Unterprogramm der Statistik-Programme  
[6330](#)

```

5 ONERRORGOTO500
10 GOSUB100:s=f:a=k+1:y=n+1
30 FORi=1 TOx:f=f*(a-i)*(y-i)/i/(z+i):s=s+f:NEXT
80 s=1-s+f:RETURN
100 a=m-k:y=m-n:z=m-k-n:f=1
110 IFa<17 ANDk<17 THENFORi=0 TOm-1:f=f*(a-i)/(m-i):NEXT:RETURN
120 f=(a+.5)*LNa+(y+.5)*LNy-(m+.5)*LNm-(z+.5)*LNz
+(1/a+1/y-1/m-1/z)/12:f=EXPf:RETURN
500 IFERR=1 THENCLS:ONERRORGOTO0
510 sr=1:RESUME80

```

3.6.18 S6410 Unterprogramm der Statistik-Programme 6410, 6610,  
6670, 6680, M6710, M6760, M6770, S6420, S6430

**LIB S6410** Unterprogramm der Statistik-Programme  
[6410](#), [6610](#), [6670](#), [6680](#),  
[M6710](#), [M6760](#), [M6770](#), [S6420](#), [S6430](#)

```

5 ONERRORGOTO500
10 IFP=<0 ORP>=1 THEN*
20 y=-LN(4*P*(1-P))
30 x=.5824238515E-5+y*(-.104527497E-5+y*(.8360937017E-7
+y*(-.3231081277E-8+y*(.3657763036E-10+y*.6936233982E-12))))
40 x=SQ R(y*(1.570796288+y*(.03706987906+y*(-.8364353589E-3
+y*(-.2250947176E-3+y*(.6841218299E-5+y*x))))))
50 IFP>.5 THENx=-x
80 y=x:RETURN
500 IFERR=1 THENCLS:ONERRORGOTO0
510 sr=1:RESUME80

```

## 3.6.19 S6420 Unterprogramm der Statistik-Programme 6420, 6630, 6640, M6730, S6440

**LIB S6420** Unterprogramm der Statistik-Programme  
[6420](#), [6630](#), [6640](#), [M6730](#), [S6440](#)

```

10 GOSUB400:IFP>1 ORP=<0 THEN*
15 IFP=1 THENx=0:GOTO80
20 IFn=1 THENq=P:P=q/2:GOSUB"LIB0:S6410":GOSUB400:x=y*y:P=q:GOTO80
30 IFn=2 THENx=-2*LNp:GOTO80
40 GOSUB"LIB0:S6410":GOSUB400:x=y:y=x*x:x=n+SQ R(2*n)*x
+2/3*(y-1)+x*(y-7)/9/SQ R(2*n)-2/405/n*(y*(3*y+7)-16):IFx<0 THENx=0
50 IFn>100 THEN80
60 b=x:GOSUB100:IFn<31 THENIFABS(b-x)>1E-4 THEN60
80 y=x:RETURN
100 q=P:GOSUB"LIB0:S6220":GOSUB400:P=y
110 z=EXP(((n-1)*LN(x/n)-LN(4*PI*x)+n-x-1/n/6)/2)
120 x=x+(P-q)/z:P=q:x=ROUND(x,-6):RETURN
400 ONERRORGOTO500:RETURN
500 IFERR=1 THENCLS:ONERRORGOTO0
510 sr=1:RESUME80

```

## 3.6.20 S6430 Unterprogramm der Statistik-Programme 6430, 6620, 6660, M6720, M6750, S6440

**LIB S6430** Unterprogramm der Statistik-Programme  
[6430](#), [6620](#), [6660](#), [M6720](#), [M6750](#), [S6440](#)

```

20 GOSUB400:ANGLE1:IFP>=1 ORP=<0 THEN*
30 GOSUB"LIB0:S6410":GOSUB400:x=y:y=x^2
40 a=(y+1)/4:b=((5*y+16)*y+3)/96:c=((3*y+19)*y+17)*y-15)/384
:d=(((79*y+776)*y+1482)*y-1920)*y-945)/92160
:e=(((27*y+339)*y+930)*y-1782)*y-765)*y+17955)/368640
50 x=x*(1+(a+(b+(c+(d*e/n)/n)/n)/n)/n)
60 IFn>(LOGP)^2+3 THEN80
70 q=P:GOSUB"LIB0:S6230":GOSUB400:P=y:GOSUB100:y=x:x=x+(P-q)/a:P=q
:IFx THENIFROUND(x-y,LOG(ABSx)-5)<>0 THEN70
80 y=x:RETURN
100 b=n+1:a=EXP((b*LN(b/(n+x*x))+LN(n/b/2/PI)-1+(1/b-1/n)/6)/2):RETURN
400 ONERRORGOTO500:RETURN
500 IFERR=1 THENCLS:ONERRORGOTO0
510 sr=1:RESUME80

```



## 3.6.21 S6440 Unterprogramm der Statistik-Programme 6440, 6650, M6740

<b>LIB S6440</b>	Unterprogramm der Statistik-Programme <a href="#">6440</a> , <a href="#">6650</a> , <a href="#">M6740</a>
------------------	--

```

10 GOSUB400:ANGLE1:IFP>1 ORP=<0 THEN*
15 IFP=1 THENx=0:GOTO80
20 IFm=1 THENm=P:P=.5-P/2:GOSUB"LIB0:S6430":GOSUB400:P=m:m=1:x=1/y^2:GOTO80
30 IFn=1 THENn=m:P=P/2:GOSUB"LIB0:S6430":GOSUB400:n=1:P=P*2:x=y^2:GOTO80
40 IFm=2 THENP=1-P:m=n:n=2:GOSUB100:P=1-P:n=m:m=2:x=1/x:GOTO80
50 IFn>m THENP=1-P:d=n:n=m:m=d:GOSUB200:x=1/x:d=m:m=n:n=d:P=1-P:GOTO80
60 GOSUB200
80 y=x:RETURN
100 GOSUB"LIB0:S6420":GOSUB400:x=y:a=n-2
110 x=x/n*(1+((x-a)/2+(((4*x-11)*a)*x+a*(7*n-10))/24+(((2*x-10)*a)*x
+a*(17*n-26))*x-a*a*(9*n-6))/48/m)/m)/m):RETURN
200 GOSUB100
210 d=x:c=P:GOSUB"LIB0:S6240":GOSUB400:P=c
220 z=n+m:z=EXP((z*LN(z/(n*x+m)))+(n-2)*LNz+LN(n*m/z)-LN(4*PI)
-(1/n+1/m-1/z)/6)/2):x=x+(y-P)/z
230 IFABS(d-x)>3E-4 THEN210 ELSERETURN
400 ONERRORGOTO500:RETURN
500 IFERR=1 THENCLS:ONERRORGOTO0
510 sr=1:RESUME80

```

## 3.7 Undokumentierte Funktionen

### 3.7.1 0400 Speicher- und Schnittstellen-Testprogramm

LIB 0400

Speicher- und Schnittstellen-Testprogramm

```

10 CLEAR:CLS:DEFSEG=0:K=PEEK(&H6B4)/4+PEEK(&H6B5)*64
20 IF K<32 THENK1=8:K2=K-K1
30 IF K=32 OR K=64 THENK1=32:K2=K-K1
40 IF K=40THENPOKE&H0000,10:POKE0,0:IFPEEK&HA000=10THENK1=32:K2=8ELSEK1=8:K2=32
50 PRINT"Main":K1:TAB(8):"Kbyte Option":K2:"Kbyte":PRINT:A=64:D=1:A1=0
70 GOSUB1000:A=A+A:D=(D+D)MOD256:IF A<K*1024 THEN70
80 FORI=1 TO9:A=K*1024-A1-1:IF I=5 OR I=9 THEND=&H00 ELSE D=&H20
100 GOSUB1010:IF A1=0 THENA1=1 ELSEA1=A1*2
110 NEXT
120 A=K*1024-A1-1:GOSUB1000:A1=A1*2:IF A1<K*1024 THEN120
130 PRINT" OK RAM":GOSUB10000:PRINT:DEFSEG=&H4000
160 ROM=(PEEK(4)+1+PEEK(5)*256)*2
170 IF PEEK(0)<>&H43 THENPRINT"NG ROM 1":GOTO1030
180 IF ROM<>1024 THEN200
190 D1=PEEK(&H5955):D2=PEEK&HA6AA:D3=PEEK&HFFFF:DEFSEG=&H5000
191 D4=PEEK0:D5=PEEK&H5955:D6=PEEK&HA6AA
192 IF PEEK&HFFFF<>&H20 THENPRINT"NG ROM 2":GOTO1030
193 PRINT"OK 512K*2 ROM?":PRINT
194 PRINTRIGHT$(HEX$(D1),2);" ";RIGHT$(HEX$(D2),2);" ";RIGHT$(HEX$(D3),2);" ";
" ";RIGHT$(HEX$(D4),2);" ";RIGHT$(HEX$(D5),2);" ";RIGHT$(HEX$(D6),2);" ";GOTO210
200 IF PEEK(ROM*128-1)<>&H20 THENPRINT"NG ROM 1":GOTO1030
201 D1=PEEK&H5955:IF ROM=256 THEND2=PEEK&H26AA ELSE D2=PEEK&HA6AA
202 PRINT"OK":STR$(ROM):"K ROM?":PRINT
203 PRINTRIGHT$(HEX$(D1),2);" ";RIGHT$(HEX$(D2),2);" ";
210 GOSUB10000
220 CLS:FOR I=0 TO 63:PRINTCHR$(&HFF):NEXT:LOCATE 0,0
230 DEFCHR$(&HFF)="FFFFFFFF":GOSUB10000:DEFCHR$(&HFF)="0000000000":GOSUB10000
250 DEFCHR$(&HFF)="AA55AA55AA":GOSUB10000:DEFCHR$(&HFF)="55AA55AA55":TRON
:GOSUB10000:TROFF
280 CLS:PRINT"Printer CHECK":PRINT:GOSUB10000
290 LPRINT"Main":K1:"Kbyte Option":K2:"Kbyte"
320 FOR I=32 TO 255:IF I=&H7F THENI=&H00
330 LPRINTCHR$(I):NEXT:LPRINT
350 PRINT"RS 232C CHECK":PRINT:GOSUB10000
360 OPEN "COM0:5,E,8,2,C,D,C,B,N" AS #1:PRINT#1,123:INPUT #1,D
390 CLOSE:IF D<>123 THENPRINT"NG RS 232C":GOTO1030
410 PRINT"OK RS 232C":GOSUB10000:PRINT
420 PRINT"MT CHECK <REC>":PRINT:GOSUB10000
430 OPEN "CAS0:TEST" FOR OUTPUT AS #1:PRINT#1,"ABC":CLOSE
460 PRINT"MT <REW> and <PLAY>":PRINT:GOSUB10000
470 OPEN "CAS0:TEST" FOR INPUT AS #1:INPUT #1,A#:CLOSE
490 IF A#<>"ABC" THENPRINT"NG MT":PRINT:GOTO1030
510 PRINT"OK MT":PRINT
520 END
1000 POKE A,D
1010 IF PEEKA=D THENRETURN
1020 IF A>=K1 THENPRINT"NG RAM 2": ELSE PRINT "NG RAM 1":
1030 BEEP:GOTO1030
10000 BEEP:IF INPUT$(1,0)=CHR$(0) THEN10000ELSE RETURN

```



## 3.7.3 0900 Konvertier-Programm

## LIB 0900 Konvertier-Programm

```

10 MODE8:DEFSEG=0:GOTO110
20 GOSUB20
30 OPEN#1 FORINPUTAS#1:IFASC(INPUT$(1,1))<>2 THENCLOSE:GOTO30 ELSERETURN
50 IF#="" THENRETURN ELSEGOSUB70:FORk=1 TOLEN(#):k#=MID$(#,k,1):MODE32(k#)
: NEXT: #="" :FORl=0 TOk-2: #=#+CHR$(PEEK(1+768)): NEXT: RETURN
70 POKE768,0:POKE769,0:POKE1608,0:POKE1609,3:RETURN
90 n=PEEK(e):e=e-1:RETURN
110 n#="" :a=2:a#="" :j#="" :i=0:p#="" :m=512:f#="" :c#="" :o=0:g=0:e=0:d=0:e#=""
120 p=0:b#="" :k=0:#="" :c=0:w=0:u=0:v=0:s=0:o#="" :i#="" :q=0:h#="" :r=0:m#=""
: k#="" :h=0:f=0:t=0:l=0:b=0:x=0:y=0:n=0:d#=""
200 IFPEEK(m)=32 THENm=m+1:GOTO200
210 IFPEEK(m)=4 ANDPEEK(m+1)=180
THENf#="UF":GOSUB2190:GOSUB2120:GOSUB2220:GOTO410
220 IFPEEK(m)<>4 ORPEEK(m+1)<>179 THENa
230 n=PEEK(m+a):IFn=32 THENa=a+1:GOTO230
240 IFn=0 ORn=34 ORn=48 ORn=49 THENf#="PF":GOSUB2190:GOSUB2120:GOSUB2220:GOTO410
250 IFn=7 ANDPEEK(m+a+1)=187
THENf#="AF":a=a+2:GOSUB2190:GOSUB2120:GOSUB2220:GOTO410
260 IFn=35 THENf#="MF":a=a+1:GOSUB2190:GOSUB2120 ELSEa
270 n=PEEK(m+a)
280 IFn=0 THEN#="" :GOTO410
290 IFn=32 THENa=a+1:GOTO270
300 IFn=44 THENa=a+1:GOSUB2250:IFn=77 THENa=1:GOSUB2220 ELSEa ELSEa
410 a=1709:o=(PEEK(a+1))*256+(PEEK(a))-1:f=o:e=o:h=o-FRE(1)+1
430 FORd=1867 TO1874:e#=#+CHR$(PEEKd): NEXT
460 b#="" :a#="" :OPEN#1 FORINPUTAS#1:p=ASC(INPUT$(1,1)):IFp<144 THENCLOSE:GOTO460
470 b#=INPUT$(10,1):CLOSE
490 IF# MOD16=15 THENa#="" :GOTO510
500 FORk=1 TO8:j#=MID$(b#,k,1):IFj#<>CHR$(?) THENa#=#+j#: NEXT
510 i=ASC(MID$(b#,10)):c=VAL("&h"+MID$(HEX$(p),3,1))-8
520 IFc<2 THENn#="MF":GOTO560
530 IFc=6 THENn#="UF":GOTO560
540 IFc=30rc=7 THENn#="AF":GOTO560
550 n#="PF"
560 #=#+a#:GOSUB50:b#=#:PRINTb#:TAB(9):n#:#:PRINT
570 IFf#<>n# THEN460
580 IFc#<>" " ANDc#<>b# THEN460
600 ONc GOTO810,, ,810,810,810
710 GOSUB30
720 n=ASC(INPUT$(1,1)):IFn=255 ORn=241 THEN730 ELSEd#=#+CHR$(n):GOTO720
730 CLOSE:#=#+d#:GOSUB50:d#=#
735 IFLEN(d#)>8 THENREAD#
740 IFLEN(d#)<8 THENd#=#+CHR$(0):GOTO740
750 IFe#=# THEN810
760 PASSd#
810 GOSUB30:MODE35<:CLOSE:f=PEEK(1609)*256+PEEK(1608)-1
1010 IFf#="UF" THEN10010 ELSEPRINT"Convertina...":
1020 IFf#="MF" THEN6010
1030 IFf#="PF" THEN1140
1110 w=48:o#=HEX$(PEEK(1715)+PEEK(1716)*256-15):POKE1616,VAL("&h"+RIGHT$(o#,2))
: POKE1617,VAL("&h"+LEFT$(o#,2))
1130 FORk=48 TO57:k#=CHR$(k):MODE30(k#): NEXT:GOTO1160
1140 w=PEEK(PEEK(1611)*256+PEEK(1610)+8):k#=CHR$(w):MODE30(k#)
1160 q=0:x=0:y=0:l=0:GOSUB70:GOSUB2040:u=n:GOSUB90:v=n
: o#=RIGHT$(HEX$(v),2)+RIGHT$(HEX$(u),2):o#=MID$(STR$(VAL(o#)),2)
: s=PEEK(1609)*256+PEEK(1608)
1170 FORk=1 TOLEN(o#):POKE768+k-1,ASC(MID$(o#,k,1)):s=s+1: NEXT
: POKE768+k-1,0:i#=HEX$(s):POKE1609,VAL(LEFT$(i#,2)):POKE1608,VAL(RIGHT$(i#,2))
1180 GOSUB90
1182 IFn=255 THEN1240
1184 IFn=254 THENq=0:x=0:y=0:l=0:GOTO1225
1186 IFn=7 THENy=y+1:IFY>2 THENy=0:l=1:GOTO1225
1188 IFy=1 THEN1225
1190 IFn=167 THENx=1:GOTO1225
1192 IFn*x*1=93 THENn=94:GOTO1223

```

## LIB 0900 Konvertier-Programm

```

1194 IFn=165 ORn=151 THENq=SGN(n-151):GOTO1223
1196 IFn*q=94 THENn=151:q=0
1223 l=0
1225 IFPEEK(1608)>=247 THEN5000
1230 h#=CHR$(n):MODE32(h#):GOTO1180
1240 GOSUB2080:MODE31(m#):GOTO1160
2030 IFPEEK(m+a)=44 ANDPEEK(m+a+1)=77 THENa=1:RETURN
2040 e=o
2050 GOSUB90:IFn=224 THENw=w+1:GOTO2050
2060 IFn=240 THENEND ELSERETURN
2080 m#=CHR$(w)
2090 r=e-f:t=o-r+1:MODE33(f+1,t,r):f=t-1:IF(FRE(1))-r<0 THEN20
2100 RETURN
2120 n=PEEK(m+a):IFn=44 ANDf#="MF" THENRETURN
2130 IFn=0 THENRETURN
2140 IFn=32 THENa=a+1:GOTO2120
2150 IFn<>34 THENa ELSEa=a+1
2160 n=PEEK(m+a):IFn=34 THENa=a+1:c#=LEFT$(c#,8):RETURN
2170 IFn=0 THENc#=LEFT$(c#,8):RETURN ELSEc#=c#+CHR$(n):a=a+1:GOTO2160
2190 n=PEEK(m+a):IFn=48 ORn=49 THENp#="PMT"+CHR$(n)+":":a=a+1:RETURN
2195 IFn=32 THENa=a+1:GOTO2190
2200 p#="PMT0:":RETURN
2220 n=PEEK(m+a):IFn=0 THENRETURN
2230 IFn=32 THENa=a+1:GOTO2220 ELSEa
2250 n=PEEK(m+a):a=a+1:IFn=32 THEN2250 ELSERETURN
5000 j#=j#+"ST error":GOTO5000
6010 IFa=1THEN6040
6020 RESTORE#
6030 RESTORE#"",1,6060:WRITE#:GOTO6030
6040 RESTORE#CHR$(26),0,6060:READ#
6060 GOSUB70:e=o
6070 GOSUB90:IFn=240 THENEND
6075 IFPEEK(1608)>=247 THEN5000
6080 IFn<>255 THENh#=CHR$(n):MODE32(h#):GOTO6070
6090 GOSUB2090:k=768:s=256*PEEK(1609)+PEEK(1608):l#=""
6100 IFk=s THENWRITE#l#:GOTO6060 ELSEl#=l#+CHR$(PEEK(k)):k=k+1:GOTO6100
10010 e=o:j=0:a=0:IFI%16>0 THEN10100
10020 IFA=0 THENPRINT"GET ";(e-f-1)/8:"data":PRINT:PRINT "Convertins...";
10030 IFE<f+5 THENEND
10040 GOSUB70:n=PEEK(e-7):IFn>10 THEN10070
10050 FORk=0 TO7:GOSUB90:POKE768+k,n:NEXT:MODE34("R("+STR$(j)+")"+CHR$(0))
:j=j+1:GOTO10030
10070 a=n%16
10080 FORk=0 TO7:GOSUB90:a=a-1:IFA>=0 THENMODE32(CHR$(n))
10085 NEXT
10090 MODE34("R("+STR$(j)+")"+CHR$(0)):j=j+1:GOTO10030
10100 GOSUB90:b=n:a=30:IFI%16=14 THENa=62
10110 PRINT"GET #, ";(e-f-a-2)/8:"data":PRINT:PRINT"Convertins...":GOSUB70
10120 FORk=0 TOa:GOSUB90:b=b-1:IFb>=0 THENMODE32(CHR$(n))
10130 NEXT:MODE34("#"+CHR$(0)):GOTO10030

```

## 4 Anhang

### 4.1 Befehls-Übersicht

Das ist meine alphabetisch geordnete Befehls-Übersicht. Die Seitenzahl verweist direkt zur Seite der CASIO-Bedienungsanleitung des Rechners.

Befehl/Funktion	Kategorie	Beschreibung	Seite
<b>&amp;H</b>	Zeichen-Funktion	Wandelt den auf &H folgenden 1- bis 4-stelligen Hexadezimalwert in einen Dezimalwert um.	159
<b>ABS</b>	Numerische Funktion	Berechnet den Absolutwert des Arguments.	141
<b>ACS</b>	Numerische Funktion	Berechnet den Wert des entsprechenden inversen trigonometrischen Funktionswertes für das Argument.	135
<b>ANGLE</b>	Numerische Funktion	Spezifiziert die Winkeleinheit.	133
<b>ASC</b>	Zeichen-Funktion	Liefert den Zeichencode, der dem ersten Zeichen (ganz links) einer Zeichenfolge entspricht.	151
<b>ASN</b>	Numerische Funktion	Berechnet den Wert des entsprechenden inversen trigonometrischen Funktionswertes für das Argument.	135
<b>ATN</b>	Numerische Funktion	Berechnet den Wert des entsprechenden inversen trigonometrischen Funktionswertes für das Argument.	135
<b>BEEP</b>	Grundbefehl	Aktiviert den Summer.	121
<b>CHR\$</b>	Zeichen-Funktion	Liefert ein Zeichen, das dem spezifizierten Zeichencode entspricht.	150
<b>CLEAR</b>	Manueller Befehl	Löscht alle Variable und bestimmt die Variablenbereich-Größe entsprechend dem eingegebenen Parameter. Außerdem werden alle offenen Daten geschlossen.	93
<b>CLOSE</b>	Ein/Ausgabe-Befehl	Schließt Dateien und deklariert das Ende der Verwendung des I/O (Eingabe/Ausgabe)-Puffers.	166
<b>CLS</b>	Grundbefehl	Löscht den Bildschirm.	119
<b>COS</b>	Numerische Funktion	Berechnet den Wert des entsprechenden trigonometrischen Funktionswertes für das Argument.	134
<b>CUR</b>	Numerische Funktion	Berechnet die Kubikwurzel des Arguments.	140
<b>DATA</b>	Grundbefehl	Nimmt Daten auf, die durch die Anweisung READ gelesen werden sollen.	112
<b>DEFSEG</b>	Grundbefehl	Spezifiziert Segment-Grundadressen.	129
<b>DEG</b>	Zeichen-Funktion	Wandelt einen Sexagesimalwert in einen Dezimalwert um.	160
<b>DIM</b>	Grundbefehl	Deklariert ein Feld.	125
<b>DMS\$</b>	Zeichen-Funktion	Wandelt einen Dezimalwert in eine Sexagesimal-Zeichenfolge um.	161
<b>EDIT</b>	Manueller Befehl	Aktiviert den BASIC-Editor-Modus.	97
<b>END</b>	Grundbefehl	Beendet die Programmausführung.	100
<b>EOF</b>	Ein/Ausgabe-Befehl	Zeigt das Ende der Datei-Eingabe an.	170
<b>ERASE</b>	Grundbefehl	Löscht ein spezifiziertes Feld.	126
<b>ERL</b>	Grundbefehl	Liefert die Nummer einer Zeile, in der ein Fehler erzeugt wurde.	132

Befehl/Funktion	Kategorie	Beschreibung	Seite
<b>ERR</b>	Grundbefehl	Liefert den Fehlercode, der einem aufgetretenen Fehler entspricht.	132
<b>EXP</b>	Numerische Funktion	Berechnet den Wert der Exponentialfunktion für das Argument.	138
<b>FACT</b>	Numerische Funktion	Berechnet die Fakultät des Arguments.	146
<b>FIX</b>	Numerische Funktion	Ermittelt den ganzzahligen Anteil des Arguments.	143
<b>FOR ~ NEXT</b>	Grundbefehl	Führt die Programmzeilen zwischen der FOR-Anweisung und der NEXT-Anweisung aus und erhöht die Steuervariable, beginnend mit dem Anfangswert. Die Ausführung wird beendet, wenn der Wert der Steuervariablen den spezifizierten Endwert überschreitet.	108
<b>FRAC</b>	Numerische Funktion	Ermittelt die Nachkommastellen des Arguments.	144
<b>FRE</b>	Manueller Befehl	Liefert die Speicherbereich-Größe entsprechend dem Argument.	95
<b>GOSUB</b>	Grundbefehl	Springt zu einem spezifizierten Unterprogramm.	103
<b>GOTO</b>	Grundbefehl	Unbedingte Verzweigung zum spezifizierten Verzweigungsziel.	102
<b>HEX\$</b>	Zeichen-Funktion	Liefert eine Hexadezimal-Zeichenfolge für einen im Argument spezifizierten Dezimalwert.	158
<b>HYP ACS</b>	Numerische Funktion	Berechnet den Wert des entsprechenden inversen Hyperbelfunktion für das Argument.	137
<b>HYP ASN</b>	Numerische Funktion	Berechnet den Wert des entsprechenden inversen Hyperbelfunktion für das Argument.	137
<b>HYP ATN</b>	Numerische Funktion	Berechnet den Wert des entsprechenden inversen Hyperbelfunktion für das Argument.	137
<b>HYP COS</b>	Numerische Funktion	Berechnet den Wert des entsprechenden Hyperbelfunktion für das Argument.	136
<b>HYP SIN</b>	Numerische Funktion	Berechnet den Wert des entsprechenden Hyperbelfunktion für das Argument.	136
<b>HYP TAN</b>	Numerische Funktion	Berechnet den Wert des entsprechenden Hyperbelfunktion für das Argument.	136
<b>IF ~ THEN ~ ELSE IF ~ GOTO ~ ELSE</b>	Grundbefehl	Führt die THEN-Anweisung oder GOTO-Anweisung aus, wenn die spezifizierte Bedingung erfüllt wird. Die ELSE-Anweisung wird ausgeführt, wenn die spezifizierte Bedingung nicht erfüllt wird.	107
<b>INKEY\$</b>	Grundbefehl	Weist eine einzelne Zeicheneingabe von der Tastatur einer Variablen zu.	123
<b>INPUT</b>	Grundbefehl	Weist Daten, die über die Tastatur eingegeben werden, Variablen zu.	122
<b>INPUT#</b>	Ein/Ausgabe-Befehl	Liest Daten aus einer sequentiellen Datei.	168
<b>INPUT\$</b>	Grundbefehl	Weist eine spezifizierte Anzahl Zeichen von der Tastatur einer Variablen zu.	124
<b>INPUT\$</b>	Ein/Ausgabe-Befehl	Liest die spezifizierte Anzahl Zeichen aus einer sequentiellen Datei.	169
<b>INT</b>	Numerische Funktion	Ermittelt die größte ganze Zahl, die den Wert des Arguments nicht überschreitet.	143
<b>LEFT\$</b>	Zeichen-Funktion	Liefert eine spezifizierte Anzahl Zeichen vom linken Ende einer Zeichenfolge.	157
<b>LEN</b>	Zeichen-Funktion	Liefert einen Wert, der die Anzahl der Zeichen einer Zeichenfolge angibt.	158

Befehl/Funktion	Kategorie	Beschreibung	Seite
<b>LET</b>	Grundbefehl	Weist den Wert eines Ausdrucks an der rechten Seite einer Gleichung der Variablen an der linken Seite zu.	111
<b>LIST</b> <b>LIST ALL</b>	Manueller Befehl	Zeigt das gesamte oder einen Teil des gegenwärtigen spezifizierten Programms an.	96
<b>LIST#</b>	Datenbank-Befehl	Zeigt alle Datenbank-Daten an.	174
<b>LLIST</b>	Ein/Ausgabe-Befehl	Ausgabe von Programminhalten an den Drucker.	162
<b>LLIST#</b>	Datenbank-Befehl	Gibt alle Datenbank-Daten auf dem Drucker aus.	175
<b>LN</b>	Numerische Funktion	Berechnet den Wert der entsprechenden Logarithmusfunktion für das Argument	139
<b>LOAD</b> <b>LOAD ALL</b>	Ein/Ausgabe-Befehl	Liest aus einer Datei in den Speicher.	172
<b>LOAD#</b>	Datenbank-Befehl	Liest Daten in den Datenbank-Bereich.	176
<b>LOCATE</b>	Grundbefehl	Bewegt den Cursor zu einer spezifizierten Position auf dem virtuellen Bildschirm.	118
<b>LOG</b>	Numerische Funktion	Berechnet den Wert der entsprechenden Logarithmusfunktion für das Argument	139
<b>LPRINT</b>	Ein/Ausgabe-Befehl	Ausgabe von text an den Drucker.	163
<b>MID\$</b>	Zeichen-Funktion	Liefert eine spezifizierte Anzahl Zeichen von einer spezifizierten Position in einer Zeichenfolge.	155
<b>NCR</b>	Numerische Funktion	Berechnet die Kombination nCr für die Werte von n und r.	147
<b>NEW</b> <b>NEW ALL</b>	Manueller Befehl	Löscht ein Programm.	93
<b>NEW#</b>	Datenbank-Befehl	Löscht Datenbank-Daten.	174
<b>NPR</b>	Numerische Funktion	Berechnet die Permutation nPr für die Werte von n und r.	147
<b>ON ERROR GOTO</b>	Grundbefehl	Spezifiziert die Zeilennummer, zu der die Ausführung verzweigt wird, wenn ein Fehler auftritt.	130
<b>ON GOSUB</b>	Grundbefehl	Sprung zum spezifizierten Unterprogramm entsprechend der spezifizierten Verzweigungsbedingung.	106
<b>ON GOTO</b>	Grundbefehl	Springt zum spezifizierten Verzweigungsziel entsprechend der spezifizierten Verzweigungsbedingung.	105
<b>OPEN</b>	Ein/Ausgabe-Befehl	Eröffnet eine Datei zur Verarbeitung.	165
<b>PASS</b>	Manueller Befehl	Spezifiziert oder löscht ein Passwort.	92
<b>PEEK</b>	Grundbefehl	Liefert den Wert, der in der spezifizierten Speicheradresse gespeichert ist.	127
<b>PI</b>	Numerische Funktion	Gibt den Wert von $\pi$ an.	146
<b>POKE</b>	Grundbefehl	Schreibt Daten in eine spezifizierte Adresse.	128
<b>POL</b>	Numerische Funktion	Wandelt kartesische Koordinaten (x, Y) in Polkoordinaten (r, $\theta$ ) um.	148
<b>PRINT</b>	Grundbefehl	Zeigt Daten auf dem Bildschirm an.	115
<b>PRINT#</b>	Ein/Ausgabe-Befehl	Ausgabe von Daten in eine sequentielle Datei.	167
<b>RAN#</b>	Numerische Funktion	Erzeugt einen Zufallswert im Bereich von 0 bis 1.	145
<b>READ</b>	Grundbefehl	Liest den Inhalt der DATA-Anweisung in den Speicher.	113
<b>READ#</b>	Datenbank-Befehl	Liest Daten aus dem Datenabank-Bereich.	177



Befehl/Funktion	Kategorie	Beschreibung	Seite
<b>REC</b>	Numerische Funktion	Wandelt Polkoordinaten ( $r, \theta$ ) in kartesische Koordinaten ( $x, Y$ ) um.	149
<b>REM ( ' )</b>	Grundbefehl	Ermöglicht das Einfügen von Anmerkungen oder Kommentaren in ein Programm. Dieser Befehl wird nicht ausgeführt.	110
<b>RESTORE</b>	Grundbefehl	Spezifiziert eine DATA-Zeile zum Lesen durch die READ-Anweisung.	114
<b>RESTORE#</b>	Datenbank-Befehl	Sucht bestimmte Daten im Datenbank-Bereich und verändert die Lesereihenfolge von Datenbank-Daten.	178
<b>RESUME</b>	Grundbefehl	Bewirkt Rückkehr aus einer Fehlerabwicklungsroutine zum Hauptprogramm.	131
<b>RETURN</b>	Grundbefehl	Rückkehr der Ausführung von einem Unterprogramm zum Hauptprogramm.	104
<b>RIGHT\$</b>	Zeichen-Funktion	Liefert eine spezifizierte Anzahl Zeichen von der rechten Seite einer Zeichenfolge.	156
<b>ROUND</b>	Numerische Funktion	Rundet das Argument an der spezifizierten Stelle.	144
<b>RUN</b>	Manueller Befehl	Führt ein Programm aus.	98
<b>SAVE SAVE ALL</b>	Ein/Ausgabe-Befehl	Sichert ein Programm in eine spezifizierten Datei.	171
<b>SAVE#</b>	Datenbank-Befehl	Gibt Datenbank-Daten zu einer Datei aus, die durch den Dateideskriptor spezifiziert wurde.	175
<b>SET</b>	Grundbefehl	Spezifiziert das Ausgabeformat von numerischen Daten.	120
<b>SGN</b>	Numerische Funktion	Ermittelt den Wert, der dem Vorzeichen des Arguments entspricht.	142
<b>SIN</b>	Numerische Funktion	Berechnet den Wert des entsprechenden trigonometrischen Funktionswertes für das Argument.	134
<b>SQR</b>	Numerische Funktion	Berechnet die Quadratwurzel des Arguments.	140
<b>STOP</b>	Grundbefehl	Unterbricht die Programmausführung.	101
<b>STR\$</b>	Zeichen-Funktion	Wandelt das Argument (numerischer Wert oder Wert eines numerischen Ausdrucks) in eine Zeichenfolge um.	152
<b>TAB</b>	Grundbefehl	Gibt eine Horizontal-Tabulatorspezifikation zum Bildschirm oder Drucker aus.	117
<b>TAN</b>	Numerische Funktion	Berechnet den Wert des entsprechenden trigonometrischen Funktionswertes für das Argument.	134
<b>TROFF</b>	Manueller Befehl	Hebt den Ablaufverfolgungs-Modus auf.	99
<b>TRON</b>	Manueller Befehl	Spezifiziert den Ablaufverfolgungs-Modus.	99
<b>VAL</b>	Zeichen-Funktion	Wandelt eine numerische Zeichenfolge in einen numerischen Wert um.	153
<b>VALF</b>	Zeichen-Funktion	Führt mit einem numerischen Ausdruck, der als Zeichenfolge ausgedrückt ist, eine Berechnung aus und liefert das Ergebnis.	154
<b>VARLIST</b>	Manueller Befehl	Zeigt Variablennamen und Feldnamen an.	97
<b>VERIFY</b>	Ein/Ausgabe-Befehl	Überprüft den Inhalt einer auf Kassette gespeicherten Datei.	173
<b>WRITE#</b>	Datenbank-Befehl	Überschreibt und löscht Datenbank-Daten.	179

## 4.2 Übersicht der Wissenschaftlichen Bibliothek

Das ist eine Übersicht der Wissenschaftlichen Bibliothek. Die Seitenzahl verweist direkt zur Seite der CASIO-Bedienungsanleitung des Rechners.

Nummer	Programm	Beschreibung	Seite
1000	Speicher- und Schnittstellen-Testprogramm		184
5010	Primfaktoren-Analyse		187
5020	Größter gemeinsamer Teiler, Kleinstes gemeinsames Vielfaches		188
5040	Gleichungssystem (Gauss-Elimination)		189
5050	Quadratische Gleichung		191
5060	Kubische Gleichung		192
5080	Numerische Lösung einer Gleichung (Newton)		194
5090	Numerische Lösung einer Gleichung (Halbierungs-Methode)		196
5100	Matrixoperationen		198
5200	Numerische Integration (Romberg-Methode)		208
5220	Gewöhnliche Differenzialgleichung (Ru.Ku.)		210
5230	Lagrangsche Interpolation		211
5250	Gamma-Funktion		212
5260	Besselsche Funktion $J_n(x)$		213
5270	Besselsche Funktion $Y_n(x)$		214
5280	Modifizierte Besselsche Funktion $I_n(x)$		215
5290	Modifizierte Besselsche Funktion $K_n(x)$		216
5300	Komplexe Zahlen		217
5350	Binär-Dezimal-Hexadezimal		221
5510	Gerade durch zwei Punkte		226
5520	Schnittwinkel von zwei Geraden		227
5530	Abstand zwischen Punkt und Gerade		228
5540	Drehbewegung		229
5550	Kreis durch drei Punkte		230
5560	Länge von Tangenten von einem Punkt zu einem Kreis		231
5570	Tangentialgleichung		232
5600	Dreiecksfläche		234
5605	Fläche eines Trapezoides		235
5610	Fläche eines Parallelogramms		236
5615	Fläche eines Kreises		237
5620	Fläche eines Sektors		238
5625	Fläche eines Segments		239
5630	Fläche einer Ellipse		240
5635	Fläche eines Polygons		240
5650	Flächeninhalt einer Kugel		242

Nummer	Programm	Beschreibung	Seite
5655	Flächeninhalt einer Kugelzone		243
5660	Flächeninhalt eines Kugelsektors		244
5665	Oberfläche eines Kreiszyllinders		245
5670	Oberfläche eines Kreiskegels		246
5675	Flächeninhalt eines Kreiskegelstumpfes		247
5700	Rauminhalt einer Kugel		248
5705	Rauminhalt einer Kugelzone		249
5710	Rauminhalt eines Kugelsektors		250
5715	Rauminhalt eines Kreiszyllinders		251
5720	Rauminhalt eines Kreiskegels		252
5725	Rauminhalt eines Kreiskegelstumpfes		253
5730	Rauminhalt eines Keils		254
5735	Rauminhalt einer Pyramide		255
5740	Rauminhalt eines Pyramidenstumpfes		256
5745	Rauminhalt eines Ellipsoiden		257
5750	Einbeschriebener Kreis und Umbeschriebener Kreis eines Polygons		258
5760	Regelmäßiger Polyeder		259
5800	Zerlegung in Faktoren		261
5810	Trigonometrische Funktionen		262
5820	Differenziale		263
5830	Integrationen		264
5840	Laplace-Transformation		265
5900	Tabelle des Periodischen Systems		267
5910	Wissenschaftliche Konstanten		272
5920	Konstante der elektrolytischen Dissoziation		274
5930	Bewegung und Energie		275
5932	Wellenbewegung		277
5934	Wechselstrom und Gleichstromkreise		278
5936	Elektrische und magnetische Felder		280
5938	Thermodynamik und Anderes		282
5950	Metrische Umwandlungen für Längen		283
5960	Metrische Umwandlungen für Flächen		285
5970	Metrische Umwandlungen für Rauminhalte		286
5980	Metrische Umwandlungen für Gewicht		288
6210	Obere Wahrscheinlichkeitsintegrale (Normalverteilung)		289
6220	Obere Wahrscheinlichkeitsintegrale ( $x^2$ Verteilung)		290
6230	Obere Wahrscheinlichkeitsintegrale (t Verteilung)		291
6240	Obere Wahrscheinlichkeitsintegrale (F Verteilung)		292
6310	Obere Summenhäufigkeit (Binominal-		293

Nummer	Programm	Beschreibung	Seite
	Verteilung)		
6320	Obere Summenhäufigkeit (Poisson-Verteilung)		294
6330	Obere Summenhäufigkeit (Hypergeometrische Verteilung)		295
6410	Prozentpunkt (Normalverteilung)		296
6420	Prozentpunkt ( $\chi^2$ Verteilung)		297
6430	Prozentpunkt (t Verteilung)		298
6440	Prozentpunkt (F Verteilung)		299
6450	Normale Zufallszahlen		300
6460	Exponentielle Zufallszahlen		301
6500	Statistische Berechnungen mit einer Variablen		302
6510	Lineare Regression ( $y=a+bx$ )		305
6520	Logarithmische Regression ( $y=a+b \ln x$ )		308
6530	Exponentielle Regression ( $y=ab^x$ )		311
6540	Potenz-Regression ( $y=ax^b$ )		314
6610	Mittelwert-Intervallschätzung (für bekannte Varianz)		318
6620	Mittelwert-Intervallschätzung (für unbekannte Varianz)		322
6630	Varianz-Intervallschätzung		326
6640	Standardabweichung-Intervallschätzung		330
6650	Varianzquotient-Intervallschätzung		333
6660	Mittelwertdifferenz-Intervallschätzung		338
6670	Verhältnis-Intervallschätzung		343
6680	Verhältnisdifferenz-Intervallschätzung		344
6710	Grundgesamtheits-Mittelwert-Test (zweiseitig): für bek. Varianz)		346
6711	Grundgesamtheits-Mittelwert-Test (rechtsseitig): für bek. Varianz)		350
6712	Grundgesamtheits-Mittelwert-Test (linksseitig): für bek. Varianz)		354
6720	Grundgesamtheits-Mittelwert-Test (zweiseitig): für unbek. Varianz)		358
6721	Grundgesamtheits-Mittelwert-Test (rechtsseitig): für unbek. Varianz)		361
6722	Grundgesamtheits-Mittelwert-Test (linksseitig): für unbek. Varianz)		364
6730	Grundgesamtheits-Varianz-Test (zweiseitig)		367
6731	Grundgesamtheits-Varianz-Test (rechtsseitig)		370
6732	Grundgesamtheits-Varianz-Test (linksseitig)		373
6740	Varianzquotient-Test (zweiseitig)		376
6741	Varianzquotient-Test (rechtsseitig)		381
6742	Varianzquotient-Test (linksseitig)		386

Nummer	Programm	Beschreibung	Seite
6750	Mittelwertdifferenz-Test (zweiseitig)		390
6751	Mittelwertdifferenz-Test (rechtsseitig)		395
6752	Mittelwertdifferenz-Test (linksseitig)		400
6760	Quotiententest (zweiseitig)		405
6761	Quotiententest (rechtsseitig)		406
6762	Quotiententest (linksseitig)		407
6770	Quotientendifferenztest (zweiseitig)		408
6771	Quotientendifferenztest (rechtsseitig)		410
6772	Quotientendifferenztest (linksseitig)		411

**4.3 Index**

Anhang .....	110	Library 5665 .....	52
Anzeige .....	10	Library 5670 .....	52
BASIC-Modus .....	17	Library 5675 .....	52
Bedienungsanleitung .....	18	Library 5700 .....	53
Befehls-Übersicht .....	110	Library 5705 .....	53
Betriebsarten .....	12	Library 5710 .....	53
CAL-Modus .....	13	Library 5715 .....	54
Casio FX-850P .....	9	Library 5720 .....	54
Datenbank-Befehle .....	18	Library 5725 .....	54
Datenbank-Funktion .....	15	Library 5730 .....	55
Eingabe/Ausgabe-Befehle .....	18	Library 5735 .....	55
Formelspeicher-Funktion .....	14	Library 5740 .....	55
Grundbefehle .....	18	Library 5745 .....	56
Library 1000 .....	25	Library 5750 .....	56
Library 5010 .....	26	Library 5760 .....	57
Library 5020 .....	26	Library 5800 .....	58
Library 5040 .....	27	Library 5810 .....	59
Library 5050 .....	28	Library 5820 .....	60
Library 5060 .....	29	Library 5830 .....	61
Library 5080 .....	30	Library 5840 .....	62
Library 5090 .....	31	Library 5900 .....	63
Library 5100 .....	32	Library 5910 .....	65
Library 5200 .....	34	Library 5920 .....	66
Library 5220 .....	35	Library 5930 .....	67
Library 5230 .....	36	Library 5932 .....	68
Library 5250 .....	36	Library 5934 .....	69
Library 5260 .....	37	Library 5936 .....	70
Library 5270 .....	38	Library 5938 .....	71
Library 5280 .....	39	Library 5950 .....	72
Library 5290 .....	40	Library 5960 .....	73
Library 5300 .....	41	Library 5970 .....	74
Library 5350 .....	42	Library 5980 .....	75
Library 5510 .....	44	Library 6210 .....	76
Library 5520 .....	44	Library 6220 .....	76
Library 5530 .....	44	Library 6230 .....	76
Library 5540 .....	45	Library 6240 .....	77
Library 5550 .....	45	Library 6310 .....	77
Library 5560 .....	46	Library 6320 .....	77
Library 5570 .....	47	Library 6330 .....	78
Library 5600 .....	48	Library 6410 .....	78
Library 5605 .....	48	Library 6420 .....	78
Library 5610 .....	49	Library 6430 .....	79
Library 5615 .....	49	Library 6440 .....	79
Library 5620 .....	49	Library 6450 .....	79
Library 5625 .....	50	Library 6460 .....	80
Library 5630 .....	50	Library 6500 .....	80
Library 5635 .....	50	Library 6510 .....	81
Library 5650 .....	51	Library 6520 .....	81
Library 5655 .....	51	Library 6530 .....	81
Library 5660 .....	51	Library 6540 .....	81

Library 6610 .....	82	Speicher .....	10
Library 6620 .....	82	Statistik.....	76
Library 6630 .....	83	Statistik Unterprogramme .....	90
Library 6640 .....	83	TRANSLIB .....	20
Library 6650 .....	84	Übersicht der Wissensch. Bibliothek .....	114
Library 6660 .....	85	.....	114
Library 6670 .....	85	Undokumentierte Funktion 0400 ..	106
Library 6680 .....	86	Undokumentierte Funktion 0800 ..	107
Library 6710 .....	86	Undokumentierte Funktion 0900 ..	108
Library 6711 .....	86	Undokumentierte Funktionen .....	106
Library 6712 .....	86	Unterprogramm M6510.....	90
Library 6720 .....	87	Unterprogramm M6710.....	92
Library 6721 .....	87	Unterprogramm M6720.....	93
Library 6722 .....	87	Unterprogramm M6730.....	94
Library 6730 .....	87	Unterprogramm M6740.....	95
Library 6731 .....	87	Unterprogramm M6750.....	96
Library 6732 .....	87	Unterprogramm M6760.....	97
Library 6740 .....	88	Unterprogramm M6770.....	98
Library 6741 .....	88	Unterprogramm S6020 .....	99
Library 6742 .....	88	Unterprogramm S6030 .....	100
Library 6750 .....	88	Unterprogramm S6210 .....	101
Library 6751 .....	88	Unterprogramm S6220 .....	101
Library 6752 .....	88	Unterprogramm S6230 .....	101
Library 6760 .....	88	Unterprogramm S6240 .....	102
Library 6761 .....	89	Unterprogramm S6310 .....	102
Library 6762 .....	89	Unterprogramm S6320 .....	102
Library 6770 .....	89	Unterprogramm S6330 .....	103
Library 6771 .....	89	Unterprogramm S6410 .....	103
Library 6772 .....	89	Unterprogramm S6420 .....	104
Library K0630 .....	22	Unterprogramm S6430 .....	104
Manuelle Befehle.....	18	Unterprogramm S6440 .....	105
Mathematik.....	25	USB-Interface.....	11
Numerische Funktionen .....	18	Wissenschaftliche Bibliothek .....	16
Preise.....	11	Wissenschaftliche Formeln.....	58
Programmbereiche .....	17	Zeichen-Funktionen.....	18
Prozessor.....	10	Zubehör .....	11
Spannungsversorgung .....	10		

```

CAPS S CAL BASIC DEGRADGRA MEMO IN EDIT LIB 8 8 8 8 DEFM PRT TR STOP
-

CAPS S CAL BASIC DEGRADGRA MEMO IN EDIT LIB 8 8 8 8 DEFM PRT TR STOP
-

CAPS S CAL BASIC DEGRADGRA MEMO IN EDIT LIB 8 8 8 8 DEFM PRT TR STOP
12345678901234567890123456789012
12345678901234567890123456789012
    
```

**Listing 4**    XXXX

**Prg. 4.1**    XXXX

**LIB**

Abbildung 4

Zeile	Beschreibung

Piktogramme:



**Beispiel**



**Hinweis**



**Achtung**



Überblicksinformationen



Handlungsanweisungen



Hintergrundinformationen



Hinweis bzw. Frage



## Casio Basic Library

### Was Sie schon immer über das Casio Scientific Library 116 wissen wollten:

- Einführung
- Voraussetzungen
- Quellcodes der Scientific Library 116



#### Autor

Dipl.-Ing.(FH)  
Manfred Becker

E-Mail: [mani.becker@web.de](mailto:mani.becker@web.de)

URL: <http://manib.ma.funpic.de>

### Hallo liebe Casio Fans,

Dieses Tutorial zeigt Ihnen die Basic-Listings der Scientific Library 116, der Programmbibliothek des Taschenrechners **CASIO FX-850P/880P**.

Sie sollten bereits etwas Erfahrung mit der Casio Basic-Programmierung haben, um die Quelltexte in diesem Tutorial verstehen zu können.

Programmieranfängern empfehle ich mein **Casio Basic Tutorial: Basic-Programmierung für Anfänger mit den Taschenrechner CASIO FX-850P/880P**

Viel Spass dabei!

<http://manib.ma.funpic.de>

ISBN-10: 3-123-45678-0  
ISBN-13: 978-3-123-45678-0

