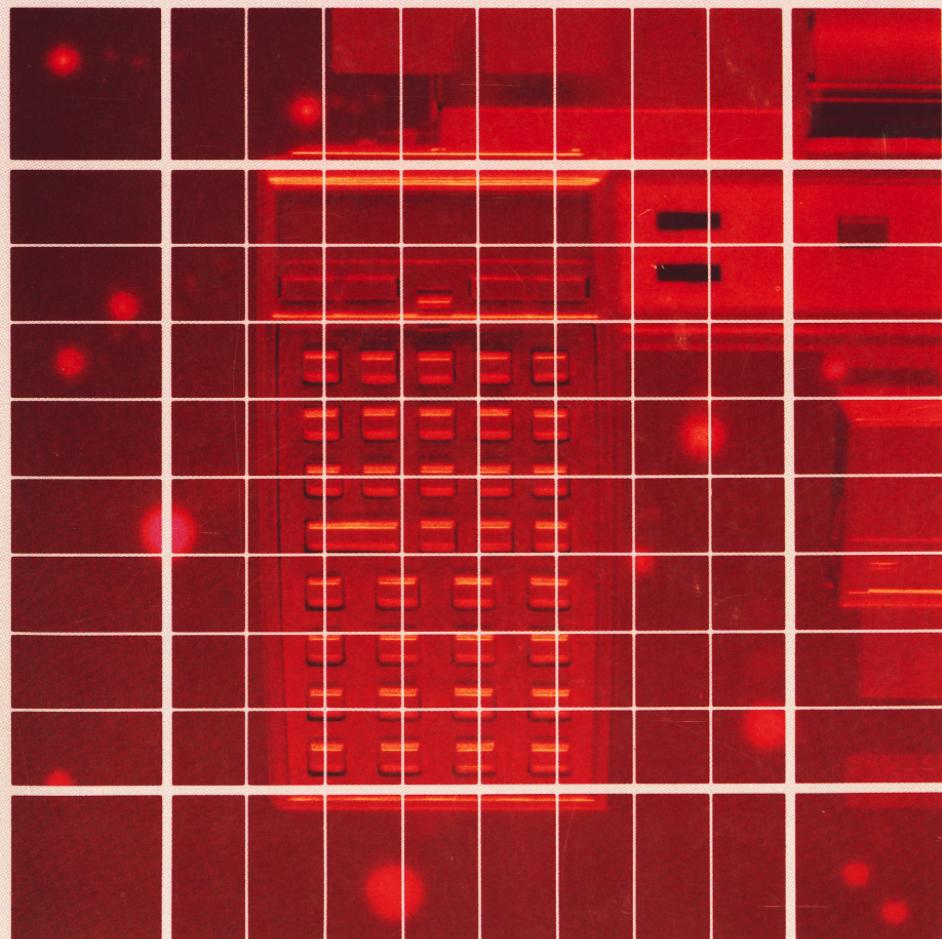


HEWLETT-PACKARD

HP-41C

Standard Programmsammlung





HP-41C

Standard-Programmsammlung

August 1979

00041-90027

Printed in U.S.A.

© Hewlett-Packard Company, 1979

Einleitung

Die in diesem Handbuch enthaltenen Programme veranschaulichen sehr deutlich die Leistungsfähigkeit und Vielseitigkeit Ihres HP-41C im Bereich der rechnergestützten Lösung von Problemen. Sie werden entdecken, daß die Programme nützlich und unterhaltsam, ja sogar fesselnd sind. Indem Sie die Programme eintasten und ausführen, gewinnen Sie praktisch sofort einen Einblick in die fortgeschrittenen Eigenschaften Ihres HP-41 C und die Programme bleiben – dank des Permanent-Speichers – auch für den zukünftigen Gebrauch im Rechner erhalten.

Ein Studium der fachkundig zusammengestellten Programme wird dazu beitragen, auch Ihre Programmierfertigkeit zu entwickeln. Der volle Nutzen des HP-41C ergibt sich aus einer erfinderischen Ausnutzung der Programmierleistung und Vielseitigkeit Ihres Rechners, die Sie speziell Ihren Anforderungen anpassen können.

Jedes der in diesem Handbuch enthaltenen 10 Programme setzt sich aus einer Beschreibung, einer Bedienungsanweisung, einem oder mehreren Zahlenbeispielen, Kernpunkten im Programm und einem Protokoll zusammen. Lesen Sie vor der Benutzung der Programme die Abschnitte „Eingeben eines Programms in den HP-41 C“ und „Aufbau der Bedienungsanweisungen“ am Anfang dieses Buches durch. Sie werden die Programme allerdings verständlicher finden und noch mehr Nutzen aus ihnen ziehen, wenn Sie zuerst das HP-41C Bedienungs- und Programmierhandbuch durcharbeiten.

Nachdem Sie ein Programm zur Ausführung gewählt haben, tasten Sie es gemäß dem Programmprotokoll ein. Die für die richtige Ausführung des Programms benötigte Information ist in den Bedienungsanweisungen enthalten. Sie werden wahrscheinlich nur die ersten paar Male auf diese Information zurückgreifen müssen. Im weiteren Verlauf wird Sie das Programm selber durch die Ausführung leiten, indem es angibt, welche Variablen einzugeben, welche Tasten zu drücken sind und welche Werte ausgegeben werden.

Unter der Rubrik Kernpunkte werden Programmietechniken gezeigt, die Ihr besonderes Interesse verdienen. Ein Studium dieser Techniken wird Ihnen nicht nur den Ablauf von Teilen des Programms verdeutlichen, sondern Ihnen auch zeigen, wie Sie diese Techniken in eigenen Programmen einsetzen können. Schauen Sie sich auch die in den Programmprotokollen enthaltenen Kommentare an, um Ihr Verständnis der einzelnen Programme zu vertiefen und mehr über effiziente und vielseitige Programmietechniken zu erfahren.

Bis auf das Spielprogramm „Black Jack“ reicht die Grundausstattung des HP-41C für die in diesem Handbuch enthaltenen Programme aus. „Black Jack“ benötigt ein zusätzliches Speichermodul. Bei der Erweiterung Ihres HP-41C Systems werden Sie merken, daß sich einige dieser Programme als Grundlage für eigene, längere Programme verwenden lassen. Vielleicht wollen Sie einige Änderungen vornehmen, um die Programme Ihren besonderen Anforderungen anzupassen. Aber gerade Programmierbarkeit gehört zum Sinn und Zweck Ihres HP-41C Rechners.

INHALTSVERZEICHNIS

Einleitung	3
Aufbau der Bedienungsanweisungen	5
Eingeben eines Programms in den HP-41C	6
UPN-Lehrgang	8
Zum Erlernen der UPN-Arbeitsweise wird der Stack angezeigt.	
Kalenderrechnungen	14
Es können fast alle Tages- und Datenumrechnungen ausgeführt werden.	
Wortratespiel	18
Ein verstecktes Wort muß erraten werden.	
Arithmetik-Lehrgang	22
Bei zehn richtigen Antworten ertönt ein Tusch.	
Hexadezimal-Dezimal-Umwandlung	28
Zahlen werden von einem System in das andere umgewandelt.	
Finanztechnische Berechnungen	32
Der HP-41C wird in einen leistungsfähigen Finanz-Rechner umgewandelt.	
Nullstellenbestimmung	38
Die Nullstellen einer Funktion werden schnell und genau berechnet.	
Kurvenanpassung	42
Bis zu vier Kurventypen können an vorgegebene Wertepaare angepaßt werden.	
Vektor-Operationen	50
Berechnungen mit komplexen Zahlen können einfach durchgeführt werden.	
Black Jack	54
Eine vereinfachte Version von "17 und 4". Es wird ein zusätzliches Speichermodul benötigt.	

Aufbau der Bedienungsanweisungen

Zu jedem in diesem Paket enthaltenen Programm sind die Bedienungsanweisungen in Tabellenform angegeben. Sie sind der Leitfaden für die Ausführung der Programme.

Die Tabelle setzt sich aus fünf Spalten zusammen:

Die erste ist mit „Nr.“ bezeichnet und gibt die laufende Nummer des jeweiligen Bedienungsschrittes an. Die Bedienungsanweisungen sind entsprechend dieser Nummerierung Zeile für Zeile zu befolgen.

Die zweite Spalte „Anweisung“ enthält Anweisungen und Kommentare im Zusammenhang mit den auszuführenden Operationen.

In der Spalte „Wert“ sind die einzutastenden Daten, gegebenenfalls deren Einheit oder eine entsprechende Zeichenfolge angegeben. Für die Dateneingabe werden die Zifferntasten 0 bis 9, die Dezimalpunkttaste, sowie **EEX** (für die Eingabe des Exponenten) und **CHS** (für negative Zahlen oder Exponenten) verwendet.

Die Spalte „Funktion“ enthält die Funktionstasten, die im Zusammenhang mit diesem Anweisungsschritt auszuführen sind.

Wenn eine Anweisung in den Spalten „Wert“ oder „Funktion“ goldfarben gedruckt ist, muß vor dem Eintasten der Anweisung der ALPHA-Modus eingeschaltet werden. So bedeutet beispielsweise die Angabe **XEQ A4C**, daß folgende Tasten zu drücken sind: **XEQ ALPHA A 4C ALPHA**. Sie können natürlich auch die Funktion A4C einer beliebigen Taste zuordnen, indem Sie **ASN ALPHA A4C ALPHA [Taste]** drücken. Danach wäre einfach **[Taste]** im USER-Modus zu drücken, um die Funktion auszuführen.

In der Spalte Anzeige finden Sie die Textausgaben, die berechneten Zwischen- und Endergebnisse und, soweit zutreffend, deren Einheiten.

Über der Spalte „Anzeige“ wird in einem Feld die für die Ausführung des Programms benötigte Mindestzahl von Speicherregistern angegeben. Vor dem Eintasten des Programms sollte gegebenenfalls die SIZE-Funktion verwendet werden, damit das Programm im Rechner Platz hat.

Im Bedienungshandbuch wird auf den Seiten 73 und 117 ausführlich beschrieben, wie die Speicherplatzbelegung des Rechners bestimmt werden kann.

Eingeben eines Programms in den HP-41C

Einige Punkte sind zu berücksichtigen, wenn Sie die in diesem Buch enthaltenen Programmprotokolle zum Eingeben eines Programms verwenden. Das Protokoll des HP 82143A Druckers ermöglicht eine praktische und leicht verständliche Eingabe von Programmen, ohne daß jeder einzelne Tastendruck gezeigt werden muß. Diese Druckerausgabe wird in diesem Handbuch verwendet. Wenn Sie erst einmal den Vorgang gemeistert haben, von einem Protokoll ein Programm einzutasten, werden Sie die Methode schnell und einfach anwenden können. Der Ablauf ist wie folgt:

1. Am Ende jedes Programmprotokolls befinden sich Status-Informationen, die für die korrekte Ausführung des Programms benötigt werden. Hier ist auch die Speicherplatzzuteilung enthalten. Bevor Sie das Programm eintasten, drücken Sie **XEQ** **ALPHA** **SIZE** **ALPHA** und geben Sie die Zuteilung an. (Die Eingabe ist dreistellig, so daß beispielsweise 10 als 010 angegeben werden muß.)

Weiter enthalten die Status-Informationen Angaben über das Anzeigeformat und die im Programm verwendeten Flags. Damit das Programm richtig abläuft, überprüfen Sie, ob das richtige Anzeigeformat in HP-41C eingeschaltet ist und ob die entsprechenden Flags je nach Angabe gesetzt oder gelöscht sind.

2. Schalten Sie den HP-41C in den PRGM-Modus (drücken Sie **PRGM**) und drücken Sie dann **■** **STO** **•** **•**, um den Rechner für die Eingabe des neuen Programms vorzubereiten.
3. Geben Sie das Programm ein. Die folgenden Hinweise werden Ihnen bei der Eingabe der Programme von den Protokollen in diesem Buch behilflich sein:
 - a) Wenn sich ein Zeichen oder eine Zeichenfolge des Protokolls in „(Anführungszeichen)“ befindet, so handelt es sich um ALPHA-Zeichen. Und diese einzugeben, drücken Sie einfach **ALPHA**, tasten Sie die Zeichen ein und drücken Sie wiederum **ALPHA**. So würde beispielsweise „SAMPLE“ als **ALPHA** SAMPLE **ALPHA** einge-tastet werden.
 - b) Die Raute vor jeder LBL-Anweisung ist das visuelle Hilfsmittel, um Programmmarken im Protokoll aufzufinden. Bei der Eingabe wird dieses Zeichen nicht beachtet.
 - c) Der Drucker stellt das Divisionszeichen durch / dar. Wenn Sie im Protokoll / sehen, drücken Sie **÷**.
 - d) Der Drucker stellt das Multiplikationszeichen durch * dar. Wenn Sie im Protokoll * sehen, drücken Sie **×**.
 - e) Im Programmprotokoll wird für die **APPEND**-Funktion das Zeichen † verwendet. Wenn Sie † sehen, drücken Sie **■** **APPEND** im ALPHA-Modus (drücken Sie **■** und die Taste K).
 - f) Die folgenden Adressformen sind für alle Operationen, die eine Registeradresse benötigen, zugelassen:
nn (eine zweistellige Zahl)
IND nn (Indirekt: **■**, gefolgt von einer zweistelligen Zahl)
X, Y, Z, T oder L (eine Stack-Adresse: **•** gefolgt von X, Y, Z, T oder L)
IND X, Y, Z, T oder L (eine indirekte Stack-Adresse: **■** **•** gefolgt von X, Y, Z, T oder L)

Indirekte Adressen werden durch **■** gefolgt von der indirekten Adresse angegeben.
 Stack-Adressen werden durch **□** gefolgt von X, Y, Z, T oder L angegeben. Indirekte Stack-Adressen werden durch **■ □** gefolgt von X, Y, Z, T oder L angegeben.

Druckerprotokoll

```

01 ♦LBL "SAM
PLE"
02 "THIS IS
A"
03 "SAMPLE
"
04 AVIEW
05 6
06 ENTER↑
07 -2
08 /
09 ABS
10 STO IND
L
11 "R3="
12 ARCL 03
13 AVIEW
14 RTN

```

Tastenfolge

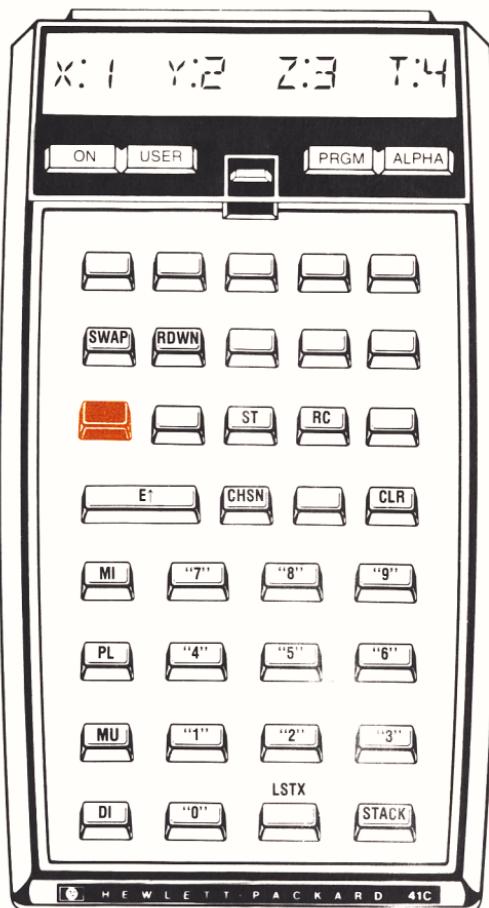
■	LBL	ALPHA	SAMPLE	ALPHA
■	ALPHA	THIS IS A	ALPHA	
■	ALPHA	■ APPEND	SAMPLE	
■	AVIEW	ALPHA		
6				
	ENTER↑			
2	CHS			
	+			
	XEQ	ALPHA	ABS	ALPHA
	STO	■	•	L
	ALPHA	R3=	■	ARCL 03
	■	AVIEW		
	ALPHA			
	■	RTN		

Anzeige

01 LBL^T SAMPLE
02^T THIS IS A
03^T SAMPLE
04 AVIEW
05 06
06 ENTER ↴
07 -2
08 /
09 ABS
10 STO IND L
11 R3=
12 ARCL 03
13 AVIEW
14 RTN

UPN-Lehrgang

Dieses Programm verdeutlicht die Arbeitsweise von UPN - (Umgekehrte Polnische Notation). Der Inhalt der vier Register des Stacks wird gleichzeitig angezeigt, so daß die Wirkung einer gegebenen Tastenfolge nicht abgeleitet werden muß, sondern vielmehr beobachtet werden kann. Die vorgesehenen Funktionen sind auf dem Tastenfeld angegeben. Die entsprechende Zuordnung ist in den Bedienungsanweisungen enthalten. Alle Funktionen übergeben die Programmkontrolle an eine Routine, die den Stack anzeigt. Es ist möglich, die Wirkung von Funktionen auf den Stack zu beobachten, die im Programm nicht enthalten sind. Führen Sie die gewünschte Funktion einfach aus und drücken Sie dann die **R/S**-Taste, der das Programm STACK zugeordnet ist. Es bestehen nur zwei Unterschiede zwischen dem eigentlichen und dem simulierten Rechner: es dürfen nur einstellige Zahlen eingegeben werden und STO und RCL beziehen sich nur auf ein einziges Register, so daß die Angabe einer Adresse entfällt.



Speichergröße: 001

Nr.	Anweisung	Wert	Funktion	Anzeige
1				
2	<p>Stellen Sie den Status ein und laden Sie das Programm. Führen Sie die Zuordnung der Routinen wie gezeigt aus und schalten Sie in den USER-Modus. Die vorgeschlagene Zuordnung ergibt die auf der vorhergehenden Seite gezeigte Tastaturanordnung.</p> <p>SWAP x:y ST STO RDWN R↓ E↑ ENTER↑ RC RCL CLR ↶ CHSN CHS PL ↑ MI ⊖ MU ⤒ DI ⤓ 9 9 8 8 7 7 6 6 5 5 4 4 3 3 2 2 1 1 0 0 LSTX LASTX STACK R/S</p>			
3	Drücken Sie eine gewünschte Tastenfolge und beobachten Sie, wie sich der Inhalt des Stacks verändert.			
4	<p>Die Funktion RUP und CLSTK werden durch und erreicht. (Sie können diese Funktionen aber auch zuordnen.)</p> <p>Um z.B. die Funktion FCN der ✓x Taste zuzuordnen.</p>		XEQ RUP XEQ CLSTK ASN ALPHA FCN ALPHA ✓x	

Beispiel 1:

Berechnen Sie den Ausdruck

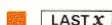
$$\frac{(2 + B) b}{8 - b}$$

für $b = 3$ **Tastenfolge:****Funktionen:**

XEQ **ALPHA** CLSTK **ALPHA**
2
ENTER↑

Anzeige:

X:0 Y:0 Z:0 T:0
X:2 Y:0 Z:0 T:0
X:2 Y:2 Z:0 T:0

3
 +
 LAST X
 ×
 8
 LAST X
 -
 ÷

X:3 Y:2 Z:0 T:0
 X:5 Y:0 Z:0 T:0
 X:3 Y:5 Z:0 T:0
 X:15 Y:0 Z:0 T:0
 X:8 T:15 Z:0 T:0
 X:3 Y:8 Z:15 T:0
 X:5 Y:15 Z:0 T:0
 X:3 Y:0 Z:0 T:0

Beispiel 2:

Belassen Sie die vorangegangenen Ergebnisse und berechnen Sie

$$\frac{2 + 4 (9 - 7)}{6 - 4}$$

Funktion
 9
 ENTER↑

Anzeige
 X:9 Y:3 Z:0 T:0
 X:9 Y:9 Z:3 T:0

7
 -
 4
 ×
 2
 +
 6
 ENTER↑
 4
 -
 ÷

X:7 Y:9 Z:3 T:0
 X:2 Y:3 Z:0 T:0
 X:4 Y:2 Z:3 T:0
 X:8 Y:3 Z:0 T:0
 X:2 Y:8 Z:3 T:0
 X:10 Y:3 Z:0 T:0
 X:6 Y:10 Z:3 T:0
 X:6 Y:6 Z:10 T:3
 X:4 Y:6 Z:10 T:3
 X:2 Y:10 Z:3 T:3
 X:5 Y:3 Z:3 T:3

Nach einem **ENTER↑** wird der Stack nicht angehoben, wenn ein neuer Wert eingegeben wird

Beachten Sie, daß das Ergebnis von Beispiel 1 die Berechnung in Beispiel 2 nicht beeinflußt hat

Beispiel 3:

Bestimmen Sie die polare Form der komplexen Zahl $3 + 4i$

4
[ENTER]
3
[R-P]
STACK

X:4 Y:5 Z:3 T:3
X:4 Y:4 Z:5 T:3
X:3 Y:4 Z:5 T:3
5
X:5 Y:53 Z:5 T:3

Bedenken Sie, daß
STACK der [R/S] -
Taste zugeordnet ist

Ein Kernpunkt des Programms

Besonders nützlich ist in diesem Programm die zur Anzeige der Stack-Register verwandte Routine STACK. Vielleicht sollten Sie diese Routine im Rechner gespeichert lassen, damit Sie bei der Lösung Ihrer eigenen Probleme den gesamten Stack anzeigen können.

01♦LBL "CLS TK"		50 FS?C 05	Eingabe 0.
02 CLST	Stack löschen	51 CLX	
03 GTO 14	x löschen wenn Stack-Lift nicht wirksam ist.	52 0	Stack-Lift wirksam machen
04♦LBL "1"		53 GTO 14	
05 FS?C 05		54♦LBL 13	
06 CLX		55 CF 05	
07 1		56♦LBL 14	
08 GTO 14		57♦LBL "STA CK"	
09♦LBL "2"	Eingabe 1	58 "X:"	
10 FS?C 05	Siehe Anmerkung	59 ARCL X	Stack anzeigen
11 CLX		60 "F Y:"	
12 2		61 ARCL Y	
13 GTO 14	Eingabe 2	62 "F Z:"	
14♦LBL "3"		63 ARCL Z	
15 FS?C 05		64 "F T:"	
16 CLX		65 ARCL T	
17 3		66 AVIEW	
18 GTO 14	Eingabe 3	67 RTN	
19♦LBL "4"		68♦LBL "ET"	Stack-Lift unwirksam machen
20 FS?C 05		69 SF 05	
21 CLX		70 ENTER↑	
22 4		71 GTO 14	
23 GTO 14	Eingabe 4	72♦LBL "RDW N"	
24♦LBL "5"		73 RDN	Zyklisch nach unten vertauschen
25 FS?C 05		74 GTO 13	
26 CLX		75♦LBL "SWA P"	
27 5		76 X<>Y	x und y vertauschen
28 GTO 14	Eingabe 5	77 GTO 14	
29♦LBL "6"		78♦LBL "RUP	
30 FS?C 05		"	
31 CLX		79 RT	Zyklisch nach oben vertauschen
32 6		80 GTO 13	
33 GTO 14	Eingabe 6	81♦LBL "PL"	Addieren
34♦LBL "7"		82 +	
35 FS?C 05		83 GTO 13	Subtrahieren
36 CLX		84♦LBL "MI"	
37 7		85 -	
38 GTO 14	Eingabe 7	86 GTO 13	
39♦LBL "8"		87♦LBL "MU"	
40 FS?C 05		88 *	Multiplizieren
41 CLX		89 GTO 13	
42 8	Eingabe 8	90♦LBL "DI"	
43 GTO 14		91 /	Dividieren
44♦LBL "9"		92 GTO 13	
45 FS?C 05		93♦LBL "CLR	
46 CLX		"	
47 9	Eingabe 9	94 SF 05	
48 GTO 14			
49♦LBL "0"			

R00 Speicher

<pre> 95 CLX 96 GTO 14 97+LBL "CHS N" 98 CHS 99 GTO 14 100+LBL "ST" 101 STO 00 102 GTO 14 103+LBL "RC" 104 FS?C 05 105 CLX 106 RCL 00 107 GTO 14 108+LBL "LST X" 109 FS?C 05 110 CLX 111 LASTX 112 GTO 14 </pre>	<p>Stack-Lift unwirksam machen und x löschen.</p> <p>Vorzeichen wechseln</p> <p>Speichern.</p> <p>x löschen wenn Stack-Lift unwirksam ist. Zurückrufen.</p> <p>Diese Anweisung braucht nicht einge-tastet zu werden.</p>		
--	--	--	--

Wichtige Status-Informationen:

Speichergröße = 001
FIX 0

Flags:

F05 gesetzt = Stack-Lift unwirksam
F29 gelöscht zur Unterdrückung des Dezimalpunktes (-kommas)

Anmerkung: Es ist praktisch, FS?C einer Taste, sagen wir der **[LN]**-Taste, durch **[ASN] [ALPHA]** FS?C **[ALPHA]** **[LN]** zuzuordnen. Sie verwenden dann **[LN]** um erst FS?C__ in der Anzeige zu erhalten und nochmal um FS?C 05 zu erlangen. Beachten Sie, daß sich der Rechner im USER-Modus befinden muß, da Sie sonst zwei **[LN]** erhalten.

Kalenderrechnungen

Dieses Programm berechnet wahlweise Kalenderdaten oder die zwischen Kalenderdaten liegende Anzahl von Tagen. Entweder wird der Zeitraum zwischen zwei vorgegebenen Kalenderdaten in Tagen berechnet oder es wird ein Kalenderdatum aus einem Anfangsdatum und der Anzahl dazwischenliegender Tage berechnet.

Kalenderdaten werden in der Form mm.ttjjjj eingegeben und in der Form MONAT tt, jjjj ausgegeben.

Weiter kann für ein Kalenderdatum der dazugehörige Wochentag berechnet und in Textform angezeigt werden.

Speichergröße: 010				
Nr.	Anweisung	Wert	Funktion	Anzeige
1	Stellen Sie den Status ein, geben Sie das Programm ein und schalten Sie den User-Mode ein WOCHENTAG			
2a	Geben Sie ein Datum ein und berechnen Sie den Wochentag	Datum*	[E]	Wochentag
3a	Wiederholen Sie Schritt 2a für ein neues Datum ANZAHL DER TAGE ZWISCHEN DEN DATEN			
2b	Geben Sie zwei der folgenden Werte ein: Erstes Datum Zweites Datum Anzahl der Tage zwischen den Daten (T)	T 1* T 2* T	[A] [B] [C]	Tag 1* Tag 2* T
3b	Berechnen Sie einen der folgenden Werte: Erstes Datum Zweites Datum Anzahl der Tage zwischen den Daten		[A] [B] [C]	Tag 1* Tag 2* T
4	Wiederholen Sie Schritt 3b für neue Daten (Werte, die sich nicht verändern, brauchen nicht erneut eingegeben zu werden.) *Daten werden in der Form mm, ttjjjj eingegeben und in der Form Monat tt, jjjj ausgegeben.			

Beispiel 1:

Auf welchen Wochentag fiel der 19. Februar 1946?

Tastenfolge:

2.191946 [E]

Anzeige:

TUESDAY (Dienstag)

Beispiel 2:

Welches Datum erhält man, wenn auf den 4. August 1978 10000 Tage dazuaddiert werden?

Tastenfolge:

8.041978 [A] 10000 [C][B]

Anzeige:

DEC 20,2005

Beispiel 3:

Ein Mann, der am 18. Dezember 1913 geboren wurde, hat einen Sohn, der am 19. Februar 1946 zur Welt kam. An welchem Tag wird der Vater doppelt so alt wie sein Sohn sein?

Tastenfolge:

12.181913 [A]
2.191946 [B]
[C]
2 [X] [C][B]

Anzeige:

DEC 18,1913
FEB 19,1946
11751
APR 23,1978

Anzahl der Tage.
Die doppelte
Anzahl Tage nach
dem ersten Datum.

Ein Kerpunkt des Programms

Dieses Programm nutzt die Eigenschaft des "austauschbaren Dezimalpunktes/-kommas", um die Datumsanzeige zu formatieren. Wenn sich das Datum in Form mm.ttjjjj im X-Register befindet, wird mit **[XEQ] IND X** ein Unterprogramm ausgeführt, das eine aus drei Zeichen bestehende Monatsbezeichnung im ALPHA-Register beläßt. Dann wird der Dezimalteil der Zahl im X-Register mit 100 multipliziert, das Flag für den Dezimalpunkt gelöscht und der Tag und das Jahr an den Inhalt der ALPHA-Anzeige angefügt. Somit ergibt ein ursprünglicher X-Wert von 12.251978 eine Anzeige von DEC 25,1978.

Anmerkung: Aufgrund seiner Länge wurde dieses Programm nur unter Verwendung von „lokalen Marken“ geschrieben. Sollte der Rechner einmal außerhalb des Programms an irgendeiner anderen Stelle des Programmspeichers stehen, kann er mit Hilfe von CAT 1 (siehe Bedienungs-Handbuch) wieder entsprechend positioniert werden.

16 Kalenderrechnungen

01♦LBL A 02 RCL 04 03 RCL 01 04 - 05 3 06 GTO 20 07♦LBL B 08 RCL 03 09 RCL 01 10 + 11 4 12♦LBL 20 13 STO 02 14 RDN 15 365.25 16 STO 05 17 30.6001 18 STO 06 19 RDN 20 RDN 21 FS?C 22 22 GTO 21 23 STO IND 02 24 122.1 25 - 26 RCL 05 27 / 28 INT 29 STO 09 30 RCL 05 31 * 32 INT 33 RCL IND 02 34 - 35 CHS 36 STO 00' 37 RCL 06 38 / 39 INT 40 STO 07 41 RCL 00 42 X<>Y 43 RCL 06 44 * 45 INT 46 - 47 STO 08 48 RCL 07 49 1 50 RCL 08 51 %	△ Tage berechnen und Code 3 in die Anzeige schreiben. △ Tage berechnen und Code 4 in die Anzeige schreiben. Steuercode speichern. Konstanten speichern. △ Tage in Anzeige schreiben. △ Tage entsprechend Steuercode speichern. Monatstag berechnen.	52 - 53 - 54 RCL 07 55 14 56 / 57 XEQ 22 58 RCL 09 59 1 E6 60 / 61 + 62 GTO 25 63♦LBL 21 64 RDN 65 FC? 06 66 STO IND 02 67 ENTER↑ 68 INT 69 STO 07 70 - 71 1 E2 72 * 73 ENTER↑ 74 INT 75 STO 08 76 - 77 1 E4 78 * 79 STO 09 80 RCL 07 81 1 82 + 83 ENTER↑ 84 1/X 85 .7 86 + 87 CHS 88 XEQ 22 89 RCL 06 90 * 91 INT 92 RCL 09 93 RCL 05 94 * 95 INT 96 + 97 RCL 08 98 + 99 X<> IND 02 100 FS?C 06 101 RTN 102♦LBL 25	Kalenderdaten in die einzelnen Komponenten mm.tt.jjjj zerlegen. Tageszahl berechnen.
--	---	---	---

R00 = Zwischenspeicherung

R05 = 365.25

R01 = △ Tage

R06 = 30.600

R02 = Zeiger

R07 = m

R03 = Tag # 1

R08 = t

R04 = Tag # 2

R09 = j

103 ENTER† 104 XEQ IND X 105 FRC 106 1 E2 107 * 108 CF 28 109 FIX 4 110 ARCL X 111 RDN 112 AVIEW 113 SF 28 114 RTN 115♦LBL 22 116 INT 117 ST+ 09 118 12 119 * 120 - 121 RTN 122♦LBL C 123 CF 29 124 FIX 0 125 ST0 01 126 FS?C 22 127 RTN 128 RCL 04 129 RCL 03 130 - 131 ST0 01 132 RTN 133♦LBL E 134 SF 06 135 SF 22 136 RCL 05 137 5 138 XEQ 20 139 RCL IND 02 140 7 141 MOD 142 13 143 + 144 XEQ IND X 145 AVIEW 146 RTN 147♦LBL 13 148 "FRIDAY" 149 RTN 150♦LBL 14 151 "SATURDA Y" 152 RTN	Wochentag berechnen: 153♦LBL 15 154 "SUNDAY" 155 RTN 156♦LBL 16 157 "MONDAY" 158 RTN 159♦LBL 17 160 "TUESDAY" " 161 RTN 162♦LBL 18 163 "WEDNESD AY" 164 RTN 165♦LBL 19 166 "THURSDA Y" 167 RTN 168♦LBL 01 169 "JAN " 170 RTN 171♦LBL 02 172 "FEB " 173 RTN 174♦LBL 03 175 "MAR " 176 RTN 177♦LBL 04 178 "APR " 179 RTN 180♦LBL 05 181 "MAY " 182 RTN 183♦LBL 06 184 "JUN " 185 RTN 186♦LBL 07 187 "JUL " 188 RTN 189♦LBL 08 190 "AUG " 191 RTN 192♦LBL 09 193 "SEP " 194 RTN 195♦LBL 10 196 "OCT " 197 RTN 198♦LBL 11 199 "NOV " 200 RTN 201♦LBL 12 202 "DEC "	
---	---	--

Wichtige
Status-Informationen

Speichergröße = 010
FIX 4

Verwendete Flags:
F06
F28

Wortratespiel

Dieses Programm ist eine Version des Wortspiels „Henker“. Der erste Spieler denkt sich ein aus sechs Zeichen bestehendes Wort aus und übergibt es dem Rechner. Der zweite Spieler versucht das Wort zu erraten, indem er verschiedene Buchstaben probiert, bis das Wort vollständig ist. Nach jedem Versuch zeigt der Rechner alle richtig erratenen Buchstaben an der entsprechenden Stelle in der Anzeige an. Wenn das Wort vollständig erraten ist, zeigt der Rechner die Anzahl der benötigten Versuche an.

Speichergröße: 019				
Nr.	Anweisung	Wert	Funktion	Anzeige
1	Stellen Sie den Status ein und geben Sie das Programm ein.			
2	Starten Sie das Programm.		XEQ WORDS	KEY IN WORD LETTER?
3	Der erste Spieler gibt ein Wort ein	sechs beliebige Zeichen	[R/S]	
4	Der zweite Spieler errät ein Zeichen.	ein beliebiges Zeichen	[R/S]	Die bis jetzt erratenen Buchstaben LETTER?
5	Wiederholen Sie Schritt 4, bis das Wort vollständig ist. Darauf erscheint DONE in der Anzeige, dann WORD IS <word> und YOU TOOK nn GUESSES.			

Beispiel:

Geben Sie „HP-41C“ ein und erraten Sie dann das Wort.

Tastenfolge:

XEQ ALPHA WORDS ALPHA
HP-41C R/S

A R/S

P R/S

C R/S

H R/S

4 R/S

Anzeige:

KEY IN WORD LETTER?

LETTER?

P

LETTER?

P C

LETTER?

HP C

LETTER?

HP 4 C

LETTER?

Beachten Sie,
daß der Rechner
im ALPHA-Modus
anhält!)

■ 1 [R/S]

■ - [R/S]

HP 41C
LETTER?
DONE
WORD IS <HP-41C>
YOU TOOK 7 GUESSES

Kernpunkte des Programms

Bei der Entwicklung dieses Programms wurden die zwei Spezialroutinen SPEL und DESPEL verwendet. Sie dienen dazu, ein Wort aus einer Anzahl Buchstaben aufzubauen und ein Wort in seine Einzelbuchstaben zu zerlegen. In dem endgültigen Programm ist nur noch DESPEL vorhanden, da die Funktion von SPEL im Programm durch eine Vergleichsroutine für Buchstaben übernommen wurde.

Die Art der Verwendung von SPEL und DESPEL in diesem Programm erfordert, daß ein Code im X-Register übergeben wird. Dieser Code bestimmt für SPEL die Quelladresse und für DESPEL die Zieladresse der Buchstaben.

Der Code hat folgende Form:

	fl. 011 für SPEL
oder	ll.0ff für DESPEL
wobei	fl das Register für den ersten Buchstaben
	ll das Register für den letzten Buchstaben
und	ff = fl-1
ist.	

Mit SPEL und DESPEL oder ähnlichen Routinen können eine ganze Reihe unterschiedlicher Zeichenketten codiert und decodiert werden. Eine ähnliche Routine wurde für das hexadezimale Umwandlungsprogramm (siehe Seite 28) verwendet.

01♦LBL "SPE L"	setzt ein gelösches ALPHA-Register	01♦LBL "DES PEL"	speichert den Zähler ll.Off
02 STO 07	voraus	02 STO 07	rettet das Wort
03♦LBL 08	speichert den Zähler	03 ASTO 00	rettet alle bis auf
04 ARCL IND	fl.011	04♦LBL 07	den letzten Buch-
07	setzt das Wort	05 " "	staben
05 ISG 07	zusammen;	06 ARCL 00	rettet den letzten
06 GTO 08	wenn es nicht der	07 ASTO 00	Buchstaben
07 RTN	letzte Buchstabe ist	08 ASHF	wenn es nicht der
	wird die Schleife	09 ASTO IND	letzte Buchstabe
	wiederholt	07	ist, wird Schleife
		10 DSE 07	wiederholt
		11 GTO 07	
		12 RTN	

20 Wortratespiel

01♦LBL "WOR DS" 02 "KEY IN WORD" 03 AON 04 PROMPT 05 ASTO 08 06 6 07 XEQ "DES PEL" 08 . 09 STO 17 10 " 11 ASTO 09 12 16.01 13 XEQ "DES PEL" 14♦LBL "LTT R" 15 CLA 16 ASTO 09 17 "LETTER? " 18 AON 19 PROMPT 20 ASTO 10 21 ISG 17 22 1.006 23 STO 18 24♦LBL 06 25 " " 26 ASTO Y 27 RCL 18 28 10 29 + 30 CLA 31 ARCL IND X 32 RDN 33 ASTO X 34 X=Y? 35 GTO 00 36 CLA 37 ARCL 10 38 ASTO Y 39 CLA 40 ARCL IND 18 41 ASTO X 42 X=Y?	Verstecktes Wort speichern. Buchstaben in R01 bis R06 speichern. Leerzeichen in R11 bis R16.	43 GTO 00 44 " 45 ASTO X 46♦LBL 00 47 CLA 48 ARCL 09 49 ARCL X 50 ASTO 09 51 AVIEW 52 10 53 RCL 18 54 + 55 CLA 56 ARCL Y 57 ASTO IND X 58 ISG 18 59 GTO 06 60 CLA 61 ARCL 08 62 ASTO Y 63 CLA 64 ARCL 09 65 ASTO X 66 X=Y? 67 GTO 00 68 PSE 69 PSE 70 GTO "LTT R" 71♦LBL 00 72 "DONE" 73 AVIEW 74 "WORD IS <" 75 ARCL 09 76 ">" 77 AVIEW 78 PSE 79 PSE 80 RCL 17 81 INT 82 "YOU TOO K" 83 ARCL X 84 "F GUESS ES" 85 AVIEW 86 RTN 87♦LBL "DES PEL"	i anzeigen; sonst Leerzeichen anzeigen. Einen Buchstaben der Anzeige hinzufügen. Schleife sechsmal wiederholen. Wenn die Worte identisch sind, dann DONE. Andernfalls eine erneute Eingabe verlangen. Wort anzeigen. Anzahl Versuche anzeigen.
--	--	---	---

R00 = Zwischenspeicher
R01 = 1. Buchstabe, GW
R02 = 2. Buchstabe, GW
R03 = 3. Buchstabe, GW
R04 = 4. Buchstabe, GW
R05 = 5. Buchstabe, GW
R06 = 6. Buchstabe, GW

R07 = Zähler
R08 = Verstecktes Wort (GW)
R09 = Spielers Wort (SW)
R10 = Augenblicklicher Buchstabe.
R11 = 1. Buchstabe, SW
R12 = 2. Buchstabe, SW
R13 = 3. Buchstabe, SW

<pre>88 STO 07 89 ASTO 00 90♦LBL 07 91 " " 92 ARCL 00 93 ASTO 00 94 ASHF 95 ASTO IND 07 96 DSE 07 97 GTO 07 98 RTN</pre> <p>Wichtige Status-Information Speichergröße = 019 FIX 0 CF 29</p> <p>Flags: F29 gelöscht, um Dezimalpunkt zu unterdrücken</p>	Unterprogramm zur Zerlegung des Wortes in seine Buchstaben.		
<p>R14 = 4. Buchstabe, SW R15 = 5. Buchstabe, SW R16 = 6. Buchstabe, SW R17 = Zähler R18 = Zähler</p>			

Arithmetik-Lehrgang

Dieses Programm erstellt Übungsaufgaben für die vier Grundrechenarten. Sie können selber den Schwierigkeitsgrad bestimmen und die Wahl der Lektion, ob Addition, Subtraktion, Division oder Multiplikation. Eine Lektion besteht aus 10 Aufgaben, wonach die Anzahl der richtigen Lösungen in Prozent angezeigt wird.

Das Programm wird mit [XEQ] [ALPHA] TEACH [ALPHA] gestartet. Der Rechner verlangt die größte in den Problemen zu verwendende Zahl. Nach der Eingabe dieser größten Zahl und [R/S] erscheint „+, -, ×, /?“ in der Anzeige, wobei der ALPHA-Indikator aufleuchtet. Drücken Sie jetzt einfach die goldene Umschalter-Taste, eine der arithmetischen Funktionen und [R/S] um mit der Übung zu beginnen. Der ALPHA-Modus wird automatisch ausgeschaltet.

Sobald eine Aufgabe gestellt ist, tasten Sie Ihre Antwort ein und drücken Sie [R/S]. Eine richtige Antwort wird mit YES belohnt, worauf eine neue Aufgabe gestellt wird. Nach einer falschen Antwort ertönt ein unangenehmes Geräusch und die Meldung NO erscheint in der Anzeige. Sie können jetzt einen zweiten Versuch machen, die Aufgabe zu lösen. Der Rechner zeigt die richtige Lösung an, wenn eine Aufgabe zweimal falsch berechnet wurde und fährt dann mit der nächsten Aufgabe fort. Wenn alle zehn Aufgaben beim ersten Versuch richtig gelöst wurden, wird ein Tusch gespielt. Das Programm springt dann wieder zu der Anzeige „+, -, ×, /?“ zurück.

Bei Beginn des Programms steht im X-Register eine Zahl (ein Startwert zwischen 0 und 1) der die Sequenz der vom Rechner gestellten Aufgaben festlegt. Um eine bestimmte Serie von Aufgaben zu wiederholen, wird einfach derselbe Startwert verwendet. Wird kein Startwert eingegeben, verwendet der Rechner die gerade im X-Register enthaltene Zahl.

Speichergröße: 010				
Nr.	Anweisung	Wert	Funktion	Anzeige
1	Stellen Sie den Status ein und geben Sie das Programm ein.			
2	Geben Sie einen Startwert ($0 \leqslant \text{Wert} < 1$) ein und starten Sie das		[XEQ] TEACH [R/S]	MAX NUMBER? +, -, ×, /?
3	Geben Sie die größte zu verwendende Zahl ein.	Startwert N		Angezeigte Aufgaben: $(n_1) + (n_2) = ?$ $(n_1) - (n_2) = ?$ $(n_1) \times (n_2) = ?$ $(n_1) / (n_2) = ?$
4	Wählen Sie eine Lektion Addition Subtraktion Multiplikation Division	+ - × /	[R/S] [R/S] [R/S] [R/S]	
5	Tasten Sie Ihre Antwort ein.	Antwort	[R/S]	YES oder NO
6	Nach 10 Aufgaben erscheint die Anzahl der richtigen Lösungen in Prozent. Sie können für eine weitere Lektion mit Schritt A fortfahren.			(Treffer)% RIGHT

Beispiel:

Geben Sie einen Startwert von .021946 ein und üben Sie dann Subtrahieren, wobei Sie einen Höchstwert von 14 verwenden.

Tastenfolge:

.021946
XEQ ALPHA TEACH ALPHA
 14 **R/S**
– R/S
 7 **R/S**
 1 **R/S**
 8 **R/S**
 7 **R/S**
 3 **R/S**
 6 **R/S**
 8 **R/S**
 11 **R/S**
 1 **R/S**
 4 **R/S**
 3 **R/S**
 4 **R/S**

Anzeige:

MAX NUMBER?
 $+, -, *, ., /?$
 $12 - 5 = ?$
YES
 $14 - 13 = ?$
YES
 $13 - 6 = ?$
NO $13 - 6 = ?$
YES
 $14 - 11 = ?$
YES
 $14 - 7 = ?$
NO $14 - 7 = ?$
NO $14 - 7 = 7$
 $13 - 2 = ?$
YES
 $14 - 13 = ?$
YES
 $14 - 10 = ?$
YES
 $12 - 9 = ?$
YES
 $14 - 10 = ?$
YES
90% RIGHT
 $+, -, *, ., /?$

Kernpunkte im Programm

In diesem Programm wird in einer kombinierten Anwendung gezeigt, wie der Rechner ein eingegebenes ALPHA-Zeichen für einen indirekten Unterprogrammaufruf und eine Textausgabe verwendet.

Während des Ablaufs des Programms wird vom Anwender das Symbol +, -, * oder /, abhängig von der Art der durchzunehmenden Lektion, als Eingabe verlangt. Das Programm speichert dieses Symbol im Register 06, erzeugt dann zwei Zahlen und führt das Unterprogramm aus, dessen Bezeichnung in R06 gespeichert ist. Dasselbe Symbol wird dann zurückgerufen und für den Aufbau der Anzeige verwendet, um die Art der Lektion anzuzeigen.

Das Programm enthält ferner einen interessanten Zufallsgenerator. Der Generator

$$r_{n+1} = \text{FRC} (9821 \times r_n + .211327)$$

wurde in einem Programm von Don Malm in der HP-65 Anwenderbibliothek gefunden. Der Generator besteht den Spektraltest (Knuth, V2, § 3.4) und da die Parameter dem Lehrsatz A (op. cit. S. 15) genügen, werden unabhängig von der Wahl von r_0 eine Million unterschiedliche Zufallszahlen zwischen 0 und 1 erzeugt.

Da der Zufallsgenerator Zahlen zwischen 0 und 1 erzeugt, müssen diese Zahlen weiterverarbeitet werden, um die für die arithmetischen Probleme erforderlichen Ganzzahlen zu erhalten. Indem die Zufallszahlen mit einer Ganzzahl N multipliziert werden, und dann nur der ganzzahlige Teil verwendet wird, können Zahlen zwischen 0 und N-1 erzeugt werden. In diesem Programm wird die vom Anwender angegebene Höchstzahl plus 1 verwendet, um Zahlen zwischen 0 und dieser Höchstzahl zu erzeugen.

<pre> 01♦LBL "TER CH" 02 CF 29 03 FIX 0 04 STO 00 05♦LBL A 06 "MAX NUM BER?" 07 PROMPT 08 1 09 + 10 STO 04 11♦LBL "AGN " 12 0 13 STO 08 14 STO 09 15 10 16 STO 07 17 "+, -, *" 18 √? 19 RDN 20 PROMPT 21 ROFF 22 ASTO 06 22♦LBL 09 23 XEQ "RND M" 24 STO 02 25 XEQ "RND M" 26 STO 05 27 RCL 02 28 XEQ IND 06 29♦LBL "TRY " 30 ARCL 05 31 ARCL 06 32 ARCL 02 33 "F=?" 34 PROMPT 35 RCL 03 36 X=Y? 37 GTO "YES " 38 "NO " 39 AVIEW 40 TONE 2 41 TONE 2 </pre>	<p>Initialisierung.</p> <p>Eingabe der größten Zahl.</p> <p>Marke für Einsprung.</p> <p>Frage nach Lektion.</p> <p>Anfang der Schleife.</p> <p>Operanden erzeugen.</p> <p>Problem erzeugen.</p> <p>Problem stellen.</p> <p>Wenn richtig, dann „YES“.</p>	<pre> 42 FS?C 00 43 GTO 00 44 SF 00 45 1 46 ST+ 09 47 GTO "TRY " 48♦LBL 00 49 ARCL 05 50 ARCL 06 51 ARCL 02 52 "F=" 53 ARCL 03 54 AVIEW 55 GTO 00 56♦LBL "YES " 57 CF 00 58 "YES" 59 AVIEW 60 1 61 ST+ 08 62♦LBL 00 63 DSE 07 64 GTO 09 65 RCL 09 66 X=0? 67 XEQ "FF" 68 RCL 08 69 .1 70 / 71 CLA 72 ARCL X 73 "F% RIGH T" 74 AVIEW 75 PSE 76 PSE 77 GTO "AGN " 78♦LBL "+" 79 + 80 STO 03 81 LASTX 82 - 83 LASTX 84 CLA 85 RTN 86♦LBL "-" 87 - </pre>	<p>Wenn 2. Versuch, dann nächste Aufgabe</p> <p>Anzahl der falschen Antwort erhöhen und Aufgabe wiederholen.</p> <p>Richtige Antwort anzeigen.</p> <p>„YES“ anzeigen.</p> <p>Anzahl der richtigen Antworten erhöhen.</p> <p>Wenn nicht fertig, Schleife wiederholen.</p> <p>Wenn alle Antworten richtig waren, Tusch spielen.</p> <p>Anzahl der richtigen Antworten in % anzeigen.</p> <p>Nächste Aufgabenserie.</p> <p>Eine + Lektion</p> <p>Eine - Lektion.</p>
---	--	---	---

R00 = Zufallszahl

R01 = nicht verwendet

R02 = n2

R03 = Antwort

R04 = 1 + Höchstzahl

R05 = n1

R06 = Art der Lektion

R07 = Zähler

R08 = Anzahl der richtigen Antworten

R09 = Anzahl der falschen Antworten

88 X<=0? 89 XEQ 00 90 STO 03 91 LASTX 92 + 93 LASTX 94 CLA 95 RTN 96♦LBL 00 97 CHS 98 RCL 02 99 X<> 05 100 X<> 02 101 RDN 102 RTN 103♦LBL "*" 104 * 105 STO 03 106 LASTX 107 / 108 LASTX 109 CLA 110 RTN 111♦LBL "/" 112 X<>Y 113 STO 03 114 * 115 STO 05 116 CLA 117 RTN 118♦LBL "RND" M" 119 RCL 00 120 9821 121 * 122 .211327 123 + 124 FRC 125 STO 00 126 SQRT 127 RCL 04 128 * 129 INT 130 RTN 131♦LBL "FF" 132 TONE 8 133 TONE 9 134 XEQ "0" 135 XEQ "0" 136 TONE 8	Eine * Lektion. Eine / Lektion Zufallsgenerator Zahl transformieren. Spielen einer Melodie.	137 TONE 8 138 TONE 8 139 TONE 7 140 TONE 8 141 TONE 8 142 TONE 7 143 TONE 8 144 TONE 9 145 XEQ "0" 146 XEQ "0" 147 TONE 9 148 TONE 8 149 XEQ "0" 150 TONE 8 151 TONE 7 152 XEQ "0" 153 TONE 7 154 TONE 6 155 RTN 156♦LBL "0" 157 X<>Y 158 X<>Y 159 X<>Y 160 X<>Y 161 X<>Y 162 X<>Y 163 RTN	Unterprogramm als Zeitschleife. Wichtige Status-informationen: Speichergröße = 010 FIX 0 CF 29 Flags: FOO gesetzt, wenn Antwort falsch F29 gelöscht zur Unterdrückung des Dezimalpunktes
--	---	---	--

Notizen

Hexadezimal-Dezimal Umwandlung

Dieses Programm wandelt Zahlen zwischen Hexadezimal- und Dezimalsystem um. Es können ganzzahlige Dezimalzahlen bis 1048575 und ganzzahlige Hexadezimalzahlen bis zu FFFF umgewandelt werden.

					Speichergröße: 021
Nr.	Anweisung	Wert	Funktion	Anzeige	
1	Stellen Sie den Status ein und schalten Sie in den USER-Modus.				
2	Initialisieren Sie den Rechner.		[A]	READY	
3	Um eine Dezimalzahl in eine Hexadezimalzahl umzuwandeln, tasten Sie die Zahl ein.	D	[E]	H	
4	Um eine Hexadezimalzahl in eine Dezimalzahl umzuwandeln, tasten Sie die Zahl im ALPHA-Modus ein.	H	[E]	D	
5	Um die Zahl wieder in Ihre ursprüngliche Form umzuwandeln, drücken Sie einfach nochmal E. ANMERKUNG: D stellt eine ganzzahlige Dezimalzahl kleiner als 1048576_{10} dar H stellt eine ganzzahlige Hexadezimalzahl kleiner als 1000000_{16} dar		[E]	H oder D	

Beispiel 1:

Wandeln Sie 123_{10} in eine Hexadezimalzahl um.

Tastenfolge	Anzeige	Kommentar:
[A] 123 [E]	READY 7 B	Programminitialisierung

Beispiel 2:

Wandeln Sie 123_{16} in eine Dezimalzahl um.

Tastenfolge:	Anzeige:
123 [E]	291.

Kernpunkt im Programm

Dieses Programm verwendet die beiden bei Ziffern- und Zeicheneingabe aktiven Flags 22 und 23, um zu entscheiden, ob die eingegebene Zahl die Basis 10 (dezimal) oder 16 (hexadezimal) hat. In der ersten Programmzeile wird F22 überprüft, um festzustellen, ob Ziffern eingegeben wurden. Wenn dies der Fall war, wird Flag 23 gelöscht, so daß das Programm mit Schritt 6 fortsetzen kann. Wenn Flag 22 nicht gesetzt ist, wird Flag 23 geprüft, wonach das Programm nach LBL 04 verzweigt, wenn Zeichen eingegeben wurden. Die Flags werden anschließend so gesetzt, daß eine automatische Rückumwandlung stattfinden kann.

01+LBL E		50 X=Y?	Wenn Zeichen Null, Schleife 5 wiederholen.
02 FS?C 22		51 GTO 05	
03 CF 23		52+LBL 06	
04 FS? 23	Wenn Alpha-Eingabe, GTO Marke 04.	53 RCL IND	
05 GTO 04		18	Codierte Hexa-Zahl aufbauen.
06 STO 19	Dezimal in codierte Hexa-Zahl umwandeln	54 X=Y?	
07 XEQ 08		55 GTO 07	
08 +	Schleife 1	56 RDN	
09+LBL 01		57 ISG 18	
10 LASTX	Inkrementiere Zähler	58+LBL 00	
11 ISG 16	Pseudo-Marke wird	59 GTO 06	
12+LBL 00	übersprungen.	60+LBL 07	
13 1 E2		61 RCL 18	
14 /		62 RCL 17	
15 INT		63 INT	
16 X#0?	Schleife 1 wiederholen, so lange Ziffern vor- handen sind.	64 10↑X	Bis zu 5 Hexa-Zeichen zählen.
17 GTO 01		65 *	
18 CLA		66 ST+ 19	Routine zur Speicherung von Kon- stanten in den richtigen Registern und Vorberei- tung zur Umwandlung.
19 LASTX	Anfang Schleife 3	67 ISG 17	
20+LBL 03		68 GTO 05	
21 1 E2	Hexa-Zahl aufbauen.	69+LBL 08	
22 *		70 16	
23 ARCL IND		71 STO 18	
X		72 1	
24 FRC		73 STO 17	
25 DSE 16	Schleife 3 wiederholen bis R16 Null ist.	74 0	
26 GTO 03		75 STO 16	
27 SF 23		76 1 E2	
28 ASTO X	Hexa-Zahl anzeigen.	77 STO 20	
29 BEEP		78 FS? 23	
30 RTN		79 GTO 09	
31+LBL 04		80 RCL 18	
32 ASTO 16	Hexa-dezimal Um- wandlung vorbereiten.	81 X<> 20	Anfang Schleife 10. Umwandlung von einem Zahlensystem in das andere.
33 .00802		82 STO 18	
34 STO 17		83+LBL 09	
35 0		84 RCL 19	
36 STO 19		85+LBL 10	
37+LBL 05	Anfang Schleife 5	86 RCL 20	
38 0		87 /	
39 STO 18		88 STO 19	
40 " "		89 FRC	
41 ASTO Y	Hexa-Zahl zerlegen.	90 RCL 20	
42 ARCL 16		91 *	
43 ASTO 16		92 RCL 17	
44 ASHF		93 *	
45 ASTO X		94 ST+ 16	
46 X=Y?		95 RCL 18	
47 GTO 08		96 ST* 17	
48 CLA		97 RCL 19	
49 ASTO Y	Wenn Leerzeichen, Schleife verlassen.	98 INT	
		99 X#0?	Wenn nicht fertig,

R00 = "0"

R06 = "6"

R01 = "1"

R07 = "7"

R02 = "2"

R08 = "8"

R03 = "3"

R09 = "9"

R04 = "4"

R10 = "A"

R05 = "5"

R11 = "B"

<pre> 100 GTO 10 101 X>> 16 102 CLA 103 FS?C 23 104 BEEP 105 RTN 106+LBL A 107 CF 22 108 CF 23 109 "0" 110 ASTO 00 111 "1" 112 ASTO 01 113 "2" 114 ASTO 02 115 "3" 116 ASTO 03 117 "4" 118 ASTO 04 119 "5" 120 ASTO 05 121 "6" 122 ASTO 06 123 "7" 124 ASTO 07 125 "8" 126 ASTO 08 127 "9" 128 ASTO 09 129 "A" 130 ASTO 10 131 "B" 132 ASTO 11 133 "C" 134 ASTO 12 135 "D" 136 ASTO 13 137 "E" 138 ASTO 14 139 "F" 140 ASTO 15 141 "READY" 142 ASTO X </pre>	<p>Schleife 10 wiederholen</p> <p>Initialisierungsroutine</p>		
--	---	--	--

Wichtige Status-Information

Größe = 021

FIX 0

Flags:

F22 Zifferneingabe

F23 ALPHA-Eingabe

R12 = "C"

R13 = "D"

R14 = "E"

R15 = "F"

R16 = ALPHA

R17 = Schleifenzähler, Ziffernzähler

R18 = Basiskonstante, Schleifenzähler

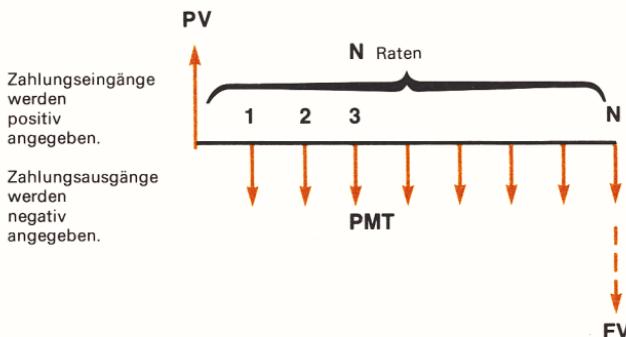
R19 = dezimalcodierte Zahl wird hier aufgebaut

R20 = Basiskonstante

Notizen

Finanztechnische Berechnungen

Mit diesem Programm wird Ihr HP-41 C in einen leistungsfähigen Finanz-Rechner umgewandelt. Das Programm berechnet die Variablen, die im Zusammenhang mit Cash-Flow-Problemen auftreten, die sich durch das untenstehende Diagramm darstellen lassen.



PV = Gegenwärtiger oder Barwert des Kapitals, Höhe eines Darlehens, einer Investition usw.

I = Periodenzinssatz

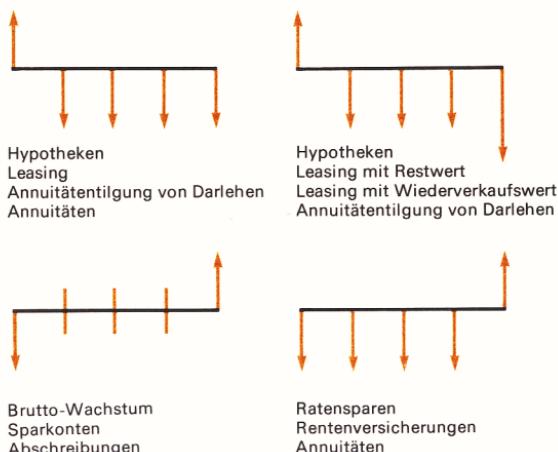
N = Anzahl der Zins- oder Zahlungsperioden

PMT = Ein- oder ausgezahlter Ratenbetrag, Annuität

FV = Endkapital, Endbetrag, Höhe der Ersparnisse usw.

In der Abbildung wird der Cash-Flow einer Annuitätentilgung eines Darlehns aus der Sicht des Kreditnehmers gezeigt. Vom Standpunkt des Geldgebers werden PV als negativ und die Annuitätenzahlungen als positiv angegeben. Durch einen Vorzeichenwechsel der Variablen PV, PMT, und FV können eine Reihe unterschiedlicher Probleme des Cash-Flows berücksichtigt werden.

Untenstehend sind die Cash-Flow-Diagramme für die vier grundlegenden, bei Zinseszinsrechnungen auftretenden Probleme zusammen mit der im Text verwendeten Terminologie angegeben.



Die fünf Tasten in der obersten Reihe (**A** bis **E**) werden zur Eingabe der finanztechnischen Parameter verwendet. Wenn Sie drei der Parameter eingegeben haben und eine der zwei übrigen Tasten drücken, wird der entsprechende Wert berechnet; wenn Sie vier der Parameter eingegeben haben, wird durch Drücken der letzten Taste der entsprechende Wert berechnet.

Um den Wert einer der fünf Variablen festzustellen, wird **RCL** und die entsprechende Taste gedrückt. Mit der Tastenfolge **RCL A** können alle in diesem Programm verwendeten Speicherregister gelöscht werden. Wenn die Register auf diese Weise gelöscht werden, erscheint in der Anzeige der Text N, I, PV, PMT, FV, um den Anwender auf die Funktion dieser Tasten aufmerksam zu machen.

Literatur:

Mehr über Cash-Flow-Analyse kann in Grant, E.L. und Ireson, W.G., Principles of Engineering Economy, Fourth Ed. The Ronald Press Company, New York 1964 nachgelesen werden.

Speichergröße: 010				
Nr.	Anweisung	Wert	Funktion	Anzeige
1	Stellen Sie den Status ein, geben Sie das Programm ein und schalten Sie den Rechner in den User-Modus.			
2	Löschen Sie die im Programm benötigten Register.		RCL A	N, I, PV, PMT, FV
3	Geben Sie die bekannten Größen ein: Anzahl der Perioden Periodenzinssatz in Prozent Barwert Ratenbetrag Endwert	N I PV* PMT* FV*	A B C D E	N I PV PMT FV
4	Berechnen Sie die gesuchte Größe: Anzahl der Perioden Periodenzinssatz Barwert Ratenbetrag Endwert		A B C D E	N = (N) I = (I)% PV = S(PV)* PMT = S(PMT)* FV = S(FV)*
5	Gehen Sie nach Schritt 4, um eine beliebige Variable erneut zu berechnen oder gehen Sie nach Schritt 3, um eine oder alle Variablen zu ändern.			

* Positiv bei Zahlungseingängen, negativ bei Auszahlungen.

34 Kaufmännische Berechnungen

Beispiel 1:

Ein Ehepaar erwirbt ein Haus, das \$ 50,000 kostet. Wie hoch sind die monatlichen Zahlungen eines 8,5%igen Darlehens von \$ 40,000 über einen Zeitraum von 30 Jahren weniger einem Monat?

Tastenfolge:

40000 [A] 40000 [C]
8.5 [ENTER↑] 12 [÷] [B]
30 [ENTER↑] 12 [×] 1 [-] [A] [D]

Anzeige:

40,000.00
0.71
PMT=\$-307.75

Beispiel 2:

Das Ehepaar von Beispiel 1 verkauft sein Haus nach 18 Monaten für \$ 25,000. Zu welchem Jahreszins hätte das Ehepaar seine ursprünglichen \$ 10,000 und die monatlichen Zahlungen von \$ 307.75 anlegen müssen, um nach 18 Monaten einen Endwert von \$ 25,000 zu erhalten?

Tastenfolge:

18 [A]
25000 [E]
10000 [CHS] [C] [B]
12 [×]

Anzeige:

25,000.00
I = 3.21 %
38.51

Monatszinssatz

Jahreszinssatz

Kernpunkt im Programm

In diesem Programm wird eine Technik der „austauschbaren Lösung“ verwendet. Jede der fünf Variablen in der Gleichung kann als Ausdruck der anderen vier Variablen angegeben werden. Die fünf Tasten in der obersten Reihe werden sowohl zur Abspeicherung der Eingabedaten als auch zur Berechnung der Ergebnisse mittels folgender Programmstruktur verwendet.

Marke f Eine der Marken A-J oder a-e

STO r Die Variable wird im Register r gespeichert

FS?C22 Das bei Dateneingabe aktive Flag wird geprüft u. gelöscht.

RTN Ende der Berechnung, wenn hier Daten eingetastet wurden.

} Berechnung des Wertes der Unbekannten.

STO r Speicherung des berechneten Wertes.

} Anzeige des neuen Wertes.

RTN

Dieser Baustein kann beliebig oft, je nach Anzahl der Variablen, wiederholt werden.

01♦LBL A 02 STO 01 03 FS?C 22 04 RTN 05 RCL 04 06 RCL 09 07 / 08 STO 00 09 RCL 05 10 - 11 RCL 03 12 RCL 00 13 + 14 / 15 LN 16 RCL 09 17 LN1+X 18 / 19 STO 01 20 "N=" 21 ARCL X 22 AVIEW 23 RTN 24♦LBL B 25 STO 02 26 1 E2 27 / 28 STO 09 29 1 30 + 31 STO 07 32 RCL 02 33 FS?C 22 34 RTN 35 RCL 04 36 X#0? 37 GTO 01 38 RCL 05 39 RCL 03 40 / 41 CHS 42 RCL 01 43 1/X 44 Y↑X 45 1 46 - 47 STO 09 48 GTO 00 49♦LBL 01 50 RCL 05	N speichern Anhalten, wenn es sich um neue Daten handelt. Andernfalls neues N errechnen. Neues N anzeigen. I und einige Funktionen von I speichern. Anhalten, wenn es sich um neue Daten handelt. Andernfalls, wenn PMT = 0, neues I durch einfache Formel berechnen. Sonst neues I nach Newtons-Methode berechnen.	51 ABS 52 RCL 04 53 RCL 01 54 * 55 RCL 03 56 + 57 ABS 58 - 59 RCL 04 60 RCL 01 61 * 62 RCL 05 63 + 64 ABS 65 RCL 03 66 ABS 67 - 68 * 69 ENTER↑ 70 ABS 71 / 72 1 E-9 73 * 74 STO 09 75♦LBL 06 76 XEQ 08 77 RCL 04 78 * 79 RCL 03 80 + 81 RCL 05 82 RCL 08 83 * 84 + 85 RCL 08 86 RCL 07 87 / 88 RCL 01 89 * 90 STO 06 91 1 92 RCL 08 93 - 94 RCL 09 95 / 96 - 97 RCL 04 98 RCL 09 99 / 100 * 101 RCL 05	Anfang Schleife.
---	--	--	------------------

R00 = verwendet

R01 = n

R02 = i

R03 = PV

R04 = PMT

R05 = FV

R06 = verwendet

R07 = $1 + i/100$

R08 = verwendet

R09 = $i/100$

102 RCL 06 103 * 104 - 105 / 106 STO 09 107 ABS 108 1 E-7 109 X<=Y? 110 GTO 06 111 RCL 09 112♦LBL 00 113 1 E2 114 * 115 STO 02 116 "I=" 117 ARCL X 118 "%" 119 AVIEW 120 RTN 121♦LBL C 122 STO 03 123 FS?C 22 124 RTN 125 RCL 04 126 XEQ 08 127 * 128 RCL 05 129 RCL 08 130 * 131 + 132 CHS 133 STO 03 134 "PV=\$" 135 ARCL X 136 AVIEW 137 RTN 138♦LBL D 139 STO 04 140 FS?C 22 141 RTN 142 XEQ 08 143 1/X 144 RCL 03 145 RCL 05 146 RCL 08 147 * 148 + 149 * 150 CHS 151 STO 04 152 "PMT=\$" 153 ARCL X	Wenn I nicht klein ist, Schleife wiederholen. Neues I anzeigen. PV speichern. Anhalten wenn es sich um neue Daten handelt. Andernfalls neuen PV berechnen. Neuen PV anzeigen. PMT speichern. Anhalten, wenn es sich um neue Daten handelt. Andernfalls neuen PMT berechnen. Neuen PMT anzeigen.	154 AVIEW 155 RTN 156♦LBL E 157 STO 05 158 FS?C 22 159 RTN 160 XEQ 08 161 RCL 04 162 * 163 RCL 03 164 + 165 RCL 08 166 / 167 CHS 168 STO 05 169 "FV=\$" 170 ARCL X 171 AVIEW 172 RTN 173♦LBL 08 174 1 175 XEQ 09 176 RCL 01 177 CHS 178 Y↑X 179 STO 08 180 - 181 RCL 09 182 / 183 RTN 184♦LBL 09 185 RCL 09 186 1 187 + 188 STO 07 189 RTN 190♦LBL 9 191 CLX 192 STO 01 193 STO 02 194 STO 03 195 STO 04 196 STO 05 197 STO 09 198 "N, I, P V, PMT,F" 199 "FV" 200 AVIEW 201 RTN	FV speichern. Anhalten wenn es sich um neue Daten handelt. Andernfalls neuen FV berechnen. Unterprogramm zur Berechnung von $\left(1 + \frac{i}{100}\right)^{-n}$ $1 - \left(1 + \frac{i}{100}\right)^{-n}$ $\frac{i/100}{\left(1 + \frac{i}{100}\right)^{-n}}$ Unterprogramm zur Berechnung von $1 + i/100$
--	---	--	--

Wichtige Status-
Informationen:
Größe = 010
FIX 2

Flags:
F22 Zifferneingabe

Notizen

Nullstellenbestimmung

Dieses Programm ermittelt die Werte einer unabhängigen Variablen x, für die eine Funktion dieser Variablen gleich Null ist. Diese Werte werden als Nullstellen dieser Funktion bezeichnet. So wird beispielsweise für die Gleichung

$$f(x) = 2x - 6$$

x = 3 als Nullstelle bezeichnet, weil

$$f(3) = 2 \times 3 - 6 = 0$$

ist.

Es gibt eine ganze Reihe von Verfahren, mit der die Nullstellen einer Funktion errechnet werden können. Diese Algorithmen benötigen im allgemeinen einen Anfangswert für die Nullstelle als Ausgangspunkt für die Iteration. Das iterative Lösungsverfahren bestimmt dann laufend bessere Funktionswerte, bis eine ausreichend genaue Lösung erreicht ist. Einige Algorithmen konvergieren nicht gegen eine Lösung, so daß die Iteration nicht abbricht. Wieder andere gewährleisten zwar eine Lösung, konvergieren aber nur sehr langsam.

Der in diesem Programm verwendete Algorithmus ermittelt in jedem Fall die Nullstelle einer Funktion, wenn die Anfangswerte beiderseits einer ungeraden Anzahl von Nullstellen vorgegeben werden. Wenn diese Anfangswerte nicht richtig beiderseits einer Nullstelle liegen, müssen neue Werte gewählt werden. Der Anwender muß also einiges über seine Funktion wissen, bevor er dieses Programm anwendet, wenn er Gewähr haben will, daß die Iteration schnell ist und gegen eine Nullstelle konvergiert.

Vor der Ausführung des Programms müssen Sie die Funktion programmieren, deren Nullstelle Sie bestimmen wollen. Dies geschieht, indem Sie **GTO** drücken und dann die Funktion ejntasten. Mit der Tastenfolge **XEQ ROOT** starten Sie dann das Programm. Die erforderlichen Eingaben bestehen aus der von Ihnen verwendeten Funktionsbezeichnung und den beiden Anfangswerten. Wenn beide Anfangswerte Funktionswerte auf derselben Seite der x-Achse zur Folge haben, erscheint kurz die Meldung „F1*F2>0“, und die Eingabe für die Anfangswerte muß wiederholt werden.

Das Programm selber benötigt die Register 01 bis 07, so daß Register 00 und alle von Register 07 aufwärts verfügbaren Register bei der Programmierung der Funktion verwendet werden können. Das mit Text versehene Ergebnis wird angezeigt, wenn der Wert der Funktion 10^{-10} unterschritten hat. Beim Eintasten des Programms kann einfach ein anderer Wert eingegeben werden, um die Rechengenauigkeit zu erhöhen.

Literatur:

Der hier verwendete Illinois-Algorithmus ist in M. Dowell & P. Jarratt „A modified regula falsi method for computing the root of an equation“ BIT 11 (1971) S. 168–174 beschrieben.

Ein ähnlicher Algorithmus, der etwas schneller konvergiert, stammt von denselben Autoren M. Dowell & P. Jarrat „The Pegasas method for computing the root of an equation“ BIT 12 (1972), S. 503-508.

Nr.	Anweisung	Wert	Funktion	Anzeige
1	Stellen Sie den Status ein und geben Sie das Programm ein.			
2	Tasten Sie Ihre Funktion ein. Verwenden Sie dabei einen globalen Namen (d.h. nicht A-J, a-e oder 00-99).			
3	Starten Sie das Programm.		XEQ ROOT	FUNCTION NAME?
4	Geben Sie die Bezeichnung Ihrer Funktion ein.	Name	R/S	GUESS1=?
5	Geben Sie den ersten Anfangswert ein.	X1	R/S	GUESS2=?
6	Geben Sie den zweiten Anfangswert ein. Entweder zeigt der Rechner jetzt eine Lösung an oder das Programm kehrt nach Schritt 5 zurück.	X2	R/S	X=(ROOT) F1*F2>0

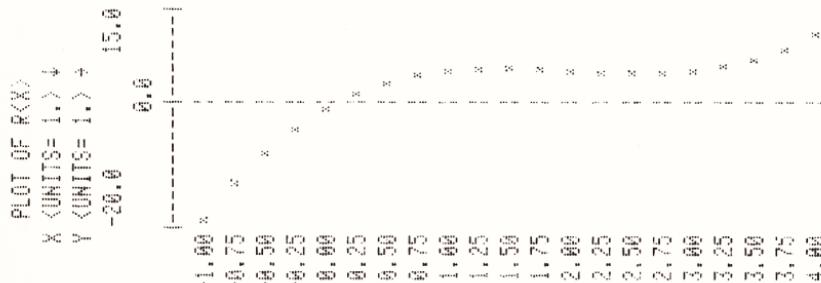
Beispiel 1:

Ermitteln Sie einen Wert x , so daß die Gleichung $R(x) = x^3 - 6x^2 + 11x - 1 = 0$ erfüllt ist. Beachten Sie, daß eine Skizze des Graphen auf eine Nullstelle zwischen 0 und 1 schließen läßt.

Tastenfolge**Anzeige**

ENTER+ ENTER+ ENTER+ 6 - x
 11 + x 1 - RTN

FUNCTION NAME?
GUESS1=?
GUESS2=?
X = 0.0958

**Beispiel 2:**

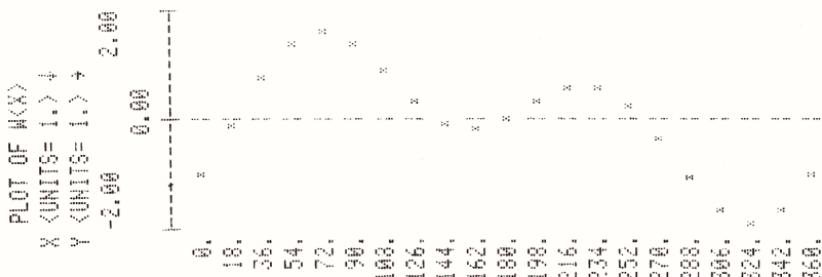
Bestimmen Sie die Nullstelle der Gleichung $W(x) = \sin(x - 30) - \cos(2x + 60)$, die zwischen 200 und 300 Grad liegt.

Tastenfolge

```

■ GTO • • PRGM
■ LBL ALPHA WAVE ALPHA
30 - SIN RCL 04
2 × 60 + COS - ■ RTN
PRGM
XEQ ALPHA ROOT ALPHA
WAVE R/S
200 R/S
300 R/S

```

Anzeige**FUNCTION NAME?****GUESS1=?****GUESS2=?****X = 260.0000****Kernpunkt im Programm**

Das Programm verlangt vom Anwender die Bezeichnung der programmierten Funktion. Diese Bezeichnung wird gespeichert und die Funktion jedesmal indirekt ausgeführt, wenn sie im Programm benötigt wird. Beachten Sie, daß die AON-Funktion vor dem PROMPT ausgeführt wird, so daß der HP-41C im ALPHA-Modus anhält. Die AOFF-Funktion muß allerdings vor dem nächsten PROMPT ausgeführt werden, da sonst der ALPHA-Modus weiter wirksam bleibt. Mit den AON- und AOFF-Funktionen kann der Modus, in dem der Rechner anhält, leicht gesteuert werden, um den Anwender noch einmal an die Art der erforderlichen Eingaben zu erinnern und ALPHA-Eingaben zu erleichtern.

Wenn die Funktionsbezeichnung im Register 3 gespeichert ist, kann das Programm die Funktion jedesmal mit XEQ IND 03 ausführen. So muß bei der Anwendung des Programms auf mehrere Funktionen nicht etwa die ganze Funktion geändert, sondern nur die entsprechende Funktionsbezeichnung eingegeben werden.

FUNCTION NAME?**AON****PROMPT**

Textausgabe, der Rechner hält im ALPHA-Modus an.

ASTO 03

Die Funktionsbezeichnung wird in Register 3 gespeichert.

AOFF

Schaltet den ALPHA-Modus aus.

.**.****.**

Führt das Programm, dessen Bezeichnung im R 3 steht, aus.

XEQ IND 03

<pre> 01♦LBL "R0F T" 02 "FUNCTION N NAME?" 03 RDN 04 PROMPT 05 ROFF 06 RSTO 03 07♦LBL A 08 "GUESS1= ?" 09 PROMPT 10 STO 01 11 "GUESS2= ?" 12 PROMPT 13 STO 02 14 RCL 01 15 STO 04 16 XEQ IND 03 17 STO 05 18 RCL 02 19 STO 04 20 XEQ IND 03 21 STO 06 22 RCL 05 23 * 24 X>0? 25 GTO 05 26♦LBL 00 27 RCL 02 28 RCL 02 29 RCL 01 30 - 31 RCL 06 32 RCL 05 33 - 34 / 35 RCL 06 36 + 37 - 38 STO 04 39 XEQ IND 03 40 STO 07 41 X=0? 42 GTO 04 43 ABS </pre>	<p>Der Rechner verlangt die Funktionsbezeichnung.</p> <p>Anfangswerte speichern.</p> <p>Anfang der Schleife.</p> <p>Neues x.</p> <p>Wenn $f(x) = 0$, dann fertig.</p>	<pre> 44 1 E-10 45 X>Y? 46 GTO 04 47 RCL 07 48 RCL 06 49 + 50 X>0? 51 GTO 01 52 RCL 02 53 STO 01 54 RCL 06 55 STO 05 56♦LBL 02 57 RCL 04 58 STO 02 59 RCL 07 60 STO 06 61 GTO 00 62♦LBL 01 63 2 64 ST/ 05 65 GTO 02 66♦LBL 04 67 "X=" 68 ARCL 04 69 PROMPT 70♦LBL 05 71 "F1*F2>0 " 72 AVIEW 73 PSE 74 GTO A </pre>	<p>Grenzwert Wenn $f(x) < 1 \text{ E}-10$, dann fertig.</p> <p>Neue Anfangswerte gemäß den Bedingungen des Illinois-Algorithmus auswählen.</p> <p>Fertig.</p> <p>Antwort anzeigen.</p> <p>Fehlermeldung.</p> <p>Rücksprung zur Eingabe.</p> <p>Wichtige Status-Information: Speichergröße = 008 DEG FIX 4</p>
--	--	---	---

R00 = nicht verwendet

R01 = X1
 R02 = X2
 R03 = Name
 R04 = X
 R05 = f(X1)
 R06 = f(X2)
 R07 = f(X3)

Kurvenanpassung

Dieses Programm ermöglicht die Anpassung folgender Kurven an vorgegebene Wertepaare (x_i, y_i) , $i = 1, 2 \dots, n$:

1. Gerade (lineare Regression): $y = a + bx$.
2. Exponentialfunktion: $y = ae^{bx}$ ($a > 0$),
3. Logarithmusfunktion: $y = a + b \ln x$,
4. Potenzfunktion: $y = ax^b$ ($a > 0$).

Die Regressionskoeffizienten a und b werden durch die Lösung des folgenden äquivalenten linearen Gleichungssystems ermittelt:

$$A_n + B \sum X_i = \sum Y_i$$

$$A \sum X_i + B \sum X_i^2 = \sum Y_i X_i$$

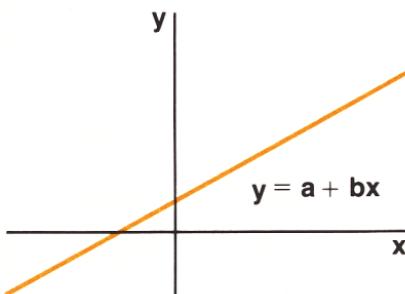
Die Beziehung der Variablen ist folgendermaßen definiert:

Regression	A	B	X_i	Y_i
Gerade	a	b	x_i	y_i
Exponentialfunktion	$\ln a$	b	x_i	$\ln y_i$
Logarithmusfunktion	a	b	$\ln x_i$	y_i
Potenzfunktion	$\ln a$	b	$\ln x_i$	$\ln y_i$

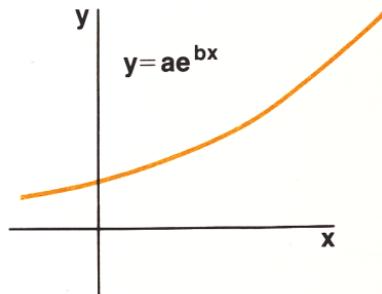
Das Bestimmtheitsmaß ist folgendermaßen definiert:

$$R^2 = \frac{A \sum Y_i + b \sum X_i Y_i - \frac{1}{n} (\sum Y_i)^2}{\sum (Y_i^2) - \frac{1}{n} (\sum Y_i)^2}$$

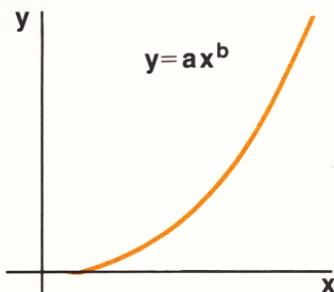
Lineare Regression



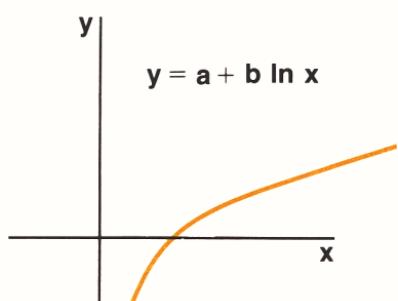
Exponential-Kurvenanpassung



Anpassung einer Potenzfunktion



Logarithmische Kurvenanpassung



Anmerkungen:

1. Im Programm wird die Methode der kleinsten Quadrate entweder auf die ursprünglichen Gleichungen (Gerade und Logarithmusfunktion) oder auf die transformierten Gleichungen (Exponential- und Potenzfunktion) angewandt.
2. Für negative Werte von x_i oder für $x_i = 0$ erfolgt im Fall der logarithmischen Kurvenanpassung eine Fehlermeldung. Das gleiche gilt für y_i bei der Exponential-Kurvenanpassung. Bei der Verwendung einer Potenzfunktion müssen sowohl alle x_i als auch y_i positiv und von Null verschieden sein.
3. Wenn die Differenz zwischen x- und/oder y-Werten klein wird, nimmt die Genauigkeit der Regressionskoeffizienten ab.

Speichergröße: 016

Nr.	Anweisung	Wert	Funktion	Anzeige
1	Stellen Sie den Status ein und geben Sie das Programm ein.			
2	Initialisieren Sie das Programm für eine LINEARE REGRESSION oder EXPONENTIAL-KURVEN-ANPASSUNG oder LOGARITHMISCHE KURVENANPASSUNG oder ANPASSUNG EINER POTENZFUNKTION.		[XEQ] LIN [XEQ] EXP [XEQ] LOG [XEQ] POW	LIN EXP LOG POW
3	Wiederholen Sie Schritte 3 und 4 für $i = 1, 2, \dots n$ Eingabe: x_i Eingabe: y_i	x_i y_i	[ENTER] [A]	(i)
4	Wenn Ihnen bei der Eingabe ein Fehler unterlaufen ist, korrigieren Sie ihn durch	x_k y_k	[ENTER] [C]	(k-1)
5	Berechnen Sie R^2 und die Regressionskoeffizienten a und b		[E] [R/S] [R/S]	$R^2 = (R^2)$ $a = (a)$ $b = (b)$

Speichergröße: 016

Nr.	Anweisung	Wert	Funktion	Anzeige
6	Berechnen Sie einen Schätzwert \hat{y} zu einem gegebenen x -Wert	x	R/S	$\hat{Y} = (\hat{y})$
7	Wiederholen Sie Schritt 6 für verschiedene x -Werte			
8	Wiederholen Sie Schritt 4 für eine erneute Ausgabe der Ergebnisse			
9	Wenn Sie das Programm auf eine andere Datenmenge anwenden wollen, initialisieren Sie es mit → und gehen Sie dann nach Schritt 3		A	LIN oder EXP oder LOG oder POW
10	Wenn Sie ein anderes Programm ausführen wollen, gehen Sie nach Schritt 2.			

Beispiel 1:

Führen Sie eine lineare Regression für folgende Koordinatenpaare durch und berechnen Sie \hat{y} für $x = 37$ und $x = 35$.

x ₁	40.5	38.6	37.9	36.2	35.1	34.6
y ₁	104.5	102	100	97.5	95.5	94

Tastenfolge:

XEQ ALPHA LIN ALPHA
 40.5 ENTER↑ 104.5 A
 38.6 ENTER↑ 102 A
 37.5 ENTER↑ 100 A
 36.2 ENTER↑ 97.5 A
 35.2 ENTER↑ 95.5 A
 35.2 ENTER↑ 95.5 C
 35.1 ENTER↑ 95.5 A
 34.6 ENTER↑ 94 A
 E
 R/S
 R/S
 37 R/S
 35 R/S

Anzeige:

LIN
 1.00
 2.00
 3.00
 4.00
 5.00
 4.00
 5.00
 6.00
 R2 = 0.99
 a = 33.53
 b = 1.76
 Y. = 98.65
 Y. = 95.13

Hoppla!
 Fehler berichtigen.
 Verwenden Sie die
 richtigen Werte.

Beispiel 2:

Legen Sie eine Exponentialfunktion durch folgende Koordinatenpaare und berechnen Sie \hat{y} für $x = 1.5$ und $x = 2$.

x_1	.72	1.31	1.95	2.58	3.14
y_1	2.16	1.61	1.61	.85	0.5

Tastenfolge:

XEQ ALPHA EXP ALPHA
 .72 ENTER \downarrow 2.16 A
 1.31 ENTER \downarrow 1.61 A
 1.95 ENTER \downarrow 1.16 A
 2.58 ENTER \downarrow .85 A
 3.15 ENTER \downarrow .05 A
 3.15 ENTER \downarrow .05 C
 3.14 ENTER \downarrow 0.5 A
 E
 R/S
 R/S
 1.5 R/S
 2.0 R/S

Anzeige:

EXP
 1.00
 2.00
 3.00
 4.00
 5.00
 4.00
 5.00
 R2 = 0.98
 a = 3.45
 b = -0.58
 Y. = 1.44
 Y. = 1.08

Wenn Sie keinen Fehler machen, können Sie zwei Schritte überspringen.

Beispiel 3:

Legen Sie eine Logarithmusfunktion durch folgende Koordinatenpaare und berechnen Sie \hat{y} für $x = 8$ und $x = 14,5$.

x_1	3	4	6	10	12
y_1	1.5	9.3	23.4	45.8	60.1

Tastenfolge:

XEQ ALPHA LOG ALPHA
 3 ENTER \downarrow 1.5 A
 4 ENTER \downarrow 9.3 A
 6 ENTER \downarrow 23.4 A
 10 ENTER \downarrow 45.8 A
 12 ENTER \downarrow 6.01 A
 12 ENTER \downarrow 6.01 C
 12 ENTER \downarrow 60.1 A
 E
 R/S
 R/S
 8 R/S
 14.5 R/S

Anzeige:

LOG
 1.00
 2.00
 3.00
 4.00
 5.00
 4.00
 5.00
 R2 = 0.98
 a = -47.02
 b = 41.39
 Y. = 39.06
 Y. = 63.67

Wieder ein Fehler!

Beispiel 4:

Legen Sie eine Potenzfunktion durch folgende Koordinatenpaare und berechnen Sie \hat{y} für $x = 18$ und $x = 23$.

x_1	10	12	15	17	20	22	25	27	30	32	35
y_1	0.95	1.05	1.25	1.41	1.73	2.00	2.53	2.98	3.85	4.59	6.02

Tastenfolge:

```

XEQ [ALPHA]      POW [ALPHA]
10 [ENTER↑] 0.95 [A]
12 [ENTER↑] 1.05 [A]
15 [ENTER↑] 1.25 [A]
17 [ENTER↑] 1.41 [A]
20 [ENTER↑] 1.73 [A]
22 [ENTER↑] 2.00 [A]
25 [ENTER↑] 2.53 [A]
27 [ENTER↑] 2.98 [A]
30 [ENTER↑] 3.85 [A]
32 [ENTER↑] 4.59 [A]
35 [ENTER↑] 60.2 [A]
35 [ENTER↑] 60.2 [C]
35 [ENTER↑] 6.02 [A]
[E]
[R/S]
[R/S]
18 [R/S]
23 [R/S]

```

Anzeige:

POW
1.00
2.06
3.00
4.00
5.00
6.00
7.00
8.00
9.00
10.00
11.00
10.00
11.00
R2 = 0.94
a = 0.03
b = 1.46
Y. = 1.76
Y. = 2.52

Fehlerkorrektur.

Kernpunkt im Programm

Dieses Programm verwendet für die meisten auszuführenden Berechnungen nur einen einzigen Code. Da die Dateneingabe für jede der vier Typen der Kurvenanpassung unterschiedlicher Art ist, sollte man meinen, daß für jeden Kurventyp ein besonderes Programm erforderlich wäre. Dies wird umgangen, indem die vorbereitenden Programme LIN, LOG, EXP und POW einen Code in Register 00 speichern. Wenn dann die Anweisung XEQ IND 00 in Zeile 32 ausgeführt wird, werden die Eingabedaten entsprechend der in Register 00 gespeicherten Funktionsbezeichnung auf vier unterschiedliche Weisen verarbeitet.

01♦LBL "LIN " 02 5 03 "LIN" 04 GTO 13 05♦LBL "EXP " 06 6 07 "EXP" 08 GTO 13 09♦LBL "LOG " 10 7 11 "LOG" 12 GTO 13 13♦LBL "POW " 14 8 15 "POW" 16♦LBL 13 17 XEQ "INI T" 18 STO 00 19 ASTO 08 20 ΣREG 10 21 CLΣ 22 BEEP 23 AVIEW 24 STOP 25♦LBL C 26 X<>Y 27 XEQ IND 00 28 Σ- 29 STOP 30♦LBL A 31 X<>Y 32 XEQ IND 00 33 Σ+ 34 STOP 35♦LBL 07 36 LN 37 RTN 38♦LBL 08 39 LN 40♦LBL 06 41 X<>Y 42 LN 43 X<>Y 44 RTN	Gerade. Exponential. Logarithmisch. Potenz. Piepton, Anzeige und Σ-Register definieren Korrektur. Dateneingabe. Logarithmisch. Potenz und Exponential.	45♦LBL E 46 RCL 15 47 RCL 11 48 RCL 10 49 RCL 10 50 XEQ 09 51 STO 03 52 RCL 12 53 RCL 11 54 RCL 10 55 RCL 14 56 XEQ 09 57 RCL 03 58 ∕ 59 STO 04 60 XEQ IND 00 61 STO 06 62 RCL 15 63 RCL 14 64 RCL 10 65 RCL 12 66 XEQ 09 67 RCL 03 68 ∕ 69 STO 05 70♦LBL 03 71 RCL 04 72 RCL 12 73 ∗ 74 RCL 05 75 RCL 14 76 ∗ 77 + 78 RCL 12 79 X↑2 80 RCL 15 81 ∕ 82 STO 09 83 - 84 RCL 13 85 RCL 09 86 - 87 ∕ 88 "R2" 89 XEQ 88 90 RCL 06 91 "a" 92 XEQ 88 93 RCL 05 94 "b"	A, b und a, b berechnen.
---	--	--	--------------------------

R00 = Index

R01 = x

R02 = y

R03 = det

R04 = A

R05 = b

R06 = a

R07 = verwendet

R08 = LIN oder EXP oder LOG oder POW

R09 = (Σy) 2/n

95 GTO 01 96♦LBL 06 97♦LBL 08 98 E↑X 99♦LBL 05 100♦LBL 07 101 RTN 102♦LBL 09 103 * 104 STO 07 105 RDN 106 * 107 RCL 07 108 - 109 RTN 110♦LBL 00 111 "Y." 112♦LBL 01 113 "F=" 114 ARCL X 115 AVIEW 116 FS? 55 117 STOP 118♦LBL 04 119 GTO IND 08 120♦LBL 08 121 RCL 05 122 Y↑X 123 GTO 09 124♦LBL 06 125 RCL 05 126 * 127 E↑X 128♦LBL 09 129 RCL 06 130 * 131 GTO 00 132♦LBL 07 133 LN 134♦LBL 05 135 RCL 05 136 * 137 RCL 06 138 + 139 GTO 00 140♦LBL 88 141 "F=" 142 ARCL X 143 AVIEW 144 RTN	Inverse Transformation Bestimmtheitsmaß r^2 berechnen. x eingeben um y zu berechnen.	145♦LBL a 146 GTO IND 08 147♦LBL "INI T" 148 CLRG 149 CF 00 150 CF 01 151 CF 02 152 SF 21 153 SF 27 154 CF 29 155 RTN Wichtige Status-Information: Speichergröße = 016 $\Sigma = 10$ F_{1-2} Flags: F44 F41 F42 F21 F27 F29	Neu initialisieren. Initialisierung.
--	---	--	---

$$R10 = \Sigma x$$

$$R11 = \Sigma x^2$$

$$R12 = \Sigma y$$

$$R13 = \Sigma y^2$$

$$R14 = \Sigma xy$$

$$R15 = n$$

Notizen

Vektor-Operationen

Mit diesem Programm kann die Addition, Subtraktion, Multiplikation und Division von Vektoren durchgeführt werden. Die Vektoren müssen wie folgt im Stack angeordnet sein, bevor eine der Routinen ausgeführt werden kann

Ursprüngliche Stackanordnung

T v_1 Z u_1 Y v_2 X u_2

Die sich ergebende Anzeige

U = u, V = v

wobei die zwei Vektoren folgendermaßen dargestellt werden:

$$u_1 + iv_1 \text{ und } u_2 + iv_2$$

Beachten Sie, daß auch die folgende Schreibweise zuweilen verwendet wird:

$$u + vi, u + jv \text{ oder } ui + vj$$

Speichergröße: 000

Nr.	Anweisung	Wert	Funktion	Anzeige
1	Geben Sie das Programm ein und ordnen Sie die Routinen folgenden (oder vier anderen) Tasten zu CADD  CSUB  CMULT  CDIV 		[ASN] CADD  [ASN] CSUB  [ASN] CMULT  [ASN] CDIV 	
2a	Speichern Sie die Eingabewerte in die Rechenregister-Stapel Imaginärteil des ersten Vektors Realteil des ersten Vektors	v ₁ u ₁	[ENTER↑] [ENTER↑]	
2b	Imaginärteil des zweiten Vektors Realteil des zweiten Vektors	v ₂ u ₂	[ENTER↑]	
3	Wählen Sie die gewünschte Funktion Vektoraddition Vektorschubtraktion Vektormultiplikation Vektordivision		CADD CSUB CMULT CDIV	U = (u), V = (v) U = (u), V = (v) U = (u), V = (v) U = (u), V = (v)
4	Wenn das Ergebnis als Teil einer weiteren Berechnung verwendet werden soll, ist es nicht nötig, das Ergebnis wieder einzugeben. Fahren Sie einfach mit der Vektoreingabe in Schritt 2b fort.			

Beispiel 1

Summieren Sie $1 + i3$ und $4 + i6$.

Tastenfolge:

■ FIX 2

6 [ENTER] 4 [ENTER] 3 [ENTER] 1

[XEQ] [ALPHA] CADD [ALPHA]

Anzeige:

U = 5.00, V = 9.00

Wählen Sie ein
passendes
Anzeigeformat.

Beispiel 2:

Berechnen Sie $s^2 + 1$ für $s = 3 + j2$

Tastenfolge:

2 [ENTER] 3 [ENTER]

2 [ENTER] 3 [XEQ]

[ALPHA] CMULT [ALPHA]

0 [ENTER] 1 [XEQ]

[ALPHA] CADD [ALPHA]

Anzeige:

U = 5.00, V = 12.00

U = 6.00, V = 12.00

Addieren Sie $1 + j0$.

Kernpunkt im Programm

In vielen Problemen wird zur Ausführung einer gewünschten Funktion nur eine Dateneingabe benötigt. Da Vektoren aber aus zwei Teilen bestehen, muß der Anwender zwei Dateneingaben vornehmen, bevor er eine Berechnung durchführen kann. Durch eine wohlüberlegte Verwendung des Stacks beim Eingeben von Daten können viele Programme wesentlich verkürzt werden.

Hier wird gezeigt, wie knapp ein Programm sein kann, wenn der Anwender mit den Eingabedaten vorsichtig umgehen muß.

Beachten sie, daß die vier Routinen als Unterprogramme in Programmen verwendet werden können, die Vektor-Operationen beinhalten, wenn Sie den Ausgabeteil gegen LBL "UV" RTN ersetzen. Die Ausgabewerte u und v stehen anschließend in den X- und Y-Registern.

Es dürfte sich auch als praktisch erweisen, die vier Routinen den Tasten **+**, **-**, **×** und **÷** zuzuordnen, so daß die Funktionen im USER-Modus direkt ausgeführt werden können.

52 Vektor-Operationen

01♦LBL "CSU B" 02 CHS 03 X<>Y 04 CHS 05♦LBL "CSD D" 06 X<>Y 07 RDN 08 + 09 RDN 10 + 11 RT 12 GTO "UV" 13♦LBL "CDT V" 14 R-P 15 1/X 16 X<>Y 17 CHS 18 GTO 00 19♦LBL "CMU LT" 20 R-P 21 X<>Y 22♦LBL 00 23 RDN 24 RDN 25 R-P 26 RT 27 * 28 RDN 29 + 30 RT 31 P-R 32♦LBL "UV" 33 "U=" 34 ARCL X 35 "F, V=" 36 ARCL Y 37 AVIEW 38 RTN Wichtige Status- Information: Speichergröße = 000	Subtrahieren. Vorzeichen des zweiten Vektors ändern und addieren. Addieren. Dividieren. Reziprokwert des zweiten Vektors bilden und multiplizieren. Multiplizieren. Anzeigeroutine.		

Notizen

Blackjack

Mit diesem Programm kann eine vereinfachte Version von „Blackjack“ (Vingt-et-un „17 und 4“) gespielt werden. Der Rechner, als Bank, hat 104 Karten, die er (ohne Erneuerung) austeilt, bis er nur noch 13 Karten hat. Dann wird das Kartenspiel neu gemischt. Der Spieler darf den Einsatz in beliebiger Höhe wählen. Wenn er keinen Einsatz angibt, wird der vorherige Wert in den Berechnungen verwendet.

Der Spieler und die Bank erhalten beide zwei Karten, wobei eine Karte der Bank aufgedeckt wird. Der Spieler kann dann entweder weitere Karten ziehen oder passen. Das Ziel des Spiels ist es, einen Gesamtwert von 21 aber nicht darüber zu erreichen. Die Bildkarten zählen jeweils 10, Asse 1 oder 11 Punkte und die restlichen Karten haben ihren Nennwert. Wenn die ersten zwei Karten des Spielers 21 ergeben, hat er „Blackjack“ und er erhält sofort das anderthalbfache seines Einsatzes, wenn die Bank nicht auch Blackjack hat.

Wenn der Spieler eine Karte zieht, die seine Punktzahl über 21 erhöht, ist er bankrott und verliert seinen Einsatz. Wenn der Spieler bei einer Punktzahl von 21 oder weniger paßt, muß die Bank so lange Karten ziehen, bis ihre Punktzahl wenigstens größer als 16 ist. Das Blatt mit der größeren Punktzahl hat dann gewonnen und die Kasse des Spielers wird auf den neuesten Stand gebracht. Wenn Bank und Spieler dieselbe Trefferzahl haben, gilt das Spiel als unentschieden.

Für dieses Programm muß der HP-41C mit einem zusätzlichen Speichermodul ausgestattet sein.

Speichergröße: 027

Nr.	Anweisung	Wert	Funktion	Anzeige
1	Geben Sie das Programm ein, überprüfen Sie den Status und ordnen Sie DL, HT und S drei beliebigen Tasten zu. Ein Startwert (0 ≤ Startwert <1) kann in R ₀₀ eingegeben werden.			
2	Geben Sie den Stand ihrer Kasse ein.	Kasse	[STO] 21	
3	Um das Kartenspiel zu mischen		[XEQ] SH	SHUFFLING
4	Geben Sie Ihren Einsatz ein	Einsatz	DL	I SHOW c* YOU HAVE 1 YOU HAVE 1 21
5a	Ziehen Sie eine Karte und wiederholen Sie diesen Schritt oder gehen Sie nach 5b		HT	YOU HAVE Karten
5b	Passen Sie. Die Bank zeigt jetzt ihr Blatt und zieht dann weitere Karten oder paßt.		S	I HAVE Karten . .
6	Wiederholen Sie nach Wunsch Schritt 4 und folgende Schritte			YOUR BANK IS S Kasse
	• Anmerkung: wenn Sie in Schritt 4 Blackjack haben, erscheint BLACKJACK in der Anzeige und [S(TAND)] (Passe) wird automatisch ausgeführt.			
	c ist eine beliebige Karte, Karten bezieht sich auf eine Folge von Karten – die Bezeichnungen der Karten sind aneinander gekettet so daß eine 10 und eine 7 als 107 angezeigt werden.			

Beispiel:

Mischen Sie das Kartenspiel, geben Sie π als Startwert ein und spielen Sie Blackjack mit einem Einsatz von \$2.

Tastenfolge:

[ASN] [ALPHA] DL [ALPHA] [$\Sigma+$]

[ASN] [ALPHA] HT [ALPHA] [\sqrt{x}]

[ASN] [ALPHA] S [ALPHA] [\sqrt{x}]

[USER]

[XEQ] [ALPHA] SH [ALPHA]

0 [STO] 21

[T] [STO] 00

2 DL

Anzeige:

ASN DL 11

ASN HT 12

ASN S 13

SHUFFLING

104

Es wird nur FRC (π) verwendet.

Anmerkung: Die DL-Funktion wurde der Taste [$\Sigma+$] zugeordnet. Bedenken Sie, daß sich der Rechner im USER-Modus befinden muß, damit Sie nicht $\Sigma+$ erhalten.

S

DL

HT

HT

S

I SHOW 2
YOU HAVE 107

I HAVE 2J
I HAVE 2JK

BUST
YOUR BANK IS \$2
I SHOW 6
YOU HAVE A5
YOU HAVE A57
YOU HAVE A575
I HAVE 6K
I HAVE 6K8
BUST
YOUR BANK IS \$4

Anmerkung:
Die S-Funktion wurde
der Taste zuge-
ordnet.

Kernpunkt im Programm

Sie können die 11 Register, die nach dem Eingeben des Programms noch zur Verfügung stehen, zur Programmierung einfacher Strategien für das Spielen von Blackjack verwenden. Die unten aufgeführte Routine überprüft die im Programm verwendeten Register und Flags um zu entscheiden, ob weitere Karten gezogen werden sollen oder Passen richtiger wäre. Wenn das Programm verliert, verdoppelt es jedesmal seinen Einsatz und gewinnt somit schließlich. Indem Sie weitere Speichermodule an ihren HP-41C anschließen, können Sie komplizierte Strategien entwickeln.

Beachten Sie, daß Sie für dieses Programm die Speicherregisterzahl auf 28 erhöhen müssen.

01♦LBL "PL"		18 XEQ "HT"	
02 2	neuer Einsatz	19 GTO 00	
03 SF 22		20♦LBL 01	
04♦LBL 02		21 FS? 09	
05 XEQ "IL"	Geben	22 XEQ "S"	Wenn kein Blackjack, dann passen
06♦LBL 00		23 RCL 27	
07 RCL 24	Punktezahl überprüfen	24 RCL 21	
08 12		25 STO 27	
09 ENTER↑	Korrektur für AS	26 -	Letzten Kassenstand retten
10 10	Wenn kein AS	27 X<0?	
11 FS? 07	Korrektur löschen.	28 GTO "PL"	Wenn Spiel gewonnen, neuen Einsatz ein- geben. Wenn unent- schieden, letzten Einsatz verwenden.
12 CLX		29 X=0?	
13 -		30 GTO 02	
14 X<=Y?		31 2	Wenn Spiel verloren Einsatz verdoppeln.
15 GTO 01	Wenn Punktezahl ≤ 12	32 ST* 22	
16 FC? 09	oder Blackjack, dann passen,	33 GTO 02	
17 GTO 01	andernfalls Karte ziehen.	34 END	

01♦LBL "CRD" " 02 CLA 03 RSTO 19 04 1 05 STO 15 06 RCL 00 07 9821 08 * 09 .211327 10 + 11 FRC 12 STO 00 13 RCL 14 14 * 15 INT 16 1 17 + 18♦LBL 02 19 RCL IND 15 20 X>Y? 21 GTO 03 22 - 23 ISG 15 24♦LBL 99 25 GTO 02 26♦LBL 03 27 DSE IND 15 28♦LBL 99 29 DSE 14 30 12 31 RCL 14 32 X>Y? 33 GTO 04 34 XEQ "SH" 35♦LBL 04 36 RCL 15 37 STO 16 38 10 39 X<=Y? 40 GTO 00 41 X<>Y 42 STO 16 43 1 44 X=Y? 45 GTO A 46 CLA	Routine für Geben einer Karte. Zufallszahlengenerator.	47 ARCL Y 48 GTO 01 49♦LBL 00 50 STO 16 51 CLX 52 10 53 X=Y? 54 GTO "10" 55 1 56 + 57 X=Y? 58 GTO J 59 1 60 + 61 X=Y? 62 GTO "Q" 63 "K" 64 GTO 01 65♦LBL A 66 "A" 67 CF 07 68 GTO 01 69♦LBL "Q" 70 "Q" 71 GTO 01 72♦LBL J 73 "J" 74 GTO 01 75♦LBL "10" 76 "10" 77♦LBL 01 78 RSTO 19 79 RCL 16 80 RTN 81♦LBL "SH" 82 "SHUFFLI NG" 83 AVIEW 84 1.013 85 ENTER↑ 86 8 87♦LBL 14 88 STO IND Y 89 ISG Y 90 GTO 14 91 104 92 STO 14 93 CLD	ALPHA-Bezeichnung der Karte speichern. Unterprogramm für Aufbau des Kartenspiels.
--	---	---	--

R00 = Zufallszahl

R01 = Asse

R02 = 2'er

R03 = 3'er

R04 = 4'er

R05 = 5'er

R06 = 6'er

R07 = 7'er

R08 = 8'er

R09 = 9'er

94 CF 00		137 FS? 07	
95 CF 01		138 CLX	
96 CF 02		139 +	
97 CF 03		140 21	
98 CF 04		141 X?Y?	
99 RTN		142 SF 09	
100♦LBL "DL"		143 FS? 09	Wenn kein Blackjack, Flag 9 setzen.
101 CF 09	Blackjack. Kein As.	144 RTN	
102 SF 07		145 21.5	
103 ABS		146 ST0 24	
104 INT		147 1.5	
105 FS?C 22		148 ST* 20	
106 ST0 22		149 "BLACKJA CK"	Sofort nach „STAND“ verzweigen.
107 RCL 22		150 AVIEW	
108 ST0 20	Alten Einsatz verwenden oder neuen Einsatz speichern.	151♦LBL "S"	
109 SF 06		152 CF 06	
110 CLA		153 FS? 07	
111 ASTO 26		154 GTO 05	
112 ASTO 25		155 11	
113 XEQ "CRD"	Erste Karte der Bank geben.	156 RCL 24	
"		157 X?Y?	
114 RCL 15		158 GTO 05	
115 ST0 17		159 10	
116 XEQ "CRD"		160 ST+ 24	
"		161♦LBL 05	
117 ST0 23		162 CF 07	
118 CF 08		163 FS? 08	
119 FS? 07		164 SF 07	
120 SF 08		165 RCL 17	
121 CLA		166 ST0 15	
122 ARCL 19		167 XEQ 04	
123 ARCL 25		168 XEQ "DH"	
124 ASTO 25		169 FS? 07	
125 "I SHOW		170 GTO 07	
"		171 11	
126 ARCL 25	Aufgedeckte Karte der Bank. Kein As.	172 RCL 23	
127 AVIEW		173 X?Y?	
128 SF 07		174 GTO 07	
129 0		175 21.5	
130 ST0 24		176 ST0 23	
131 XEQ "CRD"	Erste Karte des Spielers geben.	177 "I HAVE BLACKJAC"	
"		178 "HK"	
132 XEQ "PH"		179 AVIEW	
133 XEQ "CRD"	Zweite Karte des Spielers geben.	180 GTO 07	
"		181♦LBL 06	
134 XEQ "PH"	Blatt des Spielers anzeigen.		
135 RCL 24			
136 10			

R10 = 10'er
R11 = B
R12 = D
R13 = K
R14 = Anzahl der
restlichen Karten

R15 = Zähler
R16 = Wert der augenblicklichen Karte
R17 = Verdeckte Karte der Bank
R18 = nicht verwendet
R19 = Augenblickliche Karte im ALPHA-Format

182 XEQ "CRD"	Bank zieht eine Karte.	Bankrott überprüfen.
"		
183 XEQ "IH"	Zieht die Bank oder nicht? Wenn Spieler bankrott, dann abrechnen. Hat der Spieler Blackjack, dann Blackjack setzen. Hat die Bank mehr als 17 dann abrechnen.	Bank bankrott.
184♦LBL 07		
185 FS? 06		
186 GTO 09		
187 FC? 09		
188 GTO 08		
189 RCL 23		
190 17		
191 X<=Y?		
192 GTO 08		
193 FS? 07		
194 GTO 06		
195 11		
196 RCL 23		
197 X>Y?		
198 GTO 06		
199 7		
200 X>Y?		
201 GTO 06		
202 10		
203 ST+ 23		
204♦LBL 08		
205 21.5		
206 RCL 23		
207 X>Y?		
208 XEQ "DB"		
209 RCL 24		
210 -		
211 X=0?		
212 XEQ "P"		
213 X>0?		
214 SF 06		
215♦LBL 09		
216 RCL 20		
217 FS? 06		
218 CHS		
219 ST+ 21		
220 "YOUR BANK IS \$"		
221 ARCL 21		
222 AVIEW		
223 RTN		
224♦LBL "HT"		
225 XEQ "CRD"		
"		
226 XEQ "PH"		

R20 = Auszahlen

R21 = Kasse des Spielers

R22 = belegt

R23 = Punktzahl der Bank

R24 = Punktzahl des Spielers

R25 = Blatt der Bank

R26 = Blatt des Spielers

verwendete Flags:

F06 Spieler bankrott

F07 gesetzt = kein As / gelöscht = As

F08 gesetzt Bank hat kein As

gelöscht = Bank hat As

F09 gesetzt = kein Blackjack

gelöscht = Blackjack

F29 gelöscht um Dezimalpunkt zu unterdrücken

F21 sollte mit F55 (Drucker vorhanden)

übereinstimmen



Hewlett-Packard GmbH:

6000 Frankfurt 56, Bernerstrasse 117, Postfach 560140, Tel. (0611) 50 04-1
7030 Böblingen, Herrenbergerstrasse 110, Tel. (07031) 667-1
4000 Düsseldorf 11, Emanuel-Leutze-Strasse 1 (Seestern), Tel. (0211) 5 97 11
2000 Hamburg 60, Kapstadttring 5, Tel. (040) 6 38 04-1
8021 Taufkirchen, Eschenstraße 5, Tel. (089) 6117-1
3000 Hannover 91, Am Grossmarkt 6, Tel. (0511) 46 60 01
8500 Nürnberg, Neumeyerstrasse 90, Tel. (0911) 52 20 83/85
1000 Berlin 30, Keithstrasse 2-4, Tel. (030) 24 90 86

Hewlett-Packard (Schweiz) AG:

Zürcherstrasse 20, Postfach 307, CH-8952 Schlieren-Zürich, Tel. (01) 7 30 52 40

Hewlett-Packard Ges.m.b.H. für Österreich/für sozialistische Staaten:
Handelskai 52, Postfach 7, A-1205 Wien, Tel. (0222) 3516 21 bis 32

Hewlett-Packard S.A. Europa-Zentrale:
7, rue du Bois-du-Lan, Postfach, CH-1217 Meyrin 2-Genf, Schweiz, Tel. (022) 82 70 00

Scan Copyright ©
The Museum of HP Calculators
www.hpmuseum.org

Original content used with permission.

Thank you for supporting the Museum of HP
Calculators by purchasing this Scan!

Please do not make copies of this scan or
make it available on file sharing services.