

Programmsammlung für hp - 25

Spielprogramme

- 1 Golfspiel (ein oder mehrere Spieler)
- 2 Mensch Ärgere Dich Nicht für 3 Spieler
- 3 Mensch Ärgere Dich Nicht für 3 Spieler ähnl. 2
- 4 Mensch Ärgere Dich Nicht für 4 Spieler
- 5 Artillerie-Spiel für 2 Spieler
- 6 Sechs verschiedene Zufallszahlen (z.B. Lotto)
- 7 Schießen auf bewegtes Ziel (ein Spieler)
- 8 Rechenübungen mit Fehlerquote
- 9 Motorradrennen (1 Fahrer auf der Strecke)
- 10 Formeln zur Schräglage von Motorrädern
- 11 Superhirn (keine gleichen Farben)
- 12 Nimmspiel mit beliebigen Mengen
- 13a Codieren und Decodieren von Texten
- 13b Codieren und Dekodieren von Texten mit einf. Rechner
- 14a Auffinden von Pythagoräischen Zahlen (systematisch)
- 14b Auffinden von Pythagoräischen Zahlen (mit Formeln)
- 15 Mondlandung (verbessert)

Nützliche Programme

- 1 n über k berücksichtigt alle Definitionen
- 2 y^x bei positiver und negativer Basis
- 3 Radizieren von komplexen Zahlen
- 4 Zinsen im Sparbuch, exakter Kalender
- 5 Kombination für Orientierungsfahrt
- 6 Sechs oder acht Zahlen der Größe nach ordnen
- 7 Gebühr für Postüberweisung ins Ausland (Stand 1975)
- 8 Primfaktorzerlegung
- 9 Kürzen zweier Zahlen
- 10a Kalender, Wochentage, Differenz in Tagen
- 10b Welches Datum in x Tagen
- 11 Simonsche Regel für numerische Integration
- 12 Normiertes Zeichnen von Funktionen
- 13 wie 12, Grenzwerte werden selbst ermittelt
- 14 Primzahlen tabellarisch
- 15 Hyperbelfunktionen mittels Reihen
- 16 Biorhythmik
- 17 Sammelbesteller-Rechnung
- 18 Finanzprogramm-Paket
- 19 Zahlenumwandlung dual \leftrightarrow dezimal
- 20 Telefongebühren (Als Uhr und Abrechnung)
- 21 Überstundenzuschläge nach IG-Metall
- 22 Uhrzeit

Gerhard Streichert

Raffelisenstraße 1
8431 Postbauer-Heng

Golfspiel

```

00
01 23 04 STO 4 39 24 05 RCL 5
02 01 1 40 14 09 f → R
03 23 51 00 STO + 0 41 23 41 01 STO - 1
04 34 CLX 42 21 x ← y
05 74 R/S ^ 43 23 41 02 STO - 2
06 14 09 f → R 44 24 01 RCL 1
07 23 03 STO 3 45 14 74 f PAUSE
08 61 x 46 24 02 RCL 2
09 14 74 f PAUSE 47 13 00 GTO 00
10 14 09 f → R 48 24 00 RCL 0
11 23 41 01 STO - 1 49 14 11 09 f FIX 9
12 22 R↓
13 23 41 02 STO - 2
14 24 03 RCL 3
15 14 02 f → R
16 23 05 STO 5
17 14 74 f PAUSE

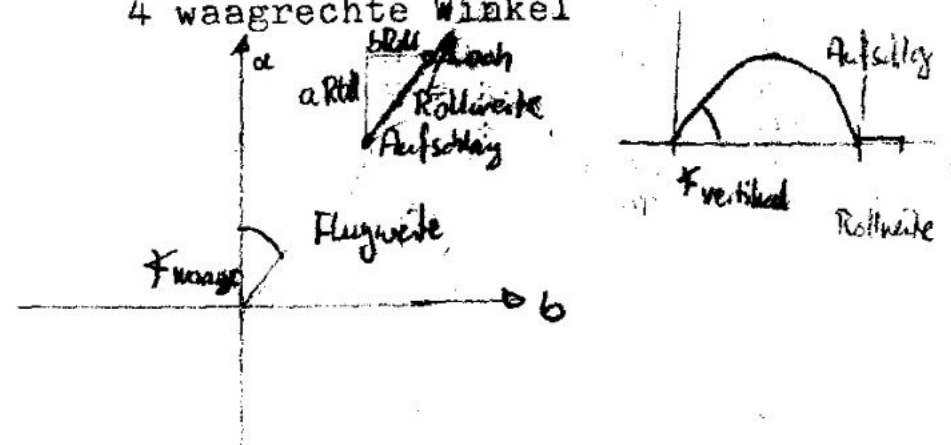
18 24 04 RCL 4
19 24 02 RCL 2
20 24 01 RCL 1
21 15 09 g → P
22 22 R+
23 05 5
24 71 +
25 14 01 f INT
26 05 5
27 61 x
28 14 74 f PAUSE
29 14 61 f x ← y
30 13 38 GTO 38
31 14 04 f sin
32 24 02 RCL 2
33 21 x ← y
34 71 +
35 24 05 RCL 5
36 24 51 f x ← y
37 13 48 GTO 48
38 24 04 RCL 4
    
```

Das Loch liegt a m vor und b m neben dem jeweiligen Standort. Schätze dem Winkel in Richtung zum Loch, Vorzeichen beachten. Wähle den Winkel in vertikaler Richtung und die Abschlußgeschw. Der Rechner berechnet zunächst die Wurfweite und die Rollweite am Boden. Bei vertikalem Winkel = 0 fliegt der Ball nicht, sondern rollt nur. Jetzt prüft der Rechner, ob die Flugrichtung genau zum Loch führt. Ist dies der Fall errechnet er den Aufschlagpunkt und die Entfernung von hier zum Loch. War die Rollweite größer, rollt der Ball ins Loch. Ansonsten berechnet der R. die neuen Koordinaten, wie auch, wenn der horizontale Winkel nicht richtig war. Der horizontale Winkel darf nur auf 5 Grad genau eingegeben werden. Als Trefferanzeige erscheint die Anzahl der Schläge mit neuen Nachkommennull.

Anweisung:

1. Programm eingeben f Prgm
2. f REG, a STO 1, b STO 2, f fix 0 oder 2
3. Horizontalwinkel auf 5 Grad abgerundet R/S
A: 0
4. Vertikalwinkel f Abschlußgeschw. R/S
5. A: (Wurfweite)(Rollweite)(Winkel von Aufschlag zum Loch) (a-Entfern) b-Entfern)
6. gehe mit neuen Daten zu 3.
7. bei neun Nullen nach dem Komma: Treffer mit Anzahl der ausgeführten Schläge.
für neues Spiel gehe zu 2

Speicher 0 Schlagzahl 5 Rollweite
1 a und Aufschlags-a 6 a-Rollweite
2 b und Aufschlags-b 7 b-Rollweite
3 waagrechte Geschwind.
4 waagrechte Winkel



Mensch-Ärgere-Dich-Nicht für 5 Spieler mit
 Spielernummer wahlweise automatisch

00				g NOP					
01	15	14	74	f PAUS Neuer Wert	38	14	74	f PAUS Wurf	
02		14	71	f x=y					
03	23	41	03	STO - 3	39		51	+	
04		24	02	RCL 2	40	14	71	f x=y	
05		14	71	f x=y	41	13	47	GTO 47	
06	23	41	02	STO - 2	42	14	41	f x=y	
07		14	73	f LAST x	43	23	01	STO 1	
08			06	6.	44	24	03	RCL 3	
09		14	71	f x=y	45	24	01	RCL 1	
10		13	29	GTO 29	46	13	0100	GTO 0100	
11			03	3	47		34	CLX	
12			32	CHS	48	23	01	STO 1	
13		24	05	RCL 5	49	24	05	RCL 5 Sieger	
14		14	71	f x=y					
15	23	41	05	STO - 5					
16			01	1					
17	23	41	05	STO - 5					
18		24	05	RCL 5					
19		14	74	f PAUS Spielernummer					
20		24	01	RCL 1					
21		24	02	RCL 2					
22		24	03	RCL 3					
23		14	74	f PAUS alter Wert					
24		23	01	STO 1					
25			22	R↓					
26		23	03	STO 3					
27			22	R↓					
28		23	02	STO 2					
29		24	06	RCL 6					
30		24	01	RCL 1					
31		24	00	RCL 0					
32		15	04	g sin ⁻¹					
33		15	01	g FRAG					
34		23	00	STO 0					
35			07	7					
36			61	x					
37		14	01	f INT					

Übernommene Regeln:

Bei Punktgleichheit wird der ursprünglich dort stehende Spieler auf Null gesetzt, auch dann wenn der andere wegen sechs nocheinmal würfelt und dieses Feld gleich wieder verläßt. Bei Sechs darf man also nocheinmal würfeln, auch wenn man damit das Ziel erreicht oder überschreitet. Man muß das Ziel (wählbar STO 6) genau erreichen, zu hohe Würfe bleiben wirkungslos.

geänderte Regeln:

Man kann auch ein Null würfeln. Jeder Spieler startet von Null aus, ohne auf eine Sechs zu warten. Jeder Spieler hat immer nur ein Männchen im Spiel. Geht dieses ins Ziel, erscheint die Nummer des Spielers, und das Programm hält an. Der Rechner zählt nicht mit, wieviele Männchen bereits im Ziel sind. Das nächste Männchen dieses Spielers beginnt automatisch bei Null.

Anweisung:

- 1 Programm eingeben manuell oder automatisch; f PRGM
2. ,xxx... STO 0; Ziel STO 6
3. 0 STO 1, STO2, STO3, STO5;
4. R/S A: Spielernummer mit Minusz. alter Wert Wurf, neuer Wert. Stop oder
5. Bei Wurf sechs A: neuer Wert, 2Wurf, neuer Wert;
6. Bei zu hohem Wurf A: alter Wert (statt neuer Wert)
7. Bei erreichen des Ziels A: neg. Spielernr. (Statt neuer Wert) und Stop auch bei automatischem E
 Notiere ein Männchen mehr im Ziel für diesen Spieler, und gehe zu 4;
8. Für neues Spiel gehe zu 2.

X hier kann zur Kontrolle die zugehörige Spielernummer aus Speicher 5 abgerufen werden, jedoch muß mit R↓ die Anzeige korrigiert werden, bevor wieder gestartet wird.

Mensch-Ärgere-Dich-Nicht ähnlich wie 2
 für deei Süieler mit Spielernummer, wahlweise automatisch

00	15	g	NOF						
01	14	f	PAUS	Neuer Wert					
02	14	f	x=y	40	14	73	f	LAST	x
03	23	41	03	STO - 3	41	06	6		
04	24	02	RCL	2	42	14	71	f	x=y
05	14	71	f	x=y	43	13	25	GTO	25
06	23	41	02	STO - 2	44	24	03	RCL	3
07		03	3		45	24	01	RCL	1
08		32	CHS		46	13	0100	GTO	01 00
09	24	05	RCL	5	47	34	CLX		
10	14	71	f	x=y	48	23	01	STO	1
11	23	41	05	STO - 5	49	24	05	RCL	5
12		01	1						
13	23	41	05	STO - 5					
14	24	05	RCL	5					
15	14	74	f	PAUS	Sp.Nr.				
16	24	01	RCL	1					
17	24	02	RCL	2					
18	24	03	RCL	3					
19	14	74	f	PAUS	alt.Wert				
20	23	01	STO	1					
21		22	R	↓					
22	23	03	STO	3					
23		22	R	↓					
24	23	02	STO	2					
25	24	06	RCL	6					
26	24	01	RCL	1					
27	24	00	RCL	0					
28	15	04	g	sin ⁻¹					
29	15	01	g	FRAC					
30	23	00	STO	0					
31		07	7						
32		61	x						
33	14	01	f	INT					
34	14	74	f	PAUS	Wurf				
35		51	+						
36	14	71	f	x=y					
37	13	47	GTO	47					
38	14	41	f	x<y					
39	23	01	STO	1					

Anweisung siehe Programm 2

geänderte Regeln gegenüber Programm 2
 Geht ein Spieler mit 6 ins Ziel, erlischt sein
 Recht auf nochmaliges Würfeln
 Wirft ein Spieler 6 und anschließend keine Null und
 keinen zu hohen Wurf, sodaß er dieses Feld gleich
 wieder verläßt, so wird dieses Feld nicht angezeigt,
 und ein ev. dort stehender anderer Spieler nicht geworfen!

4

Mensch-Ärgere-Dich-Nicht für
 vier Spieler mit Spielernummer wahlweise autom.

```

00
01 14 41 f x<y 40 15 04 g sin-1
02 23 01 STO 1 41 15 01 g FRAC
03 24 04 RCL 4 42 23 00 STO 0
04 24 01 RCL 1 43 07 7
05 14 71 f x=y 44 51 x
06 23 41 04 STO - 4 45 14 01 f INT
07 24 03 RCL 3 46 14 74 f PAUS(Wurf)
08 14 71 f x=y 47 51 +
09 23 41 03 STO - 3 48 14 61 f x/y
10 24 02 RCL 2 49 13 01 GTO 01
11 24 01 -RCL 1
12 14 71 f x=y
13 23 41 02 -STO - 2 neu
oder R/S 14 14 74 f PAUSE(alt. Wert)
15 14 73 f LAST x
16 06 6
17 14 71 f x=y
18 13 37 GTO 37
19 01 1
20 23 41 07 STO - 7
21 04 4
22 32 CHS
23 24 07 RCL 7
24 14 71 f x=y
25 23 41 07 STO - 7
26 14 74 f PAUS(Sp.Nr.)
27 24 04 RCL 4
28 23 05 STO 5
29 24 03 RCL 3
30 23 04 STO 4
31 24 02 RCL 2
32 23 03 STO 3
33 24 01 RCL 1
34 23 02 STO 2
35 24 05 BCL 5
36 23 01 STO 1
37 24 06 RCL 6
38 24 01 RCL 1
39 24 00 RCL 0
  
```

4

Regeln siehe Programm 2

Anweisung:

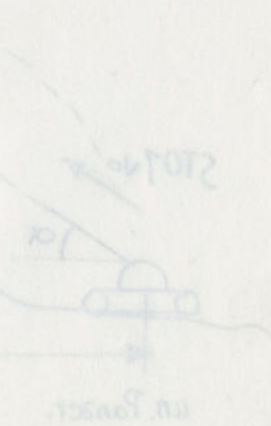
1. Programm eingeben wahlweise **manuell** GTO 19
2. ,xxx... STO 0; 0 STO 1 bis 5, STO 7;
3. Ziel STO 6
4. R/S A: neg Spielernr., Wurf, neuer Wert,R/S
5. bei Wurf 6 A: neuer Wert, (neuer (2.))Wurf, neuer Wert;
6. bei zu hohem Wurf A: Wurf, alter Wert...
7. bei Erreichen des Zieles=A: Ziel Spielernr des Siegers RCL 7 CLX STO 1 R/S siehe 4.
8. für neues Spiel gehe zu 2.

Artillerie-Spiel

```

00
01 31 Enter 39 21 x ← y
02 00 40 23 04 STO 4
03 23 00 STO 0 41 01 41
04 23 05 Sto 5 42 32 CHS
05 22 R↓ 43 23 61 03 STO .x 3
06 23 04 STO 4 44 22 R↓
07 24 01 RCL 1 45 13 06 GTO .06
08 14 09 f →R 46 24 03 RCL 3
09 24 02 RCL 2 47 15 03 g ABS
10 71 + 48 23 03 STO 3
11 15 22 g 1/x 49 23 61 07 STO .x 7
12 61 x
13 14 73 f LAST x
14 15 02 g x²
15 04 4
16 73 .
17 09 9
18 61 x
19 41 -
20 24 06 RCL 6
21 71 +
22 14 01 f INT
23 24 06 RCL 6
24 61 x
25 24 03 RCL 3
26 41 -
27 15 71 g x=0
28 13 46 GTO 46
29 14 74 f PAUSE
30 24 00 RCL 0
31 21 x ← y
32 23 00 STO 0
33 21 x ← y
34 74 R/S
35 24 05 RCL 5
36 51 +
37 24 04 RCL 4
38 23 05 STO 5

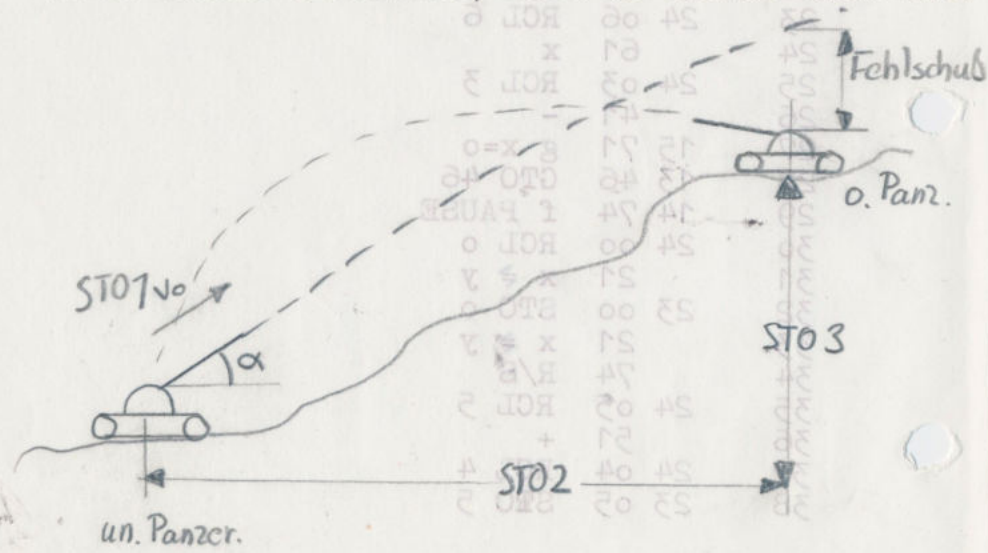
```



Artillerie-Spiel

1. Programm eingeben f PRGM,
2. Daten nach Zeichnung eingeben: v STO 1, Entfernung STO 2, Höhendiff. STO 3; für die Genauigkeit die letzte geltende Stelle z.B. 0,1 STO 6; 9 EEX 99 STO 7;
3. Winkel des unteren Panzers R/S A: (Fehlschuß) 0,00
4. Winkel des oberen Panzers R/S A: (Fehlschuß) Fehlschuß des unteren Panzers.
5. Winkeländerung des unteren Panzers R/S A: (Fehlschuß u.P.) Fehlschuß d. oberen Panzers,
6. Winkeländerung d. oberen Panzers R/S A: (Fehls. d. oberen Panzers), Fehlschuß d. unt. Panzers, gehe zu 5.
7. OF bei treffer des zuletzt schießenden. f PRGM gehe zu 3. oder 2.

Der jeweils schießende Schütze erfährt also sofort seinen Fehlschuß bzw. Treffer, muß sich diesen Wert jedoch nicht merken, da dieser noch einmal erscheint, wenn er wieder dran ist.



6

Sechs verschiedene Zufallszahlen
(z.B. Lottozahlen) :Anweisung:

```

00
01 14 11 00 f FIX 0 39 24 03 RCL 4
02 24 00 RCL 0 40 23 03 STO 3
03 24 07 RCL 7 41 24 05 RCL 5
04 14 33 f REG 42 23 04 STO 4
05 23 07 STO 7 43 24 06 RCL 6
06 22 R 44 23 05 STO 5
07 15 04 g sin-1 45 (14)74 R/S f PAUSE
08 15 01 g FRAC 46 24 00 RCL 0
09 23 00 STO 0 47 13 07 GTO 07
10 24 07 RCL 7 48 24 06 RCL 6
11 61 x 49 32 CHS
12 14 01 f INT
13 23 06 STO 6
14 15 71 g x=y0
15 13 46 GTO 46
16 24 05 RCL 5
17 14 71 f x=y
18 13 46 GTO 46
19 24 04 RCL 4
20 24 06 RCL 6
21 14 71 f x=y
22 13 46 GTO 46
23 24 03 RCL 3
24 14 71 f x=y
25 13 46 GTO 46
26 24 02 RCL 2
27 24 06 RCL 6
28 14 71 f x=y
29 13 46 GTO 46
30 24 01 RCL 1
31 14 71 f x=y
32 13 46 GTO 46
33 15 61 g x=y0
34 13 48 GTO 48
35 24 02 RCL 2
36 23 01 STO 1
37 24 03 RCL 3
38 23 02 STO 2

```

6

Sechs verschiedene Zufallszahlen
(z.B. Lottozahlen) :Anweisung:

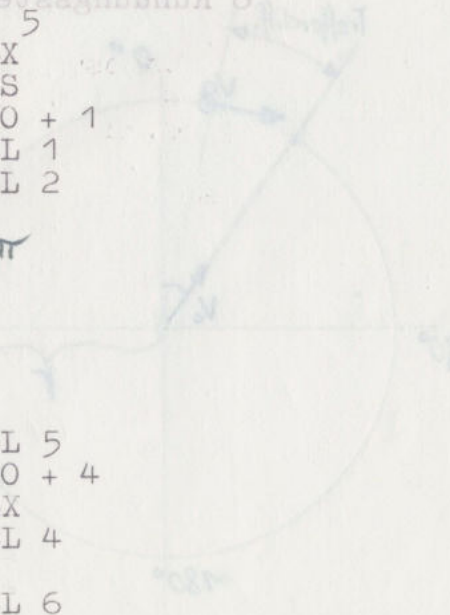
1. Programm eingeben fPRGM
 2. Max. Zahl + 1 Sto 7 (Lotto 50)
 3. Zufallszahl < 1 STO 0;
 4. R/S 1. Zahl, R/S 2. Zahl 6. Zahl mit Minuszeichen
 5. für neuen Tip gehe nach 3 oder 4
- Speicher:
1....6 getippte Zahlen
0 Zufallszahl < 1
7 Max. Zahl (bei Lotto 50) +1

Das Programm "würfelt" sechs Zufallszahlen zwischen 1 und max. Zahl mit recht guter Zufälligkeit, Diese können auch nachträglich von den Speichern abgerufen werden. Sie sind auf jeden Fall voneinander verschieden; deshalb muß die max. Zahl grösser als 6 sein.

Schießen auf bewegtes Ziel

```

00
01 23 03 STO 3 40 71 +
02 24 00 RCL 0 -1 41 14 01 f INT
03 15 04 g sin 42 24 06 RCL 6
04 15 01 g FRAC 43 61 x
05 23 00 STO 0 44 15 61 g x≠0
06 23 61 03 STO x 3 45 13 23 GTO 23
07 74 R/S 46 23 01 STO 1
08 23 02 STO 2 47 09 9
09 02 2 -8-236884
10 71 + 48 23 61 07 STO x 7
11 15 73 g π 49 13 00 GTO 00
12 61 x
13 61 x
14 32 CHS
15 23 04 STO 4
16 24 03 RCL 3
17 24 02 RCL 2
18 61 x
19 74 R/S
20 71 +
21 23 05 STO 5
22 34 CLX
23 74 R/S
24 23 51 01 STO + 1
25 24 01 RCL 1
26 24 02 RCL 2
27 61 x
28 15 73 g π
29 61 x
30 01 1
31 08 8
32 00 0
33 71 +
34 24 05 RCL 5
35 23 51 04 STO + 4
36 34 CLX
37 24 04 RCL 4
38 41 -
39 24 06 RCL 6
    
```

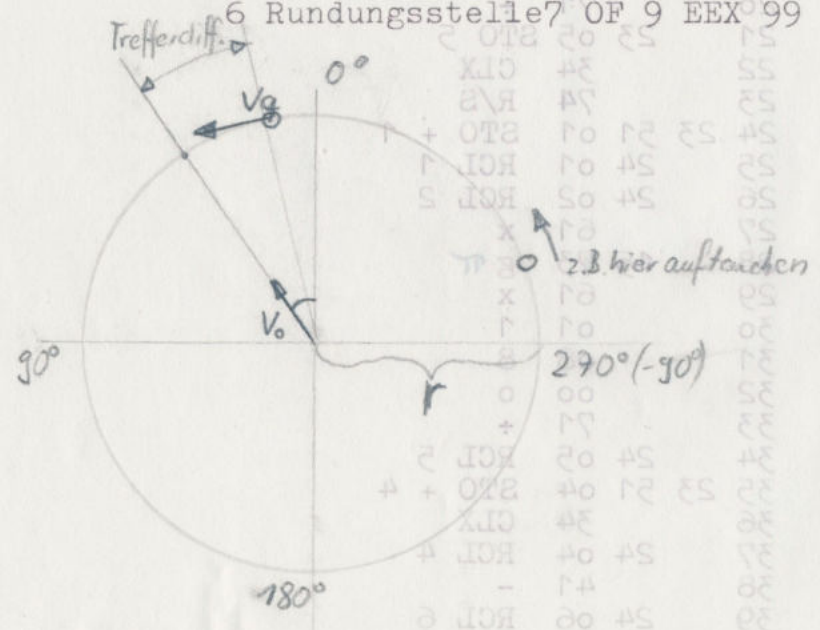


Ein feindliches Objekt taucht im Bereich von -90° bis 0° auf und bewegt sich mit unbekannter Geschwindigkeit kreisförmig um den eigenen Standort. Wähle den Winkel der Abschußrichtung. Es erscheint die Differenz zw zwischen Einschlagstelle und Objekt positiv, wenn Sie zu weit vorgehalten haben und neg. wenn sie hinter dem Objekt treffen würden.

Anweisung:

1. Programm eingeben, f PRGM, o,xxx... STO o,
2. 9 EEX 99 STO 7, Genauigkeit (z.B. o,1) STO 6
3. max. Objektgeschwindigkeit R/S Radius R/S Geschossgeschwindigkeit R/S A: o,oo
4. Winkeländerung (Winkel am Anfang Null) R/S A: Trefferdifferenz, gehe zu 3.4.
5. Bei OF Treffer, gehe zu 2. oder 3.

Speicher: 0 Zufallszahl, 1 alter Winkel
2 Radius 3 Objektgeschw.
4 jeweil. Stando 5 Ortsän.pro Schuß
6 Rundungsstelle 7 OF 9 EEX 99



Rechenübungen

00									
01	24	04	RCL 4	40	14	41	f x<y		
02	24	05	RCL 5	41		21	x < y		
03		1	Rech.zei	42	23	04	STO 4		
04	14	71	f x=y	43		21	x < y		
05	13	16	GTO 16	44	23	05	STO 5		
06		01	1	45		33	EEX		
07	23	51	06	STO + 6	46	02	? Zahl c		
08	24	03	RCL 3	47		71	÷		
09		32	CHS	48		51	+		
10	13	00	GTO 00	49	23	03	STO 3		
11	24	06	RCL 6						
12	24	07	RCL 7						
13		71	+						
14	14	74	f PAUSE						
15	13	38	GTO 38						
16		01	1						
17	23	51	07	STO + 7					
18	24	00	RCL 0						
19	15	04	g sin -1						
20	15	01	g FRAC						
21	23	00	STO 0						
22	24	01	RCL 1						
23		61	x						
24	14	01	f INT						
25	23	04	STO 4						
26	24	02	RCL 2						
27	15	61	g x≠0						
28	13	33	GTO 33						
29	24	04	RCL 4						
30	23	05	RCL 5 STO 5						
31	23	02	STO 2						
32	13	18	GTO 18						
33		34	CLX						
34	23	02	STO 2						
35		01	1) Zahl b						
36		01	1) Zahl b						
37	23	51	04	STO + 4					
38	24	04	RCL 4						
39	24	05	RCL 5						

Beispiel für Großes Einmaleins:
Schritt 03 61 x
a=10 STO 1; b=10 Schr 35,36
c=2 Schr. 46 f FIX 2

Rechenübungen

Das Programm "würfelt" zwei Zahlen deren maximaler und minimaler Wert gewählt werden können. Diese beiden Zahlen erscheinen links und rechts vom Komma. Nun rechnet man eine der vier Grundrechenarten mit diesen beiden Zahlen und gibt das Ergebnis ein. War es richtig, stellt der Rechner eine neue Aufgabe. War es falsch, stellt der Rechner dieselbe Aufgabe nochmals, mit Minuszeichen als Hinweis, daß die Lösung falsch war. Es gibt ~~Yasb~~ nur positive Zahlen. Die linke Zahl ist immer grösser ~~Yas~~ als die rechte, soda bei Differenzen keine negativen Zahlen entstehen. Außerdem zählt der Rechner die Anzahl der gestellten Aufgaben und die Anzahl der falschen Lösungen mit und errechnet auf Wunsch den Quotienten daraus.

Anweisung:

1. Programm eingeben, Punkte freilassen
2. Wähle Rechenart +-x÷ Schritt 03;
3. Linke Zahl immer grösser rechte Zahl
Min. der rechten Zahl 0
Max. der rechten Zahl a
Min. der linken Zahl b
Max. der linken Zahl a+b
Stellenzahl der rechten Zahl c
Wähle a,b,c
a STO 1, b Schritt 35,36, c Schritt 46, f FIX
4. GTO 16 0,xxx,, STO 0 R/S
5. A: linke Zahl komma rechte Zahl
Rechne nach gewählter Rechenart und gib dein Ergebnis ein R/S
6. War es richtig geht der Rechner zu 5.
7. War es falsch, erscheint dieselbe Aufgabe mit Minuszeichen, gehe zu 5.
8. GTO 11 R/S A: (Verhältnis der falschen Lösungen zur Anzahl aller Aufgaben) Rechner geht zu 5.

[b (a+b)] , [0 a]
Stellenzahl c

Motorradrennen

00			41	24 01	RCL 1
01	31	Enter	42	23 00	STO 0
02	04	4	43	24 00	RCL 0 ₋₁
03	71	÷	44	15 04	g sin
04	23 03	STO 3	45	15 01	g FRAC
05	15 02	g x ²	46	23 00	STO 0
06	24 05	RCL 5	47	24 04	RCL 4
07	71	+	48	61	x
08	09	9	49	23 05	STO 5
09	71	+			
10	15 06	g tan ⁻¹			
11	14 74	f PAUSE			
12	06	6)maxWink.			
13	00	0			
14	14 71	f x≥y			
15	13 20	GTO 20			
16	00	0			
17	23 07	STO 7			
18	15 22	g 1/x			
19	13 39	GTO 39			
20	24 05	RCL 5			
21	15 73	g π			
22	61	x			
23	24 03	RCL 3			
24	71	+			
25	23 51 07	STO + 7			
26	01	1			
27	23 51 02	STO + 2			
28	24 02	RCL 2			
29	02	2 Kurv.zahl			
30	14 71	f x≥y			
31	13 43	GTO 43			
32	24 07	RCL 7	32	24 06	RCL 6
33	23 41 07	STO - 7	33	24 07	RCL 7
34	23 51 06	STO + 6	34	14 74	f PAUSE
35	14 74	f PAUSE	35	23 06	STO 6
36	24 06	RCL 6	36	23 41 07	STO - 7
37	32	CHS	37	41	-
38	74	R/S			
39	00	0			
40	23 02	STO 2			

Das Programm gibt den Radius der nächsten Kurve in m an, und man schätzt die Geschwindigkeit, mit der man diese Kurve durchfahren kann, ohne daß der max. Schräglagenwinkel überschritten wird(wählbar mit Schritt:12 13). Die Geschwind. gibt man in km/h an.

Nun erechnet das Programm die Zeit zum Durchfahren der Kurve (Halbkreis) und den nötigen Schräglagenwinkel. War dieser größer als erlaubt, erscheint Error. Ansonsten wird die benötigte Zeit aufaddiert und es erscheint die nächste Kurve.

Beim Rennprogramm wird nach Durchfahren der gesamten Strecke (maximal 10 Kurven) zunächst kurz die neue Rundenzeit und dann die bisherige Gesamtzeit angezeigt. Rundenzahl bitte selbst mitzählen. Nach Ende des Rennens startet - wie bei Bergrennen - der nächste Fahrer.

Beim Trainingsprogramm wird nach Durchfahren der gesamten Strecke kurz die neue Rundenzeit und dann die zeitl. Verbesserung (Verschlechterung negativ) angezeigt.

Anweisung:

1. Renn- oder Trainingspr eingeben, dabei max. Schräglagenwinkel Schritt 12 13 zB. 60
2. GTO 39 Max Kurvenrad STO 4, 0.xxx STO 1
R/S--A+-0700
3. R/S A: Kurvenradius, gib Geschw. ein R/S A: (tats. Schräglagenwinkel), neuer Kurvenradius gehe zu4.
5. Bei Error SST R/S A: Erster Kurvenradius
6. Renn. : nach letzter Kurve
AB (Rundenzeit) , -Gesamtzeit bisher
7. Renn nach letzter Runde notiere Gesamtzeit
0 STO 6 nächster Fahrer beginnt bei 3.
6. Training nach letzter Kurve A: (Rundenzeit), zeitl. Verbesserung zur letzten Runde (Verschlechterung neg). *gehe zu 3 für nächste Runde*

Formeln zur Schräglage von Motorrädern in Kurven

00				
01	14	71	f x=y	Anweisung
02	13	16	GTO 16	1. Programm eingeben f PRGM
03	14	06	f tan	g DEG
04		09	9	2. r ↑ α R/S A: v
05		73	,	oder:
06		08	8	r ↑ v ↑ R/S A: α
07		01	1	3. für neue Rechnung zu 2.
08		61	x	
09		61	x	
10	14	02	f \sqrt{x}	Einheiten:
11		03	3	Kurvenradius in m r
12		73	,	Winkel in Grad α
13		06	6	Geschw. in km/h v
14		61	x	
15	13	00	GTO 00	Besonderheit:
16		22	R ↓	Die beiden Programmteile
17		03	3	werden nur durch zusätz-
18		73	,	liches Drücken der Enter-
19		06	6	taste bei der Eingabe auf-
20		71	÷	gesucht. Man spart das Taste
21	15	02	g x ²	von GTO 16 von Hand.
22		21	x \rightleftharpoons y	
23		71	÷	
24		09	9	
25		73	,	
26		08	8	
27		01	1 \Rightarrow	
28		71	÷	
29	15	06	g tan ⁻¹	
30	13	00	GTO 00	

Superhirn		(keine doppelten Farben)	
Programmteil a		Programmteil b	
00		00	
01	14 33 f REG	01	23 05 STO 5
02	23 07 STO 7	02	24 02 RCL 2
03	01 1	03	24 01 RCL 1
04	23 51 07 STO+7	04	23 02 STO 2
05	22 R ↓	05	22 R ↓
06	22 R ↓	06	23 01 STO 1
07	23 00 STO 0	07	24 05 RCL 5
08	24 00 RCL 0	08	14 61 f x≠y
09	15 04 g sin ⁻¹	09	13 12 GTO 12
10	15 01 g FRAC	10	01 1
11	23 00 STO 0	11	23 51 06 STO+6
12	24 07 RCL 7	12	24 02 RCL 2
13	61 x	13	24 05 RCL 5
14	14 01 f INT	14	14 61 f x≠y
15	15 71 g x=0	15	13 18 GTO 18
16	13 08 GTO 08	16	01 1
17	23 01 STO 1	17	23 51 07 STO+7
18	24 02 RCL 2	18	01 1
19	14 71 f x=y	19	23 51 00 STO+0
20	13 08 GTO 08	20	01 1
21	24 03 RCL 3	21	02 2
22	24 01 RCL 1	22	24 00 RCL 0
23	14 71 f x=y	23	14 71 f x=y
24	13 08 GTO 08	24	13 39 GTO 39
25	24 04 RCL 4	25	03 3
26	14 71 f x=y	26	71 *
27	13 08 GTO 08	27	15 01 g FRAC
28	15 61 g x≠0	28	15 71 g x=0
29	13 37 GTO 37	29	13 00 GTO 00
30	24 03 RCL 3	30	24 02 RCL 2
31	23 04 STO 4	31	24 03 RCL 3
32	24 02 RCL 2	32	24 04 RCL 4
33	23 03 STO 3	33	23 03 STO 3
34	24 01 RCL 1	34	22 R ↓
35	23 02 STO 2	35	23 02 STO 2
36	13 08 GTO 08	36	22 R ↓
37	00 0	37	23 04 STO 4
38	23 07 STO 7	38	13 12 GTO 12
39	23 00 STO 0	39	23 41 00 STO - 0
40	13 00 GTO 00	40	24 07 RCL 7

Gerhard Streichert
 Ralfisenstraße 1
 8431 Postbauer-Hang

Fortsetzung Teil b

41	23 41 07 STO - 7
42	24 06 RCL 6
43	23 41 06 STO - 6
44	04 4
45	14 61 f x≠y
46	13 49 GTO 49
47	34 CLx
48	15 22 g 1/x
49	22 R ↓

Bei fehlerhafter Eingabe
 Tipp trotzdem vervoll-
 ständigen und dann mit richtigem
 Tipp zu 3.

Die Aufgabe des Spiels besteht darin, eine vom Rechner gewürfelte Farbkombination von 4 Farben (dargestellt durch Zahlen) zu ermitteln. Dazu wählt man zunächst auf gut Glück eine Anfangskombination. Der Rechner teilt nun mit, wieviele der Farben bereits an der (A) richtigen Position sind und wieviele sonst noch übereinstimmen, jedoch am falschen Ort sind. (B) Durch geschicktes Verändern der eigenen Farbkombination kann man nun auf einzelne Farben und deren Position schließen bis man schließlich die gesamte Farbkombination ermittelt hat.

Es ist wichtig sich die gesamten Versuche zu protokollieren (wenn man nicht sowieso mit dem Spielbrett und den Farbsteckern arbeitet), damit man sämtliche Versuche und deren Information zur Verfügung hat.

Am besten läßt man den Rechner zunächst aus 6 verschiedenen Farben wählen.

Anweisung:

1. Programmteil a eingeben, f FIX 0, f PRGM, 0,xxx / n R/S Anzeige: 0
 0,xxx bedeutet Zufallszahl kleiner 1, n Farbenanzahl
2. Programmteil b eingeben, f PRGM
3. Eigener Tipp: 1. Farbe R/S Anz: 0
 2. Farbe R/S Anz: 0
 3. Farbe R/S Anz: 0
 4. Farbe R/S Anz: 0
4. Anz: Anzahl der Farben an richtiger Position (A)
 x=y Anzahl der Farben, die übereinstimmen, aber an falscher Position sind (B)
 gehe mit neuem Tipp zu 3.
5. Bei richtiger Farbkombination Anz: Error, das heißt, Spiel gewonnen.
 Anzahl der Versuche bitte selbst mitzählen, sie ist ein Maß für Ihre Fertigkeit

Universelles Nimmspiel

00							
01	23 04	STO 4	42	15 71	g x=0		
02	01	1	43	13 49	GTO 49		
03	51	+	44	13 08	GTO 08		
04	23 03	STO 3	45	24 04	RCL 4		
05	22	R↓	46	13 38	GTO 38		
06	23 00	STO 0	47	24 06	RCL 6		
07	14 11 00	f FIX 0	48	13 00	GTO 00		
08	74	R/S	49	24 05	RCL 5		
09	31	↑					
10	01	1					
11	14 41	f x<y	Speicher				
12	21	x=y					
13	24 04	RCL 4	0	Aktuelles N			
14	14 51	f x>y	1	-			
15	21	x=4	2	-			
16	32	CHS	3	maxZug + 1			
17	14 74	Pause	4	maxZug			
18	23 51 00	STO+0	5	35071 (= I lose)			
19	24 00	RCL 0	6	55178 (= BLIZZ)			
20	14 74	f PAUSE	7	-			
21	15 71	g x=0					
22	13 47	GTO 47					
23	24 03	RCL 3					
24	71	+					
25	15 01	g FRAC					
26	15 71	g x=0					
27	13 45	GTO 45					
28	24 00	RCL 0					
29	14 73	f LAST x					
30	14 01	f INT					
31	24 03	RCL 3					
32	61	x					
33	01	1					
34	51	+					
35	41	-					
36	15 71	g x=0					
37	01	1					
38	32	CHS					
39	14 74	f PAUSE					
40	23 51 00	STO+0					
41	24 00	RCL 0					

Das Nimm-Spiel beginnt damit, daß N gleiche Objekte (zB. Streichhölzer) auf dem Tisch liegen. Die beiden Spieler haben nun abwechselnd ein, zwei oder 3 usw. bis zur vereinbarten Höchstgrenze (=maxZug) der Gegenstände wegzunehmen. Verloren hat derjenige, der gezwungen ist, das letzte zu nehmen.

Dieses Spiel können Sie auch mit dem hp=25 spielen. Begonnen wird mit der Zahl N und der Festlegung von maxZug. Nun haben Sie und der Rechner abwechselnd eins, zwei ... maxZug zu subtrahieren.

Zu Beginn des Spieles müssen Sie Werte für N und maxZug eingeben. Für Anfänger ist zB 15 ↑ 3 möglich. Fortgeschrittene wählen 80 ↑ 10.

Der Rechner zeigt nach Ihrem Zug jedesmal die verbleibende Summe, dann seinen Zug mit Minuszeichen und schließlich den Rest an. Nun sind Sie wieder dran.

Großzügiger Weise läßt Sie der Rechner beginnen. Sie können durchaus gewinnen, wenngleich der hp=25 und sein Programmierer ein Meisterspieler ist. Es ist nämlich nicht möglich einen Fehler zu machen, ohne daß dies der Rechner merkt und sich seinen Sieg sichert.

Wenn Sie schummeln und weniger als eins nehmen, ersetzt der Rechner Ihren Zug durch eins. Ähnlich verfährt er bei zu großem Zug. Er ersetzt diesen Zug durch maxZug. Er zeigt Ihnen als erstes Ihren Zug nochmals mit Minuszeichen an, und zwar korrigiert, wenn er falsch war. Nehmen Sie am Ende des Spiels mehr Gegenstände als noch vorhanden, so fügt der Rechner wieder so viele hinzu, daß eins übrigbleibt. Er ist sogar in der Lage einen Zug wie 6,4 anzunehmen und spielt dann so weiter, daß wieder eine ganze Zahl als Rest bleibt.

Anweisung:

1. Programm eingeben, f PRGM 35071 STO 5, 55178 STO 6
2. N ↑ maxZug R/S Anz: N
3. Ihr Zug R/S, Anz: (-korrigierter Zug, Rest, - sein Zug), Rest, gehe zu 3.
4. Wenn der Rechner gewinnt Anz: BLISS, wenn er verliert ILOSE, Dazu Rechner auf den Kopf stellen.
5. Für neues Spiel gehe zu 2.

Gerhard Streichert

Raiffeisenstraße 1
8431 Postbauer-Heng

Codieren Decodieren

00							
01	14	11	00	f FIX	0	40	41
02		01		1		41	51
03	23	51	01	STO + 1		42	14 11 01
04		21		x=y		43	13 00
05		24	00	RCL	0	44	01
06		23	02	STO 2	-1	45	23 41 01
07		15	04	g sin		46	24 02
08		15	01	g FRAC		47	23 00
09		23	00	STO	0	48	13 26
10		24	07	RCL	7	49	
11		61		x			
12		14	01	f INT			
13		13	16	GTO 16	(RCL 7)	Speicher	
14		21		x=y		0	Aktuelles Z
15		41		-		1	Zählwerk
16		51		+		2	Last Z
17		24	07	RCL	7		
18		01		1		7	N
19		51		+			
20		21		x=y			
21		14	41	f x=y			
22		13	00	GTO	00		
23		24	07	RCL	7		
24		41		-			
25		13	00	GTO	00		
26		24	01	RCL	1		
27		05		5			
28		71		+			
29		01		1			
30		51		+ \XXXX			
31		14	01	f INT			
32		14	73	f LASTx			
33		15	01	g FRAC			
34		02		2			
35		71		+			
36		15	61	g xzo			
37		13	41	GTO	41		
38		73		.			
39		05		5			

1 = 1	R = 27	% = 46
9 = 9	S = 28	& = 47
A = 10	T = 29	(= 48
B = 11	U = 30) = 49
C = 12	V = 31	\$ = 50
D = 13	W = 32	ß = 51
E = 14	X = 33	: = 52
F = 15	Y = 34	
G = 16	Z = 35	
H = 17	, = 36	
I = 18	. = 37	
J = 19	? = 38	
K = 20	! = 39	
L = 21	- = 40	
M = 22	+ = 41	
N = 23	= = 42	
O = 24	; = 43	
P = 25	" = 44	
Q = 26	/ = 45	

Mit dem Programm lassen sich Texte mit beliebig vielen verschiedenen Zeichen in Codetexte mit gleichvielen Zeichen übersetzen und rückübersetzen. Da der Nachrichtenkanal meist nur eine bestimmte Anzahl verschiedener Zeichen zuläßt, ist der Klartext zunächst auf den Kanal abzustimmen. Nicht vorhandene Zeichen können durch textliche Abkürzungen ersetzt werden.

Außer Rechner und Programm wird zum Ver- und Entschlüsseln noch eine Liste der zu übertragenden Zeichen benötigt. Dabei sind die Zeichen durchnummerieren. Die Anzahl der Zeichen heißt N. Umseitig ist ein Beispiel für Schreibmaschinentext angegeben. (Folien Seite 35)

Um eine Entschlüsselung anhand der Wortlänge zu vermeiden, wird der Klartext zunächst in Gruppen zu fünf Buchstaben (Zeichen) niedergeschrieben. Davor wird eine beliebige Zeichengruppe mit 5 Zeichen gesetzt. Diese wird nicht verschlüsselt sondern gelangt direkt auf den N-Kanal. Aus ihr wird sowohl beim Ver- als auch beim Entschlüsseln die Zahl Z gebildet, indem mit der Tabelle die Buchstaben (Zeichen) in Zahlen umgesetzt werden. Jetzt schreibt man noch o. davor und tippt sie so in den Rechner als Z.

Nun gibt man die Nummer des ersten Zeichens des Klartextes in den Rechner ein und erhält nach R/S die Nummer des Codezeichens. Beim Entschlüsseln verfährt man genauso mit geändertem Programmschritt 13 (RCL 7). Mit GTO 26 R/S kann die Nummer des zuletzt verarbeiteten Zeichens ermittelt werden. Die Zahl vor dem Komma bedeutet die Nummer der 5-er-Gruppe, die nach dem Komma die Nummer des Zeichens in dieser Gruppe. Mit GTO 44 R/S kann bei fehlerhafter Eingabe der Rechner auf den vorherigen Buchstaben zurückgesetzt werden; jedoch nur einmal nacheinander anwenden! Es erscheint die Nummer des letzten richtigen Zeichens.

Anweisung = Ver-schlüsseln

1. Programm eingeben, f PRGM
2. Anzahl der Zeichen n STO 7
3. Z STO 0 D STO 1
4. Nummer des ersten/nächsten Zeichens laut Tabelle eingeben R/S A: Nummer des Codezeichens für N-Kanal.
5. gehe zu 4. bis Text durch.

EntschlüsseIn

1. Programm eingeben Schritt 13 (RCL 7), f Prgm
2. und 3. wie oben
4. Nummer des Ersten/nächsten Codezeichens eingeben R/S

A: Nummer des Klartextzeichens. Wiederhole 4. bis text durch

GTO 26 R/S A: Nummer des letzten Zeichens (Gruppe, Zeichen)
GTO 44 R/S A: Nummer des vorletzten Zeichens ("")

XRA! 2)0AR -x"QO vFOA JERYL

Bei Fehler

Ver- Entschlüsseln, für einfachen Rechner als Partner

00					
01	24	00	RCL 0	Speicher	
02	23	01	STO 1		
03	24	04	RCL 4	0	Aktuelles Z_n
04		61	x	1	Last z_{n-1}
05	24	03	RCL 3	2	frei
06		61	x	3	EEX 7 (bei 8-stelligem Partner-Rechner)
07	14	01	f INT		
08	24	03	RCL 3	4	3.14100000
09		71	+	5	N = 35
10	15	01	g FRAC	6	frei
11	23	00	STO 0	7	frei
12	24	05	RCL 5		
13		61	x		
14	14	01	f INT		
15	(41)	51	+ (-)		
16	24	05	RCL 5 (GTO 22)		
17	14	51	f $x \Rightarrow y$		
18	13	21	GTO 21		
19		41	-		
20	13	00	GTO 00		
21		21	x == y		
22	15	51	g $x \geq 0$		
23	13	00	GTO 00		
24	24	05	RCL 5		
25		51	+		
26	13	00	GTO 00		

Gerhard Streichert
Raiffeisenstraße 1
8431 Postbauer-Heng

Anweisung:

SPEICHER 3 und 4 wie oben

1. Programm eingeben, f PRGM f FIX 0, Schritt 15, 16 bei verschlüsseln: + RCL 5, beim Entsch. -GTO 22
 2. Anzahl (N = 35) der Zeichen STO 5;
 3. Z STO 0
 4. Nummer des ersten/nächsten Zeichens laut Tabelle eingeben R/S A: Nummer des Codezeichens
 5. gehe zu 4. bis Text durch.
- Zum Entschlüsseln:
1. Programm eingeben, f PRGM, f FIX 0, Schritt 15, 16 - GTO 22;
 2. und 3. wie oben;
 4. Nummer des Codezeichens eingeb. R/S A: Nummer des Klartextzeichens, Gehe zu 4. bis Text durch.

14a

00				
01	01	1		Gerhard Streichert
02	23	01	STO 1	Raiffeisenstraße 1
03	23	02	STO 2	8431 Postbauer-Heng
04	24	01	RCL 1	
05	15	02	g x ²	Speicher
06	24	02	RCL 2	
07	15	02	g x	0 Grenze
08		51	+	1 erste Kathete
09	14	02	f \sqrt{x}	2 zweite Kathete
10	23	03	STO 3	3 Hypothenuse
11	15	01	g FRAC	4 frei
12	15	61	g x ² / 0	
13	13	24	GTO 24	
14	24	02	RCL 2	
15	24	01	RCL 1	
16	14	41	f x ² - y	
17	13	24	GTO 24	
18	24	02	RCL 2	
19	14	74	f PAUSE	
20	24	01	RCL 1	
21	14	74	f PAUSE	
22	24	03	RCL 3	
23		74	R/S	
24		01	1	
25	23	51	STD + 1	
26	24	01	RCL 1	
27	24	00	RCL 0	
28	14	51	f x ² ≥ y	
29	13	04	GTO 04	
30		01	1	
31	23	01	STO +2	
32	23	51	02 RCL +2	
33	24	02	STO 2	
34	24	00	RCL 0	
35	14	51	f x ² ≥ y	
36	13	04	GTO 04	
37		00	0	
38	15	22	g 1/x	

RCL 1
 RCL 2
 kx=y
 GTO 23
 x
 2
 x
 PAUSE
 RCL 1
 x²
 RCL
 g x²
 PAUSE
 LGSTX
 LGSTX
 +
 PAUSE
 1
 23 STO+1
 ...
 ...

32

Sind die drei Seiten eines Dreiecks ganzzahlige Vielfache einer Einheit, so nennt man diese Zahlen Pythagoräische Zahlentrippele.

Das Programm untersucht alle natürlichen Zahlen bis zur vorgegebenen Obergrenze nach solchen Zahlentrippele.

Dabei läuft eine Kathete zunächst von eins bis zur Grenze, während die zweite Kathete eins ist. Danach wird die zweite Kathete zwei gesetzt und die erste läuft wieder von eins bis zur Grenze usw.

Da die beiden Katheten vertauschbar sind, würden die gleichen Zahlentrippele zweimal in verschiedener Reihenfolge auftauchen. Dies wird vermieden, da die erste Kathete immer kleiner als die zweite sein muß, damit eine Anzeige erfolgt.

- Anweisung
1. Programm eingeben, f PRGM, f FIX 0,
 2. Grenze STO 0
 3. R/S A: (1. Kathete), (2. Kathete) Hypothenuse
 4. R/S gehe zu 3. bis A: Error dann sind alle Zahlen bis zur Grenze untersucht worden.

32

Auffinden Pythagoräischer Zahlen

(Formelmäßiges Auffinden, sehr schnell, jedoch ungeordnet)

```

00
01      01  1
02      23 04 STO 4
03      23 05 STO 5
04      24 04 RCL 4
05      24 05 RCL 5
06      14 71 -f x=y
07      13 32 GTO 32
08      61 x
09      02 2
10      61 x
11      23 01 STO 1
12      24 04 RCL 4
13      15 02 g x^2
14      24 05 RCL 5
15      15 02 g x^2
16      41 -
17      23 02 STO 2
18      14 73 f LASTx
19      51 +
20      14 73 f LASTx
21      51 +
22      23 03 STO 3
23      24 01 RCL 1
24      24 02 RCL 2
25      14 51 f x>=y
26      21 x==y
27      14 74 f PAUSE
28      21 x==y
29      14 74 f PAUSE
30      24 03 RCL 3
31      74 R/S
32      01 1
33 23 51 04 STO + 4
34      24 04 RCL 4
35      24 00 RCL 0
36      14 51 f x>=y
37      13 04 GTO 04
38      01 1
39 23 51 05 STO + 5

```

Gerhard Streichert
Raiffeisenstraße 1
8431 Postbauer-Heng

Auffinden Pythagoräischer Zahlen

Sind die drei Seiten eines rechtwinkligen Dreiecks ganzzahlige Vielfache einer Einheit so nennt man diese Zahlen pythagoräische Zahlen
Formel: $a^2 + b^2 = c^2$

wobei $a, b, c \in G$

Das Programm errechnet solche Zahlen nach der Formel:

$$a = 2pq, b = p^2 - q^2, c = p^2 + q^2;$$

wobei $p, q \in G$

Dabei läuft p von eins bis zur gegebenen Grenzwiederholt durch, wobei jedesmal q um eins erhöht wird, bis auch dieses die Grenze erreicht.

Um das Erscheinen gleicher Zahlen zu vermeiden, beginnt p immer erst bei dem jeweiligen q zu laufen.

Das Programm findet sehr schnell pythagoräische Zahlen, diese sind jedoch nicht geordnet.

Mondlandung

15

```

00
01 24 05 RCL 5 41 01 1
02 23 02 STO 2 42 00 0
03 24 07 RCL 7 43 61 x
04 23 00 STO 0 44 51 +
05 24 04 RCL 4 45 14 02 x
06 23 01 STO 1 46 32 CHS
07 14 11 04 f FIX 4 47 23 01 STO 1
08 24 00 RCL 0 48 24 01 RCL 1
09 24 06 RCL 6 49 14 11 00 f FIX 0
10 71 +
11 24 01 RCL 1
12 15 41 g x<0
13 13 16 GTO 16
14 51 +
15 13 18 GTO 18 Speicher
16 21 x==y
17 41 -
18 74 R/S 0 Höhe x
19 24 02 RCL 2 1 Geschwind. v
20 14 41 f x<y 2 Treibstoff f
21 13 38 GTO 38 3 Beschleunig. a
22 22 R↓ Anfangswerte:
23 23 41 02 STO -2 4 Geschwind. v=-50
24 05 5 5 Treibstoff f=120
25 41 - 6 EEX 4
26 23 03 STO 3 7 Höhe x=500
27 02 2
28 71 +
29 24 01 RCL 1
30 51 +
31 23 51 00 STO + 0
32 24 00 RCL 0
33 15 41 g x<0
34 13 48 GTO 48
35 24 03 RCL 3
36 23 51 01 STO + 1
37 13 08 GTO 08
38 24 01 RCL 1
39 15 02 g x^2
40 24 00 RCL 0
    
```

Gerhard Streichert
Raiffeisenstraße 1
8431 Postbauer-Heng

Mondlandung

12

Dieses Mondlandeprogramm wurde weitestgehend aus dem hp-25 Buch übernommen. Durch kürzen einiger Programmteile gelang es, das Programm in der Benutzung zu vereinfachen:

Die Anfangsdaten müssen nur beim ersten Mal eingegeben werden. Nach dem Aufschlag auf dem Mond brauchen Sie nur noch R/S zu drücken um das Spiel von vorn zu beginnen.

Das Spiel kann auch mit veränderten Anfangswerten gespielt werden. Diese erscheinen dann zu Beginn der nächsten Spiele.

Anweisung

1. Programm eingeben f PRGM, Höhe STO 7, EEX 4 STO 6, Brennstoffvorrat STO 5, Geschwindigkeit STO 4;
2. R/S A+ V?X
3. Bremsstoß eingeben gehe zu 2.
4. Falls A: V, (Also keine Nachkommastellen) Ihre Aufschlagsgeschwindigkeit. Für neues Spiel R/S gehe zu 2.

Gerhard Streichert
Raiffeisenstraße 1
8431 Postbauer-Heng

(n/k) berücksichtigt alle Definitionen

00							
01	23	01	STO 1	40		61	x
02	15	71	g x=0	41	13	31	GTO 31
03	13	45	GTO 45	42		22	R↓
04	15	41	g x<0	43		22	R↓
05	13	47	GTO 47	44	13	00	GTO 00
06	15	01	g FRAC	45		01	1
07	15	61	g x≠0	46	13	00	GTO 00
08	13	47	GTO 47	47		34	CLX
09		21	x ≅ y	48	15	22	g 1/x (Error)
10	23	02	STO 2	49		00	o
11	24	01	RCL 1				
12	14	71	f x=y				
13	13	45	GTO 45				
14	24	02	RCL 2				
15	15	41	g x<0				
16	13	28	GTO 28				
17	15	01	g FRAC				
18	15	61	g x≠0				
19	13	28	GTO 28				
20	24	01	RCL 1				
21	24	02	RCL 2				
22	14	41	f x<y				
23	13	49	GTO 49				
24	24	01	RCL 1				
25		41	-				
26	14	41	f x<y				
27	23	01	STO 1				
28	24	02	RCL 2				
29	24	01	RCL 1				
30		71	÷				
31		01	1				
32	23	41	01 STO - 1				
33	23	41	02 STO - 2				
34		34	CLX				
35	24	02	RCL 2				
36	24	01	RCL 1				
37	15	71	g x=0				
38	13	42	GTO 42				
39		71	÷				

Anweisung:

1. Programm eingeben f PRGM
2. n Enter k R/S A: Ergebnis
3. Wiederhole 2. beliebig
4. bei Error: nicht definiert

Es werden alle Definitionen berücksichtigt und der jeweils kürzeste Weg ausgewählt.

y^x bei pos. und neg. Basis

00			
01	23	01	STO 1
02		21	$x \leq y$
03	23	00	STO 0
04		31	↑
05	15	03	g ABS
06	14	71	f $x=y$
07	13	21	GTO 21
08	24	01	RCL 1
09	15	01	g FRAC
10	15	71	g $x=0$
11	13	15	GTO 15
12	24	00	RCL 00
13	24	01	RCL 1
14	14	03	f y^x Error
15	24	01	RCL 1
16		02	2
17		71	÷
18	15	01	g FRAC
19	15	61	g $x \neq 0$
20	13	26	GTO 26
21	24	00	RCL 0
22	15	03	g ABS
23	24	01	RCL 1
24	14	03	f y^x
25	13	00	GTO 00
26	24	00	RCL 0
27	32		CHS
28	24	01	RCL 1
29	14	03	f y^x
30		32	CHS
31	13	00	GTO 00

1. Programm eingeben
f PRGM
2. Basis ↑ EXP. R/S
3. bei Error Wurzelprg.
für komplexe Zahlen

```

00
01 23 06 STO 6
02 22 RCL 5)GNOP
03 23 05 STO 5)GNOP
04 22 RCL 5)GNOP
05 21 x=y
06 23 04 STO 4
07 22 RCL 5)GNOP
08 03 3
09 21 X=y
10 14 41 f x<y
11 13 16 GTO 16
12 01 1
13 51 +
14 24 05 RCL 5
15 13 22 GTO 22
16 01 1
17 03 3
18 51 +
19 24 05 RCL 5
20 01 1
21 41 -
22 24 03 RCL 3
23 61 x
24 14 01 f INT
25 21 x=y
26 24 02 RCL 2
27 61 x
28 14 01 f INT
29 51 +
30 24 04 RCL 4
31 51 +
32 24 07 RCL 7
33 21 x=y
34 23 07 STO 7
35 21 x=y
36 41 -
37 14 74 f PAUS
38 24 03 RCL 3
39 71 +
40 24 01 RCL 1

```

Anweisung

1. Programm eingeben
f PRGM, 30.6 STO 2,
365.25 STO 3
2. Zinssatz dezimal STO1
3. 1. Datum ↑ o R/S
4. 2. Datum ↑ Kapit R/S
5. Wiederhole 4. bis
zum letzten Datum
A: Zinsen
6. Für neue Rechnung
o STO o gehe zu 2.

Das Datum wird entweder
so eingegeben

Tag ↑ Monat ↑ Jahr

oder

Tag ↑ Monat

je nachdem ob in

Schritt 2 und 3 R STO5

oder g NOP steht.

Wird die Jahresangabe

nicht ständig wiederholt

so muß sie am Anfang in

STO 5 eingegeben werden.

Anzeige: zunächst die

Tagesdifferenz, dann die

Zinsen für diesen Zeit-

raum und schließlich die

Zinssumme.

Beispiel Zinssatz 3.4 % = 0.034 STO 1

```

1 1 1976 10 R/S
5 11 1976 1625 R/S
7 8 1977 3451 R/S

```

```

A: N 0 0
A: 309 46.74 46.74 DM
A: 275 88.34 135.08 DM

```

Tage Zinsen Zinssumme

Speicheranweisung

1. Programm eingeben
2. Zinssatz dezimal STO1
3. 1. Datum ↑ o R/S
4. 2. Datum ↑ Kapit R/S
5. Wiederhole 4. bis
zum letzten Datum
A: Zinsen
6. Für neue Rechnung
o STO o gehe zu 2.

Speicher

- 0 Zinssumme
- 1 Zinssatz in dezimal
- 2 30.6
- 3 365.25
- 4 TAG
- 5 JAHR
- 6 jew. Kapital
- 7 N alt

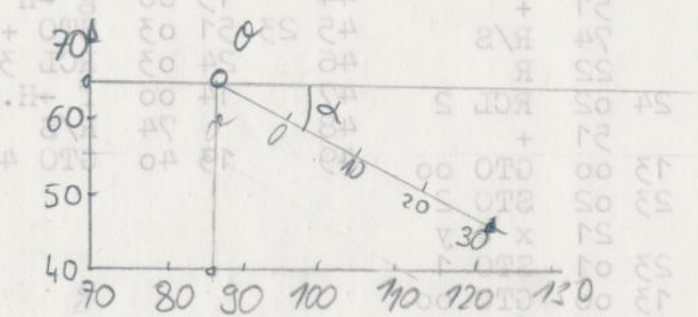
Kombination für Orientierungsfahrt

5

00									
01	21	x	y	41	33	EEX			
02	14	09	f	-R	42	-02	2		
03	24	01	RCL	1	43	71	+		
04	51		+		44	15 00	g	H	
05	74	R/S			45	23 51 03	STO	+	3
06	22		R		46	24 03	RCL	3	
07	24	02	RCL	2	47	14 00	f	H.MS	
08	51		+		48	74	R/S		
09	13	00	GTO	00	49	13 40	GTO	40	
10	23	02	STO	2					
11	21	x	y						
12	23	01	STO	1					
13	13	00	GTO	00					
14	31		f						
15	03		3						
16	73		.						
17	06		6						
18	71		+						
19	71		+						
20	06		6						
21	00		0						
22	71		+						
23	14	00	f	H.MS					
24	74		R/S						
25	13	14	GTO	14					
26	31		f						
27	06		6						
28	00		0						
29	61		x						
30	71		+						
31	03		3						
32	73		.						
33	06		6						
34	61		x						
35	74		R/S						
36	13	26	GTO	26					
37	15	00	g	H					
38	23	03	STO	3					
39	74		R/S						
40	31								

GTO 10

1. Polarkoordinaten in Rechtwinklige
 Polarursprung eingeben hier: 85 f 65 GTO R/
 Radius und Winkel eingeben hier: 40 f -30
 R/S x-Koordinate 119,64
 R/S y-Koordinate 45



2. GTO 25
 Schnitt ausrechnen (ges. Zeit)
 Strecke in m f Geschw. in km/h R/S A: zeit m.s
3. GTO 36
 Schnitt ausrechnen (mittlere Geschw.)
 Strecke in m f Zeit in min R/S A: Geschw. km/h
4. GTO 37
 Zeittabelle ausrechnen
 Startzeit in H.MS R/S
 Zeitdifferenz in HM.S
 R/S: Zeitpunkt in H.MS
 Vorsicht 60s = 1 Min mehr
 ev. f FIX 4

Sechs Speicherinhalte der Größe nach ordnen

00	x	51	04	00			
01	01	1	01	01	34	CLX	
02	23	07	STO 7	02	23	07	STO 7
03	24	01	RCL 1	03	24	01	RCL 1
04	24	02	RCL 2	04	24	02	RCL 2
05	14	51	f x=y	05	14	51	f x=y
06		21	x ≤ y	06		21	x ≤ y
07	23	01	STO 1	07	23	01	STO 1
08		22	R↓	08		22	R↓
09	23	02	STO 2	09	23	02	STO 2
10	24	01	RCL 1	10	24	03	RCL 3
11	23	00	sto 0	11	14	51	f x=y
12	24	02	RCL 2	12	21	x = y	
13	23	01	STO 1	13	23	02	STO 2
14	24	03	RCL 3	14	22	R↓	
15	23	02	STO 2	15	23	03	STO 3
16	24	04	RCL 4	16	24	04	RCL 4
17	23	03	STO 3	17	14	51	f x=y
18	24	05	RCL 5	18	21	x = y	
19	23	04	STO 4	19	23	03	STO 3
20	24	06	RCL 6	20	22	R↓	
21	23	05	STO 5	21	23	04	STO 4
22	24	00	RCL 0	22	24	05	RCL 5
23	23	06	STO 6	23	14	51	f x=y
24		01	1	24	21	x = y	
25	23	51	07 STO + 7	25	23	04	STO 4
26		03	3	26	22	R↓	
27		01	1	27	23	05	STO 5
28	24	07	RCL 7	28	24	06	RCL 6
29	14	71	f x=y	29	14	51	f x=y
30	13	00	GTO 00	30	21	x = y	
31		66	6	31	23	05	STO 5
32		71	+	32	22	R↓	
33	15	01	g FRAC	33	23	06	STO 6
34	15	71	g x=0	34	01	1	
35	13	10	GTO 10	35	23	05	STO +7
36	13	03	GTO 03	36	05	5	

Die Speicherinhalte von 37
1 bis 6 werden der Größe
nach angeordnet. Speicher
0 und 7 belegt.

8 Speicherinhalte ordnen (selbst mitzählen)

00				40		21	x ≤ y
01	24	00	RCL 0	41	23	06	STO 6
02	24	01	RCL 1	42	22	R↓	
03	14	51	f x=y	43	23	07	STO 7
04		21	x = y	44	14	74	f PAUSE
05	23	00	STO 0	45	13	01	GTO 01
06		22	R↓				
07	23	01	STO 1				Stoppe beim 7. mal Pause
08	24	02	RCL 2				
09	14	51	f x=y				
10		21	x = y				
11	23	01	STO 1				
12		22	R↓				
13	23	02	STO 2				
14	24	03	RCL 3				
15	14	51	f x=y				
16		21	x = y				
17	23	02	STO 2				
18		22	R↓				
19	23	03	STO 3				
20	24	04	RCL 4				
21	14	51	f x=y				
22		21	x = y				
23	23	03	STO 3				
24		22	R↓				
25	23	04	STO 4				
26	24	05	RCL 5				
27		21	x = y				
28	23	04	STO 4				
29		22	R↓				
30	23	05	STO 5				
31	24	06	RCL 6				
32	14	51	f x=y				
33		21	x = y				
34	23	05	STO 5				
35		22	R↓				
36	23	06	STO 6				
37	24	07	RCL 7				
38		21	x = y				
39	14	51	f x=y				

Gebühr für Postüberweisung ins Ausland
Stand 1.1.75

00					
01	31	↑	41	61	x
02	33	EEX	42	01	1
03	02	2	43	01	1
-04	71	÷	44	73	,
-05	14 01	f INT	45	05	5
-06	23 01	STO 1	46	51	+
-07	01	1	47	24 02	RCL 2
-08	00	0	48	14 41	f x < y
09	14 51	f x = y	49	21	x ≠ y
10	13 33	GTO 33			
11	24 01	RCL 1	Für Auslandspostüberwei-		
12	33	EEX	sungen erhebt die Post		
13	02	2	Gebühren in Abhängig-		
14	14 51	f x = y	keit von dem zu überwei-		
15	13 27	GTO 27	senden Betrag.		
16	24 01	RCL 1	Anweisung:		
17	33	EEX	1. Progr. eingeb. f PRGM		
18	03	3	2. ,25 STO 2		
19	14 41	f x < y	3. überweisungsbetrag R/S		
20	13 37	GTO 37	4. A: Gebühr gehe zu 3.		
21	73	,			
22	= 00	0			
23	= 04	4			
24	24 01	RCL 1	Beispiele:		
25	61	x	3,25	R/S A:	0,25
26	13 43	GTO 43	234,45	R/S A:	0,25
27	73	,	745,14	R/S A:	0,70
28	00	0	3526,20	R/S A:	2,25
29	05	5	27 234	R/S A:	12,38
30	24 01	RCL 1	345 123,34	R/S A:	115,03
31	61	x			
32	13 44	GTO 44			
33	24 01	RCL 1			
34	21	x ≠ y			
35	71	÷			
36	13 47	GTO 47			
37	24 01	RCL 1			
38	73	,			
39	00	0			
40	03	3			

Gebühr für Postüberweisung ins Ausland
Stand 1.1.75

00					
01	31	↑	41	61	x
02	33	EEX	42	01	1
03	02	2	43	01	1
-04	71	÷	44	73	,
-05	14 01	f INT	45	05	5
-06	23 01	STO 1	46	51	+
-07	01	1	47	24 02	RCL 2
-08	00	0	48	14 41	f x < y
09	14 51	f x = y	49	21	x ≠ y
10	13 33	GTO 33			
11	24 01	RCL 1	Das Programm rechnet folgendermaßen:		
12	33	EEX	Betrag ÷ 100 → x		
13	02	2	x - 3 → ja		
14	14 51	f x = y	x ≤ 10? → ja → x · 0,1 = Ergebnis		
15	13 27	GTO 27	nein → ja → x · 0,05 + 10,5 = Ergebnis		
16	24 01	RCL 1	Anweisung:		
17	33	EEX	1000 < x? → ja → x · 0,03 + 11,5 = Ergebnis		
18	03	3	ja nein → ja → x · 0,04 + 1,5 = Ergebnis		
19	14 41	f x < y	Ergebnis < 0,25 → ja → Endergebnis = 0,25		
20	13 37	GTO 37	nein → ja → Endergebnis = Ergebnis		
21	73	,			
22	= 00	0			
23	= 04	4			
24	24 01	RCL 1	Beispiele:		
25	61	x	3,25	R/S A:	0,25
26	13 43	GTO 43	234,45	R/S A:	0,25
27	73	,	745,14	R/S A:	0,70
28	00	0	3526,20	R/S A:	2,25
29	05	5	27 234	R/S A:	12,38
30	24 01	RCL 1	345 123,34	R/S A:	115,03
31	61	x			
32	13 44	GTO 44			
33	24 01	RCL 1			
34	21	x ≠ y			
35	71	÷			
36	13 47	GTO 47			
37	24 01	RCL 1			
38	73	,			
39	00	0			
40	03	3			

8

Primfaktorzerlegung

```

00  Das Programm zerlegt die gegebene Zahl in
01 23 00 STO 0 41 23 51 04 STO + 4
02 09 0 0 EX 42 23 23 GTO 23
03 09 0 1 43 24 02 RCL 2
04 09 0 0 44 (14)74 f(PAUSE)R/S
05 09 0 0 45 24 03 RCL 3
06 09 0 1 46 23 01 STO 1
07 09 0 0 47 13 23 GTO 23
08 09 0 0 48 34 CLX
09 09 0 0 49 15 22 g 1/x
10 09 0 0
11 14 41 f x<y
12 13 48 GTO 48
13 22 R ↓
14 15 03 ABS
15 23 01 STO 1
16 15 01 g FRAC
17 15 61 g x≠0
18 13 48 GTO 48
19 02 2
20 23 02 STO 2
21 01 1
22 23 04 STO 4
23 24 02 RCL 2
24 24 01 RCL 1
25 14 51 f x>y
26 13 29 GTO 29
27 24 00 RCL 0
28 13 00 GTO 00
29 21 x = y
30 71 ÷
31 14 01 f INT
32 23 03 STO 3
33 14 73 f LAST x
34 15 01 ABS
35 15 71 g x=0
36 13 43 GTO 43
37 24 04 RCL 4
38 23 51 02 STO + 2
39 01 1
40 14 71 f x=y

```

8

Primfaktorzerlegung

Das Programm zerlegt die gegebene Zahl in seine Primfaktoren. Dabei darf der Betrag der Zahl nicht kleiner als eins und - wegen der Rechengenauigkeit - nicht größer als 999999999 sein, sonst erscheint Error. Außerdem muß es sich um Ganze Zahlen handeln.

Die Primfaktoren erscheinen der Reihe nach bis die Zahl selbst erscheint. Negative Zahlen werden wie positive behandelt.

```

00  Das Programm zerlegt die gegebene Zahl in
01 23 00 STO 0 41 23 51 04 STO + 4
02 09 0 0 EX 42 23 23 GTO 23
03 09 0 1 43 24 02 RCL 2
04 09 0 0 44 (14)74 f(PAUSE)R/S
05 09 0 0 45 24 03 RCL 3
06 09 0 1 46 23 01 STO 1
07 09 0 0 47 13 23 GTO 23
08 09 0 0 48 34 CLX
09 09 0 0 49 15 22 g 1/x
10 09 0 0
11 14 41 f x<y
12 13 48 GTO 48
13 22 R ↓
14 15 03 ABS
15 23 01 STO 1
16 15 01 g FRAC
17 15 61 g x≠0
18 13 48 GTO 48
19 02 2
20 23 02 STO 2
21 01 1
22 23 04 STO 4
23 24 02 RCL 2
24 24 01 RCL 1
25 14 51 f x>y
26 13 29 GTO 29
27 24 00 RCL 0
28 13 00 GTO 00
29 21 x = y
30 71 ÷
31 14 01 f INT
32 23 03 STO 3
33 14 73 f LAST x
34 15 01 ABS
35 15 71 g x=0
36 13 43 GTO 43
37 24 04 RCL 4
38 23 51 02 STO + 2
39 01 1
40 14 71 f x=y

```

Kürzen zweier Zahlen

00									
01	14	51	f. $x \geq y$	41	24	00	RCL 0		
02		21	$x \neq y$	42	14	74	f PAUS		
03		23	01. STO 1	43	13	11	GTO 11		
04		22	R ↓	44	13	00	GTO 00		
05		23	02. STO 2						
06			02 : 2						
07		23	03. STO 3						
08			01 1						
09		23	00. STO 0					Anweisung	
10		23	05. STO 5					1. Programm eing f PRGM	
11		24	01. RCL 1					2. a f b R/S	
12		24	03. RCL 3					3. (A:) wachsender	
13			71 +					Kürzungsfaktor	
14		15	01. g FRAC					4. A: größter gemeins.	
15		15	71. g $x=0$					Teiler, die gekürzter	
16		13	27. GTO 27					Zahlen stehen in Spei-	
17		24	05. RCL 5					cher 1 und 2. für	
18	23	51	03. STO + 3					neue Rechnung zu 2.	
19			01 1						
20		14	71. f $x=y$						
21	23	51	05. STO + 5						
22		14	73. f Last x						
23		14	51. f $x \geq y$						
24		13	11. GTO 11						
25		24	00. RCL 0						
26		13	00. GTO 00						
27		14	73. f Last x						
28		23	04. STO 4						
29		24	02. RCL 2						
30		24	03. RCL 3						
31			71 +						
32		15	01. g FRAC						
33		15	61. g $x \neq 0$						
34		13	17. GTO 17						
35		14	73. f Last x						
36		23	02. STO 2						
37		24	04. RCL 4						
38		23	01. STO 1						
39		24	03. RCL 3						
40	23	61	00. STO x 0						

Das Programm ermittelt den größten gemeinsamen Teiler zweier Zahlen, indem es diese probeweise durch die Primzahlen teilt. Sind beide ohne Rest teilbar, wird dies ausgeführt, und weitere Primzahlen versucht, bis eine Zahl kleiner 1 würde.

Eine Besonderheit: Zur Ermittlung der Primzahlen werden die Zahlen der Reihe nach durch alle Zahlen ab 2 probeweise geteilt. Dadurch entstehen die Primzahlen automatisch. Da außer 2 alle Primzahlen ungerade sind, wird ab 3 in Zwischenschritten vorgegangen um Rechenzeit zu sparen.

Beispiel zu Kürzen:

$$\frac{1932 \cdot 220}{3864} = ?$$

1932 / 3864 R/S Anz: (2, 4, 12, 84, 1932), 1932

RCL 1 Anz: 1

RCL 2 Anz: 2

Zähler und Nenner können den Speicherplätzen nicht eindeutig zugeordnet werden; deshalb muß man anhand der größeren Zahl Zähler und Nenner bestimmen. Hier: 1 = Zä, 2 = Ne also

$$\frac{1 \cdot 220}{2}$$

220 / 2 R/S Anz: (2) Anz: 2 RCL 1, Anz: 1
RCL 2, Anz: 110

also $\frac{110}{1} = 110$

Bei langer Rechenzeit ohne jede Anzeige sind die Zahlen entweder überhaupt nicht kürzbar oder nur durch große Primzahlen zu kürzen.

Bei anfänglich kurzen Anzeigen und anschließend langer Rechenzeit kann das Programm abgebrochen werden. In Speicher 1 und 2 stehen dann die gekürzten Zahlen die entweder noch durch große Zahlen kürzbar sind oder sonst sehr groß sind.

10a 10b

Kalender Wochentage, Differenz in Tagen

```

00
01 23 03 STO 3 39 13 43 GTO 43
02 22 R↓ 40 22 R↓
03 23 01 STO 1 41 23 07 STO 7
04 22 R↓ 42 23 00 STO 0
05 23 02 STO 2 43 07 7
06 03 3 44 71 +
07 24 01 RCL 1 45 15 01 g ERAC
08 14 41 f x<y 46 07 7
09 13 14 GTO 14 47 61 x
10 01 1 48 (14)74 R/S (PAUSE)
11 51 + 49 24 07 RCL 7
12 24 03 RCL 3
13 13 20 GTO 20
14 01 1
15 03 3
16 51 +
17 24 03 RCL 3
18 01 1
19 41 -
20 24 04 RCL 4
21 61 x
22 14 01 f INT
23 21 x = y
24 24 05 RCL 5
25 61 x
26 14 01 f INT
27 51 +
28 24 02 RCL 2
29 51 +
30 24 06 RCL 6
31 41 -
32 24 00 RCL 0
33 15 71 g x = 0
34 13 40 GTO 40
35 34 CLX
36 23 00 STO 0
37 22 R↓
38 23 41 07 STO - 7

```

Das Programm errechnet die Differenz zwischen zwei gegebenen Daten in Tagen. Es wird jeweils der Wochentag der Daten angegeben. 0 bedeutet Sonntag, 1=montag usw. bis 6 = Samstag

Anweisung:

1. Programm eingeben f PRGM 365,25 STO 4
30,6 STO 5 **621049** STO 6 0 STO 0
2. Tag e1 Monat e1 Jahr des späteren Datums R/
A: Code für Wochentag, n1
3. Tag e1 Monat e1 Jahr des früheren Datums R/S
A: Code für wochentag, Differenz in Tagen
für neue Daten gehe zu 2.
4. Wird der Wochentag gesucht, der vom jetzigen Datum x Tage entfernt ist, führe 2. aus und addiere zu n1 die Tagesdifferenz. jetzt
GTO 43 A: Code für wochentag, unnötiges n 1

Beispiel:

5.1.1981 ab 4.3.1953

1 für Montag, N

3 für Mittwoch 10 169 Tage differenz

Datum, daß n Tage von jetzt entfernt ist

00						
01	24	00	RCL 0	41	24	01 RCL 1
02		51	+	42		51 +
03	24	04	RCL 4	43	24	06 RCL 6
04		71	+	44		41 -
05	14	01	f INT	45		74 R/S
06	23	03	STO 3	46		41 -
07	14	73	f LAST x	47		32 CHS
08	15	01	g FRAC	48	15	61 g x \div o
09	24	07	RCL 7	49	13	01 GTO 01
10		61	x			
11	14	01	f INT			
12	23	02	STO 2			
13	14	73	f LASTx			
14	15	01	g FRAC			
15	24	05	RCL 5			
16		61	x			
17	14	01	f int			
18	23	01	STO 1			
19		03	3			
20	24	02	RCL 2			
21	14	41	f x<y			
22	13	27	GTO 27			
23		01	1			
24		51	+			
25	24	03	RCL 3			
26	13	33	GTO 33			
27		01	1			
28		03	3			
29		51	+			
30	24	03	RCL 3			
31		01	1			
32		41	-			
33	24	04	RCL 4			
34		61	x			
35	14	01	f INT			
36		21	x \Rightarrow y			
37	24	05	RCL 5			
38		61	x			
39	14	01	fINT			
40		51	+			

Gerhard Streichert

Raiffeisenstraße 1
8431 Postbauer-Heng

Speicher

- 0 n₁, -
- 1 Tag
- 2 Monat
- 3 Jahr
- 4 365,25
- 5 30,6
- 6 29
- 7 365,25/30,6

Dieses Programm ergänzt das vorangegangene. Es errechnet das Datum, daß von einem gegebenen Datum n Tage entfernt ist. n \neq Tagesdiff

Das Programm muß während der Anwendung geändert werden. Dazu hält es zunächst bei Schritt 45 an. Danach gibt man den geänderten Schritt ein und führt noch einige Tastenfunktionen von Hand aus. Soll das Programm sehr oft angewendet werden, empfiehlt es sich zunächst nur N zu berechnen, diese Zahl zu notieren und erst nachdem alle N bekannt sind das Programm zu ändern. Die Anweisung b gibt genaue Auskunft über diesen Anwendungsfall

Anweisung a

1. Programm eingeben, 365,25 STO 4, 30,6 STO 5 + STO 7 29 STO 6
2. Tag₁ STO 1, Monat₁ STO 2, Jahr₁ STO 3 GTO 19 R/S
3. A:N, Tagesdifferenz + STO 0, PRGM \leftarrow BST BST RCL 0 \rightarrow RUN, f PRGM o R/S A: o
4. RCL 1 TAG₂, RCL 2 Monat₂, RCL 3 Jahr₂
5. für neue Rechnung GTO 44, PRGM \leftarrow R/S gehe zu 2. \rightarrow RUN

Anweisung b

1. siehe a,
2. siehe a, A: N notiere dieses um Tagesdiff erhöhte Ergebnis und wiederhole 1. und 2. bis alle Daten durch
3. GTO 44 PRGM \leftarrow RCL 0 \rightarrow RUN f PRGM 3I gib die notierte N ein STO 0, o R/S
4. siehe a, gehe zu 3I bis alle Zahlen durch.
5. für neue Rechnung gehe zu 5a

Formelzeichen:

- Tag₁ Monat₁ Jahr₁ bekanntes Datum von dem ausgegangen wird
- Tag₂ usw Unbekanntes Datum das errechnet wird
- N Nummer das Tages
- Tagesdifferenz kann positiv oder negativ eingegeben werden, je nachdem, ob in die Zukunft oder Vergangenheit gerechnet werden soll.
- 4. 3.1953 N= 71343o
- 31.12.1999 N= 73o533 wird als 0.1.2000 angezeigt

17

Simonsche Regel für numerische Integration

```

00
01 23 06 STO 6      46 24 07 RCL 7
02 21 x==y          47 02 2
03 23 01 STD 1      48 71 +
04 41 -             49 41 -
05 21 x==y
06 71 +
07 23 02 STO 2
08 03 3
09 71 +
10 13 03 STO 3
11 23 51 06 STO + 6
12 24 01 RCL 1
13 (14 04) f(x)
:
:
:
:
13 28 GTO 28
28 24 03 RCL 3
29 61 x
30 24 04/74 R/S RCL 4
31 61 x
32 23 07 STO 7
33 23 51 00 STO + 0
34 24 02 RCL 2
35 23 51 01 STO + 1
36 24 01 RCL 1
37 24 06 RCL 6
38 14 41 f x<y
39 13 45 GTO 45
40 24 05 RCL 5
41 15 22 g 1/x
42 23 05 STO 5
43 23 71 04 STO + 4
44 13 12 GTO 12
45 24 00 RCL 0
    
```

Simonsche Regel für numerische Integration

Anweisung

1. Programm (schwarz) eingeben; 2 STO 4 STO 5
2. f(x) ab 12 bis 27 eingeben; x steht im Speicher 1 zur Verfügung. Für Zwischenergebnisse kann Sp. 7 dienen jedoch nicht für Konstante die während des ganzen Programms gebraucht werden. Falls nicht alle Schritte bis 27 gebraucht werden, sprünge am Schluß von f(x) nach 28.
3. 0 STO 0 Schrittzahl (nur gerade) | xmin | xmax (R/S
4. Prgm ← BST BST RCL 4 SST SST → RUN R/S
5. A: Ergebnis
6. für neue Rechnung GTO 29 PRGM ← R/S → RUN
f PRGM gehe zu 2. oder 3.

Wenn f(x) = 0 dann kann sofort mit dem roten Programm gearbeitet werden. Es entfällt dann auch 4. und zur Vorbereitung 6.

Reicht der Speicherplatz nicht aus, kann zunächst der Programmteil ab 45 weggelassen werden und dann schließlich der Teil bis 11 eingespart werden. Diese Teile müssen natürlich dann von Hand ausgeführt werden. Achte auf die geänderten Sprungadressen.

Adress	Code	Operand	Operand	Operand
00				
01	23	06	STO	6
02	21	x==y		
03	23	01	STD	1
04	41	-		
05	21	x==y		
06	71	+		
07	23	02	STO	2
08	03	3		
09	71	+		
10	13	03	STO	3
11	23	51 06	STO +	6
12	24	01	RCL	1
13	(14	04)	f(x)	
:	:	:	:	:
:	:	:	:	:
:	:	:	:	:
13	28	GTO	28	
28	24	03	RCL	3
29	61	x		
30	24	04/74	R/S RCL	4
31	61	x		
32	23	07	STO	7
33	23	51 00	STO +	0
34	24	02	RCL	2
35	23	51 01	STO +	1
36	24	01	RCL	1
37	24	06	RCL	6
38	14	41	f x<y	
39	13	45	GTO	45
40	24	05	RCL	5
41	15	22	g 1/x	
42	23	05	STO	5
43	23	71 04	STO +	4
44	13	12	GTO	12
45	24	00	RCL	0

12

Normiertes Zeichnen von Funktionen
(wenn y_{\min} und y_{\max} bekannt)

00				
01	23	03	STO 3	Speicher
02		41	-	R0 dx
03		33	EEX	R1 $100/y_{\max} - y_{\min} =$
04		02	2	faktor
05		21	$x \leq y$	R2 x_{\min}, x
06		71	+	
07	23	01	STO 1	R3 y_{\min} , Subtrahend
08		74	R/S	ab R4 frei für f(x)
09	23	02	STO 2	verwendbar
10		41	-	
11		21	$x \leq y$	Anweisung
12		71	+	
13	23	00	STO 0	
14	24	01	RCL 1	
15	24	03	RCL 3	
16		61	x	
17		73	.	
18		05	5	
19		41	-	
20	23	03	STO 3	
21	24	02	RCL 2	
22			f(x)	
.			.	
.			.	
.			.	
	13	41	GTO 41	
41	24	01	RCL 1	
42		61	x	
43	24	03	RCL 3	
44		41	-	
45	14	01	f INT	
46		74	R/S	
47	24	00	RCL 0	
48	23	51	STO + 2	
49	13	21	GTO 21	

12

Normiertes Zeichnen von Funktionen
(wenn y_{\min} und y_{\max} bekannt)

Das Programm errechnet von einer gegebenen Funktion n Punkte, die es auf Werte zwischen 0 und 100 normiert. Dabei muß y_{\min} und y_{\max} vorher bekannt sein. Ist das nicht der Fall, wähle nächstes Programm.

Anweisung

1. Programm eingeben, dabei von 22 bis 40 die Funktion programmieren. x steht dabei im x-Register, f(x) soll danach ebenfalls im x-Register stehen. Die Speicher ab 5 stehen zur Verfügung. GTO 41
2. y_{\max} , y_{\min} , R/S
3. Anzahl der Punkte \uparrow x_{\max} , x_{\min} , R/S
4. Anzeige: f(x) normiert, zeichne Punkt zwischen Null und Hundert. gehe R/S gehe zu 4. bis alle Punkte gezeichnet sind.

14

Primzahlen tabellarisch

00						
01	31	Enter	40	23	06	STO 6
02	03	3	41	13	20	GTO 20
03	14	41	42		74	R/S
04	22	R↓	43	24	00	RCL 0
05	23	07	44	24	07	RCL 7
06	03	3	45	14	33	f REG
07	23	00	46	23	07	STO 7
08	13	11	47		22	R↓
09		02	48	23	00	STO 0
10	23	51	49	13	20	GTO 20
11		03				
12	24	07				RCL 7
13	24	00				RCL 0
14		71				+
15	14	41				f x<y
16	13	23				GTO 23
17	15	01				g FRAC
18	15	61				g x≠0
19	13	09				GTO 09
20		02				2
21	23	51				STO + 7
22	13	06				GTO 06
23		\$				siehe Anweis.
24	14	74				f PAUSE
25	13	20				GTO 20
26	24	01				RCL 1
27	15	61				g x≠0
28	13	42				GTO 42
29	24	02				RCL 2
30	23	01				STO 1
31	24	03				STO RCL 3
32	23	02				STO 2
33	24	04				RCL 4
34	23	03				STO 3
35	24	05				RCL 5
36	23	04				STO 4
37	24	06				RCL 6
38	23	05				STO 5
39	24	07				RCL 7

Das Programm ermittelt alle Primzahlen mit Ausnahme von 2. Es beginnt entweder bei 3 oder bei der eingegebenen ungeraden Zahl.

Das Programm zeigt entweder die Primzahlen in laufender Folge kurz an, oder es ermittelt 7 aufeinanderfolgende Primzahlen die dann in den Speichern 1 bis 7 abzufufen sind. Bei Start fährt es dann mit den nächsten 7 Primzahlen fort.

Bei großen Primzahlen dauert das Programm recht lange

Anweisung

1. Programm eingeben Schritt 23 entweder RCL 7
2. n R/S n=kleinste zu untersuchende Zahl (ungerade)
3. Anzeige: Primzahlen in laufender Folge oder:
 1. Programm eingeben Schritt 23 GTO 26
 2. wie oben
 3. Anzeige: nächste Primzahlen; Die nächsten 7 Primzahlen sind aus den Speichern 1 bis 7 abrufbar R/S Rechner geht zu 3.

Reihen für Hyperbolische Funktionen

15

00					
01	14	71	f x=y	41	23 07 STO 7
02	13	12	GTO 12	42	13 19 GTO 19
03	15	02	g x ²	43	24 00 RCL 0
04	23	03	STO 3	44	13 00 GTO 00
05		01	1		
06	23	00	STO 0		
07	23	02	STO 2		
08	23	04	STO 4		
09		00	0		
10	23	01	STO 1		
11	13	19	GTO 19		
12	23	00	STO 0		
13	23	04	STO 4		
14	15	02	g x ²		
15	23	03	STO 3		
16		01	1		
17	23	01	STO 1		
18	23	02	STO 2		
19		02	2		
20	23	51	01 STO + 1		
21	24	01	RCL 1		
22	24	01	RCL 1		
23		01	1		
24		41	-		
25		61	x		
26	23	61	02 STO x 2		
27	24	03	RCL 3		
28	23	61	04 STO x 4		
29	24	04	RCL 4		
30	24	02	RCL 2		
31		71	+		
32	23	51	00 STO + 0		
33	24	07	RCL 7		
34	24	00	RCL 0		
35	14	74	f PAUSE		
36	24	06	RCL 6		
37		71	+		
38	14	01	fINT		
39	14	71	f x=y		
40	13	43	GTO 43		

- Anweisung
1. Programm eingeben f PRGM
 2. Genauigkeit zB ,001 STO 6
 3. für sinh (x)
x ↗ R/S A: sinh (x)
 4. für cosh (x)
x R/S A: cosh (x)
 5. für neue Rechnung gehe zu 2.,3.,oder 4.

12

Reihen für Hyperbolische Funktionen

00					
01	14	71	f x=y	41	23 07 STO 7
02	13	12	GTO 12	42	13 19 GTO 19
03	15	02	g x ²	43	24 00 RCL 0
04	23	03	STO 3	44	13 00 GTO 00
05		01	1		
06	23	00	STO 0		
07	23	02	STO 2		
08	23	04	STO 4		
09		00	0		
10	23	01	STO 1		
11	13	19	GTO 19		
12	23	00	STO 0		
13	23	04	STO 4		
14	15	02	g x ²		
15	23	03	STO 3		
16		01	1		
17	23	01	STO 1		
18	23	02	STO 2		
19		02	2		
20	23	51	01 STO + 1		
21	24	01	RCL 1		
22	24	01	RCL 1		
23		01	1		
24		41	-		
25		61	x		
26	23	61	02 STO x 2		
27	24	03	RCL 3		
28	23	61	04 STO x 4		
29	24	04	RCL 4		
30	24	02	RCL 2		
31		71	+		
32	23	51	00 STO + 0		
33	24	07	RCL 7		
34	24	00	RCL 0		
35	14	74	f PAUSE		
36	24	06	RCL 6		
37		71	+		
38	14	01	fINT		
39	14	71	f x=y		
40	13	43	GTO 43		

- Anweisung
1. Programm eingeben f PRGM
 2. Genauigkeit zB ,001 STO 6
 3. für sinh (x)
x ↗ R/S A: sinh (x)
 4. für cosh (x)
x R/S A: cosh (x)
 5. für neue Rechnung gehe zu 2.,3.,oder 4.

16

Anweisung

```

00
01 01 1 41 14 01 F INT
02 23 51 00 STO + 0 42 21 x<y
03 24 00 RCL 0 43 24 05 RCL 5
04 14 74 f PAUSE 44 61 x
05 24 06 RCL 6 45 14 01 F INT
06 71 + 46 51 +
07 15 01 g FRAC 47 24 02 RCL 2
08 02 2 48 51 +
09 15 73 49 13 19 GTO 19
10 61 x
11 61 x
12 14 04 f sin
13 13 00 GTO 00
14 05 5
15 23 51 06 STO + 6
16 24 07 RCL 7
17 23 00 STO 0
18 13 01 GTO 01
19 74 R/S
20 23 03 STO 3
21 22 R↓
22 23 01 STO 1
23 22 R↓
24 23 02 STO 2
25 03 3
26 24 01 RCL 1
27 14 41 f x<y
28 13 33 GTO 33
29 01 1
30 51 +
31 24 03 RCL 3
32 13 39 GTO 39
33 01 1
34 03 3
35 51 +
36 24 03 RCL 3
37 01 1
38 41 -
39 24 04 RCL 4
40 61 x

```

Speicher

- 0 akt. Tageszahl
- 1 Monat
- 2 Tag
- 3 Jahr
- 4 365,25
- 5 0,6
- 6 23 29 33
- 7 erster Tag

Anweisung

1. Programm eingeben, 365,25 STO 4, 30,6 STO 5, g RAL
2. 23 STO 6 GTO 20
 späteres Datum: Tag ↑ Monat ↑ Jahr R/S Anzeige: N STO 7
 Geburtsdatum: Tag ↑ Monat ↑ Jahr R/S A: N' STO - 7
 GTO 16
3. R/S (A): Tagesnr. A: M-zyklus von -1,2,+.1
4. R/S für nächsten Tag wiederhole 3 beliebig oft
5. für W-Zyklus GTO 14 R/S siehe 3. mit W-Zyklus
6. für I-Zyklus GTO 14 R/S siehe 3. mit I-Zyklus
7. für neue Rechnung siehe 2.

Speicher

- 0 akt. Tageszahl
- 1 Monat
- 2 Tag
- 3 Jahr
- 4 365,25
- 5 0,6
- 6 23 29 33
- 7 erster Tag

17

Sammelbesteller-Rechnung

00					
01	15	75	g NOP(x)	40	74 R/S
02	23	51 01	STO + 1	41	24 05 RCL 5
03		24 01	RCL 1	42	14 74 f PAUSE
04		13 00	GTO 00	43	13 00 GTO 00
05		23 02	STO 2		
06		24 01	RCL 1		
07		05 5			
08	15	21	g %		
09		74	R/S		
10		41	-		
11	23	00	STO 0		
12	24	01	RCL 1		
13	24	02	RCL 2		
14	15	61	g x≠0		
15	13	19	GTO 19		
16		00	o		
17	23	01	STO 1		
18	13	00	GTO 00		
19		02	2		
20		71	+		
21	14	01	f INT		
22	15	21	g %		
23		74	R/S		
24	23	51 00	STO + o		
25		00	o		
26	23	01	STO 1		
27	24	00	RCL 0		
28		74	R/S		
29	24	00	RCL 0		
30	24	02	RCL 2		
31		71	+		
32	14	01	f INT		
33	23	05	STO 5		
34	24	02	RCL 2		
35		61	x		
36		41	-		
37	24	05	RCL 5		
38		51	+		
39	23	04	STO 4		

Speicher

- o Rechnungsendwert
- 1 Warenwert
- 2 Ratenzahl
- 3 frei
- 4 1. Rate
- 5 Folge-Raten

Gerhard Streichert

Raiffeisenstraße 1
8431 Postbauer-Heng

Anweisung

1. Programm eingeben, f PRGM
 2. Gesamtpreis R/S A: Teilsumme Gehe zu 2.
 3. GTO 05, Ratenzahl, R/S
 4. A: Unkostenvergütung R/S bei 0 Raten g.z. 5a
 5. A: Kreditspesen R/S
 6. A: Rechnungsendwert
 7. A: 1. Rate
 8. A:(Folgeraten), Folgeraten gehe zu 2. für neue Rechnung
- 5a, A: Rechnungsendwert, gehe zu 2.
- Wurden mehrere Artikel derselben Sorte bestellt:
ändere Schritt 01 zu 01 61 x
dann gilt
2. Einzelpreis Anzahl R/S A: Teilsumme gehe zu 2.

Beispiel:

3 Artikel á 5,23 DM	31 5.23 R/S A: 15,69
5 Artikel á 23,45 DM	51 23.45 R/S A: 117,25
	<u>132,94</u>
GTO 05	
3 Raten	GTO 05 3 R/S
Unkostenverg. 6,65 DM	R/S
Kreditspesen 1.33 DM	R/S
Rechnungsendwert 127.62 DM	R/S
1. Rate 43.62	R/S
Folgeraten 42.00	

Finanzprogramme (Übersicht)

Im Programmbuch des hp-25 sind einige Programme zum Finanzwesen angegeben. Nachfolgende Übersicht dient zum schnellen Auffinden und Anwenden des jeweils richtigen Programms.

Die Belegung der Speicher ist weitestgehend vereinheitlicht. Deshalb zunächst diese Tabelle: Speicher Formelz. Erklärung, Ausnahme bei Prog.

0	frei	*J	1
1	n	Anzahl der Perioden	K
2	i	Zinssatz pro Periode (dezimal)	
3	PMT	Periodische Zahlung	
4	PV	Anfangskapital (-schuld)	
5	FV	Endkapital(-schuld)	

6 PV/PMT bei Prg. 3
7 $(1+j)^{-1}$ bei Prg. 1 und 3

Speicherinhalte 0 bis 5 werden eingegeben, 6 und 7 werden vom Prg. berechnet.

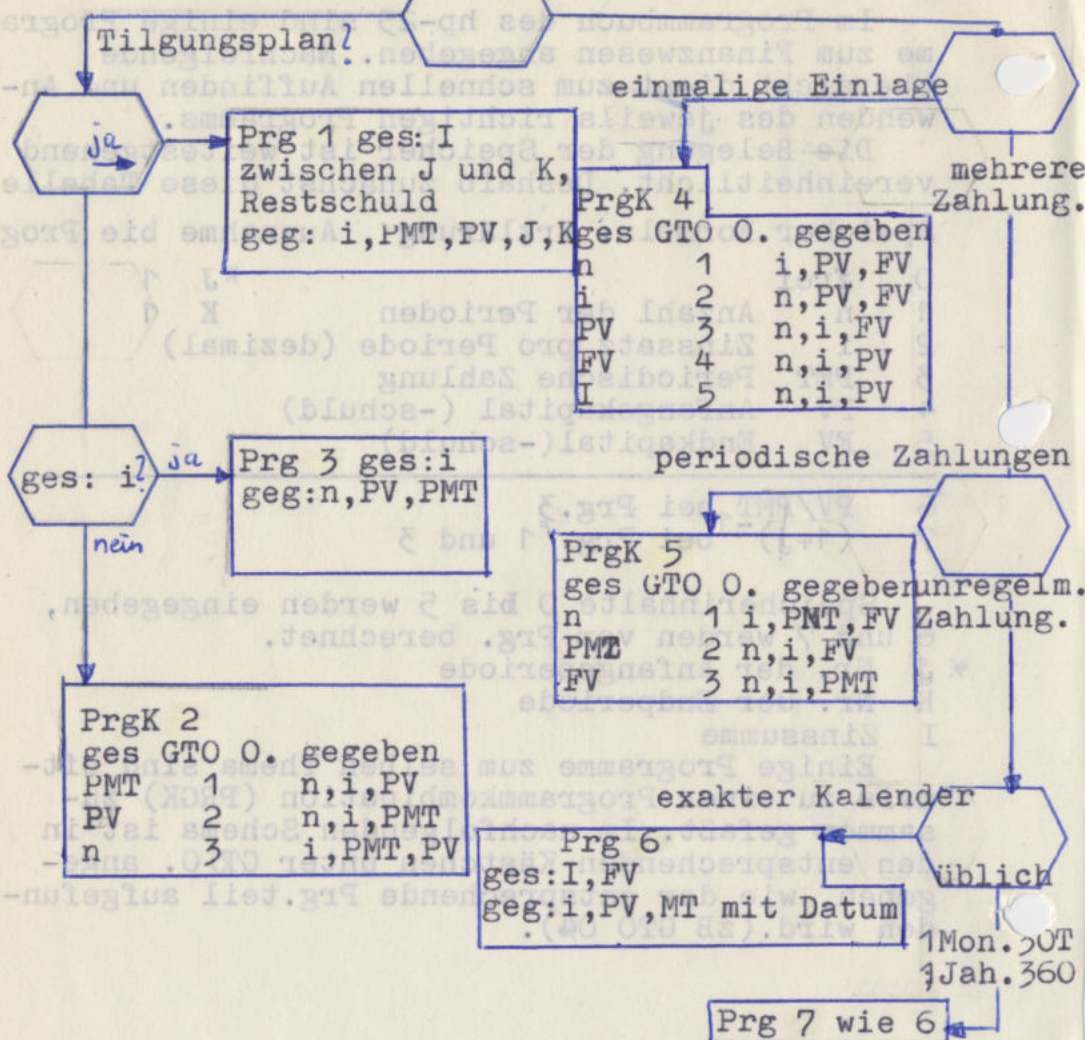
- * J Nr. der Anfangsperiode
- K Nr. der Endperiode
- I Zinssumme

Einige Programme zum selben Thema sind oftmals zu einer Programmkombination (PRGK) zusammen gefaßt. Im nachfolgenden Schema ist in den entsprechenden Kästchen unter GTO 0. angegeben, wie der entsprechende Prg.teil aufgefunden wird. (zB GTO 0%).

Gerhard Streichert
Raiffeisenstraße 1
8431 Postbauer-Heng

Finanzprogramme

Darlehen Guthaben



Gerhard Streichert
Raiffeisenstraße 1
8431 Postbauer-Heng

Erfolgssparen 6,25 Jahre sparen; 0,75 Jahre liegen lassen
14% Prämie auf Sparanteil

Formel:

$$PMT(\text{jährlich}) = \frac{FV}{\frac{1}{i} [(1+i)^{6,5} - (1+i)^{0,25}] + 0,275}$$

Prg 1

ges: I zwischen J und K, Restschuld
geg: i, PMT, PV, J, K

oo	2	3	4	o	1	Speicher			
00									
01	24	02	RCL	2	40	24	00	RCL	0
02		01	1		41		41	-	
03		51	+		42		01	1	
04	24	01	RCL	1	43		51	+	
05		32	CHS		44	24	03	RCL	3
06	14	03	f y ^x		45		61	x	
07	23	07	STO	7	46		51	+	
08		01	1		47		74	R/S	
09		41	-		48		21	x=y	
10	24	02	RCL	2	49	13	00	GTO	00
11		71	+						
12	24	03	RCL	3					
13		61	x						
14	24	04	RCL	4					
15		51	+						
16	24	07	RCL	7					
17		71	+						
18	23	06	STO	6					
19	24	02	RCL	2					
20		01	1						
21		51	+						
22	24	00	RCL	0					
23		01	1						
24		41	-						
25		32	CHS						
26	14	03	f y ^x						
27	23	07	STO	7					
28		01	1						
29		41	-						
30	24	02	RCL	2					
31		71	+						
32	24	03	RCL	3					
33		61	x						
34	24	04	RCL	4					
35		51	+						
36	24	07	RCL	7					
37		71	+						
38		41	-						
39	24	01	RCL	1					

Anweisung

1. Programm eing. f PRGM
2. Daten nach obiger Tabelle eingeben R/S
3. Anzeige I Zinssumme R/S
4. Anzeige FV Restschuld
5. für neue Rechnung gehe zu 2.

Gerhard Streichert
Raiffeisenstraße 1
8431 Postbauer-Heng

18
18

Prg 1

ges: I zwischen J und K, Restschuld
geg: i, PMT, PV, J, K

oo	2	3	4	o	1	Speicher			
00									
01	24	02	RCL	2	40	24	00	RCL	0
02		01	1		41		41	-	
03		51	+		42		01	1	
04	24	01	RCL	1	43		51	+	
05		32	CHS		44	24	03	RCL	3
06	14	03	f y ^x		45		61	x	
07	23	07	STO	7	46		51	+	
08		01	1		47		74	R/S	
09		41	-		48		21	x=y	
10	24	02	RCL	2	49	13	00	GTO	00
11		71	+						
12	24	03	RCL	3					
13		61	x						
14	24	04	RCL	4					
15		51	+						
16	24	07	RCL	7					
17		71	+						
18	23	06	STO	6					
19	24	02	RCL	2					
20		01	1						
21		51	+						
22	24	00	RCL	0					
23		01	1						
24		41	-						
25		32	CHS						
26	14	03	f y ^x						
27	23	07	STO	7					
28		01	1						
29		41	-						
30	24	02	RCL	2					
31		71	+						
32	24	03	RCL	3					
33		61	x						
34	24	04	RCL	4					
35		51	+						
36	24	07	RCL	7					
37		71	+						
38		41	-						
39	24	01	RCL	1					

- Anweisung
1. Programm eing. f PRGM
 2. Daten nach obiger Tabelle eingeben R/S
 3. Anzeige I Zinssumme R/S
 4. Anzeige FV Restschuld
 5. für neue Rechnung gehen zu 2.

Gerhard Streichert
Raiffeisenstraße 1
8431 Postbauer-Heng

Prg 2

ges PMT 1
 PV 2
 n 3

GTO 0.
 1
 2
 3

gegeben
 n,i,PV
 n,i,PMZ
 i,PMT,PV

Speicher
 1 n
 2 i
 3 PMT
 4 PV

00
 01 13 04 GTO 04
 02 13 18 GTO 18
 03 13 31 GTO 31
 04 01 1
 05 24 02 RCL 2
 06 01 1
 07 51 +
 08 24 01 RCL 1
 09 32 CHS
 10 14 03 f y^x
 11 41 -
 12 24 02 RCL 2
 13 21 x=y
 14 71 ÷
 15 24 04 RCL 4
 16 61 x
 17 13 00 GTO 00
 18 01 1
 19 24 02 RCL 2
 20 01 1
 21 51 +
 22 24 01 RCL 1
 23 32 CHS
 24 14 03 f y^x
 25 41 -
 26 24 02 RCL 2
 27 71 ÷
 28 24 03 RCL 3
 29 61 x
 30 13 00 GTO 00
 31 01 1
 32 24 04 RCL 4
 33 24 03 RCL 3
 34 71 ÷
 35 24 02 RCL 2

- Anweisung
 1. Programm eingeben f PRGM
 2. Geg Werte nach obiger Tabelle in Speicher eing
 3. Je nach gesuchtem Wert Gto o. ausführen R/S
 4. Anzeige: ges Wert
 5. für neue Rechnung gehe zu 2. oder 3.

Gerhard Streichert
 Raiffeisenstraße 1
 8431 Postbauer-Hang

18

18

Prg 2

Speicher
 1 n
 2 i
 3 PMT
 4 PV

GTO 0.
 1
 2
 3

gegeben
 n,i,PV
 n,i,PMZ
 i,PMT,PV

ges
 PMT
 PV
 n

00
 01 13 04 GTO 04
 02 13 18 GTO 18
 03 13 31 GTO 31
 04 01 1
 05 24 02 RCL 2
 06 01 1
 07 51 +
 08 24 01 RCL 1
 09 32 CHS
 10 14 03 f y^x
 11 41 -
 12 24 02 RCL 2
 13 21 x=y
 14 71 ÷
 15 24 04 RCL 4
 16 61 x
 17 13 00 GTO 00
 18 01 1
 19 24 02 RCL 2
 20 01 1
 21 51 +
 22 24 01 RCL 1
 23 32 CHS
 24 14 03 f y^x
 25 41 -
 26 24 02 RCL 2
 27 71 ÷
 28 24 03 RCL 3
 29 61 x
 30 13 00 GTO 00
 31 01 1
 32 24 04 RCL 4
 33 24 03 RCL 3
 34 71 ÷
 35 24 02 RCL 2

Gerhard Streichert
 Raiffeisenstraße 1
 8431 Postbauer-Hang

```

ges i
geg n, PV, PMT

00
01 24 0# RCL 4
02 24 03 RCL 3
03 71 ÷
04 23 05 STO 6
05 24 06 RCL 6
06 31 Enter
07 15 22 g 1/x
08 21 x==y
09 24 p1 RCL 1
10 15 02 g x2
11 71 +
12 41 -
13 23 02 STO 2
14 24 06 RCL 6
15 24 02 RCL 2
16 61 x
17 01 1
18 24 02 RCL 2
19 01 1
20 51 +
21 24 01 RCL 1
22 32 CHS
23 14 03 f yx
24 23 07 STO 7
25 41 -
26 41 -
27 24 01 RCL 1
28 24 02 RCL 2
29 15 22 g 1/x
30 01 1
31 51 +
32 71 +
33 01 1
34 51 +
35 24 07 RCL 7
36 61 x
37 01 1
38 41 -
39 24 02 RCL 2

```

Speicher

```

0 frei
1 n
2 i
3 PMT
4 PV
5 frei
6 PV/PMT
7 (1+j)-1

40 71 +
41 71 +
42 23 51 02 STO + 2
43 14 03 g ABS
44 15 03 g ABS
44 33 EEX
45 06 6
46 32 CHS
47 14 41 f x<y
48 13 14 GTO 14
49 24 02 RCL 2

```

Anweisung

1. Programm eingeben
f PRGM
2. Geg. Werte nach
obiger Tabelle eing.
3. R/S
4. Anzeige: ges Wert
in dezimal
5. EEX 2 x A: i in %
6. ev 12 x für i in %
pro Jahr

```

Speicher

0 frei
1 n
2 i
3 PMT
4 PV
5 frei
6 PV/PMT
7 (1+j)-1

40 71 +
41 71 +
42 23 51 02 STO + 2
43 14 03 g ABS
44 15 03 g ABS
44 33 EEX
45 06 6
46 32 CHS
47 14 41 f x<y
48 13 14 GTO 14
49 24 02 RCL 2

```

Speicher

```

0 frei
1 n
2 i
3 PMT
4 PV
5 frei
6 PV/PMT
7 (1+j)-1

40 71 +
41 71 +
42 23 51 02 STO + 2
43 14 03 g ABS
44 15 03 g ABS
44 33 EEX
45 06 6
46 32 CHS
47 14 41 f x<y
48 13 14 GTO 14
49 24 02 RCL 2

```

Anweisung

1. Programm eingeben
f PRGM
2. Geg. Werte nach
obiger Tabelle eing.
3. R/S
4. Anzeige: ges Wert
in dezimal
5. EEX 2 x A: i in %
6. ev 12 x für i in %
pro Jahr

```

00
01 24 0# RCL 4
02 24 03 RCL 3
03 71 ÷
04 23 05 STO 6
05 24 06 RCL 6
06 31 Enter
07 15 22 g 1/x
08 21 x==y
09 24 p1 RCL 1
10 15 02 g x2
11 71 +
12 41 -
13 23 02 STO 2
14 24 06 RCL 6
15 24 02 RCL 2
16 61 x
17 01 1
18 24 02 RCL 2
19 01 1
20 51 +
21 24 01 RCL 1
22 32 CHS
23 14 03 f yx
24 23 07 STO 7
25 41 -
26 41 -
27 24 01 RCL 1
28 24 02 RCL 2
29 15 22 g 1/x
30 01 1
31 51 +
32 71 +
33 01 1
34 51 +
35 24 07 RCL 7
36 61 x
37 01 1
38 41 -
39 24 02 RCL 2

```

ges	GTO	O.	PrgK 4 gegeben	Speicher
n	1		i, PV, FV	1 n
i	2		n, PV, FV	2 i
PV	3		n, i, FV	3
FV	4		n, i, PV	4 PV
I	5		n, i, PV	5 FV

oo				
o1	13	15	GTO 15	31 01 1
o2	13	25	GTO 25	32 41 -
o3	13	34	GTO 34	33 13 00 GTO 00
o4	13	43	GTO 43	34 24 02 RCL 2
o5	24	02	RCL 2	35 01 1
o6		01	1	36 51 +
o7		51	+	37 24 01 RCL 1
o8	24	01	RCL 1	38 32 CHS
o9	14	03	f y ^x	39 14 03 f y ^x
10		01	1	40 24 05 RCL 5
11		41	-	41 61 x
12	24	04	RCL 4	42 13 00 GTO 00
13		61	x	43 24 02 RCL 2
14	13	00	GTO 00	44 01 1
15	24	05	RCL 5	45 51 +
16	24	04	RCL 4	46 24 01 RCL 1
17		71	+	47 14 03 f y ^x
18	14	07	f ln	48 24 04 RCL 4
19	24	02	RCL 2	49 61 x
20		01	1	
21		51	+	
22	14	07	f ln	
23		71	+	
24	13	00	GTO 00	
25	24	05	RCL 5	
26	24	04	RCL 4	
27	24	01	RCL 1	
28	24	01	RCL 1	
29	15	22	g 1/x	
30	14	03	f y ^x	

- Anweisung
1. Programm eing. f PRC
 2. je nach ges. Größe in Tabelle GTO o.
 3. gegeb. Größen nach Tabelle eingeb R/S
 4. Anzeige: ges. Größe
 5. für neue Rechnung gehe zu 2. oder 3.

Gerhard Streichert
Raiffeisenstraße 1
8431 Postbauer-Heng

ges	GTO	O.	PrgK 4 gegeben	Speicher
n	1		i, PV, FV	1 n
i	2		n, PV, FV	2 i
PV	3		n, i, FV	3
FV	4		n, i, PV	4 PV
I	5		n, i, PV	5 FV

oo				
o1	13	15	GTO 15	15 15
o2	13	25	GTO 25	
o3	13	34	GTO 34	
o4	13	43	GTO 43	
o5	24	02	RCL 2	
o6		01	1	
o7		51	+	
o8	24	01	RCL 1	
o9	14	03	f y ^x	
10		01	1	
11		41	-	
12	24	04	RCL 4	
13		61	x	
14	13	00	GTO 00	
15	24	05	RCL 5	
16	24	04	RCL 4	
17		71	+	
18	14	07	f ln	
19	24	02	RCL 2	
20		01	1	
21		51	+	
22	14	07	f ln	
23		71	+	
24	13	00	GTO 00	
25	24	05	RCL 5	
26	24	04	RCL 4	
27	24	01	RCL 1	
28	24	01	RCL 1	
29	15	22	g 1/x	
30	14	03	f y ^x	

Gerhard Streichert
Raiffeisenstraße 1
8431 Postbauer-Heng

ges GTO 0. gegeben
 n 1 i, PMT, FV
 PMT 2 n, i, FV
 FV 3 n, i, PMT

Speicher

1 n
 2 i
 3 PMT
 4 PV
 5 FV

00
 01 13 04 GTO 04
 02 13 21 GTO 21
 03 13 35 GTO 35
 04 24 02 RCL 2
 05 24 05 RCL 5
 06 61 x
 07 24 03 RCL 3
 08 71 +
 09 24 02 RCL 2
 10 01 1
 11 51 +
 12 23 00 STO 0
 13 51 +
 14 14 07 f ln
 15 24 00 RCL 0
 16 14 07 f ln
 17 71 +
 18 01 1
 19 41 -
 20 13 00 GTO 00
 21 24 05 RCL 5
 22 24 02 RCL 2
 23 61 * x
 24 24 02 RCL 2
 25 01 1
 26 51 +
 27 71 +
 28 14 73 f Last x
 29 24 01 RCL 1
 30 14 03 f y^x
 31 01 1
 32 41 -
 33 71 +
 34 13 00 GTO 00
 35 24 03 RCL 3
 36 24 02 RCL 2
 37 01 1
 38 51 +

39 61 x
 40 14 73 f Last x
 41 24 01 RCL 1
 42 14 03 f y^x
 43 01 1
 44 41 -
 45 61 x
 46 24 02 RCL 2
 47 71 +
 48 13 00 GTO 00
 49

Anweisung

1. Programm eing. f PRGM
2. Geg. Werte nach obiger Tabelle in Speicher eingeben
3. Je nach gesuchtem Wert GTO 0. ausführen R/S
4. Anzeige: ges. Wert
5. für neue Rechnung gehe zu 2. oder 3.

Speicher
 1 n
 2 i
 3 PMT
 4 PV
 5 FV

61 x
 73 f Last x
 01 RCL 1
 03 f y^x
 01 1
 41 -
 61 x
 02 RCL 2
 71 +
 00 GTO 00

- Anweisung
1. Programm eing. f PRGM
 2. Geg. Werte nach obiger Tabelle in Speicher eingeben
 3. Je nach gesuchtem Wert GTO 0. ausführen R/S
 4. Anzeige: ges. Wert
 5. für neue Rechnung gehe zu 2. oder 3.

ges GTO 0. gegeben
 n 1 i, PMT, FV
 PMT 2 n, i, FV
 FV 3 n, i, PMT

00
 01 13 04 GTO 04
 02 13 21 GTO 21
 03 13 35 GTO 35
 04 24 02 RCL 2
 05 24 05 RCL 5
 06 61 x
 07 24 03 RCL 3
 08 71 +
 09 24 02 RCL 2
 10 01 1
 11 51 +
 12 23 00 STO 0
 13 51 +
 14 14 07 f ln
 15 24 00 RCL 0
 16 14 07 f ln
 17 71 +
 18 01 1
 19 41 -
 20 13 00 GTO 00
 21 24 05 RCL 5
 22 24 02 RCL 2
 23 61 * x
 24 24 02 RCL 2
 25 01 1
 26 51 +
 27 71 +
 28 14 73 f Last x
 29 24 01 RCL 1
 30 14 03 f y^x
 31 01 1
 32 41 -
 33 71 +
 34 13 00 GTO 00
 35 24 03 RCL 3
 36 24 02 RCL 2
 37 01 1
 38 51 +

ges I im Laufe eines Jahres
geg i, PV, Datum der Kontobewegung

```

00
01 23 05 STO 5 40 71 +
02 22 R↓ 41 24 02 RCL 2
03 23 07 STO 7 42 61 x
04 22 R↓ 43 24 05 RCL 5
05 21 x==y 44 14 01 f INT
06 23 06 STO 6 45 61 x
07 22 R↓ 46 14 11 02 f FIX 2
08 03 3 47 14 74 f PAUSE
09 21 x==y 48 23 51 00 STO + 0
10 14 41 f x<y 49 24 00 RCL 0
11 13 16 GTO 16
12 01 1
13 51 +
14 24 07 RCL 7
15 13 22 GTO 22
16 01 1
17 03 3 Speicher
18 51 +
19 24 p7 RCL 7 0 I Zinssumme
20 01 1 1 Nalt Tagesnummer
21 41 - 2 i Zinssatz dezimal
22 24 04 RCL 4 3 30,6
23 61 x 4 365,25
24 14 01 f INT 5 FV maßgebendes Kap.
25 21 x==y 6 Tag
26 24 03 RCL 3 7 Jahr
27 61 x
28 14 01 f INT
29 51 + Beispiel: 1977STO7
30 24 06 RCL 6 i = 0,03 STO 2
31 51 + 1.1.0 R/S
32 24 01 RCL 1 A: N 0,0
33 21 x==y 5.6.1235,67 R/S
34 23 01 STO 1 A: 155, 15,72 15,72
35 21 x==y 7.9.123 R/S
36 41 - A: 94 0,9516,67
37 14 11 00 f FIX 0 31.12 500 R/S
38 14 74 f PAUSE A: 118 4,72 21,40
39 24 04 RCL 4

```

Das Programm errechnet (ähnlich Rrg 7) die Zinsen, die im Laufe eines Jahres bei unregelmäßigen Ein- und Auszahlungen im Sparbuch anfallen. Falls die Zinsen des Vorjahres nicht gleich gutgeschrieben werden konnten, diese schon zu Beginn des Jahres dem jeweiligen Kapital zurechnen.

Nach der Vorbereitung gibt man den Zinssatz als Dezimalzahl ein STO 2.

Das Programm funktioniert zwar auch über einen Jahreswechsel hinweg, es ist aber üblich jeweils nur innerhalb eines Jahres zu rechnen. In diesem Fall werden die Programmschritte 02 und 03 mit g NOP beschrieben, und die Jahreszahl zu Beginn in Speicher 7 eingeschrieben.

Jetzt gibt man Tag und Monat des Anfangs des Zeitraumes (meist 1.1.) ein und anschließend Null. Damit sind alle Vorbereitungen abgeschlossen und man beginnt mit der Routine:

Nächstes Datum eingeben, und Kapital für den Zeitraum vom letzten Datum zu diesem Datum. Dieses Kapital steht im Sparbuch jeweils eine Zeile höher als das zugehörige Datum.

Der Rechner gibt nun die Tagesdifferenz zum letzten Datum an und danach die Zinsen für diesen Zeitraum, schließlich die bisher aufgelaufene Zinssumme, die am Schluß gleichzeitig das Endergebnis darstellt.

Bei Fehlerhafter Eingabe, die erst nach R/S bemerkt wird: Vorheriges richtiges Datum eingeben RCL 5 R/S Somit behebt der Rechner die letzte falsche Eingabe.

Bei Zinssatzwechsel bis zu diesem Datum rechnen, dann neuen Zinssatz in STO2 eingeben und weiterrechnen.

Anweisung

- 1, Programm eingeben f PRGM, 30,6 STO3, 365,25 STO 4
2. i dezimal STO 2 o STO 0
3. 1 1 1 0 R/S A: (0 N,0,0)
4. Tag | Monat | (Jahr) Kapital seit letztem Dat. R/S A: (Tagesdiff),(Zinsen), Zinssumme
5. Wiederhole 4. bis zum 31.12.

letzte Anzeige ist Endergebnis gehe z u 2.
bei Fehler: letztes Datum RCL 5 R/S gehe zu 4.

Prg 7 Zinsen im Sparbuch

18

```

00
01 23 05 STO 5 . 41 24 05 RCL 5
02 22 R↓ . 42 14 01 f INT
03 03 3 . 43 24 02 RCL 2
04 06 6 . 44 61 x
05 33 EEX . 45 24 06 RCL 6
06 03 3 . 46 61 x
07 71 + . 47 23 51 00 STO + 0
08 23 02 STO 2 . 48 24 00 RCL 0
09 00 0 . 49 13 14 GTO 14
10 23 00 STO 0
11 23 01 STO 1
12 23 03 STO 3
13 23 04 STO 4
14 74 R/S
15 24 01 RCL 1
16 22 R↓
17 23 03 STO 3
18 34 CLX
19 01 1
20 41 -
21 03 3
22 00 0
23 61 x
24 51 +
25 24 03 RCL 3
26 15 51 g x ≥ 0
27 13 32 GTO 32
28 34 CLX
29 01 1
30 41 -
31 13 33 GTO 33
32 22 R↓
33 23 01 STO 1
34 41 -
35 32 CHS
36 23 06 STO 6
37 24 04 RCL 4
38 23 51 05 STO + 5
39 24 03 RCL 3
40 23 04 STO 4
    
```

ges: Im Laufe eines Jahres
geg: i, PV, PMT, Datum der
Ein- oder Auszahlung

Speicher
0 I Zinssumme
1 Nr. des Tages vom 1.1. aus
2 i %/36000
3 PMT1
4 PMT2
5 PV/FV
6 Tagesdiff zur letzten Kontobew
7 frei (o. Zinszwischensumme bei 2A'D)

Beispiel
i=3 %, PV 1000 DM
3 ↑ 1000 R/S A: 0
5.3. 125 R/S A: 5.42
6.7. -147 R/S A: 16.67
5.8. 321 R/S A: 19.11
6.10. -369 R/S A: 25.61
30.12. 0 R/S A: 32.19

Das Programm errechnet (ähnlich Rrg 7) die Zinsen, die im Laufe eines Jahres bei unregelmäßigen Ein- und Auszahlungen im Sparbuch anfallen. Falls die Zinsen des Vorjahres nicht gleich gutgeschrieben werden konnten, diese schon zu Beginn des Jahres dem jeweiligen Kapital zurechnen.

Nach der Vorbereitung gibt man den Zinssatz als Dezimalzahl ein STO 2.

Das Programm funktioniert zwar auch über einen Jahreswechsel hinweg, es ist aber üblich jeweils nur innerhalb eines Jahres zu rechnen. In diesem Fall werden die Programmschritte 02 und 03 mit g NOP beschrieben, und die Jahreszahl zu Beginn in Speicher 7 eingeschrieben.

Jetzt gibt man Tag und Monat des Anfangs des Zeitraumes (meist 1.1.) ein und anschließend Null. Damit sind alle Vorbereitungen abgeschlossen und man beginnt mit der Routine:

Nächstes Datum eingeben, und Kapital für den Zeitraum vom letzten Datum zu diesem Datum. Dieses Kapital steht im Sparbuch jeweils eine Zeile höher als das zugehörige Datum.

Der Rechner gibt nun die Tagesdifferenz zum letzten Datum an und danach die Zinsen für diesen Zeitraum, schließlich die bisher aufgelaufene Zinssumme, die am Schluß gleichzeitig das Endergebnis darstellt.

Bei Fehlerhafter Eingabe, die erst nach R/S bemerkt wird: Vorheriges richtiges Datum eingeben RCL 5 R/S Somit behebt der Rechner die letzte falsche Eingabe.

Bei Zinssatzwechsel bis zu diesem Datum rechnen, dann neuen Zinssatz in STO2 eingeben und weiterrechnen.

- Anweisung
- 1, Programm eingeben f PRGM, 30,6 STO3, 365,25 STO 4
 2. i dezimal STO 2 0 STO 0
 3. 1 ↑ 1 ↑ 0 R/S A: (3 N,0,0)
 4. Tag ↑ Monat ↑ (Jahr) Kapital seit letztem Dat. R/S A: (Tagesdiff),(Zinsen), Zinssumme
 5. Wiederhole 4. bis zum 31.12.

letzter Anzeiger ist Endergebnis gehe zu 2.
bei Fehler: letztes Datum RCL 5 R/S gehe zu 4.

```

00
01 21 x==y 41 15 01 g FRAC
02 14 01 f INT 42 24 01 RCL 1
03 23 00 STO 0 43 61 x
04 00 0 44 23 51 03 STO + 3
05 23 03 STO 3 45 02 2
06 22 R↓ 46 23 61 01 STO x 1
07 22 R↓ 47 13 33 GTO 33
08 23 01 STO 1 48 24 03 RCL 3
09 01 1 49 13 00 GTO 00
10 00 0
11 23 02 STO 2
12 14 71 f x=y
13 13 33 GTO 33
14 01 1
15 24 00 RCL 0 Speicher
16 14 41 f x<y 0 Zahl u a
17 13 48 GTO 48 1 1, 10, 100 usw
18 02 2 2 10
19 71 + 3 Ergebnis
20 14 01 f INT
21 23 00 STO 0
22 14 73 f LASTx 4 frei
23 15 01 g FRAC . .
24 02 2 . .
25 61 x
26 (15)74 R/S(g NOP)
27 24 01 RCL 1
28 61 x
29 23 51 03 STO + 3
30 24 02 RCL 2
31 23 61 01 STO x 1
32 13 14 GTO 14
33 24 00 RCL 0
34 15 71 g x=0
35 13 48 GTO 48
36 24 02 RCL 2
37 71 +
38 14 01 f INT
39 23 00 STO 0
40 14 73 f LASTx
    
```

Gerhard Streichert
Raiffeisenstraße 1
8431 Postbauer-Heng

Das Programm interpretiert eine eingegebene Zahl entweder als Dualzahl und errechnet den dezimalen Gegenwert, oder umgekehrt. Es werden nur ganzzahlige Zahlen berechnet. Zur Sicherheit wird auf die eingegebene Zahl die Funktion f INT angewendet.

Zur Unterscheidung der gewünschten Rechenrichtung gibt man nach der Zahl für gesuchten Dezimalwert eine 10 und für gesuchten Dualwert eine 1 ein.

Da Zahlen, die größer sind als 1023, bereits mit Zehnerexponenten dargestellt werden müssen, ist eine genaue Ablesung der letzten Stellen nicht möglich. In diesem Fall ändert man Schritt 26 zu R/S. Nun werden die Dualen Ziffern 0 und 1 einzeln angezeigt und zwar von rechts nach links. Das Rechenende wird durch Anzeige der gesamten Zahl angezeigt.

mit Zehnerpotenz

Anweisung:

1. Programm eingeben, f PRGM,
2. umzurechnende Zahl (1 oder 10) R/S (A) Ergebnis für neue Rechnung gehe zu 2.

Wird das duale Ergebnis mit Zehnerexp. dargestellt, so ändere Schritt 26 in R/S und wiederhole die Rechnung. Es werden nun die dualen Ziffern 1 und 0 einzeln von rechts nach links angezeigt. Als Abschluß erscheint nochmals das Gesamtergebnis.

B. 1100111010 RB 51
1911 RB 10011
255 g x² 1 RA 1111111000000001 =
1,1111110 15

Telefongebühren

Gerhard Streichert

Raiffeisenstraße 1
8431 Postbauer-Heng

00							
01	23	03	STO 3				
02		73	.				
03		02	2	Gebühr für			
04		03	3	1 Einheit			
05	23	01	STO 1	40	14	01	f INT
06	23	00	STO 0	41	14	74	f PAUSE
07		34	CLx	42	24	01	RCL 1
08	23	02	STO 2	43		61	x
09		01	1	44	23	51	06 STO+6
10		73	.	45	13	29	GTO 29
11		02	2	46	24	06	RCL 6
12		08	8	47	23	41	06 STO-6
13	23	04	STO 4	48	23	51	07 STO+7
14	24	00	RCL 0	49	13	29	GTO 29
15	14	74	f PAUSE				
16	24	04	RCL 4				
17	23	51	02 STO+2				
18	24	02	xxxx RCL 2				
19	24	03	RCL 3				
20		71	+				Register
21		01	1	0			Summe Gebühren
22	14	51	f x=y	1			Preis für Einheit zZ 0,23
23	13	14	GTO 14	2			Zählwerk für Zeitmessung
24	24	03	RCL 3	3			Grenze für Zeitmessung
25	23	41	02 STO-2	4			Prop.-Faktor
26	24	01	RCL 1	5			-
27	23	51	00 STO+0	6			Summe Gebühren für 1 Teiln
28	13	14	GTO 14	7			Summe Gebühren für alle Teiln
29		74	R/S				
30	14	71	f x=y				
31	13	46	GTO 46				
32	15	00	g → H				
33		21	x=y				
34		06	6				
35		00	0				
36		71	+				
37		71	+				
38		01	1				
39		51	+				

Programmteil 1 dient dazu, während des Gesprächs die bisher verbrauchten Gebühren anzuzeigen. Dazu muß dem Rechner die Zeit für eine Einheit bekannt sein. Diese ermittelt sich nach folgender Tabelle:

Uhrzeit	Montag bis Freitag			Samst., Sonnt., Feiert.		
	8 - 18 Uhr	8 min	45	12	12	12 min
18 - 8 Uhr	12 min	67,5	38,6	38,6		
Zone	O/N	1	2	3	O/N	1 2 3

Die Zonennummer entnimmt man der Beilage zum AVON. Die ermittelte Zahl (Zeit/Einheit) gibt man ein und drückt R/S wenn der Gesprächspartner abhebt. Nur wird laufend der bisher verbrauchte Betrag angezeigt.

Anweisung:

Programm eingeben.

1. f PRGM Ein: Zeit/Einheit (siehe oben)
2. Beim Zustandekommen der Verbindung R/S
3. Anz: laufend die bisher verbrauchte Gebühr
4. Bei Ende des Gesprächs R/S RCL 0 Anz: Gebühr
5. Für neues Gespräch gehe zu 1.

Der Programmteil 2 dient dazu, am Monatsende die Gebühren aus der notierten Gesprächsdauer und sonstigen Angaben zu berechnen und aufzuaddieren.

Anweisung

1. Programm eingeben, einmal Programmteil 1 ausführen
2. GTO 30, 0 STO 6, STO 7
3. Ein: Zeit/Einheit (sec) ↑ Gesprächsdauer (min, sec) R/S
4. Anz: (Zahl der Einheiten), Gebühr
5. gehe zu 3. bis alle Gespräche von diesem Teilnehmer durch
6. ↑ R/S Anz: Gebührensomme dieses Teilnehmers
gehe zu 3. für nächsten Gesprächsteilnehmer
7. Wenn alle Teilnehmer durch, RCL 7 Anz: Rechnungsbetrag, der mit Postrechnung übereinstimmen sollte.

Überstundenzuschläge

hp 25

00					
01	15	00	g → H		
02		21	x → y		
03	15	00	g → H		
04		41	-	MA	
05		04	4		
06		61	x		
07	14	01	f INT		
08		04	4		
09		71	+	MA gerundet	
10	23	51	06	STO + 6	aufsummieren
11		24	01	RCL 1	
12		25	00	g → H	
13		00	+	w	
14		02	2		
15		00	0		
16		41	-	NMA	
17	15	41	g x c0		
18		34	CLx	nur pos.	
19	23	51	07	STO + 7	aufsummieren
20		13	00	STO 00	Ende Routine
21		24	03	RCL 3	
22		24	06	RCL 6	
23			06	6	
24		24	07	RCL 7	
25		14	51	f y → y	
26		21	x → y	TMA50 - (größere Zahl: 6 oder NMA)	
27		22	R+		
28		41	-		
29		25	41	g x c0	
30			34	CLx	nur pos.
31			61	x	1,5 Faktor
32		24	06	RCL 6	MA
33	23	41	06	STO - 6	Speicher löschen
34		14	73	f LAST x	TMA 50
35			41	-	
36		24	07	RCL 7	NMA
37			41	-	
38		24	05	RCL 5	
39			61	x	1,25 Faktor
40			51	+	
41		24	07	RCL 7	NMA
42	23	41	07	STO - 7	Speicher löschen
43		24	04	RCL 4	1,6 Faktor
44			61	x	
45			51	+	
46		24	00	RCL 0	GG
47			61	x	
48		24	02	RCL 2	173
49			71	+	

Gerhard Stroichert
 Helfferstraße 1
 8431 Postbauer-Hang

Anzeige Eingabe Kommentar

Register

- 0 GG oder 100
- 1 ZAS = ZÜB
- 2 173
- 3 1,5
- 4 1,6
- 5 1,25
- 6 MA
- 7 NMA

Das Programm errechnet die Überstundenzuschläge, die in einer Woche anfallen. Gibt man als Grundgehalt (GG) 100 ein, so werden die Zuschläge in Prozent errechnet.

Für die ersten sechs Überstunden gibt es 25% Zuschlag. Für die 7. und folgenden Überstunden beträgt der Zuschlag 50%. Für Nachtarbeit ab 20 Uhr (die bei uns immer Mehrarbeit ist) gibt es 60%, unabhängig davon, ob sie zu den ersten 6 gehört oder nicht.

Anmerkung: Uhrzeiten werden immer in der Form H.MS eingegeben, im Rechner dann zu H.dez umgerechnet.

Überstunden vor 6 Uhr werden nicht als Nachtarbeit erkannt. Abhilfe: Auf die Zeit nach 20 Uhr umrechnen.

Feiertagsarbeit kann nur für unbezahlte Feiertage berechnet werden. An bezahlten Feiertagen wird keine zusätzl. Grundvergütung bezahlt, sondern "nur" 100% Zuschlag. Dieser Fall wird vom Programm nicht erfasst.

Formeln: MA = (ZÜS - ZÜB) auf 0,25 gerundet
 NMA = (MA + ZÜB - 20) wenn negativ, dann durch 0 zu ersetzen
 MA und NMA wird im Lauf einer Woche aufsummiert

Hg = GG/173 * ((TMA50 * 1,5) + (TMA25 * 1,25) + NMA * 1,6)
 TMA50 = MA - (größere Zahl von 6 oder NMA) nur pos.
 TMA25 = MA - TMA50 - NMA

- Abkürzungen:
- HG Überstundenlohn (Grundvergütung + Zuschlag) brutto
 - GG Grundgehalt (monatl. brutto)
 - MA Mehrarbeit in Stunden dez./wird aufsummiert
 - NMA Nachtmehrarbeit ab 20 Uhr
 - ZÜS Zeit-Überstundenschluß
 - ZÜB Zeit-Überstundenbeginn
 - ZAS Zeit-Arbeitschluß (regulär) = ZÜB an Arbeitstagen
 - TMA50 Tagemehrarbeit, die mit 50% Zuschlag vergütet wird
 - TMA25 Tagemehrarbeit, die mit 25% Zuschlag vergütet wird

Anweisung:

1. Programm eingeben, f INGM, f BEG, 173 STO2, 1,5 STO3, 1,6 STO4, 1,25 STO5
2. GG STO 0, oder 100 STO 0, dann ist das Ergebnis in %
3. ZÜB (Zeit-Überstundenbeginn) STO 1,
4. An Arbeitstagen: RCL 1 (ZÜB), ZÜS R/S Anzeige: nicht bedeutend
5. Wiederhole 4. für Montag bis Freitag
6. An Samstag: ZÜB STO 1, ZÜS R/S Anzeige: nicht bedeutend
7. Wiederhole 6. für Sonntag
8. STO 27 R/S Anzeige: HG für diese Woche, ev in %
 Notiere HG, da kein Speicher im Rechner frei ist.
9. Für nächste Woche gehe zu 3.

UHRZEIT (Echtzeitprogramm)

```

00
01 24 01 RCL 1
02 23 51 00 STO+0
03 24 02 RCL 2
04 24 00 RCL 0
05 14 51 f x=y
06 13 10 GTO 10
07 14 00 f H.MS
08 14 74 f PAUSE
09 13 01 GTO 01
10 24 02 RCL 2
11 23 41 00 STO-0
12 13 01 GTO 01
13 13 00 GTO 00

```

Gerhard Streichert
Raiffeisenstraße 1
8431 Postbauer-Heng

Dieses Programm zeigt die aktuelle Uhrzeit an.

Anweisungß

1. Programm eingeben, GTO 08, f FIX 4
2. 3,081068401 EEX - 4 STO 1
3. 24 STO 2
4. aktuelle Uhrzeit (H.ms) g H. STO 0
5. Ist die Zeit erreicht, R/S
6. A: (aktuelle Uhrzeit)

05
01
00

(unvollständiges Programm)

STO 0
STO -0

1
2
X

STO 1

↓R

STO 7

↓R

EEX

2

÷

STO 2

14 RCL 7

gX=0

GTO 49

fPAUSE

1

STO →

RCL 0

RCL 2

1

+

X

RCL 1

RCL 2

2

÷

1

+

X

+

STO 0

fPAUSE

GTO 14

49 RCL 0

7%

↑

30 Jahre

↑

100 DM/monat

Zinssatz

in 2

↑

Laufzeit in Jahr.

decr. in 7

↑

Monatsbeitrag

in 1

Summe

in

⊙

→ 1513



Super Hirn Form und Farbe

Würfelprogramm existiert noch nicht

Programmteil SW und WS-Stifte

Programmteil blaue Stifte

06

```

STO 5
RCL 2
RCL 1
STO 2
R↓
STO 1
g NOP
RCL 5
g NOP
f x=y
GTO 14
1
STO+6
RCL 2
g NOP
RCL 5
g NOP
f x=y
GTO 22
1
STO+7
1
STO+0
1
2
RCL 0
f x=y
GTO 43
3
=
g FRAC
g x=0
GTO 00
RCL 2
RCL 3
RCL 4
STO 3
R↓
STO 2
R↓
STO 4
GTO 14
STO-0
RCL 7
RCL 6
STO 7 f PAUSE
STO-6 x=y
    
```

```

STO 5
RCL 2
RCL 1
STO 2
R↓
STO 1
L INT
RCL 5
L INT
f x=y
GTO 14
1
STO+6
RCL 1
g FRAC
RCL 5
g FRAC
f x=y
GTO 22
1
STO+6
1
STO+0
1
2
RCL 0
f x=y
GTO 43
3
=
g FRAC
g x=0
GTO 00
RCL 2
RCL 3
RCL 4
STO 3
R↓
STO 2
R↓
STO 4
GTO 14
STO-0
RCL 7
RCL 6
STO 7 f PAUSE
STO-6
    
```

GTO 22
STO-0
RCL 6
~~STO-6~~
RCL 7 ~~2~~
STO-7 ~~x~~ *Hand*