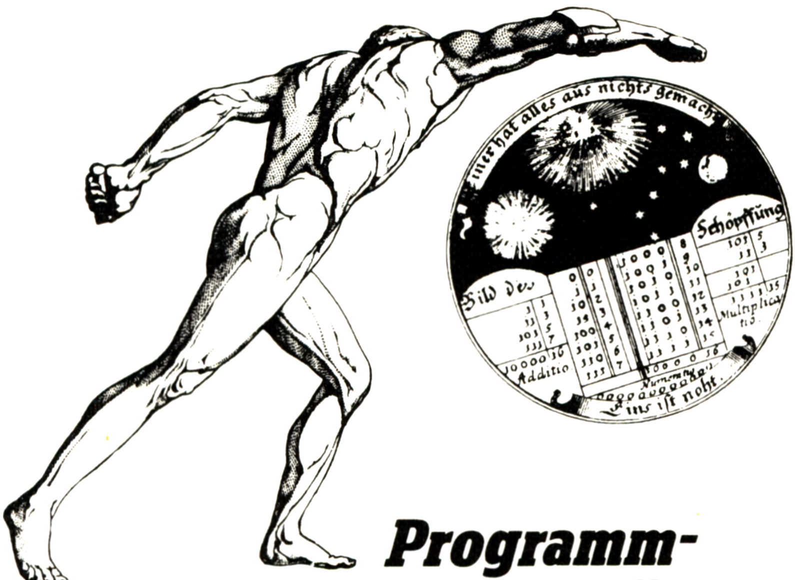
D. Herrmann G. Schnellhardt

Schneider CPC

Mathematik



Programm-Beispiele für den Anwender

1

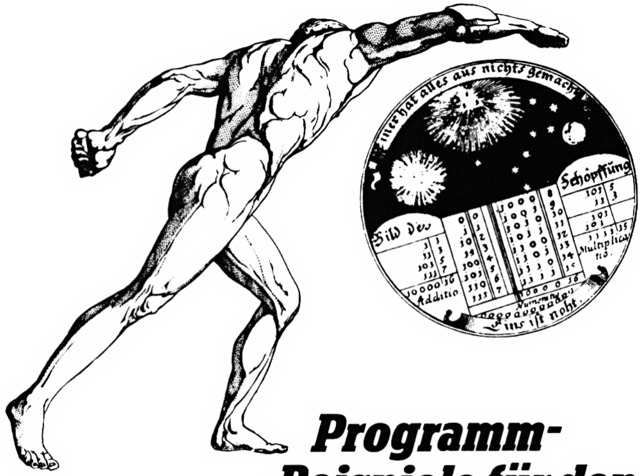
iwt

Für Eva und Barbara

D. Herrmann G. Schnellhardt

Schneider CPC

Mathematik



Programm-Beispiele für den Anwender

1

IWT

ISBN 3-88 322-152-X
1. Auflage 1985

Printed in Western Germany

Holdenrieds Druck- und Verlags-GmbH, Füssen
Umschlaggestaltung: Kaselow und Partner, München

VORWORT

Dieser Band enthält 39 mathematische Programme aus den Bereichen

- Mehr-Register-Arithmetik
- Zahlentheorie
- Kombinatorik
- Algebra
- Geometrie
- Numerische Mathematik

Besonders nützlich werden Praktiker und Computerfans die hier neu vorgelegte Langzahl-Arithmetik finden, die die Grundrechenarten für Zahlen bis 255 Stellen gestattet!

Zahlreiche Anwendungen finden auch die hier angegebenen kombinatorischen Prozeduren: Wer wollte nicht schon mal alle Permutationen oder Kombinationen eines Problems durchspielen? Vielfältige Unterstützung findet auch das Rechnen mit Polynomen, Matrizen und komplexen Zahlen: Neben dem komplexen Hornerschema werden insbesondere Algorithmen zur Polynomdivision und Matrizeninversion gegeben.

Anwendungsbezogen sind die Programme der Numerischen Mathematik. Hier werden Verfahren zur Lösung von linearen und nichtlinearen Gleichungen gegeben, zum Eigenwertproblem von Matrizen und zur numerischen Integration und Differentiation. Alle Programme werden durch vollständige Beispiele und komplette Programmausdrucke erläutert.

Dem Verlag danke ich für die Herausgabe des Bandes und für die stets freundliche Zusammenarbeit.

Anzing, im Juni 1984

A C H T U N G

Bitte beachten Sie die folgenden Korrekturen!

Seiten 21 und 24 und 29, hier bitte jeweils die Programmzeile 220 wie folgt ändern:

```
220 IF M N THEN H$=A$:A$=B$:B$=H$:H=N:N=M:M=H
```

Seite 82, hier ist das Ergebnis auf 6 Kommastellen gerundet.

INHALTSVERZEICHNIS

Seite

GEOMETRIE

NUMERISCHE MATHEMATIK

ANHANG

La machine arithmetique fait des effects
qui approchent plus de la pensee
que tout ce que font des animaux

Pascal

Daß die niedrigste aller Geistestätigkeiten
die arithmetische ist, wird dadurch belegt,
daß sie die einzige ist, welche auch durch
eine Maschine ausgeführt werden kann

Schopenhauer

EINLEITUNG

Mathematik und Computer

Schon zu allen Zeiten hat es bemerkenswerte numerische Leistungen gegeben.

Die berühmte Stieraufgabe, mit der Archimedes die Mathematiker von Alexandria zur Verzweiflung trieb, hatte keine kleinere Lösung als

10366482, 7460514, 7358060 und 4149387.

Ähnlich schwierig war die Lösung $x = 1.3688081075$ der Gleichung

$$x^3 + 2x^2 + 10x = 20$$

die Leonardo von Pisa, Fibonacci genannt, um 1220 auf 10 Dezimalen genau berechnete.
Van Ceulen (1539–1610) benötigte mehr als die Hälfte seines Lebens, um die Zahl pi auf 35 Dezimalen zu berechnen. Als Dank dafür hat man sie in seinen Grabstein in Leiden gemeißelt. Fermat (1601–1655) stellt in Briefen an seine armen Zeitgenossen mehrfach die Frage nach den Lösungen von

$$x^2 - 149y^2 = 1$$

deren kleinste Werte $x = 25801741449$ und $y = 2113761020$ sind.

In mühevoller Arbeit berechnete W. Shanks 1873 die ersten 707 Dezimalstellen von pi. Ein Großteil dieser Mühe war jedoch vergeblich, da, wie sich später herausstellte, die 528ste Stelle bereits falsch war.
An dieser Stelle muß auch der immense Rechenaufwand bedacht werden, den Ptolemaeus für die Erstellung seiner Sehnentafel Buergi und Brigg mit ihren Logarithmentafeln und Kepler mit den Rudolffinischen Tafeln hatte.
Alle diese Probleme hätten in kurzer Zeit mit Hilfe eines Rechners gelöst werden können. Es erhebt sich die Frage, ob der Computer auch prinzipiell neue Fragestellungen in Angriff nehmen kann.

1976 kam es zu dem seltenen Ereignis, daß ein mathematischer Lehrsatz Schlagzeilen in der berühmten New York Times machte. Das sensationelle Ereignis war, daß der Beweis des Vierfarbensatzes mit Hilfe von Computern gelungen war.
K. Appel und W. Haken ernteten jedoch mit ihrem Beweis bei den meisten Mathematikern nur Kopfschütteln. Ein mathematischer Beweis war traditionell etwas, was man mit Papier und Bleistift nachvollziehen konnte.

Wer aber sollte das 460-seitige Computerprogramm von Appel und Haken nachprüfen? Dazu kamen weitere 133 Seiten des Illionis Journal of Mathematics, auf denen die Autoren ihr Computerprogramm erläuterten.

Das Vier-Farben-Problem, 1852 von Francis Guthrie aufgeworfen, besagt, daß jede ebene Landkarte mit 4 Farben so gefärbt werden kann, daß je 2 benachbarte Länder verschiedenfarbig sind. Trotz erheblicher Anstrengungen widerstand die Vermutung über 120 Jahre lang den Beweisbemühungen der Mathematiker.
Die Schwierigkeit war die, alle möglichen Lagemöglichkeiten der Länder in den Griff zu bekommen. Auf Grund der Vorarbeiten von Heesch (Hannover) gelang es Appel und Haken, alle möglichen Konfigurationen in einigen tausend Fallunterscheidungen zu erfassen und vom Computer überprüfen zu lassen.

Großen Aufschwung erfuhren die Teile der Mathematik, die sich mit endlichen Mengen befassen, wie die Kombinatorik, Endliche Geometrie, Graphen- und Codierungstheorie, durch die Anwendung von Computern. Auch die Zahlentheorie erfuhr wesentliche Anregungen, so wurden z.B. sehr effektive Primzahltestverfahren entwickelt. Mit ihrer Hilfe konnte im September 1983 die z.Z. größte bekannte Primzahl

$$2^{132\,049}-1$$

mit 39751 Stellen gefunden werden. Die Rechenzeit beträgt bei den hier benützen Lucas-Lehmer-Tests nur wenige Stunden.
Konträr ist auch die Meinung der Mathematiker zu den stochastischen Primzahltests, die Primzahlen nur einer gewissen Wahrscheinlichkeit nachweisen. Beim Rabintest läßt sich die Irrtumswahrscheinlichkeit bei einigem Aufwand auf

$$10^{-60}$$

senken. Kann man dies als Beweis der Primzahleigenschaft ansehen? Oder ist nicht schon die Irrtumswahrscheinlichkeit eines Menschen größer?

Bezeichnend ist die Geschichte von Rosser, Schönfeld und Yohe (Wisconsin), die bei ihrer Arbeit über die noch unbewiesene Riemannsche Hypothese einen Rechner benützten. Als "reine" Mathematiker mißtrauten sie dem Computer und überprüften in mühseliger Kleinarbeit dessen Betriebssystem. Tatsächlich fanden sie mehrere Fehler in der internen Rechnerlogik. Diese Fehler hatte bisher noch niemand bemerkt, obwohl der Computer bereits seit mehreren Jahren in Betrieb war.

Im Gegensatz zu den Primzahltests fehlt es bei der Primzahlfaktorisierung noch an effektiven Algorithmen. So weiß man seit langer Zeit, daß eine Zahl wie

$$2^{256} + 1$$

zerlegbar ist, man kennt jedoch bis heute noch nicht ihre Primzahlzerlegung.

Die Primfaktorisierung hat große Bedeutung in der Kryptologie gewonnen. Man kann nämlich damit Verschlüsselungsverfahren konstruieren und ihr Prinzip offenlegen, ohne befürchten zu müssen, daß es "geknackt" wird (Verfahren von Rivest, Shamir und Adleman). Die Rechenzeit zur Faktorisierung einer 1000stelligen Zahl beträgt z.Z. noch 4 Mrd. Jahre. Dies hat große politische Bedeutung, falls es einmal zu einem generellen Atomwaffen-Sperrvertrag kommen sollte und die Codierung der Beobachtungssonden bekanntgegeben werden müßte.
Ein weiteres bekanntes bisher ungelöstes Problem der Zahlentheorie ist der sog. große Fermat-Satz. Er besagt, daß es keine ganzzahlige Lösung der Gleichung $\neq 0$

$$x^n + y^n = z^n$$

für $n>2$ gibt. Gleichungen, von denen nur ganzzahlige Lösungen gesucht werden, nennt man Diophantisch. Vor kurzem (1983) konnte Faltings (Wuppertal) zeigen, daß Diophantische Gleichungen vom Grad >3 höchstens endlich viele Lösungen haben können (Mordellsche Vermutung). Durch die Reduzierung auf endlich viele Fälle scheint das Problem nun mit dem Computer greifbar.
Im Falle der sog. Catalanschen Vermutung ist dies tatsächlich der Fall. Diese besagt, daß es keine zwei Potenzen >2 von natürlichen Zahlen – außer 8 und 9 – gibt, deren Differenz 1 ist. Tijdemann konnte ebenfalls zeigen, daß das Problem höchstens endlich viele Lösungen hat. Durch umfangreiche Computerberechnungen glaubt er die Catalansche Vermutung mit an Sicherheit grenzender Wahrscheinlichkeit bewiesen zu haben. Was ist von einem solchen Beweis zu halten?
Auch viele Fragestellungen der Kombinatorik, die infolge des großen Rechenaufwandes bisher ungelöst waren, konnten seitdem beantwortet werden. Bekannt ist das Eulerische Offiziersproblem (1782): Ist es möglich, 6 verschiedene Offiziere aus 6 verschiedenen Regimentern so in einem Quadrat aufzustellen, daß in jeder Reihe und Spalte jeder Offiziersrang und jedes Regiment vertreten ist?

Eine solche Anordnung nennt man ein Paar von orthogonalen Lateinischen Quadraten. Euler vermutete, daß es keine orthogonalen Quadrate der Ordnung 6, 10, 14, usw. gäbe. Erst in den Jahren 1959/60 konnten Bose und Shrikhande nach umfangreichen Rechnungen orthogonale Lateinische Quadrate der Ordnung 10 und 14 angeben. Sie erhielten dafür den Spitznamen "the Euler Spoilers".
Verwandt mit den orthogonalen Lateinischen Quadraten sind die Blockpläne der Endlichen Geometrie. Ausgangspunkt war das Schulmädchenproblem des

Geistlichen Kirkman (1851): Kann man 15 Schulmädchen an den 7 Wochentagen so in Dreierreihen spazieren führen, daß jedes Mädchenpaar an genau einem Tag zusammentrifft?
Spezielle Blockpläne sind die sog. Steinersysteme S(t,k,v). Dabei müssen v Dinge zu je t Stück in k Reihen angeordnet werden können. Die Lösung des Schulmädchen-Problems stellt somit ein (2, 3, 15)-Steinersystem dar. Es ist im Rahmen des Buches nicht möglich, die Vielzahl der ungelösten Probleme im Zusammenhang mit den Steinersystemen aufzuzählen. Erwähnt sei, daß die kleinsten Steinersysteme, deren Existenz bisher unbewiesen sind, folgende sind:

S (4, 5, 17), S (4, 5, 21), S (4, 5, 27)

Durch umfangreiche Rechnung konnten Mendelsohn und Hung vor wenigen Jahren z.B. zeigen, daß das Steinersystem S (4, 5, 15) nicht existiert. Hier bleibt noch viel Forschungstätigkeit für künftige Mathematiker und viel Rechenarbeit für die nächste, noch schnellere Computergeration zu tun.

"Aus Margarita Philosophica des Gregor Reisch, Freiburg 1503.
Boethius mit indischen Zahlen und Pythagoras als Linienrechner vor der Arithmetica sitzend"

Es macht mich traurig, daß gebildete Leute
nicht einmal von der Existenz
meines Fachgebietes wissen

P. R. Halmos

Die Furcht vor der Mathematik . .
steht der Angst erheblich näher
als die Ehrfurcht

F. Auerbach

Mathematik — die ungeliebte Wissenschaft

Kein Zweifel, die Mathematik war schon immer eine unpopuläre Wissenschaft.

Besonders dem Christentum waren die heidnischen Schriften eines Euklid-Diophant und Archimedes als heidnisch suspekt. Tertullian schreibt:

> "Nach Christus brauchen wir keinerlei Wissbegier mehr, nach den Evangelien sind keinerlei Forschungen mehr nötig".

Als Folge der Verketzerung der Heiden, wurde die berühmte Mathematikerin Hypathia, Tochter des Akademievorstehers Theon, 415 von einer fanatischen Menge zu Tode gemartert. Dies, obwohl der Kirchenlehrer Synesios von Kyrene, Bischof von Ptolemais, ein Hörer ihrer Vorlesung war. Sogar die Mathematik hatte den Beigeschmack des Heidnischen. So lautete eines der Gesetze des Justinian-Kodex: "de Maleficis, mathematicis et ceteris similis" (Über Übeltätern, Mathematikern und dgl.):

> "Vollständig verboten ist die verdammenswerte Kunst der Mathematik"

Die Mathematiker werden dort mit Giftmischern, Zauberern und Sterndeutern (Veneficis, Magis, Chaldeis) in einen Topf geworfen. Noch 1614 bezeichnete der Pater Caccini in einer Predigt, die gegen Galilei gerichtet war, die Mathematik als Teufelskunst und die Mathematiker als Urheber aller Ketzereien, die man aus allen Staaten vertreiben müsse.
Er stützte sich dabei auf den heiligen Augustinus, der in De genesi ad literam geschrieben hatte:

> "Es besteht die Gefahr, daß die Mathematiker mit dem Teufel im Bunde, den Geist trüben und in die Bande der Hölle verstricken".

Als 529 Kaiser Justinian die Athener Philosophenschule schloß, erlosch eine lange Tradition mathematischer Lehre. Noch bis 485 hatte an dieser Schule Proklos gelehrt, dem wir einen grundlegenden Euklid-Kommentar verdanken.

Es ist ein ganz wesentlicher Verdienst der arabischen Kultur ab 800 die wichtigsten Werke des Apollonius, Archimedes, Heron, Euklid, Diophant, Ptolemaeus usw. in ihre Sprache übersetzt und somit der Nachwelt erhalten zu haben. Einige Übersetzungen aus dem Griechischen ins Lateinische führte auch Boethius (480 - 524) durch. Von ihm stammt auch die im Mittelalter übliche Einteilung der Künste in das Trivium (Grammatik, Rhetorik, Logik) und das Quadrivium (Arithmetik, Geometrie, Astronomie und Musik).

Um eine mathematische Allgemeinbildung in Gang zu bringen, führte der Schotte Alkuin von York, im Auftrag Karls des Großen, 789 eine Schulreform durch. Sie forderte, daß neben Psalmen, Noten und Grammatik insbesondere auch der Computus gelehrt werden solle. Unter dem Computus Ecclesiaticus verstand man die Berechnung der beweglichen Kirchenfeiertage. Es war nämlich der Kirche ein Dorn im Auge, daß wegen der mangelhaften Rechenfertigkeit der Geistlichen das Osterfest zu verschiedenen Zeitpunkten gefeiert wurde. Alkuin selbst schrieb ein Rechenbuch zu Unterrichtszwecken.

Der Rechenunterricht scheint – wie heute – nicht die besondere Freude der Schüler gewesen zu sein. Ein Mönch namens Strabo der Klosterschule Reichenau schreibt 822 in sein Tagebuch:

> "Zur Abwechslung und Unterhaltung lösten wir mathematische Rätsel, welche Alkuin für den großen Karl gefertigt hatte. Viele vermochten nicht allen diesen Rechnungen zu folgen, und bevor wir zur Geometrie übergingen, traten diejenigen aus, welche sich fortan den Studien der Medizin, Rechtswissenschaft und den Künsten der Malerei und der Bildhauerei widmen wollten".

Ein Mathematiker, Gerbert von Aurillac (940 - 1003) bestieg als Sylvester II 999 den Stuhl Petri. Gerbert hatte einige Jugendjahre in Spanien verbracht und dort von den Arabern den Abakus (Rechenbrett) kennengelernt. Als Lehrer der Domschule Reims verfaßte er ein Lehrbuch zum Rechnen am Abakus. Auch an den aus den Domschulen hervorgegangenen Universitäten Bologna (1088), Paris (1150), Salerno (1173) und Montpellier (1180) spielte die Mathematik keinerlei Rolle.

Von der Universität Erfurt weiß man aus Quellen, daß die dort wirkenden Humanisten die Anstellung eines Mathematikers strikt ablehnten. Noch Anfang des 16. Jahrhunderts legten an der Universität Paris die Kandidaten für den Grad des Magisters der Künste kein Examen im Fach Geometrie ab. Sie mußten lediglich beeiden (!), Vorlesungen über die ersten 6 Bücher des Euklid gehört zu haben.
Das Risiko einer wirklichen Prüfung wollte man nicht eingehen. Bezeichnend ist, daß der Lehrsatz des Pythagoras am Ende des 1. Kapitels den Namen "Magister Matheseus" trug und somit als Weisheit letzter Schluß galt.

Welche Schwierigkeiten das Rechnen damals machte, wird aus der Rede Philipp Melanchtons vor der Universität Wittenberg (1517) deutlich.

". . . Deshalb können ihre Anfangsgründe (der Rechenkünste) gar nicht dunkel und schwer sein; sie sind im Gegenteil so durchsichtig, daß Kinder sie begreifen können, weil ja alles so natürlich vor sich geht. Die Regeln des Vielfachen und Teilens allerdings erfordern viel mehr Fleiß, aber ihr Sinn wird schon bald von den aufmerksameren eingesehen werden. Übung und Anwendung erfordert diese Fertigkeit wie alle anderen."

"Ein Vater meldet seinen Sohn bei einem Rechenmeister an.
Holzschnitt von Hans Weiditz, 1535"

Die Mathematik-Kenntnisse dieser Studenten aus den Lateinschulen sind entsprechend schlecht gewesen. Der Rechenunterricht wurde in der Regel von einem externen Rechenmeister übernommen, die dazu meist eigene Rechenbücher verfaßten. Der bekannteste Rechenmeister war Adam Ries(e) in Erfurt und Annaberg.
Es dauerte noch sehr lange, bis Mathematik ein reguläres Unterrichtsfach war. So wurde noch 1805 (!) im Jahresbericht eines lutherischen Gymnasiums in Essen vermerkt, daß an der Schule kein Rechenunterricht erteilt werde. Dies sei der Fall, weil ältere Schüler den Lehrern bekannt gemacht hätten, "daß in der Stadt jetzt ein fertiger Rechenmeister sei, der eine kürzere Methode habe".

Oft wurde der Mathematik-Unterricht von nicht ausgebildeten Lehrkräften, z.B. von Theologen übernommen, die in Wartestellung auf eigene Pfarrpfründe waren. Ob es wohl diese Theologen waren, über die es in der Prüfungsordnung von 1820 für Preussen heißt: "Um das Eindringen untüchtiger Objekte in den Höheren Schulen Einhalt zu gebieten . . .". Auch in Bayern waren viele Mathematik-Lehrstühle im 19. Jahrhundert noch mit Theologen besetzt.

Daneben litt die mathematische Lehre noch unter anderen Schwierigkeiten. Mollweide in Leipzig erklärte es für unmöglich, neben der, für alle Fakultäten verbindlichen Mathesis pura, auch höhere Mathematik zu bringen, weil es dabei zuviel Schreibens an der Tafel gäbe.
Bezeichnend ist auch der Bericht von Neumann in Berlin von 1818: "Als ich mich beim Professor für Mathematik meldete, sagte dieser, ja ich habe die Vorlesung angezeigt, aber sie pflegt niemals zustande zu kommen. Ich verab-

"Darstellung der Arithmetica und Geometria aus dem Quadrivium, Holzschnitte um 1500" redete mich mit 5 anderen, zu ihm zu gehen. Der Professor kam ins Auditorium und schrieb ununterbrochen Formeln an die Tafel und sprach dabei kein Wort, bis die Zeit um war. Am 2. Tag kamen nur noch 2 Zuhörer. Der Professor stellt sich wieder an die Tafel und zeichnete wieder ununterbrochen mathematische Formeln an diese. Am dritten Tag kam außer mir nur noch ein Zuhörer. Der Professor erschien, ging ans Katheder, wandte sich an uns und sagte: Sie sehen, meine Herren, es kommt kein Kolleg zustande".

"Darstellung der Musica und Astronomia, Holzschnitte um 1500"

1. Addition langer Zahlen

Jeder, der schon länger mit dem Computer rechnet, ist schon einmal an die Grenze der Rechengenauigkeit gestoßen. Während bei Großrechenanlagen meist eine Langzahl-Arithmetik hardware-mäßig implementiert ist, ist dies bei Mikrocomputern nicht der Fall.
Daher sollen im folgenden, Programme zu den Grundrechenarten für lange Zahlen angegeben werden.

Diese Programme setzen voraus, daß die eingegebenen Zahlen kommafrei und positiv sind. Dies ist keine große Einschränkung, da die Vorzeichenregeln leicht von Hand ausgeführt werden können. Kommafreie Zahlen können durch entsprechendes Ausklammern von Zehnerpotenzen erhalten werden, z.B.

$$12.3456789 = 123456789 * 10^{-7}$$
$$0.000987654 = 987654 * 10^{-9}$$

Alle Zahlen werden als Stringvariablen eingegeben und können daher bis zu 255 Stellen haben, sie werden innerhalb des Programms in Felder zerlegt. Dabei kann jedes Feld 1 bis 8 Ziffern umfassen.
Umfaßt ein Feld mehrere Ziffern, so ist zu beachten, daß der Rechner führende Nullen nicht ausdruckt. Mit Hilfe der Stringfunktion kann man auch führende Nullen ausdrucken lassen, wie es z.B. im Programm 7 geschieht.

Wird der 1. Summand durch das Feld U(I), der zweite durch V(I) dargestellt, so kann bei ziffernweiser Addition das Addieren wie folgt beschrieben werden:

```
 90  K = 0
100  FOR I=N TO 1 STEP -1
110  W = U(I) + V(I) + K
120  W(I) = W - INT(W/10) * 10
130  K = INT(W/10)
140  NEXT I
150  W(0) = K
```

Dabei ist N die Länge des größten Summanden, K der Übertrag. Die Summe ist im Feld W(I) gespeichert. Bei sehr großen Rechnungen, wie z.B. bei der Berechnung von 1000 Dezimalzahlen, zerlegt man die Summanden z.B. in Fünferblöcke U(I) bzw. V(I).

Die Fünferblock-Addition läuft wie folgt ab:

```
 9Ø  K = Ø
1ØØ  FOR I=N TO 1 STEP −1
1ØØ  W = U(I) + V(I) + K
11Ø  K = INT (W/1Ø↑5)
12Ø  W(I) = W − K * 10↑5
13Ø  NEXT I
14Ø  W(Ø) = K
```

Dabei ist N natürlich die Anzahl der Fünferblöcke.

Zum folgenden Programm

Als Beispiel werden im Programm die Zahlen:

783Ø412596412348

und

914Ø35812769267

addiert. Hier ergibt sich die Summe

8744448439181615

Das hier gezeigte Programm arbeitet mit Einerblöcken, so daß auch Nullen richtig ausgegeben werden.

```
100 REM Addition langer Zahlen
110 :
120 CLS:PRINT STRING$(30,42)
130 PRINT "    Addition langer Zahlen"
140 PRINT STRING$(30,42)
150 :
160 DEFSTR A,B
170 PRINT:INPUT "Eingabe 1.Summand";A$
180 INPUT "Eingabe 2.Summand";B$
190 :
200 'Austausch und Auffüllen der Stellen
210 N=LEN(A$):M=LEN(B$)
220 IF M>N THEN SWAP A$,B$:SWAP N,M
230 IF N=M THEN 280
240 FOR I=1 TO N-M
250  B$="0"+B$
260 NEXT I
270 :
280 'Umwandlung in Zahlen
290 DEFINT I-K,U-W
300 DIM U(N),V(N),W(N)
310 FOR I=1 TO N
320   U(I)=VAL(MID$(A$,I,1))
330   V(I)=VAL(MID$(B$,I,1))
340 NEXT I
350 :
360 'stellenweise Addition
370 J=N:K=0
380 WHILE J>=1
390   W=U(J)+V(J)+K
400   K=INT(W/10):W(J)=W MOD 10
410   J=J-1
420 WEND
430 W(0)=K
440 :
450 PRINT:PRINT"Summe =";
460 K=1-K
470 FOR I=K TO N
480   PRINT W(I);
490 NEXT I
500 END
```

```
******************************
   Addition langer Zahlen
******************************

Eingabe 1.Summand? 7830412596412348
Eingabe 2.Summand?  914035842769267

Summe = 8  7  4  4  4  4  8  4  3  9  1  8  1  6  1  5
```

2. Subtraktion langer Zahlen

Die Subtraktion langer Zahlen läuft als inverse Rechenoperation ähnlich wie die Addition ab. Da, wie anfangs erwähnt, auf das Rechnen mit Vorzeichen verzichtet wird, muß vorausgesetzt werden, daß der Minuend größer ist als der Subtrahend. Ist dies nicht der Fall, so vertauscht man Minuend mit Subtrahend und versieht die Differenz mit einem negativen Vorzeichen.

Wird der Minuend durch das Feld U(I), der Subtrahend durch V(I) dargestellt, so kann bei ziffernweiser Subtraktion das Subtrahieren wie folgt beschrieben werden:

```
 90  K = 0
100  FOR I=N TO 1 STEP -1
110  W = U(I) - V(I) + K
120  K = INT (W/10)
130  W(I) = W - K * 10
140  NEXT I
```

Dabei ist N die Länge des Minuenden, K der Übertrag. Die Differenz ist im Feld W(I) gespeichert. Übersteigt der Subtrahend den Minuend, so wird eine entsprechende Fehlermeldung ausgegeben.
Bei sehr großen Rechnungen, wie z.B. bei der Berechnung von Fünferblöcke U(I) bzw. V(I).
Die Fünferblock-Subtraktion läuft wie folgt ab:

```
 90  K = 0
100  FOR I=N TO 1 STEP -1
100  W = U(I) - V(I) - K
110  IF W >= 0 THEN K = 0: GOTO 130
120  K = 1: W = W + 10↑5
130  W(I) = W
140  NEXT I
```

Dabei ist N natürlich die Anzahl der Fünferblöcke.

Zum folgenden Programm

Als Beispiel werden im Programm die Zahlen:

78410351691 0456

und

9541217036428

subtrahiert. Es ergibt sich die Differenz

774562299874028

```
100 REM Subtraktiom langer Zahlen
110 :
120 CLS:PRINT STRING$(30,42)
130 PRINT "  Subtraktion langer Zahlen"
140 PRINT STRING$(30,42)
150 :
160 DEFSTR A,B
170 PRINT:INPUT "Eingabe Minuend";A$
180 INPUT "Eingabe Subtrahend";B$
190 :
200 'Austausch und Auffüllen der Stellen
210 N=LEN(A$):M=LEN(B$)
220 IF M>N THEN SWAP A$,B$:SWAP N,M
230 IF N=M THEN 280
240 FOR I=1 TO N-M
250  B$="0"+B$
260 NEXT I
270 :
280 'Umwandlung in Zahlen
290 DEFINT I-K,U-W
300 DIM U(N),V(N),W(N)
310 FOR I=1 TO N
320   U(I)=VAL(MID$(A$,I,1))
330   V(I)=VAL(MID$(B$,I,1))
340 NEXT I
350 :
360 'stellenweise Subtraktion
370 J=N:K=0
380 WHILE J>0
390   W=U(J)-V(J)+K
400   K=INT(W/10):W(J)=W-K*10
410   J=J-1
420 WEND
430 W(0)=K
440 IF K=-1 THEN PRINT"Minuend
              < Subtrahend":END
450 :
460 PRINT:PRINT"Differenz =";
470 K=0
480 IF W(K)=0 THEN K=K+1:GOTO 480
490 FOR I=K TO N
500   PRINT W(I);
```

```
510 NEXT I
520 END
```

```
****************************
  Subtraktion langer Zahlen
****************************

Eingabe Minuend? 784103516910456
Eingabe Subtrahend? 9541217036428

Differenz = 7  7  4  5  6  2  2  9  9  8  7  4  0  2  8
```

3. Multiplikation langer Zahlen

Nicht ganz so einfach wie die Addition und Subtraktion ist die Multiplikation langer Zahlen.
Wird der größere Faktor stellenweise in das Feld U(I) und der zweite in V(I) zerlegt, so läßt sich die stellenweise Multiplikation wie folgt programmieren:

```
100  FOR J = M TO 1 STEP -1
110  IF V(J) = 0 THEN W(J) = 0: GOTO 190
120  K = 0
130  FOR I=N TO 1 STEP -1
140  T = U(I) * V(J) + W(I+J) + K
150  W(I+J) = T - INT(T/10) * 10
160  K = INT(T/10)
170  NEXT I
180  W(J) = K
190  NEXT J
```

Dabei ist M die Stellenzahl des größeren Faktors, N die des kleineren Faktors.
Im Feld W(I) wird die Ziffernfolge des Produkts gespeichert.
Ähnlich, wie bei der Addition und Subtraktion, könnte man auch die Multiplikation in Dreier- bzw. Viererblöcken durchführen.

Zum folgenden Programm

Das Programm berechnet nach dem oben angegebenen Verfahren das Produkt zweier langer Zahlen.

Als 1. Beispiel wird die Zahl

999999999999999

quadriert. Es ergibt sich

999999999999998000000000000001

Im 2. Beispiel werden die beiden Primfaktoren

59649589127497217

und

5704689200685129054721

der Zahl

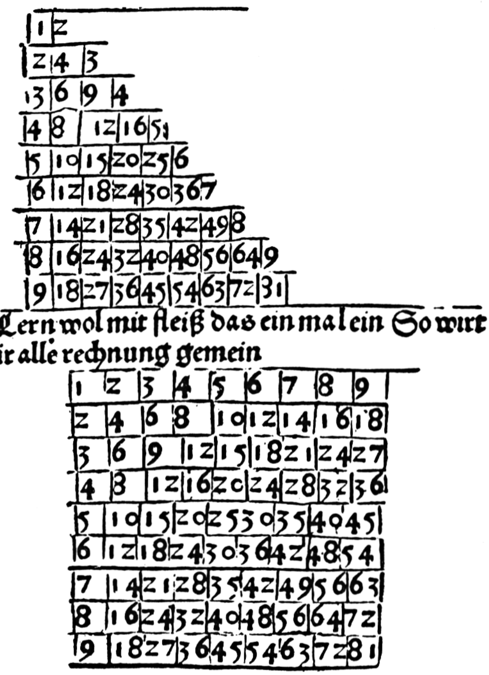
$$2^{128} + 1$$

miteinander multipliziert. Diese Primzahlfaktorisierung gelang erst 1970 den Amerikanern Morrison und Brillhart, obwohl man schon sehr lange wußte, daß diese Zahl keine Primzahl ist.

Das Programm liefert

340 82366 92093 84634 63374 60743 17682 11457

(vgl. Programmausdruck).



1	2								
2	4	3							
3	6	9	4						
4	8	12	16	5					
5	10	15	20	25	6				
6	12	18	24	30	36	7			
7	14	21	28	35	42	49	8		
8	16	24	32	40	48	56	64	9	
9	18	27	36	45	54	63	72	81	

Lern wol mit fleiß das ein mal ein So wirt dir alle rechnung gemein

1	2	3	4	5	6	7	8	9
2	4	6	8	10	12	14	16	18
3	6	9	12	15	18	21	24	27
4	8	12	16	20	24	28	32	36
5	10	15	20	25	30	35	40	45
6	12	18	24	30	36	42	48	54
7	14	21	28	35	42	49	56	63
8	16	24	32	40	48	56	64	72
9	18	27	36	45	54	63	72	81

"Multiplikationstafel aus der Arithmetik von Johannes Widmann, Leipzig (1489)"

```
100 REM Multiplikation langer Zahlen
110 :
120 CLS:PRINT STRING$(35,42)
130 PRINT "  Multiplikation langer Zahlen"
140 PRINT STRING$(35,42)
150 :
160 DEFSTR A,B
170 PRINT:INPUT "Eingabe 1.Faktor";A$
180 INPUT "Eingabe 2.Faktor";B$
190 :
200 'Austausch und Auffüllen der Stellen
210 N=LEN(A$):M=LEN(B$)
220 IF M>N THEN SWAP A$,B$:SWAP N,M
230 :
240 'Umwandlung in Zahlen
250 DEFINT I-K,U-W
260 DIM U(N),V(M),W(N+M)
270 FOR I=1 TO N
280   U(I)=VAL(MID$(A$,I,1))
290 NEXT I
300 FOR I=1 TO M
310   V(I)=VAL(MID$(B$,I,1))
320 NEXT I
330 FOR I=M+1 TO N+M
340   W(I)=0
350 NEXT I
360 :
370 'stellenweise Multiplikation
380 J=M
390 WHILE J>=0
400   IF V(J)=0 THEN W(J)=0:GOTO 470
410   I=N:K=0
420   WHILE I>=0
430     T=U(I)*V(J)+W(I+J)+K
440     K=INT(T/10):W(I+J)=T MOD 10
450     I=I-1
460   WEND
470   J=J-1
480 WEND
490 :
500 PRINT:PRINT"Produkt =";
```

```
510 K=1
520 IF W(K)=0 THEN K=K+1:GOTO 520
530 FOR I=K TO N+M
540   PRINT W(I);
550 NEXT I
560 END
```

```
***********************************
  Multiplikation langer Zahlen
***********************************

Eingabe 1.Faktor? 999999999999999
Eingabe 2.Faktor? 999999999999999

Produkt = 9  9  9  9  9  9  9  9  9  9  9  9  9  9  8  0  0  0  0  0  0  0  0
 0  0  0  0  0  0  1


Eingabe 1.Faktor? 5704689200685129054721
Eingabe 2.Faktor? 59649589127497217

Produkt = 3  4  0  2  8  2  3  6  6  9  2  0  9  3  8  4  6  3  4  6  3  3  7
 4  6  0  7  4  3  1  7  6  8  2  1  1  4  5  7
```

4. Division langer Zahlen

Wesentlich komplizierter als Addition und Multiplikation ist die Division langer Zahlen.
Das Programm führt eine stellenweise Division durch, entsprechend dem schriftlichen Rechnen. Zuerst wird geprüft, wie oft der Divisor in den Dividenden hineingeht. Sodann wird der Divisor mit dem entsprechenden Vielfachen multipliziert. Diese Multiplikation einer langen Zahl mit einem einstelligen Faktor wird bei Programm 5 erklärt. Das sich ergebende Produkt muß noch vom Dividenden subtrahiert werden. Der genaue Algorithmus kann dem folgenden Programm entnommen werden.

Zum folgenden Programm

Als Beispiel wird die schon aus dem Programm 3 bekannte Zahl

$$2^{128} + 1$$

durch den Primfaktor

5964958912749721 7

dividiert. Das Programm liefert den ganzzahligen Quotienten

57Ø46892ØØ685129Ø54721

```
100 REM Division langer Zahlen
110 :
120 CLS:PRINT STRING$(30,42)
130 PRINT"  Division langer Zahlen"
140 PRINT STRING$(30,42):PRINT
150 :
160 READ A$:PRINT"Dividend= ";A$:PRINT
170 READ B$:PRINT"Divisor= ";B$
180 :
190 M=LEN(A$):N=LEN(B$):M=M-N
200 IF M>=0 THEN 220
210 PRINT"Dividend < Divisor sein":END
220 DIM U(N+M),V(N),W(N),Q(M)
230 :
240 FOR I=1 TO N+M
250   U(I)=VAL(MID$(A$,I,1))
260 NEXT I
270 FOR I=1 TO N
280   V(I)=VAL(MID$(B$,I,1))
290 NEXT I
300 :
310 D=INT(10/(V(1)+1)):U(0)=0
320 IF D=1 THEN 450
330 K=0
340 FOR I=N+M TO 1 STEP -1
350   U=U(I)*D+K:K=INT(U/10):U(I)=U-K*10
360 NEXT I
370 U(0)=K
380 :
390 K=0
400 FOR I=N TO 1 STEP -1
410   V=V(I)*D+K:K=INT(V/10):V(I)=V-K*10
420 NEXT I
430 V(0)=K
440 :
450 J=0
460 IF U(J)=V(1) THEN Q=9:GOTO 490
470 IF V(1)=0 THEN PRINT"führ.Null":END
480 Q=INT((U(J)*10+U(J+1))/V(1))
490 Q1=(U(J)*10+U(J+1)-Q*V(1))*10+U(J+2)
500 IF V(2)*Q<=Q1 THEN 530
```

```
510 Q=Q-1:GOTO 490
520 :
530 K=0
540 FOR I=N TO 1 STEP -1
550   W=V(I)*Q+K:K=INT(W/10):W(I)=W-K*10
560 NEXT I
570 W(0)=K
580 :
590 K=0:F=0
600 FOR I=J+N TO J STEP -1
610   U=U(I)-W(I-J)+K:K=INT(U/10)
      :U(I)=U-K*10
620 NEXT I
630 IF K=-1 THEN F=1
640 :
650 Q(J)=Q
660 IF F=0 THEN 730
670 Q(J)=Q(J)-1:K=0
680 FOR I=J+N TO J STEP -1
690   U=U(I)+V(I-J)+K:K=INT(U/10)
      :U(I)=U-K*10
700 NEXT I
710 U(J-1)=U(J-1)+K
720 :
730 J=J+1
740 IF J<=M THEN 460
750 :
760 K=0
770 FOR I=M+1 TO M+N
780   U=U(I)+K*10:U(I)=INT(U/D)
      :K=U-INT(U/D)*D
790 NEXT I
800 :
810 REM Ergebnis
820 PRINT:PRINT"Quotient=" :J=0
830 IF Q(J)=0 THEN J=J+1:GOTO 830
840 FOR I=J TO M
850   PRINT Q(I);
860 NEXT I:PRINT".";
870 FOR I=M+1 TO N+M
880  PRINT U(I);
```

```
890 NEXT I
900 END
910 :
920 DATA 340282366920938463463374607431768211457
930 DATA 59649589127497217
```

```
******************************
  Division langer Zahlen
******************************

Dividend= 340282366920938463463374607431768211457

Divisor= 59649589127497217

Quotient=
 5  7  0  4  6  8  9  2  0  0  6  8  5  1  2  9  0  5  4  7  2  1 . 0  0  0
 0  0  0  0  0  0  0  0  0  0  0  0  0
```

5. Potenzieren

Eine weitere Anwendung der Mehr-Register-Arithmetik liefert das Potenzieren, da hier sehr schnell große Zahlen auftreten.
Das Potenzieren wird im Programm als wiederholte Multiplikation durchgeführt. Eine andere Möglichkeit wäre das logarithmische Rechnen, jedoch ist die mehrfachgenaue Berechnung von Logarithmen noch aufwendiger.

Teilt man die mit B zu multiplizierende Zahl in Achterblöcke R(I), so kann die Multiplikation wie folgt durchgeführt werden:

```
100  U = 0
110  FOR I=1 TO R
120  H = R(I) * B + U
130  IF H <= -1E8 THEN U = 0: GOTO 160
140  U = INT (H/1E8)
150  H = H - U * 1E8
160  R(I) = H
170  NEXT I
```

Dabei ist R die Anzahl der Achterblöcke und U der jeweilige Überlauf.

Soll die Zahl B zur N-ten Potenz erhoben werden, so muß das angegebene Programmstück noch in die Schleife

```
FOR K=1 TO N . . . . . NEXT K
```

eingebaut werden.

Die benötigte Anzahl der Achterblöcke wird logarithmisch berechnet: Der Zehnerlogarithmus der gesuchten Potenz B^N ist

```
N * LOG (B)/LOG (10)
```

Da der aufgerundete Zehnerlogarithmus einer Zahl die Stellenzahl angibt, kann daraus durch Division mit 8 die notwendige Registerzahl R berechnet werden.

Zum folgenden Programm

Das Programm berechnet nach dem oben beschriebenen Verfahren beliebige Potenzen mit einstelliger Basis.
Soll die Basis mehrstellig sein, so wird empfohlen die Registerzahl entsprechend zu vermindern.
Als Beispiel wird die hundertste Potenz von 2 berechnet. Es ergibt sich:

1267650 60022822 94014967 032053756

(vgl. Programmausdruck).

```
100 REM Berechnung grosser Potenzen
110 :
120 CLS:PRINT STRING$(35,42)
130 PRINT"     Berechnung hoher Potenzen"
140 PRINT STRING$(35,42)
150 :
160 PRINT:INPUT"Welche Basis";B
170 INPUT"Zu welcher Potenz";N
180 PRINT:PRINT B;"hoch ";N;"="
200 DIM R(100)
210 :
220 'Zahl der Fuenferbloecke
230 R=N*LOG(B)/LOG(10)
240 R%=INT(R/5)+1
250 :
260 'Multiplikationsschleife
270 U=0:R(1)=1
280 FOR K=1 TO N
290   FOR I=1 TO R%
300     H=R(I)*B+U
310     IF H<-100000  THEN U=0:GOTO 330
320     U=INT(H/100000 ):H=H-U*100000
330     R(I)=H
340   NEXT I
350 NEXT K
360 :
370 'Ausgabe
380 IF R(R%)=0 THEN R%=R%-1:GOTO 380
390 FOR I=R% TO 1 STEP -1
395   R$=STR$(R(I)):R$=RIGHT$(R$,LEN(R$)-1)
400   PRINT RIGHT$("0000"+R$,5);"   ";
410   IF (R%-I+1) MOD 10 =0 THEN PRINT
420 NEXT I
430 END
```

```
***********************************
    Berechnung hoher Potenzen
***********************************

Welche Basis? 2
Zu welcher Potenz? 100

 2 hoch  100 =
00001  26765  06002  28229  40149  67032  05376
```

6. FAKULTÄTS-BERECHNUNG

Die Berechnung der Fakultäts-Funktion liefert ebenfalls eine Anwendung der Mehr-Register-Arithmetik. Die Fakultätsfunktion n!, definiert als das Produkt aller natürlichen Zahlen von 1 bis n, wächst nämlich noch schneller als eine Potenzfunktion.
Die Fakultätsberechnung wird im Programm als wiederholte Multiplikation durchgeführt. Eine andere Möglichkeit wäre auch hier das logarithmische Rechnen, jedoch ist die mehrfachgenaue Berechnung von Logarithmen noch aufwendiger. Die Multiplikation erfolgt nach dem bei Programm 5 angegebenen Verfahren.
Komplizierter ist nur die Berechnung der benötigten Register. Sie erfolgt ebenfalls logarithmisch. Da der Logarithmus eines Produkts gleich der Summe der Logarithmen ist, ergibt sich log(n!) aus

```
100 FOR I=2 TO N
110 L=L+LOG(I)
120 NEXT I
```

Da die so ermittelte Zahl L, aufgerundet die Stellenzahl angibt, ist L/6 die benötigte Anzahl von Sechserblöcken. Da der Computer jedoch mit natürlichen Logarithmen rechnet, muß die Zahl zuvor über den Faktor

ln (10) = .4342945

auf den Zehnerlogarithmus umgerechnet werden.

Zum folgenden Programm

Das Programm berechnet die Fakultätsfunktion für dreistellige Zahlen in Fünferblöcken. Ist die Fakultät einer größeren Zahl gesucht, so wird aus Genauigkeitsgründen empfohlen mit Vierer- bzw. Dreierblöcken zu rechnen. Dazu müssen die im Programm auftretenden Sechsen entsprechend ersetzt werden.

Als Programmbeispiel wird

100 !

berechnet. Das Ergebnis kann dem Programmausdruck entnommen werden.

```
100 REM Fakultaet grosser Zahlen
110 :
120 CLS:PRINT STRING$(35,42)
130 PRINT"  Fakultaet grosser Zahlen "
140 PRINT STRING$(35,42)
150 :
160 PRINT:INPUT"Fakultaet welcher Zahl";N
170 PRINT:PRINT N;"!="
190 DIM R(100)
200 :
210 'Zahl der Fuenferbloecke
220 L=1:R(1)=1
230 FOR I=2 TO N
240 L=L+LOG(I)
250 NEXT I
260 L=L*.4342945:R%=L/5+1
270 :
280 FOR I=2 TO R%
290 R(I)=0
300 NEXT I
310 :
320 'Multiplikationsschleife
330 L=1
340 FOR I=N TO 2 STEP -1
350   L=L+LOG(I)*.434294575
360   L%=L/5+1
370   U=0
380   FOR J=1 TO L%
390     H=R(J)*I+U
400     IF H<-100000  THEN U=0:GOTO 420
410     U=INT(H/100000 ):H=H-U*100000
420     R(J)=H
430   NEXT J
440 NEXT I
450 :
460 'Ausgabe
470 IF R(R%)=0 THEN R%=R%-1:GOTO 470
480 FOR I=R% TO 1 STEP -1
490   R$=STR$(R(I)):R$=RIGHT$(R$,LEN(R$)-1)
500   PRINT RIGHT$("0000"+R$,5);"  ";
510   IF (R%-I+1) MOD 8 =0 THEN PRINT
520 NEXT I
530 END
```

```
510    IF  (R%-I+1)  MOD 8 =0 THEN PRINT
520 NEXT I
530 END
```

```
*********************************
  Fakultät großer Zahlen
*********************************

Fakultät welcher Zahl? 100

 100 !=
00933  26215  44394  41526  81699  23885  62667  00490
71596  82643  81621  46859  29638  95217  59999  32299
15608  94146  39761  56518  28625  36979  20827  22375
82511  85210  91686  40000  00000  00000  00000  00000
```

7. e AUF 1ØØØ STELLEN

Mit der Zahl π ist die Eulersche Zahl

$$e = 2.7182818\ldots$$

über die Gleichung

$$e^{\pi i} = -1$$

verbunden, dabei ist $i = \sqrt{-1}$.

e ist insbesondere als Basis der natürlichen Logarithmen von Bedeutung

$$y = \ln x \Rightarrow x = e^y$$

Ebenso stellt e den Grenzwert

$$e = \lim_{x \to \infty} \left(1 + \frac{1}{x}\right)^x$$

dar. Mit diesem Grenzwert ist auch die Zinzeszinsformel verknüpft: Wird 1 DM nicht jährlich oder monatlich, sondern kontinuierlich verzinst, so wächst der Zinseszins bei 100% nicht ins Unbeschränkte, sondern nähert sich dem Wert e DM. Die Reihenentwicklung der Exponentialfunktion liefert eine für numerische Zwecke brauchbare Formel für e

$$e = \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \frac{1}{4!} + \frac{1}{5!} + \frac{1}{6!} + \ldots.$$

dabei stellt n! die Fakultätsfunktion dar.

Erwähnenswert ist, daß auch e als Ergebnis eines Zufallsexperiments auftritt. Werden n Dinge zufällig angeordnet, so ist mit der Wahrscheinlichkeit

$$1/e = 0.3678794.$$

kein Gegenstand an seinem Ort. Solche Zufallsexperimente können mit Hilfe von Programm 12 erhalten werden. Der Versuch ist besser bekannt als das Problem der "vertauschten Briefe". Legt man n Briefe zufällig in n vorbereitete Umschläge, so liegt mit der Wahrscheinlichkeit

$$1 - 1/e = 0.6321205$$

mindestens ein Brief im richtigen Umschlag. Im Band 1 der vorliegenden Programmsammlung findet sich für n = 5 ein Grafikprogramm.

Zum folgenden Programm

Das Programm berechnet die Dezimalbruchentwicklung von e mit Hilfe der angegebenen Reihenentwicklung. Die Kehrwerte der benötigten Fakultäten werden durch wiederholte Divisionen berechnet und fortlaufend aufsummiert.

Das hier angegebene Programm läßt sich auf beliebige Stellenzahl ausdehnen. Die Variable N in Zeile 220 muß dazu so geändert werden, daß die Dezimalstellen von 1/N! bei der gewünschten Genauigkeit vernachlässigt werden können.

```
100 REM e auf 1000 Stellen
110 :
120 CLS:PRINT STRING$(30,42)
130 PRINT "     e auf 1000 Stellen"
140 PRINT STRING$(30,42)
150 :
160 DIM A(202):'Zahl der Fuenferbloecke
170 FOR I=1 TO 202
180   A(I)=0
190 NEXT I
200 :
210 B=100000 :A(0)=1
220 FOR N=450 TO 1 STEP -1
230   FOR I=0 TO 201
240     IF A(I)=0 THEN 280
250     Q=INT(A(I)/N):R=A(I)-Q*N
260     A(I)=Q:A(I+1)=A(I+1)+B*R
270   NEXT I
280   A(0)=A(0)+1
290 NEXT N
300 :
310 'Rundung
320 A(201)=A(201)+INT(A(202)/B+.5)
330 FOR I=200 TO 1 STEP -1
340   U=INT(A(I)/B):A(I)=A(I)-B*U
350   A(I-1)=A(I-1)+U
360 NEXT I
370 :
380 PRINT"e=";A(0);"."
390 FOR I=1 TO 200
400   A$=STR$(A(I)):A$=RIGHT$(A$,LEN(A$)-1)
410   PRINT RIGHT$("0000"+A$,5);"  ";
420   IF I MOD 8 =0 THEN PRINT
430 NEXT I
440 END
```

****
e auf 1000 Stellen
****
e= 2 .

71828	18284	59045	23536	02874	71352
66249	77572	47093	69995	95749	66967
62772	40766	30353	54759	45713	82178
52516	64274	27466	39193	20030	59921
81741	35966	29043	57290	03342	95260
59563	07381	32328	62794	34907	63233
82988	07531	95251	01901	15738	34187
93070	21540	89149	93488	41675	09244
76146	06680	82264	80016	84774	11853
74234	54424	37107	53907	77449	92069
55170	27618	38606	26133	13845	83000
75204	49338	26560	28864	68800	72705
71742	64364	22912	59039	54818	65870
31264	10457	49669	34792	29434	01033
74639	85002	03510	59931	78084	20832
22517	64866	71751	87375	78427	63076
80122	36490	02456	95587	91367	18709
99960	71749	18641	04258	68953	74226
37605	56454	08460	42957	75390	51037
43587	90914	02547	48265	73137	70140
32649	37577	68259	40475	49435	58471
54657	04140	40868	51052	63327	79621
13658	42489	90958	78661	05465	30091
74336	12838	01761	73296	06216	23923
00965	92466	98058	07785	54983	78516
41272	07192	10263	06389	90530	75586
98035	67741	57062	08405	90325	33337
36086	65511	16056	38466	32017	85929
31994	92993	48212	68436	63182	49742
19923	37104	37136	11443	27646	39803
65791	61003	20703	24716	31157	66078
92931	94953	77431	41698	42363	91701
14315	74683	58961	40066	05718	45316
49749	84728				

8. LINEARE DIOPHANTISCHE GLEICHUNG

Ein Standardverfahren zur Lösung einer linearen Diophantischen Gleichung

$$ax + by = c$$

gleichbedeutend mit der Kongruenz

$$ax = c \bmod y$$

ist der erweiterte Euklidsche Algorithmus. Mit seiner Hilfe kann der größte gemeinsame Teiler

$$\text{ggT}(a,b)$$

als Linearkombination von a und b bestimmt werden.

Dies soll an einem Beispiel für a = 286 und b = 121 gezeigt werden:

$$\begin{aligned}
262 : 121 &= 2 \text{ Rest } 44 \Rightarrow 286 = 2 \cdot 121 + 44 \\
121 : 44 &= 2 \text{ Rest } 33 \Rightarrow 121 = 2 \cdot 44 + 33 \\
44 : 33 &= 1 \text{ Rest } 11 \Rightarrow 44 = 1 \cdot 33 + 11 \\
33 : 11 &= 3 \qquad\qquad \Rightarrow 33 = 3 \cdot 11
\end{aligned}$$

Somit gilt

$$\text{ggT}(286, 121) = 11$$

Durch Rückwärtsrechnen folgt

$$\begin{aligned}
11 &= 44 - 33 \\
&= 44 - (121 - 2 \cdot 44) \\
&= 3 \cdot 44 - 121 \\
&= 3 \cdot (286 - 2 \cdot 121) - 121 \\
&= 3 \cdot 286 - 7 \cdot 121
\end{aligned}$$

Dies liefert die gesuchte Linearkombination

$$3 \cdot 286 - 7 \cdot 121 = 11$$

allgemein

$$ax + by = \text{ggT}(a,b)$$

Sind x_0, y_0 die Koeffizienten der Linearkombination, so hat die Diophantische Gleichung

$$ax + by = c$$

die allgemeine Lösung:

$$x = cx_0 + \frac{a}{\text{ggT}(a,b)} \cdot t$$

$$y = cy_0 - \frac{b}{\text{ggT}(a,b)} \cdot t; \quad t \text{ ganzzahlig}$$

Diese Lösung existiert nur, wenn c ein Vielfaches von ggT (a,b) ist; andernfalls ist die Gleichung nicht lösbar (vgl. [7]).

Zum folgenden Programm

Das Programm berechnet mit Hilfe des erweiterten Euklidschen Algorithmus nach dem oben angegebenen Verfahren eine spezielle Lösung der Gleichung

$$ax + by = ggT\ (a,b).$$

Damit wird die allgemeine Lösung bestimmt.

Das im Programm angegebene Verfahren kann prinzipiell auch für Diophantische Gleichungen der Form

$$ax - by = c$$

($b > 0$) angewandt werden. In diesem Fall ergibt sich für den ggT ein negativer Wert, im Gegensatz zur üblichen Definition; zwei aufeinander stoßende negative Vorzeichen sind als positiv zu lesen.

Als Beispiel wird die Diophantische Gleichung

$$17x + 6y = 5$$

gelöst. Das Programm liefert die Parameterlösung

$$x = -5 - 6t$$
$$y = 15 + 17t$$

dabei durchläuft t alle ganzen Zahlen. Für jeden Wert von t ergibt sich wieder eine spezielle Lösung, z.B. für

$$t = 0 : x = -5,\ y = 15$$
$$t = 1 : x = -11,\ y = 32$$
$$t = -1 : x = 1,\ y = -2$$

usw. Zahlreiche Diophantische Gleichungen findet man in eingekleideter Form in den historischen Aufgaben des Anhangs.

```
100 REM Lineare Diophantische Gleichung
110 :
120 CLS:PRINT STRING$(30,42)
130 PRINT" Lin.Diophant.Gleichung"
140 PRINT STRING$(30,42)
150 :
160 DEFINT A-C,X-Y
170 PRINT"a*x + b*y = c"
180 INPUT"Eingabe a,b,c";A,B,C
190 :
200 REM Initialisieren
210 A0=A:A1=B:X0=1
220 X1=0:Y0=0:Y1=1
230 :
240 REM Erweiteter Euklidscher Algorithmus
250 Q=INT(A0/A1)
260 IF Q=A0/A1 THEN 320
270 A2=A0-Q*A1:X2=X0-Q*X1:Y2=Y0-Q*Y1
280 A0=A1:A1=A2:X0=X1:X1=X2
290 Y0=Y1:Y1=Y2:GOTO 250
300 :
310 REM
320 PRINT"ggT(";A;",";B;")=";A2:PRINT
330 PRINT"Loesung der Gleichung";
340 PRINT" a*x+b*y=ggT(a,b)"
350 IF Y2<0 THEN 370
360 PRINT TAB(8)A;"*";X2;"+";B;"*";Y2;
          "=";A2:GOTO 380
370 PRINT TAB(8)A;"*";X2,"-";B;"*";
          ABS(Y2);"=";A2
380 :
390 IF C/A2<>INT(C/A2) THEN PRINT
       "existiert nicht":END
400 PRINT
410 PRINT"Allg.Loesung der Diophant.
          Gleichung:"
420 PRINT TAB(8)"x=";C*X2/A2;"-t*"B
430 PRINT TAB(8)"y=";C*Y2/A2;"+t*";A;
440 PRINT SPC(5)"t ganzzahlig"
450 END
```

```
*******************************
 Lin.Diophant.Gleichung
*******************************
a*x  + b*y  = c
Eingabe a,b,c? 17,6,5

ggT( 17  ,  6 )= 1

Lösung der Gleichung a*x+b*y=ggT(a,b)
        17 *-1  + 6  *  3  =  1

Allg.Lösung der Diophant.
Gleichung:
       x=-5 -t* 6
       y= 15 +t* 17       t ganzzahlig
```

9. KETTENBRUCH / BRUCHAPPROXIMATION

Sucht man zu einem vorgegebenen Dezimalbruch einen Quotienten gleichen Werts – z.B. ein Windungszahlverhältnis beim Transformator – so kann dieser mit Hilfe eines Kettenbruchs bestimmt werden.

Ein Kettenbruch hat folgende Form

$$e = 2 + \cfrac{1}{1 + \cfrac{1}{2 + \cfrac{2}{3 + \cfrac{3}{4 + \cfrac{4}{\dots}}}}}$$

Solche Kettenbrüche können für rationale, d.h. für Bruchzahlen durch den Euklidschen Algorithmus berechnet werden.

Für 67/29 ergibt sich z.B.

$$\begin{aligned} 67 &: 29 = 2 \text{ Rest } 9 \\ 29 &: 9 = 3 \text{ Rest } 2 \\ 9 &: 2 = 4 \text{ Rest } 1 \\ 2 &: 1 = 2 \end{aligned}$$

Somit folgt

$$\frac{67}{29} = 2 + \frac{9}{29} = 2 + \cfrac{1}{\cfrac{29}{9}}$$

$$\frac{29}{9} = 3 + \frac{2}{9} = 3 + \cfrac{1}{\cfrac{9}{2}}$$

$$\frac{9}{2} = 4 + \frac{1}{2}$$

Setzt man diese Gleichungen rückwärts ineinander, so ergibt sich der Kettenbruch

$$\frac{67}{29} = 2 + \cfrac{1}{3 + \cfrac{1}{4 + \cfrac{1}{2}}}$$

Der Euklidsche Algorithmus zeigt, daß der Kettenbruch einer rationalen Zahl abbricht, jedoch nicht der einer irrationalen Zahl. Ein Beweis dazu findet sich in [14]. Kettenbrüche können somit durch folgendes Programm berechnet werden

```
100 INPUT "DEZIMALBRUCH"; X
110 PRINT INT (X)
120 IF X=INT(X) THEN 150
130 X=1/(X-INT(X))
140 GOTO 110
150 END
```

jedoch muß dabei noch die beschränkte Genauigkeit des Rechners beachtet werden.

Zum folgenden Programm

Das folgende Programm berechnet zu einem eingegebenen Dezimalbruch maximal 8 Teilnenner nach dem oben angegebenen Programmstück. Dabei wird versucht Rechnerungenauigkeiten, so weit wie möglich, abzufangen.

Der so berechnete Kettenbruch wird auf dem Bildschirm ausgegeben und die zugehörige rationale Zahl berechnet.

Als Programmbeispiel wurde

$$\pi = 3.14159265$$

eingegeben. Man erhält die Bruchapproximation

$$\frac{2551690445}{80115557}$$

Der ausgegebene Kettenbruch kann dem Programmausdruck entnommen werden.

```
100 REM Kettenbruch/Bruchapproximation
110 :
120 CLS:PRINT STRING$(35,42)
130 PRINT" Kettenbruch/Bruchapproximation"
140 PRINT STRING$(35,42)
150 :
160 PRINT:INPUT"Dezimalzahl";Y
170 DIM R(12)
180 :
210 K=8
220 FOR I=1 TO K
230   R(I)=INT(Y+.000001)
240   IF ABS(R(I)-Y)<.000001 THEN 280
250   Y=1/(Y-R(I))
260 NEXT I:GOTO 290
270 :
280 K=I
290 PRINT R(1);" + 1"
300 FOR I=2 TO K-1
310   PRINT TAB(I*6-7);" --------"
320   PRINT TAB(I*6-6);R(I);" + 1"
330 NEXT I
340 PRINT TAB(K*6-7);" -----"
350 PRINT TAB(K*6-6);R(K)
360 :
370 Z=R(K):N=1
380 FOR J=K-1 TO 1 STEP -1
390   H=Z:Z=R(J)*Z+N:N=H
400 NEXT J
410 PRINT Z
420 PRINT"=";STRING$(LEN(STR$(Z)),45);"=";
430 PRINT Z/N
440 PRINT N
450 END
```

```
 Kettenbruch/Bruchapproximation
************************************

Dezimalzahl? 3.14159265
 3   + 1
     -------
      7   + 1
           -------
            15   + 1
                 -------
                  1   + 1
                       -------
                        288   + 1
                              -------
                               2   + 1
                                    -------
                                     4   + 1
                                          ---
                                          272
 251690445
=---------------= 3.14159265
 80115557
```

10. PERMUTATIONEN

Da endliche n–elementige Mengen stets auf die Menge

$$\{1, 2, 3, \ldots, n\}$$

abgebildet werden können, arbeiten die folgenden Kombinatorik-Programme stets mit dieser Anfangsmenge der natürlichen Zahlen.
Jede Anordnung der Menge 1, 2, 3, . . ., n, bei der jedes Element genau einmal vorkommt, heißt Permutation.

Permutationen treten vielfach in der Kombinatorik und Wahrscheinlichkeitsrechnung auf. Man benötigt sie insbesondere auch in Operations Research, wenn man z.B. auf der Suche nach der kürzesten Rundreise alle Anordnungen der zu besuchenden Städte durchmustern muß.

Es gibt

$$n!$$

Permutationen einer n–elementigen Menge; dabei ist n! die in Programm 6 behandelte Fakultätsfunktion.

Die Fakultäten der Zahlen wachsen sehr schnell, so gilt

6! = 72Ø, 7! = 5Ø4Ø, 8! = 4Ø32Ø
9! = 36288Ø, 1Ø! = 36288ØØ
11! = 399168ØØ

Somit gibt es für 10 Leute bei 10 Sitzplätzen

36288ØØ

verschiedene Sitzordnungen. Spielt in einer Fußballmannschaft jeder auf jeder Position, so gibt es

399168ØØ

verschiedene Mannschaftsaufstellungen.

Zu folgendem Programm

Das folgende Programm erzeugt nach einem Algorithmus von Ord-Smith die Permutation einer n–elementigen Menge in lexikographischer Anordnung.

Dabei heißen zwei Zahlenfolgen a_i und b_i lexikographisch geordnet, wenn für die erste Stelle mit

$$a_i \neq b_i$$

gilt

$$a_i \leqslant b_i$$

Der Programmausdruck zeigt die Permutation einer 4–elementigen Menge.

```
100 REM Permutationen
110 :
120 CLS:PRINT STRING$(25,42)
130 PRINT"      Permutationen"
140 PRINT STRING$(25,42)
150 :
160 INPUT"Wieviele Elemente 1..n";N:PRINT
170 DIM X(N),Y(N)
180 :
190 REM Initialisieren
200 FOR I=1 TO N
210     X(I)=I
220 NEXT I
230 F=0:P=0:N1=N-1
240 :
250 REM wiederholen bis Flag f wieder =0
260     FOR I=1 TO N
270         PRINT X(I);
280     NEXT I:PRINT
290     P=P+1
300     GOSUB 350:'Unterprogramm Permutation
310 IF F=1 THEN 250
320 PRINT P;"Permutationen"
330 END
340 :
350 REM Rekursive Erzeugung der Permutationen
360 IF F=1 THEN 410
370 F=1
380 FOR K=1 TO N
390   Y(K)=N
400 NEXT K
410 IF Y(N1)<>N THEN 450
420 Y(N1)=N1
430 H=X(N):X(N)=X(N1):X(N1)=H:GOTO 600
440 :
450 FOR J=1 TO N1
460     K=N-J
470     IF Y(K)<>K THEN 520
480     Y(K)=N
490 NEXT J
500 F=0:K=1:GOTO 550
```

```
510 :
520 M=Y(K)
530 H=X(M):X(M)=X(K):X(K)=H
540 Y(K)=M-1:K=K+1
550 M=N
560 :
570 H=X(M):X(M)=X(K):X(K)=H
580 M=M-1:K=K+1
590 IF K<M THEN 570
600 RETURN
```

```
**************************
      Permutationen
**************************
Wieviele Elemente 1..n? 4

 1   2   3   4
 1   2   4   3
 1   3   2   4
 1   3   4   2
 1   4   2   3
 1   4   3   2
 2   1   3   4
 2   1   4   3
 2   3   1   4
 2   3   4   1
 2   4   1   3
 2   4   3   1
 3   1   2   4
 3   1   4   2
 3   2   1   4
 3   2   4   1
 3   4   1   2
 3   4   2   1
 4   1   2   3
 4   1   3   2
 4   2   1   3
 4   2   3   1
 4   3   1   2
 4   3   2   1
 24 Permutationen
```

11. ZUFALLSPERMUTATION

Für viele Zwecke benötigt man zufällige Permutationen, z.B. für die bei Programm 8 erwähnten Monte-Carlo-Simulation der vertauschten Briefe.

Als Anfangswert wird die Permutation P(I) der Zahlen 1, . ., n auf die natürliche Reihenfolge gesetzt.

Sodann werden n–1 ganzzahlige Zufallszahlen ausgelost und die Permutation mit entsprechendem Index ausgetauscht. Dies wird durch folgendes Programmstück durchgeführt.

```
260 FOR J=N TO 2 STEP -1
270 R = INT (J * RND(1)) + 1
280 H = P(J) : P(J) = P(R) : P(R) = H
290 NEXT J
```

Die so erhaltene Zufallspermutation wird ausgedruckt und bei Bedarf eine neue erzeugt.

Zum folgenden Programm

Nach Abfrage, wieviele Zahlen die Permutation umfassen soll und wieviele Permutationen erwünscht sind, werden nach dem angegebenen Verfahren mit Hilfe des eingebauten Zufallszahlengenerators die benötigten zufälligen Permutationen erzeugt.
Der Programmausdruck zeigt 20 Zufallspermutationen der Zahlen 1 bis 9.

```
100 REM Zufallspermutation
110 :
120 CLS:PRINT STRING$(25,42)
130 PRINT"  Zufallspermutationen"
140 PRINT STRING$(25,42)
150 :
160 INPUT"Wieviele Zahlen";N
170 INPUT"Wieviele Permutationen";M:PRINT
180 DIM P(N)
190 :
200 REM Initialisieren
210 I=1
220 :
230 RANDOMIZE TIMER
240 WHILE I<=M
250   FOR J=1 TO N
260     P(J)=J
270   NEXT J
280   :
290   FOR J=N TO 2 STEP -1
300     R=INT(J*RND)+1:'Auslosen der Reihenfolge
310     H=P(J):P(J)=P(R):P(R)=H
320   NEXT J
330   FOR J=1 TO N
340     PRINT P(J);
350   NEXT J:PRINT
360   I=I+1
370 WEND
380 END
```

```
**************************
  Zufallspermutationen
**************************
Wieviele Zahlen ? 9
Wieviele Permutationen ? 20

 2  3  8  6  9  4  5  1  7
 4  3  5  6  7  1  8  9  2
 8  3  4  5  9  2  1  6  7
 5  4  2  6  7  9  8  1  3
 1  3  8  6  5  7  2  9  4
 8  5  6  7  9  3  2  1  4
 4  8  5  2  7  9  6  1  3
 8  3  9  1  2  5  7  4  6
 1  4  5  6  9  3  2  7  8
 4  5  8  3  7  6  2  9  1
 5  7  2  9  4  6  3  8  1
 1  9  2  4  5  6  8  7  3
 7  3  4  6  8  9  5  2  1
 7  3  8  6  9  5  4  2  1
 7  1  5  4  3  6  8  9  2
 4  6  7  1  9  8  3  2  5
 6  4  5  9  8  3  7  2  1
 8  4  7  5  9  2  3  1  6
 9  3  2  4  1  7  8  5  6
 9  8  5  2  6  7  4  3  1
```

12. KOMBINATIONEN

Ordnet man nur k Elemente einer n-elementigen Menge ohne Wiederholung an, so erhält man eine k-Kombination oder auch Kombination k.-ter Klasse.

Die Anzahl der k-Kombinationen einer n-elementigen Menge schreibt man

$$\binom{n}{k}$$

und bezeichnet dies als Binomialkoeffizient. Diese tragen ihren Namen daher, weil sie als Koeffizienten in den binomischen Formeln, z.B.

$$(a+b)^3 = \binom{3}{0}a^3 + \binom{3}{1}a^2b + \binom{3}{2}ab^2 + \binom{3}{3}b^3$$
$$= a^3 + 3a^2b + 3ab^2 + b^3$$

auftreten.

Die Binomialkoeffizienten $\binom{n}{i}$ $i = 0, 1, 2, \ldots, n$ finden sich jeweils in der (n+1)-ten Zeile des Pascalschen Dreiecks:

$$\begin{array}{ccccccccccccc}
 & & & & & & 1 & & & & & & \\
 & & & & & 1 & & 1 & & & & & \\
 & & & & 1 & & 2 & & 1 & & & & \\
 & & & 1 & & 3 & & 3 & & 1 & & & \\
 & & 1 & & 4 & & 6 & & 4 & & 1 & & \\
 & 1 & & 5 & & 10 & & 10 & & 5 & & 1 & \\
1 & & 6 & & 15 & & 20 & & 15 & & 20 & & 1
\end{array}$$

Die Binomialkoeffizienten können auch über die Fakultätsfunktion berechnet werden

$$\binom{n}{k} = \frac{n!}{k!\,(n-k)!}$$

So kann beim Schafkopf (24 Karten, 3 Spieler) jeder Spieler

$$\binom{24}{8} = 735471$$

Blätter haben. Beim Lottospiel gibt es

$$\binom{49}{6} = 13983816$$

verschiedene Lottotips, beim neuen Mittwochslotto entsprechend

$$\binom{38}{7} = 12620256$$

Zum folgenden Programm

Das Programm erzeugt nach einem Algorithmus von Kurtzberg alle k-Kombinationen (oder k-ten Klasse) einer n-elementigen Menge.

Der Programmausdruck zeigt die

4-Kombinationen

einer 6-elementigen Menge.


Eyn Newe
Unnd wolgegründte
underweysung aller Kauffmanß Rech-
nung in dreyen Büchern/mit schönen Re
geln vñ fragstucken begriffen. Sunder-
lich was fortl unnd behendigkait in der
Welschē Practica vñ Tolletn gebraucht
wirdt/ des gleychen fürmalß weder in
Teützscher noch in Welscher sprach nie
gedrückt. durch Petrum Apianū
von Leyßnick/d Astronomei
zu Ingolstat Ordina-
riū/ verfertiger.

"Titelblatt der Arithmetik von Petrus Apianus, Ingolstadt 1527
mit der ersten gedruckten Darstellung des Pascalschen Dreiecks"

```
100 REM Kombinatioen
110 :
120 CLS:PRINT STRING$(25,42)
130 PRINT"    Kombinationen"
140 PRINT STRING$(25,42)
150 :
160 INPUT"Wieviele Elemente";N
170 INPUT"Zu welcher Klasse";M
180 DIM X(M)
190 :
200 REM Initialisieren
210 FOR I=1 TO M
220     X(I)=N
230 NEXT I
240 L=0:PRINT
250 :
260 REM Binomialkoeffizient m aus n
270 P=1
280 IF M=0 THEN 340
290 FOR I=1 TO M
300     P=P*(N-I+1)/I
310 NEXT I
320 :
330 WHILE L<P
340     GOSUB 440:'Kombinationen
350     FOR I=1 TO M
360     PRINT X(I);
370     NEXT I:PRINT
380     L=L+1
390 WEND:PRINT
400 PRINT P;"Kombinationen"
410 END
420 :
430 REM Rekursion
440 FOR K=1 TO M
450     J=K-1:A=M-J:B=N-J
460     IF X(A)<B THEN 530
470 NEXT K
480 :
490 FOR K=1 TO M
500     X(K)=K
```

```
510 NEXT K:GOTO 570
520 :
530 B=X(A)
540 FOR K=A TO M
550     B=B+1:X(K)=B
560 NEXT K
570 RETURN
```

```
**************************
   Kombinationen
**************************
Wieviele Elemente? 6
Zu welcher Klasse? 4

 1  2  3  4
 1  2  3  5
 1  2  3  6
 1  2  4  5
 1  2  4  6
 1  2  5  6
 1  3  4  5
 1  3  4  6
 1  3  5  6
 1  4  5  6
 2  3  4  5
 2  3  4  6
 2  3  5  6
 2  4  5  6
 3  4  5  6
 15 Kombinationen
```

13. VARIATIONEN

Ordnet man k Elemente einer n-elementigen Menge unter Berücksichtigung der Reihenfolge an, so erhält man eine k-Variation oder Variation zu k-ten Klasse.
Es gibt

$$\frac{n!}{(n-k)!}$$

k-Variationen einer n-elementigen Menge.

Wollen von 20 Personen 8 Leute an einem bestimmten Tisch sitzen, so gibt es dort

$$\frac{2Ø!}{12!} = 5Ø7911Ø4ØØ$$

verschiedene Sitzordnungen. Aus einem Gremium von 30 Personen gibt es

$$\frac{3Ø!}{27!} = 2436Ø$$

verschiedene Möglichkeiten einen Präsidenten, dessen 1. und 2. Stellvertreter auszuwählen.

Zum folgenden Programm

Das Programm erzeugt alle k-Variationen oder Variationen zur k.-ten Klasse einer n-elementigen Menge.

Der Programmausdruck zeigt alle 3 Variationen einer 4-elementigen Menge.

```
100 REM Variationen
110 :
120 CLS:PRINT STRING$(25,42)
130 PRINT"      Variationen"
140 PRINT STRING$(25,42)
150 :
160 INPUT"Wieviele Elemente";N
170 INPUT"zu welcher Klasse";K:PRINT
180 DIM X(N),Y(K)
190 :
200 REM Initialisieren
210 V=0:M=K
220 IF K=N THEN M=M-1
230 FOR I=1 TO N
240   X(I)=I
250 NEXT I
260 FOR I=1 TO M
270   Y(I)=I
280 NEXT I
290 :
300 I=M
310 FOR J=1 TO K
320   PRINT X(J);
330 NEXT J:PRINT
340 V=V+1
350 WHILE I>0
360   IF Y(I)>=N THEN 400
370   Y(I)=Y(I)+1
380   H=X(I):X(I)=X(Y(I)):X(Y(I))=H:GOTO 300
390   :
400   H=X(I):X(I)=X(Y(I)):X(Y(I))=H
410   Y(I)=Y(I)-1
420   IF Y(I)>I THEN 400
430   I=I-1
440 WEND
450 PRINT:PRINT V; "Variationen"
460 END
```

```
***************************
      Variationen
***************************
Wieviele Elemente ? 4
zu welcher Klasse ? 3

 1  2  3
 1  2  4
 1  3  2
 1  3  4
 1  4  2
 1  4  3
 2  1  3
 2  1  4
 2  3  1
 2  3  4
 2  4  1
 2  4  3
 3  1  2
 3  1  4
 3  2  1
 3  2  4
 3  4  1
 3  4  2
 4  1  2
 4  1  3
 4  2  1
 4  2  3
 4  3  1
 4  3  2

 24 Variationen
```

14. Ø1-TUPEL

Wählt man k Elemente aus einer n-elementigen Menge aus und läßt dabei beliebige Wiederholung zu, so erhält man die

k-tupel

der Menge. Im Fall von k=2 spricht man auch von Paaren, bei k=3 von Tripel.

Es gibt insgesamt

$$n^k$$

k-Tupel einer n-elementigen Menge.

Besondere Bedeutung haben die 8-Tupel der Binärzahlen Ø und 1, Byte genannt, für Computer. Alle ganzen Zahlen werden mit Hilfe zweier Bytes dargestellt. Da das erste Byte das Vorzeichen angibt, können somit INTEGER-Zahlen mit 15 Bit dargestellt werden. Es sind dies die Zahl Ø bis 32767.

Weitere Anwendungen finden die Ø1-Tupel, da mit ihrer Hilfe alle Teilmengen einer gegebenen Menge erzeugt werden können: Ist die i.-te Stelle des Ø1-Tupel Eins, so gehört das Element i zur entsprechenden Teilmenge, sonst nicht.

Somit hat eine n-elementige Menge soviele Teilmengen, wie der Ø1-Tupel der Länge n gibt, nämlich

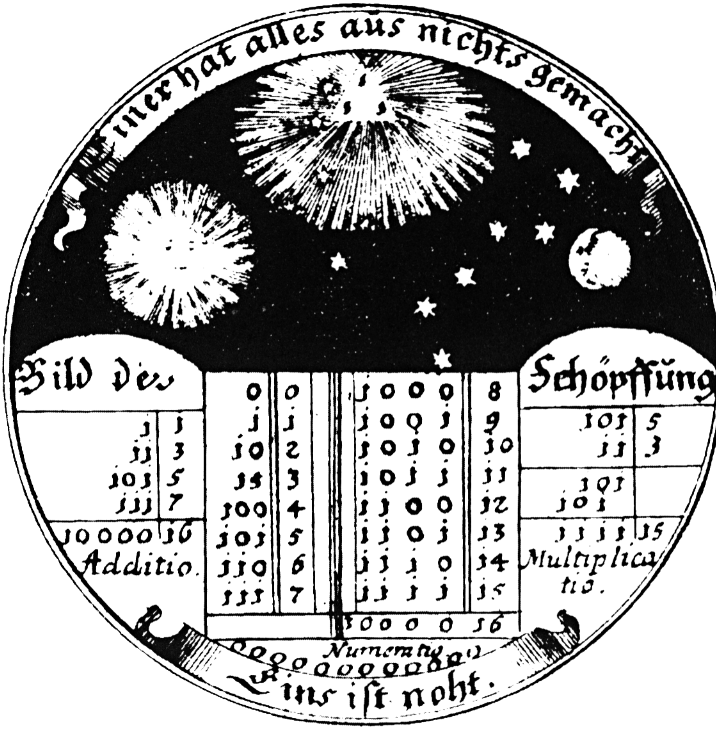
$$2^n$$

Wichtig sind Ø1-Tupel auch in der Boole'schen Algebra und Aussagenlogik, da mit ihrer Hilfe alle Wahrheitswerte von Aussageverknüpfungen ermittelt werden können (siehe Logikprogramme im Band 1 der vorliegenden Programmsammlung). Allerdings wird im IBM-BASIC "wahr" nicht als 1, sondern als −1 codiert; dies kann aber im Programm durch eine Vorzeichenänderung erreicht werden.

Zum folgenden Programm

Das Programm erzeugt nach einem Algorithmus von Berztiss alle Ø1-Tupel der gewünschten Länge.
Läßt man in Zeile 270 statt X(K) das Element A(K) einer Menge – falls X(K) = 1 – ausdrucken, so erhält man alle Teilmengen dieser Menge.

Der Programmausdruck zeigt alle Ø1-Tupel der Länge 5.



"Entwurf von Leibniz für eine Gedenkmünze, die das von ihm gefundene Binärsystem darstellt (1697)"

```
100 REM 01-Tupel
110 :
120 CLS:PRINT STRING$(25,42)
130 PRINT"    01-Tupel"
140 PRINT STRING$(25,42)
150 :
160 PRINT:INPUT"Laenge der 01-tupel";N
170 DIM X(N+1)
180 :
190 REM Initialisieren
200 FOR K=1 TO N
210   X(K)=0
220 NEXT K
230 J=INT(2^N+.00001):L=1
240 :
250 WHILE L<=J
260   FOR K=1 TO N
270     PRINT X(K);
280   NEXT K:PRINT
290   GOSUB 350:'Unterprogramm
300   L=L+1
310 WEND
320 PRINT:PRINT J;" 01-Tupel"
330 END
340 :
350 REM rekursive Erzeugung der 01-Tupel
360 I=1
370 IF X(I)<>0 THEN 390
380 X(I)=1:GOTO 400
390 X(I)=0:I=I+1:GOTO 370
400 RETURN
```

```
***************************
   01-Tupel
***************************
Länge der 01-tupel ? 5

 0   0   0   0   0
 1   0   0   0   0
 0   1   0   0   0
 1   1   0   0   0
 0   0   1   0   0
 1   0   1   0   0
 0   1   1   0   0
 1   1   1   0   0
 0   0   0   1   0
 1   0   0   1   0
 0   1   0   1   0
 1   1   0   1   0
 0   0   1   1   0
 1   0   1   1   0
 0   1   1   1   0
 1   1   1   1   0
 0   0   0   0   1
 1   0   0   0   1
 0   1   0   0   1
 1   1   0   0   1
 0   0   1   0   1
 1   0   1   0   1
 0   1   1   0   1
 1   1   1   0   1
 0   0   0   1   1
 1   0   0   1   1
 0   1   0   1   1
 1   1   0   1   1
 0   0   1   1   1
 1   0   1   1   1
 0   1   1   1   1

 32  01-Tupel
```

15. PARTITIONEN

Unter einer Partition der Zahl n versteht man die Zerlegung von n in Summanden, wobei es auf die Anordnung nicht ankommt. So hat die Zahl 6 folgende 11 Partitionen:

$$
\begin{aligned}
6 &= 6 \\
&= 5 + 1 \\
&= 4 + 2 \\
&= 4 + 1 + 1 \\
&= 3 + 3 \\
&= 3 + 2 + 1 \\
&= 3 + 1 + 1 + 1 \\
&= 2 + 2 + 2 \\
&= 2 + 2 + 1 + 1 \\
&= 2 + 1 + 1 + 1 + 1 \\
&= 1 + 1 + 1 + 1 + 1 + 1
\end{aligned}
$$

Solche Partitionen werden benötigt bei Geldwechsel- und Frankatur-Problemen.

Zum folgenden Programm

Das Programm erzeugt sämtliche Partitionen einer Zahl nach einem Algorithmus von McKay.

Der Programmausdruck zeigt die 42 Partitionen der Zahl 1Ø.

```
100 REM Partitionen
110 :
120 CLS:PRINT STRING$(25,42)
130 PRINT"   Partitionen"
140 PRINT STRING$(25,42)
150 :
160 INPUT"Welche Zahl";N
170 DIM P(N)
180 PRINT"Die Partitionen von";N;" sind:"
190 :
200 REM Initialisieren
210 P(1)=N:K=1:F=0:Z=0
220 :
230 REM Wiederholen bis Flag f=0
240 GOSUB 330:'Unterprogramm
250 FOR J=1 TO K
260   PRINT P(J);
270 NEXT J:PRINT
280 Z=Z+1
290 IF F=1 THEN 240
300 PRINT Z;" Partitionen"
310 END
320 :
330 REM Rekursion
340 IF F=1 THEN 410
350 F=1
360 FOR I=1 TO K
370   IF P(I)=1 THEN 480
380 NEXT I
390 I=K:GOTO 480
400 :
410 L=K-I:K=I:P(I)=P(I)-1
420 IF P(K)>L THEN 460
430 L=L-P(K):K=K+1
440 P(K)=P(K-1):GOTO 420
450 :
460 K=K+1:P(K)=L+1
470 IF P(I)<>1 THEN I=K
480 IF P(I)=1 THEN I=I-1
490 IF I=0 THEN F=0
500 RETURN
```

```
*************************
   Partitionen
*************************
Welche Zahl ? 10

Die Partitionen von 10 sind:
 10
 9   1
 8   2
 8   1   1
 7   3
 7   2   1
 7   1   1   1
 6   4
 6   3   1
 6   2   2
 6   2   1   1
 6   1   1   1   1
 5   5
 5   4   1
 5   3   2
 5   3   1   1
 5   2   2   1
 5   2   1   1   1
 5   1   1   1   1   1
 4   4   2
 4   4   1   1
 4   3   3
 4   3   2   1
 4   3   1   1   1
 4   2   2   2
 4   2   2   1   1
 4   2   1   1   1   1
 4   1   1   1   1   1   1
 3   3   3   1
 3   3   2   2
 3   3   2   1   1
 3   3   1   1   1   1
 3   2   2   2   1
 3   2   2   1   1   1
 3   2   1   1   1   1   1
 3   1   1   1   1   1   1   1
 2   2   2   2   2
 2   2   2   2   1   1
 2   2   2   1   1   1   1
 2   2   1   1   1   1   1   1
 2   1   1   1   1   1   1   1   1
 1   1   1   1   1   1   1   1   1   1

 42  Partitionen
```

16. KUBISCHE GLEICHUNG

Die kubische Gleichung

$$ax^3 + bx^2 + cx + d = 0$$

kann mit Hilfe der Cardano-Formel gelöst werden. Die Cardano-Formel ist für das Rechnen von Hand äußerst umständlich, so daß man meist zu einem Iterationsverfahren greift (vgl. Programm 31). Sie liefert jedoch im Gegensatz zu den Iterationsverfahren sämtliche Lösungen der Gleichung.

Dividiert man die kubische Gleichung durch a und führt die Substitution

$$y = x + \frac{b}{3a}$$

durch, so erhält man die sogenannte reduzierte Gleichung

$$y + 3py + 2q = 0$$

mit

$$2q = \frac{2b^3}{27a^3} - \frac{bc}{3a^2} + \frac{d}{a} \qquad 3p = \frac{3ac - b^2}{3a^2}$$

Aus p und q berechnet sich die kubische Resultante

$$D = q^2 + p^3$$

Ihr Vorzeichen gibt Auskunft über die Art der Lösungen:

Ist $D > 0$, so gibt es 1 reelle, 2 konjugiert komplexe Lösungen
Ist $D < 0$, so gibt es 3 reelle Lösungen
Ist $D = 0$, so gibt es mehrfache, reelle Wurzeln

Mit Hilfe der 3. Wurzel

$$\sqrt[3]{-q \pm \sqrt{D}}$$

können nun die gesuchten Lösungen berechnet werden. Im Fall $D<0$ allerdings, ist in der Cardano-Formel eine komplexe Wurzel zu ziehen. Deswegen wird hier eine trigonometrische Lösung durchgeführt. Die hierbei benötigte Umkehrfunktion arccos(x) der Cosinus-Funktion ist nicht in BASIC implementiert. Sie muß daher über die Arctangens-Funktion ausgedrückt werden.

Es gilt

$$\arccos x = \begin{cases} \arctan \dfrac{\sqrt{1-x^2}}{x} & \text{für } \emptyset < x \leqslant 1 \\ \arctan \dfrac{\sqrt{1-x^2}}{x} + \pi & \text{für } -1 \leqslant x < \emptyset \\ \dfrac{\pi}{2} & \text{für } x = \emptyset \end{cases}$$

Zum folgenden Programm

Das Programm berechnet die Lösung der kubischen Gleichung nach dem angegebenen Verfahren von Cardano.

Eine berühmte kubische Gleichung ist

$$x^3 + 2x^2 + 1\emptyset x - 2\emptyset = \emptyset$$

Sie wurde Leonardo von Pisa (ca. 1170 – 1250), genannt Fibonacci, von Johann von Palermo, einem Höfling von Friedrich II, zur Lösung gestellt. Fibonacci fand die Lösung

$$x = 1.3688\emptyset81\emptyset75$$

die auf 8(!) Dezimalen genau ist.

Die Eingabe der Koeffizienten ins Programm liefert, neben der genannten Lösung, noch die beiden komplexen Lösungen

$$-1.6844\emptyset4\emptyset5 \pm 3.43133135i \quad \text{(gerundet)}$$

```
100 REM Kubische Gleichung
110 :
120 CLS:PRINT STRING$(25,42)
130 PRINT"  Kubische Gleichung"
140 PRINT STRING$(25,42)
150 :
160 PRINT"a*x^3+b*x^2+c*x+d=0"
170 PRINT"Eingabe der Koeffizienten a,b,c,d";
180 INPUT A,B,C,D
190 IF A=0 THEN PRINT"
      Keine kubische Gleichung ":END
200 A1=A:A=B/A1:B=C/A1:C=D/A1
210 :
220 'Parameter der reduzierten Gleichung
230 P=-A^2/9+B/3
240 Q=A^3/27-A*B/6+C/2
250 :
260 'Kubische Resultante
270 R=Q^2+P^3
280 :
290 'Fallunterscheidung
300 PRINT:PRINT"Loesung:"
310 ON SGN(R)+2 GOTO 490,430,330
320 :
330 '1 reelle,2 komplexen Nullstellen
340 U=SQR(R)-Q:U=SGN(U)*ABS(U)^(1/3)
350 V=-SQR(R)-Q:V=SGN(V)*ABS(V)^(1/3)
360 X1=U+V-A/3
370 'Real- und Imaginaerteil
380 X2=-(U+V)/2-A/3:X3=(U-V)/2*SQR(3)
390 PRINT"x1=";X1
400 PRINT"X2=";X2;"+i*";X3
410 PRINT"X2=";X2;"-i*";X3:END
420 :
430 'reelle Doppelloesung
440 U=SGN(-Q)*ABS(Q)^(1/3)
450 X1=U*2-A/3:X2=-U-A/3
460 PRINT"x1=";X1
470 PRINT"X2=x3=";X2:END
480 :
490 '3 reelle Loesungen
500 X=-Q/SQR(-P^3)
```

```
510 :
520 'Definition des Arccosinus
530 DEF FNARCCOS(X)=ATN(SQR(1-X^2)/X)
540 ON SGN(X)+2 GOTO 550,560,570
550 PHI=FNARCCOS(X)+PI:GOTO 580
560 PHI=FNARCCOS(X):GOTO 580
570 PHI=PI/2
580 X1=2*SQR(-P)*COS(PHI/3)-A/3
590 X2=-2*SQR(-P)*COS((PHI+PI)/3)-A/3
600 X3=-2*SQR(-P)*COS((PHI-PI)/3)-A/3
610 PRINT"x1=";X1
620 PRINT"x2=";X2
630 PRINT"x3=";X3
640 END
```

```
*************************
  Kubische Gleichung
*************************
a*x^3+b*x^2+c*x+d=0
Eingabe der Koeffizienten a,b,c,d? 1,2,10,-20

Loesung:
x1= 1.368808
X2=-1.684404 +i* 3.431331
X2=-1.684404 -i* 3.431331
```

17. GLEICHUNG VIERTEN GRADES

Ebenso wie die kubische Gleichung kann auch die Gleichung 4. Grades

$$ax^4 + bx^3 + cx^2 + dx + e = \emptyset$$

mit Hilfe von Wurzeltermen gelöst werden. Nach Ferrari kann die Lösung der Gleichung 4. Grades auf die kubische Gleichung

$$8y^3 - 4cy^2 + (2bd-8e)y + e(4c-b^2) - d^2 = \emptyset$$

zurückgeführt werden. Setzt man eine Lösung y in die quadratische Gleichung

$$x + \frac{1}{2}(b+A)x + (y+\frac{by-d}{A}) = \emptyset$$

mit

$$A = \pm\sqrt{8y + b^2 - 4c}$$

ein, so liefert deren Lösung schließlich die gesuchten Lösungen der ursprünglichen Gleichung 4. Grades.
Es ist klar, daß die Lösung nach Ferrari für das Rechnen von Hand zu kompliziert ist. Daher werden Gleichungen 4. Grades meist mit Iterationsverfahren angegegangen. Da jedoch Gleichungen 4. Grades nicht notwendig reelle Lösungen haben, kann es hier zu Konvergenzschwierigkeiten kommen. Da aber die Gleichungen vom 3. und 4. Grad häufig vorkommen – insbesondere im Zusammenhang mit dem charakteristischen Polynom einer Matrix – ist das vorliegende Programm sehr nützlich.

Nachdem Ferrari 1545 seine Formel veröffentlicht hatte, bemühten sich die Mathematiker – insbesondere Euler und Lagrange – über 250 Jahre lang eine entsprechende Formel für die Gleichung 5. Grades zu finden. Erst 1824 zeigte Abel, daß Gleichungen von höherem als dem 4. Grad prinzipiell nicht mit Radikalen, d.h. mit Wurzeltermen zu lösen sind. Hier muß man stets zu Iterationsverfahren greifen.

Zum folgenden Programm

Das Programm löst die allgemeine Gleichung 4. Grades nach dem Verfahren von Ferrari. Die dabei auftretende kubische Gleichung wird nach der Cardano-Formel von Programm 17 gelöst.

Als Beispiel wird die Gleichung

$$x^4 - 24x^3 + 15\emptyset x^2 - 2\emptyset\emptyset x - 375 = \emptyset$$

behandelt; sie ergibt sich in Programm 36 als charakteristisches Polynom der dort angegebenen Matrix.

Das Programm liefert die reellen Lösungen

—1, +5, +5, 15

```
100 REM Gleichung vierten Grades
110 :
120 CLS:PRINT STRING$(30,42)
130 PRINT"  Gleichung vierten Grades"
140 PRINT STRING$(30,42)
150 :
160 PRINT"a*x^4+b*x^3+c*x^2+d*x+e = 0"
170 INPUT "Koeffizienten a,b,c,d,e";A,B,C,D,E
180 IF A=0 THEN PRINT"
    Keine Gleichung 4.Grades":END
190 PRINT:PRINT"Loesung:"
200 :
210 IF B=0 AND D=0 THEN 830:'Biquadr.Gleich.
220 GOSUB 340:'kubische Gleichung
230 :
240 'quadratische Gleichung
250 J=8*Y+B^2-4*C
260 IF ABS(J)<.00000005  THEN J=0
270 H1=SQR(J):G=H1:H=(B*H1+J)/2
280 I=Y*H1+B*Y-D:GOSUB 670
290 :
300 H1=-H1:H=(B*H1+J)/2:G=H1
310 I=Y*H1+B*Y-D:GOSUB 670
320 END
330 :
340 'kubische Gleichung
350 A1=-C/2:B1=B*D/4-E:C1=(E*(4*C-B^2)-D^2)/8
360 P=-A1^2/9+B1/3
370 Q=A1^3/27-A1*B1/6+C1/2
380 'Kubische Resultante
390 R=Q^2+P^3
400 :
410 ON SGN(R)+2 GOTO 550,490,430
420 :
430 '1 reelle,2 komplexen Nullstellen
440 U=SQR(R)-Q:U=SGN(U)*ABS(U)^(1/3)
450 V=-SQR(R)-Q:V=SGN(V)*ABS(V)^(1/3)
460 Y=U+V-A1/3
470 RETURN
480 :
```

```
490 'reelle Doppelloesung
500 U=SGN(-Q)*ABS(Q)^(1/3)
510 :
520 Y=2*U-A1/3
530 RETURN
540 :
550 '3 reelle Loesungen
570 X=-Q/SQR(-P^3)
580 'Definition des Arccosinus
590 DEF FNARCCOS(X)=ATN(SQR(1-X^2)/X)
600 ON SGN(X)+2 GOTO 610,630,620
610 PHI=FNARCCOS(X)+PI:GOTO 640
620 PHI=FNARCCOS(X):GOTO 640
630 PHI=PI/2
640 Y=2*SQR(-P)*COS(PHI/3)-A1/3
650 RETURN
660 :
670 'quadratische Gleichung
680 D1=H^2-4*G*I
690 IF G=0 AND H=0 AND I=0 THEN PRINT"
    Vierfache Loesung x=";E^(1/4):END
700 IF D1<0 THEN 770
710 :
720 'reelle Loesung
730 PRINT (-H+SQR(D1))/(2*G)
740 PRINT (-H-SQR(D1))/(2*G)
750 RETURN
760 :
770 'komplexe Loesung
780 D1=-D1
790 PRINT -H/(2*G);"+i*";SQR(D1)/(2*G)
800 PRINT -H/(2*G);"-i*";SQR(D1)/(2*G)
810 RETURN
820 :
830 'biquadratische Gleichung
840 K=C^2-4*A*E
850 IF K<0 THEN 960
860 Y1=(-C+SQR(C^2-4*A*E))/(2*A)
870 Y2=(-C-SQR(C^2-4*A*E))/(2*A)
880 IF Y1<0 THEN 900
890 PRINT SQR(Y1):PRINT-SQR(Y1):GOTO 910
900 PRINT"I*";SQR(-Y1):PRINT"-I*";SQR(-Y1)
```

```
910 IF Y2<0 THEN 930
920 PRINT SQR(Y2):PRINT-SQR(Y2):GOTO 940
930 PRINT"I*";SQR(-Y2):PRINT"-I*";SQR(-Y2)
940 END
950 :
960 RE=-C/(2*A):IM=SQR(-K)/(2*A)
970 BT=SQR(RE^2+IM^2)
980 R1=SQR((BT+RE)/2):I1=SQR((BT-RE)/2)
990 BE=SQR(R1^2+I1^2)
1000 R2=SQR((BE+R1)/2 ):I2=SQR((BE-R1)/2 )
1010 PRINT R2;"-I*";I2
1020 PRINT R2;"+I*";I2
1030 PRINT-R2;"-I*";I2
1040 PRINT-R2;"+I*";I2
1050 END
```

```
******************************
  Gleichung vierten Grades
******************************
a*x^4+b*x^3+c*x^2+d*x+e = 0
Koeffizienten a,b,c,d,e? 1,-24,150,-200,-375

Loesung:
 5 +i* 1.9301E-04
 5 -i* 1.9301E-04
-1
 15
```

18. POLYNOMBERECHNUNG

Das folgende Programm bestimmt zu vorgegebenen Nullstellen das zugehörige Polynom.

Die Koeffizienten a_i $(1 \leqslant i \leqslant n)$ eines normierten Polynoms n-ten Grades

$$p(x) = x^n + a_{n-1}\,x^{n-1} + a_{n-2}\,x^{n-2} + a_{n-1}\,x^{n-1} + \ldots + a_o$$

sind nach dem Satz von Vieta elementarsymmetrische Funktionen der Nullstellen x_i:

$$x_1 + x_2 + x_3 + \ldots . + x_n = -a_{n-1}$$

$$x_1x_2 + x_1x_3 + x_1x_4 + \ldots + x_{n-1}\,x_n = a_{n-2}$$

$$x_1x_2x_3 + x_1x_2x_4 + x_1x_2x_5 + \ldots + x_{n-2}\,x_{n-1}\,x_n = -a_{n-3}$$

$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$x_1 \cdot x_2 \cdot x_3 \cdot x_4 \cdot \ldots . \cdot x_n = (-1)^n \cdot a_o$$

Zum folgenden Programm

Das Programm berechnet die Koeffizienten des gesuchten Polynoms nach oben genannten Formeln iterativ innerhalb zweier verschachtelter Schleifen

```
220  A(Ø) = Ø : A(N) = 1
230  FOR I=N TO 1 STEP -1
240  FOR J=H TO 1 STEP -1
250  A(N-J) = A(N-J) - A(N-J+1) * X(N-I+1)
260  NEXT J
270  NEXT I
```

Als Programmbeispiel wird das Polynom mit den Nullstellen

$$x_1 = 1,\ x_2 = 2,\ x_3 = 3,\ x_4 = 4,\ x_5 = 5 \text{ und } x_6 = 6$$

gesucht. Es ergibt sich das Polynom

$$P(x) = x^6 - 21x^5 + 175x^4 - 735x^3 + 1624x^2 - 1764x + 72Ø.$$

Das Programm funktioniert nur für reelle Nullstellen. Für komplexe Nullstellen müssen die entsprechenden quadratischen Polynome multipliziert werden, dies kann mittels Programm 20 erfolgen.

```
100 REM Polynomberechnung anhand
110 '    gegebener Nullstellen
120 :
130 CLS:PRINT STRING$(25,42)
140 PRINT"  Polynomberechnung"
150 PRINT STRING$(25,42)
160 :
170 PRINT:INPUT "Wieviele Nullstellen";N
180 DIM A(N),X(N)
190 :
200 FOR I=1 TO N
210   PRINT I;". Nullstelle";:INPUT X(I)
220   A(I)=0
230 NEXT I
240 :
250 A(0)=0:A(N)=1
260 FOR I=N TO 1 STEP -1
270   FOR J=N TO 1 STEP -1
280     A(N-J)=A(N-J)-A(N-J+1)*X(N-I+1)
290   NEXT J
300 NEXT I:PRINT
310 :
320 PRINT"Koeffizienten des Polynoms:"
330 FOR I=N TO 0 STEP -1
340   PRINT"a(";I;")=";A(I)
350 NEXT I
360 END
```

```
**************************
  Polynomberechnung
**************************

Wieviele Nullstellen? 6
 1 . Nullstelle? 1
 2 . Nullstelle? 2
 3 . Nullstelle? 3
 4 . Nullstelle? 4
 5 . Nullstelle? 5
 6 . Nullstelle? 6

Koeffizienten des Polynoms:
a( 6 )= 1
a( 5 )=-21
a( 4 )= 175
a( 3 )=-735
a( 2 )= 1624
a( 1 )=-1764
a( 0 )= 720
```

19. KOMPLEXES HORNERSCHEMA

Mit Hilfe des Hornerschemas kann der Wert des Polynoms

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + a_{n-2} x^{n-2} + \ldots + a_o.$$

an der Stelle x ohne Potenzieren ausgewertet werden. Dies kann mit folgendem Programmstück geschehen

```
100 F = A(N)
110 FOR I=N-1 TO 0 STEP -1
120 F=F * X + A(I)
130 NEXT I
```

Dabei ist N der Polynomgrad und F der gesuchte Funktionswert. Das Hornerschema ist auch im Betriebssystem vieler Rechner enthalten; mit ihm werden alle transzendenten Funktionen wie sin, arctan usw. über eine Polynomapproximation berechnet. Das angegebene Hornerschema kann auch für komplexe Zahlen angewendet werden. Dazu muß die komplexe Addition und Multiplikation geeignet sein. Sind die komplexen Zahlen als Zahlenpaare

(RE(1), IM(1))   (RE(2), IM(2))

gegeben, so kann die Addition wie folgt definiert werden

```
500 REM KOMPLEXE ADDITION
510 RE = RE(1) + RE(2)
520 IM = IM(1) + IM(2)
```

Entsprechend die Multiplikation

```
600 REM KOMPLEXE MULTIPLIKATION
610 RE = RE(1) * RE(2) - IM(1) * IM(2)
620 IM = RE(1) * IM(2) + RE(2) * IM(1)
```

Diese Rechenoperationen werden als Unterprogramm definiert und bei Bedarf angesprungen. Da jedoch in BASIC alle Variablen global sind, muß zuvor eine geeignete Parameterübergabe durchgeführt werden.

Da hier die komplexen Zahlen als Zahlenpaare behandelt werden, müssen entsprechend auch die Koeffizienten des Polynoms und der Argumentwert als Zahlenpaare eingelesen werden.

Zum folgenden Programm

Als Beispiel wird der Funktionswert des komplexen Polynoms

$$p(z) = (1 + 1.5i)z^3 - z^2 + 3z + (1 + 2i)$$

an der Stelle

$$z = 1 - i$$

berechnet. Die Koeffizienten werden im Programm mittels DATA-Werten eingelesen.
Es ergibt sich der Polynomwert

$$5-4i.$$

```
100 REM Komplexes Hornerschema
110 :
120 CLS:PRINT STRING$(25,42)
130 PRINT" Komplexes Hornerschema"
140 PRINT STRING$(25,42)
150 :
160 READ N:'Polynomgrad
170 PRINT"Polynomgrad=";N
180 DIM R(N),I(N),RE(2),IM(2)
190 :
200 PRINT"Koeffizienten:"
210 FOR K=N TO 0 STEP -1
220   READ RE,IM:'Real- Imag.teil
230   GOSUB 470
240   R(K)=RE:I(K)=IM
250 NEXT K
260 PRINT"Argument=";
270   READ RE,IM
280   GOSUB 470
290   X=RE:Y=IM
300 :
310 REM Hornerschema
320 RE(1)=R(N):IM(1)=I(N)
330 FOR K=N-1 TO 0 STEP -1
340   RE(2)=X:IM(2)=Y
350   GOSUB 560:'kompl.Mult.
360   RE(1)=RE:IM(1)=IM
370   RE(2)=R(K):IM(2)=I(K)
380   GOSUB 510:'kompl.Add.
390   RE(1)=RE:IM(1)=IM
400 NEXT K
410 :
420 PRINT:PRINT"Polynomwert=";
430 GOSUB 470
440 END
450 :
460 REM Ausgabeprozedur
470 IF IM>=0 THEN
    PRINT RE;"+i*";IM:GOTO 490
480 PRINT RE;"-i*";ABS(IM)
490 RETURN
500 :
```

```
510 REM Komplexe Addition
520 RE=RE(1)+RE(2)
530 IM=IM(1)+IM(2)
540 RETURN
550 :
560 REM Komplexe Multiplikation
570 RE=RE(1)*RE(2)-IM(1)*IM(2)
580 IM=RE(1)*IM(2)+IM(1)*RE(2)
590 RETURN
600 :
610 DATA 3
620 DATA 1,1.5,-1,0,3,0,1,2
630 DATA 1,-1
```

```
*************************
 Komplexes Hornerschema
*************************
Polynomgrad= 3
Koeffizienten:
 1 +i* 1.5
-1 +i* 0
 3 +i* 0
 1 +i* 2
Argument= 1 -i* 1

Polynomwert= 5 -i* 4
```

20. POLYNOM-MULTIPLIKATION

Zwei Polynome vom Grad n und m

$$a_n x^n + a_{n-1} x^{n-1} + a_{n-2} x^{n-2} + \ldots + a_o$$

$$b_m x^m + b_{m-1} x^{m-1} + b_{m-2} x^{m-2} + \ldots + b_o$$

werden multipliziert, indem man jeden Term des ersten Polynoms mit jedem des zweiten multipliziert.

Dies kann mit Hilfe zweier verschachtelter Schleifen geschehen

```
290 FOR I=0 TO N
300 FOR J=0 TO M
310 K = I + J
320 C(K) = C(K) + A(I) * B(J)
330 NEXT J
340 NEXT I
```

Das Verfahren läßt sich direkt auch auf Reihen übertragen; d.h. es liefert auch das Cauchy-Produkt zweier Reihenentwicklungen.

Zum folgenden Programm

Das Programm berechnet nach dem angegebenen Verfahren das Produkt zweier Polynome und gibt die entsprechenden Koeffizienten aus.

Als Beispiel werden die Polynome

$$p(x) = x^4 - 5x^3 + 3x^2 + 6x - 2 \ \text{-}\ a_o$$

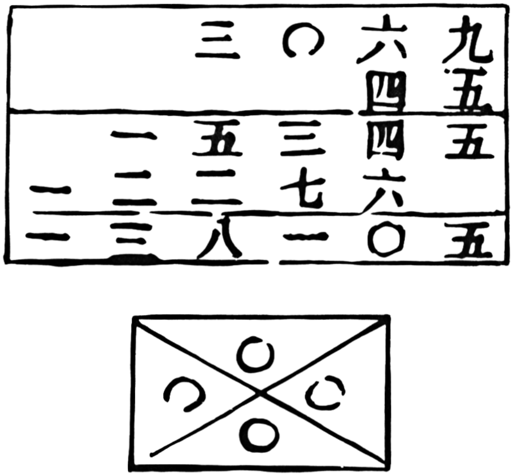
$$q(x) = x^2 - 1$$

multipliziert. Eingabe der Koeffizienten von links nach rechts ins Programm liefert die Koeffizienten des Produktpolynoms

1, –5,2, 11, –5, –6,2

Das Ergebnis ist somit

$$x^6 - 5x^5 + 2x^4 + 11x^3 - 5x^2 - 6x + 2$$



»Darstellung der Multiplikation
3069·45 = 138105
aus einer chinesischen Arithmetik
des Jahres 1355«.

```
100 REM Polynom-Multiplikation
110 :
120 CLS:PRINT STRING$(25,42)
130 PRINT" Polynom-Multiplikation"
140 PRINT STRING$(25,42)
150 :
160 INPUT"Grad des 1.Polynoms";N
170 DIM A(N)
180 FOR I=N TO 0 STEP -1
190   PRINT"Koeffizient x(";I;")=";
200    INPUT A(I)
210 NEXT I:PRINT
220 :
230 INPUT"Grad des 2.Polynoms";M
240 DIM B(M),C(N+M)
250 FOR I=M TO 0 STEP -1
260   PRINT"Koeffizient x(";I;")=";
270    INPUT B(I)
280 NEXT I:PRINT
290 :
300 FOR I=1 TO N+M
310 C(I)=0
320 NEXT I
330 FOR I=0 TO N
340 FOR J=0 TO M
350   K=I+J
360   C(K)=C(K)+A(I)*B(J)
370 NEXT J
380 NEXT I
390 :
400 PRINT"Koeffizienten d.Produkts:"
410 FOR I=N+M TO 0 STEP -1
420   PRINT C(I);
430 NEXT I:PRINT
440 END
```

```
**************************
 Polynom-Multiplikation
**************************
Grad des 1.Polynoms? 4
Koeffizient x( 4 )=? 1
Koeffizient x( 3 )=? -5
Koeffizient x( 2 )=? 3
Koeffizient x( 1 )=? 6
Koeffizient x( 0 )=? -2

Grad des 2.Polynoms? 2
Koeffizient x( 2 )=? 1
Koeffizient x( 1 )=? 0
Koeffizient x( 0 )=? -1

Koeffizienten d.Produkts:
 1 -5  2  11 -5 -6  2
```

21. POLYNOM-DIVISION

Die Polynom-Division findet vielfache Anwendung beim Abdividieren von gefundenen Nullstellen, bei der Partialbruchzerlegung u.s.w.

Die Polynom-Division kann in völliger Analogie zur gewöhnlichen Division durchgeführt werden:
Es wird geprüft, wie oft der entsprechende Koeffizient des Divisors in den Dividenden hineingeht. Das entsprechende Vielfache des Divisors wird dann vom Koeffizienten des Dividenden subtrahiert.

Zum folgenden Programm

Das Programm dividiert zwei Polynome nach dem gewöhnlichen Divisionsverfahren.
Es wurde versucht Rundungsfehler aufzufangen, indem sehr kleine Reste Null gesetzt werden. Die Polynomgrade und Koeffizienten werden in Form von DATA-Werten eingelesen.

Als Beispiel werden die Polynome

$$2x^7 + 9x^6 - 5x^5 + 42x^4 - 24x^3 + 10x^2 + 21x - 8$$

$$x^3 + 5x^2 - 3x + 4$$

dividiert. Das Programm liefert den Quotienten

$$2x^4 - x^3 + 6x^2 + x - 7$$

mit dem Rest

$$24x^2 - 4x + 20$$

```
100 REM Polynom-Division
110 :
120 CLS:PRINT STRING$(25,42)
130 PRINT"    Polynom-Division"
140 PRINT STRING$(25,42)
150 :
160 PRINT"Koeffizienten des Dividenden:"
170 READ N:'rem Polynomgrad
180 DIM A(N+1),B(N+1),Q(N+1),R(N+1)
190 FOR I=N+1 TO 1 STEP -1
200   READ A(I):PRINT A(I);
210 NEXT I:PRINT:PRINT
220 :
230 PRINT"Koeffizienten des Divisors"
240 READ M
250 FOR I=M+1 TO 1 STEP -1
260   READ B(I):PRINT B(I);
270 NEXT I:PRINT:PRINT
280 :
290 IF N>=M THEN 320
300 PRINT"Polynomgrad des Dividenden"
310 PRINT"<= der des Divisors":END
320 R=N-M
330 FOR I=N+1 TO M+1 STEP -1
340   Q(I-M)=A(I)/B(M+1)
350   FOR J=0 TO M
360     A(I-J)=A(I-J)-Q(I-M)*B(M+1-J)
370     IF ABS(A(I-J))>=ABS(Q(I-M)*
      B(M+1-J))*.0000001 THEN 390
380     A(I-J)=0
390   NEXT J
400 NEXT I
410 :
420 PRINT"Koeffizienten d.Quotienten:"
430 FOR I=R+1 TO 1 STEP -1
440   PRINT Q(I);
450 NEXT I:PRINT:PRINT
460 :
470 PRINT"Koeffizienten d.Restpolynoms:"
480 IF A(R)=0 THEN R=R-1:GOTO 480
490 FOR I=R TO 1 STEP -1
500   R(I)=A(I)
```

```
510   PRINT R(I);
520 NEXT I:PRINT
530 END
540 :
550 DATA 7
560 DATA 2,9,-5,42,-24,10,21,-8
570 DATA 3
580 DATA 1,5,-3,4
```

```
****************************
   Polynom-Division
****************************
Koeffizienten des Dividenden:
 2  9 -5  42 -24  10  21 -8

Koeffizienten des Divisors
 1  5 -3  4

Koeffizienten d.Quotienten:
 2 -1  6  1 -7

Koeffizienten d.Restpolynoms:
 24 -4  20
```

22. MATRIZEN-MULTIPLIKATION

Matrizen sind rechteckige Anordnungen von reellen Zahlen. Eine Matrix hat die Ordnung (m, n), wenn sie m Zeilen und n Spalten besitzt.

Zwei Matrizen **A** und **B** werden multipliziert, indem man jede Zeile von **A** mit jeder Spalte von **B** multipliziert. Dies zeigt, daß das Produkt von **A** und **B** nur existiert, wenn die Spaltenzahl von **A** mit der Zeilenzahl von **B** übereinstimmt. Ist **A** von der Ordnung (n, m) und **B** der Ordnung (p, q) mit m=p, so ist die Produktmatrix von der Ordnung (n, q).

Die Matrizenmultiplikation kann mit Hilfe von folgendem Programmstück durchgeführt werden

```
320 FOR I=1 TO N
330 FOR J=1 TO Q
340 S=Ø
350 FOR K=1 TO M
360 S=S + A(I,K) * B(K,J)
370 NEXT K
380 C(I,J) = S
390 NEXT J
400 NEXT I
```

Zum folgenden Programm

Das folgende Programm berechnet das Produkt zweier Matrizen, die in Form von DATA-Werten eingelesen werden. Dabei wird geprüft, ob die Spaltenzahl der ersten Matrix mit der Zeilenzahl der zweiten übereinstimmt, andernfalls wird eine Fehlermeldung ausgegeben.

Im Programmbeispiel werden die Matrizen

$$\begin{pmatrix} 8 & 5 & 2 & 3 \\ 6 & 4 & 1 & 0 \\ 1 & 2 & 2 & 4 \end{pmatrix} \qquad \begin{pmatrix} 1 & -2 \\ -2 & 3 \\ 1 & 1 \\ 1 & 2 \end{pmatrix}$$

miteinander multipliziert. Die Produktmatrix ist

$$\begin{pmatrix} 3 & -7 \\ -1 & 1 \\ 3 & 14 \end{pmatrix}$$

(vgl. Programmausdruck).

```
100 REM Matrizenmultiplikation
110 :
120 CLS:PRINT STRING$(25,42)
130 PRINT"  Matrizenmultiplikation"
140 PRINT STRING$(25,42)
150 :
160 PRINT"Gegebene Matrizen:"
170 READ N,M:'Ordnung der 1.Matrix
180 DIM A(N,M)
190 FOR I=1 TO N
200 FOR J=1 TO M
210   READ A(I,J):PRINT A(I,J);
220 NEXT J:PRINT
230 NEXT I:PRINT
240 :
250 READ P,Q:'Ordnung der 2.Matrix
260 DIM B(P,Q)
270 IF M=P THEN 320
280 PRINT"Eingabefehler:Spaltenzahl"
290 PRINT"der 1.Matrix muß mit der"
300 PRINT"Zeilenzahl der 2.Matrix"
310 PRINT"übereinstimmen":END
320 FOR I=1 TO P
330 FOR J=1 TO Q
340   READ B(I,J):PRINT B(I,J);
350 NEXT J:PRINT
360 NEXT I:PRINT
370 :
380 PRINT"Produktmatrix:"
390 FOR I=1 TO N
400 FOR J=1 TO Q
410   S=0
420   FOR K=1 TO M
430     S=S+A(I,K)*B(K,J)
440   NEXT K
450   C(I,J)=S:PRINT S;
460 NEXT J:PRINT
470 NEXT I:PRINT
480 END
490 :
500 DATA 3,4
```

```
510 DATA 8,5,2,3
520 DATA 6,4,1,0
530 DATA 1,2,2,4
540 :
550 DATA 4,2
560 DATA 1,-2
570 DATA -2,3
580 DATA 1,1
590 DATA 1,2
```

```
**************************
  Matrizenmultiplikation
**************************
Gegebene Matrizen:
 8  5  2  3
 6  4  1  0
 1  2  2  4

 1 -2
-2  3
 1  1
 1  2

Produktmatrix:
 3  7
-1  1
 3  14
```

23. ZWEIERPOTENZEN VON MATRIZEN

Da für quadratische Matrizen die Zeilenzahl mit der Spaltenzahl übereinstimmt, können quadratische Matrizen mit sich selbst multipliziert werden.

Solche Potenzen von Matrizen haben zahlreiche Anwendungen in der numerischen und Wirtschaftsmathematik, Wahrscheinlichkeitsrechnung und Graphentheorie.

Als Beispiel werden hier die (homogenen) Markowketten herausgegriffen:
Hat ein System endlich viele Zustände und geht ein Zustand i, unabhängig von seinem vorhergehenden Zustand, in den Zustand j mit der Wahrscheinlichkeit

$$p_{ij}$$

über, so stellt das System eine Markowkette (auch Markoff geschrieben) dar.
Die Matrix P mit den Elementen

$$p_{ij}$$

heißt eine stochastische Matrix. Solche Matrizen sind dadurch gekennzeichnet, daß ihre Zeilensumme den Wert 1 haben, da die Summe aller Übergangswahrscheinlichkeiten natürlich den Wert 1 hat.
Hat dieses System ein stationäres Verhalten, d.h. strebt es einer Grenzverteilung zu, so kann diese Grenzverteilung durch fortgesetztes Potentieren berechnet werden [10].
Multipliziert man die Matrix mit sich selbst und quadriert das jeweilige Ergebnis, so erhält man die Zweierpotenzen der Matrix.

Zum folgenden Programm

In einem bestimmten Land gebe es 3 Parteien A, B und C. Folgendes Wahlverhalten sei gegeben:
80% aller A-Wähler wählen bei der nächsten Wahl wieder A, 10% B und 10% C.
70% aller B-Wähler wählen bei der nächsten Wahl wieder B, 15% A und 15% C.
60% aller C-Wähler wählen bei der nächsten Wahl wieder C, 20% A und 20% B.

Damit ergibt sich folgende stochastische Matrix

$$\begin{pmatrix} Ø.8 & Ø.1 & Ø.1 \\ Ø.15 & Ø.7 & Ø.15 \\ Ø.2 & Ø.2 & Ø.6 \end{pmatrix}$$

Wie man dem Programmausdruck entnimmt, nähern sich die Zeilen der Matrizenpotenzen dem Vektor

(Ø.461912194 Ø.3Ø74191Ø8 Ø.23Ø668699)

Dieser stellt die Grenzverteilung dar; d.h. auf lange Sicht wählen

46.2 % aller Wähler	A
3Ø.7 %	B
23.1 %	C

```
100 REM Zweierpotenz einer Matrix
110 :
120 CLS:PRINT STRING$(30,42)
130 PRINT" Zweierpotenz einer Matrix"
140 PRINT STRING$(30,42)
150 :
160 READ N:'Ordnung der Matrix
170 READ P:'gewuenschte Zweierpotenz
180 DIM A(N,N),B(N,N)
190 :
200 PRINT"Gegebene Matrix:"
210 FOR I=1 TO N
220 FOR J=1 TO N
230 READ A(I,J):PRINT A(I,J);
240 NEXT J:PRINT
250 NEXT I:PRINT
260 :
270 P=INT(LOG(P)/LOG(2)+.5)
280 FOR L=1 TO P
290   PRINT 2^L;". Potenz:"
300   FOR I=1 TO N
310   FOR J=1 TO N
320     S=0
330     FOR K=1 TO N
340       S=S+A(I,K)*A(K,J)
350     NEXT K
360     B(I,J)=S:PRINT S;
370   NEXT J:PRINT
380   NEXT I:PRINT
390   :
400   FOR I=1 TO N
410   FOR J=1 TO N
420     A(I,J)=B(I,J)
430   NEXT J
440   NEXT I
450 NEXT L
460 END
470 :
480 DATA 3
490 DATA 16
500 DATA .8,.1,.1
```

```
510 DATA .15,.7,.15
520 DATA .2,.2,.6
```

```
******************************
 Zweierpotenz einer Matrix
******************************
Gegebene Matrix:
  0.8    0.1   0.1
  0.15   0.7   0.15
  0.2    0.2   0.6

 2 . Potenz:
  0.675 0.17  0.155
  0.255 0.535 0.21
  0.31  0.28  0.41

 4 . Potenz:
  0.547025    0.2491      0.203875
  0.37365     0.388375    0.237975
  0.40775     0.3173      0.27495


 8 . Potenz:
 0.475442597  0.297697678  0.226859726
 0.446546517  0.319420823  0.234032661
 0.453719452  0.312043548  0.234237002

 16 . Potenz:
 0.461912195  0.307419108  0.230668699
 0.461128662  0.307993905  0.230877434
 0.461337398  0.307836579  0.230826025
```

24. MITTELWERTE

Neben dem arithmetischen Mittel A der Zahlen x_i $(1 \leqslant i \leqslant n)$

$$A = \frac{1}{n} (x_1 + x_2 + x_3 + x_4 \ldots + x_n)$$

sind auch andere Mittelwerte von Bedeutung:

das geometrische Mittel

$$G = \sqrt[n]{x_1 \cdot x_2 \cdot x_3 \cdot x_4 \cdot \ldots \ldots \cdot x_n}$$

das harmonische Mittel

$$\frac{1}{H} = n \left(\frac{1}{x_1} + \frac{1}{x_2} + \frac{1}{x_3} + \frac{1}{x_4} + \ldots \frac{1}{x_n} \right)$$

das quadratische Mittel

$$Q = \sqrt{\frac{1}{n} \left(x_1^2 + x_2^2 + x_3^2 + x_4^2 + \ldots . + x_n^2\right)}$$

Das meistgenutzte arithmetische Mittel ist sehr empfindlich gegen Ausreißer; d.h. bei kleinen Zahlenreihen wird das arithmetische Mittel durch einen zufällig großen Wert verfälscht. In diesem Fall sollte man lieber den Median (vgl. Programm 25) nehmen.

Das geometrische Mittel ist stets bei prozentualen Größen zu nehmen. Wird z.B. eine Gehaltserhöhung von 3% erst zum 1. Juni ausgezahlt, so ist die effektive Gehaltserhöhung nur

$$\left(\sqrt[12]{1.\text{Ø}3^7} - 1\right) \cdot 1\text{ØØ}\ \% = 1.7\ \%$$

Hier findet man häufig falsche Angaben, auch in angesehenen Tageszeitungen.

Fährt ein Auto eine Strecke bergauf mit 30 km/h und bergab mit 90 km/h, so ist die Durchschnittgeschwindigkeit keineswegs 60 km/h, sondern nur das harmonische Mittel von 45 km/h.

Für das arithmetische, geometrische und harmonische Mittel gilt die Ungleichung von Cauchy

$$A \geq G \geq H$$

Das quadratische Mittel hat große Bedeutung in der Statistik, da die Standardabweichung einer Meßreihe gleich dem quadratischen Mittel der einzelnen Abweichungen ist.

Zum folgenden Programm

Das Programm berechnet für beliebige Zahlenreihen das arithmetische, geometrische, harmonische und quadratische Mittel nach den oben angegebenen Formeln.

Als Beispiel wurde die Zahlenreihe:

19, 22, 2Ø, 18, 19, 23, 17, 21, 25, 17

eingegeben. Es ergeben sich folgende gerundete Mittel:

arithmetisches M.:	2Ø.1Ø
geometrisches M.:	19.95
harmonisches M.:	19.8Ø
quadratisches M.:	2Ø.26

```
100 REM Mittelwerte
110 :
120 CLS:PRINT STRING$(25,42)
130 PRINT"      Mittelwerte "
140 PRINT STRING$(25,42)
150 :
160 INPUT "Wieviele Zahlen ";N
170 A=0:G=1:H=0:Q=0:DF=1
180 :
190 PRINT:PRINT"Eingabe der Zahlen:"
200 FOR I=1 TO N
210 PRINT USING" ##. Zahl";I;:INPUT X
220 A=A+X:G=G*X:Q=Q+X^2
230 IF X=0 THEN DF=0:GOTO 250
240 H=H+1/X
250 NEXT I:PRINT
260 :
270 A=A/N
280 G=G^(1/N)
290 Q=SQR(Q/N)
300 IF H<>0 THEN H=N/H
310 :
320 PRINT"Arithmetisches Mittel=";
330 PRINT USING "#####.###";A
340 PRINT" Geometrisches Mittel=";
350 PRINT USING "#####.###";G
360 IF DF=0 THEN 370 ELSE 380
370 PRINT" Harmonisches Mittel
          nicht definiert":GOTO 400
380 PRINT"  Harmonisches Mittel=";
390 PRINT USING "#####.###";H
400 PRINT " Quadratisches Mittel=";
410 PRINT USING "#####.###";Q
420 END
```

```
*************************
     Mittelwerte
*************************
Wieviele Zahlen ? 10

Eingabe der Zahlen:
  1. Zahl? 19
  2. Zahl? 22
  3. Zahl? 20
  4. Zahl? 18
  5. Zahl? 19
  6. Zahl? 23
  7. Zahl? 17
  8. Zahl? 21
  9. Zahl? 25
 10. Zahl? 17

Arithmetisches Mittel=   20.100
 Geometrisches Mittel=   19.948
  Harmonisches Mittel=   19.801
 Quadratisches Mittel=   20.256
```

25. STATISTISCHE MITTELWERTE

Neben dem arithmetischen Mittel werden in der Statistik auch noch andere Mittelwerte betrachtet.

Der Modus ist der häufigste Wert einer Stichprobe oder einer Meßwertreihe. Für Durchschnittslohnangaben ist sicher der Modus aussagekräftiger als der Mittelwert, der durch Bezieher sehr großer Einkommen verfälscht wird.

Denkt man sich alle Stichprobenwerte der Größe nach geordnet, so stellt das in der Mitte liegende Element den Median dar. Ist die Zahl der Werte gerade, so wählt man den Mittelwert der beiden mittleren Elemente zum Median; z.B. beim Stichprobenumfang 100 ist der Median das Mittel aus dem 50. und 51. Stichprobenwert. Der Median hat sehr große Bedeutung beim Testen auf vorgegebene Verteilungen (siehe z.B. [16]). Bei einer geringen Zahl von Stichprobenwerten ist das arithmetische Mittel besonders empfindlich gegen "Ausreißer". So verwendet die Stiftung Warentest als mittleren Preis eines Gerätes den Median, wenn weniger als 5 Geräte gekauft werden.

Weitere Parameter, die die Verteilung von Zahlen kennzeichnen, sind die Varianz, die Schiefe und der Exzeß.

Sind x_i $(1 <= i <= n)$ die Stichprobenwerte, so stellt

$$V = \frac{\Sigma x_i 2 - \frac{(\Sigma x_i)2}{n}}{n-1}$$

die Varianz dar. Sie ist ein Maß für die Streuung der Werte um ihren Mittelwert. Die Quadratwurzel aus der Varianz wird Standardabweichung σ genannt. Im Falle einer Normalverteilung gibt letztere Zahl den Bereich um den Mittelwert an, in dem sich 65% aller Werte befinden.

Die Schiefe wird berechnet aus

$$\gamma = \left(\frac{\Sigma x_i 3}{n} - 3\,\frac{\Sigma x_i 2}{n} \cdot \frac{\Sigma x_i}{n} + 2\left(\frac{\Sigma x_i}{n}\right)^3\right) / \sigma 3$$

Sie ist ein Maß für die Symmetrie der Verteilung.

Der Exzeß

$$\epsilon = \left(\frac{\Sigma x_i 4}{n} - 4\,\frac{\Sigma x_i 3}{n} - \frac{\Sigma x_i}{n} + 6\,\frac{\Sigma x_i 2}{n}\left(\frac{\Sigma x}{n}\right)^2 - 3\left(\frac{\Sigma x_i}{n}\right)^4\right) / \sigma 4 - 3$$

(auch Wölbung genannt) ist ein Maß für die Steilheit der Kurve. Positiver Exzeß zeigt, daß die Verteilungskurve steiler als die Normalverteilung verläuft, negative Werte zeigen einen flacheren Verlauf. Für die Normalverteilung verschwindet sowohl Schiefe wie Exzeß.

Zum folgenden Programm

Gibt man die Stichprobenwerte in Form von DATA-Werten ein, so werden die oben genannten statistischen Kennwerte nach den angegebenen Formeln berechnet. Existieren zwei Werte mit gleich großer Häufigkeit, so wird der kleinere als Modus gewählt.

Für die Stichprobe

(18, 2Ø, 21, 19, 22, 23, 17, 21, 18, 24)

können die sich ergebenden Maßzahlen dem Programmausdruck entnommen werden.
Die Schiefe ist klein, da genau 5 Werte unter und 5 Werte über dem Mittel von 2Ø.3 liegen.
Der Exzeß ist negativ, dies zeigt eine flache Verteilung an.

```
100 REM Statistische Mittelwerte
110 :
120 CLS:PRINT STRING$(30,42)
130 PRINT"  Statistische Mittelwerte"
140 PRINT STRING$(30,42)
150 :
160 READ N
170 DIM X(N),H(N)
180 :
190 'Einlesen der Datawerte
200 M=0:S=0:T=0:U=0
210 FOR I=1 TO N
220 READ X
230 M=M+X:S=S+X^2
240 T=T+X^3:U=U+X^4
250 H(I)=1:X(I)=X
260 NEXT I
270 A=M/N
280 :
290 'Bubblesort
300 F=1
310 FOR I=1 TO N-1
320   IF X(I)<=X(I+1) THEN 340
330   H=X(I):X(I)=X(I+1):X(I+1)=H:F=0
340 NEXT I
350 IF F=0 THEN 300
```

```
360 :
370 PRINT:PRINT USING"Mittelwert=              ####.####";A
380 V=S/N-A^2
390 PRINT USING"Varianz=                 ####.####";V
400 SG=SQR(ABS(V))
410 PRINT USING"Standardabweichung= ####.####";SG
420 :
430 SF=(T/N-3*A*S/N+2*A^3)/SG^3
440 PRINT USING"Schiefe                  ####.####";SF
450 :
460 E=(U/N-4*T/N*A+6*S*A^2/N-3*A^4)/SG^4-3
470 PRINT USING"Exzess=                   ####.####";E
480 :
490 IF N MOD 2 <>0 THEN ME=X((N+1)/2):GOTO 510
500 ME=(X(N/2)+X(N/2+1))/2
510 PRINT USING"Median=                  ####.####";ME
520 :
530 'Frequenzzaehlung
540 FOR I=1 TO N-1
550   FOR J=I+1 TO N
560     IF X(I)=X(J)  THEN H(I)=H(I)+1
570   NEXT J
580 NEXT I
590 K=0:M=0
600 FOR I=1 TO N
610   IF H(I)>K THEN K=H(I):L=I
620 NEXT I
630 FOR I=1 TO N
```

```
640   IF (H(I)=K) AND (I<>L) THEN M=1
650 NEXT I
660 IF K=0 THEN PRINT"Modus nicht definiert":GOTO 690
670 PRINT USING"Modus=                    ####.####";X(L)
680 IF M=1 THEN PRINT"Modus nicht eindeutig"
690 END
700 :
710 DATA 10
720 DATA 18,20,21,19,22,23,17,21,18,24
```

```
******************************
  Statistische Mittelwerte
******************************

Mittelwert=              20.3000
Varianz=                  4.8100
Standardabweichung=       2.1932
Schiefe                   0.1274
Exzess=                  -1.1552
Median=                  20.5000
Modus=                   18.0000
Modus nicht eindeutig
```

"Pythagoras zeigt an Glocken, gespannten Saiten und Flöten den Zusammenhang von Harmonie und ganzen Zahlen. Aus Theorica Musice, Mailand 1492"

26. PYTHAGOREISCHE ZAHLEN

Ganzzahlige Lösungen $\neq \emptyset$ der Diophantischen Gleichung

$$x^2 + y^2 = z^2$$

heißen pythagoreische Zahlen. Ein Dreieck ist genau dann rechtwinklig, wenn seine Seitenlängen pythagoreische Zahlen sind.

Setze man zwei solcher rechtwinkeligen Dreiecke mit gleicher Kathete zusammen, so erhält man ein Dreieck, dessen Höhe ganzzahlig ist. Die Seitenlängen eines solchen Dreiecks nennt man auch Heronsche Zahlen, da sie rationale Werte der Heronschen Flächenformel liefern.

Bekannte pythagoreische Tripel sind

(3, 4, 5)
(8, 6, 1Ø)
(5, 12, 13)
(9, 12, 15)

Pythagoreische Zahlen können durch die Formeln

$$x = u^2 - v^2$$
$$y = 2uv$$
$$z = u^2 + v^2$$

erzeugt werden; dabei durchlaufen u und v alle natürlichen Zahlen. Die obengenannten Tripel erhält man für

$u = 2, \; v = 1$
$u = 3, \; v = 1$
$u = 3, \; v = 2$
$u = 4, \; v = 2$

Wie man an den Beispielen sieht, sind auch stets alle Vielfachen eines solchen Tripels wieder pythagoreisch. Im folgenden Programm werden daher nur teilerfremde Tripel erzeugt.

Zum folgenden Programm

Das folgende Programm berechnet nach dem oben angegebenen Verfahren beliebig viele pythagoreische Zahlen, wobei Vielfache anderer Tripel ausgeschlossen werden.
Die ersten 20 vom Programm erzeugten Zahlentripel können dem Programmausdruck entnommen werden.

```
100 REM Pythagoreische Zahlen
110 :
120 CLS:PRINT STRING$(25,42)
130 PRINT"  Pythagoreische Zahlen"
140 PRINT STRING$(25,42)
150 :
160  REM  DEFINT A-C,N-Z
170 'Startwerte
180 U=2:Z=0
190 PRINT" Nr.----------A-----------B----------C---"
200 :
210 V=U-1
220 A=U^2-V^2:B=2*U*V:C=U^2+V^2
230 PRINT USING "###      #####      #####     #####";Z,A,B,C
240 Z=Z+1
250 IF (Z MOD 18)=0 THEN 260 ELSE 310
260 PRINT SPC(65)"Taste "
270 IF INKEY$="" THEN 270
280 CLS
290 PRINT" Nr.----------A-----------B----------C---"
300 :
310 V=V-2
320 IF V<=0 THEN U=U+1:GOTO 210
330 V1=V:U2=U
340 N=U2/V1:U1=U2 MOD V1
350 IF U1<>0 THEN U2=V1:V1=U1:GOTO 340
360 IF V1<>1 THEN 310
370 GOTO 220
380 END
```

```
*************************
 Pythagoreische Zahlen
*************************
```

Nr.	A	B	C
0	3	4	5
1	5	12	13
2	7	24	25
3	15	8	17
4	9	40	41
5	21	20	29
6	11	60	61
7	35	12	37
8	13	84	85
9	33	56	65
10	45	28	53
11	15	112	113
12	39	80	89
13	55	48	73
14	63	16	65
15	17	144	145
16	65	72	97
17	77	36	85

27. KOORDINATEN BESONDERER DREIECKSPUNKTE

Sind $\underline{p}1$, $\underline{p}2$, $\underline{p}3$ die Ortsvektoren der Dreiecks-Eckpunkte, so ergeben sich die Ortsvektoren der besonderen Dreieckspunkte aus folgenden Formeln:

Schwerpunkt

$$s = \frac{\underline{p}1 + \underline{p}2 + \underline{p}3}{3}$$

Inkreismittelpunkt

$$i = \frac{a_1\underline{p}1 + a_2\underline{p}2 + a_3\underline{p}3}{a_1 + a_2 + a_3}$$

Höhenschnittpunkt

$$h = \frac{t_1\underline{p}1 + t_2\underline{p}2 + t_3\underline{p}3}{t_1 t_2 t_3}$$

Umkreismittelpunkt

$$u = \frac{c_1 s_1 \underline{p}1 + c_2 s_2 \underline{p}2 + c_3 s_3 \underline{p}3}{2\, s_1 s_2 s_3}$$

dabei stellen a_1, a_2, a_3 die Seitenlängen,
s_1, s_2, s_3 die Sinuswerte,
c_1, c_2, c_3 die Cosinuswerte,
t_1, t_2, t_3 die Tangenswerte

des Dreiecks dar.

Zum folgenden Programm

Das Programm berechnet die 4 besonderen Punkte eines Dreiecks nach den oben angegebenen Vektorgleichungen.
Als Programmbeispiel wird das räumliche Dreieck

(5 | Ø | –4), (–2 | 6 | 3), (6 | –3 | –5)

behandelt. Es ergeben sich folgende Werte,

Schwerpunkt:

(3 | 1 | –2)

Inkreismittelpunkt:

(4.6Ø337933 | −Ø.5Ø521699 | −3.6Ø337933)

Höhenschnittpunkt:

(14.6 | 17.Ø666654 | −13.56)

Umkreismittelpunkt:

(−2.8 | −7.Ø3333264 | 3.8)

Zur Kontrolle kann man den Satz anwenden, daß der Schwerpunkt die Strecke Umkreismittelpunkt – Höhenschnittpunkt im Verhältnis 1:2 teilt

$$s = \frac{1}{3}(2 \cdot u + h)$$

Dies ist, wie man leicht nachrechnet, erfüllt.

```
100 REM Koordinaten besonderer Dreieckspunkte
110 :
120 CLS:PRINT STRING$(40,42)
130 PRINT" Koordinaten besonderer Dreieckspunkte"
140 PRINT STRING$(40,42)
150 :
160 INPUT"Koordinaten x,y,z des 1.Punkts";X(1),X(2),X(3)
170 INPUT"Koordinaten x,y,z des 2.Punkts";Y(1),Y(2),Y(3)
180 INPUT"Koordinaten x,y,z des 3.Punkts";Z(1),Z(2),Z(3)
190 :
200 'Seitenvektoren
210 P(1,1)=Y(1)-Z(1):P(1,2)=Y(2)-Z(2):P(1,3)=Y(3)-Z(3)
220 P(2,1)=Z(1)-X(1):P(2,2)=Z(2)-X(2):P(2,3)=Z(3)-X(3)
230 P(3,1)=X(1)-Y(1):P(3,2)=X(2)-Y(2):P(3,3)=X(3)-Y(3)
240 :
250 :
260 'Seitenlaengen
270 FOR I=1 TO 3
280   S=0
290   FOR J=1 TO 3
300     S=S+P(I,J)^2
310   NEXT J
320   A(I)=SQR(S)
330 NEXT I
340 :
```

```
350 'Cosinuswerte
360 C=0
370 FOR I=1 TO 3
380   C=C+P(2,I)*P(1,I)
390 NEXT I
400 C(3)=-C/(A(1)*A(2)):C=0
410 FOR I=1 TO 3
420   C=C+P(3,I)*P(1,I)
430 NEXT I
440 C(2)=-C/(A(1)*A(3)):C=0
450 FOR I=1 TO 3
460   C=C+P(3,I)*P(2,I)
470 NEXT I
480 C(1)=-C/(A(2)*A(3))
490 :
500 'Sinus- und Tangenswerte
510 FOR I=1 TO 3
520   S(I)=SQR(1-C(I)^2)
530   IF C(I)=0 THEN T(I)=1E+30:GOTO 550
540   T(I)=S(I)/C(I)
550 NEXT I
560 :
570 PRINT:PRINT"Schwerpunkt:"
580 FOR I=1 TO 3
590   PRINT(X(I)+Y(I)+Z(I))/3;
600 NEXT I:PRINT
```

```
610 :
620 PRINT"Inkreismittelpunkt:"
630 U=A(1)+A(2)+A(3)
640 FOR I=1 TO 3
650  PRINT(A(1)*X(I)+A(2)*Y(I)+A(3)*Z(I))/U;
660 NEXT I:PRINT
670 :
680 T=T(1)*T(2)*T(3)
690 IF T=0 THEN 750
700 PRINT"Hoehenschnittpunkt:"
710 FOR I=1 TO 3
720   PRINT(T(1)*X(I)+T(2)*Y(I)+T(3)*Z(I))/T;
730 NEXT I:PRINT
740 :
750 S=2*S(1)*S(2)*S(3)
760 IF S=0 THEN 810
770 PRINT"Umkreismittelpunkt"
780 FOR I=1 TO 3
790   PRINT(C(1)*S(1)*X(I)+C(2)*S(2)*Y(I)+C(3)*S(3)*Z(I))/S;
800 NEXT I:PRINT
810 END
```

```
****************************************
Koordinaten x,y,z des 1.Punkts? 5,0,-4
Koordinaten x,y,z des 2.Punkts? -2,6,3
Koordinaten x,y,z des 3.Punkts? 6,-3,-5

Schwerpunkt:
 3  1 -2
Inkreismittelpunkt:
 4.60337933  -0.505216989  -3.60337933
Hoehenschnittpunkt:
 14.6000006  17.066669    -13.6000003
Umkreismittelpunkt
-2.79999994  -7.03333456    3.79999987
```

28. POLYGON-BERECHNUNG

Das folgende Programm berechnet Flächeninhalt und Umfang eines Vielecks bei bekannten Koordinaten der Eckpunkte. Der Flächeninhalt wird nach der Formel, vgl. [4]

$$A = [(x_1 + x_2)(y_1 - y_2) + (x_2 + x_3)(y_2 - y_3) + \ldots$$
$$+ (x_n + x_1)(y_n - y_1)] / 2$$

berechnet.

Da der Abstand des Punktes $(x_{i+1} \mid y_{i+1})$ vom Punkt $(x_i \mid y_i)$ nach Pythagoras gleich

$$d = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}$$

ist, ergibt sich der Umfang des Vielecks durch Aufsummieren der Abstände der einzelnen Eckpunkte.

Der Schwerpunkt ergibt sich aus

$$x_s = \frac{1}{6A} \sum_{i=1}^{n} (x_i y_{i+1} - x_{i+1} y_i)(x_i + x_{i+1}) \qquad \text{mit } x_{n+1} = x_1$$

$$y_s = \frac{1}{6A} \sum_{i=1}^{n} (x_i y_{i+1} - x_{i+1} y_i)(y_i + y_{i+1}) \qquad y_{n+1} = y_1$$

Zum folgenden Programm

Das Programm berechnet für ein beliebiges Vieleck Umfang und Flächeninhalt. Bei der Eingabe der Koordinaten muß darauf geachtet werden, daß der Umlaufsinn eingehalten wird, da die Formel nicht für "überschlagene" Vielecke gilt.

Als Programmbeispiel wird das Sechseck

(4 | 2), (1Ø | Ø), (12 | 9), (8 | 11), (3 | 5), (Ø | 3)

gewählt. Es ergibt sich

Flächeninhalt = 67
Umfang = 35.5551423

Schwerpunkt = (7.35323384 | 5.2Ø895523)

"Visieren von Winkeln mit Hilfe des Jakobstabs. Holzschnitt aus dem 16. Jahrhundert"

```
100 REM Polygonberechnung
110 :
120 CLS:PRINT STRING$(25,42)
130 PRINT"  Polygonberechnung"
140 PRINT STRING$(25,42)
150 :
160 INPUT"Wieviele Eckpunkte";N
170 DIM X(N+1),Y(N+1)
180 :
190 PRINT"Gib Koordinaten der Eckpunkte"
200 PRINT"im gleichen Umlaufssinn ein !"
210 FOR I=1 TO N
220 PRINT"X(";I;"),Y(";I;")";:INPUT X(I),Y(I)
230 NEXT I
240 X(N+1)=X(1):Y(N+1)=Y(1)
250 :
260 'Flaechenberechnung
270 F=0
280 FOR I=1 TO N
290 F=F+(X(I)+X(I+1))*(Y(I)-Y(I+1))
300 NEXT I
310 F=ABS(F)/2
320 :
330 'Umfang
340 U=0
350 FOR I=1 TO N
360 U=U+SQR((X(I+1)-X(I))^2+(Y(I+1)-Y(I))^2)
370 NEXT I
380 :
390 'Schwerpunktskoordinaten
400 XS=0:YS=0
410 FOR I=1 TO N
420 XS=XS+(X(I)*Y(I+1)-X(I+1)*Y(I))*(X(I)+X(I+1))
430 YS=YS+(X(I)*Y(I+1)-X(I+1)*Y(I))*(Y(I)+Y(I+1))
440 NEXT I
450 XS=XS/(6*F):YS=YS/(6*F)
460 :
470 PRINT:PRINT"Flaecheninhalt=";F
480 PRINT"Umfang=";U
490 PRINT"Schwerpunkt (";XS;",";YS;")"
500 END
```

```
*************************
  Polygonberechnung
*************************
Wieviele Eckpunkte? 6
Gib Koordinaten der Eckpunkte
im gleichen Umlaufssinn ein !
X( 1 ),Y( 1 )? 4,2
X( 2 ),Y( 2 )? 10,0
X( 3 ),Y( 3 )? 12,9
X( 4 ),Y( 4 )? 8,11
X( 5 ),Y( 5 )? 3,5
X( 6 ),Y( 6 )? 0,3

Flaecheninhalt= 67
Umfang= 35.5551423
Schwerpunkt ( 7.35323383 , 5.20895522 )
```

29. DETERMINANTE

Determinanten haben zahlreiche Anwendungen: Mit ihrer Hilfe kann geprüft werden, ob

- Vektoren linear abhängig sind
- ein lineares Gleichungssystem lösbar ist
- zwei Polynome gemeinsame Nullstellen haben
- ein Extremum einer Funktion mehrerer Variablen vorliegt

usw.

Die Resultante der Polynome

$$f(x) = x^3 + 2x^2 + 3x + 5$$
$$g(x) = 6x^2 + 8x + 9$$

$$\mathrm{Det}\,(f,g) = \begin{vmatrix} 1 & 2 & 3 & 5 & 0 \\ 0 & 1 & 2 & 3 & 5 \\ 6 & 8 & 9 & 0 & 0 \\ 0 & 6 & 8 & 9 & 0 \\ 0 & 0 & 6 & 8 & 9 \end{vmatrix}$$

z.B. ist genau dann Null, wenn f und g mindestens eine gemeinsame Nullstelle haben. Hier gilt Det (f,g) = 2873, somit haben f und g keine gemeinsamen Nullstellen. Ersetzt man g durch die Ableitung f' von f, so gibt die Determinante Auskunft, ob f mehrfache Nullstellen hat.

Erwähnenswert ist auch die Cramersche Regel, die erlaubt einzelne Unbekannte eines Gleichungssystems zu finden, ohne das ganze System lösen zu müssen. Ein Programm dazu findet sich in [15].

Zum folgenden Programm

Das Programm berechnet die Determinante durch fortgesetztes Entwickeln nach der 1. Spalte.
Dazu werden die Zeilen 2 bis n mit dem Element $a_{11} \neq 0$ multipliziert. Dadurch vervielfacht sich zunächst der Wert der Determinante um

$$a_{11}^{n-1}$$

Der entsprechende Faktor wird sodann abdividiert. Subtrahiert man von jeder der Zeilen 2 bis n das

a_{i1}-fache von Zeile 1

so wird in der 1. Spalte der Vektor

(1 | Ø | Ø | . . . . | Ø)

erzeugt.

Entwickelt man nun nach der 1. Spalte, so verbleibt eine Determinante der Ordnung n–1.
Setzt man das Verfahren fort, so verringert sich die Ordnung der Determinante bei jedem Schritt, bis sie nur noch zweireihig ist. Im letzteren Fall kann der Wert leicht ermittelt werden.
Die Determinante wird im Programm in Form von DATA-Werten eingelesen.

Als Beispiel wird die Determinante

$$\begin{vmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 2 & 3 & 4 & 5 & 6 \\ 3 & 3 & 3 & 4 & 5 & 6 \\ 4 & 4 & 4 & 4 & 5 & 6 \\ 5 & 5 & 5 & 5 & 5 & 6 \\ 6 & 6 & 6 & 6 & 6 & 6 \end{vmatrix}$$

berechnet. Das Programm liefert den Wert –6.

```
100 REM Determinante
110 :
120 CLS:PRINT STRING$(25,42)
130 PRINT"    Determinante"
140 PRINT STRING$(25,42)
150 :
160 READ N:'Ordnung der Determinante
170 DIM A(N,N)
180 :
190 'Einlesen der Derterminante
200 FOR I=1 TO N
210 FOR J=1 TO N
220    READ A(I,J):PRINT A(I,J);
230 NEXT J:PRINT
240 NEXT I:PRINT
250 :
260 REM Berechnung der Determinante
270 GOSUB 310
280 PRINT"Determinante=";D
290 END
300 :
310 REM Unterprogramm Determinante
320 D=1:M=N
330 WHILE M>2
340   P=0:REM Pivotsuche
350   IF ABS(A(1,1))>.00001 THEN 490
360   FOR K=2 TO N
370      IF ABS(A(1,K))<=ABS(A(1,K-1))
         THEN 390
380      P=A(1,K):L=K
390   NEXT K
400   IF P=0 THEN D=0:GOTO 640
410   :
420   REM Spaltentausch
430   FOR J=1 TO M
440     H=A(J,L):A(J,L)=A(J,1):A(J,1)=H
450   NEXT J
460   D=-D
470   :
480   REM Entwicklung nach der 1.Spalte
490   FOR J=2 TO M
500   FOR K=2 TO M
```

```
510      A(J,K)=A(1,1)*A(J,K)-A(J,1)*A(1,K)
520     NEXT K
530     NEXT J
540     T=SGN(A(1,1))*ABS(A(1,1)^(M-2)):
          D=D/T
550     FOR J=2 TO M
560     FOR K=2 TO M
570        A(J-1,K-1)=A(J,K)
580     NEXT K
590     NEXT J
600     M=M-1
610 WEND
620 'Zweireihige Restdeterminante
630 D=D*(A(1,1)*A(2,2)-A(1,2)*A(2,1))
640 RETURN
650 :
660 DATA 6
670 DATA 1,2,3,4,5,6
680 DATA 2,2,3,4,5,6
690 DATA 3,3,3,4,5,6
700 DATA 4,4,4,4,5,6
710 DATA 5,5,5,5,5,6
720 DATA 6,6,6,6,6,6
```

```
*************************
   Determinante
*************************
 1   2   3   4   5   6
 2   2   3   4   5   6
 3   3   3   4   5   6
 4   4   4   4   5   6
 5   5   5   5   5   6
 6   6   6   6   6   6

Determinante=-6
```

30. MÜLLER-ITERATION

Die Iteration ist eine Programmiermethode, bei der durch zahlreiche Iterationsschritte ein gesuchter Punkt immer näher eingekreist und schließlich mit der vorgegebenen Genauigkeit bestimmt wird.

Iterationsverfahren werden hauptsächlich zur Lösung von nichtlinearen Gleichungen genutzt, da z.B. Nullstellen von Polynomen vom Grad >4 nicht mehr formelmäßig berechnet werden können. Iterationen finden jedoch auch Anwendung bei der Lösung von großen linearen Gleichungssystemen (z.B. beim Gauß-Seidel-Verfahren).

Ein einfaches Verfahren ist die sog. Fixpunkt-Iteration

$$x_{n+1} = F(x_n)$$

Dabei wird der Graph der Funktion F(x) mit der Geraden y = x zum Schnitt gebracht.

Bei der Regula falsi

$$x_{n+1} = x_n - f(x_n)\frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})}$$

wird die Nullstelle der Verbindungsgerade zweier Graphenpunkte von f(x) bestimmt.

Bessere Konvergenz erzielt das Newton-Verfahren

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

bei dem jeweils die Nullstelle der Tangente bestimmt wird. Die Newton-Iteration läßt sich auf nichtlineare Systeme und somit auch auf komplexe Gleichungen ausdehnen (siehe Programm 32). Programme zu den erwähnten Iterationsverfahren finden sich z.B. in [15] und [17].

Die Müller-Iteration (1956) (in der englischsprachigen Literatur Muller genannt) kann als Verallgemeinerung der Regula falsi aufgefaßt werden. Im Gegensatz zu dieser wird durch 3 Kurvenpunkte eine Parabel gelegt und deren Nullstelle bestimmt. Die Bestimmung der Parabelgleichung erfolgt mittels Interpolation.

Sind x_1, x_2 und x_3 die 3 Kurvenpunkte bzw. die Startwerte, so berechnet man den nächsten Iterationswert aus

$$x = x_3 + l\,(x_3 - x_2)$$

Der Faktor l berechnet sich aus dem Quotienten

$$l = \frac{-2Df(x_3)}{G \pm \sqrt{G^2 - 4DCf(x_3)}}$$

Die Konstanten ergeben sich aus

$$G = L^2\,f(x_1) - D^2\,f(x_2) + (L+D)\,f(x_3)$$

$$C = L\,(Lf(x_1) - D\,f(x_2) + f(x_3))$$

mit

$$L = \frac{x_3 - x_2}{x_2 - x_1}$$

$$D = \frac{x_3 - x_1}{x_2 - x_1}$$

Das Vorzeichen im Nenner von l wird aus numerischen Gründen so gewählt, daß der Nenner möglichst groß wird.

Wegen der im Nenner auftretenden Wurzel könnte das Verfahren auch ins Komplexe gehen. Da die komplexe Newton-Iteration effektiver ist, ist das hier angegebene Müller-Verfahren auf das Reelle beschränkt.

Zum folgenden Programm

Die zu lösende nichtlineare Gleichung ist im Programm in Zeile 170 zu definieren. Der Startwert, die Abbruchgenauigkeit und die max. Zahl der Iterationen werden über INPUT eingegeben.
Liegt der Startwert in der Nähe der gesuchten Nullstelle, so ist die Konvergenz gut. Bei Funktionen, die keine reellen Nullstellen haben, kann die Iteration manchmal zwischen zwei Werten hin- und herpendeln und so Konvergenz vortäuschen. Zur Kontrolle wird daher im Programm noch der Funktionswert des letzten Iterationsschritts mit ausgedruckt.

Als Programmbeispiel wird das kubische Polynom

$$x^3 - x - 1$$

behandelt. Beim Startwert 1 ergibt sich in 5 Iterationsschritten die Nullstelle

$$x = 1.324718$$

auf Maschinengenauigkeit. Der zugehörige Funktionswert ist

$$f(x) = 5.960465 \cdot 10^{-7}$$

```
100 REM Muelleriteration
110 :
120 CLS:PRINT STRING$(25,42)
130 PRINT"    Mueller-Iteration"
140 PRINT STRING$(25,42)
150 :
160 'Definieren der Funktion
170 DEF FNF(X)=X^3-X-1
180 PRINT"f(x)=x^3-x-1"
190 :
200 INPUT"Startwert";X0
210 INPUT"Abbruchgenauigkeit";EPS
220 INPUT"Iterationszahl";IT
230 :
240 'Initialisieren
250 K=1:D=.5
260 X3=X0:X1=X3-D:X2=X3+D
270 PRINT X0
280 :
290 'Mueller-Interpolation
300 IF X2=X1 THEN X2=X2+1E-08
310 L1=(X3-X2)/(X2-X1)
320 D1=(X3-X1)/(X2-X1)
330 IF K>1 THEN 350
340 E1=FNF(X1):E2=FNF(X2)
350 E3=FNF(X3)
360 A1=L1^2*E1-D1^2*E2+(L1+D1)*E3
370 C1=L1*(L1*E1-D1*E2+E3)
380 B=A1^2-4*D1*C1*E3
390 :
400 'Iteration
410 IF B<0 THEN B=0
420 IF A1<>0 THEN A1=A1+SGN(A1)*SQR(B)
430 IF ABS(A1)+ABS(B)=0 THEN A1=4*D1*E3
440 IF A1=0 THEN A1=.0000001
450 L=-2*D1*E3/A1:X=X3+L*(X3-X2)
460 PRINT X
470 :
480 'Konvergenztest
490 IF ABS(X-X3)<EPS*ABS(X3) THEN 560
500 IF K<IT THEN 510 ELSE 530
```

```
510 K=K+1:X1=X2:X2=X3:X3=X
520 E1=E2:E2=E3:GOTO 300
530 PRINT"Keine Konvergenz nach";IT;"Iterationen"
540 PRINT"Letzter ";
550 :
560 PRINT"Funktionswert=";FNF(X3)
570 END
```

```
***************************
   Mueller-Iteration
***************************
f(x)=x^3-x-1
Startwert? 1
Abbruchgenauigkeit? 1e-06
Iterationszahl? 50
 1
 1.31344632
 1.32456943
 1.32471808
 1.32471796
Funktionswert= 5.43427E-07
```

31. MÜLLER-ITERATION (ZWEIDIMENSIONAL)

Die Müller-Iteration (Programm 31) läßt sich auch auf Funktionen

$$f(x, y)$$

zweier Variablen ausdehnen.

Dabei wird die x-y-Ebene in einfachen Müller-Iterationsschritten rechteckig in x- und y-Richtung durchlaufen

$$x = x_3 + l_1 (x_3 - x_2)$$

$$y = y_3 + l_2 (y_3 - y_2)$$

Das Verfahren bricht ab, wenn die Änderung der x- bzw. y-Werte die vorgegebene Abbruchgenauigkeit unterschreitet.

Zum folgenden Programm

Startwerte und Abbruchgenauigkeit werden mittels INPUT eingegeben. Das Programm ist gegen das Verschwinden des Nenners gesichert, jedoch nicht das Überschreiten des Wertebereichs (Overflow).

Als Beispiel wird die Funktion

$$f(x, y) = e^y - x^2 - 2$$

behandelt.

Für die Startwerte x=1, y=1 erhält man in 2 Schritten die Lösung

$$x = 0.84751501, y = 1$$

Analog liefern die Startwerte x=1, y=Ø die Lösung

$$x = -0.7637183, y = 0.9490546$$

Da die Funktion unendlich viele Nullstellen hat, erhält man für verschiedene Startwerte wechselnde Lösungen. Sie erfüllen die Gleichung

$$y = \ln (x^2 + 2)$$

die man erhält, wenn man die gegebene Funktionsgleichung nach y auflöst.

```
100 REM Muelleriteration (2-dimensional)
110 :
120 CLS:PRINT STRING$(39,42)
130 PRINT"   Mueller-Iteration (2-dim.)
140 PRINT STRING$(40,42)
150 :
160 'Definieren der Funktion
170 DEF FNF(X,Y)=EXP(Y)-X^2-2
180 PRINT:PRINT"f(x,y)=exp(y)-x^2-2"
190 :
200 PRINT:INPUT "Startwerte";X0,Y0
210 INPUT"Abbruchgenauigkeit";EPS
220 INPUT"Iterationszahl";IT:PRINT
230 :
240 'Initialisieren
250 K=1:B1=.5:B2=B1
260 :
270 X=X0:Y=Y0
280 X3=X0:X1=X3-B1:X2=X3+B2
290 PRINT X0,Y0
300 :
310 'Mueller-Interpolation
320 IF X2=X1 THEN X2=X2+1E-08
330 L1=(X3-X2)/(X2-X1)
340 D1=(X3-X1)/(X2-X1)
350 IF K>1 THEN 370
360 E1=FNF(X1,Y):E2=FNF(X2,Y)
370 E3=FNF(X3,Y)
380 GOSUB 600
390 B1=L*(X3-X2):X=X3+B1
410 :
420 'Konvergenztest
430 IF ABS(B1)+ABS(B2)<EPS*ABS(X3) THEN 570
440 X0=X:Y3=Y0:Y1=Y3-B2:Y2=Y3+B2
450 IF Y2=Y1 THEN Y2=Y2+.0000001
460 L1=(Y3-Y2)/(Y2-Y1)
470 D1=(Y3-Y1)/(Y2-Y1)
480 E1=FNF(X,Y1):E2=FNF(X,Y2):E3=FNF(X,Y3)
490 GOSUB 600
500 B2=L*(Y3-Y2):Y=Y3+B2
```

```
510 IF ABS(B1)+ABS(B2)<EPS*ABS(X3) THEN 570
520 IF K<IT THEN K=K+1:Y0=Y:GOTO 270
530 :
540 PRINT"Keine Konvergenz nach";IT;"
        Iterationen"
550 PRINT"Letzter ";
560 :
570 PRINT"Funktionswert=";FNF(X3,Y3)
580 END
590 :
600 'Unterprogramm
610 A1=L1^2*E1-D1^2*E2+(L1+D1)*E3
620 C1=L1*(L1*E1-D1*E2+E3)
630 B=A1^2-4*D1*C1*E3
640 IF B<0 THEN B=0
650 IF A1<>0 THEN A1=A1+SGN(A1)*SQR(B)
660 IF ABS(A1)+ABS(B)=0 THEN A1=4*D1*E3
670 IF A1=0 THEN A1=.0000001
680 L=-2*D1*E3/A1
690 RETURN
```

```
****************************************
   Mueller-Iteration (2-dim.)
*****************************************

f(x,y)=exp(y)-x^2-2

Startwerte? 1,1
Abbruchgenauigkeit? 1e-06
Iterationszahl? 30

 1               1
 .847515091      1
Funktionswert= 0
```

```
f(x,y)=exp(y)-x^2-2

Startwerte? 1,0
Abbruchgenauigkeit? 1e-06
Iterationszahl? 30

 1               0
-1               1.20627398
-0.764252978     1.00921891
-0.763719359     0.949423254
-0.763718694     0.949054804
Funktionswert= 5.68107E-07
```

32. NEWTON-ITERATION IM KOMPLEXEN

Die Newton-Iteration im Reellen

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

kann auch auf die komplexe Ebene ausgedehnt werden

$$z_{n+1} = z_n - \frac{f(z_n)}{f'(z_n)}$$

Um eine komplexe Arithmetik zu vermeiden, wird die Funktion f und ihre Ableitung f' in Real- und Imaginärteil zerlegt

$$f(z) = u + iv$$
$$f'(z) = u_x + iu_y$$

Damit erhält man

$$\frac{f}{f'} = \frac{u + iv}{u_x + iu_y} = \frac{uu_x + vu_y}{u_x{}^2 + u_y{}^2} + i\,\frac{vu_x - uu_y}{u_x{}^2 + u_y{}^2}$$

durch Erweitern mit dem Konjugiert-Komplexen des Nenners. Dabei stellen

$$u_x,\ u_y$$

die partiellen Ableitungen von u nach x bzw. y dar. Trennt man Real- und Imaginärteil, so erhält man die komplexe Newton-Iteration

$$x_{n+1} = x_n - \frac{uu_x + vu_y}{u_x{}^2 + u_y{}^2}$$

$$y_{n+1} = y_n - \frac{vu_x - uu_y}{u_x{}^2 + u_y{}^2}$$

Polynome können mit Hilfe der binomischen Formel in Real- und Imaginärteil zerlegt werden. Z.B. gilt

$$\begin{aligned} z^3 + 1 &= (x + iy)^3 + 1 \\ &= x^3 + 3ix^2y - 3xy^2 - iy^3 + 1 \end{aligned}$$

Dies liefert den Realteil $u = x^3 - 3xy^2 + 1$ und Imaginärteil $v = 3x^2\,y - y^3$.

Andere Funktionen können mit Hilfe eines geeigneten Tafelwerks zerlegt werden. Nach [4] gilt z.B.

$$\sin z = \sin x \cosh y + i \cos x \sinh y$$

Dabei treten die hyperbolischen Funktionen sinh bzw. cosh auf.

Zum folgenden Programm

Der Realteil u bzw. Imaginärteil w der Funktion muß im Programm als Unterprogramm in den Zeilen 170—180 vereinbart werden. Die partiellen Ableitungen von u nach x und y folgen in den Programmzeilen 230—240

Die Startwerte, die Abbruchgenauigkeit und die maximale Zahl der Iterationen wird über INPUT-Anweisungen eingegeben.

Als Programmbeispiel wird die Gleichung

$$z^3 + 1$$

gelöst. Die Zerlegung in Real- und Imaginärteil ist

$$u = x^3 - 3xy^2 + 1, \quad v = 3x^2y - y^3$$

die partiellen Ableitungen von u nach x bzw. y sind

$$u_x = 3x^2 - 3y^2, \quad u_y = -6xy$$

Eingabe der Startwerte x=1, y=1 liefert die Nullstelle

Ø.5 + i * Ø.866Ø254

Entsprechend liefern die Startwerte x=1, y=—1 die Nullstelle

Ø.5 — i * Ø.866Ø254

bzw. x=1, y=Ø die Nullstelle

—1 + i * Ø.

Die exakten Nullstellen sind hier

$$-1 \text{ und } \frac{1 \pm \sqrt{3}}{2}$$

```
100 REM Newton-Verfahren im Komplexen
110 :
120 CLS:PRINT STRING$(30,42)
130 PRINT"  Komplexe Newton-Iteration"
140 PRINT STRING$(30,42)
150 :
160 'Definieren Real- u.Imaginaerteil
170 DEF FNU(X,Y)=X^3-3*X*Y^2+1
180 DEF FNV(X,Y)=3*X^2*Y-Y^3
190 PRINT"Realteil=x^3-3*x*y^2+1"
200 PRINT"Imaginaerteil=3*x^2*y-y^3"
210 :
220 'Def. der part.Ableit. von u(x,y)
230 DEF FNX(X,Y)=3*X^2-3*Y^2
240 DEF FNY(X,Y)=-6*X*Y
250 :
260 PRINT:INPUT"Startwert Real,Imag.";XO,YO
270 INPUT"Genauigkeit";EPS
280 INPUT"max.Iterationszahl";IT
290 :
300 'Anfangswerte
310 K=0
320 PRINT XO,YO
330 :
340 'Iteration
350 K=K+1
360 UX=FNX(XO,YO):UY=FNY(XO,YO)
370 G=UX^2+UY^2
380 IF G=0 THEN PRINT"
        Gradient ist Nullvektor":END
390 X=XO+(FNV(XO,YO)*UY-FNU(XO,YO)*UX)/G
400 Y=YO-(FNV(XO,YO)*UX+FNU(XO,YO)*UY)/G
410 PRINT X,Y
420 :
430 'Konvergenztest
440 IF (X-XO)^2+(Y-YO)^2 < EPS*ABS(X) THEN 490
450 IF K<IT THEN XO=X:YO=Y:GOTO 350
```

```
460 PRINT"Keine Konvergenz nach ";IT;"
        Iterationen"
470 PRINT"Letzter Wert=";:GOTO 500
480 :
490 PRINT"Nullstelle=";
500 IF Y<0 THEN 520
510 PRINT X;"+i*";Y:END
520 PRINT X;"-i*";ABS(Y)
530 END
```

```
******************************
  Komplexe Newton-Iteration
******************************
Realteil=x^3-3*x*y^2+1
Imaginaerteil=3*x^2*y-y^3

Startwert Real,Imag.? 1,1
Genauigkeit? 1e-06
max.Iterationszahl? 20
 1              1
 0.666666667    0.833333333
 0.508691916    0.841099874
 0.499329996    0.866269172
 0.499999911    0.866024903
 0.5            0.866025404
Nullstelle= .5 +i* 0.866025404
```

```
*******************************
  Komplexe Newton-Iteration
*******************************
Realteil=x^3-3*x*y^2+1
Imaginaerteil=3*x^2*y-y^3

Startwert Real,Imag.? 1,-1
Genauigkeit? 1e-06
max.Iterationszahl? 20
 1             -1
 0.666666667   -0.833333333
 0.508691916   -0.841099874
 0.499329996   -0.866269172
 0.499999911   -0.866024903
 0.5           -0.866025404
Nullstelle= .5 -i* 0.866025404
```

```
Startwert Real,Imag.? 1,0
Genauigkeit? 1e-06
max.Iterationszahl? 20
 1               0
 0.333333333     0
-2.77777778      0
-1.89505185      0
-1.35618683      0
-1.08535856      0
-1.00653708      0
-1.00004236      0
-1               0
Nullstelle=-1 +i* 0
```

33. NEVILLE-INTERPOLATION

Unter Interpolation versteht man Aufsuchen von Zwischenwerten einer tabellierten Funktion. Im Gegensatz dazu wird bei der Extrapolation ein Wert außerhalb des tabellierten Bereichs gesucht.

Sind die Funktionswerte Z(I) an den Stellen X(I) gegeben und soll an der Stelle X interpoliert werden, so läßt sich die Neville-Interpolation wie folgt beschreiben

```
260 FOR K=1 TO N
270 FOR I=K+1 TO N
280 IF I=N+1 THEN 330
290 Z(I) = (Z(I) * Y(K) – Z(K) * Y(I)) / (Y(K) – Y(I)
300 NEXT I
310 NEXT K
```

Dabei gilt

$$Y(I) = X - X(I)$$

N ist die Anzahl der vorgegebenen Funktionswerte, Z(N) der gesuchte, interpolierte Funktionswert.

Die in Programmzeile 340 auftretenden Differenzen heißen dividierte Differenzen. Mit ihrer Hilfe kann auch das Polynom vom Grad höchstens N–1 bestimmen, das durch die vorgegebenen Punkte geht. Ein Programm zur Newton-Interpolation findet sich in [17].

Zum folgenden Programm

Im Programm wird nach dem Neville-Verfahren im angegebenen Punkt interpoliert.

Folgende Funktionsstellen (x|y) seien gegeben:

$$(-4 \mid -26),\ (-1 \mid 4),\ (0 \mid 2),\ (2 \mid 16)$$

Gesucht werde der Funktionswert im Punkt x=1.

Das Programm liefert den Funktionswert

$$f(1) = 4$$

Das interpolierende Polynom ist hier

$$p(x) = x^3 + 2x^2 - x + 2$$

wie man leicht nachrechnet.

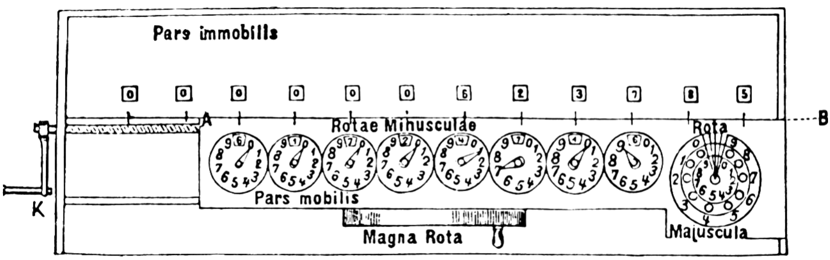
```
100 REM Neville-Interpolation
110 :
120 CLS:PRINT STRING$(25,42)
130 PRINT"  Neville-Interpolation"
140 PRINT STRING$(25,42)
150 :
160 INPUT"Wieviele Stuetzstellen";N
170 DIM X(N),Y(N),Z(N)
180 :
190 PRINT:PRINT"Eingabe der Stuetzstellen"
200 FOR I=1 TO N
210   PRINT USING" x(##),y(##)";I,I;:
      INPUT X(I),Y(I)
220 NEXT I
230 :
240 PRINT:INPUT"x-Wert";X
250 FOR I=1 TO N
260   Z(I)=Y(I):Y(I)=X-X(I)
270 NEXT I
280 :
290 'Interpolation
300 FOR K=1 TO N
310   FOR I=K+1 TO N
320     IF I=N+1 THEN 380
330     IF Y(I)=Y(K) THEN PRINT"y-Werte
        muessen verschieden sein":END
340     Z(I)=(Y(K)*Z(I)-Y(I)*Z(K))/(Y(K)-Y(I))
350   NEXT I
360 NEXT K
370 :
380 PRINT:PRINT"Interpol.Funktionswert=";Z(N)
390 END
```

```
*************************
  Neville-Interpolation
*************************
Wieviele Stuetzstellen? 4

Eingabe der Stuetzstellen
 x( 1),y( 1)? -4,-26
 x( 2),y( 2)? -1,4
 x( 3),y( 3)? 0,2
 x( 4),y( 4)? 2,16

x-Wert? 1

Interpol.Funktionswert= 4
```



"Skizze der Rechenmaschine von Gottfried Leibniz (Entwurf 1673)"

34. GAUSS-JORDAN-VERFAHREN

Löst man im Gleichungssystem

$$a_{11} x_1 + a_{12} x_2 + \ldots + a_{1n} x_n = b_1$$

$$a_{21} x_1 + a_{22} x_2 + \ldots + a_{2n} x_n = b_2$$

$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$a_{n1} x \;\; + a_{n2} x_2 + \ldots + a_{nn} x_n = b_n$$

die k-te Gleichung nach der Unbekannten x_k auf, so erhält man

$$x_k = (-a_{k1} x_1 - a_{k2} x_2 - \ldots - a_{kn} x_n + b_k) / a_{kk}$$

Dies setzt voraus, daß das Diagonalelement $a_{kk} \neq \emptyset$ ist. Ist $a_{kk} = \emptyset$ oder sehr klein, so führt man eine entsprechende Spalten- oder Zeilenvertauschung durch. Findet sich keine Zeile oder Spalte, die ein Element $\neq \emptyset$ liefert, so ist die entsprechende Matrix singulär, d.h. die Determinante der Matrix verschwindet und das Gleichungssystem ist nicht eindeutig lösbar.

Setzt man diese Unbekannte x_k in alle anderen Gleichungen ein, so verschwindet sie aus diesen Gleichungen, d.h. sie wird eliminiert. Führt man dies für alle Unbekannten durch, so bleibt in der 1. Zeile nur noch die x_1, in der 2. noch x_2 und entsprechend in der k-ten Zeile x_k übrig.

Man hat somit ein Gleichungssystem erhalten, in dem die Unbekannten nur noch in der Diagonale der Matrix stehen

$$\begin{array}{llllll} d_{11} x_1 & & & & & = c_1 \\ & d_{22} x_2 & & & & = c_2 \\ & & d_{33} x_3 & & & = c_3 \\ & & & \cdot & & \\ & & & & \cdot & \ldots \\ & & & & \cdot & \\ & & & & d_{nn} x_n & = c_n \end{array}$$

Dieses Gleichungssystem kann sehr einfach gelöst werden

$$x_k = \frac{c_k}{d_{kk}}; \; d_{kk} \neq \emptyset$$

Lineare Gleichungssysteme finden zahlreiche Anwendungen bei

– Dimensionierung elektrischer Schaltkreise
– der Lösung von partiellen Differentialgleichungen
– Vektorrechnung
– Input- und Outputmodellen der Wirtschaft
– Ausgleichsrechnung
– elastische Verformungen von Fachwerken, Karrosserien

usw.

Zur Konstruktion eines Flugzeugflügels treten z.B. Gleichungssysteme mit mehr als 20.000 Unbekannten auf.

Zum folgenden Programm

Das Programm löst beliebige lineare Gleichungssysteme, bei denen die Zahl der Unbekannten mit der Zahl der Gleichungen übereinstimmt.
Gibt es mehr Unbekannte als Gleichungen, so heißt das System unterbestimmt, im umgekehrten Fall überbestimmt. Zur Lösung von überbestimmten Gleichungssystemen, wie sie bei der linearen Regression auftreten, gibt es spezielle Verfahren wie z.B. die Methode der kleinsten Quadrate.

Als Programmbeispiel wird das Gleichungssystem mit 6 Unbekannten behandelt.

$$\begin{aligned}
6x_1 - 3x_2 + 2x_3 + x_4 - x_5 + x_6 &= 11 \\
-3x_1 - 7x_2 + 4x_4 - 2x_5 + x_7 &= -5 \\
4x_1 - 3x_2 + 6x_3 - x_4 + 2x_5 + x_7 &= 28 \\
2x_1 + 4x_2 + 5x_3 - 7x_4 - 3x_5 + 2x_7 &= -6 \\
-x_1 + 5x_2 - 4x_3 + 8x_5 - 2x_7 &= 25 \\
3x_1 + 4x_3 - 2x_4 + 5x_5 - 6x_7 &= -4
\end{aligned}$$

Die Anzahl der Unbekannten und die Gleichungsmatrix mit rechter Seite wird im Programm in Form von DATA-Werten eingelesen.

Das Programm liefert hier die exakten Lösungen

$$x_1 = 1,\ x_2 = 2,\ x_3 = 3,\ x_4 = 4,\ x_5 = 5 \text{ und } x_6 = 6.$$

```
100 REM Gauss-Jordan-Verfahren
110 :
120 CLS:PRINT STRING$(27,42)
130 PRINT" Gauss-Jordan-Verfahren"
140 PRINT STRING$(27,42)
150 :
160 DEFINT I-N
170 READ N:'Anzahl der Unbekannten
180 DIM A(N,N+1),X(N)
190 :
200 PRINT"Gegebenes Gleichungsystem:"
210 FOR I=1 TO N
220 FOR J=1 TO N+1
230   READ A(I,J):PRINT A(I,J);
240 NEXT J:PRINT
250 NEXT I:PRINT
260 :
270 FOR K=1 TO N
280   IF K=N THEN 370
290  G=A(K,K):M=K
300   FOR I=K+1 TO N
310     IF ABS(A(I,K))<=G THEN 330
320     G=ABS(A(K,K)):M=I
330   NEXT I
340   FOR J=K TO N+1
350     H=A(K,J):A(K,J)=A(M,J):A(M,J)=H
360   NEXT J
370   IF ABS(A(K,K))<.0000001 THEN
      PRINT"Matrix singulaer":END
380   FOR I=1 TO N
390     IF I=K THEN 440
400     F=A(I,K)/A(K,K)
410     FOR J=K+1 TO N+1
420       A(I,J)=A(I,J)-A(K,J)*F
430     NEXT J
440   NEXT I
450 NEXT K
460 :
470 PRINT"Loesung:"
480 FOR I=1 TO N
490   X(I)=A(I,N+1)/A(I,I)
500   PRINT"x(";I;")=";X(I)
```

```
510 NEXT I
520 END
530 :
540 DATA 6
550 DATA 6,-3,2,1,-1,1,11
560 DATA -3,-7,0,4,-2,1,-5
570 DATA 4,-3,6,-1,2,1,28
580 DATA 2,4,5,-7,-3,2,-6
590 DATA -1,5,-4,0,8,-2,25
600 DATA 3,0,4,-2,5,-6,-4
```

```
***************************
 Gauss-Jordan-Verfahren
***************************
Gegebenes Gleichungsystem:
 6 -3  2  1 -1  1  11
-3 -7  0  4 -2  1 -5
 4 -3  6 -1  2  1  28
 2  4  5 -7 -3  2 -6
-1  5 -4  0  8 -2  25
 3  0  4 -2  5 -6 -4

Loesung:
x( 1 )= 0.999999998
x( 2 )= 2.00000002
x( 3 )= 3.00000002
x( 4 )= 4.00000002
x( 5 )= 5
x( 6 )= 6
```

35. MATRIZEN-INVERSION NACH FADDEJEW

Eine einfache Methode zur Invertierung einer quadratischen Matrix, die zugleich noch das charakteristische Polynom liefert, ist das Verfahren von Faddejew [9].

Ist die zu invertierende Matrix $\mathbf{A}_1$ von der Ordnung n, so wird iterativ die Matrix $\mathbf{A}_n$ in n−1 Schritten über die Gleichung

$$\mathbf{A}_{i+1} = \mathbf{A}_1 \, (\mathbf{A}_i - c_i \, \mathbf{E})$$

berechnet; dabei ist **E** die Einheitsmatrix und c_i = Spur $(\mathbf{A}_i)/i$. Die Spur einer Matrix ist die Summe der Diagonalelemente.

Die gesuchte inverse Matrix ergibt sich dann aus

$$\mathbf{A}_1^{-1} = (\mathbf{A}_{n-1} - c_{n-1} \, \mathbf{E}) \, \frac{1}{c_n}$$

Der Wert c_n liefert außerdem noch den Wert der Determinante

$$\mathrm{Det}(\mathbf{A}) = (-1)^{n+1} \, c_n$$

Programme zu anderen Inversionsverfahren finden sich z.B. in [17].

Zum folgenden Programm

Das Programm berechnet die inverse Matrix nach dem angegebenen Verfahren. Die zu invertierende Matrix wird in Form von DATA-Werten eingelesen.

Als Beispiel soll die Inverse von

$$\mathbf{A} = \begin{pmatrix} 5 & 7 & 6 & 5 \\ 7 & 10 & 8 & 7 \\ 6 & 8 & 10 & 9 \\ 5 & 7 & 9 & 10 \end{pmatrix}$$

berechnet werden. Das Programm liefert hier die exakte Inverse

$$\mathbf{A}^{-1} = \begin{pmatrix} 68 & -41 & -17 & 10 \\ -41 & 25 & 10 & -6 \\ -17 & 10 & 5 & -3 \\ 10 & -6 & -3 & 2 \end{pmatrix}$$

Durch Matrizen-Multiplikation (Programm 22) kann geprüft werden, daß gilt

$$\mathbf{A}\mathbf{A}^{-1} = \mathbf{A}^{-1}\,\mathbf{A} = \mathbf{E}.$$

```
100 REM Matrizeninversion nach Faddejew
110 :
120 CLS:PRINT STRING$(25,42)
130 PRINT"  Matrizeninversion"
140 PRINT STRING$(25,42)
150 :
160 DEFINT I-N
170 READ N:'Ordnung der Matrix
180 DIM A(N,N),A1(N,N),B(N,N),G(N,N),H(N,N)
190 :
200 PRINT"Gegebene Matrix:"
210 FOR I=1 TO N
220 FOR J=1 TO N
230   READ A(I,J):PRINT A(I,J);
240   A1(I,J)=A(I,J)
250 NEXT J:PRINT
260 NEXT I:PRINT
270 :
280 FOR I=1 TO N
290   S=0
300   FOR J=1 TO N
310     S=S+A(J,J)
320   NEXT J
330   C=S/I
340   :
350   FOR J=1 TO N
360   FOR K=1 TO N
370     IF J=K THEN H(J,J)=A(J,J)-C
        ELSE H(J,K)=A(J,K)
380   NEXT K
390   NEXT J
400   :
410   IF I=N-1 THEN 420 ELSE 480
420   FOR J=1 TO N
430   FOR K=1 TO N
440     G(J,K)=H(J,K)
450   NEXT K
460   NEXT J
470   :
480   FOR J=1 TO N
490   FOR K=1 TO N
500     S=0
```

```
510      FOR L=1 TO N
520      S=S+A1(J,L)*H(L,K)
530      NEXT L
540      A(J,K)=S
550    NEXT K
560    NEXT J
570 NEXT I
580 :
590 IF C=0 THEN PRINT"Matrix singulaer":END
600 PRINT"Inverse Matrix:"
610 FOR I=1 TO N
620 FOR J=1 TO N
630   G(I,J)=G(I,J)/C
640   PRINT G(I,J);
650 NEXT J:PRINT
660 NEXT I:PRINT
670 D=(-1)^(N+1)*C
680 PRINT"Determinante=";D
690 END
700 :
710 DATA 4
720 DATA 5,7,6,5
730 DATA 7,10,8,7
740 DATA 6,8,10,9
750 DATA 5,7,9,10
```

```
**************************
  Matrizeninversion
**************************
Gegebene Matrix:
 5  7  6  5
 7  10  8  7
 6  8  10  9
 5  7  9  10

Inverse Matrix:
 68 -41 -17  10
-41  25  10 -6
-17  10  5 -3
 10 -6 -3  2

Determinante= 1
```

36. CHARAKTERISTISCHES POLYNOM NACH FADDEJEW

Das charakteristische Polynom hat genau die Eigenwerte der Matrix als Nullstellen.

Eigenwerte λ und Eigenvektoren **x** sind Lösung der Gleichung

$$\mathbf{A}\mathbf{x} = \lambda\, \mathbf{x}$$

Eigenwerte von Matrizen geben an

- Klassifikation von Flächen n-ter Ordnung
- Schwingungsfrequenzen eines Systems
- Kreiseleigenschaften von festen Körpern
- Stationäres Verhalten von Markow-Ketten

usw.

Wie beim vorhergehenden Programm entstehen beim Faddejew-Verfahren [9]

$$\mathbf{A}_{i+1} = \mathbf{A}_1\, (\mathbf{A}_i - c_i\, \mathbf{E}) \qquad i = 1, 2, \ldots, n-1$$

die Koeffizienten c_i aus

$$c_i = \mathrm{Spur}\, (\mathbf{A}_i)\, / i$$

Versieht man die Werte c_i mit alternierenden Vorzeichen, so erhält man die Koeffizienten des charakteristischen Polynoms der Matrix. Bestimmt man die Nullstellen des Polynoms mit Hilfe der Programme 17, 18, 31, 32 oder 33, so erhält man die gesuchten Eigenwerte der Matrix.

Programme zu anderen Eigenwertverfahren, die auch gleichzeitig die Eigenvektoren bestimmen, finden sich in [17].

Zum folgenden Programm

Analog zum vorhergehenden Programm werden die Koeffizienten c_i bestimmt und mit den richtigen Vorzeichen versehen. Die Matrix wird wieder in Form von DATA-Werten eingelesen.

Als Beispiel wird die Matrix

$$\mathbf{A} = \begin{pmatrix} 6 & 4 & 4 & 1 \\ 4 & 6 & 1 & 4 \\ 4 & 1 & 6 & 4 \\ 1 & 4 & 4 & 6 \end{pmatrix}$$

behandelt. Das Programm liefert das charakteristische Polynom

$$p(x) = x^4 - 24x^3 + 15\emptyset x^2 - 2\emptyset\emptyset x - 375$$

dessen Nullstellen bereits in Programm 17 bestimmt worden sind. Die Eigenwerte von **A** sind somit

$$\lambda_1 = -1 \ ; \ \lambda_2 = 5 \ ; \ \lambda_3 = -5 \ ; \ \lambda_4 = 15.$$

```
100 REM Charakt.Polynom nach Faddejew
110 :
120 CLS:PRINT STRING$(30,42)
130 PRINT" Charakt.Polynom einer Matrix"
140 PRINT STRING$(30,42)
150 :
160 READ N:'Ordnung der Matrix
170 DIM A(N,N),A1(N,N),G(N,N),H(N,N),C(N)
180 :
190 PRINT"Gegebene Matrix:"
200 FOR I=1 TO N
210 FOR J=1 TO N
220   READ A(I,J):PRINT A(I,J);
230   A1(I,J)=A(I,J)
240 NEXT J:PRINT
250 NEXT I:PRINT
260 :
270 FOR I=1 TO N
280   S=0
290   FOR J=1 TO N
300     S=S+A(J,J)
310   NEXT J
320   C(I)=S/I
330   :
340   FOR J=1 TO N
350   FOR K=1 TO N
360     IF J=K THEN H(J,J)=A(J,J)-C(I)
                ELSE H(J,K)=A(J,K)
370   NEXT K
380   NEXT J
390   :
400   IF I=N-1 THEN 410 ELSE 470
410   FOR J=1 TO N
420   FOR K=1 TO N
430     G(J,K)=H(J,K)
440   NEXT K
450   NEXT J
460   :
470   FOR J=1 TO N
480   FOR K=1 TO N
490     S=0
500     FOR L=1 TO N
```

```
510        S=S+A1(J,L)*H(L,K)
520        NEXT L
530        A(J,K)=S
540     NEXT K
550     NEXT J
560 NEXT I
570 :
580 PRINT"Koeff.des charakt.Polynoms:"
590 IF (N MOD 2)=0 THEN S=-1 ELSE S=1
600 PRINT-S;
610 FOR I=1 TO N
620 C(I)=S*C(I)
630 PRINT C(I);
640 NEXT I:PRINT
650 END
660 :
670 DATA 4
680 DATA 6,4,4,1
690 DATA 4,6,1,4
700 DATA 4,1,6,4
710 DATA 1,4,4,6
```

```
*******************************
 Charakt.Polynom einer Matrix
*******************************
Gegebene Matrix:
 6   4   4   1
 4   6   1   4
 4   1   6   4
 1   4   4   6

Koeff.des charakt.Polynoms:
 1 -24  150 -200 -375
```

37. NUMERISCHE DIFFERENTIATION

Obwohl Funktionen – im Gegensatz zur Integration – analytisch differenziert werden können, ist dennoch das folgende Programm von Interesse, da es einen Überblick über Funktions- und Ableitungswerte im ganzen Intervall ermöglicht.

Das Programm benützt die 3-Punkte-Formeln zur Ableitung:

$$f'(x) = \frac{-3f(x) + 4f(x+h) - f(x+2h)}{2^h}$$

$$f''(x) = \frac{(f(x) - 2f(x+h) + f(x+2h)}{h^2}$$

Da die Formeln nach links "übergreifen", ist zu beachten, daß die Funktion auch links vom betrachteten Intervall definiert sein muß.

Da in BASIC die Verkettung von Funktionen implementiert ist, können die beiden Formeln direkt als Funktionen mittels

DEF FN. .

vereinbart werden.

Dazu muß nur in Zeile 170 die zu differenzierende Funktion als

DEF FNF(x) = . . .

definiert werden.

Das Intervall und die Schrittweite h werden über eine INPUT-Anweisung eingegeben. Wegen der unvermeidlichen Rundungsfehler werden alle Funktionswerte im Programm auf 4 Dezimale gerundet. Dies kann, wenn gewünscht, leicht geändert werden. Die Schrittweite zur Ausgabe ist im Programm mit der Variablen H (Zeile 370) gegeben; diese kann notfalls geändert werden.

Zum folgenden Programm

Als Programmbeispiel wird die Funktion

$$f(x) = x^2e^{-x} + \sin x$$

gewählt. Das Intervall soll [1,2], die Schrittweite Ø.1 sein. Wie man dem Programmausdruck entnimmt, wächst der Funktionswert bis x = 1.6 an und nimmt rechts davon wieder ab. Die Funktion hat somit zwischen 1.6 und 1.7 ein Maximum, wie es auch am Vorzeichenwechsel der 1. Ableitung erkennbar ist. Die 2. Ableitung ist im ganzen Intervall negativ, d.h. der Graph ist rechtsgekrümmt.

```
100 REM Numerische Differentiation
110 :
120 CLS:PRINT STRING$(25,42)
130 PRINT" Numerische Differentiation"
140 PRINT STRING$(25,42)
150 :
160 'Definieren der Funktion
170 DEF FNF(X)=X^2*EXP(-X)+SIN(X)
180 PRINT:PRINT"f(x)=x^2*exp(-x)+sin(x)"
190 :
200 '1.Ableitung
210 DEF FNA(X)=(-3*FNF(X)+4*FNF(X+H)-FNF(X+2*H))/(2*H)
220 :
230 '2.Ableitung
240 DEF FNB(X)=(FNF(X)-2*FNF(X+H)+FNF(X+2*H))/H^2
250 :
260 INPUT "Intervall A,B";A,B
270 IF A>B THEN H=A:A=B:B=H
280 INPUT"Schrittweite";H:PRINT
290 :
300 :
310 PRINT"  x         F(x)          F'(x)        F''(x)"
320 PRINT STRING$(39,45)
330 X=A
340 WHILE X<=B+H/2
350   F=FNF(X):F1=FNA(X):F2=FNB(X)
360   PRINT USING" ##.# #####.#### ######.#### #####.####";X,F,F1,F2
370   X=X+H
380 WEND
390 END
```

```
***************************
 Numerische Differentiation
***************************

f(x)=x^2*exp(-x)+sin(x)
Intervall A,B? 1,2
Schrittweite? .1

    x          F(x)          F'(x)         F''(x)
---------------------------------------------------
   1.0        1.2094         0.9106        -1.2854
   1.1        1.2940         0.7848        -1.3399
   1.2        1.3658         0.6525        -1.3735
   1.3        1.4241         0.5159        -1.3885
   1.4        1.4688         0.3769        -1.3867
   1.5        1.4995         0.2374        -1.3700
   1.6        1.5164         0.0989        -1.3396
   1.7        1.5196        -0.0372        -1.2969
   1.8        1.5094        -0.1696        -1.2431
   1.9        1.4862        -0.2971        -1.1792
   2.0        1.4506        -0.4187        -1.1062
```

38. NUMERISCHE INTEGRATION (TRAPEZREGEL)

Da viele Funktionen wie

$$\sin(x^2),\ e^{-x^2},\ \frac{\sin x}{x}$$

nicht analytisch integriert werden können, ist man hier auf numerische Integration angewiesen. Es gibt zahlreiche Integrationsmethoden. Ein bekanntes Verfahren ist die Trapezregel (auch Sehnentrapez-Regel) genannt:

$$\int_a^b f(x)dx = \frac{h}{2}\left[f(a) + 2f(a+h) + 2f(a+2h) + \ldots + 2f(b-h) + f(b)\right]$$

mit der Schrittweite $h = \frac{b-a}{n}$, n Zahl der Intervallunterteilungen.

Die Trapezregel eignet sich auch zur Extrapolation (siehe [33], [35], [36]). Weitere Programme zur numerischen Integration finden sich in [15] und [17].

Zum folgenden Programm

Das Programm berechnet das bestimmte Integral einer Funktion nach der angegebenen Trapezformel.

Die zu integrierende Funktion ist in Zeile 170 mittels

DEF FNF(X) = . . .

zu definieren. Die Intervallgrenzen a, b und Anzahl n der Unterteilungen werden über INPUT eingegeben.

Als Beispiel wird das Integral

$$\int_{\emptyset}^{1} \frac{4}{1+x^2}\,dx$$

mit 5Ø Unterteilungen berechnet. Das Programm liefert

3.141526

Vergleicht man dies mit dem exakten Wert

$\pi = 3.14159265$

so liefert dies den Fehler

$7 \cdot 10^{-5}$.

```
100 REM Numerische Integration
110 :
120 CLS:PRINT STRING$(25,42)
130 PRINT"   Trapezregel"
140 PRINT STRING$(25,42)
150 :
160 'Definieren der Integrandenfunktion
170 DEF FNF(X)=4/(X^2+1)
180 PRINT "f(x)=4/(x^2+1)"
190 :
200 INPUT "Intervallgrenzen";A,B
210 IF A>B THEN H=A:A=B:B=H
220 INPUT "Anzahl der Intervallteilungen";N
230 :
240 H=(B-A)/N:'Schrittweite
250 I=FNF(A)+FNF(B)
260 S=0
280 FOR K=1 TO N-1
290   S=S+FNF(A+K*H)
300 NEXT K
310 INTEGRAL=(I+2*S)*H/2
320 :
330 PRINT"Integral=";INTEGRAL
340 END
```

```
*************************
   Trapezregel
*************************
f(x)=4/(x^2+1)
Intervallgrenzen? 0,1
Anzahl der Intervallteilungen? 50
Integral= 3.14152599
```

39. HAMMING-VERFAHREN

Bekannte Verfahren zur Integration von gewöhnlichen Differentialgleichungen sind die Verfahren von Runge-Kutta und Adams-Bashforth (siehe [33], [35], [37]. Programme dazu finden sich z.B. in [17].
Ein neueres Verfahren stammt von R.W. Hamming (1959) [12].

Die Differentialgleichung

$$y' = f(x,y)$$

mit dem Anfangswert

$$y(a) = y\emptyset \text{ mit } x \in [a,b]$$

habe die Lösungsfunktion u(x). Im folgenden wird

$$u(x_i) = u_i,\ f(x_i, u_i) = f_i$$

gesetzt. Das Hamming-Verfahren läuft wie folgt in 5 Schritten ab:

(1) Zunächst wird mit dem Prädiktor der Funktionswert

$$v_{i+1} = u_{i-3} + \frac{4}{3} h\ (2f_i - f_{i-1} + 2f_{i-2})$$

vorhergesagt.

(2) Dieser Prädiktorwert wird mittels des vorhergehenden Korrektorwerts z_i modifiziert

$$w_{i+1} = v_{i+1} - \frac{112}{121}(z_i - v_i)$$

(3) Für diesen modifizierten Wert wird die rechte Seite der Differentialgleichung bestimmt

$$w'_{i+1} = f(x_{i+1}, w_{i+1})$$

(4) Der neue Korrektor lautet

$$z_{i+1} = \frac{1}{8}(9u_i - u_{i-2} + 3h\ (w_{i+1} + 2f_i - f_{i-1}))$$

(5) Der gesuchte Funktionswert u_{i+1} wird schließlich mit Hilfe des Korrektors verbessert

$$u_{i+1} = z_{i+1} + \frac{9}{121}(z_{i+1} - v_{i+1})$$

Ist der rechte Rand des Integrationsintervalls [a, b] noch nicht erreicht, so wird x um die Schrittweite h erhöht und das Hamming-Verfahren erneut angewandt.

Wie man der Prädiktorformel entnimmt, benötigt man bei jedem Schritt die Kenntnis der Funktionswerte f_i, f_{i-1} und f_{i-2}. Man sagt, das Verfahren ist dreifach zurückgreifend. Da aber nur der Startwert $u_\emptyset$ und somit auch $f_\emptyset$ bekannt ist, müssen die fehlenden Funktionswerte f_1, f_2 mit Hilfe eines anderen Verfahrens berechnet werden. Man wählt hier meist die Methode von Runge-Kutta.

Zum folgenden Programm

Als Beispiel soll die gewöhnliche Differentialgleichung

$$y' = y - 2x/y$$

mit dem Anfangswert $y(\emptyset) = 1$ im Intervall $[\emptyset,1]$ integriert werden.

Die Differentialgleichung wird in Zeile 170 in der Form

170 F = Y – 2 * X/Y

vereinbart. Das Intervall [a,b], der Anfangswert $y_\emptyset$ und die Schrittweite h werden in From von DATA-Werten eingelesen. Die numerische Lösung kann dem Programm-Ausdruck entnommen werden.

Die exakte Lösung der Differentialgleichung lautet

$$y = \sqrt{2x + 1}$$

Der relative Fehler am rechten Intervallrand ist somit

$$3.6 \cdot 1\emptyset^{-6}.$$

```
100 REM Hamming-Verfahren
110 :
120 CLS:PRINT STRING$(25,42)
130 PRINT"    Hamming-verfahren"
140 PRINT STRING$(25,42)
150 :
160 'Differentialgleichung in der Form f'=f(x,y)
170 DEF FNF(X,Y)=Y-2*X/Y
180 PRINT TAB(5)"y'=y-2*x/y"
190 :
200 INPUT "Intervallgrenzen a,b";A,B
210 IF A>B THEN H=A:A=B:B=H
220 INPUT "Anfangswert der Funktion";Y
230 INPUT "Integrations-Schrittweite";H
240 DIM Y(5),F(5)
250 :
260 'Anfangswerte
270 X=A:Y(1)=Y
280 PRINT"    x            y"
290 PRINT STRING$(25,45)
300 PRINT USING " ##.#        #####.######";X,Y
310 :
320 'Startprozedur nach Runge-Kutta
330 FOR I=1 TO 3
340   K1=H*FNF(X,Y)
350   K2=H*FNF(X+H/2,Y+K1/2)
360   K3=H*FNF(X+H/2,Y+K2/2)
370   K4=H*FNF(X+H,Y+K3)
380   Y=Y+(K1+2*K2+2*K3+K4)/6
390   X=X+H:Y(I+1)=Y
400   F(I+1)=FNF(X,Y)
410   PRINT USING " ##.#        #####.######";X,Y
420 NEXT I
430 :
440 'Hamming-Verfahren
450 K=0:P1=0:X=X+H
460 WHILE X<=B+H/2
470   'Praediktor
480   P=Y(1)+4*H*(2*F(2)-F(3)+2*F(4))/3
490   'Modifikator
500   M=P-112*(K-P1)/121
```

```
510     Y=M:M=FNF(X,Y)
520     'Korrektor
530     K=(9*Y(4)-Y(2)+3*H*(M+2*F(4)-F(3)))/8
540     'Schrittfunktion
550     Y(5)=K+9*(K-P)/121
560     PRINT USING " ##.#        #####.######";X,Y
570     F(5)=FNF(X,Y(5)):P1=P
580     FOR I=1 TO 4
590       Y(I)=Y(I+1):F(I)=F(I+1)
600     NEXT I
610     X=X+H
620 WEND
630 END
```

```
***************************
    Hamming-verfahren
***************************
    y'=y-2*x/y
Intervallgrenzen a,b? 0,1
Anfangswert der Funktion? 1
Integrations-Schrittweite? .1
    x              y
---------------------------
  0.0           1.000000
  0.1           1.095446
  0.2           1.183217
  0.3           1.264912
  0.4           1.341564
  0.5           1.414103
  0.6           1.483179
  0.7           1.549159
  0.8           1.612438
  0.9           1.673320
  1.0           1.732057
```

ANHANG A

HISTORISCHE RECHENAUFGABEN

Altbabylonische Keilschrifttafeln (ca. 2000 v. Chr.)

1) Länge und Breite habe ich multipliziert und so die Fläche gemacht. Wiederum was die Länge über die Breite hinausgeht zur Fläche, habe ich addiert und (es ergibt) 183. Wiederum Länge und Breite addiert (gibt) 27. Länge, Breite und Fläche (ist) was?. *Lösung: (15, 12, 180) (14, 13, 182)*

2) 1/4 Breite und Länge sind 7 Handbreiten, Länge und Breite zusammen sind 10 Handbreiten. *Lösung (6,4)*

3) Die Fläche und die Seite des Quadrates habe ich addiert und 0,75 ist es. *Lösung: 1/2*

4) Ein Balken, 0,5 lang (steht senkrecht) an einer Wand. Oben ist er um 0,1 herabgekommen. Von unten (wie weit hat er sich entfernt)? *Lösung: 3/10*

5) Finde, wie lange es dauert, bis eine Geldsumme sich bei 1/5 Jahreszins verdoppelt. *Lösung: 3.801784*

Papyrus Rhind (ca. 1700 v. Chr.)

6) Haufen: sein Ganzes, 2/3 hinzu, (von der Summe) 1/3 hinweg, bleibt 30. *Lösung: 27*

7) 7 Häuser, in jedem 7 Katzen, jede frißt 7 Mäuse, jede von diesen 7 Ähren, von jeder Ähre könnte man 7 Scheffel Korn ernten, wieviele Dinge sind es insgesamt)? *Lösung: 19607*

8) Zu verteilen 100 Brote unter 5 Personen in arithmetischer Folge, daß 1/7 der Summe der drei ersten Anteile gleich der Summe der letzten drei Anteile ist. *Lösung: (38 1/3, 29 1/6, 20, 10 5/6, 1 2/3)*

9) Ein Quadrat und ein weiteres, dessen Seitenlänge 3/4 des ersten ist, bilden die Fläche 100. Wie lang ist die Seite des ersten? *Lösung: 8*

10) Haufen, 2/3 davon, 1/2 davon, 1/7 davon, zusammen genommen ergeben 33. *Lösung: 14 28/97*

Anthologia Graeca (ca. 500 n. Chr.)

11) Wieviele Äpfel werden gebraucht, wenn 4 von 6 Personen 1/3, 1/8, 1/4 und 1/5 jeweils vom Ganzen erhalten, während die 5. Person 10 Äpfel erhält und genau ein Apfel für die 6. Person übrig bleibt. *Lösung: 120*

12) Demarchos verlebte 1/4 seines Lebens als Junge, 1/5 als Heranwachsender, 1/3 als Mann und verbrachte 13 Jahre im Alter. Wie alt ist er geworden? *Lösung: 60*

13) Ziegelmacher, ich muß dringend an meinem Haus weiterbauen. Es ist schön heute und ich brauche nicht mehr als die üblichen 300 Ziegel. Du allein kannst soviele am Tage machen, dein Sohn 200 und dein Schwiegerson 250. Wenn ihr alle drei miteinander arbeitet, in welcher Zeit könnt ihr die geten Ziegel herstellen? *Lösung: 2/5*

14) Ich bin ein Löwe aus Erz; aus meinen Augen, aus meinem Mund und aus meiner rechten Fußsohle springen Fontänen hervor. Mein rechtes Auge füllt das Becken in 2 Tagen, das linke in 3 und mein Fuß in 4 Tagen. Wie lange dauerts, wenn alles vereint? *Lösung: 3 33/37*

Algebra des al-Hwarizmi (ca. 825 n. Chr.)

15) Ein Quadrat und zehn seiner Wurzeln sind gleich 39. *Lösung 3, –13*

16) Ein Quadrat und 21 sind gleich zehn seiner Wurzeln. *Lösung 3,7*

17) Ich habe 10 in zwei Teile geteilt. Dann habe ich den einen Teil durch den anderen geteilt und der Quotient war 4. *Lösung (8,2)*

18) Ich habe 10 in zwei Teile geteilt. Dann habe ich jeden Teil mit sich selbst multipliziert, das kleinere vom größeren abgezogen und der Rest war 40. *Lösung (7,3)*

"Aufgaben zur Schärfung des Verstandes" von Alkuin (ca. 790)

19) Wenn 100 Buschel Korn so auf 100 Leute verteilt werden, daß jeder Mann 3 Buschel, jede Frau 2 Buschel und jedes Kind 1/2 Buschel erhält, ist die Frage, wieviel Männer, Frauen und Kinder waren es?
Lösung: (17, 5, 78) (14, 10, 76) (11, 15, 14) (8, 20, 72) (5, 25, 70)

20) 30 Körbe, 10 davon voll, 10 halbvoll und 10 leer, sollen so unter drei Söhne verteilt werden, daß die Körbe und Inhalte gleich verteilt sind. Wie kann dies geschehen? *Lösung: z.B. 10 halbvolle*

21) Ein Hund jagt ein Kaninchen, das 150 Fuß Vorsprung hat. Der Hund springt immer 9 Fuß weit, wenn das Kaninchen 7 Fuß springt. Nach wieviel Sprüngen hat der Hund das Kaninchen eingeholt? *Lösung: 75*

22) Ein Wolf, eine Ziege und ein Kohlkopf müssen in einer Fähre einen Fluß überqueren, die außer dem Fährmann noch einen weiteren Gegenstand befördern kann. Wie muß der Fährmann sie über den Fluß bringen, damit weder der Wolf die Ziege noch die Ziege den Kohl frißt?

Lilavati des Bhaskara II (1150)

23) Das Quadrat aus 1/8 einer Herde Affen tummelt sich im Walde, während die übrigen zwölf auf dem Gipfel eines Hügels brüllen. *Lösung: 16, 48*

24) Jemand hat 300 Rupien und 6 Pferde. Ein anderer hat 10 Pferde vom gleichen Wert, aber eine Schuld von 100 Rupien. Beide haben dasselbe Vermögen. Was ist der Preis eines Pferdes? *Lösung: 57 1/7*

25) Schönes Mädchen mit den glitzernden Augen, sage mir, so du die richtige Kunst der Umkehrung verstehst: Welches ist die Zahl, die mit 3 vervielfacht, sodann um 3/4 des Produkts vermehrt, durch 7 geteilt, um ein Drittel des Quotienten vermindert, durch Ausziehung der Quadratwurzel, Addition von 8 und Division durch 10 die Zahl 2 hervorbringt? *Lösung: 28*

26) Von einem Schwarm Bienen läßt sich 1/5 auf einer Kadambablüte, 1/3 auf der Silindhablume nieder. Der dreifache Unterschied der beiden Zahlen flog nach den Blüten einer Kutuja, eine Biene blieb übrig, die in der Luft hin und her schwebt, gleichzeitig angezogen von dem lieblichen Duft einer Jasmine und eines Pandamus. Sage mir, reizendes Weib, die Anzahl der Bienen!
Lösung: 15

Liber abaci des Leonardo von Pisa (1202),-

27) Jemand reist in Geschäften nach Lucca, von da nach Florenz und von da heim nach Pisa. In jeder Stadt verdient er so viel, daß sich sein Geld jeweils verdoppelt; in jeder Stadt gibt er aber auch 12 Denare aus. Nach der Heimkehr hat er nichts mehr. *Lösung: 10 1/2*

28) Erhält ein Mann vom zweiten 7 Denare, so hat er fünfmal so viel wie der zweite; erhält der zweite Mann vom ersten fünf Denare, so hat er siebenmal so viel wie der erste. Wieviel hatte jeder ursprünglich?
Lösung: (12 7/17, 46 2/17)

29) Ein König sandte 30 Mann in seinen Obstgarten um Bäume zu pflanzen. Wenn sie 1000 Bäume in 9 Tagen pflanzen können, wie lange würden 36 Mann für 4400 Bäume benötigen? *Lösung: 33*

30) Ein Mann hinterläßt seinem ältesten Sohn 1 Bezant und 1/7 des Verbleibenden. Von dem Rest gab er dem zweitältesten 2 Bezant und wieder 1/7 des verbleibenden Rests. Diese Teilung setzte er fort, er gab dem nächstjüngeren 1 Bezant mehr als dem vorhergehenden und je 1/7 des verbleibenden Rests. Wieviele Söhne waren es, wenn der letzte Sohn den Rest erhält, aber alle Söhne gleichviel? *Lösung: 6*

Algorismus Ratisbonensis (1450)

31) Item ain fraw hat veigen vnd hat auch kinder vnd sy gibt iglichen kind 12 veigen, so pleibt ir 37 feigen. Nu nympt sy dij veigen widerumb von den kindern vund gibt ander wais hyn vnd gibt iglichen kind 15 feigen, so zw rint (fehlen) ir 44 veigen. Nu frag ich, wije vil sind der feigen vnd der kinder gewesen? *Lösung: (361, 27)*

32) Item: diuidatur 8 in tales 2 partes, ut veniant in diuisione 3. (Teile 8 so in zwei Summanden, daß ihr Quotient 4 ist). *Lösung: (6 2/5, 1 3/5)*

33) Si tunc vixisses quantum vixisti et iterum tantum et dimidium et dimidium dimidij tanti, 100 annos compleuisses. (Wenn du noch solange leben würdest, wie du schon gelebt hast und nocheinmal so viel, und noch halb mal soviel und ein viertelmal soviel, so würdest du 100 Jahre alt werden).
Lösung: 26 2/3

34) Item ein thurnn (Turm) ist 1/4 im ertrich (Erdreich) vnd 10 schuch im wasser vnd 3/5 ym luft. Nu frag ich, wye lang der thurnn seij vnd uil schuch im wasser sey vnd vieuil im luft. *Lösung: 66 2/3*

Bamberger Rechenbuch (1483)

35) Item ist ein Faß, das hat drei Zapfen. Wenn man den ersten zieht, geht es aus in 2 Tagen. Mit dem anderen Zapfen geht es aus in 3 Tagen. Mit dem dritten Zapfen geht es aus in 4 Tagen. Und wenn man sie alle drei zieht, wielange muß es ausrinnen? *Lösung: 12/13*

36) Es lauf ein Has gegen ein Holz und ein Hund lauf ihm hinten nach. Und wenn der Has 12 Sprünge tut, so tut der Hund 15, und der Has ist vor dem Hund 100 Schritt. Nun ist die Frag, wann der Hund den Hasen erläuft, in wieviel Schritten? *Lösung: 33 1/3*

37) Es sind zwei Gesellen, die gehen nach Rom. Einer geht alle Tage 6 Meilen. Der ander geht am ersten Tag 1 Meile, an dem anderen Tag 2, etc. Um alle Tag eine Meile mehr denn vorher. Nun willst du wissen, in wieviel Tagen einer soviel gegangen ist als der andere? *Lösung: 11*

38) Item ein Turm gebaut nach solchen Sitten: 1/4 des Turmes ist im Erdreich, 1/5 im Wasser und 100 Schuh in der Luft. Nun frag ich, wieviel Schuh sind im Wasser und Erdreich und wieviel Schuh sind an dem ganzen Turm?
Lösung: 181 9/11

"Titelblatt von Adam Rieses Rechenbuch (2. Aufl. 1529). Es zeigt einen Wettbewerb zwischen einem Zahlen- und einem Linienrechner"

Rechenbuch des Adam Riese (1526)

39) Item ein kauffmann zeucht (verleiht) hinweg mit gelt / gewinnt ein drittheil seines Hauptguts (Kapital) / vnd 4 fl. mehr / legt an hauptgut vnd gewinn / gewinnt den vierdten theil / bringt zusammen 40. *Lösung: 21*

40) Item einer spricht zum andern / wann ich noch so vil / ein drittheil / vnd ein vierdtheil so viel hett / so wer meines gelts über 100 fl. so viel als jetzt darunder. *Lösung: 55 35/43*

41) Item ein fuhrmann fehrt von Leiptzig ghen Nürenberg in 6 tagen / vnd ein ander fuhrman fehrt desselbigen tags aus von Nürenberg / kompt in 8 tagen ghen Leiptzig / in wie vile tagen komen sie zusamen? *Lösung: 3 3/7*

42) Item 21 Personen / Männer und Frawen / haben vertrunken 81 d / ein Mann sol geben 5 d vnd eine Fraw 3 d. Nu frag ich wie viel jeglicher in sonderheit gewesen sind? *Lösung: (9, 12)*

43) Item einer hat zween silberne Becher / vnd ein oberlied (Deckel) / so dasselbig auff edn ersten gesetzt wirdt / helt er viermal des andern gewicht. Wirdt es aber auff den andern gesetzt / so ist er dreymal schwerer dann der erste / vnd das oberlied wigt 16 loth / wieviel wigt ein jeglicher Becher in sonderheit? *Lösung: (7 3/11, 5 9/11)*

Algebra von Christopher Clavius (1608)

44) Um seinen Sohn zum Rechnen zu ermuntern, zahlt ein Vater seinem Sohn 8 Pfennige für die Lösung einer Aufgabe, wenn sie richtig gelöst wurde und zieht ihm 5 Pfennige ab, wenn sie falsch war. Nach 26 Aufgaben schuldet weder der Vater dem Sohn etwas noch umgekehrt. Wieviele Aufgaben hat der Sohn richtig gelöst? *Lösung: 10*

45) Wenn ich jedem Bettler vor der Tür 7 Pfennige geben würde, würden mir 24 Pfennige verbleiben. 32 Pfennige würden fehlen um jedem 9 Pfennige geben zu können. Wieviele Bettler sind es und wieviel Geld habe ich? *Lösung: (28, 220)*

46) Einem Diener werden 100 Denare und ein Mantel als Jahresgehalt versprochen. Nach 7 Monaten verläßt der Diener das Haus und erhält 20 Denare und den Mantel. Wieviel ist der Mantel wert? *Lösung: 92*

Vollständige Anleitung zur Algebra von Euler (1770)

47) Ein Vater hinterläßt seinen 3 Söhnen ein Vermögen von 1600 Reichstalern. Nach seinem Testament soll der älteste Sohn 200 Rtl. mehr haben als der zweite, der zweite aber 100 Rtl. mehr als der dritte; wieviel bekommt jeder? *Lösung: (700, 500, 400)*

48) Suche ein Zahl von der Beschaffenheit, daß, wenn man sie mit 5 multipliziert, das Produkt so viel unter 40 liegt, wie die Zahl selbst unter 12. *Lösung: 7*

49) Zwey Bäuerinnen haben zusammen 100 Eyer, die erste spricht: wann ich die meinigen zu je 8 überzähle, so bleiben 7 übrig; die andere spricht: wann ich die meinigen zu je 10 überzähle, so bleiben mir auch 7 übrig; wieviel hat jede Eyer gehabt? *Lösung: (63, 37) (23, 77)*

50) Eine Gesellschaft von Männern und Weibern sind in einem Wirtshaus; ein Mann verzehrt 25 Cop., ein Weib aber 16 Cop. und es findet sich, daß die Weiber insgesamt einen Cop. mehr verzehrt haben als die Männer; wieviel Männer und Weiber sind es gewesen? *Lösung: (7 + 16 t, 11 + 25 t) t ganzzahlig*

ANHANG B

ZITATE

Wenn jemand die Mathematiker fragt: Ihr Wunderlichen – von was für Zahlen sprecht Ihr eigentlich – Was werden sie wohl antworten? daß sie von Zahlen reden, die man nur denken kann, unmöglich aber auf irgendeine andere Art handhaben.

Plato

Geometrie ist die Erkenntnis von Gegenständen ewigen Seines.

Plato

Wer die Geometrie begreift, vermag alles auf der Welt zu verstehen.

Galilei

Ceux qui ne sont pas mathematiciens sont portes a considerer les mathematiques comme une science inhumaine.

E. Borel

Gott existiert, wie die Mathematik widerspruchsfrei ist, der Teufel, weil wir es nicht beweisen können.

A. Weil

Die Zahl ist das Wesen aller Dinge.

Pythagoras

Ich hörte mich anklagen, als sei ich ein Feind der Mathematik überhaupt, die doch niemand höher schätzen kann als ich, da sie gerade das leistet, was mir zu bewirken völlig versagt blieb.

Goethe

Die Mathematiker sind eine Art Franzosen: Redet man mit ihnen, so übersetzen sie es in ihre Sprache, und dann ist es etwas ganz anderes.

Goethe

Ein Mathematiker, der nicht irgendwie ein Dichter ist, wird nie ein vollkommener Mathematiker sein.

Weierstraß

Die Wissenschaft der reinen Mathematik in ihrer modernen Entwicklung darf von sich behaupten, die ureigenste Schöpfung des menschlichen Geistes zu sein.

Whitehead

Es gibt jedoch noch einen anderen Grund für die hohe Wertschätzung der Mathematik: sie allein bietet den exakten Naturwissenschaften ein gewisses Maß an Sicherheit, das ohne Mathematik nicht denkbar wäre.

Einstein

Zur Mathematik führt kein Königsweg.

Menachmos

Es gibt nichts Unpopuläreres als die moderne Mathematik, und auch darin steckt ein Stück Symbolik der unendlichen Ferne, der Distanz. Alle großen Werke des Abendlandes von Dante bis Parsifal sind unpopulär.

Spengler

Nicht etwa, daß bei größerer Verbreitung des Einblicks in die Methode der Mathematik notwendigerweise viel mehr Kluges gesagt werden würde als heute, aber es würde sicher viel weniger Unkluges gesagt.

K. Menger

Die ganzen Zahlen hat der liebe Gott gemacht — alles andere ist Menschenwerk.

Kronecker

Die Zahlen sind freie Schöpfungen des menschlichen Geistes, sie dienen als Mittel, um die Verschiedenheit der Dinge leichter und schärfer zu fassen.

Dedekind

Der Begriff der Mathematik ist der Begriff der Wissenschaft überhaupt. Alle Wissenschaften sollen daher Mathematik werden.

Novalis

Wer ein mathematisches Buch nicht mit Andacht ergreift und es wie Gottes Wort liest, versteht es nicht.

Novalis

Die Mathematik ist wie die Gottseligkeit, zu allen Dingen nütze, aber wie diese nicht jedermanns Sache.

J. Kraus

Ich glaube . ., daß es, im strengsten Verstand, für den Menschen nur eine einzige Wissenschaft gibt, und diese ist die reine Mathematik.

Lichtenberg

Die sogenannten Mathematiker von Profession haben sich, auf die Unmündigkeit der übrigen Menschen gestützt, einen Kredit von Tiefsinn erworben, der viel Ähnlichkeit mit dem vom Heiligen hat, den die Theologen für sich haben.

Lichtenberg

Die Mathematik ist eine Art Spielzeug, welches die Natur uns zuwarf zum Troste und zur Unterhaltung in der Finsternis.

d'Alembert

So kann also die Mathematik definiert werden als diejenige Wissenschaft, in der wir niemals das kennen, worüber wir sprechen, und niemals wissen, ob das, was wir sagen, wahr ist.

B. Russell

Stets sind sie eilig, nur zu messen und zu rechnen, halten es für die Hauptsache, und le calcul! le calcul! ist ihr Feldgeschrei. Aber ich sage: ou le calcul commence, l'intelligence des phenomenes cesse: während Einer bloße Zahlen und Zeichen im Kopf hat, kann er nicht dem Kausalzusammenhang auf die Spur kommen.

Schopenhauer

Das Wesen der Mathematik liegt gerade in ihrer Freiheit.

G. Cantor

Die Geometrie ist einzig und ewig, ein Widerschein aus dem Geiste Gottes, daß die Menschen an ihr teilhaben, ist eine Ursache dafür, daß der Mensch Ebenbild Gottes ist.

Kepler

Die Mathematik ist eine gar herrliche Wissenschaft, aber die Mathematiker taugen oft den Henker nicht.

Lichtenberg

Wer die Sicherheit der Mathematik verachtet, stürzt sich in das Chaos der Gedanken.

da Vinci

Befaßt Euch zuerst mit Mathematik, denn die Mathematik ist die Zahl, und die Zahl ist Gott.

J.H. Fabre

Die Natur ist mathematisch, die Meisterwerke der Kunst sind im Einklang mit der Natur; sie drücken die Naturgesetze aus und bedienen sich ihrer. Folglich ist das Kunstwerk mathematisch, und der Wissenschaftler kann das strengste Urteil anlegen und unfehlbare Formeln anwenden.

Le Corbusier

BILDQUELLENNACHWEIS

Aulis Verlag	Seite 96
Bibliographisches Institut AG	Seite 72
Deutsches Museum, Bildstelle B IV	Seiten 16, 17
Niedersächsische Landesbibliothek	Seite 156
Vandenhoeck & Ruprecht	Seiten 12, 15

Im übrigen danken wir den Verlagen für die freundliche Unterstützung.

ANHANG C

LITERATURVERZEICHNIS

[1] Becker O.: Die Grundlagen der Mathematik, Freiburg, München 1964

[2] Becker O.: Zur Geschichte der griechischen Mathematik, Darmstadt 1965

[3] Bell E.T.: Die großen Mathematiker, Düsseldorf, Wien 1967

[4] Bronstein I.N./Semendjajew K.A.: Taschenbuch der Mathematik Zürich, Frankfurt/M. 1967

[5] Courant R./Robbins H.: Was ist Mathematik? Berlin, Göttingen, Heidelberg 1962

[6] Davies P./Hersh R.: The Mathematical Experience Boston, Basel, Stuttgart 1981

[7] Dreszer J.: Mathematik-Handbuch, Zürich, Frankfurt/M., Thun 1975

[8] Engel A.: Elementarmathematik vom algorithmischen Standpunkt Stuttgart 1977

[9] Faddejew D.K./Faddejewa W.N.: Numerische Methoden der linearen Algebra, München, Wien 1973

[10] Ferschl F.: Markovketten, Berlin, Heidelberg, New York 1970

[11] Goldstine H.: The Computer from Pascal to von Neumann Princeton 1980

[12] Hamming R.W.: Numerical Methods for Scientists and Engineers Tokyo, Düsseldorf, Johannesburg, London 1973

[13] Halder H./Heise W.: Einführung in die Kombinatorik München 1976

[14] Hardy G.H./Wright E.M.: Einführung in die Zahlentheorie München 1958

[15] Herrmann D.: Mathematik-Programme in BASIC, Köln 1982

[16] Herrmann D.: Wahrscheinlichkeitsrechnung/Statistik 30 BASIC-Programme, Wiesbaden 1983

[17] Herrmann D.: Numerische Mathematik – 40 BASIC-Programme Wiesbaden im Druck

[18] Hoppe E.: Mathematik und Astronomie im klassischen Altertum Wiesbaden 1966

[19] Jeger M.: Einführung in die Kombinatorik 1, Stuttgart 1973

[20] Karlson P.: Zauber der Zahlen, Frankfurt/M. 1965

[21] Kleine Enzyklopädie Mathematik, Thun, Frankfurt/M. 1977

[22] Menninger K.: Zahlwort und Ziffer, Göttingen 1979

[23] Meschkowski H.: Problemgeschichte der Mathematik I, II
Mannheim, Wien, Zürich 1979, 1981

[24] Meschkowski H.: Meyers Handbuch über die Mathematik
Mannheim, Wien, Zürich 1972

[25] Neugebauer O.: Vorgriechische Mathematik
Berlin, Heidelberg, New York 1969

[26] Otte M.: Mathematiker über Mathematik
Berlin, Heidelberg, New York 1974

[27] Popp W.: Wege des exakten Denkens, München 1981

[28] Scharlau W./Opolka H.: Von Fermat bis Minkowski
Berlin, Heidelberg, New York 1980

[29] Schneider E.: Mathematik ernst und heiter, Wiesbaden 1968

[30] Schubart H. Einführung in die klassische und moderne Zahlentheorie
Braunschweig 1974

[31] Smith D.E.: History of Mathematics II, New York 1958

[32] Steen L.A.: Mathematics Today, New York 1978

[33] Stoer J./Bulirsch R.: Einführung in die Numerische Mathematik I, II
Berlin, Heidelberg, New York 1972, 1973

[34] Struik D.J.: Abriß der Geschichte der Mathematik, Berlin 1976

[35] Stummel F./Hainer K.: Praktische Mathematik, Stuttgart 1971

[36] Toernig W.: Numerische Mathematik für Ingenieure und Physiker, I, II
Berlin, Heidelberg, New York 1979

[37] Tropfke J.: Geschichte der Elementarmathematik I
Berlin, New York 1980

[38] Wussing H.: Vorlesungen zur Geschichte der Mathematik, Berlin 1979

[39] Wussing H./Arnold W.: Biographien bedeutender Mathematiker
Köln 1978

[40] van der Waerden B.L.: Erwachende Wissenschaft I
Basel, Stuttgart 1966

Für den Anwender des Schneider CPC bietet dieses Buch eine Reihe von Programmen aus den Bereichen Finanzmathematik, Unternehmensforschung und Betriebswirtschaft. Sie finden z. B. Programme für Zins-, Rendite-, Renten-, Tilgungs-, Optimierungs-, Abschreibungs-Berechnung.

In Vorb. 2. Q. '85. 216 Seiten.
Kart. DM 44,– / Fr. 44.–
ISBN 3-88322-153-8

Das Buch führt den Leser mit vielen Beispielen aus der Praxis in das Datenbanksystem dBase II ein. Es sind keine Vorkenntnisse von dBase erforderlich. Das Werk ist ähnlich einer (PU) Programmierten Unterweisung aufgebaut und ist daher sehr gut verständlich.

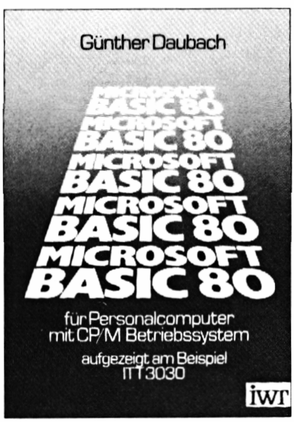
1985. Ca. 300 Seiten.
Kart. DM 48,– / Fr. 48.–
ISBN 3-88322-154-6

Dieses Buch behandelt alle grafischen Fähigkeiten des Schneider CPC und führt in die Grafikprogrammierung ein. Von der elementaren Programmierung bis hin zur komplexen mehrfarbigen Abbildung im hochaufgelösten Grafikmodus werden zahlreiche Beispiele erläutert und entsprechend illustriert.

In Vorb. 2. Q. '85. 288 Seiten.
Kart. DM 48,– / Fr. 48.–
ISBN 3-88322-147-3

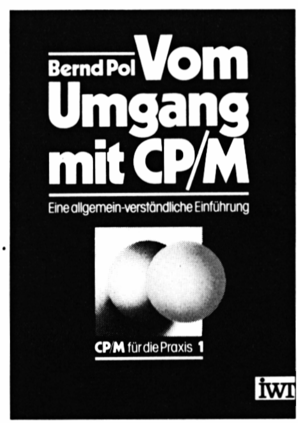
Dieses Buch wendet sich an SuperCalc-Anwender. Anfänger führt es schnell zur sicheren Handhabung, Fortgeschrittene finden ausgereifte Anwendungen. Beispiele mit ausführlicher Beschreibung machen das Buch auch für VisiCalc-Anwender interessant. Die Unterschiede zwischen VisiCalc und SuperCalc sind in einem Kapitel dargestellt.

In Vorb. 2. Q. '85. Ca. 200 Seiten.
Kart. DM 48,– / Fr. 48.–
ISBN 3-88322-155-4



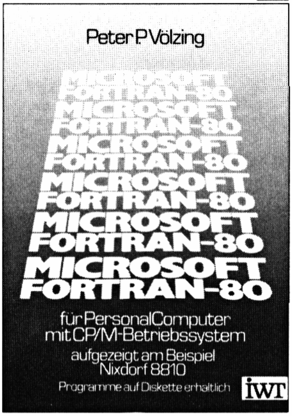
Teil 1 gibt eine Einführung in die Grundlagen; Sprach-Interpreter und Compiler werden erläutert. Teil 2 enthält eine umfassende Beschreibung aller Anweisungen, Befehle und Funktionen, abgerundet mit zahlreichen Beispielen. Teil 3 beschreibt die Anwendung des Compilers und seine Unterschiede zum Interpreter.

1983. 306 Seiten. **Spiralh. DM 56,–/ Fr. 56.–**
ISBN 3-88322-024-8



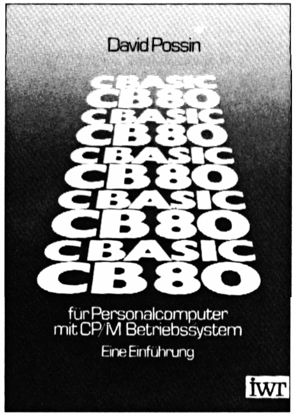
Das Buch ist in drei Teile gegliedert: Teil 1 führt in die Eigenschaften von Computern und CP/M im besonderen ein. Aufbauend auf diesen Kenntnissen werden im 2. Teil die zentralen CP/M-Hilfsprogramme vorgestellt. Der 3. Teil geht – nach einer Einführung in die Funktionsweise des 8080-Prozessors – auf die CP/M-Systembesonderheiten ein.

1982. 386 S. Mit zahlr. prakt. Beispielen.
Geb. DM 48,–/Fr. 48.–
ISBN 3-88322-004-3



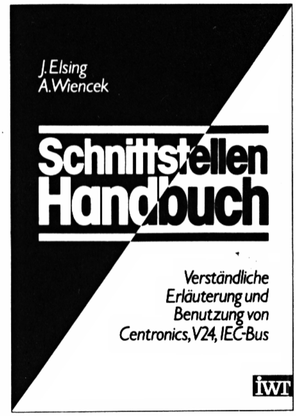
Dieses Buch beschreibt den Leistungsumfang von FORTRAN-80 für Personalcomputer, wie es im Release 3.44 von MicroSoft festgelegt wurde. Alle Sprachelemente sind in detaillierter Form abgehandelt und in Beispielen dargestellt. Das Buch kann in dieser Form als Programmierhandbuch und auch als Nachschlagewerk genutzt werden.

1984. 344 Seiten.
Spiralh. DM 56,-/Fr. 56.-
ISBN 3-88322-079-5



Dieses Buch ermöglicht die schnelle Einarbeitung in die Programmiersprache CBasic. Alle Eigenschaften werden mit Beispielen erklärt. Syntaxdiagramme zeigen grafisch die Einsatzmöglichkeiten. Die Funktionen des Compilers, des Linkers und des Bibliotheksprogrammes LIB werden erläutert.

1983. 190 Seiten.
Spiralh. DM 56,-/Fr. 56.-
ISBN 3-88322-034-5



Dieses Buch behandelt die Problematik der Schnittstellen V 24 und dem IEC-Bus. Die Übertragung zu peripheren Geräten (Drucker, Plotter, Terminal u. a.) ist gut verständlich aufgezeigt. Hardwareaufbau und Beispielanschlüsse machen das Buch zu einem guten Nachschlagewerk.

In Vorb. I.Q.'85. 200 Seiten.
Geb. Ca. DM 48,-/ ca. Fr. 48.-
ISBN 3-88322-094-9

Wer hat nicht bereits verzweifelt versucht, das ›Computerchinesenglisch‹ zu verstehen? Hier hilft das Wörterbuch der Computerei mit seinen über tausend Begriffen. Außerdem sind die wichtigsten Begriffe erklärt. Ein handliches Nachschlagewerk für jeden, der sich mit Computerei beschäftigt.

1983. 2., erw. Aufl. 144 Seiten.
Kart. DM 32,–/Fr. 32.–
ISBN 3-88322-026-4

Die Programmiersprache ›C‹ ist besonders zur Erstellung schneller, maschinennaher Programme geeignet, weist jedoch gegenüber der Assemblerprogrammierung wesentliche Vorteile auf, wie z. B. lokale und globale Variable, Prozeduren, Funktionen usw. Diese Einführung setzt beim Leser keine umfassenden Kenntnisse voraus.

1984. 256 Seiten.
Geb. DM 56,-/Fr. 56.-
ISBN 3-88322-041-8

Dieses Buch führt Sie zur sicheren Handhabung der Tabellenkalkulation. Zahlreiche Anwendungen mit ausführlicher Beschreibung schließen sich an. Unter anderem: Werbeplanung, Reisekosten, Kapazitätsauslastung, Baufinanzierung, Reaktionszeiten.– Auch für die englische Version nutzbar.

In Vorb. Juni 1984. Ca. 250 Seiten. **Spiralh.**
Ca. DM 56,–/ca. Fr. 56.–
ISBN 3-88322-074-4

D. Herrmann, G. Schnellhardt

Schneider CPC Band 1: Mathematik

Dieses Buch enthält 39 mathematische Programme aus den Bereichen: — Mehr-Register-Arithmetik — Zahlentheorie — Kombinatorik — Algebra — Geometrie — Numerische Mathematik.

Die hier neu vorgelegte Langzahl-Arithmetik gestattet die Grundrechenarten für Zahlen bis 255 Stellen. Zahlreiche Anwendungen finden auch die angegebenen kombinatorischen Prozeduren.

Sie finden auch das Rechnen mit Polynomen, Matrizen und komplexen Zahlen: Neben dem komplexen Hornerschema werden insbesondere Algorithmen zur Polynomdivision und Matrizeninversion gegeben.

Anwendungsbezogen sind die Programme der Numerischen Mathematik. Hier werden Verfahren zur Lösung von linearen und nichtlinearen Gleichungen gegeben, zum Eigenwertproblem von Matrizen und zur numerischen Integration und Differentiation.

ISBN 3-88322-152-X

IWT

D. Herrmann G. Schnellhardt

Schneider CPC Mathematik

1

AMSTRAD CPC

MÉMOIRE ÉCRITE
MEMORY ENGRAVED
MEMORIA ESCRITA

https://acpc.me/