

Lösungen der Übungsaufgaben und Verständnisfragen

©Manfred Brill, 2005

Inhaltsverzeichnis

1	Zahlen	2
2	Mengenlehre	16
3	Logik	26
4	Relationen und Abbildungen	34
5	Lineare Gleichungssysteme und der Gauß-Algorithmus	44
6	Zahlentheorie	62
7	Graphentheorie	79
8	Algebraische Strukturen	94
9	Vektoralgebra	107
10	Vektorräume	116
11	Lineare Abbildungen	135
12	Folgen und Reihen	144
13	Differenzialrechnung	157
14	Integralrechnung	171
15	Wahrscheinlichkeitstheorie und Statistik	198

Einleitung

Sie finden in diesem Dokument alle Aufgaben und die zugehörigen Lösungen aus

Manfred Brill: *Mathematik für Informatiker*, 2. Auflage, Hanser Verlag, München.

Darüberhinaus gibt es zu jedem Kapitel Verständnis- und Methodenfragen. Die Verständnisfragen definieren die Lernziele der einzelnen Kapitel und sollen Ihnen helfen Ihr Wissen zu überprüfen. Die Methodenfragen zielen auf das in den Kapiteln vermittelte Methodenwissen. Sie werden häufig Überschneidungen feststellen. Verständnisfragen zielen darauf, beispielsweise einen Algorithmus nachvollziehen zu können; Methodenfragen darauf, dass Sie die Algorithmen anwenden (und programmieren) können.

Kapitel 1

Zahlen

Verständnisfragen

Sachfragen

1. Wie lauten die Peano-Axiome?
2. Wie können die Zahlensysteme der natürlichen, ganzen, rationalen, reellen und komplexen Zahlen charakterisiert werden?
3. Wie ist der Absolutbetrag einer Zahl definiert? Welche Eigenschaften besitzt er?
4. Was versteht man unter dem Logarithmus einer Zahl zu einer Basis a ?
5. Beschreiben Sie den Unterschied zwischen einer reellen und einer komplexen Zahl!
6. Was versteht man unter Polarkoordinaten einer komplexen Zahl?
7. Beschreiben Sie die Multiplikation und Division komplexer Zahlen in der Gauß'schen Zahlenebene!
8. Was ist eine Einheitswurzel?
9. Beschreiben Sie den Begriff der Indextransformation bei Summen und Produkten!
10. Beschreiben Sie die Analogie zwischen Summen und Produkten und Schleifen in einer Programmiersprache!
11. Beschreiben Sie die Zahlensysteme mit den Basen $b = 10$, $b = 2$ und $b = 16$!
12. Warum kann das Konstruktionsverfahren für die Dezimalbruchdarstellung einer reellen Zahl nie zu der Situation führen, dass ab irgendeiner Nachkommastelle alle Koeffizienten d_i Null werden?
13. Wie wird die Darstellung einer gegebenen Dezimalzahl zu einer Basis b berechnet?
14. Was versteht man unter Festkomma-Darstellung? Wie werden die Grundrechenarten realisiert?
15. Was versteht man unter Gleitkomma-Darstellung?
16. Beschreiben Sie ein Modell der Gleitkomma-Arithmetik für Addition und Multiplikation!

Methodenfragen

1. Eine allgemeine Potenz a^x mit Hilfe des natürlichen Logarithmus ausdrücken können.
2. Den Logarithmus zu einer allgemeinen Basis a mit Hilfe des natürlichen Logarithmus ausdrücken können.
3. Die Rechengesetze für die Logarithmen anwenden können.
4. Die Arithmetik mit komplexen Zahlen durchführen können.
5. Zwischen der kartesischen Darstellung und der Polarkoordinatendarstellung komplexer Zahlen umrechnen können.
6. Eigenschaften komplexer Zahlen in der Gauß'schen Zahlenebene visualisieren können.
7. Allgemeine Summen und Produkte berechnen können.
8. Indextransformationen ausführen können.
9. Die Fakultät berechnen können.
10. Die Darstellung einer gegebenen ganzen oder rationalen Dezimalzahl in einem Stellenwertsystem aufstellen können.
11. Zwischen zwei verschiedenen Zahlensystemen umrechnen können; insbesondere zwischen Dezimalsystem, Dualsystem und Hexadezimalsystem.
12. Rationale und reelle Zahlen in Fest- und Gleitkommazahlen umwandeln können.
13. Die Kenngrößen eines Gleitkommasystems beschreiben können.
14. Gleitkommaarithmetik durchführen können.
15. Matrizen addieren und transponieren können.
16. Matrizen multiplizieren können, mindestens unter Zuhilfenahme des Falk'schen Schemas.

Übungsaufgaben

1. Weisen Sie nach, dass $\log_{10}(2)$ irrational ist!

Lösung:

Angenommen, es gibt zwei natürliche Zahlen m, n mit

$$\log_{10}(2) = \frac{m}{n},$$

dann gilt nach Definition des Logarithmus

$$2^n = 10^m.$$

Das ist aber ein offensichtlicher Widerspruch, denn die rechte Seite dieser Gleichung ist durch 5 teilbar, die linke Seite aber nicht!

2. Weisen Sie nach, dass die Näherung $\text{ld } x \approx \ln x + \log_{10} x$ einen Fehler von weniger als 1% aufweist!

Lösung:

Es ist $\ln(2)^{-1} \approx 1,442\,695\,040\,888\,963$, deshalb gilt $\text{ld}(x) \approx 1,442\,695\,040\,888\,963 \cdot \ln(x)$. Mit dem gleichen Argument gilt $\log_{10}(x) \approx 1,434\,294\,481\,903\,251 \cdot \ln(x)$. Der relative Fehler der Darstellung ist dann ungefähr gegeben durch

$$\frac{1,442\,695 - 1,434\,294}{1,442\,695} \approx 0,582\%.$$

3. Implementieren Sie in der Programmiersprache Ihrer Wahl die Näherung aus Aufgabe 2 und $\text{ld}(x) = \frac{\ln(x)}{\ln(2)}$. Vergleichen Sie die damit erzielten Ergebnisse!

Lösung:

Hier eine C++-Version:

```
const double ln2 = log(2.0);
double logapprox(double x)
{
    return log(x) + log10(x);
}
double ld(double x)
{
    return log(x)/ln2;
}
```

Hier eine Java-Version:

```
static private double ln2 = Math.log(2.0);

static private double approx(double x)
{
    return Math.log(x) + Math.log10(x);
}

static private double ld(double x)
{
    return Math.log(x)/ln2;
}
```

Und hier ein Ergebnis (Pentium III, g++ unter Cygwin):

x-Wert	$\ln(x) + \log_{10}(x)$	$\ln(x) / \ln(2)$
0.5	-0.994177	-1
0.6	-0.732674	-0.736966
0.7	-0.511577	-0.514573
0.8	-0.320054	-0.321928
0.9	-0.151118	-0.152003
1	-1.59239e-16	-1.60171e-16
1.1	0.136703	0.137504
1.2	0.261503	0.263034
1.3	0.376308	0.378512
1.4	0.4826	0.485427
1.5	0.581556	0.584963
1.6	0.674124	0.678072
1.7	0.761077	0.765535
1.8	0.843059	0.847997
1.9	0.920607	0.925999
2	0.994177	1
2.1	1.06416	1.07039
2.2	1.13088	1.1375

2.3	1.19464	1.20163
2.4	1.25568	1.26303
2.5	1.31423	1.32193

4. Stellen Sie die folgenden Zahlen und die dazu konjugiert komplexen Zahlen in der Gauß'schen Zahlenebene dar und berechnen Sie $|z|$ und $|\bar{z}|$: $z_1 = 3 + 4i$, $z_2 = 1 - 2i$, $z_3 = -2i$, z_4 mit $|z_4| = 5$, $\operatorname{Re}(z_4) = 3$.

Lösung:

Es ist immer $|z| = |\bar{z}|$; $|z_1| = 5$, $|z_2| = \sqrt{5}$, $|z_3| = 2$.

z_4 liegt auf dem Kreis um den Ursprung mit Radius 5, und auf der Linie $\operatorname{Re}(z) = 3$, Dann sind die beiden Schnittpunkte gegeben durch den Imaginärteil $3^2 + y^2 = 25$; damit ergeben sich die beiden Lösungen $\operatorname{Im}(z_4) = \pm 4$. Eine Lösung ist also z_1 , die andere gegeben durch $z_4 = 3 - 4i$.

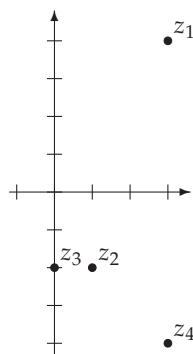


Abbildung 1.1: Lösung zu Aufgabe 5

5. Berechnen Sie die folgenden Summen, Produkte und Quotienten: $z_1 = (1 + 2i) + (3 + 4i)$, $z_2 = (8 - 5i) + (3 + 9i)$, $z_3 = (3 + 7i)(5 + 6i)$, $z_4 = (7 - i)^2$, $z_5 = \frac{1}{2+i}$, $z_6 = \frac{1}{3-5i}$, $z_7 = (5 + 3i)^{-2}$.

Lösung:

$$z_1 = 4 + 6i, z_2 = 11 + 4i, z_3 = -27 + 53i, z_4 = 48 - 14i, z_5 = \frac{2}{5} - \frac{1}{5}i, z_6 = \frac{3}{34} + \frac{5}{34}i, z_7 = \frac{4}{289} - \frac{15}{578}i.$$

6. Bestimmen Sie die Polarkoordinatendarstellung von $(1 + i)^{\frac{1}{2}}$, $(-8 + i8\sqrt{3})^{\frac{1}{4}}$ und $(i)^{\frac{1}{3}}$.

Lösung:

Die beiden Wurzeln für $(1 + i)^{\frac{1}{2}}$:

$$z_1 = \sqrt[4]{2}(\cos 22,5^\circ + i \sin 22,5^\circ)$$

$$z_2 = \sqrt[4]{2}(\cos 202,5^\circ + i \sin 202,5^\circ)$$

Die vier Wurzeln für $(-8 + i8\sqrt{3})^{\frac{1}{4}}$:

$$z_1 = 2(\cos 30^\circ + i \sin 30^\circ)$$

$$z_2 = 2(\cos 120^\circ + i \sin 120^\circ)$$

$$z_3 = 2(\cos 210^\circ + i \sin 210^\circ)$$

$$z_4 = 2(\cos 300^\circ + i \sin 300^\circ)$$

Die drei Wurzeln für $i^{\frac{1}{3}}$:

$$z_1 = \cos 30^\circ + i \sin 30^\circ$$

$$z_2 = \cos 150^\circ + i \sin 150^\circ$$

$$z_3 = \cos 270^\circ + i \sin 270^\circ$$

7. Berechnen Sie die 5-ten Einheitswurzeln und stellen Sie diese in der komplexen Zahlenebene dar!

Lösung:

$$z_i = \cos\left((i-1)\frac{2\pi}{5}\right) + i \sin\left((i-1)\frac{2\pi}{5}\right), 1 \leq i \leq 5.$$

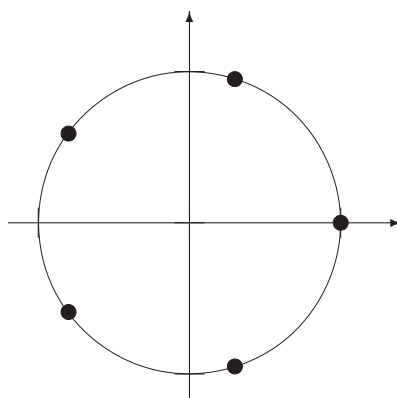


Abbildung 1.2: Lösung zu Aufgabe 7

8. Berechnen Sie mit Hilfe eines Programms die Fakultäten $6!$, $9!$, $10!$ und $12!$ und vergleichen Sie die Näherungen mit der Stirling'schen Formel! Vergleichen Sie insbesondere den relativen Fehler mit $\frac{1}{12n}$.

Lösung:

Die Fakultät als rekursive Funktion:

```
double fakultaet(int n)
{
    if (n==1)
        return 1;
    else
        return n * fakultaet(n-1);
}

double stirling(int n)
{
    return sqrt(2 * M_PI * n) * pow(n/M_E, n);
}

double relativ(int n)
{
    double fak = fakultaet(n);
    return (fak - stirling(n))/fak;
}
```

Und hier die Fakultäten, die Näherung durch die Stirling'sche Formel und der Vergleich des relativen Fehlers mit $\frac{1}{12n}$:

```

n! fuer n=6 ist 720
Die Näherung durch die Stirling'sche Formel 710.078
Der relative Fehler: 0.0137803
1/(12n) im Vergleich dazu: 0.0138889

n! fuer n=9 ist 362880
Die Näherung durch die Stirling'sche Formel 359537
Der relative Fehler: 0.00921276
1/(12n) im Vergleich dazu: 0.00925926

n! fuer n=10 ist 3.6288e+06
Die Näherung durch die Stirling'sche Formel 3.5987e+06
Der relative Fehler: 0.00829596
1/(12n) im Vergleich dazu: 0.00833333

n! fuer n=12 ist 4.79002e+08
Die Näherung durch die Stirling'sche Formel 4.75687e+08
Der relative Fehler: 0.00691879
1/(12n) im Vergleich dazu: 0.00694444

```

9. Berechnen Sie für die Zahlen $x_1 = 5, x_2 = 2, x_3 = 1, x_4 = 2, y_1 = 1, y_2 = 4, y_3 = 3$ und $y_4 = 1$

$$\sum_{i=1}^4 x_i, \prod_{i=1}^4 x_i, \sum_{i=1}^4 x_i y_i \text{ und } \prod_{i=1}^4 x_i y_i.$$

Lösung:

$$\sum_{i=1}^4 x_i = 10, \prod_{i=1}^4 x_i = 20, \sum_{i=1}^4 x_i y_i = 18 \text{ und } \prod_{i=1}^4 x_i y_i = 240.$$

10. Berechnen Sie

$$\sum_{i=1}^{10} \frac{1}{i}, \sum_{i=1}^{10} (i+3) \text{ und } \prod_{i=1}^{10} (6i-2).$$

Lösung:

$$\sum_{i=1}^{10} \frac{1}{i} = \frac{7381}{2520} \approx 2,928\,968, \sum_{i=1}^{10} (i+3) = 85, \prod_{i=1}^{10} (6i-2) = 74\,385\,581\,670\,400.$$

11. Wenden Sie das Divisionsverfahren auf die natürliche Zahl $n = 674$ für die Basen $b = 5$ und $b = 2$ an!

Lösung:

Für $b = 5$ ist $k = 5$, denn es gilt $5^4 = 625 \leq 674 < 5^5$. Das Divisionsverfahren ergibt die Darstellung $(10144)_5$:

$$674 = 1 \cdot 5^4 + 49$$

$$49 = 0 \cdot 5^3 + 49$$

$$49 = 1 \cdot 5^2 + 24$$

$$24 = 4 \cdot 5^1 + 4$$

$$4 = 4 \cdot 5^0.$$

Für $b = 2$ ist $k = 10$, das Divisionsverfahren

$$674 = 1 \cdot 2^9 + 162$$

$$162 = 0 \cdot 2^8 + 162$$

$$162 = 1 \cdot 2^7 + 34$$

$$34 = 0 \cdot 2^6 + 34$$

$$34 = 1 \cdot 2^5 + 2$$

$$2 = 0 \cdot 2^4 + 2$$

$$2 = 0 \cdot 2^3 + 2$$

$$2 = 0 \cdot 2^2 + 2$$

$$2 = 1 \cdot 2^1 + 0$$

$$0 = 0 \cdot 2^0.$$

führt zum Ergebnis $674 = (1010100010)_2$.

12. Wandeln Sie die Zahl $(745)_8$ in das System zur Basis $b = 3$ um!

Lösung:

Die Zahl $(745)_8$ ist als Dezimalzahl gegeben durch $7 \cdot 64 + 4 \cdot 8 + 5 = 485$. Zur Basis 3 ist die Darstellung dann $(122222)_3$ wegen

$$485 = 1 \cdot 3^5 + 242$$

$$242 = 2 \cdot 3^4 + 80$$

$$80 = 2 \cdot 3^3 + 26$$

$$26 = 2 \cdot 3^2 + 8$$

$$8 = 2 \cdot 3^1 + 2$$

$$2 = 2 \cdot 3^0.$$

13. Formal kann mit Darstellungen zu einer Basis b gerechnet werden, wie wir das vom Dezimalsystem gewohnt sind. Prüfen Sie dies an Hand der Beispiele $(543)_6 + (242)_6 = (1225)_6$, $(213)_6 - (132)_6 = (41)_6$ und $(153)_6 \cdot (23)_6 = (4443)_6$ nach!

Lösung:

Mit Hilfe des Stellenwertsystems ergibt sich beispielsweise für die Addition

$$\begin{aligned} (543)_6 + (242)_6 &= 5 \cdot 6^2 + 4 \cdot 6 + 3 \cdot 1 + 2 \cdot 6^2 + 4 \cdot 6 + 2 \cdot 1 \\ &= 1 \cdot 6^3 + 2 \cdot 6^2 + 2 \cdot 6^1 + 4 \cdot 1 \\ &= (1225)_6 \end{aligned}$$

Es ist $8 = (12)_6$ und deshalb

$$\begin{array}{r} (5 \ 4 \ 3)_6 \\ + (2 \ 4 \ 2)_6 \\ \hline (1 \ 2 \ 2 \ 5)_6 \end{array}$$

wenn Sie alle Berechnungen und Überträge wie in der Schule gelernt zur Basis 6 durchführen.

Für die Subtraktion erhalten Sie

$$\begin{array}{r} (2 \ 1 \ 3)_6 \\ - (1 \ 3 \ 2)_6 \\ \hline (0 \ 4 \ 1)_6 \end{array}$$

und für die Multiplikation

$$\begin{array}{r} (153)_6 \cdot (23)_6 \\ \hline (350)_6 \\ (543)_6 \\ \hline (4443)_6 \end{array}$$

14. Berechnen Sie die Entwicklung zur Basis $b = 7$ von $\frac{1}{5}$ und von $\frac{2}{5}$ zur Basis 2!

Lösung:

$\frac{1}{5} = 0,2 = (0, \overline{1254})_7$, denn es ist $1 = 0 \cdot 5 + 1$ und

$$7 = 1 \cdot 5 + 2$$

$$14 = 2 \cdot 5 + 4$$

$$28 = 5 \cdot 5 + 3$$

$$21 = 4 \cdot 5 + 1.$$

Es ist $\frac{2}{5} = 0,2 = (0, \overline{0110})_2$ wegen $2 = 0 \cdot 5 + 2$ und

$$4 = 0 \cdot 5 + 4$$

$$8 = 1 \cdot 5 + 3$$

$$6 = 1 \cdot 5 + 1$$

15. Die Darstellung im Zweierkomplement wird in den Mikroprozessoren von Intel verwendet, mit den Längen $N = 16, 32$ und 32 werden damit die Datentypen *short integer*, *word integer* und *long* realisiert. Darüber hinaus gibt es die Datentypen *unsigned short integer*, *unsigned word integer* und *unsigned long*. Geben Sie die minimal und maximal darstellbaren ganzen Zahlen für diese 6 Datentypen an!

Lösung:

Die Lösung finden Sie in Tabelle 1.1.

Tabelle 1.1: Lösung von Aufgabe 15

Datentyp	i_{min}	i_{max}
unsigned short int	0	65 535
short int	-32 768	32 767
unsigned int	0	4 294 967 295
int	-2 147 483 648	2 147 483 647
unsigned long int	0	4 294 967 295
long int	-2 147 483 648	2 147 483 647

16. Wandeln Sie die Dezimalzahlen $1, 0, -0,5, 6, 625$ und $3\,456$ in das IEC single-Format um!

Lösung:

Die Zahl $1, 0$ ist gegeben durch $1_2 \cdot 2^0$. Also ist die Mantisse gegeben durch $00\,000\,000\,000\,000\,000\,000\,000$, die Charakteristik ist gegeben durch $127 + 0 = 127$, also durch $01\,111\,111$.

$0,5$ ist gegeben durch $(0,1)_2 \cdot 2^0$. Durch Normalisieren erhält man den Dualbruch $(1,0)_2 \cdot 2^{-1}$. Die Mantisse ist also $00\,000\,000\,000\,000\,000\,000\,000$, die Charakteristik $127 - 1 = 126 = (01\,111\,110)_2$.

$6,625$ entspricht in Dualdarstellung $(110,101)_2 \cdot 2^0 = (1,10101)_2 \cdot 2^2$. Dann ist die Mantisse gegeben durch $m = 01\,010\,100\,000\,000\,000\,000\,000$; die Charakteristik ist $127 + 2 = 129 = (10\,000\,001)_2$.

3456 ist als Dualzahl gegeben durch $(110\ 110\ 000\ 000)_2 \cdot 2^0 = (1,101\ 100\ 000\ 00)_2 \cdot 2^{11}$. Die Mantisse ist dann gegeben durch $m = (10\ 110\ 000\ 000\ 000\ 000\ 000\ 000)_2$, die Charakteristik durch $c = 127 + 11 = 138 = (10\ 001\ 010)_2$.

17. Welche Dezimalzahlen werden im IEC single-Format durch die Charakteristik $c_1 = (10\ 000\ 000)_2$ und die Mantisse $m_1 = (10\ 000\ 000\ 000\ 000\ 000\ 000\ 000)_2$ beziehungsweise $c_2 = (10\ 100\ 000)_2$, $m_2 = (00\ 100\ 000\ 000\ 000\ 000\ 000\ 000)_2$ dargestellt?

Lösung:

Die erste Bitfolge stellt $x_1 = 3$, denn die Charakteristik entspricht der Zahl 128, dadurch ergibt sich der Exponent 1. Die Mantisse entspricht $(1,1)_2 = 1,5$. Insgesamt ergibt sich $x_1 = 1,5 \cdot 2^1 = 3$.

Die zweite Zahl x_2 hat den Exponenten 33, denn es ist $(10\ 100\ 000)_2 = 160$. Die Mantisse entspricht $(1,001)_2 = 1,125$. Dann ist $x_2 = 1,125 \cdot 2^{33}$.

18. Wie sieht die größte und kleinste darstellbare Gleitkommazahl in einem normalisierten System mit den Parametern b, p, e_{min} und e_{max} aus? Welche Zahlen ergeben sich für die IEC-Grundformate?

Lösung:

Die kleinste Zahl ist

$$-1, \underbrace{00 \dots 00}_{(p-1)\text{-mal}} 1 \cdot b^{e_{min}},$$

die größte ist

$$1, \underbrace{(b-1)(b-1) \dots (b-1)(b-1)}_{p\text{-mal}} \cdot b^{e_{max}}.$$

Die kleinste Zahl für IEC single ist $-1, \underbrace{00 \dots 00}_{22\text{-mal}} 1 \cdot 2^{-125}$, die größte $1, \underbrace{11 \dots 11}_{23\text{-mal}} \cdot 2^{128}$.

IEC double berechnet man analog.

19. Weisen Sie nach, dass der Abstand zwischen 1.0 und $\frac{1}{b}$ durch $\frac{\epsilon_M}{b}$ gegeben ist!

Lösung:

$\frac{1}{b}$ ist gegeben durch $0,10 \dots 0 \cdot b^0$. Die Zahlen zwischen $\frac{1}{b}$ und 1 haben den Abstand $0,0 \dots 01 \cdot b^0 = b^{-p}$. Dies entspricht

$$\frac{\epsilon_M}{b} = \frac{b^{1-p}}{b}.$$

20. Definieren Sie für Festkommazahlen mit Skalierungsfaktor s und Verschiebung V eine Multiplikation, sodass das Produkt $x_1 \odot x_2$ der beiden ganzen Zahlen x_1, x_2 dem Produkt der durch sie dargestellten reellen Zahlen entspricht!

Lösung:

$f(x_1 \odot x_2) = s(x_1 \odot x_2) + V = f(x_1)f(x_2) = (sx_1 + V)(sx_2 + V)$. Durch Auflösen erhält man die Darstellung $x_1 \odot x_2 = sx_1x_2 + V(x_1 + x_2) + \frac{1}{s}V(V-1)$.

21. Berechnen Sie Summe und Differenz für $x_1 = 100, x_2 = 99,7$ in einem Gleitkommasystem mit $b = 10$ und $p = 2$.

Lösung:

Durch Angleichen der Exponenten ist die Summe gegeben durch $0,1997 \cdot 10^3 = 0,2 \cdot 10^3 = 200$.

Die Differenz ist gegeben durch $0,0003 \cdot 10^3 = 0,3 \cdot 10^0 = 0,3$.

22. Berechnen Sie im Gleitkommazahlensystem mit $b = 10, p = 4$ die Summe $0,4462 \cdot 10^{-4} + 0,2413 \cdot 10^{-3} + 0,1234 \cdot 10^0$ von links nach rechts und von rechts nach links. Notieren Sie die jeweils bei jeder Addition auftretenden Rundungsfehler!

Lösung:

Von links nach rechts ergibt sich als erste Summe $0,2859 \cdot 10^{-3}$. Durch Angleichen der Exponenten ist dann die Summe gegeben durch $0,12368592 \cdot 10^0$, durch Runden ergibt sich dann das Endergebnis $0,1237 \cdot 10^0$.

Von rechts nach links: Die erste Summe berechnet sich als $0,1236413 \cdot 10^0$, durch Runden ergibt sich $0,1236 \cdot 10^0$. Die Addition von $0,4462 \cdot 10^{-4}$ ergibt nach Runden $0,1236 \cdot 10^0$.

Summen sollten immer der Größe nach geordnet berechnet werden; beginnend mit den kleinsten Summanden!

23. Das arithmetische Mittel zweier Zahlen ist definiert als $\frac{x_1+x_2}{2}$. Berechnen Sie mit Gleitkommazahlen mit $b = 10, p = 2$ für $x_1 = 0.99, x_2 = 0.98$ das arithmetische Mittel. Betrachten Sie den absoluten und den relativen Fehler!

Alternativ ergibt sich das arithmetische Mittel durch $x_2 + (x_1 - x_2)/2$. Berechnen Sie das arithmetische Mittel durch diese Formel und vergleichen Sie die Ergebnisse!

Lösung:

Für die Summe $x_1 + x_2$ ergibt sich nach Runden das Ergebnis $0,2 \cdot 10^1$. Multiplikation mit $0,5 \cdot 10^0$ ergibt dann das Endergebnis $0,1 \cdot 10^1 = 1$.

Für die Alternative wird zuerst die Differenz $x_1 - x_2$, dafür ergibt sich $0,01 \cdot 10^0$. Multiplikation dieser Zahl mit $0,5 \cdot 10^0$ ergibt $0,01 \cdot 10^0$; als Näherung für das arithmetische Mittel insgesamt dann $0,99 \cdot 10^0$.

24. Gegeben sind die beiden Rekursionen $f_0 = 1 - \frac{1}{e}, f_{k+1} = 1 - (k+1)f_k$ und $g_k = \frac{1-g_{k+1}}{k+1}$ mit einem Startwert $G_N, N > k$. Die Rekursion für g_k ergibt sich durch Auflösen der Rekursion für f nach f_k . Berechnen Sie f_{30} mit dem Computer und vergleichen Sie das Ergebnis mit g_{30} . Dabei verwenden Sie für g_{50} einen beliebigen Startwert zwischen -10^{10} und 10^{10} . Interpretieren Sie die Ergebnisse!

Lösung:

Bei der „Rückwärtsiteration“ wird der Fehler, der durch eine ungenaue Zahldarstellung entsteht mit $\frac{1}{(50-30)!}$, bei der Vorwärtsiteration mit $30!$ multipliziert.

Ein Quelltext in C++:

```
// -----
// Implementierung der Integralrekursionen für Aufgabe 24, Kapitel Zahlen
// -----
#include <cmath>
#include "intrek.h"

double forwardIteration(unsigned int n)
{
    unsigned int counter;
    double iterationValue;

    iterationValue = (exp(1.0)-1.0)/exp(1.0);

    for (counter = 1; counter <= n; counter++)
        iterationValue = 1.0 - counter*iterationValue;
    return iterationValue;
}
```

```
double backwardIteration(unsigned int start, unsigned n, double istart)
{
    unsigned int counter;
    double iterationValue;

    iterationValue = istart;

    for (counter = start; counter > n; counter--)
        iterationValue = (1.0-iterationValue)/(double)(counter);
    return iterationValue;
}

// -----
//   Integralrekursion aus Aufgabe 24, Kapitel Zahlen
//   Hauptprogramm
// -----
#include <cmath>
#include <iostream.h>

#include "intrek.h"

int main(void)
{
    char vor='v', zwischen='n';
    int i, N, startN;
    double start=0.0;

    // Anzahl der Nachkommastellen in der Ausgabe hochsetzen!
    cout.precision(12);

    cout << "-----" << endl;
    cout << " Integralrekursion" << endl;
    cout << "-----" << endl;

    cout << "Vorwärts- oder Rückwärtsrekursion?" << endl;
    cout << "v = Vorwärts" << endl;
    cin >> vor;
    if (vor == 'v' || vor == 'V') {
        cout << "Welches n soll denn berechnet werden?" << endl;
        cin >> N;
        cout << "Sollen Zwischenergebnisse ausgegeben werden?" << endl;
        cout << "j/n" << endl;
        cin >> zwischen;
        cout << "Das Ergebnis der Vorwärtsrekursion" << endl;
        cout << "Der Startwert: " << (exp(1)-1.0)/exp(1.0) << endl;
        cout << endl;
        if (zwischen == 'j' || zwischen == 'J') {
            for (i=1; i<=N; i++)
                cout << "k= " << i << ": " << forwardIteration(i) << endl;
        }
        else
            cout << "I_" << N << forwardIteration(N) << endl;
    }
    else {
        cout << "Welches n soll denn berechnet werden?" << endl;
        cin >> N;
        cout << "Von welchem Startindex soll berechnet werden?" << endl;
    }
}
```

```

cin >> startN;
cout << "Welcher Startwert soll verwendet werden?" << endl;
cin >> start;
cout << "Sollen Zwischenergebnisse ausgegeben werden?" << endl;
cout << "j/n" << endl;
cin >> zwischen;
cout << "Das Ergebnis der Rückwärtsrekursion" << endl;
cout << "Der Startwert: " << start << endl;
cout << endl;
if (zwischen == 'j' || zwischen == 'J') {
    for (i=startN; i>= N; i--)
        cout << "k= " << i << ": " <<
            backwardIteration(startN, i, start) << endl;
}
else
    cout << "I_" << N << i" :" <<
        backwardIteration(startN, N, start) << endl;
}
return 0;
}

```

Hier die Ausgabe für die Vorwärts-Iteration:

```

Das Ergebnis der Vorwärts-Iteration
Der Startwert: 0.632120558829

```

```

k= 1: 0.367879441171
k= 2: 0.264241117657
k= 3: 0.207276647029
k= 4: 0.170893411885
k= 5: 0.145532940573
k= 6: 0.126802356562
k= 7: 0.112383504069
k= 8: 0.100931967445
k= 9: 0.0916122929942
k= 10: 0.0838770700583
k= 11: 0.0773522293588
k= 12: 0.0717732476946
k= 13: 0.0669477799697
k= 14: 0.0627310804239
k= 15: 0.0590337936419
k= 16: 0.0554593017296
k= 17: 0.051918705973
k= 18: -0.0294536707515
k= 19: 1.55961974428
k= 20: -30.1923948856
k= 21: 635.040292597
k= 22: -13969.8864371
k= 23: 321308.388054
k= 24: -7711400.3133
k= 25: 192785008.833
k= 26: -5012410228.65
k= 27: 135335076174
k= 28: -3.78938213288e+12
k= 29: 1.09892081854e+14
k= 30: -3.29676245561e+15

```

Im Gegensatz dazu die Ergebnisse der Rückwärts-Iteration, beginnend bei einem beliebigen Startwert für I_{50} und einem Ergebnis für I_{30} :

Das Ergebnis der Rückwärtsrekursion
 Der Startwert für n=50: 1000000

```

k= 50: 1000000
k= 49: -19999.98
k= 48: 408.183265306
k= 47: -8.48298469388
k= 46: 0.201765631785
k= 45: 0.0173529210482
k= 44: 0.0218366017545
k= 43: 0.0222309863238
k= 42: 0.0227388142715
k= 41: 0.0232681234697
k= 40: 0.0238227286959
k= 39: 0.0244044317826
k= 38: 0.0250152709799
k= 37: 0.0256574928689
k= 36: 0.0263335812738
k= 35: 0.0270462894091
k= 34: 0.0277986774455
k= 33: 0.0285941565457
k= 32: 0.0294365407107
k= 31: 0.0303301081028
k= 30: 0.0312796739322
  
```

25. Berechnen Sie die Produkte $(AB)C$ und $A(BC)$ für

$$A = \begin{pmatrix} 2 & 1 & -1 \\ 0 & 2 & 1 \end{pmatrix}, B = \begin{pmatrix} 4 & -2 & 0 \\ -3 & -1 & -1 \\ 5 & 0 & 2 \end{pmatrix}, C = \begin{pmatrix} -1 & 2 \\ 4 & 0 \\ 3 & 1 \end{pmatrix}.$$

Lösung:

$(AB)C = A(BC)$ und

$$ABC = \begin{pmatrix} -29 & -3 \\ -7 & -2 \end{pmatrix}$$

26. Berechnen Sie $A \cdot B$ mit dem Algorithmus von Strassen:

$$A = \begin{pmatrix} -1 & 2 & 1 & 5 \\ 0 & -3 & 4 & 2 \\ 1 & 5 & 6 & 1 \end{pmatrix}, B = \begin{pmatrix} 2 & 1 & 4 \\ 3 & 5 & 2 \\ 7 & -1 & 5 \\ 0 & 3 & -3 \end{pmatrix}$$

Lösung:

Um den Algorithmus von Strassen anzuwenden, müssen beide Matrizen durch Nullen zu 4×4 Matrizen aufgefüllt werden.

Dann ergeben sich die folgenden Zwischenmatrizen:

$$\begin{aligned}
 W_1 &= \begin{pmatrix} 35 & 20 \\ 0 & -15 \end{pmatrix}, W_2 = \begin{pmatrix} 32 & 37 \\ 0 & 0 \end{pmatrix}, \\
 W_3 &= \begin{pmatrix} 11 & 0 \\ -15 & 0 \end{pmatrix}, W_4 = \begin{pmatrix} 27 & -14 \\ 0 & 0 \end{pmatrix}, \\
 W_5 &= \begin{pmatrix} -21 & 0 \\ 23 & 0 \end{pmatrix}, W_6 = \begin{pmatrix} 27 & 17 \\ 15 & 15 \end{pmatrix}, W_7 = \begin{pmatrix} -72 & 17 \\ 42 & 2 \end{pmatrix}.
 \end{aligned}$$

Als Endergebnis ergibt sich wie im Buch das Matrixprodukt

$$A \cdot B = \begin{pmatrix} 11 & 23 & -10 \\ 19 & -13 & 8 \\ 59 & 23 & 41 \end{pmatrix},$$

wenn man die überflüssigen Nullen wieder entfernt.

27. Für reelle Zahlen gilt die Kürzungsregel: Aus $ab = ac$ und $a \neq 0$ folgt $b = c$, und aus $ad = 0$ folgt $a = 0 \vee d = 0$. Gelten diese Regeln für die Matrix-Arithmetik?

Lösung:

Für singuläre Matrizen $A \neq 0$ können die Regeln verletzt werden!

Kapitel 2

Mengenlehre

Verständnisfragen

Sachfragen

1. Wann sind zwei Mengen gleich?
2. Was ist eine Potenzmenge?
3. Was ist der Durchschnitt und die Vereinigung zweier Mengen?
4. Was ist die Differenz und das Komplement zweier Mengen?
5. Wie lautet die Summen- und die Produktregel der Kombinatorik?
6. Was ist eine Permutation?
7. Was ist eine Kombination?
8. Was ist das Pascal'sche Dreieck?
9. Wie lautet die Vandermond'sche Gleichung und wie lautet ihre Interpretation in der Kombinatorik?
10. Beschreiben Sie das Inklusions-Exklusions-Prinzip!
11. Motivieren Sie die Formel für das Inklusions-Exklusions-Prinzip!
12. Was ist ein Derangement?

Methodenfragen

1. Die Potenzmenge einer gegebenen Menge aufstellen können.
2. Eine Mengengleichheit nachweisen können.
3. Die verschiedenen Mengenoperationen durchführen können.
4. Die Rechenregeln für die Mengenoperationen anwenden können.
5. Mengenbeziehungen und Mengenaussagen durch Venn-Diagramme beschreiben können.

6. Mit Hilfe der Charakteristik Mengen auf dem Computer darstellen können und damit die Mengenoperationen durchführen können.
7. Die Summen- und Produktregel anwenden können.
8. Permutation und Kombinationen mit und ohne Wiederholung berechnen können.
9. Binomialkoeffizienten berechnen und anwenden können.
10. Modellprobleme der Kombinatorik beschreiben und identifizieren können.
11. Das Inklusions/Exklusions-Prinzip anwenden können.
12. Derangements berechnen können.

Übungsaufgaben

1. Zählen Sie die Elemente der folgenden Mengen auf: $M_1 = \{x \mid x \text{ ist eine reelle Zahl mit } x^2 = 1\}$, $M_2 = \{x \mid x \text{ ist eine natürliche Zahl kleiner als } 12\}$, $M_3 = \{x \mid x \text{ ist das Quadrat einer ganzen Zahl und kleiner als } 100\}$.

Lösung:

$$M_1 = \{1, -1\}, M_2 = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11\}, M_3 = \{1, 4, 9, 16, 25, 36, 49, 64, 81\}.$$

2. Beschreiben Sie die folgenden Mengen mit Hilfe von Eigenschaften: $M_1 = \{0, 3, 6, 9, 12\}$, $M_2 = \{-3, -2, -1, 0, 1, 2, 3\}$, $M_3 = \{1, 3, 5, 7, 9\}$, $M_4 = \{x \mid 2, 4, 6, 8, 10\}$.

Lösung:

$$M_1 = \{x \mid x = k \cdot 3, k = 0, 1, 2, 3, 4\}, M_2 = \{x \mid x \in \mathbb{Z}, |x| \leq 3\}, M_3 = \{x \mid x = 2k - 1, k \in \mathbb{N}, k \leq 5\}, M_4 = \{x \mid x = 2k, k \in \mathbb{N}, k \leq 5\}.$$

3. Gegeben sind die Mengen $A = \{2, 4, 6\}$, $B = \{2, 6\}$ und $C = \{4, 6, 8\}$. Welche dieser Mengen ist eine Teilmenge einer der anderen?

Lösung:

Es ist $B \subset A$.

4. Weisen Sie nach, dass aus $A \subset B$ und $B \subset C$ auch $A \subset C$ folgt! Visualisieren Sie diesen Sachverhalt mit Hilfe eines Venn-Diagramms!

Lösung:

Es ist zu zeigen, dass für ein $x \in A$ auch $x \in C$ gilt. Wegen $A \subset B$ folgt aus $x \in A$ sofort $x \in B$. Wegen $B \subset C$ folgt die Behauptung.

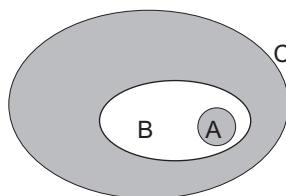


Abbildung 2.1: Venn-Diagramm zu Aufgabe 4

5. Bestimmen Sie für die Mengen $A = \{1, 2, 3, 4, 5\}$, $B = \{0, 3, 6\}$ die Mengen $A \cup B$, $A \cap B$, $A \setminus B$ und $B \setminus A$.

Lösung:

Es ist $A \cup B = \{0, 1, 2, 3, 4, 5, 6\}$, $A \cap B = \{3\}$, $A \setminus B = \{1, 2, 4, 5\}$, $B \setminus A = \{0, 6\}$.

6. Visualisieren Sie die Gleichung $(S \cup T) \setminus V = (S \setminus V) \cup (T \setminus V)$ mit Hilfe eines Venn-Diagramms und führen Sie den Nachweis der Mengengleichheit!

Lösung:

Ein $x \in (S \cup T) \setminus V$ liegt in S oder in T , aber nicht in V . Dann gilt aber auch $x \in S \setminus V$ oder $x \in T \setminus V$.

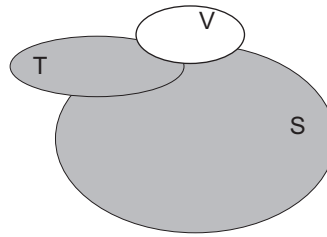


Abbildung 2.2: Venn-Diagramm zu Aufgabe 6

Ein $x \in (S \setminus V) \cup (T \setminus V)$ liegt in S oder in T ; aber nicht in V .

7. Vereinfachen Sie den Ausdruck $(S^c \cap T^c)^c \cap (S^c \cap T^c)$!

Lösung:

Mit den De Morgan'schen Regeln folgt:

$$\begin{aligned} (S^c \cap T^c)^c \cap (S^c \cap T^c) &= (S \cup T) \cap (S^c \cap T^c) \\ &= (S \cup T) \cap (S \cup T)^c \\ &= \emptyset. \end{aligned}$$

8. Bestimmen Sie für $S = \{1, 2\}$, $T = \{a, b\}$ und $V = \{b, c\}$ die folgenden Mengen: $S \times (T \cap V)$, $(S \times T) \cup (S \times V)$ und $(S \times T) \cap (S \times V)$!

Lösung:

$$S \times (T \cap V) = \{(1, b), (2, b)\},$$

$$(S \times T) \cup (S \times V) = \{(1, a), (1, b), (1, c), (2, a), (2, b), (2, c)\}$$

$$(S \times T) \cap (S \times V) = \{(1, b), (2, b)\}.$$

9. Bestimmen Sie die Potenzmenge der folgenden Mengen: $M_1 = \{a\}$, $M_2 = \{a, b\}$, $M_3 = \{\emptyset, a, b\}$!

Lösung:

$$\mathbb{P}(M_1) = \{\emptyset, \{a\}\},$$

$$\mathbb{P}(M_2) = \{\emptyset, \{a\}, \{b\}, \{a, b\}\},$$

$$\mathbb{P}(M_3) = \{\emptyset, \{\emptyset\}, \{a\}, \{b\}, \{\emptyset, a\}, \{\emptyset, b\}, \{a, b\}, \{\emptyset, a, b\}\}.$$

10. Implementieren Sie eine Darstellung einer Menge mit n Elementen mit Hilfe der Charakteristik und beschreiben Sie damit für zwei Teilmengen A und B die Mengen $A \cup B$, $A \cap B$, $A \setminus B$.

Lösung:

Hier finden Sie eine Implementierung mit Hilfe der STL-Klasse `vector`:

```
/*!
 * Die Klasse repräsentiert eine Menge und ihre Teilmengen
 * mit Hilfe von Charakteristiken.
 *
 * Die Mengen werden durch Teilmengen der natürlichen Zahlen repräsentiert,
 * beginnend bei 1.
 *
 * Eine Menge mit drei Elementen also durch {1, 2, 3}.
 * Ihre Teilmenge {2,3} durch die Charakteristik (0,1,1).
 *
 * Die Charakteristiken werden mit Hilfe der STL-Klasse vector<bool>
 * realisiert.
 * Die STL-Klasse bitset kann nicht verwendet werden, da diese Klasse
 * es nicht zulässt, die Anzahl der Elemente dynamisch zu spezifizieren.
 */
#ifndef VLSET_H
#define VLSET_H
#include <vector>

class vlSet{
public:
    ///! Defaultkonstruktor; Menge mit einem Element.
    vlSet();
    ///! Kopierkonstruktor
    vlSet(const vlSet&);
    ///! Konstruktor mit Anzahl der Elemente; alle sind enthalten.
    vlSet(int n);
    ///! Konstruktor mit Anzahl der Elemente, mit konstantem Wert
    vlSet(int n, bool val);
    ///! Konstruktor mit Anzahl der Elemente und Feld
    vlSet(int n, bool*);

    ///! Ist Element i in der Teilmenge enthalten?
    bool in(int i);

    ///! Funktion, die die Elemente der Teilmenge auf cout ausgibt
    void print(void);

    ///! Funktion für das Komplement einer Teilmenge
    vlSet komplement(void);
    ///! Funktion für den Durchschnitt zweier Teilmengen
    vlSet schnitt(const vlSet);

    ///! Funktion für die Vereinigung zweier Teilmengen
    vlSet vereinigung(const vlSet);

    ///! Funktion für den Durchschnitt zweier Teilmengen
    vlSet differenz(const vlSet);

private:
    ///! Die Charakteristik; mit einer Defaultlänge von 1
    vector<bool> chi;
    vector<int> elemente;
};
#endif // VLSET_H

#include "vlSet.h"

vlSet::vlSet()
```

```
{
    chi = vector<bool>(false);
    elemente = vector<int>(1);
}

vlSet::vlSet(const vlSet& kopie)
{
    int i, n = kopie.elemente.size();
    chi = vector<bool>(n);
    elemente = vector<int>(n);

    for (i=0; i<n; i++)
    {
        this->chi[i] = kopie.chi[i];
        this->elemente[i] = kopie.elemente[i];
    }
}

vlSet::vlSet(int n)
{
    int i;

    chi = vector<bool>(n, false);
    elemente = vector<int>(n);

    for (i=0; i<n; i++)
        elemente[i] = i+1;
}

vlSet::vlSet(int n, bool val)
{
    int i;
    chi = vector<bool>(n, val);
    elemente = vector<int>(n);

    for (i=0; i<n; i++)
        elemente[i] = i+1;
}

vlSet::vlSet(int n, bool *values)
{
    int i;

    chi = vector<bool>(n);
    elemente = vector<int>(n);

    for (i=0; i<n; i++)
        elemente[i] = i+1;

    for (i = 0; i<n; i++)
        chi[i] = values[i];
}

bool vlSet::in(int i)
{
    return chi[i];
}
```

```
void vlSet::print(void)
{
    int i, n = chi.size();

    for (i=0; i<n; i++)
    {
        if (chi[i])
            cout << elemente[i] << " ";
    }

    cout << endl;
}

vlSet vlSet::komplement(void)
{
    int i, n = elemente.size();
    vlSet ergebnis(n);

    for (i=0; i<n; i++)
        ergebnis.chi[i] = !chi[i];

    return ergebnis;
}

vlSet vlSet::schnitt(const vlSet b)
{
    int i, n = elemente.size();
    vlSet ergebnis(n);

    if (n != b.elemente.size()) return ergebnis;

    for (i=0; i<n; i++) {
        if (chi[i] && b.chi[i])
            ergebnis.chi[i] = true;
    }
    return ergebnis;
}

vlSet vlSet::vereinigung(const vlSet b)
{
    int i, n = elemente.size();
    vlSet ergebnis(n);

    if (n != b.elemente.size()) return ergebnis;

    for (i=0; i<n; i++) {
        if (chi[i] || b.chi[i])
            ergebnis.chi[i] = true;
    }
    return ergebnis;
}

vlSet vlSet::differenz(const vlSet b)
{
    int i, n = elemente.size();
    vlSet ergebnis(n);
```

```
if (n != b.elemente.size()) return ergebnis;

for (i=0; i<n; i++) {
    if (chi[i] && (!b.chi[i]))
        ergebnis.chi[i] = true;
}

return ergebnis;
}
```

Und hier ein Testprogramm:

```
#include <iostream>
#include "vlSet.h"

int main()
{
    bool ba[3] = {true, false, true},
        bb[3] = {false, false, true},
        bc[3] = {false, true, true};

    vlSet a(3, ba), b(3, bb), c(3, bc), resultat(3);

    cout << "a = " << endl;
    a.print();

    cout << "b = " << endl;
    b.print();

    cout << "c = " << endl;
    c.print();

    cout << "Das Komplement von a = {1, 3}" << endl;
    resultat = a.komplement();

    resultat.print();

    cout << "Der Durchschnitt von a und b:" << endl;
    resultat = a.schnitt(b);

    resultat.print();

    cout << "Die Vereinigung von b und c:" << endl;
    resultat = b.vereinigung(c);

    resultat.print();

    cout << "Die Differenz von c und b:" << endl;
    resultat = c.differenz(b);

    resultat.print();
    return 0;
}
```

Hier die Ausgaben des Testprogramms:

```
a =
1 3
b =
```

```

3
c =
2 3
Das Komplement von a = {1, 3}:
2
Der Durchschnitt von a und b:
3
Die Vereinigung von b und c:
2 3
Die Differenz von c und b:
2

```

11. Auf wie viele Arten können Sie aus 9 deutsch-, 3 englisch- und 11 französisch-sprachigen Büchern zwei verschiedensprachige auswählen?

Lösung:

Auf $9 \cdot 3 + 9 \cdot 11 + 3 \cdot 11 = 159$ Arten.

12. Auf wie viele Arten können Sie zwei Felder eines Schachbretts auswählen, die in verschiedenen Zeilen und Spalten liegen?

Lösung:

Ein Schachbrett hat $8 \cdot 8 = 64$ Felder. Nach der Wahl des ersten Felds, für das es 64 Möglichkeiten gibt, stehen noch $7 \cdot 7$ Felder zur Verfügung. Also gibt es $64 \cdot 49 = 3136$ Arten der Auswahl.

13. Auf wie viele Arten können Sie ein Skatspiel mit 32 Karten mischen?

Lösung:

Das Mischen entspricht einer 32-Permutation aus 32 Elementen ohne Wiederholung, also ist die Antwort gegeben durch $32!$.

14. Ein Passwort besteht aus zwei Buchstaben und vier Ziffern, wobei die Ziffern, aber nicht die Buchstaben mehrfach auftreten dürfen. Klein- und Großschreibung ist als signifikant anzusehen. Wie viele Passwörter können Sie bilden?

Lösung:

Die zwei Buchstaben auf die sechs Stellen zu verteilen entspricht $P(6, 2) = 6 \cdot 5 = 30$. Für eine getroffene Auswahl für die beiden Buchstaben gibt es dann $52 \cdot 51$ Möglichkeiten, für die Ziffern gibt es 10^4 Möglichkeiten. Insgesamt gibt es dann $30 \cdot (52 \cdot 51 + 10^4) = 382680$ Passwörter.

15. Weisen Sie die Gleichungen $k \binom{n}{k} = n \binom{n-1}{k-1}$, $\binom{n}{k} = \frac{n}{n-k} \binom{n-1}{k}$ nach!

Lösung:

Es ist

$$\begin{aligned}
 n \binom{n-1}{k-1} &= n \frac{(n-1)!}{(k-1)!(n-1-(k-1))!} \\
 &= \frac{n!}{(k-1)!(n-k)!} \\
 &= \frac{kn!}{k!(n-k)!} \\
 &= k \binom{n}{k}.
 \end{aligned}$$

und

$$\begin{aligned}\frac{n}{n-k} \binom{n-1}{k} &= \frac{n}{n-k} \frac{(n-1)!}{k!(n-k-1)!} \\ &= \frac{n!}{k!(n-k)!} = \binom{n}{k}.\end{aligned}$$

16. Weisen Sie nach, dass für eine endliche Menge die Anzahl der Teilmengen mit gerader Anzahl gleich der Anzahl der Teilmengen mit ungerader Anzahl ist!

Lösung:

Es gibt für ein $k \leq n$ genau $\binom{n}{k}$ Teilmengen mit k Elementen. Es ist also zu zeigen, dass

$$\sum_{k=0, k \text{ gerade}}^n \binom{n}{k} = \sum_{k=0, k \text{ ungerade}}^n \binom{n}{k}$$

gilt. Wenn Sie die Summe auf der rechten Seite nach links bringen, erhalten Sie die Gleichung

$$\sum_{k=0}^n (-1)^k \binom{n}{k} = 0.$$

Diese Formel stimmt allerdings, mit der binomischen Formel gilt

$$\sum_{k=0}^n (-1)^k \binom{n}{k} = (1-1)^n = 0.$$

17. Wie viele 10-stellige Dezimalzahlen enthalten die Ziffern 2 und 5, aber nicht 0, 1 oder 9?

Lösung:

Mit dem Inklusions-Exklusions-Prinzip folgt ähnlich der Argumentation im Buch als Lösung $7^{10} - 2 \cdot 6^{10} + 5^{10}$.

18. Eine Menge M mit 100 Elementen hat 3 Teilmengen A_1, A_2 , und A_3 mit $|M_1| = 60, |A_2| = 65, |A_3| = 20$ mit $A_1 \cup A_2 \cup A_3 = M$ und $\forall i, j |A_i \cap A_j| = 20$. Bestimmen sie $|A_1 \cap A_2 \cap A_3|$.

Lösung:

Laut Aufgabentext ist $N_2 = |A_i \cap A_j| = 20$.

Es ist $M = A_1^c \cup A_2^c \cup A_3^c$. Die Mächtigkeit von $A_1^c \cup A_2^c \cup A_3^c$ kann mit Hilfe des Inklusions-Exklusions-Prinzip berechnet werden als

$$100 = |A_1^c \cup A_2^c \cup A_3^c| = |A_1| + |A_2| + |A_3| - \binom{3}{2} N_2 + N_3.$$

Gesucht ist N_3 . Durch Auflösen ergibt sich

$$N_3 = 100 - (60 + 65 + 20) + 3 \cdot 20 = 100 - 145 + 60 = 15.$$

19. Implementieren Sie eine Funktion, die für eine natürliche Zahl n die Anzahl der Derangements D_n berechnet, und vergleichen Sie Ihre Ergebnisse mit den Werten aus Abschnitt 2.4!

Lösung:

Man könnte die Implementierung einfach mit der im Buch angegebenen Formel und mit Hilfe der Fakultät durchführen. Aber wenn Sie $n!$ mit jedem Summanden multiplizieren, dann erkennt man, dass

$$D_n = (-1)^n + \sum_{k=1}^{n-2} (-1)^{n-k} \prod_{j=0}^{k-1} (n-j)$$

gilt.

Dies führt zu einer Implementierung, die keine Fakultät benötigt. Das Produkt kann bei jedem Summanden mitberechnet werden.

```
long derangement(long n)
{
    long k, sign, faktor, result;

    if (n % 2 == 0)
        sign = 1;
    else
        sign = -1;
    result = sign;
    faktor = n;
    for (k=1; k<n-1; k++) {
        sign = -sign;
        result += sign*faktor;
        faktor *= n-k;
    }
    return result;
}
```

Und hier die Ausgabe für $n = 6$:

```
-----
Lösung der Aufgabe 19 zu Kapitel 2
-----
D_n fuer n=6 ist: 265
```

Eine Java-Implementierung:

```
private static long derangement(long n) {
    long k;
    long result, sign, faktor;

    if (n % 2 == 0)
        sign = 1;
    else
        sign = -1;
    result = sign;
    faktor = n;
    for (k=1; k<n-1; k++) {
        sign = -sign;
        result += sign*faktor;
        faktor *= n-k;
    }
    return result;
}
```

Kapitel 3

Logik

Verständnisfragen

Sachfragen

1. Was ist eine logische Aussage?
2. Wie ist die Konjunktion und die Disjunktion definiert?
3. Beschreiben Sie das Exklusive Oder, die Implikation und die Äquivalenz!
4. Wie ist der Peirce- und der Sheffer-Operator definiert?
5. Was ist eine Tautologie, was ein Widerspruch?
6. Wie können Sie für einen gegebenen logischen Ausdruck feststellen, ob er eine Tautologie oder ein Widerspruch darstellen?
7. Wann sind zwei logische Ausdrücke gleich?
8. Wie lauten die De Morgan'schen Regeln der Aussagenlogik?
9. Kann jeder logische Ausdruck durch einen Ausdruck ersetzt werden, der nur die Operatoren \neg, \wedge, \vee enthält? Gibt es eine analoge Aussage für den Sheffer- oder den Peirce-Operator?
10. Was ist ein Minterm?
11. Beschreiben Sie die disjunktive und die konjunktive Normalform eines logischen Ausdrucks!
12. Beschreiben Sie den Begriff des Karnaugh-Diagramms und seine Anwendung!
13. Was versteht man unter „SOPE“?
14. Was ist ein Prädikat?
15. Beschreiben Sie den All- und den Existenz-Quantor!
16. Beschreiben Sie die Negation von Prädikaten, die Quantoren beinhalten!
17. Was ist ein Gegenbeispiel?
18. Beschreiben Sie direkten Beweis!
19. Beschreiben Sie den Beweis durch Kontraposition!

20. Beschreiben Sie den Widerspruchsbeweis!
21. Beschreiben Sie den Beweis durch vollständige Induktion!

Methodenfragen

1. Verbale Aussagen in logische Ausdrücke umformen können.
2. Wahrheitstafeln für logische Ausdrücke aufstellen können.
3. Die Rechenregeln für logische Ausdrücke anwenden können.
4. Logische Ausdrücke mit \neg, \vee, \wedge , mit $|$ oder mit \downarrow schreiben können.
5. Für eine gegebene Wahrheitstafel eine Normalform aufstellen können!
6. Eine Normalform mit Hilfe von Karnaugh-Diagrammen oder den Rechenregeln vereinfachen können!
7. Den Wahrheitswert von Prädikaten feststellen können.
8. Prädikate mit Quantoren aufbauen und lesen können.
9. Prädikate mit Quantoren negieren können.
10. Kleine und einfache Beweise selbst führen können.
11. Beweise nachvollziehen können.
12. Induktionsbeweise durchführen können.

Übungsaufgaben

1. Wie viele Aussagenverbindungen gibt es für zwei logische Aussagen? Beschreiben Sie alle möglichen Verbindungen mit Hilfe einer Wahrheitstafel!

Lösung:

Es gibt insgesamt $16 = 2^{2^2} = 2^4$ mögliche Aussagenverbindungen, denn es gibt 2^n Zeilen in der Wahrheitstafel. In Tabelle 3.1 finden Sie eine Zusammenstellung.

p	q	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}
f	f	w	f	w	f	w	f	w	f	w	f	w	f	w	f	w	f
f	w	w	w	f	f	w	w	f	f	w	w	f	f	w	w	f	f
w	f	w	w	w	w	f	f	f	f	w	w	w	w	f	f	f	f
w	w	w	w	w	w	w	w	w	w	f	f	f	f	f	f	f	f

Tabelle 3.1: Sämtliche zweistellige Aussagenverbindungen

In der Tabelle 3.1 finden Sie die im Buch eingeführten, „klassischen“ Aussageverbindungen wieder. f_8 ist die Konjunktion, f_2 die Disjunktion. f_5 ist die Implikation und f_7 die Äquivalenz und f_{10} ist das exklusive oder. f_9 ist der Sheffer-Operator, f_{15} der Peirce-Operator.

f_1 und f_{16} sind die beiden Konstanten w und f , also Tautologie und Widerspruch. f_4 und f_6 stimmen mit p bzw. mit q überein. f_3 ist die Implikation $q \Rightarrow p$, dieser Operator wird auch *Konversion* genannt. f_{12} , die Negation der Implikation, wird auch *Inhibition* genannt. Und f_{14} ist die Negation der Implikation $f_3 = (q \Rightarrow p)$.

2. Bestimmen Sie die Wahrheitstafel für $(p \wedge q) \Rightarrow (p \vee q)$ und $p \Rightarrow (q \Rightarrow (r \vee \neg p))$.

Lösung:

$(p \wedge q) \Rightarrow (p \vee q)$ ist eine Tautologie.

$p \Rightarrow (q \Rightarrow (r \vee \neg p))$ hat die folgende Wahrheitstabelle:

p	q	r	$p \Rightarrow (q \Rightarrow (r \vee \neg p))$
f	f	f	w
f	f	w	w
f	w	f	w
f	w	w	w
w	f	f	w
w	f	w	w
w	w	f	f
w	w	w	w

3. Ist der Ausdruck $(p \Rightarrow q) \Leftrightarrow (\neg p \wedge q)$ eine Tautologie?

Lösung:

Der Ausdruck ist keine Tautologie. Wenn die Wahrheitswerte der Variablen p und q übereinstimmen ist der Ausdruck falsch.

4. Weisen Sie nach, dass $((p \Rightarrow q) \wedge (\neg p \Rightarrow q)) \Leftrightarrow q$ eine Tautologie ist!

Lösung:

Der Nachweis kann über eine Wahrheitstafel wie in Tabelle 3.2 erfolgen:

p	q	$p \Rightarrow q$	$\neg p \Rightarrow q$	$(p \Rightarrow q) \wedge (\neg p \Rightarrow q)$	$((p \Rightarrow q) \wedge (\neg p \Rightarrow q)) \Leftrightarrow q$
f	f	w	f	f	w
f	w	w	w	w	w
w	f	f	w	f	w
w	w	w	w	w	w

Tabelle 3.2: Wahrheitstafel zu Aufgabe 4

5. Weisen Sie nach, dass das Distributivgesetz $p \wedge (q \vee r) = (p \wedge q) \vee (p \wedge r)$ gilt!

Lösung:

Der Nachweis erfolgt über die Wahrheitstafel 3.3.

p	q	r	$q \vee r$	$p \wedge (q \vee r)$	$p \wedge q$	$p \wedge r$	$(p \wedge q) \vee (p \wedge r)$
f	f	f	f	\mathbf{f}	f	f	\mathbf{f}
f	f	w	w	\mathbf{f}	f	f	\mathbf{f}
f	w	f	w	\mathbf{f}	f	f	\mathbf{f}
f	w	w	w	\mathbf{f}	f	f	\mathbf{f}
w	f	f	f	\mathbf{f}	f	f	\mathbf{f}
w	f	w	w	\mathbf{w}	f	w	\mathbf{w}
w	w	f	w	\mathbf{w}	w	f	\mathbf{w}
w	w	w	w	\mathbf{w}	w	w	\mathbf{w}

Tabelle 3.3: Die Wahrheitstafel zu Aufgabe 5

6. Weisen Sie nach, dass $((p \Rightarrow q) \wedge (\neg p \Rightarrow q)) \Leftrightarrow q$ eine Tautologie ist!

Lösung:

Der Nachweis erfolgt über die Wahrheitstafel in Tabelle 3.4.

Tabelle 3.4: Wahrheitstafel zu Aufgabe 6

p	q	$p \Rightarrow q$	$\neg p \Rightarrow q$	$(p \Rightarrow q) \wedge (\neg p \Rightarrow q)$	$(p \Rightarrow q) \wedge (\neg p \Rightarrow q) \Leftrightarrow q$
f	f	w	f	f	w
f	w	w	w	w	w
w	f	f	w	f	w
w	w	w	w	w	w

7. Für eine Wechselschaltung könnte die Wahrheitstafel 3.5 stehen. Stellen Sie diese Wechselschaltung mit \vee , \wedge und \neg dar!

Tabelle 3.5: Eine Wechselschaltung

p	q	x
w	w	w
w	f	f
f	w	f
f	f	w

Lösung:

Es gibt zwei Minterme, in disjunktiver Normalform ergibt sich dann $(p \wedge q) \vee (\neg p \wedge \neg q)$. Der Wahrheitsverlauf ist identisch mit der Äquivalenz $p \Leftrightarrow q$.

Die konjunktive Normalform kann analog mit Mintermen aufgestellt werden. Statt den Zeilen, die im vorgeschriebenen Verlauf ein w enthalten verwendet man die Zeilen mit f , stellt die Minterme mit der Disjunktion auf und verbindet die Minterme mit der Konjunktion. Dann erhält man für die Wechselschaltung $(\neg p \vee q) \wedge (p \vee \neg q)$.

8. Die Wahrheitstafel 3.6 enthält die vorgegebenen Wahrheitsverläufe für zwei logische Schaltungen a und b . Schreiben Sie beide logische Ausdrücke in konjunktiver und disjunktiver Normalform und bestimmen Sie eine SOPE-Darstellung!

Tabelle 3.6: Zwei Schaltungen

p	q	r	a	b
f	f	f	f	w
f	f	w	w	w
f	w	f	f	w
f	w	w	f	f
w	f	f	f	f
w	f	w	w	f
w	w	f	f	f
w	w	w	w	f

Tabelle 3.7: Eingangsgrößen für Aufgabe 12

A	B	C	D	Ziffer
f	f	f	f	0
f	f	f	w	1
f	f	w	f	2
f	f	w	w	3
f	w	f	f	4
f	w	f	w	5
f	w	w	f	6
f	w	w	w	7
w	f	f	f	8
w	f	f	w	9

Lösung:

Für die Schaltung a gibt es drei Minterme, also lautet die disjunktive Normalform

$$a = (\neg p \wedge \neg q \wedge r) \vee (p \wedge \neg q \wedge r) \vee (p \wedge q \wedge r).$$

Die konjunktive Normalform lautet

$$a = (p \vee q \vee r) \wedge (p \vee \neg q \vee r) \wedge (p \vee \neg q \vee \neg r) \wedge (\neg p \vee q \vee \neg r) \wedge (p \vee q \vee \neg r).$$

Als SOPE erhält man

$$a = (p \wedge r) \vee (r \wedge \neg q) = r \wedge (p \vee \neg q).$$

Die Normalformen für die Schaltung b lauten

$$\begin{aligned} b &= (\neg p \wedge \neg q \wedge \neg r) \vee (\neg p \wedge \neg q \wedge r) \vee (\neg p \wedge q \wedge \neg r) \\ &= (p \vee \neg q \vee \neg r) \wedge (\neg p \vee q \vee r) \wedge (\neg p \vee q \vee \neg r) \wedge (\neg p \vee \neg q \vee r) \wedge (\neg p \vee \neg q \vee \neg r). \end{aligned}$$

Als SOPE erhält man

$$(\neg p \wedge \neg q) \vee (\neg p \wedge \neg r) = \neg p \wedge (\neg q \vee \neg r).$$

9. Schreiben Sie $(p \Rightarrow (q \Rightarrow r)) \Rightarrow ((p \Rightarrow q) \Rightarrow r)$ in disjunktiver und konjunktiver Normalform und stellen Sie eine SOPE-Darstellung auf!

Lösung:

Die Wahrheitstafel für den Ausdruck finden Sie in Tabelle 3.8. Damit können die gesuchten Normalformen aufgestellt werden.

p	q	r	$q \Rightarrow r$	$p \Rightarrow q$	$p \Rightarrow (q \Rightarrow r)$	$(p \Rightarrow q) \Rightarrow r$	$(p \Rightarrow (q \Rightarrow r)) \Rightarrow ((p \Rightarrow q) \Rightarrow r)$
f	f	f	w	w	w	f	f
f	f	w	w	w	w	w	w
f	w	f	f	w	w	f	f
f	w	w	w	w	w	w	w
w	f	f	w	f	w	w	w
w	f	w	w	f	w	w	w
w	w	f	f	w	f	f	w
w	w	w	w	w	w	w	w

Tabelle 3.8: Die Wahrheitstafel für Aufgabe 9

Die disjunktive Normalform enthält 6 Minterme:

$$\begin{aligned} &(\neg p \wedge \neg q \wedge \neg r) \vee (\neg p \wedge q \wedge \neg r) \vee (p \wedge \neg q \wedge \neg r) \\ &\vee (p \wedge \neg q \wedge r) \vee (p \wedge q \wedge \neg r) \vee (p \wedge q \wedge r) \end{aligned}$$

Die konjunktive Normalform besteht aus einem Minterm:

$$(p \vee r),$$

damit ist auch eine SOPE gefunden.

10. In einem Auto soll ein Warnton den Fahrer darauf aufmerksam machen, dass das Licht eingeschaltet ist, obwohl der Zündschlüssel abgezogen und die Tür bereits geöffnet wurden. Entwerfen Sie eine Schaltung, die dieses Problem löst!

Lösung:

$L = w$ entspricht „Licht an“, $S = w$ entspricht „Schlüssel steckt“, $T = w$ entspricht „Tür ist geschlossen“ und $W = w$ entspricht „Warnton ertönt“. Dann ist W darzustellen als

$$W = L \wedge \neg S \wedge \neg T.$$

11. Eine Öldruckkontrolle soll für folgende zwei Fälle Gefahr signalisieren: der Motor läuft, es ist aber kein Öldruck vorhanden, oder der Motor läuft und die Öltemperatur ist zu hoch. Entwickeln Sie eine SOPE-Darstellung für einen logischen Ausdruck, der diese Kontrolle realisiert!

Lösung:

$m = w$ soll „Motor läuft“ entsprechen, $d = w$ entspricht „Öldruck vorhanden“ und $t = w$ soll für „Temperatur in Ordnung“ stehen. Dann können Sie eine Wahrheitstafel für die drei Aussagen m, d und t und eine Normalform aufstellen; man erhält

$$f(m, d, t) = (m \wedge \neg d) \vee (m \wedge \neg t).$$

12. Mit dem BDC-7-Segment-Decoder in Abbildung 3.1 können die Ziffern 0 bis 9 durch Signale auf den Leitungen A, B, C und D dargestellt werden. Stellen Sie einen logischen Ausdruck auf, der die Darstellung der Ziffern ermöglicht und die Eingaben der Wahrheitstafel 3.7 verwendet!

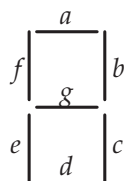


Abbildung 3.1: Ein 7-Segment-Decoder

Lösung:

Mit den logischen Ausdrücken

$$\begin{aligned} z_0 &= \neg A \wedge \neg B \wedge \neg C \wedge \neg D, \\ z_1 &= \neg A \wedge \neg B \wedge \neg C \wedge D, \\ z_2 &= \neg A \wedge \neg B \wedge C \wedge \neg D, \\ z_3 &= \neg A \wedge \neg B \wedge C \wedge D, \\ z_4 &= \neg A \wedge B \wedge \neg C \wedge \neg D, \\ z_5 &= \neg A \wedge B \wedge \neg C \wedge D, \\ z_6 &= \neg A \wedge B \wedge C \wedge \neg D, \\ z_7 &= \neg A \wedge B \wedge C \wedge D, \\ z_8 &= A \wedge \neg B \wedge \neg C \wedge \neg D, \\ z_9 &= A \wedge \neg B \wedge \neg C \wedge D \end{aligned}$$

erhalten Sie die Steuerung

$$\begin{aligned} a &= \neg(z_1 \vee z_4), \\ b &= \neg(z_5 \vee z_6), \\ c &= \neg z_2, \\ d &= \neg(z_1 \vee z_4 \vee z_7), \\ e &= z_0 \vee z_2 \vee z_6 \vee z_8, \\ f &= \neg(z_1 \vee z_2 \vee z_3 \vee z_7), \\ g &= \neg(z_0 \vee z_1 \vee z_7). \end{aligned}$$

13. Formulieren Sie die Peano-Axiome aus Kapitel 1 mit Hilfe von Prädikaten und Quantoren!

Lösung:

P1 $1 \in \mathbb{N}$.

P2 $\forall n \in \mathbb{N} \exists n^+ \in \mathbb{N} (n^+ = n + 1)$.

P3 $\forall n \in \mathbb{N} (n^+ = 1 \Rightarrow n \notin \mathbb{N})$.

P4 $\forall m, n \in \mathbb{N} (m + 1 = n + 1 \Rightarrow m = n)$.

P5 $\forall n \in \mathbb{N} ((p(1) \wedge (p(n) \Rightarrow p(n^+)) \Rightarrow p(n))$.

14. Formulieren Sie die Existenz eines inversen Elements für die Addition und Multiplikation von rationalen Zahlen mit Hilfe von Prädikaten und Quantoren!

Lösung:

Addition: $\forall p \in \mathbb{Q} \exists q \in \mathbb{Q} (p + q = 0)$.

Multiplikation: $\forall p \in \mathbb{Q}, p \neq 0 \exists q \in \mathbb{Q} (p \cdot q = 1)$.

15. Die folgenden Prädikate sind gegeben: $q(x) =$ „ x ist durch zwei teilbar“, $s(x) =$ „ x ist durch drei teilbar“, $t(x) =$ „ x ist durch sechs teilbar“. Als Grundbereich soll immer $G = \mathbb{N}$ verwendet werden. Formulieren Sie die Aussagen

$$\forall x (s(x) \Rightarrow q(x)), \forall x (t(x) \Rightarrow s(x)), \\ \forall x ((q(x) \wedge s(x)) \Leftrightarrow t(x)), \forall x (\neg q(x) \wedge \neg s(x) \wedge \neg t(x))$$

verbal und untersuchen Sie ihren Wahrheitswert. Bilden Sie auch die Negationen und formen Sie die negierten Ausdrücke so um, dass das Negationszeichen unmittelbar vor den Einzelaussagen steht!

Lösung:

$\forall x (s(x) \Rightarrow q(x))$: Jede natürliche Zahl, die durch 3 teilbar ist, ist auch durch 2 teilbar. Diese Aussage ist falsch, denn $x = 9$ ist zwar durch 3 teilbar, aber keine gerade Zahl.

Die Negation davon ist $\exists x \neg(s(x) \Rightarrow q(x)) = \exists x (s(x) \wedge \neg q(x))$. Verbal: „es gibt eine natürliche Zahl, die durch 3 und nicht durch 2 teilbar ist“. Eine wahre Aussage, $x = 9$ beispielsweise macht das Prädikat wahr.

$\forall x (t(x) \Rightarrow s(x))$: Jede natürliche Zahl, die durch 6 teilbar ist, ist auch durch 3 teilbar. Die Aussage ist richtig, im Abschnitt über die Beweisverfahren wurde diese Aussage bewiesen.

Die Negation davon ist $\exists x \neg(t(x) \Rightarrow s(x)) = \exists x (t(x) \wedge \neg s(x))$. Verbal: „es gibt eine natürliche Zahl, die durch 6 und nicht durch 3 teilbar ist“. Eine offensichtlich falsche Aussage.

$\forall x ((q(x) \wedge s(x)) \Leftrightarrow t(x))$: Eine natürliche Zahl ist genau dann durch 2 und 3 teilbar, wenn sie durch 6 teilbar ist. Die Aussage ist richtig; liegt an $6 = 2 \cdot 3$.

Die Negation davon ist $\exists x \neg((q(x) \wedge s(x)) \Leftrightarrow t(x)) = \exists x (q(x) \wedge s(x) \oplus t(x))$. Verbal: „es gibt eine natürliche Zahl, die entweder durch 2 und 3 oder durch 6 teilbar ist“. Eine falsche Aussage.

$\forall x (\neg q(x) \wedge \neg s(x) \wedge \neg t(x))$: Jede natürliche Zahl ist nicht durch 2, 3 und 6 teilbar. Diese Aussage ist falsch, denn $6 \in \mathbb{N}$.

Die Negation davon ist $\exists x \neg(\neg q(x) \wedge \neg s(x) \wedge \neg t(x)) = \exists x (q(x) \vee s(x) \vee t(x))$. Verbal: „es gibt eine natürliche Zahl, die durch 2 oder durch 3 oder durch 6 teilbar ist“. eine Wahre Aussage.

16. Bilden Sie die Negation der folgenden Aussagen: $\forall p, q \in \mathbb{Q} \exists r \in \mathbb{R} (p < r < q)$, $\forall x \in \mathbb{R} \exists n \in \mathbb{N} (n > x)$, $\exists p \in \mathbb{Q} (p^2 = 2)$ und beschreiben Sie die Aussage und ihre Negation verbal.

Lösung:

$\forall p, q \in \mathbb{Q} \exists r \in \mathbb{R} (p < r < q)$: verbal ist diese Aussage „für alle rationalen Zahlen p und q gibt es eine reelle Zahl r mit $p < r < q$ “. Eine wahre Aussage, beispielsweise können Sie r als das arithmetische Mittel von p und q setzen.

Die Negation davon ist $\exists p, q \in \mathbb{Q} \forall r \in \mathbb{R} \neg(p < r < q) = \exists p, q \in \mathbb{Q} \forall r \in \mathbb{R} (r \notin (p, q))$.
Verbal: „es gibt zwei rationale Zahlen, sodass im Intervall (p, q) keine reelle Zahl liegt“. Eine falsche Aussage.

$\forall x \in \mathbb{R} \exists n \in \mathbb{N} (n > x)$: verbal ist dies Aussage „für jede reelle Zahl x gibt es eine natürliche Zahl, die größer als x ist“. Eine wahre Aussage, die unter der Bezeichnung *Archimedes-Axiom* bekannt ist.

Die Negation davon ist $\exists x \in \mathbb{R} \forall n \in \mathbb{N} \neg(n > x) = \exists x \in \mathbb{R} \forall n \in \mathbb{N} (n \leq x)$. Verbal: „es gibt eine reelle Zahl x , so dass alle natürlichen Zahlen kleiner oder höchstens gleich x sind“. Eine falsche Aussage.

$\exists p \in \mathbb{Q} (p^2 = 2)$: verbal ist diese Aussage „es gibt eine rationale Zahl, deren Quadrat 2 ist“. Eine falsche Aussage, die Tatsache, dass $\sqrt{2}$ irrational ist war der Grund für die Einführung der reellen Zahlen.

Die Negation davon ist $\forall p \in \mathbb{Q} \neg(p^2 = 2) = \forall p \in \mathbb{Q} (p^2 \neq 2)$. Eine wahre Aussage!

17. Beweisen Sie $a \geq 0, b \geq 0 \Rightarrow \frac{(a+b)^2}{4} \geq ab$.

Lösung:

Der Beweis kann direkt geführt werden:

$$\frac{(a+b)^2}{4} - ab = \frac{a^2 + 2ab + b^2 - 4ab}{4} = \frac{(a-b)^2}{4} \geq 0.$$

Quadrate sind immer positiv!

18. Beweisen Sie durch einen direkten und einen indirekten Beweis, dass das Produkt zweier ungerader Zahlen ungerade ist.

Lösung:

Direkter Beweis:

Für zwei ungerade Zahlen $2n + 1$ und $2m + 1$ multiplizieren Sie einfach aus: $(2n + 1)(2m + 1) = 2(2nm + n + m) + 1$. Das Ergebnis ist offensichtlich wieder eine ungerade Zahl.

Indirekter Beweis:

Angenommen, es gibt zwei ungerade Zahlen $2n + 1$ und $2m + 1$, deren Produkt gerade ist. Dann gibt es eine natürliche Zahl k mit $(2n + 1)(2m + 1) = 2k$. Multiplizieren Sie die Klammern auf der linken Seite aus, dann erhalten Sie die Gleichung $2(2nm + n + m) = 2k - 1$, ein Widerspruch.

19. Beweisen Sie $2^n < n!$ für $n > 3$.

Lösung:

Die Induktionsbasis ist $n = 4$: $2^4 = 16 < 4! = 24$. Der Induktionsschritt folgt aus $2^{n+1} = 2^n \cdot 2 < 2n! < (n+1)n! = (n+1)!$.

20. Beweisen Sie durch vollständige Induktion die Aussagen $\sum_{i=1}^n i = \frac{n(n+1)}{2}$, $\sum_{i=1}^n (4i - 2) = 2n^2$.

Lösung:

Der Induktionsschritt für die erste Aussage kann geführt werden als

$$\sum_{i=1}^{n+1} i = \frac{n(n+1)}{2} + n + 1 = \frac{(n+1)(n+2)}{2}.$$

Die zweite Behauptung folgt mit dem Induktionsschritt

$$\sum_{i=1}^{n+1} (4i - 2) = 2n^2 + (4(n+1) - 2) = 2n^2 + 4n + 2 = 2(n^2 + 2n + 1) = 2(n+1)^2.$$

Kapitel 4

Relationen und Abbildungen

Verständnisfragen

Sachfragen

1. Was ist eine Relation?
2. Beschreiben Sie die Komposition von Relationen!
3. Was ist eine inverse Relation?
4. Was versteht man unter einer identischen Relation?
5. Was ist eine Äquivalenzrelation?
6. Erläutern Sie die Eigenschaften reflexiv, symmetrisch, antisymmetrisch, asymmetrisch und transitiv einer Relation! Nennen Sie Beispiele!
7. Was ist eine Partition?
8. Erläutern Sie den Zusammenhang zwischen Partitionen und Äquivalenzrelationen!
9. Was ist eine teilgeordnete Menge?
10. Was ist eine strikte Teilordnung?
11. Erläutern Sie den Begriff der oberen bzw. unteren Schranke und des Supremums und Infimums!
12. Was ist ein minimales und was ist ein maximales Element einer teilgeordneten Menge?
13. Was versteht man unter einer verträglichen Ordnung in einer teilgeordneten Menge?
14. Was ist eine topologische Sortierung?
15. Was ist eine Abbildung?
16. Was ist eine Funktion?
17. Erläutern Sie die Eigenschaft rechtseindeutig für eine Relation!
18. Was ist der Graph einer Funktion?
19. Was ist ein Definitions- und ein Wertebereich?

20. Was ist das Bild einer Abbildung?
21. Erläutern Sie die Eigenschaften injektiv, surjektiv und bijektiv von Abbildungen!
22. Was ist eine inverse Abbildung? Wann existiert sie?
23. Erläutern Sie den Zusammenhang zwischen Relationen, Abbildungen und Datenbanken!
24. Wann sind zwei Mengen gleichmächtig?
25. Was ist eine abzählbare Menge?
26. Was beschreibt das Symbol \aleph_0 ?
27. Wie kann nachgewiesen, dass \mathbb{Z} abzählbar ist?
28. Wie kann nachgewiesen, dass \mathbb{Q} abzählbar ist?
29. Wie kann nachgewiesen, dass \mathbb{R} überabzählbar ist?
30. Erläutern Sie das Cauchy'sche Diagonalisierungsschema!
31. Erläutern Sie das Cantor'sche Diagonalisierungsschema!
32. Erläutern Sie den Zusammenhang zwischen Mengen und Abbildungen!
33. Wie kann nachgewiesen werden, dass die Menge aller Algorithmen abzählbar ist?
34. Wie kann nachgewiesen werden, dass die Menge aller totalen Abbildungen $f : M^* \rightarrow M^*$ überabzählbar ist?
35. Gibt es nicht-berechenbare Funktionen?

Methodenfragen

1. Die verschiedenen Darstellungsmöglichkeiten für Relationen anwenden können.
2. Die Komposition von Relationen bilden können.
3. Für eine gegebene Relation entscheiden können, welche Eigenschaften sie hat.
4. Für eine gegebene Äquivalenzrelation die Äquivalenzklassen bilden können.
5. Entscheiden können, ob eine Äquivalenz- oder Ordnungsrelation vorliegt.
6. Ein kartesisches Produkt lexikographisch anordnen können.
7. Elemente aus M^* für ein gegebenes Alphabet lexikographisch anordnen können.
8. Supremum, Infimum, minimale und maximale Elemente einer teilgeordneten Menge bestimmen können.
9. Das Hasse-Diagramm einer teilgeordneten Menge konstruieren können.
10. Für eine teilgeordnete Menge eine topologische Sortierung bestimmen können.
11. Definitions- und Wertebereich einer Abbildung bestimmen können.
12. Werte einer gegebenen Abbildung und insbesondere einer Funktion bestimmen können.
13. Abbildungen und insbesondere Funktionen verketteten können.
14. Den Graphen einer Abbildung und insbesondere einer Funktion darstellen können.

15. Feststellen können, ob eine gegebene Abbildung bzw. Funktion injektiv und/oder surjektiv ist.
16. Für eine bijektive Abbildung die inverse Abbildung angeben können.
17. Die inverse Funktion einer bijektiven Verkettung von Funktionen bilden können.
18. Projektion, Restriktion und Verbund für gegebene Relationen bilden können.
19. Für eine unendliche Teilmenge von \mathbb{N} nachweisen können, dass sie abzählbar ist.
20. Funktionswerte der Funktion $\phi : M^* \rightarrow \mathbb{N}$ für ein gegebenes Alphabet berechnen können.

Übungsaufgaben

1. Stellen Sie die Relation $R = \{(1, 2), (1, 5), (1, 6), (2, 2), (2, 4), (3, 4), (3, 6), (4, 6), (5, 5)\}$ auf $M = \{1, 2, 3, 4, 5, 6\}$ als Pfeildiagramm, als Tabelle, als gerichteter Graph und als binäre Matrix dar!

Lösung:

Die Darstellung als binäre Matrix ist gegeben durch

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

In Tabelle 4.1 sehen Sie die Tabellendarstellung; in den Abbildungen 4.1 und 4.2 die Pfeildarstellung und der gerichtete Graph.

x	y
1	2
1	5
1	6
2	2
2	4
3	4
3	6
4	6
5	5

Tabelle 4.1: Die Tabellendarstellung zur Aufgabe 1

2. Bilden Sie für $R_1 = \{(a, b), (a, c)\}$ und $R_2 = \{(b, d), (c, a)\}$ die Kompositionen $R_2^{-1} \circ R_1^{-1}$ und $R_1^{-1} \circ R_2^{-1}$!

Lösung:

$$R_2^{-1} \circ R_1^{-1} = \{(d, a), (a, a)\}, R_1^{-1} \circ R_2^{-1} = \{(b, c), (c, c)\}.$$

3. Ist für reflexive Relationen $R_1, R_2 \subset M^2$ die Komposition $R_1 \circ R_2 \subset M^2$ reflexiv? Beweisen Sie, dass für reflexive Relationen $R \subset R \circ R$ gilt!

Lösung:

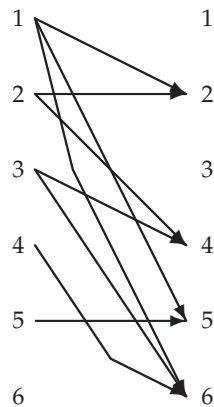


Abbildung 4.1: Das Pfeildiagramm zur Aufgabe 1

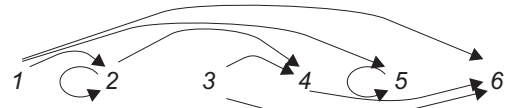


Abbildung 4.2: Der gerichtete Graph zur Aufgabe 1

Wenn beide Relationen reflexiv sind, dann ist $(x, x) \in R_1$ und $(x, x) \in R_2$ für alle $x \in M$. Zu zeigen ist, dass auch $\forall x \in M (x, x) \in R_2 \circ R_1$ gilt. Das folgt aber aus der Definition der Komposition.

Für $(x, y) \in R$ müssen Sie $(x, y) \in R \circ R$ nachweisen. R ist reflexiv, also ist $(x, x) \in R$. Mit $(x, y) \in R$ folgt dann $(x, y) \in R \circ R$.

4. Ist die Aussage $R \circ R \subseteq R$ äquivalent zur Definition einer transitiven Relation?

Lösung:

Wenn Sie aufschreiben, was $R \circ R \subseteq R$ bedeutet, steht da die Definition der Transitivität!

5. Geben Sie sämtliche Partitionen der Menge $M = \{9, 12, 21\}$ an. Weisen Sie nach, dass $R = \{(x, y) \in M^2 \mid x \text{ hat die gleiche Quersumme wie } y\}$ eine Äquivalenzrelation ist, und bestimmen Sie die Äquivalenzklassen!

Lösung:

Die möglichen Partitionen sind $K_1 = \emptyset, K_2 = M; K_1 = \{9\}, K_2 = \{12\}, K_3 = \{21\}; K_1 = \{9\}, K_2 = \{12, 21\}; K_1 = \{12\}, K_2 = \{9, 21\}; K_1 = \{21\}, K_2 = \{9, 12\}$.

Dass die angegebene Relation reflexiv, symmetrisch und transitiv ist, können Sie leicht nachrechnen. Die Äquivalenzklassen sind $[9] = \{9\}, [12] = \{12, 21\}$.

6. Welche der folgenden Relationen sind Äquivalenzrelationen, wenn M die Menge aller Menschen und W die Menge aller Wassertürme in Deutschland darstellt? $R_m = \{(x, y) \in M^2 \mid x \text{ ist am gleichen Tag wie } y \text{ geboren}\}$, $R_{10} = \{(n, m) \in \mathbb{N}^2 \mid n + m = 10\}$, $R_w = \{(x, y) \in W^2 \mid x \text{ ist höher als } y\}$.

Lösung:

R_m ist eine Äquivalenzrelation.

R_{10} ist keine Äquivalenzrelation, denn sie ist nicht reflexiv, $(4, 4) \notin R_{10}$.

R_w ist keine Äquivalenzrelation, sie enthält überhaupt kein Paar (x, x) .

7. Die Relation $R \subset \mathbb{N}^2 \times \mathbb{N}^2$ soll gegeben sein durch $((a, b), (c, d)) \in R \Leftrightarrow a \cdot d = b \cdot c$. Ist R eine Äquivalenzrelation?

Lösung:

Reflexivität: ist $(a, b) \in \mathbb{N}^2$ ein Paar von natürlichen Zahlen, dann ist $((a, b), (a, b)) \in R$, denn es ist $a \cdot b = b \cdot a$.

Symmetrie: falls $((a, b), (c, d)) \in R$, dann ist $a \cdot d = b \cdot c$. Dann ist auch $c \cdot b = d \cdot a$, und $((c, d), (a, b)) \in R$.

Transitivität: falls $((a, b), (c, d)) \in R$ und $((c, d), (e, f)) \in R$, dann muss nachgewiesen werden, dann auch $((a, b), (e, f)) \in R$.

Es ist $a \cdot d = b \cdot c$ und $c \cdot f = d \cdot e$; damit gilt auch

$$b = \frac{a \cdot d}{c}, f = \frac{d \cdot e}{c}.$$

Dann gilt

$$a \cdot f = a \frac{d \cdot e}{c} = \frac{a \cdot d}{c} \cdot e = b \cdot e.$$

Damit ist R eine Äquivalenzrelation.

8. Welche der folgenden Mengen von Teilmengen von $M = \{1, 2, 3, 4, 5, 6\}$ ist eine Partition:
 $T_1 = \{\{1, 2\}, \{2, 3, 4\}, \{4, 5, 6\}\}$, $T_2 = \{\{1\}, \{2, 3, 6\}, \{4\}, \{5\}\}$, $T_3 = \{\{2, 4, 6\}, \{1, 3, 5\}\}$,
 $T_4 = \{\{1, 4, 5\}, \{2, 6\}\}$?

Lösung:

T_1 ist keine Partition, denn $\{1, 2\} \cap \{2, 3, 4\} = \{2\} \neq \emptyset$.

T_2 und T_3 sind Partitionen, die Teilmengen sind disjunkt, als Vereinigung erhalten Sie M .

T_4 ist keine Partition; die Teilmengen sind zwar disjunkt, aber in der Vereinigung fehlt das Element $3 \in M$.

9. Welche der folgenden durch binäre Matrizen gegebenen Relationen sind Teilordnungen?

$$R_1 = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, R_2 = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix}, R_3 = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \end{pmatrix}.$$

Lösung:

Alle drei Relationen sind reflexiv, denn auf der Diagonalen steht 1.

R_1 ist nicht antisymmetrisch, denn sowohl das Element 23 und 32 der Matrix ist Null. Also kann R_1 keine Teilordnung sein. Auch R_3 ist nicht antisymmetrisch, da sowohl das Element 12 als auch das Element 21 der Matrix 0 ist. R_2 ist antisymmetrisch; denn für ein Element a_{ij} der binären Matrix mit $i \neq j$ mit $a_{ij} = 0$ ist $a_{ji} = 1$ bzw. für $a_{ij} = 1$ ist $a_{ji} = 0$.

R_2 ist auch transitiv; es gilt $R_2 \circ R_2 = R_2$. Damit ist R_2 eine Teilordnung.

10. Beweisen Sie, dass für die teilgeordnete Menge (M, R) auch (M, R^{-1}) eine teilgeordnete Menge ist.

Lösung:

Wenn R reflexiv ist, dann ist auch R^{-1} reflexiv. Auch die Antisymmetrie für R^{-1} ist klar, wenn R antisymmetrisch ist; und mit dem gleichen Argument ist R^{-1} transitiv, wenn R diese Eigenschaft hat.

11. Es sei $S = \{1, 2, 3, 4\}$. Als Ordnung sei die lexikographische Ordnung mit \leq auf S^2 gegeben. Finden Sie alle Paare in S^2 , die kleiner als $(2, 3)$ oder größer als $(3, 1)$ sind!

Lösung:

Kleiner als $(2, 3)$ sind alle Paare, die als erste Komponente eine 1 haben, und die beiden Paare $(2, 1)$ und $(2, 2)$.

Größer als $(3, 1)$ sind alle Paare, deren erste Komponente größer als 3 sind, und die Paare $(3, 2)$, $(3, 3)$, $(3, 4)$.

12. Ordnen Sie die Bitstrings 0, 01, 11, 001, 010, 011, 0001 und 0101 lexikographisch an, mit der Teilordnung $0 < 1$ für ein Bit!

Lösung:

Es ist $0 < 0001 < 001 < 01 < 010 < 0101 < 011 < 11$.

13. Zeichnen Sie das Hasse-Diagramm für die Teilbarkeit ohne Rest auf den Mengen $M_1 = \{1, 2, 3, 4, 5, 6, 7, 8\}$, $M_2 = \{1, 2, 3, 5, 7, 11, 13\}$, $M_3 = \{1, 2, 3, 6, 12, 24, 36, 48\}$ und $M_4 = \{1, 2, 4, 8, 16, 32, 64\}$!

Lösung:

Die Diagramme finden Sie in Abbildung 4.3.

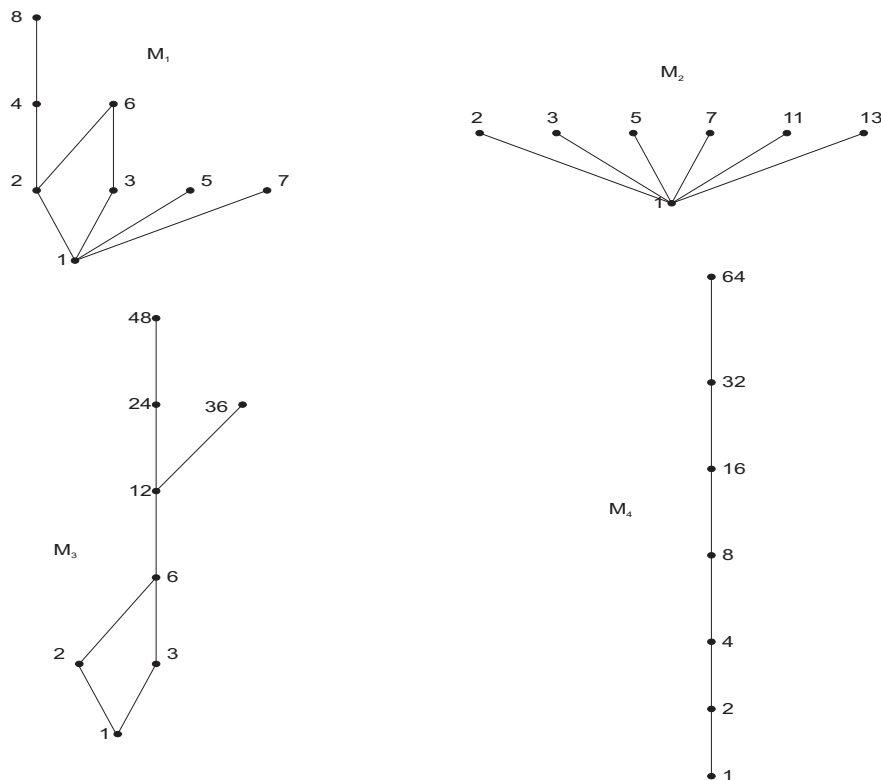


Abbildung 4.3: Die Hasse-Diagramme für Aufgabe 13

14. Gegeben ist die teilgeordnete Menge $(\{3, 5, 9, 15, 24, 45\}, |)$. Geben Sie die maximalen und minimalen Elemente an! Geben Sie alle oberen Schranken für die Teilmenge $\{3, 5\}$ und das Supremum an, falls es existiert. Bestimmen Sie die unteren Schranken der Teilmenge $\{15, 45\}$ und das Infimum, falls es existiert.

Lösung:

24 und 45 sind maximale Elemente, 3 und 5 minimale Elemente.

15 und 45 sind obere Schranken für $\{3, 5\}$. Das Supremum ist 15.

15, 5 und 3 sind untere Schranken für $\{15, 45\}$, das Infimum ist 15.

15. Für eine Menge M und ihre Potenzmenge $\mathbb{P}(M)$ sollen die Teilmengen $A, B \subseteq M$ betrachtet werden. Begründen Sie, dass $A \cap B$ das Infimum und $A \cup B$ das Supremum von $\{A, B\} \in \mathbb{P}(M)$ ist!

Lösung:

$A \cap B$ ist sicher eine untere Schranke für $\{A, B\}$, denn es ist $A \cap B \subseteq A$, $A \cap B \subseteq B$. Angenommen, es gibt eine untere Schranke, also eine Teilmenge $U \in \mathbb{P}(M)$ mit $A \cap B \subset U$ und $U \subseteq A$, $U \subseteq B$. Es soll also ein $x \in U$ geben, das nicht in $A \cap B$ liegt, aber in beiden Mengen A und B . Dann muss es im Schnitt liegen, ein Widerspruch. Also ist der Durchschnitt $A \cap B$ das Infimum.

Analog wird für $A \cup B$ argumentiert. Die Vereinigung ist sicher eine obere Schranke für $\{A, B\}$. Und es kann keine kleinere obere Schranke geben; eine Annahme kann zum Widerspruch geführt werden.

16. Bestimmen Sie eine topologische Sortierung für die teilgeordnete Mengen in 4.17 und 4.4!



Abbildung 4.4: Hasse-Diagramme für Aufgabe 16

Lösung:

Für die Teilordnung aus Abbildung 4.17 gibt es keine eindeutige topologische Sortierung; eine Möglichkeit ist $a, c, b, e, d, g, f, h, j$. Den Verlauf für diese Lösung sehen Sie in Abbildung 4.5.

Für die linke Teilordnung in Abbildung 4.4 ist eine topologische Sortierung gegeben durch a, b, d, c, e, g, f, h . Den Verlauf für diese Lösung sehen Sie in Abbildung 4.6.

Für die rechte Teilordnung in Abbildung 4.4 ist eine topologische Sortierung gegeben durch $c, f, g, a, b, e, d, i, h, j, k, l, m$. Den Verlauf für diese Lösung sehen Sie in Abbildung 4.7.

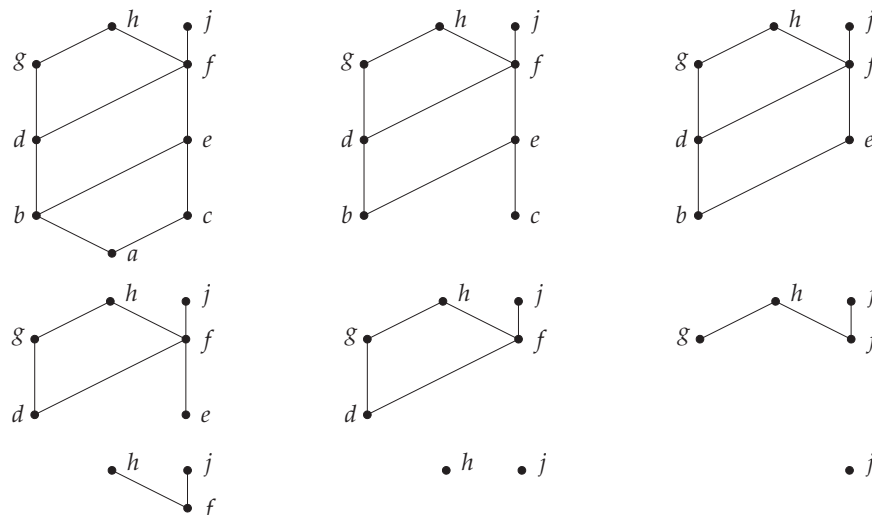


Abbildung 4.5: Lösung für die Teilordnung aus Abbildung 4.17 im Buch

17. M sei die Menge aller Bitfolgen. Welche der folgenden Relationen aus $M \times \mathbb{N}$ ist eine Abbildung: $f_1(x)$ ist die Position einer 0 in x ; $f_2(x)$ ist die Anzahl der 1 in x , $f_3(x)$ ist die kleinste natürliche Zahl i , sodass das i -te Bit eine 1 ist; dabei wird für das leere Wort $f_3(\epsilon) = 0$ gesetzt.

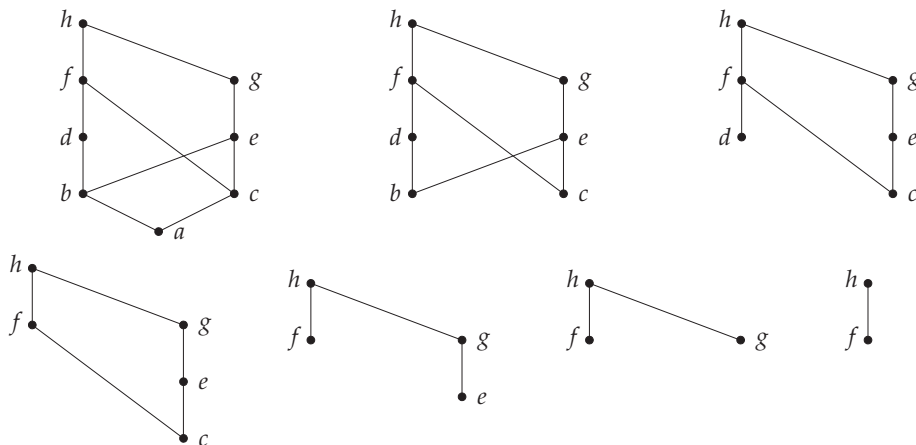


Abbildung 4.6: Lösung für die Teilordnung aus Abbildung 4.4 links

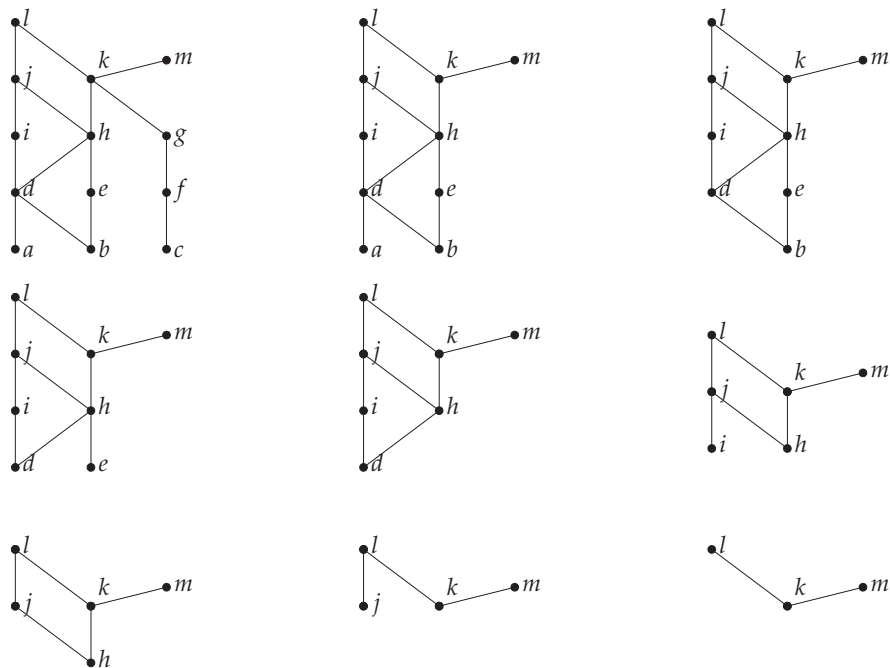


Abbildung 4.7: Lösung für die Teilordnung aus Abbildung 4.4 rechts

Lösung:

f_1 ist keine Abbildung; für die Bitfolge 1010 gibt es den Wert 2 und den Wert 4.

f_2 ist eine Abbildung; sie ist eine rechtseindeutige Relation.

f_3 ist eine Abbildung, denn sie ist eine rechtseindeutige Relation. Allerdings gibt es Bitfolgen, für die es keine Werte gibt wie beispielsweise 00.

18. Geben Sie die folgenden Werte an: $\lfloor 1, 1 \rfloor$, $\lceil \frac{3}{4} \rceil$, $\lfloor \frac{1}{2} + \lceil \frac{1}{2} \rceil \rfloor$ und $\lfloor \frac{1}{2} \cdot \lfloor \frac{5}{2} \rfloor \rfloor$. Weisen Sie die folgenden Eigenschaften nach: $\lfloor x + m \rfloor = \lfloor x \rfloor + m$, $\lceil x + m \rceil = \lceil x \rceil + m$ für $m \in \mathbb{Z}$.

Lösung:

Die gesuchten Werte sind

$$\lfloor 1, 1 \rfloor = 1 = \lceil \frac{3}{4} \rceil, \lfloor \frac{1}{2} + \lceil \frac{1}{2} \rceil \rfloor = \lfloor \frac{1}{2} + 1 \rfloor = 1, \lfloor \frac{1}{2} \cdot \lfloor \frac{5}{2} \rfloor \rfloor = \lfloor \frac{1}{2} \cdot 1 \rfloor = 0.$$

$$\lfloor x + m \rfloor = \lfloor x \rfloor + m:$$

Angenommen, es ist $\lfloor x \rfloor = n \in \mathbb{Z}$. Dann ist die Ungleichung $n \leq x < n + 1$ erfüllt. Wenn Sie zu dieser Ungleichungskette m addieren, erhalten Sie

$$n + m \leq x + m < n + m + 1.$$

Dann ist aber offensichtlich $\lfloor x + m \rfloor = n + m = \lfloor x \rfloor + m$.

$$\lceil x + m \rceil = \lceil x \rceil + m:$$

Angenommen, es ist $\lceil x \rceil = n \in \mathbb{Z}$. Dann ist die Ungleichung $n - 1 < x \leq n$ erfüllt. Wenn Sie zu dieser Ungleichungskette m addieren, erhalten Sie

$$n + m - 1 < x + m \leq n + m.$$

Dann ist aber offensichtlich $\lceil x + m \rceil = n + m = \lceil x \rceil + m$.

19. Untersuchen Sie die folgenden Abbildungen $f : \mathbb{Z} \rightarrow \mathbb{Z}$ auf Injektivität, Surjektivität und Bijektivität: $f_1(n) = n - 1$, $f_2(n) = n^2 + 1$, $f_3(n) = n^3$, $f_4(n) = \lceil \frac{n}{2} \rceil$.

Lösung:

f_1 ist bijektiv. Sie ist injektiv, aus $f(n) = f(m)$ folgt $n - 1 = m - 1$ und damit $n = m$. Sie ist surjektiv, denn für $m \in \mathbb{Z}$ ist das Urbild gegeben durch $m + 1$: $f(m + 1) = m$.

f_2 ist weder injektiv noch surjektiv. Sowohl n als auch $-n \in \mathbb{Z}$ haben das gleiche Bild $f_2(n) = f_2(-n) = n^2 + 1$. Alle ganze Zahlen kleiner als 1 haben kein Urbild, denn es ist $n^2 + 1 \geq 1$.

f_3 ist injektiv, denn aus $f_3(n) = f_3(m)$ folgt $n^3 = m^3$ und auch $n = m$. Sie ist nicht surjektiv, für 2 gibt es keine ganze Zahl mit $n^3 = 2$.

f_4 ist nicht injektiv, denn es gilt $f_4(1) = \lceil \frac{1}{2} \rceil = 1 = \lceil \frac{2}{2} \rceil = f_4(2)$. f_4 ist surjektiv. Für $m \in \mathbb{Z}$ ist das Urbild gegeben durch $2m$.

20. Bestimmen Sie für die reellen Funktionen $f(x) = x + 1$, $g(x) = 2x$ und $h(x) = x^2$ die Verkettungen $f \circ g$, $f \circ h$, $g \circ f$, $g \circ h$, $h \circ f$ und $h \circ g$!

Lösung:

Es gilt

$$(f \circ g)(x) = 2x + 1,$$

$$(f \circ h)(x) = x^2 + 1,$$

$$(g \circ f)(x) = 2(x + 1),$$

$$(g \circ h)(x) = 2x^2,$$

$$(h \circ h)(x) = (x + 1)^2,$$

$$(h \circ g)(x) = 4x^2.$$

21. Zeichnen Sie die Graphen der folgenden Funktionen $f_i : \mathbb{R} \rightarrow \mathbb{R}$: $f_1(x) = \lfloor 2x \rfloor$, $f_2(x) = \lceil \frac{x}{2} \rceil$, $f_3(x) = \lfloor x \rfloor + \lfloor \frac{x}{2} \rfloor$, $f_4(x) = \lfloor 2x \rfloor \cdot \lceil \frac{x}{2} \rceil$.

Lösung:

Die Graphen finden sie in Abbildung 4.8.

22. Finden Sie für $f(x) = \lfloor x \rfloor$ die folgenden Bilder der inversen Funktion: $f^{-1}(0)$, $f^{-1}(\{-1; 0, 1\})$, $f^{-1}(\{x \mid 0 < x < 1\})$.

Lösung:

$$f^{-1}(0) = [0; 1); f^{-1}(\{-1; 0, 1\}) = [-1; 2); f^{-1}(\{x \mid 0 < x < 1\}) = \emptyset.$$

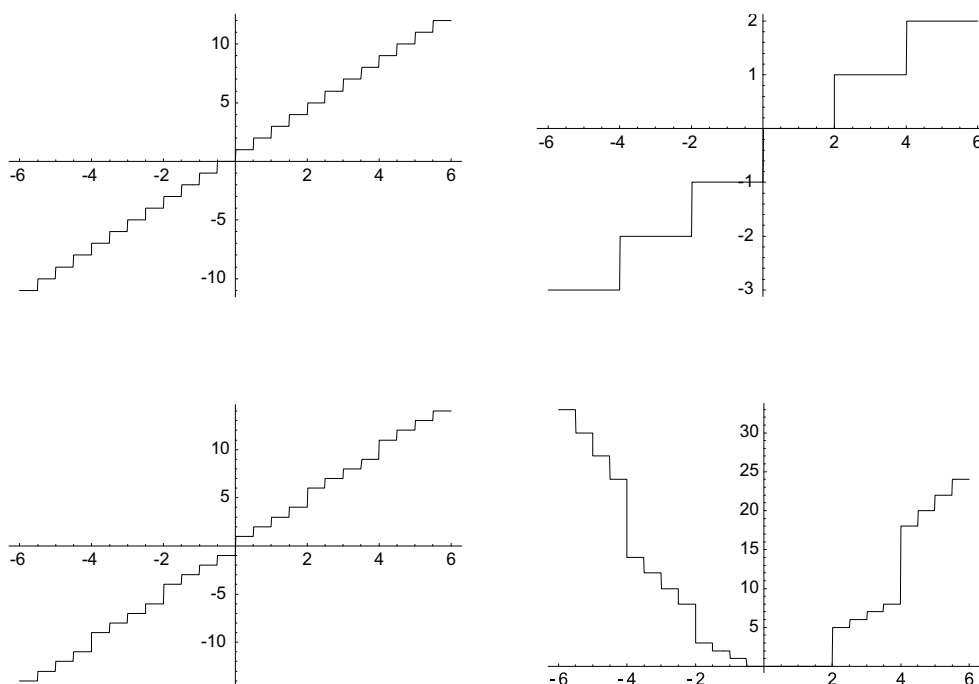


Abbildung 4.8: Die Graphen für Abbildung 21. Oben links f_1 , oben rechts f_2 , unten links f_3 , unter rechts f_4

23. Geben Sie an, welche der folgenden Mengen abzählbar ist, und geben Sie die Zuordnung zwischen \mathbb{N} und den abzählbaren Mengen an: die negativen ganzen Zahlen, die ungeraden natürlichen Zahlen, die reellen Zahlen zwischen 0 und $\frac{1}{2}$ und alle ganzen Zahlen, die ein Vielfaches von 7 sind.

Lösung:

Die negativen ganzen Zahlen sind abzählbar; die Zuordnung ist einfach gegeben durch $-n$ für $n \in \mathbb{N}$. Auch die ungeraden Zahlen sind abzählbar; sie sind eine Teilmenge einer abzählbaren Menge. Die Zuordnung ist gegeben durch $2n + 1$ für $n \in \mathbb{N}$.

Die reellen Zahlen zwischen 0 und $\frac{1}{2}$ sind überabzählbar; die Diagonalisierung kann genauso wie im Fall $[0; 1]$ durchgeführt werden.

Alle Vielfachen von 7 sind abzählbar. \mathbb{Z} ist abzählbar, und mit $7z$ für $z \in \mathbb{Z}$ ist eine Zuordnung gegeben.

24. Bestimmen Sie die Funktionswerte der Abbildung $\phi : M^* \rightarrow \mathbb{N}$ für das Alphabet $M_{10} = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ für die folgenden Worte: $w_1 = 55$, $w_2 = 101$, $w_3 = 999$. Bestimmen Sie für das Alphabet $M_2 = \{0, 1\}$ die Funktionswerte für die folgenden Worte: $w_4 = 0101$, $w_5 = 00011$, $w_6 = 111000$.

Lösung:

$$\phi(w_1) = 6 \cdot 11 + 6 \cdot 121 = 792.$$

$$\phi(w_2) = 2 \cdot 11 + 1 \cdot 11^2 + 2 \cdot 11^3 = 2805.$$

$$\phi(w_3) = 10 \cdot 11 + 10 \cdot 11^2 + 10 \cdot 11^3 = 14630.$$

$$\phi(w_4) = 1 \cdot 3 + 2 \cdot 3^2 + 1 \cdot 3^3 + 2 \cdot 3^4 = 210.$$

$$\phi(w_5) = 1 \cdot 3 + 1 \cdot 3^2 + 1 \cdot 3^3 + 2 \cdot 3^4 + 2 \cdot 3^5 = 687.$$

$$\phi(w_6) = 2 \cdot 3 + 2 \cdot 3^2 + 2 \cdot 3^3 + 1 \cdot 3^4 + 1 \cdot 3^5 + 1 \cdot 3^6 = 1131.$$

Kapitel 5

Lineare Gleichungssysteme und der Gauß-Algorithmus

Verständnisfragen

Sachfragen

1. Was ist ein lineares Gleichungssystem?
2. Was ist die erweiterte Koeffizientenmatrix eines linearen Gleichungssystems?
3. Was ist eine obere bzw. untere Dreiecksmatrix?
4. Was ist ein gestaffeltes lineares Gleichungssystem?
5. Wie verläuft die Rückwärts- bzw. Vorwärtssubstitution?
6. Den Verlauf der Gauß-Elimination erläutern können.
7. Die Matrix-Darstellung der Gauß-Elimination herleiten können.
8. Bedingungen für die Invertierbarkeit einer quadratischen Matrix kennen.
9. Die LU-Zerlegung und ihre Anwendung bei der Lösung von linearen Gleichungssystemen erläutern können.
10. Was versteht man unter Scheinskalierung?
11. Was versteht man unter der Pivotwahl? Welche Strategien kennen Sie?
12. Was ist eine Determinante?
13. Wie wird eine 2×2 Determinante berechnet?
14. Welche Eigenschaften haben Determinanten?
15. Wie wird eine 3×3 Determinanten berechnet?
16. Erläutern Sie den Laplace'schen Entwicklungssatz zur Berechnung allgemeiner Determinanten!
17. Was ist ein Eigenwert einer quadratischen Matrix?

18. Was ist ein Eigenvektor einer quadratischen Matrix?
19. Wie werden Eigenwerte und Eigenvektoren einer quadratischen Matrix berechnet?
20. Was versteht man unter der Ähnlichkeit von Matrizen?

Methodenfragen

1. Die erweiterte Koeffizientenmatrix eines linearen Gleichungssystems aufstellen können.
2. Ein gestaffeltes lineares Gleichungssystem erkennen und mit Rückwärts- oder Vorwärtssubstitution lösen können.
3. Elementare Zeilenumformungen durchführen können.
4. Gauß-Elimination durchführen können.
5. Elementarmatrizen zu elementaren Zeilenumformungen aufstellen können.
6. Feststellen können, ob eine gegebene quadratische Matrix invertierbar ist.
7. Die Inverse einer invertierbaren quadratischen Matrix mit Gauß-Jordan bestimmen können.
8. Die Inverse einer invertierbaren quadratischen Matrix mit linearen Gleichungssystemen bestimmen können.
9. Mit einer gegebenen LU-Zerlegung der Koeffizientenmatrix ein lineares Gleichungssystem lösen können.
10. Die LU-Zerlegung mit und ohne Zeilenvertauschungen durchführen können.
11. Die Gauß-Elimination und die LU-Zerlegung mit Scheinskalierung und Spaltenpivotsuche implementieren können.
12. 2×2 , 3×3 und beliebige $n \times n$ -Determinanten berechnen können.
13. Die Eigenschaften der Determinante anwenden können.
14. Der Laplace'schen Entwicklungssatz anwenden können.
15. Eigenwerte und Eigenvektoren einer quadratischen Matrix berechnen können.
16. Entscheiden können, ob eine Matrix zu einer Diagonalmatrix ähnlich ist.

Übungsaufgaben

1. Bestimmen Sie in der erweiterten Koeffizientenmatrix $\begin{pmatrix} a & b & c & -2 \\ c & a & b & 8 \\ b & c & a & 0 \end{pmatrix}$ die Koeffizienten a, b und c so, dass das zugehörige Gleichungssystem genau die Lösung $x_1 = 1, x_2 = -1$ und $x_3 = 2$ besitzt!

Lösung:

a, b, c ergeben sich als Lösung des Gleichungssystems mit der Koeffizientenmatrix

$$\begin{pmatrix} 1 & -1 & 2 \\ -1 & 2 & 1 \\ 2 & b & -1 \end{pmatrix}.$$

Das Gleichungssystem erhalten Sie, indem Sie die gegebene Lösung in die Gleichungen einsetzen und nach a, b und c sortieren. Die Lösung dieses Gleichungssystems ist gegeben durch $a = -1, b = 3, c = 1$.

2. Bestimmen Sie die Koeffizienten des Polynoms $p(x) = a_3x^3 + a_2x^2 + a_1x + a_0$ so, dass $p(-1) = 0, p(1) = 2, p(2) = 3, p(3) = 12$ erfüllt sind!

Lösung:

Die Koeffizienten des kubischen Polynoms ergeben sich als Lösung eines linearen Gleichungssystems. Die Gleichungen für das System erhalten Sie, indem sie die gegebenen Punkte in die Polynome einsetzen; die rechte Seite ist durch die vorgegebenen Funktionswerte gegeben. Als Lösung erhält man $a_3 = 1, a_2 = -2, a_1 = 0, a_0 = 3$.

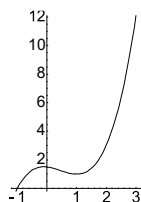


Abbildung 5.1: Das gesuchte Polynom aus Aufgabe 2

3. Bestimmen Sie die Elementarmatrizen E_1, E_2, E_3 und E_4 mit $E_1A = B, E_2B = A, E_3A = C, E_4C = A$ für die Matrizen $A = \begin{pmatrix} 3 & 4 & 1 \\ 2 & -7 & -1 \\ 8 & 1 & 5 \end{pmatrix}, B = \begin{pmatrix} 8 & 1 & 5 \\ 2 & -7 & -1 \\ 3 & 4 & 1 \end{pmatrix}, C = \begin{pmatrix} 3 & 4 & 1 \\ 2 & -7 & -1 \\ 2 & -7 & 3 \end{pmatrix}$.

Lösung:

Die Elementarmatrix E_1 ist die Matrix, die die erste mit der dritten Zeile vertauscht. Auch E_2 ist durch diese Matrix gegeben. E_3 ist die Matrix, die von der dritten Zeile das zweifache der ersten Zeile subtrahiert; E_4 addiert zur dritten Zeile das zweifache der ersten Zeile hinzu.

Es ist

$$E_1 = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}, E_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -2 & 0 & 1 \end{pmatrix}, E_4 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 2 & 0 & 1 \end{pmatrix}$$

4. Lösen Sie die linearen Gleichungssysteme $Ax = \mathbf{b}$ mit Hilfe der Gauß-Elimination für die erweiterten Koeffizientenmatrizen $\left(\begin{array}{ccc|c} 1 & 0 & 1 & 3 \\ 2 & 2 & 1 & 10 \\ 0 & 2 & 1 & 4 \end{array}\right)$ und $\left(\begin{array}{cccc|c} 9 & -11 & 10 & 4 & 9 \\ 2 & -2 & 2 & 1 & 3 \\ 7 & -16 & 11 & 2 & -7 \\ 1 & 0 & 0 & 2 & 8 \end{array}\right)$.

Lösung:

Die Lösung für das 3×3 Gleichungssystem ist $\mathbf{x} = (3, 2, 0)^T$; für das 4×4 System erhält man die Lösung $\mathbf{x} = (2, 1, -1, 3)^T$.

5. Bestimmen Sie die Inversen der Matrizen $A = \begin{pmatrix} 1 & 6 & 4 \\ 2 & 4 & -1 \\ -1 & 2 & 5 \end{pmatrix}$ und $B = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 3 & 0 \\ 1 & 3 & 5 \\ 1 & 3 & 5 & 7 \end{pmatrix}$!

Lösung:

A ist nicht invertierbar, der Gauß-Jordan Algorithmus bricht ab. B ist invertierbar und eine untere Dreiecksmatrix, also muss nur noch das untere Dreieck eliminiert werden. Das Ergebnis des Gauß-Jordan-Algorithmus ist

$$B^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -\frac{1}{3} & \frac{1}{3} & 0 & 0 \\ 0 & -\frac{1}{5} & \frac{1}{5} & 0 \\ 0 & 0 & -\frac{1}{7} & \frac{1}{7} \end{pmatrix}$$

6. Schreiben Sie eine Funktion in der Programmiersprache Ihrer Wahl, die die Rückwärtssubstitution für eine obere Dreiecks-Matrix A und eine rechte Seite \mathbf{b} durchführt. Testen Sie diese

Funktion an selbst gewählten linearen Gleichungssystemen!

Lösung:

Hier ein Java-Code mit Hilfe der Ninja-Klassen (Ninja steht für *Numerical Intensive Java*). Auf www.matheinfo.de finden Sie Links zum Download dieser frei zur Verfügung stehenden Klassen für die Arbeit mit Vektoren, Matrizen in Java. Auch die *BLAS*, die *Basic Linear Algebra Subroutines*, ist in *Ninja* enthalten.

Hier der Java-Code:

```
import com.ibm.math.array.*;    // import the Array package

public class backsub {
    private static void backSubstitution(doubleArray2D A, doubleArray1D b)
    {
        // Rücksubstitution für eine gegebene obere Dreiecksmatrix
        // und eine rechte Seite. Das Ergebnis steht anschließend auf der
        // übergebenen rechten Seite!
        int i, j, n = b.size();
        double summe;

        for (i=n-1; i>=0; i--) {
            summe = b.get(i);
            for (j=i+1; j<n; j++) {
                summe -= A.get(i, j)*b.get(j);
            }
            b.set(i, summe/A.get(i, i));
        }
    }

    public static void main(String[] args) {
        int n = 3;                // problem size
        doubleArray1D b = new doubleArray1D(n);
        doubleArray1D x = new doubleArray1D(n);
        doubleArray2D A = new doubleArray2D(n, n);

        initArrays(A, b);

        // b auf x kopieren, damit später die Probe durchgeführt werden kann
        x.assign(b);

        backSubstitution(A, x);

        System.out.println("Die berechnete Lösung x");
        System.out.println("x[0] = "+x.get(0));
        System.out.println("x[1] = "+x.get(1));
        System.out.println("x[2] = "+x.get(2));

        if (validate(A, x, b))
            System.out.println("Die Probe war erfolgreich!");
    }
}
```

Mit Hilfe der Funktion `initArrays` werden die Koeffizientenmatrix und die rechte Seite besetzt. Achten Sie darauf, dass die Funktion `backSubstitution` den übergebenen Vektor `b` überschreibt!

```
private static void initArrays(doubleArray2D A, doubleArray1D b) {
    b.set(0, 1.0); b.set(1, 1.0); b.set(2, 0.6);
}
```

```

A.set(0,0, 1.0); A.set(0, 1, -1.0); A.set(0, 2, 0.0);
A.set(1,0, 0.0); A.set(1, 1, 5.0); A.set(1, 2, -3.0);
A.set(2,0, 0.0); A.set(2, 1, 0.0); A.set(2, 2, 1.8);
}

```

Für diese Werte, also

$$A = \begin{pmatrix} 1 & -1 & 0 \\ 0 & 5 & -3 \\ 0 & 0 & 1,8 \end{pmatrix}, \mathbf{b} = \begin{pmatrix} 1 \\ 1 \\ 0,6 \end{pmatrix}$$

erhält man die Lösung

$$\mathbf{x} = \begin{pmatrix} 1,4 \\ 0,4 \\ 0,3333 \end{pmatrix}.$$

Mit der folgenden Funktion `validate` wird die Probe durchgeführt:

```

private static boolean validate(doubleArray2D A, doubleArray1D x,
                               doubleArray1D b) {
    // Probe der berechneten Lösung
    boolean ok = true;
    double eps = 1e-09, error;
    int n = x.size();
    doubleArray1D bb = new doubleArray1D(n);

    // Berechne Ax mit Hilfe von dgemv der BLAS!
    Blas.dgemv(Blas.NoTranspose, 1.0, A, x, 0.0, bb);

    for (int i=0; i<n; i++) {
        error = Math.abs(b.get(i)-bb.get(i));
        if (error > eps) {
            ok = false;
        }
    }
    return ok;
}

```

7. Schreiben Sie eine Funktion in der Programmiersprache Ihrer Wahl, die die Gauß-Elimination für eine Matrix `A` und eine rechte Seite `b` durchführt. Testen Sie diese Funktion an selbst gewählten linearen Gleichungssystemen und mit der Funktion aus Aufgabe 6!

Lösung:

Mit der Rückwärtssubstitution aus der Lösung zu Aufgabe 6 und dem folgenden Quellcode können lineare Gleichungssysteme wie

$$\begin{pmatrix} 1 & -1 & 0 \\ -1 & 6 & -3 \\ 0 & -3 & 3,6 \end{pmatrix} \mathbf{x} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

gelöst werden. Die obere Dreiecksmatrix aus der Lösung von Aufgabe 6 ist das Ergebnis der Gauß-Elimination für diese erweiterte Koeffizientenmatrix. Die Besetzung der Matrix muss entsprechend angepasst werden. Ganz wichtig: die Funktion `gausselimination` überschreibt das obere Dreieck; also muss für die Probe die Originalmatrix gesichert werden!

```

public class gauss {

    public static void main(String[] args) {
        int n = 3; // problem size
        doubleArray1D b = new doubleArray1D(n);
    }
}

```

```
doubleArray1D x = new doubleArray1D(n);
doubleArray2D A = new doubleArray2D(n,n);
doubleArray2D Aalt = new doubleArray2D(n,n);

initArrays(A,b);

// b auf x kopieren, damit später die Probe durchgeführt werden kann
x.assign(b);
// Das gleiche für A
Aalt.assign(A);

// Die Gauss-Zerlegung aufrufen
gaussElimination(A,x);

// Rückwärtsauflösung durchführen
backSubstitution(A, x);

System.out.println("Die berechnete Lösung x");
System.out.println("x[0] = "+x.get(0));
System.out.println("x[1] = "+x.get(1));
System.out.println("x[2] = "+x.get(2));

if (validate(Aalt,x,b))
    System.out.println("Die Probe war erfolgreich!");
else
    System.out.println("Die Probe hat nicht gepasst!");
}

/!/
* Gauß-Elimination für eine quadratische Matrix und einer
* gegebenen rechten Seite. Beide Argumente werden überschrieben!
*/
private static void gaussElimination(doubleArray2D A, doubleArray1D b)
{
    int i, j, k, n = b.size();

    for (j=0; j<n; j++)
        for (i=j+1; i<n; i++) {
            for (k=j+1; k<n; k++)
                A.set(i,k, A.get(i,k) - (A.get(i,j)/A.get(j,j))*A.get(j,k));
            b.set(i,b.get(i) - (A.get(i,j)/A.get(j,j))*b.get(j));
        }
}

private static void initArrays(doubleArray2D A, doubleArray1D b) {

    b.set(0, 1.0); b.set(1, 0.0); b.set(2, 0.0);

    A.set(0,0, 1.0); A.set(0, 1, -1.0); A.set(0, 2, 0.0);
    A.set(1,0, -1.0); A.set(1, 1, 6.0); A.set(1, 2, -3.0);
    A.set(2,0, 0.0); A.set(2, 1, -3.0); A.set(2, 2, 3.6);
}
}
```

Die obere Dreiecksmatrix kann man sich auch ausgeben lassen; insgesamt erhält man eine Ausgabe wie:

```
Der erste Spalte nach Gauß
A[0,0] = 1.0
Der zweite Spalte nach Gauß
A[0,1] = -1.0
A[1,1] = 5.0
Der dritte Spalte nach Gauß
A[0,2] = 0.0
A[1,2] = -3.0
A[2,2] = 1.8000000000000003
Die berechnete Lösung x
x[0] = 1.4
x[1] = 0.39999999999999997
x[2] = 0.33333333333333326
Die Probe war erfolgreich!
```

8. Die *Hilbert-Matrix* ist eine quadratische Matrix mit $h_{ij} = \frac{1}{i+j-1}$, $1 \leq i, j \leq n$. Invertieren Sie H_n mit Hilfe der Gauß-Elimination und Rücksubstitution aus Aufgabe 7. Vergleichen Sie Ihre Ergebnisse mit der exakten Inversen H_n^{-1} , die durch

$$h_{ij}^{-1} = (-1)^{i+j}(i+j-1) \binom{n+i-j}{n-j} \binom{n+j-1}{n-i} \binom{i+j-2}{i-1}^2.$$

gegeben ist!

Lösung:

Die Inverse der Hilbertmatrix H_n kann mit Hilfe der Gauss-Elimination in den letzten beiden Aufgaben gelöst werden. Als rechte Seiten verwendet man die n kanonischen Einheitsvektoren; dann erhalten Sie die jeweiligen Spalten der Matrix.

Wenn Sie zur Klasse der letzten Aufgabe noch eine Funktion beifügen, die die Hilbertmatrix besetzt wie

```
private static doubleArray2D hilbertmatrix(int n) {
    doubleArray2D hn = new doubleArray2D(n,n);
    int i,j;

    for (i=0; i<n; i++)
        for (j=0; j<n; j++)
            hn.set(i, j, 1.0/(i+j+1.0));
    return hn;
}
```

dann erhält man beispielsweise für $n = 3$ die erste Spalte

```
Die berechnete erste Spalte der Inversen für n=3:
x[0] = 9.0000000000000043
x[1] = -36.000000000000234
x[2] = 30.000000000000224
Die Probe war erfolgreich!
```

Die Probe-Funktion ist offensichtlich zufrieden.

Das Ganze geht gut, bis Sie in den Bereich von $n = 11$ und $n = 12$ kommen. Das liegt daran, dass die Hilbertmatrix ein Beispiel einer schlecht-konditionierten Matrix ist. Die Rundungsfehler, die bei der Gauß-Elimination unweigerlich auftreten, werden mit einem immer größeren Faktor, der Kondition der Matrix, multipliziert. Ist diese Kondition größer wie die

Maschinengenauigkeit, kann das Ergebnis nicht mehr sinnvoll sein. Die Kondition der Hilbertmatrix steigt exponentiell an; bei $n = 11$ ergibt sich auf Cygwin und JDK1.4 noch die Ausgabe

```
Die berechnete erste Spalte der Inversen für n=11:
x[0] = 120.89701771433465
x[1] = -7249.145595417011
x[2] = 141286.79323580535
x[3] = -1318138.0520180147
x[4] = 6917893.199034784
x[5] = -2.2130997809192467E7
x[6] = 4.477813681939939E7
x[7] = -5.755993554357505E7
x[8] = 4.5560038581912935E7
x[9] = -2.024568544551592E7
x[10] = 3864540.6361000347
Die Probe war erfolgreich!
```

Dabei vergleicht die Probe-Funktion das Ergebnis von Ax mit \mathbf{b} mit einer Genauigkeit von 10^{-9} . Bei $n = 12$ ergibt sich auf meinem Rechner dann

```
Die berechnete erste Spalte der Inversen für n=12:
x[0] = 141.5584489814937
x[1] = -9986.581068394009
x[2] = 230509.64239294577
x[3] = -2570134.73345685
x[4] = 1.6325579265833031E7
x[5] = -6.4343784039349094E7
x[6] = 1.6454066371450692E8
x[7] = -2.777840891763152E8
x[8] = 3.0733657354688287E8
x[9] = -2.1432348592910305E8
x[10] = 8.544011827110982E7
x[11] = -1.4842115901764408E7
Die Probe hat nicht gepasst!
```

Für noch größere Werte von n werden die Ergebnisse immer unsinniger ...

9. In vielen Anwendungen tauchen lineare Gleichungssysteme mit *tridiagonaler* Koeffizientenmatrix auf. Für $n = 5$ hat die tridiagonale Koeffizientenmatrix die Form

$$A = \begin{pmatrix} a_1 & b_1 & 0 & 0 & 0 \\ c_1 & a_2 & b_2 & 0 & 0 \\ 0 & c_2 & a_3 & b_3 & 0 \\ 0 & 0 & c_3 & a_4 & b_4 \\ 0 & 0 & 0 & c_4 & a_5 \end{pmatrix}.$$

Wenn die Gauß-Elimination für eine solche Koeffizientenmatrix ohne Zeilenvertauschung durchführbar ist, dann muss eine LU -Zerlegung existieren, bei der L eine bidiagonale untere und U eine bidiagonale obere Dreiecksmatrix ist. Stellen Sie einen Algorithmus auf, der für eine gegebene $n \times n$ -Tridiagonalmatrix diese LU -Zerlegung berechnet wenn keine Zeilenvertauschungen nötig sind. Implementieren Sie den Algorithmus in der Programmiersprache Ihrer Wahl und testen Sie die Funktion an selbst gewählten Beispielen!

Lösung:

Die LU -Zerlegung einer Tridiagonalmatrix muss aus zwei Matrizen bestehen, die jeweils eine Haupt- und eine Nebendiagonale besitzen. Das kann man leicht verifizieren; nehmen Sie an, es wäre falsch, dann sieht man schnell, dass das Produkt LU keine Tridiagonalmatrix ist.

Es ist also

$$\begin{pmatrix} a_1 & b_1 & 0 & 0 & 0 \\ c_1 & a_2 & b_2 & 0 & 0 \\ 0 & c_2 & a_3 & b_3 & 0 \\ 0 & 0 & c_3 & a_4 & b_4 \\ 0 & 0 & 0 & c_4 & a_5 \end{pmatrix} = \begin{pmatrix} 1 & & & & \\ l_1 & 1 & & & \\ & l_2 & 1 & & \\ & & l_3 & 1 & \\ & & & l_4 & 1 \end{pmatrix} \begin{pmatrix} m_1 & r_1 & & & \\ & m_2 & r_2 & & \\ & & m_3 & r_3 & \\ & & & m_4 & r_4 \\ & & & & m_5 \end{pmatrix}.$$

Die Unbekannten Elemente l_i, m_i und r_i sind dann durch Koeffizientenvergleich bestimmbar:

$$\begin{aligned} a_1 &= m_1, & b_1 &= r_1 \\ c_1 &= l_1 m_1, & a_2 &= l_1 r_1 + m_2, & b_2 &= r_2 \\ c_2 &= l_2 m_2, & a_3 &= l_2 r_2 + m_3, & b_3 &= r_3 \\ c_3 &= l_3 m_3, & a_4 &= l_3 r_3 + m_4, & b_4 &= r_4 \\ c_4 &= l_4 m_4, & a_5 &= l_4 r_4 + m_5. \end{aligned}$$

Also bestimmen sich die Unbekannten immer in der Reihenfolge $m_1, r_1, l_1, m_2, r_2, l_2$ und so weiter. Dabei ist einfach $r_i = b_i$. Dann kann das in einer Schleife formuliert werden:

$$\begin{aligned} m_1 &= a_1, \\ l_1 &= \frac{c_1}{m_1}, m_2 = a_2 - l_1 b_1, \\ l_2 &= \frac{c_2}{m_2}, m_3 = a_3 - l_2 b_2, \dots \end{aligned}$$

Auch die Auflösung der beiden bidiagonalen Gleichungssysteme kann einfach formuliert werden. Die Vorwärtsauflösung für das Gleichungssystem $Ly = \mathbf{d}$ ist gegeben durch

$$y_1 = d_1, y_i = d_i - l_{i-1} y_{i-1}, i = 2, \dots, n.$$

Zur Berechnung von y_2 benötigt man also nur die erste Komponente der rechten Seite, d_1 und l_1 . Das bedeutet, dass Sie die Vorwärtsauflösung in die Schleife für die Berechnung der beiden Faktoren L und U integrieren können.

Anschließend wird noch mit einer Rückwärtsauflösung das Gleichungssystem $Ux = \mathbf{y}$ gelöst:

$$x_n = \frac{y_n}{m_n}, x_i = \frac{1}{m_i} (y_i - r_i x_{i+1}), i = n-1, \dots, 1.$$

```

/*!
 * LU-Zerlegung und Auflösung für ein tridiagonales Gleichungssystem.
 * Die Matrix ist durch die Vektoren a, b und c gegeben.
 *
 * Der Vektor a wird überschrieben!
 */
private static void LUandSolve(doubleArray1D c, doubleArray1D a,
                               doubleArray1D b, doubleArray1D x,
                               doubleArray1D d)
{
    int j, n = d.size();
    doubleArray1D l = new doubleArray1D(n);

    x.set(0, d.get(0));
    for (j=1; j<n; j++) {
        l.set(j-1, c.get(j-1)/a.get(j-1));
        a.set(j, a.get(j)-l.get(j-1)*b.get(j-1));
        x.set(j, d.get(j)-l.get(j-1)*x.get(j-1));
    }
}

```

```

}
// Es wird nicht überprüft, ob die auftretenden Diagonalelemente
// ungleich Null sind!#
x.set(n-1, x.get(n-1)/a.get(n-1));
for (j=n-2; j>=0; j--) {
    x.set(j, (x.get(j)-b.get(j)*x.get(j+1))/a.get(j));
}
}

```

Für das tridiagonale Gleichungssystem

$$\begin{pmatrix} 1 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 2 & 1 \end{pmatrix} \mathbf{x} = \begin{pmatrix} 2 \\ 4 \\ 3 \end{pmatrix}$$

ist die exakte Lösung gegeben durch

$$\mathbf{x} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}.$$

Dies wird auch durch den Code berechnet:

```

Die berechnete Lösung x
x[0] = 1.0
x[1] = 1.0
x[2] = 1.0

```

10. Implementieren Sie die *LU*-Zerlegung mit Scheinskalierung und Spaltenpivotwahl in der Programmiersprache Ihrer Wahl. Lösen Sie damit das lineare Gleichungssystem mit der 20×20 -Matrix A mit

$$a_{ij} = \begin{cases} 2, & i + j = 21, \\ \cos(0,1 \cdot i + 0,05 \cdot j + 0,04 \cdot i \cdot j), & \text{sonst} \end{cases}$$

und der rechten Seite $b_i = 1, 1 \leq i \leq 20$.

Lösung:

Für diese Aufgabe müssen *drei* Funktionen implementiert werden. Neben der *LU*-Zerlegung eine Vorwärtssubstitution für das Gleichungssystem $L\mathbf{y} = P\mathbf{b}$ mit der Permutationsmatrix P und einer Rückwärtssubstitution für $U\mathbf{x} = \mathbf{y}$.

Die Rückwärtssubstitution kann aus den bisherigen Aufgaben übernommen werden. Die Vorwärtssubstitution ist analog zu programmieren. Die Permutationsmatrix wird als Vektor gespeichert. Ist $p_i = k$, dann müssen die Zeilen i und k vertauscht werden. Insgesamt erhält man dann folgenden Quellcode für diese beiden Auflösungen:

```

private static void forwardSubstitution(doubleArray2D A, doubleArray1D b,
                                       intArray1D p)
{
    // Vorwärtssubstitution für eine gegebene untere Dreiecksmatrix
    // und eine rechte Seite. Das Ergebnis steht anschließend auf der
    // übergebenen rechten Seite! Im Feld p steht eine
    // eventuell durchgeführte Zeilenvertauschung.
    //
    // Die Diagonale von A wird implizit auf 1 gesetzt!
    int i, j, n = b.size();
    double h, c;

```

```

// Zeilenvertauschung in der rechten Seite
for (i=0; i<n-1; i++) {
    if (p.get(i) != i) {
        h = b.get(i);
        b.set(i, b.get(p.get(i)));
        b.set(p.get(i), h);
    }
}

for (i=1; i<n; i++) {
    c = b.get(i);
    for (j=0; j<i; j++)
        c -= A.get(i, j)*b.get(j);
    b.set(i, c);
}

}

private static void backSubstitution(doubleArray2D A, doubleArray1D b)
{
    // Rücksubstitution für eine gegebene obere Dreiecksmatrix
    // und eine rechte Seite. Das Ergebnis steht anschließend auf der
    // übergebenen rechten Seite!

    int i, j, n = b.size();
    double summe;

    for (i=n-1; i>=0; i--) {
        summe = b.get(i);
        for (j=i+1; j<n; j++) {
            summe -= A.get(i, j)*b.get(j);
        }
        b.set(i, summe/A.get(i, i));
    }
}
}

```

Wie bisher wird auch hier die rechte Seite überschrieben; also müssen Sie diesen Vektor vorher sichern, wenn Sie nochmal die Probe durchführen möchten.

Die *LU*-Zerlegung hat die Matrix *A* und die Permutationsmatrix *P* als Argument. Auch hier steht nach dem Aufruf der Funktion auf *A* das Ergebnis der *LU*-Zerlegung. Die Diagonale von *L* wird implizit auf 1 gesetzt.

```

/*!
 * LU für eine quadratische Matrix und einer
 * gegebenen rechten Seite. Beide Argumente werden überschrieben!
 *
 * Die Zerlegung wird mit Spaltenpivotsuche und Scheinskalierung
 * durchgeführt. Die Zeilenvertauschungen werden auf dem Integer-Feld
 * p abgelegt und werden der Vorwärts-Auflösung übergeben.
 */
private static void LUzerlegung(doubleArray2D A, intArray1D p)
{
    int i, j, k, n = p.size();
    double q, s, h, max, summe;

    // Den Permutationsvektor initialisieren
    for (i=0; i<n; i++) p.set(i, i);
}

```

```

for (k=0; k<n-1; k++) {
    // Die Spaltensumme berechnen
    max = 0.0;
    for (i=k; i<n; i++) {
        summe = 0.0;
        for (j=k; j<n; j++) {
            summe += Math.abs(A.get(i, j));
        }

        // Spaltenpivotsuche mit Scheinskalierung
        q = Math.abs(A.get(i, k))/summe;
        if (q > max) {
            max = q;
            p.set(k, i);
        }
    }
    if (max == 0) {
        // Throw exeption!
        return;
    }

    // Vertauschen der Zeilen, falls erforderlich
    if (p.get(k) != k) {
        for (j=0; j<n; j++) {
            h = A.get(k, j);
            A.set(k, j, A.get(p.get(k), j));
            A.set(p.get(k), j, h);
        }
    }

    // Elimination
    for (i=k+1; i<n; i++) {
        A.set(i, k, A.get(i, k)/A.get(k, k));
        for (j=k+1; j<n; j++) {
            A.set(i, j, A.get(i, j)-A.get(i, k)*A.get(k, j));
        }
    }
}
}
}

```

Wenn Sie diese Funktionen auf die Matrix aus Aufgabe 7 anwenden, dann wird keine Zeilenvertauschung durchgeführt.

Im folgenden Code-Fragment sehen Sie die Besetzung der vorgegebenen Matrix und das Hauptprogramm:

```

private static void initArrays(doubleArray2D A, doubleArray1D b) {
    int i, j, n=b.size();

    b.assign(1.0);
    for (i=0; i<n; i++)
        for (j=0; j<n; j++) {
            if (i+j==21)
                A.set(i, j, 2.0);
            else
                A.set(i, j,
                    Math.cos(0.1*(double)i+0.05*(double)j+
                        0.04*(double)i*(double)j));
        }
}

```

```

}

public static void main(String[] args) {
    int i, n = 20;           // problem size
    doubleArray1D b = new doubleArray1D(n);
    doubleArray1D x = new doubleArray1D(n);
    doubleArray2D A = new doubleArray2D(n,n);
    doubleArray2D Aalt = new doubleArray2D(n,n);
    intArray1D p = new intArray1D(n);

    initArrays(A,b);

    // b auf x kopieren, damit später die Probe durchgeführt werden kann
    x.assign(b);
    // Das gleiche für A
    Aalt.assign(A);

    // Die Gauss-Zerlegung aufrufen
    LUzerlegung(A, p);

    // Ausgabe des Permutationsvektors
    System.out.println("Der Permutationsvektor der LU-Zerlegung");
    for (i=0; i<n; i++)
        System.out.println("p[" + i + "] = "+p.get(i));

    // Die Vorwärtssubstitution für Ly = Pb
    forwardSubstitution(A, x, p);

    // Rückwärtsauflösung durchführen
    backSubstitution(A, x);

    System.out.println("\nDie berechnete Lösung x");
    for (i=0; i<n; i++)
        System.out.println("x[" + i + "] = "+x.get(i));

    if (validate(Aalt,x,b))
        System.out.println("\nDie Probe war erfolgreich!");
    else
        System.out.println("\nDie Probe hat nicht gepasst!");
}

```

Damit erhalten Sie die folgenden Ausgaben der Permutationsmatrix und der erhaltenen Lösung; die Probefunktion wird dabei erfolgreich durchgeführt:

```

Der Permutationsvektor der LU-Zerlegung
p[0] = 1
p[1] = 16
p[2] = 19
p[3] = 18
p[4] = 17
p[5] = 15
p[6] = 14
p[7] = 13
p[8] = 12
p[9] = 11
p[10] = 10
p[11] = 11
p[12] = 12

```

```

p[13] = 13
p[14] = 14
p[15] = 15
p[16] = 17
p[17] = 17
p[18] = 18
p[19] = 19

Die berechnete Lösung x
x[0] = 2.536362184292622
x[1] = -1.7691770080297828
x[2] = 0.0871597654482613
x[3] = 0.06404845762858202
x[4] = 0.045846727606265444
x[5] = 0.02852318961741817
x[6] = 0.012378367951613147
x[7] = 6.161281447016294E-4
x[8] = -0.004428505321647933
x[9] = -0.003952624773600349
x[10] = -0.0012861512578206155
x[11] = 7.37151462158858E-4
x[12] = 0.0010712592221248283
x[13] = 3.816974115166879E-4
x[14] = -6.382871244859799E-5
x[15] = 2.3828117914932964E-4
x[16] = 8.403897600632952E-4
x[17] = 0.0011846824049227677
x[18] = 0.0010839925860701616
x[19] = 6.192427701304958E-4

Die Probe war erfolgreich!

```

11. Invertieren Sie mit der LU -Zerlegung mit Scheinskalierung und Spaltenpivotwahl die Hilbertmatrix für verschiedene Werte von n . Vergleichen Sie Ihre Ergebnisse mit den Ergebnissen aus Aufgabe 8!

Lösung:

Hier wird jetzt die LU -Zerlegung aus der Aufgabe 10 auf die Hilbertmatrix angewandt. Wieder ist für $n = 3$ die Invertierung (immer am Beispiel der ersten Spalte der inversen Hilbertmatrix durchgeführt) ohne Zeilenvertauschung möglich:

```

Der Permutationsvektor der LU-Zerlegung
p[0] = 0
p[1] = 1
p[2] = 2

Die berechnete Lösung x
x[0] = 9.0000000000000043
x[1] = -36.000000000000234
x[2] = 30.000000000000224

Die Probe war erfolgreich!

```

Auch für $n = 11$ ist alles noch in Ordnung:

```

Der Permutationsvektor der LU-Zerlegung
p[0] = 0
p[1] = 1
p[2] = 2

```

```

p[3] = 3
p[4] = 4
p[5] = 5
p[6] = 6
p[7] = 7
p[8] = 8
p[9] = 9
p[10] = 10

Die berechnete Lösung x
x[0] = 120.89701771573164
x[1] = -7249.14559544495
x[2] = 141286.79323595553
x[3] = -1318138.0520184098
x[4] = 6917893.199035297
x[5] = -2.2130997809192814E7
x[6] = 4.477813681939951E7
x[7] = -5.755993554357506E7
x[8] = 4.556003858191293E7
x[9] = -2.024568544551592E7
x[10] = 3864540.6361000347

Die Probe war erfolgreich!

```

Es wird keine Zeilenvertauschung durchgeführt; die Ergebnisse sind identisch! Auch für größere Werte verändert sich nichts mehr.

12. Beweisen Sie durch Nachrechnen, dass alle für 2×2 -Determinanten aufgestellten Regeln auch für den Fall 3×3 gelten!

Lösung:

Die Regeln können durch Nachrechnen nachgewiesen werden.

Eine Determinante wird mit einem Faktor λ multipliziert, indem eine beliebige Zeile oder Spalte der Determinante mit λ multipliziert wird. Sinnvoller Weise multiplizieren wir die Zeile, nach der entwickelt wird, beispielsweise die erste Zeile:

$$\begin{vmatrix} \lambda a_{11} & \lambda a_{12} & \lambda a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = \lambda a_{11} \begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix} - \lambda a_{12} \begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} + \lambda a_{13} \begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix} = \lambda \det(A).$$

Werden beispielsweise zwei Zeilen vertauscht, dann ändert die Determinante ihr Vorzeichen:

$$\begin{vmatrix} a_{21} & a_{22} & a_{23} \\ a_{11} & a_{12} & a_{13} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = -a_{11} \begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix} + a_{12} \begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} - a_{13} \begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix} = -\det(A).$$

Auch alle weiteren Regeln ergeben sich auf diese Weise.

13. Berechnen Sie $D_1 = \begin{vmatrix} 1 & -2 & 7 \\ 0 & 3 & 2 \\ 5 & -1 & 4 \end{vmatrix}$, $D_2 = \begin{vmatrix} 1 & 2 & -3 & 5 \\ 0 & 12 & 0 & 1 \\ 1 & 0 & -1 & 2 \\ -1 & 2 & 2 & 1 \end{vmatrix}$ und $D_3 = \begin{vmatrix} 1 & 5 & 5 & 0 \\ -2 & 1 & -2 & 3 \\ 0 & 1 & 1 & 0 \\ 1 & 2 & 4 & 1 \end{vmatrix}$.

Lösung:

D_1 kann mit der Regel von Sarrus oder mit dem Entwicklungssatz berechnet werden. Es ist $D_1 = 1 \cdot 3 \cdot 4 + (-2) \cdot 2 \cdot 5 + 7 \cdot 0 \cdot (-1) - 7 \cdot 3 \cdot 5 - 1 \cdot 2 \cdot (-1) - (-2) \cdot 0 \cdot 4 = 12 - 20 - 105 + 2 = -111$.

D_2 ist (Entwicklung nach der ersten Spalte)

$$D_2 = \begin{vmatrix} 12 & 0 & 1 \\ 0 & -1 & 2 \\ 2 & 2 & 1 \end{vmatrix} + \begin{vmatrix} 2 & -3 & 5 \\ 12 & 0 & 1 \\ 2 & 2 & 1 \end{vmatrix} + \begin{vmatrix} 2 & -3 & 5 \\ 12 & 0 & 1 \\ 0 & -1 & 2 \end{vmatrix} = 102.$$

D_3 ist (Entwicklung nach der vierten Zeile)

$$D_3 = 3 \cdot \begin{vmatrix} 1 & 5 & 5 \\ 0 & 1 & 1 \\ 1 & 2 & 4 \end{vmatrix} = 3 \cdot 2 = 6.$$

14. Berechnen Sie die Determinante von A^{-1} für $A = \begin{pmatrix} 1 & 0 & 3 & 4 \\ -2 & 1 & 0 & 3 \\ 1 & 4 & 1 & 5 \\ 0 & 2 & 2 & 0 \end{pmatrix}$.

Lösung:

Um die Determinante einer Inversen zu berechnen wird die $\det(A)$ berechnet; $\det(A^{-1}) = \frac{1}{\det(A)} = \frac{1}{142}$.

Bei der Berechnung der Determinante von A kann vorteilhaft nach der vierten Zeile entwickelt werden, es ist

$$\det(A) = 2 \cdot \begin{vmatrix} 1 & 3 & 4 \\ -2 & 0 & 3 \\ 1 & 1 & 5 \end{vmatrix} - 2 \cdot \begin{vmatrix} 1 & 0 & 4 \\ -2 & 1 & 3 \\ 1 & 4 & 5 \end{vmatrix} = 2 \cdot 28 - 2 \cdot (-43) = 142.$$

Möglich wäre natürlich auch, die Inverse mit Gauß-Jordan und dann die Determinante zu berechnen.

15. Berechnen Sie Eigenwerte und Eigenvektoren der Matrizen $A = \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}$ und $B = \begin{pmatrix} 4 & 0 & 1 \\ -2 & 1 & 0 \\ -2 & 0 & 1 \end{pmatrix}$.

Lösung:

Die Eigenwerte von A sind gegeben durch die Nullstellen des charakteristischen Polynoms $(1 - \lambda)^2 - 4 = 1 - 2\lambda + \lambda^2 - 4 = \lambda^2 - 2\lambda - 3$. Als Nullstellen erhält man $\lambda_1 = -1$ und $\lambda_2 = 3$. Die Eigenvektoren sind dann als Lösung der zugehörigen Gleichungssysteme gegeben durch $\mathbf{x}_1 = (-1, 1)^T$, $\mathbf{x}_2 = (1, 1)^T$.

Das charakteristische Polynom von B ist gegeben durch $(1 - \lambda)((4 - \lambda)(1 - \lambda)^2 + 2)$; möglich ist beispielsweise die Entwicklung nach der ersten Zeile. Dann sind die Nullstellen gegeben durch $\lambda_1 = 1$, $\lambda_2 = 2$, $\lambda_3 = 3$. Die zugehörigen Eigenvektoren sind $\mathbf{x}_1 = (0, 1, 0)^T$, $\mathbf{x}_2 = (-1, 2, 2)^T$, $\mathbf{x}_3 = (-1, 1, 1)^T$.

16. Bestimmen Sie die Eigenwerte von A^{25} für $A = \begin{pmatrix} -1 & -2 & -2 \\ 1 & 2 & 1 \\ -1 & -1 & 0 \end{pmatrix}$.

Lösung:

Die Eigenwerte von A^{25} sind die 25-te Potenz der Eigenwerte von A . Es gilt $\lambda_1 = -1$, $\lambda_2 = 1$, $\lambda_3 = 1$.

17. Beweisen Sie, dass eine 2×2 -Matrix A genau dann zu einer Diagonalmatrix ähnlich ist, wenn $(a_{11} - a_{22})^2 + 4a_{12}a_{21} > 0$ erfüllt ist.

Lösung:

Die Matrix A muss zwei reelle Eigenwerte haben, dann kann die Konstruktion wie für die Beispielmatrix im Buch durchgeführt werden. Mit der Formel von Vieta folgt nach Aufstellen

des charakteristischen Polynoms die Behauptung:

$$\begin{aligned} \det(A - \lambda I) &= \begin{vmatrix} a_{11} - \lambda & a_{12} \\ a_{21} & a_{22} - \lambda \end{vmatrix} \\ &= (a_{11} - \lambda)(a_{22} - \lambda) - a_{12}a_{21} = (a_{11}a_{22} - a_{12}a_{21}) - \lambda(a_{11} + a_{22}) + \lambda^2. \end{aligned}$$

Die Diskriminante ist dann größer Null, wenn

$$D = b^2 - 4ac = (a_{11} + a_{22})^2 - 4(a_{11}a_{22} - a_{12}a_{21}) = (a_{11} - a_{22})^2 + 4a_{12}a_{21} > 0$$

erfüllt ist.

18. Berechnen sie auf der Basis der LU -Zerlegung der Hilbertmatrix die Determinante $\det(H_n)$ und vergleichen Sie die Werte mit dem exakten Wert

$$\det(H_n) = \frac{(\prod_{i=2}^{n-1} i!)^4}{\prod_{i=2}^{2n-1} i!} \approx 2^{-2n^2}.$$

Lösung:

Die Determinante ergibt sich als Produkt der Diagonalelemente der Matrix U ; denn es ist $\det(A) = \det(LU) = \det(L)\det(U) = \det(U)$.

Man könnte die Berechnung der Determinanten direkt in die LU -Zerlegung integrieren; hier ist eine Lösung, die einfach das Ergebnis der bereits verwendeten Funktion für die LU -Zerlegung aufgreift. Wichtig ist, dass Sie daran denken, dass sich bei jeder bei der Zerlegung durchgeführten Zeilenvertauschung sich das Vorzeichen der Determinante umkehrt!

```

/*
 * Berechnung der Determinanten einer Matrix auf der Basis
 * einer LU-Zerlegung
 */
private static double LUDeterminante(doubleArray2D A, intArray1D p) {
    int i, n = p.size();
    double det;

    det = 1.0d;

    // Für jede Zeilenvertauschung bei der LU-Zerlegung
    // bändert die Determinante einmal das Vorzeichen.
    for (i = 0; i < n; i++) {
        det *= A.get(i, i);
        if (p.get(i) != i) det = -det;
    }
    return det;
}

```

Für kleine n ist die Schätzung der Determinanten nicht sehr gut, aber beispielsweise für $n = 5$ erhält man die folgende Ausgabe, wenn der Vergleich auf Gleichheit mit einer Genauigkeit von 10^{-9} durchgeführt wird:

```

Die berechnete Determinante für n=5 ist 3.749295132517303E-12
Die Schätzung für n=5 ist 8.881784197001244E-16
Die Probe war erfolgreich!

```

Beachten Sie, dass die Determinanten der Hilbertmatrizen exponentiell gegen Null konvergieren!

Hier nochmals das Hauptprogramm für diese Aufgabe:

```
public static void main(String[] args) {
    int i, n = 5;
    double det, estimate, eps = 1e-09d;
    doubleArray2D A = new doubleArray2D(n,n);
    intArray1D p = new intArray1D(n);

    A = hilbertmatrix(n);
    LUzerlegung(A, p);
    det = LUDeterminante(A, p);

    System.out.println("Die berechnete Determinante für n="
        + n + " ist " + det);

    // Die Schätzung für die Determinante der Hilbertmatrix berechnen
    estimate = Math.exp(-2.0*n*n*Math.log(2.0));
    System.out.println("Die Schätzung für n=" + n + " ist " + estimate);

    if (Math.abs(estimate-det) < eps)
        System.out.println("Die Probe war erfolgreich!");
    else
        System.out.println("Die Probe hat nicht gepasst!");
}
```

Kapitel 6

Zahlentheorie

Verständnisfragen

Sachfragen

1. Erläutern Sie die ganzzahlige Division!
2. Was versteht man unter $a \bmod b$ für $a, b \in \mathbb{Z}$?
3. Erläutern sie $a|b$ für $a, b \in \mathbb{Z}$!
4. Was ist eine Primzahl?
5. Was ist eine zusammengesetzte Zahl?
6. Wie viele Primzahlen gibt es?
7. Wie lautet der Fundamentalsatz der Zahlentheorie?
8. Kennen Sie einen einfachen Primzahltest?
9. Erläutern Sie das Sieb des Erathostenes!
10. Erläutern Sie die Funktion $\pi(n)$!
11. Wie lautet die Definition des größten gemeinsamen Teilers?
12. Welche Berechnungsmöglichkeiten für den größten gemeinsamen Teiler kennen Sie?
13. Was ist eine diophantische Gleichung?
14. Wie verläuft der Euklidische Algorithmus?
15. Wie verläuft der verallgemeinerte Euklidische Algorithmus?
16. Wie kann der größte gemeinsame Teiler rekursiv berechnet werden?
17. Was sind zwei ganze Zahlen kongruent zum Modul m ?
18. Erläutern Sie die modulare Addition und Multiplikation!
19. Erläutern Sie den Zusammenhang zwischen Rest- und Äquivalenzklassen!
20. Wie lautet die Kürzungsregel für die modulare Division?

21. Was ist eine modulare Inverse? Wie kann sie bestimmt werden?
22. Beschreiben Sie den Graphen der Funktion $n^2 \bmod m$!
23. Erläutern Sie die modulare Wurzel und den modularen Logarithmus!
24. Wie lautet der kleine Satz von Fermat?
25. Wie ist die Euler'sche Funktion definiert?
26. Wie kann die Euler'sche Funktion berechnet werden?
27. Wie lautet der Satz von Euler?
28. Erläutern Sie die Aufgabenstellung der Kryptographie!
29. Erläutern Sie die Verschiebe-Chiffre!
30. Erläutern Sie die Tausch-Chiffre!
31. Was muss der Schlüssel einer Tausch-Chiffre erfüllen?
32. Erläutern Sie das Prinzip eines Public-Key-Kryptosystems!
33. Erläutern Sie das Kryptosystem nach Hellman, Merkle, Diffie!
34. Erläutern Sie das RSA-System!
35. Erläutern Sie die digitale Unterschrift mit Hilfe des RSA-Systems!
36. Erläutern Sie das ElGamal-System!

Methodenfragen

1. Den Quotienten und den Rest bei der ganzzahligen Division angeben können.
2. Teiler einer Zahl angeben können.
3. Die Regeln für die Teilbarkeit anwenden können.
4. Primfaktoren einer Zahl finden können.
5. Die Primfaktorzerlegung für kleine Zahlen konstruieren können.
6. Das Sieb des Erathostenes mit Papier und Bleistift durchführen können.
7. Das Sieb des Erathostenes programmieren können.
8. Den Euklidischen Algorithmus durchführen können.
9. Den verallgemeinerten Euklidischen Algorithmus durchführen können.
10. Einfache diophantische Gleichungen lösen können.
11. Den größten gemeinsamen Teiler zweier Zahlen rekursiv berechnen können.
12. Den Euklidischen Algorithmus und seine Verallgemeinerung programmieren können.
13. Die Funktion $\pi(n)$ mit Hilfe eines Siebs berechnen können.
14. Kongruenzen aufstellen und nachrechnen können.
15. Modulare Arithmetik ausführen können; insbesondere die Kürzungsregel der modularen Division.

16. Die modulare Inverse berechnen können und zur Gleichungslösung verwenden können.
17. Modulare Potenzrechnung, Wurzeln und Logarithmen bestimmen können.
18. Einen Primzahltest auf der Basis des kleinen Satzes von Fermat durchführen können.
19. Die Euler'sche Funktion berechnen können.
20. Eine Verschiebe-Chiffre anwenden und programmieren können.
21. Entscheiden können, ob ein gegebenes Zahlenpaar als Schlüssel einer Tausch-Chiffre verwendet werden kann.
22. Eine Tausch-Chiffre anwenden und programmieren können.
23. Das Krypto-System nach Hellman, Merkle, Diffie anwenden und programmieren können.
24. Das RSA-System anwenden und programmieren können.
25. Das ElGamal-System anwenden und programmieren können.

Übungsaufgaben

1. Geben Sie den Quotienten q und den Rest r für die folgenden Zahlenpaare an: $a = 7, b = 5$; $a = 32, b = 11$, $a = -32, b = 11$, $a = 625, b = 24$ und $a = 774, b = 354$.

Lösung:

$$7 = 1 \cdot 5 + 2, \text{ also } q = 1, r = 2.$$

$$32 = 2 \cdot 11 + 10, \text{ also } q = 2, r = 10.$$

$$-32 = (-3) \cdot 11 + 1, \text{ also } q = -3, r = 1.$$

$$625 = 26 \cdot 24 + 1, \text{ also } q = 26, r = 1.$$

$$774 = 2 \cdot 354 + 76, \text{ also } q = 2, r = 76.$$

2. Bestimmen Sie die Primzahlfaktorisation von 1 001, 7 777 und 10 121 804.

Lösung:

$$7\,777 = 7 \cdot 11 \cdot 101, \quad 10\,121\,804 = 2^2 \cdot 7 \cdot 11 \cdot 59 \cdot 557.$$

3. Alle zusammengesetzten Zahlen bis 10 000 enthalten in ihrer Primzahlfaktorisation eine Primzahl, die kleiner als 100 ist. Ist diese Aussage richtig?

Lösung:

Die Aussage folgt aus Satz 5.8. Ist eine natürliche Zahl kein Vielfaches einer Primzahl p mit $p^2 \leq n$, dann ist n eine Primzahl. 100 ist die Wurzel aus 10 000.

4. Führen Sie das Sieb des Erathostenes auf Papier und mit Hilfe des Computers für die Zahlen bis 180 durch!

Lösung:

Die Primzahlen kleiner als 180 sind gegeben durch

$$2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, \\ 101, 103, 107, 109, 113, 127, 131, 137, 139, 149, 151, 157, 163, 167, 173, 179.$$

Sie müssen alle Vielfache bis einschließlich 13 untersuchen, da $13^2 = 169$ und $17^2 = 289$.

Hier ein C++-Source-Code, der eine Liste der Primzahlen bis zu einer vorgegebenen Zahl ausgibt:

```

void erathostenes(unsigned long N, unsigned long &phi, bool *prim);

void main(void)
{
    unsigned long n, phi;

    cout << "Geben Sie N an:" << endl;
    cin >> n;

    bool *prim = new bool[n];

    erathostenes(n, phi, prim);

    // Ausgeben der Primzahlen
    cout << endl;
    for(int i=1; i < n; i++) {
        if(prim[i]) cout << i+1 << " ";
    }
    cout << endl;
    cout << " phi(" << n << ") = " << phi << endl;
}

void erathostenes(unsigned long N, unsigned long &phi, bool *prim)
{
    unsigned long wurzel = (unsigned long)sqrt(N);
    int i, j;

    // Initialisieren des Feldes
    for(i=0; i < N; i++) {
        prim[i] = true;
    }
    // Feststellen der Nichtprimzahlen
    // Auf Grund der Aussage, die die Grundlage des Siebs bildet,
    // muss nur bis wurzel(N) überprüft werden.
    //
    // Die Vielfachen werden gestrichen, übrig bleiben die
    // Primzahlen.
    // Auf count steht nach Durchlauf des Siebs die Anzahl der
    // Primzahlen kleiner oder gleich size; 1 ist keine Primzahl,
    // also wird sofort 1 subtrahiert.
    phi = N-1;
    for(i=2; i <= wurzel; i++) {
        for(j=2*i; j <= N; j+=i) {
            if (prim[j-1]) {
                prim[j-1] = false;
                phi--;
            }
        }
    }
}

```

Und hier eine Java-Klasse mit der gleichen Funktionalität:

```

public class Erathostenes {
    private static long phi;
    private static boolean[] prim;

    // Das Sieb als Konstruktor
    public Erathostenes (long N)

```

```
{
    int i, j;
    prim = new boolean[(int)N];
    long wurzel;

    // Initialisieren des Feldes
    for (i=0; i<N; i++) {
        prim[i] = true;
    }

    wurzel = (long)Math.sqrt(N);
    phi = N-1;
    for (i=2; i <= wurzel; i++) {
        for (j=2*i; j <= N; j+=i) {
            if (prim[j-1]) {
                prim[j-1] = false;
                phi--;
            }
        }
    }
} // Konstruktor

public static void main (String[] args) throws IOException {
    long N = readInt();
    Erathostenes sieb = new Erathostenes(N);
    System.out.println();
    for (int i=1; i<N; i++) {
        System.out.print(i+1+" ");
    }
    System.out.println();

    System.out.println("phi(" + N + ") = " + phi);
} // main

public static int readInt() throws IOException {
    int n;
    BufferedReader in = new BufferedReader(
        new InputStreamReader(System.in));

    System.out.print("Geben sie N an: ");
    n = Integer.parseInt(in.readLine());

    return n;
} // readInt
}
```

5. Berechnen Sie $\text{ggT}(144, 89)$ und $\text{ggT}(168, 9)$ mit dem Euklidischen Algorithmus und mit der rekursiven Methode!

Lösung:

144 und 89 sind relativ prim:

$$\begin{aligned}
 144 &= 1 \cdot 89 + 55, \\
 89 &= 1 \cdot 55 + 34, \\
 55 &= 1 \cdot 34 + 21, \\
 34 &= 1 \cdot 21 + 13, \\
 21 &= 1 \cdot 13 + 8, \\
 13 &= 1 \cdot 8 + 5, \\
 8 &= 1 \cdot 5 + 3, \\
 5 &= 1 \cdot 3 + 2, \\
 3 &= 1 \cdot 2 + 1, \\
 2 &= 2 \cdot 1.
 \end{aligned}$$

Rekursiv:

$$\text{ggT}(144, 89) = \text{ggT}(89, 55) = \text{ggT}(55, 43) = \dots = \text{ggT}(1, 0).$$

$\text{ggT}(168, 9) = 3$:

$$\begin{aligned}
 168 &= 18 \cdot 9 + 6, \\
 9 &= 1 \cdot 6 + 3, \\
 6 &= 2 \cdot 3.
 \end{aligned}$$

Rekursiv:

$$\text{ggT}(168, 9) = \text{ggT}(9, 6) = \text{ggT}(6, 3) = \text{ggT}(3, 3) = \text{ggT}(3, 0) = 3.$$

6. Implementieren Sie den Euklidischen Algorithmus und testen Sie die Funktion mit den Ergebnissen aus Aufgabe 5!

Lösung:

```

int gcd(int a, int b)
{
    int h;

    do {
        h = a - (a/b)*b;
        a = b;
        b = h;
    } while (h !=0 );
    return a;
}

```

7. Lösen Sie die Gleichungen $13x + 19y = 1000$ und $6930x + 1098y = 18$.

Lösung:

Die Lösung der ersten Gleichung ist $x = 3000, y = -2000$. Die beiden Zahlen 13 und 19 sind relativ prim; wir lösen die Gleichung $x \cdot 13 + y \cdot 19 = 1$ und multiplizieren anschließend die gefundenen Lösungen mit dem Faktor 1000.

Die Rekursion verläuft wie folgt:

$$\begin{aligned}
 W_0 &= (19, 1, 0), W_1 = (13, 0, 1), \\
 q_0 &= 1, W_2 = W_0 - W_1 = (6, 1, -1), \\
 q_1 &= 6, W_3 = W_1 - 2 \cdot W_2 = (1, -2, 3).
 \end{aligned}$$

Damit ist $(-2) \cdot 19 + 3 \cdot 13 = 1$ und die Lösung der ursprünglichen Aufgabe lautet $x = 3\,000, y = -2\,000$.

Die zweite Gleichung ist lösbar, denn es ist $\text{ggT}(6\,930, 1\,098) = 18$. Die Rekursion verläuft wie folgt:

$$\begin{aligned} W_0 &= (6\,930, 1, 0), W_1 = (1\,098, 0, 1), \\ q_0 &= 6, W_2 = (342, 1, -6), \\ q_1 &= 3, W_3 = (72, -3, 19), \\ q_2 &= 4, W_4 = (54, 13, -82), \\ q_3 &= 1, W_5 = (18, -16, 101). \end{aligned}$$

Also ist die Lösung $x = -16, y = 101$.

8. Implementieren Sie den verallgemeinerten Euklidischen Algorithmus und testen Sie die Funktion mit den Ergebnissen aus Aufgabe 7!

Lösung:

```
void extendedGcd(int a, int b, int ergebnis[3])
{
    int q, a1, a2, a3, b1, b2, b3, h1, h2, h3;
    a1 = a; a2 = 1; a3 = 0;
    b1 = b; b2 = 0; b3 = 1;

    do {
        q = a1/b1;
        h1 = a1 - q*b1;
        h2 = a2 - q*b2;
        h3 = a3 - q*b3;

        a1 = b1;
        a2 = b2;
        a3 = b3;
        b1 = h1;
        b2 = h2;
        b3 = h3;
    } while (h1 != 0 );
    ergebnis[0] = a1;
    ergebnis[1] = a2;
    ergebnis[2] = a3;
}
```

9. Sind die Zahlen 45, 49 und 50 als $25x + 35y$ darstellbar?

Lösung:

Es ist $\text{ggT}(25, 35) = 5$, also ist die Gleichung für die rechten Seiten 45 und 50 lösbar, mit den Lösungen $x = 27, y = -18$ und $x = 30, y = -20$.

10. Stellen Sie die Verknüpfungstabellen für \odot und \oplus für $m = 3$ und $m = 6$ auf!

Lösung:

Die Verknüpfungstabellen für $m = 3$ finden Sie in Tabelle 6.1, für $m = 6$ in Tabelle 6.2.

11. Beweisen Sie die 3er und die 9er Regel: Eine Zahl n ist genau dann durch 3 bzw. 9 teilbar, wenn ihre Quersumme durch 3 bzw. 9 teilbar ist!

\oplus	0	1	2
0	0	1	2
1	1	2	0
2	2	0	1

\odot	0	1	2
0	0	0	0
1	0	1	2
2	0	2	1

Tabelle 6.1: Die Verknüpfungstabellen für Aufgabe 10 im Fall $m = 3$

\oplus	0	1	2	3	4	5
0	0	1	2	3	4	5
1	1	2	3	4	5	0
2	2	3	4	5	0	1
3	3	4	5	0	1	2
4	4	5	0	1	2	3
5	5	0	1	2	3	4

\odot	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	1	2	3	4	5
2	0	2	4	0	2	4
3	0	3	0	3	0	3
4	0	4	2	0	4	2
5	0	5	4	3	2	1

Tabelle 6.2: Die Verknüpfungstabellen für Aufgabe 10 im Fall $m = 6$

Lösung:

Die Beweise verlaufen analog zur 11-er Regel. Es ist $10^i \equiv 1 \pmod m$, sowohl für $m = 3$ als auch für $m = 9$. Dann ist für $n = \sum_{i=0}^l a_i 10^i$

$$n \equiv \left(\sum_{i=0}^l a_i \right) \pmod m.$$

Wenn die Quersumme durch m teilbar ist, dann ist

$$\sum_{i=0}^l a_i \equiv 0 \pmod m.$$

12. Stellen Sie eine Wertetabelle für die Funktionen $n^2 \pmod{11}$ und $n^3 \pmod{11}$ auf und erstellen Sie den Graph!

Lösung:

Die Wertetabelle finden Sie in Tabelle 6.3; in 6.1 die Graphen.

Tabelle 6.3: Die Wertetabelle für die Funktionen $n^2 \pmod{11}$ und $n^3 \pmod{11}$ in Aufgabe 12

n	0	1	2	3	4	5	6	7	8	9	10
$n^2 \pmod{11}$	0	1	4	9	5	3	3	5	9	4	1
$n^3 \pmod{11}$	0	1	8	5	9	4	7	2	6	3	10

13. Verwenden Sie die Lösung der Aufgabe 12, um die zweite modulare Wurzel aus $n = 4$ und $n = 5$ sowie die dritte modulare Wurzel aus $n = 2$ zu bestimmen.

Lösung:

Die zweite Wurzel aus 4 sind $n = 2$ und $n = 9$.

Die zweite Wurzel aus 5 sind $n = 4$ und $n = 7$.

Die dritte Wurzel aus 2 ist $n = 7$.

14. Welche Zahlen haben zum Modul 8 eine modulare Inverse? Geben Sie die Inversen an!

Lösung:

Außer der 1 die Zahlen zwischen 2 und 7, die relativ prim mit 8 sind; das sind dann 3, 5 und 7.



Abbildung 6.1: Der Graph der Funktionen $n^2 \bmod 11$ (links) und $n^3 \bmod 11$ (rechts) für Aufgabe 12

Mit dem verallgemeinerten Euklidischen Algorithmus erhält man $-8 + 3 \cdot 3 = 1$; also ist die modulare Inverse von 3 zum Modul 8 gleich 3.

Für 5 ergibt sich $2 \cdot 8 + (-3) \cdot 5 = 1$; also ist die modulare Inverse von 5 zum Modul 8 gegeben durch -3 bzw. 5.

Für 7 ergibt sich $8 + (-1) \cdot 7 = 1$, also ist die modulare Inverse von 7 zum Modul 8 gegeben durch -1 bzw. 7.

15. Beweisen Sie, dass für die ganzen Zahlen a, b, q, r aus $a = q \cdot b + r$ die Gleichung $\text{ggT}(a, b) = \text{ggT}(b, r)$ folgt.

Lösung:

Es reicht zu zeigen, dass die Menge der Teiler von a und b mit der von b und r übereinstimmt; dann muss auch $\text{ggT}(a, b) = \text{ggT}(b, r)$ gelten.

Angenommen, d ist eine Zahl mit $d|a$ und $d|b$. Dann teilt d auch $r = a - b \cdot q$.

Angenommen, c ist eine Zahl mit $c|b$ und $c|r$. Dann teilt c auch $b \cdot q + r = a$.

16. Berechnen Sie $28^{12} \bmod 13$, $3^{15} \bmod 13$, $15^{83} \bmod 13$, $17^{20} \bmod 50$ und $49^{28} \bmod 58$.

Lösung:

$28^{12} \bmod 13 \equiv 1$, denn 28 und 13 sind relativ prim, der kleine Fermat'sche Satz kann angewandt werden.

Es gilt $3^{12} \equiv 1 \bmod 13$ und damit $3^{15} \equiv 3^3 \bmod 13 \equiv 1 \bmod 13$.

$15^{83} \equiv 7 \bmod 13$, denn mit dem kleinen Satz von Fermat gilt

$$15^{83} = 15^{72} \cdot 15^{11} \equiv (15^{12})^6 \cdot 15^{11} \bmod 13 \equiv 15^{11} \bmod 13.$$

$15^{12} \equiv 1 \bmod 13$; also ist die Lösung gegeben durch die modulare Inverse von 15 zum Modul 13, die durch 7 gegeben ist. Multiplikation mit 7 entspricht der Division durch 15, es ist $\text{ggT}(13, 15) = 1$.

$17^{20} \bmod 50 \equiv 1$, denn 17 und 50 sind relativ prim, und $\varphi(50) = 20$; der Satz von Euler kann angewandt werden.

Auch hier kann der Satz von Euler angewandt werden: $49^{28} \bmod 58 \equiv 1$, denn $\varphi(58) = 28$.

17. Berechnen Sie die Werte der Euler'schen Funktion für $41 \leq n \leq 60$.

Lösung:

$$\begin{aligned} \varphi(41) &= 40, \varphi(42) = \varphi(2 \cdot 3 \cdot 7) = 12, \varphi(43) = 42, \\ \varphi(44) &= \varphi(2^2 \cdot 11) = 20, \varphi(45) = \varphi(3^2 \cdot 5) = 30, \varphi(46) = \varphi(2 \cdot 23) = 22, \\ \varphi(47) &= 46, \varphi(48) = \varphi(2^4 \cdot 3) = 16, \varphi(49) = \varphi(7^2) = 42, \\ \varphi(50) &= \varphi(2 \cdot 5^2) = 20, \varphi(51) = \varphi(3 \cdot 17) = 32, \varphi(52) = \varphi(2^2 \cdot 13) = 24, \\ \varphi(53) &= 52, \varphi(54) = \varphi(2 \cdot 3^3) = 18, \varphi(55) = \varphi(5 \cdot 11) = 40, \\ \varphi(56) &= \varphi(2^3 \cdot 7) = 24, \varphi(57) = \varphi(3 \cdot 19) = 36, \varphi(58) = \varphi(2 \cdot 29) = 28, \\ \varphi(59) &= 58, \varphi(60) = \varphi(2^2 \cdot 3 \cdot 5) = 16. \end{aligned}$$

18. Implementieren Sie die Tausch-Chiffre und führen Sie sie mit den Schlüsseln (5,21) und (15,7) durch für die Nachricht „mathematik fuer informatiker“!

Lösung:

Für den Schlüssel (5,21) wird „mathematik“ wie folgt abgebildet. Zuerst auf die Zahlen

$$12, 0, 19, 7, 4, 12, 0, 19, 8, 10;$$

dann die Tauschchiffre

$$19, 1, 10, 18, 7, 19, 1, 10, 13, 3,$$

und als Zeichenkette „tbkshtbknds“.

Das Wort „fuer“ wird auf die Zahlen 5, 20, 4, 17 abgebildet; nach der Tauschchiffre ist 2, 5, 7, 20 und als Zeichenkette „cfhu“.

Für „informatik“ ergeben sich die Zahlen

$$8, 13, 5, 14, 17, 12, 0, 19, 8, 10$$

und nach der Tauschchiffre

$$13, 14, 2, 9, 20, 19, 1, 10, 13, 3.$$

Das entspricht der Zeichenkette „nocjutbknd“

Für die Schlüssel (15,7) ergibt sich für „mathematik“ die Tauschchiffre

$$7, 1, 4, 24, 3, 7, 1, 4, 5, 19;$$

als Zeichenkette „hbeydhbeft“.

Das Wort „fuer“ wird auf die Tauschchiffre 2, 5, 7, 20 und die Zeichenkette „cfhu“ abgebildet.

Für „informatik“ ergibt sich die Tauschchiffre

$$5, 14, 10, 21, 16, 7, 1, 4, 5, 19$$

und als Zeichenkette „fokvqhbeft“.

Und hier der Quellcode:

```
// Funktion, die aus char zwischen a und z eine zahl zwischen 0 und 25 macht
int c2i(const char c)
{
    return int(c)-97;
}

// Funktion, die aus int zwischen 1 und 26 char zwischen a und z macht
char i2c(int i)
{
```

```

    return char(i+97);
}

// Funktion, die den Eingabe-String in ein Integer-Feld umwandelt
void s2i(int laenge, const char satz[], int isatz[])
{
    for (int i = 0; i<laenge; i++)
        isatz[i] = c2i(satz[i]);
}

// Funktion, die ein Integer-Feld in einen String umwandelt
void i2s(int laenge, const int chiffre[], char satz[])
{
    for (int i = 0; i<laenge; i++)
        satz[i] = i2c(chiffre[i]);
}

// Funktion, die die Verschiebe-Chiffre mit S durchfuehrt
// alles modulo 26.
void verschiebeChiffre(int laenge, int s, int satz[])
{
    for (int i=0; i<laenge; i++)
    {
        satz[i] = (satz[i] + s) % 26;
        // Beim Dechiffrieren wird die negative Verschiebung in diese
        // Funktion eingesetzt. Dadurch können negative Restklassenvertreter
        // auftauchen. Abfangen und 26 dazu addieren.
        if (satz[i] < 0 ) satz[i] += 26;
    }
}

// Funktion, die multipliziert, modulo 26.
void mult(int laenge, int t, int satz[])
{
    for (int i=0; i<laenge; i++)
        satz[i] = (satz[i]*t) % 26;
}

// Tausch-Chiffre: zuerst verschieben, dann multiplizieren.
void tauschChiffre(int laenge, int s, int t, int chiffre[])
{
    verschiebeChiffre(laenge, s, chiffre);
    mult(laenge, t, chiffre);
}

// Funktion, die eine Tauschchiffre dechiffriert.
// s ist die Verschiebung, t der multiplikative Faktor.
// Als Faktor muss die multiplikative Inverse zum Modul 26 verwendet werden.
// Chiffrieren und deChiffrieren unterscheiden sich durch die Reihenfolge.
// Bei der Dechiffrierung wird zuerst multipliziert, dann -s abgezogen!
void tauschDeChiffre(int laenge, int s, int t, int chiffre[])
{
    mult(laenge, t, chiffre);
    verschiebeChiffre(laenge, s, chiffre);
}

```

19. Wie können Sie eine Tausch-Chiffre mit Schlüssel (s, t) angreifen, wenn Sie bereits für zwei Buchstabenpaare die Zuordnung kennen? Führen Sie das gefundene Verfahren durch für das

Paar $c \mapsto d, h \mapsto s!$ Dechiffrieren Sie mit dem gefundenen Schlüssel den Geheimtext „zds-wvmc“!

Lösung:

Durch die zwei bekannten Paare kann ein modulares lineares Gleichungssystem aufgestellt werden. Für die Funktion $f(n)$ sind die beiden Werte $f(2) = 3$ und $f(7) = 18$ bekannt. Dann sind s und t gesucht mit

$$\begin{aligned} 2 \cdot t + s &\equiv 3 \pmod{26}, \\ 7 \cdot t + s &\equiv 18 \pmod{26}. \end{aligned}$$

Auflösen der ersten Gleichung nach s ergibt

$$s \equiv (3 - 2 \cdot t) \pmod{26}.$$

Das können Sie in die zweite Gleichung einsetzen:

$$7 \cdot t + 3 - 2 \cdot t \equiv 18 \pmod{26}.$$

Dann erfüllt t die Gleichung

$$5 \cdot t \equiv 15 \pmod{26}.$$

Die multiplikative Inverse zu 5 ist die Zahl 21. Wir multiplizieren beide Seiten der Gleichung damit und erhalten

$$\begin{aligned} t &\equiv (21 \cdot 15) \pmod{26} \\ &\equiv 315 \pmod{26} \\ &\equiv 3. \end{aligned}$$

Dann ist $s \equiv (3 - 6) \pmod{26} \equiv (-3) \pmod{26} \equiv 23$.

Es wurde der Schlüssel $(23, 3)$ verwendet.

20. Für das Teilmengen-Summenproblem gibt es einen Empfänger mit privatem Schlüssel 1, 2, 5, 11, 32, 87, 141, $w = 901$, $m = 1234$ und öffentlichem Schlüssel 901, 568, 803, 39, 450, 645, 1173. Ver- und entschlüsseln Sie die Nachricht „mathematik fuer informatiker“!

Lösung:

Die Teilmengen-Summen Chiffre für das Wort „mathematik“:

$$3131, 2642, 2722, 1508, 3092, 3131, 2642, 2722, 2681, 3326.$$

Die Teilmengen-Summen Chiffre für das Wort „fuer“:

$$2564, 3895, 3092, 2917.$$

Die Teilmengen-Summen Chiffre für das Wort „informatiker“:

$$2681, 2603, 2564, 3776, 2917, 3131, 2642, 2722, 2681, 3326, 3092, 2917.$$

Hier der Quelltext der dafür benutzten C++-Funktionen. Für die 7-Bitdarstellung der Buchstaben wird die STL-Klasse `bitset` verwendet.

```
unsigned long knapsack(char wort, unsigned long publicKey[7])
{
    unsigned long ch;
    int i;
    bitset<7> bits(wort);
```

```
    ch = 0;
    for (i=0; i<7; i++)
        ch += key[i]*bits[i];
    return ch;
}

// Es wird davon ausgegangen, dass die im privaten Schlüssel
// abgelegten Zahlen // bereits so angeordnet sind,
// dass jede größer ist als die Summe aller ihrer Vorgänger.
unsigned long deKnapsack(unsigned long Bprime, unsigned long privateKey[7])
{
    int i, first;
    unsigned long summe;
    bitset<7> bits; // Alle bits sind auf Null gesetzt!

    // Zuerst den kleinsten privaten Schlüssel finden
    // in der Summe vorkommt.
    first = 6;
    summe = privateKey[first];
    while (Bprime <= summe) {
        summe = privateKey[first-1];
        first--;
    }
    bits.set(6-first);
    if (Bprime > summe )
        for (i=first-1; i>=0; i--) {
            if (Bprime >= summe+privateKey[i]) {
                summe += privateKey[i];
                bits.set(6-i);
            }
        }
    // Jetzt aus der Bitfolge wieder eine natürliche Zahl machen
    return bits.to_ulong();
}

void summenChiffre(int laenge, unsigned long publicKey[7],
                  char satz[], int chiffre[])
{
    int i;
    for (i=0; i<laenge; i++)
        chiffre[i] = knapsack(satz[i], key);
}

void summenDeChiffre(int laenge, unsigned long winv, unsigned long m,
                    unsigned long privateKey[7], unsigned long chiffre[],
                    char satz[])
{
    int i;
    unsigned long asciiCode, Bprime;

    for (i=0; i<laenge; i++) {
        Bprime = (chiffre[i]*winv) % m;
        asciiCode = deKnapsack(Bprime, privateKey);
        satz[i] = char(asciiCode);
    }
}
```

21. Bilden Sie ein eigenes Kryptosystem auf der Basis des Teilmengen-Summenproblems und testen Sie dieses System mit den Nachrichten „hanser“ und „mathematik fuer informatiker“!

Lösung:

Für ein Kryptosystem nach dem Teilmengen-Summenproblem benötigen Sie als aller erstes zwei natürliche Zahlen m und w , die relativ prim sind. Dann gibt es für w zum Modul m die modulare Inverse w^{-1} .

Anschließend müssen Sie für jeden Teilnehmer 7 Zahlen p_0, p_1, \dots, p_6 suchen, die

$$\sum_{i=0}^k p_i < p_{k+1}, k = 0, \dots, 5.$$

erfüllen. Diese bilden den privaten Schlüssel für den Teilnehmer. Als öffentlichen Schlüssel veröffentlichen Sie anschließend die 7 Zahlen a_0, a_1, \dots, a_6 mit

$$a_k = (w \cdot p_i) \bmod m.$$

Als Modul wählen wir jetzt $m = 1\,026$ und als $w = 511$. Überzeugen Sie sich mit Hilfe des verallgemeinerten Euklidischen Algorithmus, es ist $\text{ggT}(1026, 511) = 1$; als modulare Inverse erhalten Sie $w^{-1} = 769$.

Als privaten Schlüssel eines neuen Teilnehmers werden die Zahlen

$$p_0 = 3, p_1 = 4, p_2 = 11, p_3 = 23, p_4 = 45, p_5 = 87, p_6 = 256$$

gewählt. Dann ist der öffentlichen Schlüssel für diesen Teilnehmer

$$a_0 = 507, a_1 = 1018, a_2 = 491, a_3 = 467, a_4 = 423, a_5 = 339, a_6 = 514.$$

Für die Nachricht „hanser“ ergibt sich dann die Chiffre

$$1992, 2039, 2754, 2869, 2462, 2355.$$

Für „mathematik“ ist die Chiffre

$$2929, 2039, 2439, 1992, 2462, 2929, 2039, 2439, 2506, 2845;$$

„fuer“ wird zu 2287, 2953, 2462, 2355 und „informatiker“ zu

$$2506, 2754, 2287, 3268, 2355, 2929, 2039, 2439, 2506, 2845, 2462, 2355.$$

22. Übertragen Sie für den öffentlichen Schlüssel $n = 34\,571 = 181 \cdot 191$, $e = 7\,901$ und für den privaten Schlüssel $d = 11\,501$ die Nachricht „mathematik fuer informatiker“ mit einem RSA-System!

Lösung:

Ich hoffe, Sie haben nicht versucht, diese Aufgabe per Hand zu lösen. Ein Problem bei der Implementierung sind die großen Zahlen, die auftreten. Eine Lösung wäre, die Potenzen so zu berechnen, dass nach jeder Multiplikation sofort wieder die Restklasse bestimmt wird.

In Java können Sie die Klasse `BigInteger` für beliebig große Zahlen verwenden, die eine Funktion `modPow` zur Verfügung stellt.

In C++ wäre die Potenz dann wie folgt implementierbar:

```
// Produkt als sukzessives Addieren mit zwischenzweitlichem Rest.
unsigned long produktMod(unsigned long a, unsigned long b, unsigned long m)
{
```

```

unsigned long i, summe;

summe = 0;
for (i=0; i<b; i++)
    summe = (summe + a) % m;

return summe;
}
// Potenz als sukzessives Multiplizieren mit zwischenzeitlichem Rest.
unsigned long potenzMod(unsigned long a, unsigned long b, unsigned long m)
{
    unsigned long i, potenz;

    potenz = 1;
    for (i=0; i<b; i++)
        potenz = produktMod(potenz, a, m);

    return potenz;
}

```

Das RSA-System ist dann etwa so implementierbar:

```

void rsaChiffre(int laenge, unsigned long publicKey, unsigned long m,
               unsigned long chiffre[])
{
    int i;

    for (i=0; i<laenge; i++) {
        chiffre[i] = potenzMod(chiffre[i], publicKey, m);
    }
}

void rsaDeChiffre(int laenge, unsigned long privateKey, unsigned long m,
                  unsigned long chiffre[])
{
    int i;

    for (i=0; i<laenge; i++) {
        chiffre[i] = potenzMod(chiffre[i], privateKey, m);
    }
}

```

Die RSA-Chiffre für das Wort „mathematik“:

3 165, 22 783, 31 817, 24 968, 16 790, 3 165, 22 783, 31 817, 21 486, 23 456.

Die RSA-Chiffre für das Wort „fuer“:

13 390, 32 585, 16 790, 27 537.

Die RSA-Chiffre für das Wort „informatiker“:

21 486, 28 638, 13 390, 5 441, 27 537, 3 165, 22 783, 31 817, 21 486, 23 456, 16 790, 27 537.

23. Bilden Sie ein eigenes RSA-Kryptosystem und testen Sie dieses System mit den Nachrichten „hanser“ und „mathematik fuer informatiker“!

Lösung:

Zuerst müssen Sie zwei Primzahlen auswählen, beispielsweise $p = 167$ und $q = 97$. Dann ist $m = p \cdot q = 16199$ und $\varphi(m) = (p - 1)(q - 1) = 15936$.

Für die zu $\varphi(m)$ relativ prime Zahl kann beispielsweise $d = 41$ gewählt werden. Mit dem verallgemeinerten Euklidischen Algorithmus kann dazu die modulare Inverse zum Modul $\varphi(m)$ berechnet werden; es ist $e = 7385$.

Die RSA-Chiffre für das Wort „hanser“ lautet dann:

12 722, 6 305, 9 007, 1 049, 12 847, 11 150.

Die RSA-Chiffre für das Wort „mathematik“:

15 023, 6 305, 127, 12 722, 12 847, 15 023, 6 305, 127, 1 447, 3 261.

Die RSA-Chiffre für das Wort „fuer“:

11 366, 12 759, 12 847, 11 150.

Die RSA-Chiffre für das Wort „informatiker“:

1 447, 9 007, 11 366, 14 854, 11 150, 15 023, 6 305, 127, 1 447, 3 261, 12 847, 11 150.

24. Übertragen Sie für $p = 7919$, $a = 511$, dem öffentlichen Schlüssel $y = 4143$ und privatem Schlüssel $x = 73$ eines ElGamal-Systems die Nachricht „mathematik fuer informatiker“!

Lösung:

Ich hoffe, Sie haben genauso wie bei den Aufgaben zu RSA nicht versucht, diese Aufgabe per Hand zu lösen. Ein Problem bei der Implementierung sind die großen Zahlen, die auftreten. Eine Lösung wäre, die Potenzen so zu berechnen, dass nach jeder Multiplikation sofort wieder die Restklasse bestimmt wird. Die Lösung besteht wie beim RSA-Verfahren entweder in der Verwendung der Klasse `BigInteger` für Java oder die C++-Funktionen, die schon in Aufgabe 22 vorgestellt wurden.

Das ElGamal-Verfahren ist dann etwa so implementierbar:

```
void elGamalChiffre(int laenge, unsigned long p, unsigned long a,
                  unsigned long y, unsigned long r,
                  unsigned long chiffre[], unsigned long &z)
{
    // Die Nachricht besteht aus der chiffre und der Zahl z!
    int i;
    unsigned long yr;

    yr = potenzMod(y, r, p);

    for (i=0; i<laenge; i++)
        chiffre[i] = produktMod(chiffre[i], yr, p);

    z = potenzMod(a, r, p);
}

void elGamalDeChiffre(int laenge, unsigned long privateKey, unsigned long p,
                    unsigned long chiffre[], unsigned long z)
{
    int i;
    long potenz, inverse;

    potenz = potenzMod(z, privateKey, p);
    inverse = modulareInverse(potenz, p);
```

```

for (i = 0; i < laenge; i++)
    chiffre[i] = produktMod(chiffre[i], inverse, p);
}

```

Die ElGamal-Chiffre für das Wort „mathematik“:

791, 5 063, 6 218, 2 571, 3 639, 791, 5 063, 6 218, 2 215, 1 503.

Die ElGamal-Chiffre für das Wort „fuer“:

3 283, 5 862, 3 639, 6 930.

Die ElGamal-Chiffre für das Wort „informatiker“:

2 215, 435, 3 283, 79, 6 930, 791, 5 063, 6 218, 2 215, 1 503, 3 639, 6 930.

25. Bilden Sie ein eigenes ElGamal-Kryptosystem und testen Sie dieses System mit den Nachrichten „hanser“ und „mathematik fuer informatiker“!

Lösung:

Sie benötigen eine Primzahl p und eine dazu relativ prime Basis a . Möglich wäre $p = 6\,133$ und $a = 471$.

Für den Empfänger soll der private Schlüssel $x = 99$ gewählt sein. Daraus ergibt sich mit $y \equiv a^x \pmod{p}$ der öffentliche Schlüssel $y = 1\,071$. Zusätzlich wird noch die zufällige Zahl $r = 7$ ausgewählt.

Die zusätzlich übertragene Zahl ist $z = 5\,762$.

Die ElGamal-Chiffre für das Wort „hanser“ lautet:

2 355, 3 199, 4 260, 2 781, 4 469, 5 530.

Die ElGamal-Chiffre für das Wort „mathematik“:

876, 3 199, 32, 2 355, 4 469, 876, 3 199, 32, 5 739, 241.

Die ElGamal-Chiffre für das Wort „fuer“:

1 720, 3 416, 4 469, 5 530.

Die ElGamal-Chiffre für das Wort „informatiker“:

5 739, 4 260, 1 720, 1 511, 5 530, 876, 3 199, 32, 5 739, 241, 4 469, 5 530.

Kapitel 7

Graphentheorie

Verständnisfragen

Sachfragen

1. Was ist ein ungerichteter Graph?
2. Erläutern Sie den Begriff *Adjazenz*!
3. Erläutern Sie den Eckengrad in einem Graphen!
4. Welchen Zusammenhang gibt es zwischen den Eckengraden und der Anzahl der Kanten eines ungerichteten Graphen?
5. Wieviele Ecken ungeraden Grades hat ein ungerichteter Graph?
6. Was ist ein Weg in einem Graphen?
7. Was ist ein Kreis in einem Graphen?
8. Wie ist die Länge eines Wegs und eines Kreises in einem Graphen definiert?
9. Was versteht man unter einem zusammenhängenden Graphen?
10. Was ist eine Brücke in einem zusammenhängenden Graphen?
11. Was ist der Abstand zweier Ecken in einem Graphen?
12. Erläutern Sie den Begriff des Eulergraphen und das Königsberger Brückenproblem!
13. Was ist ein Hamiltonkreis und ein Hamiltongraph?
14. Was ist ein gerichteter Graph?
15. Was ist eine Adjazenzmatrix?
16. Was ist eine Inzidenzmatrix?
17. Wie kann ein gerichteter Graph dargestellt werden?
18. Was ist ein Baum? Was ist ein Wald?
19. Was ist ein binärer Baum?

20. Was ist das Niveau eines binären Baums? Kennen Sie eine Abschätzung für das Niveau eines Baums?
21. Welche Aussagen kennen Sie über die Anzahl der Ecken eines regulären binären Baums?
22. Was ist ein aufspannender Baum?
23. Welchen Zusammenhang zwischen der Anzahl der Ecken und der Kanten gibt es in einem Baum?
24. Wieviele Blätter enthält ein regulärer binärer Baum?
25. Was ist ein aufspannender Baum?
26. Wie hängen aufspannende Bäume und kürzeste Wege in einem Graphen zusammen?
27. Beschreiben Sie die Breitensuche!
28. Beschreiben Sie die Tiefensuche!
29. Was ist ein bewerteter Graph?
30. Was ist ein minimal aufspannender Baum?
31. Beschreiben Sie den Kruskal-Algorithmus!
32. Beschreiben Sie den Dijkstra-Algorithmus!
33. Beschreiben Sie das Problem des Handlungsreisenden!
34. Wie kann das Problem des Handlungsreisenden gelöst werden?
35. Was ist ein planarer Graph?
36. Wie lautet die Eulerformel?
37. Kennen Sie ein notwendiges Kriterium für die Planarität eines gegebenen Graphen?
38. Was ist eine Färbung eines Graphen?
39. Was ist die chromatische Zahl $\chi(G)$ eines Graphen G ?
40. Wie lautet der Vierfarbensatz?
41. Was ist ein bipartiter Graph?
42. Wie kann ein bipartiter Graph charakterisiert werden?
43. Was ist ein Matching?
44. Was ist ein gesättigtes Matching? Was ist ein maximales Matching?
45. Was ist ein alternierender Weg in einem bipartiten Graphen?
46. Was ist ein vergrößernder Weg in einem bipartiten Graphen?
47. Wie kann ein Matching in einem bipartiten Graphen vergrößert werden?
48. Wie lautet der Heiratssatz?
49. Was ist eine minimale Eckenüberdeckung?
50. Wie lautet der Dualitätssatz von König?
51. Wie verläuft der Ungarische Algorithmus?

Methodenfragen

1. Eckengrade in einem Graphen bestimmen können.
2. Einfache Graphen auf der Basis einer gegebenen Ecken- und Kantenmenge grafisch darstellen können.
3. Wege und Kreise in einem Graphen erkennen und darstellen können.
4. Den Abstand zwischen zwei Ecken in einem Graphen bestimmen können.
5. Entscheiden können, ob ein gegebener Graph ein Eulergraph ist.
6. Einen Eulerkreis konstruieren können, wenn der Graph ein Eulergraph ist.
7. Die Adjazenz- oder Inzidenzmatrix eines gegebenen Graphen aufstellen können.
8. Für eine gegebene Adjazenz- oder Inzidenzmatrix den Graphen beschreiben und grafisch darstellen können.
9. Die Adjazenzliste eines gegebenen Graphen aufstellen können.
10. Für eine gegebene Adjazenzliste einen Graphen beschreiben und grafisch darstellen können.
11. Datenstrukturen für die Darstellung eines Graphen implementieren und anwenden können.
12. Entscheiden können, ob ein gegebener Graph ein Baum ist.
13. Die Eigenschaften eines Baums bestimmen können.
14. Für einen gegebenen Graphen einen aufspannenden Baum konstruieren können!
15. Die Breitensuche durchführen und implementieren können.
16. Die Tiefensuche durchführen und implementieren können.
17. Den Kruskal-Algorithmus durchführen und implementieren können.
18. Den Dijkstra-Algorithmus durchführen und implementieren können.
19. Die MST-Heuristik für die Lösung des Problem des Handlungsreisenden durchführen können.
20. Die Eulerformel auf einen planaren Graphen anwenden können.
21. Überprüfen können, ob ein Graph notwendige Bedingungen für die Planarität erfüllt.
22. Einen Graphen färben können.
23. Überprüfen können, ob ein gegebener Graph bipartit ist.
24. Feststellen können, ob ein Matching gesättigt oder maximal ist.
25. Ein gesättigtes Matching bestimmen können.
26. Vergrößernde Wege finden und anwenden können.
27. Den Ungarischen Algorithmus durchführen können.

Übungsaufgaben

1. Zeichnen Sie die Graphen G_1 mit der Inzidenzmatrix B und G_2 mit der Adjazenzmatrix A :

$$B = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}, A = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \end{pmatrix}$$

Lösung:

Lesen Sie die Inzidenzmatrix spaltenweise, dann können Sie direkt den Graphen zeichnen. Der Graph für B hat 5 Ecken und 8 Kanten. Anschließend kann man versuchen, das Layout zu verbessern. Bei der Adjazenzmatrix reicht es, das obere Dreieck zu betrachten und die Kanten zu zeichnen. Auch hier bietet es sich an, das Layout anschließend zu verbessern.

Ein Werkzeug für das Layout von Graphen ist GraphViz von AT&T, das Sie unter der URL [hier noch die URL einfügen](#) finden. Der ungerichtete Graph zur Adjazenzmatrix kann dann als Eingabedatei wie folgt definiert werden:

```
graph G {
  1 -- 2;
  1 -- 4;
  1 -- 5;
  2 -- 3;
  2 -- 6;
  3 -- 4;
  3 -- 6;
  4 -- 5;
  5 -- 6;
```

Für den Graphen mit der Inzidenzmatrix B ist die Eingabedatei wie folgt gegeben:

```
graph G {
  1 -- 2;
  1 -- 4;
  1 -- 5;
  2 -- 3;
  2 -- 5;
  3 -- 4;
  3 -- 5;
  4 -- 5;
```

Die Ergebnisse finden Sie in Abbildung 7.1 und 7.2.

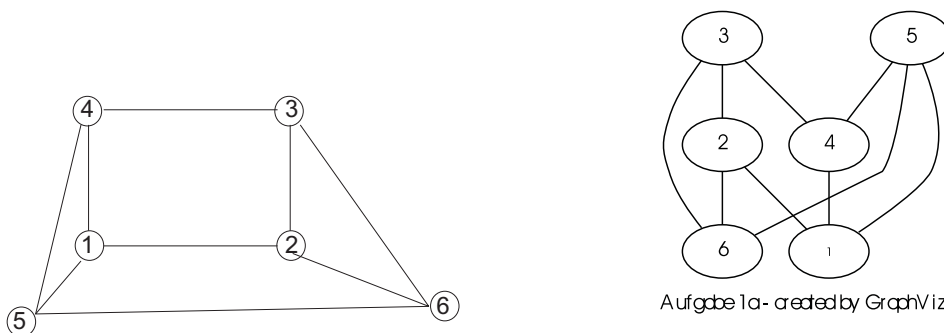


Abbildung 7.1: Layout des Graphen zur Adjazenzmatrix A in Aufgabe 1; links manuell; rechts berechnet mit GraphViz

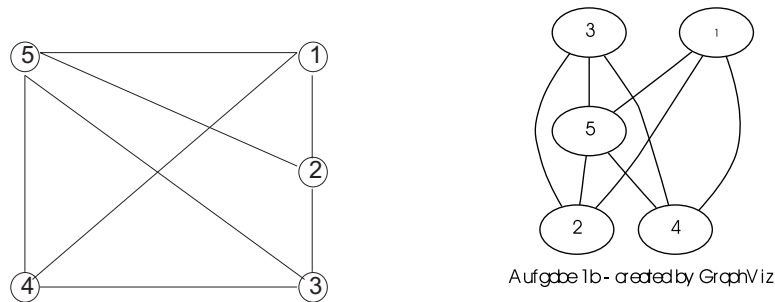


Abbildung 7.2: Layout des Graphen zur Inzidenzmatrix B in Aufgabe 1; links manuell; rechts mit GraphViz

2. Beweisen Sie, dass für einen Graphen $G(E, K)$ mit den Ecken u_1, \dots, u_n , Adjazenzmatrix A und Inzidenzmatrix B die Gleichung $BB^T = \text{diag}(d(u_1), \dots, d(u_n)) + A$ erfüllt ist.

Lösung:

Die Matrix BB^T ist eine $n \times n$ Matrix. Für die Elemente des Produkts ausserhalb der Diagonalen ist $b_{ij}b_{lj} = 1$ genau dann, wenn es eine Kante zwischen den Ecken u_i und u_j gibt. In der Diagonalen gilt $\sum_{k=1, k \neq i}^m b_{ik}b_{ki} = d(u_i)$.

3. Beweisen Sie, dass eine Kante in einem Graphen genau dann eine Brücke ist, wenn sie in keinem Kreis des Graphen liegt.

Lösung:

Angenommen, k ist eine Brücke und liegt in einem Kreis. Dann kann nach dem Herausnehmen von k immer noch jeder Knoten im Kreis erreicht werden; ein Widerspruch, denn nach dem Herausnehmen einer Brücke zerfällt der Graph in mindestens zwei Zusammenhangskomponenten.

Ist umgekehrt $k = u_i u_j$ eine Brücke, dann gibt es keinen weiteren Weg von u_i nach u_j , k kann nicht in einem Kreis liegen.

4. Beweisen Sie, dass für die Inzidenzmatrix B eines gerichteten Graphen $GR(E, K)$ die Zeilensummen gleich $d^+(u) - d^-(u)$ sind und die Spalten sich zu 0 summieren.

Lösung:

In jeder Spalte j gibt es zwei Einträge ungleich Null. Einer der Einträge entspricht einem Anfangs-, einer einem Endpunkt. Daraus folgt die Spaltensumme.

$d^+(u)$ entspricht der Anzahl der 1-Einträge in jeder Zeile, $d^-(u)$ der (-1) -Einträge.

5. Gibt es einen einfachen Graphen, der 15 Ecken vom Grad 5 hat?

Lösung:

Nein, denn die Summe der Eckengrade muss eine gerade Zahl sein!

6. Stellen Sie für die Graphen in Abbildung 7.3 die Adjazenz- und Inzidenzmatrix und die Adjazenzliste auf!

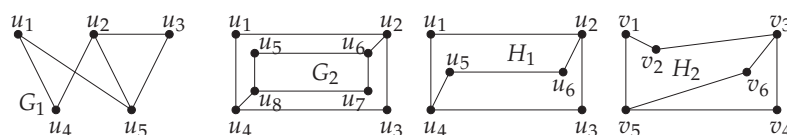


Abbildung 7.3: Die Graphen für die Aufgaben 6 und 7

Lösung:

Die Adjazenzmatrix von G_1 ist eine 5×5 -Matrix:

$$G_1 = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \end{pmatrix}.$$

Für die Inzidenzmatrix müssen die Kanten nummeriert werden:

$$k_1 = u_1u_4, k_2 = u_1u_5,$$

$$k_3 = u_2u_3, k_4 = u_2u_5,$$

$$k_5 = u_2u_5, k_6 = u_3u_5.$$

Damit ergibt sich die 5×6 -Inzidenzmatrix

$$IG_1 = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 \end{pmatrix}$$

G_2 hat die 8×8 -Adjazenzmatrix

$$G_2 = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \end{pmatrix}$$

Für die Inzidenzmatrix müssen die Kanten nummeriert werden:

$$k_1 = u_1u_2, k_2 = u_2u_3,$$

$$k_3 = u_3u_4, k_4 = u_4u_1,$$

$$k_5 = u_2u_6, k_6 = u_4u_8,$$

$$k_7 = u_5u_6, k_8 = u_6u_7,$$

$$k_9 = u_7u_8, k_{10} = u_8u_5.$$

Damit ergibt sich die 8×10 -Inzidenzmatrix

$$IG_2 = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \end{pmatrix}.$$

H_1 hat die 6×6 -Adjazenzmatrix

$$H_1 = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \end{pmatrix}.$$

Für die Inzidenzmatrix müssen die Kanten nummeriert werden:

$$\begin{aligned} k_1 &= u_1u_2, k_2 = u_2u_3, \\ k_3 &= u_3u_4, k_4 = u_4u_1, \\ k_5 &= u_4u_5, k_6 = u_5u_6, \\ k_7 &= u_2u_6. \end{aligned}$$

Damit ergibt sich die 6×7 -Inzidenzmatrix

$$IH_1 = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}.$$

H_2 hat die 6×6 -Adjazenzmatrix

$$H_2 = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}.$$

Für die Inzidenzmatrix müssen die Kanten nummeriert werden:

$$\begin{aligned} k_1 &= v_1v_2, k_2 = v_2v_3, \\ k_3 &= v_3v_4, k_4 = v_4v_5, \\ k_5 &= v_5v_1, k_6 = v_5v_6, \\ k_7 &= v_3v_6. \end{aligned}$$

Damit ergibt sich die 6×7 -Inzidenzmatrix

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}.$$

7. Sind die Graphen H_1 und H_2 in Abbildung 7.3 isomorph?

Lösung:

Wenn die beiden Graphen isomorph sind, dann muss es eine adjazenzerhaltende Abbildung zwischen ihnen geben.

Es ist $d(u_1) = 2$ und diese Ecke in H_1 hat nur Nachbarn, die Grad 3 besitzen. Dann ist als Bild für u_1 nur v_4 oder v_6 möglich. Dann wird jetzt $\varphi(u_1) = v_6$ definiert; sollte dies nicht zu einem Isomorphismus führen, kann die Variante $\varphi(u_1) = v_4$ betrachtet werden.

u_2 ist adjazent zu u_1 , also sind die einzig möglichen Bilder für u_2 nur die Ecken v_3 oder v_5 . Wir setzen willkürlich $\varphi(u_2) = v_3$. Als Bild für u_3 wird $\varphi(u_3) = v_4$ gesetzt; auch $\varphi(u_3) = v_2$ wäre noch möglich gewesen.

Für die restlichen Ecken wird $\varphi(u_4) = v_5$ und $\varphi(u_5) = v_1, \varphi(u_6) = v_2$ gesetzt.

Stellt man die Adjazenzmatrizen beider Graphen auf, wobei die Ecken in H_2 durch die Bilder $\varphi(u_1), \dots, \varphi(u_6)$ nummeriert werden (das Element 12 ist also beispielsweise gegeben durch die Information, ob es zwischen $\varphi(u_1) = v_6$ und $\varphi(u_2) = v_3$ eine Kante gibt), dann erhält man

$$H_1 = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \end{pmatrix}, H_2 = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \end{pmatrix}.$$

Sie stimmen offensichtlich überein; dann ist φ eine kantenerhaltende Abbildung; die beiden Graphen sind isomorph.

8. Stellen Sie fest, ob die Graphen in Abbildung 7.4 Eulergraphen sind, und geben Sie gegebenenfalls einen Eulerkreis an!

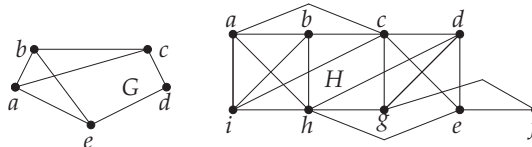


Abbildung 7.4: Graphen zu Aufgabe 8

Lösung:

Der Graph G hat nur eine Ecke mit geradem Grad, nämlich c .

Der Graph H hat nur Ecken mit geradem Grad, es ist $d(f) = 2$, alle anderen Ecken haben Grad 4.

Ein Eulerkreis ist $a, i, h, g, d, e, f, g, c, e, h, d, c, a, b, i, c, b, h, a$.

9. Wie viele Kanten hat ein Baum mit 10 000 Ecken? Wie viele Blätter hat ein regulärer binärer Baum mit 99 Ecken?

Lösung:

Ein Baum hat $|K| = |E| - 1$ Kanten; also gibt es im vorliegenden Fall 9 999 Kanten. Ein regulärer binärer Baum hat $\frac{n+1}{2}$ Blätter, im vorliegenden Fall also $\frac{101}{2} = 50$ Blätter.

10. Welche der folgenden Codes sind Präfix-Codes? a) a:11, e:00, t:10, s:01; b) a:0, e:1, t:01, s:001, c) a:101, e:11, t:001, s:011, n:010. Geben Sie, wenn möglich, den Klartext für die Codes 01111110 und 00110101101111 an!

Lösung:

Die Fälle a) und c) stellen Präfix-Codes dar; bei b) ist die Zuordnung nicht eindeutig; denn 01 kann sowohl t als auch ae bedeuten; 001 könnte sowohl für s als auch für aae oder für at stehen.

01111110 steht in Code a) für das Wort *saat*, in c) gibt es keinen Sinn.

00110101101111 steht in Code a) für *eassta*, in Code c) für *tasse*.

11. Geben Sie die Codes für $\{a, e, i, k, o, p, u\}$ für den binären Baum in Abbildung 7.5 an!

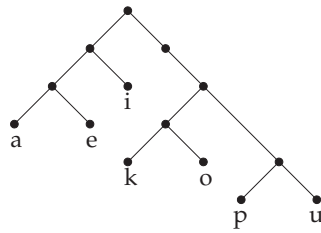


Abbildung 7.5: Der binäre Baum für Aufgabe 11

Lösung:

a: 000, e: 001, i: 01, k: 1100, o: 1101, p: 1110, u: 1111.

12. Bestimmen Sie einen Huffman-Code für die Zeichen $\{a, b, c, d, e, f\}$ mit den Häufigkeiten 12%, 32%, 4%, 20%, 16% und 16%. Bestimmen Sie die mittlere Codewort-Länge!

Lösung:

Das Ergebnis ist a: 1110, b: 10, c: 1111, d: 00, e: 110, f: 01. Die mittlere Code-Länge ist 2,16.

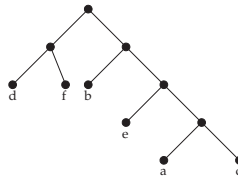


Abbildung 7.6: Der binäre Baum zur Lösung von Aufgabe 12

13. Konstruieren Sie einen minimalen aufspannenden Baum mit dem Kruskal- und dem Prim-Algorithmus für die bewerteten Graphen aus Abbildung 7.7.

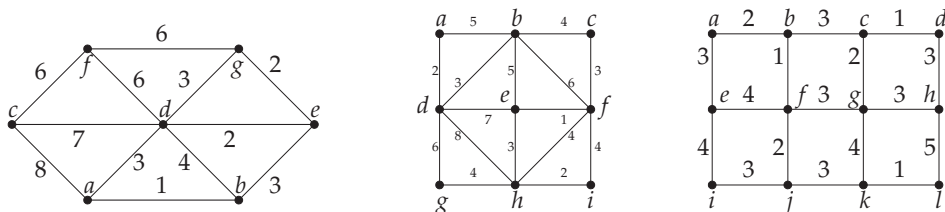


Abbildung 7.7: Die bewerteten Graphen zu Aufgabe 13

Lösung:

Die Lösungen finden Sie in den Tabellen 7.1, 7.2 und 7.3.

14. Ein *maximal aufspannender Baum* in einem zusammenhängenden und bewerteten Graphen ist ein Baum mit dem größtmöglichen Gewicht. Formulieren Sie einen Algorithmus, der einen solchen Baum konstruiert, und wenden Sie ihn auf den Graphen links in Abbildung 7.7 an!

Lösung:

Sie können den Kruskal- oder Prim-Algorithmus „umbauen“; statt des Minimums verwenden Sie immer das Maximum. Tabelle 7.4 zeigt das Ergebnis, wenn Sie die Kanten der Größe nach sortieren und immer eine Kante mit größtmöglichem Gewicht auswählen, solange kein Kreis geschlossen wird.

Kruskal-Algorithmus			Prim-Algorithmus		
n	Kante	Gewicht	n	Kante	Gewicht
1	ab	1	1	ab	1
2	ed	2	2	be	3
3	eg	2	3	ed	2
4	be	3	4	eg	2
5	fg	6	5	fg	6
6	fc	6	6	fc	6
		20			20

Tabelle 7.1: Kruskal- und Prim-Algorithmus für den Graph links in Abbildung 7.7

Kruskal-Algorithmus			Prim-Algorithmus		
n	Kante	Gewicht	n	Kante	Gewicht
1	ef	1	1	ef	1
2	ad	2	2	cf	3
3	hi	2	3	eh	3
4	bd	3	4	hi	2
5	cf	3	5	bc	4
6	eh	3	6	bd	3
7	bc	4	7	ad	2
8	gh	4	8	gh	4
		22			22

Tabelle 7.2: Kruskal- und Prim-Algorithmus für den Graph in Abbildung 7.7 Mitte

15. Bestimmen Sie die kürzesten Wege im Graphen in Abbildung 7.8 für die folgenden Eckenpaare: a) a, h , b) a, d , c) a, f , d) b, h !

Lösung:

- Der Weg zwischen a und h ist gegeben durch ac, cd, de, eg, gh ; er hat die gewichtete Länge 16.
- Der Weg zwischen a und d ist gegeben durch ac, cd ; er hat die gewichtete Länge 6.
- Der Weg zwischen c und f ist gegeben durch cd, df ; er hat die gewichtete Länge 8.
- Der Weg zwischen b und h ist gegeben durch bd, de, eg, gh ; er hat die gewichtete Länge 15.

16. Führen Sie den Dijkstra-Algorithmus für den Graphen in ?? auf Seite ?? durch, ausgehend von Ecke c !

Lösung:

$\{ce, ef, cb, cd, ba\}$

17. Bestimmen Sie einen Hamiltonkreis mit Hilfe der MST Heuristik für den Graphen in Abbildung 7.9!

Lösung:

Aus dem Graphen kann ein vollständiger Graph gemacht werden, in dem Sie die fehlenden Kanten bilden und mit der Länge der entsprechenden Wege belegen. Im ersten Schritt wird ein minimal aufspannender Baum bestimmt, beispielsweise ausgehend von Ecke a . Das Ergebnis ist die Kantenmenge $\{ac, cd, de, db\}$.

Kruskal-Algorithmus			Prim-Algorithmus		
n	Kante	Gewicht	n	Kante	Gewicht
1	cd	1	1	bf	1
2	kl	1	2	ab	2
3	bf	1	3	fj	2
4	cg	2	4	ae	3
5	ab	2	5	ij	3
6	fj	2	6	fg	3
7	bc	3	7	eg	2
8	jk	3	8	cd	1
9	gh	3	9	gh	3
10	ij	3	10	hl	3
11	ae	3	11	kl	1
		24			24

Tabelle 7.3: Kruskal- und Prim-Algorithmus für den Graph in Abbildung rechts 7.7

An den Kruskal-Algorithmus angelehnt		
n	Kante	Gewicht
1	ac	8
2	cd	7
3	cf	6
4	fg	6
5	bd	4
6	be	3
		34

Tabelle 7.4: An den Kruskal-Algorithmus angelehnter Algorithmus zur Bestimmung des maximal aufspannenden Baums für den Graph links in Abbildung 7.7

Jetzt werden alle Kanten in diesem Baum verdoppelt und bereits durchlaufene Kanten werden übersprungen; dadurch erhält man den Hamiltonkreis $acdef$ mit Länge 15.

18. Bestimmen Sie die chromatische Zahl für den Graphen, der aus K_5 durch Entfernen einer beliebigen Kante entsteht, und für den Graphen $G(E, K)$ mit $E = \{0, 1, 2, \dots, n-1\}$, $K = \{(p, q) \in E^2 \mid q \equiv p+1 \pmod n\}$.

Lösung:

K_5 ohne eine beliebige Kante hat die chromatische Zahl 4, wie Sie in Abbildung 7.10 erkennen können; der gierige Färbe-Algorithmus ergibt das ebenfalls sehr schnell, wenn Sie mit der Ecke oben beginnen, die immer noch Grad 4 hat.

Die Graphen mit der Kantenmenge $K = \{(p, q) \in E^2 \mid q \equiv p+1 \pmod n\}$ sind Kreise. Für gerades n haben Kreise die chromatische Zahl 2, für ungerades n 3.

19. Welche chromatische Zahl hat der Graph in Abbildung 7.11? Bestimmen Sie eine Färbung!

Lösung:

Es ist $\chi(G) = 4$. Eine Färbung kann mit dem gierigen Algorithmus bestimmt werden. Hier der Verlauf des Verfahrens:

- (a) $i = 1$, farbe = 1; $f(a) = 1$, dann wird $f(b) = f(c) = f(d) = f(e) = -1$.
 $f(f) = 1$, dann wird $f(g) = -1$.
- (b) $i = 2$, farbe = 2; $f(b) = 2$, dann wird $f(c) = -2$.
 $f(d) = 2$, dann wird $f(e) = f(g) = -2$.

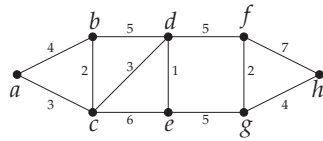


Abbildung 7.8: Der Graph für Aufgabe 15

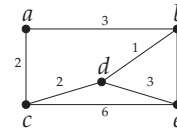
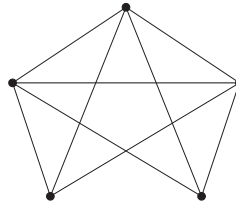


Abbildung 7.9: Der Graph für Aufgabe 17

Abbildung 7.10: Der K_5 mit einer weggelassenen Kante

- (c) $i = 3$, farbe = 3; $f(c) = 3$.
 $f(e) = 3$, dann wird $f(g) = -3$.
 (d) $i = 4$, farbe = 4; $f(g) = 4$.

20. Suchen Sie einen Graphen mit 30 Kanten, in dem jede Ecke Grad 5 hat, und bestimmen Sie seine chromatische Zahl!

Lösung:

Ein Graph mit 12 Ecken, die alle Grad 5 haben, erfüllt wegen

$$\sum_{i=1}^{12} d(u_i) = 2 \cdot 30$$

die Forderung.

Beispielsweise der Graph mit den Zusammenhangskomponenten $G = K_6 \cup K_6$; er hat die chromatische Zahl 6.

21. Bestimmen Sie für den bipartiten Graphen G in Abbildung 7.12 ein maximales Matching, ausgehend von dem eingezeichneten gesättigten Matching.

Lösung:

In Abbildung 7.13 sehen Sie nochmals die Ausgangssituation und das Endergebnis. Bei der Durchführung des Verfahrens und auch beim Nachvollziehen der Lösung ist es angebracht, den sukzessive in den Schritten 3 und 4 aufgebauten Wurzelbaum immer aufzuzeichnen. In der angegebenen Lösung finden Sie die Bäume nach einer Entscheidung in Schritt 4.

Das Ausgangsmatching M ist gesättigt. Im Schritt 2 des ungarischen Algorithmus ist $E(M) \cap S = \{a_2, a_3, a_4\}$. Als freie Ecke aus $S \setminus E(M)$ kann a_1 gewählt werden; dann ist $A = \{a_1\}$, $I = \emptyset$ und $T = \{a_1\}$.

Im Schritt 3 ist $N(A) = \{b_2\} \neq \emptyset$; und es kann $y = b_2$ gewählt werden. Im Schritt 4 für diese Wahl ist $z = a_2$ und $A = \{a_1, a_2\}$, $I = \{b_2\}$ und $E(T) = \{a_1, a_2, b_2\}$, $K(T) = \{a_1 b_2, b_2 a_2\}$.

Schritt 3 wird wiederum durchgeführt. Es ist $N(A) = \{b_2, b_3\} \neq I$. Wir wählen $y = b_3$ und die Kante $a_2 b_3$. In Schritt 4 gibt es zu y eine inzidente Kante in M , nämlich $a_3 b_3$. Dann ist $A = \{a_1, a_2, a_3\}$, $I = \{b_2, b_3\}$, $E(T) = \{a_1, a_2, a_3, b_2\}$, $K(T) = \{a_1 b_2, b_2 a_2, a_3 b_2\}$. Wir gehen zurück zu Schritt 3.

Hier ist $N(A) = \{b_1, b_2, b_3\} \neq I$; als Ecke aus $N(A) \setminus I$ kann $y = b_1$ gewählt werden. Zu diesem y gibt es im Matching M keine inzidente Kante; also ist ein vergrößernder Weg gefunden.

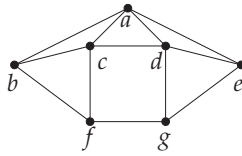


Abbildung 7.11: Der Graph für Aufgabe 19

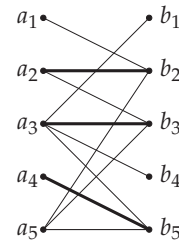


Abbildung 7.12: Der Graph für Aufgabe 21

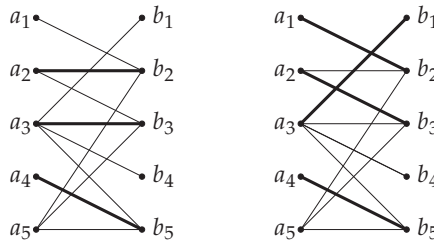


Abbildung 7.13: Ein maximales Matching für Abbildung 21, links die Ausgangssituation, rechts die Lösung

Es gibt ein neues Matching $M_1 = \{a_1b_2, a_2b_3, a_3b_1, a_4b_5\}$.

Wir sind zurück in Schritt 2 des ungarischen Algorithmus. Jetzt ist $E(M_1) = \{a_1, a_2, a_3, a_4\}$, und es gibt noch eine Ecke in S , nämlich a_5 , die nicht im Matching liegt.

Also setzen wir $a = a_5$ als freie Ecke. Es ist $I = \emptyset$ und $A = \{a_5\}$; $N(A) = \{b_2, b_3, b_5\}$. Als Ecke y kann $y = b_2$ gewählt werden. Dazu gibt es in Schritt 4 eine inzidente Kante. Also wird der Baum erweitert, es ist $A = \{a_1, a_5\}$, $I = \{b_2\}$.

Zurück in Schritt 3 ist $N(A) = \{b_2, b_5\}$. Als Ecke y wird jetzt $y = b_5$ gewählt. Dann gibt es in Schritt 4 eine in M_1 inzidente Kante mit $z = a_4$, der Baum wird erweitert; $A = \{a_1, a_4, a_5\}$, $I = \{b_2, b_5\}$.

Zurück in Schritt 3 ist $N(A) = \{b_2, b_3, b_4, b_5\}$. Als y kann $y = b_3$ gewählt werden. Auch hier gibt es in Schritt 4 eine inzidente Kante mit $z = a_2$. Dann ist $A = \{a_1, a_2, a_4, a_5\}$ und $I = \{b_2, b_3, b_5\}$.

Zurück im Schritt 3 ist $N(A) = \{b_2, b_3, b_5\} = I$. Dann wird aber die Ecke $a = a_5$ aus S entfernt, und wir gehen zurück zu Schritt 2. Dort gilt jetzt $E(M_1) = \{a_1, a_2, a_3, a_4\}$, und der ungarische Algorithmus stoppt!

22. Für den bipartiten Graphen $G(E_1 \cup E_2, K)$ mit $E_1 = \{a_1, a_2, a_3, a_4, a_5, a_6, a_7\}$, $E_2 = \{b_1, b_2, b_3, b_4, b_5, b_6, b_7, b_8\}$ und $K = \{a_1b_1, a_1b_2, a_2b_1, a_2b_2, a_3b_2, a_3b_5, a_4b_1, a_4b_2, a_5b_2, a_5b_3, a_5b_4, a_5b_6, a_5b_7, a_5b_8, a_6b_3, a_6b_4, a_6b_5, a_6b_6, a_6b_8, a_7b_5, a_7b_8\}$ ist $M = \{a_2b_1, a_4b_2, a_5b_7, a_6b_5\}$ ein gesättigtes Matching. Bestimmen Sie ein maximales Matching!

Lösung:

In Abbildung 7.14 finden Sie den bipartiten Graphen und das gegebene gesättigte Matching $M = \{a_2b_1, a_4b_2, a_5b_7, a_6b_5\}$. Bei der Durchführung des Verfahrens und auch beim Nachvollziehen der Lösung ist es angebracht, den sukzessive in den Schritten 3 und 4 aufgebauten Wurzelbaum immer aufzuzeichnen. In der angegebenen Lösung finden Sie die Bäume nach einer Entscheidung in Schritt 4.

In Schritt 2 des ungarischen Algorithmus ist $E(M) \cap E_1 = \{a_2, a_4, a_5, a_6\}$. Als freie Ecke aus $E_1 \setminus E(M)$ wird a_1 gewählt. In Schritt 3 ist $N(A) = \{b_1, b_2\} \neq \emptyset$, es kann $y = b_1$ gewählt werden. In Schritt 4 gibt es die zu $y = b_1$ inzidente Kante a_2b_1 , also ist $z = a_2$, $A = \{a_1, a_2\}$, $I = \{b_1\}$, $E(T) = \{a_1, a_2, b_1\}$, $K(T) = \{a_1b_1, b_1a_2\}$. Mit diesem Wurzelbaum gehen wir zurück zu Schritt 3.

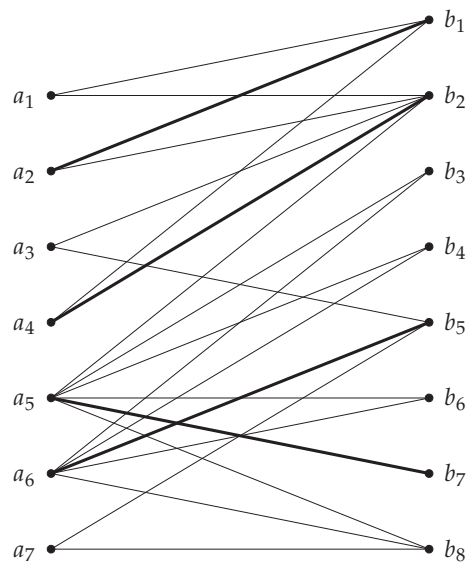


Abbildung 7.14: Der bipartite Graph für Aufgabe 22, das gegebene gesättigte Matching $\{a_2b_1, a_4b_2, a_5b_7, a_6b_5\}$ ist fett dargestellt

Es ist $N(A) = \{b_1, b_2\} \neq I$, also gibt es die Wahl $y = b_2 \notin N(A) \setminus I$. Wir wählen die Kante a_4b_2 und gehen damit zu Schritt 4. Es ist $A = \{a_1, a_2, a_4\}$ und $I = \{b_1, b_2\}$. In Abbildung 7.15 sehen Sie den dadurch entstandenen Wurzelbaum.

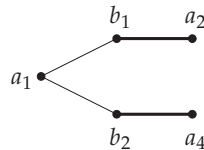


Abbildung 7.15: Der Wurzelbaum T nach dem zweimaligen Durchlaufen von Schritt 3 und 4 für den Ausgangsknoten a_1 . Die im Matching enthaltenen Kanten des Baums sind fett dargestellt.

In Schritt 3 ist $N(A) = \{b_1, b_2\} = I$. Also setzen wir $S = S \setminus \{a_1\}$ und gehen zu Schritt 2 zurück. Es gibt keinen zunehmenden Weg.

In Schritt 2 wählen wir $a = a_3$; also ist $A = \{a_3\}$ und $I = \emptyset$, $N(A) = \{b_2, b_5\}$. In Schritt 3 wählen wir $y = b_2$ und $x = a_3$. In Schritt 4 ist $z = a_4$, insgesamt kann der Wurzelbaum erweitert werden, es ist $A = \{a_3, a_4\}$, $I = \{b_2\}$ und $N(A) = \{b_1, b_2, b_5\}$.

Zurück in Schritt 3 ist $y = b_1$ und $x = a_4$. Wieder in Schritt 4 gibt es $z = a_2$, damit ist $A = \{a_2, a_3, a_4\}$, $I = \{b_1, b_2\}$ und $N(A) = \{b_1, b_2, b_5\}$.

In Schritt 3 ist $N(A) \setminus I = \{b_5\}$, also $y = b_5$ und $x = a_3$. In Schritt 4 ist $z = a_6$, also $A = \{a_2, a_3, a_4, a_6\}$, $I = \{b_1, b_2, b_5\}$ und $N(A) = \{b_1, b_2, b_3, b_4, b_5, b_6, b_8\}$.

In Schritt 3 zurück ist $N(A) \setminus I = \{b_3, b_4, b_6, b_8\}$, $y = b_3$ und $x = a_6$ ist eine mögliche Wahl. In Schritt 4 gibt es für diese Wahl kein z mit einer zu b_3 inzidenten Kante in M . Abbildung 7.16 zeigt den ausgehend von $a = a_3$ aufgebauten Wurzelbaum. Wir haben einen vergrößernden Weg gefunden, die Kante a_6b_5 kann durch die beiden Kanten a_3b_5 und a_6b_3 ersetzt werden. Es ist dann $M = \{a_2b_1, a_3b_5, a_4b_2, a_5b_7, a_6b_3\}$.

Zurück in Schritt 2 ist jetzt $E(M) \cap E_1 = \{a_2, a_3, a_4, a_5, a_6\}$, denn die Ecke a_1 steht ja nicht mehr zur Verfügung. Als neue Wurzel eines Baums steht nur noch a_7 zur Verfügung, dann ist $A = \{a_7\}$, $I = \emptyset$ und $N(A) = \{b_5, b_8\}$. In Schritt 3 wird $y = b_5$ und $x = a_7$ gewählt. Dann ist in Schritt 4 $z = a_3$ und $A = \{a_3, a_7\}$, $I = \{b_5\}$ und $N(A) = \{b_2, b_5, b_8\}$.

Zurück in Schritt 3 wird $y = b_2$ und $x = a_3$ gewählt. In Schritt 4 ist $z = a_4$ und $A = \{a_3, a_4, a_7\}$, $I = \{b_2, b_5\}$ und $N(A) = \{b_1, b_2, b_5, b_8\}$.

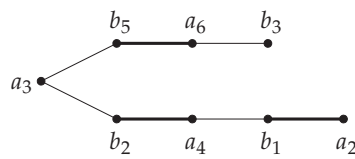


Abbildung 7.16: Der Wurzelbaum T für den Ausgangsknoten a_3 . Die im Matching enthaltenen Kanten des Baums sind fett dargestellt.

Jetzt wird in Schritt 3 $y = b_1$ und $x = a_4$ gewählt. In Schritt 4 ist dann $z = a_2$ und $A = \{a_2, a_3, a_4, a_7\}$, $I = \{b_1, b_2, b_5\}$ und $N(A) = \{b_1, b_2, b_5, b_8\}$.

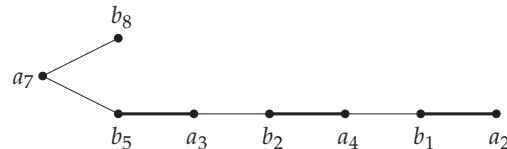


Abbildung 7.17: Der Wurzelbaum T für den Ausgangsknoten a_7 . Die im Matching enthaltenen Kanten des Baums sind fett dargestellt.

In Schritt 3 ist $y = b_8$ und $x = a_7$. In Abbildung 7.17 finden Sie den Wurzelbaum nach Hinzufügen der Kante a_7b_8 . Zu b_8 gibt es keine im aktuellen Matching inzidente Kante. Also ist a_7b_8 ein vergrößernder Weg, das dem Matching hinzugefügt werden kann.

Das Matching ist dann gegeben durch $M = \{a_2b_1, a_3b_5, a_4b_2, a_5b_7, a_6b_3, a_7b_8\}$.

Zurück in Schritt 2 ist $E(M) \cap E_1 = \emptyset$, denn die Ecke a_1 wurde ganz zu Beginn des Algorithmus gestrichen. Damit erfüllt das Matching M das Abbruchkriterium, ein maximales Matching ist gefunden! In Abbildung 7.18 ist der ganze bipartite Graph und das maximale Matching nochmals dargestellt!

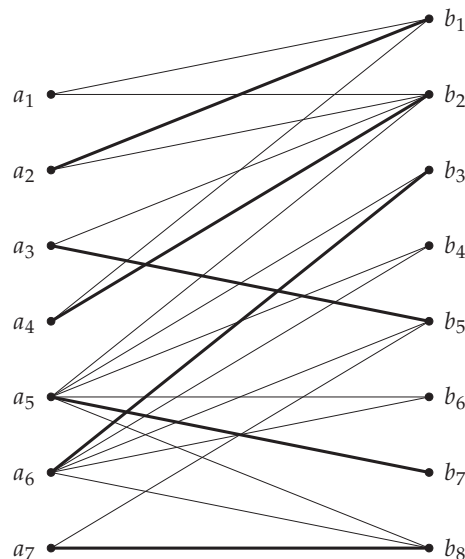


Abbildung 7.18: Der bipartite Graph für Aufgabe 22, das berechnete maximale Matching $\{a_2b_1, a_3b_5, a_4b_2, a_5b_7, a_6b_3, a_7b_8\}$ ist fett dargestellt

Kapitel 8

Algebraische Strukturen

Verständnisfragen

Sachfragen

1. Was ist eine algebraische Operation?
2. Was ist eine Halbgruppe?
3. Nennen Sie Beispiele für Halbgruppen!
4. Was ist eine Gruppe?
5. Nennen Sie Beispiele für Gruppen!
6. Was ist eine abelsche Gruppe?
7. Nennen Sie Beispiele für abelsche Gruppen!
8. Nennen Sie Gruppen, die nicht abelsch sind!
9. Nennen Sie Beispiele für endliche Gruppen!
10. Ist $(\mathbb{Z}_n \setminus \{0\}, \odot)$ immer eine Gruppe?
11. Was ist eine Untergruppe?
12. Was ist ein Homomorphismus?
13. Was ist der Kern eines Homomorphismus?
14. Was ist das Bild eines Homomorphismus?
15. Wann ist ein Homomorphismus injektiv?
16. Was ist ein Isomorphismus?
17. Haben Kern und Bild eines Homomorphismus eine algebraische Struktur?
18. Was ist die Klein'sche Vierergruppe? Warum wurde Sie eingeführt?
19. Was ist ein Ring?
20. Nennen Sie Beispiele für Ringe und für Ringe mit Einselement!

21. Was ist ein Körper?
22. Was ist ein Nullteiler? Nennen Sie Beispiele!
23. Gibt es Ringe mit Nullteilern, die trotzdem kein Körper sind?
24. Wie ist ein Polynom über einem Körper definiert?
25. Wie werten Sie Polynome auf dem Computer aus?
26. Beschreiben Sie die Addition, Multiplikation und Division von Polynomen!
27. Vergleichen Sie einen Polynomring und den Ring der ganzen Zahlen!
28. Was ist ein Linearfaktor?
29. Beschreiben Sie die Linearfaktorzerlegung eines Polynoms!
30. Wie kann die Polynomdivision über \mathbb{Z}_2 zur Fehlererkennung in der Kanalcodierung verwendet werden?
31. Was ist eine Bool'sche Algebra?
32. Nennen Sie Beispiele von Bool'schen Algebren!
33. Ist in einer Bool'schen Algebra eine Teilordnung definiert?

Methodenfragen

1. Nachweisen können, dass eine Menge mit einer gegebenen Operation eine Halbgruppe, eine Gruppe oder eine abelsche Gruppe ist.
2. Nachweisen können, dass eine Teilmenge einer Gruppe eine Untergruppe darstellt.
3. Elementare Eigenschaften in einer Gruppe nachweisen können.
4. Nachweisen können, dass eine Abbildung ein Homomorphismus ist.
5. Den Kern und das Bild eines Homomorphismus bestimmen können.
6. Nachweisen können, dass ein Homomorphismus ein Isomorphismus ist.
7. Isomorphismen zwischen zwei gegebenen Gruppen mit gleicher Anzahl von Elementen finden können.
8. Nachweisen können, dass eine Menge mit zwei gegebenen algebraischen Operationen ein Ring oder ein Körper ist.
9. Die Polynomarithmetik in einem gegebenen Körper durchführen können.
10. Die Polynomarithmetik implementieren können.
11. Die Fehlererkennung mit Hilfe von Polynomen an Beispielen durchführen können.
12. Nachweisen können, dass eine Menge mit zwei gegebenen algebraischen Operationen und einer unären Operation eine Bool'sche Algebra ist.

Übungsaufgaben

1. Weisen Sie die Gruppeneigenschaften der Mengen $\{-1, 1\} \subset \mathbb{Z}$ und $\{1, -1, i, -i\} \subset \mathbb{C}$ mit der Multiplikation als algebraische Operation nach!

Lösung:

Zuerst zu $M_1 = \{1, -1\}$.

Die Multiplikation ist abgeschlossen, denn es ist $1 \cdot 1 = (-1) \cdot (-1) = 1 \in M_1$ und $(-1) \cdot 1 = 1 \cdot (-1) = -1 \in M_1$.

Das Assoziativgesetz ist erfüllt:

$$\begin{aligned} 1 \cdot ((-1) \cdot 1) &= -1 = (1 \cdot (-1)) \cdot 1 = 1 \cdot (1 \cdot (-1)) = (1 \cdot 1) \cdot (-1); \\ (-1) \cdot ((-1) \cdot 1) &= 1 = ((-1) \cdot (-1)) \cdot 1 = (-1) \cdot (1 \cdot (-1)) = ((-1) \cdot 1) \cdot (-1). \end{aligned}$$

Das neutrale Element ist 1, wie in \mathbb{Z} . Das inverse Element für -1 ist -1 selbst wegen $(-1) \cdot (-1) = 1$. Als ist (M_1, \cdot) eine Gruppe der Ordnung 2. Sie ist auch abelsch, wie Sie beim Nachweis der Abgeschlossenheit ganz oben bereits sehen.

$M_2 = \{1, -1, i, -i\}$.

Die Multiplikation ist abgeschlossen. Für die Elemente 1 und -1 erhalten Sie die gleichen Ergebnisse wie in M_1 . Es ist $i \cdot i = -1$, so war die imaginäre Einheit gerade definiert. $i \cdot (-i) = (-i) \cdot i = 1$.

$1 \cdot a = a$ und $(-1) \cdot a = -a$ für alle $a \in M_2$, damit ist die Abgeschlossenheit nachgewiesen; und auch die Existenz des neutralen Elements e ist bewiesen.

Bleibt die Frage nach einem inversen Element für jedes $a \in M_2$ ungleich 1 zu beantworten. Das inverse Element zu -1 ist wieder -1 selbst, wie in M_1 . Das inverse Element zu i ist $-i$, denn es gilt $i \cdot (-i) = 1$; inverses Element zu $-i$ ist i .

M_2 ist abelsch, wie die bisher betrachteten Verknüpfungen beweisen. (M_2, \cdot) ist eine Gruppe der Ordnung 4; fasst man M_1 ist eine Untergruppe von M_2 .

2. Weisen Sie nach, dass für Gruppen (G_1, \circ) und (G_2, \bullet) das kartesische Produkt $G_1 \times G_2$ mit der algebraischen Operation $(x_1, y_1) \diamond (x_2, y_2) = (x_1 \circ x_2, y_1 \bullet y_2)$ eine Gruppe ist!

Lösung:

Die definierte Abbildung \diamond ist eine algebraische Operation, denn da $x_1 \circ x_2 \in G_1$ und $y_1 \bullet y_2 \in G_2$ gilt ist auch

$$(x_1, y_1) \diamond (x_2, y_2) = (x_1 \circ x_2, y_1 \bullet y_2) \in G_1 \times G_2.$$

Das Assoziativgesetz ist erfüllt, da dies in den einzelnen Komponenten erfüllt ist:

$$\begin{aligned} (x_1, y_1) \diamond ((x_2, y_2) \diamond (x_3, y_3)) &= (x_1 \circ (x_2 \circ x_3), y_1 \circ (y_2 \circ y_3)) \\ &= ((x_1 \circ x_2) \circ x_3, (y_1 \bullet y_2) \bullet y_3) \\ &= ((x_1, y_1) \diamond (x_2, y_2)) \diamond (x_3, y_3). \end{aligned}$$

Das neutrale Element ist (e_1, e_2) , falls e_1 das neutrale Element in G_1 und e_2 das in G_2 ist:

$$(x, y) \diamond (e_1, e_2) = (x \circ e_1, y \bullet e_2) = (x, y).$$

Die inversen Elemente sind gegeben als kartesisches Produkt der inversen Elemente:

$$(x, y) \diamond (x^{-1}, y^{-1}) = (x \circ x^{-1}, y \bullet y^{-1}) = (e_1, e_2).$$

3. Beweisen Sie, dass für eine Gruppe (G, \circ) und $a, b, c \in G$ $ab = ac \Rightarrow b = c$ gilt!

Lösung:

In einer Gruppe gibt es zu a ein Element a^{-1} , sodass $a \circ a^{-1} = e$ ist, wenn e das neutrale Element kennzeichnet.

Dann gilt

$$\begin{aligned} a \circ b &= a \circ c \\ \Leftrightarrow a^{-1} \circ a \circ b &= a^{-1} \circ a \circ c \\ \Leftrightarrow e \circ b &= e \circ c \\ \Leftrightarrow b &= c \end{aligned}$$

Als Widerspruchsbeweis:

Angenommen, es gibt zwei Elemente $b \neq c$ mit $a \circ b = a \circ c \Rightarrow b = c$. Dann gilt

$$b = a^{-1} \circ (a \circ b) = a^{-1} \circ (a \circ c) = c,$$

ein Widerspruch.

4. Bestimmen Sie alle Gruppen mit $n \leq 3$ Elementen durch Aufstellen der Verknüpfungstabellen!

Lösung:

Sie müssen das neutrale Element wählen und für die restlichen Elemente entscheiden, wie die inversen Elemente aussehen. In jeder Zeile und jeder Spalte muss jedes Element genau einmal auftreten, da die inversen Elemente und auch das neutrale Element eindeutig bestimmt sind.

Dann bleibt für $n = 2$ und auch für $n = 3$ nur eine Möglichkeit übrig, nämlich die Tabellen 8.1 und 8.2.

\circ	e	a
e	e	a
a	a	e

Tabelle 8.1: Die Verknüpfungstabelle der Gruppe der Ordnung 2

\circ	e	a	b
e	e	a	b
a	a	b	e
b	b	e	a

Tabelle 8.2: Die Verknüpfungstabelle der Gruppe der Ordnung 3

5. Weisen Sie nach, dass die folgenden 6 Funktionen von $\mathbb{R} \setminus \{0, 1\}$ nach $\mathbb{R} \setminus \{0, 1\}$ mit der Komposition als algebraische Operation eine Gruppe bilden: $\iota(x) = x$, $\rho(x) = 1 - \frac{1}{x}$, $\sigma(x) = \frac{1}{1-x}$, $\alpha(x) = 1 - x$, $\beta(x) = \frac{x}{x-1}$ und $\gamma(x) = \frac{1}{x}$.

Lösung:

Die Verkettung ist abgeschlossen auf der Menge $\{\iota, \rho, \sigma, \alpha, \beta, \gamma\}$, beispielsweise gilt

$$(\rho \circ \alpha)(x) = 1 - \frac{1}{\alpha(x)} = 1 - \frac{1}{1-x} = \beta(x).$$

Die Verkettung ist immer assoziativ.

Die Funktion ι ist das neutrale Element, es gilt immer $f \circ \iota = \iota \circ f = f$, was Sie nachrechnen können.

α , β und γ sind zu sich selbst invers, es gilt beispielsweise

$$(\beta \circ \beta)(x) = \frac{\beta(x)}{\beta(x) - 1} = \frac{x}{x - (x-1)} = x.$$

ρ^{-1} ist durch σ gegeben wegen

$$(\rho \circ \sigma)(x) = 1 - \frac{1}{\frac{1}{1-x}} = 1 - (1-x) = x.$$

Das inverse Element σ^{-1} ist ρ wegen

$$(\sigma \circ \rho)(x) = \frac{1}{1 - \rho(x)} = \frac{1}{1 - (1 - \frac{1}{x})} = x.$$

In Tabelle 8.3 finden Sie die komplette Verknüpfungstafel; Sie erkennen darin auch, dass die Gruppe nichtabelsch ist; beispielsweise ist $\gamma \circ \sigma = \alpha$ und $\sigma \circ \gamma = \beta$.

\circ	ι	ρ	σ	α	β	γ
ι	ι	ρ	σ	α	β	γ
ρ	ρ	σ	ι	β	γ	α
σ	σ	ι	ρ	γ	α	β
α	α	γ	β	ι	σ	ρ
β	β	α	γ	ρ	ι	σ
γ	γ	β	α	σ	ρ	ι

Tabelle 8.3: Die Verknüpfungstafel zu Aufgabe 6

6. Weisen Sie nach, dass für das gleichschenklige Dreieck in Abbildung 8.1 die folgenden Abbildungen eine Gruppe mit der Verkettung als algebraische Operation bilden: I , die Identität, R , die Rotation im Uhrzeigersinn um den Inkreismittelpunkt, die N auf L , L auf M und M auf N abbildet; S , die entgegengesetzte Rotation gegen den Uhrzeigersinn, und A , B , C die Spiegelungen um die Achsen LX , MY und NZ .

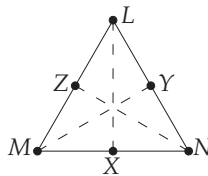


Abbildung 8.1: Ein gleichschenkliges Dreieck für Aufgabe 6

Lösung:

Die 6 Elemente stellen die möglichen Symmetrien dar; zwei Drehungen um jeweils 120° um den Inkreismittelpunkt, die Drehung um 0° , also die Identität und die drei Spiegelungen. Solche Symmetriegruppen sind der Ursprung der Gruppentheorie, man kann auch für andere geometrische Objekte wie beispielsweise das Quadrat, Fünfeck oder den Tetraeder oder Hexaeder entsprechende Symmetriegruppen bilden.

Die Verkettung der Operationen ist abgeschlossen, denn es gilt beispielsweise

$$\begin{aligned} R(L) &= N, R(M) = L, R(N) = M, \\ A(L) &= L, A(M) = N, A(N) = M, \\ A \circ R(L) &= A(N) = M, A \circ R(M) = A(L) = L, A \circ R(N) = A(M) = N. \end{aligned}$$

Dann ist offensichtlich $A \circ R = C$. Analog können Sie nachweisen, dass $R \circ A = B$. Die Komposition von Abbildungen ist immer assoziativ, in Tabelle 8.4 finden Sie alle 36 möglichen Verknüpfungen.

Die Gruppe ist nicht abelsch, auch das können Sie in der Tabelle ablesen, beispielsweise ist $A \circ R = C$ und $R \circ A = B$.

\circ	I	R	S	A	B	C
I	I	R	S	A	B	C
R	R	S	I	B	C	A
S	S	I	R	C	A	B
A	A	C	B	I	S	R
B	B	A	C	R	I	S
C	C	B	A	S	R	I

Tabelle 8.4: Die Verknüpfungstafel für Aufgabe 6

7. Weisen Sie nach, dass die Abbildung $A : (\mathbb{C} \setminus \{0\}, \cdot) \rightarrow (\mathbb{C} \setminus \{0\}, \cdot)$ mit $A(z) = |z|$ und $L : (\mathbb{R} \setminus \{0\}, \cdot) \rightarrow (\mathbb{R}, +)$ mit $L(x) = \text{ld}(x^2)$ Homomorphismen sind. Bestimmen Sie den Kern und das Bild der Homomorphismen!

Lösung:

Sowohl in \mathbb{R} als auch in \mathbb{C} gilt $|x_1 \cdot x_2| = |x_1| \cdot |x_2|$.

$N(A) = \{0\}$; $R(A) = \{z \in \mathbb{C} \mid \text{Im}(z) = 0, \text{Re}(z) \geq 0\}$.

Auch L ist ein Homomorphismus, denn mit den Rechenregeln für den Logarithmus gilt

$$\begin{aligned} L(x \cdot y) &= \text{ld}((xy)^2) \\ &= \text{ld}(x^2 y^2) \\ &= \text{ld}(x^2) + \text{ld}(y^2) \\ &= L(x) + L(y). \end{aligned}$$

L ist nicht injektiv, denn es gilt $L(x) = L(-x)$, der Kern ist gegeben durch $N(L) = \{1, -1\}$. L ist aber surjektiv, Urbilder für ein $y \in \mathbb{R}$ sind $x = \pm\sqrt{2^y}$; also ist $R(L) = \mathbb{R}$.

8. Ist (M, \circ) eine Gruppe und $x \in M$. Die *Konjugation* ist durch $\gamma_x : M \rightarrow M$, $\gamma_x(n) = x \circ n \circ x^{-1}$ definiert. Weisen Sie nach, dass γ_x ein Isomorphismus und die Menge aller Konjugationen eine Untergruppe von $S(M)$ ist!

Lösung:

Eine Konjugation ist ein Homomorphismus, denn es gilt

$$\begin{aligned} \gamma_x(m_1 \circ m_2) &= x \circ (m_1 \circ m_2) \circ x^{-1} \\ &= x \circ m_1 \circ m_2 \circ x^{-1} \\ &= x \circ m_1 \circ x^{-1} \circ x \circ m_2 \circ x^{-1} \\ &= \gamma_x(m_1) \circ \gamma_x(m_2). \end{aligned}$$

Es ist $N(\gamma_x) = \{e\}$, also ist eine Konjugation injektiv:

$$\begin{aligned} \gamma_x(m) = e &\Leftrightarrow x \circ m \circ x^{-1} = e, \\ &\Leftrightarrow m \circ x^{-1} = x^{-1} \circ e \\ &\Leftrightarrow m \circ x^{-1} = x^{-1} \\ &\Leftrightarrow m = x^{-1} \circ x \\ &\Leftrightarrow m = e. \end{aligned}$$

Das Urbild für ein $m \in M$ ist $n = x^{-1} \circ m \circ x$:

$$\gamma_x(n) = x \circ x^{-1} \circ m \circ x \circ x^{-1} = m.$$

Jetzt zur Untergruppeneigenschaft:

Die Identität ist eine Konjugation, mit $x = e$. Die Abgeschlossenheit folgt aus

$$(\gamma_x \circ \gamma_y)(n) = x \circ y \circ n \circ y^{-1} \circ x^{-1} = \gamma_{x \circ y}(n).$$

Für jede Konjugation ist die Inverse gegeben durch $\gamma_x^{-1} = \gamma_{x^{-1}}$; also ist die Menge aller Konjugationen eine Untergruppe von $S(M)$.

9. Weisen Sie nach, dass die Gruppen aus Aufgabe 5 und 6 isomorph sind!

Lösung:

Die Notation in den Tabellen 8.3 und 8.4 wurde schon so gewählt, dass man die Isomorphie direkt ablesen kann. Die Isomorphie F ist definiert durch

$$F(I) = \iota, F(R) = \rho, F(S) = \sigma, F(A) = \alpha, F(B) = \beta, F(C) = \gamma.$$

Eine weitere, zu diesen beiden Gruppen isomorphe Gruppe ist gegeben durch die sechs 2×2 Matrizen

$$i = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, r = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}, s = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}, \\ a = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, b = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, c = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix},$$

wenn die Matrixarithmetik mit modularer Arithmetik in \mathbb{Z}_2 durchgeführt wird. Auch hier ist die Isomorphie schon durch die Bezeichnungen gegeben. Und zu S_3 , der Gruppe der Permutationen auf der Menge $\{0, 1, 2\}$ finden Sie ebenfalls leicht eine Isomorphie!

10. Weisen Sie die Rechenregeln eines Rings in Satz 8.8 nach!

Lösung:

$$\forall a \in R \ a \cdot 0 = 0 \cdot a = 0:$$

Es ist $0 \cdot a = (0 + 0) \cdot a = 0 \cdot a + 0 \cdot a$, denn im Ring R ist das Distributivgesetz erfüllt. $(R, +)$ ist eine abelsche Gruppe, dort ist das neutrale Element der Addition eindeutig bestimmt, also ist $0 \cdot a = 0$. Analog beweisen Sie $a \cdot 0 = 0$.

$$\forall a, b \in R \ (-a) \cdot b = -a \cdot b = a \cdot (-b):$$

Aus $a \cdot b + (-a) \cdot b = (a + (-a)) \cdot b = 0 \cdot b = 0$ folgt $(-a) \cdot b = -a \cdot b$. Analog ergibt sich $a \cdot (-b) = -a \cdot b$.

$$\forall a, b \in R \ (-a) \cdot (-b) = a \cdot b:$$

Mit den eben nachgewiesenen Regeln ist $(-a) \cdot (-b) = -(a \cdot (-b)) = a \cdot b$.

11. Weisen Sie für den Körper mit vier Elementen und den Verknüpfungstabellen 8.7 und 8.8 auf Seite 208 die Distributivgesetze nach!

Lösung:

Die Additionstabelle 8.8 ist symmetrisch bezüglich der Diagonale. Das neutrale Element ist 0. $(R, +)$ eine abelsche Gruppe.

Die Multiplikationstabelle 8.7 zeigt, dass 1 das neutrale Element der Multiplikation ist. Das Assoziativgesetz ist ebenfalls erfüllt, was Sie exemplarisch nachprüfen sollten. Also ist (R, \cdot) eine Halbgruppe.

Bleibt das Distributivgesetz zu überprüfen. Auch das nur exemplarisch, interessant sind nur die Fälle, bei denen die Klammer nicht mit 0 oder 1 multipliziert werden. Es ist beispielsweise

$$a \cdot (1 + b) = a \cdot a = b, a \cdot 1 + a \cdot b = a + 1 = b$$

und

$$(b + 1) \cdot a = a \cdot a = b, b \cdot a + 1 \cdot a = 1 + a = b.$$

Tabelle 8.5: Eine Addition?

\diamond	a	b	c	d
a	a	b	c	d
b	b	c	d	a
c	c	d	a	b
d	d	a	b	c

Tabelle 8.6: Eine Multiplikation?

\times	a	b	c	d
a	a	a	a	a
b	a	b	c	d
c	a	c	d	b
d	a	d	b	c

12. Definieren die Verknüpfungstafeln 8.5 und 8.6 für die Menge $\{a, b, c, d\}$ einen Körper?

Lösung:

Die Tabellen definieren zwei abelsche Gruppen, bleibt, die Distributivgesetze zu überprüfen. Es gilt $c \times (c \diamond d) = c \neq c \times c \diamond c \times d = a$.

Also lautet die Antwort *Nein*.

13. Ist in den Körpern \mathbb{Z}_5 und \mathbb{Z}_7 die Gleichung $x^2 = 2$ lösbar? Wenn ja, wie sieht die Lösung aus?

Lösung:

In \mathbb{Z}_7 hat die Gleichung die beiden Lösungen $x = 3, x = 4$, denn es gilt $3 \cdot 3 = 9 \equiv 2 \pmod{7}$ und $16 \equiv 2 \pmod{7}$. In \mathbb{Z}_5 existiert keine Lösung.

14. Berechnen Sie Summe und Produkt der Polynome $p(x) = x^6 + x^3 - 1$ und $q(x) = x^4 + x^3 + x$ über \mathbb{R}, \mathbb{Z}_3 und \mathbb{Z}_2 !

Lösung:

Die Summen: Über \mathbb{R} und \mathbb{Z}_3 : $(p + q)(x) = x^6 + x^4 + 2x^3 + x - 1$; über \mathbb{Z}_2 $(p + q)(x) = x^6 + x^4 + x - 1$.

Die Produkte: Über \mathbb{R} und \mathbb{Z}_3 : $(p \cdot q)(x) = x^{10} + x^9 + 2x^7 + x^6 - x^3 - x$, über \mathbb{Z}_2 : $(p \cdot q)(x) = x^{10} + x^9 + x^6 - x^3 - x$.

15. Dividieren Sie das Polynom $p_1(x) = x^5 - 1$ mit Rest durch $q_1(x) = x^3 - 1$ und $p_2(x) = -144 - 23x^2 + 2x^4$ durch $q_2(x) = -24 - 6x + 4x^2 + x^3$.

Lösung:

$r_1(x) = x^2, s_1(x) = x^2 - 1$ und $r_2(x) = 2x - 8, s_2(x) = 9x^2 + 48x - 336$. ???

16. Berechnen Sie die Linearfaktorzerlegung des Polynoms $x^6 - 4x^4 - 4x^2 + 16$ über den Körpern \mathbb{Q}, \mathbb{R} und \mathbb{C} !

Lösung:

Nur über \mathbb{C} zerfällt das Polynom in Linearfaktoren:

$$p(x) = (x - 2)(x + 2)(x - \sqrt{2})(x + \sqrt{2})(x - i\sqrt{2})(x + i\sqrt{2}).$$

17. Setzen Sie die Matrix $\begin{pmatrix} 3 & -5 \\ 2 & 1 \end{pmatrix}$ als Variable x in das Polynom $x^2 + 2x + 5$ ein! Setzen Sie in das gleiche Polynom die Zahl 3 ein und berechnen Sie das Ergebnis in \mathbb{Z} und \mathbb{Z}_5 !

Lösung:

$\begin{pmatrix} 0 & -30 \\ 12 & -2 \end{pmatrix}$, in \mathbb{Z} : 20, in \mathbb{Z}_5 : 0.

18. Zeigen Sie, dass alle Elemente aus \mathbb{Z}_6 , wenn Sie diese in das Polynom $p(x) = x^3 - x$ einsetzen, als Ergebnis 0 ergeben. Welche Eigenschaft von \mathbb{Z}_6 verursacht dies?

Lösung:

Es gilt immer $x^3 = x$.

19. Implementieren Sie in der objektorientierten Programmiersprache Ihrer Wahl eine Klasse, die Polynome und die Polynomarithmetik, insbesondere die Polynomdivision, zur Verfügung stellt!

Lösung:

```
//-----  
// vlPolynom.java  
// Implementierung von Polynomen und der Polynomarithmetik  
// über den reellen Zahlen (Datentyp double).  
// -----  
// Autor:          Manfred Brill  
// Letzte Änderung: 22. 06. 2004  
//-----  
public class vlPolynom {  
  
    public vlPolynom() {  
        degree = 0;  
        a = new double[1];  
        a[0] = 0.0;  
    }  
  
    ///! Konstruktor mit Grad, ohne Koeffizienten  
    public vlPolynom(int n) {  
        a = new double[n+1];  
        degree = n-1;  
        for (int i=0; i<=degree; i++) a[i]=0.0;  
    }  
  
    ///! Konstruktor mit Grad und Feld mit Koeffizienten  
    public vlPolynom(int n, double[] c) {  
        a = new double[n+1];  
        degree = n-1;  
        for (int i=0; i<=degree; i++) a[i] = c[i];  
    }  
  
    ///! Kopierkonstruktor  
    public vlPolynom(vlPolynom copy) {  
        a = new double[copy.getDegree()+1];  
        degree = copy.getDegree();  
  
        for (int i=0; i<=degree; i++) a[i] = copy.getCoefficient(i);  
    }  
  
    ///! Grad zurückgeben  
    public int getDegree() {  
        return degree;  
    }  
  
    ///! Grad setzen  
    public void setDegree(int n) {  
        degree=n;  
    }  
  
    ///! Koeffizienten setzen  
    public void setCoefficients(int n, double c[]) {  
        degree = n-1;  
        a = new double[n];  
    }  
}
```

```
        for (int i=0; i<=degree; i++) a[i] = c[i];
    }

    ///! Koeffizient i setzen
    public void setCoefficient(int i, double c) {
        a[i] = c;
    }

    ///! Koeffizienten abfragen
    public void getCoefficients(double c[]) {
        for (int i=0; i<=degree; i++) c[i] = a[i];
    }

    ///! i-ter Koeffizient abfragen
    public double getCoefficient(int i) {
        return a[i];
    }

    ///! Auswerten des Polynoms
    public double eval(double x) {
        double value;

        value = a[degree+1];
        for (int i=degree; i>=0; i--) {
            value = x*value + a[i];
        }
        return value;
    }

    ///! Polynom-Addition
    public void add(vlPolynom p) {

        int i, maxdegree, mindegree, d;

        d = p.getDegree();

        if (this.degree >= d) {
            for (i=0; i<=d; i++) a[i] += p.getCoefficient(i);
        }
        else {
            double help[] = new double[degree+1];

            for (i=0; i<=degree; i++) help[i] = a[i];
            // Grad vergrößern und Koeffizienten neu anlegen

            a = new double[d+1];

            for (i=0; i<=degree; i++) a[i] = help[i] + p.getCoefficient(i);

            for (i=degree+1; i<=d; i++) a[i] = p.getCoefficient(i);
            degree = d;
        }
    }

    ///! Polynom-Multiplikation
    /** Voraussetzung: q[m] != 0;
    * Wird zur Zeit nicht überprüft!
    */
    public void divide(vlPolynom q, vlPolynom s, vlPolynom r) {
```

```
int i, j, m = q.getDegree();

double ahelp[] = new double[degree+1], qc[] = new double[m+1],
    sc[] = new double[degree - m+1], rc[] = new double[m];

// Koeffizienten kopieren
for (i=0; i<= degree; i++)
    ahelp[i] = a[i];
q.getCoefficients(qc);

for (i=degree-m; i>= 0; i--) {
    sc[i] = ahelp[m+i]/qc[m];
    for (j=m+i-1; j>=i; j--)
        ahelp[j] -= sc[i]*qc[j-i];
}

for (i=0; i<= m-1; i++)
    rc[i] = ahelp[i];

s.setCoefficients(degree-m+1, sc);
r.setCoefficients(m, rc);
}

///Polynom-Multiplikation
public vlPolynom multiply(vlPolynom q) {
    int i, j, pd, qd, deg, min;
    double sum;

    qd = q.getDegree();
    deg = degree+qd;

    vlPolynom result = new vlPolynom(deg+1);

    for (i=0; i<=deg; i++) {
        sum = 0.0;
        if (i > degree)
            min = degree;
        else
            min = i;

        for (j=0; j<=i; j++) {
            min = i-j;
            if (j <= degree && i-j <= qd)
                sum += a[j]*q.getCoefficient(i-j);
        }
        result.setCoefficient(i, sum);
    }
    return result;
}

public void print() {
    System.out.println("Ausgabe einer Instanz der Klasse vlPolynom");
    System.out.println("Grad des Polynoms: " + degree);

    for (int i=0; i <= degree; i++) {
        System.out.println("Index " + i + ": " + a[i]+" ");
    }
}
```

```
private int degree;
private double a[];
}
```

20. Zeigen Sie, dass die Menge $\mathbb{B} = \{n \in \mathbb{N} \mid n|105\}$ mit $\square = \text{ggT}$, $\boxplus = \text{kgV}$, $\mathbf{0} = 1$, $\mathbf{1} = 105$ und $\setminus x$ als größtes Element $y \in \mathbb{B}$ mit $\text{ggT}(x, y) = 1$ eine Boolesche Algebra ist! Finden Sie eine Potenzmenge $\mathbb{P}(M)$, die zu \mathbb{B} isomorph ist!

Lösung:

Die Primzahlfaktorisierung von 105 ist gegeben durch $105 = 3 \cdot 5 \cdot 7$. Also ist

$$\mathbb{B} = \{1, 3, 5, 7, 15, 21, 35, 105\}.$$

Die Axiome aus Definition 8.13 müssen nachgerechnet werden.

■ $x \boxplus \mathbf{0} = x$, $x \square \mathbf{1} = 1$:

Es ist $x \boxplus \mathbf{0} = \text{kgV}(x, 1) = x$. Die Aussage für die Multiplikation ist ebenfalls erfüllt, denn es gilt $x \square \mathbf{1} = \text{ggT}(x, 105) = x$, denn jedes $x \in \mathbb{B}$ ist ein Teiler von 105.

■ $x \boxplus \setminus x = \mathbf{1}$, $x \square \setminus x = \mathbf{0}$:

Nach der Definition ist $\setminus x = \frac{105}{x}$. Dann gilt

$$x \boxplus \setminus x = \text{kgV}\left(x, \frac{105}{x}\right) = \frac{x \cdot \frac{105}{x}}{1} = 105 = \mathbf{1}.$$

Es gilt

$$x \square \setminus x = \text{ggT}\left(x, \frac{105}{x}\right) = 1 = \mathbf{0}.$$

In \mathbb{B} sind die komplementären Teiler relativ prim!

■ $(x \boxplus y) \boxplus z = x \boxplus (y \boxplus z)$, $(x \square y) \square z = x \square (y \square z)$:

Es ist $\text{ggT}(\text{ggT}(x, y), z) = \text{ggT}(x, \text{ggT}(y, z))$ und $\text{kgV}(\text{kgV}(x, y), z) = \text{kgV}(x, \text{kgV}(y, z))$. Diese Aussagen wurden nicht im Kapitel über Zahlentheorie bewiesen; aber das sollte nicht schwer fallen. Man könnte beispielsweise einen Widerspruchsbeweis führen!

■ $x \boxplus y = y \boxplus x$, $x \square y = y \square x$:

Es ist $\text{ggT}(x, y) = \text{ggT}(y, x)$ und $\text{kgV}(x, y) = \text{kgV}(y, x)$.

■ $x \boxplus (y \square z) = x \boxplus y \square x \boxplus z$, $x \square (y \boxplus z) = x \square y \boxplus x \square z$:

Auch dies ist wieder auf Grund von zahlentheoretischen Aussagen erfüllt:

$$\text{kgV}(x, \text{ggT}(y, z)) = \text{ggT}(\text{kgV}(x, y), \text{kgV}(x, z)),$$

$$\text{ggT}(x, \text{kgV}(y, z)) = \text{kgV}(\text{ggT}(x, y), \text{ggT}(x, z)).$$

\mathbb{B} hat wie oben angegeben 8 Elemente. Dann ist diese Boolesche Algebra isomorph zu einer Algebra mit einer Potenzmenge $\mathbb{P}(M)$ einer Menge mit $|M| = 3$, beispielsweise $M = \{a, b, c\}$. Diese Boolesche Algebra ist dann gegeben durch $(\mathbb{P}(M), \cup, \cap, \setminus, \emptyset, M)$.

21. Ersetzen Sie in Aufgabe 20 die Zahl 105 durch eine Primzahl und suchen Sie eine dazu isomorphe Boolesche Algebra!

Lösung:

Die Trägermenge enthält nur 2 Elemente, die 1 und die Primzahl selbst. Diese Boolesche Algebra ist isomorph zur kleinst möglichen Booleschen Algebra mit zwei Elementen.

22. Beweisen Sie die De Morgan'sche Regel in einer Booleschen Algebra aus Satz 8.17 und die Absorptionsregel $\forall x, y \in \mathbb{B} \ x \sqcap (x \boxplus y) = x$ aus Satz 8.16!

Lösung:

Zuerst überlegen wir uns, dass das Element y mit $x \boxplus y = \mathbf{1} \wedge x \sqcap y = \mathbf{0}$ eindeutig bestimmt ist als $y = \neg x$.

Sind x und y beliebige Elemente der Algebra \mathbb{B} , die die Behauptung erfüllen, dann gilt

$$y = y \boxplus \mathbf{0} = y \boxplus x \sqcap \neg x = (y \boxplus x) \sqcap (y \boxplus \neg x).$$

und

$$(y \boxplus x) \sqcap (y \boxplus \neg x) = \mathbf{1} \sqcap (y \boxplus \neg x) = (y \boxplus \neg x).$$

Genauso gilt

$$\neg x = \neg x \boxplus \mathbf{0} = \neg x \boxplus x \sqcap \neg y.$$

Dann ist

$$\neg x \boxplus x \sqcap y = (\neg x \boxplus x) \sqcap (\neg x \boxplus y) = (x \boxplus \neg x) \sqcap (y \boxplus \neg x) = \mathbf{1} \sqcap (y \boxplus \neg x) = (y \boxplus \neg x).$$

Dann folgt aber insgesamt $y = \neg x$.

Es wird nur die erste de Morgan'sche Regel bewiesen, die zweite folgt analog. Es ist

$$\begin{aligned} (x \boxplus y) \boxplus (\neg x \sqcap \neg y) &= ((x \boxplus y) \boxplus \neg x) \sqcap ((x \boxplus y) \boxplus \neg y) \\ &= (\mathbf{1} \boxplus y) \sqcap (x \boxplus \mathbf{1}) = \mathbf{1}. \end{aligned}$$

und

$$\begin{aligned} (x \boxplus y) \sqcap (\neg x \sqcap \neg y) &= ((x \sqcap \neg x) \sqcap \neg y) \boxplus (\neg x \sqcap (y \sqcap \neg y)) \\ &= (\neg y \sqcap \mathbf{0}) \boxplus (\neg x \sqcap \mathbf{0}) = \mathbf{0}. \end{aligned}$$

Dann ist mit der Überlegung zu Beginn der Lösung $\neg(x \boxplus y) = \neg x \sqcap \neg y$.

Die Absorptionsregel $\forall x, y \in \mathbb{B} \ x \sqcap (x \boxplus y) = x$:

$$x \sqcap (x \boxplus y) = (x \boxplus \mathbf{0}) \sqcap (x \boxplus y) = x \boxplus \mathbf{0} \sqcap y = x \boxplus y \sqcap \mathbf{0} = x \boxplus \mathbf{0} = x.$$

23. Weisen Sie nach, dass in einer Booleschen Algebra $x \leq y \Leftrightarrow x \boxplus y = y$ und $x \leq y \Leftrightarrow \neg y \leq \neg x$ gelten.

Lösung:

Aus $x \boxplus y = y$ folgt

$$x \sqcap y = x \sqcap (x \boxplus y) = x$$

mit dem Absorptionsgesetz aus Satz 8.16. Dann ist $x \leq y$.

$x \leq y$ ist äquivalent zu $x \sqcap y = x$. Dann folgt

$$x \boxplus y = (x \sqcap y) \boxplus y = y,$$

wieder mit Satz 8.16.

Die Aussage $x \leq y \Leftrightarrow \neg y \leq \neg x$ folgt mit den de Morgan'schen Regeln in Satz 8.17 und der eben bewiesenen Äquivalenz:

$$x \leq y \Leftrightarrow x \sqcap y = x \Leftrightarrow x \boxplus y = y \Leftrightarrow \neg(\neg x \sqcap \neg y) = y \Leftrightarrow \neg y \sqcap \neg x = \neg y.$$

Kapitel 9

Vektoralgebra

Verständnisfragen

Sachfragen

1. Beschreiben Sie den Zusammenhang zwischen Punkten und Vektoren!
2. Wie ist die Vektoraddition definiert?
3. Wie ist die Differenz zweier Vektoren definiert?
4. Wie ist die Multiplikation eines Vektors mit einer reellen Zahl definiert?
5. Erläutern Sie die Koordinaten eines Vektors in \mathbb{R}^2 und \mathbb{R}^3 !
6. Wie sind Vektoraddition und Multiplikation mit einem Skalar in Koordinaten definiert?
7. Welche Rechenregeln für die Vektorarithmetik kennen Sie?
8. Wie ist die Addition zwischen einem Punkt und einem Vektor definiert?
9. Wie lautet die Parameterdarstellung einer Geraden?
10. Wie lautet die Parameterdarstellung einer Ebene?
11. Wie lautet die implizite Darstellung einer Geraden im \mathbb{R}^2 ?
12. Wie hängen implizite Darstellung und Parameterdarstellung einer Geraden zusammen?
13. Wie lautet die implizite Darstellung einer Ebene im \mathbb{R}^3 ?
14. Wie hängen implizite Darstellung und Parameterdarstellung einer Ebene zusammen?
15. Wie können schiefe Achsen in einem Koordinatensystem definiert werden?
16. Wie lautet die Definition des Skalarprodukts im \mathbb{R}^n ?
17. Nennen Sie Rechenregeln des Skalarprodukts!
18. Wie ist die Norm eines Vektors definiert? Welche Eigenschaften hat die Norm?
19. Wie ist der Abstand zweier Punkte definiert? Welche Eigenschaft hat der Abstand zweier Punkte?

20. Wie lautet die Cauchy-Schwarz'sche Ungleichung?
21. Wann sind zwei Vektoren im \mathbb{R}^n orthogonal?
22. Wie lautet der Satz von Pythagoras im \mathbb{R}^n ?
23. Was ist ein Normalenvektor auf einer Geraden im \mathbb{R}^n ?
24. Erläutern Sie die Orthogonalprojektion eines Vektors auf einen gegebenen Vektor \mathbf{v} !
25. Wie kann der Abstand zwischen einem Punkt und einer Geraden bestimmt werden?
26. Wie ist das Vektorprodukt definiert?
27. Nennen Sie Rechenregeln für das Vektorprodukt!
28. Wie lautet die Rechte-Hand-Regel?
29. Kann das Vektorprodukt geometrisch interpretiert werden?
30. Wie lautet die Definition des Spatprodukts?
31. Kann das Spatprodukt geometrisch interpretiert werden?
32. Erläutern Sie den Zusammenhang zwischen Vektoren, Punkten und Matrizen!
33. Formulieren Sie das Skalarprodukt im \mathbb{R}^n mit Hilfe von Matrixarithmetik!

Methodenfragen

1. Mit Vektoren rechnen können.
2. Mit Geraden und Ebenen in impliziter Darstellung und Parameterdarstellung arbeiten können!
3. Zwischen impliziter Darstellung und Parameterdarstellung umrechnen können!
4. Koordinaten eines Punkts in einer Ebene bestimmen können.
5. Skalarprodukt berechnen können.
6. Den Winkel zwischen zwei Vektoren berechnen können.
7. Überprüfen können, ob zwei Vektoren orthogonal sind.
8. Die Orthogonalprojektion eines Vektors berechnen können.
9. Abstand zwischen einem Punkt und Gerade oder Ebene berechnen können.
10. Vektorprodukt berechnen können.
11. Flächeninhalte mit Hilfe des Vektorprodukts berechnen können.
12. Spatprodukt berechnen können.
13. Volumen mit Hilfe des Spatprodukts berechnen können.
14. Feststellen können, ob zwei Geraden im \mathbb{R}^3 windschief sind.
15. Matrixarithmetik auf Vektoren anwenden können, insbesondere $A\mathbf{x}$ für eine Matrix A und einen Vektor \mathbf{x} berechnen können.

Übungsaufgaben

1. Gegeben sind die drei Vektoren $\mathbf{u} = (-3, 1, 2)$, $\mathbf{v} = (4, 0, -8)$, $\mathbf{w} = (6, -1, -4)$. Bestimmen Sie die Komponenten von $\mathbf{u} - \mathbf{v}$, $6\mathbf{u} + 2\mathbf{w}$, $-\mathbf{v} + \mathbf{u}$, $(2\mathbf{u} - 7\mathbf{w}) - (8\mathbf{w} + \mathbf{v})$. Bestimmen Sie den Vektor $\mathbf{x} \in \mathbb{R}^3$ mit $2\mathbf{u} - \mathbf{v} + \mathbf{x} = 7\mathbf{x} + \mathbf{w}$.

Lösung:

$$\mathbf{u} - \mathbf{v} = (-7, 1, 10), \quad 6\mathbf{u} + 2\mathbf{w} = (-6, 4, 4), \quad -\mathbf{v} + \mathbf{u} = (-7, 1, 10), \quad (2\mathbf{u} - 7\mathbf{w}) - (8\mathbf{w} + \mathbf{v}) = (-100, 17, 72).$$

$$\mathbf{x} = \frac{1}{3}(-8, 1, 8).$$

2. Beweisen Sie, dass es keine Skalare $\lambda_1, \lambda_2, \lambda_3$ gibt mit $\lambda_1(-2, 9, 6) + \lambda_2(-3, 2, 1) + \lambda_3(1, 7, 5) = (0, 5, 4)$.

Lösung:

Die Gleichung entspricht einem linearen Gleichungssystem

$$-2\lambda_1 - 3\lambda_2 + \lambda_3 = 0,$$

$$9\lambda_1 + 2\lambda_2 + 7\lambda_3 = 5,$$

$$6\lambda_1 + \lambda_2 + 5\lambda_3 = 4.$$

Durch Gauß-Elimination können Sie nachweisen, dass dieses lineare Gleichungssystem keine Lösung besitzt.

3. Beweisen Sie, dass bei der Definition der Vektorarithmetik die reellen durch komplexe Komponenten ersetzt werden dürfen, wenn die für \mathbb{R}^n definierten Operationen übernommen werden. Der dadurch entstehende Vektorraum heißt \mathbb{C}^n .

Lösung:

Dass die geforderten Rechengesetze erfüllt sind liegt daran, dass auch \mathbb{C} ein Körper ist. Also kann man an Stelle von \mathbb{R} einen Körper, beispielsweise \mathbb{Z}_2 oder \mathbb{Z}_p für eine Primzahl $p \in \mathbb{Z}$ verwenden.

4. Beweisen Sie $G(\mathbf{u}, \mathbf{v}) = G(\mathbf{u}', \mathbf{v}') \Leftrightarrow ((G \cap G' \neq \emptyset) \wedge (\exists \mu \in \mathbb{R} \mathbf{v}' = \mu\mathbf{v}))$.

Lösung:

\Rightarrow : es gibt Skalare mit $\mathbf{u} = \mathbf{u}' + \lambda\mathbf{v}'$, $\mathbf{u}' = \mathbf{u} + \mu\mathbf{v}$.

\Leftarrow : Es gibt genau einen Punkt im Schnitt, nennen wir den \mathbf{x} , zusammen mit der Gleichheit der Richtungsvektoren folgt die Behauptung.

5. Bestimmen Sie die Parameterdarstellung der Ebene, die die Punkte $(-4, -1, -1)$, $(-2, 0, 1)$ und $(-1, -2, 3)$ enthält!

Lösung:

$$E(\lambda, \mu) = (-4, -1, -1)^T + \lambda(2, 1, 0)^T + \mu(3, -1, 2)^T.$$

6. Zeigen Sie, dass die Geraden $G(P, \mathbf{v})$ und $G(P', \mathbf{v}')$ mit $P = (3, 4, 1)$, $\mathbf{v} = (-2, 1, -1)$, $P' = (5, 1, 7)$ und $\mathbf{v}' = (2, -1, 1)$ parallel sind!

Lösung:

Es ist $\mathbf{v} = -\mathbf{v}'$. Die Richtung ist entgegengesetzt; aber die Geraden sind parallel.

7. Beweisen Sie die Äquivalenz der Parameterdarstellung und der impliziten Darstellung einer Ebene!

Lösung:

Die implizite Darstellung einer Ebene ist gegeben durch die Gleichung

$$ax + by + cz + d = 0.$$

Die Parameterform ist gegeben durch den Punkt P und die beiden linear unabhängigen Richtungsvektoren \mathbf{v} , \mathbf{w} mit

$$P + \lambda \mathbf{v} + \mu \mathbf{w}, \lambda, \mu \in \mathbb{R}.$$

Zuerst der Übergang von einer impliziten zur Parameterdarstellung: Mindestens eine der Zahlen a , b oder c müssen ungleich Null sein. Es wird angenommen, dass $a \neq 0$.

Dann setzen wir $y = z = 0$ und lösen nach a auf:

$$ax + d = 0 \Rightarrow x = -\frac{d}{a}.$$

Dann kann als Punkt P auf der Ebene $P = (-\frac{d}{a}, 0, 0)$ verwendet werden. Ist $a = 0$, wird einfach nach einer der anderen Koordinaten aufgelöst.

Dann muss jeder Punkt

$$\left(-\frac{d}{a}, 0, 0\right) + \lambda(v_x, v_y, v_z) + \mu(w_x, w_y, w_z)$$

ein Element der Ebene sein, also die Gleichung $ax + by + cz + d = 0$ erfüllen:

$$a \cdot \left(-\frac{d}{a} + \lambda v_x + \mu w_x\right) + b\lambda v_y + b\mu w_y + c\lambda v_z + c\mu w_z + d = 0.$$

Zusammengefasst nach λ und μ :

$$\lambda(av_x + bv_y + cv_z) + \mu(aw_x + bw_y + cw_z) = 0.$$

Diese Gleichung muss für alle λ und μ erfüllt sein. Für $\lambda \neq 0, \mu = 0$ ergibt sich

$$v_x = \frac{-bv_y - cv_z}{a}.$$

Aus $\lambda = 0, \mu \neq 0$ ergibt sich

$$w_x = \frac{-bw_y - cw_z}{a}.$$

v_y und v_z sind frei wählbar, also beispielsweise

$$v_z = 1, v_y = 0 : v_x = -\frac{c}{a}$$

und

$$w_y = 1, w_z = 0 : w_x = -\frac{b}{a}.$$

Insgesamt ist die Parameterform im Fall $a \neq 0$ durch

$$P = \left(-\frac{d}{a}, 0, 0\right), \mathbf{v} = \begin{pmatrix} -\frac{c}{a} \\ 0 \\ 1 \end{pmatrix}, \mathbf{w} = \begin{pmatrix} -\frac{b}{a} \\ 1 \\ 0 \end{pmatrix}.$$

gegeben.

Ist die Parameterform einer Ebene durch

$$P + \lambda \mathbf{v} + \mu \mathbf{w}, \lambda, \mu \in \mathbb{R}$$

gegeben, dann sind die Koeffizienten der Gleichung

$$ax + by + cz + d = 0$$

gesucht. Das Vektorprodukt $\mathbf{v} \times \mathbf{w}$ steht senkrecht auf der Ebene, damit ist die Normale \mathbf{n} und die Koeffizienten a, b und c gegeben. d erhalten Sie durch Einsetzen des Ursprungs in die Gleichung $\langle \mathbf{n}, \mathbf{XP} \rangle = 0$.

8. Berechnen Sie die Skalarprodukte $\langle (3, 1, 4, -5), (2, 2, -4, -3) \rangle$ und $\langle (-1, 1, 0, 4, -3), (-2, -2, 0, 2, 1) \rangle$.

Lösung:

Es ist $\langle (3, 1, 4, -5), (2, 2, -4, -3) \rangle = 7$ und $\langle (-1, 1, 0, 4, -3), (-2, -2, 0, 2, 1) \rangle = 5$.

9. Für welche $\lambda \in \mathbb{R}$ sind die Vektoren $\mathbf{u} = (2, 1, 3)$ und $\mathbf{v} = (1, 7, \lambda)$ orthogonal?

Lösung:

Damit die beiden Vektoren orthogonal sind muss $\langle \mathbf{u}, \mathbf{v} \rangle = 0$ sein. Daraus können Sie eine Gleichung für λ aufstellen; es ist $\lambda = -3$.

10. Weisen Sie die Gleichung $\|\mathbf{x} + \mathbf{y}\|^2 + \|\mathbf{x} - \mathbf{y}\|^2 = 2\|\mathbf{x}\|^2 + 2\|\mathbf{y}\|^2$ nach. Warum trägt sie die Bezeichnung *Parallelogrammgleichung*?

Lösung:

Die Aussage lässt sich durch Nachrechnen beweisen; Sie müssen einfach die Definition der Norm durch das Skalarprodukt verwenden.

$\mathbf{x} + \mathbf{y}, \mathbf{x} - \mathbf{y}$ bilden die Diagonalen im von \mathbf{x}, \mathbf{y} aufgespannten Parallelogramm.

11. Weisen Sie nach, dass die Vektoren $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ mit $\|\mathbf{x} + \mathbf{y}\| = \|\mathbf{x} - \mathbf{y}\|$ orthogonal sind. Welche geometrische Interpretation hat diese Aussage?

Lösung:

Es ist $\|\mathbf{x} + \mathbf{y}\|^2 - \|\mathbf{x} - \mathbf{y}\|^2 = 0 \Leftrightarrow 4 \langle \mathbf{x}, \mathbf{y} \rangle = 0$.

$\mathbf{x} + \mathbf{y}$ und $\mathbf{x} - \mathbf{y}$ stellen die Diagonalen in einem Rechteck dar!

12. Bestimmen Sie für $\mathbf{x} = (3, 2, -1), \mathbf{y} = (2, 6, 7)$ die Vektoren $\mathbf{x} \times \mathbf{y}, \mathbf{x} \times (\mathbf{y} - 2\mathbf{x}), (\mathbf{x} \times \mathbf{y}) - 2\mathbf{y}$.

Lösung:

$\mathbf{x} \times \mathbf{y} = (20, -23, 14), \mathbf{x} \times (\mathbf{y} - 2\mathbf{x}) = (20, -23, 14)$ und $(\mathbf{x} \times \mathbf{y}) - 2\mathbf{y} = (16, -35, 0)$

13. Bestimmen Sie einen Vektor, der auf $(-6, 4, 2), (3, 1, 5)$ oder $(-2, 1, 5), (3, 0, -3)$ senkrecht steht!

Lösung:

Wenn Sie das Vektorprodukt bilden erhält man einen zu den Argumenten orthogonalen Vektor.

Für $(-6, 4, 2) \times (3, 1, 5)$ erhält man $(18, 36, -18)$;

für $(-2, 1, 5) \times (3, 0, -3)$ das Ergebnis $(-3, 9, -3)$.

14. Vereinfachen Sie den Ausdruck $(\mathbf{u} + \mathbf{v}) \times (\mathbf{u} - \mathbf{v})$!

Lösung:

$2\mathbf{v} \times \mathbf{u}$

15. Beweisen Sie, dass für den von den Vektoren $\mathbf{x}, \mathbf{y} \in \mathbb{R}^3$ eingeschlossenen Winkel $\tan(\theta) = \frac{\|\mathbf{x} \times \mathbf{y}\|}{\langle \mathbf{x}, \mathbf{y} \rangle}$ gilt!

Lösung:

Im Buch finden Sie die Ausdrücke für Sinus und Kosinus von θ ; wenn Sie diese dividieren erhalten Sie das Ergebnis.

16. Bestimmen Sie die Fläche des Dreiecks mit den Eckpunkten $(2, 2, 0)$, $(-1, 2, 2)$ und $(0, 4, 3)$!

Lösung:

Mit Satz 9.14 folgt $\frac{\sqrt{77}}{2}$.

17. Beschreiben Sie die Schnittbildung zwischen drei Ebenen im \mathbb{R}^3 ! Welche Lösungen erhalten Sie für die Beispiele $x_1 + x_3 - 1 = 0$, $x_3 - 1 = 0$, $x_2 - 2 = 0$ und $x_1 = 0$, $x_1 + x_3 = 0$, $x_3 = 0$?

Lösung:

Sind die Ebenen in impliziter Darstellung gegeben, dann müssen alle Punkte im Schnitt der drei Ebenen liegen. Ist $a_i x_1 + b_i x_2 + c_i x_3 + d_i = 0$ die implizite Darstellung der Ebene E_i , dann muss $Q = (q_1, q_2, q_3)$ aus dem Schnitt drei lineare Gleichungen erfüllen:

$$\begin{aligned} a_1 q_1 + b_1 q_2 + c_1 q_3 &= -d_1, \\ a_2 q_1 + b_2 q_2 + c_2 q_3 &= -d_2, \\ a_3 q_1 + b_3 q_2 + c_3 q_3 &= -d_3. \end{aligned}$$

Ob sich die drei Ebenen in einem Punkt schneiden, oder ob der Schnitt durch eine Gerade oder Ebene gegeben ist kann jetzt am Lösungsverhalten des linearen Gleichungssystems abgelesen werden.

Für das Beispiel $x_1 + x_3 - 1 = 0$, $x_3 - 1 = 0$, $x_2 - 2 = 0$ ist das lineare Gleichungssystem gegeben durch

$$\begin{pmatrix} 1 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} q_1 \\ q_2 \\ q_3 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix}.$$

Mit Hilfe der Gauss-Elimination erhalten Sie eine eindeutige Lösung. Das bedeutet, dass der Punkt $Q = (0, 2, 1)$ der Schnittpunkt der Geraden ist.

Für das Beispiel $x_1 = 0$, $x_1 + x_3 = 0$, $x_3 = 0$ ist $\mathbf{n}_2 = \mathbf{n}_1 + \mathbf{n}_3$, die Normalenvektoren sind also komplanar. Bei der Lösung des linearen Gleichungssystems können Sie x_2 frei wählen; die drei Ebenen schneiden sich in der Gerade $G(0, (0, 1, 0))$.

18. Welchen Abstand haben die Geraden $G_1((0, 0, 0), (1, 0, 0))$ und $G_2((3, 4, 1), (-2, 1, -1))$?

Lösung:

Ob zwei gegebene Geraden windschief sind oder sich in einem Punkt schneiden kann mit Hilfe des Spatprodukts entschieden werden. Das Spatprodukt der Vektoren $(3, 4, 1)$, $(1, 0, 0)$ und $(-2, 1, -1)$ ist 5; also sind die Geraden windschief.

Die beiden Punkte $X_1 \in G_1$, $X_2 \in G_2$ können durch ein lineares Gleichungssystem bestimmt werden. Es ist $X_1 = P_1 + \lambda_1 \mathbf{v}_1$, $X_2 = P_2 + \lambda_2 \mathbf{v}_2$. Die Verbindung zwischen X_1 und X_2 muss auf beiden Geraden senkrecht stehen, darauf erhält man das lineare Gleichungssystem

$$\begin{aligned} \langle \mathbf{P}_1 \mathbf{P}_2, \mathbf{v}_1 \rangle &= \lambda_1 \|\mathbf{v}_1\|^2 - \lambda_2 \langle \mathbf{v}_2, \mathbf{v}_1 \rangle, \\ \langle \mathbf{P}_1 \mathbf{P}_2, \mathbf{v}_2 \rangle &= \lambda_1 \langle \mathbf{v}_1, \mathbf{v}_2 \rangle - \lambda_2 \|\mathbf{v}_2\|^2. \end{aligned}$$

Für das Beispiel konkret

$$\begin{aligned} \lambda_1 + 2\lambda_2 &= 3, \\ -2\lambda_1 - 1 - 6\lambda_2 &= -3. \end{aligned}$$

Daraus ergibt sich die Lösung $\lambda_1 = 6$, $\lambda_2 = \frac{3}{2}$ und $X_1 = (6, 0, 0)$, $X_2 = (6, \frac{5}{2}, \frac{5}{2})$.

19. Formulieren Sie die Gauß-Elimination und die LU-Zerlegung aus Kapitel 5 mit Hilfe des euklidischen Skalarprodukts. Schreiben Sie eine Funktion, die das Skalarprodukt realisiert, oder verwenden Sie die `BLAS`, wenn sie Ihnen zur Verfügung steht. Bauen Sie das euklidische Skalarprodukt in Ihre Implementierung der LU-Zerlegung ein!

Lösung:

Die Rückwärtssubstitution war gegeben durch

$$x_i = \frac{1}{a_{ii}} \left(b_i - \sum_{k=i+1}^n a_{ik} x_k \right), i = n, n-1, \dots, 1.$$

Die darin auftretende Summe ist interpretierbar als Skalarprodukt im \mathbb{R}^{n-i} zwischen dem entsprechenden Teil des Zeilenvektors und den bisher berechneten Lösungskomponenten.

Die BLAS finden Sie beispielsweise auf www.netlib.org und als Teil des bereits verwendeten Java-Pakets *Ninja*. Das Skalarprodukt ist realisiert als `Blas.ddot`. Die entsprechenden Ausschnitte der Vektoren und Matrizen können Sie in *Ninja* durch die `section`-Funktion angeben. Dadurch ergibt sich der folgende Source-Code für die Rückwärtssubstitution:

```
private static void backSubstitution(doubleArray2D A, doubleArray1D b)
{
    // Rücksubstitution für eine gegebene obere Dreiecksmatrix
    // und eine rechte Seite. Das Ergebnis steht anschließend auf der
    // übergebenen rechten Seite!

    // Verwendet die BLAS-Routine ddot und die section-Klassenfunktionen
    // von doubleArray2D und doubleArray1D

    int i, j, n = b.size();

    // Erstes Lösungselement ohne Skalarprodukt
    b.set(n-1, b.get(n-1)/A.get(n-1, n-1));

    for (i=n-2; i>=0; i--) {
        b.set(i, (b.get(i) - Blas.ddot(b.section(new Range(i+1,n-1)),
            A.section(i, new Range(i+1,n-1))))/A.get(i,i));
    }
}
```

Der Rest des Programms verläuft analog wie auf Seite 47.

Für die komplette *LU*-Zerlegung benötigen wir eine entsprechende Vorwärtssubstitution und die eigentliche Zerlegung durch die Gauß-Elimination.

Zuerst zur Vorwärtssubstitution. Dabei wird implizit eingebaut, dass die Diagonalelemente der unteren Dreiecksmatrix nach erfolgter *LU*-Zerlegung alle 1 sind. Dann ist die Vorwärtssubstitution gegeben durch

$$x_1 = b_1, x_i = b_i - \sum_{k=1}^{i-1} a_{ik} x_k, i = 2, \dots, n.$$

Die Summe entspricht einem Skalarprodukt zwischen der ersten $i - 1$ Elementen der i -ten Zeile und den bisher berechneten $i - 1$ Elementen der Lösung. Dies kann wiederum durch die Funktion `Blas.ddot` mit der `section`-Funktion realisiert werden:

```
// Vorwärtssubstitution für eine gegebene untere Dreiecksmatrix
// und eine rechte Seite. Das Ergebnis steht anschließend auf der
// übergebenen rechten Seite! Im Feld p steht eine eventuell
// durchgeführte Zeilenvertauschung.
```

```
//
// Die Diagonale von A ist 1; diese Funktion gehört zur LU-Zerlegung!
private static void forwardSubstitution(doubleArray2D A, doubleArray1D b,
                                       intArray1D p)
{
    int i, j, n = b.size();
    double h;

    // Zeilenvertauschung in der rechten Seite
    for (i=0; i<n-1; i++) {
        if (p.get(i) != i) {
            h = b.get(i);
            b.set(i, b.get(p.get(i)));
            b.set(p.get(i), h);
        }
    }

    // Jetzt die Vorwärts-Substitution;
    // das erste Element bleibt unverändert,
    // die Diagonalelemente sind alle 1!
    for (i=1; i<n; i++) {
        b.set(i, b.get(i) -
              Blas.ddot(b.section(new Range(0, i-1)),
                       A.section(i, new Range(0, i-1))));
    }
}
}
```

Nun zur *LU*-Zerlegung. Neben dem Skalarprodukt gibt es in der BLAS eine ganze Reihe von Funktionen, die Sie sehr gewinnbringend einsetzen können. Und Sie können erwarten, dass diese Bibliothek häufig in einer optimierten Form vorliegen.

Bei der Spaltenpivot-Wahl mit Scheinskalierung werden die Absolutbeträge der Zeilenelemente der Matrix berechnet. Dazu gibt es in der BLAS die Funktion `Blas.dasum`. Damit können wir den Teil

```
summe = 0.0;
for (j=k; j<n; j++) {
    summe += Math.abs(A.get(i, j));
}
// Spaltenpivotsuche mit Scheinskalierung
q = Math.abs(A.get(i, k))/summe;
if (q > max) {
    max = q;
    p.set(k, i);
}
```

durch die Zeile

```
// Spaltenpivotsuche mit Scheinskalierung
q = Math.abs(A.get(i, k))/Blas.dasum(A.section(i, new Range(k, n-1)));
if (q > max) {
    max = q;
    p.set(k, i);
}
```

ersetzen.

Die Blas-Bibliothek enthält auch eine Funktion zum Vertauschen der Elemente zweier Vektoren, `Blas.dswap`. Dann kann die Zeilenvertauschung in der Pivotsuche

```
// Vertauschen der Zeilen, falls erforderlich
if (p.get(k) != k) {
    for (j=0; j<n; j++) {
        h = A.get(k, j);
        A.set(k, j, A.get(p.get(k), j));
        A.set(p.get(k), j, h);
    }
}
```

durch die Zeile

```
if (p.get(k) != k) {
    Blas.dswap(A.section(k, new Range(0, n-1)),
              A.section(p.get(k), new Range(0, n-1)));
}
```

ersetzt werden.

Im Eliminationsschritt finden Sie die Zeile

$$a_{ik} = a_{ik} - \frac{a_{ip_j}}{a_{p_j p_j}} a_{p_j k}$$

Dies kann durch `Blas.daxpy` realisiert werden. Diese Funktion berechnet $\mathbf{y} = \mathbf{y} + \alpha \mathbf{x}$ für zwei Vektoren \mathbf{x} und \mathbf{y} . Statt

```
// Elimination
for (i=k+1; i<n; i++) {
    A.set(i, k, A.get(i, k)/A.get(k, k));
    for (j=k+1; j<n; j++)
        A.set(i, j, A.get(i, j)-A.get(i, k)*A.get(k, j));
}
```

kann man

```
// Elimination
for (i=k+1; i<n; i++) {
    A.set(i, k, A.get(i, k)/A.get(k, k));
    Blas.daxpy(-A.get(i, k), A.section(k, new Range(k+1, n-1)),
              A.section(i, new Range(k+1, n-1)));
}
```

verwenden.

Kapitel 10

Vektorräume

Verständnisfragen

Sachfragen

1. Was ist ein allgemeiner Vektorraum?
2. Was ist ein Unterraum?
3. Erläutern Sie den Begriff der Linearkombination!
4. Was ist ein Spann von Vektoren?
5. Wann sind Vektoren linear unabhängig bzw. abhängig?
6. Erläutern Sie den Begriff der Basis!
7. Was versteht man unter Basis-Koordinaten?
8. Was ist ein Koordinatenvektor?
9. Erläutern Sie den Begriff der Dimension!
10. Was ist eine Basistransformation?
11. Wie kann die Basistransformationsmatrix gebildet werden?
12. Was ist ein Zeilenraum einer Matrix?
13. Was ist ein Spaltenraum einer Matrix?
14. Was ist der Nullraum einer Matrix?
15. Wie kann die Lösbarkeit eines linearen Gleichungssystems mit Hilfe des Rangbegriffs beschrieben werden?
16. Wie kann die allgemeine Lösung eines linearen Gleichungssystems beschrieben werden?
17. Was ist ein Skalarprodukt?
18. Was ist eine Norm?
19. Wie kann mit Hilfe des Skalarprodukts eine Norm und ein Abstands begriff definiert werden?

20. Was ist eine Orthonormalbasis?
21. Wie können die Basiskoeffizienten bezüglich einer Orthonormalbasis bestimmt werden?
22. Hat jeder n -dimensionaler Vektorraum eine Orthonormalbasis?
23. Erläutern Sie die QR -Zerlegung einer $m \times n$ -Matrix!
24. Was ist eine Householder-Transformation?
25. Wie kann mit Hilfe von Householder-Transformationen eine QR -Zerlegung berechnet werden?

Methodenfragen

1. Für eine Menge, einer Addition und einer skalaren Multiplikation die Vektorraum-Gesetze nachrechnen können.
2. Überprüfen können, ob eine Teilmenge eines Vektorraums ein Unterraum ist.
3. Gegebene Vektoren auf lineare Unabhängigkeit überprüfen können.
4. Die Basiskoordinaten eines Elements eines gegebenen Spans von Basisvektoren berechnen können.
5. Die Dimension eines Spans bestimmen können.
6. Die Basistransformationsmatrix zwischen zwei Unterräumen aufstellen können.
7. Spalten-, Zeilen- und Nullraum einer gegebenen Matrix bestimmen können.
8. Eine Basis für einen gegebenen Spann bestimmen können.
9. Die Basis des Nullraums einer Matrix bestimmen können.
10. Für einen n -dimensionalen Vektorraum eine Orthonormalbasis mit Hilfe des Gram-Schmidt-Verfahrens bestimmen können.
11. Die QR -Zerlegung einer Matrix mit Hilfe des Gram-Schmidt-Verfahrens bestimmen können.
12. Die lineare Ausgleichsrechnung mit Hilfe der QR -Zerlegung lösen können.
13. Für einen gegebenen Vektor die Householder-Transformation bilden und anwenden können.
14. Die QR -Zerlegung mit Hilfe von Householder-Transformationen anwenden und implementieren können.

Übungsaufgaben

1. Weisen Sie die Gültigkeit der Vektorraumaxiome für $F[0; 1]$ nach!

Lösung:

Die Vektorraumgesetze folgen aus den Eigenschaften der Arithmetik der reellen Zahlen und der punktweisen Definition der Vektorraum-Operationen.

2. Ist die Menge $V = \{(x, y) \in \mathbb{R}^2\}$ mit $(x, y) + (x', y') = (x + x', y + y')$, $\lambda(x, y) = (2\lambda x, 2\lambda y)$ ein Vektorraum?

Lösung:

Nein, denn weder $(\lambda\mu)(x, y) = \lambda(\mu(x, y))$ noch $1(x, y) = (x, y)$ sind erfüllt!

3. Ist die Menge $\{f \in F[0;1] \mid f(1) = 0\}$ mit den aus $F[0;1]$ übernommenen Operationen ein Vektorraum?

Lösung:

Diese Menge ist ein Vektorraum. Einerseits ist sie ein Unterraum der $F[0;1]$. Man kann auch einfach nachweisen, dass die Addition abgeschlossen ist, die skalare Multiplikation auch und die Gültigkeit der Vektorraumgesetze nachrechnen.

4. Welche der folgenden Teilmengen von $P_3[0;1]$ sind Unterräume: $U_1 = \{p \in P_3[0;1] \mid a_0 = 0\}$, $U_2 = \{p \in P_3[0;1] \mid a_0 + a_1 + a_2 + a_3 = 0\}$?

Lösung:

Beides sind Unterräume. Das Nullpolynom ist in beiden enthalten; und die beiden Vektorraumoperationen sind abgeschlossen.

Für zwei Polynome mit $a_0 = b_0 = 0$ hat auch die Summe das konstante Element 0.

Sind p und q zwei Polynome mit $p(x) = a_3x^3 + a_2x^2 + a_1x + a_0$ und $q(x) = b_3x^3 + b_2x^2 + b_1x + b_0$ mit $a_3 + a_2 + a_1 + a_0 = 0 = b_3 + b_2 + b_1 + b_0$, dann ist auch $(p+q)(x)$ in dieser Menge, denn $(a_3 + b_3) + (a_2 + b_2) + (a_1 + b_1) + (a_0 + b_0) = (a_3 + a_2 + a_1 + a_0) + (b_3 + b_2 + b_1 + b_0) = 0$. Analog können Sie die Abgeschlossenheit der skalaren Multiplikation nachweisen.

5. Sind die Vektoren $(0, 3, 1, -1)^T$, $(6, 0, 5, 1)^T$, $(4, -7, 1, 3)^T \in \mathbb{R}^4$ linear unabhängig?

Lösung:

Nein. Wenn Sie die drei Vektoren in einer 3×4 -Matrix schreiben und Gauß-Elimination durchführen, dann erhalten Sie die eine Matrix vom Rang 2. Dann können die Vektoren aber nicht unabhängig sein!

Beispielsweise erhält man aus

$$\begin{pmatrix} 4 & -7 & 1 & 3 \\ 0 & 3 & 1 & -1 \\ 6 & 0 & 5 & 1 \end{pmatrix}$$

durch Gauß-Elimination

$$\begin{pmatrix} 4 & -7 & 1 & 3 \\ 0 & 3 & 1 & -1 \\ 0 & 0 & 0 & \end{pmatrix}.$$

6. \mathbf{x}_3 sei ein Vektor eines Vektorraums, der nicht in $[\mathbf{x}_1, \mathbf{x}_2]$ liegt. Zeigen Sie, dass $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$ linear unabhängig sind!

Lösung:

Angenommen, die Vektoren wären linear abhängig. Dann könnten Sie \mathbf{x}_3 als Linearkombination darstellen, ein Widerspruch. Denn dann würde \mathbf{x}_3 ja im Spann von \mathbf{x}_1 und \mathbf{x}_2 liegen!

7. Welche der Unterräume $U_1 = \left[\begin{pmatrix} 1 \\ 1 \\ 5 \end{pmatrix}, \begin{pmatrix} 2 \\ 3 \\ 13 \end{pmatrix} \right]$, $U_2 = \left[\begin{pmatrix} 1 \\ -1 \\ -2 \end{pmatrix}, \begin{pmatrix} 3 \\ -2 \\ -3 \end{pmatrix} \right]$ und $U_3 = \left[\begin{pmatrix} 1 \\ -1 \\ -1 \end{pmatrix}, \begin{pmatrix} 4 \\ -3 \\ -1 \end{pmatrix}, \begin{pmatrix} 3 \\ -1 \\ 3 \end{pmatrix} \right]$ sind gleich?

Lösung:

Die Basis der drei Spalten kann durch Gauss-Elimination bestimmt werden. Schreiben Sie wie im Buch auf Seite 255 angegeben die gegebenen Vektoren als Spaltenvektoren in einer Matrix und führen elementare Zeilenumformungen durch. Dann können Sie die Basen der drei Unterräume ablesen.

Für U_1 und U_2 erhalten Sie die Basen $\{\mathbf{e}_1, \mathbf{e}_2\}$, für U_3 erhält man die Zeilenstufenform

$$\begin{pmatrix} 1 & 0 & -5 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{pmatrix},$$

dann ist $\{\mathbf{e}_1, \mathbf{e}_2\}$ auch eine Basis von U_3 . Dann stimmen aber alle drei Unterräume überein!

8. Für welche Werte von $a \in \mathbb{R}$ sind die Vektoren $(0, 1, a)^T$ und $(1, a, 0)^T$ linear abhängig?

Lösung:

Wenn Sie die beiden Vektoren als Spaltenvektoren einer 4×2 -Matrix schreiben und Gauss-Elimination durchführen erhalten Sie die Matrix

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}.$$

Diese hängt nicht mehr von a ab, also sind die Vektoren für alle $a \in \mathbb{R}$ linear unabhängig!

9. Welche Dimension hat der Unterraum $[p_0, p_1, p_2, p_3] \subset P_3$ mit $p_0(x) = 1$, $p_1(x) = x$, $p_2(x) = \frac{1}{2}(3x^2 - 1)$, $p_3(x) = \frac{1}{2}(5x^3 - 3x)$?

Lösung:

Sie können die Basiskoeffizienten der vier Polynome bezüglich der Monom-Basis $\{1, x, x^2, x^3\}$ ablesen und als Spaltenvektoren in einer Matrix schreiben:

$$\begin{pmatrix} 1 & 0 & -\frac{1}{2} & 0 \\ 0 & 1 & 0 & -\frac{3}{2} \\ 0 & 0 & \frac{3}{2} & 0 \\ 0 & 0 & 0 & \frac{5}{2} \end{pmatrix}.$$

Diese Matrix wird durch elementare Zeilenumformungen auf I_4 überführt. Dann sind die vier Polynome aber linear unabhängig, der Unterraum hat die Dimension 4.

10. Bestimmen Sie die Basistransformationsmatrix P für $B = \{(1, 1)^T, (2, 1)^T\}$ und $B' = \{\mathbf{e}_1, \mathbf{e}_2\}$ und die Koordinaten des Vektors $\mathbf{z} = (-3, 5)^T$ bezüglich B !

Lösung:

Es ist $\mathbf{e}_1 = -(1, 1)^T + (2, 1)^T$ und $\mathbf{e}_2 = 2(1, 1)^T - (2, 1)^T$, also lautet die Basistransformationsmatrix

$$P = \begin{pmatrix} -1 & 2 \\ 1 & -1 \end{pmatrix}.$$

Als Koordinaten des Vektors $\mathbf{z} = (-3, 5)^T$ (angegeben bezüglich den kanonischen Einheitsvektoren) erhalten Sie durch $P\mathbf{z}$ das Ergebnis $(13, -8)^T$.

11. Die kubischen Bernsteinpolynome sind definiert durch $B_0(x) = (1-x)^3$, $B_1(x) = 3x(1-x)^2$, $B_2(x) = 3x^2(1-x)$ und $B_3(x) = x^3$. Weisen Sie nach, dass diese Polynome eine Basis von $P_3[0; 1]$ sind und bestimmen Sie die Basistransformationsmatrix für den Übergang zur Monombasis und zur Basis $\{H_0^3, H_1^3, H_2^3, H_3^3\}$ der Hermite-Polynome!

Lösung:

Wenn Sie die Definition der Bernstein-Polynome ausmultiplizieren und so ablesbaren Basiskoeffizienten bezüglich der Monom-Basis $\{1, x, x^2, x^3\}$ als Spaltenvektoren in eine Matrix

schreiben, erhalten Sie die untere Dreiecksmatrix

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{pmatrix}.$$

Dann sind die vier Bernstein-Polynome aber linear unabhängig.

Durch Nachrechnen können Sie überprüfen, dass die folgenden Gleichungen gelten:

$$\begin{aligned} B_0(x) + B_1(x) + B_2(x) + B_3(x) &= 1, \\ \frac{1}{3}B_1(x) + \frac{2}{3}B_2(x) + B_3(x) &= x, \\ \frac{1}{3}B_2(x) + B_3(x) &= x^2. \end{aligned}$$

Dann können Sie (es ist $B_3(x) = x^3$) die Basistransformationsmatrix zwischen Bernstein- und Monombasis ablesen:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & \frac{1}{3} & 0 & 0 \\ 1 & \frac{2}{3} & \frac{1}{3} & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix},$$

Zwischen Hermite und Bernstein können Sie das analog durchführen, oder die Inverse der Basistransformation zwischen Hermite- und Monombasis verwenden und die beiden Matrizen multiplizieren. Man erhält die Matrix

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ \frac{4}{3} & \frac{1}{9} & 0 & 0 \\ 2 & \frac{4}{9} & \frac{1}{9} & 0 \\ 4 & 2 & \frac{4}{3} & 1 \end{pmatrix}$$

12. B und B' seien Basen eines Vektorraums, P und Q die beiden Basistransformationsmatrizen. Weisen Sie nach, dass $PQ = QP = I$ erfüllt ist!

Lösung:

$B = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ und $B' = \{\mathbf{y}_1, \dots, \mathbf{y}_n\}$ seien die Basis des n -dimensionalen Vektorraums U . Dann gilt mit den Basistransformationsmatrizen P und Q und jeden Vektor $\mathbf{u} \in U$ für die Koordinatenvektoren

$$\mathbf{u}_B = P\mathbf{u}_{B'}, \mathbf{u}_{B'} = Q\mathbf{u}_B.$$

Angenommen, es gilt $QP \neq I_n$. Dann muss es mindestens einen Vektor $\mathbf{v} \in V$ geben mit $QP\mathbf{v} \neq \mathbf{v}$.

Sind \mathbf{v}_B und $\mathbf{v}_{B'}$ die entsprechenden Koordinatenvektoren dieses Vektors, dann gilt aber

$$\mathbf{v}_{B'} = Q\mathbf{v}_B = QP\mathbf{v}_{B'};$$

ein Widerspruch. Analog kann $PQ = I_n$ nachgewiesen werden.

13. Lösen Sie das lineare Gleichungssystem $\begin{pmatrix} 1 & 6 & 4 \\ 2 & 4 & -1 \\ -1 & 2 & 5 \end{pmatrix} \mathbf{x} = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}$ und bestimmen Sie eine Basis des Nullraums der Koeffizientenmatrix!

Lösung:

Die erweiterte Koeffizientenmatrix kann durch Gauß-Elimination auf die Form

$$\begin{pmatrix} 1 & 6 & 4 & 1 \\ 0 & -8 & -9 & -1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

gebracht werden. Das System ist nicht eindeutig lösbar. Man kann $x_3 = \lambda$ setzen und erhält danach durch Rücksubstitution die allgemeine Lösung $x_1 = \frac{1}{4}(1 + 11\lambda)$ und $x_2 = \frac{1}{8}(1 - 9\lambda)$. Für $\lambda = 0$ erhält man die spezielle Lösung

$$\mathbf{x}_s = \left(\frac{1}{4}, \frac{1}{8}, 0 \right)^T.$$

Eine Basis des Nullraums kann jetzt abgelesen werden, denn es gilt $\mathbf{x} = \mathbf{x}_s + \mathbf{x}_h$ mit Lösung des homogenen Gleichungssystems mit der gleichen Koeffizientenmatrix. Dann ist

$$NR(A) = \left[\left(\frac{11}{4}, -\frac{9}{8}, 1 \right)^T \right].$$

14. Welche der beiden Matrizen ist orthogonal: $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, $\begin{pmatrix} 0 & 1 & \frac{1}{2}\sqrt{2} \\ 1 & 0 & 0 \\ 0 & 0 & \frac{1}{2}\sqrt{2} \end{pmatrix}$?

Lösung:

Die Einheitsmatrix ist orthogonal.

Die zweite Matrix hat zwar Spaltenvektoren, die alle Länge 1 besitzen. Aber das Skalarprodukt zwischen zweiter und dritter Spalte ist nicht 0, sondern $\frac{1}{2}\sqrt{2}$. Also liegt kein Orthonormalsystem vor!

15. Zeigen Sie, dass für eine Matrix A mit linear unabhängigen Spaltenvektoren und einer rechten Seite \mathbf{b} , die orthogonal zum Spaltenraum von A ist, $\mathbf{x} = \mathbf{0}$ die Ausgleichslösung von $A\mathbf{x} = \mathbf{b}$ ist.

Lösung:

Für A existiert die QR -Zerlegung. Da \mathbf{b} orthogonal zum Spaltenraum ist, gilt $Q^T \mathbf{b} = \mathbf{0}$, denn die rechte Seite steht auf allen Spalten von Q senkrecht.

R ist eine invertierbare Matrix, also hat das lineare Gleichungssystem $R\mathbf{x} = Q^T \mathbf{b} = \mathbf{0}$ nur die triviale Lösung.

16. Berechnen Sie die Ausgleichslösung von $A\mathbf{x} = \mathbf{b}$ mit $A = \begin{pmatrix} 1 & 1 \\ -1 & 1 \\ -1 & 2 \end{pmatrix}$, $\mathbf{b} = \begin{pmatrix} 7 \\ 0 \\ -7 \end{pmatrix}$.

Lösung:

Q und R können durch die Anwendung des Gram-Schmidt-Verfahrens auf die Spalten von A berechnet werden.

Man erhält

$$Q = \begin{pmatrix} \frac{1}{3}\sqrt{3} & \frac{5}{42}\sqrt{42} \\ -\frac{1}{3}\sqrt{3} & \frac{1}{42}\sqrt{42} \\ -\frac{1}{3}\sqrt{3} & \frac{2}{21}\sqrt{42} \end{pmatrix}, R = \begin{pmatrix} \sqrt{3} & -\frac{2}{3}\sqrt{3} \\ 0 & \frac{1}{3}\sqrt{42} \end{pmatrix}.$$

Es ist $Q^T \mathbf{b} = \left(\frac{14}{3}\sqrt{4}, \frac{5}{6}\sqrt{42} - \frac{2}{3}\sqrt{42} \right)^T$ und die Lösung von $R\mathbf{x} = Q^T \mathbf{b}$ ergibt sich als

$$\mathbf{x} = \left(5, \frac{1}{2} \right)^T.$$

17. Berechnen Sie die QR -Zerlegung der Matrix $A = \begin{pmatrix} 1 & 0 \\ 1 & 3 \\ 1 & 4 \\ 1 & 7 \end{pmatrix}$ mit Hilfe von Householder-Transformationen und lösen Sie damit das lineare Ausgleichsproblem mit der rechten Seite $\mathbf{b} = (1, 2, 6, 4)^T$.

Lösung:

Für den ersten Spaltenvektor $\mathbf{a}_1 = (1, 1, 1, 1)^T$ ist $\mathbf{u}_1 = (\frac{3}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2})^T$. Dann ist

$$P_1 A = \begin{pmatrix} -2 & -7 \\ 0 & \frac{2}{3} \\ 0 & \frac{5}{3} \\ 0 & \frac{14}{3} \end{pmatrix}.$$

Im nächsten Schritt wird der Spaltenvektor $(\frac{2}{3}, \frac{5}{3}, \frac{14}{3})^T$ auf den ersten kanonischen Einheitsvektor im \mathbb{R}^3 gedreht, mit $\mathbf{u}_2 = (\frac{17}{15}, \frac{1}{3}, \frac{14}{15})^T$. Es ergibt sich dann

$$P_2 A = \begin{pmatrix} 2 & -7 \\ 0 & -5 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}.$$

Im unteren Teil von A kann die Information über \mathbf{u}_1 und \mathbf{u}_2 abgelegt werden. Es fehlt zwar ein Speicherplatz, aber die beiden Vektoren können ja so abgelegt werden, dass durch eine Skalierung die erste Koordinate immer 1 ist. Dann ergibt sich nach Ablauf des Algorithmus

$$P_2 A = \begin{pmatrix} 2 & -7 \\ \frac{1}{3} & -5 \\ \frac{1}{3} & \frac{5}{17} \\ \frac{1}{3} & \frac{14}{17} \end{pmatrix}.$$

Auch $Q^T \mathbf{b}$ hätte man gleich berechnen können, durch $P_2 P_1 \mathbf{v} = (-\frac{13}{2}, \frac{5}{2}, \frac{99}{34}, -\frac{5}{34})^T$. Dann ergibt sich das lineare Gleichungssystem

$$\begin{pmatrix} -2 & -7 \\ 0 & -5 \end{pmatrix} \mathbf{x} = \begin{pmatrix} -\frac{13}{2} \\ \frac{5}{2} \end{pmatrix}$$

und die Lösung

$$\mathbf{x} = \begin{pmatrix} \frac{3}{2} \\ \frac{1}{2} \end{pmatrix}.$$

18. Implementieren Sie die QR -Zerlegung mit Householder-Transformationen in der Programmiersprache Ihrer Wahl und lösen Sie damit das Ausgleichsproblem aus den Aufgaben 16 und 17!

Lösung:

Auch bei der QR -Zerlegung kann eine Pivot-Suche eingesetzt werden, um die Stabilität zu erhöhen. Hier werden Spalten vertauscht, um sicherzustellen, dass die euklidische Norm der jeweils ersten Spalte der restlichen Matrix maximal wird. Dann bricht der Algorithmus erst dann zusammen, wenn die restliche Matrix wirklich (numerisch) die Nullmatrix ist.

Diese Spaltenpivotsuche entspricht einer Umnummerierung der Variablen, also hat man mit der Permutationsmatrix P eine Zerlegung der Form

$$QRP = A$$

berechnet.

Bei der Auflösung des linearen Ausgleichsproblems wird die QR -Zerlegung mit Spaltenpivotsuche durchgeführt. Anschließend wird auf die rechte Seite \mathbf{b} Q^T angewandt und das gestaffelte System $R\mathbf{x} = Q^T \mathbf{b}$ berechnet. Abschließend muss die Spaltenvertauschung rückgängig gemacht werden mit $P^{-1}\mathbf{x}$.

```

/*!
 * Lösung der Aufgabe 18, Kapitel 10.
*/

import com.ibm.math.array.*;    // Ninja Klassen importieren

public class qrtest {

    public static void main(String[] args) {
        int i, m = 4, n=3;
        double tol = 0.000001;

        doubleArray1D b = new doubleArray1D(m);
        doubleArray1D x = new doubleArray1D(n);
        doubleArray1D diag = new doubleArray1D(m);
        doubleArray2D A = new doubleArray2D(m,n);
        intArray1D p = new intArray1D(n);

        initArrays(m, n, A, b);

        // Die QR-Zerlegung durchführen
        QRzerlegung(A, diag, p);

        // Berechnung der Lösung des Ausgleichsproblems
        QRsol(A, diag, p, b, x);

        System.out.println("\nDie berechnete Lösung x");
        for (i=0; i<n; i++)
            System.out.println("x[" + i + "] = "+x.get(i));

        if (validate(x))
            System.out.println("\nDie Probe war erfolgreich!");
        else
            System.out.println("\nDie Probe hat nicht gepasst!");
    }

    /*!
     * Anwendung von  $Q^T$  auf einen gegebenen Vektor als Teil der
     * Auflösung eines linearen Ausgleichsproblems mit Hilfe
     * der QR-Zerlegung. Die Funktion QRzerlegung muss vorher aufgerufen
     * werden!
    */
    private static void QTb(doubleArray2D A, doubleArray1D diag,
        doubleArray1D x, doubleArray1D y)
    {
        int i, j, m = A.size(0), n = A.size(1);
        double gamma, summe;

        // Kopieren von x auf y, x wird nicht veraendert!
        Blas.dcopy(x,y);

        // Householder-Transformationen anwenden
        for (j=0; j<n; j++) {
            summe = 0.0;
            for (i=j; i<m; i++)
                summe += A.get(i, j)*y.get(i);
            gamma = summe/(diag.get(j)*A.get(j, j));
            for (i=j; i<m; i++)

```

```

        y.set(i, y.get(i)+gamma*A.get(i,j));
    }
}

/*!
 * Auflösungsfunktion für ein lineares überbestimmtes
 * Asungleichsproblem, nach Durchführung der QR-Zerlegung
 * mit der Funktion Qrzerlegung. Auf A steht das Ergebnis der
 * Zerlegung; die Diagonale von R ist auf dem Vektor diag
 * gespeichert. p enthält die durchgeführten Spaltenvertauschungen,
 * b die rechte Seite und x die berechnete Lösung.
 */
private static void QRsol(doubleArray2D A, doubleArray1D diag,
                          intArray1D p, doubleArray1D b,
                          doubleArray1D x)
{
    int i, j, m = A.size(0), n = A.size(1);
    double summe;
    doubleArray1D work = new doubleArray1D(m);

    // Q^T auf die rechte Seite anwenden mit der Funktion QTb
    QTb(A, diag, b, work);

    // Auflösen von Rx = Q^T b
    work.set(n, work.get(n)/diag.get(n));
    for (i=n-1; i>=0; i--) {
        summe = 0.0;
        for (j=i+1; j<n; j++)
            summe += A.get(i,j)*work.get(j);
        work.set(i, (work.get(i)-summe)/diag.get(i));
    }

    // Pivotsuche rückgängig machen durch Rücknummerierung
    // der berechneten Lösung
    for (i=0; i<n; i++)
        x.set(p.get(i), work.get(i));
}

/*!
 * QR-Zerlegung der Matrix A mit Householder-Transformationen.
 * Zur besseren Stabilität ist die Pivotwahl nach Businger-
 * Golub (vgl. Wilkinson/Reinsch: Linear Algebra) eingebaut.
 *
 * Die Matrix A enthaelt nach Durchlauf der Routine oberhalb
 * der Diagonale die Matrix R mit A=QR. Die Diagonale von R
 * steht im Vektor diag. Dadurch kann auf der Diagonale von A
 * und darunter die Information ueber die orthogonale Matrix Q
 * abgelegt werden. Der Vektor p enthaelt Information über
 * die Permutation der Zeilen.
 */
private static void QRzerlegung(doubleArray2D A, doubleArray1D diag,
                                intArray1D p)
{
    int i, j, k, tausch, m = A.size(0), n = A.size(1);
    double q, s, h, sigma, summe, qrkk, alphak, beta;
    doubleArray1D work = new doubleArray1D(n);

    // Den Permutationsvektor initialisieren und die Spaltensummen

```

```

// für die Pivotsuche auf work abspeichern.
for (j=0; j<n; j++) {
    summe = 0.0;
    for (i=0; i<m; i++)
        summe += A.get(i, j)*A.get(i, j);
    work.set(j, summe);

    p.set(j, j);
}

// Householder-Transformationen
for (k=0; k<n; k++) {
    sigma = work.get(k);
    tausch = k;

    // Pivotwahl nach Businger-Golub
    for (j=k+1; j<n; j++) {
        if (sigma <= work.get(j)) {
            sigma = work.get(j);
            tausch = j;
        }
    }

    // Spaltentausch durchführen
    if (tausch != k) {
        i = p.get(k);
        p.set(k, p.get(tausch));
        p.set(tausch, i);
        work.set(tausch, work.get(k));
        work.set(k, sigma);
        for (i=0; i<m; i++) {
            sigma = A.get(i, k);
            A.set(i, k, A.get(i, tausch));
            A.set(i, tausch, sigma);
        }
    }

    // Q berechnen
    // Abbruch für work(k) < eps, nicht realisiert!
    qrkk = A.get(k, k);
    if (qrkk < 0.0)
        alphak = Math.sqrt(work.get(k));
    else
        alphak = -Math.sqrt(work.get(k));

    diag.set(k, alphak);
    sigma = work.get(k) - qrkk*alphak;
    beta = 1.0/sigma;
    A.set(k, k, qrkk - alphak);

    // Orthogonalprojektion auf den Rest der Matrix anwenden
    for (j=k+1; j<n; j++) {
        sigma = 0.0;
        for (i=k; i<m; i++)
            sigma += A.get(i, k)*A.get(i, j);
        sigma *= beta;
        for (i=k; i<m; i++)
            A.set(i, j, A.get(i, j) - sigma*A.get(i, k));
    }
}

```

```

        work.set(j, work.get(j)-A.get(k,j)*A.get(k,j));
    }
}

private static void initArrays(int m, int n, doubleArray2D A,
                               doubleArray1D b) {
    int i, j;

    b.set(0, 17.0);
    b.set(1, 57.0);
    b.set(2, 121.0);
    b.set(3, 209.0);

    for (i=0; i<m; i++) A.set(i, 0, 1.0);
    A.set(0, 1, 2.0);
    A.set(1, 1, 4.0);
    A.set(2, 1, 6.0);
    A.set(3, 1, 8.0);

    A.set(0, 2, 4.0);
    A.set(1, 2, 16.0);
    A.set(2, 2, 36.0);
    A.set(3, 2, 64.0);
}

private static boolean validate(doubleArray1D x) {
    // Probe der berechneten Lösung
    boolean ok = true;
    double eps = 1e-09, error;
    int n = x.size();

    error = Math.abs(x.get(0)-1.0);
    if (error > eps) {
        ok = false;
    }
    error = Math.abs(x.get(1)-2.0);
    if (error > eps) {
        ok = false;
    }
    error = Math.abs(x.get(2)-3.0);
    if (error > eps) {
        ok = false;
    }
    return ok;
}
}

```

Die Funktion `initArrays` besetzt die rechte Seite und die 4×3 -Matrix A als

$$A = \begin{pmatrix} 1 & 2 & 4 \\ 1 & 4 & 16 \\ 1 & 6 & 36 \\ 1 & 8 & 64 \end{pmatrix}, \mathbf{b} = \begin{pmatrix} 17 \\ 57 \\ 121 \\ 209 \end{pmatrix}.$$

Die lineare Ausgleichslösung dafür ist der Vektor $\mathbf{x} = (1, 2, 3)^T$. Die Ausgabe des *Java*-Programms ist:

Die berechnete Lösung \mathbf{x}

```
x[0] = 1.000000000000000282
x[1] = 1.99999999999999976
x[2] = 3.00000000000000004
```

Die Lösungen für die Aufgaben 16 und 17 werden mit dieser Implementierung auch reproduziert; dazu kann man die Funktion `initArrays` und `validate` entsprechend verändern:

```
private static void initArrays16(doubleArray2D A, doubleArray1D b) {
    int i, j, m = A.size(0), n = A.size(1);

    b.set(0, 7.0);
    b.set(1, 0.0);
    b.set(2, -7.0);

    A.set(0,0, 1.0); A.set(1,0, -1.0); A.set(2,0, -1.0);
    A.set(0,1, 1.0); A.set(1,1,1.0); A.set(2,1,2.0);
}

private static void initArrays17(doubleArray2D A, doubleArray1D b) {
    int i, j, m = A.size(0), n = A.size(1);
    double h1 = Math.sqrt(3.0), h2 = Math.sqrt(42.0);

    b.set(0, 1.0);
    b.set(1, 2.0);
    b.set(2, 6.0);
    b.set(3, 4.0);

    for (i=0; i<m; i++) A.set(i,0,1.0);
    A.set(0,1,0.0); A.set(1,1,3.0); A.set(2,1,4.0); A.set(3,1,7.0);
}

private static boolean validate16(doubleArray1D x) {
    // Probe der berechneten Lösung
    boolean ok = true;
    double eps = 1e-09, error;

    error = Math.abs(x.get(0)-5.0);
    if (error > eps) {
        ok = false;
    }
    error = Math.abs(x.get(1)-0.5);
    if (error > eps) {
        ok = false;
    }
    return ok;
}

private static boolean validate17(doubleArray1D x) {
    // Probe der berechneten Lösung
    boolean ok = true;
    double eps = 1e-09, error;

    error = Math.abs(x.get(0)-1.5);
    if (error > eps) {
        ok = false;
    }
    error = Math.abs(x.get(1)-0.5);
    if (error > eps) {
        ok = false;
    }
}
```

```

}
return ok;
}

```

19. Berechnen Sie die Ausgleichsgerade zur Tabelle 10.1 mit Ihrer Implementierung aus Aufgabe 18!

Tabelle 10.1: Die Messwerte zu Aufgabe 19

150	150	150	200	200	200	250	250	250	300	300	300
77,4	76,7	78,2	84,1	84,5	83,7	88,9	89,2	89,7	94,2	94,7	95,9

Lösung:

Die Lösung aus Aufgabe 18 wurde leicht verändert. Denkbar wäre auch, aus dem Quelltext eine Package zu machen und die Messdaten beispielsweise einzulesen. Mit den gegebenen Messwerten wird die Matrix in `initArrays` besetzt. Bei der Berechnung der Ausgleichsgerade benötigt man eine $m \times 2$ Matrix und eine rechte Seite $\mathbf{b} \in \mathbb{R}^m$, wenn m die Anzahl der Messwerte ist. Die erste Spalte der Matrix A ist 1.0, die zweite Spalte besteht aus den gemessenen x -Werten. Die y -Werte bilden die rechte Seite \mathbf{b} . Wendet man die QR -Zerlegung aus Aufgabe 18 auf dieses Ausgleichsproblem an, dann erhält man

```

Die berechnete Lösung x
x[0] = 60.483333333333308
x[1] = 0.11533333333333454

```

Die erste Komponente der Lösung ist der Nulldurchgang, die zweite die Steigung der Ausgleichsgerade. Also lautet die Ausgleichsgerade $f(x) = 0,115333x + 60,4833$.

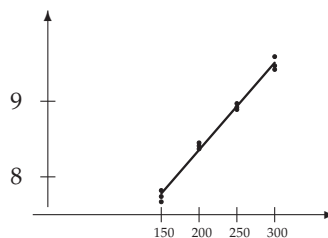


Abbildung 10.1: Die Messwerte und die berechnete Ausgleichsgerade für Aufgabe 19

Hier der Quelltext:

```

import com.ibm.math.array.*; // Ninja Klassen importieren

public class regressgerade {

    public static void main(String[] args) {
        int i, m=12, n=2;

        doubleArray1D xwerte = getXValues(m);
        doubleArray1D ywerte = getYValues(m);

        doubleArray1D x = regress(xwerte, ywerte);

        System.out.println("\nDie berechnete Lösung x");
        for (i=0; i<n; i++)
            System.out.println("x[" + i + "] = "+x.get(i));
    }
}

```

```

}

/*!
 * Zusammenfassen der Besetzung der Felder und der Lösung des
 * linearen Ausgleichproblems
 */
private static doubleArray1D regress(doubleArray1D xwerte,
                                   doubleArray1D ywerte) {
    int m = xwerte.size(), n=2;

    doubleArray2D A = new doubleArray2D(m,n);
    doubleArray1D b = new doubleArray1D(m);

    initArrays(xwerte, ywerte, A, b);

    return QR(A,b);
}

/*!
 * Zusammenfassung der QR-Zerlegung und der Auflösung zu einem
 * Funktionsaufruf.
 */
private static doubleArray1D QR(doubleArray2D A, doubleArray1D b) {
    int m = A.size(0), n = A.size(1);
    doubleArray1D x = new doubleArray1D(n);
    doubleArray1D diag = new doubleArray1D(m);
    intArray1D p = new intArray1D(n);

    // Die QR-Zerlegung durchführen
    QRzerlegung(A, diag, p);

    // Berechnung der Lösung des Ausgleichsproblems
    QRsol(A, diag, p, b, x);

    return x;
}

/*!
 * Anwendung von  $Q^T$  auf einen gegebenen Vektor als Teil der
 * Auflösung eines linearen Ausgleichsproblems mit Hilfe
 * der QR-Zerlegung. Die Funktion QRzerlegung muss vorher aufgerufen
 * werden!
 */
private static void QTb(doubleArray2D A, doubleArray1D diag,
                       doubleArray1D x, doubleArray1D y)
{
    int i, j, m = A.size(0), n = A.size(1);
    double gamma, summe;

    // Kopieren von x auf y, x wird nicht veraendert!
    Blas.dcopy(x,y);

    // Householder-Transformationen anwenden
    for (j=0; j<n; j++) {
        summe = 0.0;
        for (i=j; i<m; i++)
            summe += A.get(i, j)*y.get(i);
        gamma = summe/(diag.get(j)*A.get(j, j));
        for (i=j; i<m; i++)

```

```

        y.set(i, y.get(i)+gamma*A.get(i,j));
    }
}

/*!
 * Auflösungsfunktion für ein lineares überbestimmtes
 * Asungleichsproblem, nach Durchführung der QR-Zerlegung
 * mit der Funktion Qrzerlegung. Auf A steht das Ergebnis der
 * Zerlegung; die Diagonale von R ist auf dem Vektor diag
 * gespeichert. p enthält die durchgeführten Spaltenvertauschungen,
 * b die rechte Seite und x die berechnete Lösung.
 */
private static void QRsol(doubleArray2D A, doubleArray1D diag,
                          intArray1D p, doubleArray1D b,
                          doubleArray1D x)
{
    int i, j, m = A.size(0), n = A.size(1);
    double summe;
    doubleArray1D work = new doubleArray1D(m);

    // Q^T auf die rechte Seite anwenden mit der Funktion QTb
    QTb(A, diag, b, work);

    // Auflösen von Rx = Q^T b
    work.set(n, work.get(n)/diag.get(n));
    for (i=n-1; i>=0; i--) {
        summe = 0.0;
        for (j=i+1; j<n; j++)
            summe += A.get(i,j)*work.get(j);
        work.set(i, (work.get(i)-summe)/diag.get(i));
    }

    // Pivotsuche rückgängig machen durch Rücknummerierung
    // der berechneten Lösung
    for (i=0; i<n; i++)
        x.set(p.get(i), work.get(i));
}

/*!
 * QR-Zerlegung der Matrix A mit Householder-Transformationen.
 * Zur besseren Stabilität ist die Pivotwahl nach Businger-
 * Golub (vgl. Wilkinson/Reinsch: Linear Algebra) eingebaut.
 *
 * Die Matrix A enthaelt nach Durchlauf der Routine oberhalb
 * der Diagonale die Matrix R mit A=QR. Die Diagonale von R
 * steht im Vektor diag. Dadurch kann auf der Diagonale von A
 * und darunter die Information ueber die orthogonale Matrix Q
 * abgelegt werden. Der Vektor p enthaelt Information über
 * die Permutation der Zeilen.
 */
private static void QRzerlegung(doubleArray2D A, doubleArray1D diag,
                                intArray1D p)
{
    int i, j, k, tausch, m = A.size(0), n = A.size(1);
    double q, s, h, sigma, summe, qrkk, alphak, beta;
    doubleArray1D work = new doubleArray1D(n);

    // Den Permutationsvektor initialisieren und die Spaltensummen

```

```

// für die Pivotsuche auf work abspeichern.
for (j=0; j<n; j++) {

    // Aus work stehen die euklidischen Normen der Spaltenvektoren
    summe = 0.0;
    for (i=0; i<m; i++)
        summe += A.get(i, j)*A.get(i, j);
    work.set(j, summe);

    p.set(j, j);
}

// Householder-Transformationen
for (k=0; k<n; k++) {
    sigma = work.get(k);
    tausch = k;

    // Pivotwahl nach Businger-Golub
    for (j=k+1; j<n; j++) {
        if (sigma <= work.get(j)) {
            sigma = work.get(j);
            tausch = j;
        }
    }

    // Spaltentausch durchführen
    if (tausch != k) {
        i = p.get(k);
        p.set(k, p.get(tausch));
        p.set(tausch, i);
        work.set(tausch, work.get(k));
        work.set(k, sigma);
        for (i=0; i<m; i++) {
            sigma = A.get(i, k);
            A.set(i, k, A.get(i, tausch));
            A.set(i, tausch, sigma);
        }
    }

    // Q berechnen
    // Abbruch für work(k) < eps, nicht realisiert!
    qrkk = A.get(k, k);
    if (qrkk < 0.0)
        alphak = Math.sqrt(work.get(k));
    else
        alphak = -Math.sqrt(work.get(k));

    diag.set(k, alphak);
    sigma = work.get(k) - qrkk*alphak;
    beta = 1.0/sigma;
    A.set(k, k, qrkk-alphak);

    // Orthogonalprojektion auf den Rest der Matrix anwenden
    for (j=k+1; j<n; j++) {
        sigma = 0.0;
        for (i=k; i<m; i++)
            sigma += A.get(i, k)*A.get(i, j);

```

```
        sigma *= beta;
        for (i=k; i<m; i++)
            A.set(i, j, A.get(i, j)-sigma*A.get(i, k));
        work.set(j, work.get(j)-A.get(k, j)*A.get(k, j));
    }
}

private static void initArrays(doubleArray1D xwerte, doubleArray1D ywerte,
                             doubleArray2D A, doubleArray1D b) {

    // Besetzung der Matrix und der rechten Seite für die übergebenen
    // Messwerte (x_i, y_i)
    int i, j, m = A.size(0), n = A.size(1);
    Range spalten = new Range(0, m-1);

    b.assign(ywerte);

    A.section(spalten, 0).assign(1.0);
    A.section(spalten, 1).assign(xwerte);
}

private static doubleArray1D getXValues(int m) {
    doubleArray1D x = new doubleArray1D(m);

    x.set(0, 150.0);
    x.set(1, 150.0);
    x.set(2, 150.0);

    x.set(3, 200.0);
    x.set(4, 200.0);
    x.set(5, 200.0);

    x.set(6, 250.0);
    x.set(7, 250.0);
    x.set(8, 250.0);

    x.set(9, 300.0);
    x.set(10, 300.0);
    x.set(11, 300.0);

    return x;
}

private static doubleArray1D getYValues(int m) {
    doubleArray1D y = new doubleArray1D(m);

    y.set(0, 77.4);
    y.set(1, 76.7);
    y.set(2, 78.2);

    y.set(3, 84.1);
    y.set(4, 84.5);
    y.set(5, 83.7);

    y.set(6, 88.9);
    y.set(7, 89.2);
    y.set(8, 89.7);
}
```

```

y.set(9, 94.2);
y.set(10, 94.7);
y.set(11, 95.9);

return y;
}
}

```

20. Für exakte Eingangsgrößen x_i wurden die Ausgangsgrößen y_i wie in Tabelle 10.2 beobachtet. Als Ansatz für den Ausgleich wird $f(x) = a_1 \frac{x}{1+x} + a_2(1 - e^{-x})$ verwendet. Stellen Sie die Matrix A und die rechte Seite \mathbf{b} auf und lösen Sie das lineare Ausgleichsproblem durch eine QR-Zerlegung!

Tabelle 10.2: Die Messwerte zu Aufgabe 20

x_i	0,2	0,5	1,0	1,5	2,0	3,0
y_i	0,3	0,5	0,8	1,0	1,2	1,3

Lösung:

Der Unterschied zur Berechnung der Ausgleichsgerade besteht in der Besetzung der Matrix A für das Ausgleichsproblem. Die rechte Seite \mathbf{b} enthält wieder die gemessenen y -Werte. Würden alle Messwerte auf der angegebenen Funktion mit den Koeffizienten a_1 und a_2 liegen, dann gilt für den k -ten Messwert

$$a_1 \frac{x_k}{1+x_k} + a_2(1 - e^{-x_k}) = y_k.$$

Also ist der Vektor $\mathbf{x} = (a_1, a_2)^T$, die erste Spalte der Matrix A besteht aus den Werten $\frac{x_k}{1+x_k}$, $1 \leq k \leq m$, die zweite Spalte aus $1 - e^{-x_k}$, $1 \leq k \leq m$. Wendet man die QR-Zerlegung aus Aufgabe 18 auf dieses Ausgleichsproblem an, dann erhält man

```

Die berechnete Lösung x
x[0] = 0.38244529084477336
x[1] = 1.0391925936468365

```

$x[0]$ entspricht a_1 , die zweite Komponente a_2 . Die berechnete Ausgleichslösung ist die Funktion $f(x) = 0,382\,445 \frac{x}{1+x} + 1,039\,19(1 - e^{-x})$. In Abbildung 10.2 ist sie mit den Messwerten dargestellt.

Der Quelltext unterscheidet sich von der Lösung zu Aufgabe 19 nur durch die Besetzung der Felder, und dass die beiden Funktionen implementiert wurden. Wenn Sie darin für `basis1` die konstante Funktion $f(x) = 1$ und für `basis2` die Funktion $f(x) = x$ setzen, dann erhalten Sie die Lösung aus Aufgabe 19 zurück. Hier die Implementierung:

```

private static void initArrays(doubleArray1D xwerte, doubleArray1D ywerte,
                               doubleArray2D A, doubleArray1D b) {

    // Besetzung der Matrix und der rechten Seite für die übergebenen
    // Messwerte (x_i, y_i)
    int i, m = A.size(0);
    Range spalten = new Range(0, m-1);

    b.assign(ywerte);

    for (i=0; i<m; i++) {
        A.set(i, 0, basis1(xwerte.get(i)));

```

```
A.set(i,1,basis2(xwerte.get(i)));  
}  
}  
  
private static double basis1(double x) {  
    return x/(1.0+x);  
}  
  
private static double basis2(double x) {  
    return 1.0-Math.exp(-x);  
}
```

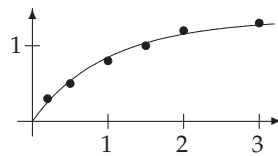


Abbildung 10.2: Die Messwerte und die berechnete Ausgleichsfunktion für Aufgabe 20

Kapitel 11

Lineare Abbildungen

Verständnisfragen

Sachfragen

1. Was ist eine lineare Abbildung?
2. Erläutern Sie den Zusammenhang zwischen Unterräumen, linearer Unabhängigkeit und linearen Abbildungen!
3. Was ist der Rang einer linearen Abbildung?
4. Wann ist eine lineare Abbildung injektiv?
5. Wann sind zwei Vektorräume isomorph?
6. Ist die Verkettung von linearen Abbildungen linear?
7. Wie lautet die Inverse der Verkettung zweier linearer Abbildungen?
8. Erläutern Sie den Zusammenhang zwischen Basisvektoren und linearen Abbildungen!
9. Wie kann die Matrixdarstellung einer linearen Abbildung aufgestellt werden?
10. Was ist ein affiner Raum?
11. Erläutern Sie den Zusammenhang zwischen Vektorräumen und affinen Räumen!
12. Was ist eine affine Kombination?
13. Ist die Summe zweier Punkte in einem affinen Raum unabhängig vom Koordinatensystem?
14. Was versteht man unter baryzentrischen Koordinaten?
15. Wann ist eine Abbildung affin?
16. Wie kann eine affine Abbildung mit Hilfe einer Matrix dargestellt werden?
17. Nennen Sie Beispiele von affinen Abbildungen!
18. Erläutern Sie den Zusammenhang zwischen linearen und affinen Abbildungen!
19. Wie lautet die lineare Abbildung, die zu einer gegebenen invertierbaren affinen Abbildung im A^3 Vektoren abbildet?

20. Ist die Verkettung affiner Abbildung kommutativ?
21. Wann ist eine quadratische Matrix diagonalisierbar?
22. Wann ist eine quadratische Matrix durch eine orthogonale Matrix diagonalisierbar?
23. Welche Eigenschaft besitzen die Eigenwerte einer symmetrischen Matrix?
24. Wie lautet die Singulärwertzerlegung einer rechteckigen Matrix?
25. Was ist eine quadratische Form?
26. Erläutern Sie den Zusammenhang zwischen quadratischen Formen und Kegelschnitten!
27. Wie kann mit Hilfe einer quadratischen Form entschieden werden, welche Form der zugehörige Kegelschnitt besitzt?
28. Was versteht man unter einer Hauptachsentransformation?

Methodenfragen

1. Überprüfen können, ob eine gegebene Abbildung linear ist.
2. Den Rang einer linearen Abbildung bestimmen können.
3. Den Kern einer linearen Abbildung bestimmen können.
4. Entscheiden können, ob eine lineare Abbildung injektiv oder surjektiv ist.
5. Eine lineare Abbildung invertieren können.
6. Die Inverse einer Verkettung zweier linearer Abbildungen bilden können.
7. Die Matrixdarstellung einer gegebenen linearen Abbildung aufstellen können!
8. Mit Punkten und Vektoren in affinen Räumen arbeiten können.
9. Konvexkombinationen und affine Kombinationen bilden können.
10. Mit homogenen Koordinaten formal umgehen können.
11. Affine Abbildungen mit homogenen Koordinaten und Matrizen anwenden können.
12. Die Verkettung affiner Abbildungen bilden können.
13. Die Verkettung affiner Abbildungen von rechts nach links lesen und anschaulich interpretieren können.
14. Eine Matrix diagonalisieren können.
15. Eine symmetrische Matrix diagonalisieren können.
16. Die Singulärwertzerlegung einer rechteckigen Matrix bestimmen können.
17. Die quadratische Form eines Kegelschnitts aufstellen können.
18. Hauptachsentransformationen für Kegelschnitte durchführen können.

Übungsaufgaben

1. Ist die Abbildung $T : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ mit $T(x_1, x_2) = (x_1 + 2x_2, 3x_1 - x_2)^T$ linear?

Lösung:

Sind $(x_1, x_2), (y_1, y_2)$ zwei Vektoren, dann gilt $T((x_1, x_2) + (y_1, y_2)) = ((x_1 + y_1) + 2(x_2 + y_2), 3(x_1 + y_1) - (x_2 + y_2))$. Dies stimmt mit $T(x_1, x_2) + T(y_1, y_2)$ überein. Analog können Sie $\lambda T(x_1, x_2) = T(\lambda x_1, \lambda x_2)$ nachweisen.

Die Matrixdarstellung ist

$$T = \begin{pmatrix} 1 & 2 \\ 3 & -1 \end{pmatrix}.$$

2. Sind die Abbildungen $T : U \rightarrow \mathbb{R}$ mit $T(\mathbf{x}) = \|\mathbf{x}\|$ und $K : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ mit $K\mathbf{x} = \mathbf{x} \times \mathbf{x}_0$ mit einem festen Vektor $\mathbf{x}_0 \in \mathbb{R}^3$ linear?

Lösung:

T ist nicht linear; sonst müsste für eine Norm in der Dreiecksungleichung immer $\|\mathbf{x} + \mathbf{y}\| = \|\mathbf{x}\| + \|\mathbf{y}\|$ gelten. Das ist aber nicht der Fall.

K ist linear, das folgt aus den Eigenschaften des Vektorprodukts.

3. $B = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\} = \{(1, 2, 1)^T, (2, 9, 0)^T, (3, 3, 4)^T\}$ ist eine Basis des \mathbb{R}^3 . Stellen Sie eine Matrixdarstellung für die lineare Abbildung $F : \mathbb{R}^3 \rightarrow \mathbb{R}^2$ mit

$$F(\mathbf{x}_1) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, F(\mathbf{x}_2) = \begin{pmatrix} -1 \\ 1 \end{pmatrix}, F(\mathbf{x}_3) = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

auf, falls in \mathbb{R}^2 die kanonischen Basisvektoren als Basis verwendet werden!

Lösung:

Sie erinnern sich, die Bilder der Basis sind die Spaltenvektoren der darstellenden Matrix:

$$A = \begin{pmatrix} 1 & -1 & 0 \\ 0 & 1 & 1 \end{pmatrix}.$$

4. Bestimmen Sie die Matrixdarstellung der Abbildung $F : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ mit $F(\mathbf{e}_1) = (1, 2, 1)^T$, $F(\mathbf{e}_2) = (1, 0, -2)^T$ bezüglich der kanonischen Basen in \mathbb{R}^2 und \mathbb{R}^3 ! Bestimmen Sie die Matrixdarstellung der gleichen Abbildung zu den Basen $B_2 = \{(1, 1)^T, (1, 2)^T\}$, $B_3 = \{(0, 1, 1)^T, (1, 0, 1)^T, (1, 1, 0)^T\}$!

Lösung:

Sie erinnern sich, die Bilder der Basis sind die Spaltenvektoren der darstellenden Matrix!

Bezüglich der kanonischen Basis ist dann die Matrix durch

$$\begin{pmatrix} 1 & 1 \\ 2 & 0 \\ 1 & -2 \end{pmatrix}$$

gegeben.

Wenn Sie die jeweiligen Basistransformationsmatrizen und Inversen bilden und entsprechend ausmultiplizieren, dann erhalten Sie bezüglich den angegebenen Basen die Matrixdarstellung.

Sie müssen zuerst von B_2 in die kanonischen Basisvektoren umrechnen, diese Matrix ist gegeben durch

$$\Phi_2 = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}.$$

Dann gehen Sie mit A nach \mathbb{R}^3 mit den kanonischen Basisvektoren; und abschließend müssen Sie diese Basiskoeffizienten in die Basis B_3 umrechnen. Diese letzte Umrechnung ist gegeben durch

$$\Phi_3^{-1} = \begin{pmatrix} -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} \end{pmatrix}.$$

Die darstellende Matrix ist dann als Matrixprodukt $\Phi_3^{-1} A \Phi_2$ gegeben:

$$B = \begin{pmatrix} -\frac{1}{2} & -2 \\ -\frac{1}{2} & -1 \\ \frac{5}{2} & 4 \end{pmatrix}.$$

5. Welche darstellende Matrix hat die Summe von zwei linearen Abbildungen $F + G$?

Lösung:

Die Spaltenvektoren der Matrix sind die Bilder der Basisvektoren. Für einen Basisvektor \mathbf{x} gilt aber $(F + G)(\mathbf{x}) = F(\mathbf{x}) + G(\mathbf{x})$.

Dann ist die darstellende Matrix als Summe der darstellenden Matrizen gegeben.

6. Beweisen Sie, dass eine 2×2 Matrix A genau dann orthogonal ist, wenn sie die Form $\begin{pmatrix} \cos(\varphi) & -\sin(\varphi) \\ \sin(\varphi) & \cos(\varphi) \end{pmatrix}$ oder $\begin{pmatrix} \cos(\varphi) & \sin(\varphi) \\ -\sin(\varphi) & \cos(\varphi) \end{pmatrix}$ aufweist!

Lösung:

„ \Leftarrow “:

Es ist $-\cos(\varphi)\sin(\varphi) + \sin(\varphi)\cos(\varphi) = 0$ und $-\cos(\varphi)\sin(\varphi) + (-\sin(\varphi))(-\cos(\varphi)) = 0$. Die Längen der Spaltenvektoren für beide angegebenen Matrizen sind immer 1, da $\cos^2(\varphi) + \sin^2(\varphi) = 1$ ist.

„ \Rightarrow “:

Die Spaltenvektoren der orthogonalen Matrix $A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$ müssen die Länge 1 haben. Also können Sie die Vektoren in Polarkoordinaten darstellen, es ist $r = 1$.

Der Winkel φ ist gegeben durch

$$\varphi = \begin{cases} \arccos(a_{11}), & a_{21} \geq 0, \\ -\arccos(a_{11}), & a_{21} < 0. \end{cases}$$

Für $a_{21} \geq 0$ ist dann die erste Spalte gegeben durch $a_{11} = \cos(\varphi)$, $a_{21} = \sin(\varphi)$. Im Fall $a_{21} < 0$ ist die erste Spalte gegeben durch $a_{11} = \cos(-\varphi) = \cos(\varphi)$ und $a_{21} = \sin(-\varphi) = -\sin(\varphi)$.

Auch für die zweite Spalte muss es analog einen Winkel ψ geben, so dass der Spaltenvektor mit Hilfe von Polarkoordinaten entsprechend dargestellt werden kann. Darüberhinaus soll die Matrix orthogonale Spaltenvektoren haben. Dann muss aber $\psi = \varphi \pm 90^\circ$ erfüllt sein!

Im Fall $\psi = \varphi + 90^\circ$ gilt dann $\cos(\varphi + 90^\circ) = -\sin(\varphi)$ und $\sin(\varphi + 90^\circ) = \cos(\varphi)$, für $\psi = \varphi - 90^\circ$ entsprechend $\cos(\varphi - 90^\circ) = \sin(\varphi)$ und $\sin(\varphi - 90^\circ) = -\cos(\varphi)$.

7. Geben Sie eine affine Abbildung an, die das Dreieck mit den Eckpunkten $(0,0)$, $(2,1)$, $(3,0)$ auf ein rechtwinkliges Dreieck mit rechtem Winkel im Punkt $(1,1)$ abbildet!

Lösung:

In einem ersten Schritt kann eine Scherung bestimmt werden, die das gegebene Dreieck auf ein Dreieck abbildet, das in $(0,0)$ einen rechten Winkel hat. Anschließend wird einfach eine Translation hinzugefügt.

Die Scherung hat die Matrixdarstellung

$$SH = \begin{pmatrix} 1 & \sigma_1 & 0 \\ \sigma_2 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Der Punkt $(2,1)$ wird auf einen Punkt auf der y -Achse abbilden, dann muss also

$$2 + \sigma_1 = 0$$

erfüllt sein. Daraus ergibt sich $\sigma_1 = -2$. Der Punkt $(3,0)$ muss gleichzeitig auf einen Punkt auf der x -Achse abgebildet werden, daraus ergibt sich

$$3\sigma_2 = 0$$

und $\sigma_2 = 0$. Dann ist die Scherung gegeben durch

$$\begin{pmatrix} 1 & -2 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Die Translation um $\mathbf{t} = (1,1)^T$ kann noch hinzugefügt werden, insgesamt ergibt sich die affine Abbildung

$$\begin{pmatrix} 1 & -2 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}.$$

8. Eine Reflexion an der y -Achse im A^3 ist gegeben durch die 4×4 -Matrix

$$Ref_y = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Stellen Sie die Matrix-Darstellung der Reflexion an einer beliebigen im Raum liegenden Linie, die durch den Ursprung geht, auf!

Lösung:

Durch eine Koordinatentransformation wird die Gerade durch den Ursprung auf die y -Achse abgebildet; dann die Reflexion durchgeführt und anschließend wieder die Koordinatentransformation rückgängig gemacht. Die Koordinatentransformation beschreiben wir durch die beiden Winkel φ und θ . φ beschreibt den Winkel zwischen der y -Achse und der Achse in der xy -Ebene, θ den Winkel zwischen y -Achse und der Achse in der yz -Ebene. Dann ist die allgemeine Reflexion an einer Achse gegeben durch $R_x(-\theta)R_z(\varphi)Ref_xR_z(-\varphi)R_x(-\theta)$:

$$Ref_y = \begin{pmatrix} \cos \theta^2 \cos 2\varphi - \sin \theta^2 & \cos \theta \sin 2\varphi & -\sin 2\theta \sin \varphi^2 \\ \cos \theta \sin 2\varphi & -\cos 2\varphi & \sin \theta \sin 2\varphi \\ -\sin 2\theta \sin \varphi^2 & \sin \theta \sin 2\varphi & \cos \theta^2 + \sin \theta^2 \cos 2\varphi \end{pmatrix}$$

Setzen Sie $\theta = \varphi = 0$, dann erhalten Sie Ref_y zurück. Interessant ist der Fall $\theta = 0$, einer Reflexion an einer Achse in der yz -Ebene:

$$Ref_{(\theta=0)} = \begin{pmatrix} \cos(2\varphi) & \sin(2\varphi) & 0 \\ \sin(2\varphi) & -\cos(2\varphi) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

9. Leiten Sie die Matrix-Darstellung der Rotation um eine beliebige Achse im Raum mit Hilfe von Kugelkoordinaten und Matrix-Algebra her!

Lösung:

Analog zur Lösung von Aufgabe 8 wird die Drehachse durch die beiden Winkel φ, θ beschrieben. Also wird die Drehachse auf eine Koordinatenachse transformiert, dann die Drehung durchgeführt und anschließend wieder zurückgedreht. Entscheidet man sich für die x -Achse, dann benötigt man den Winkel φ in der xy -Ebene und den Winkel θ in der xz -Ebene. Die Drehung der Achse auf die x -Achse ist dann durch die Rotationen $R_y(-\theta)$ und $R_z(-\varphi)$ gegeben. Dann ist die allgemeine Rotation mit Winkel β gegeben durch

$$R_y(\theta)R_z(\varphi)R_x(\beta)R_z(-\varphi)R_y(-\theta).$$

Haben Sie sich dazu entschieden, die Drehachse auf die y oder z -Achse zu drehen, ergibt sich ein anderer Ausdruck, aber das gleiche Endergebnis.

10. Gegeben ist eine beliebige Kombination von Punkten in einem affinen Raum: $P = \sum_{i=1}^n \lambda_i X_i$. Weisen Sie nach, dass P ein Punkt ist im Fall $\sum \lambda_i = 1$ und ein Vektor im Fall $\sum \lambda_i = 0$! Hat die Kombination einen Sinn für andere Werte der Summe $\sum \lambda_i$?

Lösung:

Es gilt

$$\sum_{i=1}^n \lambda_i X_i = \begin{pmatrix} \sum_{i=1}^n \lambda_i x_i \\ \sum_{i=1}^n \lambda_i \end{pmatrix}.$$

falls die Punkte gegeben sind durch $X_i = (x_i, 1)^T$. Für $\sum \lambda_i = 1$ erhält man einen Punkt für die affine Kombination. Ist die Summe $\sum \lambda_i = 0$, dann ist das Ergebnis ein Vektor. Hat die Summe einen Wert $\sum \lambda_i \neq 0$, dann kann durch diesen Wert dividiert werden; das Ergebnis ist ein Punkt.

11. Diagonalisieren Sie die Matrizen $\begin{pmatrix} 3 & 1 \\ 1 & 3 \end{pmatrix}$, $\begin{pmatrix} -2 & 0 & -36 \\ 0 & -3 & 0 \\ -36 & 0 & -23 \end{pmatrix}$, $\begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix}$ und $\begin{pmatrix} -7 & 24 & 0 & 0 \\ 24 & 7 & 0 & 0 \\ 0 & 0 & -7 & 24 \\ 0 & 0 & 24 & 7 \end{pmatrix}$!

Lösung:

Die Eigenwerte von $\begin{pmatrix} 3 & 1 \\ 1 & 3 \end{pmatrix}$ sind gegeben durch 2 und 4. Der Eigenvektor zu $\lambda_1 = 4$ ist gegeben durch $\frac{1}{2}\sqrt{2}(1, 1)^T$, der Eigenvektor zu $\lambda_2 = 2$ ist $\frac{1}{2}\sqrt{2}(-1, 1)^T$. Also ist die Matrix diagonalisierbar, sie ist ähnlich zu $diag(4, 2)$.

Auch $\begin{pmatrix} -2 & 0 & -36 \\ 0 & -3 & 0 \\ -36 & 0 & -23 \end{pmatrix}$ ist diagonalisierbar, sie hat die Eigenwerte $\lambda_1 = -50$, $\lambda_2 = -3$ und $\lambda_3 = 25$ mit den Eigenvektoren $(3, 0, 4)^T$, $(0, 1, 0)^T$ und $(-4, 0, 3)^T$.

$\begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix}$ hat die Eigenwerte $\lambda_1 = 0$, $\lambda_2 = 3$ mit den linear unabhängigen Eigenvektoren $(1, 1, 1)^T$, $(-1, 0, 1)^T$ und $(-1, 1, 0)^T$. Dann ist sie ähnlich zu $diag(0, 3, 3)$.

$\begin{pmatrix} -7 & 24 & 0 & 0 \\ 24 & 7 & 0 & 0 \\ 0 & 0 & -7 & 24 \\ 0 & 0 & 24 & 7 \end{pmatrix}$ hat die Eigenwerte $\lambda_1 = -25$ und $\lambda_2 = 25$ mit den linear unabhängigen Eigenvektoren $(0, 0, -4, 3)^T$, $(-4, 3, 0, 0)^T$, $(0, 0, 3, 4)^T$ und $(3, 4, 0, 0)^T$. Die Matrix ist dann ähnlich zu $diag(-25, -25, 25, 25)$.

12. Diagonalisieren Sie die Matrix $A = I_3 - \mathbf{v}\mathbf{v}^T$ für $\mathbf{v} = (1, 0, 1)^T \in \mathbb{R}^3$!

Lösung:

Die Matrix ist gegeben durch

$$A = \begin{pmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{pmatrix}.$$

A hat die Eigenwerte $\lambda_1 = -1$ und $\lambda_2 = 1$ mit den linear unabhängigen Eigenvektoren $(1, 0, 1)^T$, $(-1, 0, 1)^T$ und $(0, 1, 0)^T$.

Dann ist A ähnlich zu $\text{diag}(-1, 1, 1)$.

13. Beweisen Sie, dass für jeden Vektor $\mathbf{v} \in \mathbb{R}^n$, $\mathbf{v} \neq \mathbf{0}$, die Matrix $I_n - \beta\mathbf{v}\mathbf{v}^T$ diagonalisierbar ist!

Lösung:

Der Vektor $\mathbf{v} \neq \mathbf{0}$ ist ein Eigenvektor der Abbildung, denn es gilt

$$(I_n - \beta\mathbf{v}\mathbf{v}^T)\mathbf{v} = (1 - \beta\|\mathbf{v}\|^2)\mathbf{v}.$$

Jeder zu \mathbf{v} orthogonale Vektor \mathbf{x} wird auf $\mathbf{0}$ abgebildet, denn es ist dann $\mathbf{v}^T\mathbf{x} = 0$. Zu \mathbf{v} können Sie mit dem Gram-Schmidt-Verfahren $n - 1$ orthogonale und linear unabhängige Vektoren bilden. Diese spannen den Kern der Abbildung auf und gehören zum Eigenwert 0. Dann ist die Abbildung aber diagonalisierbar!

14. Gegeben ist die Diagonalmatrix $D = \text{diag}(-1, 3, 3)$, $\mathbf{v}_1 = (2, -1, 1)^T$, $V_2 = \begin{pmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}$ und $(W_1, W_2)^T = (\mathbf{v}_1 | V_2)^{-1}$. Weisen Sie nach, dass $-\mathbf{v}_1 W_1 + 3V_2 W_2$ zu D ähnlich ist!

Lösung:

Der Ausdruck ergibt die Matrix

$$\begin{pmatrix} -1 & 4 & 4 \\ 2 & 1 & -2 \\ -2 & 2 & 5 \end{pmatrix},$$

Diese Matrix hat die Eigenwerte $\lambda_1 = -1$ und $\lambda_2 = 3$. Die Eigenvektoren $(2, -1, 1)^T$, $(1, 0, 1)^T$ und $(1, 1, 0)^T$ sind linear unabhängig, dann ist A ähnlich zu D .

15. Bestimmen Sie die zugehörige quadratische Form für $x^2 + y^2 - z^2 + 4xy - 5yz$.

Lösung:

Für $n = 3$ ist eine quadratische Form für die Matrix $A = \begin{pmatrix} a & d & e \\ d & b & f \\ e & f & c \end{pmatrix}$ identisch mit einem Ausdruck $ax^2 + by^2 + cz^2 + 2dxy + 2exz + 2fyz$.

Dann ist

$$A = \begin{pmatrix} 1 & 2 & 0 \\ 2 & 1 & -\frac{5}{2} \\ 0 & -\frac{5}{2} & -1 \end{pmatrix}.$$

16. Wenden Sie die Hauptachsentransformation auf die quadratischen Formen $2x^2 + 2y^2 - 2xy$ und $-3x^2 + 5y^2 + 2xy$ an! Welche Kegelschnitte gehören zu diesen quadratischen Formen?

Lösung:

Wir müssen die Matrix A ablesen und diagonalisieren.

Für $2x^2 + 2y^2 - 2xy$ ist die Matrix A gegeben als

$$A = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}.$$

Diese Matrix hat die Eigenwerte $\lambda_1 = 3$, $\lambda_2 = 1$ und die Eigenvektoren $(1, 1)^T$, $(-1, 1)^T$. Also ist A diagonalisierbar. Es liegt eine Ellipse vor.

Für die zweite Form ist die Matrix gegeben durch

$$B = \begin{pmatrix} -3 & 1 \\ 1 & 5 \end{pmatrix}.$$

Diese Matrix hat die Eigenwerte $\lambda_1 = 1 - \sqrt{17}$ und $\lambda_2 = 1 + \sqrt{17}$ mit den linear unabhängigen Eigenvektoren $(-4 - \sqrt{17}, 1)^T$ und $(-4 + \sqrt{17}, 1)^T$. Die Diagonalmatrix hat also zwei von Null verschiedene Werte unterschiedlichen Vorzeichens, also liegt eine Hyperbel vor!

17. Eine Matrix heißt *nilpotent*, wenn es ein $n \in \mathbb{N}$ gibt mit $A^n = 0$. Was können Sie über die Eigenwerte einer nilpotenten Matrix aussagen?

Lösung:

Alle Eigenwerte einer nilpotenten Matrix sind Null.

Nehmen wir das Gegenteil an, dann existiert ein $\lambda \neq 0$, $\mathbf{x} \neq \mathbf{0}$ mit $A\mathbf{x} = \lambda\mathbf{x}$ und $A^n\mathbf{x} = \lambda^n\mathbf{x} = \mathbf{0}$, ein Widerspruch.

18. Weisen Sie nach, dass für jede $m \times n$ -Matrix A die Matrizen $A^T A$ und AA^T die gleichen Eigenwerte besitzen!

Lösung:

Wenn A die Singulärwertzerlegung UDV besitzt, dann gilt $AA^T = UD^2U^T$ und $A^T A = V^T D^2 V$. Beide Matrizen sind diagonalisierbar, die Quadrate der singulären Werten von A sind jeweils die Eigenwerte.

19. Bestimmen Sie eine Matrix P , die für eine $m \times n$ -Matrix A die Gleichung $P^2 = A^T A$ erfüllt!

Lösung:

Wenn A die Singulärwertzerlegung UDV besitzt, dann gilt $A^T A = V^T D^2 V$. Die Matrix $P = V^T D V$ erfüllt die gegebene Gleichung.

20. Bestimmen Sie eine Singulärwertzerlegung der Matrix $A = \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix}$ und lösen Sie damit das lineare Ausgleichsproblem mit der rechten Seite $\mathbf{b} = (2, 0, 0, -1)^T$!

Lösung:

Der Rang der Matrix A ist offensichtlich 1. Die Matrix $A^T A$ ist gegeben durch

$$A^T A = \begin{pmatrix} 4 & 4 \\ 4 & 4 \end{pmatrix}.$$

Diese Matrix hat die Eigenwerte $\lambda_1 = 8$ und $\lambda_2 = 0$; zu λ_1 gehört der Eigenvektor $(1, 1)^T$, zu λ_2 der Eigenvektor $(-1, 1)^T$.

Dann ist der Singulärwert $\mu_1 = \sqrt{\lambda_1} = 2\sqrt{2}$. Die Matrix V ist gegeben durch Orthonormalisieren der beiden Eigenvektoren, die die Spalten definieren. Insbesondere ist dann

$$\mathbf{v}_1 = \begin{pmatrix} \frac{1}{2}\sqrt{2} \\ \frac{1}{2}\sqrt{2} \end{pmatrix}.$$

\mathbf{u}_1 ist gegeben durch $\mathbf{u}_1 = \frac{1}{\mu_1} A\mathbf{v}_1$, dann gilt

$$\mathbf{u}_1 = \frac{1}{2\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} \frac{1}{2}\sqrt{2} \\ \frac{1}{2}\sqrt{2} \end{pmatrix} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}.$$

Die Lösung des Ausgleichsproblems ist dann gegeben durch

$$\mathbf{x} = \begin{pmatrix} D^{-1} & 0 \\ 0 & 0 \end{pmatrix} V^T U^T \mathbf{b} = \sum_{i=1}^r \frac{1}{\sqrt{\mu_i}} \langle \mathbf{u}_i, \mathbf{b} \rangle \mathbf{v}_i,$$

in unserem Fall ist $r = 1$ und die Lösung ist gegeben durch

$$\mathbf{x} = \frac{1}{2\sqrt{2}} \left\langle \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}, \begin{pmatrix} 2 \\ 0 \\ 0 \\ -1 \end{pmatrix} \right\rangle \begin{pmatrix} \frac{1}{2}\sqrt{2} \\ \frac{1}{2}\sqrt{2} \end{pmatrix} = \begin{pmatrix} \frac{1}{8} \\ \frac{1}{8} \end{pmatrix}.$$

Kapitel 12

Folgen und Reihen

Verständnisfragen

Sachfragen

1. Nennen Sie Beispiele für reelle Zahlenfolgen!
2. Was ist eine arithmetische Folge?
3. Was ist eine geometrische Folge?
4. Erläutern Sie den Begriff der monotonen Folge!
5. Erläutern Sie den Begriff der beschränkten Folge!
6. Was ist das Infimum und Supremum einer beschränkten Folge?
7. Erläutern Sie den Begriff der ε -Umgebung!
8. Wie ist die Konvergenz einer Folge definiert?
9. Ist der Grenzwert einer Folge eindeutig?
10. Was ist eine Cauchy-Folge?
11. Wie hängen Monotonie, Beschränktheit und Konvergenz von Folgen zusammen?
12. Wie lauten die Limes-Sätze für Folgen?
13. Was ist eine Reihe?
14. Wann ist eine Reihe konvergent?
15. Was ist eine geometrische Reihe? Wann konvergiert sie?
16. Wie lautet das Cauchy'sche Hauptkriterium für Reihen?
17. Kennen Sie ein notwendiges Kriterium für die Konvergenz einer Reihe?
18. Was ist die harmonische Reihe? Ist sie konvergent?
19. Was ist eine alternierende Reihe?
20. Wie lautet das Leibniz-Kriterium?

21. Wie lautet das Majoranten-Kriterium?
22. Wie lautet das Quotienten-Kriterium?
23. Wie lautet das Wurzel-Kriterium?
24. Erläutern Sie den Begriff der absoluten Konvergenz!
25. Gilt für konvergente Reihen das Assoziativgesetz? Mit anderen Worten, wann dürfen Sie in einer Reihe Summanden vertauschen, ohne das Konvergenzverhalten zu verändern?
26. Was ist eine Potenzreihe?
27. Wie ist das Konvergenzintervall und der Konvergenzradius einer Potenzreihe definiert?
28. Wie kann der Konvergenzradius einer Potenzreihe bestimmt werden?
29. Wie sind die Landau'schen Symbole definiert?
30. Nennen Sie wichtige Vergleichsfolgen und ihre verbale Beschreibungen!
31. Was ist eine Norm?
32. Was ist eine p -Norm? Wie ist die Maximumsnorm definiert?
33. Wie kann das Gesamtschrittverfahren motiviert werden?
34. Wie kann das Einzelschrittverfahren motiviert werden?
35. Kennen Sie ein Konvergenzkriterium für Gesamt- und Einzelschrittverfahren?

Methodenfragen

1. Erkennen können, ob eine gegebene Folge eine arithmetische oder geometrische Folge ist.
2. Rekursiv gegebene Folgen bilden können.
3. Eine Folge auf Monotonie überprüfen können.
4. Eine Folge auf Schranken, Infimum und Supremum überprüfen können.
5. Eine Folge auf dem Zahlenstrahl und in einem kartesischen Koordinatensystem darstellen können.
6. Eine gegebene Folge auf Konvergenz untersuchen können.
7. Für ein gegebenes ε bei einer konvergenten Folge das $N(\varepsilon)$ bestimmen können.
8. Die Limes-Sätze für Folgen anwenden können.
9. Die Teilsummen von Reihen bilden können.
10. Kriterien für die Konvergenz und Divergenz von Reihen anwenden können.
11. Den Konvergenzradius einer Potenzreihe bestimmen können.
12. Die Landau'schen Symbole anwenden können.
13. Vektornormen berechnen können.
14. Die Konvergenz einer Folge von Vektoren überprüfen können.
15. Das Gesamtschrittverfahren durchführen und programmieren können.
16. Das Einzelschrittverfahren durchführen und programmieren können.

Übungsaufgaben

1. Untersuchen Sie die Folgen $a_n = \frac{n}{n+1}$, $b_n = 3 - 2n$ und $c_n = n + \ln(n)$ auf Monotonie und Beschränktheit!

Lösung:

(a_n) ist streng monoton steigend und beschränkt mit Schranken 0 und 1, (b_n) ist streng monoton fallend und nach oben beschränkt mit Schranke 1; nach unten unbeschränkt. (c_n) ist streng monoton steigend und nach unten beschränkt mit Schranke 1.

2. Untersuchen sie die Folge $a_n = \sqrt{25n^2 + 7n + 1} - 5n$ und weisen Sie nach, dass sie streng monoton fallend und durch $C = 0,7$ nach unten beschränkt ist.

Lösung:

Dass $0,7$ eine untere Schranke der Folge (a_n) ist folgt aus

$$\begin{aligned} a_n > 0.7 &\Leftrightarrow \sqrt{25n^2 + 7n + 1} > 5n + \frac{7}{10} \\ &\Leftrightarrow 25n^2 + 7n + 1 > 25n^2 + 7n + \left(\frac{7}{10}\right)^2 \\ &\Leftrightarrow 1 > \frac{49}{100} \end{aligned}$$

$$\begin{aligned} a_{n+1} - a_n &= \sqrt{25(n+1)^2 + 7(n+1) + 1} - (\sqrt{25n^2 + 7n + 1} + 5) \\ &= \sqrt{25n^2 + 7n + 26 + (50n + 7)} - \sqrt{25n^2 + 7n + 26 + 10\sqrt{25n^2 + 7n + 1}}. \end{aligned}$$

Es ist $10\sqrt{25n^2 + 7n + 1} > 50n + 7$, damit folgt $a_{n+1} - a_n < 0$.

3. Suchen Sie den kleinsten Index n mit $a_n < 0,01$ für die Folge $a_n = \frac{2n}{n^2+2}$.

Lösung:

Die Ungleichung

$$\frac{2n}{n^2+2} < \frac{1}{100}$$

ist erfüllt für $n > 200$.

4. Bestimmen Sie Infimum und Supremum für die Folgen $a_n = \frac{2n}{n+1}$, $b_n = \frac{1}{n^2+1}$ und $c_n = (-1)^n \frac{n+2}{3^n}$!

Lösung:

Die Folge (a_n) hat das Infimum 1 und Supremum 2, (b_n) 0 als Infimum und $\frac{1}{2}$ als Supremum und (c_n) besitzt das Infimum -1 und Supremum $\frac{4}{9}$.

5. Zeigen Sie, dass die Folgen $a_n = \frac{1}{2^n}$ und $b_n = 1 + (-1)^n \frac{1}{n^2}$ konvergent sind mit Grenzwerten $a = 0$ und $b = 1$!

Lösung:

Für beliebiges $\varepsilon > 0$ und $n > \log_2 \frac{1}{\varepsilon}$ ist $|\frac{1}{2^n}| < \varepsilon$.

Für $n > \frac{1}{\varepsilon} \sqrt{\varepsilon}$ ist $|b_n - b| < \varepsilon$.

6. Angenommen, a und b sind positive reelle Zahlen mit $a > b$. Mit a_1 und b_1 soll ihr arithmetisches bzw. geometrisches Mittel bezeichnet werden: $a_1 = \frac{a+b}{2}$, $b_1 = \sqrt{a \cdot b}$. Allgemein werden die Folgen

$$a_{n+1} = \frac{a_n + b_n}{2}, b_{n+1} = \sqrt{a_n \cdot b_n}$$

gebildet. Beweisen Sie durch vollständige Induktion $a_n > a_{n+1} > b_{n+1} > b_n$. Zeigen Sie, dass die beiden Folgen konvergieren und dass die beiden Grenzwerte übereinstimmen!

Lösung:

Das arithmetische Mittel ist immer größer als das geometrische Mittel, es ist

$$\frac{a+b}{2} > \sqrt{ab}, a > b > 0.$$

Dies beweist man mit Hilfe der binomischen Formel:

$$\frac{a+b}{2} > \sqrt{ab} \Leftrightarrow \frac{a^2 + 2ab + b^2}{4} > ab \Leftrightarrow \frac{a^2 - 2ab - b^2}{4} > 0 \Leftrightarrow \frac{(a-b)^2}{4} > 0.$$

Dann ist aber auch $a_n > b_n$ für die beiden Folgen. Es ist $b_{n+1} > b_n$ wegen

$$\sqrt{a_n b_n} > b_n \Leftrightarrow a_n > b_n$$

und auch $a_{n+1} < a_n$ wegen

$$\frac{a_n + b_n}{2} < a_n \Leftrightarrow \frac{b_n}{2} < \frac{a_n}{2} \Leftrightarrow b_n < a_n.$$

Die Folge (a_n) ist also monoton fallend und durch b_1 beschränkt; damit ist sie auch konvergent. Genauso folgt die Konvergenz der monoton wachsenden Folge (b_n) , die durch a_1 beschränkt ist.

Für die Folge $d_n = (a_n - b_n)^2$ gilt

$$\begin{aligned} (a_n - b_n)^2 &= (a_n + b_n)^2 - 4a_n b_n \\ &= 4a_{n+1}^2 - 4b_{n+1}^2 \\ &= 4(a_{n+1} - b_{n+1})(a_{n+1} + b_{n+1}) \end{aligned}$$

Dann gilt

$$d_{n+1} < \frac{1}{8b_1} d_n.$$

Die Folge (d_n) ist eine Nullfolge; die Konvergenz ist quadratisch!

Nach Gauß wird der gemeinsame Grenzwert der beiden Folgen *arithmetisch-geometrisches Mittel von a und b* $AGM(a, b)$ genannt.

7. Zeigen Sie, dass die nachstehenden Folgen den angegebenen Grenzwert g besitzen. Berechnen Sie die Anzahl der Folgenglieder, die außerhalb der ε -Umgebung um den Grenzwert liegen: $a_n = \frac{3n+2}{2n-1}$, $g = \frac{3}{2}$, $\varepsilon = 0, 1$; $b_n = \frac{3-4n+n^2}{2+3n-n^2}$, $g = -1$, $\varepsilon = 0, 005$.

Lösung:

Klammern Sie die höchste Potenz aus und verwenden Sie die Limesätze!

Bei der Folge (a_n) liegen die ersten 17 Folgenglieder außerhalb der ε -Umgebung mit $\varepsilon = 0, 1$, bei (b_n) sind dies die ersten 197 Folgenglieder für $\varepsilon = 0, 005$.

8. Erstellen Sie eine grafische Darstellung der Folgen $a_n = (-1)^n \frac{n+1}{n}$, $a_n = \frac{n^3}{n!}$ und $a_n = \sqrt[n]{3^n + 5^n}$. Schätzen Sie gegebenenfalls den Grenzwert und weisen Sie die Konvergenz nach!

Lösung:

Die Folge $a_n = (-1)^n \frac{n+1}{n}$ ist in Abbildung 12.1 dargestellt. Es ist klar zu erkennen, dass es keinen eindeutigen Grenzwert gibt. Die Folge (a_{2n}) konvergiert gegen 1, die Folge mit ungeraden Indizes gegen -1 .

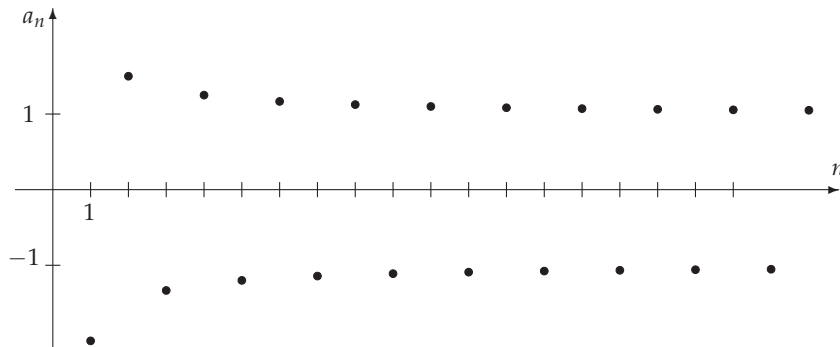


Abbildung 12.1: $a_n = (-1)^n \frac{n+1}{n}$ für $n \leq 20$

Einige Folgenglieder der Folge $a_n = \frac{n^3}{n!}$ finden Sie in Abbildung 12.2. Die ersten Glieder scheinen zu wachsen, dann werden die Glieder sehr schnell sehr klein. Die Konvergenz der Folge gegen Null sieht man durch

$$\frac{n^3}{n!} = \frac{1}{(n-3)!} \cdot \frac{n}{n^2 - 3n + 1},$$

das Produkt zweier Nullfolgen ist eine Nullfolge.

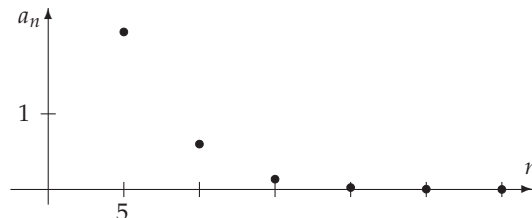
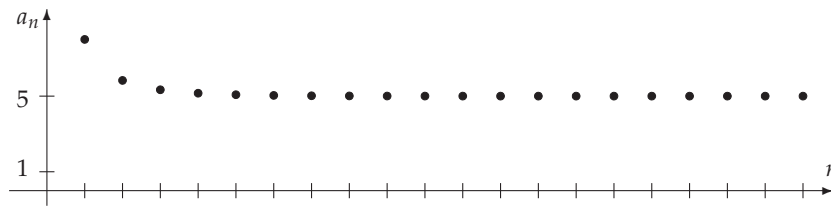


Abbildung 12.2: $a_n = \frac{n^3}{n!}$ für $n = 5, \dots, 10$

Die Folge $a_n = \sqrt[n]{3^n + 5^n}$ finden Sie in Abbildung 12.3. Sie konvergiert, wenn man der Abbildung Glauben schenken darf nach gegen 5.

Nach unten ist die Folge durch die konstante Folge 5, nach oben wegen $\sqrt[n]{3^n + 5^n} < \sqrt[n]{25}$ durch die Folge $b_n = \sqrt[n]{25}$ beschränkt; für $q > 1$ ist $\sqrt[n]{q}$ eine konvergente Folge mit Grenzwert 1. Mit dem Sandwichsatz folgt dann auch die vermutete Konvergenz von (a_n) gegen den Grenzwert 5.

9. Die Folge $a_n = (1 + \frac{1}{n})^n$ konvergiert gegen die Euler'sche Zahl e .
- Zeigen Sie, dass für $0 \leq a < b$ die Ungleichungen $\frac{b^{n+1} - a^{n+1}}{b-a} < (n+1)b^n$ und $b^n[(n+1)a - nb] < a^{n+1}$ erfüllt sind!
 - Verwenden Sie $a = 1 + \frac{1}{n+1}$ und $b = 1 + \frac{1}{n}$, um die Monotonie der Folge (a_n) zu untersuchen!

Abbildung 12.3: $a_n = \sqrt[n]{3^n + 5^n}$

(c) Verwenden Sie $a = 1$ und $b = 1 + \frac{1}{2n}$, um nachzuweisen, dass $a_{2n} < 4$ und $a_n < 4$ erfüllt sind!

(d) Weisen Sie nach, dass (a_n) konvergiert!

Lösung:

Teilaufgabe (a):

Die erste Ungleichung kann mit vollständiger Induktion bewiesen werden. Die Induktionsbasis $n = 1$:

$$\frac{b^2 - a^2}{b - a} = \frac{(b - a)(b + a)}{b - a} = b + a < 2b,$$

für $0 \leq a < b$.

Der Induktionsschritt folgt mit:

$$\begin{aligned} (b^{n+1} - a^{n+1}) &= (b^n - a^n)(b + a) - ab^n + a^n b \\ &< 2b(b^n - a^n) + b^2(a^{n-1} - b^{n-1}). \end{aligned}$$

Dann ist

$$\begin{aligned} \frac{b^{n+1} - a^{n+1}}{b - a} &= 2b \frac{b^n - a^n}{b - a} + b^2 \frac{a^{n-1} - b^{n-1}}{b - a} \\ &< 2bnb^{n-1} - b^2(n-1)b^{n-2} \\ &= (n+1)b^n. \end{aligned}$$

Dann gilt

$$\begin{aligned} -a^{n+1} &< b^n(n+1)(b-a) - b^{n+1} \\ &= b^n((n+1)b - (n+a)a + b) \\ &= b^n(nb - (n+1)a). \end{aligned}$$

Damit ist auch die zweite Ungleichung bewiesen.

Teilaufgabe (b): Für $a = 1 + \frac{1}{n+1}$ und $b = 1 + \frac{1}{n}$ ist $0 \leq a < b$ erfüllt; einsetzen in die zweite Ungleichung in Teilaufgabe a) erhält man durch Einsetzen

$$\left(1 + \frac{1}{n}\right)^n \left[(n+1) \left(1 + \frac{1}{n+1}\right) - n \left(1 + \frac{1}{n}\right) \right] = \left(1 + \frac{1}{n}\right)^n$$

und damit $a_n < a_{n+1}$.

Teilaufgabe (c): Setzt man diesmal die angegebenen Werte für a und b in die 2. Ungleichung ein, dann erhält man

$$\left(1 + \frac{1}{2n}\right)^n < 2 \Leftrightarrow a_{2n} < 4.$$

Dann gilt wegen der Monotonie auch $a_{2n} < a_{2n+1} < a_{2n+2} < 4$, also ist 4 eine obere Schranke für die Folge (a_n) .

Teilaufgabe (d): Eine monotone und beschränkte Folge ist konvergent!

10. Bestimmen Sie den Grenzwert der Folge $(\frac{F_{n+1}}{F_n})!$

Lösung:

Die Fibonacci-Folge kann mit Hilfe des goldenen Schnitts

$$\theta = \frac{1 + \sqrt{5}}{2}$$

durch

$$F_n = \frac{\sqrt{5}}{5}(\theta^n - (1 - \theta)^n)$$

dargestellt werden.

Dann gilt für den Quotienten

$$\frac{F_{n+1}}{F_n} = \frac{\theta^{n+1} - (1 - \theta)^{n+1}}{\theta^n - (1 - \theta)^n},$$

mit den Limesätzen folgt

$$\lim_{n \rightarrow \infty} \frac{F_{n+1}}{F_n} = \theta,$$

denn $(1 - \theta)^n$ ist eine Nullfolge!

11. Untersuchen Sie die Reihen $\sum_{i=1}^{\infty} \frac{n}{5^n}$, $\sum_{i=1}^{\infty} \frac{3^{2n}}{(2n)!}$, $\sum_{i=1}^{\infty} n \cdot (\frac{1}{2})^{n-1}$ und $\sum_{i=1}^{\infty} \frac{(-1)^{n+1}}{n \cdot 5^{2n-1}}$ auf Konvergenz!

Lösung:

Die ersten drei Reihen konvergieren auf Grund des Quotientenkriteriums, die Konvergenz der alternierenden Reihe folgt aus dem Leibniz-Kriterium.

12. Beweisen Sie das Wurzelkriterium!

Lösung:

Der Beweis verläuft analog zum Nachweis des Quotientenkriteriums.

Erfüllt die Folge (a_n) das Wurzelkriterium, dann gibt es ein $0 < q < 1$ mit

$$\sqrt[n]{a_n} < q.$$

Also ist

$$a_n < q^n,$$

und damit erhält man eine konvergente geometrische Reihe als Majorante.

13. Programmieren Sie in der Programmiersprache Ihrer Wahl eine Funktion, die in einer Schleife die Teilsummen der harmonischen Reihe berechnet und in jedem Schleifendurchgang ausgibt. Wenn Sie die Schleife lang genug laufen lassen, scheint auf dem Computer die harmonische Reihe zu konvergieren. Beschreiben Sie die Gründe für diesen Irrtum!

Lösung:

Wie Sie wissen divergiert die harmonische Reihe, die Folge der Teilsummen wächst über jede Grenze. Egal welches Gleitpunkt-Zahlensystem Sie benutzen, jedes dieser Systeme hat eine größte darstellbare Zahl. Wird diese erreicht, erhalten Sie danach immer nur diese größte Zahl als Summenwert.

Bei der Summierung werden auch die Genauigkeitsunterschiede zwischen `float` und `double` deutlich. Für die die Teilsumme $S_{1\,000}$ ergibt sich die folgende Ausgabe:

Die Summe für float-Genauigkeit bis 1/100000 :12.090851

Die Summe für float-Genauigkeit bis 1/100000 :12.090146129863335

Die Summe für float-Genauigkeit bis 1/5000000 :15.403683

Die Summe für float-Genauigkeit bis 1/5000000 :16.002164235298594

Bei float-Genauigkeit erhält man jetzt schon keinen neuen Wert.

Die Summe für float-Genauigkeit bis 1/10000000 :15.403683

Die Summe für float-Genauigkeit bis 1/10000000 :16.695311365857272

Bei double geht es noch weiter:

Die Summe für float-Genauigkeit bis 1/100000000 :15.403683

Die Summe für float-Genauigkeit bis 1/100000000 :18.997896413852555

Die Summe für float-Genauigkeit bis 1/2000000000 :15.403683

Die Summe für float-Genauigkeit bis 1/2000000000 :21.993628682662845

Wann ist eigentlich bei double Schluß?

14. Weisen Sie nach, dass $\sum_{i=1}^{\infty} \frac{1}{i^2+x^2}$ für $x \in (0; \infty)$ und $\sum_{i=1}^{\infty} \frac{\sin(ix)}{i^{\frac{3}{2}}}$ für alle $x \in \mathbb{R}$ konvergieren!

Lösung:

Es gibt die konvergenten Majoranten $\sum \frac{1}{i^2}$ und $\sum \frac{1}{i^{\frac{3}{2}}}$.

15. Bestimmen Sie den Konvergenzradius der Potenzreihen $\sum_{i=1}^{\infty} ix^i$, $\sum_{i=1}^{\infty} \frac{x^i}{i}$, $\sum_{i=1}^{\infty} \sin(i\pi)x^i$, $\sum_{i=1}^{\infty} \ln(i)x^i$, $\sum_{i=1}^{\infty} \frac{x^i}{i^{\ln(i)}}$, $\sum_{i=1}^{\infty} \frac{\ln(i)}{i}x^i$.

Lösung:

$\sum_{i=1}^{\infty} ix^i$: $R = 1$, am Rand divergent;

$\sum_{i=1}^{\infty} \sin(i\pi)x^i$: $R = \infty$;

$\sum_{i=1}^{\infty} \frac{x^i}{i}$: $R = \infty$;

$\sum_{i=1}^{\infty} \ln(i)x^i$: $R = 1$, am Rand divergent;

$\sum_{i=1}^{\infty} \frac{x^i}{i^{\ln(i)}}$: $R = 1$, am Rand konvergent;

$\sum_{i=1}^{\infty} \frac{\ln(i)}{i}x^i$: $R = 1$, für $x = 1$ divergent, für $x = -1$ konvergent.

16. Die Besselfunktion der Ordnung 1 ist durch die Potenzreihe $J_1(x) = \sum_{i=0}^{\infty} \frac{(-1)^i x^{2i+1}}{i!(i+1)!2^{2i+1}}$ gegeben. Bestimmen Sie das Konvergenzintervall und vergleichen Sie die ersten 5 Teilsummen von J_1 !

Lösung:

Auch die Besselfunktion 1. Ordnung konvergiert für alle $x \in \mathbb{R}$, das folgt aus dem Quotientenkriterium oder dem Wurzelkriterium wie im Fall der Besselfunktion 0. Ordnung auf Seite 310. Die Funktion J_1 ist im Gegensatz zur Besselfunktion 0.-Ordnung nicht symmetrisch zur y -Achse, sondern ähnlich wie die Sinus-Funktion punktsymmetrisch zum Ursprung - eine ungerade Funktion. In Abbildung ?? finden Sie eine grafische Darstellung von J_1 ; Abbildung ?? zeigt die Teilsummen S_3 , S_5 , S_7 und S_9 .

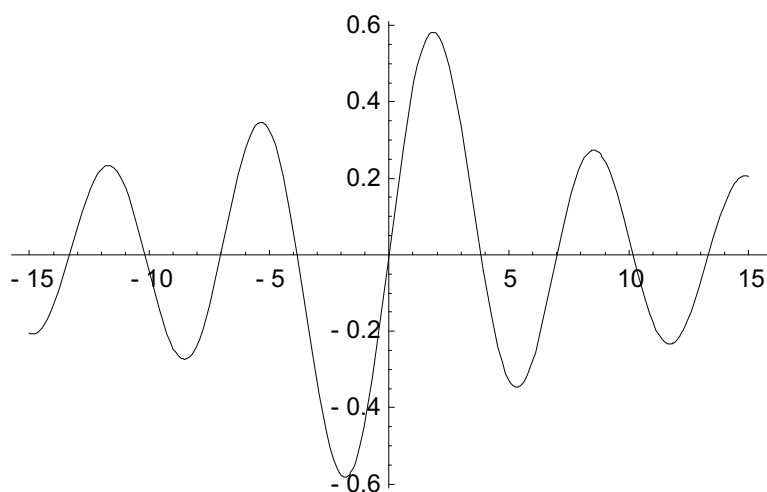


Abbildung 12.4: Der Verlauf der Bessel-Funktion $J_1(x)$

Die ersten 5 Teilsummen für $J_1(x)$ sind gegeben durch

$$S_1(x) = \frac{x}{2},$$

$$S_3(x) = \frac{x}{2} - \frac{x^3}{16},$$

$$S_5(x) = \frac{x}{2} - \frac{x^3}{16} + \frac{x^5}{384},$$

$$S_7(x) = \frac{x}{2} - \frac{x^3}{16} + \frac{x^5}{384} - \frac{x^7}{18\,432},$$

$$S_9(x) = \frac{x}{2} - \frac{x^3}{16} + \frac{x^5}{384} - \frac{x^7}{18\,432} + \frac{x^9}{1\,474\,560}.$$

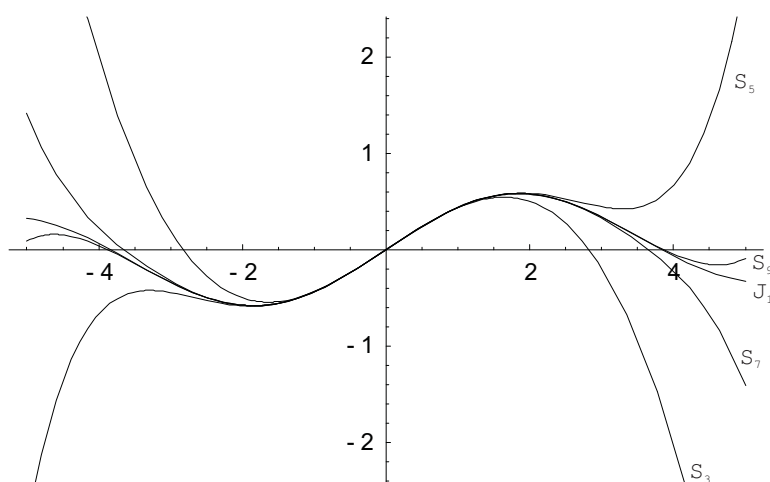


Abbildung 12.5: Die Teilsummen S_3, S_5, S_7, S_9 der Besselfunktion $J_1(x)$

17. Benutzen Sie Ihre Implementierung des Euklidischen Algorithmus und zählen Sie für konkrete Eingaben von a und b mit $b < a$ die Anzahl der benötigten Schritte. Vergleichen Sie diese mit der theoretischen Obergrenze!

Lösung:

Die Werte in der folgenden Tabelle wurden mit `g++` und `unsigned long` als Ganzzahltyp berechnet.

a	10^2	10^3	10^4	10^5	10^6	10^7	10^8	10^9
$\lceil \frac{\log(\sqrt{5a+\frac{1}{2}})}{\log \theta} \rceil$	10	15	19	24	29	34	38	43
Max. Anzahl Schritte für $\text{ggT}(a, b), b < a$	7	11	15	19	24	28	32	37

18. Beweisen Sie $O(A) + O(A) = O(A)$, $o(A) + o(A) = o(A)$ und $O(A)O(B) = O(AB)$, $o(A)o(B) = o(AB)$ für zwei reelle Zahlenfolgen A und B .

Lösung:

Die Beweise werden für das o durchgeführt; für O verlaufen Sie analog.

Ist $A = (a_n)$ eine Folge und $(b_n) = o(A)$, $(c_n) = o(A)$, dann ist

$$\lim_{n \rightarrow \infty} \frac{b_n}{a_n} = 0, \lim_{n \rightarrow \infty} \frac{c_n}{a_n} = 0.$$

Das gleiche gilt sicher auch für die Folgen

$$\frac{b_n + c_n}{a_n}$$

und

$$\frac{b_n \cdot c_n}{a_n}$$

19. Beschreiben Sie Folgen mit $(a_n) = O(1)$ und $(a_n) = o(1)$!

Lösung:

Für eine Folge mit $(a_n) = O(1)$ ist der Quotient

$$\frac{a_n}{1} = a_n$$

beschränkt; also auch die Folge (a_n) selbst.

Für $(a_n) = o(1)$ ist mit dem gleichen Argument die Folge eine Nullfolge.

20. Weisen Sie die folgende Ungleichung nach: $\frac{1}{\sqrt{n}} \|\mathbf{x}\|_2 \leq \|\mathbf{x}\|_\infty \leq \|\mathbf{x}\|_2 \leq \sqrt{n} \|\mathbf{x}\|_\infty$.

Lösung:

Die Ungleichung $\|\mathbf{x}\|_\infty \leq \|\mathbf{x}\|_2$ ist klar, Sie müssen einfach die Definition der beiden Normen aufschreiben.

Es ist

$$\|\mathbf{x}\|_2^2 = \sum_{i=1}^n x_i^2 \leq \sum_{i=1}^n \|\mathbf{x}\|_\infty^2 = n \|\mathbf{x}\|_\infty^2$$

und damit sind auch die anderen Ungleichungen bewiesen.

21. Implementieren Sie das Gesamt- und Einzelschrittverfahren und lösen Sie damit selbst gewählte Beispiele!

Lösung:

Die Implementierung der beiden Verfahren, wieder mit Hilfe der `NINJA`-Klassen und `BLAS` in Java:

```
private static void gesamtSchritt(doubleArray2D A, doubleArray1D b,
                                doubleArray1D x, double epsilon)
{
    int i, j, counter, n = x.size();
    double summe, fehler;
    doubleArray1D xneu = new doubleArray1D(n);

    fehler = 1.0E10;
    counter = 0;

    while (fehler > epsilon) {
        for (i=0; i<n; i++) {
            summe = 0.0;
            for (j=0; j<i; j++)
                summe += A.get(i, j)*x.get(j);
            for (j=i+1; j<n; j++)
                summe += A.get(i, j)*x.get(j);
            xneu.set(i, (b.get(i)-summe)/A.get(i, i));
        }

        fehler = Blas.dnorm2(x.minus(xneu));
        x.assign(xneu);
        counter++;
    }
    System.out.println("Das Konvergenzkriterium beim Gesamtschrittverfahren"
        + " war nach "+counter+" Iterationen" +
        " erfüllt!");
}

private static void einzelSchritt(doubleArray2D A, doubleArray1D b,
                                  doubleArray1D x, double epsilon)
{
    int i, j, counter, n = x.size();
    double summe, fehler;
    doubleArray1D xalt = new doubleArray1D(n);

    fehler = 1.0E10;
    counter = 0;

    while (fehler > epsilon) {
        xalt.assign(x);
        for (i=0; i<n; i++) {
            summe = 0.0;
            for (j=0; j<i; j++)
                summe += A.get(i, j)*x.get(j);
            for (j=i+1; j<n; j++)
                summe += A.get(i, j)*x.get(j);
            x.set(i, (b.get(i)-summe)/A.get(i, i));
        }

        fehler = Blas.dnorm2(x.minus(xalt));
        counter++;
    }
    System.out.println("Das Konvergenzkriterium beim Einzelschrittverfahren" +
        " war nach "+counter+" Iterationen" +
        " erfüllt!");
}
```

```

private static void initArrays(doubleArray2D A, doubleArray1D b,
                             doubleArray1D x) {

    // Besetzen des Gleichungssystems und des Startvektors für die
    // iterative Lösung.
    b.set(0, 7.0); b.set(1, 7.0); b.set(2, 3.0);

    A.set(0,0, 4.0); A.set(0, 1, 1.0); A.set(0, 2, 2.0);
    A.set(1,0, 2.0); A.set(1, 1, 4.0); A.set(1, 2, 1.0);
    A.set(2,0, 0.0); A.set(2, 1, 1.0); A.set(2, 2, 2.0);

    x.set(0, 1.0); x.set(1, 2.0); x.set(2,3.0);
}

```

Wenn man beispielsweise das Gleichungssystem von Seite 318 verwendet, dann benötigt das Gesamtschrittverfahren für eine Abbruchgenauigkeit von $\epsilon = 10^{-7}$ 91 Iterationen, das Einzelschrittverfahren nur 19.

Sehr häufig werden iterative Verfahren zur Lösung von linearen Gleichungssystemen verwendet, um diskretisierte partielle Differenzialgleichung, beispielweise die Wärmeleitungsgleichung zu lösen. Dabei entstehen durch das Ersetzen der Ableitungen durch Differenzialquotienten sehr dünn besetzte Bandmatrizen, beispielweise eine Matrix wie

$$A = \begin{pmatrix} 4 & -1 & 0 & \dots \\ -1 & 4 & -1 & \dots \\ 0 & \dots & \dots & \dots \\ \dots & -1 & 4 & -1 \\ 0 & \dots & -1 & 4 \end{pmatrix}.$$

Nur die Diagonale und die beiden Nebendiagonalen sind ungleich Null. Je nach Diskretisierungsmethode werden die Matrizen etwas komplexer; aber die dünne Struktur (im angelsächsischen Sprachraum wird von *sparse matrices* gesprochen) bleibt auf jeden Fall bestehen.

Dann kann natürlich darauf verzichtet werden, das Produkt zwischen Matrix A und dem Iterationsvektor bei den beiden Gesamt- und Einzelschrittverfahren zu berechnen; man würde nur unnötig oft mit 0 multiplizieren. Für ein Element des Iterationsvektors beim Gesamtschrittverfahren ergibt sich dann folgende Berechnung (dabei wurden die Vektorelemente mit $0, \dots, n-1$ nummeriert, wie in den Programmiersprachen):

$$\begin{aligned} x_0^{(k+1)} &= 0,25(b_0 + x_1^{(k)}), \\ x_i^{(k+1)} &= 0,25(b_i + x_{i-1}^{(k)} + x_{i+1}^{(k)}), \\ x_{n-1}^{(k+1)} &= 0,25(b_{n-1} + x_{n-2}^{(k)}). \end{aligned}$$

Die Berechnungsvorschrift für das Einzelschrittverfahren können Sie sicher leicht davon ableiten. Im folgenden Quelltext sind noch zusätzlich alle Komponenten der rechten Seite \mathbf{b} gleich 1,0; dann kann auf das Feld dafür gänzlich verzichtet werden.

```

private static void gesamtSchrittSparse(doubleArray1D x, double epsilon) {
    // Gesamtschrittverfahren für eine dünn besetzte Matrix
    int i, j, counter, n=x.size();
    double fehler;
    doubleArray1D xneu = new doubleArray1D(n);

    fehler = 1.0E10;
    counter = 0;
}

```

```

while (fehler > epsilon) {
    xneu.set(0, 0.25*(1.0+x.get(1)));
    for (i=1; i<n-1; i++)
        xneu.set(i, 0.25*(1.0+x.get(i-1)+x.get(i+1)));
    xneu.set(n-1, 0.25*(1.0+x.get(n-2)));
    fehler = Blas.dnorm2(x.minus(xneu));
    x.assign(xneu);
    counter++;
}
System.out.println("Das Konvergenzkriterium beim Gesamtschrittverfahren"
    + " war nach "+counter+" Iterationen" +
    " erfüllt!");
}

private static void einzelSchrittSparse(doubleArray1D x, double epsilon) {
    // Gesamtschrittverfahren für eine dünn besetzte Matrix
    int i,j,counter, n=x.size();
    double fehler;
    doubleArray1D xalt = new doubleArray1D(n);

    fehler = 1.0E10;
    counter = 0;

    while (fehler > epsilon) {
        xalt.assign(x);
        x.set(0, 0.25*(1.0+x.get(1)));
        for (i=1; i<n-1; i++)
            x.set(i, 0.25*(1.0+x.get(i-1)+x.get(i+1)));
        x.set(n-1, 0.25*(1.0+x.get(n-2)));
        fehler = Blas.dnorm2(x.minus(xalt));
        counter++;
    }
    System.out.println("Das Konvergenzkriterium beim Einzelschrittverfahren"
        + " war nach "+counter+" Iterationen" +
        " erfüllt!");
}
}

```

Bei einer Abbruchgenauigkeit von $\varepsilon = 10^{-7}$ benötigt dieses Gesamtschrittverfahren 50 Iterationen. Für die gleiche Genauigkeitsanforderung benötigte das Einzelschrittverfahren genau 50%, nämlich 25 Iterationen.

Kapitel 13

Differenzialrechnung

Verständnisfragen

Sachfragen

1. Nennen Sie Beispiele für polynomiale, rationale, algebraische und transzendente Funktionen!
2. Erläutern Sie den Funktionenlimes!
3. Erläutern Sie den Begriff einer Asymptote einer Funktion!
4. Wie lautet das ε - δ -Kriterium für den Funktionenlimes einer Funktion!
5. Erläutern Sie den Begriff der Stetigkeit!
6. Wann ist eine Funktion monoton?
7. Wann ist eine Funktion beschränkt?
8. Wie lautet der Zwischenwertsatz?
9. Erläutern Sie das Bisektionsverfahren!
10. Erläutern Sie die Regula-Falsi!
11. Erläutern Sie die Nullstellenbestimmung durch quadratische Interpolation!
12. Wann ist eine Funktion differenzierbar?
13. Wie hängen Differenzierbarkeit und Stetigkeit einer Funktion zusammen?
14. Wie lautet die Produkt- und Quotientenregel?
15. Wie lautet die Kettenregel?
16. Erläutern Sie den Zusammenhang zwischen der Ableitung einer Funktion und der Ableitung ihrer Umkehrfunktion!
17. Erläutern Sie das Newton-Verfahren.
18. Erläutern Sie den Begriff der Konvergenzordnung!
19. Wie lautet der Satz von Rolle?

20. Wie lautet der Mittelwertsatz?
21. Erläutern Sie die Näherung einer Funktion durch ein Taylor-Polynom!
22. Wie lautet die Laplace-Form des Restglieds bei der Taylor-Entwicklung?
23. Nennen Sie einige Beispiele für Taylorreihen!
24. Erläutern Sie den Zusammenhang zwischen Potenzreihen und Taylorreihen!
25. Wann ist eine Funktion konvex bzw. konkav?
26. Was ist ein lokales Maximum bzw. Minimum einer Funktion?
27. Nennen Sie ein notwendiges und ein hinreichendes Kriterium für ein lokales Extremum einer differenzierbaren Funktion!
28. Was versteht man unter einem Wendepunkt?
29. Wie können die Wendepunkte einer differenzierbaren Funktion bestimmt werden?

Methodenfragen

1. Transformationen für Funktionen durchführen können.
2. Funktionenlimes berechnen können.
3. Asymptoten für gegebene Funktionen bestimmen können.
4. Das ε - δ -Kriterium für eine gegebene Funktion anwenden können.
5. Überprüfen können, ob eine gegebene Funktion stetig ist.
6. Eine gegebene Funktion auf Monotonie und Beschränktheit überprüfen können.
7. Das Bisektionsverfahren anwenden und implementieren können.
8. Die Regula Falsi anwenden und implementieren können.
9. Die Nullstellenbestimmung durch quadratische Interpolation anwenden und implementieren können.
10. Eine gegebene Funktion ableiten können.
11. Die Ableitungsregeln, insbesondere die Kettenregel, anwenden können.
12. Die Ableitung einer Umkehrfunktion berechnen können.
13. Das Newton-Verfahren anwenden und implementieren können.
14. Das Taylor-Polynom einer Funktion aufstellen können.
15. Monotonie von differenzierbaren Funktionen untersuchen können.
16. Das Krümmungsverhalten von differenzierbaren Funktionen untersuchen können.
17. Lokale Extrema von differenzierbaren Funktionen bestimmen können.
18. Wendepunkte von differenzierbaren Funktionen bestimmen können.

Übungsaufgaben

1. In Abbildung 13.1 ist ein Graph einer Funktion f gegeben. Verwenden Sie diesen, um die Graphen der folgenden Funktionen zu skizzieren: $f(2x)$, $f(-x)$, $-f(-x)$, $f(x+4)$, $f(x)+4$, $2f(x)+3$.

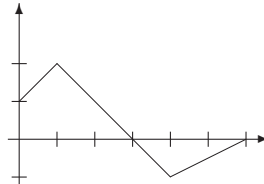


Abbildung 13.1: Ein Funktionsgraph für Aufgabe 1

Lösung:

Lösung.

2. Berechnen Sie die Funktionenlimes (falls sie existieren) für die Funktionen $f(x) = \frac{x^2+8}{x+4}$, $x_0 = -4$; $g(x) = \frac{x^2+7x+10}{x^2-x-6}$, $x_0 = -2$ und $h(x) = \frac{\sqrt{x+1}-\sqrt{1-x}}{\sqrt{x}}$, $x_0 = 0$.

Lösung:

f hat einen Pol in $x_0 = -4$, g hat an der Stelle $x_0 = -2$ einen Grenzwert, denn der Zähler hat die beiden Nullstellen $x = 5$ und $x = -2$; das Nennerpolynom die Nullstellen $x = 3$ und ebenfalls $x = -2$. Also kann g geschrieben werden als

$$g(x) = \frac{(x+5)(x+2)}{(x-3)(x+2)}.$$

Dann ist der Funktionenlimes für $x_0 = -2$ gegeben als

$$\lim_{x \rightarrow -2} \frac{x+5}{x-3} = -\frac{3}{5}.$$

h hat eine Unendlichkeitsstelle bei $x_0 = 0$.

3. Beweisen Sie, dass $f(x) = \sin\left(\frac{1}{x}\right)$ für $x_0 = 0$ keinen Funktionenlimes besitzt!

Lösung:

Das Argument $\frac{1}{x}$ geht gegen Unendlich, die Sinus-Funktion hat aber keine Asymptote für $x \rightarrow \infty$.

4. Bestimmen Sie die Asymptoten für $f(x) = \frac{2x^2}{x^2+1}$ und $g(x) = \frac{1+x^2+2x^3}{2+x^2}$.

Lösung:

Für $f(x)$ erhalten Sie durch Ausklammern der höchsten Potenz

$$f(x) = \frac{2}{1 + \frac{1}{x^2}} \rightarrow 2$$

sowohl für $x \rightarrow \infty$ als auch $x \rightarrow -\infty$.

Für $g(x)$ erhalten Sie durch Ausklammern der höchsten Potenz

$$g(x) = \frac{2x + 1 + \frac{1}{x^2}}{1 + \frac{2}{x^2}}.$$

Für $x \rightarrow \infty$ hat g ∞ als Grenzwert, im negativen ist der Grenzwert $-\infty$.

5. Bestimmen Sie die Unendlichkeitsstellen der Funktion $f(x) = \frac{1}{\cos(x)\sin(x)}$.

Lösung:

Die Unendlichkeitsstellen sind die Nullstellen des Nenners. Diese sind gegeben durch die Nullstellen der Sinus- und der Kosinusfunktion, also $x = k \cdot \frac{\pi}{2}, k \in \mathbb{Z}$.

6. Beweisen Sie mit Hilfe des ε - δ -Kriteriums $\lim_{x \rightarrow 3} x^2 = 9$.

Lösung:

Zuerst wird ein Wert für δ geschätzt. Gesucht ist eine Zahl δ mit $|x^2 - 9| < \varepsilon$ für $0 < |x - 3| < \delta$.

Mit der dritten binomischen Formel ist

$$|x^2 - 9| = |x - 3||x + 3|.$$

Falls $x \in (2; 4)$ liegt, dann gilt $5 < x + 3 < 7$ und $|x + 3| < 7$.

Jetzt gibt es zwei Bedingungen an x , nämlich

$$|x - 3| < 1, |x - 3| < \frac{\varepsilon}{7}.$$

Dann kann für den Grenzwert als δ das Minimum zwischen 1 und $\frac{\varepsilon}{7}$ verwendet werden.

7. Zwischen welchen Zahlen liegt eine positive Nullstelle von $f(x) = x^3 - 1\,000x - 500\,000$?

Lösung:

In $[0, 10^4]$, denn dort sind die Voraussetzungen des Zwischenwertsatzes erfüllt.

8. Implementieren Sie das Bisektionsverfahren und Regula Falsi und bestimmen Sie damit auf zwei und vier Dezimalstellen genau die im Intervall $[\frac{\pi}{2}; \pi]$ liegende Lösung von $5 \sin(x) = x$.

Lösung:

In diesem Kapitel folgen noch einige Aufgaben mit verschiedenen Algorithmen. Deshalb werden zuerst virtuelle Klassen implementiert, von denen dann die konkreten Algorithmen abgeleitet werden. In diesen konkreten Algorithmen ist auch die Beispielfunktion, hier $f(x) = 5 \sin(x) - x$ implementiert. Einen Graphen der Beispielfunktion finden Sie in Abbildung 13.2.

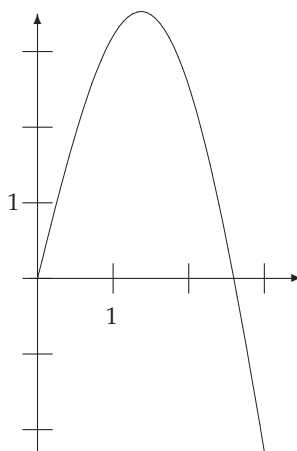


Abbildung 13.2: Die Funktion $f(x) = 5 \sin(x) - x$

```
//-----  
// nullstellenAlgorithmus.java  
// Virtuelle Oberklasse für die Algorithmen für  
// die Bestimmung von Nullstellen eindimensionaler Funktionen.  
// -----  
// Autor: Manfred Brill  
// Letzte Änderung: 14. 11. 2004  
//-----  
public abstract class nullstellenAlgorithmus {  
  
    protected double epsilon; // Abbruchgenauigkeit  
    protected double startwert; // Auswertestelle  
  
    protected abstract double funktion(double x); // Funktionsprototyp  
  
    public abstract double berechne();  
    // Hier wird der jeweilige Algorithmus implementiert.  
    public nullstellenAlgorithmus() {  
        epsilon = 1.0E-3;  
        startwert = 0.0;  
    }  
  
    public nullstellenAlgorithmus(double eps, double start) {  
        epsilon = eps;  
        startwert = start;  
    }  
  
    public nullstellenAlgorithmus(double eps) {  
        epsilon = eps;  
        startwert = 0.0;  
    }  
  
    public void setStartwert(double x) {  
        startwert = x;  
    }  
  
    public void setEpsilon(double eps) {  
        epsilon = eps;  
    }  
  
    public double getStartwert() {  
        return startwert;  
    }  
  
    public double getEpsilon() {  
        return epsilon;  
    }  
}
```

Die Bisektion wird jetzt hiervon abgeleitet:

```
public class bisektion extends nullstellenAlgorithmus {  
    private double xlinks, xrechts;  
    private int maxIter, aktIter; // Integer-Variable für "Notbremse"  
  
    public bisektion() {  
        xlinks = 0.0;  
        xrechts = 1.0;  
        startwert=0.5;  
    }  
}
```

```
        epsilon=1.0E-3;
        maxIter = 50;
        aktIter = 0;
    }

    public bisektion(double links, double rechts, double eps) {
        xlinks = links;
        xrechts = rechts;
        epsilon = eps;
        startwert = 0.5*(links + rechts);
        maxIter = 50;
        aktIter = 0;
    }

    public bisektion(double links, double rechts) {
        xlinks = links;
        xrechts = rechts;
        epsilon = 1.0E-3;
        startwert = 0.5*(links + rechts);
        maxIter = 50;
        aktIter = 0;
    }

    public void setMaxIter(int max) {
        maxIter = max;
    }

    public double funktion(double x) {
        return 5.0*Math.sin(x) - x;
    }

    public int getAnzahlIterationen() {
        return aktIter;
    }

    public double berechne() {
        int counter=0;
        double xmid, dx, fmid, f, result;

        f = funktion(xlinks);
        fmid = funktion(xrechts);

        if (f*fmid >= 0) {
            System.out.println("Keine Klammer übergeben für Bisektion");
            return xlinks;
        }

        if (f < 0) {
            // Orientieren
            result = xlinks;
            dx = xrechts - xlinks;
        }
        else {
            result = xrechts;
            dx = xlinks - xrechts;
        }

        // Schleife für die Bisektion
```

```

while ((counter <= maxIter)&&
      ((Math.abs(dx)>=epsilon) || (Math.abs(fmid)>=epsilon))) {
    dx = 0.5*dx;
    xmid = result + dx;
    fmid = funktion(xmid);
    counter++;
    if (fmid <= 0.0) result = xmid;
}
aktIter = counter;

return result;
}
}

```

Für $\varepsilon = 0,05$ erhält man dann die folgende Ausgabe:

```

Die berechnete Lösung der Bisektion: 2.6016314162540475
Die Abbruchgenauigkeit war 0.05.
Es wurden 5 Iterationen durchgeführt.
Der Funktionswert an der Lösung:
-0.031117695287938396.

```

$\varepsilon = 0.0005$ ergibt:

```

Die berechnete Lösung der Bisektion: 2.5958789882994764
Die Abbruchgenauigkeit war 5.0E-4.
Es wurden 13 Iterationen durchgeführt.
Der Funktionswert an der Lösung:
-7.378228040719748E-4.

```

Für die Regula Falsi wird eine entsprechende Klasse `regulaFalsi` implementiert; dabei unterscheidet sich nur die Funktion `regulaFalsi.berechne()` von der Klasse `bisektion`.

```

public double berechne() {
    int counter=0;
    double xs, dx, x1, y1, fx, fy;

    x1 = xlinks;
    y1 = xrechts;

    fx = funktion(x1);
    fy = funktion(y1);

    if (fx*fy >= 0) {
        System.out.println("Keine Klammer übergeben für die Regula False");
        return xlinks;
    }
    fx = funktion(x1);
    fy = funktion(y1);

    // Schleife für die Bisektion
    while ((counter <= maxIter)&&((fx>=epsilon) || fy >= epsilon)) {

        xs = x1 - fx*((y1-x1)/(fy-fx));
        if (fx*funktion(xs)<0.0) {
            y1 = xs;
            fy = funktion(xs);
        }
        else {
            x1 = xs;

```

```

        fx = funktion(xs);
    }
    counter++;
}
aktIter = counter;

return 0.5*(x1+y1);
}

```

Mit Abbruchgenauigkeit $\varepsilon = 0,05$ erhält man dann die Ausgabe

```

Die berechnete Lösung der Regula Falsi: 2.593950915988187
Die Abbruchgenauigkeit war 0.05.
Es wurden 3 Iterationen durchgeführt.
Der Funktionswert an der Lösung:
0.009425588181722677.

```

Abbruchgenauigkeit $\varepsilon = 0,0005$:

```

Die berechnete Lösung der Regula Falsi: 2.5957265157620166
Die Abbruchgenauigkeit war 5.0E-4.
Es wurden 5 Iterationen durchgeführt.
Der Funktionswert an der Lösung:
6.625448183594784E-5.

```

9. Motivieren Sie die Fallunterscheidung für die Bestimmung von h bei der Nullstellenbestimmung durch quadratische Interpolation!

Lösung:

Bei einer Iteration liegen drei Näherungswerte x_{n-2}, x_{n-1} und x_n mit zugehörigen Funktionswerten $y_{n-2} = f(x_{n-2}), y_{n-1} = f(x_{n-1}), y_n = f(x_n)$ vor. Dabei ist der Punkt x_n die beste Näherung. Wir setzen ein quadratisches Polynom an durch

$$p_2(x) = C + B(x - x_n) + A(x - x_n)^2.$$

Aus den Interpolationsbedingungen erhält man schnell

$$A = \frac{h_1 d_2 - h_2 d_1}{h_2 h_1 (h_2 - h_1)},$$

$$B = \frac{h_2^2 d_1 + h_1^2 d_2}{h_2 h_1 (h_2 - h_1)}, C = y_n,$$

mit den Hilfsgrößen

$$h_2 = x_{n-2} - x_n, h_1 = x_{n-1} - x_n,$$

$$d_2 = y_{n-2} - y_n, d_1 = y_{n-1} - y_n.$$

Jetzt berechnen wir eine neue Näherung x_{n+1} durch den Ansatz $h = x_{n+1} - x_n$. Dazu bestimmen wir h so, dass h eine Nullstelle des quadratischen Polynoms

$$C + Bh + Ah^2 = 0$$

ist. Der Fall $A = B = 0$ tritt nur dann auf, wenn $y_{n-2} = y_{n-1} = y_n$ ist, den wir vernachlässigen können. Ist $A = 0$ und $B \neq 0$, erhält man nur eine Lösung für h , die drei aktuellen Näherungen liegen auf einer Geraden. Für $A \neq 0$ und $B = 0$ erhält man zwei Lösungen für h , wir wählen willkürlich eine der beiden aus.

Im allgemeinen Fall erhält man zwei Lösungen, wobei man für die Berechnung von x_{n+1} die mit dem kleineren Betrag benutzt. Insgesamt:

$$h = \begin{cases} -\frac{C}{B} & A = 0, B \neq 0, \\ \sqrt{-\frac{C}{A}} & A \neq 0, B = 0, \\ \frac{-2\operatorname{sgn}(B)C}{|B| + \sqrt{B^2 - 4AC}} & \text{sonst.} \end{cases}$$

Dabei bezeichnet $\operatorname{sgn}(B)$ das Vorzeichen der Zahl B .

10. Implementieren Sie die Nullstellenbestimmung mit Hilfe von quadratischer Interpolation und bestimmen Sie damit auf zwei und vier Dezimalstellen genau die im Intervall $[\frac{\pi}{2}; \pi]$ liegende Lösung von $5 \sin(x) = x$.

Lösung:

Eine grafische Darstellung der Funktion finden Sie in Abbildung 13.2 auf 160.

Mit Hilfe der in Aufgabe 8 implementierten Klassen wird eine Klasse für den Algorithmus quadratischeInterpolation abgeleitet; wichtig ist die berechne-Funktion:

```
public double berechne() {
    int counter=0;
    double a, b, c, d, e, fa, fb, fc, p, q, r, s, xm;

    a = xlinks;
    b = xrechts;
    fa = funktion(a); fb = funktion(b);
    if (fa*fb > 0.0) {
        System.out.println
            ("Keine Klammer übergeben für quadratische Interpolation");
        return xlinks;
    }
    c = b;
    fc = fb;
    xm = 0.5*(a+b);
    e = 1.0; d = e;
    // Schleife für die Bisektion
    while (counter <= maxIter&&
        (Math.abs(fb) > epsilon || Math.abs(xm) > epsilon)) {

        if ((fb > 0.0 && fc > 0.0) || (fb < 0.0 && fc < 0.0)) {
            c = a;
            fc = fa;
            d = b-a;
            e = d;
        }
        if (Math.abs(fc) < Math.abs(fb)) {
            a = b;
            b = c;
            c = a;
            fa = fb;
            fb = fc;
            fc = fa;
        }
        xm = 0.5*(c-b);

        s = fb/fa;
        if (a == c) {
            p = 2.0*xm*s;
```

```

        q = 1.0-s;
    }
    else {
        q = fa/fc;
        r = fb/fc;
        p = s*(2.0*xm*q*(q-r)-(b-a)*(r-1.0));
        q = (q-1.0)*(r-1.0)*(s-1.0);
    }
    if (p>0.0) q = -q;
    p = Math.abs(p);
    e = d;
    d = p/q;
    a = b;
    fa = fb;

    // Neuen Punkt berechnen
    b = b+d;
    fb = funktion(b);

    counter++;
}
aktIter = counter;
return b;
}

```

Für die `signum`-Funktion verwenden Sie entweder eine eigene Implementierung oder die ab Java 1.5 zur Verfügung stehende `Math.signum`-Funktion.

Für das Nullstellenproblem aus Aufgabe 8 mit den beiden verwendeten Abbruchgenauigkeiten erhält man folgende Ergebnisse:

```

Die berechnete Lösung der quadratischen Interpolation: 2.5957364004011003
Die Abbruchgenauigkeit war 0.05.
Es wurden 4 Iterationen durchgeführt.
Der Funktionswert an der Lösung:
1.4128800668977703E-5.

```

```

Die berechnete Lösung der quadratischen Interpolation: 2.595739079312127
Die Abbruchgenauigkeit war 5.0E-4.
Es wurden 5 Iterationen durchgeführt.
Der Funktionswert an der Lösung:
1.7806880414639181E-9.

```

11. Wo ist die Funktion $f(x) = |1 + x| - |1 - x|$ monoton?

Lösung:

Auf $[-1, 1]$ streng monoton wachsend, sonst konstant

12. Welche der folgenden Funktionen sind umkehrbar, wie lauten die Umkehrfunktionen: $f_1(x) = \sqrt{x+1}$, $f_2(x) = \sqrt{x^2+1}$, $f_3(x) = \ln|x| - x$ und $f_4(x) = \frac{x-2}{x+3}$?

Lösung:

$$f_1^{-1}(x) = x^2 - 1;$$

$$f_2^{-1}(x) = \sqrt{x^2 - 1};$$

f_3 ist umkehrbar, allerdings ohne dass die inverse Funktion geschlossen darstellbar ist;

$$f_4^{-1}(x) = \frac{3x+2}{1-x}.$$

13. Differenzieren Sie die folgenden Funktionen: $(100 - 4x^2 + 3x)(1 + 2x^2)$, $\frac{x^2}{-1+x^2}$, $\frac{1}{-1+x}$ und $\frac{4x}{(-1+x^2)(1+x)}$ und $\sqrt[3]{1-x^2}$!

Lösung:

$$3 + 392x + 18x^2 + 32x^3; \frac{-2x}{(x^2-1)^2}; \frac{-1}{(x-1)^2} + 4 \frac{1+x^2+2x^3}{(-1+x^2)^2(1+x)^2}; -\frac{2}{3} \frac{x}{(1-x^2)^{2/3}}.$$

14. An welcher Stelle x_0 hat die Ableitung der Funktion $f(x) = (x-1) \ln(x)$ den Wert 1?

Lösung:

Die Ableitung der angegebenen Funktion ist gegeben durch

$$f'(x) = \ln(x) + \frac{x-1}{x}.$$

Die gesuchte Stelle x ist dann eine Lösung der Nullstellenaufgabe

$$\ln(x) + \frac{x-1}{x} - 1 = 0.$$

Die Funktion $g(x) = \ln(x) + \frac{x-1}{x} - 1$ hat die Funktionswerte

$$g(1) = -1, g(2) = 0,93\ 147\ 181.$$

Also muss nach dem Zwischenwertsatz im Intervall $[1; 2]$ eine Lösung liegen, das zeigt auch die Abbildung 13.3. Sie können Bisektion, Regula Falsi oder auch quadratische Interpolation verwenden, es gilt $x \sim 1,763\ 211\ 941$.

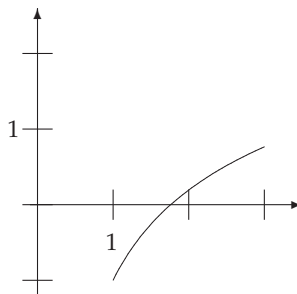


Abbildung 13.3: Die Funktion $g(x) = \ln(x) + \frac{x-1}{x} - 1$

15. Leiten Sie ab: $e^{-kx} \sin(\omega x)$, $\arcsin(\sqrt{x})$, $\arctan(x) + \operatorname{arccot}(\frac{1}{x})$ und $\arccos(1-2x)$.

Lösung:

$$e^{-kx}(\omega \cos(\omega x) - k \sin(\omega x)), \frac{1}{2} \frac{1}{\sqrt{x(1-x)}}, \frac{2}{1+x^2}, \frac{1}{\sqrt{x(1-x)}}$$

16. Bestimmen Sie die Ableitungen der Hyperbelfunktionen: $\sinh(x) = \frac{e^x + e^{-x}}{2}$ und $\cosh(x) = \frac{e^x - e^{-x}}{2}$.

Lösung:

$$\cosh(x), \sinh(x)$$

17. Implementieren Sie das Newton-Verfahren und bestimmen Sie damit auf zwei und vier Dezimalstellen genau die im Intervall $[\frac{\pi}{2}; \pi]$ liegende Lösung von $5 \sin(x) = x$.

Lösung:

Wie bereits bei den Aufgaben 8 und 10 wird eine entsprechende Klasse abgeleitet. Diese hat die folgende berechne-Funktion:

```

public double berechne() {
    int counter=0;
    double x, dx, fx;

    x = startwert;
    fx = funktion(x);

    // Schleife für die Iteration
    while ((counter <= maxIter)&&
           ((Math.abs(fx)>=epsilon))) {

        x = x - fx/ableitung(x);
        fx = funktion(x);
        counter++;

    }
    aktIter = counter;
    return x;
}

```

Für die beiden gewünschten Abbruchgenauigkeiten $\varepsilon = 5 \cdot 10^{-2}$ und $\varepsilon = 5 \cdot 10^{-4}$ und Startwert $x_1 = \pi$ erhält man dann die folgenden Ergebnisse:

```

Die berechnete Lösung des Newton-Verfahrens: 2.595856717362374
Die Abbruchgenauigkeit war 0.05.
Es wurden 2 Iterationen durchgeführt.
Der Funktionswert an der Lösung:
-6.203713029453439E-4.

```

```

-----
Die berechnete Lösung des Newton-Verfahrens: 2.59573908305506
Die Abbruchgenauigkeit war 5.0E-4.
Es wurden 3 Iterationen durchgeführt.
Der Funktionswert an der Lösung:
-1.795737869514369E-8.

```

18. Gesucht ist eine Lösung der nichtlinearen Gleichung $e^{2x} - \sin x - 2 = 0$ mit $x \geq 0$. Verwenden Sie für die Regula Falsi die Startwerte $x_0 = 0, x_1 = 1$, für das Newton-Verfahren $x_0 = 0, 25$. Vergleichen Sie die Ergebnisse!

Lösung:

Die Startwerte für die Regula Falsi sind schlecht gewählt. Wenn wie in der Lösung von Aufgabe 8 der Mittelpunkt des letzten berechneten Intervalls von der Regula Falsi zurückgegeben wird, dann erhält man nach der maximalen Anzahl der Iterationen eine sehr schlechte Näherung. Das liegt daran, dass immer nur die linke Grenze verändert wird im Verlauf des Verfahrens. Diese liegt immer näher an der gesuchten Näherung, der Korrekturterm wird immer kleiner.

Das Newton-Verfahren konvergiert schnell gegen eine Näherung:

```

Die berechnete Lösung der Regula Falsi: 0.4719103765419831
Die Abbruchgenauigkeit war 0.05.
Es wurden 10001 Iterationen durchgeführt.
Der Funktionswert an der Lösung:
0.11519249547652777.
-----
Die berechnete Lösung des Newton-Verfahrens: 0.4438497254104666
Die Abbruchgenauigkeit war 5.0E-4.
Es wurden 3 Iterationen durchgeführt.

```

Der Funktionswert an der Lösung:
1.1460918404848641E-4.

19. Bestimmen Sie eine Approximation von $f(x) = \sin(2x)$ durch ein Polynom vom Grad 4 an der Entwicklungsstelle $x_0 = \frac{\pi}{6}$!

Lösung:

$$p_4(x) = \frac{\sqrt{3}}{2} + \frac{1}{2}(x-a) - \frac{\sqrt{3}}{3}(x-a)^2 - \frac{1}{6}(x-a)^3 + 15\sqrt{3}(x-a)^4.$$

20. Stellen Sie das Taylorpolynom für $f(x) = \ln\left(\frac{1+x}{1-x}\right)$ auf und schätzen Sie das Restglied ab!

Lösung:

Es gilt $f^{(n)}(0) = (-2)^n(n-1)!$ und $f^{(n)}(x) = \frac{f^{(n)}(0)}{(1+x)^n}$. Damit kann das Restglied abgeschätzt werden durch $\frac{2}{n+1}|x|^{n+1}$.

21. Berechnen Sie die folgenden Grenzwerte: $\lim_{x \rightarrow 0} \frac{\sin(5x)}{x}$, $\lim_{x \rightarrow \infty} x(1 - \cos(\frac{1}{x}))$ und $\lim_{x \rightarrow 0} \sin(x) \cdot \ln(x)$.

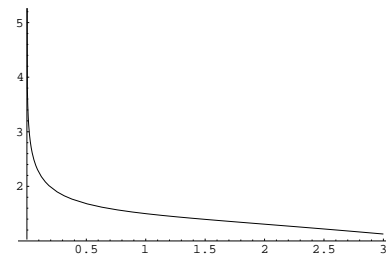
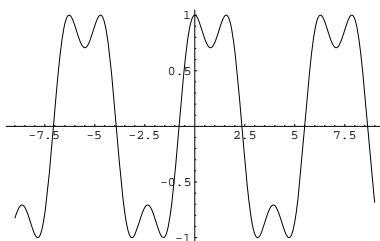
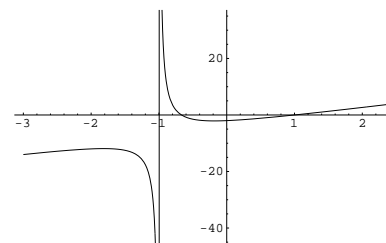
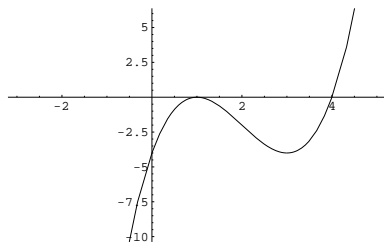
Lösung:

5; 0; 0

22. Untersuchen Sie die Funktionen $f(x) = x^3 - 6x^2 + 9x - 4$, $f(x) = \frac{3x^2 - x - 2}{x+1}$, $f(x) = \sin^3(x) + \cos^3(x)$, $f(x) = \frac{x}{2} - \frac{x+2}{4} \ln(x+1)$ auf Nullstellen, lokale Extrema, Wendepunkte, Krümmung, und skizzieren Sie den Graphen.

Lösung:

So sollte es aussehen:



23. Berechnen Sie die Lösung des Interpolationsproblems $x_0 = 0, x_1 = 1, x_2 = 3, y_0 = 0, y_1 = 1, y_2 = 3$ mit Hilfe der Lagrange-Polynome und des Newton-Algorithmus. Berechnen Sie die Funktionswerte des Interpolationspolynoms an den Stellen $x = 0,5$ und $x = 2,5$!

Lösung:

Die Lösung ist $p(x) = x$; daraus müssen sich durch den Neville-Algorithmus die Funktionswerte $p(0,5) = 0,5$ und $p(2,5) = 2,5$ ergeben!

24. Interpolieren Sie die Funktion $\ln(x)$ durch ein quadratisches Polynom in den Stützstellen $x = 10, 11, 12$. Schätzen Sie den Interpolationsfehler an der Stelle $x = 11,1$ ab!

Lösung:

$$p(x) = \ln 10 + (x - 10)(-\ln 10 + \ln 11 + \frac{1}{2}(x - 11)(\ln 10 - 2 \ln 11 + \ln 12)).$$

An der Stelle 11,1 ist der Fehler kleiner oder gleich $3,310^{-5}$, die dritte Ableitung von \ln auf $[10; 12]$ ist abschätzbar durch $\frac{1}{500}$.

25. Interpolieren Sie die Funktion $f(x) = e^{-x}$ durch Hermite-Interpolation an den Stellen $x_0 = 0,3$ und $x_1 = 0,4$. Bestimmen Sie den Fehler an der Stelle $x = 0,34$!

Lösung:

$$p(x) = 0,7408(H_0^3(t) - 0,1H_1^3(t)) + 0,6703(0,1H_2^3(t) - H_3^3(t)),$$

Der Fehler an der Stelle 0,34 ist $1,69492 \cdot 10^{-7}$.

26. Implementieren Sie den Neville-Algorithmus und stellen Sie die Fehlerfunktion $r(x) = f(x) - P_{10}(x)$ für $f(x) = e^{-3x}$ im Intervall $[0; 5]$ grafisch dar! Das Polynom P_{10} soll durch die 11 Stützstellen $x_k = 0,5 \cdot k, 0 \leq k \leq 10$ bestimmt werden.

Lösung:

Es ist kein Programmierfehler, wenn Sie wie in Abbildung 13.4 die großen Abweichungen am Rand erhalten, Diese Abweichungen ergeben sich durch den Runge-Effekt!

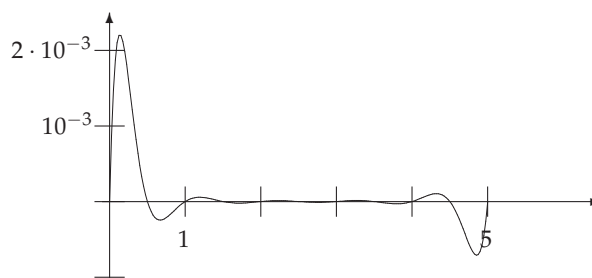


Abbildung 13.4: Die Fehlerfunktion $r(x) = e^{-3x} - P_{10}(x)$ für Aufgabe 26

Der Neville-Algorithmus als Java-Funktion:

```
private static double eval(double z) {
    int i, k;
    doubleArray1D t = new doubleArray1D(N);

    for (i=0; i<N; i++) {
        t.set(i, y.get(i));
        for (k=i-1; k>=0; k--)
            t.set(k, t.get(k+1) +
                (t.get(k+1)-t.get(k)) * (z-x.get(i)) / (x.get(i)-x.get(k)));
    }

    return t.get(0);
}

private static doubleArray1D x;
private static doubleArray1D y;
private static int N;
```

Kapitel 14

Integralrechnung

Verständnisfragen

Sachfragen

1. Erläutern Sie den Begriff des bestimmten Integrals!
2. Wie lautet der Mittelwertsatz der Integralrechnung?
3. Erläutern Sie den Begriff der Stammfunktion!
4. Ist die Stammfunktion einer Funktion eindeutig bestimmt?
5. Wie hängt das Integral einer stetigen Funktion von der oberen Integrationsgrenze ab?
6. Wie lautet der Hauptsatz der Differenzial- und Integralrechnung?
7. Was versteht man unter partieller Integration?
8. Wie lautet die Substitutionsregel?
9. Was ist ein unbestimmtes Integral?
10. Was ist eine Quadraturformel?
11. Was ist eine Rechtecksregel?
12. Erläutern Sie die Trapezregel und ihre Herleitung!
13. Erläutern Sie die Kepler'sche Fassregel und ihre Herleitung!
14. Wie lautet die Simpson-Regel!
15. Was versteht man unter einer Newton-Côtes-Formel?
16. Erläutern Sie die Gauß-Quadratur!
17. Was ist eine gewöhnliche Differenzialgleichung?
18. Was versteht man unter einer allgemeinen, was unter einer speziellen Lösung einer gewöhnlichen Differenzialgleichung?
19. Wie kann das Euler-Verfahren motiviert werden?

20. Erläutern Sie den Verlauf des Euler-Verfahrens!
21. Was ist ein Einschrittverfahren?
22. Wie ist der lokale Diskretisierungsverfahren eines Diskretisierungsverfahrens definiert?
23. Was ist die Konsistenzordnung eines Diskretisierungsverfahrens?
24. Wie ist der globale Fehler eines Diskretisierungsverfahrens definiert?
25. Welche Konsistenzordnung besitzt das Euler-Verfahren?
26. Welche Konvergenzordnung besitzt das Euler-Verfahren? Welche Voraussetzungen müssen dafür erfüllt sein?
27. Wie kann die Konvergenzordnung eines Diskretisierungsverfahrens visualisiert werden?
28. Was versteht man unter einem Differenzialgleichungssystem?
29. Was ist ein Vektorfeld?
30. Wie kann ein Differenzialgleichungssystem mit dem Euler-Verfahren gelöst werden?
31. Erläutern Sie den bei einem Einzelschrittverfahren verwendete Ansatz!
32. Wie lautet das Verfahren von Heun?
33. Welche Quadraturformel liegt dem Verfahren von Heun zu Grunde?
34. Wie lautet das Verfahren von Runge-Kutta?
35. Welche Quadraturformel liegt dem Verfahren von Runge-Kutta zu Grunde?

Methodenfragen

1. Bestimmte Integrale durch Grenzwerte von Summen berechnen können.
2. Flächeninhalte durch bestimmte Integrale berechnen können.
3. Die Stammfunktion einer gegebenen Funktion angeben können.
4. Mit Hilfe eines Richtungsfelds eine Stammfunktion skizzieren können.
5. Mit Hilfe des Hauptsatzes der Differenzial- und Integralrechnung bestimmte und unbestimmte Integrale berechnen können.
6. Partielle Integration anwenden können.
7. Die Substitutionsregel anwenden können.
8. Entscheiden können, ob ein unbestimmtes Integral existiert.
9. Ein existierendes unbestimmtes Integral berechnen können.
10. Mit Hilfe von Quadraturformeln ein Integral annähern können.
11. Newton-Côtes-Formeln implementieren und anwenden können.
12. Gauss-Legendre-Formeln anwenden und implementieren können.
13. Für einfache gewöhnliche Differenzialgleichungen die allgemeine und die spezielle Lösung angeben können.

14. Das Euler-Verfahren anwenden und implementieren können.
15. Den globalen Fehler eines Diskretisierungsverfahrens grafisch darstellen können.
16. Das Heun-Verfahren anwenden und implementieren können.
17. Das Verfahren von Runge-Kutta anwenden und implementieren können.

Übungsaufgaben

1. Berechnen Sie über eine Grenzwertbetrachtung für die Unter- und Obersummen das bestimmte Integral über $f(x) = 4 - x^2$ von $a = 0$ nach $b = 2$.

Lösung:

Für den Grenzübergang bilden wir Unterteilungen des Intervalls $[0; 2]$ mit Schrittweite $h = \frac{1}{2^k}$. Das entspricht $n = 2^k$ und $x_{i+1} - x_i = \frac{1}{2^{k-1}}$.

Die Funktion $f(x) = 4 - x^2$ ist streng monoton fallend im Integrationsintervall; also ist die Obersumme gegeben durch

$$O_k = \frac{1}{2^{k-1}} \sum_{i=1}^n f(x_{i-1}) = 8 - \frac{1}{2^{3k-1}} \sum_{i=1}^{n-1} i^2.$$

Verwendet man die Summenformel

$$\sum_{i=1}^n i^2 = \frac{1}{n(n+1)(2n+1)},$$

dann erhält man für die Obersumme den Grenzwert $\frac{16}{3}$. Analog ergibt sich der gleiche Grenzwert für die Untersumme.

2. Schätzen Sie die bestimmten Integrale $\int_1^2 e^{-x^2} dx$ und $\int_{2\pi}^{3\pi} \frac{\sin(x)}{x} dx$ nach unten und oben ab!

Lösung:

Der Integrand ist monoton steigend, also werden Minimum und Maximum an den Rändern angenommen:

$$e^{-1} \leq \int_1^2 e^{-x^2} dx \leq e^{-2},$$

$$0 \leq \int_{2\pi}^{3\pi} \frac{\sin x}{x} dx \leq 0,13\pi.$$

3. Skizzieren Sie das Richtungsfeld für die Funktion $f(x) = -\frac{1}{x^2}$ und einige mögliche Stammfunktionen. Vergleichen Sie Ihre Skizzen mit der exakten Stammfunktion!

Lösung:

Die exakte Stammfunktion für das Richtungsfeld ist gegeben durch

$$F(x) = x^{-1} + C.$$

In Abbildung 14.1 sehen Sie eine Skizze des Richtungsfelds und zwei eingezeichnete Lösungen.

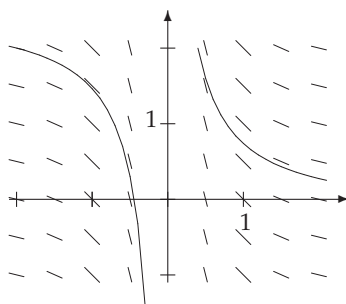


Abbildung 14.1: Das Richtungsfeld für die Funktion $f(x) = -\frac{1}{x^2}$; die linke Stammfunktion ist gegeben durch den Anfangswert $(-2, 2)$, die Stammfunktion im ersten Quadranten ist gegeben durch den Startwert $(0, 4, 2)$

4. Geben Sie die folgenden unbestimmten Integrale an: $\int (2e^x - 10^x) dx$, $-\int a^x dx$, $\int (e^t - 4^t) dt$, $\int \frac{1 - \cos^2(x)}{\cos^2(x)} dx$ und $\int \frac{1}{2 - 2x^2} dx$!

Lösung:

$$\int (2e^x - 10^x) dx = e^x - \frac{10^x}{\ln(10)} + C,$$

$$-\int a^x dx = \frac{-a^x}{\ln(a)} + C,$$

$$\int (e^t - 4^t) dt = e^t - \frac{4^t}{\ln(4)} + C,$$

$$\int \frac{1 - \cos^2(x)}{\cos^2(x)} dx = \frac{\sin(x) - x \cos(x)}{\cos(x)} + C,$$

$$\int \frac{1}{2 - 2x^2} dx = \frac{1}{4} (\ln(x - 1) - \ln(x + 1)) + C.$$

5. Berechnen Sie die folgenden bestimmten Integrale: $\int_0^3 (x - 1) dx$, $\int_0^{2\pi} \sin(x) dx$, $\int_1^4 (x^2 - 4x) dx$ und $\int_1^5 \frac{5}{x^2} dx$!

Lösung:

$$\int_0^3 (x - 1) dx = \frac{3}{2},$$

$$\int_0^{2\pi} \sin(x) dx = 0$$

$$\int_1^4 (x^2 - 4x) dx = -9,$$

$$\int_1^5 \frac{5}{x^2} dx = 4$$

6. Berechnen Sie mit Hilfe der Substitutionsregel oder partieller Integration: $\int \frac{8x-3}{4x^2-3x+2} dx$, $\int \frac{1}{x \ln x} dx$, $\int \frac{x}{\cos^2 x^2} dx$, $\int x e^x dx$, $\int \frac{\ln x}{x} dx$ und $\int e^x \sin x dx$.

Lösung:

$$\int \frac{8x-3}{4x^2-3x+2} dx = \ln(2 - 3x + 4x^3) + C,$$

$$\int \frac{1}{x \ln(x)} dx = \ln(\ln(x)) + C,$$

$$\int \frac{x}{\cos^2 x^2} dx = \frac{1}{2} \tan(x^2) + C,$$

$$\int x e^x dx = e^x(x-1) + C,$$

$$\int \frac{\ln(x)}{x} dx = \frac{1}{2} \ln(x)^2 + C,$$

$$\int e^x \sin(x) dx = -\frac{1}{2} e^x (\cos(x) - \sin(x)) + C$$

7. Bestimmen Sie eine Rekursionsformel für die Integrale $I_n = \int_0^1 x^n e^{-x} dx$.

Lösung:

Der Startwert der Rekursion ist das Integral

$$I_0 = \int_0^1 e^{-x} dx = \int_{-1}^0 e^x dx = 1 - \frac{1}{e}.$$

Dabei wurde die Substitution $x = -x$ verwendet.

Partielle Integration führt dann für I_{n+1} zu

$$I_{n+1} = 1 - (n+1)I_n.$$

8. Berechnen Sie die Fläche, die vom Graphen der Funktion $f(x) = x^2 - 5x + 6$ und der x -Achse im Intervall $[1; b]$ eingeschlossen wird!

Lösung:

Die Funktion hat Nullstellen in $x = 2$ und $x = 3$. Links von $x = 2$ und rechts von $x = 3$ ist die Funktion oberhalb der x -Achse. In Abbildung 14.2 sehen Sie den Verlauf der Funktion mit Definitionsbereich $[1; 4]$.

Für ein $b \leq 2$ ist die Fläche oberhalb der x -Achse gegeben durch

$$A(b) = \frac{1}{3}b^3 + 6b - \frac{5}{2}b^2 - \frac{23}{6}.$$

Für $b = 2$ erhält man die Fläche $A(2) = \frac{5}{6}$.

Für ein $b \in (2; 3]$ muss das Vorzeichen für den Bereich von 2 bis b umgekehrt werden:

$$A(b) = \int_1^2 f(x) dx - \int_2^b f(x) dx = \frac{5}{6} - \int_2^b (x^2 - 5x + 6) dx = -\frac{1}{3}b^3 + \frac{5}{2}b^2 - 6b + \frac{11}{2}.$$

Für $b = 3$ erhält man dann die Fläche $A(3) = 1$.

Rechts von $x = 3$ ist dann

$$A(b) = 1 + \int_3^b f(x) dx = \frac{1}{3}b^3 - \frac{5}{2}b^2 + 6b - \frac{7}{2}.$$

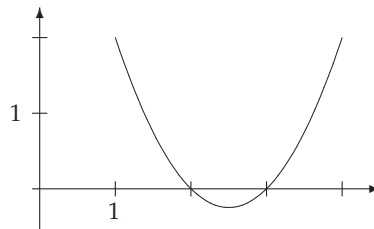


Abbildung 14.2: Der Verlauf des Integranden in Aufgabe 8

9. Berechnen Sie die Fläche zwischen den Graphen von $f(x) = x + 1$ und $g(x) = e^{-x}$ im Intervall $[0; 3]$!

Lösung:

Beide Integranden verlaufen oberhalb der x -Achse; und schneiden sich im Punkt $(0, 1)$. Also ist die gesuchte Fläche gegeben durch

$$A = \int_0^3 x + 1 - e^{-x} dx = \left[x + \frac{1}{2}x^2 - e^{-x} \right]_0^3 = \frac{17}{2} - \frac{1}{e^3}.$$

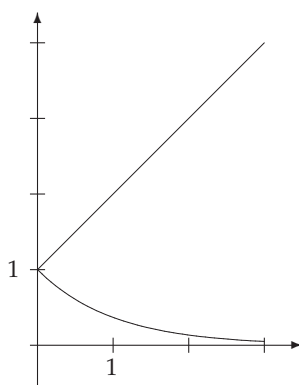


Abbildung 14.3: Der Verlauf des Integranden in Aufgabe 9

10. Untersuchen Sie, ob die folgenden uneigentlichen Integrale existieren: $\int_0^\infty \cos(x) dx$, $\int_{-\infty}^\infty \frac{1}{1+x^2} dx$, $\int_1^4 \frac{x}{\sqrt{|x^2-4|}} dx$, $\int_0^4 \frac{2}{x\sqrt{x}} dx$ und $\int_0^{32} \frac{1}{\sqrt[5]{x^3}} dx$!

Lösung:

$\int_0^\infty \cos(x) dx$ existiert nicht;

$\int_{-\infty}^\infty \frac{1}{1+x^2} dx$ existiert und hat den Wert π ;

$\int_1^4 \frac{x}{\sqrt{|x^2-4|}} dx$ existiert und hat den Wert $3\sqrt{3}$,

$\int_0^4 \frac{2}{x\sqrt{x}} dx$ existiert nicht,

$\int_0^{32} \frac{1}{\sqrt[5]{x^3}} dx$ existiert und hat den Wert 10.

11. Leiten Sie die Kepler'sche Fassregel durch Integration des quadratischen Interpolationspolynoms her!

Lösung:

Die Stützstellen seien durch die äquidistanten Punkte x_{i-1} , x_i und x_{i+1} gegeben. Wenn die Teilintervalle äquidistant sind, dann ist das Polynom mit $h = x_i - x_{i-1} = x_{i+1} - x_i$ bestimmt als

$$p(x) = f_{i-1} + \frac{f_i - f_{i-1}}{h}(x - x_{i-1}) + \frac{f_{i+1} - 2f_i + f_{i-1}}{2h^2}(x - x_{i-1})(x - x_i).$$

Dann ist $\int_{x_{i-1}}^{x_{i+1}} p(x) dx$ gegeben durch

$$\begin{aligned} \int_{x_{i-1}}^{x_{i+1}} p(x) dx &= f_{i-1} \int_{x_{i+1}}^{x_{i+1}} dx \\ &+ \frac{f_i - f_{i-1}}{h} \int_{x_{i+1}}^{x_{i+1}} (x - x_{i-1}) dx \\ &+ \frac{f_{i+1} - 2f_i + f_{i-1}}{2h^2} \int_{x_{i+1}}^{x_{i+1}} (x - x_{i-1})(x - x_i) dx. \end{aligned}$$

Alle Integrale können mit der Substitution $u = x - x_{i-1}$ gelöst werden es ist dann

$$\int_{x_{i+1}}^{x_{i+1}} (x - x_{i-1}) dx = 2(f_i - f_{i-1})h, \int_{x_{i+1}}^{x_{i+1}} (x - x_{i-1})(x - x_i) dx = \frac{h}{3}(f_{i+1} - 2f_i + f_{i-1}).$$

Addiert man diese drei Ergebnisse, ergibt sich die Fassregel.

12. Nähern Sie die Integrale $\int_0^1 \sqrt{3x+4} dx$ und $\int_0^1 4\sqrt{1-x^2} dx$ mit der Trapez- und der Simpsonregel an und vergleichen Sie die Ergebnisse mit den exakten Werten!

Lösung:

Die Stammfunktion von $\int \sqrt{3x+4} dx$ finden Sie als Beispiel für die Anwendung der Substitutionsregel im Buch. Es ist

$$\int \sqrt{3x+4} dx = \frac{2}{9}(3x+4)^{\frac{3}{2}}.$$

Für das bestimmte Integral gilt dann $I \approx 2,337835$.

Die Trapezregel mit $n = 2$ ergibt die Näherung 2,322875656. Für $n = 5$ und der Zerlegung $x_0 = 0, x_i = \frac{i}{4}$ ergibt sich die Näherung 2,336883252.

Für $n = 11$ und $h = \frac{1}{10}$ erhält man 2,337682861.

Die Fassregel ergibt die Näherung 2,337763805.

Für $k = 2$ und $h = 0,25$ ergibt die Simpson-Regel die Näherung 2,337830413.

$k = 5$ und $h = 0,1$ bei der Simpson-Regel ergibt 2,337835241.

Die exakte Lösung des zweiten Integrals ist π , Sie finden die Integration wiederum als Beispiel der Substitutionsregel; das Ergebnis ist die Fläche eines Kreises mit Radius 1.

Die Trapezregel für $n = 2$ ergibt die Näherung 2, $n = 5$ ergibt 2,995709068, $n = 11$ die Näherung 3,104518326.

Die Fassregel ergibt die Näherung 2,976067743; $k = 2$ die Näherung 3,083595155 und $k = 5$ die Näherung 3,127008159.

Implementierungen in Java finden Sie in der Lösung der Aufgabe 14 auf Seite 178!

13. Berechnen Sie das Integral $\int_{1,05}^{1,35} f(x) dx$ mit Hilfe der Simpson-Regel, falls für f die Werte in Tabelle 14.1 bekannt sind!

Tabelle 14.1: Wertetabelle für Aufgabe 13

x	1,05	1,10	1,15	1,2	1,25	1,3	1,35
$f(x)$	2,36	2,5	2,74	3,04	3,46	3,98	4,6

Lösung:

Die Näherung durch die Simpson-Regel für $k = 3$ und $h = 0,05$ ist 0,957333333.

Die Implementierung in Java finden Sie in der Lösung der Aufgabe 14 auf Seite 178!

14. Implementieren Sie verschiedene Newton-Côtes-Formeln und die Gauß-Quadratur in der Programmiersprache Ihrer Wahl und testen Sie die Implementierung an Hand des Integrals $\int_0^{\frac{\pi}{2}} x^2(x^2 - 2) \sin(x) dx$!

Lösung:

Die Stammfunktion von $\int x^2(x^2 - 2) \sin(x) dx$ ist gegeben durch

$$-(28 - 14x^2 + x^4) \cos(x) + 4x(-7 + x^2) \sin(x).$$

Das bestimmte Integral ergibt sich dann als

$$\int_0^{\frac{\pi}{2}} x^2(x^2 - 2) \sin(x) dx = 28 - 14\pi + \frac{\pi^3}{2} \approx -0,479159.$$

Die Trapezregel ergibt die folgenden Näherungen: für $n = 3$ $-0,020944909$, für $n = 5$ $-0,361461274608$ und für $n = 10$ $-0,455783021609$.

Die Simpson-Regel ergibt die folgenden Näherungen: bei der Fassregel erhält man $-0,329850875$, für $k = 2$ die Näherung $-0,47496673$ und für $k = 5$ die Näherung $-0,479079334$.

Gauss-Legendre mit $n = 2$: $-0,583495559$; Gauss-Legendre mit $n = 3$: $-0,473832940$, mit $n = 10$ (die Gewichte und Stützstellen finden Sie im Buch) erhält man $-0,479158453$.

Bei der Implementierung in *Java* wird zuerst eine abstrakte Oberklasse implementiert, das für besetzte Gewichte und der fest implementierten Funktion der Aufgabe in der `berechne()`-Funktion die Quadraturformel bildet. Die Gauß-Legendre-Quadratur überschreibt diese Funktion.

```
//-----
// quadraturFormel.java
// Virtuelle Oberklasse für die Algorithmen für
// Quadraturformeln.
//-----
//Autor:          Manfred Brill
//Letzte Änderung: 14. 11. 2004
//-----
public abstract class quadraturFormel {

    protected double a, b; // Integrationsgrenzen
    protected int n;      // Anzahl der Stützstellen
    protected double h;   // Schrittweite
    protected double y[]; // Funktionswerte
    protected double w[]; // Gewichte

    protected quadraturFormel() {
        a = 0.0;
        b = 1.0;
        n = 3;
        h = (b-a)/(double)(n-1);
        y = new double[n];
        w = new double[n];
    }

    protected quadraturFormel(double aa, double bb, int nn) {
        a = aa; b = bb;
        n = nn;
        h = (b-a)/(double)(n-1);
        y = new double[n];
        w = new double[n];
    }
}
```

```
}

public void setGrenzen(double aa, double bb) {
    a = aa; b = bb;
}

public void setA(double aa) {
    a = aa;
}

public void setB(double bb) {
    b = bb;
}

public void setN(int j) {
    n = j;
}

public void setSchrittweite(double delta) {
    h = delta;
}

public int getAnzahlStuetzstellen() {
    return n;
}

public double getSchrittweite() {
    return h;
}

public void setY(double yy[]) {
    for (int i=0; i<n; i++) y[i] = yy[i];
}

public void setGewichte(double ww[]) {
    for (int i=0; i<n; i++) w[i] = ww[i];
}

protected double funktion(double x) {
    // Funktion für Aufgabe 14
    return x*x*(x*x-2.0)*Math.sin(x);
}

public double berechne() {
    double summe = 0.0;

    for (int i=0; i<n; i++) {
        summe += w[i]*y[i];
    }
    return summe;
}
}
```

```
// Die Kepler'sche Fassregel
public class fassRegel extends quadraturFormel {

    public fassRegel() {
        double ww;
```

```

    a = 0.0;
    b = 1.0;
    n = 3;
    h = (b-a)/(double)(n-1);

    y[0] = funktion(a);
    y[1] = funktion((b-a)/2.0);
    y[2] = funktion(b);

    ww = (b-a)/6.0;
    w[0] = ww;
    w[1] = 4.0*ww;
    w[2] = ww;
}

public fassRegel(double aa, double bb, double yy[]) {
    double ww;
    a = aa; b = bb; n = 3;
    h = (b-a)/(double)(n-1);

    for (int i=0; i<n; i++) y[i] = yy[i];

    ww = (b-a)/6.0;
    w[0] = ww;
    w[1] = 4.0*ww;
    w[2] = ww;
}

///! Konstruktor mit eingebauter Funktion der Aufgabe
public fassRegel(double aa, double bb) {
    double ww;
    a = aa; b = bb; n = 3;
    h = (b-a)/(double)(n-1);

    y[0] = funktion(a);
    y[1] = funktion((b-a)/2.0);
    y[2] = funktion(b);

    ww = (b-a)/6.0;
    w[0] = ww;
    w[1] = 4.0*ww;
    w[2] = ww;
}
}

```

```

// Die Trapezregel
public class trapezRegel extends quadraturFormel {

    public trapezRegel() {

        a = 0.0;
        b = 1.0;
        n = 3;
        h = (b-a)/(double)(n-1);

        y = new double[3];
        y[0] = funktion(a);
        y[1] = funktion((b-a)/2.0);
        y[2] = funktion(b);
    }
}

```

```

        w = new double[3];
        w[0] = h/2.0;
        w[1] = h;
        w[2] = h/2.0;
    }

    public trapezRegel(double aa, double bb, int nn, double yy[]) {
        a = aa; b = bb; n = nn;
        h = (b-a)/(double)(n-1);
        y = new double[n];
        w = new double[n];

        for (int i=0; i<n; i++) y[i] = yy[i];

        w[0] = h/2.0;
        for (int i=1; i<n-1; i++) w[i] = h;
        w[n-1] = h/2.0;
    }

    /// Konstruktor mit eingebauter Funktion der Aufgabe
    public trapezRegel(double aa, double bb, int nn) {
        int i;

        a = aa; b = bb; n = nn;
        h = (b-a)/(double)(n-1);
        y = new double[n];
        w = new double[n];

        y[0] = funktion(a);
        for (i=1; i<n-1; i++) y[i] = funktion(a + (double)i *h);
        y[n-1] = funktion(b);

        w[0] = h/2.0;
        for (i=1; i<n-1; i++) w[i] = h;
        w[n-1] = h/2.0;
    }
}

```

```

/// Die Simpson-Regel
public class simpsonRegel extends quadraturFormel {
    private int k;

    public simpsonRegel() {

        a = 0.0;
        b = 1.0;
        n = 3;
        k = 1;
        h = (b-a)/(double)(n-1);

        y = new double[2];
        y[0] = funktion(a);
        y[1] = funktion((b-a)/2.0);
        y[2] = funktion(b);

        w = new double[2];
        w[0] = h/3.0;
        w[1] = 4.0*h;
    }
}

```

```

        w[2] = h/3.0;
    }

    public simpsonRegel(double aa, double bb, int nn, double yy[]) {
        int i;

        a = aa; b = bb; n = nn; k = (n-1)/2;
        h = (n-1)/2;

        h = (b-a)/(double)(n-1);
        y = new double[n];
        w = new double[n];

        for (i=0; i<n; i++) y[i] = yy[i];

        w[0] = h/3.0;
        for (i=1; i<k; i++) w[2*i] = (2.0*h)/3.0;
        for (i=1; i<=k; i++) w[2*i-1] = (4.0*h)/3.0;
        w[n-1] = h/3.0;
    }

    ///! Konstruktor mit eingebauter Funktion der Aufgabe
    public simpsonRegel(double aa, double bb, int nn) {
        int i;

        a = aa; b = bb; n = nn; k = (n-1)/2;
        h = (b-a)/(double)(n-1);
        y = new double[n];
        w = new double[n];

        y[0] = funktion(a);
        for (i=1; i<n-1; i++) y[i] = funktion(a + (double)i *h);
        y[n-1] = funktion(b);

        w[0] = h/3.0;
        for (i=1; i<k; i++) w[2*i] = (2.0*h)/3.0;
        for (i=1; i<=k; i++) w[2*i-1] = (4.0*h)/3.0;
        w[n-1] = h/3.0;
    }
}

/// Gauss-Legendre-Quadratur für n=2, 3 und 10
public class gaussLegendre extends quadraturFormel {

    public gaussLegendre() {

        a = -1.0;
        b = 1.0;
        n = 2;
        h = (b-a)/(double)(n-1);

        w = new double[2];
    }

    public gaussLegendre(double aa, double bb, int nn) {
        a = aa; b = bb; n = nn;

        h = (b-a)/(double)(n-1);
        w = new double[n]; y = new double[n];
    }
}

```

```
}  
  
// Berechne-Funktion wird hier überschrieben!  
public double berechne() {  
    double ergebnis;  
  
    // Nur die Fälle n=2, 3 und 10 sind implementiert!  
    switch (n) {  
        case 2:    ergebnis = legendre2();  
                  break;  
        case 3:    ergebnis = legendre3();  
                  break;  
        case 10:   ergebnis = legendre10();  
                  break;  
        default:   n=2;  
                  ergebnis = legendre2();  
                  break;  
    }  
    return ergebnis;  
}  
  
// Koordinatentransformation vom Intervall [a,b] auf [-1,1]  
private double phi(double x) {  
    return 0.5*(a+b) + 0.5*(b-a)*x;  
}  
  
private double legendre2() {  
    double x = Math.sqrt(3.0)/3.0;  
  
    return 0.5*(b-a)*(funktion(phi(-x)) + funktion(phi(x)));  
}  
  
private double legendre3() {  
    double y = Math.sqrt(15.0)/5.0;  
    double w1 = 5.0/9.0;  
    double w2 = 8.0/9.0;  
  
    return 0.5*(b-a)*(w1*funktion(phi(-y)) + w2*funktion(phi(0.0)) +  
        w1*funktion(phi(y)));  
}  
  
private double legendre10() {  
    double x[] = new double[10];  
    double yy, summe;  
    int i;  
  
    x[0] = -0.9739075285;  
    x[1] = -0.8650633666;  
    x[2] = -0.67940956582;  
    x[3] = -0.4333953941;  
    x[4] = -0.1488743389;  
    x[5] = 0.1488743389;  
    x[6] = 0.4333953941;  
    x[7] = 0.67940956582;  
    x[8] = 0.8650633666;  
    x[9] = 0.9739075285;  
  
    w[0] = 0.0666713443;
```

```
w[1] = 0.1494513491;
w[2] = 0.2190863625;
w[3] = 0.2692667193;
w[4] = 0.2955242247;
w[5] = 0.2955242247;
w[6] = 0.2692667193;
w[7] = 0.2190863625;
w[8] = 0.1494513491;
w[9] = 0.0666713443;

summe = 0.0;
for (i=0; i<10; i++) {
    summe += w[i]*funktion(phi(x[i]));
}
return 0.5*(b-a)*summe;
}
}
```

Und hier das verwendete Hauptprogramm für die Berechnungen:

```
/*
 * Quadraturen für die Lösung der Aufgabe 14, Kapitel 14
 */

public class int14 {

    // -----
    public static void main (String[] args) {
        System.out.println("-----");

        kepler();

        System.out.println("-----");

        trapez();

        System.out.println("-----");

        simpson();

        System.out.println("-----");

        gauss();
    }

    public static void kepler() {
        double solution=0.0;
        fassRegel quad = new fassRegel(0.0, Math.PI/2.0);

        solution = quad.berechne();
        System.out.println(
            "Die berechnete Lösung der Fass-Regel: "+solution);
    }

    public static void trapez() {
        double solution=0.0;
        int n=10;
        trapezRegel quad = new trapezRegel(0.0, Math.PI/2.0, n);
    }
}
```

```

    solution = quad.berechne();
    System.out.println(
        "Die berechnete Lösung der Trapez-Regel für n = "+n+": "+solution);
    }

    public static void simpson() {
        double solution=0.0;
        int n=11;
        simpsonRegel quad = new simpsonRegel(0.0, Math.PI/2.0, n);

        solution = quad.berechne();
        System.out.println(
            "Die berechnete Lösung der Simpson-Regel für n = "+n+": "+solution);
    }

    public static void gauss() {
        double solution=0.0;
        int n=10;
        gaussLegendre quad = new gaussLegendre(0.0, Math.PI/2.0, n);

        solution = quad.berechne();
        System.out.println(
            "Die berechnete Lösung der Gauss-Legendre-Quadratur für n = "+n+": "
                +solution);
    }
}

```

15. Verwenden Sie Ihre Implementierungen und Ergebnisse aus Aufgabe 14, um die Fehler der Newton-Côtes-Formeln in einer doppelt-logarithmischen Darstellung aufzutragen! Welche Form muss die Kurve besitzen?

Lösung:

Die Ordnung der Quadraturformel ist die Steigung der Geraden, falls eine doppelt-logarithmische Skala verwendet wird. Es ist $e(h) = Ch^p(b-a)f^{(p)}(\xi)$. Durch Logarithmieren auf beiden Seiten erhält man eine Gerade mit Steigung p . Abbildung 14.4 zeigt das grafische Ergebnis, Tabelle 14.2 die numerischen Resultate.

Tabelle 14.2: Die numerischen Ergebnisse für Trapez- und Simpson-Regel für Aufgabe 15

n	h	Trapez-Regel	Simpson-Regel
2	1,0	$1,409 \cdot 10^{-1}$	$4,789 \cdot 10^{-1}$
3	0,5	$3,565 \cdot 10^{-2}$	$5,793 \cdot 10^{-4}$
5	0,25	$8,940 \cdot 10^{-3}$	$3,701 \cdot 10^{-5}$
9	0,125	$2,237 \cdot 10^{-3}$	$2,326 \cdot 10^{-6}$
17	0,062 5	$5,593 \cdot 10^{-4}$	$1,456 \cdot 10^{-7}$
33	0,031 25	$1,398 \cdot 10^{-4}$	$9,103 \cdot 10^{-9}$
65	0,0156 25	$3,496 \cdot 10^{-5}$	$5,69 \cdot 10^{-10}$

16. Bestimmen Sie für das Anfangswertproblem $y' = y + 2$, $y(0) = 1$ eine Näherung für $y(2)$ durch die Verfahren von Euler, Heun und Runge-Kutta mit Schrittweite $h = 1!$

Lösung:

Die exakte Lösung des Anfangswertproblems ist gegeben durch $y(x) = -2 + Ce^x$; mit dem Anfangswert ist dann $y(x) = -2 + 3e^x$.

Die exakte Lösung ist $y(2) \approx 20,167$.

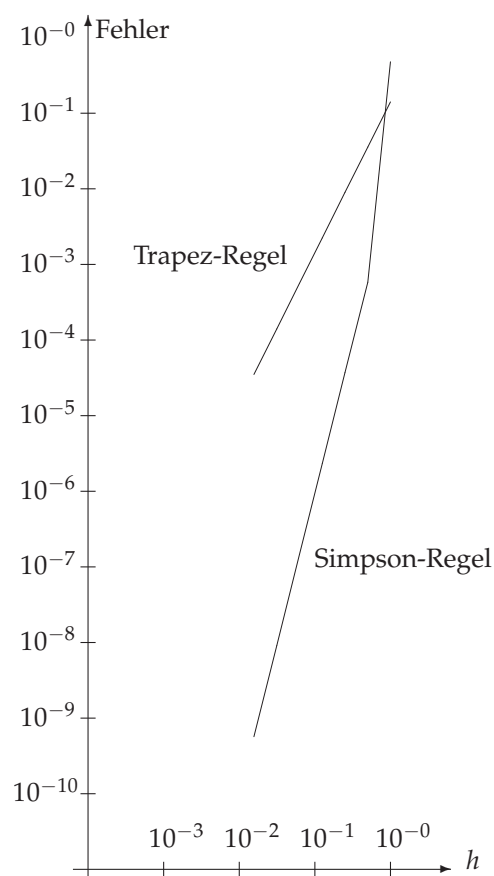


Abbildung 14.4: Die Fehler bei Trapez- und Simpson-Regel in einer doppelt-logarithmischen Skala für Aufgabe 15

Das Euler-Verfahren mit $h = 1$ berechnet die Näherung $y_2 = 10,0$; Heun berechnet $y_2 = 16,75$; Runge-Kutta berechnet $y_2 = 20,005$.

17. Implementieren Sie das Euler-, das Heun- und das Runge-Kutta-Verfahren in der Programmiersprache Ihrer Wahl und testen Sie Ihre Implementierung an Hand des Anfangswertproblems $y' = y + 2$, $y(0) = -1$ und $y' = e^{2x} - 2y$, $y(0) = 1$ auf dem Intervall $[0; 2]$. Skizzieren Sie Ihre Ergebnisse, und erstellen Sie eine doppelt-logarithmische Darstellung der globalen Fehler!

Lösung:

Die exakte Lösung des Anfangswertproblems

$$y' = y + 2, y(0) = -1$$

ist $y(x) = e^x - 2$.

Die exakte Lösung des Anfangswertproblems

$$y' = e^{2x} - 2y, y(0) = 1$$

ist gegeben durch

$$y(x) = \frac{1}{4}e^{2x} + \frac{3}{4}e^{-2x}$$

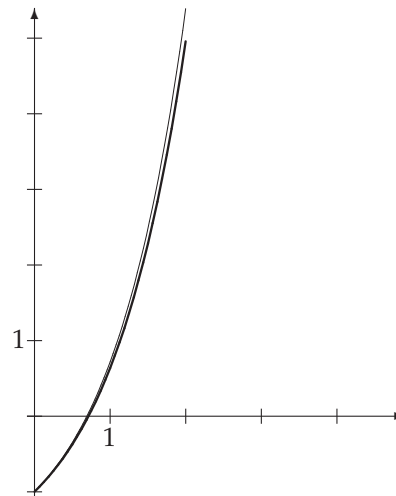


Abbildung 14.5: Ergebnis des Euler-Verfahrens ($n = 17$) für Aufgabe 17, $y' = y + 2$. Die berechnete Näherung ist fett dargestellt.

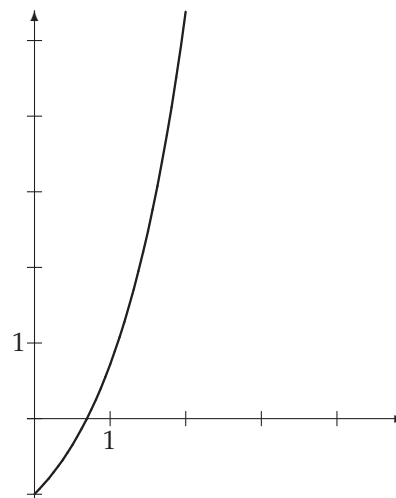


Abbildung 14.6: Ergebnis des Heun-Verfahrens ($n = 17$) für Aufgabe 17, $y' = y + 2$. Die berechnete Näherung ist fett dargestellt.

mit $y(2) \approx 13,663$.

In den Abbildungen 14.5, 14.6 und 14.7 sehen Sie die Ergebnisse für die rechte Seite $f(x, y) = y + 2$; in den Abbildungen 14.8, 14.9 und 14.10 die für die rechte Seite $f(x, y) = e^{2x} - 2y$. Die Lösungen wurden mit dem folgenden Java-Code berechnet:

```
//-----
// einzelschrittVerfahren.java
// Virtuelle Oberklasse für die Algorithmen für
// Einzelschrittverfahren.
//-----
//Autor:           Manfred Brill
//Letzte Änderung: 14. 11. 2004
```

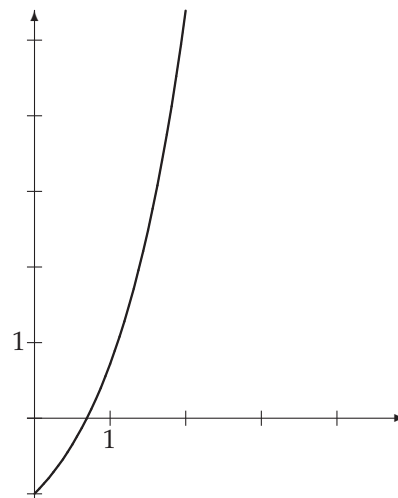


Abbildung 14.7: Ergebnis des Runge-Kutta-Verfahrens ($n = 17$) für Aufgabe 17, $y' = y + 2$. Die berechnete Näherung ist fett dargestellt.

```
//-----
public abstract class einzelschrittVerfahren {

    protected double a, b;    // Integrationsgrenzen
    protected int n;         // Anzahl der Stützstellen
    protected double h;      // Schrittweite
    protected double y0;     // Anfangswert für x=a
    protected double x[], y[]; // Ergebnisvektoren des Verfahrens

    protected einzelschrittVerfahren() {
        a = 0.0;
        b = 1.0;
        n = 3;
        h = (b-a)/(double) (n-1);
        y0 = 0.0;
        x = new double[n];
        y = new double[n];
    }

    protected einzelschrittVerfahren(double aa, double bb,
                                     double yy, int nn) {

        a = aa; b = bb;
        n = nn;
        h = (b-a)/(double) (n-1);
        y0 = yy;

        x = new double[n];
        y = new double[n];
    }

    public void getErgebnis(double xw[], double yw[]) {
        for (int i=0; i<n; i++) {
            xw[i] = x[i];
            yw[i] = y[i];
        }
    }
}
```

```
}  
  
public void printErgebnis() {  
    for (int i=0; i<n; i++)  
        System.out.println(""+x[i]+" "+y[i]+"");  
}  
  
public void setGrenzen(double aa, double bb, double yy) {  
    a = aa; b = bb;  
    y0 = yy;  
}  
  
public void setA(double aa) {  
    a = aa;  
}  
  
public void setB(double bb) {  
    b = bb;  
}  
  
public void setAnfangsWert(double yy) {  
    y0 = yy;  
}  
  
public void setN(int j) {  
    n = j;  
}  
  
public void setSchrittweite(double delta) {  
    h = delta;  
}  
  
public double getSchrittweite() {  
    return h;  
}  
  
// Rechte Seite der gewöhnlichen Differenzialgleichung  
protected double funktionBuch(double x, double y) {  
    // Buchfunktion, Seite 388  
    return x+y;  
}  
  
// Rechte Seite der gewöhnlichen Differenzialgleichung  
protected double funktion(double x, double y) {  
    // Funktion für Aufgabe 17  
    return y+2.0;  
}  
  
protected double funktion2(double x, double y) {  
    // Funktion für Aufgabe 17  
    return Math.exp(2.0*x) - 2.0*y;  
}  
  
// Die implementieren Einzelschrittverfahren  
// müssen diese Funktion implementieren!  
protected abstract void berechne();  
}
```

```

\end{lstlisting}

\begin{lstlisting}{}
// Das Eulerverfahren
public class eulerVerfahren extends einzelschrittVerfahren {

    public eulerVerfahren (double aa, double bb, double yy, int nn) {
        a = aa; b = bb;
        n = nn;
        h = (b-a)/(double) (n-1);
        y0 = yy;

        x = new double[n];
        y = new double[n];
    }

    public void berechne() {
        int i;

        x[0] = a;
        y[0] = y0;

        for (i=1; i<n; i++) {
            x[i] = x[i-1]+h;
            y[i] = y[i-1] + h*funktion(x[i-1], y[i-1]);
        }
    }
}

```

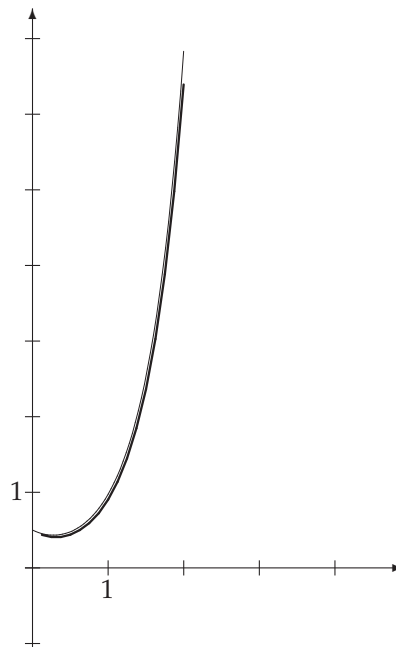


Abbildung 14.8: Ergebnis des Euler-Verfahrens ($n = 17$) für Aufgabe 17, $y' = e^{2x} - 2y$. Die exakte Lösung ist fett dargestellt

In Abbildung 14.11 finden Sie die globalen Fehler für diese Verfahren und der rechten Seite

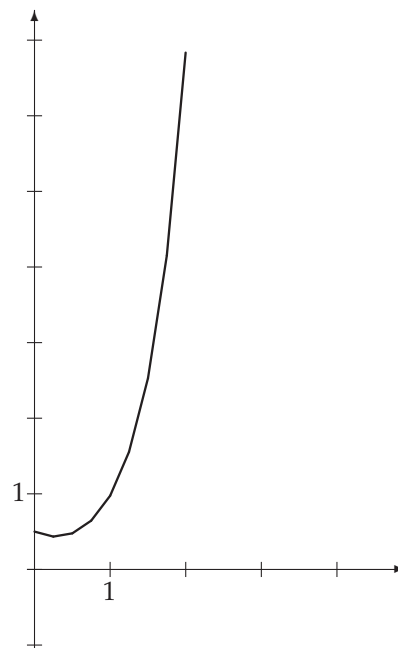


Abbildung 14.9: Ergebnis des Runge-Kutta-Verfahrens ($n = 9$) für Aufgabe 17, $y' = e^{2x} - 2y$.

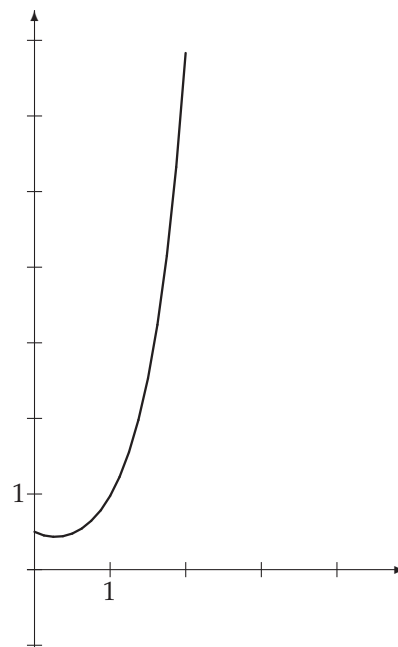


Abbildung 14.10: Ergebnis des Runge-Kutta-Verfahrens ($n = 17$) für Aufgabe 17, $y' = e^{2x} - 2y$.

$f(x, y) = y + 2$ in doppelt-logarithmischer Darstellung.

```
// Das Heun-Verfahren
public class heunVerfahren extends einzelschrittVerfahren {
```

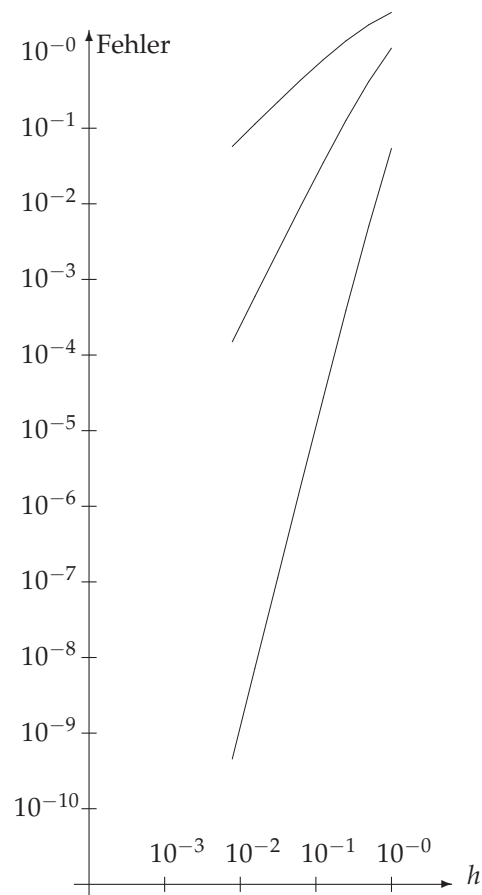


Abbildung 14.11: Die globalen Fehler bei Euler-, Heun- und Runge-Kutta-Verfahren für $f(x, y) = y + 2$ in einer doppelt-logarithmischen Skala für Aufgabe 17

```

public heunVerfahren (double aa, double bb, double yy, int nn) {
    a = aa; b = bb;
    n = nn;
    h = (b-a)/(double) (n-1);
    y0 = yy;

    x = new double[n];
    y = new double[n];
}

public void berechne() {
    int i;
    double k1, k2;

    x[0] = a;
    y[0] = y0;

    for (i=1; i<n; i++) {
        k1 = funktion(x[i-1], y[i-1]);
        k2 = funktion(x[i-1]+h, y[i-1]+h*k1);
    }
}

```

```

        x[i] = x[i-1]+h;
        y[i] = y[i-1] + (h/2.0)*(k1+k2);
    }
}

```

```

// Das Runge-Kutta-Verfahren
public class rungekuttaVerfahren extends einzelschrittVerfahren {
    public rungekuttaVerfahren (double aa, double bb, double yy, int nn) {
        a = aa; b = bb;
        n = nn;
        h = (b-a)/(double)(n-1);
        y0 = yy;

        x = new double[n];
        y = new double[n];
    }

    public void berechne() {
        int i;
        double k1, k2, k3, k4, hh=h/2.0;

        x[0] = a;
        y[0] = y0;

        for (i=1; i<n; i++) {
            k1 = funktion(x[i-1], y[i-1]);
            k2 = funktion(x[i-1]+hh, y[i-1]+hh*k1);
            k3 = funktion(x[i-1]+hh, y[i-1]+hh*k2);
            k4 = funktion(x[i-1]+h, y[i-1]+h*k3);

            x[i] = x[i-1]+h;
            y[i] = y[i-1] + h*(k1/6.0+k2/3.0 + k3/3.0 + k4/6.0);
        }
    }
}

```

18. Skizzieren Sie die Vektorfelder $\mathbf{f}(x, y) = (y, x)$, $\mathbf{f}(x, y) = (-x, 2y)$, $\mathbf{f}(x, y) = \frac{1}{\sqrt{x^2+y^2}}(y, -x)$ und $\mathbf{f}(x, y) = (y^2 - 2xy, 3xy - 6x^2)!$

Lösung:

Die Skizzen finden Sie in den Abbildungen 14.12 bis 14.15.

19. Erweitern Sie Ihre Implementierung der Einzelschrittverfahren, sodass Sie damit Differenzialgleichungssysteme lösen können. Nähern Sie damit das System mit dem Vektorfeld $\mathbf{f}(x, y) = (-y, x)$ und $(y_1(0) = 1, y_2(0) = 0)$ an. Vergleichen Sie Ihre Ergebnisse mit den Abbildungen 14.16 und 14.20!

Lösung:

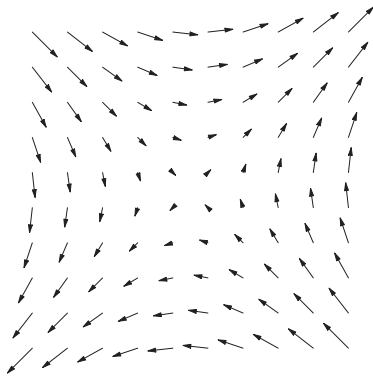
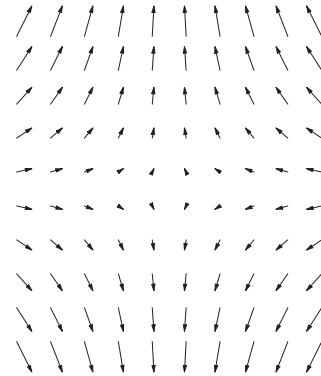
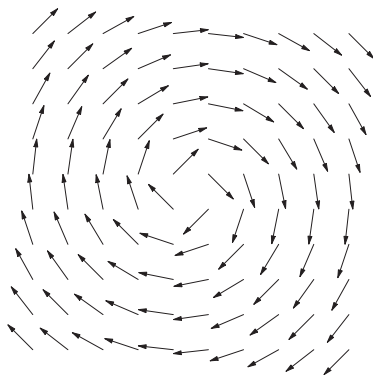
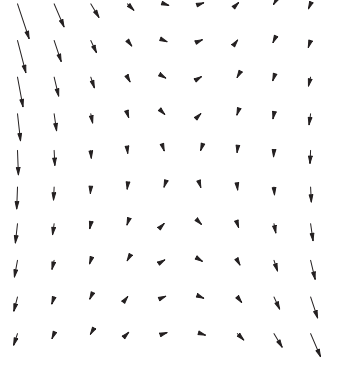
Sie sollten einen geschlossenen Kreis beim Euler-Verfahren nur bei kleinen Schrittweiten erhalten.

Hier die Implementierung in Java:

```

//-----
// einzelschrittVerfahren2D.java
// Virtuelle Oberklasse für die Algorithmen für
// Einzelschrittverfahren zur Lösung eines

```

Abbildung 14.12: $f(x, y) = (y, x)$ Abbildung 14.13: $f(x, y) = (-x, 2y)$ Abbildung 14.14: $f(x, y) = \frac{1}{\sqrt{x^2+y^2}}(y, -x)$ Abbildung 14.15: $f(x, y) = (y^2 - 2xy, 3xy - 6x^2)$

```
// zweidimensionalen Vektorfelds.
//-----
//Autor:          Manfred Brill
//Letzte Änderung: 14. 11. 2004
//-----
public abstract class einzelschrittVerfahren2D {

    protected int n;          // Anzahl der Schritte
    protected double h;       // Schrittweite
    protected double x0, y0;  // Anfangswert
    protected double x[], y[]; // Ergebnisvektoren des Verfahrens

    protected einzelschrittVerfahren2D() {
        n = 3;
        h = 0.1;
        x0 = 0.0; y0 = 0.0;
        x = new double[n];
        y = new double[n];
    }

    protected einzelschrittVerfahren2D(double xx, double yy,
                                         int nn, double delta) {
        n = nn;
        h = delta;
        x0 = xx;
    }
}
```

```
        y0 = yy;

        x = new double[n];
        y = new double[n];
    }

    public void getErgebnis(double xw[], double yw[]) {

        for (int i=0; i<n; i++) {
            xw[i] = x[i];
            yw[i] = y[i];
        }
    }

    public void printErgebnis() {

        for (int i=0; i<n; i++)
            System.out.println("(" + x[i] + ", " + y[i] + ")");
    }

    public void setAnfangswert(double xx, double yy) {
        x0 = xx; y0 = yy;
    }

    public void setN(int j) {
        n = j;
    }

    public void setSchrittweite(double delta) {
        h = delta;
    }

    public double getSchrittweite() {
        return h;
    }

    // Rechte Seite der gewöhnlichen Differenzialgleichung,
    // x-Komponente
    protected double funktionx(double x, double y) {
        // Funktion 1 für Aufgabe 17
        return -y;
    }

    // Rechte Seite der gewöhnlichen Differenzialgleichung,
    // y-Komponente
    protected double funktiony(double x, double y) {
        // Funktion 2 für Aufgabe 17
        return x;
    }

    // Die implementieren Einzelschrittverfahren
    // müssen diese Funktion implementieren!
    protected abstract void berechne();
    protected abstract double xSchritt(double x, double y);
    protected abstract double ySchritt(double x, double y);
}

// Das Eulerverfahren
```

```
public class eulerVerfahren2D extends einzelschrittVerfahren2D {
    public eulerVerfahren2D (double xx, double yy,
                             int nn, double delta) {
        n = nn;
        h = delta;
        x0 = xx;
        y0 = yy;

        x = new double[n];
        y = new double[n];
    }

    public double xSchritt(double x, double y) {
        return x + h*funktionx(x, y);
    }

    public double ySchritt(double x, double y) {
        return y + h*funktiony(x, y);
    }

    public void berechne() {
        int i;
        double savex;

        x[0] = x0;
        y[0] = y0;

        for (i=1; i<n; i++) {
            savex = x[i-1];
            x[i] = xSchritt(x[i-1], y[i-1]);
            y[i] = ySchritt(savex, y[i-1]);
        }
    }
}

// Das Runge-Kutta-Verfahren
public class rungekuttaVerfahren2D extends einzelschrittVerfahren2D {
    public rungekuttaVerfahren2D (double xx, double yy,
                                   int nn, double delta) {
        n = nn;
        h = delta;
        x0 = xx; y0 = yy;

        x = new double[n];
        y = new double[n];
    }

    public double xSchritt(double x, double y) {
        double k1, k2, k3, k4, hh;

        hh = h/2.0;

        k1 = funktionx(x, y);
        k2 = funktionx(x+hh, y+hh*k1);
        k3 = funktionx(x+hh, y+hh*k2);
        k4 = funktionx(x+h, y+h*k3);

        return x + h*(k1/6.0 + k2/3.0 + k3/3.0 + k4/6.0);
    }
}
```

```

}

public double ySchritt(double x, double y) {
    double k1, k2, k3, k4, hh;

    hh = h/2.0;

    k1 = funktiony(x, y);
    k2 = funktiony(x+hh, y+hh*k1);
    k3 = funktiony(x+hh, y+hh*k2);
    k4 = funktiony(x+h, y+h*k3);

    return y + h*(k1/6.0 + k2/3.0 + k3/3.0 + k4/6.0);
}
}

```

20. Das *Räuber-Beute-Modell* nach Lotka-Volterra beschreibt die Größe der Beute-Population y_1 und der Räuberpopulation y_2 über die Zeit. Das Modell ist gegeben durch das Differenzialgleichungssystem

$$y_1'(t) = 10y_1(t)(1 - y_2(t)), y_2'(t) = y_2(t)(y_1(t) - 1).$$

Verwenden Sie Ihre Implementierung der Einzelschrittverfahren für die Anfangswerte $y_1(0) = 3$, $y_2(0) = 1$ und vergleichen Sie Ihre Ergebnisse!

Lösung:

In Abbildung 14.16 sehen Sie die mit Runge-Kutta berechneten Paare (y_1, y_2) . Die Abbildung 14.17 zeigt die y_1 - und y_2 -Werte über der Zeitachse aufgetragen. Für die Lösung wurde der Code aus Aufgabe 19 verwendet.

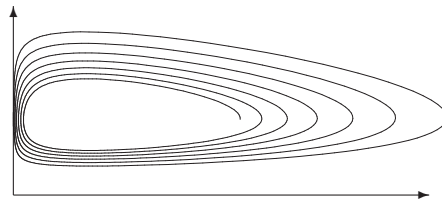


Abbildung 14.16: Das Räuber-Beute-Modell mit Runge-Kutta, $h = 0,01$ und $n = 1600$

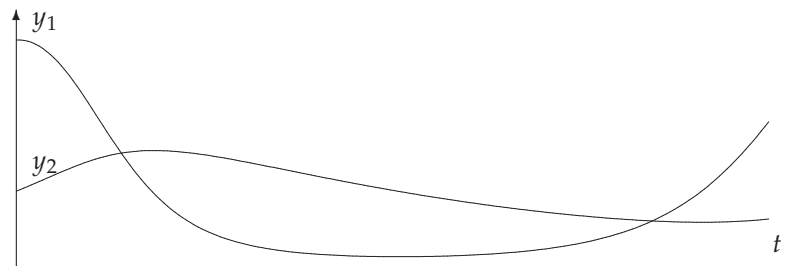


Abbildung 14.17: Die berechneten y_1 und y_2 -Werte über der Zeitachse

Kapitel 15

Wahrscheinlichkeitstheorie und Statistik

Verständnisfragen

Sachfragen

1. Erläutern Sie den Begriff der absoluten und relativen Häufigkeit einer Stichprobe!
2. Erläutern Sie den Begriff der Klassenhäufigkeit in der beschreibenden Statistik!
3. Erläutern Sie den Begriff der Summenhäufigkeit!
4. Erläutern Sie den Begriff der empirischen Dichte!
5. Erläutern Sie den Begriff des Modalwerts!
6. Erläutern Sie den Begriff des arithmetischen Mittels!
7. Erläutern Sie den Begriff des geometrischen Mittels!
8. Erläutern Sie den Begriff der Spannweite, der Varianz und der Standardabweichung!
9. Was versteht man unter einer Kontingenztafel?
10. Was ist die Stichproben-Kovarianz?
11. Wie ist der Bravais-Pearson-Korrelationskoeffizient definiert?
12. Was ist eine Ereignis-Algebra?
13. Erläutern Sie den Ansatz der Laplace-Wahrscheinlichkeit!
14. Was ist eine geometrische Wahrscheinlichkeit?
15. Was ist ein Wahrscheinlichkeitsmaß?
16. Was versteht man unter einem Wahrscheinlichkeitsraum?
17. Was ist eine bedingte Wahrscheinlichkeit?
18. Was sagt der Satz von der totalen Wahrscheinlichkeit aus?

19. Wie lautet der Satz von Bayes?
20. Wann sind zwei Ereignisse unabhängig?
21. Was ist eine Zufallsvariable?
22. Was ist eine Verteilungsfunktion einer Zufallsvariablen?
23. Welche Eigenschaften hat eine Verteilungsfunktion?
24. Was ist eine diskrete Zufallsvariable?
25. Was ist eine stetige Zufallsvariable?
26. Erläutern Sie den Begriff der Wahrscheinlichkeitsdichte einer stetigen Zufallsvariablen!
27. Gibt es für eine stetige Zufallsvariable Ereignisse, die fast unmöglich sind?
28. Wie ist der Erwartungswert einer Zufallsvariablen definiert?
29. Wie ist die Varianz und die Standardabweichung einer Zufallsvariablen definiert?
30. Welche Rechenregeln für Erwartungswert und Varianz einer Zufallsvariablen kennen Sie?
31. Wann sind zwei Zufallsvariablen unabhängig?
32. Erläutern Sie den Korrelationskoeffizienten zweier Zufallsvariablen!
33. Wann sind zwei Zufallsvariable unkorreliert?
34. Erläutern Sie den Zusammenhang zwischen Unabhängigkeit und Unkorreliertheit!
35. Wie lautet die Ungleichung von Tschebyscheff?
36. Was ist ein Quantil einer Zufallsvariablen?
37. Nennen Sie wichtige diskrete und stetige Verteilungen!
38. Was ist eine Bernoulli-Variablen?
39. Erläutern Sie Verteilungs- und Dichtefunktion der Normalverteilung!
40. Erläutern Sie den Zusammenhang zwischen $N(0, 1)$ und einer beliebigen $N(\mu, \sigma)$ -Verteilung!
41. Was ist ein $k\sigma$ -Bereich?
42. Wie lautet das schwache Gesetz der großen Zahlen?
43. Was ist eine erwartungstreue Punktschätzung?
44. Was ist eine asymptotisch erwartungstreue Punktschätzung?
45. Nennen sie Beispiele für erwartungstreue und asymptotisch erwartungstreue Punktschätzer!
46. Erläutern Sie das Maximum-Likelihood-Prinzip!

Methodenfragen

1. Die empirische Dichte und die absoluten und relativen Häufigkeiten einer Stichprobe berechnen können.
2. Für eine Stichprobe Klassen bilden können.
3. Summenhäufigkeiten berechnen können.
4. Modalwert und Mittelwerte einer Stichprobe berechnen können.
5. Spannweite, Varianz und Standardabweichung einer Stichprobe berechnen können.
6. Ein Kontingenztable für eine zweidimensionale Stichprobe aufstellen können.
7. Stichproben-Kovarianz und Bravais-Pearson-Korrelationskoeffizient berechnen können.
8. Ein Streudiagramm erstellen und interpretieren können.
9. Für ein Zufallsexperiment eine geeignete Ereignis-Algebra konstruieren können.
10. Nachweisen können, dass eine gegebene Menge eine Ereignis-Algebra darstellt.
11. Laplace-Wahrscheinlichkeiten bestimmen können.
12. Die Rechenregeln für ein Wahrscheinlichkeitsmaß anwenden können.
13. Bedingte Wahrscheinlichkeiten berechnen und anwenden können.
14. Überprüfen können, ob zwei gegebene Ereignisse unabhängig sind.
15. Verteilungs- und Dichtefunktionen von Zufallsvariablen anwenden und bestimmen können.
16. Wahrscheinlichkeiten für Zufallsvariable berechnen können.
17. Erwartungswert, Varianz und Standardabweichung einer Zufallsvariablen berechnen können.
18. Rechenregeln für Erwartungswert und Varianz einer Zufallsvariablen anwenden können.
19. Zufallsvariable auf Unabhängigkeit überprüfen können.
20. Den Korrelationskoeffizienten bestimmen und interpretieren können.
21. Die Ungleichung von Tschebyscheff anwenden können.
22. Quantile bestimmen können.
23. Mit diskreten und stetigen Verteilungen arbeiten können.
24. Die typischen Anwendungsbereiche diskreter und stetiger Verteilungen kennen.
25. Mit der Normalverteilung arbeiten können.
26. $k\sigma$ -Bereiche der Normalverteilung bestimmen können.
27. Punktschätzer anwenden und einschätzen können.
28. Das Maximum-Likelihood-Prinzip anwenden können.

Übungsaufgaben

1. Die folgenden Daten stellen eine Stichprobe einer Umfrage über das Alter von Eigenheimen in Jahren dar:

82, 70, 69, 19, 71, 70, 13, 70, 70, 23, 62, 53, 32, 65, 66, 55, 79, 15, 21, 69

59, 18, 21, 66, 72, 19, 65, 57, 24, 71, 51, 50, 82, 13, 13, 82, 82, 54, 64, 36.

Stellen Sie die relativen und absoluten Häufigkeiten grafisch dar und bestimmen Sie die Lageparameter der Stichprobe!

Lösung:

Modalwerte für die Stichprobe sind die Ausprägungen 70 und 82. Das arithmetische Mittel der Stichprobe ist gegeben durch $\bar{x} = 51,825$, der Median durch $med = 60,5$. Das geometrische Mittel der Stichprobe ist gegeben durch $x_g = 44,52$.

Die absoluten Häufigkeiten finden sie in Abbildung 15.1, die relativen Häufigkeiten in Abbildung 15.2 und die empirische Verteilungsfunktion in Abbildung 15.3!

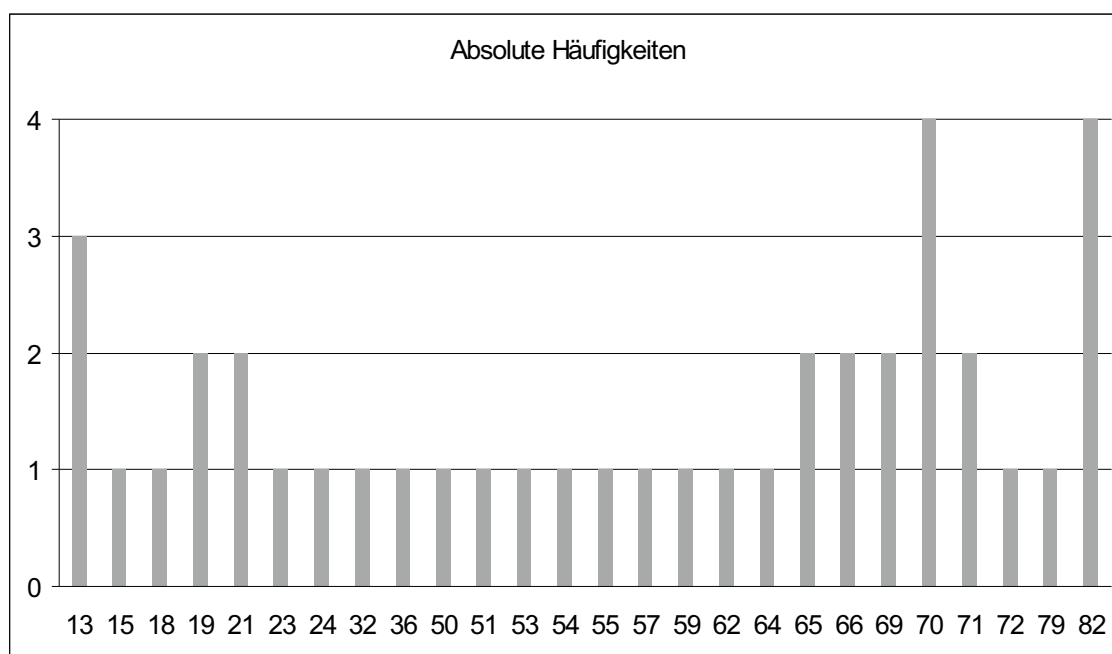


Abbildung 15.1: Die absoluten Häufigkeiten für die Daten in Aufgabe 1

2. Bestimmen Sie die relative Häufigkeit und die relative Summenhäufigkeit für eine Klassenbreite von $w = 50$ für die Lebensdauer-Stichprobe auf Seite 401, und stellen Sie diese grafisch dar!

Lösung:

Die relative Klassenhäufigkeiten bei Klassenbreite 50 finden Sie in Abbildung 15.4; die relative Summenhäufigkeit in Abbildung 15.5.

3. Bestimmen Sie die Lage- und Streuungsparameter für die Daten der der Lebensdauer-Stichprobe auf Seite 401 und der Stichprobe aus Aufgabe 1!

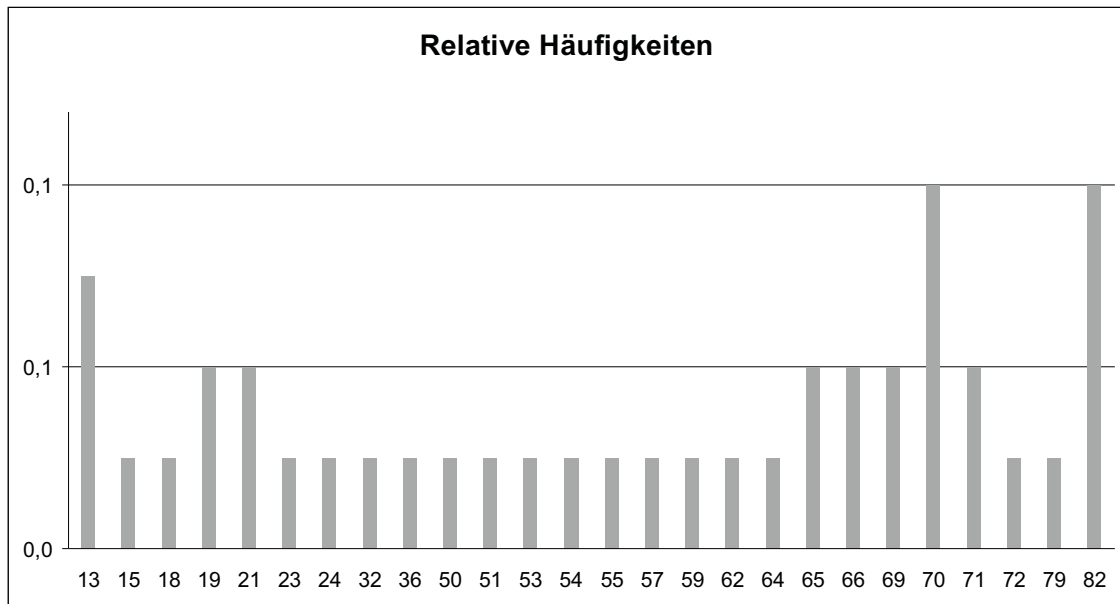


Abbildung 15.2: Die relativen Häufigkeiten für die Daten in Aufgabe 1

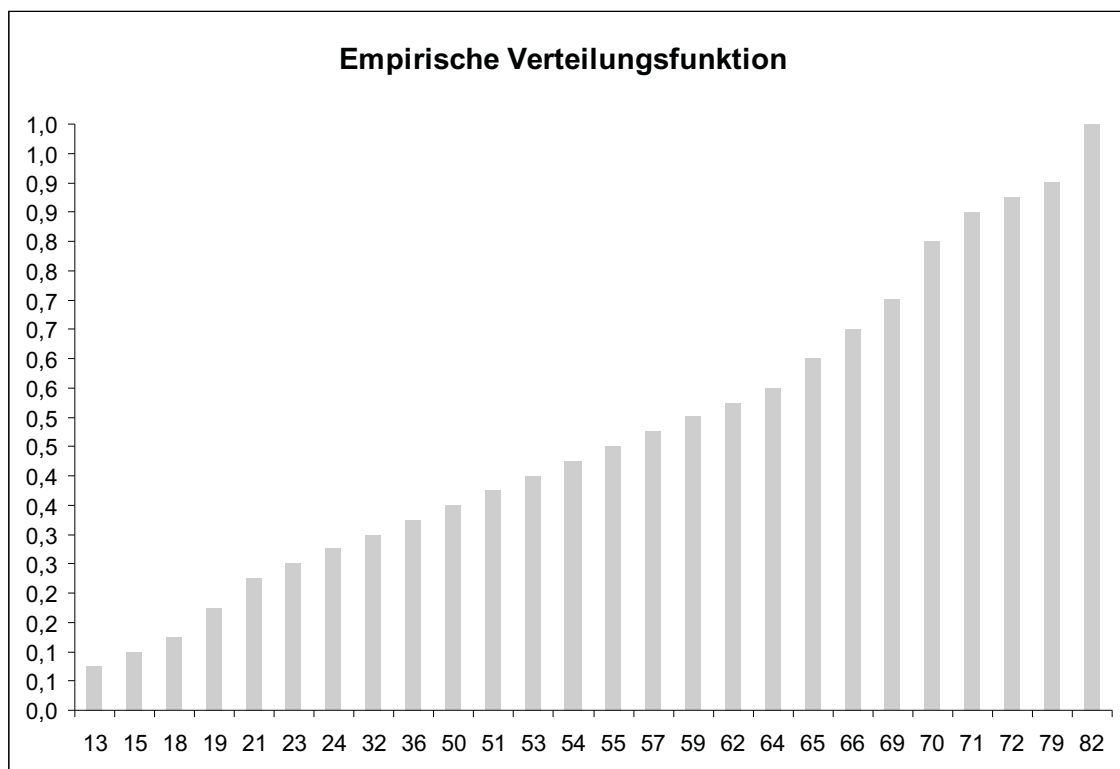


Abbildung 15.3: Die empirische Verteilungsfunktion für die Daten in Aufgabe 1

Lösung:

Die Ergebnisse für die Stichprobe der Lebensdauern auf Seite 401:

$\bar{x} \approx 313,166$, $x_g \approx 266,85$, $x_m = 325$. Die Spannweite ist 470, die Varianz $s^2 \approx 2\,1283,59$ und

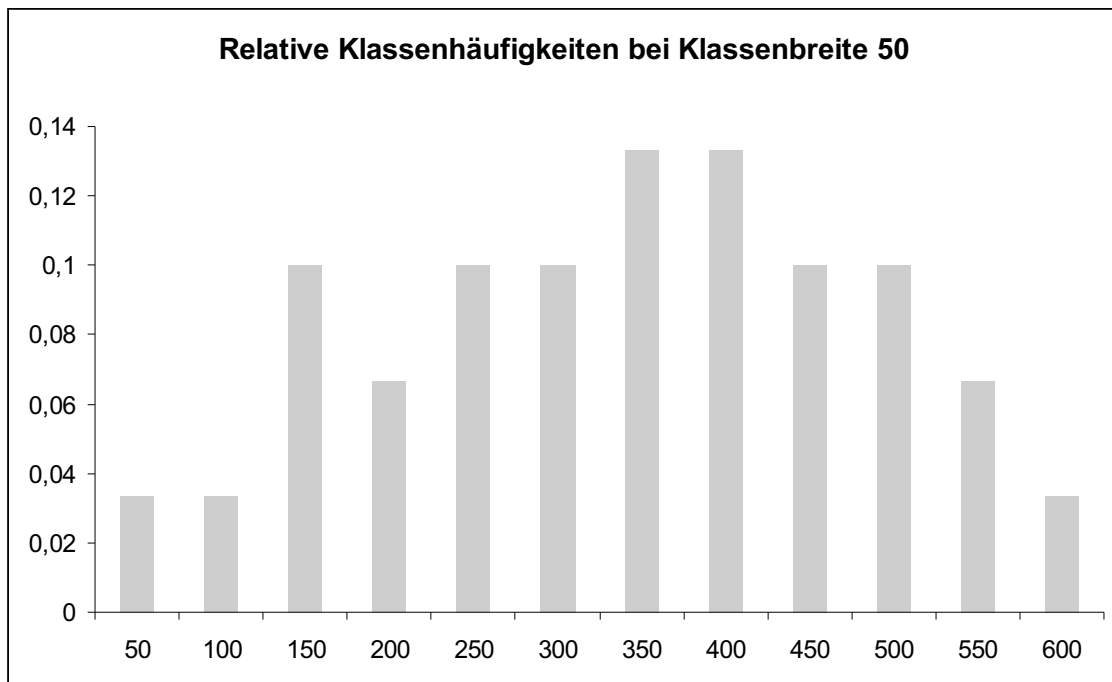


Abbildung 15.4: Die relativen Häufigkeiten für die Daten in Aufgabe 2

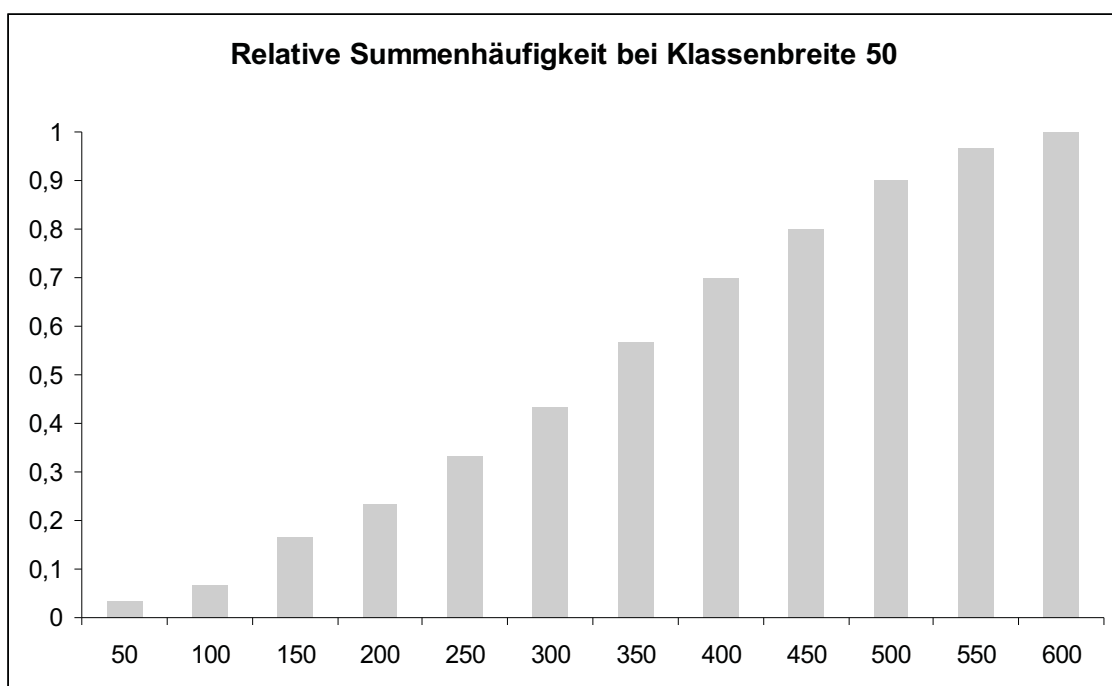


Abbildung 15.5: Die relativen Summenhäufigkeiten für die Daten in Aufgabe 2

die Standardabweichung $s \approx 145,89$.

Für die Stichprobe in Aufgabe 1 sind die Lageparameter: Modalwerte für die Stichprobe sind die Ausprägungen 70 und 82. Das arithmetische Mittel der Stichprobe ist gegeben durch $\bar{x} =$

51,825, der Median durch $x_{med} = 60,5$. Das geometrische Mittel der Stichprobe ist gegeben durch $x_g = 44,52$.

Die Spannweite der Stichprobe ist 6, die Varianz $s^2 \approx 560,507$ und die Standardabweichung ist $s \approx 23,675$.

4. Weisen Sie nach, dass der Median einer Stichprobe mit n Werten die Gleichungen

$$\sum_{i=1}^n \frac{x_i - x_{med}}{|x_i - x_{med}|} = 0, x_{med} = \frac{\sum_{i=1}^n \frac{x_i}{|x_i - x_{med}|}}{\sum_{i=1}^n \frac{1}{|x_i - x_{med}|}}$$

erfüllt!

Lösung:

Nehmen wir einmal an, dass es eine gerade Anzahl von Stichproben gibt. Dann war $x_{med} = \frac{1}{2}(x_k + x_{k+1})$ mit $k = \frac{n}{2}$. In der ersten Summe ist für alle Indices mit $i \leq k$ der Summand gleich -1 , für die restlichen Summanden ist der Wert konstant 1, und die Summe ist insgesamt Null.

Ist die Anzahl der Stichproben n ungerade, dann war der Median gegeben durch die Zahl $x_{med} = x_k$ mit $k = \frac{n+1}{2}$. Für diesen Index ist der Summand 0; die Summanden mit $i < k$ sind wieder alle -1 , die Summanden mit $i > k$ sind 1; und es gibt genauso so viele Summanden mit -1 wie mit 1.

Die zweite Gleichung für den Median folgt aus

$$x_{med} - \frac{\sum_{i=1}^n \frac{x_i}{|x_i - x_{med}|}}{\sum_{i=1}^n \frac{1}{|x_i - x_{med}|}} = \frac{\sum_{i=1}^n \frac{x_{med} - x_i}{|x_i - x_{med}|}}{\sum_{i=1}^n \frac{1}{|x_i - x_{med}|}} = 0.$$

Auf der Basis beider Gleichungen kann eine Berechnungsroutine für den Median ohne Sortieren der Stichprobe aufgestellt werden. Beispielsweise kann das Bisektionsverfahren für die Bestimmung der Nullstelle eingesetzt werden.

5. Weisen Sie nach, dass das arithmetische Mittel für eine Stichprobe mit n Werten die Funktion $\sum_{i=1}^n (x_i - x)^2$ minimiert!

Lösung:

Für die Funktion $f(x) = \sum_{i=1}^n (x_i - x)^2$ ist die erste und die zweite Ableitung gegeben durch

$$f'(x) = -2 \sum_{i=1}^n x_i + 2nx, f''(x) = 2n.$$

Für ein lokales Extremum der Funktion f muss $f'(x) = 0$ erfüllt sein. Die Nullstelle der ersten Ableitung ist gegeben durch

$$2nx = 2 \sum_{i=1}^n x_i \Leftrightarrow x = \frac{1}{n} \sum_{i=1}^n x_i = \bar{x}.$$

Die zweite Ableitung $f''(x) = 2n > 0$ ist immer positiv, also ist \bar{x} ein lokales Minimum!

6. Bestimmen Sie die Streuungsparameter für die Daten der Lebensdauer-Stichprobe auf Seite 401 und der Stichprobe aus Aufgabe 1!

Lösung:

Die Lösung finden Sie in der Lösung von Aufgabe 3!

7. Wie groß ist die Wahrscheinlichkeit, dass in einer Gruppe von 60 Personen zwei Personen am gleichen Tag Geburtstag haben?

Lösung:

Auf Seite 408 finden sie im Beispiel die Berechnung für den Fall von $n = 30$ Personen; dort war die Wahrscheinlichkeit, dass zwei Personen am gleichen Tag Geburtstag haben (bei 365 Tagen im Jahr) $p = 0,706 = 1 - 0,294$.

Jetzt ist die Wahrscheinlichkeit, dass alle Personen an einem verschiedenen Tag Geburtstag haben gegeben durch

$$P(A^c) = \frac{\prod_{i=306}^{365} i}{365^{30}} \approx 0,005877.$$

Daraus ergibt sich die gesuchte Wahrscheinlichkeit als

$$P(A) = 1 - P(A^c) \approx 0,994123.$$

8. Weisen Sie nach, dass für zwei unabhängige Ereignisse A und B auch A^c und B^c sowie A und B^c unabhängig sind!

Lösung:

Es gilt

$$A = (A \cap B) \cup (A \cap B^c),$$

und damit folgt aus den Wahrscheinlichkeitsaxiomen und der vorausgesetzten Unabhängigkeit von A und B

$$P(A) = P(A \cap B) + P(A \cap B^c) = P(A)P(B) + P(A \cap B^c)$$

und damit

$$P(A \cap B^c) = P(A)(1 - P(B)) = P(A)P(B^c).$$

Dann sind A und B^c unabhängig.

Wenn Sie in dieser Herleitung A durch A^c ersetzen erhalten Sie die Unabhängigkeit von A^c und B und auch von A^c und B^c .

9. Weisen Sie nach, dass für eine Zufallsvariable $P(X \in [\mu - 3\sigma; \mu + 3\sigma]) \geq \frac{8}{9}$ und $P(X \in [\mu - 4\sigma; \mu + 4\sigma]) \geq \frac{15}{16}$ gilt!

Lösung:

Allgemein gilt, das $P(x \in [\mu - k\sigma; \mu + k\sigma]) \geq 1 - \frac{1}{k^2}$. Also ist $P = \frac{8}{9}$ für $k = 3$ und $P = \frac{15}{16}$ für $k = 4$.

10. Für welchen Wert von a ist die Funktion $f(x) = a(x - 1)$ für $x \in [1; 3]$ und $f(x) = 0$ sonst Dichtefunktion einer Zufallsvariablen X ? Bestimmen Sie die zugehörige Verteilungsfunktion, $P(X > 2)$, $E(X)$, $Var(X)$, das 75%- und das 90%-Quantil!

Lösung:

Für eine Dichtefunktion muss

$$\int_{-\infty}^{\infty} f_X(t) dt = 1$$

erfüllt sein. Für die gegebene Funktion f folgt daraus die Bedingung

$$a \int_1^3 (x - 1) dx = 1$$

und $a = \frac{1}{2}$.

Dann ist die Verteilungsfunktion gegeben als

$$F_X(t) = \begin{cases} 0 & -\infty < t \leq 1, \\ \frac{1}{4}(t-1)^2 & 1 < t \leq 3, \\ 1 & t > 3. \end{cases}$$

Dann ist

$$P(X > 2) = 1 - P(X \leq 2) = 1 - F_X(2) = \frac{3}{4}.$$

Der Erwartungswert ist gegeben als

$$E(X) = \frac{1}{2} \int_1^3 x(x-1) dx = \frac{7}{3},$$

die Varianz als

$$\text{Var}(X) = \frac{1}{2} \int_1^3 \left(x - \frac{7}{3}\right)^2 (x-1) dx = \frac{2}{9}.$$

Das 75%-Quantil $t_{0,75}$ ist gegeben als Lösung der Gleichung

$$F_X(t_{0,75}) = 0,75.$$

Dadurch ergibt sich eine quadratische Gleichung für $t_{0,75}$ und die Lösung $t_{0,75} = 1 + \sqrt{3}$.

$t_{0,9}$ ist gegeben als Nullstelle der entsprechenden quadratischen Gleichung mit rechter Seite 0,9 als $t_{0,9} \approx 2,64$.

11. Weisen Sie nach, dass die Varianz einer geometrischen Verteilung durch $\text{Var}(X) = \frac{1-p}{p^2}$ gegeben ist.

Lösung:

Wie im Fall des Erwartungswerts der geometrischen Verteilung auf Seite 426 kann hier die zweite Ableitung der geometrischen Reihe als Potenzreihe verwendet werden. Es gilt also

$$\sum_{i=1}^{\infty} i \cdot q^{i-1} = \frac{1}{(1-q)^2}, \quad \sum_{i=2}^{\infty} i \cdot (i-1) q^{i-2} = \frac{2}{(1-q)^3}.$$

Die Varianz ist gegeben durch

$$\text{Var}(X) = E(X^2) - (E(X))^2,$$

also müssen wir zuerst $E(X^2)$ bestimmen. Es gilt

$$\begin{aligned} E(X^2) &= \sum_{i=1}^{\infty} i^2 p (1-p)^{i-1} \\ &= p \sum_{i=1}^{\infty} i^2 (1-p)^{i-1} \\ &= p \sum_{i=1}^{\infty} (i + i(i-1)) (1-p)^{i-1} \\ &= p \left[\sum_{i=1}^{\infty} i (1-p)^{i-1} + (1-p) \sum_{i=2}^{\infty} i(i-1) (1-p)^{i-2} \right] \\ &= \frac{1}{p} + p(1-p) \frac{2}{p^3} \\ &= \frac{2-p}{p^2}. \end{aligned}$$

Dann folgt

$$\text{Var}(X) = E(X^2) - (E(X))^2 = \frac{2-p}{p^2} - \frac{1}{p^2} = \frac{1-p}{p^2}.$$

12. Bei der Produktion eines Speicherbausteins sind im Durchschnitt 5% Ausschuss. Die Losgröße beträgt 200 Stück. Wie groß ist die Wahrscheinlichkeit, dass in einer Stichprobe von 20 Stück genau 5 Bausteine Ausschuss sind, dass alle verwendbar oder höchstens 2 Bausteine defekt sind? Bestimmen Sie den Erwartungswert und die Varianz dieser Verteilung!

Lösung:

Es handelt sich um einer relativ kleine Stichprobe, das gezogene Stück wird nicht in die Grundgesamtheit zurückgelegt; dann verwenden wir die hypergeometrische Verteilung.

Wir verwenden eine $\text{Hyp}(200; 10, 20)$ -Verteilung. Dann ist die gesuchte Wahrscheinlichkeit, dass in der Stichprobe genau 5 Stück Ausschuss sind

$$P_5 = \frac{\binom{10}{5} \binom{190}{15}}{\binom{200}{20}} \approx 0,001\,028.$$

Die Wahrscheinlichkeit, dass unter den 20 gezogenen Stücken genau 5 Ausschuss sind beträgt rund 0,1%.

Dass alle verwendbar, also kein Ausschuss in der Stichprobe ist, hat die Wahrscheinlichkeit

$$P_0 = \frac{\binom{10}{0} \binom{190}{20}}{\binom{200}{20}} \approx 0,339\,78.$$

Die Wahrscheinlichkeit, dass höchstens 2 Stück Ausschuss sind ist gegeben durch

$$P(X \leq 2) = P(X = 0) + P(X = 1) \approx 0,934\,71.$$

Der Erwartungswert ist $E(X) = n \frac{M}{N} = 10$, die Varianz ist $\text{Var}(X) = n \frac{M}{N} (1 - \frac{M}{N}) \frac{N-n}{N-1} \approx 8,859$.

13. Weisen Sie nach, dass für eine $P(\lambda)$ -verteilte Variable $\text{Var}(X) = \lambda$ gilt!

Lösung:

Für den Erwartungswert gilt $E(X) = \lambda$. Dann gilt

$$\begin{aligned} \text{Var}(X) &= \sum_{i=0}^{\infty} (i - \lambda)^2 \frac{\lambda^i}{i!} e^{-\lambda} \\ &= \sum_{i=0}^{\infty} i^2 \frac{\lambda^i}{i!} e^{-\lambda} - 2\lambda \sum_{i=0}^{\infty} i \frac{\lambda^i}{i!} e^{-\lambda} + \lambda^2 e^{-\lambda} \sum_{i=0}^{\infty} \frac{\lambda^i}{i!} \\ &= \sum_{i=0}^{\infty} i(i-1) \frac{\lambda^i}{i!} e^{-\lambda} + \sum_{i=0}^{\infty} i \frac{\lambda^i}{i!} e^{-\lambda} - 2\lambda \sum_{i=0}^{\infty} i \frac{\lambda^i}{i!} e^{-\lambda} + \lambda^2 e^{-\lambda} \sum_{i=0}^{\infty} \frac{\lambda^i}{i!} \\ &= \sum_{i=2}^{\infty} i(i-1) \frac{\lambda^i}{i!} e^{-\lambda} + \lambda - 2\lambda^2 + \lambda^2 e^{\lambda} e^{-\lambda} \\ &= \lambda^2 + \lambda - 2\lambda^2 + \lambda^2 = \lambda. \end{aligned}$$

14. Bei der Produktion von Stoff sind im Durchschnitt auf 100m 10 Fehler enthalten. Der Stoff wird zur Weiterverarbeitung in Abschnitte von jeweils 3m zerschnitten. Wie groß ist die Wahrscheinlichkeit, dass in einem Abschnitt kein Fehler enthalten ist?

Lösung:

Im Durchschnitt ergeben sich

$$\frac{m}{N} = 0,1 \text{ Fehler pro Meter.}$$

Für einen Abschnitt von 3 Meter folgt dann, dass es

$$n \frac{m}{N} = \lambda = 0,3$$

Fehler pro Abschnitt gibt.

Die Wahrscheinlichkeit, dass ein Abschnitt von 3 Metern keinen Fehler aufweist ist dann gegeben durch eine $P(0,3)$ -Verteilung mit

$$P(X = 0) = \frac{0,3^0}{0!} e^{-0,3} \approx 0,741.$$

15. Stellen Sie die Poisson-Verteilung für $\lambda = 9$, $n = 1000$ und $\lambda = 1$, $n = 200$ als Stabdiagramm dar!

Lösung:

In Abbildung 15.6 finden Sie das Stabdiagramm für $\lambda = 9$, Abbildung 15.8 den Fall $\lambda = 1$. Man erkennt gut, dass im Fall $\lambda = 1$ die Verteilung unsymmetrisch ist. Mit wachsendem λ wird die Verteilung immer „symmetrischer“.

Wie bereits im Buch vermerkt ist die Poisson-Verteilung eine gute Approximation einer Binomialverteilung. In der Aufgabenstellung ist neben λ auch ein Wert für n gegeben; daraus ergibt sich für die Binomialverteilung $p = \frac{\lambda}{n} = \frac{9}{1000}$ bzw. $p = \frac{\lambda}{n} = \frac{1}{1000}$. In den Abbildungen 15.7 und 15.9 finden Sie die Stabdiagramme für diese Binomialverteilungen.

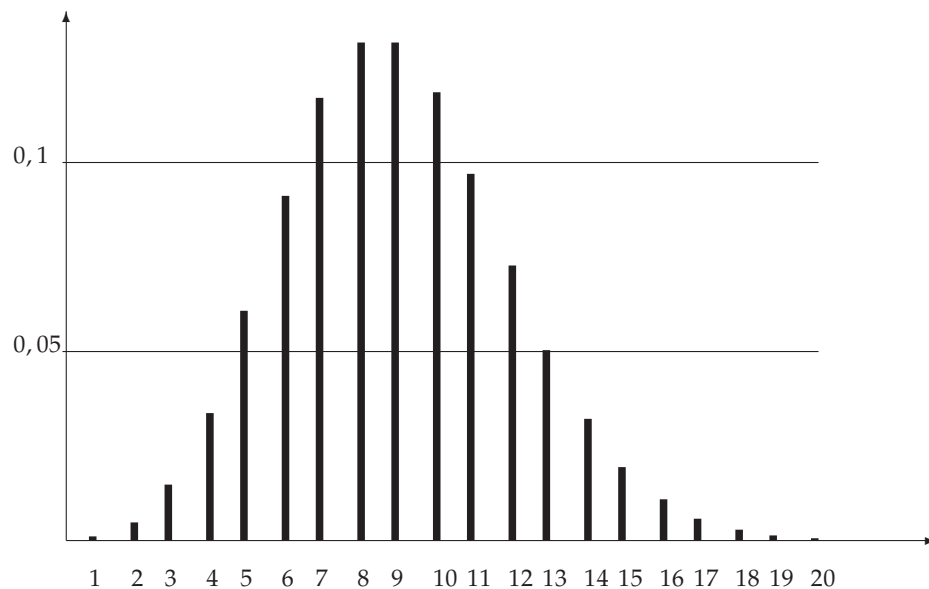


Abbildung 15.6: Das Stabdiagramm der Poisson-Verteilung mit $\lambda = 9$ für Aufgabe 15

16. Weisen Sie nach, dass für eine $E(\lambda)$ -verteilte Zufallsvariable $E(X) = \frac{1}{\lambda}$ und $Var(X) = \frac{1}{\lambda^2}$ gilt!

Lösung:

Der Erwartungswert einer $E(\lambda)$ -verteilten Zufallsvariablen X ist gegeben durch das Integral

$$E(X) = \int_0^{\infty} x \lambda e^{-\lambda x} dx.$$

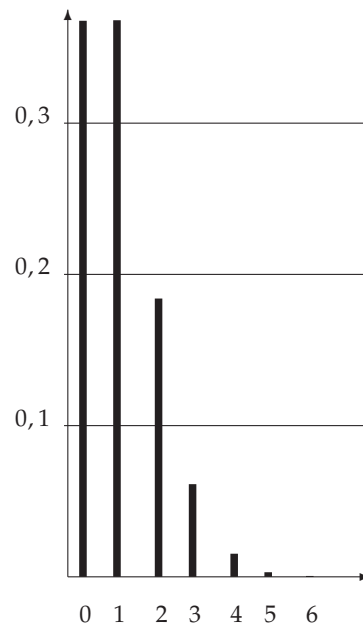


Abbildung 15.9: Das Stabdiagramm der Binomialverteilung mit $\lambda = 1$, $n = 1000$ und $p = \frac{1}{1000}$ für Aufgabe 15

$f(x) = x$, also ist $f'(x) = 1$ und $g'(x) = \lambda e^{-\lambda x}$, dann ist $g(x) = -e^{-\lambda x}$.

$$\begin{aligned} E(X) &= \int_0^{\infty} x \lambda e^{-\lambda x} dx \\ &= \left[-x e^{-\lambda x} \right]_0^{\infty} + \int_0^{\infty} e^{-\lambda x} dx \\ &= \left[-\frac{1}{\lambda} e^{-\lambda x} \right]_0^{\infty} \\ &= \frac{1}{\lambda}. \end{aligned}$$

Für die Bestimmung der Varianz berechnen wir $E(X^2)$, wieder mit partieller Integration:

$$\begin{aligned} E(X^2) &= \int_0^{\infty} x^2 \lambda e^{-\lambda x} dx \\ &= \left[x^2 e^{-\lambda x} \right]_0^{\infty} + 2 \int_0^{\infty} x e^{-\lambda x} dx \\ &= \frac{2}{\lambda} \cdot \frac{1}{\lambda} = \frac{2}{\lambda^2}. \end{aligned}$$

Dann ist die Varianz gegeben durch

$$\text{Var}(X) = E(X^2) - E(X)^2 = \frac{2}{\lambda^2} - \frac{1}{\lambda^2} = \frac{1}{\lambda^2}.$$

17. Erfahrungswerte zeigen, dass die Reparaturzeit eines PCs in einem Unternehmen exponentiell verteilt ist. Die mittlere Reparaturzeit beträgt 4 Stunden. Wie groß ist die Wahrscheinlichkeit, dass für einen PC zwischen 5 und 8 Stunden, höchstens 10 Stunden und mindestens 3 Stunden benötigt werden?

Lösung:

Wir müssen in einem ersten Schritt den Parameter λ der angenommenen Exponentialverteilung bestimmen. In der Aufgabenstellung finden Sie den Erwartungswert, nämlich 4 Stunden

für die Reparaturzeit. Der Erwartungswert der Exponentialverteilung ist $E(X) = \frac{1}{\lambda} = 4$, dann ist also $\lambda = 0,25$.

Es ist dann

$$P(5 < X < 8) = P(X < 8) - P(X < 5) = (1 - e^{-0,25 \cdot 8}) - (1 - e^{-0,25 \cdot 5}) \approx 0,151.$$

Dass eine Reparatur höchstens 10 Stunden dauert ist 0,9197. Die Wahrscheinlichkeit, dass die Reparatur mindestens 3 Stunden dauert ist gegeben durch

$$P(X \geq 3) = 1 - P(X < 3) \approx 0,472.$$

18. Weisen Sie nach, dass $P(|X - \mu| \leq k\sigma) = 2\Phi(k) - 1$ gilt.

Lösung:

Zur Erinnerung: ist X eine $N(\mu, \sigma^2)$ -verteilte Zufallsvariable, dann ist

$$Y = \frac{X - \mu}{\sigma}$$

$N(0,1)$ -verteilt!

$$\begin{aligned} P(|X - \mu| \leq k\sigma) &= P\left(\left|\frac{X - \mu}{\sigma}\right| \leq k\right) \\ &= P\left(-k < \frac{X - \mu}{\sigma} < k\right) \\ &= P(Y < k) - P(Y < -k) \\ &= \Phi(k) - \Phi(-k) \\ &= \Phi(k) - (1 - \Phi(k)) \\ &= 2\Phi(k) - 1. \end{aligned}$$

19. Implementieren Sie die Approximation der Normalverteilung $\Phi(0;1)$ nach der Formel auf Seite 431 in der Programmiersprache Ihrer Wahl und vergleichen Sie die Ergebnisse mit Tafelwerken!

Lösung:

Die Approximation von Ponton (Sie finden Sie unter der URL <http://www.uiowa.edu/grpporc/crisp/crisp.3.6.htm>) dient zur Näherung der sogenannten „Error Function“

$$\operatorname{erfc}(t) = \frac{1}{\sqrt{2\pi}} \int_0^t e^{-\frac{x^2}{2}} dx.$$

Die standardisierte Normalverteilung ist dann gegeben durch

$$\Phi(t) = \begin{cases} \frac{1}{2} + \operatorname{erfc}(t), & t \geq 0, \\ \frac{1}{2} - \operatorname{erfc}(t), & t < 0. \end{cases}$$

Dann ergibt sich der folgende Java-Code:

```
private static double ponton(double x) {
    double k, value = 0.0;

    k = 0.5 + (1.0/Math.sqrt(Math.PI)-0.5) *
        Math.exp(-x*x/Math.sqrt(2.0*Math.PI));
    value = k * Math.sqrt(1.0-Math.exp(-x*x/2.0));
}
```

```

if (x >= 0.0)
    return 0.5 + value;
else
    return 0.5 - value;
}

```

Die Tabellen, beispielsweise in Greiner/Tinhofer, Stochastik für Studienanfänger der Informatik aus dem Hanser-Verlag sind typischerweise so aufgebaut, dass in einer quadratischen Matrix Werte für $\Phi(t)$ stehen. In den Zeilen finden Sie Werte von t , in den Spalten die Überschriften $0.00, 0.01, \dots, 0.09$. In der Zeile mit $t = 0.5$ und Spaltenüberschrift 0.04 können Sie dann den Wert für $t = 0.54$ ablesen: $\Phi(0.54) \approx 0.70540$. Der Java-Code liefert die Näherung 0.70521 , das entspricht einem Fehler von $\delta = 1.866 \cdot 10^{-4}$. In Tabelle 15.1 finden Sie die berechneten Werte, die Tabellenwerte aus Greiner/Tinhofer und die Fehler für die komplette Zeile $t = 0.5$. Je näher die Argumente bei 0 liegen desto besser ist die Approximation von Ponton. Beispielsweise ist für $t = 0.05$ in Greiner/Tinhofer ein Wert von 0.51994 angegeben, Ponton berechnet 0.5199386205616786 , ein Fehler von $1,379 \cdot 10^{-6}$.

x	$\Phi(x)$	Greiner/Tinhofer	Fehler
0,5	0,6913083625030305	0,69146	$1,5163749696944784 \cdot 10^{-4}$
0,51	0,6948119739173705	0,69497	$1,5802608262949214 \cdot 10^{-4}$
0,52	0,6982975085457394	0,69847	$1,7249145426068146 \cdot 10^{-4}$
0,53	0,7017647134242829	0,70194	$1,7528657571708184 \cdot 10^{-4}$
0,54	0,7052133408637061	0,7054	$1,8665913629389852 \cdot 10^{-4}$
0,55	0,7086431484939236	0,70884	$1,9685150607640090 \cdot 10^{-4}$
0,56	0,712053899306031	0,71226	$2,0610069396898023 \cdot 10^{-4}$
0,57	0,7154453616915879	0,71566	$2,1463830841206288 \cdot 10^{-4}$
0,58	0,7188173094792161	0,71904	$2,2269052078394136 \cdot 10^{-4}$
0,59	0,7221695219685125	0,7224	$2,3047803148756874 \cdot 10^{-4}$

Tabelle 15.1: Die Approximation nach Ponton, Werte aus Greiner/Tinhofer und die Fehler für eine Zeile der tabellierten Standardnormalverteilung in Aufgabe 19

Ponton gibt eine noch einfachere Approximation an; er ersetzt die Näherung aus der Aufgabenstellung durch

$$K(t) = 0,5 + 0,064 \cdot e^{-0,4 \cdot t^2}, \Phi(t; 0, 1) \approx 0,5 \pm K(t) \sqrt{1 - e^{-\frac{t^2}{2}}}.$$

Für $t = 0.59$ ergibt sich damit ein Näherungswert von $0,72209537820118128$, eine Abweichung vom Tabellenwert von $3,046217988188493 \cdot 10^{-4}$.

20. Wie groß ist die Wahrscheinlichkeit, dass eine $N(50; 10)$ -verteilte Zufallsvariable im Intervall $(45; 55)$, unterhalb von 45, oberhalb von 60 und um mehr als 5 vom Erwartungswert abweicht?

Lösung:
Es gilt

$$\begin{aligned}
P(45 \leq X \leq 55) &= \Phi(55; 10, 0, 1) - \Phi(45; 10, 0, 1) \\
&= \Phi(0, 5) - \Phi(-0, 5) = 2\Phi(0, 5) - 1 \\
&\approx 0,38292.
\end{aligned}$$

Oberhalb von 60 liegt die Zufallsvariable mit Wahrscheinlichkeit

$$P(X > 60) = 1 - \Phi(1) \approx 0,15866.$$

Die Wahrscheinlichkeit, dass sie um mehr als 5 cm vom Erwartungswert abweicht ist

$$P(|X - \mu| > 5) = 1 - P(|X - \mu| \leq 5) = 1 - \left(2\Phi\left(\frac{5}{10}\right) - 1\right) \approx 0,617.$$

21. Weisen Sie nach, dass bei Binomial- und Poisson-Verteilung der Maximum-Likelihood-Schätzer mit dem arithmetischen Mittel übereinstimmt!

Lösung:

Die Maximum-Likelihood-Funktion für die Poisson-Verteilung ist

$$\begin{aligned} L(\mathbf{x}; \lambda) &= \prod_{i=1}^n \frac{\lambda^{x_i}}{x_i!} e^{-\lambda} \\ &= e^{-n\lambda} \prod_{i=1}^n \frac{\lambda^{x_i}}{x_i!} \\ &= e^{-n\lambda} \frac{\lambda^{n\bar{x}}}{\prod_{i=1}^n x_i!}. \end{aligned}$$

Logarithmieren ergibt dann

$$LL(\lambda) = \ln \left(e^{-n\lambda} \frac{\lambda^{n\bar{x}}}{\prod_{i=1}^n x_i!} \right) = -n\lambda + n\bar{x} \ln(\lambda) - \ln \left(\prod_{i=1}^n x_i! \right).$$

Dann ist

$$LL'(\lambda) = -n + \frac{n\bar{x}}{\lambda},$$

Nullsetzen dieser Ableitung ergibt den Maximum-Likelihood-Schätzer $\lambda = \bar{x}$.

Die Maximum-Likelihood-Funktion für die Binomialverteilung (mit $p \neq 1, p \neq 0$) ist

$$L(\mathbf{x}; p) = \prod_{i=1}^n \binom{n}{x_i} p^{x_i} (1-p)^{n-x_i}.$$

Logarithmieren dieser Funktion ergibt die Funktion

$$\begin{aligned} LL(p) &= \sum_{i=1}^n \ln \left(\binom{n}{x_i} p^{x_i} (1-p)^{n-x_i} \right) \\ &= \sum_{i=1}^n \ln \left(\binom{n}{x_i} \right) + \ln(p)n\bar{x} + \ln(1-p)n(n-\bar{x}) \end{aligned}$$

Die erste Ableitung von LL ist gegeben durch

$$\begin{aligned} LL'(p) &= \frac{n\bar{x}}{p} - \frac{n(n-\bar{x})}{1-p} \\ &= \frac{n\bar{x} - pn^2}{p(1-p)} \end{aligned}$$

Die Nullstelle dieser Ableitung ist gegeben durch

$$p = \frac{\bar{x}}{n}.$$

Zur Erinnerung, $n \cdot p$ ist der Erwartungswert einer binomialverteilten Zufallsvariablen! Also ist das arithmetische Mittel \bar{x} der Maximum-Likelihood-Schätzer für den Erwartungswert der Zufallsvariablen.