

Doina Logofătu

Grundlegende Algorithmen mit Java

- Vom Algorithmus zum fertigen Programm
- Lern- und Arbeitsbuch für Informatiker und Mathematiker



Mit Online-Service
zum Buch



Doina Logofătu

Grundlegende Algorithmen mit **Java**

60 gelöste Probleme, 240 Aufgaben, über 130 Bilder

Vom Algorithmus zum fertigen Programm
Lern- und Arbeitsbuch für Informatiker und Mathematiker



*Take your life in your own hands and what happens?
A terrible thing: no one is to blame.*
Erica Jong

Geleitwort

Das vorliegende Buch „Grundlegende Algorithmen mit Java“ vermittelt Grundlagen zu Algorithmen und zur Komplexitätstheorie und illustriert danach allgemeine Lösungsprinzipien (z. B. „Teile und Herrsche“, „Dynamische Programmierung“). Drei ausgewählte Problemstellungen – „Verschachtelte Schachteln“ als einführendes Beispiel, das „Data Ordering Problem“ als Beispiel für ein NP-vollständiges Problem und Berechnung von Potenzsummen als mathematisch nichttriviales Problem werden ausführlicher in eigenen Kapiteln behandelt.

Jedes Lösungsprinzip wird nach einem theoretischen Vorspann anhand von Problemen - von trivialen Beispielen bis hin zu umfangreicheren Aufgaben aus Programmierwettbewerben - erläutert, die anhand von Übungsaufgaben vertieft werden können. In dem Buch wird ein besonderer Schwerpunkt auf das Prinzip der dynamischen Programmierung gesetzt.

Zu allen näher besprochenen Algorithmen gibt es jeweils, nach einigen Kommentaren zur Implementierung, ein komplettes Java-Programm. Das Buch setzt einfache Kenntnisse in Java voraus, wobei einige Konzepte und Grundlegende Klassen am einführenden Beispiel der verschachtelten Schachteln wiederholt werden. Vom über prozedurale Programmierung hinaus gehenden Sprachumfang von Java kommt hauptsächlich die Objektorientierung zum Einsatz (insbesondere für abstrakte Datentypen), Nebenläufigkeit und Netzwerktechniken kommen nicht vor. Dafür finden sich einige Sprachkonstrukte und Klassen, die in Java 5, teils sogar Java 6 eingeführt worden sind, und man lernt manche wenig bekannte Bereiche kennen, z. B. die *Buffer*-Klassen aus dem Paket *java.nio* und *java.util.Arrays*. Fast alle Programme erhalten ihre Eingaben aus einer Datei (oder von der Tastatur) und schreiben die Ergebnisse in eine andere Datei; im Kapitel "Rekursion" kommen daneben auch graphische Ausgaben fraktaler Strukturen vor.

Neben vielen Abbildungen lockern auch einige kleine Bilder, jeweils passend zum Thema, den Text auf. Am Ende des Buches finden sich ein Literaturverzeichnis und ein Stichwortverzeichnis.

Dr. Eric Müller

Vorwort

Viele Problemstellungen dieses Buches habe ich bereits in meinem Buch *Algorithmen und Problemlösungen mit C++* behandelt, das im letzten Jahr erschienen ist. Die Theorie und die Problembeschreibungen sind größtenteils die gleichen, geändert haben sich natürlich die Programme und die Abschnitte, die sich auf die Programmiersprache beziehen. Neu in diesem Buch ist das erste Kapitel über Algorithmen und die Komplexitätstheorie und das vierte Kapitel *Data Ordering Problem*, in dem ein NP-vollständiges Problem analysiert und mit verschiedenen Ansätzen gelöst wird. Hinzugekommen sind auch neue Probleme wie z. B. die Huffman-Codierung im *Greedy*-Kapitel, Sudoku im *Backtracking*-Kapitel und *LCS (longest common substring)* im Kapitel über die Dynamische Programmierung. Ich habe die Problemstellungen und Erklärungen aus dem C++-Buch überarbeitet und mir außerdem ein paar zusätzliche Übungen ausgedacht. Wie im alten Buch möchte ich jetzt durch unterhaltsame Aufgaben Ihre Begeisterung und Ihre Neugier für die Informatik wecken und Ihnen die Fähigkeit vermitteln, für neue Probleme eigene Lösungen zu finden.

Grundlegende Algorithmen mit Java beinhaltet in den neun Kapiteln 60 Probleme bzw. Aufgaben, die vollständig analysiert und in Java gelöst werden, knapp 240 Übungsaufgaben und gut 130 Abbildungen und Bilder. Die nötigen Grundlagen am Anfang jedes Kapitels ermöglichen einen theoretischen Überblick über die Thematik. Zu jedem Problem wird beschrieben, wie die Eingabe- und Ausgabedateien aufgebaut sind, und ein Beispiel dafür angegeben. Damit können Sie selbstgeschriebene Programme überprüfen. Dann folgt der Abschnitt *Problemanalyse und Entwurf der Lösung*, der einen detaillierten algorithmischen/mathematischen Lösungsansatz und ein Java-Programm präsentiert. Die Programme sind kompakt und die Schlüsselwörter in blauer Farbe, um eine gute Lesbarkeit zu gewährleisten. Darum befinden sich auch die Kommentare meistens nicht direkt im Code, sondern daneben in blauen Kästchen. Die Programme sind mit dem JDK 1.6. kompiliert worden, den SUN® kostenlos zur Verfügung stellt (<http://java.sun.com/javase/downloads>). Zu jedem Problem gehören Übungen, die Sie meist auffordern, Programme zu ändern oder neue Programme zu schreiben, damit Sie das gerade Erlernte wiederholen können und ihre Programmierfähigkeiten verbessern.

Alle Aufgaben bzw. Probleme wenden die am jeweiligen Kapitelanfang vorgestellten mathematischen Konzepte bzw. algorithmischen Verfahren an und vertiefen sie. Die Absicht, die dahinter steht, ist die, dass Sie die Theorie dadurch erlernen, indem Sie sehen und üben, wie sie in der Praxis, also in den Problemen, eingesetzt wird. Viele Probleme sind klassisch, wie z. B. Fibonacci-Zahlen, Koch'sche Schneeflockenkurve, Türme von Hanoi, *N-Damen*, Haus des Nikolaus, Kartenfärbung, Konvexe Hülle, Multiplikation einer Matrizenfolge und Edit-Distanz. Aufgaben aus den Programmierwettbewerben *Association for Computing Machinery (ACM)*, *International Olympiad in Informatics (IOI)* und *Central-European Olympiad in Informatics (CEOI)* inspirierten mich dazu, zahlreiche Probleme für das Buch zu formulieren.

Ab und zu finden Sie, quasi als Belohnung für Ihren Fleiß, zwischen zwei Kapiteln Überraschungsbilder wie: Don Quijotes Windmühlen, Schwäne auf der Isar, Kapelle in Dorohoi, Statue in Bremen, Strandkörbe auf Norderney.

Den Online-Service zum Buch finden Sie hier:

www.algorithmen-und-problemlösungen.de

Ich bitte Sie, meine verehrten Leser, mir Ihre Anmerkungen, Lob und Kritik zu senden:

doinabooks@yahoo.com

Dafür bedanke ich mich im Voraus.

Viel Vergnügen beim Lesen und spannendes Lernen!

München,
im August 2007

Doina Logofătu

www.doina-logofatu.de

Danksagung

Ganz besonders herzlich bedanke ich mich bei Herrn *Dr. Eric Müller*, der mir beim Schreiben dieses zweiten Buches wieder treu zur Seite stand. Unermüdlich hat er sehr viele schöne Ideen, wunderbare Vorschläge und Korrekturen und auch ein komplettes Programm für das Problem Orangensport im Backtracking-Kapitel beigesteuert. *Dr. Eric Müller* gewann zwischen 1986 und 1988 dreimal einen zweiten Preis (Silbermedaille) bei den Internationalen Mathematik-Olympiaden (*IMO*) und wirkt seit einigen Jahren bei der Vorbereitung der deutschen Teilnehmer auf diesen Schülerwettbewerb mit.

Ebenso bedanke ich mich ganz besonders herzlich beim General, der mir meisterhaft bei der Erstellung der Java-Programme geholfen und mir somit viel Zeit erspart hat.

Mein besonderer Dank gebührt Herrn *Prof. Dr. Rolf Drechsler*, dem Leiter der Arbeitsgruppe Rechnerarchitektur der Universität Bremen. Von ihm habe ich gelernt, mich besser in die Position des Lesers zu versetzen und meine Aufgaben „eine nach der anderen“ zu erledigen. Außerdem hat er mir die Vorlagen für die Testmusterkompaktierung (Problem 12, Kapitel 7, *Backtracking*) und das *Data Ordering Problem* (Kapitel 4) gegeben. Über diese Probleme und andere Themen habe ich mit *Görschwin Fey*, *Daniel Große*, *Dr. Rüdiger Ebendt*, *Sebastian Kinder* und *Junhao Shi* interessante Gespräche geführt. Dafür danke ich ihnen.

Für Informationen zu ACM Problemen danke ich den Herren *Prof. Dr. Miquel Revilla Samos* (Universität Valladolid, Spanien), *Prof. Dr. Cristian Giumale* und *Prof. Dr. Nicolae Țăpuș* (beide Universität Bukarest, Rumänien).

Für die Erlaubnis, Fotos im Buch verwenden zu dürfen, bedanke ich mich bei den Herren *Prof. Dr. Stephen Arthur Cook* (erstes Kapitel), *Michael W. Davidson* (Fotos von Euklid, Euler und Fermat), *Robert D. Colburn* (Foto von Richard Bellman) und *Wolfgang Weege* (das Spiegelfoto im Rekursions-Kapitel).

Ich danke Katja Schnelle-Romaus (Karsunke) dafür, dass sie mich vor ein paar Jahren beim Erlernen der deutschen Sprache unterstützt hat. Sie schenkte mir damals ein Buch mit der Geschichte über die Feldmaus Frederik und steigerte dadurch meine Lernmotivation um ein Vielfaches.

Ein tiefer Dank gebührt meinen Mathematiklehrern *Rodica Ungureanu* (Gymnasium „Gr. Ghica“) und *Victor Barnea* (Volksschule 7) in meiner kleinen Heimatstadt Dorohoi im Nordosten Rumäniens. Sie haben es mit ihren anspruchsvollen und interessanten Aufgaben geschafft, mich immer wieder für die Mathematik zu begeistern.

Und schließlich danke ich allen, die die Fertigstellung des Buches ermöglicht haben.

München,
im August 2007

Doina Logofătu

Inhaltsverzeichnis

GELEITWORT von <i>Dr. Eric Müller</i>	VII
VORWORT	IX
DANKSAGUNG	XI
1 ALGORITHMEN – GRUNDLEGENDE KONZEPTE	1
Abstammung des Wortes Algorithmus	1
Alternative Definitionen	1
Beispiele für Algorithmen	2
<i>Euklidischer Algorithmus</i>	2
<i>Das Sieb des Eratosthenes</i>	3
<i>Binäre Suche</i>	4
<i>Rezept für Tiramisu</i>	6
Vom Problem zur Lösung	7
Eigenschaften eines Algorithmus	10
Algorithmik	10
Das <i>RAM</i> -Rechnermodell	11
Die Komplexität von Algorithmen	12
Optimalität, Reduktion, Beispiele	14
Wachstum von $O(g(n))$	17
Die reelle Zeit eines Algorithmus (polynomial vs. exponentiell)	17
Klassifizierung der Probleme (P, NP, NP-vollständig, NP-hart)	18
Probleme NP-vollständig (NP- <i>complete</i>)	19
Das Erfüllbarkeitsproblem (<i>SAT</i>)	20
Die Klasse der NP-hart Probleme	21
Aufgaben	22

2 VERSCHACHELTE SCHACHTELN	25
Problembeschreibung	25
Problemanalyse und Entwurf der Lösung	26
Der Algorithmus	27
Das Programm	29
Die Programmanalyse	32
Aufgaben	37
Anmerkungen	38
3 GREEDY	39
Grundlagen	39
Problem 1. Rucksackproblem	40
Problem 2. Kartenfärbung	43
Problem 3. Springer auf dem Schachbrett	45
Problem 4. Minimaler Spannbaum (Kruskal-Algorithmus)	48
Problem 5. Huffman-Kodierung	56
4 DATA ORDERING PROBLEM	65
Problembeschreibung	65
Problemdomäne, Definitionen	66
DOP und DOPI sind NP-vollständig	71
Algorithmen für DOP und DOPI	72
Zufällige-Lösung-Algorithmen (RAN)	73
Exakt-Algorithmen (EX)	74
<i>Greedy_Min</i> -Algorithmen (GM)	74
<i>Greedy_Min Simplified</i> -Algorithmen (GMS)	75
Algorithmen mit unterer Schranke (LB)	75
Implementierungsdetails	77
Programm	84
Auswertung der Ergebnisse	94
Aufgaben	96
5 REKURSION	99
Vollständige Induktion	99
Rekursion: Grundlagen	105
Problem 1. Quersumme und Spiegelung einer natürlichen Zahl	106
Problem 2. Die Zahl 4	108
Problem 3. Rest großer Potenzen	111
Problem 4. Die Torte (lineare Rekursion)	115
Problem 5. Die Ackermannfunktion (verschachtelte Rekursion, "compound recursion")	118
Problem 6. Rekursive Zahlenumwandlung (Dezimalsystem in System mit Basis P)	120

Problem 7. Summe zweier Wurzeln (verzweigte Rekursion)	123
Problem 8. Collatz-Funktion (nicht-monotone Rekursion)	125
Problem 9. Quadrate und Quadrätchen	127
Problem 10. Quadrate (direkte Rekursion)	130
Problem 11. Quadrate und Kreise (indirekte Rekursion)	133
Problem 12. Die Koch'sche Schneeflockenkurve	135
6 TEILE UND HERRSCHE	145
Grundlagen	145
Problem 1. Größter gemeinsamer Teiler mehrerer Zahlen	146
Problem 2. Die Türme von Hanoi	148
Problem 3. Integral mit Trapezregel	150
Problem 4. <i>Quicksort</i>	152
Problem 5. <i>Mergesort</i> (Sortieren durch Verschmelzen)	155
Problem 6. <i>Quad</i> -Bäume	157
Problem 7. Diskrete Fourier-Transformation (DFT)	162
7 BACKTRACKING	169
Problem 1. Das Problem der n Damen	169
Allgemeine Bemerkungen zum Backtracking-Verfahren	175
Problem 2. Das Problem der n Türme	178
Problem 3. Das Problem der Türme auf den ersten m Reihen	179
Problem 4. Das Problem der aufsteigenden Türme auf den ersten m Reihen	180
Problem 5. Die Freundschafts-Jugendherberge	181
Problem 6. Partitionen einer natürlichen Zahl	182
Problem 7. Erdkunde-Referate	185
Problem 8. Alle Wege des Springers	188
Problem 9. Das Fotoproblem	191
Problem 10. Der ausbrechende Ball	193
Problem 11. Orangensport	196
Problem 12. Testmusterkompaktierung	205
Problem 13. Sudoku	214
Problem 14. Das Haus des Nikolaus	221
Noch 10 Probleme	224
8 DYNAMISCHE PROGRAMMIERUNG	231
Grundlagen, Eigenschaften des Verfahrens	231
1. <i>Ursprung des Konzeptes</i>	231
2. <i>Optimalitätsprinzip</i>	231
3. <i>Überlappung der Probleme, Speicherung der optimalen Teilproblemlösungen (Memoization)</i>	232
4. <i>Einführendes Beispiel – die Fibonacci-Folge</i>	232
5. <i>Bottom-up versus top-down</i>	234

6. Vergleich mit anderen Verfahren	234
Aufgaben	235
Problem 1. Das Zählen der Kaninchen	236
Problem 2. Längste aufsteigende Teilfolge	240
Problem 3. Längste gemeinsame Teilfolge (LCS)	245
Problem 4. Zahlen-Dreieck	249
Problem 5. Domino	253
Problem 6. Verteilung der Geschenke	258
Problem 7. Ähnliche Summe	261
Problem 8. Schotten auf dem Oktoberfest	266
Problem 9. Springer auf dem Schachbrett	275
Problem 10. Summen von Produkten	280
Problem 11. Minimale Triangulierung eines konvexen Vielecks	286
Problem 12. Multiplikation einer Matrizenfolge	291
Problem 13. Edit-Distanz	297
Problem 14. Arbitrage	305
9 POTENZSUMMEN	311
Problembeschreibung	311
Problemanalyse. Algebraische Modellierung	311
Von der Rekursionsgleichung zum Algorithmus	313
Der Algorithmus	316
Programm	318
Aufgaben	321
LITERATURVERZEICHNIS	323
STICHWORTVERZEICHNIS	327

Literaturverzeichnis

- [Aig02] Martin Aigner, Günter M. Ziegler, *Das BUCH der Beweise*, Springer Verlag, 2002.
- [Ban83] Horia Banea, *Probleme de matematică traduse din revista sovietică KVANT*, Editura Didactică și Pedagogică, București, 1983.
- [Bar93] Barnsley, M. F., *Fractals everywhere. Second edition*, Academic Press Inc., Boston, 1993.
- [Bur94] Stan, M., Burleson, W., *Limited-weight codes for low-power I/O*, in *Int'l Workshop on Low Power Design*, 1994.
- [Coo71] Cook, S., A., *The complexity of theorem-proving procedures*, Proc. 3rd Annual ACM Symp. Theory of Computing, S. 151-158.
- [Cor04] Th. H. Cormen, C. E. Leiserson, R. Rivest, C. Stein, *Algorithmen-Eine Einführung*, Oldenbourg Wissenschaftsverlag, München, 2004.
- [Die00] Reinhard Diestel, *Graphentheorie*, Springer-Verlag, Heidelberg, 2000.
- [Dre99] Drechsler, N., Drechsler, R., *Exploiting don't cares during data sequencing using genetic algorithms*, in *ASP Design Automation Conf.*, pp. 303-306, 1999
- [Fla02] David Flanagan, *JAVA in a nutshell. A desktop quick reference*, O' Reilly, Sebastopol, 2002.
- [For96] Otto Forster, *Algorithmische Zahlentheorie*, Vieweg Verlag, Wiesbaden, 1996.
- [Gan91] Mircea Ganga, *Teme și probleme de matematică*, Editura Tehnică, București, 1991.
- [Gar79] Garey, M. R., Johnson, D. S., *Computers and Intractability: a Guide to the Theory of NP-Completeness*, W. H. Freeman, New York.

- [GM] *Gazeta Matematică*, mathematisches Magazin, Sammlung
- [Gra94] Graham R. L., Knuth D. E., Patashnik O., *Concrete Mathematics*, Addison-Wesley, 1994.
- [Heu03] Volker Heun, *Grundlegende Algorithmen. Einführung in den Entwurf und die Analyse effizienter Algorithmen*, 2. Auflage, Vieweg Verlag, Wiesbaden, 2003.
- [Iva02] Cornelia Ivaşc, Mona Prună, Luminiţa Condurache, Doina Logofătu, *Informatica C++. Manual pentru clasa a XI-a*, Editura Petriom, Bucureşti, 2002.
- [Knu97] Donald E. Knuth, *The Art of Computer Programming, Volume I: Fundamental Algorithms*, Addison-Wesley Longman, Amsterdam; Auflage: 3. A., 1997.
- [Kre99] Donald L. Kreher, Douglas R. Stinson, *Combinatorial algorithms. Generation, Enumeration, and Search.*, CRC Press, 1999.
- [Krü06] Guido Krüger, *Hanbuch der Java-Programmierung*, 4., aktualisierte Auflage, Addison-Wesley, München, 2006.
- [Log01] Doina Logofătu, *C++. Probleme rezolvate şi algoritmi*, Editura Polirom, Iaşi, 2001.
- [Log05a] Doina Logofătu, *Suma puterilor asemenea*, GInfo, 15/2 2005, S. 40-43, WWW: http://www.ginfo.ro/revista/15_2/focus2.pdf.
- [Log05b] Doina Logofătu, *Şirul lui Catalan*, GInfo, 15/5, 2005, S. 36-41, WWW: http://www.ginfo.ro/revista/15_5/mate1.pdf.
- [Log05c] Doina Logofătu, *De la problema cutiilor speciale la elemente POO cu C++*, GInfo, 15/5, S. 27-30, WWW: http://www.ginfo.ro/revista/15_5/focus1.pdf.
- [Log06] Doina Logofătu, *Bazele programării în C. Aplicaţii*, Editura Polirom, Iaşi, 2006.
- [Log06a] Doina Logofătu, Rolf Drechsler, *Efficient Evolutionary Approaches for the Data Ordering Problem with Inversion*, 3rd European Workshop on Hardware Optimisation Techniques (EvoHOT), LNCS 3907, S. 320-331, Budapest, 2006.
- [Log06b] Doina Logofătu, *Greedy Approaches for the Data Ordering Problem with and without Inversion*, Proceedings of the ROSYCS 2006, Romanian Symposium on Computer Science, S. 65-80, Iaşi.
- [Log06c] Doina Logofătu, *Algorithmen und Problemlösungen mit C++*, Vieweg-Verlag, Wiesbaden, 2006.
- [Log07] Doina Logofătu, *Algoritmi fundamentali în C++. Aplicaţii*, Polirom, Iaşi, 2007.
- [Mär01] *Das große Märchenbuch*, Droemersch Verlagsanstalt Th. Knauer Nachf., München, 2001.

- [Mat02] J. Matoušek, J. Nešetřil, *Diskrete Mathematik. Eine Entdeckungsreise*, Springer Verlag, 2002.
- [Mur95] Murgai, R., Fujita, M., Krishnan, S. C., *Data sequencing for minimum transition*, in Int'l Workshop on Logic Synth., 1995.
- [Mur97] Murgai, R., Fujita, M., Krishnan, S. C., *Data sequencing for minimum-transition transmission*, in IFIP Int'l Conf. on VLSI, 1997.
- [Mur98] Murgai, R., Fujita, M., Oliveira, A., *Using complementation and resequencing to minimize transitions*, in Design Automation Conf., S. 694-697, 1998.
- [Năs83] C. Năstăsescu, C. Niță, M. Brandiburu, D. Joița, *Exerciții și probleme de algebră*, Editura Didactică și Pedagogică, București, 1983.
- [Nit04] Manfred Nitzsche, *Graphen für Einsteiger. Rund um das Haus von Nikolaus*, Vieweg Verlag, Wiesbaden, 2004.
- [Pei98] Heinz-Otto Peitgen, Hartmut Jürgens, Dietmar Saupe, *Bausteine der Chaos. Fraktale*, Rowohlt Taschenbuch Verlag GmbH, 1998
- [Ran97] Doina Rancea, *Limbajul Pascal. Manual clasa a IX-a*, Computer Libris Agora, Cluj, 1997.
- [Ran99] Doina Rancea, *Limbajul Pascal. Algoritmi fundamentali*, Computer Libris Agora, Cluj, 1999.
- [Ski03] Steven S. Skiena, Miguel A. Revilla, *Programming Challenges. The Programming Contest Training Manual*, Springer-Verlag, New York, 2003.
- [Sip97] Sipser, M., *Introduction to the Theory of Computation*, PWS-Kent, Belmont, California.
- [Sta86] D. Stanton, D. White, *Constructive Combinatorics*, Springer-Verlag, New-York, 1986.
- [Str66] Karl Strubecker, *Einführung in die höhere Mathematik, Band I: Grundlagen*, R. Oldenbourg, München-Wien, 1966.
- [Str67] Karl Strubecker, *Einführung in die höhere Mathematik, Band II: Differentialrechnung einer reellen Veränderlichen*, R. Oldenbourg, München-Wien, 1967.
- [Sun06] *Java™ Platform, Standard Edition 6 API Specification*, [Online Referenz: http://java.sun.com/javase/6/docs/api/](http://java.sun.com/javase/6/docs/api/), Sun Microsystems, Inc, 2006.
- [Tom81] Ioan Tomescu, *Probleme de combinatorică și teoria grafurilor*, Editura Didactică și Pedagogică, București, 1981.

Online Referenzen

<http://de.wikipedia.org/wiki/Hauptseite>
<http://acm.uva.es/problemset/>
<http://ceoi.inf.elte.hu/>
<http://mathworld.wolfram.com/>
<http://microscopy.fsu.edu/optics/timeline/people/>
<http://olympiads.win.tue.nl/ioi/>
<http://www.acm.org/>
<http://www.answers.com/>
<http://www.cut-the-knot.org>
<http://www.erichfried.de/>
<http://home.cogeco.ca/~ve3ll/jatutor0.htm>
<http://leepoint.net/notes-java/>
<http://www.math.utah.edu/mathcircle/notes/mladen2.pdf>
<http://www.matheboard.de>
<http://www.mathematische-basteleien.de>
<http://www.mathe-online.at/galerie.html>
http://www.nserc.gc.ca/news/2006/p060214_cook.htm
<http://www-i1.informatik.rwth-aachen.de/~algorithmus>
<http://www-math.mit.edu/~rstan/ec/catadd.pdf>
<http://java.sun.com/javase/6/docs/api/java/util/ArrayDeque.html>
<http://leepoint.net/notes-java/flow/loops/foreach.html>
<http://www.aifb.uni-karlsruhe.de/Lehre/Sommer2003/proksy/vlf/pks0309.pdf>

Stichwortverzeichnis

-
3-Schritte-Modell 232

Θ -Notation 12

Ω -Notation 13

π 114, 165

A

Abbruchbedingung 106

abs 174, 181, 190, 277

absteigende Teilfolge 240

Abstraktion 11

Abu Ja'far Muhammad ibn Musa al-Khwarizmi 1

Abwärtskompatibilität 36

Ackermann, Wilhelm 118

Ackermannfunktion 118

add 42, 54, 60, 91, 131, 134, 140, 147, 165

addFirst 91

addLast 90

adjazent 222

Adjazenzmatrix 8, 68, 222

ähnliche Summe 261

ALG *doTransformBase10ToP()*

ALG *rekonstruiereTeilfolge()* 29

ALG *write_Triangulation()* 288

ALG_Backtracking_Iterativ 176

ALG_Backtracking_Rekursiv 176

ALG_BINÄRE_SUCHE 5

ALG_COST_TRIANGULATIONS 287

ALG *DivideEtImpera* 146

ALG *doTransformBase10ToP* 121

ALG_EUKLID 2

ALG_EXAKT_DOPI 74

ALG_FFT 163

ALG_FibIt 233

ALG_FibRec 233

ALG *Greedy* 39

ALG_GREEDY_COMPACT 212

ALG_HUFFMAN 59

ALG_KRUSKAL 51

ALG_LCS 246

ALG_N_DAMEN 173

ALG_NACHFOLGER_BS 80

ALG_NACHFOLGER_PERM 78

ALG_OPTIM_COMPACT_REC 208

ALG_POTENZSUMMEN 316

ALG_RAN_DOP 73

ALG_RAN_DOPI 73

ALG_REVERSE 79

ALG_REVERSE2 80

ALG_SCHNELLES_POTENZIEREN 108

ALG_SIEB_ERATOSTHENES 4

ALG_SWAP 79

ALG_TIRAMISU 6

ALG_VERSCH_SCHACHTELN 28

ALG_WARSHALL_ARBITRAGE 307

algebraische (kartesische) Form 164

Algorithmen mit unterer Schranke 75

Algorithmik 10

al-jabr 1

Allgemeinheit 10

allocate 242, 272

Analyse der Komplexität 16

Anzahl aller Kombinationen 181

Anzahl aller n -Permutationen 178
 Anzahl aller Variationen 179
 Anzahl der Gebiete für schneidende Geraden 304
 Anzahl der schwarzen Pixel 158
 Anzahl der surjektiven Abbildungen 186
 Anzahl der Ziffern einer Zahl 107
 Anzahl der Züge, Türme von Hanoi 150
 Anzahl vollständiger Binärbäume 63
 append 35, 110, 160
 Arbitrage 305
 ArithmeticException 320
 arraycopy 80, 87
 ArrayDeque 83, 89
 ArrayList 41, 52, 54, 60, 91, 146, 154, 165
 Arrays 30, 33
 asList 33
assert 30, 36
 AssertionError 36
 Aufrundungsfunktion 238
 aufsteigende Teilfolge 27, 240
 Ausnahme 34
Ausnahmetyp 34
 Aussage 100
 Auswertung der Ergebnisse 94

B

Backtracking 169, 235
 Bauer, Wolf, Ziege und Kohlkopf 226
 Bauern auf dem Schachbrett 228
 Baum 49, 105, 157, 233
 bedingter Ausdruck?: 43, 62, 126, 147, 161, 209, 255
 Bedingung 25, 172
 Bellman, Richard 231
 Berliner Schachzeitung 169
 Bernoullische Ungleichung 105
 Beschreibung 10
 Bestimmung des minimalen Pfades 231
 Beweis 11, 100, 109, 111, 128, 240
 Beweis durch vollständige Induktion 101, 103
 Bezzel, Max 169
 BigDecimal 114

BigInteger 111, 239
 bijektive Funktion 18
 Binärbaum 56
 binäre Suche 4
 Binärwort 56, 65
 Binomialkoeffizienten 235, 239, 273, 318
 Binomischer Lehrsatz (Newton'sche Binomialformel) 102, 312
 Bit-Operatoren
 BitSet 84, 201, 218
 Blattknoten 56
 Boolean 62, 89
Bottom-up-Ansatz 234
bus-invert-Methode 66

C

$\mathbb{C}(\mathbb{Z})$ 164
carré latin 214
 Cäsar, Julius 145
 Catalan-Zahl 63, 224, 275, 286, 292
catch 31, 34
 Cauchy-Schwarz-Ungleichung (Schwarzsche Ungleichung) 102
 Character 217
 charAt 60, 160, 209, 302
 clear 87, 203, 210, 219
 clone 203
 close 31, 42, 45, 48, 54, 61, 94, 110
 Co-NP 19
 Code mit beliebiger Länge 56
 Code mit fester Länge 56
 Collatz-Funktion 106, 125
 Collections 40, 42
compaction_factor 211
 Comparable 29, 33, 42, 61
 compare 43
 compareTo 30, 33, 40, 43, 62
compound recursion 106, 118
 Cook, Stephen Arthur, Prof. Dr. 20
 currentTimeMillis 237

D

Darstellung 7, 32, 121

Data Ordering Problem 65
Datenabstraktion 32, 40, 163, 255, 319
Datenkanal 65
Decke aus Holz und Gold 167
default-Konstruktor 32
Dekodierung 56
delete 303
Deque 82, 89
Determinismus 10
Die Elemente 2
Die Zahl 4 108
digit 217
Dimension 131, 133, 138
direkte Rekursion 106, 130
direkter Beweis 100, 102, 109, 111, 128
Diskrete Fourier-Transformation (DFT) 162
Divide et Impera 145
Division mit Rest 111
Domino 253
Don Quijotes Windmühlen 168
DOP, DOPI 66, 71
Double 52
Double Ended Queue 82
drawLine 139
drawOval 133
drawRect 131, 134
Dreiecksungleichung 67
Drei-Gläser-Methode 78
Dynamische Programmierung 27, 231

E

Edit-Distanz 297
Einheitswurzeln 162
elementare Operation 11
Endlichkeit 10
Endlosschleife 106
enstrySet 55, 60, 92
Entscheidungsproblem 18, 72
Eratosthenes von Kyrene 3
Erfüllbarkeitsproblem 20
Ergebnisse DOP 95
Ergebnisse DOPI 96
Euklid 2

Euklidischer Abstand 288
Euklidischer Algorithmus 2, 320
Euler, Leonhard 164, 214
Exakt-Algorithmen (EX) 74
EXIT_ON_CLOSE 131, 134
Exklusiv-ODER 66

F

Fast Fourier Transformation (FFT)
fat recursion 106, 123
Fibonacci (Leonardo Pisano) 236
Fibonacci-Folge 105, 232
File 30, 35, 41, 44, 53, 60, 110
FileReader 35
fill 31, 259, 277
finally 31, 34, 45, ..., 110, ...
FloatBuffer 272
flush 238
foreach-Schleife 54, 62, 90, 166, 174, 187, 204, 210, 242, 260, 277
Formatter 123
Foto 158
Fotoproblem 191
Fraktal 135
Fraktal *Space-Filling* 141

G

gcd 111
Geheimnisprinzip 32
Genauigkeit 208, 317
Generator 135
Generierung aller *n*-BitStrings 80
Generierung aller *n*-Permutationen 78, 170
Generierung aller Variationen 179
geometrische Reihe 128
Gesamtanzahl der Transitionen 67
get 42, 52, 60, 91, 147, 165, ..., 210, 242
getContentPane 131, 134, 140
getDefaultToolkit 131
getFirst 91
getKey 55
getLast 91
getScreenSize 131

getSize 131, 133, 138, 140
 getValue 55, 61
 Gewicht eines Graphen 49
 Gewicht eines Huffman-Baumes 57
 Gewichtsfunktion eines Graphen 49
 Glaspypamide im Louvre, Paris 38
 Gleichung 102, 123, 225, 281
 Graffiti in Benalmadena 310
 Graph 9, 18, 48, 75, 233
 Graphenfärbungsproblem 20
 Graphenisomorphismusproblem 20
 Graphics 131, 133, 138
Greedy 39, 235
Greedy-Min-Algorithmen (GM) 74
Greedy-Min-Simplified-Algorithmen (GMS) 74
 Gregory, James 114
 Grenzwert 14
 größter gemeinsamer Teiler 2, 146, 320

H

Hamiltonkreis 19
 Hamming-Distanz 66, 298
 Handlungsreisendenproblem 18, 39
 HashSet 201
 hasNext 55, 60, 219
 hasNextDouble 152, 166
 hasNextFloat 42
 hasNextInt 30, 35, 110, 116, 124, 147, 154
 hasNextLong 119, 121, 126, 284
 hasNextPrevious 61, 249
 Haus des Nikolaus 221
 height 131, 133, 138
 Hoare, C. A. R. 153
 Holzskulptur an einem Stuhl 230
 Huffman, Albert 56
 Huffman-Baum 56
 Huffman-Kodierung 56

I

imaginäre Einheit 1
 Imaginärteil 162
 indirekte Rekursion 106, 133

Induktionsanfang 100
 Induktionsschluss 100
 Induktionsvoraussetzung 100
 Initiator 135
 insert 31, 35, 303
 IntBuffer 241
 Integer 52, 60, 89, ..., 154
 Integer.MAX_VALUE 47, 258, 276
 Integral 150
 Interface 33
 intersects 204
 intValue 53, 90
 Invertierung 66
 IOException 41, 44, 47, 53, 60, 93, 111
 isDigit 217
 isEmpty 55, 60, 204
 Isomorphismus 18
 isProbablePrime 111
 Iterator 54, 60, 91
 iterator 55, 60, 91

J

JFrame 131, 134, 139
 JPanel 131, 133, 138

K

Kapelle in Dorohoi 24
 Kapselung 32
 Kartenfärbungsproblem 7, 43, 224
 kartesische (algebraische) Form 164
 kartesisches Produkt 225
k-Bonacci-Folge 238
 key 53, 93
 Kirche auf Norderney 63
 Klammerung von n Klammern-Paaren 224
 Klassifizierung der Probleme 18
 kleinstes gemeinsames Vielfaches 321
 Kloster Sucevița 322
 Koch, Helge von 135
 Koch'sche Schneeflockenkurve 135
 Kodierung 56
 Kombinationen 181
 Kommandozeilen-Parameter 93

Kompaktierungsmenge 206
kompatible Tests 206
kompatible Zeichen 206
komplexe Zahlen 162
Komplexität 10, 12, 41, 59, 74, 153, 155, 208, 234, 317
konjunktive Normalform (KNF) 21
konstante Kategorie 12
Konstruktoren 29, 32
konvexe Hülle 15
Koordinate 127, 136
Korrektheit 10
Kreis, in Graph 19
Kruskal, Joseph Bernard 51
Kruskal-Algorithmus 48, 83

L

Labyrinth 228
längste absteigende Teilfolge 240
längste aufsteigende Teilfolge 27, 240
längste gemeinsame Teilfolge 245
Laufzeit 14
Leibnitz, Gottfried Wilhelm 114
length 85, 160, 209, 277
Levenshtein, Vladimir 298
Levenshtein-Distanz 298
lexikographisch 27, 78, 179, 221, 288
Liber Abaci 236
lineare Rekursion 106, 115
List 41, 52, 60, 91, 147, 154
ListIterator 54, 61, 91, 249
listIterator 55, 61, 92, 249
Locale 41, 53, 124
logischer Ausdruck 21, 82
Long 120
Long.MAX_VALUE 295
lower bound 76
Lucas, Edouard 148
Lucas-Zahlen 238
Ludwig XI., König 145

M

Machin, John 115

magische Quadrate 169, 177
Manna-Pnuelli-Funktion 120
Map Colouring Problem 7
Map.Entry 54, 60, 91
Math.abs 174, 181, 190, 277
Math.cos 165
Math.max 54, 93
Math.min 53, 93, 131, 133
Math.PI 165
Math.pow 289
Math.random 85
Math.round 238
Math.sin 165
Math.sqrt 138, 289
max 54, 93
maximal aufsteigende Teilfolge 26
Mehrfachvererbung 33
Memoization 232
Mengenüberdeckungsproblem 20
MergeSort 146, 155
Metrik 67, 297
Metrik-Bedingungen 298
min 53, 131, 133
minimale Triangulierung 286
minimaler Spannbaum 49, 76
Mittelpunkt eines Segments 136
Modell der flachen Erde 12
modPow 111
Multigraph 75
multimap-Konzept 52, 83
Multiplikation einer Matrizenfolge 291
Multiplikationsreihe 293

N

Nachfolger-Algorithmus 78
natürliche Ordnung 33, 41
Nauck, Franz, Dr. 169
n-Damen-Problem 169
n-dimensionale Schachtel 25
next 55, 60, 219
nextBoolean 85
nextClearBit 87
nextDouble 52, 124, 152, 166
nextFloat 42, 308

nextInt 30, 35, 44, 47, 60, 110, 116, 122,
139, 147, 154, 174, 186, 190, 192
nextLong 108, 113, 119, 122, 126, 284
nicht monotone (offene) Rekursion 106,
125
NP-hard 21
NP-vollständig 19, 71
 n -Türme-Problem 178
NullPointerException 33

O

obere Schranke 13
offene (nicht monotone) Rekursion 106,
125
O-Notation 13
Optimalität 15, 231
Optimalitätsprinzip 231
Optimierungsproblem 18
Orangensport 196
Österreichkarte 8
overloading 32
overriding 32

P

pack 131, 134, 140
paint 131, 133, 138
parseInt 93
Partitionen einer natürlichen Zahl 182
passende Wörter 244
passt-Bedingung 25
Peano, Giuseppe 99
Peanosche Axiome 99
Permutationen 25, 67, 77
Pi (π) 114, 165
Pisano, Leonardo (Fibonacci) 236
Pivot-Element 153
point-region 157
Polygon 286
Polymorphismus 32
Polynom 163, 282, 310
position 242
Potenzieren, rekursives 107
Potenzieren, schnelles 108

Potenzsummen 311
pow 289
Präfixcode 56
pre-order Darstellung 158
previous 249
Primfaktorzerlegung 321
Primzahl 3, 101
print 31, 36, 41, 42, 45, 62
printf 48, 54, 94, 116
println 31, 36, 42, 48, 54, 110, 113
printStackTrace 31, 34
PrintStream 45, 47, 60, 93, 110, 113
PrintWriter 30, 35
Problem der Türme auf den ersten m Rei-
hen 179
Problem der aufsteigenden Türme auf den
ersten m Reihen 180
Produktmatrix 292
put 52, 54, 60, 91, 242, 272

Q

Quad-Bäume 157
Quadracci-Folge 238
Quadrant 127, 158
Quersumme einer Zahl 106
QuickSort 146, 152

R

RAM (Random Access Machine) 11
Random 84
random 85
raumfüllende Fraktale 141
Realteil 162
Rechtecksregel 152
recurrere 105
Reduktion 15
Reelle Zeit 17
Rekursion 99
Rekursionsformel 268, 313
Rekursionsgleichung 268, 313
rekursive Datenstrukturen 105
rekursive Zahlenumwandlung 120
rekursives Potenzieren 107

remove 55, 61, 177, 210
replace 302
Rest großer Potenzen 111
Rezept 6
round 238
Rucksackproblem 20, 40
Rumänienkarte 22

S

SAT 20
Satz (Erdős, Szekeres) 240
Satz von Vieta (Vieta Relationen) 123, 282
Scanner 30, 35, 41, 44, 47, 53, 60, 108
Scheduling-Problem 10, 20
Schluss von n auf $n+1$ 99
schnelles Potenzieren 108
Schnittpunkt einer Menge von n Strecken 15
Schotten auf dem Oktoberfest 266
Schwäne auf der Isar 143
Schwarzsche Ungleichung (Cauchy-Schwarz-Ungleichung) 102
Set 201
set 87, 203
setDefaultCloseOperation 131, 134
setLocation 134, 140
setPreferredSize 131, 134, 140
setVisible 131, 134, 140
Sieb des Eratosthenes 3
Sierpinski-Dreieck 140
size 60, 90, 147, 255
Skulptur auf einem Brunnen in Bern 334
sort 30, 33, 40, 42
Sortieren 27
space-filling curves 141
Spannbaum 49
Spiegelung einer Zahl 106
Springer auf dem Schachbrett 45, 188, 275
sqrt 138, 289
statische lokale Klasse 33, 255, 289
Statue in Bremen 98
Strandkörbe auf Norderney 309
String 41, 44, 46, 53, 60, 110
StringBuffer 35

StringBuilder 31, 35, 110, 160, 301
substring 302
Sudoku 214
Summe zweier Wurzeln 123
Summen von Produkten 280
Summenformel 16, 100, 104, 177
surjektive Funktion 185
swap 78
symbolische Kodierung 7, 8
Symmetrie 67, 129
System.in 174, 183, 186, 190, 269

T

Teilbarkeit 101, 105, 199
Teilbarkeit durch eine Primzahl 101
Teilbaum 159
Teilblock 214
Teile und Herrsche 145
Teiler 101, 105, 199
Teilfolge 245
Teilproblem 145, 231
Testmusterkompaktierung 205
Throwable 31, 34
Tiramisu 6
Toolkit 131, 134
Top-Down-Ansatz 234
toString 40, 43, 121, 166, 255
Transformation 7
Transition 66
Trapezregel 150
Travelling Salesman Problem (TSP) 18
TreeMap 52, 60, 91
Triangulierung eines konvexen Polygons 286
Tribonacci-Folge 238
try 30, 34, ..., 110
Turm der Wallfahrtskapelle St. Bartholomä 64
Türme von Hanoi 105, 148

U

überladene Methoden 32
überlagerte Methoden 32

Überlappung der Probleme 232
 ungünstigster Fall 14
 untere Schranke 13
 Untergraph 48
 unzusammenhängend, Graph 49
 useLocale 41, 53, 124, 152

V

value 53, 93
 Vereinfachen 320
 Verpackungsproblem 20
 verschachtelte Rekursion 106, 118
 verschachtelte Schachteln 25
 Verschmelzungsoperation 57
 versteckte Basen 229
 Verteilung der Geschenke 258
 verzweigte Rekursion 106, 123
 Vielfache 321
 Vielfache, das kleinste V. 321
 Vieta Relationen (Satz von Vieta) 123, 282
 vollständige Induktion 99
 vollständiger Binärbaum 56
 vollständiger Graph 75

W

Wachstum von $O(g(n))$ 17
 Wahrscheinlichkeit 268
 Wald 49
Walking Man an der Leopoldstraße 144
 Warnsdorff-Regel 46
 Widerspruch 26, 240
 width 131, 133, 138
 WindowConstants 131, 134
worst-case complexity 14
 Wrapper-Klasse 120
 Wurf einer Münze 266
 Wurzelsatz von Vieta 123, 282

Z

Zahlen-Dreieck 249
 Zahlensystem 120
 Zahlenumwandlung, rekursiv 120

Zeichen 56
 Zeichenmenge 57
 Zeiteinheit 11
 Zerlegung 198
 Zufällige-Lösung-Algorithmen 73
 zusammenhängend, Graph 49
 Zutaten 6
 Zuweisung 68



Skulptur auf einem Brunnen in Bern