

(falso rosto provisório)

Algoritmos



Preencha a **ficha de cadastro** no final deste livro
e receba gratuitamente informações
sobre os lançamentos e as promoções da
Editora Campus/Elsevier.

Consulte também nosso catálogo
completo e últimos lançamentos em
www.elsevier.com.br

(falso provisório)

Algoritmos

Do original: *Introduction to Algorithms*, 3rd edition
Copyright © 2009 by The MIT Press.

© 2012, Elsevier Editora Ltda.

Todos os direitos reservados e protegidos pela Lei no 9.610, de 19/02/1998.
Nenhuma parte deste livro, sem autorização prévia por escrito da editora, poderá ser reproduzida ou transmitida sejam
quais forem os meios empregados: eletrônicos, mecânicos, fotográficos, gravação ou quaisquer outros.

Coordenação de produção: Silvia Lima
Copidesque: Ivone Teixeira
Revisão: Globaltec Editorial & Marketing
Editoração Eletrônica: Globaltec Editorial & Marketing

Elsevier Editora Ltda.
Conhecimento sem Fronteiras
Rua Sete de Setembro, 111 – 16º andar
20050-006 – Centro – Rio de Janeiro – RJ – Brasil

Rua Quintana, 753 – 8º andar
04569-011 – Brooklin – São Paulo – SP

Serviço de Atendimento ao Cliente
0800-0265340
sac@elsevier.com.br

ISBN original 978-0-262-03384-8
ISBN 978-85-352-3699-6

Nota: Muito zelo e técnica foram empregados na edição desta obra. No entanto, podem ocorrer erros de digitação,
impressão ou dúvida conceitual. Em qualquer das hipóteses, solicitamos a comunicação ao nosso Serviço de Atendi-
mento ao Cliente, para que possamos esclarecer ou encaminhar a questão.
Nem a editora nem o autor assumem qualquer responsabilidade por eventuais danos ou perdas a pessoas ou bens,
originados do uso desta publicação.

CIP-BRASIL. CATALOGAÇÃO-NA-FONTE
SINDICATO NACIONAL DOS EDITORES DE LIVROS, RJ

A385
3 ed.

Algoritmos / Thomas H. Cormen... [et al.] ; [tradução
Arlete Simille Marques]. - Rio de Janeiro : Elsevier,
2012. il.

Tradução de: *Introduction to algorithms*, 3rd ed.

Apêndices
nclui bibliografia e índice
ISBN 978-85-352-3699-6

1. Programação (Computadores).
2. Algoritmos de computador.
3. Estruturas de dados (Computadores). I. Cormen,
Thomas H.

11-3715

CDD: 005.1
CDU: 004.4



PREFÁCIO

Antes de existirem computadores, havia algoritmos. Mas, agora que temos computadores, há ainda mais algoritmos, e os algoritmos estão no coração da computação.

Este livro é uma introdução abrangente ao moderno estudo de algoritmos para computadores. Apresenta muitos algoritmos e os examina com considerável profundidade, embora torne seu projeto e sua análise acessíveis a leitores de todos os níveis. Tentamos manter as explicações em um nível elementar sem sacrificar a profundidade do enfoque ou o rigor matemático.

Cada capítulo apresenta um algoritmo, uma técnica de projeto, uma área de aplicação ou um tópico relacionado. Algoritmos são descritos em linguagem comum e em pseudocódigo projetado para ser fácil de ler por qualquer pessoa que tenha estudado um pouco de programação. O livro contém 244 figuras — algumas com várias partes — que ilustram como os algoritmos funcionam. Visto que enfatizamos a *eficiência* como um critério de projeto, incluímos análises cuidadosas dos tempos de execução de todos os nossos algoritmos.

O texto foi planejado primariamente para uso em cursos de graduação e pós-graduação em algoritmos ou estruturas de dados. Como discute questões de engenharia relacionadas ao projeto de algoritmos, bem como aspectos matemáticos, é igualmente adequado para profissionais técnicos autodidatas.

Nesta terceira edição, mais uma vez atualizamos o livro inteiro. As mudanças são abrangentes e incluem novos capítulos, revisão de pseudocódigos e um estilo de redação mais ativo.

Ao professor

Este livro foi projetado para ser ao mesmo tempo versátil e completo. Você descobrirá sua utilidade para uma variedade de cursos, desde graduação em estruturas de dados até pós-graduação em algoritmos. Como oferecemos uma quantidade consideravelmente maior de material da que poderia ser abordada em um curso típico de um período, você pode considerar o livro como um bufê de vários pratos do qual pode selecionar e extrair o material que melhor atender ao curso que deseja ministrar.

Você verá que é fácil organizar seu curso usando apenas os capítulos de que precisar. Os capítulos são relativamente autônomos, de modo que você não precisa se preocupar com uma dependência inesperada e desnecessária de um capítulo em relação a outro. Cada capítulo apresenta primeiro o material mais fácil e, em seguida, o material mais difícil; os limites das seções são pontos de parada naturais. Em cursos de graduação você poderia utilizar somente as primeiras seções de um capítulo; em cursos de pós-graduação, o capítulo inteiro.

Incluímos 957 exercícios e 158 problemas. Cada seção termina com exercícios, e cada capítulo com problemas. Em geral, os exercícios são perguntas curtas que testam o domínio básico do assunto. Alguns são exercícios simples de autoavaliação, enquanto outros são mais substanciais e apropriados para o aluno resolver com mais tempo e calma. Os problemas são estudos de casos mais elaborados que, muitas vezes, apresentam novo material; frequentemente consistem em várias perguntas que conduzem o aluno pelas etapas exigidas para chegar a uma solução.

Ao contrário da prática que adotamos em edições anteriores deste livro, nesta apresentamos soluções para alguns problemas, mas de modo algum para todos os problemas e exercícios. Essas soluções estão disponíveis no site da editora: www.elsevier.com.br/cormen. Seria interessante você visitar esse site para verificar se ele contém a solução para um exercício ou problema que

planeja apresentar a seus alunos. Esperamos que o conjunto de soluções reunidos no site aumente ao longo do tempo, de modo que você deve visitá-lo toda vez que ministrar o curso.

Assinalamos com asteriscos (*) as seções e os exercícios mais adequados para alunos de pós-graduação do que de graduação. Uma seção marcada com asterisco não é necessariamente mais difícil que outra que não tenha asterisco, mas pode exigir o entendimento de matemática mais avançada. De modo semelhante, exercícios assinalados por asteriscos podem exigir um conhecimento mais avançado ou criatividade acima da média.

Ao aluno

Esperamos que este livro didático proporcione uma introdução agradável à área de algoritmos. Tentamos tornar cada algoritmo acessível e interessante. Para ajudá-lo quando encontrar algoritmos pouco familiares ou difíceis, descrevemos cada um deles etapa por etapa. Também apresentamos explicações cuidadosas dos fundamentos matemáticos necessários para entender a análise dos algoritmos. Se você já tiver alguma familiaridade com um tópico, perceberá que os capítulos estão organizados de modo que você possa apenas ler rapidamente as seções introdutórias e passar rapidamente para o material mais avançado.

Este é um livro extenso, e sua turma provavelmente só examinará uma parte de seu material. Porém, procuramos torná-lo útil para você, agora como livro didático, e também mais tarde, em sua carreira, como um guia de referência de matemática ou um manual de engenharia.

Quais são os pré-requisitos para a leitura deste livro?

- Você deve ter alguma experiência em programação. Em particular, deve entender procedimentos recursivos e estruturas de dados simples como arranjos e listas ligadas.
- Você deve ter alguma facilidade com demonstrações matemáticas, em especial por indução. Algumas partes do livro dependem de algum conhecimento de cálculo diferencial elementar. Além disso, as Partes I e VIII deste livro ensinam todas as técnicas matemáticas de que você necessitará.

Apresentamos soluções para alguns deles, que estão disponíveis em: www.elsevier.com.br/cormen. Você pode consultar o site e comparar suas soluções com as nossas.

Ao profissional

A ampla variedade de tópicos neste livro faz dele um excelente manual sobre algoritmos. Como cada capítulo é relativamente autônomo, você pode se concentrar nos tópicos que mais o interessem.

A maioria dos algoritmos que discutimos tem grande utilidade prática. Portanto, abordamos questões de implementação e outras questões de engenharia. Muitas vezes damos alternativas práticas para os poucos algoritmos cujo interesse é primordialmente teórico.

Se desejar implementar qualquer dos algoritmos, verá que a tradução do nosso pseudocódigo para a sua linguagem de programação favorita é uma tarefa razoavelmente direta. Projetamos o pseudocódigo para apresentar cada algoritmo de forma clara e sucinta. Consequentemente, não abordamos tratamento de erros e outras questões de engenharia de software que exigem características específicas do seu ambiente de programação. Tentamos apresentar cada algoritmo de modo simples e direto sem permitir que as idiossincrasias de determinada linguagem de programação obscureçam sua essência.

Se você estiver usando este livro por conta própria, sem seguir um curso, pode ser que você não consiga ter acesso às soluções de problemas e exercícios. No nosso site: <http://mitpress.mit.edu/algorithms/> há links para as respostas de alguns problemas e exercícios para que você possa verificar suas respostas. Por favor, não nos mande suas respostas.

Aos nossos colegas

Apresentamos indicações e bibliografia extensivas para a literatura corrente. Cada capítulo termina com um conjunto de notas do capítulo que dão detalhes e referências históricas. Contudo, as notas dos capítulos não oferecem uma referência completa para toda a área de algoritmos. Embora talvez seja difícil de acreditar, dado o tamanho deste livro, restrições de espaço nos impediram de incluir muitos algoritmos interessantes.

Apesar dos inúmeros pedidos dos alunos, preferimos manter a nossa política de não apresentar referências para as soluções de problemas e exercícios, para evitar que eles cedam à tentação de consultar uma solução dada em vez de determiná-la.

Mudanças na terceira edição

O que mudou entre a segunda e a terceira edições deste livro? Sobre a magnitude das mudanças entre essas duas edições e entre a primeira e a segunda, dizemos o mesmo que dissemos na segunda edição: dependendo do ponto de vista de cada leitor, a mudança pode não ser muito grande ou pode ser bem grande.

Um rápido exame do sumário mostra que a maior parte dos capítulos e seções da segunda edição aparecem na terceira edição. Eliminamos dois capítulos e uma seção, mas acrescentamos três novos capítulos e duas novas seções além desses novos capítulos.

Mantivemos a organização híbrida das duas primeiras edições. Em vez de organizar os capítulos só por domínios de problemas ou só de acordo com técnicas, este livro tem elementos de ambos. Contém capítulos baseados em técnicas de divisão e conquista, programação dinâmica, algoritmos gulosos, análise amortizada, NP-completude e algoritmos de aproximação. Mas também traz partes inteiras dedicadas a ordenação, estruturas de dados para conjuntos dinâmicos e algoritmos para problemas de grafos. Entendemos que, embora você precise saber como aplicar técnicas para projetar e analisar algoritmos, os problemas raramente informam de antemão quais técnicas são as mais adequadas para resolvê-los.

Damos a seguir um resumo das mudanças mais significativas para a terceira edição:

- Acrescentamos novos capítulos sobre árvores de van Emde Boas e algoritmos multithread, e agora a parte de fundamentos do material sobre matrizes ocupa um dos apêndices.
- Revisamos o capítulo sobre recorrências de modo a dar um tratamento mais abrangente à técnica de divisão e conquista, e suas duas primeiras seções aplicam essa técnica para resolver dois problemas. A segunda seção desse capítulo apresenta o algoritmo de Strassen para multiplicação de matrizes, que transferimos do capítulo sobre operações com matrizes.
- Eliminamos dois capítulos que raramente eram ensinados: heaps binomiais e redes de ordenação. Uma ideia fundamental no capítulo sobre redes de ordenação, o princípio 0-1, aparece, nesta edição, dentro do Problema 8-7 como o lema de ordenação 0-1 para algoritmos de comparação e permutação. O tratamento dos heaps de Fibonacci não depende mais do heaps binomiais como precursor.
- Revisamos o tratamento de programação dinâmica e algoritmos gulosos. Agora a programação dinâmica começa com um problema mais interessante, corte de hastes de aço, do que o problema de programação de linha de montagem na segunda edição. Além disso, enfatizamos a memoização com um pouco mais de intensidade do que o fizemos na segunda edição, e apresentamos a noção do grafo de subproblema como um modo de entender o tempo de execução de um algoritmo de programação dinâmica. Em nosso exemplo de abertura sobre algoritmos gulosos, o problema de seleção de atividades, chegamos ao algoritmo guloso mais diretamente do que o fizemos na segunda edição.
- O modo de eliminar um nó em árvores de busca binária (que inclui árvores rubronegras) agora garante que o nó requisitado para eliminação seja o nó realmente eliminado. Nas duas primeiras edições, em certos casos, algum outro nó seria eliminado e seu conteúdo seria transferido para o nó enviado para o procedimento de eliminação. Com o nosso novo

modo de eliminar nós, se outros componentes de um programa mantiverem ponteiros para os nós na árvore, não terminarão erroneamente com ponteiros inativos para nós que já foram eliminados.

- O material sobre redes de fluxo agora baseia os fluxos inteiramente em arestas. Essa abordagem é mais intuitiva do que o fluxo em rede usado nas duas primeiras edições.
- Como o material sobre fundamentos de matrizes e o algoritmo de Strassen passou para outros capítulos, o capítulo sobre operações com matrizes é menor do que o da segunda edição.
- Modificamos o tratamento do algoritmo de correspondência de cadeias de Knuth-Morris-Pratt.
- Corrigimos vários erros. A maioria deles aparece na errata da segunda edição publicada em nosso site, mas outros, não.
- Atendendo a muitos pedidos, mudamos a sintaxe (até certo ponto) do nosso pseudocódigo. Agora usamos “=” para indicar atribuição e “==” para testar igualdade, exatamente como fazem as linguagens C, C++, Java e Python. De modo semelhante, eliminamos as palavras-chave **do** e **then** e adotamos “//” como símbolo para comentários de final de linha. Agora também usamos a notação de ponto para indicar atributos de objetos. Nosso pseudocódigo continua sendo orientado a procedimento e não a objeto. Em outras palavras, em vez de executar métodos em objetos, simplesmente chamamos procedimentos e passamos objetos como parâmetros.
- Adicionamos 100 novos exercícios e 28 novos problemas. Além disso, atualizamos muitas citações bibliográficas e acrescentamos várias outras novas.
- Finalmente, repassamos o livro inteiro e reescrevemos sentenças, parágrafos e seções para tornar a linguagem mais clara e mais ativa.

Agradecimentos para a terceira edição

Estamos trabalhando com a MIT Press há mais de duas décadas, e que maravilhosa relação é a nossa! Agradecemos a Ellen Faran, Bob Prior, Ada Brunstein e Mary Reilly por sua ajuda e apoio.

Estávamos em lugares diferentes enquanto produzíamos a terceira edição, trabalhando no Dartmouth College Department of Computer Science, no MIT Computer

Science and Artificial Intelligence Laboratory e no Columbia University Department of Industrial Engineering and Operations Research. Agradecemos a nossas respectivas universidades e colegas por proporcionarem tais ambientes de suporte tão estimulantes.

Mais uma vez, Julie Sussman, P.P.A., nos salvou como editora técnica. Mais de uma vez ficamos pasmados com os erros que deixamos passar, mas que Julie percebeu. Ela também nos ajudou a melhorar a apresentação em vários lugares do livro. Se houvesse uma Calçada da Fama para editores técnicos, Julie seria, com certeza, a nossa candidata número um. Ela é nada menos que fenomenal. Obrigado, obrigado, obrigado, Julie! Priya Natarajan também descobriu alguns erros que pudemos corrigir antes da impressão deste livro. Quaisquer erros que restarem (e não temos dúvidas de que ainda há alguns) são da responsabilidade dos autores (e provavelmente foram inseridos depois que Julie leu o material).

O tratamento para as árvores de van Emde Boas baseia-se nas notas de Erik Demaine que, por sua vez, foram influenciadas por Michael Bender. Incorporamos também ideias de Javed Aslam, Bradley Kuszmaul e Hui Zha nesta edição.

O capítulo sobre multithreading foi baseado em notas escritas originalmente em conjunto com Harald Prokop. O material foi influenciado pelo de vários outros que trabalharam no projeto Cilk no MIT, entre eles Bradley Kuszmaul e Matteo Frigo. O projeto do pseudocódigo multithread foi inspirado pelas extensões Cilk do MIT para C e por extensões da Cilk++ para C++ pela Cilk Arts.

Agradecemos também aos muitos leitores da primeira e segunda edições que indicaram erros ou apresentaram sugestões para melhorar este livro. Corrigimos todos os erros de boa-fé indicados e incorporamos o máximo de sugestões possível. Ficamos contentes com o grande crescimento do número de tais contribuintes, mas tristes porque tornou-se impraticável nomear todos eles.

Finalmente, agradecemos a nossas esposas — Nicole Cormen, Wendy Leiserson, Gail Rivest e Rebecca Ivry — e a nossos filhos — Ricky, Will, Debby e Katie Leiserson; Alex e Christopher Rivest; e Molly, Noah e Benjamin Stein — por seu amor e apoio durante a preparação deste livro. A paciência e o estímulo de nossas famílias tornou esse projeto possível. Dedicamos este livro afetuosamente a elas.

THOMAS H. CORMEN
CHARLES E. LEISERSON
RONALD L. RIVEST
CLIFFORD STEIN

*Lebanon, New Hampshire
Cambridge, Massachusetts
Cambridge, Massachusetts
New York, New York*

Fevereiro de 2009

¹Investigamos vários programas de desenho que são executados sob Mac OS X, mas todos apresentaram deficiências significativas em comparação com o MacDraw Pro. Por algum tempo tentamos produzir as ilustrações para este livro com um programa de desenho diferente, muito conhecido. Constatamos que demorava no mínimo cinco vezes mais tempo para produzir cada ilustração do que com o MacDraw Pro, e as ilustrações resultantes não eram tão boas. Daí a decisão de voltar para o MacDraw Pro executado em Macintoshes mais antigos.

SUMÁRIO

Prefácio	xi
Parte I Fundamentos	
Introdução	1
1 O papel dos algoritmos na computação.....	3
1.1 Algoritmos.....	3
1.2 Algoritmos como tecnologia.....	7
2 Dando a partida.....	11
2.1 Ordenação por inserção	11
2.2 Análise de algoritmos.....	16
2.3 Projeto de algoritmos.....	21
3 Crescimento de funções.....	32
3.1 Notação assintótica	32
3.2 Notações padrão e funções comuns	39
4 Divisão e conquista	48
4.1 O problema do subarranjo máximo.....	50
4.2 Algoritmo de Strassen para multiplicação de matrizes.....	56
4.3 Método de substituição para resolver recorrências	61
4.4 Método da árvore de recursão para resolver recorrências.....	65
4.5 Método mestre para resolver recorrências	69
4.6 Prova do teorema mestre	72
5 Análise probabilística e algoritmos aleatorizados	84
5.1 O problema da contratação.....	84
5.2 Variáveis aleatórias indicadoras	87
5.3 Algoritmos aleatorizados.....	90
5.4 Análise probabilística e usos adicionais de variáveis aleatórias indicadoras	95
Parte II Ordenação e estatísticas de ordem	
Introdução	107
6 Ordenação por heap.....	110
6.1 Heaps	110
6.2 Manutenção da propriedade de heap	112
6.3 Construção de um heap	114
6.4 O algoritmo de ordenação por heap.....	116
6.5 Filas de prioridades	118
7 Quicksort	123
7.1 Descrição do quicksort	123
7.2 O desempenho do quicksort.....	126
7.3 Uma versão aleatorizada do quicksort	129
7.4 Análise do quicksort	130
8 Ordenação em tempo linear.....	138
8.1 Limites inferiores para ordenação	138
8.2 Ordenação por contagem.....	140
8.3 Ordenação digital.....	142
8.4 Ordenação por balde	145
9 Medianas e estatísticas de ordem	155
9.1 Mínimo e máximo	155
9.2 Seleção em tempo linear esperado.....	156
9.3 Seleção em tempo linear do pior caso	159

Parte III Estruturas de dados	
Introdução	165
10 Estruturas de dados elementares	168
10.1 Pilhas e filas	168
10.2 Listas ligadas	171
10.3 Implementação de ponteiros e objetos	175
10.4 Representação de árvores enraizadas	178
11 Tabelas de espalhamento	184
11.1 Tabelas de endereço direto	184
11.2 Tabelas de espalhamento	186
11.3 Funções hash	190
11.4 Endereçamento aberto	196
11.5 Hashing perfeito	202
12 Árvores de busca binária	209
12.1 O que é uma árvore de busca binária?	209
12.2 Consultas em uma árvore de busca binária	211
12.3 Inserção e eliminação	215
12.4 Árvores de busca binária construídas aleatoriamente	219
13 Árvores vermelho-preto	226
13.1 Propriedades de árvores vermelho-preto	226
13.2 Rotações	229
13.3 Inserção	230
13.4 Eliminação	236
14 Aumentando estruturas de dados	248
14.1 Estatísticas de ordem dinâmicas	248
14.2 Como aumentar uma estrutura de dados	252
14.3 Árvores de intervalos	255
Parte IV Técnicas avançadas de projeto e análise	
Introdução	261
15 Programação dinâmica	262
15.1 Corte de hastes	263
15.2 Multiplicação de cadeias de matrizes	270
15.3 Elementos de programação dinâmica	276
15.4 Subsequência comum mais longa	285
15.5 Árvores de busca binária ótimas	290
16 Algoritmos gulosos	302
16.1 Um problema de seleção de atividades	302
16.2 Elementos da estratégia gulosa	308
16.3 Códigos de Huffman	312
16.4 Matroides e métodos gulosos	319
16.5 Um problema de programação de tarefas como um matroide	324
17 Análise amortizada	330
17.1 Análise agregada	330
17.2 O método de contabilidade	333
17.3 O método do potencial	335
17.4 Tabelas dinâmicas	338
Parte V Estruturas de dados avançadas	
Introdução	350
18 Árvores B	352
18.1 Definição de B-árvores	355
18.2 Operações básicas em B-árvores	357
18.3 Eliminar uma chave em uma B-árvore	363
19 Heaps de Fibonacci	367
19.1 Estrutura de heaps de Fibonacci	369
19.2 Operações de heaps intercaláveis	370
19.3 Decrementar uma chave e eliminar um nó	376
19.4 Limitando o grau máximo	379

20	Árvores de van Emde Boas	386
20.1	Abordagens preliminares.....	387
20.2	Uma estrutura recursiva	389
20.3	A árvore de van Emde Boas	396
21	Estruturas de dados para conjuntos disjuntos	408
21.1	Operações em conjuntos disjuntos	408
21.2	Representação de conjuntos disjuntos por listas ligadas	410
21.3	Florestas de conjuntos disjuntos	413
21.4	Análise da união pelo posto com compressão de caminho	416
Parte VI Algoritmos de grafos		
Introdução		427
22	Algoritmos elementares em grafos	429
22.1	Representações de grafos.....	429
22.2	Busca em largura	432
22.3	Busca em profundidade	439
22.4	Ordenação topológica.....	446
22.5	Componentes fortemente conexas.....	448
23	Árvores geradoras mínimas	454
23.1	Desenvolvendo uma árvore geradora mínima	455
23.2	Algoritmos de Kruskal e Prim.....	459
24	Caminhos mínimos de fonte única	468
24.1	O algoritmo de Bellman-Ford.....	474
24.2	Caminhos mínimos de fonte única em grafos acíclicos dirigidos.....	477
24.3	Algoritmo de Dijkstra	479
24.4	Restrições de diferença e caminhos mínimos	484
24.5	Provas de propriedades de caminhos mínimos.....	488
25	Caminhos mínimos entre todos os pares	498
25.1	Caminhos mínimos e multiplicação de matrizes.....	499
25.2	O algoritmo de Floyd-Warshall	504
25.3	Algoritmo de Johnson para grafos esparsos	510
26	Fluxo máximo	516
26.1	Redes de fluxo.....	516
26.2	O método Ford-Fulkerson	520
26.3	Emparelhamento máximo em grafo bipartido	533
26.4	Algoritmos push-relabel	536
26.5	O algoritmo relabel-to-front	545
Parte VII Tópicos selecionados		
Introdução		558
27	Algoritmos multithread	560
27.1	Os fundamentos do multithread dinâmico	562
27.2	Multiplicação multithread de matrizes.....	575
27.3	Ordenação por intercalação multithread.....	578
28	Operações com matrizes	590
28.1	Resolver sistemas de equações lineares	590
28.2	Inversão de matrizes.....	601
28.3	Matrizes simétricas positivas definidas e aproximação de mínimos quadrados.....	604
29	Programação linear	613
29.1	Forma-padrão e forma de folgas.....	618
29.2	Formular problemas como programas lineares.....	624
29.3	O algoritmo simplex	628
29.4	Dualidade	639
29.5	A solução básica viável inicial.....	643
30	Polinômios e a FFT	652
30.1	Representação de polinômios.....	654
30.2	DFT e FFT	658
30.3	Implementações eficientes de FFT	664

31	Algoritmos da teoria dos números	672
31.1	Noções da teoria elementar dos números	673
31.2	Máximo divisor comum	677
31.3	Aritmética modular.....	681
31.4	Solução de equações lineares modulares.....	687
31.5	O teorema chinês do resto.....	690
31.6	Potências de um elemento	693
31.7	O sistema de criptografia de chave pública RSA.....	696
31.8	Teste de primalidade	701
31.9	Fatoração de inteiros.....	709
32	Correspondência de cadeias.....	716
32.1	O algoritmo ingênuo de correspondência de cadeias.....	718
32.2	O algoritmo Rabin-Karp.....	720
32.3	Correspondência de cadeias com autômatos finitos.....	724
32.4	O algoritmo Knuth-Morris-Pratt.....	729
33	Geometria computacional	738
33.1	Propriedades de segmentos de reta.....	738
33.2	Determinando se dois segmentos quaisquer se interceptam	743
33.3	Determinando a envoltória convexa	749
33.4	Localizando o par de pontos mais próximos	757
34	Problemas NP-completos	764
34.1	Tempo polinomial.....	768
34.2	Verificação em tempo polinomial.....	773
34.3	NP-completude e redutibilidade	777
34.4	Provas da NP-completude	785
34.5	Problemas NP-completos.....	791
35	Algoritmos de aproximação	805
35.1	O problema de cobertura de vértices	807
35.2	O problema do caixeiro-viajante.....	809
35.3	O problema de cobertura de conjuntos.....	813
35.4	Aleatorização e programação linear.....	817
35.5	O problema da soma de subconjuntos.....	821
Parte VIII Apêndice: Fundamentos de matemática		
	Introdução	830
A	Somatórios.....	831
A.1	Fórmulas e propriedades de somatórios	831
A.2	Limitando somatórios.....	834
B	Conjuntos etc.	840
B.1	Conjuntos	840
B.2	Relações	844
B.3	Funções	845
B.4	Grafos.....	847
B.5	Árvores	851
C	Contagem e probabilidade.....	858
C.1	Contagem	858
C.2	Probabilidade.....	862
C.3	Variáveis aleatórias discretas	867
C.4	Distribuições geométrica e binomial.....	870
C.5	As caudas da distribuição binomial	875
D	Matrizes	882
D.1	Matrizes e operações com matrizes.....	882
D.2	Propriedades básicas de matrizes.....	886
	Bibliografia.....	892
	Índice.....	903