



Algorithms

FOURTH EDITION

ROBERT SEDGEWICK | KEVIN WAYNE

Algorithms

FOURTH EDITION

Algorithms

Table of Contents

Cover

Title Page

Copyright Page

Contents

Preface

1. Fundamentals

1.1 Basic Programming Model

Primitive data types

Loops and conditionals

Arrays

Static methods

Recursion

APIs

Strings

Input and output

Binary search

1.2 Data Abstraction

Objects

Abstract data types

Implementing ADTs

Designing ADTs

1.3 Bags, Queues, and Stacks

APIs

Arithmetic expression evaluation

Resizing arrays

Table of Contents

Iterators

Linked lists

1.4 Analysis of Algorithms

Running time

Order-of-growth classifications

1.5 Case Study: UnionFind

Dynamic connectivity

Quick find

Quick union

Weighted quick union

2. Sorting

2.1 Elementary Sorts

Rules of the game

Selection sort

Insertion sort

Shellsort

2.2 Mergesort

Abstract in-place merge

Top-down mergesort

Bottom-up mergesort

2.3 Quicksort

2.4 Priority Queues

Priority queue API

Elementary implementations

Binary heap

Heapsort

2.5 Applications

Comparators

Stability

Median and order statistics

Table of Contents

3. Searching

3.1 Symbol Tables

Symbol table API

Ordered symbol table API

Sequential search

Binary search

3.2 Binary Search Trees

Basic implementation

Order-based methods

Deletion

3.3 Balanced Search Trees

2-3 search trees

Redblack BSTs

Deletion

3.4 Hash Tables

Hash functions

Separate chaining

Linear probing

3.5 Applications

Set data type

Whitelist and blacklist filters

Dictionary lookup

Inverted index

File indexing

Sparse matrix-vector multiplication

4. Graphs

4.1 Undirected Graphs

Glossary

Undirected graph type

Adjacency-lists representation

Table of Contents

- Depth-first search
- Breadth-first search
- Connected components
- Degrees of separation

4.2 Directed Graphs

- Glossary
- Digraph data type
- Depth-first search
- Strong connectivity
- KosarajuSharir algorithm
- Transitive closure

4.3 Minimum Spanning Trees

- Cut property
- Greedy algorithm
- Edge-weighted graph data type
- Prim's algorithm
- Kruskal's algorithm

4.4 Shortest Paths

- Properties of shortest paths
- Edge-weighted digraph data types
- Generic shortest paths algorithm
- Dijkstra's algorithm
- Shortest paths in edgeweighted DAGs
- Critical-path method
- BellmanFord algorithm
- Negative cycle detection
- Arbitrage

5. Strings

5.1 String Sorts

- Key-indexed counting

Table of Contents

- LSD string sort
- MSD string sort
- 3-way string quicksort

5.2 Tries

- String symbol table API
- R-way tries
- Ternary search tries

5.3 Substring Search

- Bruteforce algorithm
- KnuthMorrisPratt algorithm
- BoyerMoore algorithm
- Rabin-Karp fingerprint algorithm

5.4 Regular Expressions

- Describing patterns with REs
- Applications
- Nondeterministic finite-state automata
- Simulating an NFA
- Building an NFA corresponding to an RE

5.5 Data Compression

- Rules of the game
- Reading and writing binary data
- Limitations
- Run-length coding
- Huffman compression
- LZW compression

6. Context

Event-Driven Simulation

- Hard-disc model
- Collision prediction
- Collision resolution

Table of Contents

B-trees

Cost model

Search and insert

Suffix Arrays

Suffix sorting

Network Flow

FordFulkerson algorithm

Reductions

Sorting

Shortest path

Bipartite matching

Linear programming

Intractability

Longest-paths problem

Boolean satisfiability

NP-completeness

Index

Algorithms and Clients