Sumary of COSC229 - Algorithms

Tadao Takaoka

June 2002

The topics covered are given in the following with time complexities.

1. Data structures
   1.1 Binary search trees   ---   O(n log n) average for n insertions
   1.2 AVL trees   ---   O(n log n) worst case for n insertions
   1.3 Splay trees   ---   O(n log n) worst case for n insertions
   1.4 2-3 trees   ---   O(n log n) worst case for n insertions
   1.5 Heaps   --- O(n) for n insertions and O(log n) for delete-min
   1.6 Binomial queues   ---   O(log n) for delete-min and O(log n) for meld

2. Sorting
   2.1 Bubble sort   ---   O(n^2)
   2.2 Minimum selection sort   ---   O(n^2)
   2.3 Insertion sort   ---   O(n^2)
   2.3 Heap sort   ---   O(n log n) worst case
   2.4 Quick sort   ---   O(n log n) average
   2.5 Merge sort   ---   O(n log n) worst case
   2.6 Radix sort   ---   O(dn) where d is number of digits

3. Graph algorithms
   3.1 Basic definitions
   3.2 Depth-first search   ---   O(m + n)
   3.2 Breadth-first search   ---   O(m + n)
   3.3 Tarjan’s algorithm for strongly connected components   ---   O(m+n)
   3.4 Dijkstra’s algorithm for the single source shortest path problem   ---   O(n^2)
                   with heap   ---   O(m log n)
   3.5 Warshall’s algorithm for reflexive-transitive closure   ---   O(n^3)
   3.6 Floyd’s algorithm for the all pairs shortest path problem   ---   O(n^3)
   3.7 Kruskal’s algorithm for the minimum cost spanning tree   ---   O(m log n)
   3.8 Prim’s algorithm for the same problem   ---   O(m log n)

4. Pattern matching
   4.1 Naive algorithm   ---   O(m n)
   4.2 KMP algorithm   ---   O(m + n)
   4.3 Boyer-Moore algorithm   ---   O(n/m) average
   4.4 Bird’s algorithm for two-dimensional pattern matching   ---   O(m^2 + n^2)

5. Computational geometry
   5.1 Intersection of line segments   ---   O(1)
   5.2 Interior point in a polygon   ---   O(n)
   5.3 Naive algorithm for convex hull   ---   O(n^2)
   5.4 Graham’s algorithm for convex hull   ---   O(n log n)
   4.5 Lee’s algorithm for Voronoi diagram   ---   O(n log n)
6. Semi-numerical algorithms
6.1 Basics form number theory
6.2 Horner’s algorithm for polynomial evaluation --- O(n)
6.3 Repeated squaring for exponentiation --- O(m) where exponent is m bit number
6.4 Theory of Fibonacci numbers --- O(log n) for the n-th Fibonacci number
6.5 Euclidean algorithm for greatest common divisors --- O(kD(k)) where a and b are k bit numbers and D(k) is the time for division
6.6 Euclidean algorithm for multiplicative inverses
6.7 Analysis of Euclidean algorithm
6.8 Discrete Fourier transform --- O(n^2) by naive method and O(n log n) by FFT
6.9 Convolution --- O(n^2) by naive method and O(n log n) by FFT
6.10 How to multiply polynomials fast --- (n^2) by naive method and O(n log n) by FFT
6.11 How to multiply multiple precision numbers fast --- O(n^2) by naive method and O(n log n) by FFT