Algorithms and Data Structures Cheatsheet

extracted from https://algs4.cs.princeton.edu/cheatsheet/ (Sedgewick's Algorithms book)

We summarize the performance characteristics of classic algorithms and data structures for sorting, priority queues, symbol tables, and graph processing. We also summarize some of the mathematics useful in the analysis of algorithms, including commonly encountered functions; useful formulas and approximations; properties of logarithms; asymptotic notations; and solutions to divide-and- conquer recurrences.

Sorting.

The table below summarizes the number of compares for a variety of sorting algorithms, as implemented in this textbook. It includes leading constants but ignores lower-order terms.

ALGORITHM	IN PLACE	STABLE.	BEST	AVERAGE	WORST	REMARKS
selection sort	✔		$\frac{1}{2} n^2$	$\frac{1}{2} n^2$	$\frac{1}{2} n^2$	*n* exchanges; quadratic in best case
insertion sort	✔	✔	n	$\frac{1}{4} n^2$	$\frac{1}{2} n^2$	use for small or partially-sorted arrays
bubble sort	✔	✔	n	$\frac{1}{2} n^2$	$\frac{1}{2} n^2$	rarely useful; use insertion sort instead

shellsort	✔		$n \log_3 n$	unknown	$c\ n^{3/2}$	tight code; subquadratic
mergesort		✔	½ $n \lg n$	$n \lg n$	$n \lg n$	$n \log n$ guarantee; stable
quicksort	✔		$n \lg n$	$2\ n \ln n$	½ n^2	$n \log n$ probabilistic guarantee; fastest in practice
heapsort	✔		n †	$2\ n \lg n$	$2\ n \lg n$	$n \log n$ guarantee; in place

† $n \lg n$ if all keys are distinct

Priority queues.

The table below summarizes the order of growth of the running time of operations for a variety of priority queues, as implemented in this textbook. It ignores leading constants and lower-order terms. Except as noted, all running times are worst-case running times.

DATA STRUCTURE	INSERT	DEL-MIN	MIN	DEC-KEY	DELETE	MERGE
array	1	n	n	1	1	n
binary heap	$\log n$	$\log n$	1	$\log n$	$\log n$	n
***d*-way heap**	$\log_d n$	$d \log_d n$	1	$\log_d n$	$d \log_d n$	n
binomial heap	1	$\log n$	1	$\log n$	$\log n$	$\log n$
Fibonacci heap	1	$\log n$ †	1	1 †	$\log n$ †	1

† amortized guarantee

Symbol tables.

The table below summarizes the order of growth of the running time of operations for a variety of symbol tables, as implemented in this textbook. It ignores leading constants and lower-order terms.

DATA	worst case SEARCH	worst case INSERT	worst case DELETE	average case SEARCH	average case INSERT	average case DELETE
sequential search (in an unordered list)	*n*	*n*	*n*	*n*	*n*	*n*
binary search (in a sorted array)	log *n*	*n*	*n*	log *n*	*n*	*n*
binary search tree (unbalanced)	*n*	*n*	*n*	log *n*	log *n*	sqrt(*n*)
red-black BST (left-leaning)	log *n*	log *n*	log *n*	log *n*	log *n*	log *n*
AVL	log *n*	log *n*	log *n*	log *n*	log *n*	log *n*
hash table (separate-chaining)	*n*	*n*	*n*	1 †	1 †	1 †
hash table (linear-probing)	*n*	*n*	*n*	1 †	1 †	1 †

† uniform hashing assumption

Graph processing.

The table below summarizes the order of growth of the worst-case running time and memory usage (beyond the memory for the graph itself) for a variety of graph-processing problems, as implemented in this textbook. It ignores leading constants and lower-order terms. All running times are worst-case running times.

PROBLEM	ALGORITHM	TIME	SPACE
path	DFS	$E + V$	V
shortest path (fewest edges)	BFS	$E + V$	V
cycle	DFS	$E + V$	V

directed path	DFS	$E + V$	V
shortest directed path (fewest edges)	BFS	$E + V$	V
directed cycle	DFS	$E + V$	V

topological sort	DFS	$E + V$	V
bipartiteness / odd cycle	DFS	$E + V$	V
connected components	DFS	$E + V$	V
strong components	Kosaraju–Sharir	$E + V$	V
strong components	Tarjan	$E + V$	V
strong components	Gabow	$E + V$	V
Eulerian cycle	DFS	$E + V$	$E + V$
directed Eulerian cycle	DFS	$E + V$	V
transitive closure	DFS	$V(E + V)$	V^2
minimum spanning tree	Kruskal	$E \log E$	$E + V$
minimum spanning tree	Prim	$E \log V$	V
minimum spanning tree	Boruvka	$E \log V$	V
shortest paths (nonnegative weights)	Dijkstra	$E \log V$	V
shortest paths (no negative cycles)	Bellman–Ford	$V(V + E)$	V
shortest paths (no cycles)	topological sort	$V + E$	V
all-pairs shortest paths	Floyd–Warshall	V^3	V^2
maxflow–mincut	Ford–Fulkerson	$E V (E + V)$	V
bipartite matching	Hopcroft–Karp	$V^{½} (E + V)$	V
assignment problem	successive shortest paths	$n^3 \log n$	n^2

Last modified on September 12, 2020.