資料結構和演算法

Data Structure and Algorithm

TZU-CHUN HSU[1]

[1]vm3y3rmp40719@gmail.com

[1]Department of Computer Science, Zhejiang University

2021 年 1 月 20 日

Version 2.1.1

Disclaimer

本文「資料結構與演算法」為「資料結構」和「演算法」筆記的總整理，內容主要參考 Introduction to Algorithms[2] 和洪捷先生的演算法參考書 [1]，以及 wjungle 網友在 PTT 論壇上提供的資料結構筆記 [3][4]。
本文作者為 TZU-CHUN HSU，本文及其 LaTeX 相關程式碼採用 **MIT 協議**，更多內容請訪問作者之 GitHub 分頁Oscarshu0719。

1 Summary

Trees

Tree	Insert x	Delete x	Search x	Remark
BST	$O(\log n) \sim O(n)$			Create: $O(n \log n) \sim O(n^2)$
AVL tree	$O(\log_m n)$			$F_{h+2} - 1 \leq n \leq 2^h - 1$
B tree				$1 + 2\frac{\lceil \frac{m}{2} \rceil^{h-1} - 1}{\lceil \frac{m}{2} \rceil - 1} \leq n \leq \frac{m^h - 1}{m - 1}$
RBT				$h \leq 2\log(n+1)$
Splay tree				Worst: $O(n)$, Amortized: $O(\log n)$

Priority queues

Operations	Max (Min)	Min-max & Deap & SMMH	Leftist	Binomial	Fibonacci
Insert x	$O(\log n)$	$O(\log n)$	$O(\log n)$	$O(\log n), O(1)^*$	$O(1)^*$
Delete max	$O(\log n)$	$O(\log n)$			
Delete min	$O(n)$	$O(\log n)$	$O(\log n)$	$O(\log n)$	$O(\log n)^*$
Delete x				$O(\log n)$	$O(\log n)^*$
Merge	$O(n)$		$O(\log n)$	$O(\log n)$	$O(1)^*$
Decrease key				$O(\log n)$	$O(1)^*$
Search x	$O(n)$				
Find max	$O(1)$	$O(1)$			
Find min		$O(1)$		$O(\log n)$	$O(1)$
Remark			$\textit{shortest}(\text{root}) \leq \log(n+1) - 1$		

Sorting algorithms

Method	Time complexity: Best	Worst	Average	Space complexity	Stable
Insertion	$O(n)$	$O(n^2)$		$O(1)$	$\surd$
Selection	$O(n^2)$			$O(1)$	$\times$
Bubble	$O(n)$	$O(n^2)$		$O(1)$	$\surd$
Shell	$O(n^{1.5})$	$O(n^2)$		$O(1)$	$\times$
Quick	$O(n \log n)$	$O(n^2)$	$O(n \log n)$	$O(\log n) \sim O(n)$	$\times$
Merge	$O(n \log n)$			$O(n)$	$\surd$
Heap	$O(n \log n)$			$O(1)$	$\times$
LSD Radix	$O(n \times k)$			$O(n + k)$	$\surd$
Bucket/MSD Radix	$O(n)$	$O(n^2)$	$O(n + k)$	$O(n \times k)$	$\surd$
Counting	$O(n + k)$				$\surd$

Dynamic Programming algorithms		
Problem	Time complexity	Space complexity
Making change	$O(kn)$	$O(n)$
Fractional Knapsack problem	$\Theta(n \log n)$	$O(n)$
0/1 Knapsack problem (DP)	$O(n2^{\log W})$	$O(n2^{\log W})$
0/1 Knapsack problem (Branch-and-Bound)	$O(2^n)$	
Longest Common Subsequence (LCS)	$O(mn)$	$O(mn)$
Longest Increasing Subsequence (LIS)	$O(n^2)$	$O(n^2)$
Longest Common Substring	$O(mn)$	$O(mn)$
Minimum Edit Distance	$O(mn)$	$O(mn)$
Matrix-chain Multiplication	$O(n^3)$	$O(n^2)$
Traveling Salesperson problem	$\Theta(n^2 2^n)$	$O(n2^n)$
Optimal Binary Search Tree (OBST)	$\Theta(n^3)$	$\Theta(n^2)$

Graph algorithms		
Problem	Time complexity	Remark
Depth-First Search (DFS)	$O(\lvert V\rvert + \lvert E\rvert)$	
Kosaraju's	$O(\lvert V\rvert + \lvert E\rvert)$	
Kruskal's	$O(\lvert E\rvert \log \lvert V\rvert)$	
Prim's (Adjacency matrix)	$O(\lvert V\rvert^2)$	
Prim's (Adjacency list)	$O(\lvert V\rvert\lvert E\rvert)$	
Prim's (Min-Heap, Adjacency list)	$O(\lvert E\rvert \log \lvert V\rvert)$	
Prim's (Fibonacci heap, Adjacency list)	$O(\lvert E\rvert + \lvert V\rvert \log \lvert V\rvert)$	
Sollin's (Borůvka's)	$O(\lvert E\rvert \log \lvert V\rvert)$	
Dijkstra's (Min-heap)	$\Theta((\lvert E\rvert + \lvert V\rvert) \log \lvert V\rvert)$	Greedy, no negative edges or cycles
Dijkstra's (Fibonacci-heap)	$\Theta(\lvert E\rvert + \lvert V\rvert \log \lvert V\rvert)$	
Bellman-Ford	$O(\lvert V\rvert\lvert E\rvert)$	DP
Floyd-Warshall	$\Theta(\lvert V\rvert^3)$	DP, no negative cycles
Johnson's	$\Theta(\lvert V\rvert\lvert E\rvert + \lvert V\rvert^2 \log \lvert V\rvert)$	No negative cycles
Ford-Fulkerson	$O(\lvert E\rvert\lvert f^*\rvert)$	Greedy，f^* 為最大流
Edmond-Karp	$O(\lvert V\rvert\lvert E\rvert^2)$	
Push-relabel	$O(\lvert V\rvert^2\lvert E\rvert)$	

References

[1] 洪捷. 演算法—名校攻略秘笈. 鼎茂圖書出版股份有限公司, 9 edition, 2017.

[2] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms, Third Edition*. The MIT Press, 3 edition, 2009.

[3] wjungle@ptt. 演算法 @tkb 筆記. https://drive.google.com/file/d/0B8-2o6L73Q2VVmNWQk9DY3hsUm8/view?usp=sharing, 2017.

[4] wjungle@ptt. 資料結構 @tkb 筆記. https://drive.google.com/file/d/0B8-2o6L73Q2VeFpGejlYRk1WeFk/view?usp=sharing, 2017.