Back Track

|  |  |
| --- | --- |
| 演算法 | Time Complexity |
| DFS、BFS | Adjacency Matrix：O(n2)  Adjacency List：O(|E|+|V|) |

Minimal Spanning Tree(DS多談Adj Matrix、演算法多談Adj List)

|  |  |  |  |
| --- | --- | --- | --- |
| 演算法 | Time Complexity | 策略 |  |
| Kruskal’s | Adjacency Matrix：O(V2)  Adjacency List(Binary Heap製作)：O(|E|log|E|)  Adjacency List(Fibonacci Heap製作)：O(|E|log|V|) | Greedy | Adjacency Matrix  Min-Heap  Disjoint Sets Tree |
| Prim’s | Adjacency Matrix：O(V2)  Adjacency List(Binary Heap製作)：O(|E|log|V|)  Adjacency List(Fibonacci Heap製作)：O(|V|log|V|+|E|) | Greedy | Adjacency Matrix  Min-Heap  Array |

Shortest Path

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 演算法 | Time Complexity | 策略 | 負邊 | 負環 |
| Dijkstra’s | Adjacency Matrix：O(V2)  Adjacency List：O(V2)  Binary Heap：O(|E|+|V|log|V|)  Fibonacci Heap：O(|E|+|V|log|V|) | Greedy | X | X |
| Bellman-Ford | Adjacency Matrix：O(V3)  Adjacency List：O(V×E) | Dynamic Programming | O | X |
| Floyd-Warshall | Adjacency Matrix：O(V3) | Dynamic Programming | O | X |

Comparison of Various Structures

|  |  |  |  |
| --- | --- | --- | --- |
|  | Array | Linked-List | AVL Tree |
| Insert | O(1) | O(n) | O(log n) |
| Delete | O(1) | O(n) | O(log n) |
| Search | O(log n) //Binary Search | O(n) | O(log n) |
| Search kth item | O(1) | O(k) | O(log n) |
| Delete kth item | O(n-k) //後面n-k個都要往前移一格 | O(k) | O(log n) |
| Output in order | O(n) | O(n) | O(n) |

[Thormas C演算法版]

|  |  |  |  |
| --- | --- | --- | --- |
| Operation | Binary Heap(Worst, DS特愛考) | Binomial Heap(Worst) | Fibonacci Heap(攤) |
| Create-Heap | O(1) | O(1) | O(1) |
| Insert | O(log n) | O(log n) //[1] | O(1) //Lazy Merge |
| Delete | O(log n) //[6] | O(log n) | O(log n) |
| Find-Min | O(1) | O(log n) //[2] | O(1) //[3] |
| Extract-Min | O(log n) | O(log n) | O(log n) |
| Union | O(n) | O(log n) | O(1) //[4] |
| Decrease key | O(log n) | O(log n) //[5] | O(1) |

[1]Binomial Heap的Worst下，Insert O(log n)，但分攤下為O(1)

[2]因為Binomial Heap最差下需比較log n棵樹的Root找min

[3]Fibonacci Heap有設定指標指向Min

[4]因為Fibonacci Heap採用Lazy Merge

[5]例：看高度，因此O(log n)

[6]用最差情況Delete Min來考量：O(log n)

Sort

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | Time Complexity | | | Space Complexity | Stable/Unstable |
|  |  | Best | Average | Worst |
| 初等 | Insertion | **O(n)** | O(n2) | O(n2) | O(1) | **Stable** |
| Selection | O(n2) | O(n2) | O(n2) | O(1) | **Unstable** |
| Bubble | **O(n)** | O(n2) | O(n2) | O(1) | **Stable** |
| Shell | **O(n3/2)** | O(n2) | O(n2) | O(1) | **Unstable** |
| 高等 | Quick | O(log n) | O(log n) | **O(n2)** | **O(log n)~O(n)** | **Unstable** |
| Merge | O(log n) | O(log n) | O(log n) | **O(n)** | **Stable** |
| Heap | O(log n) | O(log n) | O(log n) | **O(1)** | **Unstable** |
| 線性 | Radix(LSD) | **O(d\*(n+r))** ⟹ O(n) | | | **O(r\*n)** | **Stable** |
| Bucket(MSD) | **O(n+r)** ⟹ O(n) //比Radix快 | | | **O(r\*n)** | **Unstable** |
| Counting | **O(n+k)** ⟹ O(n) | | | **O(n+k)** | **Stable** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | Best | Average | Worst |
| 初  等 | Insertion | T(n)=1 | T(n)=T(n-1)+Θ(n) | T(n)=T(n-1)+(n-1) |
| Selection | T(n)=T(n-1)+1 | T(n)=T(n-1)+Θ(n) | T(n)=T(n-1)+(n-1) |
| Bubble | T(n)=1 | T(n)=T(n-1)+Θ(n) | T(n)=T(n-1)+(n-1) |
| 高  等 | Quick | T(n)=2T(n/2)+Θ(n) | T(n)=1/n∑[T(j-1)+T(n-j)+Θ(n)] | T(n)=T(n-1)+Θ(n) |
| Merge | T(n)=2T(n/2)+Θ(n) | T(n)=2T(n/2)+Θ(n) | T(n)=2T(n/2)+Θ(n) |
| Heap | T(n)=Θ(n)+Θ(nlogn) | T(n)=Θ(n)+Θ(nlogn) | T(n)=Θ(n)+Θ(nlogn) |

Greedy

|  |  |
| --- | --- |
| Huffman | O(n log n) |
| Fraction Knapsack Problem | O(n log n) |
| Kruskal’s | Adjacency List：O(|E|log|E|) |
| Prim’s | Adjacency Matrix：O(V2) |
| Sollin’s | O(V2) |
| Dijkstra’s | Adjacency Matrix：O(V2) |
| Convex Hull | O(n log n) |

Dynamic Programming

|  |  |
| --- | --- |
| LCS / LIS | O(mn) / O(n2) |
| 0/1 Knapsack Problem | O(nW) |
| Chain Matrix | O(n3) |
| Bellman-Ford | Adjacency Matrix：O(V3) |
| Floyd-Warshell | Adjacency Matrix：O(V3) |
| OBST | O(n3) |
| TSP(Traveling Salesman Problem) | O(n22n) |

Divide-and-Conquer

|  |  |
| --- | --- |
| Tower of Hanoi | O(2n) |
| Binary Search | O(log n) |
| Quick Sort | O(n log n) |
| Merge Sort | O(n2) |
| Bucket Sort(MSD) | O(n+r) |
| Closet Pair | O(n log n) |
| Strassen’s Matrix Multiplication | O(nlog 7) |

演算法解題技巧

|  |  |
| --- | --- |
| Branch and Bound解KP | O(nW) //Worst Case亦為NPC |
| Prune and Search選第k小之數 | T(n)=T(n/5)+T(3n/4)+ Θ(n)= Θ(n) |
| 陣列合併 | Θ(klogk)+ Θ(k)= Θ(klogk) |
| 列出子集 | Θ(2n) |
| 找1-1函數 | Θ(n) |
| 名人問題 | Θ(n2) |
| 平面上極大點 | Θ(nlogn) |
| 點之Rank | T(n)=2T(n/2)+ Θ(n)= Θ(nlogn) |
| 最大連續整數和 | Θ(n) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Unsorted Single  Linked List | Sorted Single  Linked List | Unsorted Double  Linked List | Sorted Double  Linked List |
| Search(L, k) | O(n) | O(1) | O(n) | O(logn) |
| Insert(L, p) | O(1) | O(n) | O(1) | O(n) |
| Delete(L, p) | O(n) | O(n) | O(1) | O(1) |
| Successor(L, p) | O(1) | O(1) | O(1) | O(1) |
| Predecessor(L, p) | O(n) | O(n) | O(1) | O(1) |
| Minimum(L) | O(n) | O(1) | O(n) | O(1) |
| Maximum(L) | O(n) | O(1) | O(n) | O(1) |