

CODE DI PRIORITA'

Liste NON Ordinate	
inserimento	$O(1)$
removeMin	$O(n)$
Minimo	$O(n)$
Liste Ordinate	
Inserimento	$O(n)$
removeMin	$O(1)$
Minimo	$O(1)$
Heap	
Inserimento – Up-Heap	$O(\log n)$
Rimozione – Down-Heap	$O(\log n)$
Replace	$O(\log n)$
Heapsort	$O(n \log n)$
Heapify	$O(\log n)$
Raddoppio dim array	$O(n \log n)$

ALGORITMI DI ORDINAMENTO

Ordinamento in Place	
Insertion sort	$O(n^2)$
Array2heap	$O(n \log n)$
Selection sort	$O(n^2)$
Merge sort	$O(n \log n)$
Radix Sort	
Bucket sort	$O(n+N)$ ove n/N fase 1/2
Radix Sort	$O(n+N)$ ove n/N fase 1/2
Quick Sort	
Caso peggiore	$O(n^2)$
Tempo atteso	$O(n \log n)$

INSIEMI

Liste	
Find, add, remove	$O(n)$
Fusione generica	$O(n_a + n_b)$ ove n_a / n_b dim.
Spazio usato	$O(n)$

HASHING

Calcolo polinomio	$O(n^2)$
Regola di Horn	$O(n)$
Costi Peggiori / Attesi	
Get	$O(n) / O(1)$
Put	$O(n) / O(1)$
remove	$O(n) / O(1)$

MAPPE

Operazioni Varie con Liste	
Put	$O(1)$
Get	$O(n)$
Remove	$O(n)$

MAPPE ORDINATE

Operazioni Base	
Get	$O(\log n)$
Put	$O(n)$ se assente
Put	$O(\log n)$ se presente
Remove	$O(n)$
RANGE QUERY	$O(\log n + s)$ s è costo accesso

ALBERI BINARI DI RICERCA – BST (Binary Search Tree)

Nel caso peggiore $h = n$ mentre nel migliore $h = \log n$

Operazioni Base	H altezza albero
Ricerca	$O(h)$
Inserimento	$O(h)$
Rimozione	$O(h)$
Range Query	$O(h+s)$ s n.chiavi intervallo
Spazio Richiesto	$O(n)$

Alberi AVL (Adelson-Velskij e Landis)

Altezza	$O(\log n)$ sempre
Rotazioni	$O(1)$
Find, add, remove	$O(\log n)$
SUBMAP	$O(\log n + s)$

CAMMINI MINIMI

Dijkstra (connes, non neg)	$O((n+m) \cdot \log n)$
Bellman-Ford(nocicli,neg)	$O(n \cdot m)$
Tra tutti i nodi	$O(n \cdot (n+m) \cdot \log n)$

ALBERI MINIMI RICOPRENTI – MST (Minimum Spanning Tree)

Prim-Jarnik	$O((n+m) \cdot \log n)$
Kruskal	$O((n+m) \cdot \log n)$

UNION-FIND

Sequenza	
Find	$O(1)$
Makecluster	$O(1)$
Union	$O(n \cdot \text{elem.con id modifica})$
Collegata	Ove a e b sono liste
Find	$O(1)$
Makecluster	$O(1)$
Union	$O(\min\{a.size, b.size\})$
Analisi ammortizzata	$O(k+n \log n)$
Alberi	H altezza albero
Find	$O(h)$
Makecluster	$O(1)$
Union no ottimizzazioni	$O(h)$
Union con ottimizzazioni	$O(k \log n)$

GRAFI (ove n numero nodi, m numero archi, deg grado nodo)

Lista di Archi	
Iteratore su Archi nodo(v)	$O(m)$
sonoAdiacenti(v, w)	$O(m)$
Inserisci Vertice	$O(1)$
Inserisci Arco	$O(1)$
Rimuovi Vertice(v)	$O(m)$
Rimuovi Arco	$O(1)$
Spazio	$O(n+m)$
Lista di Adiacenza	
Iteratore su Archi nodo(v)	$O(n + \deg(v))$
sonoAdiacenti(v, w)	$O(\min\{\deg(v), \deg(w)\})$
Inserisci Vertice	$O(1)$
Inserisci Arco	$O(1)$
Rimuovi Vertice(v)	$O(\deg(v))$
Rimuovi Arco	$O(1)$
Spazio	$O(n+m)$
Matrice di Adiacenza	
Iteratore su Archi nodo(v)	$O(n)$
sonoAdiacenti(v, w)	$O(1)$
Inserisci Vertice	$O(n)$
Inserisci Arco	$O(1)$
Rimuovi Vertice(v)	$O(n^2)$
Rimuovi Arco	$O(1)$
Spazio	$O(n^2)$
Operazioni Varie	
DFS / DFS-Diretta	$O(n+m)$
BFS	$O(n+m)$
Kosaraju(ConnettivitàForte)	$O(n+m)$
Chiusura Transitiva	Lista: $O(n(n+m))$,Mat: $O(n^3)$