

# Esame di algoritmi e strutture dati

9 febbraio 2022

Modalità d'esame disponibili:

1. esonero (35%) + esercizi 5 e 6 (35%) + progetto (30%, obbligatorio per iscritti a partire dall'A.A. 20-21): 1 ora;
2. esercizi 1-4 (70%) + progetto (30%): 2 ore;
3. esercizi 1-5, senza progetto (opzione disponibile solo per iscritti fino all'A.A. 19-20): 2 ore.

Tutti gli esercizi hanno stesso punteggio. A seconda della modalità d'esame, il punteggio totale verrà opportunamente normalizzato, garantendo un massimo totale di 33 punti. Con un punteggio finale strettamente maggiore di 30 si ottiene la lode.

## Esercizio 1

Progettare un algoritmo **ricorsivo** (pseudocodice) con segnatura  $somma(SCL\ T) \rightarrow Intero$  che, presa in input una struttura collegata lineare (SCL)  $T$  i cui nodi contengono valori interi, restituisca la somma dei valori in essa contenuti.

Ad esempio, data l'SCL di input  $T = 1\ 3\ 5\ 2\ 7\ 12\ 9\ 14$ , l'algoritmo deve restituire il valore  $1 + 3 + 5 + 2 + 7 + 12 + 9 + 14 = 53$ .

## Esercizio 2

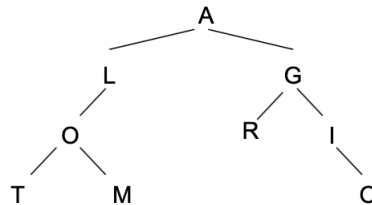
- Si illustri lo pseudocodice dell'algoritmo di ordinamento *QuickSort*.
- Se ne dimostri la complessità temporale nel caso peggiore e se ne illustri la complessità temporale nel caso medio.
- Si mostri un'esecuzione dell'algoritmo presentato sul seguente array di input:  $A = (17, 65, 43, 5, 31, 28, 76, 42)$ .

## Esercizio 3

1. Si fornisca la definizione di albero AVL.
2. Si mostrino i passi di costruzione di un albero AVL quando vengono inserite le seguenti chiavi, nell'ordine riportato: 4, 2, 24, 72, 11, 62, 35, 43, 80, 10, 22, 83.
3. Si illustrino i passi da eseguire per cancellare, nell'ordine, i nodi 24 e 62 dall'albero risultante, ed ottenere, dopo ciascuna eliminazione, un albero AVL.

## Esercizio 4

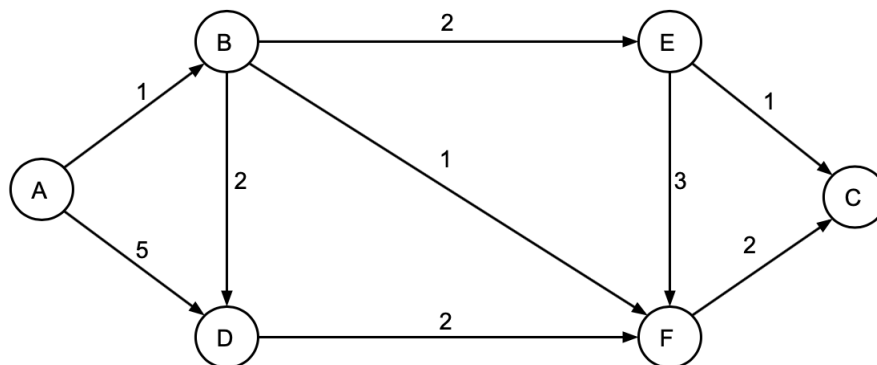
1. Si descriva la rappresentazione con vettore posizionale di un albero binario.
2. Si mostri la rappresentazione con vettore posizionale dell'albero in figura:



3. Si indichi un algoritmo in cui tale rappresentazione viene usata, illustrando la finalit  dell'algoritmo stesso e la sua complessit  temporale e spaziale nel caso peggiore.

## Esercizio 5

1. Si illustri l'algoritmo di Dijkstra, spiegando in una frase quale ne sia lo scopo.
2. Si dimostri il costo dell'algoritmo nella versione presentata (si ignorino possibili ottimizzazioni).
3. Si mostri l'esecuzione dell'algoritmo sul grafo in figura, prendendo come sorgente il nodo  $A$  e riportando il risultato restituito dall'algoritmo:



## Esercizio 6

Si consideri l'insieme di chiavi  $K = \{46, 22, 35, 45, 105, 12, 6, 67, 10, 21\}$ .

1. Definire un'opportuna funzione hash  $h : \mathbb{N} \rightarrow [0, m - 1]$ , per  $m = 12$ .
2. Mostrare i passi d'inserimento in una tabella hash di dimensione  $m$  degli elementi di  $K$ , nell'ordine riportato, adottando uno schema ad indirizzamento aperto.
3. Mostrare i passi d'inserimento in una tabella hash di dimensione  $m$  degli elementi di  $K$ , nell'ordine riportato, adottando uno schema con liste di collisione.