

Cognome e Nome _____

Matricola _____

Appello del 18 febbraio 2019

Esercizio 1 (6 punti)

$$\text{Sia } T(n) = \begin{cases} T(n-1) + cn & \text{se } n > 1 \\ d & \text{se } n = 1 \end{cases}$$

con c e d costanti.

1. **[2 punti]** Si dica, giustificando la risposta, se il **teorema generale** è utilizzabile per dare una stima esplicita (non ricorsiva) di $T(n)$
2. **[4 punti]** Ricordando che $\sum_{i=1}^n i = \frac{n(n+1)}{2}$, si dia una stima esplicita (non ricorsiva) di $T(n)$ facendo uso di un metodo a scelta.

Esercizio 2 (10 punti)

Scrivere un metodo (in pseudocodice o in Java)

static int selezione (int[] a, int k)

che, presi come parametro un array a di numeri interi e un intero k , restituisce il k -esimo valore più piccolo presente tra gli elementi dell'array (senza contare gli elementi ripetuti). Se k è minore di 1, viene restituito l'elemento più piccolo dell'array, mentre se k è maggiore del numero di elementi distinti presenti nell'array a , viene restituito l'elemento più grande dell'array. Il metodo **non** deve modificare l'array.

Ad esempio, se $a=\{1, 8, 3, 1, 5, 10, 1, 4\}$ e $k=2$, il metodo deve restituire 3 (poiché 3 è il secondo valore più piccolo presente nell'array).

Si analizzi la complessità temporale del metodo proposto: tale metodo può richiamare qualsiasi algoritmo di ordinamento e/o di ricerca visto a lezione e deve avere complessità temporale **$O(n \log n)$** nel caso peggiore (*soluzioni con complessità temporale peggiore danno luogo a una valutazione minore, pari a un massimo di 5 punti*), dove n è la lunghezza dell'array a in input.

Esercizio 3 (9 punti)

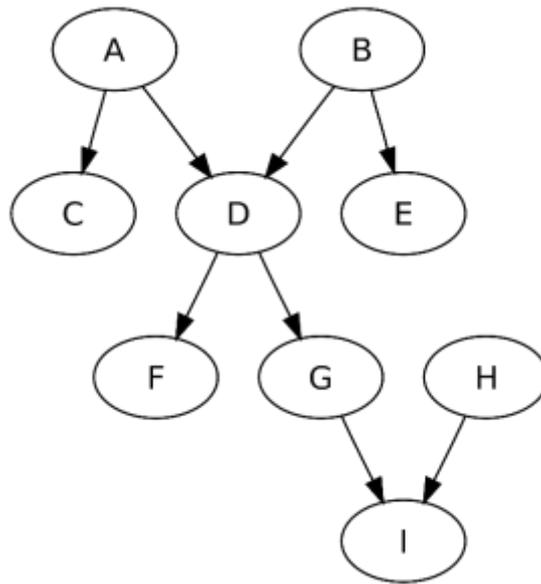
Si consideri la sequenza di chiavi intere

5, 49, 10, 15, 27, 134, 73, 135, 6, 146

- a. **[3 punti]** Mostrare **passo-passo** l'heap di minimo che si ottiene a partire dall'heap vuoto inserendo nell'ordine indicato le chiavi sopra elencate
- b. **[3 punti]** Mostrare la tabella hash di dimensione **17** con liste di trabocco che si ottiene a partire dalla tabella vuota inserendo nell'ordine indicato le chiavi sopra elencate
- c. **[3 punti]** Mostrare la tabella hash di dimensione **17** con indirizzamento aperto (tramite hashing doppio) che si ottiene a partire dalla tabella vuota inserendo nell'ordine indicato le chiavi sopra elencate (si assuma che il secondo parametro della funzione dell'hashing doppio sia **16**).

Esercizio 4 (10 punti)

Si consideri il grafo diretto aciclico (DAG) in figura.



- [2 punti] Mostrare la rappresentazione del grafo tramite liste di adiacenza
- [2 punti] Mostrare la rappresentazione del grafo tramite matrice di adiacenza
- [4 punti] Eseguire **passo-passo** la visita in profondità del grafo, indicando il tempo di inizio e fine visita per ogni nodo (si fa riferimento all'algoritmo DFS che prende in input un grafo e richiama al suo interno l'algoritmo DFS-VISIT su vari nodi sorgente).
- [2 punti] Sfruttando la visita del passo precedente, individuare un ordinamento topologico dei nodi del DAG.

Regole per lo svolgimento della prova scritta:

- Per svolgere il compito si hanno a disposizione **100** minuti.
- Scrivere **subito** nome, cognome, matricola e numero del compito su OGNI FOGLIO (**compreso questo**).
- Durante la prova scritta **non** è possibile abbandonare l'aula.
- Non è ammesso **per nessun motivo** comunicare in qualsiasi modo con altre persone
- È possibile consultare appunti, libri, dispense o qualsiasi altro materiale.
- Qualsiasi strumento elettronico di calcolo o comunicazione (telefoni cellulari, calcolatrici, palmari, computer, etc...) deve essere **completamente disattivato** e **depositato in vista sulla cattedra**
- Mettere in vista sul banco il proprio libretto (o altro documento di identità).

Cognome e Nome _____

Matricola _____

Appello del 18 febbraio 2019

Esercizio 1 (6 punti)

$$\text{Sia } T(n) = \begin{cases} T(n-1) + cn & \text{se } n > 1 \\ d & \text{se } n = 1 \end{cases}$$

con c e d costanti.

1. **[2 punti]** Si dica, giustificando la risposta, se il **teorema generale** è utilizzabile per dare una stima esplicita (non ricorsiva) di $T(n)$
2. **[4 punti]** Ricordando che $\sum_{i=1}^n i = \frac{n(n+1)}{2}$, si dia una stima esplicita (non ricorsiva) di $T(n)$ facendo uso di un metodo a scelta.

Esercizio 2 (10 punti)

Scrivere un metodo (in pseudocodice o in Java)

static int selezione (int[] a, int k)

che, presi come parametro un array a di numeri interi e un intero k , restituisce il k -esimo valore più piccolo presente tra gli elementi dell'array (senza contare gli elementi ripetuti). Se k è minore di 1, viene restituito l'elemento più piccolo dell'array, mentre se k è maggiore del numero di elementi distinti presenti nell'array a , viene restituito l'elemento più grande dell'array. Il metodo **non** deve modificare l'array.

Ad esempio, se $a=\{1, 8, 3, 1, 5, 10, 1, 4\}$ e $k=2$, il metodo deve restituire 3 (poiché 3 è il secondo valore più piccolo presente nell'array).

Si analizzi la complessità temporale del metodo proposto: tale metodo può richiamare qualsiasi algoritmo di ordinamento e/o di ricerca visto a lezione e deve avere complessità temporale **$O(n \log n)$** nel caso peggiore (soluzioni con complessità temporale peggiore danno luogo a una valutazione minore, pari a un massimo di 5 punti), dove n è la lunghezza dell'array a in input.

Esercizio 3 (9 punti)

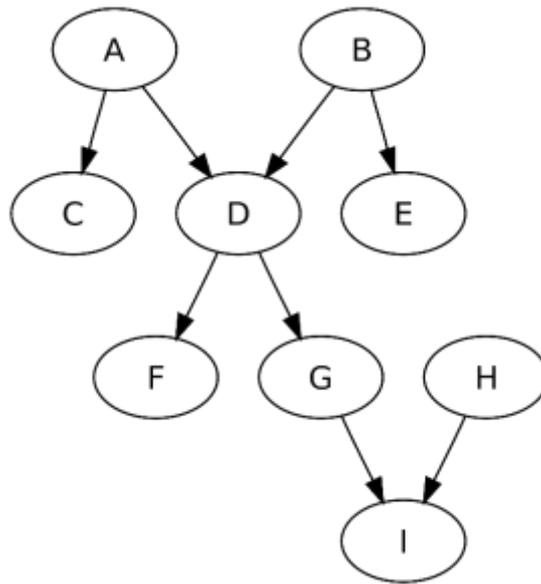
Si consideri la sequenza di chiavi intere

8, 52, 13, 137, 18, 76, 138, 9, 149, 30

- a. **[3 punti]** Mostrare **passo-passo** l'heap di minimo che si ottiene a partire dall'heap vuoto inserendo nell'ordine indicato le chiavi sopra elencate
- b. **[3 punti]** Mostrare la tabella hash di dimensione **17** con liste di trabocco che si ottiene a partire dalla tabella vuota inserendo nell'ordine indicato le chiavi sopra elencate
- c. **[3 punti]** Mostrare la tabella hash di dimensione **17** con indirizzamento aperto (tramite hashing doppio) che si ottiene a partire dalla tabella vuota inserendo nell'ordine indicato le chiavi sopra elencate (si assuma che il secondo parametro della funzione dell'hashing doppio sia **16**).

Esercizio 4 (10 punti)

Si consideri il grafo diretto aciclico (DAG) in figura.



- [2 punti] Mostrare la rappresentazione del grafo tramite liste di adiacenza
- [2 punti] Mostrare la rappresentazione del grafo tramite matrice di adiacenza
- [4 punti] Eseguire **passo-passo** la visita in profondità del grafo, indicando il tempo di inizio e fine visita per ogni nodo (si fa riferimento all'algoritmo DFS che prende in input un grafo e richiama al suo interno l'algoritmo DFS-VISIT su vari nodi sorgente).
- [2 punti] Sfruttando la visita del passo precedente, individuare un ordinamento topologico dei nodi del DAG.

Regole per lo svolgimento della prova scritta:

- Per svolgere il compito si hanno a disposizione **100** minuti.
- Scrivere **subito** nome, cognome, matricola e numero del compito su **OGNI FOGLIO (compreso questo)**.
- Durante la prova scritta **non** è possibile abbandonare l'aula.
- Non è ammesso **per nessun motivo** comunicare in qualsiasi modo con altre persone
- È possibile consultare appunti, libri, dispense o qualsiasi altro materiale.
- Qualsiasi strumento elettronico di calcolo o comunicazione (telefoni cellulari, calcolatrici, palmari, computer, etc...) deve essere **completamente disattivato** e **depositato in vista sulla cattedra**
- Mettere in vista sul banco il proprio libretto (o altro documento di identità).