

Cognome e Nome \_\_\_\_\_

Matricola \_\_\_\_\_

## Appello del 4 giugno 2019

### Esercizio 1 (6 punti)

$$\text{Sia } T(n) = \begin{cases} 2T\left(\frac{n}{2}\right) + cn^2 & \text{se } n > 1 \\ d & \text{se } n = 1 \end{cases}$$

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]** 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 Java

**static boolean equivalenti (int[] a, int[] b)**

che, presi come parametro due array **a** e **b** di numeri interi, restituisce *true* se e solo se **a** e **b** contengono gli stessi elementi (non tenendo in conto le ripetizioni degli elementi e/o l'ordine in cui essi compaiono).

Se **a** e **b** valgono entrambi *null*, viene restituito *true*; se solo uno dei due array vale *null*, viene restituito *false*.

Ad esempio, se  $a=\{1, 8, 3, 1, 5, 10, 1, 4\}$  e  $b=\{8, 1, 5, 10, 4, 3, 3\}$ , il metodo deve restituire *true*.

**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)

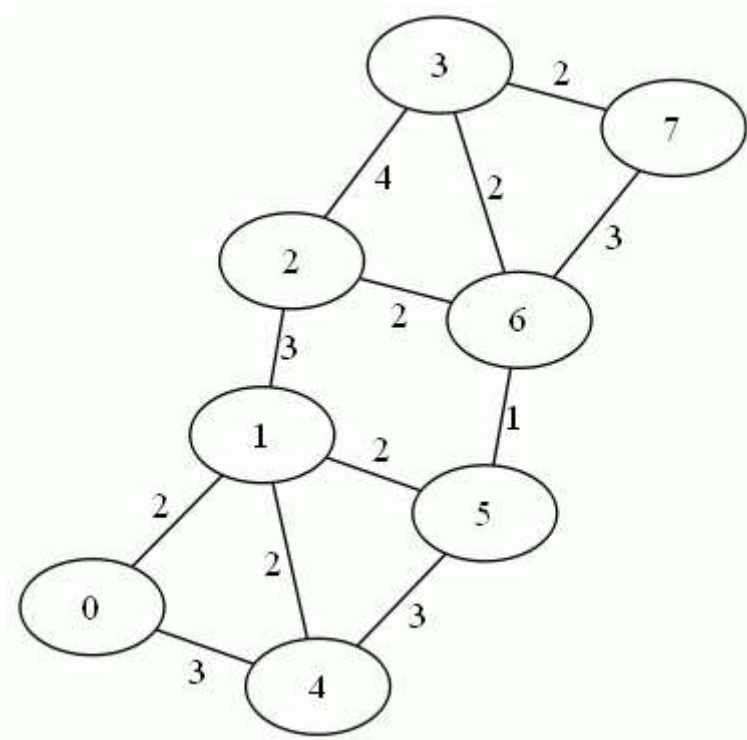
Si consideri la sequenza di chiavi intere

**10, 20, 5, 50, 31, 14, 2, 6, 1, 8, 25**

- a. **[3 punti]** Mostrare **passo-passo** l'heap di minimo che si ottiene applicando la procedura **buildMinHeap** all'array contenente le chiavi nella sequenza sopra indicata.
- b. **[3 punti]** Mostrare la tabella hash di dimensione **13** 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 **13** 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 **12**).

#### Esercizio 4 (10 punti)

Si consideri il grafo non diretto in figura.



- [2 punti]** Mostrare la rappresentazione del grafo tramite liste di adiacenza
- [2 punti]** Mostrare la rappresentazione del grafo tramite matrice di adiacenza
- [6 punti]** Mostrare una possibile esecuzione **passo-passo** dell'algoritmo di **Dijkstra** per l'albero dei cammini minimi a partire dal nodo sorgente **x**, dove **x** è ottenuto *prendendo l'ultima cifra del numero della propria matricola e calcolando il resto della divisione per 8*.  
Ad ogni passo, bisogna mostrare il contenuto della coda con priorità utilizzata dall'algoritmo.

#### 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).