

# Algoritmi e Strutture Dati

Docente: Sabrina De Capitani di Vimercati  
Appello online del 23 Giugno 2017  
Tempo a disposizione: 2:30 ore

## Domanda 1)

Rispondere brevemente, ma in modo completo, alle seguenti domande.

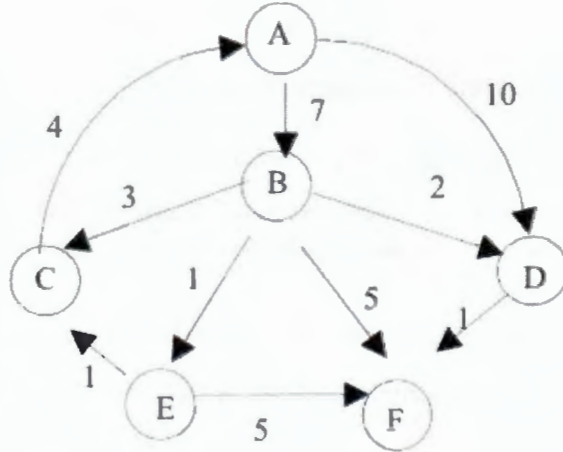
- √ 1. Nell'ambito delle tabelle di hash, cosa si intende per *indirizzamento aperto*?
- √ 2. *Matrice di adiacenza* e *liste di adiacenza*: a cosa servono? Dato un nodo, quale è la complessità computazionale di una operazione che calcola il grado del nodo con matrice di adiacenza e con le liste di adiacenza?
- √ 3. Si richiede di descrivere la struttura dati astratta *pila*, incluse le operazioni che la caratterizzano.
- √ 4. Dire cosa si intende per *lista bidirezionale con sentinella*. A cosa serve la sentinella? Quali vantaggi offre?
- √ 5. Dato un grafo  $G$ , dire cosa si intende per *albero di copertura di costo minimo*.
- √ 6. Si richiede di scrivere l'equazione ricorsiva che rappresenta la complessità computazionale del seguente algoritmo.

```
Algo(A,i,j) /* A array che contiene numeri interi; i,j numeri interi */
{
  c:=0;
  if j-i>10 then
    for k:=1 to 5
      c:= c+Algo(A,i+k,(i+j)/2)
    c := c+ Algo(A,i,j-1)
    c := c+ Algo(A,i+1,j)
    for k1:= i to j
      for k2:= k1+1 to j
        if A[k1] = A[k2] then c := c+1
  return c
}
```

- √ 7. Nell'ambito di una visita DFS, cosa si intende per archi in  $T$ , archi all'indietro, archi in avanti, ed archi di attraversamento?
- √ 8. A cosa serve l'algoritmo di Pape D'Esopo? Quale è la sua complessità?
- √ 9. Descrivere i diversi metodi di attraversamento di un albero (*previsita*, *postvisita* e *invisita*) e mostrare un esempio di visita per i tre metodi. Si richiede di usare un unico albero e di listare ordine con cui i nodi vengono visitati (albero deve avere almeno 10 nodi).
- √ 10. Descrivere (e mostrare un esempio di funzionamento) la procedura "restauraheap".

✓ **Esercizio 1)**

Dato il seguente grafo orientato e pesato si richiede di calcolare i cammini minimi da A usando l'algoritmo di Dijkstra. Compilare la tabella allegata (ogni riga rappresenta il contenuto corrente della coda e all'interno di ciascuna cella deve essere specificata la distanza ed il predecessore; iterazione 0 corrisponde ad inizializzazione).



✓ **Esercizio 2)**

Costruire la tabella hash risultante dall'inserimento dei valori 60, 39, 79, 11, 80, 32 nell'assunzione di hashing interno con funzione primaria di hash  $H(k) = k \bmod 10$ , numero di celle  $m=10$  e gestione delle collisioni tramite scansione lineare ( $h = 1$ ).

✓ **Esercizio 3)**

Si consideri un albero binario  $T$  in cui a ciascun nodo  $v$  è associato un numero reale  $t.val$ . Dato un qualsiasi cammino che porta dalla radice ad una foglia, il costo di tale cammino corrisponde alla somma dei valori associati a tutti i nodi attraversati (inclusa la radice e la foglia di destinazione). Si richiede di scrivere un algoritmo (pseudocodice) che restituisce il costo massimo tra tutti i cammini radice-foglia. Nel caso di albero vuoto, l'algoritmo deve restituire 0.