

ESAME DI ALGORITMI E STRUTTURE DATI
A.A. 2015/2016 – APPELLO DEL 20/06/2016

TEMPO A DISPOSIZIONE: 120 MINUTI

ESERCIZIO 1:

Sia **P** una pila contenente i seguenti valori { **3, 2, 6, 7, 8, 9** } dove '**3**' è l'ultimo elemento inserito nella pila.

- a) Fornire una rappresentazione grafica dell'albero **A** ottenuto dall'algoritmo `creaAVL(pila P) -> albero` quando come input viene fornita la pila **P**.
- b) Fornire, motivando dettagliatamente la risposta, la sequenza di interi stampati dall'algoritmo `stampa(pila P)` quando come input viene fornita la pila **P**.

NOTA: La funzione `stampaVisitaInAmpiezza(albero A)` stampa i nodi dell'albero **A** nell'ordine in cui questi sono visitati da una *visita in ampiezza*.

```
creaAVL(pila P)
    sia A un albero binario di ricerca inizialmente vuoto;
    while(not P.isEmpty())
        A.insert(P.pop());
return A;
```

```
stampa(pila P)
    sia A un albero binario di ricerca;
    A <- creaAVL(pila P);
    stampaVisitaInAmpiezza(A);
```

ESERCIZIO 2:

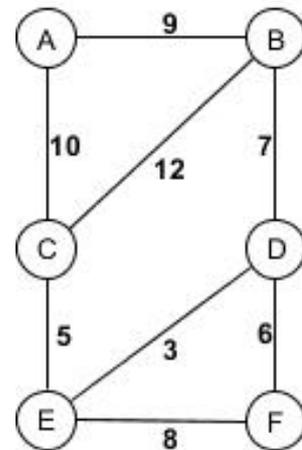
- i) Si scriva lo pseudocodice dell'algoritmo `MergeSort` in grado di ordinare gli elementi contenuti in un array **A** di **n** interi preso in ingresso.
- ii) Descrivere i passi eseguiti dall'algoritmo `MergeSort` sul seguente array in input.

0	1	2	3	4	5	6	7
6	3	4	5	2	9	8	4

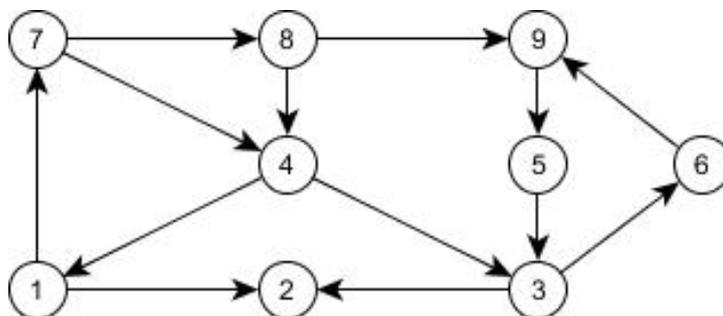
- iii) Indicare la complessità dell'algoritmo `MergeSort` nel caso peggiore e dire, motivando la risposta, se l'algoritmo `MergeSort` è o non è un algoritmo di ordinamento ottimo.
- iv) Indicare quale è la tecnica algoritmica adottata nel `MergeSort`.

ESERCIZIO 3:

- i) Si scriva lo pseudocodice dell'algoritmo di Kruskal per il calcolo di un minimo albero ricoprente di un grafo non orientato, connesso e pesato sugli archi.
- ii) Calcolare il minimo albero ricoprente del grafo rappresentato in figura adottando l'algoritmo di Kruskal. Descrivere dettagliatamente i vari passi eseguiti dall'algoritmo.

**ESERCIZIO 4:**

- i) Si scriva lo pseudocodice dell'algoritmo visitaBFS (vertice v) → albero che esegue la visita in ampiezza di un grafo orientato.
- ii) Descrivere i passi dell'algoritmo di visita quando applicato al grafo orientato rappresentato in figura partendo dal nodo 1 e presentare l'albero BFS prodotto dalla visita.

**ESERCIZIO 5:**

Indicare i vantaggi e gli svantaggi della rappresentazione dei dati indicizzata (es. tramite array) e della rappresentazione dei dati collegata (es. liste).

ESERCIZIO 6 (PER LA LODE):

Fornire la definizione formale delle seguenti notazioni asintotiche:

$$f(n) = O(g(n))$$

$$f(n) = \Omega(g(n))$$

$$f(n) = \Theta(g(n))$$