

Algoritmi e Strutture Dati

Docente: Sabrina De Capitani di Vimercati

Prima prova in itinere del 19 Novembre 2016

Tempo a disposizione: 1:30 ore

Domanda 1)

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

1. Dire cosa si intende per *complessità computazionale asintotica* e di definire la notazione sintotica “O grande” e “omega grande”.
2. Fornire la definizione di struttura dati *pila* indicando sia la specifica sintattica sia la specifica semantica.
3. Descrivere la realizzazione *p/p/f* per gli alberi.
4. Dire cosa si intende per *albero di decisione* e a cosa serve. Fornire un esempio.
5. Dato un grafo G si richiede di descrivere la sua realizzazione tramite *matrici* e *liste di adiacenza*. Si richiede di mostrare un esempio per ognuna delle realizzazioni discusse e di indicare lo spazio di memoria occupato.
6. Nell’ambito delle tabelle di hash, descrivere i metodi di scansione *lineare*, *quadratica* e *hashing doppio*.
7. Descrivere l’operazione di *cancellazione* in un albero binario di ricerca. Si richiede di fornire degli esempi.

Esercizio 1)

Dato il frammento di codice:

```
...
count=0
j=0
for i=0 to log(n)
{
  j=2 <
  while j/2^{n}
  {
    Istruzione
    j=2j
  }
}
...
```

si richiede di valutarne la complessità computazionale nei seguenti due casi:

1. Istruzione corrisponde a `count++`
2. Istruzione corrisponde a `count=count + f(i)` dove la funzione $f(n)$ ha complessità $O(n)$.

Esercizio 2)

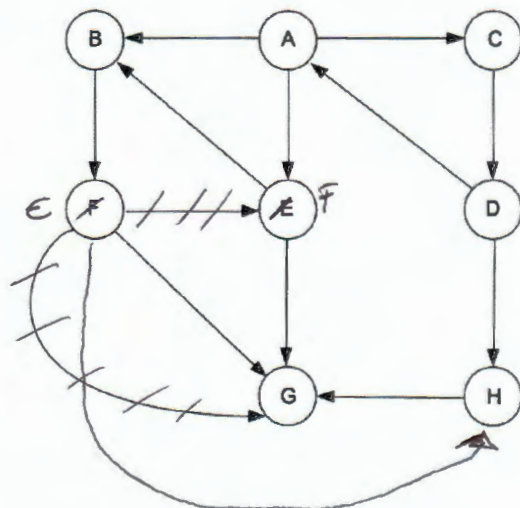
Si richiede di ordinare la seguente sequenza di numeri memorizzati in un array utilizzando l'algoritmo *Heapsort*. Mostrare il contenuto dell'array e la struttura ad albero dello heap ad ogni passo di esecuzione dell'algoritmo di ordinamento.

6	18	23	55	1	17	32	46	59	57
---	----	----	----	---	----	----	----	----	----

Esercizio 3)

Dato il seguente grafo si richiede di eseguire una visita *DFS* sul grafo partendo dal nodo A e di illustrare il corrispondente albero di copertura. Mostrare inoltre come la visita partiziona gli archi nei sottoinsiemi che contengono gli archi d'albero, in avanti, all'indietro e di attraversamento. Si noti che le liste di adiacenza dei nodi del grafo sono come segue:

- ListaAdj[A] = [B,C]
- ListaAdj[B] = [E]
- ListaAdj[C] = [D]
- ListaAdj[D] = [A,H]
- ListaAdj[E] = [F,G,H]
- ListaAdj[F] = [G,B]
- ListaAdj[G] = []
- ListaAdj[H] = [G]



SPACCHIATO