

# Informatica Generale II - Prova teorica

A.A. 20052006

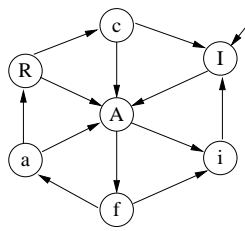
Esame: 25 gennaio 2007

**Codice: MGSY**

1. Si consideri la rappresentazione di un grafo mediante *matrice delle adiacenze* (sia  $n$  il numero di nodi). Quale tra le seguenti affermazioni è *falsa*?
  - (a) l'elemento nella riga  $i$  e nella colonna  $j$  della matrice è pari a 1 se nel grafo rappresentato c'è un arco dal nodo  $i$  al nodo  $j$ , è pari a 0 nel caso contrario;
  - (b) l'operazione di accesso ai successori di un nodo richiede l'accesso ad  $n$  elementi della matrice;
  - (c) tale rappresentazione richiede un'occupazione di memoria proporzionale al numero di nodi del grafo;
  - (d) tale rappresentazione richiede un'occupazione di memoria proporzionale al numero massimo di archi del grafo;
  - (e) non rispondo
2. Si consideri il limite superiore asintotico  $\mathcal{O}(f(n))$  alla complessità di un problema. Quale tra le seguenti affermazioni è *falsa*:
  - (a) un algoritmo ha complessità  $\mathcal{O}(f(n))$  se, per ogni  $n$  e per qualche input di dimensione  $n$ , l'algoritmo impiega una quantità di risposte proporzionali a  $f(n)$ ;
  - (b) esiste almeno un algoritmo di soluzione avente complessità  $\mathcal{O}(f(n))$ ;
  - (c) esistono tre opportune costanti  $a, b, c$  tali che il numero di istruzioni  $t(n)$  che vengono eseguite nel caso peggiore con input di dimensione  $n$  verifica per ogni  $n > c$  la seguente relazione:  $t(n) < a * f(n) + b$ ;
  - (d) La notazione  $\mathcal{O}(f(n))$  fornisce una valutazione approssimata per eccesso della complessità di un algoritmo;
  - (e) non rispondo
3. La seguente funzione `foo`:

```
void foo(Node * x, Node * y) {  
    y->next = x->next;  
    x->next = y;  
}
```

  - (a) concatena due liste concatenate  $x$  e  $y$
  - (b) inserisce il nodo  $y$  tra il nodo  $x$  e il successore di  $x$
  - (c) inserisce la lista puntata da  $x$  dopo il nodo  $y$
  - (d) inserisce la lista puntata da  $y$  dopo il nodo  $x$
  - (e) non rispondo
4. Dato il grafo e la sua rappresentazione con matrice delle adiacenze riportati in figura 1:
  - (a) La visita in ampiezza dal vertice I produce la seguente sequenza d'uscita: I A f i a R c
  - (b) Nessuna delle risposte è accettabile;
  - (c) La visita in ampiezza dal vertice I produce la seguente sequenza d'uscita: I A f a R c i
  - (d) La visita in ampiezza dal vertice I produce la seguente sequenza d'uscita: I c i A f R a
  - (e) non rispondo
5. Quali degli alberi in figura 2 hanno una lunghezza del cammino minore di 15?
  - (a) A,B
  - (b) A,B,C



	c	R	a	f	i	I	A
c	0	0	0	0	0	1	1
R	1	0	0	0	0	0	1
a	0	1	0	0	0	0	1
f	0	0	1	0	1	0	0
i	0	0	0	0	0	1	0
I	0	0	0	0	0	0	1
A	0	0	0	1	1	0	0

Figura 1:

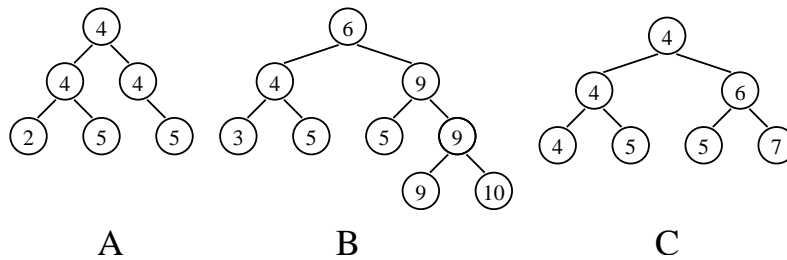


Figura 2:

- (c) B,C  
 (d) A,C  
 (e) non rispondo
6. Si supponga di avere la sequenza di numeri 3,8,6,4,5. Dopo 3 iterazioni soltanto di Bubblesort su tale sequenza, il risultato sarà:
- (a) 3,4,5,8,6  
 (b) 3,4,8,6,5  
 (c) 3,8,4,5,6  
 (d) 3,4,8,5,6  
 (e) non rispondo
7. Il costruttore è una funzione speciale che:
- (a) viene definita all'interno di una struttura e ha lo scopo di allocare dinamicamente memoria nello heap per i campi della struttura;  
 (b) viene definita all'interno di una struttura, ha come identificatore lo stesso nome della struttura, non ha tipo di ritorno e serve per inizializzare i campi della struttura;  
 (c) viene definita all'interno di una struttura, ha come identificatore lo stesso della struttura, non ha tipo di ritorno e deve inizializzare tutti i campi della struttura;  
 (d) viene definita all'interno di una struttura, può essere identificata tramite un qualunque identificatore valido e deve ritornare un tipo identico al tipo della struttura;  
 (e) non rispondo
8. Si considerino i seguenti due frammenti di codice:

```
const int dim=10;
...
int vett[dim];

int i;
...
```

```
i=10;
...
int vett [i];
```

- (a) Il primo frammento è corretto mentre il secondo produce un errore a tempo di compilazione poiché la dimensione dell'array deve essere rappresentata da una costante;
  - (b) I due frammenti sono equivalenti quanto a risultato poiché , sebbene il primo frammento rappresenti una allocazione statica di memoria per l'array, mentre il secondo una allocazione dinamica di memoria per l'array, lo spazio allocato si trova in entrambi i casi all'interno dello stack;
  - (c) I due frammenti di codice sono esattamente equivalenti e hanno come risultato la allocazione statica di un vettore di 10 elementi;
  - (d) I due frammenti non sono equivalenti in quanto rappresentano rispettivamente una allocazione statica di memoria all'interno dello stack e una allocazione dinamica all'interno dello heap;
  - (e) non rispondo
9. In quale ordine partendo dalla radice vengono visitati i nodi dell'albero in figura 3 da un algoritmo di attraversamento post-order?

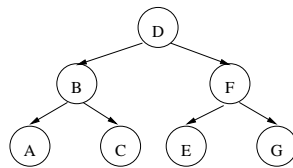


Figura 3: Albero

- (a) ABCDEFG
  - (b) GFEDCBA
  - (c) ACBEGFD
  - (d) CABGEGD
  - (e) non rispondo
10. Quale è la lunghezza del cammino dell'albero in figura 4

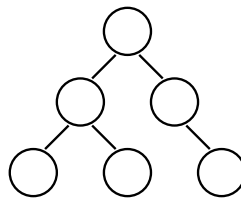


Figura 4:

- (a) 8
- (b) 9
- (c) 6
- (d) 2
- (e) non rispondo