

Informatica Generale II - Prova teorica

A.A. 2005/2006

Esame: 8 settembre 2006

Codice: MWJC

1. Si consideri il seguente frammento di codice:

```
struct Tipo1 {
    int a;

    Tipo1() { a = 0; }
    Tipo1(int _a) { a = _a; }
};

struct Tipo2 {
    Tipo1 a;

    Tipo2(int _a) {
        a = Tipo1(_a);
    }
};
```

- (a) è errato perché Tipo1 e Tipo2 contengono due campi con lo stesso nome.
(b) è corretto, ma Tipo2 non è istanziabile.
(c) è errato perché Tipo2 non ha un costruttore di default.
(d) è corretto e compilabile.
(e) non rispondo
2. 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) tale rappresentazione richiede un'occupazione di memoria proporzionale al numero di nodi del grafo;
(b) l'operazione di accesso ai successori di un nodo richiede l'accesso ad n elementi della matrice;
(c) 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;
(d) tale rappresentazione richiede un'occupazione di memoria proporzionale al numero massimo di archi del grafo;
(e) non rispondo
3. Quale tra le seguenti affermazioni relative all'ordinamento a bolle (bubblesort) è *falsa*?
- (a) la complessità dell'algoritmo dipende dai valori dei dati d'ingresso;
(b) non presenta in alcun caso costo d'esecuzione pari a $\mathcal{O}(n)$;
(c) considerato che la prima iterazione dell'algoritmo termina avendo posto in cima all'array la più piccola componente dell'array medesimo, dopo $(n-1)$ iterazioni (n essendo la dimensione dei dati d'ingresso) l'array è sicuramente ordinato sebbene non sempre tutte le $(n-1)$ iterazioni siano necessarie;
(d) ha complessità asintotica $\mathcal{O}(n^2)$;
(e) non rispondo
4. In quale ordine partendo dal nodo C vengono visitati i nodi del grafo in figura 1 da un algoritmo di visita in profondità?
- (a) CBLHDAa

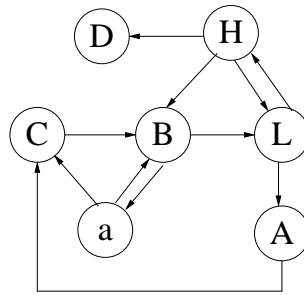


Figura 1:

- (b) CBLaADH
- (c) CBLaHAD
- (d) Nessuna risposta è accettabile;
- (e) non rispondo

5. Supponendo di aver definito una funzione funzEsame con la seguente intestazione:

```
void funzEsame (int *i, int *j);
```

L'invocazione della suddetta funzione funzEsame avrà la forma:

- (a) `int i=10, j=5;`
`funzEsame(*i, *j);`
- (b) `int i=10, j=5;`
`funzEsame(&i, &j);`
- (c) `int i=10, j=5;`
`funzEsame(i, j);`
- (d) `int i=10, j=5;`
`funzEsame();`
- (e) non rispondo

6. In quale ordine partendo dalla radice vengono visitati i nodi dell'albero in figura 2 da un algoritmo di attraversamento pre-order?

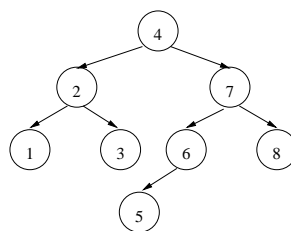


Figura 2: Albero

- (a) 87654321
- (b) 12345678
- (c) 42137658
- (d) 85673124
- (e) non rispondo

7. Si osservino i due alberi binari di ricerca in figura 3 e si consideri la complessità dell'algoritmo di ricerca su tali alberi (sia n il numero di nodi dell'albero):

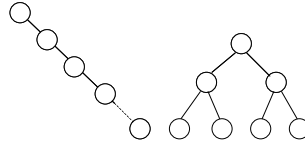


Figura 3: Alberi

- (a) la complessità non dipende dalla conformazione dell'albero per cui un algoritmo di ricerca ha eguale efficienza in ambedue i casi;
 - (b) il caso peggiore per l'algoritmo si ha quando esso opera su un albero come quello rappresentato a destra (complessità $\mathcal{O}(n)$), mentre il caso migliore si ha quando opera su un albero come quello rappresentato a sinistra (complessità $\mathcal{O}(\log(n))$);
 - (c) la complessità dell'algoritmo di ricerca non ha a che fare con l'altezza dell'albero ed è in ogni caso pari a $\mathcal{O}(\log(n))$
 - (d) il caso peggiore per l'algoritmo si ha quando esso opera su un albero come quello rappresentato a sinistra (complessità $\mathcal{O}(n)$), mentre il caso migliore si ha quando opera su un albero come quello rappresentato a destra (complessità $\mathcal{O}(\log(n))$);
 - (e) non rispondo
8. Una pila è:
- (a) un multiinsieme di elementi gestiti secondo la politica *fifo* (first in first out);
 - (b) un multiinsieme di elementi gestiti secondo la politica *lifo* (last in first out);
 - (c) un insieme di elementi gestiti secondo la politica *lifo* (last in first out);
 - (d) un multiinsieme di elementi in cui ogni eliminazione ha per oggetto l'elemento inserito per primo;
 - (e) non rispondo
9. Si supponga di avere la sequenza di numeri 1,6,4,2,3. Dopo 3 iterazioni soltanto di Bubblesort su tale sequenza, il risultato sarà:
- (a) 1,6,2,3,4
 - (b) 1,2,3,6,4
 - (c) 1,2,6,3,4
 - (d) 1,2,6,4,3
 - (e) non rispondo
10. Un albero è ordinato se
- (a) è un albero dove è specificato l'ordine dei figli di qualche nodo.
 - (b) è un albero con radice ed è specificato l'ordine dei figli della radice.
 - (c) è un albero con radice ed è specificato l'ordine dei figli di qualche nodo.
 - (d) è un albero con radice ed è specificato l'ordine dei figli di ciascun nodo.
 - (e) non rispondo