

Programmazione I

Prova scritta - 2 luglio 2013 - 1h20min

NOTA: Nei programmi si trascuri ogni problema legato al tipo ed al valore di ritorno della funzione **main**, inoltre si sottintenda la presenza delle direttive `#include <iostream> / #include <fstream> / using namespace std ;` e non si prenda come un buon esempio la formattazione utilizzata (spesso compressa per motivi di spazio). Si interpreti “terminazione forzata”, come l'abbreviazione di “terminazione forzata del programma da parte del sistema operativo”. Infine, laddove si trovi l'affermazione che un programma o frammento di codice produce un certo risultato, è da intendersi che, in accordo alle regole del linguaggio, tale programma o frammento di codice produce quel risultato per qualsiasi esecuzione su qualsiasi macchina.

PARTE 1 – RISPOSTA SINGOLA - Ogni domanda ha una sola risposta VERA.

- Una risposta esatta fa acquisire il punteggio positivo riportato a fianco della domanda
 - Una risposta errata fa perdere il punteggio negativo riportato a fianco della domanda
 - Una risposta lasciata in bianco viene valutata 0
1. (2, -0.5) Quale delle seguenti affermazioni è vera?
 - a) Gli elementi di un array possono essere di tipo diverso tra loro
 - b) L'indice di un array può essere di tipo **double**
 - c) È un errore inserire in una lista di inizializzazione di un array più elementi di quanti ne possa contenere l'array stesso
 - d) È un errore inserire in una lista di inizializzazione di un array meno elementi di quanti ne possa contenere l'array stesso
 2. (2, -.5) Dato il seguente programma:

```
void fun(int a[], int n)
{ for (int i = 0 ; i < n ; i++) cin>>a[i] ;}

main() { int c[3] ; fun(c, 3) ; }
```

 - a) La funzione **fun** modifica solo una copia locale dell'array che le viene passato
 - b) La funzione **fun** modifica l'array **c** che le viene passato
 - c) La variabile **c** ha classe di memorizzazione statica
 - d) Nessuna delle altre risposte è corretta
 3. (4, -.5) Dato il seguente programma:

```
int *a, num_elem ;

void inserisci(int n)
{ if (a == 0) a = new int[10] ; a[num_elem] = n ; num_elem++ ;}
void stampa() { for (int i = 0 ; i < num_elem ; i++) cout<<a[i]<<" " ; }

main(){ inserisci(0) ; stampa() ; inserisci(4) ; stampa() ; }
```

 - a) il programma stampa **0 0 4**
 - b) nessuna delle altre risposte è vera
 - c) il programma contiene un errore di *memory leak*
 - d) il programma contiene un errore di gestione della memoria diverso dal *memory leak*

4. (3, -.5) Tenendo presente eventuali problemi di conversione (il tipo **unsigned int** è gerarchicamente superiore al tipo **int**) e di rappresentabilità, il seguente frammento di codice
- ```
int delta_w; unsigned int wsum;
cin>>delta_w>>wsum;
if (wsum - delta_w >= 0) cout<<"maggiore"<<endl ;
```
- a) Non stampa nulla se **delta\_w** è positivo e strettamente maggiore di **wsum**  
 b) Stampa **maggiore** se **delta\_w** è negativo e **wsum** contiene un valore negativo  
 c) Stampa **maggiore** qualsiasi sia la coppia di valori interi venga inserita dall'utente  
 d) Nessuna delle altre risposte è vera

**PARTE 2 – (POSSIBILI) RISPOSTE MULTIPLE -  
 Ogni domanda può avere da una a quattro risposte CORRETTE.**

- Ogni risposta esatta viene calcolata: +1
  - Ogni risposta errata viene calcolata: -0.5
  - Una risposta lasciata in bianco viene calcolata: 0
5. Data una sequenza di N elementi tutti dello stesso tipo
- a) Se la sequenza è memorizzata in *array* e si vuole leggere il valore di un elemento di cui si conosce solo la posizione nella sequenza (si sa che è l'elemento i-esimo, con  $i < N$ ), allora la modifica può essere effettuata a costo  $O(1)$
- b) Se la sequenza è memorizzata in una lista semplice e si vuole leggere il valore di un elemento di cui si conosce solo la posizione nella sequenza (si sa che è l'elemento i-esimo, con  $i < N$ ), allora la modifica può essere effettuata a costo  $O(1)$
- c) Se la sequenza è memorizzata in un *array* utilizzando un terminatore per delimitare la fine della sequenza stessa, non è necessario mantenere ulteriori informazioni per poter accedere sequenzialmente a tutti gli elementi della sequenza
- d) Aggiungere un elemento in testa alla sequenza ha costo  $O(N)$  se la sequenza è implementata mediante una lista semplice
6. Dato il seguente programma
- ```
bool fun(int &a) {
if (a > 10) { ++a ; return true ; } return false ; }
main() {
int m = 10 ; if (m < 30 && fun(m)) cout<<(m*2) ; else cout<<m ; }
```
- a) Se eseguito, il programma stampa 10
 b) Se eseguito, il programma stampa 20
 c) Se eseguito, il programma stampa 22
 d) Il valore della variabile **m** non è modificato in conseguenza della invocazione della funzione **fun** all'interno del **main**
7. Quali delle seguenti sequenze di due istruzioni definiscono una variabile e le assegnano correttamente il valore 3.5?
- a) `a = 3.5 ; float a ;`
 b) `const float a; a = 3.5 ;`
 c) `int a ; a = 3.5 ;`
 d) Nessuna delle precedenti sequenze

8. Data la seguente stringa rappresentata mediante un *array* di caratteri
`char stringa[20];`
- a) per ottenere la stringa vuota basta assegnare `'\0'` al primo elemento dell'*array*;
 - b) la lunghezza massima che può avere la stringa è strettamente minore di 20;
 - c) se *i* è l'indice dell'elemento dell'*array* contenente il terminatore `'\0'`, allora ciascuno degli elementi dell'*array* di indice compreso tra **0** ed **i-1** contiene la rappresentazione grafica di uno dei caratteri della stringa;
 - d) riducendo la lunghezza della stringa memorizzata nella variabile **stringa**, si riduce lo spazio occupato in memoria dalla variabile stessa.
9. Quali delle seguenti affermazioni sono vere?
- a) Un file binario può contenere sequenze di byte uguali a quelle che possono essere contenute in un file di testo
 - b) Un file di testo non contiene numeri, ma rappresentazioni grafiche di caratteri (alcuni dei quali possono eventualmente essere cifre numeriche)
 - c) Nessuna delle altre risposte è vera
 - d) Se si memorizza un array di caratteri in un file binario mediante scrittura non formattata, allora il file risultante contiene la sequenza dei codici dei caratteri contenuti nell'*array*
10. Dato il seguente programma:
- ```
void fun(int) ;
int g = 10 ;
main() { int g = 5 ; fun(g) ; cout<<g ; }
void fun(int a){ cout<<(g++)<<" " ; }
```
- a) Se eseguito, il programma stampa 10 11
  - b) Se eseguito, il programma stampa 10 5
  - c) All'interno della funzione **fun** si stampa e si modifica il valore della variabile **g** definita nella funzione **main**
  - d) All'interno della funzione **main** non è visibile la variabile globale **g**

**PARTE 3 – DOMANDE APERTE –**

- **Una risposta esatta fa acquisire il punteggio positivo riportato a fianco della domanda**
- **Una risposta errata può eventualmente causare una penalità che dipende dalla gravità dell'errore**
- **Una risposta lasciata in bianco viene calcolata: 0**

11. **(5 pt)** Descrivere il tipo enumerato in non più di sei righe.

12. **(5 pt)** Scrivere una funzione che prende in ingresso un vettore contenente le coordinate di una sequenza di segmenti consecutivi in uno spazio bidimensionale, e ritorna la somma delle lunghezze dei segmenti. Ad esempio, nel caso di due segmenti consecutivi, il vettore delle coordinate potrebbe essere  $[(1, 2), (3, 2), (4, 1)]$ , e la somma delle lunghezze sarebbe 3.41. Assumere che tutti gli header file necessari siano stati inclusi.



# Programmazione I

## Prova scritta - 2 luglio 2013

Nome: \_\_\_\_\_ Cognome: \_\_\_\_\_

Matricola: \_\_\_\_\_

Indicare le risposte corrette apponendo una croce nella casella corrispondente. Per superare la prova bisogna aver raggiunto almeno **9** punti nelle domande a risposta singola/multipla, ed almeno **15** complessivamente. **Questa è l'unica pagina che dovete consegnare.** Per comodità avete anche un copia di questa pagina per calcolare il voto da sole/soli durante la correzione.

|    | Risposte |   |   |   | Punti/<br>Penalità |      |
|----|----------|---|---|---|--------------------|------|
|    | A        | B | C | D |                    |      |
| 1  |          |   |   |   | 2                  | -0,5 |
| 2  |          |   |   |   | 2                  | -0,5 |
| 3  |          |   |   |   | 4                  | -0,5 |
| 4  |          |   |   |   | 3                  | -0,5 |
| 5  |          |   |   |   |                    |      |
| 6  |          |   |   |   |                    |      |
| 7  |          |   |   |   |                    |      |
| 8  |          |   |   |   |                    |      |
| 9  |          |   |   |   |                    |      |
| 10 |          |   |   |   |                    |      |

**Risposta alla domanda 11 (5 pt):**

**Risposta alla domanda 12 (5 pt):**



# Programmazione I

## Prova scritta - 2 luglio 2013

Nome: \_\_\_\_\_ Cognome: \_\_\_\_\_

Matricola: \_\_\_\_\_

**Indicare le risposte corrette apponendo una croce nella casella corrispondente. Per superare la prova bisogna aver raggiunto almeno 9 punti nelle domande a risposta singola/multipla, ed almeno 15 complessivamente. Usate questa copia per calcolare il voto da sole/soli durante la correzione.**

|    | Risposte |   |   |   | Punti/<br>Penalità |      |
|----|----------|---|---|---|--------------------|------|
|    | A        | B | C | D |                    |      |
| 1  |          |   |   |   | 2                  | -0,5 |
| 2  |          |   |   |   | 2                  | -0,5 |
| 3  |          |   |   |   | 4                  | -0,5 |
| 4  |          |   |   |   | 3                  | -0,5 |
| 5  |          |   |   |   |                    |      |
| 6  |          |   |   |   |                    |      |
| 7  |          |   |   |   |                    |      |
| 8  |          |   |   |   |                    |      |
| 9  |          |   |   |   |                    |      |
| 10 |          |   |   |   |                    |      |

**Risposta alla domanda 11 (5 pt):**

**Risposta alla domanda 12 (5 pt):**