

Programmazione I

Prova scritta - 4 luglio 2016 - 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. (3, -0.5) Dato il seguente programma, e ricordando che il tipo **unsigned int** è gerarchicamente superiore al tipo **int**,

```
main() {
    int i ; cin>>i ; unsigned int u = 1 ;
    cout<<u+i ;
}
```

- a) per qualsiasi numero intero, positivo o negativo, immesso dall'utente, il programma stampa un numero non negativo
- b) se l'utente immette **-1** da *stdin*, il programma viene terminato forzatamente
- c) se l'utente immette **-2** da *stdin*, il programma stampa **-1**
- d) nessuna delle altre risposte è vera

2. (3, -0.5) Dato il seguente programma:

```
int global ;
void fun(int &local){ local-- ; cout<<local<<" "<<global ; }
main() { int &ref = global; ref = 2 ; fun(ref) ; }
```

- a) Se eseguito, il programma stampa **1 2**
- b) Se eseguito, il programma stampa 1 1
- c) Nessuna delle altre risposte è vera
- d) Il programma contiene un errore di gestione della memoria

3. (2, -0.5) Dato il seguente programma

```
main() {
    int i = 0 ; cin>>i ;
    if (i == '2') cout<<"Secondo" ;
    else cout<<"Altro" ;
}
```

- a) Affinché il programma stampi **Secondo**, può essere necessario che l'utente immetta un valore diverso da **2** da *stdin*
- b) Se l'utente immette **2** da *stdin* il programma stampa **Secondo**
- c) Se l'utente immette **z** da *stdin* il programma entra in un ciclo infinito
- d) Nessuna delle altre risposte è vera

4. (3, -0.5) Il seguente frammento di codice:

```
for (int i=5; i >0 ; i--) { if (i==3) i++; cout<<i*2<<" " ;}
```

- a) Stampa 10 8 6
- b) **Contiene un ciclo infinito**
- c) Nessuna delle altre risposte è vera
- d) Stampa i numeri pari 10 8 8 6 4 2

**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 lettura 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 lettura 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) A seconda del tipo degli elementi (uguale per tutti gli elementi) della sequenza e del valore di N, memorizzare la sequenza in una lista può essere più efficiente, in termini di occupazione della memoria, di memorizzare la sequenza in un *array* con terminatore (ossia un array in cui si utilizza un terminatore per delimitare la fine della sequenza)
6. Dato il seguente programma
- ```
int g = 1 ;

int fun(char &g)
{ return 2 * (++g); }

main()
{
 char m = 1 ; char n = static_cast<char>(2.1) + fun(m) ;
 cout<<static_cast<int>(m * n) ;
}
```
- a) Se eseguito, il programma stampa **12**
  - b) Nella inizializzazione della variabile **n**, la conversione **static\_cast<char>(2.1)** causa perdita di informazione
  - c) Nell'ultima istruzione del **main**, la conversione esplicita causa perdita di informazione
  - d) **Quando invocata nel main, la funzione fun non modifica la variabile globale g**

7. Dato il seguente programma:

```
int fun(int b[], int n)
{ int s = 0 ;
 for (int i = 0 ; i < n ; i++) { s += b[i] ; }
 delete [] b ;
 return s ; }

main()
{ int *a = new int[2] ; a[0] = 2 ; a[1] = 4 ;

 cout<<fun(a, 2)<<" " ;
 for (int i = 0 ; i < 2 ; i++) cout<<a[i]<<" " ; }
```

- a) il programma stampa **6 2 4**
- b) **nella funzione fun, all'interno del ciclo for non si modifica l'array dinamico allocato con l'operatore new all'inizio della funzione main**
- c) **quando l'esecuzione arriva all'inizio del ciclo for nella funzione main, il puntatore a non punta ad un oggetto correttamente allocato in memoria**
- d) **l'esecuzione dell'istruzione delete [] b ; nella funzione fun dealloca l'array dinamico puntato dal parametro b**

### 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**

8. (6 pt) Descrivere sintassi, semantica e vantaggi del tipo *enumerato*, in non più di otto righe, più eventuali frammenti di codice o ulteriori righe scritte in una qualsiasi notazione (a supporto della descrizione della sintassi).



9. (6 pt) Senza utilizzare funzioni o oggetti di libreria per l'output formattato all'interno di stringhe, scrivere una funzione che prenda in ingresso un numero intero e crei e ritorni una nuova stringa contenente la rappresentazione in base 10 del numero, preceduta dal segno del numero. Se, ad esempio, viene passato il numero 34, al funzione ritorna la stringa "+34", mentre se viene passato il numero -278, la funzione ritorna la stringa "-278". Si ottiene il punteggio massimo se si fa l'assunzione che il numero passato possa avere un numero arbitrariamente grande di cifre, e si gestisce correttamente tale assunzione.



# Programmazione I

## Prova scritta - 4 luglio 2016

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 una copia di questa pagina per calcolare il voto da sole/soli durante la correzione.

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

**Risposta alla domanda 8 (6 pt):**

**Risposta alla domanda 9 (6 pt):**



# Programmazione I

## Prova scritta - 4 luglio 2016

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 |          |   |   |   | 3                  | -0,5 |
| 2 |          |   |   |   | 3                  | -0,5 |
| 3 |          |   |   |   | 2                  | -0,5 |
| 4 |          |   |   |   | 3                  | -0,5 |
| 5 |          |   |   |   |                    |      |
| 6 |          |   |   |   |                    |      |
| 7 |          |   |   |   |                    |      |

**Risposta alla domanda 8 (6 pt):**

**Risposta alla domanda 9 (6 pt):**