

# Programmazione I

## Prova scritta - 22 gennaio 2014 - 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, -.5) Assumendo: 1) che nel seguente programma l'operazione di apertura del file in lettura nella funzione **fun2** abbia successo, e 2) che nel file sia correttamente memorizzato l'*array* scritto dalla funzione **fun1**

```
void fun1(const char b[], int m){ ofstream f("numeri");
        f.write(b, m * sizeof(char)) ; }
void fun2(int c[], int n){ ifstream f("numeri");
        for (int i = 0 ; i < n ; i++)
            if (!(f>>c[i])) break ; }

main() {
    int c[] = {2, 3}; char b[] = {'6', ' ', '2'};
    fun1(b, 3) ;
    fun2(c, 2) ;
    for (int i = 0 ; i < 2 ; i++) cout<<c[i]<<" " ;
}
```

- a) il programma si interrompe in un qualche punto prima di iniziare l'esecuzione del ciclo **for** all'interno del **main**
- b) il programma stampa 2 3
- c) il programma stampa 6 2
- d) nessuna delle altre risposte è corretta

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

```
main(){ int i = 1 , a[3] = {10} ;
    do { cin>>a[i] ;
        if (a[i] != a[i-1] + 1)
            continue ;
        else
            i++ ; } while(i < 3) ;
    for(int j = 0 ; j < 3 ; j++)
        cout<<a[j]<<" " ;
}
```

- a) Se eseguito per intero, nel ciclo **for** finale il programma stampa **10 11 12**
- b) Il programma viene terminato forzatamente prima dell'inizio del ciclo **for** finale
- c) Nessuna delle altre risposte è vera
- d) Il programma contiene un errore di gestione della memoria
3. (+3, -0.5) Il seguente frammento di codice:
- ```
char c; cin>>c;
if (static_cast<int>(c) > 60)
    cout<<"Maggiore" ;
```
- a) stampa o meno **Maggiore** a seconda del valore della costante carattere 'c'
- b) stampa **Maggiore** per ogni valore maggiore di 60 correttamente memorizzato in c
- c) contiene un errore di gestione della memoria
- d) nessuna delle altre risposte è corretta

## 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**
4. Dato un qualsiasi programma in C/C++:
- a) A tempo di compilazione, il compilatore decide in che punto dello spazio di indirizzamento collocare le variabili locali delle funzioni
- b) A tempo di compilazione, il compilatore decide in che punto dello spazio di indirizzamento collocare le variabili globali
- c) Ogni volta che viene creato un record di attivazione per una data funzione, ciascuna variabile locale alla funzione viene collocata sempre nella stessa posizione all'interno di tale record di attivazione
- d) Nel codice macchina generato dal compilatore, l'unica informazione dinamica che è necessario conoscere per poter accedere alle variabili locali di una funzione è l'indirizzo del record di attivazione della funzione stessa
5. Assumendo che i valori di tipo **double** siano rappresentati, mediante lo standard IEEE 754 in base 2, con una precisione di 53 cifre in base 2:
- a) Un qualsiasi numero reale che sia strettamente maggiore di zero e minore di 1, e che, nella sua rappresentazione in base 2, abbia un numero di cifre dopo la virgola inferiore a 53, è rappresentabile in modo esatto mediante il tipo **double**.
- b) Un qualsiasi numero reale che sia strettamente maggiore di zero e minore di 1, e che, nella sua rappresentazione in base 10, abbia una sola cifra dopo la virgola, è rappresentabile in modo esatto mediante il tipo **double**.
- c) Mediante il tipo di dato **double** si può rappresentare in modo esatto qualsiasi numero reale compreso nell'intervallo di rappresentabilità del tipo di dato stesso.
- d) Esistono numeri strettamente maggiori di zero e minori di 1, e con più di 53 cifre dopo la virgola nella loro rappresentazione in base 2, che si possono rappresentare mediante il tipo **double**.

6. Dato il seguente programma:

```
void inizializza(int a[][3], int N)
{ for(int i = 0 ; i < N ; i++)
  for(int j = 0 ; j < 3 ; j++)
    a[i][j] = 1 ; }
main() { int c[2][3] ; inizializza(c, 2) ; }
```

- La matrice *c* è vuota, ossia i suoi elementi non hanno nessun valore, quando viene passata alla funzione *inizializza*, che poi provvederà a dare un valore a ciascun elemento
- In quanto all'accesso agli elementi della matrice *a* all'interno della funzione *inizializza*, conoscere il numero di colonne della matrice serve solo per evitare accessi fuori dalla matrice stessa
- In quanto all'accesso agli elementi della matrice *a* all'interno della funzione *inizializza*, conoscere il numero di righe della matrice serve solo per evitare accessi fuori dalla matrice stessa
- Dopo l'invocazione della funzione *inizializza*, tutti gli elementi della matrice *c* contengono il valore 1

7. Dato un programma scritto in linguaggio C/C++

- Il tempo necessario per inizializzare il contenuto di un record di attivazione aumenta all'aumentare del numero e delle dimensioni dei parametri formali
- Il record di attivazione di ogni funzione contiene (oltre ad altre informazioni) le variabili globali a cui accede la funzione
- Per deallocare un record di attivazione è necessario reinizializzare il contenuto delle celle di memoria precedentemente occupate dal record stesso
- Il tempo necessario per inizializzare il contenuto di un record di attivazione aumenta all'aumentare del numero e delle dimensioni delle variabili locali inizializzate

8. Dato il seguente programma:

```
1:int a = 2;
2:
3:main()
4:{
5:    for (int i = 4 ; i > 2 ; i--) {
6:        int a = i / 2 ;
7:        cout<<static_cast<int>(a * i)<<" ";
8:    }
9:    cout<<static_cast<int>(a)<<" ";
10:}
```

- il programma stampa 8 3 1
- la variabile *a* definita alla riga 1 non è visibile solo in parte della funzione **main**
- il programma stampa 8 3 2
- la variabile *i* è visibile anche alla riga 9

9. Dato un puntatore **int \*p**, che punta ad un *array* di **int** allocato dinamicamente in memoria, l'istruzione: **p=0** ;

- fa sì che *p* non punti più ad alcun *array* in memoria dinamica;
- anche se *p* non era l'unico riferimento alla zona di memoria, impedisce di eliminare successivamente l'oggetto dalla memoria;
- non elimina l'*array* dalla memoria dinamica;
- modifica il valore di altri puntatori che puntano allo stesso *array*.

### 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**

10. **(5 pt)** Descrivere sintassi e semantica per il tipo **char** e le costanti di tale di tipo, in non più di sei righe.

11. **(6 pt)** Scrivere una funzione che prende in ingresso un vettore di interi, che si assume essere ordinato in senso crescente e che può contenere elementi ripetuti. Un possibile esempio di vettore in ingresso è: [3, 4, 4, 4, 6, 10, 10]. La funzione crea e restituisce un vettore di coppie di interi, contenente una coppia per ogni valore distinto presente nel vettore di ingresso, ed in cui ciascuna coppia contiene a sua volta il valore dell'elemento stesso ed il numero di occorrenze di tale elemento nel vettore di ingresso. Ad esempio, se la funzione prende in ingresso il vettore di cui all'esempio precedente, allora ritorna il vettore di coppie [(3, 1), (4, 3), (6, 1), (10, 2)].



# Programmazione I

## Prova scritta - 22 gennaio 2014

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 |          |   |   |   | 3                  | -0,5 |
| 2 |          |   |   |   | 3                  | -0,5 |
| 3 |          |   |   |   | 3                  | -0,5 |
| 4 |          |   |   |   |                    |      |
| 5 |          |   |   |   |                    |      |
| 6 |          |   |   |   |                    |      |
| 7 |          |   |   |   |                    |      |
| 8 |          |   |   |   |                    |      |
| 9 |          |   |   |   |                    |      |

**Risposta alla domanda 10 (5 pt):**

**Risposta alla domanda 11 (6 pt):**



# Programmazione I

## Prova scritta - 22 gennaio 2014

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 |          |   |   |   | 3                  | -0,5 |
| 4 |          |   |   |   |                    |      |
| 5 |          |   |   |   |                    |      |
| 6 |          |   |   |   |                    |      |
| 7 |          |   |   |   |                    |      |
| 8 |          |   |   |   |                    |      |
| 9 |          |   |   |   |                    |      |

**Risposta alla domanda 10 (5 pt):**

**Risposta alla domanda 11 (6 pt):**