

Programmazione I

Prova scritta - 7 giugno 2018 - 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
- (2, -.5) Date le istruzioni `cout<<'4'<<'1'<<endl ;` e `cout<<41<<endl ;` e supponendo che l'operatore di uscita sia configurato per stampare i numeri interi in base 10.
 - Tali istruzioni immettono sullo *stdout* la stessa sequenza di codici carattere dell'istruzione `cout<<4<<1<<endl ;`
 - Nessuna delle altre risposte è vera
 - La seconda istruzione immette su *stdout* un numero minore di byte rispetto alla prima
 - La seconda istruzione immette su *stdout* un numero maggiore di byte rispetto alla prima
 - (3, -.5) Si consideri l'invocazione della seguente funzione, e si supponga che l'array **a** abbia dimensione uguale al valore dell'argomento **size**, e che $v > 0$ e $size > 0$:

```
int fun(int a[], int v, int size)
{
    int count = 0, i = 0 ;
    while (i < size)
        if (a[i++]%v == 0)
            break;
        else
            count++ ;
    return count ;
}
```

 - la funzione ritorna il numero di elementi consecutivi non multipli di **v** a partire dall'inizio dell'array **a**;
 - se l'array non contiene nessun multiplo di **v**, il ciclo **while** è infinito;
 - nessuna delle altre risposte è corretta
 - se l'array **a** contiene almeno un elemento non multiplo di **v** la funzione ritorna l'indice del primo di tali elementi, altrimenti ritorna il valore **size**;
 - (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, -.5) Tenendo presente eventuali problemi di conversione (il tipo **unsigned int** è gerarchicamente superiore al tipo **int**, e qualsiasi operazione tra **unsigned int** ha per risultato un **unsigned int**) e di rappresentabilità, quale della seguenti risposte è vera riguardo al 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 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: -.5
  - Una risposta lasciata in bianco viene calcolata: 0
5. Indicare quali delle seguenti affermazioni sono vere
- a) Dato un programma scritto in un linguaggio che può essere sia compilato che interpretato, l'esecuzione di tale programma da parte di un interprete è tipicamente più lenta dell'esecuzione della versione compilata del programma
- b) Il linguaggio macchina permette di scrivere programmi portabili tra diverse architetture
- c) Il linguaggio macchina non è un linguaggio di alto livello
- d) Un programma in linguaggio macchina è una sequenza di byte da non interpretarsi come sequenza di caratteri
6. 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**

7. Data la seguente stringa rappresentata mediante un *array* di caratteri  
`char stringa[5];`
- per ottenere la stringa vuota basta assegnare `'\0'` al primo elemento dell'*array*, senza bisogno di modificare il valore di alcuno degli elementi successivi
  - la lunghezza massima che può avere la stringa è strettamente minore di 5
  - 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;
  - riducendo la lunghezza della stringa memorizzata nella variabile `stringa`, si riduce lo spazio occupato in memoria dalla variabile stessa.

8. Dato il seguente programma:

```
const int N = 1000 ;
struct ss { int b; char a[N] ; } ;
void fun(ss d) { d.a[N/2] = 1 ; d.b = 3 ; cout<<d.b ; }
main()
{
 ss c ; c.b = 13 ; c.a[N/2] = 2 ;
 fun(c) ; cout<<" "<<c.b<<" "<<static_cast<int>(c.a[N/2]) ;
}
```

- Quando `fun` è invocata il contenuto del parametro attuale `c` è copiato nel parametro formale `d`
- In quanto al campo `a` del parametro attuale `c` passato alla funzione `fun`, l'*array* in esso contenuto è copiato nel campo `a` del parametro formale `d`.
- Un passaggio per riferimento sarebbe stato più conveniente in termini di tempo di esecuzione
- Il programma stampa: **3 13 2**

### 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**

9. (6 pt) Descrivere l'idea alla base e lo schema generale della rappresentazione in virgola mobile dei numeri reali, in non più di otto righe più eventuali schemi (non verrà valutata la quantità ma la qualità di quello che si scrive, ed il tentativo di scrivere in modo estremamente fitto per aumentare la quantità, così come il superamento del numero massimo di righe, comporteranno una penalità).



10. **(5 pt)** Scrivere una funzione che prenda in ingresso un vettore di interi, e ritorni un nuovo vettore con le seguenti caratteristiche: il primo elemento del nuovo vettore è uguale al primo elemento del vettore originale, il nuovo vettore contiene solo elementi presenti anche nel vettore originale (ma può non contenere tutti gli elementi presenti nel vettore originale), il nuovo vettore ha gli elementi ordinati in senso crescente. Ad esempio, dato il vettore [5, 5, 3, 4, 7, 10, 6, 4], la funzione ritorna [5, 5, 6, 7, 10]. La funzione non modifica il vettore originale.



# Programmazione I

## Prova scritta - 7 giugno 2018

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

Risposta alla domanda 9 (6 pt):

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



# Programmazione I

## Prova scritta - 7 giugno 2018

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

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

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