

Programmazione I

Prova scritta - 24 giugno 2015 - 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

```
main()
{
    int i = 10 ; char s[10] ; ifstream f("dati.txt");
    f>>i ; f>>s ;
    cout<<i<<" "<<s<<endl ;
}
```

e supponendo che il file *dati.txt* esista e contenga la sequenza di caratteri **5 cinque** (c'è uno spazio tra il carattere 5 e la stringa *cinque*) seguiti dal carattere *newline*:

- a) Il programma memorizza il valore 5 nella variabile *i*, poi si blocca indefinitamente sull'istruzione `f>>s ;` perché manca il carattere *newline* tra il carattere 2 e la stringa *cinque* all'interno del file *dati.txt*
- b) Nessuna delle altre risposte è corretta
- c) Il programma stampa **5 cinque**
- d) L'*ifstream f* va in stato di errore sull'istruzione `f>>s ;`

2. (2, -0,5) La memoria occupata da un oggetto **X** allocato dinamicamente all'interno di una funzione

- a) è allocata all'inizio dell'esecuzione della funzione;
- b) è liberata se si assegna l'indirizzo di un nuovo oggetto dinamico al puntatore che contiene l'indirizzo di **X** in memoria;
- c) è liberata se viene deallocato il puntatore che contiene l'indirizzo di **X** in memoria;
- d) se non esplicitamente deallocata, è liberata solo al termine del programma.

3. (3, -.5) Data la seguente stringa rappresentata mediante un array

```
char s[10] = "Sol" ;
```

le istruzioni:

```
s[2] = 't' ; s[3] = s[2] ; s[4] = 'o' ; s[5] = '\\0' ;
```

- a) contengono uno o più errori di accesso alla memoria
- b) trasformano la stringa da "Sol" a "Sotto"
- c) non modificano la lunghezza della stringa
- d) nessuna delle altre risposte è vera

4. (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(const int a[], int v, int size)
```

```
{
```

```
    int count = 0, i = 0 ;
```

```
    while (i < size)
```

```
        if (a[i++] % v == 0) continue;
```

```
        else count++ ;
```

```
    return count ;
```

```
}
```

- a) se l'array **a** contiene almeno un elemento multiplo di **v** la funzione ritorna l'indice del primo di tali elementi, altrimenti ritorna il valore **size**;
- b) se l'array non contiene nessun multiplo di **v**, il ciclo **while** è infinito;
- c) la funzione contiene un errore di gestione della memoria;
- d) la funzione ritorna il numero di elementi non multipli di **v** all'interno dell'array **a**.

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. Dato un qualsiasi programma in C/C++:

- a) Il compilatore decide la posizione degli oggetti con classe di memorizzazione statica, all'interno dello spazio di indirizzamento del processo corrispondente a tale programma
- b) Il compilatore decide la posizione degli oggetti con classe di memorizzazione dinamica, all'interno dei record di attivazione delle funzioni in cui tali oggetti sono allocati
- c) Il compilatore decide la posizione degli oggetti con classe di memorizzazione automatica, all'interno dei record di attivazione delle funzioni in cui tali oggetti sono definiti
- d) Il tempo necessario per eseguire il passaggio di un array ad una funzione è indipendente dalle dimensioni dell'array

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 è necessario per conoscere l'indirizzo in memoria di ciascun elemento della matrice
- In quanto all'accesso agli elementi della matrice *a* all'interno della funzione *inizializza*, conoscere il numero di righe della matrice è necessario per conoscere l'indirizzo in memoria di ciascun elemento della matrice
- Dopo l'invocazione della funzione *inizializza*, tutti gli elementi della matrice *c* contengono il valore 1

7. Dato il seguente programma e supponendo che il valore massimo memorizzabile in un **char** sia 127:

```
int fun(int a) { return a ; };
int main() {
  char i = numeric_limits<char>::max() ; double h = 1.3 ;
  const double g = i + fun(h) ;
  cout<<g ;
}
```

- Il programma stampa **128**
- Il programma stampa un numero diverso da **128**
- La somma effettuata nella seconda riga della funzione **main** è di tipo **int**
- C'è perdita di informazione nel passaggio del parametro attuale nella invocazione della funzione **fun** nella seconda riga della funzione **main**

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)** In non più di 10 righe, descrivere il significato di classe di memorizzazione statica, automatica e dinamica (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).

9. (7 pt) Scrivere una funzione che prende in ingresso un array di interi e, in nove invocazioni su dieci, lo lascia inalterato. Invece, in una invocazione su dieci, la funzione modifica l'array eliminando la sequenza di elementi consecutivi finali, dell'array, tali che la somma degli elementi della parte restante dell'array ha valore massimo. Ad esempio, se, in una delle invocazioni in cui la funzione deve modificare l'array, le viene passato [2, -3, 5, -10, 2], la funzione trasforma tale array in [2, -3, 5].

Programmazione I

Prova scritta - 24 giugno 2015

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/ Penalità	
	A	B	C	D		
1					3	-0,5
2					2	-0,5
3					3	-0,5
4					3	-0,5
5						
6						
7						

Risposta alla domanda 8 (6 pt):

Risposta alla domanda 9 (7 pt):

Programmazione I

Prova scritta - 24 giugno 2015

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/ Penalità	
	A	B	C	D		
1					3	-0,5
2					2	-0,5
3					3	-0,5
4					3	-0,5
5						
6						
7						

Risposta alla domanda 8 (6 pt):

Risposta alla domanda 9 (7 pt):