

Programmazione I

Prova scritta - 29 gennaio 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. (3, -.5) Dato il seguente programma

```
int g = 1 ;

double fun() { ++g ; return g + 3 ; }

main()
{
  const int m = 2 ;
  double l = 1 + fun() / m ;
  cout<<static_cast<int>(l * g) ;
}
```

- a) Se eseguito, il programma stampa 7
- b) Se eseguito, il programma stampa 6
- c) Il programma stampa un numero diverso da 7 e da 6
- d) Nessuna delle altre risposte è vera

2. (2, -0.5) Il seguente frammento di codice:

```
for (int i=4; i >0 ; i--) { if (i==3) i = 2; cout<<i*2<<" " ;}
```

- a) Stampa 8 4 2
- b) Contiene un ciclo infinito
- c) Nessuna delle altre risposte è vera
- d) Stampa 8 4 4 2

3. (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)
            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) la funzione ritorna il numero di elementi non multipli di **v** all'interno dell'array **a**
- c) se l'array non contiene nessun multiplo di **v**, il ciclo **while** è infinito
- d) la funzione contiene un errore di gestione della memoria

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. La memoria occupata da un oggetto allocato dinamicamente all'interno di una funzione
- a) è allocata all'inizio dell'esecuzione del programma
 - b) è allocata quando l'istruzione di allocazione viene eseguita
 - c) è liberata quando la funzione termina
 - d) se non esplicitamente deallocata, è liberata solo al termine del programma
5. Dato un programma scritto in linguaggio C/C++
- a) Il tempo necessario per inizializzare il contenuto di un record di attivazione aumenta all'aumentare del numero e delle dimensioni dei parametri formali in esso contenuti
 - b) Il record di attivazione di ogni funzione contiene (oltre ad altre informazioni) le variabili globali a cui accede la funzione
 - c) Per deallocare un record di attivazione è necessario reinizializzare il contenuto delle celle di memoria precedentemente occupate dal record stesso
 - d) In quanto alle variabili locali di una funzione, il record di attivazione della funzione contiene solo quelle inizializzate

6. Data la seguente funzione a cui viene passato un array **a** di dimensione **N**:

```
bool fun(unsigned int a[], int i, unsigned int N) {
    if ( ( i < 0 || i >= N ) && a[i] % 2 == 1 )
        return false ;
    return true ;
}
```

- a) Se il parametro formale **i** è compreso tra 0 ed N-1 (estremi inclusi) la funzione ritorna **true**
- b) Nessuna delle altre risposte è vera
- c) Solo se il parametro formale **i** non è compreso tra 0 ed N-1 (estremi inclusi) la funzione controlla il valore dell'elemento **i**-esimo e ritorna **false** se tale elemento è dispari
- d) Ci sono valori del parametro **i** per i quali la funzione legge al di fuori dell'array **a**

7. Dato il seguente programma:

```
void scrivi(ofstream &f) { f<<"prova" ; f.close() ; }
main()
{
    ofstream of("dati.txt") ;
    scrivi(of) ;
    ifstream is("dati.txt") ;
    char buf[256] ;
    is>>buf ;
    cout<<buf ;
}
```

- a) il programma stampa **prova** su *stdout*
- b) quando si riapre il file *dati.txt* in lettura all'interno della funzione *main* non è garantito che il suo contenuto sia stato aggiornato
- c) quando si riapre il file *dati.txt* in lettura all'interno della funzione *main*, il file contiene una sequenza di byte, ciascuno dei quali contiene a sua volta il codice di uno dei caratteri precedentemente scritti nel file stesso mediante la funzione *scrivi*
- d) anche se il parametro formale *f* della funzione *scrivi* muore in uscita dalla funzione stessa, l'invocazione della funzione *f.close()* nella funzione *scrivi* è comunque necessaria per provocare l'aggiornamento del file *dati.txt*

8. Dato il seguente programma:

```
const int N = 100 ; struct ss { int b; char a[N] ; } ;
void fun(ss & d) { cout<<d.a[N/2]<<" " ; d.b = 4 ; }
main()
{
    ss c ; c.b = 8 ; c.a[N/2] = '2' ; cout<<c.b<<" " ;
    fun(c) ; cout<<c.b ;
}
```

- a) Il passaggio di **c** per riferimento permette a **fun** di modificare **c**
- b) Il programma stampa **8 2 4**
- c) Il passaggio di **c** per riferimento ha come costo la copia nel parametro formale **d** del contenuto di **c** stesso
- d) L'esecuzione di **fun(c)**; ha effetti collaterali sulla porzione della funzione **main** che segue l'invocazione della funzione **fun** stessa

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. (4 pt.) Cosa scrive su *stdout* il seguente programma, supponendo che l'utente immetta *****me** da *stdin*?

```
main()
{
    const int N = 10; char stringa[] = "esatto";
    int i ;
    for (i = 0; i < N ; i++) {
        char c; cin>>c;
        if (c == '*') continue;
        stringa[i] = c;
        if (c == 'e') break;
    }
    stringa[i+1] = '\\0' ;
    cout<<stringa<<endl ;
}
```

10. (6 pt) Scrivere una funzione che prenda in ingresso un vettore di interi senza modificarlo, e restituisca un nuovo vettore contenente solo gli elementi del primo vettore tali che tutti gli elementi che li precedono sono minori di loro. Ad esempio, preso in ingresso il vettore [2, 26, 5, 2, 41, 40], la funzione dovrebbe restituire [2, 26, 41]

Programmazione I

Prova scritta - 29 gennaio 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/ Penalità	
	A	B	C	D		
1					3	-0,5
2					2	-0,5
3					3	-0,5
4						
5						
6						
7						
8						

Risposta alla domanda 9 (4 pt):

Risposta alla domanda 10 (6 pt):

Programmazione I

Prova scritta - 29 gennaio 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/ Penalità	
	A	B	C	D		
1					3	-0,5
2					2	-0,5
3					3	-0,5
4						
5						
6						
7						
8						

Risposta alla domanda 9 (4 pt):

Risposta alla domanda 10 (6 pt):