

**Programmazione I / Informatica generale - Preappello**  
**Prova scritta**  
**17 Dicembre 2007**

NOTA: Si trascuri ogni problema legato al tipo ed al valore di ritorno della funzione **main**. Inoltre, nei programmi è da sottintendere la presenza delle direttive

```
#include <iostream>
using namespace std ;
```

**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 calcolata: 0

1. (+2, -2) Assumendo che le costanti `INT_MIN` e `INT_MAX` contengano, rispettivamente, il più piccolo ed il più grande valore di tipo `int`, il seguente programma:

```
1:  main() {
2:      int x, z ;
3:      cin>>x>>z ;
4:      int y = x * z ;
5:      if (y < INT_MIN || y > INT_MAX)
6:          cout<<"Overflow" ;
7: }
```

- a) Stampa **Overflow** se la moltiplicazione tra `x` e `z` ha generato overflow
- b) Si interrompe prima della riga 5
- c) Non stampa nulla
- d) Nessuna delle precedenti risposte è corretta

2. (+2; -2) Il seguente codice:

```
struct pers {char nome[25]; int anno;} ;
void fun(pers &p, const char n[], int a)
{
    pers d ;
    strcpy(d.nome, n) ;
    d.anno = a;
    p = d ;
}
main()
{
    pers z = {"michele", 1985};
    fun(z, "mario", 1985) ;
    cout<<z.nome<<" "<<z.anno<<endl ;
}
```

- a) Contiene un errore che causa corruzione della memoria
- b) Stampa **mario 1985**
- c) Stampa **michele 1985**
- d) Nessuna delle precedenti risposte è corretta

3. (+2, -1) Dato il seguente programma:

```
void fun(const int a[], int n)
{
    for (int i = 0 ; i < n ; i++)
        cin>>a[i] ;
}
main()
{
    int c[3] ;
    fun(c, 3) ;
}
```

- a) La funzione **fun** modifica solo una copia locale dell'array che le viene passato
- b) La funzione **fun** modifica l'array **c** che le viene passato
- c) La variabile **c** ha classe di memorizzazione statica
- d) Nessuna delle precedenti risposte è corretta

4. (+2, -1) Il seguente programma:

```
int a = 10, b = 20 ;
int &fun(bool f)
{
    if (f) return a ;
    else return b ;
}
main()
{
    int &c = fun(false) ;
    cout<<c;
}
```

- a) Stampa 10
- b) Stampa 20
- c) Contiene un errore di gestione della memoria
- d) Nessuna delle precedenti risposte è corretta

5. (2, -1) Dato il seguente programma:

```
main()
{
    int i = 10 ;
    do {cin>>i ;
        cout<<i<<" , " ; }
    while(cin) ;
}
```

se l'utente inserisce da *stdin* i caratteri **2** e **3**, separati da uno spazio, quindi preme Invio, ed infine preme CTRL+D (su sistema UNIX)

- a) il programma stampa su *stdout* **2, 3,**
- b) il programma stampa su *stdout* **2, 3, 10,**
- c) il programma stampa su *stdout* **2, 3, 3,**
- d) nessuna delle precedenti risposte è corretta

6. (3, -1) Assumendo che il tipo `int` sia memorizzato su 4 byte, il seguente programma:

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

main() {
    const int N = 3 ;
    int b[N] = {1, 4, 2}, c[] = {1, 1, 1};
    fun1(b, 3) ;
    fun2(c, 3) ;
    for (int i = 0 ; i < N ; i++) cout<<c[i]<<" " ;
}
```

- a) si interrompe prima di iniziare l'esecuzione al ciclo `for` all'interno del `main`
- b) stampa 1 1 1
- c) stampa 1 4 2
- d) nessuna delle precedenti risposte è corretta

**PARTE 2 – (POSSIBILI) RISPOSTE MULTIPLE -**  
**Ogni domanda può avere una o più risposte CORRETTE.**

- Ogni risposta esatta viene calcolata: +1
- Ogni risposta errata viene calcolata: -0.5
- Una risposta lasciata in bianco viene calcolata: 0

7. Dato il seguente programma:

```
main()
{
    ofstream f("nome.txt") ; // apro il file in scrittura
    f<<1<<2<<endl ;
    f<<3 ;
    ifstream f2("nome.txt") ; // riapro lo stesso file in lettura
    int i ;
    while(f2>>i) ;
        cout<<i<<" , " ;
}
}
```

- a) al termine del programma, il file **nome.txt** non è un file binario contenente i numeri 1, 2 e 3
  - b) al termine del programma, il file **nome.txt** è un file di testo contenente i numeri 1, 2 e 3
  - c) il programma stampa almeno **1, 2,**
  - d) quando viene riaperto in lettura alla creazione dell'*ifstream* **f2**, il file **nome.txt** può non contenere tutti i numeri che sono stati immessi nello *ofstream* **f**
8. Dato il seguente programma:
- ```
struct stringa {char s[10]; } ;
stringa fun(stringa t)
{
    strcpy(t.s, "pippo") ;
    return t ;
}
main()
{
    stringa c = {"minnie"}, d = {"pluto"};
    d = fun(c) ;
    cout<<d.s<<" "<<c.s ;
    d = c ;
    cout<<d.s<<" "<<c.s ;
}
}
```
- a) La prima istruzione di scrittura su *stdout* stampa **pippo minnie**
  - b) La seconda istruzione di scrittura su *stdout* stampa **minnie minnie**
  - c) La prima istruzione di stampa su *stdout* stampa **pippo pippo**
  - d) Nessuna delle precedenti risposte è vera

9. Dato il seguente programma:

```
int fun(int b[], int n)
{
    int s = 0 ;
    for (int i = 0 ; i < n ; i++) {
        s += b[i] ; b[i] *= 2 ;}
    delete [] b ;
    return s ;
}

main()

{
    int *a = new int[2] ;
    a[0] = 3 ; a[1] = 4 ;

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

- a) il programma stampa 7 3 4
- b) nella funzione **fun**, all'interno del ciclo **for** si modifica l'array dinamico allocato con l'operatore **new** all'inizio della funzione **main**
- c) nella funzione **main**, all'interno del ciclo **for** si stampano gli elementi dell'array dinamico allocato con l'operatore **new** all'inizio della funzione **main**
- d) il programma contiene un errore di gestione della memoria

10. Dato il seguente programma

```
bool fun(int &a)
{
    if (a > 10) {
        ++a ;
        return true ;
    }
    return false ;
}

main()
{
    int m = 22 ;

    if (m < 30 && fun(m))
        cout<<(m*2) ;
    else
        cout<<m ;
}
```

- a) Se eseguito, il programma stampa 46
- b) Se eseguito, il programma stampa 22
- c) A parte l'inizializzazione, il valore della variabile **m** non è mai modificato
- d) Il valore della variabile **m** è modificato in conseguenza della invocazione della funzione **fun** all'interno del **main**

11. Dato il seguente programma:

```
1:  const int M = 10 ;
2:  void fun(char a[], int n)
3:  {
4:      n = n > M - 1 ? M - 1 : n ;
5:      for (int i = n - 1 ; i < M ; i++)
6:          a[i] = ' ' ;
7:  }
8:  main()
9:  {
10: char b[M] = "paolo" ;
11: fun(b, 3) ;
12: cout<<b ;
13: }
```

- a) il programma stampa **pa**
- b) la stringa memorizzata in **b** ha lunghezza 5 prima dell'invocazione della funzione **fun** alla riga 11
- c) l'istruzione alla riga 12 può stampare un numero di caratteri maggiore di 9
- d) al di là dell'invocazione fatta nel **main**, si ha che, in generale, la funzione **fun** può causare corruzione della memoria durante la sua esecuzione, anche se le è stato passato un array di M elementi correttamente allocato in memoria

12. Dato il seguente programma:

```
1: float b = 3.5;
2: float fun(float &a)
3: {
4:     int i ;
5:     for (i = 0 ; i < 2 ; i++)
6:         a *= 2 ;
7:     b++ ;
8:     return b+i ;
9: }
10:
11: main()
12: {
13:     float b = 1.5 ;
14:     float a = fun(b) ;
15:     cout<<a + b<<endl ;
16: }
```

- a) la funzione **fun** modifica il valore della variabile **b** definita alla riga 13
- b) il parametro formale **a** definito alla riga 2 è visibile alla riga 15
- c) la variabile **i** definita alla riga 4 non è visibile alla riga 6
- d) il programma stampa 12.5

### PARTE 3 – DOMANDE APERTE –

- Una risposta esatta fa acquisire il punteggio positivo riportato a fianco della domanda
- Una risposta errata causa una penalità che dipende dalla gravità dell'errore, ed al più uguale al punteggio negativo riportato a fianco della domanda
- Una risposta lasciata in bianco viene calcolata: 0

13. (3 pt.) Cosa stampa a video il seguente codice?

```
struct s {int a, b ;} ;

int fun(s t, s &q)
{
    q = t ;
    t.a = q.b ;
    t.b = q.a ;
    return t.b ;
}

main()
{
    s p = {1, 2}, q = {3, 4} ;
    p.b = fun(p, q) ;
    cout<<p.a<<" "<<p.b<<endl ;
    cout<<q.a<<" "<<q.b<<endl ;
}
```

14. (4 pt.) Scrivere una funzione che prenda in ingresso un vettore di interi, e ruoti il contenuto del vettore di una posizione verso destra. Ad esempio, se il vettore è di 3 elementi, con valori {1, 5, 2}, dopo la rotazione il contenuto del vettore diviene {2, 1, 5}.





**Programmazione I – Preappello**  
**Prova Scritta**  
**17 Dicembre 2007**

Nome: \_\_\_\_\_ Cognome: \_\_\_\_\_

Matricola: \_\_\_\_\_ Corso di Laurea: \_\_\_\_\_

**Avete due copie di questa pagina, usatene una per calcolare il voto da soli dopo la correzione. Indicare le risposte corrette apponendo una croce nella casella corrispondente.**

|    | Risposte |   |   |   | Punti/<br>Penalità |    |
|----|----------|---|---|---|--------------------|----|
|    | A        | B | C | D |                    |    |
| 1  |          |   |   |   | 2                  | -2 |
| 2  |          |   |   |   | 2                  | -2 |
| 3  |          |   |   |   | 2                  | -1 |
| 4  |          |   |   |   | 2                  | -1 |
| 5  |          |   |   |   | 2                  | -1 |
| 6  |          |   |   |   | 3                  | -1 |
| 7  |          |   |   |   |                    |    |
| 8  |          |   |   |   |                    |    |
| 9  |          |   |   |   |                    |    |
| 10 |          |   |   |   |                    |    |
| 11 |          |   |   |   |                    |    |
| 12 |          |   |   |   |                    |    |

**Risposta alla domanda 13:**

**Risposta alla domanda 14:**



**Programmazione I – Preappello**  
**Prova Scritta**  
**17 Dicembre 2007**

Nome: \_\_\_\_\_ Cognome: \_\_\_\_\_

Matricola: \_\_\_\_\_ Corso di Laurea: \_\_\_\_\_

**Indicare le risposte corrette apponendo una croce nella casella corrispondente.**

|    | Risposte |   |   |   | Punti/<br>Penalità |    |
|----|----------|---|---|---|--------------------|----|
|    | A        | B | C | D |                    |    |
| 1  |          |   |   |   | 2                  | -2 |
| 2  |          |   |   |   | 2                  | -2 |
| 3  |          |   |   |   | 2                  | -1 |
| 4  |          |   |   |   | 2                  | -1 |
| 5  |          |   |   |   | 2                  | -1 |
| 6  |          |   |   |   | 3                  | -1 |
| 7  |          |   |   |   |                    |    |
| 8  |          |   |   |   |                    |    |
| 9  |          |   |   |   |                    |    |
| 10 |          |   |   |   |                    |    |
| 11 |          |   |   |   |                    |    |
| 12 |          |   |   |   |                    |    |

**Risposta alla domanda 13:**

**Risposta alla domanda 14:**