

# Programmazione I

## Prova scritta - 15 febbraio 2012 - 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) Il seguente programma:

```
struct pers {char nome[25]; int anno;} ;
pers fun(pers p, const char *s)
{ strcpy(p.nome, s) ; p.anno = 1986; return p ; }
main()
{
  pers z = {"luigi", 1985};
  pers y = fun(z, "lorenzo") ;
  cout<<z.nome<<" "<<y.nome ;
  strcpy(z.nome, "anna") ; cout<<" "<<y.nome<<" "<<y.anno ;
}
```

- a) Stampa `lorenzo lorenzo anna 1986`
- b) Stampa `luigi lorenzo lorenzo 1985`
- c) Stampa `lorenzo lorenzo lorenzo 1986`
- d) Nessuna delle altre risposte è corretta

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

```
main()
{
  int i = 1 , a[3] = {0} ;
  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 il programma stampa `0 1 2`
- 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) Dato il seguente programma:

```
main() {
    int i ; cin>>i ; unsigned int u = 10 ;
    cout<<u+static_cast<unsigned int>(i) ;
}
```

- per qualsiasi numero intero, positivo o negativo, immesso dall'utente, il programma stampa un numero non negativo
- se l'utente immette -10 da *stdin*, il programma viene terminato forzatamente
- se l'utente immette -20 da *stdin*, il programma stampa -10
- nessuna delle altre risposte è vera

4. (3, -0.5) 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 a[], int n){ ofstream f("nome");
    f.write(a,sizeof(char) * n) ; }
void fun2(int a[], int n){ ifstream f("nome");
    for (int i = 0 ; i < n ; i++)
        if (!(f>>a[i])) break ; }
main() {
    char b[] = {'1', ' ', '4'}; int c[] = {2, 2};
    fun1(b, 3) ;
    fun2(c, 2) ;
    for (int i = 0 ; i < 2 ; i++) cout<<c[i]<<" " ;
}
```

- il programma si interrompe in un qualche punto prima di iniziare l'esecuzione del ciclo **for** all'interno del **main**
- il programma contiene un errore di gestione della memoria
- il programma stampa 1 4
- nessuna delle altre risposte è corretta

5. (2, -0.5) Dato il seguente programma

```
main() {
    int i = 0 ; cin>>i ;
    if (i == '1') cout<<"Primo" ;
    else cout<<"Secondo" ;
}
```

- Se l'utente non immette da *stdin* la sequenza di cifre che rappresenta il codice del carattere 1, il programma stampa **Secondo**
- Se l'utente immette 1 da *stdin* il programma stampa **Primo**
- Se l'utente immette z da *stdin* il programma entra in un ciclo infinito
- 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: -0.5
  - Una risposta lasciata in bianco viene calcolata: 0
6. In C/C++, una stringa implementata mediante *array* di caratteri:
- va sempre allocata dinamicamente;
  - può essere copiata in un'altra stringa con una istruzione di assegnamento

- c) nessuna delle altre risposte è vera;
- d) deve sempre essere terminata dal carattere di fine stringa '\0' .

7. La seguente istruzione:

```
cout<<static_cast<char>('b' + '3' - 'c') ;
```

- a) stampa 2
- b) può causare problemi di *overflow*
- c) immette il codice di un carattere stampabile sullo *stream* di uscita standard (**cout**)
- d) nessuna delle altre risposte è corretta

8. Dato il seguente programma:

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

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

9. Dato il seguente programma:

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

```
main()
{
    int * const a = fun(3) ;
    for (int i = 0 ; i < 3 ; i++)
        cout<<a[i]<<" " ;
}
```

- a) il programma stampa 0 1 2
- b) la funzione **fun** ritorna l'indirizzo del primo elemento di un *array* che non muore alla fine dell'esecuzione della funzione stessa
- c) nessuna delle altre risposte è vera
- d) il programma contiene un errore di gestione della memoria

10. Dato il seguente programma

```
int fun(int &g) { return 2 * (++g); }

int g = 2 ;
main() {
    char n = static_cast<int>(1.4) * fun(g) ;
    cout<<static_cast<int>(g * n) ;
}
```

- Se eseguito, il programma stampa 18
- Nell'ultima istruzione del **main**, la conversione `static_cast<int>(g * n)` non causa perdita di informazione
- Nessuna delle altre risposte è vera
- La conversione `static_cast<int>(1.4)` causa perdita di informazione

### 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

11. (4 pt) Cosa scrive su *stdout* il seguente programma?

```
int *fun(const int a[], const int b[])
{
    int *d = new int[4] ;
    for (int i = 0 ; i < 2 ; i++) {
        d[2 * i] = a[i] ; d[2 * i + 1] = b[i] ;
    }
    return d ;
}

main() {
    int a[] = {2, 1}, b[] = {4, 5} ;
    int *c = fun(a, b) ;
    for(int i = 0 ; i < 4 ; i++) cout<<c[i]<<" " ;
}
```

**12. (5 pt)** Scrivere una funzione che prenda in ingresso un vettore di interi positivi e ritorni una stringa, creata all'interno della funzione stessa e contenente, per ciascun valore intero minore o uguale a 9 contenuto nel vettore di interi, la cifra (il carattere) che rappresenta tale numero in base 10, e per ciascun valore intero maggiore di 9, la lettera *p* se il numero è pari o la lettera *d* se il numero è dispari. Ad esempio, preso in ingresso il vettore [2, 26, 5, 453, 10, 1], la funzione ritorna la stringa "2p5dp1".



# Programmazione I

## Prova scritta - 15 febbraio 2012

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

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

**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					3	-0,5
3					3	-0,5
4					3	-0,5
5					2	-0,5
6						
7						
8						
9						
10						

**Risposta alla domanda 11 (4 pt):**

**Risposta alla domanda 12 (5 pt):**



# Programmazione I

## Prova scritta - 15 febbraio 2012

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

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

**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					3	-0,5
3					3	-0,5
4					3	-0,5
5					2	-0,5
6						
7						
8						
9						
10						

**Risposta alla domanda 11 (4 pt):**

**Risposta alla domanda 12 (5 pt):**