

Programmazione I / Informatica generale

Prova scritta - 2 Settembre 2008

NOTA: Nei programmi si trascuri ogni problema legato al tipo ed al valore di ritorno della funzione **main** e si sottintenda la presenza delle direttive

```
#include <iostream>
```

```
#include <fstream>
```

```
using namespace std ;
```

Infine, laddove si trovi l'affermazione che un programma o frammento di codice fornisce un certo risultato, è da intendersi che, in accordo alle regole del linguaggio, tale programma o frammento di codice fornisce 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 calcolata: 0

1. (3, -0.5) Dato il seguente programma:

```
main()
{
    int a ; cin>>a ;
    if ( (a > 1) == false )
        cout<<a ;
}
```

- a) Se l'utente immette 1 su *stdin* il programma stampa 1
- b) Il programma contiene un errore logico
- c) Nessuna delle altre risposte è vera
- d) Il programma causa un errore a tempo di compilazione

2. (3, -0.5) Il seguente programma

```
main()
{
    bool finito = false ; int i = 0 ;
    while( finito == false ) {
        cout<<i<<" " ;
        if (i == 2)
            finito = true ;
        cout<<i<<" " ;
        i++ ;
    }
}
```

- a) stampa 0 0 1 1 2 2
- b) stampa 0 0 1 1 2
- c) stampa 0 0 1 1
- d) nessuna delle altre risposte è vera

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

```
void fun(int &) ;
int global ;
main()
{
    global = 3;
    fun(global) ;
}

void fun(int &local)
{
    local++ ;
    cout<<local<<" "<<global ;
}
```

- a) Se eseguito, il programma stampa **4 3**
- b) Se eseguito, il programma stampa 4 4**
- c) Nessuna delle altre risposte è vera
- d) Il programma causa un errore a tempo di compilazione

4. (3, -.5) Quale delle seguenti affermazioni è vera?

- a) Un file binario non può contenere valori uguali a quelli che possono essere contenuti in un file di testo
- b) Un file di testo non contiene numeri, ma rappresentazioni grafiche di caratteri (alcuni dei quali possono eventualmente essere cifre numeriche)
- c) Nessuna delle altre risposte è vera
- d) In conseguenza dell'oggetto in esso memorizzato, un file binario può di fatto contenere una sequenza di valori tutti rappresentanti caratteri**

5. (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 **2 prova** (c'è uno spazio tra il carattere **2** e la stringa *prova*) seguiti dal carattere *newline*:

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

6. Dato il seguente frammento di programma:

```
enum stato {pieno, vuoto} ;
stato v[3] = {pieno, pieno, vuoto} ;
bool fun(int i)
{
    if ( ( i < 0 && i >= 3 ) || v[i] == pieno )
        return false ;
    return true ;
}
```

- a) Se invocata passando il valore 2, la funzione *fun* ritorna *true*
- b) Il frammento di programma contiene un errore logico
- c) Il frammento di programma contiene un errore di accesso alla memoria
- d) Nessuna delle altre risposte è vera

7. Dato il seguente programma

```
int g = 1 ;

double fun(int n)
{ g = g + 1 ; return n * 3 ; }

main()
{
    const double g = 3 ;
    double l = static_cast<int>(fun(g / 2)) / 2 ;
    cout<<static_cast<int>(l * g) ;
}
```

- a) Se eseguito, il programma stampa 3
- b) La conversione esplicita presente nell'espressione a destra dell'inizializzazione nella penultima istruzione della funzione *main* non causa alcuna perdita di informazione
- c) Se non fosse stata presente la conversione esplicita nell'espressione a destra dell'inizializzazione nella penultima istruzione della funzione *main* il programma avrebbe stampato un valore diverso
- d) Il programma causa un errore a tempo di compilazione, perché la prima istruzione della funzione *fun* prova a modificare la costante *g* definita nella prima istruzione della funzione *main*

8. Dato il seguente programma:

```
struct parola {char *s ; int n ;} ;

main()
{
    parola v[2] ;
    for (int i = 0 ; i < 2 ; i++) {
        v[i].s = new char[100] ;
        cin>>v[i].s ;
        v[i].n = strlen(v[i].s) ;
        cout<<v[i].s<<" "<<v[i].n<<endl ;
    }
}
```

e definita una parola come una sequenza di caratteri non separati da spazi:

- a) Purché l'utente non inserisca parole più lunghe di 100 caratteri, il programma legge correttamente da *stdin* due parole e stampa su *stdout* ciascuna delle due parole seguita dalla rispettiva lunghezza
- b) Purché l'utente non inserisca parole più lunghe di 99 caratteri, il programma legge correttamente da *stdin* due parole e stampa su *stdout* ciascuna delle due parole seguita dalla rispettiva lunghezza
- c) Nessuna delle altre risposte è vera
- d) Anche se il programma non dealloca esplicitamente gli oggetti che alloca dinamicamente durante la sua esecuzione, la memoria occupata da tali oggetti è restituita ugualmente al sistema operativo dopo la terminazione del programma stesso

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

9. (4 pt.) Cosa scrive su *stdout* il seguente programma?

```
void fun(char [], int);

main()
{
    char string[] = "luca";
    cout<<string<<endl;
    fun(string, strlen(string));
    cout<<string<<endl;
}

void fun(char s[], int n)
{
    int i = 0, j = n - 1 ;
    char c = s[j] ;
    while (s[i] != '\0'){
        s[i] = c ;
        i++ ; j-- ;
        c = s[j] ;
    }
}
```

10. (6 pt.) Definita una parola come una sequenza di caratteri non separati da alcun carattere spazio, scrivere il corpo della seguente funzione

```
int fun(const char s[]) ;
```

che ritorna la lunghezza massima delle parole contenute nella stringa memorizzata nell'*array* di caratteri *s* (la funzione prende in ingresso solo tale array e nessun altro parametro, inoltre non usa variabili globali). Esempio: se si passa alla funzione la stringa "*primo secondo terzo*", la funzione ritorna 7.

Programmazione I / Informatica generale

Prova scritta - 2 Settembre 2008

Nome: _____ Cognome: _____

Matricola: _____ Corso di Laurea: _____

Indicare le risposte corrette apponendo una croce nella casella corrispondente. Per superare la prova bisogna aver raggiunto almeno 10 punti nelle domande a risposta singola/multipla, ed almeno 15 complessivamente. Avete due copie di questa pagina, usatene una per calcolare il voto da sole/soli durante la correzione e decidere se consegnare.

	Risposte				Punti/ Penalità	
	A	B	C	D		
1					3	-0.5
2					3	-0.5
3					3	-0.5
4					3	-0.5
5					3	-0.5
6						
7						
8						

Risposta alla domanda 9 (4 pt):

Risposta alla domanda 10 (6 pt):

Programmazione I / Informatica generale

Prova scritta - 2 Settembre 2008

Nome: _____ Cognome: _____

Matricola: _____ Corso di Laurea: _____

Indicare le risposte corrette apponendo una croce nella casella corrispondente. Per superare la prova bisogna aver raggiunto almeno 10 punti nelle domande a risposta singola/multipla, ed almeno 15 complessivamente. Avete due copie di questa pagina, usatene una per calcolare il voto da sole/soli durante la correzione e decidere se consegnare.

	Risposte				Punti/ Penalità	
	A	B	C	D		
1					3	-0.5
2					3	-0.5
3					3	-0.5
4					3	-0.5
5					3	-0.5
6						
7						
8						

Risposta alla domanda 9 (4 pt):

Risposta alla domanda 10 (6 pt):