

Esercitazioni di Fondamenti di Programmazione

Anno accademico 2018/2019

Es. 7.1 – Debug

Il seguente programma: (1) accetta da tastiera un numero n ; (2) accetta da tastiera n numeri e li memorizza in un array; (3) calcola il massimo di questi numeri; (4) somma tale massimo a tutti gli elementi dell'array; (5) moltiplica tra loro tutti gli elementi dell'array risultante; e (6) stampa a schermo il risultato finale.

```
#include <iostream>
using namespace std;
int main() {
    int n;
    cout << "Inserisci n:" << endl;
    cin >> n;
    if(n < 1) {
        cout << "Numero non valido" << endl;
        return 0;
    }
    int* v = new int[n];

    cout << "Inserisci " << n << " numeri:" << endl;
    for(int i = 1; i <= n; i++)
        cin >> v[i];

    int max = 0;
    for(int i = 1; i <= n; i++)
        max = (v[i-1] > max) ? v[i-1] : max;

    for(int i = 1; i <= n; i++)
        v[i-1] += max;

    int mul = v[0];
    for(int i = 1; i <= n; i++)
        mul *= v[i-1];

    cout << "Risultato del calcolo:" << endl;
    cout << mul;

    delete[] v;
    return 0;
}
```

Tale programma contiene alcuni bug, come dimostrato dal seguente output:

```
Inserisci n:
3
Inserisci 3 numeri:
4 7 10
Risultato del calcolo:
7546      <- dovrebbe restituire 4760!
```

Il risultato corretto dovrebbe essere invece 4760, in quanto:

$\max\{4, 7, 10\} = 10$ $\{4+10, 7+10, 10+10\} = \{14, 17, 20\}$ $14*17*20 = 4760$.

Utilizzare il debugger visuale DDD per trovare e correggere tutti i bug.

Es. 7.2 – Liste 1

PRIMA PARTE:

Scrivere un programma che:

- crea una lista di interi vuota;
- inserisce l'intero 50 nella lista;
- inserisce l'elemento 60 in fondo alla lista;
- inserisce l'elemento 70 in testa alla lista.

Debuggare le singole operazioni con il debugger visuale DDD.

SECONDA PARTE:

Scrivere la funzione `stampa ()` che prende per argomento un lista di interi e ne stampa a video gli elementi.

Estendere il programma per:

- invocare la funzione `stampa ()` sulla lista appena creata;
- eliminare il secondo elemento dalla lista;
- invocare nuovamente la funzione `stampa ()` sulla lista.

Output di esempio:

```
70 50 60
70 60
```

Es. 7.3 – Inserisci ordinato

Scrivere una funzione `inserisciOrdinato(L, n)` che prende in ingresso una lista di interi `L` (supposta già ordinata) e inserisce un nuovo intero `n` in una posizione tale da farla rimanere ordinata.

Scrivere un programma che crea una lista vuota, legge 6 interi da tastiera e li inserisce in modo ordinato nella lista, e stampa la lista risultante.

Output di esempio:

```
Inserisci 6 numeri:  
1 5 3 -1 0 7  
Risultato:  
-1 0 1 3 5 7
```