

Esercitazione del 12/10/2009

Esercizio 1 - Implementare in C++ un programma che, letti da input le misure dei lati di un triangolo (rappresentati da numeri interi), stampi di che tipo è il triangolo (“Equilatero”, “Isoscele”, “Scaleno”).

```
#include<iostream.h>

int main()
{
    int lato1, lato2, lato3;

    cout<<"Inserisci lato1: ";
    cin>>lato1;
    cout<<"Inserisci lato2: ";
    cin>>lato2;
    cout<<"Inserisci lato3: ";
    cin>>lato3;

    if ((lato1 == lato2) && (lato2 == lato3))
        cout<<"Il triangolo e' equilatero"<<endl;

    else if((lato1 == lato2) || (lato2 == lato3) || (lato3==lato1))
        cout<<"Il triangolo e' isoscele "<<endl;

    else
        cout<<"Il triangolo e' scaleno"<<endl;

    return 0;
}
```

Esercizio 2 - Implementare in C++ un programma che, letta da input una sequenza di numeri positivi terminata da un “tappo” negativo, memorizzi in due totali separati le somme alternate dei numeri. In altre parole, in due totali separati la somma dei numeri in posizione pari e quella dei numeri in posizione dispari nella sequenza. Stampare poi su standard output i risultati.

Esempio: se la sequenza fosse

3 6 1 2 3 9 5 -4

Il totale dei numeri in posizione dispari sarebbe $T_1=3+1+3+5=12$, e quello di quelli in posizione pari $T_2=6+2+9=17$.

```
#include<iostream.h>

#include<iostream.h>

int main()
{

    int num, conta, sommaP = 0, sommaD = 0;
    cout<<"Inserisci sequenza terminata da un numero negativo "<<endl;
    cin>>num;

    while(num >= 0)
    {
        conta++;
        if(conta % 2 == 0)
            sommaP = sommaP + num;
        else
            sommaD = sommaD + num;

        cin>>num;
    }

    cout<<"Somma numeri in posizione pari = "<<sommaP<<endl;
    cout<<"Somma numeri in posizione dispari = "<<sommaD<<endl;

    return 0;
}
```

ESERCIZIO 3: Una sequenza di 20 numeri contiene delle sottosequenze separate da 0 .

Esempio: la sequenza 12030456700123533013 contiene 6 sottosequenze:

123 - 3 - 4567 - VUOTA - 123533 - 13

Quesito: : Si scriva un programma C++ che legga da input una sequenza di 20 numeri e stampi il numero di sottosequenze separate da 0 .

```
#include<iostream.h>

int main()
{
    int num, conta_ss = 0;

    cout<<"Inserisci sequenza di interi"<<endl;

    for(int i=0; i<20; i++)
    {
        cin>>num;
        if(num == 0)
            conta_ss++;
    }

    if(num != 0)
        conta_ss++;
    cout<<"Il numero di sottosequenze e' "<<conta_ss<<endl;

    return 0;
}
```

ESERCIZIO 4: Una famiglia S di insiemi di interi positivi può essere rappresentata da una sequenza di interi $seq(S)$, terminata da -1, ed in cui ogni insieme della famiglia sia separato dal successivo da uno 0.

Esempio: la famiglia $\{\{7,3\},\{5,1,7,4\},\{2\},\{\},\{3,9\}\}$ è rappresentata dalla sequenza $7,3,0,5,1,7,4,0,2,0,0,3,9,0,-1$.

Quesito: Si scriva un programma C++ che, letta una tale sequenza $seq(S)$:

- 1) fornisca in uscita la cardinalità della famiglia (cioè il numero degli insiemi che essa contiene); nell'esempio evidenziato sopra, il programma dovrebbe restituire 5;
- 2) fornisca in uscita la cardinalità massima degli insiemi della famiglia; nell'esempio evidenziato sopra, il programma dovrebbe restituire 4.

```
#include <iostream.h>

int main()
{
    int num, prec, card_fam=0, card_max=0, card_ins=0;
    cin>>num;
    while (num!=-1)
    {
        if (num!=0)
            card_ins++;
        else
        {
            card_fam++;
            if (card_ins>card_max)
                card_max=card_ins;
            card_ins=0;
        }
        prec = num;
        cin>>num;
    }

    if(prec != 0)
    {
        card_fam++;
        if (card_ins>card_max)
            card_max=card_ins;
    }

    cout<<"La cardinalita' della famiglia e': "<<card_fam<<endl;
    cout<<"La cardinalita' massima degli insiemi e': "<<card_max<<endl;

    return 0;
}
```