

```

/*
Programma : cppTryCatchThrow

Autore : Riontino Raffaele 4 informatici (corso serale)
         ITIS Molinari - Milano          17/02/2011

Funzionalità : Prove sull'utilizzo del blocco "try catch".
               Questo blocco serve per delimitare una porzione di codice
               detta "pericolosa", cioè che in caso di errore (come ad esempio
               la divisione per 0 , il mal funzionamento della stampante , ecc)
               può portare alla chiusura improvvisa del programma e alla perdita
               dei dati su cui si stava lavorando.
               Nella parte try si delimita : la porzione di codice da controllare,
               il controllo sul dato pericoloso e l'eventuale eccezione da sollevare
               (definita dall'istruzione throw); mentre nella parte catch viene catturata
               l'eventuale eccezione sollevata e la visualizzazione dell'errore riscontrato
               in modo da evitare un arresto improvviso del programma.
               try {
                   cin >> num2;
                   if (num2 == 0) throw "errore denominatore";
                   ris = num1 / num2;
               }
               catch(string e) {cerr << "Eccezione : " << e;}
               Nel caso l'eccezione non si verifica il blocco catch non viene eseguito

*/

#include <iostream>
#include <conio.h>
#include <string>
#include <exception>
using namespace std;

char menu();

int main()
{
    char scelta;
    string str;
    do{
        try {
            scelta = menu();
            switch(scelta)
            {
                case 'i' : throw 1;
                           break;
                case 'c' : throw 'c';
                           break;
                case 'f' : throw 2.5;
                           break;
                case 'b' : throw true;
                           break;
                case 'd' : float num1 , num2;
                           cout << "\ninserisci il numeratore : ";
                           cin >> num1;
                           cout << "inserisci il denominatore (0 per provocare l'errore): ";
                           cin >> num2;
                           if (num2 == 0) throw (str = "denominatore = 0");
                           cout << "ris = " << (num1 / num2);
                           getch();
                           break;
                case 'e' : break;
                default : throw "1" + 2;    //generica
            }
        }
        catch (int e)
        {cerr << "\nErrore int : " << e; getch();}
        catch (char e)
        {cerr << "\nErrore char : " << e; getch();}
        catch (float e)

```

```

        {cerr << "\nErrore float : " << e; getch();}
    catch (bool e)
    {cerr << "\nErrore booleano : " << e; getch();}
    catch (string e)
    {cerr << "\nEccezione generata : " << e; getch();}
    catch (...)//eccezione generica
    {cerr << "\nEccezione generica !!"; getch();}
}while(scelta != 'e');
}//fine main

```

```

char menu()
{
    system("cls");
    cout << "prova sul controllo delle eccezioni\n\n";
    cout << "\n\ti - int";
    cout << "\n\tc - char";
    cout << "\n\tf - float";
    cout << "\n\td - divisione per 0";
    cout << "\n\tb - booleano";
    cout << "\n\te - esci";
    cout << "\n\n\ttutti gli altri tasti eccezione generica\n";
    return (getch());
}

```