

FACULTATEA CALCULATOARE, INFORMATICA SI  
MICROELECTRONICA

UNIVERSITATEA TEHNICA A MOLDOVEI

MEDII INTERACTIVE DE DEZVOLTARE A  
PRODUSELOR SOFT

LUCRAREA DE LABORATOR#2

---

## Realizarea unui simplu GUI Calculator

---

Autor:  
Onta Vladislav

lector asistent:  
Irina COJANU  
lector superior:  
Radu MELNIC

# Lucrarea de laborator #2

## 1 Scopul lucrarii de laborator

Realizarea un simplu GUI calculator care suporta urmatoarele functii: +, -, /, \*, putere, radical, InversareSemn(+/-), operatii cu numere zecimale.

## 2 Mersul lucrarii de laborator

Pe parcursul lucrarii am creat un calculator ce poate efectua operatiile de baza. Calculatorul per-mite efectuarea operatiilor ca +, -, /, \*, putere, radical, InversareSemn(+/-). Operatiile date se pot efectua in lant (nu este necesar mereu sa apasam pe semnul egal).

## Secvente de cod

### Calculator

```
#pragma once

#include <stdlib.h>
#include <ctype.h>
#include <string.h>
#include "check.c"
#include "f_package.c"
#include "get.c"
#include "queue_stack.c"
#include "eval_expr.c"

using namespace System::Runtime::InteropServices;

namespace EvalExpr2 {

    using namespace System;
    using namespace System::ComponentModel;
    using namespace System::Collections;
    using namespace System::Windows::Forms;
    using namespace System::Data;
    using namespace System::Drawing;

    / <summary>
    /// Summary for MyForm
    /// </summary>
    public ref class MyForm : public System::Windows::Forms::Form
    {
    public:
```

```

MyForm(void)
{
    InitializeComponent();
    //
    //TODO: Add the constructor code here
    //
}

protected:
    /// <summary>
    /// Clean up any resources being used.
    /// </summary>
    ~MyForm()
    {
        if (components)
        {
            delete components;
        }
    }

private: System::Windows::Forms::TextBox^ txtDisplay;
private: System::Windows::Forms::Button^ bkspace;
private: System::Windows::Forms::Button^ C;

private: System::Windows::Forms::Button^ bdiv;

private: System::Windows::Forms::Button^ b7;
private: System::Windows::Forms::Button^ b8;
private: System::Windows::Forms::Button^ b9;
private: System::Windows::Forms::Button^ bmult;

private: System::Windows::Forms::Button^ b4;
private: System::Windows::Forms::Button^ b5;
private: System::Windows::Forms::Button^ b6;
private: System::Windows::Forms::Button^ bminus;
private: System::Windows::Forms::Button^ beq;

private: System::Windows::Forms::Button^ bplus;

private: System::Windows::Forms::Button^ b3;

private: System::Windows::Forms::Button^ b2;

private: System::Windows::Forms::Button^ b1;
private: System::Windows::Forms::Button^ paran1;
private: System::Windows::Forms::Button^ paran2;
private: System::Windows::Forms::Button^ bdot;
private: System::Windows::Forms::Button^ b0;

```

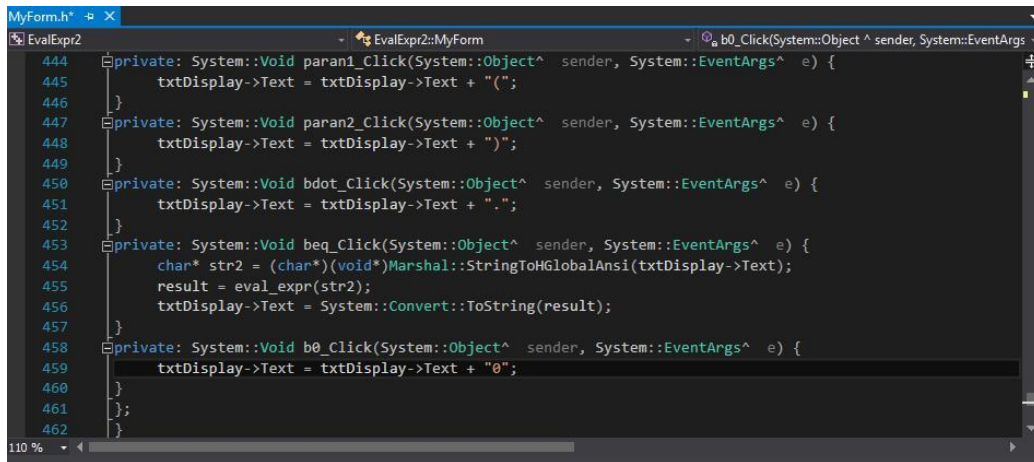
```
private: System::Windows::Forms::Button^ semn;
```

```
protected:
```

```
protected:
```

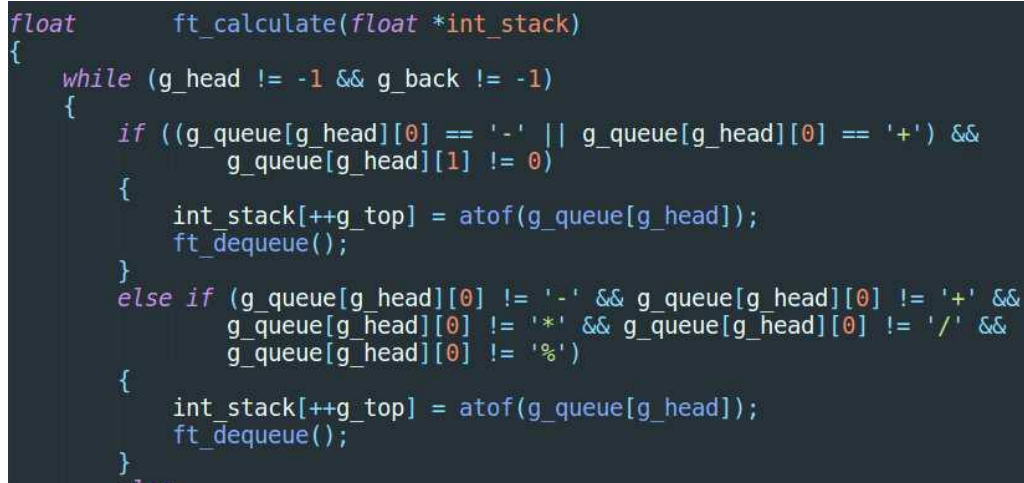
```
private:  
    /// <summary>  
    /// Required designer variable.  
    /// </summary>  
    System::ComponentModel::Container ^components
```

## Partea grafica



```
MyForm.h* x
EvalExpr2
- EvalExpr2::MyForm
- b0_Click(System::Object ^ sender, System::EventArgs e) {
444 private: System::Void paran1_Click(System::Object^ sender, System::EventArgs^ e) {
445     txtDisplay->Text = txtDisplay->Text + "(";
446 }
447 private: System::Void paran2_Click(System::Object^ sender, System::EventArgs^ e) {
448     txtDisplay->Text = txtDisplay->Text + ")";
449 }
450 private: System::Void bdot_Click(System::Object^ sender, System::EventArgs^ e) {
451     txtDisplay->Text = txtDisplay->Text + ".";
452 }
453 private: System::Void beq_Click(System::Object^ sender, System::EventArgs^ e) {
454     char* str2 = (char*)(void*)Marshal::StringToHGlobalAnsi(txtDisplay->Text);
455     result = eval_expr(str2);
456     txtDisplay->Text = System::Convert::ToString(result);
457 }
458 private: System::Void b0_Click(System::Object^ sender, System::EventArgs^ e) {
459     txtDisplay->Text = txtDisplay->Text + "0";
460 }
461 };
462 }
```

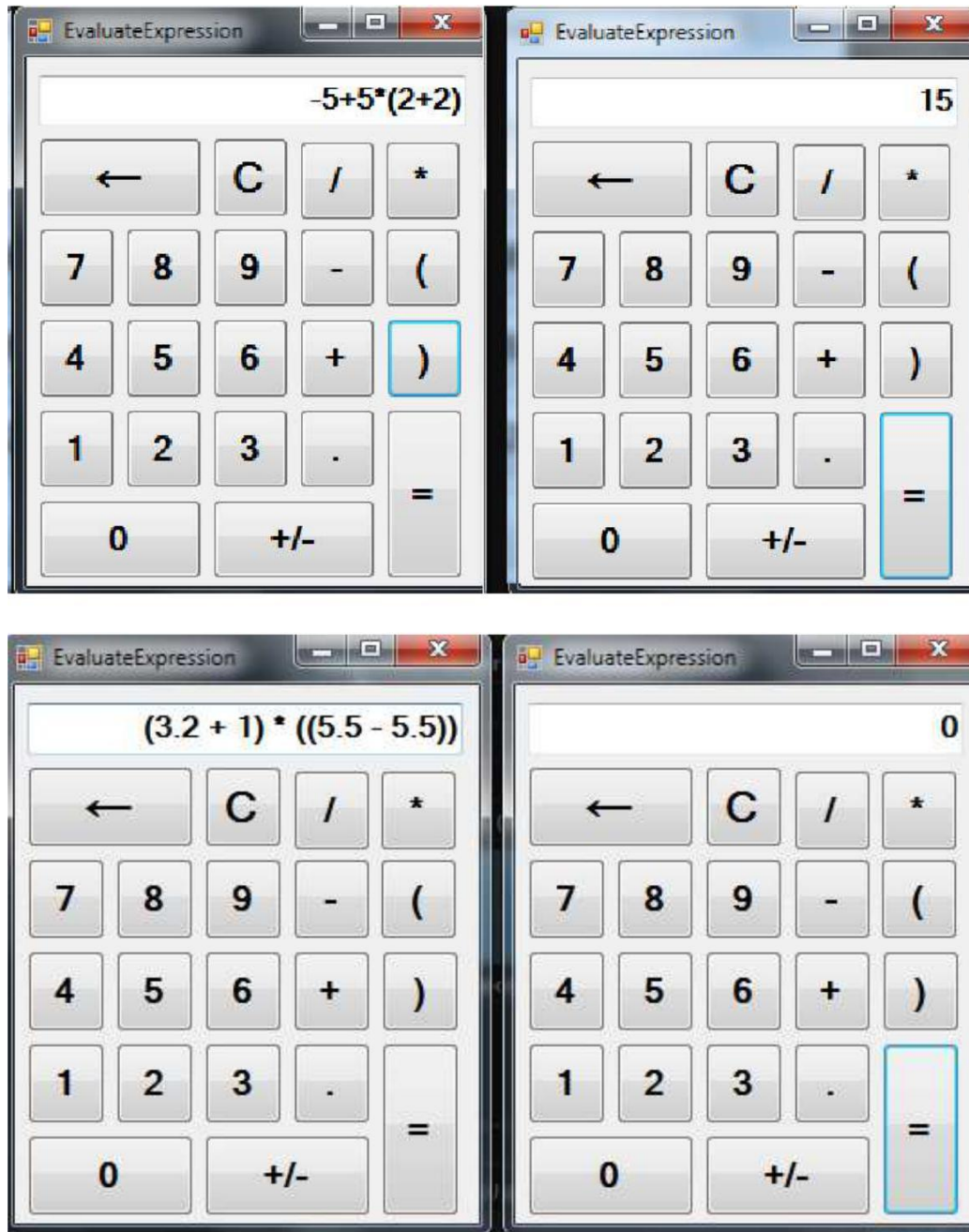
## Partea functionala



```
float ft_calculate(float *int_stack)
{
    while (g_head != -1 && g_back != -1)
    {
        if ((g_queue[g_head][0] == '-' || g_queue[g_head][0] == '+') &&
            g_queue[g_head][1] != 0)
        {
            int_stack[++g_top] = atof(g_queue[g_head]);
            ft_dequeue();
        }
        else if (g_queue[g_head][0] != '-' && g_queue[g_head][0] != '+' &&
            g_queue[g_head][0] != '*' && g_queue[g_head][0] != '/' &&
            g_queue[g_head][0] != '%')
        {
            int_stack[++g_top] = atof(g_queue[g_head]);
            ft_dequeue();
        }
    }
}
```

## Screenshoturi

Calculator



## Concluzie

În lucrarea dată am creat o versiuniune a calculatorului real. Fiecare calculator avînd specificul său și construcția sa. Am învățat să ”împachetăm” programele noastre într-un mod mai prielnic unui User. Astfel putem crea În urma efectuării lucrării mi-am îmbogățit cunoștințele în ceea ce privește crearea aplicațiilor GUI. Acum putem să nu folosim mereu consola dar să comunicăm cu programul nostru printr-o formă mai plăcută și mai familiară unui simplu utilizator. Drept IDE am ales **Visual Studio** din cauza că am găsit mai multă informație despre el. Am utilizat limbajul **C++** iar partea funcțională e scrisă în **C**. Un avantaj foarte mare a unui mediu interactiv este faptul că ne permite să cream un lucru într-un timp foarte scurt și totodată să evităm multe greșeli. După părerea mea lucrarea dată este foarte importantă pentru un student TI, personal am acumulat multe cunoștințe noi în urma efectuării acesteia.