

FACULTATEA CALCULATOARE, INFORMATICA SI MICROELECTRONICA

UNIVERSITATEA TEHNICA A MOLDOVEI

MEDII INTERACTIVE DE DEZVOLTARE A PRODUSELOR SOFT

LUCRAREA DE LABORATOR#2

GUI Development

Autor:

Nadejda HERGHELEGIU

lector asistent:

Irina COJANU

lector superior:

Svetlana COJOCARU

Laboratory work #2

1 Scopul lucrării de laborator

- Familiarizarea cu un IDE, framework ce permite dezvoltarea unei aplicații grafice GUI și elaborarea unui calculator datorită acestor interfețe.

2 Obiective

- Realizeaza un simplu GUI Calculator
- Operatiile simple: +,-,*,/,putere,radical,InversareSemn(+/-),operatii cu numere zecimale.
- Divizare proiectului in doua module - Interfata grafica(Modul GUI) si Modulul de baza(Core Module).

3 Laboratory work implementation

3.1 Tasks and Points

- Setarea interfeței vizuale;
- Crearea metodelor pentru prelucrarea evenimentelor și atașarea lor la GUI;
- Crearea codului *Core*;
- crearea metodelor pentru efectuarea operațiilor matematice.

3.2 Analiza lucrării de laborator

Link la repozitoriu: <https://insert.your.link.here>

Am efectuat lucrarea pe platforma .NET[?], folosind drept mediu de dezvoltare Microsot Visual Studio[?] și C# în calitate de limbaj de programare. Proiectul este alcătuit din două module: GUI și Core, care conține metodele pentru efectuarea calculelor și sunt folosite în modulul GUI.

Pentru crearea modulului grafic am utilizat un proiect de tip WinForms unde am adăugat pe o formă butoanele necesare și un label în care va fi afișat textul introdus.

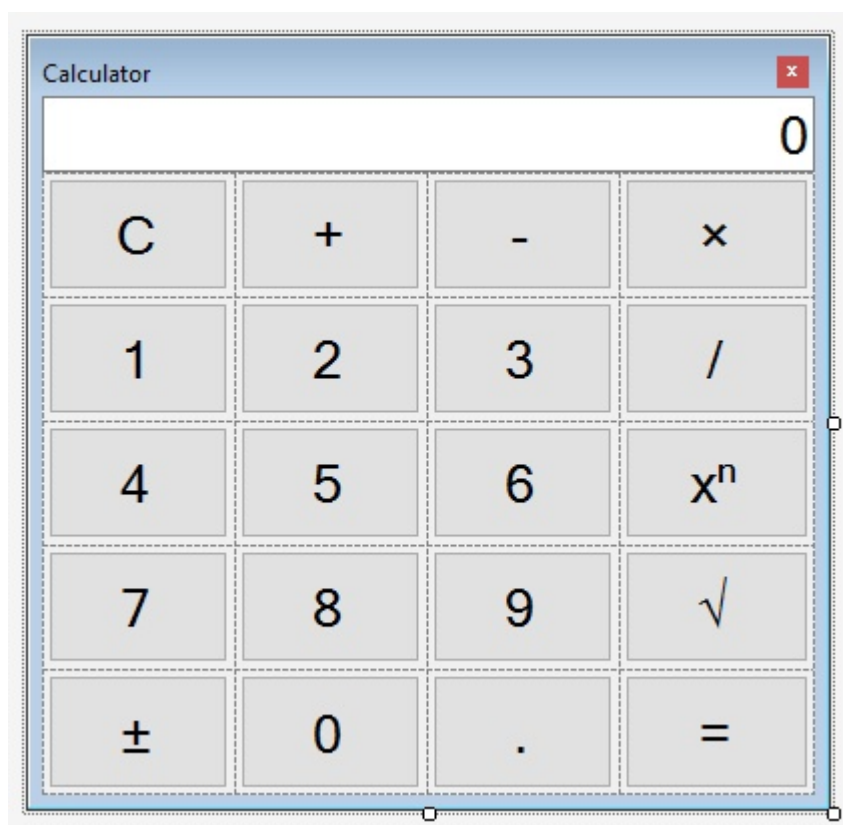


Figure 3.1 – Interfața aplicației

După aceasta am creat metode pentru fiecare eveniment necesar și le-am legat cu acțiunile butoanelor. Pe lângă evenimentele de click au fost prelucrate și cele de KeyPress. Am întâlnit o problemă cu evenimentul apelat la tastarea butonului "Enter", se executa evenimentul implit din

windows, care simulează click pe butonul activ la moment, în loc de metoda definită. Aceasta a fost soluționată prin supraîncărcarea metodei **ProcessCmdKey**.

```
1 reference
protected override bool ProcessCmdKey(ref Message msg, Keys keyData)
{
    if (keyData == Keys.Enter)
    {
        PerformClickForOther((char) 13);
        return true;
    }
    return base.ProcessCmdKey(ref msg, keyData);
}
```

Figure 3.2– Supraîncărcarea metodei ProcessCmdKey

3.3 Imagini

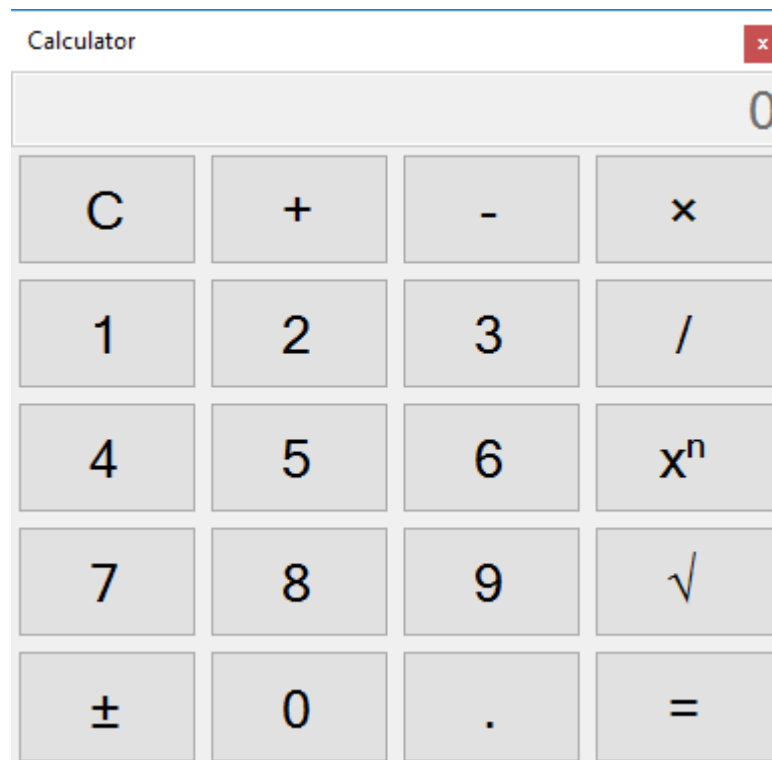


Figure 3.3– Fereastra inițială

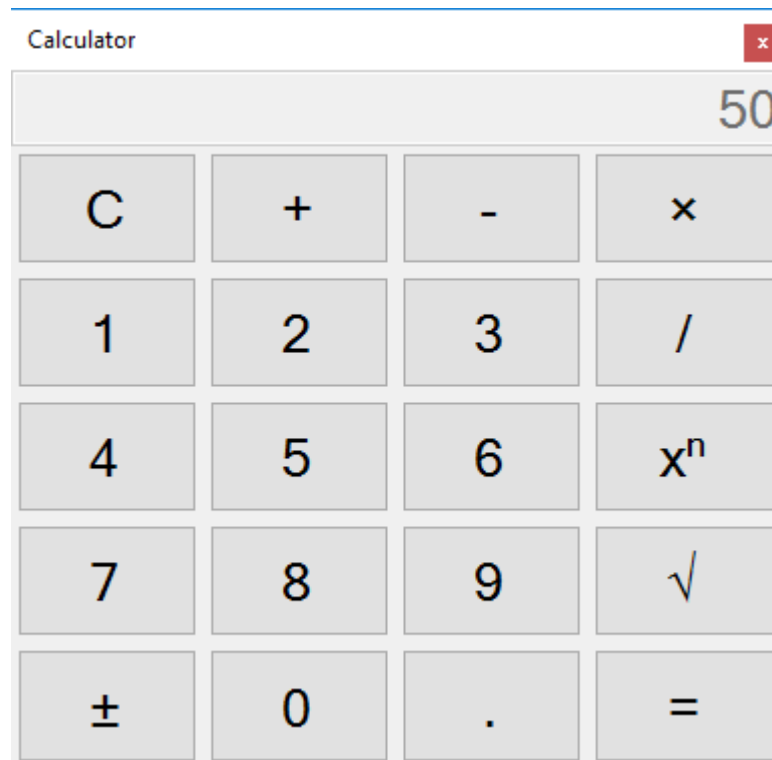


Figure 3.4– Introducerea datelor

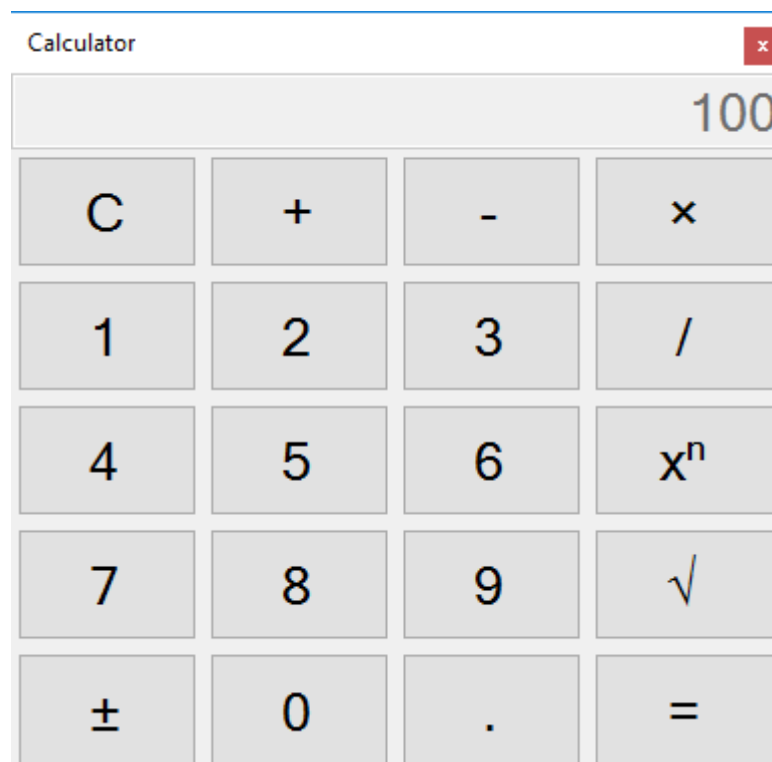


Figure 3.5– Rezultatul expresiei 50+50

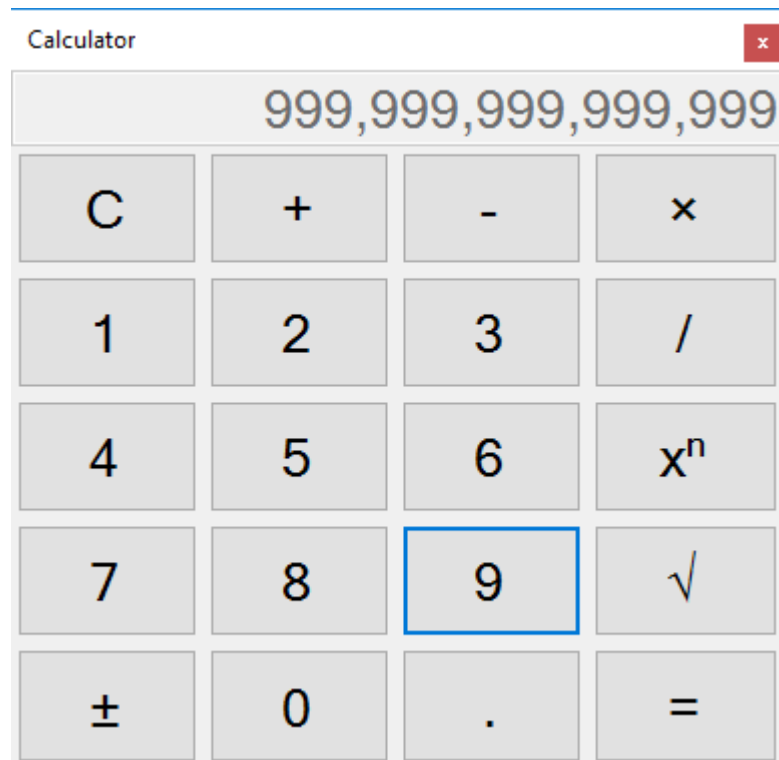


Figure 3.6– Numărul maxim

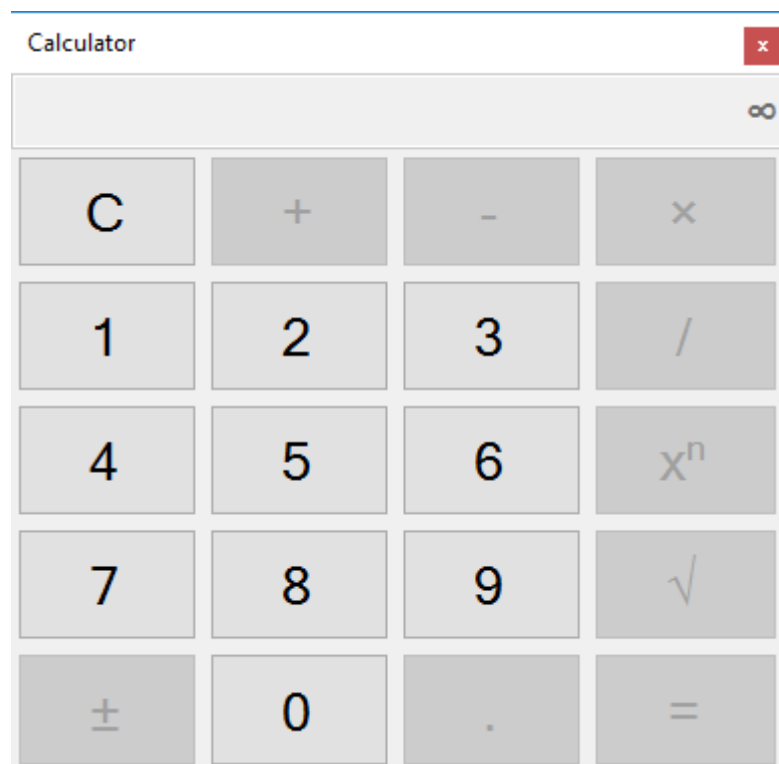


Figure 3.7– Caz de overflow

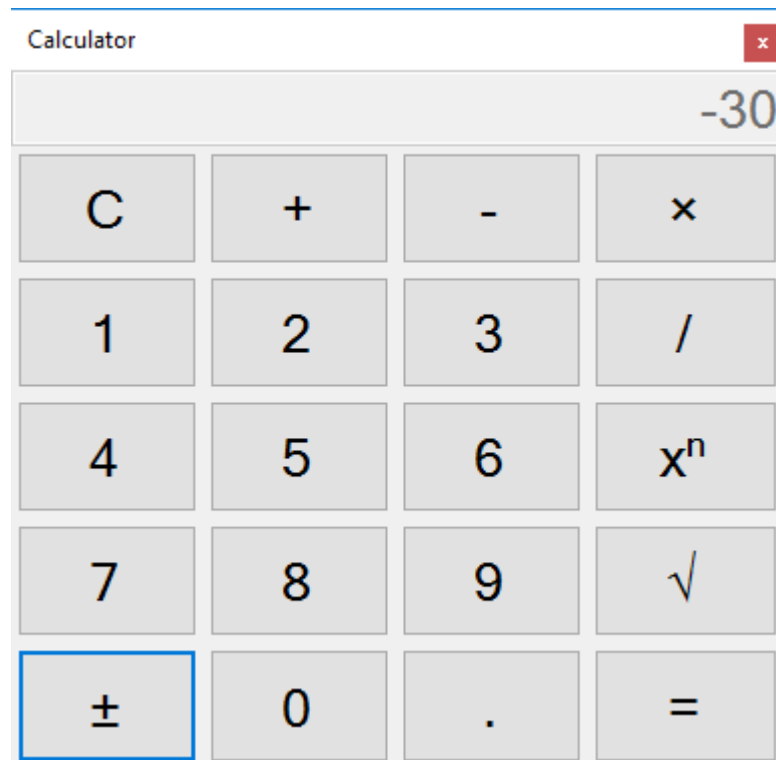


Figure 3.8– Operația de schimbare a semnului

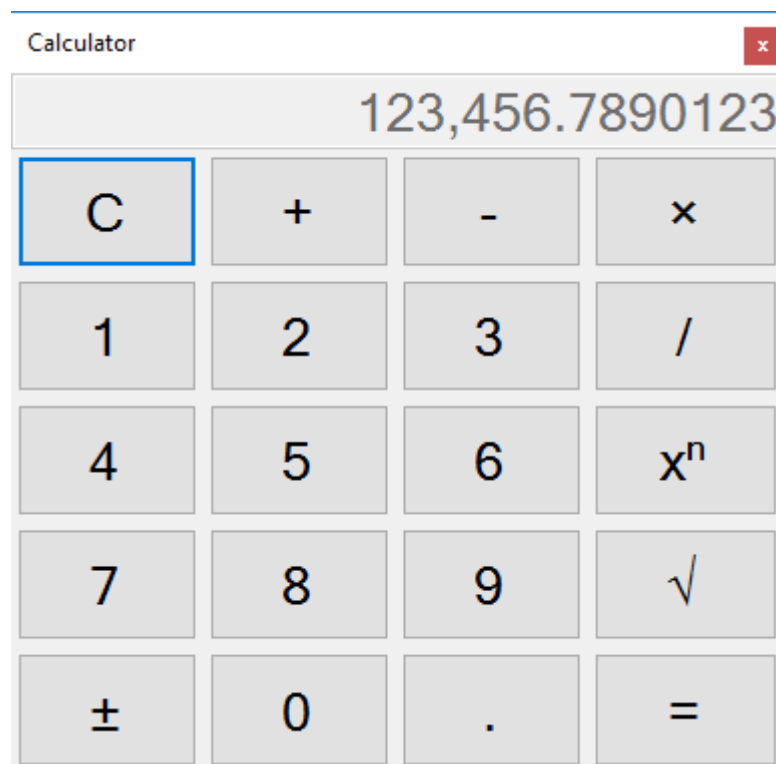


Figure 3.9– Introducerea unui număr zecimal

Concluzie

Pentru a elabora aplicația cerută am utilizat mediul de dezvoltare *Microsoft Visual Studio*, sistemul de control al versiunilor *git* și limbajul de programare *C#*. De asemenea, în timpul efectuării lucrării, am învățat multe lucruri noi despre WinForms, limbajul *C#*, cât și despre principiile Programării Orientate pe Obiecte.

Elaborând această lucrare, am creat pentru prima dată o soluție compusă din mai mult de un proiect.

References

- 1 .NET, *official page*, <https://dotnetfoundation.org/>
- 2 Visual Studio, *official page*, <https://www.visualstudio.com/>