Text

Description automatically generated

REPUBLIKA E SHQIPERISE

UNIVERSITETI POLITEKNIK I TIRANES

FAKULTETI I TEKNOLOGJISE SE INFORMACIONIT

**Dega**: Inxhinieri Informatike

**Cikli i studimeve**: Master, viti II

**Lenda**: Programim i Avancuar

**Laborator IV: Makine Llogaritese**

Punoi:Sabina Ahmetaj Pranoi: Dr. Hakik Paci

Tirane, Nentor 2024

A screenshot of a calculator

Description automatically generated

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Pune\_4\_Sabina\_Ahmetaj\_

{

public partial class Form1 : Form

{

private double currentvalue = 0;

private double previousvalue = 0;

private string currentoperation = string.Empty;

private List<string> history = new List<string>();

public Form1()

{

InitializeComponent();

this.CenterToScreen();

}

private void Form1\_Load(object sender, EventArgs e)

{

textdisplay1.Text = "0";

textdisplay2.Text = string.Empty;

}

private void buttonnum\_click(object sender, EventArgs e)

{

Button button = (Button)sender;

if (textdisplay1.Text == "0")

{

textdisplay1.Text = button.Text;

}

else

{

textdisplay1.Text += button.Text;

}

}

private void buttonoperator\_click(object sender, EventArgs e)

{

Button button = (Button)sender;

previousvalue = double.Parse(textdisplay1.Text);

currentoperation = button.Text;

textdisplay2.Text = previousvalue.ToString() + " " + currentoperation;

textdisplay1.Text = "0";

{

MessageBox.Show("Invalid input.Please enter a valid number.");

return;

}

currentoperation = button.Text;

textdisplay2.Text = previousvalue.ToString() + " " + currentoperation;

textdisplay1.Text = "0";

}

private void buttonequal\_Click(object sender, EventArgs e)

{

double parsedCurrentValue;

if (!double.TryParse(textdisplay1.Text, out parsedCurrentValue))

{

MessageBox.Show("Invalid input. Please enter a valid number");

return;

}

currentvalue = parsedCurrentValue;

string operationResult = string.Empty;

textdisplay2.Text += " " + currentvalue.ToString();

switch (currentoperation)

{

case "+":

textdisplay1.Text = (previousvalue + currentvalue).ToString();

break;

case "-":

textdisplay1.Text = (previousvalue - currentvalue).ToString();

break;

case "\*":

textdisplay1.Text = (previousvalue \* currentvalue).ToString();

break;

case "÷":

if (currentvalue != 0)

textdisplay1.Text = (previousvalue / currentvalue).ToString();

else

MessageBox.Show("Division by 0 cannot be done");

break;

case "^":

textdisplay2.Text = $" {previousvalue} ^ {currentvalue} ";

textdisplay1.Text = Math.Pow(previousvalue, currentvalue).ToString();

break;

case "%":

textdisplay2.Text = $" {previousvalue} & {currentvalue} ";

textdisplay1.Text = (previousvalue \* (currentvalue / 100)).ToString();

break;

default:

textdisplay2.Text = $"{textdisplay1.Text} = ";

break;

}

//textdisplay1.Text = operationResult;

//string historyEntry = $"{previousvalue} {currentoperation} {currentvalue} = {operationResult}";

//history.Add(historyEntry);

//textdisplay2.Text = string.Empty;

//previousvalue = Double.Parse(textdisplay1.Text);

//currentoperation = string.Empty;

}

private void buttonclear\_click(object sender, EventArgs e)

{

textdisplay1.Text = "0";

textdisplay2.Text = string.Empty;

previousvalue = 0;

currentoperation = string.Empty;

}

private void buttondelete\_click(object sender, EventArgs e)

{

if (textdisplay1.Text.Length > 1)

{

textdisplay1.Text = textdisplay1.Text.Substring(0, textdisplay1.Text.Length - 1);

}

else

{

textdisplay1.Text = "0";

}

}

private void buttonpm\_click(object sender, EventArgs e)

{

double grade = double.Parse(textdisplay1.Text);

grade = -grade;

textdisplay1.Text = grade.ToString();

}

private void buttondecimal\_click(object sender, EventArgs e)

{

if (!textdisplay1.Text.Contains("."))

{

textdisplay1.Text += ".";

}

}

private void buttonraport\_click(object sender, EventArgs e)

{

double grade = double.Parse(textdisplay1.Text);

if (grade != 0)

{

textdisplay1.Text = (1 / grade).ToString();

}

else

{

MessageBox.Show("Can not be divided by 0");

}

}

private void buttonsr\_click(object sender, EventArgs e)

{

double grade = double.Parse(textdisplay1.Text);

if (grade >= 0)

{

textdisplay1.Text = Math.Sqrt(grade).ToString();

}

else

MessageBox.Show("The square root of a negative number can not be found");

}

private void buttonminimize\_Click(object sender, EventArgs e)

{

this.WindowState = FormWindowState.Minimized;

}

private void buttonmax\_Click(object sender, EventArgs e)

{

if (this.WindowState == FormWindowState.Normal)

{

this.WindowState = FormWindowState.Maximized;

}

else

{

this.WindowState = FormWindowState.Normal;

}

}

private void buttonexit\_Click(object sender, EventArgs e)

{

Application.Exit();

}

private void buttonexp\_click(object sender, EventArgs e)

{

double x;

x = Convert.ToDouble(textdisplay1.Text) \* Convert.ToDouble(textdisplay1.Text);

textdisplay1.Text = Convert.ToString(x);

}

private void buttonce\_click(object sender, EventArgs e)

{

textdisplay1.Text = "0";

}

private void buttonpercent\_click(object sender, EventArgs e)

{

if (string.IsNullOrEmpty(currentoperation))

{

double current = double.Parse(textdisplay1.Text);

textdisplay1.Text = (current / 100).ToString();

}

else

{

currentvalue = double.Parse(textdisplay1.Text);

double result = previousvalue \* (currentvalue / 100);

textdisplay1.Text = result.ToString();

textdisplay2.Text = $"{previousvalue} {currentoperation} {currentvalue}%";

}

}

private void buttonhistory\_click(object sender, EventArgs e)

{

if (history.Count == 0)

{

MessageBox.Show("No history available");

return;

}

string historyDisplay = string.Join("\n", history);

MessageBox.Show(historyDisplay, "Calculation history");

}

private void buttonmultiply\_click(object sender, EventArgs e)

{

previousvalue = double.Parse(textdisplay1.Text);

currentoperation = "\*";

textdisplay2.Text = previousvalue.ToString() + "\*";

textdisplay1.Text = "0";

}

private void buttonadd\_click(object sender, EventArgs e)

{

previousvalue = double.Parse(textdisplay1.Text);

currentoperation = "+";

textdisplay2.Text = previousvalue.ToString() + "+";

textdisplay1.Text = "0";

}

private void buttonminus\_click(object sender, EventArgs e)

{

previousvalue = double.Parse(textdisplay1.Text);

currentoperation = "-";

textdisplay2.Text = previousvalue.ToString() + "-";

textdisplay1.Text = "0";

}

private void buttondivide\_click(object sender, EventArgs e)

{

previousvalue = double.Parse(textdisplay1.Text);

currentoperation = "÷";

textdisplay2.Text = previousvalue.ToString() + "÷";

textdisplay1.Text = "0";

}

}

}

A screenshot of a calculator

Description automatically generated

**Butonet e Numrave (0-9):**

Funksioni: Shfaq numrat e shtypur ne ekranin kryesor (textdisplay1). Nese ekrani eshte "0", numri i shtypur e zevendeson ate. Perndryshe, numri i shtypur shtohet pas numrave ekzistues.

\***Butoni i Operatoreve Matematike (+, -, , ÷):**

Funksioni: Percakton operatorin matematik qe do te perdoret per llogaritjen. Gjithashtu, ruan vleren aktuale te ekranit ne previousvalue dhe e shfaq kete vlere se bashku me operatorin ne ekranin dytesor (textdisplay2).

**Butoni "=" (Equal):**

Funksioni: Kryen operacionin matematik te percaktuar nga operatori aktual (currentoperation) duke perdorur previousvalue dhe vleren aktuale (currentvalue). Rezultati shfaqet ne ekranin kryesor (textdisplay1), dhe operacioni i plote shfaqet ne ekranin dytesor (textdisplay2).

**Butoni "C" (Clear):**

Funksioni: Fshin te gjitha vlerat dhe e kthen kalkulatorin ne gjendjen e tij fillestare. (textdisplay1 shfaq "0" dhe textdisplay2 eshte bosh.)

**Butoni "CE" (Clear Entry):**

Funksioni: Fshin vetem vleren aktuale ne ekranin kryesor (textdisplay1), duke e zevendesuar ate me "0".

**Butoni "Del" (Delete):**

Funksioni: Fshin karakterin e fundit te vleres ne ekranin kryesor (textdisplay1). Nese mbetet vetem nje karakter, shfaq "0".

**Butoni "+/-" (Plus/Minus):**

Funksioni: Nderron shenjen e vleres aktuale ne ekranin kryesor. Nese eshte pozitiv, behet negativ, dhe anasjelltas.

**Butoni "." (Decimal):**

Funksioni: Shton nje pike dhjetore ne vleren aktuale, nese nuk ka tashme nje pike dhjetore

**Butoni "%" (Percent):**

Funksioni: Llogarit perqindjen e nje vlere. Nese nuk ka operator aktiv, ndan vleren aktuale me 100. Nese ka nje operator, llogarit perqindjen ne raport me previousvalue.

**Butoni "1/x":**

Funksioni: Llogarit reciprokun (1 mbi vleren) e vleres aktuale ne ekranin kryesor. Nese vlera eshte 0, shfaqet nje mesazh gabimi.

**Butoni "√" (Square Root):**

Funksioni: Llogarit rrenjen katrore te vleres aktuale. Nese vlera eshte negative, shfaqet nje mesazh gabimi.

**Butoni "x²" (Square):**

Funksioni: Llogarit katrorin e vleres aktuale dhe e shfaq rezultatin ne ekranin kryesor.