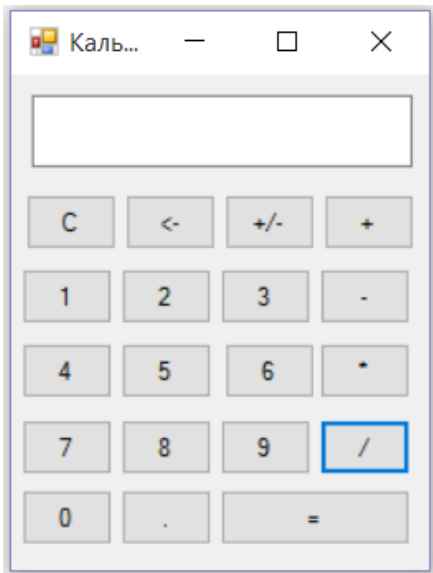


ОПОРНА СХЕМА

для виконання ПРАКТИЧНОГО ЗАВДАННЯ



СТВОРИТИ «КАЛЬКУЛЯТОР» для додавання, віднімання, множення та ділення довільних чисел (ІНТЕРФЕЙС СПРОЕКТУВАТИ СХОЖИМ ДО «СТАНДАРТНОГО» КАЛЬКУЛЯТОРА WINDOWS)

1. Спроекувати інтерфейс

додатку: створити форму, розмістити елементи керування, налаштувати властивості:

- з використанням можливостей візуального конструктора;
- програмно;



2. Реалізувати обчислення –

створити обробники подій для виконання дій додавання, віднімання, множення та ділення довільних чисел:

- забезпечити зчитування необхідних даних.
- виконати арифметичні обчислення.
- забезпечити виведення даних.



4. Усунути недоліки

- проаналізувати методи виправлення різних видів помилок



3. Виконати перевірку коректності роботи створеного застосунку:

- перевірка правильності виконання обчислень;
- перевірка коректності вхідних даних.

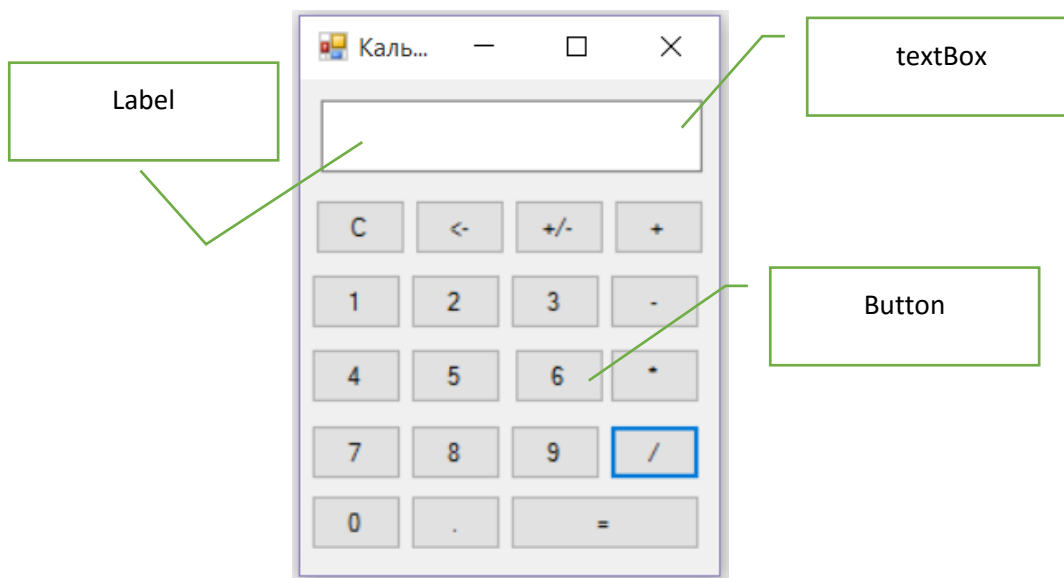
1 Проектування інтерфейсу калькулятора

1.1 Створення проекту:

File → New → Project → Visual C# → Windows Forms App → вказати назву проекту та шлях до папки. По замовчуванню проект буде містити форму Form1.

**Оскільки проект може складатися з великої кількості файлів, доцільно для його зберігання створити окрему папку*

1.2 Розміщення елементів керування на формі



2 Реалізація обчислень - створення обробників подій

2.1 Оголошення змінних програми:

```
float a, b; //для зберігання чисел
int op; //для зберігання вибраної арифметичної операції
// bool znak = true; //для зберігання знаку числа
```

2.2 Створення обробників подій для виведення цифр в текстове поле калькулятора:

```
private void button17_Click(object sender, EventArgs e)
{
    textBox1.Text = textBox1.Text + "0";
}

private void button5_Click(object sender, EventArgs e)
{
    textBox1.Text = textBox1.Text + "1"; } ...
```

2.3 Створення обробників подій для зберігання чисел та арифметичних операторів:

```
private void button4_Click(object sender, EventArgs e)
{
    a = float.Parse(textBox1.Text);
    textBox1.Clear();
    op = 1;
    label1.Text = a.ToString() + "+";
}

private void button8_Click(object sender, EventArgs e)
{
    a = float.Parse(textBox1.Text);
    textBox1.Clear();
    op = 2;
    label1.Text = a.ToString() + "-";
} .....
```

**функція `float.Parse()` перетворює текст в число з плаваючою комою;*

**функція `Clear()` очищує текстове поле;*

**функція `ToString()` перетворює числові дані в текстові.*

2.4 Створення обробника події натискання на кнопку «дорівнює»:

```
private void button19_Click(object sender, EventArgs e)
{
    switch(op)
    {
        case 1:
            b = a + float.Parse(textBox1.Text);
            textBox1.Text = b.ToString();
            break;
        case 2:
            b = a - float.Parse(textBox1.Text);
            textBox1.Text = b.ToString();
            break;
        case 3:
            b = a * float.Parse(textBox1.Text);
            textBox1.Text = b.ToString();
            break;
        case 4:
            b = a / float.Parse(textBox1.Text);
            textBox1.Text = b.ToString();
            break;
        default:
            break;
    }
    label1.Text = "=";
}
```

3. Перевірка коректності роботи додатку

Проблемні питання

1. Чи завжди КОРИСТУВАЧІ ВИКОРИСТОВУЮТЬ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ПЕРЕДБАЧУВАНО?
2. Які можуть виникнути ПРОБЛЕМИ І ПОМИЛКИ?
3. ХТО ВИНЕН ПРИ ВИНИКНЕННІ ПОМИЛОК – ПРОГРАМІСТ ЧИ КОРИСТУВАЧ?

4. Усунення недоліків

- ✦ ПРИ ВВЕДЕННІ ДАНИХ НЕОБХІДНО ЇХ ПЕРЕВІРЯТИ НА КОРЕКТНІСТЬ
- ✦ УНЕМОЖЛИВИТИ ВВЕДЕННЯ НЕКОРЕКТНИХ ДАНИХ – НАЙКРАЩИЙ ВАРІАНТ

ЗАВДАННЯ ДЛЯ САМОСТІЙНОГО ВИКОНАННЯ

1) Реалізувати програмний код для видалення останнього введеного символу (кнопка "<-“ калькулятора).

***Властивість Length** зберігає кількість символів в текстовому полі.

Приклад:

```
int lenght = textBox1.Text.Length - 1;
string text = textBox1.Text;
textBox1.Clear();
for (int i = 0; i < lenght; i++)
{
    textBox1.Text = textBox1.Text + text[i];
}
```

Контрольні питання

1. Яким чином розмістити можна розмістити елементи керування на формі?
2. Як створити процедуру натискання на кнопку?
3. Які елементи управління зазвичай використовуються для введення даних?
4. Які елементи управління можна використовувати для виведення даних?
5. Як зчитати дані з текстових полів?

Домашнє завдання

-
1. Вдосконалити Калькулятор таким чином, щоб була можливість виконувати піднесення до ступеня, знаходження кореня, додати математичні функції (sin, cos, exp ...).
 2. Оформити звіт по виконаній роботі.
 3. Повторити теоретичні відомості про організацію циклічних обчислень, синтаксис операторів циклу, базові алгоритми опрацювання масивів.