

Лабораторна робота № 14

Створення додатку для табулювання функції однієї змінної з використанням візуальних компонентів Microsoft Visual C++

Мета роботи: Ознайомлення з можливостями візуальних компонентів *Windows Form C++* в середовищі Visual Studio 2008.

Завдання Створити та запрограмувати візуальний проект Windows Form для табулювання функції згідно варіанту. Для виводу таблиці значень аргументу та функції використати елемент текстове поле (textBox) у багатострічковому режимі. У проекті передбачити можливість вмикати-вимикати виконання додаткового завдання за допомогою «прапорця» (checkBox) та запрограмувати кнопку «Очистити» для скидання результатів попереднього табулювання.

№ варіанта	Функція	Додатково знайти:
1.	$y = \sqrt{7 + 2\sin x + 3\cos x}$	Добуток значень функції, більших за задане число М
2.	$y = \sin 5x + \cos 7x - \sin 11x$	Суму від'ємних значень функції
3.	$y = \sqrt{x^2 + \log_5(x + 1)}$	Суму значень функції із заданого інтервалу [a; b]
4.	$y = x^{10} + 10^x - 2$	Кількість від'ємних значень функції
5.	$y = 5\sin x - \log_5 x $	Кількість значень функції, що належать заданому інтервалу [a; b]
6.	$y = 2\cos \pi x^3 + 3\sin^2 \ln x$	Добуток значень функції, менших заданого числа А
7.	$y = \sin 5x + \cos \frac{x}{\pi}$	Середнє арифметичне значень функції, менших заданого числа А
8.	$y = \sqrt{2 + x } + \sin 2\pi x$	Суму значень функції із заданого інтервалу [a; b]
9.	$y = x^3 \sin \pi x + 3x^2$	Добуток від'ємних значень функції
10.	$y = 5\lg(5\sin x + \sin 5x + 5)$	Суму значень функції, що належать заданому інтервалу [a; b]
11.	$y = x^3 + x^2 + x + \lg x$	Добуток значень функції із заданого інтервалу [a; b]
12.	$y = e^{3x} \sqrt{\sin(x-2) + 3}$	Кількість від'ємних значень функції
13.	$y = x \sin x + 3x^3 \cos \pi x$	Кількість значень функції, що належать заданому інтервалу [a; b]
14.	$y = 5 \operatorname{tg}(x^2 + x)$	Добуток значень функції, менших заданого числа А
15.	$u = 2x^3 - 3x^2 + 7x + 9$	Середнє арифметичне значень функції, менших заданого числа А

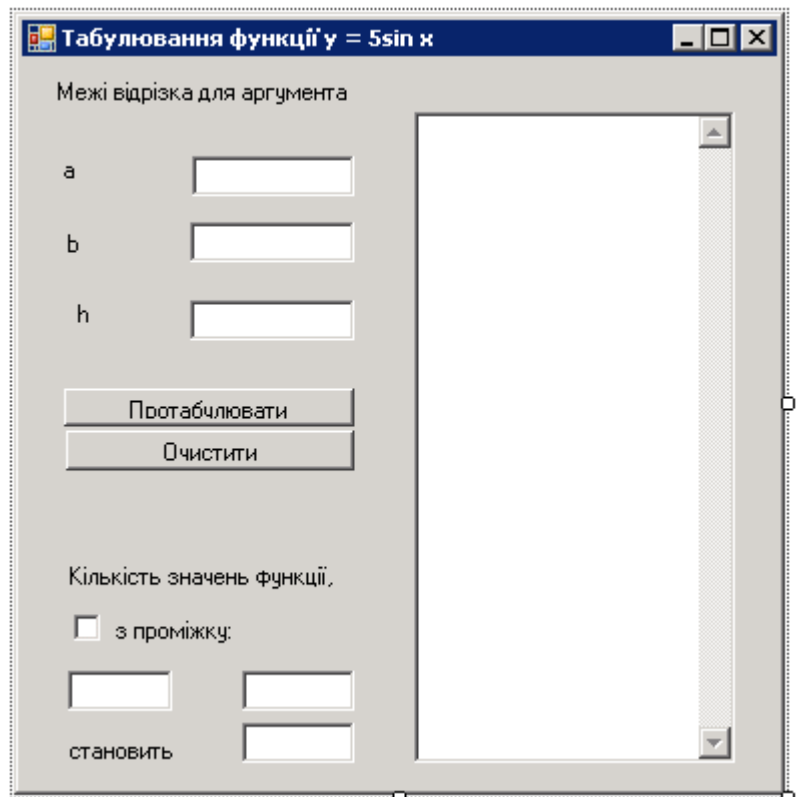
Хід виконання роботи.

1. Запускаємо Microsoft Visual Studio. Створюємо новий проект File=>New=>Project, вибираємо типи проекту CLR, шаблон Windows Forms Application. Задаємо назву проекту, папку на диску де буде збережено проект

2. Створюємо дизайн форми. Орієнтовний вигляд вікна проекту показано на малюнку. В ньому використано 7 текстових полів (textBox), 6 підписів (label), дві кнопки (button) і прапорець checkBox.

3. Для налаштування текстового поля для виводу результатів слід встановити значення true для властивості Multiline (багаторядковий) та Vertical (Вертикальна) для властивості ScrollBar

4. Перейти до написання коду функцій можна подвійним кліком лівої клавіші миші, або вибором команди Показати код (View Code) з контекстного меню елемента. У зразку використано три функції:



А). Функція опрацювання події зміни вибору прапорця.

```
private: System::Void checkBox1_CheckedChanged(System::Object^ sender, System::EventArgs^ e) { }
```

Її завдання вмикати або вимикати поля для розв'язання додаткового завдання. Код цієї функції можна оформити, приблизно так:

```
{ if(checkBox1->Checked) {
    textBox5->Enabled = true;
    textBox6->Enabled = true;
    textBox7->Enabled = true; }
else{textBox5->Enabled = false;
    textBox6->Enabled = false;
    textBox7->Enabled = false; }}
```

Б). Функція очистки текстових полів кнопкою «Очистити».

```
private: System::Void button2_Click(System::Object^ sender, System::EventArgs^ e) { }
```

Її слід заповнити інструкціями виду

```
textBox1->Text = "";
textBox2->Text = "";
...
checkBox1->Checked = false;
```

В). Функція табулювання функції та виводу результатів.

```
private: System::Void button1_Click(System::Object^ sender, System::EventArgs^ e) { }
```

5. Для підключення до проекту математичних функцій слід скористатися конструкцією

```
#define _USE_MATH_DEFINES  
#include <math.h>
```

у заголовку коду форми.

6. Для форматного виводу кожного рядка у результуючу таблицю функції можна скористатися інструкцією методом Format класу String. Код можна оформити наступним чином:

```
textBox4->Text += "    " + String::Format("{0:f}", x) + "    " +  
String::Format("{0:f}", y) + "\r\n";
```

7. Слід врахувати в коді останньої функції, що додаткове завдання може бути вимкнене. Тобто, відповідні місця коду, які стосуються додаткового завдання помістити в умовний оператор виду:

```
if (checkBox1->Checked) {    }
```