

Лабораторна робота 5

Тема. Створення настільного додатку на основі технології Microsoft .NET з використанням абстрактних класів на мові C++.

Мета. Ознайомитись із можливостями створення та використання абстрактних класів у Visual Studio C++. Навчитись створювати дочірні MDI-форми.

Хід роботи

1 Приклад виконання завдання

Завдання. Створити настільний додаток на основі технології Microsoft .NET з використанням мови CLR (Common Language Runtime) для розв'язку наступного завдання: нехай дано абстрактний клас та два похідних для реалізації рисування графіків функцій.

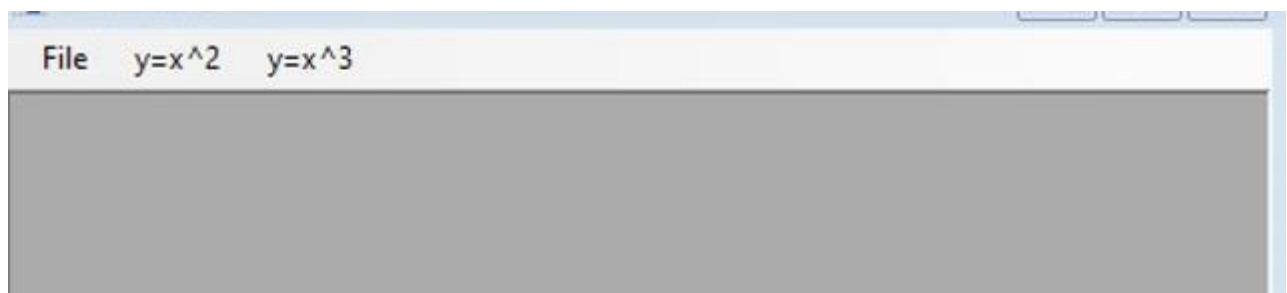


Рисунок 5.1 – Головне меню програми

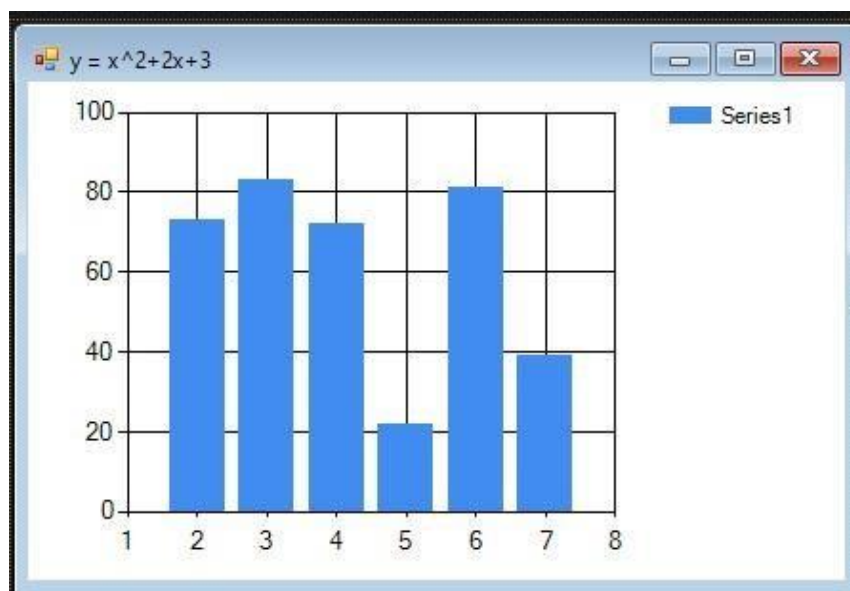


Рисунок 5.2 – Форма для виведення графіка 1 функції

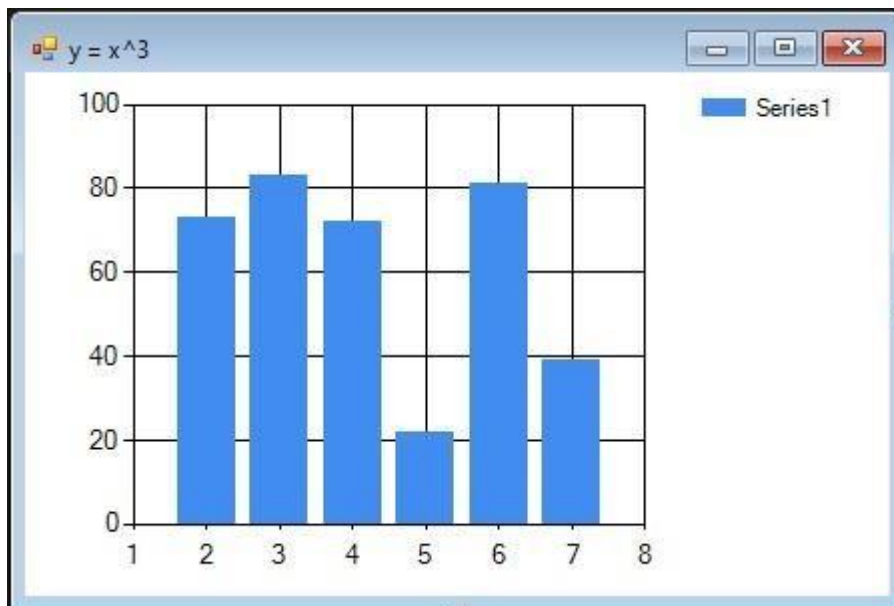


Рисунок 5.3 – Форма для виведення графіка 2 функції

Лістинг 5.1 – Header.h

```
#pragma once
#define n 9
class function
{
protected:
    double x;
public:
    void SetX(double _x) {
        x = _x;
    }
    double GetX() {
        return x;
    }
    virtual void SetY() = 0;
};

class function1 : public function {
    double y;
public:
    double GetY() {
        return y;
    }
}

void SetY() {
    y = x * x + 2 * x + 3;
}
par[n];

class function2 : public function {
    double y;
public:
    double GetY() {
        return y;
    }
    void SetY() {
        y = x * x * x;
    }
}
kub[n];
```

Лістинг 5.2 – menuStrip1

```

private:      System::Void      yx2ToolStripMenuItem_Click(System::Object^
sender,
    System::EventArgs^ e) {
    MyForm1^ form1 = gcnew MyForm1();    form1-
    >MdiParent = this;    form1->Dock                                =
System::Windows::Forms::DockStyle::Fill;                                form1->chart1-
>Series["Series1"]->XValueMember = "Значення x";
    form1->chart1->Series["Series1"]->YValueMembers    = "Значення
y(y)";
    form1->chart1->Series["Series1"]->ChartType =
    SeriesChartType::Line;    form1->chart1->Series["Series1"]-
    >Color = Color::Red;    form1->chart1-
>Series["Series1"]>IsVisibleInLegend = false;    form1->chart1-
>Visible = true;    for (int i = 0; i < n; i++) {    form1-
>chart1->Series[0]->Points->AddXY(par[i].GetX(),
    par[i].GetY());
    }
    form1->Show();
}
private:      System::Void
inputToolStripMenuItem_Click(System::Object^
sender, System::EventArgs^ e) {
    array<String^>^ Spisok = System::IO::File::ReadAllLines("mas.txt");
    int i = 0;
    for each (String ^ str in Spisok) {    for (int j = 0;
    j < str->Split('\t')->Length; j++) {    double x
= Convert::ToInt32(str->Split('\t')[j]);
    par[i].SetX(x); kub[i].SetX(x);
    par[i].SetY(); kub[i].SetY();
    i++;
    }
    }
}
private:      System::Void
closeToolStripMenuItem_Click(System::Object^
sender, System::EventArgs^ e) {    this->Close();
}
private:      System::Void      yx3ToolStripMenuItem_Click(System::Object^
sender,
    System::EventArgs^ e) {
    MyForm2^ form2 = gcnew MyForm2();    form2-
    >MdiParent = this;    form2->Dock                                =
System::Windows::Forms::DockStyle::Fill;                                form2->chart1-
>Series["Series1"]->XValueMember = "Значення x";
    form2->chart1->Series["Series1"]->YValueMembers    = "Значення
y(y)";
    form2->chart1->Series["Series1"]->ChartType =

```

```
SeriesChartType::Line; form2->chart1->Series["Series1"]-  
>Color = Color::Red; form2->chart1-  
>Series["Series1"]>IsVisibleInLegend = false; form2->chart1-  
>Visible = true; for (int i = 0; i < n; i++) { form2-  
>chart1->Series[0]->Points->AddXY(kub[i].GetX(),  
kub[i].GetY());  
}  
form2->Show();  
}  
};
```



Рисунок 5.4 – Точки

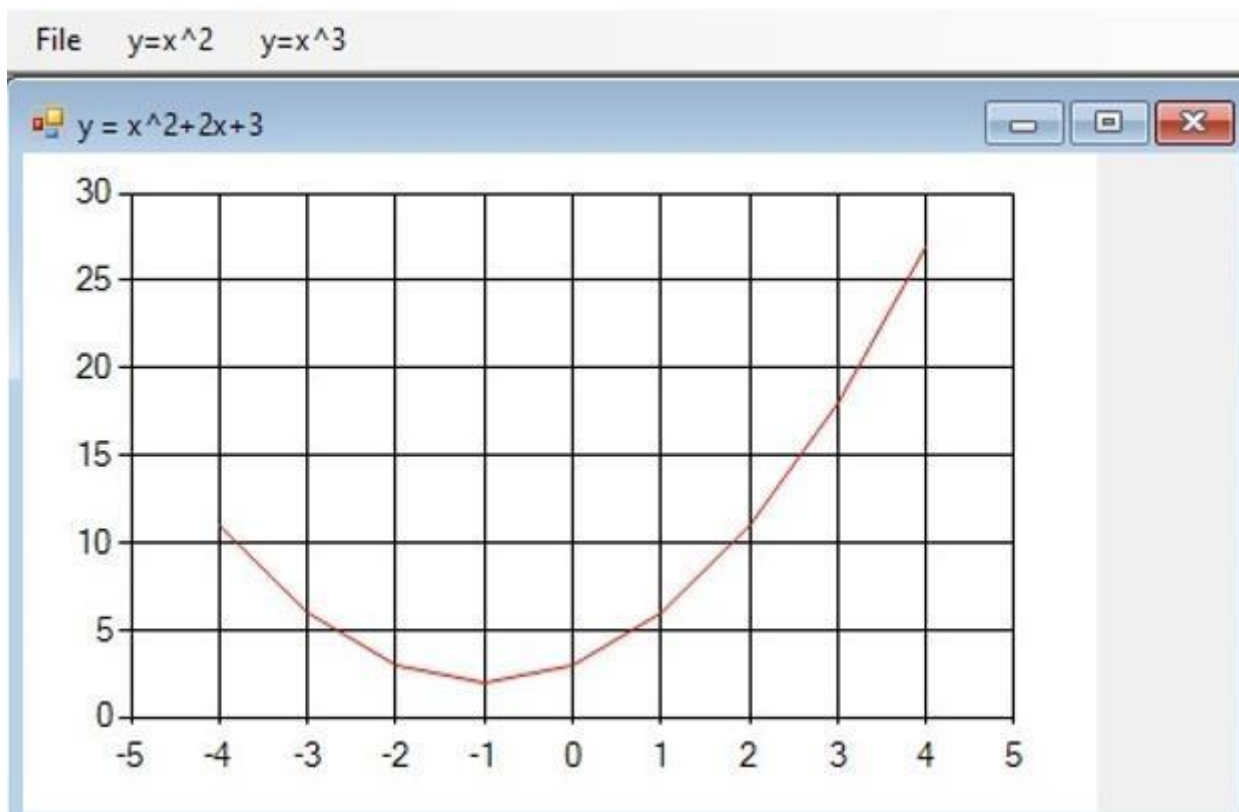


Рисунок 5.5 – Робота кнопки для виведення графіка 1 функції

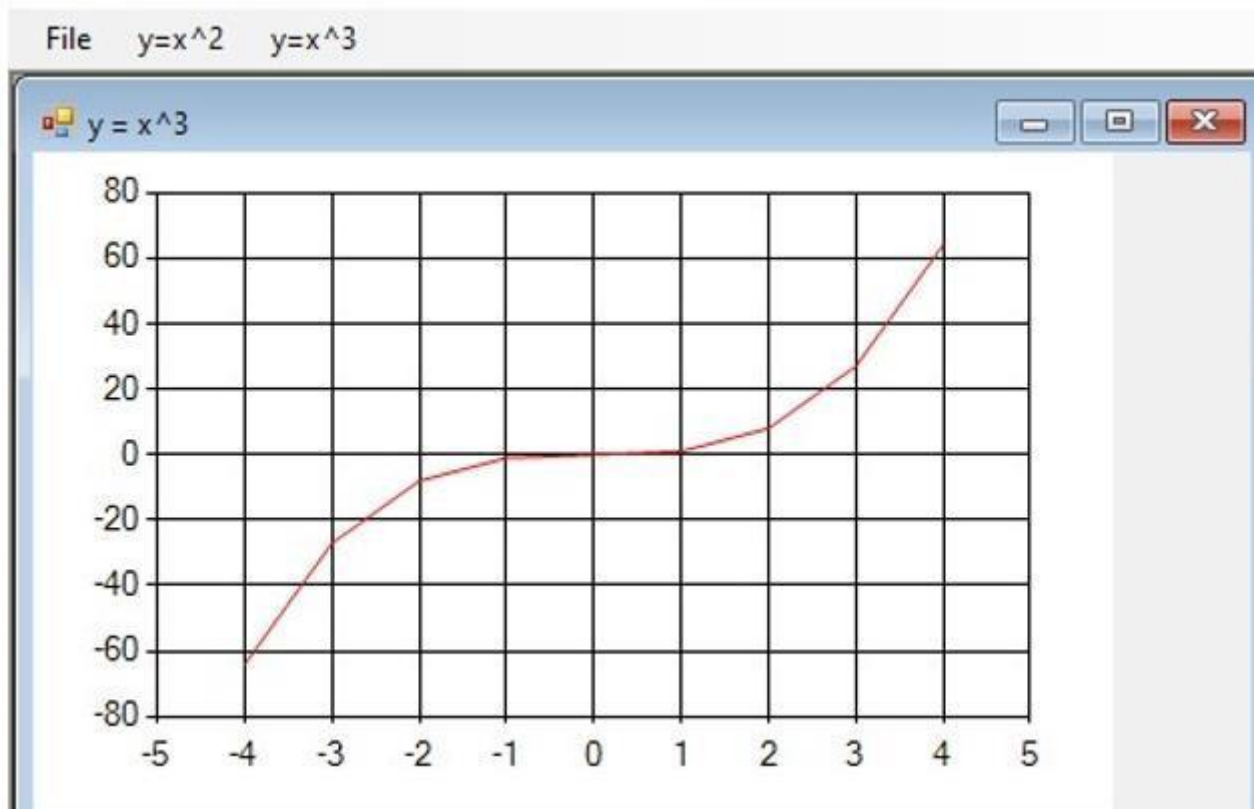


Рисунок 5.6 – Робота кнопки для виведення графіка 2 функції

Висновок: на лабораторній роботі я ознайомилася із можливостями створення та використання абстрактних класів у Visual Studio C++. Навчилася створювати дочірні MDI-форми.