

Факультет комп'ютерно-інтегрованих технологій,  
мехатроніки і робототехніки

кафедра механічної інженерії

Звіт з практичних робіт  
з навчальної вибіркової дисципліни  
**«Мови програмування в механіці  
технічних систем: C++; C#)»**

Виконав: студент групи ПМ-140

Фомюк М.С

Перевірив:

Мудревський П.І

Житомир

2020 р.

# Практичне заняття № 1

**Мета:** набути навичок створення консольних і графічних застосунків на мові C# в середовищі Microsoft Visual Studio 2019.

## Завдання :

### Лістинг програми:

```
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 lab1
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void label1_Click(object sender, EventArgs e)
        {
        }

        private void Form1_Load(object sender, EventArgs e)
        {
        }

        private void ok_Click(object sender, EventArgs e)
        {
            MessageBox.Show("Hello " + userName.Text);
        }
    }
}
```

### Результат виконання програми:

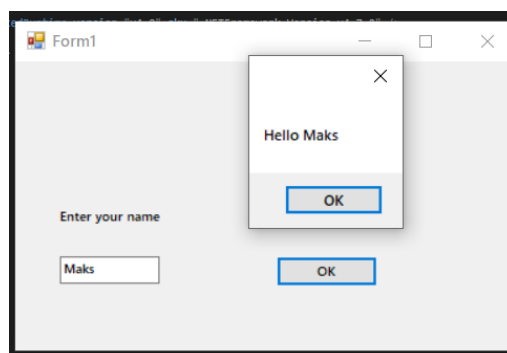


Рис. 1. Результат виконання програми

**Висновок:** на цьому практичному занятті я набув навичок створення консольних і графічних застосунків на мові C# в середовищі Microsoft Visual Studio 2019.

## Практичне заняття № 2

Мета роботи: ознайомитися з теоретичними відомостями про загальну бібліотеку класів

.NET та типи даних C#

### Завдання 1:

Лістинг програми:

```
using System;

namespace Pract2
{
    class TestMain
    {
        public static void Main() // Code Starts Here
        {
            bool b = true; System.Console.WriteLine(b); // boolean
            byte by = 200; System.Console.WriteLine(by); // byte
            sbyte sby = -100; System.Console.WriteLine(sby); // signed byte
            decimal dec = 10m; System.Console.WriteLine(dec); // decimal (28-29 signif)
            double dbl = 12.2e23; System.Console.WriteLine(dbl); // double prec. float fl=12.2f;
            System.Console.WriteLine(fl); // single prec.

            char c = 'A'; System.Console.WriteLine(c); // unicode character
            short sh = 10; System.Console.WriteLine(sh); // signed 16 bit
            ushort ush = 10; System.Console.WriteLine(ush); // unsigned 16 bit
            int i = 10; System.Console.WriteLine(i); // signed 32 bit
            uint ui = 10; System.Console.WriteLine(ui); // unsigned 32 bit
            long l = 10; System.Console.WriteLine(l); // signed 64 bit
            ulong ul = 10; System.Console.WriteLine(ul); // unsigned 64 bit
        }
    }
}
```

Результат виконання програми:

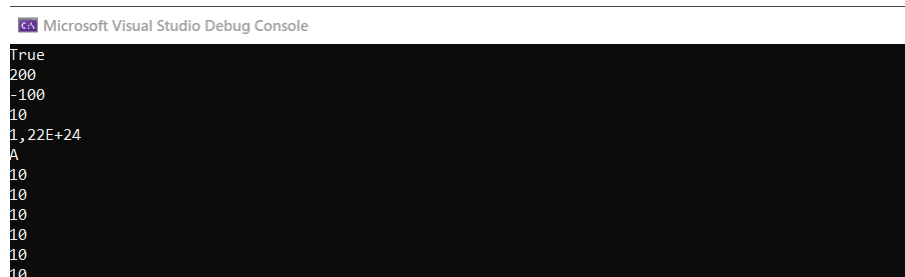


Рис. 2.1 Результат виконання програми

### Завдання 2:

Лістинг програми:

```
using System;

namespace Pract2._2z
{
    class TestMain
    {
        public static void Main() // Code Starts Here
        {
            int x;
            x = 'A'; // implicit cast (неявне приведення)
        }
    }
}
```

```

    x = (int)34.34F; // explicit cast (явне приведення)
    decimal dec = x; // implicit cast
    dec = (decimal)6.673e-11; //explicit cast
    dec = 12 + 'A'; // implicit cast
    System.Console.WriteLine(dec);
}
}
}

```

Результат виконання програми:

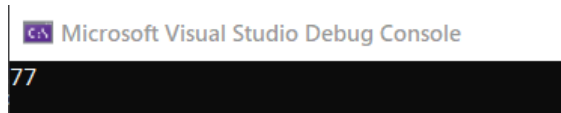


Рис. 2.2 Результат виконання програми

**Висновок:** на цьому практичному занятті я ознайомитися з теоретичними відомостями про загальну бібліотеку класів .NET та типи даних C#.

## Практичне заняття № 3

**Мета роботи:** ознайомитися з теоретичними відомостями про загальну роботу з генератором випадкових чисел на мові програмування C#.

### Завдання :

Лістинг програми:

```

using System;

namespace Pract3
{
    class Program
    {
        static void Main(string[] args)
        {
            Random rnd = new Random();
            int s = rnd.Next(0, 100);
            Console.WriteLine(s);
            float f = (float)rnd.Next(0, 1000);
            Console.WriteLine(f/100);
            char rc = (char)rnd.Next('a', 'z');
            Console.WriteLine(rc);
            string text = "";
            for(int i = 0; i<=9; i++)
            {
                char st = (char)rnd.Next('a', 'z');
                text += st;
            }
            Console.WriteLine(text);
            int randomNumber = rnd.Next(0, 10);
            if (randomNumber < 5)
            {
                Console.WriteLine("false");
            }
            else Console.WriteLine("true");
        }
    }
}

```

```

    }
}
}

```

### Результат виконання програми:

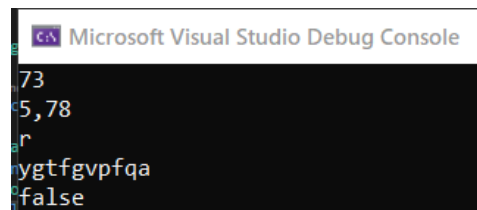


Рис. 3 Результат виконання програми

Висновок: на цьому практичному занятті я ознайомитися з теоретичними відомостями про загальну роботу з генератором випадкових чисел на мови програмування C#.

## Практичне заняття №4

**Мета роботи:** продовжити ознайомлення з генератором випадкових чисел, застосувати його для виведення випадкових символів і рядків.

### Завдання:

4. Вивести довжину прізвища і імені

```

using System;

namespace Pract4
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.OutputEncoding = System.Text.Encoding.UTF8;
            string str = "Фомюк Максим Сергійович";
            string f1; // Прізвище

            string f2; // Ім'я
            string f3; // По-батькові
            int i;
            int k = str.Length; // Довжина рядка
            int k1;
            int k2; // Довжина імені
            int n; // Кількість букв "А" в рядку
            string[] strWords;

            Console.WriteLine("Довжина рядка = " + k + "\n ");
            // Визначаємо кількість букв "А" в рядку
            n = 0;
            for (i = 0; i <= (k - 1); i++)
            {
                if ((str[i] == 'A') || (str[i] == 'a')) n = n + 1;
            }
        }
    }
}

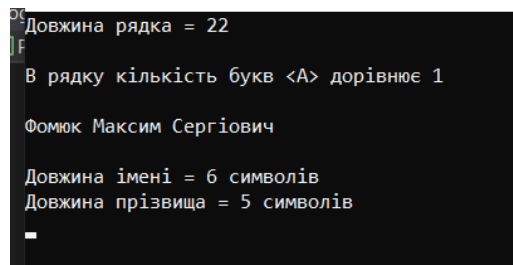
```

```

        Console.WriteLine("В рядку кількість букв <A> дорівнює " + n + "\n");
        // Рядок ділимо на складові частини.
        // В підсумку повертається масив, що містить підрядки
        strWords = str.Split(" ".ToCharArray());
        f1 = strWords[0];
        f2 = strWords[1];
        f3 = strWords[2];
        Console.WriteLine(f1 + " " + f2 + " " + f3 + " " + "\n"); //Визначаємо довжину імені
        k1 = f2.Length;
        k2 = f1.Length;
        Console.WriteLine("Довжина імені = " + k1 + " символів");
        Console.WriteLine("Довжина прізвища = " + k2 + " символів");
        /* Виводимо першу букву по-батькові
        Console.WriteLine(f3[0]); */
        Console.ReadLine();
    }
}

```

Результат виконання програми:



```

Довжина рядка = 22
В рядку кількість букв <A> дорівнює 1
Фомюк Максим Сергійович
Довжина імені = 6 символів
Довжина прізвища = 5 символів

```

Рис. 4 Результат виконання програми

Висновок: на цьому практичному занятті я продовжив ознайомлення з генератором випадкових чисел, застосував його для виведення випадкових символів та рядків.

## Практичне заняття №5

**Мета роботи:** ознайомитися з функціональними можливостями ряда елементів Windows Forms, реалізувати програму на мові програмування C#, яка моделює роботу з генератором випадкових чисел

### Завдання :

Лістинг програми:

```

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 Pract5
{
    public partial class Form1 : Form
    {
        public Form1()
        {

```

```

        InitializeComponent();
    }

    private void button1_Click(object sender, EventArgs e)
    {
        string dataType = listBox1.SelectedItem.ToString();
        Random ran1 = new Random();

        switch (dataType)
        {
            case "Int":
                int num1 = ran1.Next(0, 100);
                textBox1.Text = num1.ToString();
                break;

            case "Float":
                int num2 = ran1.Next(0, 10000);
                textBox1.Text = (num2/1000).ToString("F3");
                break;

            case "Char":
                char rc = (char)ran1.Next('a', 'z');
                textBox1.Text = rc.ToString();
                break;

            case "String":
                string text = "";
                for (int i = 0; i <= 9; i++)
                {
                    char st = (char)ran1.Next('a', 'z');
                    text += st;
                }
                textBox1.Text = text.ToString();
                break;

            case "Bool":
                int randomNumber = ran1.Next(0, 10);
                if (randomNumber < 5)
                {
                    textBox1.Text = "false";
                }
                else
                {
                    textBox1.Text = "true";
                }
                break;
        }
    }

}

private void listBox1_SelectedIndexChanged(object sender, EventArgs e)
{
}

private void label2_Click(object sender, EventArgs e)
{
}

}

```

Результат виконання програми:

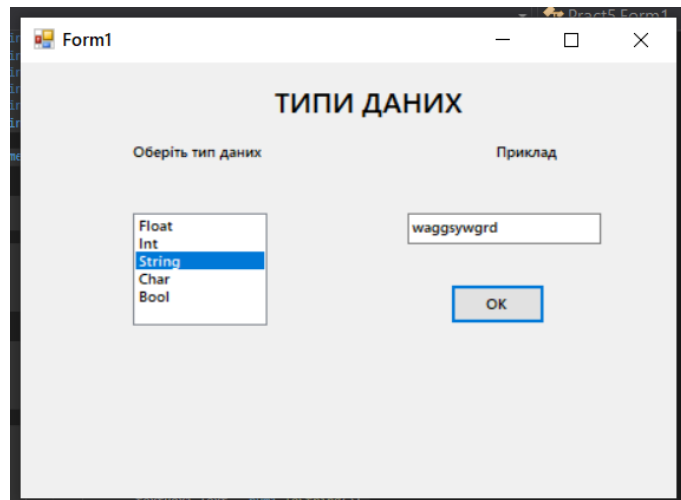


Рис. 5 Результат виконання програми

**Висновок:** на цьому практичному занятті я ознайомився з функціональними можливостями ряду елементів Windows Forms, реалізував програму на мові програмування C#, яка моделює роботу з генератором випадкових.

## Практичне заняття №6

**Мета:** вивчення засобів C# для роботи з одновимірними масивами.

### Завдання:

**10.** Дан масив  $Z$  цілих чисел, що містить 35 елементів. Обчислити і вивести  $R=S+P$ , где  $S$  – сума парних елементів, менших за 3,  $P$  – добуток непарних елементів, більших за 1.

Лістинг програми:

```
using System;
using System.Text;

namespace Pract6
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.OutputEncoding = Encoding.Unicode;
            Random rnd = new Random();
            int[] MyMassiv = new int[35];
            int i, s, f, r;

            for (i = 0; i < MyMassiv.Length; i++)
```

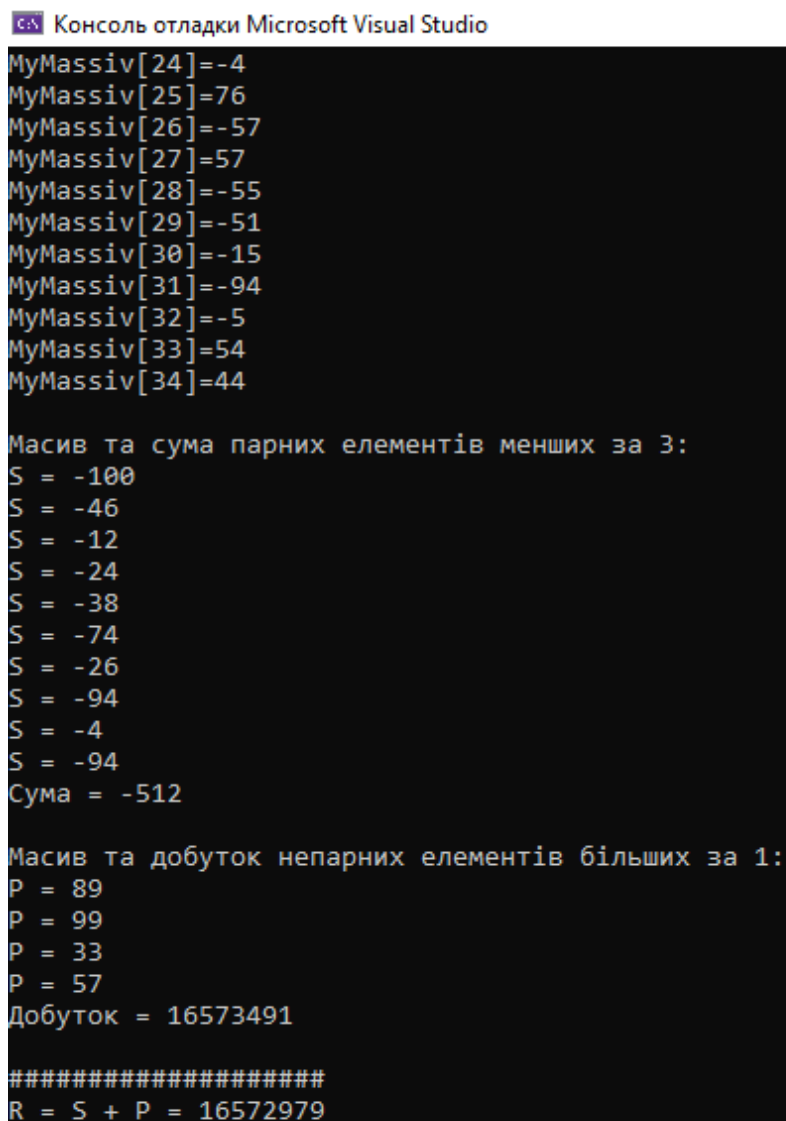


```

{
    MyMassiv[i] = rnd.Next(-100, 100);           //присвоюємо ітому
елементу рандомне значення
    Console.WriteLine("MyMassiv[{0}]= {1}", i, MyMassiv[i]);
}
s = 0;
f = 0;
for (i = 1; i < MyMassiv.Length; i++)
{
    if (MyMassiv[i] % 2 == 0 & MyMassiv[i] < 3)
        s = MyMassiv[i] + s;
    else if (MyMassiv[i] % 2 > 0 & MyMassiv[i] > 1)
    {
        f = MyMassiv[i] + f;
    }
}
r = s + f;
Console.WriteLine("#####");
Console.WriteLine(r);
}
}
}

```

### Результат виконання програми:



```

Консоль отладки Microsoft Visual Studio
MyMassiv[24]=-4
MyMassiv[25]=76
MyMassiv[26]=-57
MyMassiv[27]=57
MyMassiv[28]=-55
MyMassiv[29]=-51
MyMassiv[30]=-15
MyMassiv[31]=-94
MyMassiv[32]=-5
MyMassiv[33]=54
MyMassiv[34]=44

Масив та сума парних елементів менших за 3:
S = -100
S = -46
S = -12
S = -24
S = -38
S = -74
S = -26
S = -94
S = -4
S = -94
Сума = -512

Масив та добуток непарних елементів більших за 1:
P = 89
P = 99
P = 33
P = 57
Добуток = 16573491

#####
R = S + P = 16572979

```

Рис. 6 Результат виконання програми

Висновок: на даній практичній роботі я, вивчив принцип роботи з одновимірними масивами.

## Практичне заняття №7

**Мета роботи:** ознайомитися з функціональними можливостями ряда елементів Windows Forms, реалізувати програму для роботи з одновимірними масивами на мові програмування C#, з використанням генератора випадкових чисел.

### Завдання:

*10. Дан масив Z цілих чисел, що містить 35 елементів. Обчислити і вивести  $R=S+P$ , где  $S$  – сума парних елементів, менших за 3,  $P$  – добуток непарних елементів, більших за 1.*

Лістинг програми:

```
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 Pract7
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        int[] MyMassiv = new int[35];
        int i, s, f, r;
        private void textBox2_TextChanged(object sender, EventArgs e)
        {
        }

        private void button1_Click(object sender, EventArgs e)
        {
            Random rn = new Random();
            textBox1.Text = "";
            for (i = 0; i < MyMassiv.Length; i++)
            {
                MyMassiv[i] = rn.Next(-50, 50);
                textBox1.Text += "MyMassiv[" + Convert.ToString(i) + "] = " +
                Convert.ToString(MyMassiv[i]) + Environment.NewLine;
            }
        }
        private void button2_Click(object sender, EventArgs e)
        {
            textBox2.Text = "";
            s = 0;
        }
    }
}
```

```

f = 0;
for (i = 1; i < MyMassiv.Length; i++)
{
    if (MyMassiv[i] % 2 == 0 & MyMassiv[i] < 3)
        s = MyMassiv[i] + s;
    else if (MyMassiv[i] % 2 > 0 & MyMassiv[i] > 1)
    {
        f = MyMassiv[i] + f;
    }
}
r = s + f;
textBox2.Text += "Сума = " + Convert.ToString(r) + Environment.NewLine;
}
}
}

```

Результат виконання програми:

Form1

Початковий масив

- MyMassiv[0] = -32
- MyMassiv[1] = 2
- MyMassiv[2] = 27
- MyMassiv[3] = 35
- MyMassiv[4] = 9
- MyMassiv[5] = 10
- MyMassiv[6] = -43
- MyMassiv[7] = -6
- MyMassiv[8] = 15
- MyMassiv[9] = -28
- MyMassiv[10] = 26
- MyMassiv[11] = -33
- MyMassiv[12] = 48

Замінити

Змінений масив

- P = 9
- P = 15
- P = 19
- P = 27
- P = 29
- P = 31
- P = 33
- P = 45
- P = 27
- P = 13
- Добуток: 374739167
- R = S + P = 374739075

Замінити

Рис. 7 Результат виконання програми

Висновок: на цій практичній роботі я ознайомився з функціональними можливостями ряду елементів Windows Forms, реалізував програму для роботи з одновимірними масивами на мові програмування C#, з використанням генератора випадкових чисел.

## Практичне заняття № 8

### Двовимірні масиви

**Мета:** вивчити оператори оголошення двовимірних масивів, способи наповнення масивів, введення-виведення елементів масивів, накопичення суми і добутку елементів масивів.

#### Завдання 1:

### Варіант 3:

**Задача 3** Створити двовимірний масив  $A$  розміром  $5 \times 6$ ,

$$A = (a_{ij}; i = 0, 1, \dots, 4; j = 0, 1, \dots, 5)$$

Елементи масиву розрахувати за формулою:

$$a_{ij} = \frac{i-j}{2,5i^2+j+1}.$$

Обчислити кількість елементів, що дорівнюють нулю і визначити їх індекси. Елементи, що дорівнюють нулю замінити на значення 10.

---

### Лістинг програми:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Pract8._1
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.OutputEncoding = Encoding.Unicode;
            Console.InputEncoding = Encoding.Unicode;
            const int n = 5; // кількість рядків масиву A
            const int m = 6; // кількість стовпців масиву A
            int i, j;
            double s;
            double[,] A = new double[n, m];
            //Обчислення елементів масиву за формулою і виведення їх на екран
            Console.WriteLine("Масив A ");
            for (i = 0; i < n; i++)
            {
                for (j = 0; j < m; j++)
                {
                    A[i, j] = (double)(i - j)/(2.5*i*i+j+1);
                    Console.Write(A[i, j].ToString("F2") + "\t " + "\t");
                }
                Console.WriteLine();
            }
            Console.WriteLine();
            Console.WriteLine("Індекси нульових елементів:");

            s = 0;
            for (i = 0; i <= (n - 1); i++)
            {
                for (j = 0; j <= m - 1; j++)
                {
                    if(A[i, j]==0)
                    {
                        s ++ ;
                        Console.Write((i+1) + ";" + (j+1) + "\t ");
                        A[i, j] = 10;
                    }
                }
            }

            Console.WriteLine();
            Console.WriteLine("Кількість нульових елементів:{0}", s);
        }
    }
}
```



```

double[,] A = new double[n, m]
{{1, 3, 5.4, 8.1, 10.3, 15}, {-7, 21, -15, 23, 27.5, 34 }, {-54, 23.7, -100, 45.8, -
22.5, 105}}};
Console.WriteLine("Масив A: ");
for (i = 0; i < n; i++)
{
    for (j = 0; j < m; j++)
        Console.Write("{0}\t", A[i, j]);
    Console.WriteLine();
}
Console.WriteLine();
int kv; //кількість від'ємних елементів в кожному рядку
double s; //сума від'ємних елементів
double sr; //середнє арифметичне від'ємних елементів
s = 0;
kv = 0;
for (i = 0; i < n; i++)
{
    for (j = 0; j < m; j++)
    {
        if (A[i, j] > 0)
        {
            s = s + A[i, j];
            kv = kv + 1;
        }
    }
}
sr = s / kv;
Console.WriteLine("Середнє арифметичне ДОДАТНІХ елементів = {0,7:F3}", sr);
Console.ReadLine();
}
}
}

```

Результат виконання програми:

```

Масив A:
1      3      5,4      8,1      10,3      15
-7      21     -15      23      27,5      34
-54     23,7    -100     45,8     -22,5     105

Середнє арифметичне ДОДАТНІХ елементів = 24,831

```

Рис. 8.2 Результат виконання програми

## Практичне заняття №10

**Мета:** вивчити можливості побудови графіків за допомогою елементу керування Chart. Написати і налагодити програму побудови на екрані графіка заданої функції.

### Завдання:

Варіант 3:

$$3. \quad y = (x - 2)^2 + 3.$$

$$y = x^2 + 3x$$

### Лістинг програми:

```
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 Pract10
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            // Зчитуємо з форми потрібні значення
            double Xmin = double.Parse(textBox1.Text);
            double Xmax = double.Parse(textBox2.Text);
            double Step = double.Parse(textBox3.Text);
            // Кількість точок графіку
            int count = (int)Math.Ceiling((Xmax - Xmin) / Step) + 1;
            // Масив значень X - спільний для обох графіків
            double[] x = new double[count];
            // Два масиви Y - по одному для кожного графіка
            double[] y1 = new double[count];
            double[] y2 = new double[count];
            // Розраховуємо точки для графіків функції
            for (int i = 0; i < count; i++)
            {
                // Обчислюємо значення X
                x[i] = Xmin + Step * i;
                // Обчислюємо значення функцій в точці X
                y1[i] = Math.Pow(x[i] - 2, 2) + 3;
                y2[i] = Math.Pow(x[i], 2) + (3 * x[i]);
            }
            // Налаштовуємо осі графіку
            chart1.ChartAreas[0].AxisX.Minimum = Xmin; chart1.ChartAreas[0].AxisX.Maximum = Xmax;
            // Визначаємо крок сітки
            chart1.ChartAreas[0].AxisX.MajorGrid.Interval = Step;
            // Додаємо обчислення значень в графіки
            chart1.Series[0].Points.DataBindXY(x, y1);
            chart1.Series[1].Points.DataBindXY(x, y2);
        }

        private void textBox1_TextChanged(object sender, EventArgs e)
        {
        }
    }
}
```

### Результат виконання програми:

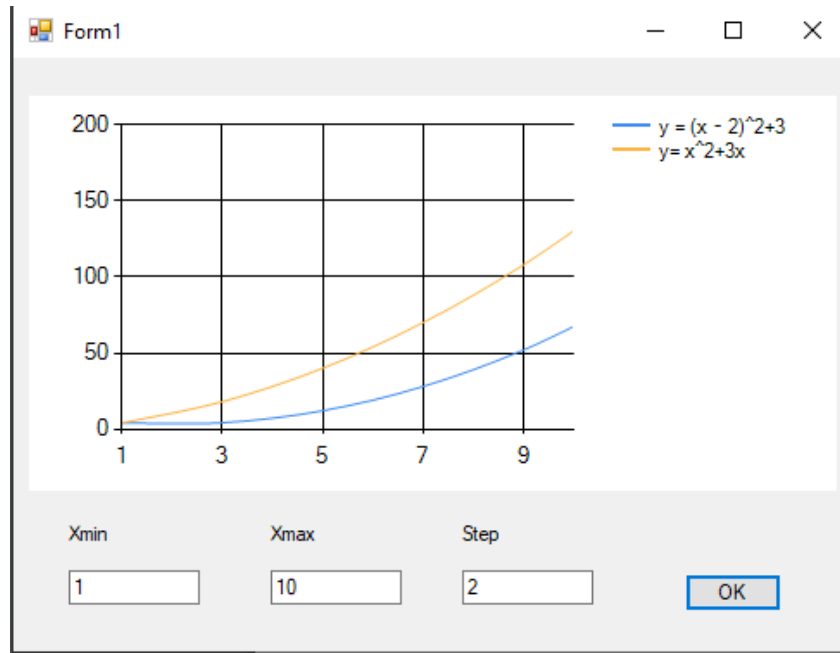


Рис. 10 Результат виконання програми

**Висновок:** я вивчив можливості побудови графіків за допомогою елементу керування Chart. Написав і налагодив програму побудови на екрані графіка заданої функції.

## Практичне заняття №11

### Програмування з використанням одновимірних масивів

**Мета:** Вивчити метод Гауса, один з методів розв'язання лінійних алгебраїчних рівнянь.

Розробити програму реалізації даного методу на мові C#.

створити консольний додаток: вивчити оператори оголошення двовимірних масивів, способи наповнення масивів, введення- виведення елементів масивів, розв'язати систему рівнянь методом Гауса.

для Windows- додатку: вивчити елементи керування: мітка Label, кнопка Button, поле введення TextBox, список ListBox, один компонент для представлення таблиці на формі DataGridView, розв'язати систему рівнянь методом Гауса.

**Завдання :**



3

$$3x_1 + x_2 - 2x_3 = 6$$

$$0.5x_1 + 2x_2 + 0.5x_3 = 9$$

$$x_1 - x_2 + 3x_3 = 9$$

3 Варіант:

Лістинг програми:

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace ConsoleApp24
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.OutputEncoding = Encoding.Unicode;
            Console.InputEncoding = Encoding.Unicode;
            int n;
            double d, s;
            Console.Write("Введіть розмірність: ");
            n = int.Parse(Console.ReadLine());
            double[,] A = new double[n, n];
            double[,] a = new double[n, n];
            double[] B = new double[n];
            double[] b = new double[n];
            double[] x = new double[n];
            Console.WriteLine("Введіть коефіцієнти і вільні члени: ");
            for (int i = 0; i < n; i++)
            {
                for (int j = 0; j < n; j++)
                {
                    Console.Write($"A[{i + 1}][{j + 1}]= ");
                    A[i, j] = double.Parse(Console.ReadLine());
                    a[i, j] = A[i, j];
                }
                Console.Write($"B[{i + 1}]= ");
                B[i] = double.Parse(Console.ReadLine());
                b[i] = B[i];
            }
            for (int k = 0; k < n; k++)
            {
                for (int j = k + 1; j < n; j++)
                {
                    d = A[j, k] / A[k, k];
                    for (int i = k; i < n; i++)
                    {
                        A[j, i] = A[j, i] - d * A[k, i];
                    }
                    B[j] = B[j] - d * B[k];
                }
            }
            for (int k = n - 1; k >= 0; k--)
            {
                d = 0;
                for (int j = k; j < n; j++)
                {
                    s = A[k, j] * x[j];
                    d += s;
                }
                x[k] = (B[k] - d) / A[k, k];
            }
            Console.Clear();
            Console.WriteLine();
            Console.WriteLine("_____");
        }
    }
}

```

```

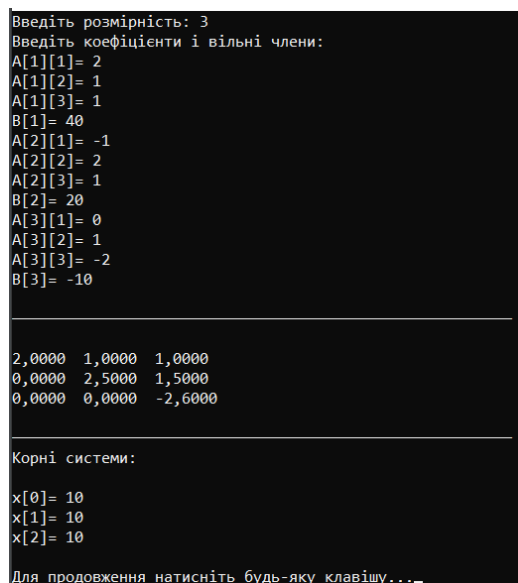
Console.WriteLine();
for (int i = 0; i < n; i++)
{
    for (int j = 0; j < n; j++)
    {
        Console.Write("{0,5:F4}\t", A[i, j]);

    }

    Console.WriteLine();
}
Console.WriteLine();
Console.WriteLine("_____");
Console.WriteLine("Корні системи: ");
Console.WriteLine();
for (int i = 0; i < n; i++)
{
    Console.WriteLine($"x[{i}]= {x[i]}");
}
Console.WriteLine();
Console.Write("Для продовження натисніть будь-яку клавішу...");
Console.ReadKey();
}
}
}

```

### Результат виконання програми:



```

Введіть розмірність: 3
Введіть коефіцієнти і вільні члени:
A[1][1]= 2
A[1][2]= 1
A[1][3]= 1
B[1]= 40
A[2][1]= -1
A[2][2]= 2
A[2][3]= 1
B[2]= 20
A[3][1]= 0
A[3][2]= 1
A[3][3]= -2
B[3]= -10

2,0000  1,0000  1,0000
0,0000  2,5000  1,5000
0,0000  0,0000 -2,6000

Корні системи:
x[0]= 10
x[1]= 10
x[2]= 10
Для продовження натисніть будь-яку клавішу...

```

Рис. 11 Результат виконання програми

### Лістинг програми:

```

// Програма для розв'язку системи лінійних рівнянь. Введення коефіцієнтів
// передбачено через DataGridView
using System;
using System.Data; // - для DataGridView
using System.Windows.Forms;
// Додавимо директиву Globalization:
using System.Globalization;
// Інші директиви using видалені, оскільки вони не використовуються в даній
// програмі
namespace ГайсGrid
{
    public partial class Form1 : Form
    {
        // Дані змінні оголошуємо поза всіх процедур, щоб
        // їх було видно з будь-якої з процедур:

```

```

int n; // - розмірність СЛАУ
DataTable Таблица;
public Form1()
{
    InitializeComponent();
}
private void Form1_Load(object sender, EventArgs e)
{
    this.Text = "Розв'язання системи рівнянь";
    // Щоб при старті програми фокус знаходився в текстовому полі:
    textBox1.TabIndex = 0;
    dataGridView1.Visible = false; // сітку даних поки не видно
    label1.Text = "Ведіть кількість невідомих:";
    button1.Text = "Ввести"; // - початковий напис на кнопці
    Таблица = new DataTable();
}
private void button1_Click(object sender, EventArgs e)
{
    int i, j;
    // Матриця коефіцієнтів лінійних рівнянь:
    Double[, ] A;
    // Вектор вільних членів :
    Double[] L;
    // ознака введення числових даних:
    var Число_ли = false;
    var tmp = "тимчасова робоча змінна";
    if (button1.Text == "Ввести")
        for (; ; )
        {
            // Нескінченний цикл, поки користувач не введе
            // саме число:
            Число_ли = int.TryParse(textBox1.Text,
                NumberStyles.Integer,
                NumberFormatInfo.CurrentInfo,
                out n);
            if (Число_ли == false) return;
            // Задаємо інший напис на кнопці:
            button1.Text = "Розв'язати";
            // Тепер текстове поле недоступне:
            textBox1.Enabled = false;
            // Тепер сітку даних видно:
            dataGridView1.Visible = true;
            dataGridView1.DataSource = Таблица;
            // Створюємо "шапку" таблиці
            for (i = 1; i <= n; i++)
            {
                tmp = "X" + Convert.ToString(i);
                Таблица.Columns.Add(new DataColumn(tmp));
            }
            // Колонка правої частини системи:
            Таблица.Columns.Add(new DataColumn("L"));
            return;
        } // - кінець тіла вічного циклу
    else // - button1.Text = "Розв'язати"
    { // Натиснули кнопку "Розв'язати"
        // Таблица.Rows.Count - кількість рядків
        if (Таблица.Rows.Count != n)
        {
            MessageBox.Show(
                "Кількість рядків не дорівнює кількості стовпців");
            return;
        }
        // Тепер можемо визначитись з розмірністю масивів:
        // Матриця коефіцієнтів лінійних рівнянь:
        A = new Double[n, n];
        // Вектор вільних членів:
        L = new Double[n];
        // Заповнення матриці коефіцієнтів системи A[j, i]
        for (j = 0; j <= n - 1; j++)
    }

```

```

    {
        for (i = 0; i <= n - 1; i++)
        {
            A[j, i] = ВернутьЧисло(j, i, ref Число_ли);
            if (Число_ли == false) return;
        } // - кінець тіла внутрішнього циклу по i
        // Права частина системи B[j, 0]
        L[j] = ВернутьЧисло(j, i, ref Число_ли);
        if (Число_ли == false) return;
    } // - кінець тіла зовнішнього циклу по j
} // - button1.Text = "Розв'язати"
// ~ ~ ~ ~ ~
// Розв'язання системи  $A \cdot x = L$  методом Гаусса:
gauss(n, A, ref L);
// L - вектор вільних членів системи, сюди
// ж повертається розв'язок x
var s = "Невідомі дорівнюють:\n";
for (j = 1; j <= n; j++)
{
    tmp = L[j - 1].ToString();
    s = s + "X" + j.ToString() + " = " + tmp + ";\n";
} // - кінець циклу по j
MessageBox.Show(s);
}
Double ВернутьЧисло(int j, int i,
ref Boolean Число_ли)
{
    // j - номер рядка, i - номер стовпця
    // Передаємо аргумент Число_ли за посиланням
    Double rab; // - робоча змінна
    var tmp = Таблиця.Rows[j][i].ToString();
    Число_ли = Double.TryParse(tmp,
    NumberStyles.Number,
    NumberFormatInfo.CurrentInfo,
    out rab);
    if (Число_ли == false)
    {
        tmp = String.Format("Номер рядка {0}, номер стовпця " +
        "{1}, " + "\n в даному полі - не число", j + 1, i + 1);
        MessageBox.Show(tmp);
    }
    return rab;
}
void gauss(int n, double[,] A, ref double[] LL)
{
    // n - розмір матриці
    // A - матриця коефіцієнтів лінійних рівнянь
    // LL - права частина, сюди ж повертаються значення невідомих
    int i, j, l = 0;
    Double c1, c2, c3;
    for (i = 0; i <= n - 1; i++) // Цикл за елементами рядка
    {
        c1 = 0;
        for (j = i; j <= n - 1; j++)
        {
            c2 = A[j, i];
            if (Math.Abs(c2) > Math.Abs(c1))
            {
                l = j; c1 = c2;
            }
        }
        for (j = i; j <= n - 1; j++)
        {
            c3 = A[l, j] / c1;
            A[l, j] = A[i, j]; A[i, j] = c3;
        } // j
        c3 = LL[l] / c1; LL[l] = LL[i]; LL[i] = c3;
        for (j = 0; j <= n - 1; j++)
        {

```

```

        if (j == i) continue;
        for (l = i + 1; l <= n - 1; l++)
        {
            A[j, l] = A[j, l] - A[i, l] * A[j, i];
        } // l
        LL[j] = LL[j] - LL[i] * A[j, i];
    } // j
} // i
}
}

```

**Висновок:** на цьому практичному занятті я вивчив оператори оголошення двовимірних масивів, способи наповнення масивів, введення-виведення елементів масивів, розв'язав систему рівнянь методом Гауса.