

# ЛАБОРАТОРНА РОБОТА №5

**Тема:** Робота з одновимірними масивами.

## Мета роботи

Дослідити роботу операторів циклу при роботі з одновимірними масивами. Навчитися використовувати компонент DataGridView.

## Постановка завдання

Розробити проект з інтерфейсом користувача, використовуючи необхідні компоненти та оброблювач подій мовою C#.

## Теоретичні відомості

### 1. Масиви

Масиви використовують для зберігання однотипних даних. Кількість об'єктів, які зберігає масив, не може змінюватися. Елементи масиву можуть мати будь-який тип. Вказівник на масив оголошується так:

тип[] ім'я;

Щоб далі працювати з масивом, необхідно виділити для нього пам'ять:

ім'я = new тип[розмір];

Якщо потрібно відразу задати елементи, то це можна зробити так:

тип[] ім'я = {елемент\_1, елемент\_2, ..., елемент\_n};

Тоді масив буде мати n елементів.

Доступ до елементів масиву здійснюється за ім'ям масиву та номеру елемента, вказаного у квадратних дужках. Нумерація починається з нуля.

### 2. Функції з класу System.Array для роботи з масивами

Функція	Результат
масив.Length	Кількість елементів в масиві
масив.GetLength	Кількість елементів у заданому вимірі масиву
Array.BinarySearch	Виконує пошук елемента у відсортованому масиві
Array.Clear	Обнуляє задані елементи масиву
Array.Copy	Копіює вказаний діапазон елементів в інший масив
масив.CopyTo	Копіює всі елементи масиву в інший
Array.IndexOf	Індекс першого елемента, рівного заданому
Array.LastIndexOf	Індекс останнього елемента, рівного заданому
Array.Reverse	Змінює порядок елементів на обернений
Array.Sort	Сортування елементів масиву

### 3. Оператор перебору значень foreach

Цей оператор повторює тіло циклу для кожного елемента масиву. Його використовують лише з метою отримання інформації, не слід таким чином змінювати елементи масиву (для цього підійде оператор циклу for).

```
foreach (тип змінна in масив)
{
    // тіло циклу
    // тепер змінна дає доступ до елемента масиву
}
```

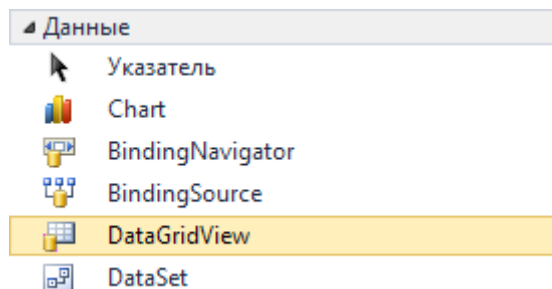
### Приклади виконання завдань

#### Приклад 1

Для заданого символьного масиву A(7) поміняти місцями максимальний та останній елементи.

#### Розв'язання

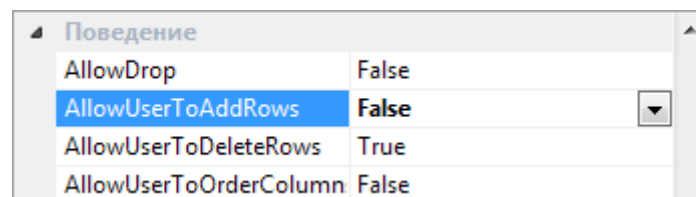
Розташувати на формі DataGridView та кнопку.



З контекстного меню компонента виберіть команду «Правка столбцов».

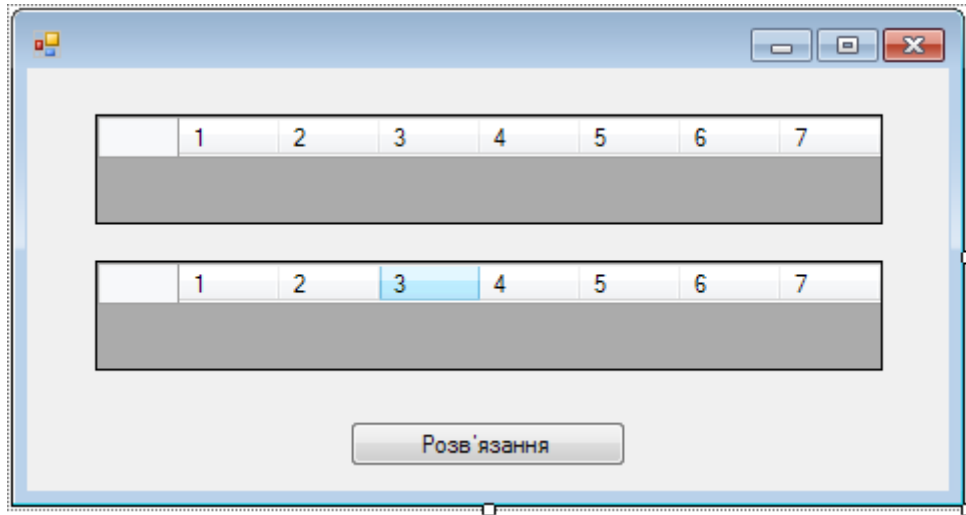
У вікні, що відкриється, натисніть на кнопку «Добавить». У полі «Текст заголовка» введіть «1». Натисніть кнопку «Добавить». Зробіть так само ще шість разів, кожен раз вводячи наступну цифру.

Перейдіть на вкладку властивостей DataGridView та поставте властивості AllowUserToAddRows значення False:



Тепер користувач під час виконання програми не зможе додавати нові рядки до таблиці.

Після цього зробіть копію DataGridView, і розмістіть її нижче оригіналу.



Перейдіть до файлу «Form1.cs» та напишіть код у `public Form1()`:

```
public Form1()
{
    InitializeComponent();
    dataGridView1.Rows.Add(); // додаємо один рядок до першої
    dataGridView2.Rows.Add(); // та другої таблиці
}
```

Для оброблювача натискання на кнопку такий код:

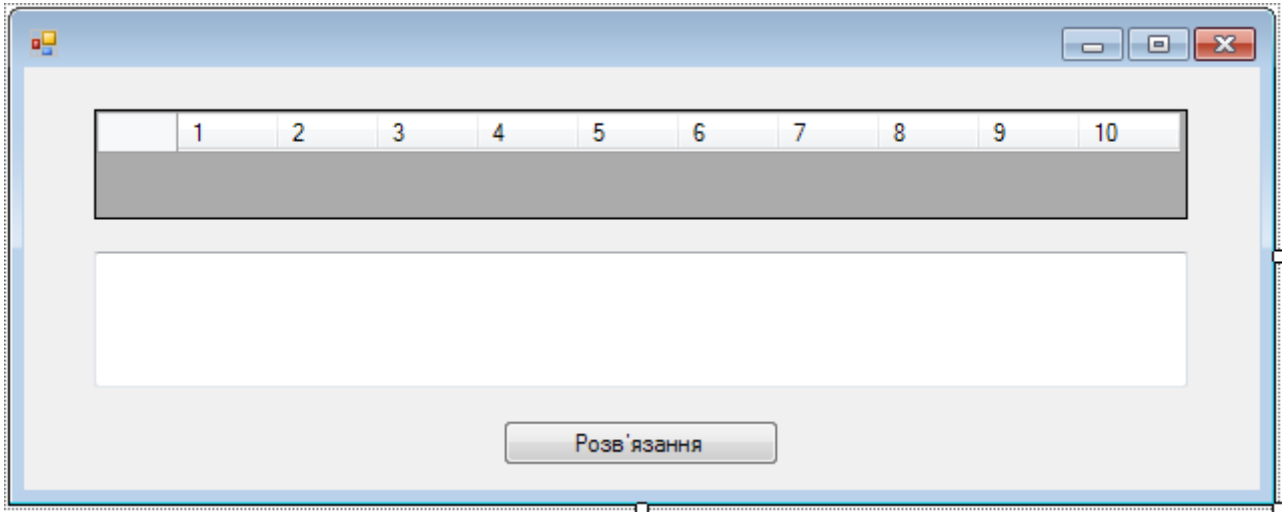
```
private void button1_Click(object sender, EventArgs e)
{
    const byte n = 7;
    char[] a = new char[n];
    for (byte i = 0; i < n; i++)
        a[i] = Convert.ToChar(dataGridView1[i, 0].Value);
    char max = a[0];
    byte nmax = 0;
    for (byte i = 1; i < n; i++)
        if (a[i] > max)
        {
            max = a[i];
            nmax = i;
        }
    a[nmax] = a[n-1];
    a[n-1] = max;
    for (byte i = 0; i < n; i++)
        dataGridView2[i, 0].Value = a[i];
}
```

## Приклад 2

Знайти суму і кількість парних елементів цілого масиву A(10).

### Розв'язання

Створимо форму:



Напишемо код програми:

```
public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();
        dataGridView1.Rows.Add();
    }

    private void button1_Click(object sender, EventArgs e)
    {
        const byte n = 10;
        int[] a = new int[n];
        for (byte i = 0; i < n; i++)
            a[i] = Convert.ToInt32(dataGridView1[i, 0].Value);
        int s = 0, count = 0;
        foreach (byte i in a) // цикл для всіх елементів масиву a
            if (i % 2 == 0) // і послідовно набуває значень
                { // всіх елементів масиву
                    s += i;
                    count++;
                }
        textBox1.Text = "Результат:\n";
        textBox1.AppendText(
            "Кількість парних елементів:\t " + count.ToString() + '\n');
        textBox1.AppendText("Їх сума:\t\t\t " + s.ToString());
    }
}
```

'\n' – це символ табуляції