

- 25) Дан массив, содержащий 14 элементов. Все отрицательные элементы заменить на 3. Вывести исходный и полученный массив.

ЛАБОРАТОРНАЯ РАБОТА 7. ПРОГРАММИРОВАНИЕ С ИСПОЛЬЗОВАНИЕМ МНОГОМЕРНЫХ МАССИВОВ

Цель лабораторной работы: изучить свойства компонента **dataGridView**. Написать программу с использованием двумерных массивов.

7.1. Двухмерные массивы

Многомерные массивы имеют более одного измерения. Чаще всего используются двумерные массивы, которые представляют собой таблицы. Каждый элемент массива имеет два индекса, первый определяет номер строки, второй - номер столбца, на пересечении которых находится элемент. Нумерация строк и столбцов начинается с нуля. Объявить двумерный массив можно одним из предложенных способов:

```
тип [,] имя__массива;  
тип [,] имя__массива = new тип [размер1, размер2];  
тип [,] имя__массива={{элементы 1-ой строки}, ... , {элементы n-ой строки}};  
тип [,] имя__массива= new тип [,]{{элементы 1-ой строки}, ... , {элементы n-ой строки}};
```

Пример кода использующего многомерные массивы:

```
// объявление и инициализация двумерного массива  
int[,] array2D = new int[,] { { 1, 2 }, { 3, 4 }, { 5, 6 }, { 7, 8 } };  
// Объявление такого массива с указанием размерности (кол-во строки столбцов)  
int[,] array2Da = new int[4, 2] { { 1, 2 }, { 3, 4 }, { 5, 6 }, { 7, 8 } };  
// Объявление двумерного массива элементами, которого являются строки  
string[,] array2Db = new string[3, 2] { { "one", "two" }, { "three", "four" },  
                                         { "five", "six" } };  
  
// Объявление трехмерного массива  
int[, ,] array3D = new int[, ,] { { { 1, 2, 3 }, { 4, 5, 6 } },  
                                   { { 7, 8, 9 }, { 10, 11, 12 } } };  
// Объявление трехмерного массива с указанием размерности  
int[, ,] array3Da = new int[2, 2, 3] { { { 1, 2, 3 }, { 4, 5, 6 } },  
                                         { { 7, 8, 9 }, { 10, 11, 12 } } };  
  
// Доступ к элементам массива  
System.Console.WriteLine(array2D[0, 0]);  
System.Console.WriteLine(array2D[0, 1]);  
System.Console.WriteLine(array2D[1, 0]);  
System.Console.WriteLine(array2D[1, 1]);  
System.Console.WriteLine(array2D[3, 0]);  
System.Console.WriteLine(array2Db[1, 0]);  
System.Console.WriteLine(array3Da[1, 0, 1]);  
System.Console.WriteLine(array3D[1, 1, 2]);  
  
// Результаты работы программы (выводятся в консоль):  
// 1  
// 2
```

```
// 3
// 4
// 7
// three
// 8
// 12
```

7.2. Элемент управления DataGridView

При работе с двумерными массивами ввод и вывод информации на экран удобно организовывать в виде таблиц. Элемент управления **DataGridView** может быть использован для отображения информации в виде двумерной таблицы. Для обращения к ячейке в этом элементе необходимо указать номер строки и номер столбца. Например: `dataGridView1.Rows[2].Cells[7].Value = "*"`; данный код позволяет записать во вторую строку в 7 ячейку знак звездочка.

7.3. Порядок выполнения задания

Задание: Создать программу для определения целочисленной матрицы 15 на 15. Разработать обработчик для поиска минимального элемента на дополнительной диагонали матрицы. Результат, после нажатия кнопки типа **Button**, вывести в **textBox**.

Окно программы приведено на рис. 7.1.

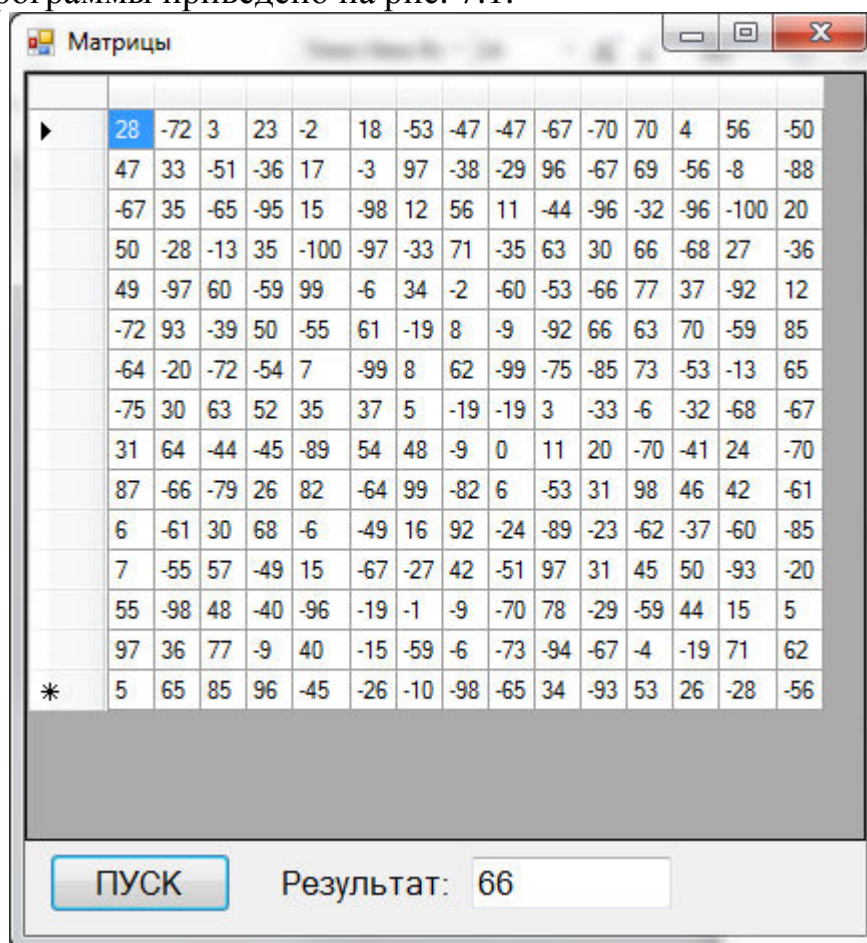


Рис. 7.1. Окно программы для работы с двумерным массивом

Текст обработчика события нажатия на кнопку приведен ниже.

```
private void button1_Click(object sender, EventArgs e)
{
    dataGridView1.RowCount = 15; //Указываем количество строк
    dataGridView1.ColumnCount = 15; //Указываем количество столбцов
    int[,] a = new int[15,15]; //Инициализируем массив
    int i,j;
    //Заполняем матрицу случайными числами
    Random rand = new Random();
    for (i=0; i<15; i++)
        for (j=0; j<15; j++)
            a[i,j] = rand.Next(-100,100);
    //Выводим матрицу в dataGridView1
    for (i=0; i<15; i++)
        for (j=0; j<15; j++)
            dataGridView1.Rows[i].Cells[j].Value =
Convert.ToString(a[i,j]);
    //производим поиск максимального элемента на дополнительной диагонали
    int m = int.MinValue;
    for (i = 0; i < 15; i++)
        if (a[i, 14 - i] > m) m = a[i, 14 - i];
    // выводим результат
    textBox1.Text = Convert.ToString(m);
}
```

7.4. Индивидуальные задания

- 1) Дана матрица $A(3,4)$. Найти наименьший элемент в каждой строке матрицы. Вывести исходную матрицу и результаты вычислений.
- 2) Дана матрица $A(3,3)$. Вычислить сумму второй строки и произведение первого столбца. Вывести исходную матрицу и результаты вычислений.
- 3) Дана матрица $A(4,4)$. Найти наибольший элемент в главной диагонали. Вывести матрицу и наибольший элемент.
- 4) Дана матрица $A(3,4)$. Найти сумму элементов главной диагонали и эту сумму поставить на место последнего элемента. Вывести исходную и полученную матрицу.
- 5) Дана матрица $A(4,3)$. Вычислить наибольший элемент матрицы. Вывести исходную матрицу и наибольший элемент.
- 6) Дана матрица $A(4,3)$. Найти количество положительных элементов.
- 7) Дана матрица $A(3,4)$. Найти количество отрицательных элементов.
- 8) Даны матрицы $X(15,15)$ и $Y(15,15)$. Вычислить и вывести элементы новой матрицы $z_{ij}=12x_{ij}-0.85y_{ij}^2$.
- 9) Даны матрицы $A(6,6)$, $B(6,6)$ и $C(6,6)$. Получить матрицу $D(6,6)$, элементы которой вычисляются по формуле $d_{ij}=\max\{a_{ij},(b_{ij}+c_{ij})\}$. Матрицу $D(6,6)$ вывести.
- 10) Вычислить сумму S элементов главной диагонали матрицы $B(10,10)$. Если $S>10$, то исходную матрицу преобразовать по формуле $b_{ij}=b_{ij}+13.5$; если $S\leq 10$, то $b_{ij}=b_{ij}^2-1.5$. Вывести сумму S и преобразованную матрицу.

- 11) Дана матрица $F(15,15)$. Вывести номер и среднее арифметическое элементов строки, начинающейся с 1. Если такой строки нет, то вывести сообщение “строки нет”.
- 12) Дана матрица $F(7,7)$. Найти наименьший элемент в каждом столбце. Вывести матрицу и найденные элементы.
- 13) Найти наибольший элемент главной диагонали матрицы $A(15,15)$ и вывести всю строку, в которой он находится.
- 14) Найти наибольшие элементы каждой строки матрицы $Z(16,16)$ и поместить их на главную диагональ. Вывести полученную матрицу.
- 15) Вычислить суммы элементов матрицы $Y(12,12)$ по столбцам и вывести их.
- 16) Найти наибольший элемент матрицы $A(10,10)$ и записать нули в ту строку и столбец, где он находится. Вывести наибольший элемент, исходную и полученную матрицу.
- 17) Дана матрица $R(9,9)$. Найти наименьший элемент в каждой строке и записать его на место первого элемента строки. Вывести исходную и полученную матрицы.
- 18) Определить количество положительных элементов каждой строки матрицы $A(10,20)$ и запомнить их в одномерном массиве N . Массив N вывести.
- 19) Вычислить количество N положительных элементов последнего столбца матрицы $X(5,5)$. Если $N < 3$, то вывести все положительные элементы матрицы, если $N \geq 3$, то вывести сумму элементов главной диагонали матрицы.
- 20) Вычислить и вывести сумму элементов матрицы $A(12,12)$, расположенных над главной диагональю матрицы.

ЛАБОРАТОРНАЯ РАБОТА 8. ПРОГРАММИРОВАНИЕ С ИСПОЛЬЗОВАНИЕМ СРЕДСТВ ДЛЯ ОТОБРАЖЕНИЯ ГРАФИЧЕСКОЙ ИНФОРМАЦИИ

Цель лабораторной работы: изучить возможности построения графиков с помощью компонента отображения графической информации **Chart**. Написать и отладить программу построения на экране графика заданной функции.

8.1. Как строится график с помощью компонента Chart

Обычно результаты расчетов представляются в виде графиков и диаграмм. Библиотека .NET Framework имеет мощный элемент управления Chart для отображения на экране графической информации (рис. 8.1).