

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ПОВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ
УНИВЕРСИТЕТ»

Кафедра ПиПЭВС

ОТЧЕТ

по дисциплине: " Алгоритмизация и
программирование "

Выполнила: студентка группы УИТС-21
Имомалиев Н.Н.

Проверил: Курасов П. А.
Оценка _____

Йошкар-Ола
2022 г.

ЛАБОРАТОРНАЯ РАБОТА 3. КЛАССЫ. СТРУКТУРЫ

1. Цель и содержание

Цель лабораторной работы: изучить структуру и принципы объявления классов, освоить технологию создания экземпляров классов (объектов).

Задачи лабораторной работы:

- научиться объявлять классы;
- научиться создавать объекты классов;
- научиться работать с полями данных и методами классов.

Класс «Матрица MxN». Реализовать инициализацию элементов матрицы случайными числами, вывод транспонированной матрицы, нахождение среднего арифметического всех элементов, а также вывод информации об объекте.
--

Рисунок 1 - Выражение для вычисления

```
using System;
```

```
namespace pr3
```

```
{
```

```
class Program
```

```
{
```

```
static void Main(string[] args)
```

```
{
```

```
// сначала надо позаботиться о переменных и массиве
```

```
Console.Write("Введите количество строк = ");
```

```
int m = int.Parse(Console.ReadLine()); ;
```

```
Console.Write("Введите количество столбцов = ");
```

```
int n = int.Parse(Console.ReadLine()); ;
```

```
int[,] mas = new int[m, n]; //массив размерностью m на n
```

```

Random rnd = new Random(); // объект, который будем использовать
для генерации рандомных целых чисел

Console.WriteLine("Матрица случайных чисел:");
for(int i=0;i<m;i++) //строки
{
    for (int j = 0; j < n; j++) //столбцы
    {
        mas[i, j] = rnd.Next(-50, 50); // диапазон случайных чисел от -50 до
50        Console.Write(mas[i, j] + "\t"); //через пробел будут выводиться
    }
    Console.WriteLine(); //переход на новую строку
}

int sum = 0; // сумма нечётных элементов
int score = 0; //счетчик количества элементов
foreach (int i in mas)
{
    if (i % 2 != 0)
    {
        sum += i;
        score++;
    }
}

// Выведем инфу о среднем значении нечётных элементов
if (score != 0)
    Console.WriteLine("Среднее значение нечётных элементов = {0}",
sum/score);
else
    Console.WriteLine("Нечётных элементов нет");

//найти максимальный элемент в каждом столбце.

```

```

int[] max_elem = new int[n]; //создадим одномерный массив, который
равен количеству столбцов
for (int i = 0; i < n; i++)
{
    max_elem[i] = mas[0, i]; //по умолчанию у нас будет так, что первая
строка будет максимальными элементами
}
//будем перебирать каждый столбец, в поисках максимального
элемента
for (int i = 0; i < m; i++)
{
    for (int j = 0; j < n; j++)
    {
        if (mas[i, j] >= max_elem[j])
            max_elem[j] = mas[i, j];
    }
}
if (max_elem.Length != 0)
{
    Console.WriteLine("Список максимальных элементов в соответствии
со столбцом: ");
    foreach (int i in max_elem)
    {
        Console.Write(i + "\t");
    }
}
else
{
    Console.WriteLine("Вы ввели матрицу размерностью 0 на 0 или
некорректное значение");
}

```

```
    }  
    Console.ReadKey();  
  }  
}
```

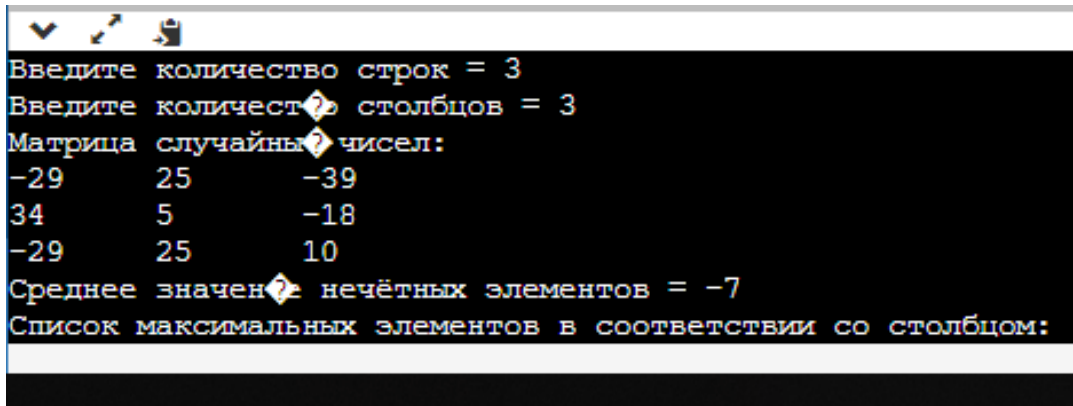


Рисунок 2 - Результат выполнения программы