**5 МАССИВЫ. АЛГОРИТМЫ ОБРАБОТКИ МАССИВОВ. РЕКУРСИЯ**

Задание 1. В массиве из 15 целых чисел найти наибольший элемент и поменять его местами с первым элементом.

Листинг программы:

static void Main(string[] args)

{

Random rand = new Random();

int[] Array = new int[15];

int Max = Array[0];

int j = 0;

Console.Write("initial array: ");

for (int x = 0; x < Array.Length; x++)

{

Array[x] = rand.Next(100);

Console.Write($"{Array[x]}; ");

}

for (int i = 0; i < Array.Length; i++)

{

if (Max < Array[i])

{

Max = Array[i];

j = i;

}

}

Array[j] = Array[0];

Array[0] = Max;

Console.WriteLine();

Console.WriteLine("the resulting array: " + string.Join("; ", Array));

Console.ReadLine();

}

}

Таблица 5.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Введите число элементов массива a 10  Введите 1 элемент массива: 4  Введите 2 элемент массива: 6  Введите 3 элемент массива: 7  Введите 4 элемент массива: 83  Введите 5 элемент массива: 2  Введите 6 элемент массива: 100  Введите 7 элемент массива: 76  Введите 8 элемент массива: 4  Введите 9 элемент массива: 1  Введите 10 элемент массива: 14 | Нечетный элемент:7  Нечетный элемент:7  Нечетный элемент:83  Нечетный элемент:1 |

Анализ результатов:

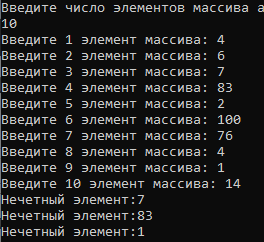


Рисунок 5.1 – Результат работы программы

Источник: собственная разработка

Задание 2. Во всех задачах сформировать и вывести на экран одномерный массив согласно варианту. Произвести его сортировку и бинарный поиск числа k (k - вводится с клавиатуры). Напишите программу для решения задачи:

Пусть дано 100 целых чисел. Распечатайте их в обратном порядке по 6 чисел в строке.

Листинг программы:

public static int[] Massiv(int a, int b)

{

int x = 100;

int[] mas = new int[x];

Random rnd = new Random();

for (int i = 0; i < mas.Length; i++)

{

mas[i] = rnd.Next(a, b);

}

return mas;

}

public static void PrintMas(int[] mas)

{

foreach (var item in mas)

{

Console.Write(" " + item);

}

}

public static void PrintArrayReverse(int[] mas,int countInLine=6)

{

var reversMas = mas.Reverse().ToArray();

for (int i = 0; i < reversMas.Length; i++)

{

if (i%countInLine==0)

{

Console.WriteLine();

}

Console.Write($"{reversMas[i]} ");

}

}

public static void Sort(int[] mas)

{

Array.Sort(mas);

Array.Reverse(mas);

foreach (var item in mas)

{

Console.Write(" " + item);

}

}

public static int Binary(int[] mas, int k)

{

Array.Sort(mas);

var sortmas = Array.BinarySearch(mas, k);

return sortmas;

}

privte static void PrintArray(int[] array)

{

Console.WriteLine(string.Join(", ",array));

}

static void Main(string[] args)

{

Console.Write("Enter the initial boundary of the array: ");

var a = Convert.ToInt32(Console.ReadLine());

Console.Write("Enter the final boundary of the array: ");

var b = Convert.ToInt32(Console.ReadLine());

Console.Write("Enter K: ");

var k = Convert.ToInt32(Console.ReadLine());

var mas = Massiv(a, b);

PrintArray(mas);

Console.WriteLine();

Console.Write("Array on the contrary: ");

PrintArrayReverse(mas,6);

Console.WriteLine();

var itemIndex = Binary(mas, k);

Console.WriteLine(itemIndex);

Console.ReadLine();

}

}

}

Таблица 5.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| initial boundary of the array: 1  final boundary of the array: 15  Enter K: 5 | Reverse array |

Анализ результатов:

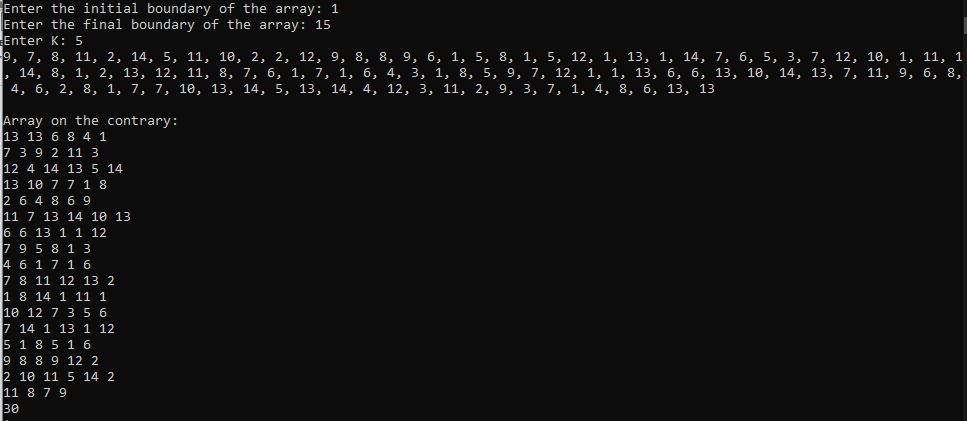


Рисунок 5.2 – Результат работы программы

Источник: собственная разработка

Задание 3. В каждой задаче задается квадратная целочисленная матрица NxN (значение N вводится с клавиатуры, N&lt;10). Программа должна заполнять матрицу случайными числами из диапазона [a, b] (a, b вводятся с клавиатуры) и осуществлять вывод на экран исходной матрицы. Затем необходимо произвести необходимые действия и напечатать результаты.

Листинг программы:

class program

{

public static void Main()

{

int i, j;

Console.WriteLine("Введите размер массива:");

i = Convert.ToInt32(Console.ReadLine());

j = i;

int summa = 0, k = 0, num = 0;

int[,] a = new int[i, j];

int[] b = new int[j];

Random rand = new Random();

for (i = 0; i < 5; i++) //заполнение двумерного массива случайными числами

for (j = 0; j < 5; j++)

{

a[i, j] = rand.Next(-100, 100);

}

//вычисление сумму столбцов

for (i = 0; i < 5; i++)

for (j = 0; j < 5; j++)

{

summa += a[j, i];

if (a[j, i] > 0) num++; //подсчитывание количества положительных элементов столбцов

if (j == 4)

{

b[k] = summa; //внесение в новый массив суммы столбцов

Console.WriteLine("Количество положительных элементов в столбце= " + k + num);

summa = 0; //обнуление суммы столбцов

num = 0; //обнуление количества положительных элементов в столбце

}

}

}

}

Таблица 5.3 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 5 | Количество положительных элементов в столбце= 02  Количество положительных элементов в столбце= 02  Количество положительных элементов в столбце= 03  Количество положительных элементов в столбце= 03  Количество положительных элементов в столбце= 03 |

Анализ результатов:

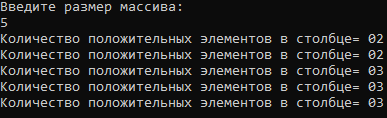


Рисунок 5.3 – Результат работы программы

Источник: собственная разработка

Задание 4. Решить задачу на проверку условий после выполнения расчетов. Дан двумерный массив. Выяснить, является ли произведение элементов второго столбца массива трехзначным числом.

Листинг программы:

static int N = 4;

static int M = 10;

static void Main()

{

int[,] matrix = new int[N, N];

Random r = new Random();

int mul = 1;

for (int i = 0; i < N; i++)

{

for (int j = 0; j < N; j++)

{

matrix[i, j] = r.Next(1, M);

if (j == 1) mul \*= matrix[i, j];

Console.Write(matrix[i, j] + "\t");

}

Console.WriteLine();

}

if (mul / 100 >= 1 && mul / 100 < 10)

Console.WriteLine($"Answer:{mul} Yes, is three-digit!");

else Console.WriteLine($" Answer: {mul} - No, is't three-digit!!");

Console.ReadKey();

}

}

}

Таблица 5.4 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 3  9  8  6 | Answer: 1296 - No, is't three-digit!! |

Анализ результатов:

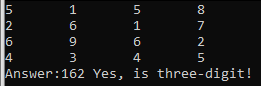


Рисунок 5.4 – Результат работы программы

Источник: собственная разработка

Задание 5. Разработать рекурсивный алгоритм и программу решения задачи, в которой вычислить: f(n)=(n-1)!/n!. Исходные данные вводятся с клавиатуры.

Листинг программы:

class Program

{

// Рекурсивный метод

static int Factorial(int n)

{

int result;

if (n == 1)

return 1;

result = Factorial(n - 1) / n;

return result;

}

static void Main(string[] args)

{

label1:

Console.WriteLine("Введите число: ");

try

{

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

Console.WriteLine("{0}! = {1}", n, Factorial(n));

}

catch (FormatException)

{

Console.WriteLine("Некорректное число");

goto label1;

}

Console.ReadLine();

}

}

Таблица 5.5 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Введите число: 6 | Результат: 0 |

Анализ результатов:

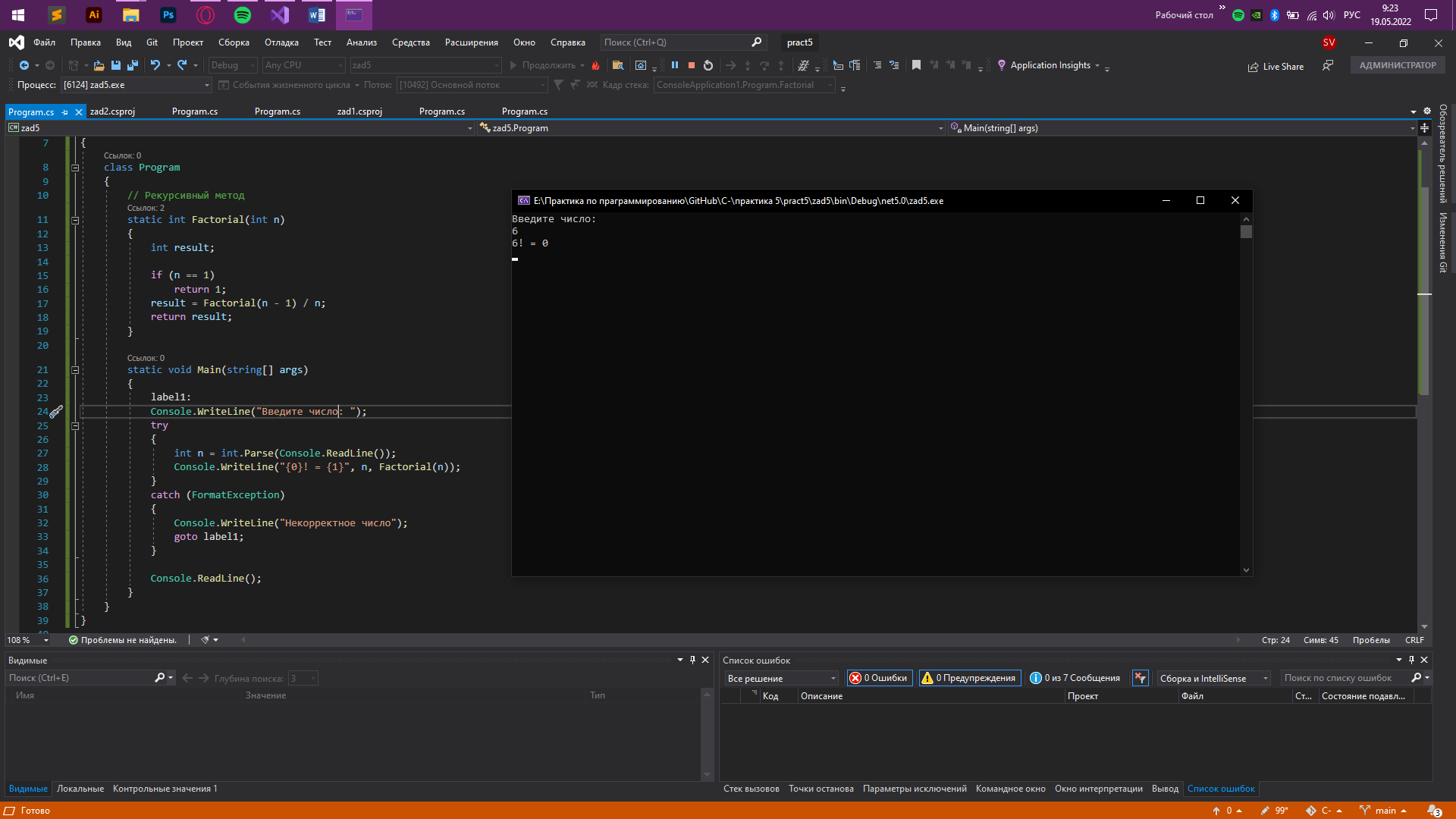


Рисунок 5.5 – Результат работы программы

Источник: собственная разработка