**5. Массивы. Классы**

Задание №1. Найти порядковый номер максимального элемента массива

вещественных чисел.

Листинг программы:

Console.Write("Enter the number of elements in the array: ");

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

double[] array = new double[num];

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

{

Console.Write($"Enter array elements at index {i}: ") ;

array[i] = double.Parse(Console.ReadLine());

}

double max = 0;

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

{

if (array[i] > max)

max = array[i];

max = i;

}

Console.Write($"Max elements : {IndexOf(array,max)} ");

static double IndexOf(double [] arr, double value)

{

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

{

if (arr[i] == value)

{

return i;

}

}

return -1;

}

Таблица 5.1 – Входные и выходные данных задание №1

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Число в массиве: 4  Нулевой элемент: 12  Первый элемент: 2334  Второй элемент: 1  Третий элемент:3 | Максимальный элемент под индексом - 1 |

Анализ результатов:

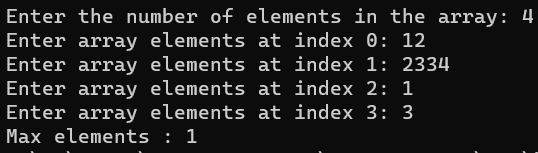


Рисунок 5.1 – Результат работы программы из задания №1

Задание №2. Пусть даны целые числа а 1 , ..., а n , каждое из которых отлично от нуля. Если в последовательности отрицательные и положительные члены чередуются, то ответом должна служить сама исходная последовательность. Иначе - получите все отрицательные члены последовательности, сохранив порядок их следования.

Листинг программы:

Console.Write("Enter how many columns and rows will be in the matrix: ");

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

int[] array = new int[n];

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

{

Console.Write($"Enter element at index: {i} - ");

array[i] = int.Parse(Console.ReadLine());

}

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

{

Console.Write( array[i] + " ");

}

Console.WriteLine();

for (int i = 0; i < array.Length-1; i++)

{

if (array[i] > 0 && array[i + 1] < 0 || array[i] < 0 && array[i + 1] > 0)

{

Console.Write(array[i] + " ");

if (i==array.Length-2)

{

Console.Write(array[i+1]);

}

}

else

{

array[i] = -array[i];

Console.Write( array[i] + " ");

if (i == array.Length - 2)

{

Console.Write(-array[i + 1]);

}

}

}

Таблица 5.2 – Входные и выходные данных задание №2

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Размер матрицы: 5  0 – 1  1 – 2  2 - 3  3 – 4  4 - 5 | -1 -2 -3 -4 -5 |

Анализ результатов:

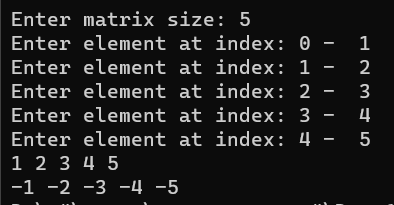


Рисунок 5.2 – Результат работы программы из задания №2

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. Подсчитать количество чисел, меньших заданного H. Вычислить количество нечётных элементов k-того столбца

(значение k вводится с клавиатуры).

Листинг программы:

Console.Write("Enter how many columns and rows will be in the matrix: ");

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

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

if (n <=10)

{

Console.WriteLine("Enter the matrix fill range");

Console.WriteLine("Enter a: ");

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

Console.WriteLine("Enter b: ");

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

Random random = new Random();

for (int i = 0; i < matrix.GetLength(0); i++)

{

for (int j = 0; j < matrix.GetLength(1); j++)

{

matrix[i, j] = random.Next(a, b);

}

}

for (int x= 0; x < matrix.GetLength(0); x++)

{

for (int y = 0; y < matrix.GetLength(1); y++)

{

Console.Write(" " + matrix[x,y]);

}

Console.WriteLine();

}

Console.WriteLine("Enter h: ");

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

int count = 0;

for (int i = 0; i < matrix.GetLength(0); i++)

{

for (int j = 0; j < matrix.GetLength(1); j++)

{

if (matrix[i,j] < h)

{

count++;

}

}

}

Console.WriteLine($"Number of numbers less than {h} = {count}");

Console.Write("Enter column number:");

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

int counte1 = 0;

if (colum >= 0 && colum<=n)

{

for (int i = 0; i < matrix.GetLength(0); i++)

{

if (matrix[i, colum] % 2 != 0)

{

counte1++;

}

}

if (counte1==0)

{

Console.WriteLine("No odd ones");

}

Console.WriteLine($"Number of odd column elements {colum} = {counte1}");

}

else

{

Console.WriteLine("No such column");

}

}

else

{

Console.WriteLine("You entered a number greater than 10, restart the program");

Таблица 5.3 – Входные и выходные данных задание №3

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Количество строк и столбцов – 4  A = 1  B = 7  H = 4  Номер столбца = 2 | Чисел меньше 4 = 6  Нечетных элементов в столбце 2 = 0 |

Анализ результатов:

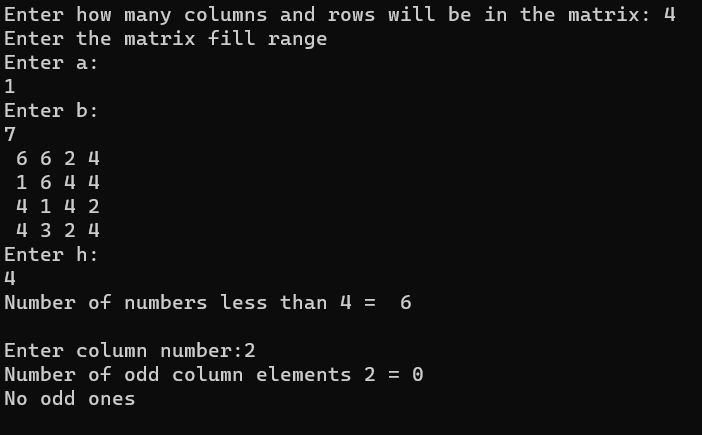


Рисунок 5.3 – Результат работы программы из задания №3

Задание №4. Дан двумерный массив. Составить программу, которая определяет максимальное из двух чисел: суммы элементов k-й строки и суммы элементов s-го столбца массива.

Листинг программы:

Console.Write("Enter how many columns and rows will be in the matrix: ");

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

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

Random random = new Random();

for (int i = 0; i < matrix.GetLength(0); i++)

{

for (int j = 0; j < matrix.GetLength(1); j++)

{

matrix[i, j] = random.Next(10);

}

}

for (int x = 0; x < matrix.GetLength(0); x++)

{

for (int y = 0; y < matrix.GetLength(1); y++)

{

Console.Write(" " + matrix[x, y]);

}

Console.WriteLine();

}

Console.WriteLine("Enter the line number, the amount you want to know");

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

Console.WriteLine("Enter the column number, the amount you want to know");

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

if (row >= 0 && row <= n || col >= 0 && col < n)

{

int sumRow = matrix.CalcRow(row);

int sumCol = matrix.CalcCol(col);

int max = Math.Max(sumRow, sumCol);

Console.WriteLine($"Sum of numbers in a row {row} = {sumRow}");

Console.WriteLine($"Sum of numbers in a column {col} = {sumCol}");

if (max == sumRow)

{

Console.WriteLine($"The maximum of two numbers is the sum of the numbers in the string {row} ");

}

else

{

Console.WriteLine($"The maximum of two numbers is the sum of the numbers in the column {col} ");

}

}

else

{

Console.WriteLine("No such column and row ");

}

internal static class DoubleArrayExtention

{

public static int CalcRow(this int [,] matrix , int row)

{

int sum = 0;

for (int j = 0; j < matrix.GetLength(1); j++)

{

sum = sum+ matrix[row, j];

}

return sum;

}

public static int CalcCol(this int[,] matrix, int col)

{

int sum = 0;

for (int i = 0; i < matrix.GetLength(1); i++)

{

sum = sum+ matrix[i,col];

}

return sum;

}

}

Таблица 5.4 – Входные и выходные данных задание №4

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Размер матрицы: 3  Сумму 1 строки и сумма первого столбца | Сумма 1 строки = 10  Сумма 1 столбца = 4  Максимальное – сумма 1 строки |

Анализ результатов:

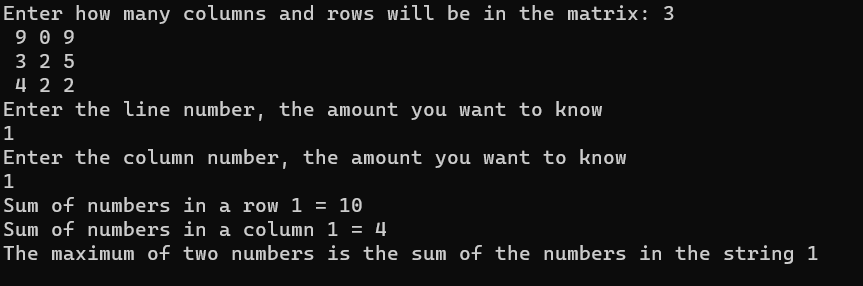


Рисунок 5.4 - Результат работы программы из задания №4

Задание №5. Разработать рекурсивный алгоритм и программу решения задачи, в которой вычислить: f(n)=(n!+2)/n!. Исходные данные вводятся с клавиатуры.

Листинг программы:

Console.Write("Enter n: ");

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

Console.WriteLine(Factorial(Factorial(n) + 2 ) / Factorial(n));

int Factorial(int num)

{

if (num == 0)

{

return 1;

}

else

{

return num \* Factorial(num - 1);

}

}

Таблица 5.5 – Входные и выходные данных задание №1

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| N = 2 | Ответ = 12 |

Анализ результатов:

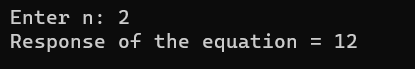


Рисунок 5.5 – Результат работы программы из задания №5