1. **«ПРОЦЕДУРЫ И ФУНКЦИИ. ПЕРЕГРУЗКА ФУНКЦИЙ (МЕТОДОВ) И ОПЕРАТОРОВ»**

Задание 1. Найти количество отрицательных элементов массива А.

Листинг программы:

using System;

class Program

{

static void Main(string[] args)

{

Console.Write("Введите элементы массива через пробел: ");

string[] input = Console.ReadLine().Split();

int[] A = new int[input.Length];

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

{

A[i] = int.Parse(input[i]);

}

int count = 0;

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

{

if (A[i] < 0)

{

count++;

Console.WriteLine($"Отрицательный элемент: {A[i]}");

}

}

Console.WriteLine($"Количество отрицательных элементов: {count}");

}

}

Таблица 1.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 4 6 -4 -6 -10 34 | -4 -6 -10 |

Анализ результатов:

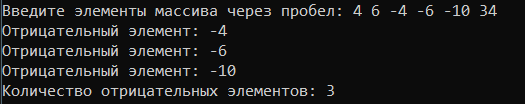


Рисунок 1.1 – Результат работы программы

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 даны целые числа а 1 , ..., а 99 . Получите новую последовательность, выбросив из исходной максимальный и минимальный члены.

Листинг программы:

using System;

class Program

{

static void Main(string[] args)

{

int[] a = new int[99];

for (int i = 0; i < 99; i++)

{

a[i] = i + 1;

}

Console.WriteLine("Исходный массив:");

foreach (int item in a)

{

Console.Write($"{item} ");

}

Console.WriteLine();

int min = a[0];

int max = a[0];

int minIndex = 0;

int maxIndex = 0;

for (int i = 1; i < a.Length; i++)

{

if (a[i] < min)

{

min = a[i];

minIndex = i;

}

if (a[i] > max)

{

max = a[i];

maxIndex = i;

}

}

int[] b = new int[a.Length - 2];

int j = 0;

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

{

if (i != minIndex && i != maxIndex)

{

b[j] = a[i];

j++;

}

}

Console.WriteLine("Новый массив:");

foreach (int item in b)

{

Console.Write($"{item} ");

}

Console.WriteLine();

Array.Sort(b);

Console.WriteLine("Отсортированный массив:");

foreach (int item in b)

{

Console.Write($"{item} ");

}

Console.WriteLine();

Console.Write("Введите число k для поиска: ");

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

int left = 0;

int right = b.Length - 1;

int mid;

while (left <= right)

{

mid = (left + right) / 2;

if (b[mid] == k)

{

Console.WriteLine($"Число {k} найдено в массиве на позиции {mid}");

return;

}

if (b[mid] > k)

{

right = mid - 1;

}

else

{

left = mid + 1;

}

}

Console.WriteLine($"Число {k} не найдено в массиве");

}

}

Таблица 1.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 56 | 54 |

Анализ результатов:

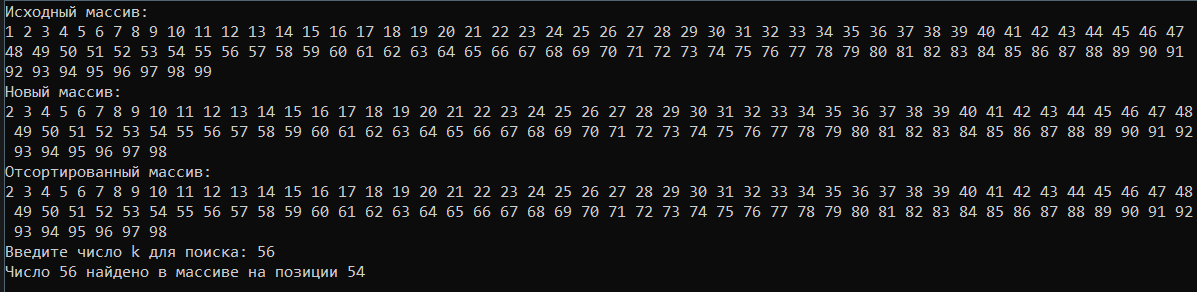


Рисунок 1.2 – Результат работы программы

Источник: собственная разработка

Задание 3. Вычислить среднее арифметическое элементов, больших заданного G. Вычислить количество чётных элементов k-той строки (значение k вводится с клавиатуры).

Листинг программы:

using System;

class Program

{

static void Main(string[] args)

{

Console.Write("Введите значение N: ");

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

Console.Write("Введите значение a: ");

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

Console.Write("Введите значение b: ");

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

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

Random random = new Random();

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

{

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

{

matrix[i, j] = random.Next(a, b + 1);

}

}

Console.WriteLine("Исходная матрица:");

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

{

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

{

Console.Write("{0,4}", matrix[i, j]);

}

Console.WriteLine();

}

Console.Write("Введите значение G: ");

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

int sum = 0;

int count = 0;

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

{

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

{

if (matrix[i, j] > g)

{

sum += matrix[i, j];

count++;

}

}

}

if (count > 0)

{

double average = (double)sum / count;

Console.WriteLine("Среднее арифметическое элементов, больших {0}: {1:F2}", g, average);

}

else

{

Console.WriteLine("Нет элементов, больших {0}", g);

}

Console.Write("Введите номер строки k (от 0 до {0}): ", n - 1);

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

int evenCount = 0;

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

{

if (matrix[k, j] % 2 == 0)

{

evenCount++;

}

}

Console.WriteLine("Количество чётных элементов в строке {0}: {1}", k, evenCount);

Console.ReadLine();

}

}

Таблица 1.3 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 4 3 6  2 3 | 5,07 2 3 |

Анализ результатов:

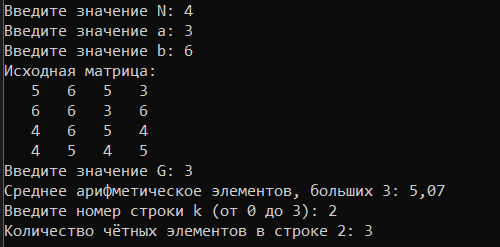


Рисунок 1.3 – Результат работы программы

Источник: собственная разработка

Задание 4. Дан двумерный массив. Определить в какой строке массива сумма элементов больше: в первой или в предпоследней.

Листинг программы:

using System;

class Program

{

static void Main(string[] args)

{

Console.Write("Введите размерность массива: ");

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

Console.Write("Введите минимальное значение: ");

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

Console.Write("Введите максимальное значение: ");

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

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

Random rand = new Random();

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

{

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

{

arr[i, j] = rand.Next(min, max + 1);

}

}

Console.WriteLine("Исходный массив:");

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

{

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

{

Console.Write("{0,3}", arr[i, j]);

}

Console.WriteLine();

}

int sumFirstRow = 0;

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

{

sumFirstRow += arr[0, j];

}

int sumLastButOneRow = 0;

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

{

sumLastButOneRow += arr[n - 2, j];

}

if (sumFirstRow > sumLastButOneRow)

{

Console.WriteLine("Сумма элементов в первой строке больше.");

}

else if (sumFirstRow < sumLastButOneRow)

{

Console.WriteLine("Сумма элементов в предпоследней строке больше.");

}

else

{

Console.WriteLine("Суммы элементов в первой и предпоследней строках равны.");

}

Console.ReadLine();

}

}

Таблица 1.4 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 6 2 15 | Сумма элементов в первой строке больше. |

Анализ результатов:

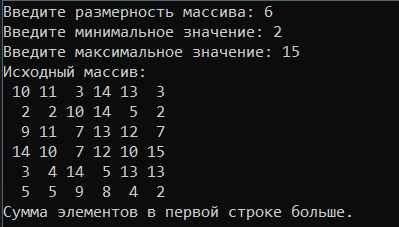


Рисунок 1.4 – Результат работы программы

Источник: собственная разработка

Задание 5. Разработать рекурсивный алгоритм и программу решения задачи, в которой вычислить: f(n)=(n+2)!/(n+4)!. Исходные данные вводятся с клавиатуры.

Листинг программы:

using System;

namespace RecursiveFunction

{

class Program

{

static void Main(string[] args)

{

Console.Write("Введите n: ");

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

double result = Factorial(n + 2) / Factorial(n + 4);

Console.WriteLine($"Результат: {result}");

}

static double Factorial(int n)

{

if (n == 0)

{

return 1;

}

else

{

return n \* Factorial(n - 1);

}

}

}

}

Таблица 1.5 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 4 | 0,017857142857142856 |

Анализ результатов:



Рисунок 1.5 – Результат работы программы

Источник: собственная разработка