Отчет по практической работе №6

Наименование работы: массивы. Класс Array.

Индивидуальные задания (вариант №9):

1. Составить программу, выдающую индексы заданного элемента или сообщающую, что такого элемента в массиве нет.
2. Составить программу с использованием одномерных массивов для решения задачи на переупорядочивание элементов массива. В качестве алгоритма сортировки использовать сортировку методом «пузырька».

В одномерном массиве, состоящем из n целых элементов, вычислить:

минимальный по модулю элемент массива;

сумму модулей элементов массива, расположенных после первого элемента, рав­ного нулю.

Преобразовать массив таким образом, чтобы в первой его половине располагались элементы, стоявшие в четных позициях, а во второй половине - элементы, стояв­шие в нечетных позициях.

1. Элемент матрицы называется локальным минимумом, если он строго меньше всех имеющихся у него соседей. Подсчитать количество локальных минимумов задан­ной матрицы размером 10 на 10. Найти сумму модулей элементов, расположенных выше главной диагонали.
2. Составить программу с использованием двумерных динамических массивов для решения задачи согласно варианту индивидуального задания №3.

Вопросы входного контроля:

1. Как объявляются одномерные массивы в языке C++/С#?

Ответ: тип\_д[] arr\_name = new тип\_д[размер];

1. Каким образом производится инициализация массива в языке C++/ С#?

Ответ: тип\_д[] arr\_name = new тип\_д[размер]{данные};

1. Какими должны быть размерности при описании статического массива в языке C++/ С#?

Ответ: разными.

Ход работы:

1. Разработал программу в соответствии с заданием №1. Были реализованы класс, метод, математические функции.

Листинг программы:

using System;

namespace Lab6

{

class Program

{

static void Main(string[] args)

{

Uan.Uans();

}

public class Uan

{

public static void Uans()

{

int[] arr = new int[100];

Random rnd = new Random();

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

{

arr[i] = rnd.Next(0, 1000);

}

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

{

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

}

Console.WriteLine();

Console.Write("Введите цифру от 0 до 1000: ");

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

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

{

if (arr[i] == n)

{

Console.WriteLine($"Элемент {arr[i]} его индекс: {i}");

}

else

Console.WriteLine($"Элемента {n} в списке нет!");

}

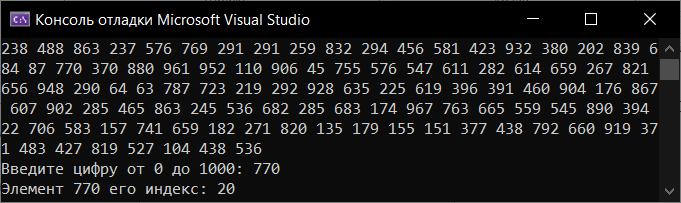
}

}

}

}

Результат выполнения программы Рис 1.1:



**Рисунок 1.1 – результат программы 6.1.**

1. Разработал программу в соответствии с заданием №2. Были реализованы класс, метод, математические функции.

Листинг программы:

using System;

using System.Linq;

namespace Lab6

{

class Program

{

static void Main(string[] args)

{

Tu.Tus();

}

public class Uan

{

public static void Uans()

{

int[] arr = new int[100];

Random rnd = new Random();

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

{

arr[i] = rnd.Next(0, 1000);

}

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

{

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

}

Console.WriteLine();

Console.Write("Введите цифру от 0 до 1000: ");

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

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

{

if (arr[i] == n)

{

Console.WriteLine($"Элемент {arr[i]} его индекс: {i}");

}

else

Console.WriteLine($"Элемента {n} в списке нет!");

}

}

}

public class Tu

{

public static void Tus()

{

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

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

int[] a = new int[sizeArray];

Random r = new Random();

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

{

a[i] = r.Next(-50, 100);

}

Console.WriteLine("Исходный массив:");

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

{

Console.Write(a[i] + ", ");

}

int min = a.Min(t => Math.Abs(t));

Console.WriteLine("Минимальный элемент = {0}, Index: {1}", min, Array.IndexOf(a, Math.Abs(min)));

int sum = 0;

bool flag = false;

for (int i = 1; i < sizeArray; ++i)

{

if (a[i] == 0)

flag = true;

if (flag)

sum += Math.Abs(a[i]);

}

int tmp;

Console.WriteLine("Сумма модулей элементов = " + sum);

for (int i = 0; i < sizeArray / 2; i++)

{

if (i % 2 == 0)

{

tmp = a[i + a.Length / 2];

a[i + a.Length / 2] = a[i];

a[i] = tmp;

}

}

// выводим в label2 для наглядности

foreach (int i in a)

{

Console.Write(i + ", ");

}

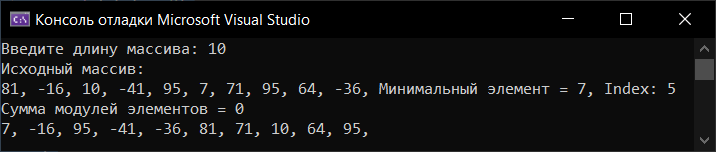
}

}

}

}

Результат выполнения программы Рис 2.1:



**Рисунок 2.1 – результат программы 6.2.**

1. Разработал программу в соответствии с заданием №3. Были реализованы класс, метод, математические функции.

Листинг программы:

using System;

using System.Linq;

namespace Lab6

{

class Program

{

static void Main(string[] args)

{

Fi.Fis();

}

public class Uan

{

public static void Uans()

{

int[] arr = new int[100];

Random rnd = new Random();

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

{

arr[i] = rnd.Next(0, 1000);

}

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

{

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

}

Console.WriteLine();

Console.Write("Введите цифру от 0 до 1000: ");

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

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

{

if (arr[i] == n)

{

Console.WriteLine($"Элемент {arr[i]} его индекс: {i}");

}

else

Console.WriteLine($"Элемента {n} в списке нет!");

}

}

}

public class Tu

{

public static void Tus()

{

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

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

int[] a = new int[sizeArray];

Random r = new Random();

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

{

a[i] = r.Next(-50, 100);

}

Console.WriteLine("Исходный массив:");

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

{

Console.Write(a[i] + ", ");

}

int min = a.Min(t => Math.Abs(t));

Console.WriteLine("Минимальный элемент = {0}, Index: {1}", min, Array.IndexOf(a, Math.Abs(min)));

int sum = 0;

bool flag = false;

for (int i = 1; i < sizeArray; ++i)

{

if (a[i] == 0)

flag = true;

if (flag)

sum += Math.Abs(a[i]);

}

int tmp;

Console.WriteLine("Сумма модулей элементов = " + sum);

for (int i = 0; i < sizeArray / 2; i++)

{

if (i % 2 == 0)

{

tmp = a[i + a.Length / 2];

a[i + a.Length / 2] = a[i];

a[i] = tmp;

}

}

// выводим в label2 для наглядности

foreach (int i in a)

{

Console.Write(i + ", ");

}

}

}

public class Fi

{

public static void Fis()

{

int[,] mas = new int[10, 10];

Random rnd = new Random();

for (int i = 0; i < 10; i++)

{

for (int j = 0; j < 10; j++)

{

mas[i, j] = -5 + rnd.Next(15);

Console.Write(mas[i, j] + " ");

}

Console.WriteLine();

}

//подсчет локальных минимумов

int count = 0;

for (int i = 1; i < 9; i++)

{

for (int j = 1; j < 9; j++)

{

if (mas[i - 1, j] > mas[i, j] && mas[i + 1, j] > mas[i, j])

if (mas[i, j - 1] > mas[i, j] && mas[i, j + 1] > mas[i, j])

count++;

}

}

Console.WriteLine("Количество локальных минимумов: " + count);

//сумма модулей элементов, расположенных выше главной диагонали

int summa = 0;

for (int i = 0; i < 10; i++)

{

for (int j = i + 1; j < 10; j++)

{

summa += Math.Abs(mas[i, j]);

}

}

Console.WriteLine("Сумма модулей элементов, расположенных выше главной диагонали = " + summa);

Console.ReadKey();

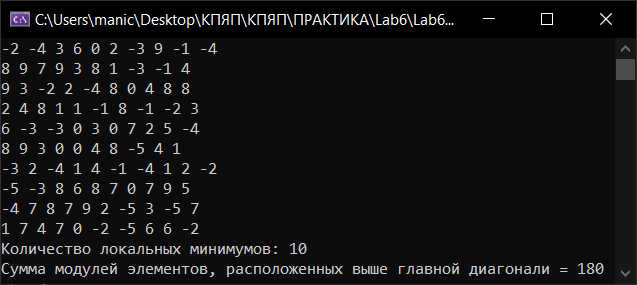
}

}

}

}

Результат выполнения программы Рис 3.1:



**Рисунок 3.1 – результат программы 6.3.**

1. Разработал программу в соответствии с заданием №4. Были реализованы класс, метод, математические функции.

Листинг программы:

using System;

using System.Linq;

namespace Lab6

{

class Program

{

static void Main(string[] args)

{

Fi.Fis();

}

public class Uan

{

public static void Uans()

{

int[] arr = new int[100];

Random rnd = new Random();

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

{

arr[i] = rnd.Next(0, 1000);

}

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

{

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

}

Console.WriteLine();

Console.Write("Введите цифру от 0 до 1000: ");

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

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

{

if (arr[i] == n)

{

Console.WriteLine($"Элемент {arr[i]} его индекс: {i}");

}

else

Console.WriteLine($"Элемента {n} в списке нет!");

}

}

}

public class Tu

{

public static void Tus()

{

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

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

int[] a = new int[sizeArray];

Random r = new Random();

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

{

a[i] = r.Next(-50, 100);

}

Console.WriteLine("Исходный массив:");

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

{

Console.Write(a[i] + ", ");

}

int min = a.Min(t => Math.Abs(t));

Console.WriteLine("Минимальный элемент = {0}, Index: {1}", min, Array.IndexOf(a, Math.Abs(min)));

int sum = 0;

bool flag = false;

for (int i = 1; i < sizeArray; ++i)

{

if (a[i] == 0)

flag = true;

if (flag)

sum += Math.Abs(a[i]);

}

int tmp;

Console.WriteLine("Сумма модулей элементов = " + sum);

for (int i = 0; i < sizeArray / 2; i++)

{

if (i % 2 == 0)

{

tmp = a[i + a.Length / 2];

a[i + a.Length / 2] = a[i];

a[i] = tmp;

}

}

// выводим в label2 для наглядности

foreach (int i in a)

{

Console.Write(i + ", ");

}

}

}

public class Fi

{

public static void Fis()

{

int[,] mas = new int[10, 10];

Random rnd = new Random();

for (int i = 0; i < 10; i++)

{

for (int j = 0; j < 10; j++)

{

mas[i, j] = -5 + rnd.Next(15);

Console.Write(mas[i, j] + " ");

}

Console.WriteLine();

}

//подсчет локальных минимумов

int count = 0;

for (int i = 1; i < 9; i++)

{

for (int j = 1; j < 9; j++)

{

if (mas[i - 1, j] > mas[i, j] && mas[i + 1, j] > mas[i, j])

if (mas[i, j - 1] > mas[i, j] && mas[i, j + 1] > mas[i, j])

count++;

}

}

Console.WriteLine("Количество локальных минимумов: " + count);

//сумма модулей элементов, расположенных выше главной диагонали

int summa = 0;

for (int i = 0; i < 10; i++)

{

for (int j = i + 1; j < 10; j++)

{

summa += Math.Abs(mas[i, j]);

}

}

Console.WriteLine("Сумма модулей элементов, расположенных выше главной диагонали = " + summa);

Console.ReadKey();

}

}

public class Fier

{

public static void Fiers()

{

Console.Write("Введите длину матрицы: ");

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

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

Random rnd = new Random();

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

{

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

{

mas[i, j] = -5 + rnd.Next(15);

Console.Write(mas[i, j] + " ");

}

Console.WriteLine();

}

//подсчет локальных минимумов

int count = 0;

for (int i = 1; i < n-1; i++)

{

for (int j = 1; j < n-1; j++)

{

if (mas[i - 1, j] > mas[i, j] && mas[i + 1, j] > mas[i, j])

if (mas[i, j - 1] > mas[i, j] && mas[i, j + 1] > mas[i, j])

count++;

}

}

Console.WriteLine("Количество локальных минимумов: " + count);

//сумма модулей элементов, расположенных выше главной диагонали

int summa = 0;

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

{

for (int j = i + 1; j < n; j++)

{

summa += Math.Abs(mas[i, j]);

}

}

Console.WriteLine("Сумма модулей элементов, расположенных выше главной диагонали = " + summa);

Console.ReadKey();

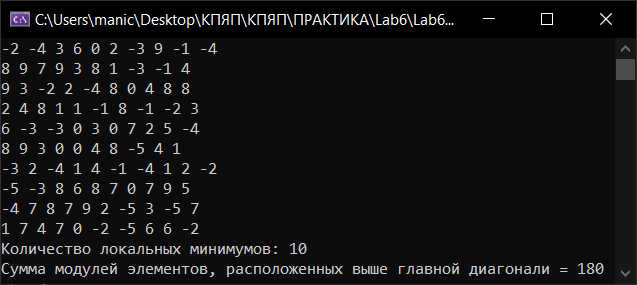
}

}

}

}

Результат выполнения программы Рис 4.1:



**Рисунок 4.1 – результат программы 6.4.**

Отметка\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Подпись преподавателя\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_