**Пермский филиал федерального государственного автономного образовательного учреждения высшего**

**“Национальный исследовательский университет**

**“Высшая школа экономики”**

Факультет экономики, менеджмента и бизнес-информатики

Дмитриев Арсений Алексеевич

**Лабораторная работа №5. «Функции и массивы»**

Отчет по практической работе

**студента образовательной программы «Программная инженерия»**

**по направлению подготовки *09.03.04 Программная инженерия***

**руководитель**

**к. тех. н., доцент каф.**

**инф. техн. в биз.**

**Викентьева О.Л.**

**Пермь, 2019 г.**

**Постановка задачи**

**Общая:**

1. Сформировать динамический одномерный массив, заполнить его случайными числами и вывести на печать.

2. Выполнить указанное в варианте задание и вывести полученный массив на печать.

3.  Сформировать динамический двумерный массив, заполнить его случайными числами и вывести на печать.

4.  Выполнить указанное в варианте задание и вывести полученный массив на печать.

5.  Сформировать динамический рваный массив, заполнить его случайными числами и вывести на печать.

6. Выполнить указанное в варианте задание и вывести полученный массив на печать.

**7.**При реализации функций необходимо продемонстрировать использование параметров разных типов и различные способы организации функций (параметры по умолчанию, перегрузку функций, и .т.д.)

**Вариант №7:**

1. Сформировать динамический одномерный массив, заполнить его случайными числами и вывести на печать.

2. Удалить все четные элементы одномерного массива и вывести полученный массив на печать.

3.  Сформировать динамический двумерный массив, заполнить его случайными числами и вывести на печать.

4.  Добавить К строк в конец матрицы и вывести полученную матрицу на печать.

5.  Сформировать динамический рваный массив, заполнить его случайными числами и вывести на печать.

6. Удалить первую строку, в которой встречаются нули и вывести полученный массив на печать.

**7.**При реализации функций необходимо продемонстрировать использование параметров разных типов и различные способы организации функций (параметры по умолчанию, перегрузку функций, и .т.д.)

**Анализ**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Функция | Входные данные | Классы входных данных | Выходные данные |
| Основные функции | | | | |
| 1 | Формирование одномерного массива | Размер массива | Целое число >=1 | Одномерный массив |
| Целое число <=0 | Сообщение об ошибке |
| 1.1 | Формирование одномерного массива с помощью ДСЧ | Вещественное число |
| Нечисло |
| 1.2 | Формирование одномерного массива с клавиатуры | Элементы массива | Целое число | Одномерный массив |
| Вещественное число | Сообщение об ошибке |
| Нечисло |
| 2 | Удаление четных элементов из одномерного массива | * Массив * Размер массива | Непустой массив, есть и четные, и нечетные | Измененный массив |
|
|
| Непустой массив, только нечетные элементы | Массив без изменений |
|
|
| Непустой массив, только четные элементы | Пустой массив |
|
|
| Пустой массив | Сообщение об ошибке |
| 3 | Формирование двумерного массива | Количество строк в массиве | Целое число >=1 | Двумерный массив |
| Целое число <=0 | Сообщение об ошибке |
| Формирование двумерного массива с помощью ДСЧ | Вещественное число |
| Нечисло |
| 3.1 | Количество столбцов в массиве | Целое число >=1 | Двумерный массив |
| Целое число <=0 | Сообщение об ошибке |
| Вещественное число |
| Нечисло |
| 3.2 | Формирование двумерного массива с клавиатуры | Элементы массива | Целое число | Двумерный массив |
| Вещественное число | Сообщение об ошибке |
| Нечисло |
| 4 | Добавление К строк в конец матрицы | К - число добавляемых строк | Целое число >=1 | Измененный массив |
| Целое число <=0 | Сообщение об ошибке |
| Вещественное число |
| Нечисло |
| 5 | Формирование рваного массива | Количество строк в массиве | Целое число >=1 | Рваный массив |
| Целое число <=0 | Сообщение об ошибке |
| Формирование рваного массива с помощью ДСЧ | Вещественное число |
| Нечисло |
| 5.1 | Количество элементов в каждой строке | Целое число >=1 | Рваный массив |
| Целое число <=0 | Сообщение об ошибке |
| Вещественное число |
| Нечисло |
| 5.2 | Формирование рваного массива с клавиатуры | Элементы массива | Целое число | Рваный массив |
| Вещественное число | Сообщение об ошибке |
| Нечисло |
| 6 | Удаление первой строки, содержащей хотя бы один ноль | Массив | Непустой массив, содержащий нули | Измененный массив |
|
| Непустой массив, в котором нет элементов равных 0 | Массив без изменений |
|
| Пустой массив | Сообщение об ошибке |
|
| Вспомогательные функции | | | | |
| 7 | Печать массива | Массив | Непустой массив | Напечатанный массив |
| Пустой массив | Сообщение о пустоте |
| 8 | Ввод целого числа | * Строка, введенная пользователем * Нижняя граница допустимого диапазона * Верхняя граница диапазона | Введенная строка – число, попадающее в допустимый диапазон от нижней до верхней границы | Целое число |
|
|
|
| Строка - не целое число | Сообщение об ошибке |
| Строка - нечисло |
| Строка - целое число, не попадающее в допустимый диапазон |
|
|
| Строка - целое число, > 2^31-1 или <-2^31 |
|
| 9 | Печать диалогового меню для пользователя |  |  |  |
|

**Проектирование**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Функция | Входные данные | Выходные данные |
| Основные функции | | | |
| 1 | CreateRandomArray  (Формирование одномерного массива с помощью ДСЧ) | int sizeOfArray - размер массива | int[] arr - массив |
|
|
|
| 2 | ReadArray  (Формирование одномерного массива с клавиатуры) | int sizeOfArray - размер массива | int[] arr - массив |
|
|
|
| 3 | ArrayDeleteEven  (Удаление четных элементов из одномерного массива) | int[] arr - массив | int[] newArr - измененный массив, Сообщение об ошибке |
|
|
|
| 4 | CreateRandomMatrix (Формирование двумерного массива с помощью ДСЧ) | int row - количество строк, int column - количество столбцов | int[,] matr - двумерный массив |
|
|
|
| 5 | ReadMatrix  (Формирование двумерного массива с клавиатуры) | int row - количество строк, int column - количество столбцов | int[,] matr - двумерный массив |
|
|
|
| 6 | MatrixAddRows  (Добавление К строк в конец матрицы) | int[,] matr - двумерный массив, int k - количество добавляемых строк | int[,] newMatr - изменный массив |
|
|
|
| 7 | CreateRandomJagged (Формирование рваного массива с помощью ДСЧ) | int row - количество строк | int[][] jag - рваный массив |
|
|
|
| 8 | ReadJagged  (Формирование рваного массива с клавиатуры) | int row - количество строк | int[][] jag - рваный массив |
|
|
|
| 9 | JaggedDeleteZeroRow  (Удаление первой строки, содержащей хотя бы один ноль) | int[][] jag - рвваный массив | int[][] newJag - изменный массив, Сообщение об ошибке |
|
|
|
| Вспомогательные функции | | | |
| 10 | PrintArray  (Печать массива) | int[] arr - одномерный массив ИЛИ int[,] matr -двумерный массив ИЛИ int[][] jag - рваный массив |  |
|
|
|
| 11 | ReadInteger  (Ввод целого числа) | int lowerBoud - нижняя граница допустимого диапазона, int upperBound - верхняя граница допустимого диапазона | int number - целое число |
|
|
|
| 12 | MainMenu  (Печать диалогового меню для пользователя) |  |  |
|
|
|

**Программа**

using System;

namespace Laba5

{

class Program

{

const int MAX\_ELEM = 1000;

const int MIN\_ELEM = -1000;

const int MAX\_LENGTH = 1000; // Максимальное кол-во элементов в одномерном массиве/строке двумерного или рваного массивов

const int MAX\_DEPTH = 25; // Максимальное кол-во строк в двумерном или рваном массиве

const int MIN\_SIZE = 1;

static int ReadInteger(string userInstruction, int lowerBound = MIN\_ELEM, int upperBound = MAX\_ELEM)

{

int num = 0;

bool ok = false;

string input;

do

{

try

{

Console.Write(userInstruction);

input = Console.ReadLine();

num = Convert.ToInt32(input);

if (num >= lowerBound && num <= upperBound) ok = true;

else Console.WriteLine("Ошибка: введенное число должно попадать в промежуток от {0} до {1}", lowerBound, upperBound);

}

catch (FormatException)

{

Console.WriteLine("Ошибка: введите целое число");

}

catch (OverflowException)

{

Console.WriteLine("Ошибка: введено слишком большое (или маленькое) число");

}

} while (!ok);

return num;

}

static void PrintArray(int[] arr, string message = "Сформированный одномерный массив:")

{

Console.WriteLine();

if (arr.Length == 0) Console.WriteLine("Массив пустой");

else

{

Console.WriteLine(message);

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

{

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

}

Console.WriteLine();

}

}

static void PrintArray(int[,] arr, string message = "Сформированный двумерный массив:")

{

Console.WriteLine();

if (arr.GetLength(0) == 0) Console.WriteLine("Массив пустой");

else

{

Console.WriteLine(message);

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

{

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

{

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

}

Console.WriteLine();

}

}

}

static void PrintArray(int[][] arr, string message = "Сформированный рваный массив:")

{

Console.WriteLine();

if (arr.GetLength(0) == 0) Console.WriteLine("Массив пустой");

else

{

Console.WriteLine(message);

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

{

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

{

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

}

Console.WriteLine();

}

}

}

static int RedefineOption(int userChoise, bool arr, bool mat, bool jag)

{

if ((userChoise >= 0 && userChoise <= 3) || (arr && mat && jag)) return userChoise;

bool[] flags = { arr, mat, jag };

if(userChoise == 4)

{

for(int i = 0; i<3 && !flags[i]; i++)

{

userChoise++;

}

} else if(userChoise == 5)

{

if (flags[0] == false) userChoise++;

else if(flags[1] == false) userChoise++;

}

return userChoise;

}

#region Array

static int[] CreateRandomArray(int sizeOfArray)

{

int[] arr = new int[sizeOfArray];

Random rand = new Random();

for (int i = 0; i < sizeOfArray; i++) arr[i] = rand.Next(MIN\_ELEM, MAX\_ELEM);

return arr;

}

static int[] ReadArray(int sizeOfArray)

{

int[] arr = new int[sizeOfArray];

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

arr[i] = ReadInteger($"Введите {i + 1} элемент: ");

return arr;

}

static int[] ArrayDeleteEven(int[] arr)

{

int sizeArr = arr.Length;

int sizeNew = 0;

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

{

if (arr[i] % 2 != 0) sizeNew++;

}

int[] newArr = new int[sizeNew];

int newInd = 0;

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

{

if (arr[i] % 2 != 0)

{

newArr[newInd] = arr[i];

newInd++;

}

}

if (sizeNew == sizeArr) Console.WriteLine("\nВ массиве нет четных элементов для удаления!");

else PrintArray(newArr, "После удаления четных элементов массив выглядит так:");

return newArr;

}

#endregion

#region Matrix

static int[,] CreateRandomMatrix(int row, int column)

{

int[,] matr = new int[row, column];

Random rand = new Random();

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

{

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

{

matr[i, j] = rand.Next(MIN\_ELEM, MAX\_ELEM);

}

}

return matr;

}

static int[,] ReadMatrix(int row, int column)

{

int[,] matr = new int[row, column];

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

{

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

{

matr[i, j] = ReadInteger($"Введите {j + 1} элемент {i + 1} строки: ");

}

}

return matr;

}

static int[,] MatrixAddRows(int[,] matr, int k)

{

int row = matr.GetLength(0);

int col = matr.GetLength(1);

int[,] newMatr = new int[row + k, col];

Console.WriteLine("Выберите вариант формирования:");

Console.WriteLine(" 1: Сформировать с помощью ДСЧ");

Console.WriteLine(" 2: Сформировать вручную, с клавиатуры");

int userChoise = ReadInteger("Введите номер выбранной опции: ", 1, 2);

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

{

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

{

newMatr[i, j] = matr[i, j];

}

}

int[,] addMatr;

if (userChoise == 1) addMatr = CreateRandomMatrix(k, col);

else addMatr = ReadMatrix(k, col);

for (int i = row; i < row + k; i++)

{

int t = i - row;

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

{

newMatr[i, j] = addMatr[t, j];

}

}

PrintArray(newMatr, "После добавления строк массив стал таким: ");

return newMatr;

}

#endregion

#region Jagged

static int[][] CreateRandomJagged(int row)

{

int[][] jag = new int[row][];

Random rand = new Random();

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

{

int size = ReadInteger($"Введите длину {i + 1} строки: ", MIN\_SIZE, MAX\_LENGTH);

jag[i] = new int[size];

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

{

jag[i][j] = rand.Next(MIN\_ELEM, MAX\_ELEM);

}

}

return jag;

}

static int[][] ReadJagged(int row)

{

int[][] jag = new int[row][];

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

{

Console.WriteLine();

int size = ReadInteger($"Введите длину {i + 1} строки: ", MIN\_SIZE, MAX\_LENGTH);

jag[i] = new int[size];

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

{

jag[i][j] = ReadInteger($"Введите {j + 1} элемент строки: ");

}

}

return jag;

}

static int[][] JaggedDeleteZeroRow(int[][] jag)

{

int rowDel = -1;

int row = jag.GetLength(0);

bool find = false;

for (int i = 0; i < row && !find; i++)

{

for (int j = 0; j < jag[i].Length; j++)

{

if (jag[i][j] == 0)

{

find = true;

rowDel = i;

}

}

}

if (!find)

{

Console.WriteLine("В массиве отсутствуют нули! Ничего не было удалено!");

return jag;

}

else

{

int[][] jag1 = new int[row - 1][];

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

{

jag1[i] = new int[jag[i].Length];

for (int j = 0; j < jag[i].Length; j++)

{

jag1[i][j] = jag[i][j];

}

}

for (int i = rowDel + 1; i < row; i++)

{

jag1[i - 1] = new int[jag[i].Length];

for (int j = 0; j < jag[i].Length; j++)

{

jag1[i - 1][j] = jag[i][j];

}

}

PrintArray(jag1, "После удаления строк массив имеет такой вид:");

return jag1;

}

}

#endregion

#region inteface

static void CreateNewArray(string instruction, out int[] arr)

{

Console.WriteLine(instruction);

int sizeOfArray = ReadInteger("Введите размер массива: ", MIN\_SIZE, MAX\_LENGTH);

Console.WriteLine("Выберите вариант формирования:");

Console.WriteLine(" 1: Сформировать с помощью ДСЧ");

Console.WriteLine(" 2: Сформировать вручную, с клавиатуры");

int userChoise = ReadInteger("Введите номер выбранной опции: ", 1, 2);

arr = new int[sizeOfArray];

Console.WriteLine();

if (userChoise == 1) arr = CreateRandomArray(sizeOfArray);

else arr = ReadArray(sizeOfArray);

PrintArray(arr);

}

static void CreateNewArray(string instruction, out int[,] matr)

{

Console.WriteLine(instruction);

int row = ReadInteger("Введите количество строк в матрице: ", MIN\_SIZE, MAX\_DEPTH);

int column = ReadInteger("Введите количество столбцов: ", MIN\_SIZE, MAX\_LENGTH);

Console.WriteLine("Выберите вариант формирования:");

Console.WriteLine(" 1: Сформировать с помощью ДСЧ");

Console.WriteLine(" 2: Сформировать вручную, с клавиатуры");

Console.WriteLine();

int userChoise = ReadInteger("Введите номер выбранной опции: ", 1, 2);

matr = new int[row, column];

if (userChoise == 1) matr = CreateRandomMatrix(row, column);

else matr = ReadMatrix(row, column);

PrintArray(matr);

}

static void CreateNewArray(string instruction, out int[][] jag)

{

Console.WriteLine(instruction);

int row = ReadInteger("Введите количество строк в массиве: ", MIN\_SIZE, MAX\_DEPTH);

Console.WriteLine("Выберите вариант формирования:");

Console.WriteLine(" 1: Сформировать с помощью ДСЧ");

Console.WriteLine(" 2: Сформировать вручную, с клавиатуры");

Console.WriteLine();

int userChoise = ReadInteger("Введите номер выбранной опции: ", 1, 2);

jag = new int[row][];

if (userChoise == 1) jag = CreateRandomJagged(row);

else jag = ReadJagged(row);

PrintArray(jag);

}

static void MainMenu(bool arr, bool mat, bool jag)

{

string word = (arr || mat || jag) ? "новый " : "";

Console.WriteLine("\nДоступные опции:");

Console.WriteLine($" 1: Сформировать {word}одномерный массив");

Console.WriteLine($" 2: Сформировать {word}двумерный массив");

Console.WriteLine($" 3: Сформировать {word}рваный массив");

int ind = 4;

if (arr)

{

Console.WriteLine($" {ind++}: Удалить все четные элементы в одномерном массиве");

}

if (mat)

{

Console.WriteLine($" {ind++}: Добавить строки в конец матрицы");

}

if (jag)

{

Console.WriteLine($" {ind++}: Удалить первую строку в рваном массиве, содержащую нули");

}

Console.WriteLine(" 0: Завершить работу программы");

}

static void Initial(out int[] arr, out int[,] matr, out int[][] jag, ref bool[] flags)

{

Console.WriteLine("Программа работает с одномерными, двумерными и рваными массивами");

Console.WriteLine("Для начала сформируем массив:");

MainMenu(flags[1], flags[2], flags[3]);

int userChoise = ReadInteger("Введите номер выбранной опции: ", 0, 3);

arr = null;

matr = null;

jag = null;

switch (userChoise)

{

case 1:

CreateNewArray("", out arr);

break;

case 2:

CreateNewArray("", out matr);

break;

case 3:

CreateNewArray("", out jag);

break;

}

flags[userChoise] = true;

}

static void EmptyArraySituation(string cause, out int[] arr, ref bool arrF, ref bool finish)

{

Console.WriteLine($"После {cause} одномерный массив стал пустым");

Console.WriteLine("Выберите вариант продолжения работы:");

Console.WriteLine(" 1: Сформировать новый одномерный массив");

Console.WriteLine(" 2: Продолжить работу с другими типами массивов");

Console.WriteLine(" 0: Прекратить работу программы");

arr = null;

int userChoise = ReadInteger("Введите номер выбранной опции: ", 0, 2);

if (userChoise == 1)

{

CreateNewArray("Формирование нового одномерного массива:", out arr);

arrF = true;

finish = false;

}

else if (userChoise == 2)

{

arrF = false;

finish = false;

}

else

{

finish = true;

}

}

static void EmptyArraySituation(string cause, out int[][] jag, ref bool jagF, ref bool finish)

{

Console.WriteLine($"После {cause} одномерный массив стал пустым");

Console.WriteLine("Выберите вариант продолжения работы:");

Console.WriteLine(" 1: Сформировать новый одномерный массив");

Console.WriteLine(" 2: Продолжить работу с другими типами массивов");

Console.WriteLine(" 0: Прекратить работу программы");

jag = null;

int userChoise = ReadInteger("Введите номер выбранной опции: ", 0, 2);

if (userChoise == 1)

{

CreateNewArray("Формирование нового рваного массива:", out jag);

jagF = true;

finish = false;

}

else if (userChoise == 2)

{

jagF = false;

finish = false;

}

else

{

finish = true;

}

}

#endregion

static void Main(string[] args)

{

bool[] flags = { false, false, false, false };

int[] arr;

int[,] matr;

int[][] jag;

Initial(out arr,out matr,out jag,ref flags);

bool arrF = flags[1], matF = flags[2], jagF = flags[3];

int options = 4;

bool finish = flags[0];

do

{

MainMenu(arrF, matF, jagF);

int userChoise = ReadInteger("Введите номер выбранной опции: ", 0, options);

Console.WriteLine();

userChoise = RedefineOption(userChoise, arrF , matF, jagF);

switch (userChoise)

{

case 0:

{

finish = true;

Console.WriteLine("Завершаю работу");

break;

}

case 1:

{

if (arrF) CreateNewArray("Формирование нового одномерного массива:", out arr);

else

{

CreateNewArray("Формирование одномерного массива",out arr);

options++;

arrF = true;

}

break;

}

case 2:

{

if (matF) CreateNewArray("Формирование нового двумерного массива:",out matr);

else

{

CreateNewArray("Формирование двумерного массива:",out matr);

options++;

matF = true;

}

break;

}

case 3:

{

if (jagF) CreateNewArray("Формирование нового рваного массива:", out jag);

else

{

CreateNewArray("Формирование рваного массива:", out jag);

options++;

jagF = true;

}

break;

}

case 4:

{

arr = ArrayDeleteEven(arr);

if (arr.Length == 0)

{

EmptyArraySituation("удаления четных элементов массива", out arr, ref arrF, ref finish);

if (!arrF) options--;

}

break;

}

case 5:

{

int pos = ReadInteger("Сколько строк нужно добавить: ", 1, MAX\_DEPTH-matr.GetLength(0));

matr = MatrixAddRows(matr, pos);

break;

}

case 6:

{

jag = JaggedDeleteZeroRow(jag);

if (jag.GetLength(0) == 0)

{

EmptyArraySituation("после удаления рядов с одним или более нулями", out jag, ref jagF, ref finish);

if (!jagF) options--;

}

break;

}

}

} while (!finish);

}

}

}

**Тестирование**

1. Формирование одномерного массива

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Критерии\тесты | Т1 | Т2 | Т3 | Т4 |
| Критерии входных данных: размер массива | | | | |
| Целое число >=1 и <=1000 | + | + | + | + |
| Целое число <1 или >1000 |  | + |  |  |
| Вещественное число |  |  | + |  |
| Нечисло | + |  |  |  |
| Критерии входных данных: элементы массива | | | | |
| |Целое число| <=10000 | + | + | + | + |
| |Целое число| >10000 |  | + |  |  |
| Вещественное число |  | + |  |  |
| Нечисло |  |  | + |  |
| Критерии длины набора | | | | |
| Единичный набор |  |  | + |  |
| Длина >1 | + | + |  | + |
| Критерии выходных данных | | | | |
| Массив | + | + | + | + |
| Сообщение об ошибке | + | + | + |  |

|  |  |  |  |
| --- | --- | --- | --- |
| № | Ввод | Ожидаемый результат | Результат работы |
| Т1 | t 2 -11 18 | Сообщение об ошибке  Массив | Сообщение об ошибке  Массив |
| Т2 | -13 3 2,8 5 ^^ 14 0 | Сообщение об ошибке  Массив | Сообщение об ошибке  Массив |
| Т3 | 100,1 1 ft 3 | Сообщение об ошибке  Массив | Сообщение об ошибке  Массив |
| Т4 | 4 2 12 -8 13 | Массив | Массив |

1. Удаление четных элементов

|  |  |  |  |
| --- | --- | --- | --- |
| № | Ввод | Ожидаемый результат | Результат работы |
| Т1 | 2 4 3 1 8 7 | 3 1 7 | 3 1 7 |
| Т2 | 1 3 -5 17 | 1 3 -5 17 | 1 3 -5 17 |
| Т3 | 2 8 -44 0 -6 | Пустой массив | Пустой массив |

|  |  |  |  |
| --- | --- | --- | --- |
| Критерии\тесты | Т1 | Т2 | Т3 |
| Критерии входных данных: массив | | | |
| Не пустой | + | + | + |
| Критерии длины набора: четные элементы | | | |
| Четных =0 |  | + |  |
| Четных >=1 | + |  |  |
| Четных = размер массива |  |  | + |
| Критерии выходных данных | | | |
| Массив без изменений |  | + |  |
| Измененный массив | + |  |  |
| "Массив стал пустым" |  |  | + |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Критерии\тесты | Т1 | Т2 | Т3 | Т4 |
| Критерии входных данных: число строк | | | | |
| Целое число >=1 и <=25 | + | + | + | + |
| Целое число <1 или >25 | + |  |  |  |
| Вещественное число |  | + |  |  |
| Нечисло |  |  | + |  |
| Критерии входных данных: число столбцов | | | | |
| Целое число >=1 и <=1000 | + | + | + | + |
| Целое число <1 или >1000 |  |  | + |  |
| Вещественное число | + |  |  |  |
| Нечисло |  | + |  |  |
| Критерии входных данных: элементы массива | | | | |
| |Целое число| <=10000 | + | + | + | + |
| |Целое число| >10000 |  | + |  |  |
| Вещественное число |  |  | + |  |
| Нечисло | + |  |  |  |
| Критерии выходных данных | | | | |
| Массив | + | + | + | + |
| Сообщение об ошибке | + | + | + |  |

1. Формирование матрицы

|  |  |  |  |
| --- | --- | --- | --- |
| № | Ввод | Ожидаемый результат | Результат работы |
| Т1 | 0 2 1,1 4  2 a 3 c -1 7  E 14 3 17 0 | Сообщение об ошибке  Массив | Сообщение об ошибке  Массив |
| Т2 | 2,5 3 vtb 2  6 -100000 17  222222 4 1  3 -12 | Сообщение об ошибке  Массив | Сообщение об ошибке  Массив |
| Т3 | А\* 2 -3 3  4 1,1 2 2  1,001 5 4,1 -1 | Сообщение об ошибке  Массив | Сообщение об ошибке  Массив |
| T4 | 3 5  1 2 3 4 5  -100 -101 -102 -1 -2  6 10 19 24 0 | Массив | Массив |

1. Добавление строк в конец матрицы

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Критерии\тесты | Т1 | Т2 | Т3 | Т4 |
| Вход: Число добавляемых строк | | | | |
| Целое число >=1 и <=(25-текущее) | + | + | + | + |
| Целое число <1 или >(25-текущие) | + |  |  |  |
| Вещестевенное число |  | + |  |  |
| Нечисло |  |  | + |  |
| Критерии входных данных: добавляемые элементы | | | | |
| |Целое число| <=10000 | + | + | + | + |
| |Целое число| >10000 |  | + |  |  |
| Вещестевенное число | + |  |  |  |
| Нечисло |  |  | + |  |
| Критерии выходных данных | | | | |
| Измененный массив | + | + | + | + |
| Сообщение об ошибке | + | + | + |  |

|  |  |  |  |
| --- | --- | --- | --- |
| № | Ввод | Ожидаемый результат | Результат работы |
| Т1 | -5 1 (3 4)  3 4,4 5 6 7 | Сообщение об ошибке  Измененный массив | Сообщение об ошибке  Измененный массив |
| Т2 | 11,92 2 (1 5)  4 -1 -10101 1 1 1  3 7 34564 16 -26 13 | Сообщение об ошибке  Измененный массив | Сообщение об ошибке  Измененный массив |
| Т3 | Btw 1 (6 3)  12 d 4 df 4 | Сообщение об ошибке  Измененный массив | Сообщение об ошибке  Измененный массив |
| Т4 | 3 (4 2)  1 5  -4 6  9 17 | Измененный массив | Измененный массив |

(a, b) – размеры матрицы, к которой добавляют строки. Эти матрицы заполняются с помощью ДСЧ

1. Формирование рваного массива

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Критерии\тесты | Т1 | Т2 | Т3 | Т4 |
| Критерии входных данных: число строк | | | | |
| Целое число >=1 и <=25 | + | + | + | + |
| Целое число <1 или >25 | + |  |  |  |
| Вещественное число |  | + |  |  |
| Нечисло |  |  | + |  |
| Критерии входных данных: число элементов каждой строки | | | | |
| Целое число >=1 и <=1000 | + | + | + | + |
| Целое число <1 или >1000 |  | + |  |  |
| Вещественное число |  |  | + |  |
| Нечисло | + |  |  |  |
| Критерии входных данных: элементы массива | | | | |
| |Целое число| <=10000 | + | + | + | + |
| |Целое число| >10000 |  |  | + |  |
| Вещественное число | + |  |  |  |
| Нечисло |  | + |  |  |
| Критерии выходных данных | | | | |
| Массив | + | + | + | + |
| Сообщение об ошибке | + | + | + |  |

|  |  |  |  |
| --- | --- | --- | --- |
| № | Ввод | Ожидаемый результат | Результат работы |
| Т1 | 30 3  1 1  ed 3 1 4 10  2 1,1 1,2 2 3 | Сообщение об ошибке  Массив | Сообщение об ошибке  Массив |
| Т2 | 5,8 2  0 4 3 f 5 t 10 3  3 3 3 3 | Сообщение об ошибке  Массив | Сообщение об ошибке  Массив |
| Т3 | \*\_\* 2  1 35  2 1000009 -89 15 | Сообщение об ошибке  Массив | Сообщение об ошибке  Массив |
| Т4 | 4  3 1 2 3  1 -1586  4 326 58 510 129  2 -39 210 | Массив | Массив |

1. Удаление первой строки, содержащей ноль

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Критерии\тесты | Т1 | Т2 | Т3 | Т4 | Т5 |
| Критерии входных данных: массив | | | | | |
| Не пустой | + | + | + | + | + |
| Критерии длины набора: четные элементы | | | | | |
| Строк с нулями = 0 | + |  |  |  |  |
| Строк с нулями >= 1 |  | + | + | + |  |
| Строк всего = 1, Строк с нулями = 1 |  |  |  |  | + |
| Критерии поиска: первая строка с нулями | | | | | |
| В начале |  | + |  |  |  |
| В середине |  |  | + |  |  |
| В конце |  |  |  | + |  |
| Критерии выходных данных | | | | | |
| Массив без изменений | + |  |  |  |  |
| Измененный массив |  | + | + | + |  |
| "Массив стал пустым" |  |  |  |  | + |

|  |  |  |  |
| --- | --- | --- | --- |
| № | Ввод | Ожидаемый результат | Результат работы |
| Т1 | 3  6 1 4 -8 3 15 2  5 6 14 12 -9 -7  1 1 | Массив без изменений | Массив без изменений |
| Т2 | 3  4 2 1 0 6  3 1 7 -19  4 0 0 0 1 | 1 7 -19  0 0 0 1 | 1 7 -19  0 0 0 1 |
| Т3 | 4  2 -4 10  3 1 7 15  1 0  3 -8 12 0 | -4 10  1 7 15  -8 12 0 | -4 10  1 7 15  -8 12 0 |
| Т4 | 4  6 1 49 -15 8 10 -78  3 18 -7 1113  4 14 -5 163 1  5 0 1 2 3 4 | 1 49 -15 8 10 -78  18 -7 1113  14 -5 163 1 | 1 49 -15 8 10 -78  18 -7 1113  14 -5 163 1 |
| Т5 | 1  6 4 8 5 1 0 -12 | Пустой массив | Пустой массив |