Министерство образования и науки Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования

«**Пермский национальный исследовательский политехнический университет**»

Кафедра «Информационные технологии и автоматизированные системы»

ОТЧЁТ

по лабораторной работе №6

Дисциплина: «Объектно-ориентированное программирование»

Тема: Класс Array. Строки. Класс String.

Вариант 6

Выполнил:

Студент группы ИВТ-20-2б

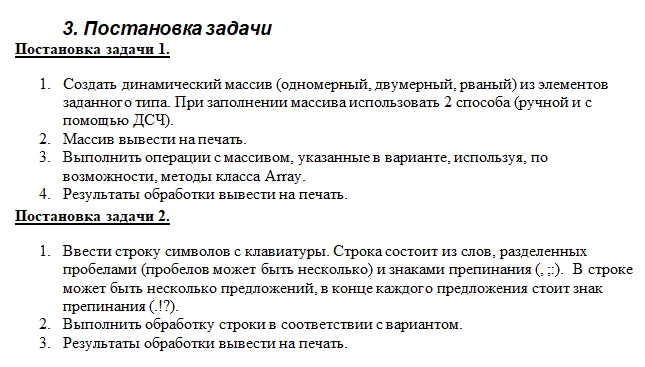
Карелов Вадим Андреевич

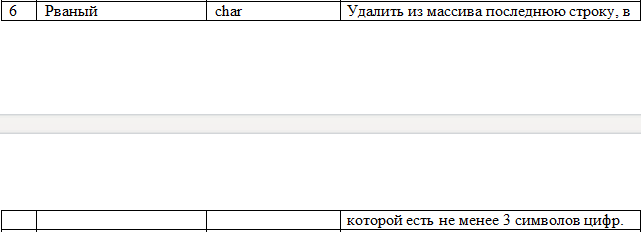
Проверила:

Доцент кафедры ИТАС

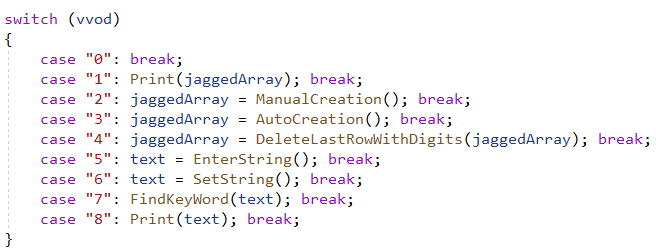
Викентьева О.Л.

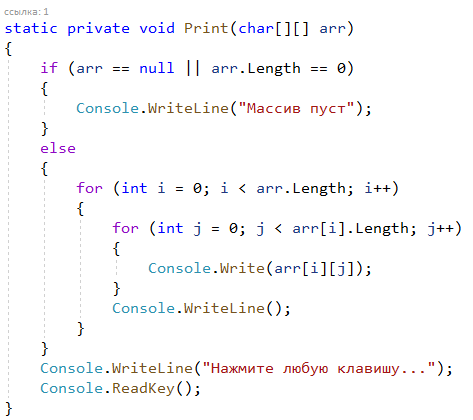
Пермь, 2021

**Постановка задачи**



**Анализ**

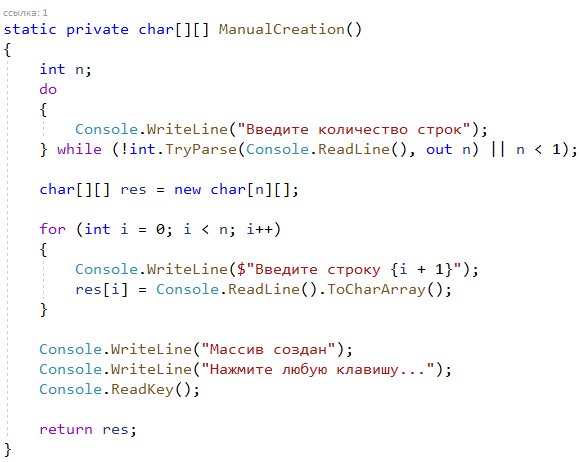
1. Главное меню. Позволяет перейти к любому пункту задания. 
2. Задание 1. Работа с динамическим массивом.
   1. Вывод массива на экран.



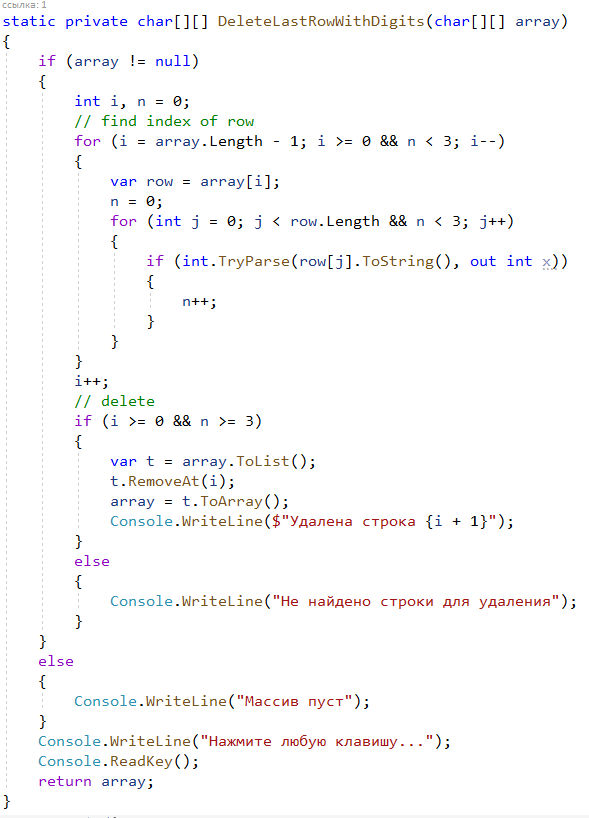
* 1. Создание массива. Автоматическое создание.



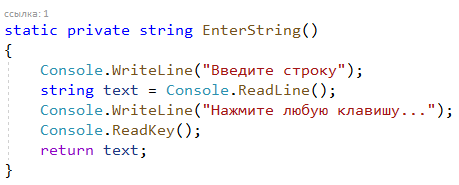
* 1. Создание массива. Ручное создание.



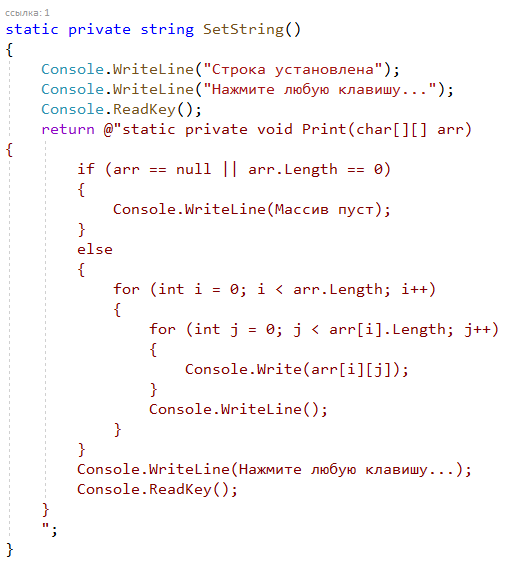
* 1. Удаление последней строки, в которой не меньше 3-х чисел.



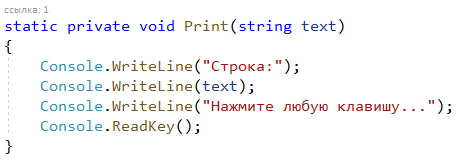
1. Задание 2. Работа со строками.
   1. Ввод строки



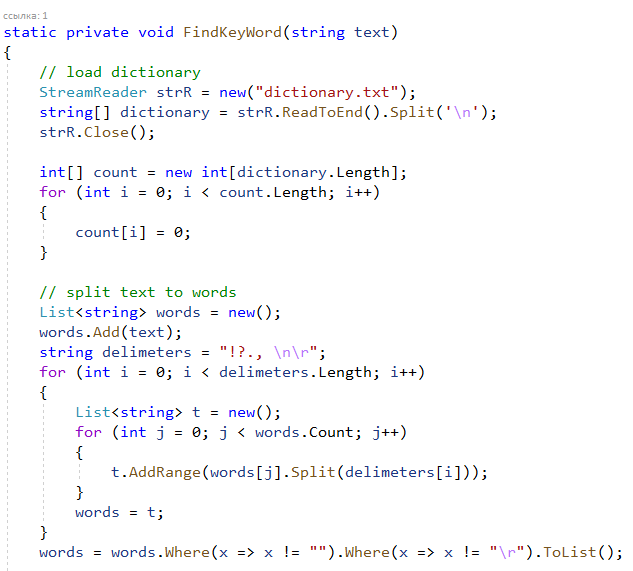
* 1. Автоматическое заполнение строки кусочком кода из C#.

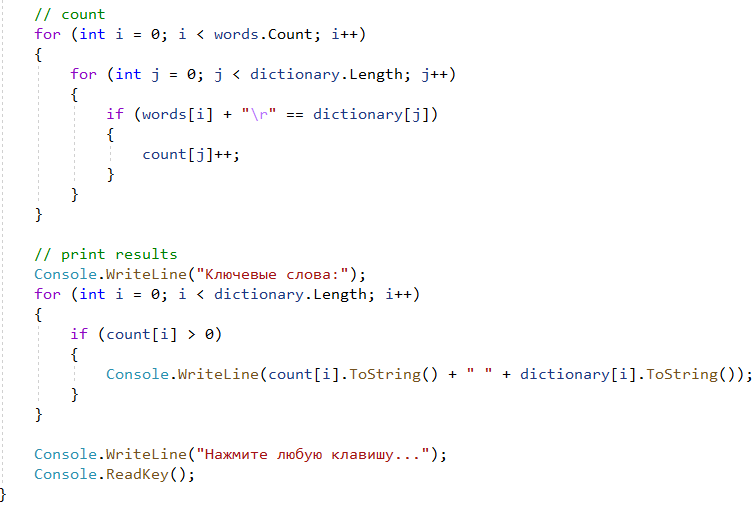


* 1. Вывод строки на экран



* 1. Поиск ключевых слов





**Код**

using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

namespace LW\_2\_06

{

class Program

{

private static Random rn = new();

static void Main(string[] args)

{

char[][] jaggedArray = null;

string text = "";

string vvod = "";

while (vvod != "0")

{

Console.Clear();

Console.WriteLine(@"0 - завершение работы программы

Задание 1

1 - вывод массива на экран

2 - ввод массива char[][] вручную

3 - заполнение массива char[][] ДСЧ

4 - удалить последнюю строку, в которой не меньше 3-х цифр

Задание 2

5 - ввод строки вручную

6 - ввод приготовленного текста

7 - найти ключевые слова C#

8 - распечатать строку

");

vvod = Console.ReadLine();

switch (vvod)

{

case "0": break;

case "1": Print(jaggedArray); break;

case "2": jaggedArray = ManualCreation(); break;

case "3": jaggedArray = AutoCreation(); break;

case "4": jaggedArray = DeleteLastRowWithDigits(jaggedArray); break;

case "5": text = EnterString(); break;

case "6": text = SetString(); break;

case "7": FindKeyWord(text); break;

case "8": Print(text); break;

}

}

}

static private void Print(char[][] arr)

{

if (arr == null || arr.Length == 0)

{

Console.WriteLine("Массив пуст");

}

else

{

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

{

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

{

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

}

Console.WriteLine();

}

}

Console.WriteLine("Нажмите любую клавишу...");

Console.ReadKey();

}

static private char[][] AutoCreation()

{

int n;

do

{

Console.WriteLine("Введите количество строк");

} while (!int.TryParse(Console.ReadLine(), out n) || n < 1);

string letters = "qwertyuiopasdfghjklzxcvbnmQWERTYUIOPASDFGHJKLZXCVBNM0123456789 ";

char[][] res = new char[n][];

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

{

int length = rn.Next(0, 30);

string row = "";

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

{

row += letters[rn.Next(0, letters.Length)]; // t

}

res[i] = row.ToCharArray();

}

Console.WriteLine("Массив создан");

Console.WriteLine("Нажмите любую клавишу...");

Console.ReadKey();

return res;

}

static private char[][] ManualCreation()

{

int n;

do

{

Console.WriteLine("Введите количество строк");

} while (!int.TryParse(Console.ReadLine(), out n) || n < 1);

char[][] res = new char[n][];

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

{

Console.WriteLine($"Введите строку {i + 1}");

res[i] = Console.ReadLine().ToCharArray();

}

Console.WriteLine("Массив создан");

Console.WriteLine("Нажмите любую клавишу...");

Console.ReadKey();

return res;

}

static private char[][] DeleteLastRowWithDigits(char[][] array)

{

if (array != null)

{

int i, n = 0;

// find index of row

for (i = array.Length - 1; i >= 0 && n < 3; i--)

{

var row = array[i];

n = 0;

for (int j = 0; j < row.Length && n < 3; j++)

{

if (int.TryParse(row[j].ToString(), out int x))

{

n++;

}

}

}

i++;

// delete

if (i >= 0 && n >= 3)

{

var t = array.ToList();

t.RemoveAt(i);

array = t.ToArray();

Console.WriteLine($"Удалена строка {i + 1}");

}

else

{

Console.WriteLine("Не найдено строки для удаления");

}

}

else

{

Console.WriteLine("Массив пуст");

}

Console.WriteLine("Нажмите любую клавишу...");

Console.ReadKey();

return array;

}

static private string EnterString()

{

Console.WriteLine("Введите строку");

string text = Console.ReadLine();

Console.WriteLine("Нажмите любую клавишу...");

Console.ReadKey();

return text;

}

static private string SetString()

{

Console.WriteLine("Строка установлена");

Console.WriteLine("Нажмите любую клавишу...");

Console.ReadKey();

return @"static private void Print(char[][] arr)

{

if (arr == null || arr.Length == 0)

{

Console.WriteLine(Массив пуст);

}

else

{

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

{

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

{

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

}

Console.WriteLine();

}

}

Console.WriteLine(Нажмите любую клавишу...);

Console.ReadKey();

}

";

}

static private void Print(string text)

{

Console.WriteLine("Строка:");

Console.WriteLine(text);

Console.WriteLine("Нажмите любую клавишу...");

Console.ReadKey();

}

static private void FindKeyWord(string text)

{

// load dictionary

StreamReader strR = new("dictionary.txt");

string[] dictionary = strR.ReadToEnd().Split('\n');

strR.Close();

int[] count = new int[dictionary.Length];

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

{

count[i] = 0;

}

// split text to words

List<string> words = new();

words.Add(text);

string delimeters = "!?., \n\r";

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

{

List<string> t = new();

for (int j = 0; j < words.Count; j++)

{

t.AddRange(words[j].Split(delimeters[i]));

}

words = t;

}

words = words.Where(x => x != "").Where(x => x != "\r").ToList();

// count

for (int i = 0; i < words.Count; i++)

{

for (int j = 0; j < dictionary.Length; j++)

{

if (words[i] + "\r" == dictionary[j])

{

count[j]++;

}

}

}

// print results

Console.WriteLine("Ключевые слова:");

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

{

if (count[i] > 0)

{

Console.WriteLine(count[i].ToString() + " " + dictionary[i].ToString());

}

}

Console.WriteLine("Нажмите любую клавишу...");

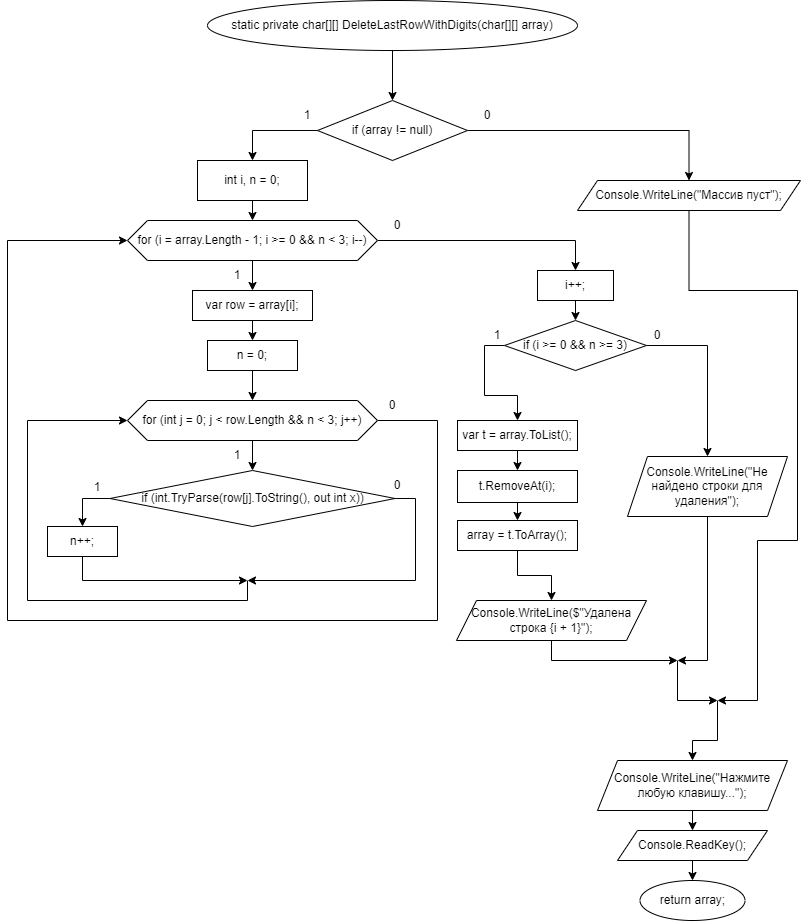
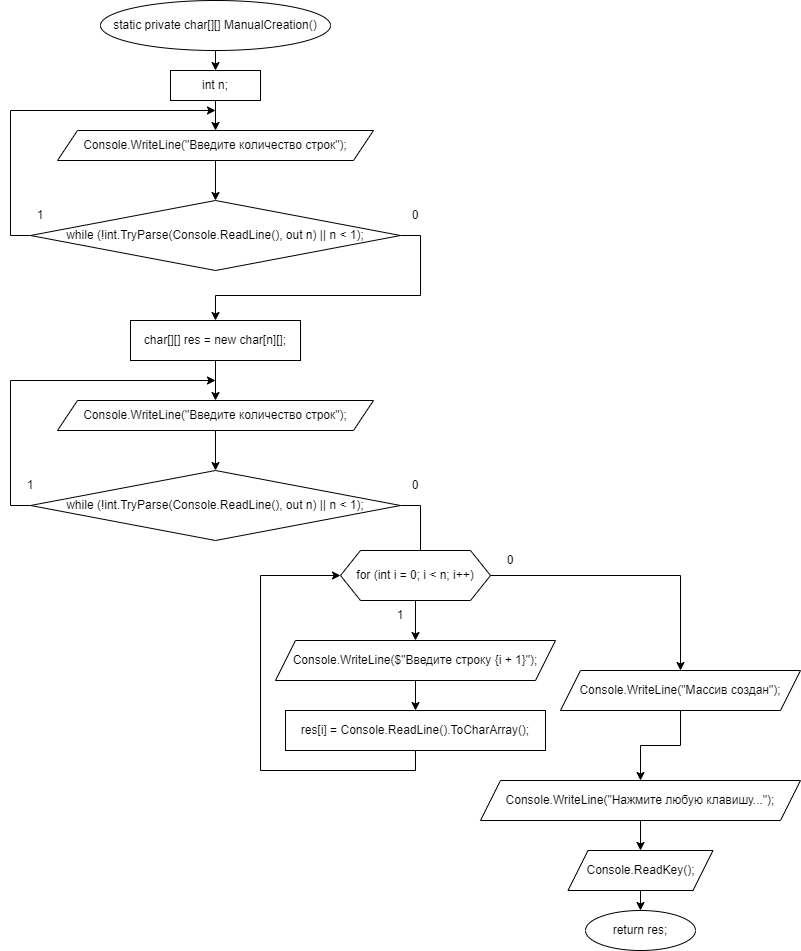
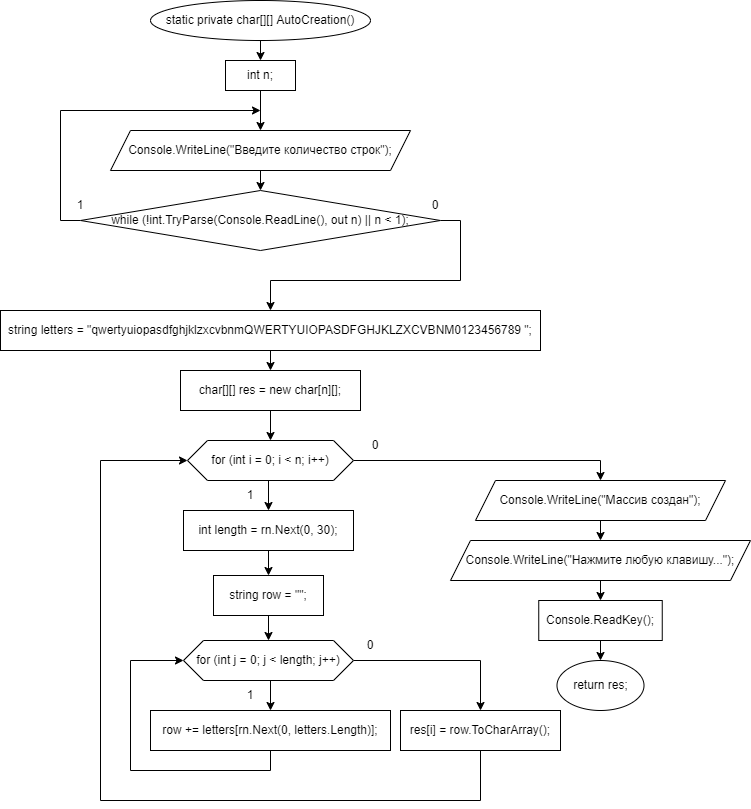
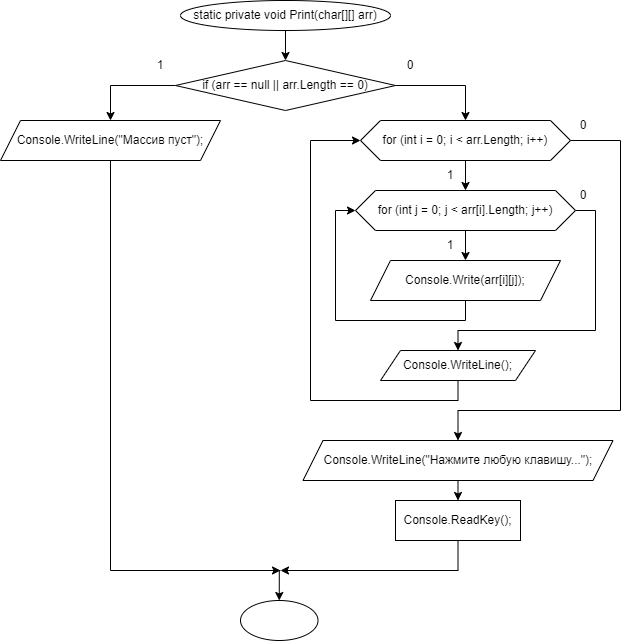
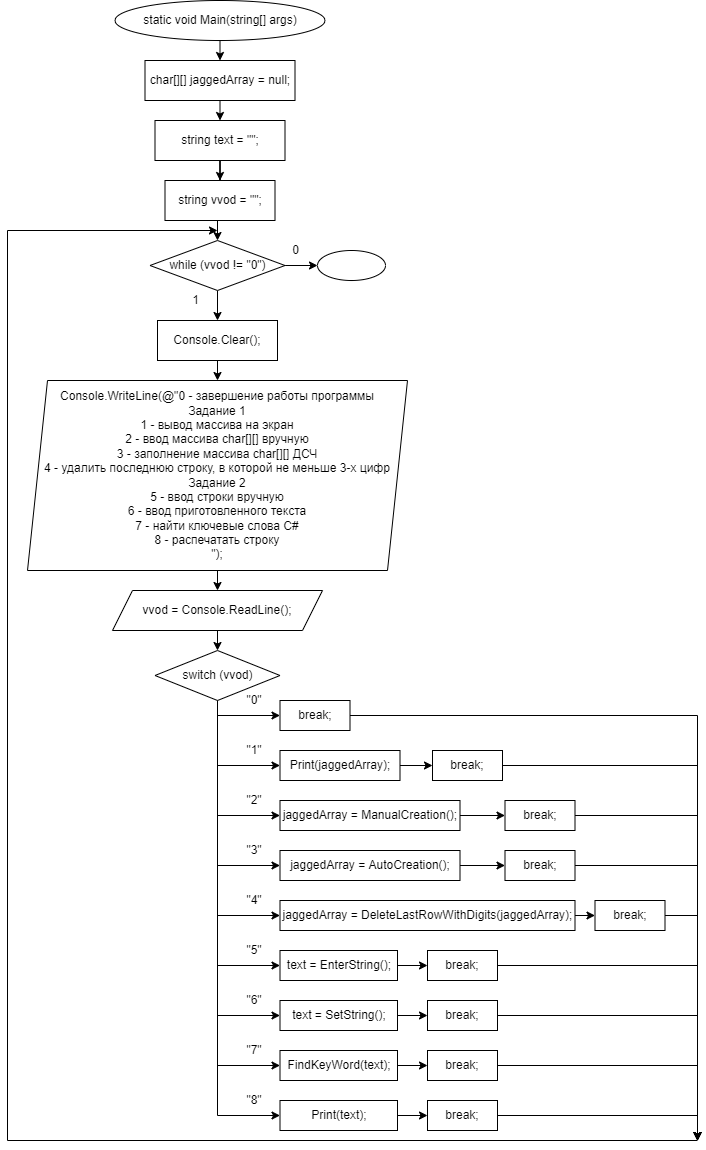
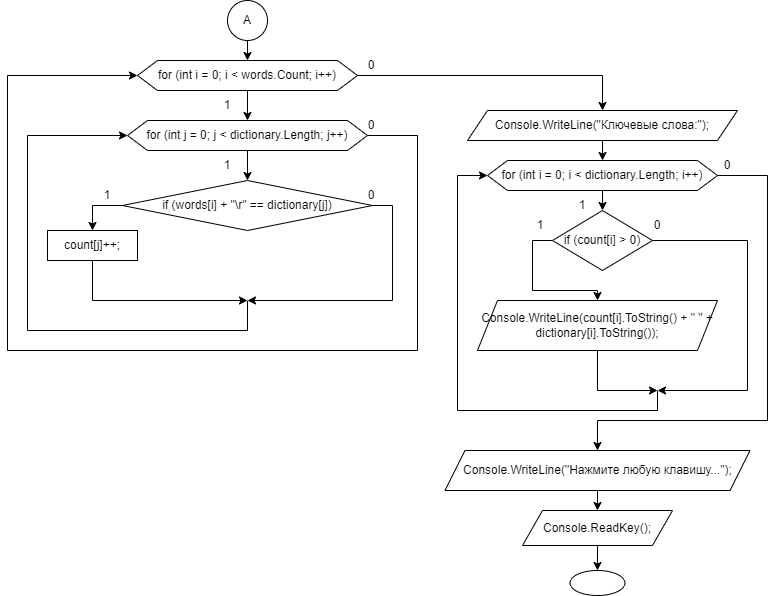
Console.ReadKey();

}

}

}

**Блок-схема**



**Тесты**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Первая часть задания** | | | | |
| **№ теста** | **Входные данные** | **Ожидаемый результат** | **Полученный результат** | **Примечания** |
| **Ввод размера и элементов** | | | | |
| 1 | 5 | 5 строк | 5 строк | ввод размера массива |
| 2 | 0 | Сообщение об ошибке | Ошибка ввода! |
| 3 | -10 | Сообщение об ошибке | Ошибка ввода! |
| 4 | 2,25 | Сообщение об ошибке | Ошибка ввода! |
| 5 | sf | Сообщение об ошибке | Ошибка ввода! |
| 6 | 1 | 1 строка | w |
| 7 | 1 | 1 | 1 | ввод элементов массива с клавиатуры |
| 8 | fsgdnh | fsgdnh | fsgdnh |
| 9 | q | q | q |
| 10 | sdfffd | sdfffd | sdfffd |
| 11 | qwe | qwe | qwe |
| 12 | 1002 | 1002 | 1002 |
| Печать массива | | | | |
| 13 | пустой массив | массив пуст | Массив пуст |  |
| 14 | werrw | werrw | werrw |  |
| Удаление элементов из массива | | | | |
| 15 | fsdfsdafa | fsdfsdafa | fsdfsdafa | Массив не пустой |
| af8979sdaf |  |  |
| 16 | dfads | dfads | dfads | Элемента нет |
| adfad | adfad | adfad |
| 17 | Массив пуст | Массив пуст | Массив пуст | Массив пустой |

**Черный ящик**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Входные данные | Номер теста | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 |
| размер - корректный | + |  |  |  |  | + |
| размер = 0 |  | + |  |  |  |  |
| размер - отрицательный |  |  | + |  |  |  |
| размер - дробный |  |  |  | + |  |  |
| размер - строка |  |  |  |  | + |  |
| элемент - корректный | + |  |  |  |  |  |
| массив пуст |  | + |  |  |  |  |
| массив не пуст | + |  |  |  |  | + |
| есть элементы для удаления | + |  |  |  |  | + |
| элементов для удаления нет |  | + |  |  |  |  |
|  |  |  |  |  |  |  |
| выходные данные | Номер теста | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 |
| удаление | + |  |  |  |  | + |
| сообщение об ошибке |  | + | + | + | + |  |
| нет элементов для удаления |  | + |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Вторая часть задания** | | | | |
| **№ теста** | **Входные данные** | **Ожидаемый результат** | **Полученный результат** | **Примечания** |
| **Ввод количества строк** | | | | |
| 1 | 2 | 2 строки | 2 строки | **ввод количества строк** |
| 2 | 0 | Сообщение об ошибке | Сообщение об ошибке |
| 3 | -10 | Сообщение об ошибке | Сообщение об ошибке |
| 4 | 5,63 | Сообщение об ошибке | Сообщение об ошибке |
| 5 | е | Сообщение об ошибке | Сообщение об ошибке |
| 6 | adgfasg | adgfasg | adgfasg | **ввод строк** |
| **Печать строки** | | | | |
| 7 | Пустая строка | Пустая строка | Пустая строка |  |
| 8 | ggsdg | ggsdg | ggsdg |  |
| **Поиск ключевых слов** | | | | |
| 9 | for if dassd | 2 | 2 |  |
| 10 | afag | 0 | 0 |  |
| 11 | Пустая строка | 0 | 0 |  |

**Черный ящик**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Входные данные | Номер теста | | | | | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| размер - корректный | + |  |  |  |  | + |  |  |  |  |  |
| размер = 0 |  | + |  |  |  |  |  |  |  |  |  |
| размер - отрицательный |  |  | + |  |  |  |  |  |  |  |  |
| размер - дробный |  |  |  | + |  |  |  |  |  |  |  |
| размер - строка |  |  |  |  | + |  |  |  |  |  |  |
| Корректный элемент |  |  |  |  |  | + |  |  |  |  |  |
| пустая строка |  |  |  |  |  |  | + |  |  |  |  |
| не пустая строка |  |  |  |  |  |  |  | + |  |  |  |
| есть ключевые слова |  |  |  |  |  |  |  |  | + |  |  |
| нет ключевых слов |  |  |  |  |  |  |  |  |  | + | + |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| выходные данные | Номер теста | | | | | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| успешный ввод | + |  |  |  |  | + | + | + |  |  |  |
| ошибка |  | + | + | + | + |  |  |  |  |  |  |
| 0 ключевых слов | + |  |  |  |  |  |  |  |  | + | + |
| несколько ключевых слов |  | + |  |  |  |  |  |  | + |  |  |