Министерство образования 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.

Выполнил:

Cтудент группы ИВТ-20-2б

Тедеев Александр Зурабович

Проверила:

Доцент кафедры ИТАС

Викентьева Ольга Леонидовна

Пермь 2021

**Постановка задачи**

Задача 1.

1. Создать динамический массив (одномерный, двумерный, рваный) из элементов заданного типа. При заполнении массива использовать 2 способа (ручной и с помощью ДСЧ).
2. Массив вывести на печать.
3. Выполнить операции с массивом, указанные в варианте, используя, по возможности, методы класса Array.
4. Результаты обработки вывести на печать.



Задача 2.

1. Ввести строку символов с клавиатуры. Строка состоит из слов, разделенных пробелами (пробелов может быть несколько) и знаками препинания (, ;:). В строке может быть несколько предложений, в конце каждого предложения стоит знак препинания (.!?).
2. Выполнить обработку строки в соответствии с вариантом.
3. Результаты обработки вывести на печать.



**Анализ**

1. Для выполнения первого задания необходимо организовать заполнение массива через ДСЧ и ручной ввод:
   1. Функция заполнения одномерного массива символов с помощью ДСЧ. Будет использоваться строка al, в которую будут записаны буквы латинского алфавита. При обработка каждого элемента массива символов будут присваиваться случайные элементы из строки al.

string al = "qwertyuiopasdfghjklzxcvbnmQWERTYUIOPASDFGHJKLZXCVBNM";

arr[i] = al[rnd.Next(0, al.Length - 1)];

* 1. Функция ручного заполнения одномерного массива символов. Будет организован цикл с постусловием для проверки правильности ввода элементов. Ввод символов кириллицы является неверным.

if (temp.Length <= arr.Length && !ContainsCyrillicLetters(temp))

{

arr = temp.ToCharArray();

isFine = true;

}

* + 1. В функции проверяется наличие в введённой строке символов кириллицы, данный шаг был выполнен связи с тем, что функция для генерации случайным образом вставляет только латинские символы. После проверки строка преобразуется в массив символов.
    2. Функция ContainsCyrillicLetters() возвращает результат проверки на наличие символов кириллицы в введённой строке. Для чего в данной функции будет использоваться строка al, содержащая только символы кириллицы.

string al="йцукенгшщзхъёфывапролджэячсмитьбюЁЙЦУКЕНГШЩЗХЪФЫВАПРОЛДЖЭЯЧСМИТЬБЮ";

* 1. Удаление последней гласной буквы производится в функции RemoveLastVowelFrom(). Необходимо организовать цикл, который будет обрабатывать массив символов с конца.

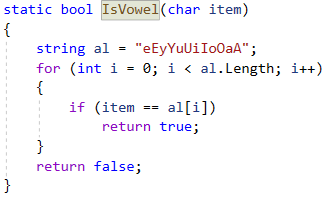
for (int i = arr.Length - 1; i >= 0 && !containsVowels; i--)

{

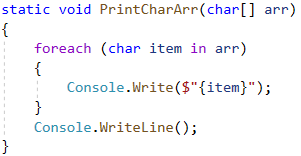
...

}

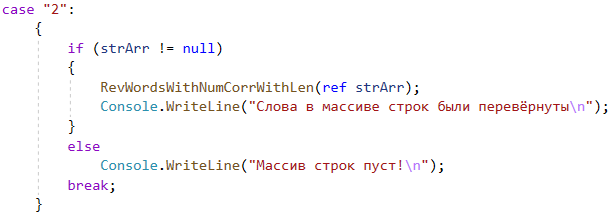
* 1. Вводится временный массив символов (tmpArr) размерностью на 1 элемент меньшей, переменная логического типа containsVowels. В цикле происходит посимвольное сравнение символов по признаку гласности.
  2. Функция IsVowel() возвращает результат сравнения.



* 1. Если было обнаружено хотя бы одно вхождение гласной буквы, то используются нестатические методы класса Array для копирования данных из временного массива tmpArr в основной – arr.
  2. Печать результата обеспечивает функция PrintCharArr()



1. Для выполнения второго задания возможны два способа ввода: ручной и использование заготовленной строки. Требования к строке: не менее одного предложения, наличия знаков разделителей, возможны несколько пробелов.
   1. Использование заготовленной строки. В функции Menu() при вводе 2 происходит присваивание переменной строкового типа значения.
   2. Ручной ввод. В функции Menu() при вводе 1 происходит вызов метода Readline() класса Console для ввода строки.
   3. Функция Menu(). При вводе 3, происходит простой выход из функции для дальнейшей работы. При нажатии 4 происходит выход из программы, а затем и полное завершение работы программы.
   4. После завершения функции Menu() производится вызов функции Task(), которая необходима для разбивание строки на массив строк (по словам).
      1. При вводе 1 производится разбиение большой строки на массив строк по словам в функции SplitMyString(), которая возвращает массив строк.
      2. При вводе 2 вызывается функция RevWordsWithNumCorrWithLen()



* + - 1. Функция RevWordsWithNumCorrWithLen() разбита на 3 этапа: Создание рванного массива символов; переворачивание слова, номер которого совпадает с его длиной; перезапись в исходный массив строк данных из равного массива символов.
      2. Создание рванного массива символов из массива строк. Используется метод класса string ToCharArray()

jagArr[i] = strArr[i].ToCharArray();

* + - 1. Переворачивание слова, номер которого совпадает с его длиной.

if (jagArr[i].Length == (i + 1) && jagArr.Length != 1)

{

Array.Reverse(jagArr[i]);

}

* + - 1. Перезапись в исходный массив строк данных из равного массива символов. Используется массив строк strArr, в цикле при обработке строки strArr[i] из рваного массива символов (jagArr) происходит присваивание строки из рванного массива символов (jagArr[i]) в i-ый элемент массива строк strArr[i] через конструктор класса string.

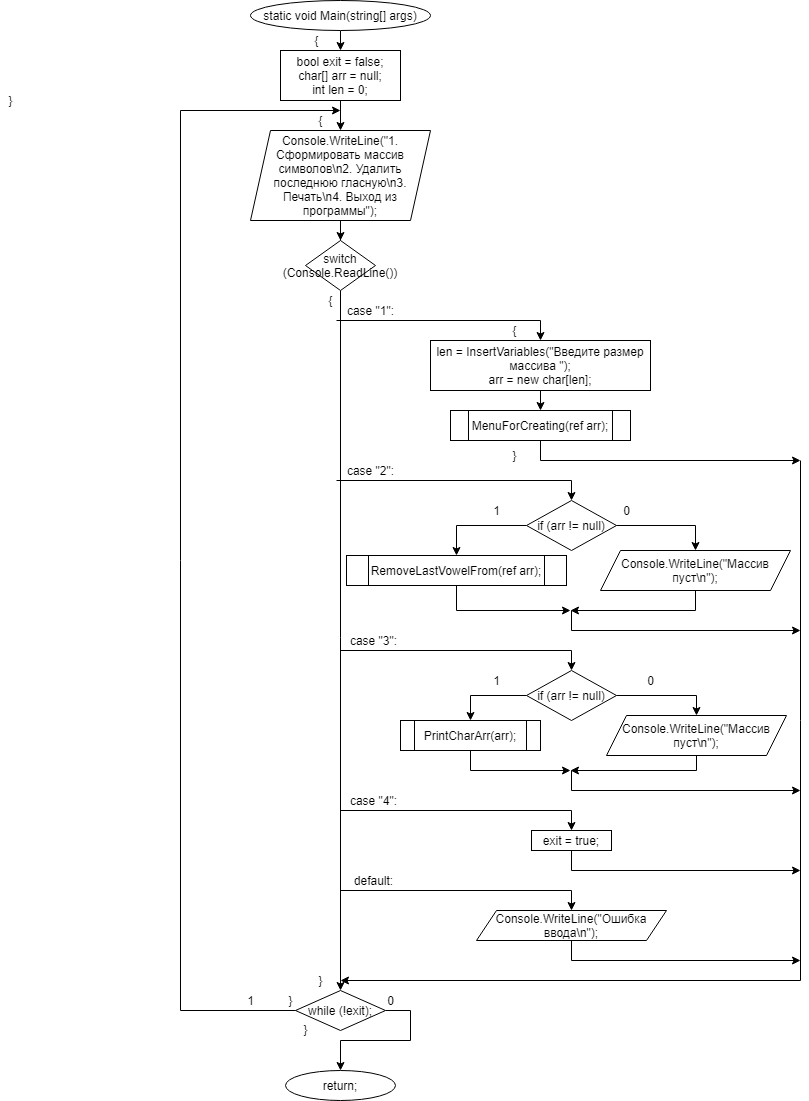
string[] strArr = new string[jagArr.Length];

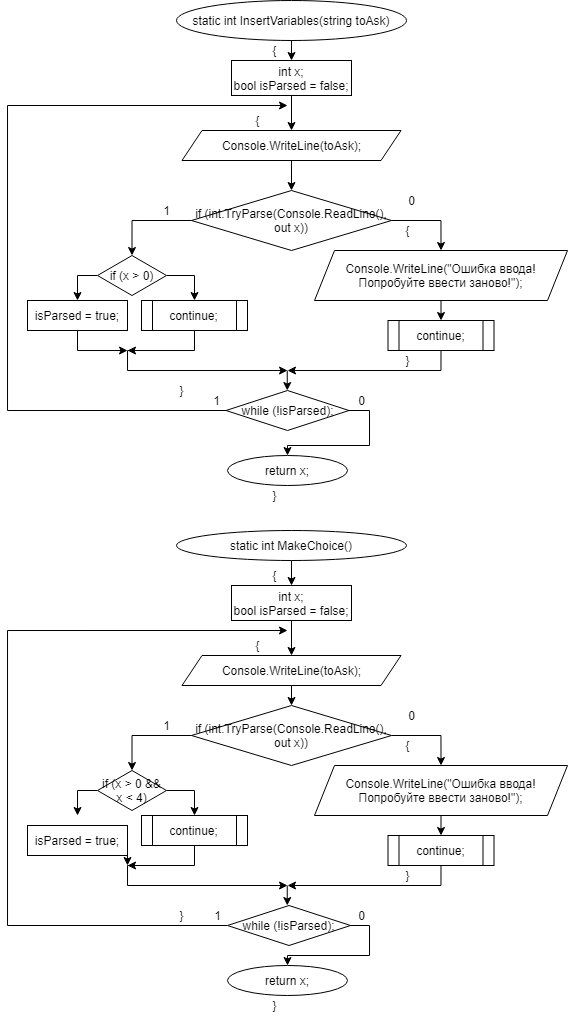
string tmp = new string(jagArr[i]);

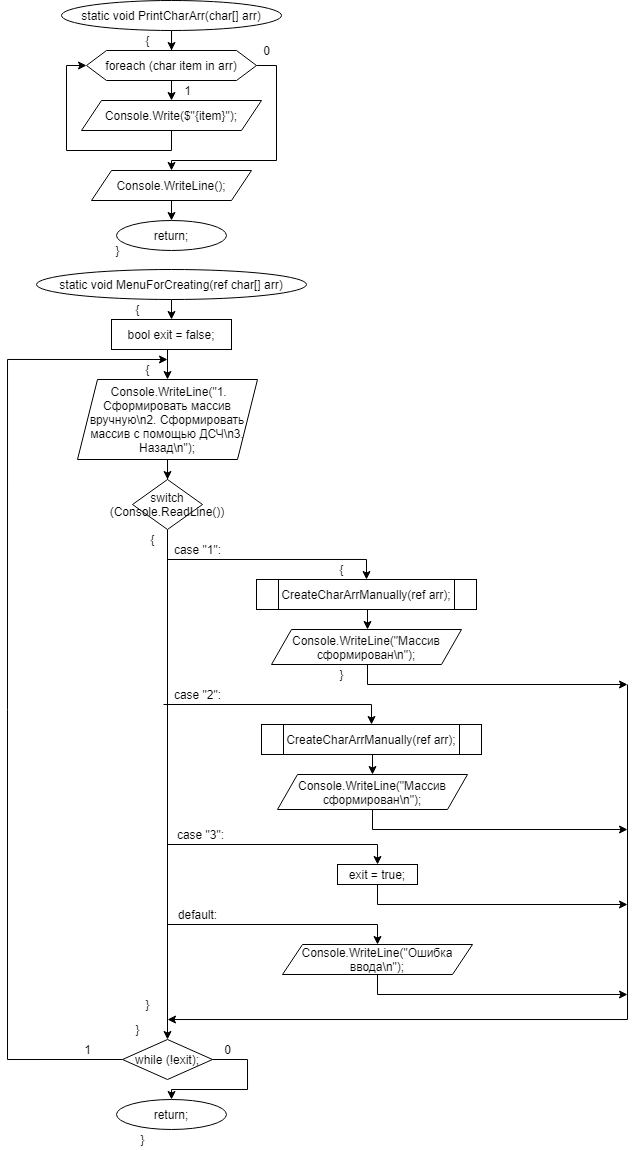
strArr[i] = tmp;

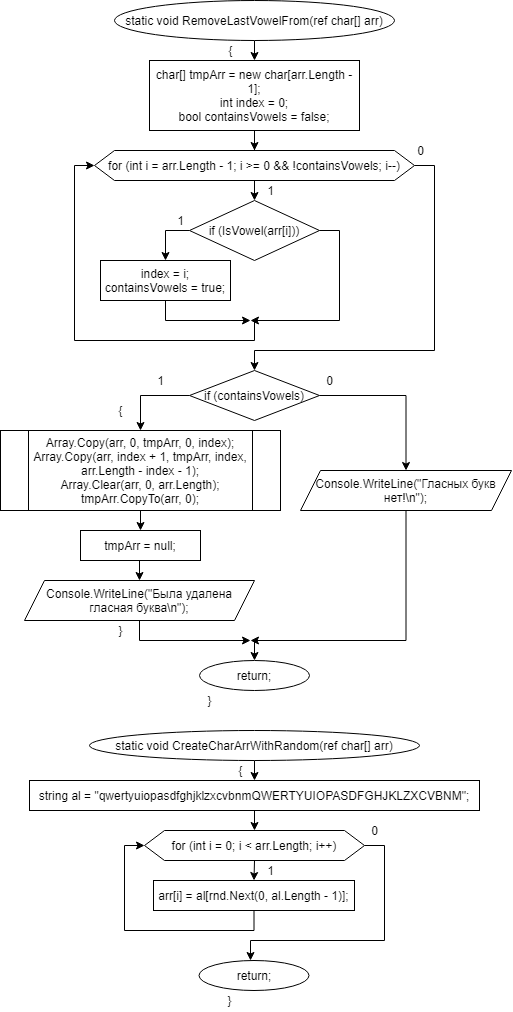
**Блок-схема**

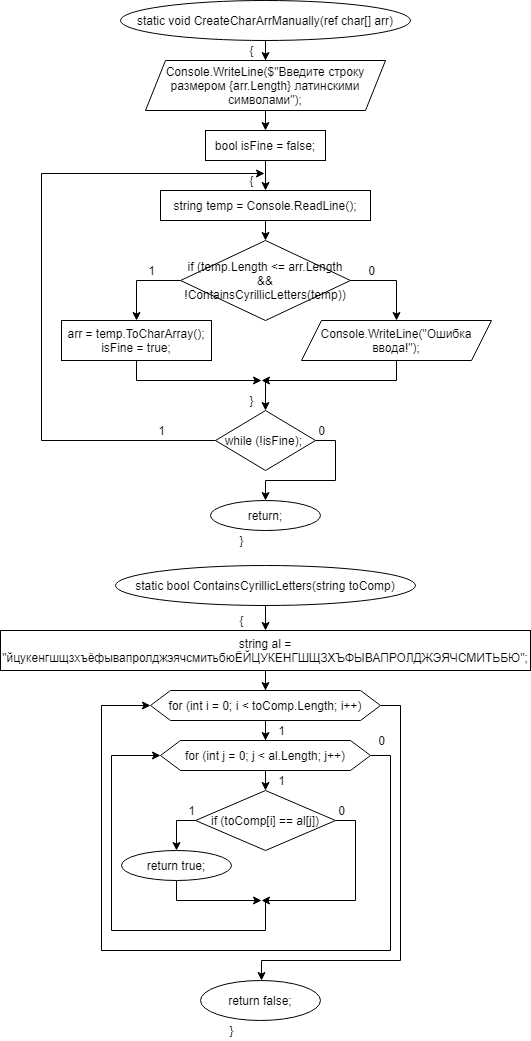
**Задача 1**

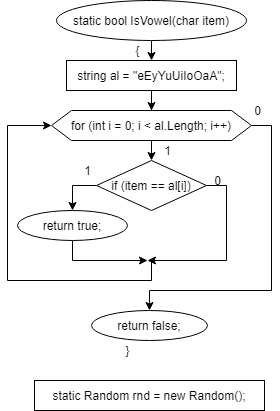




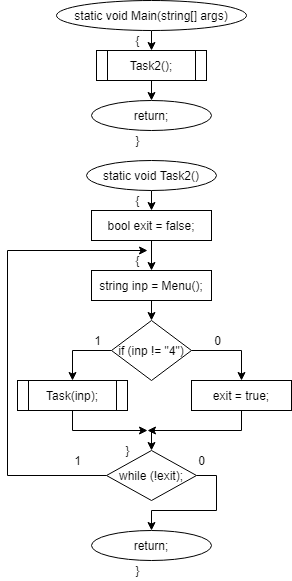


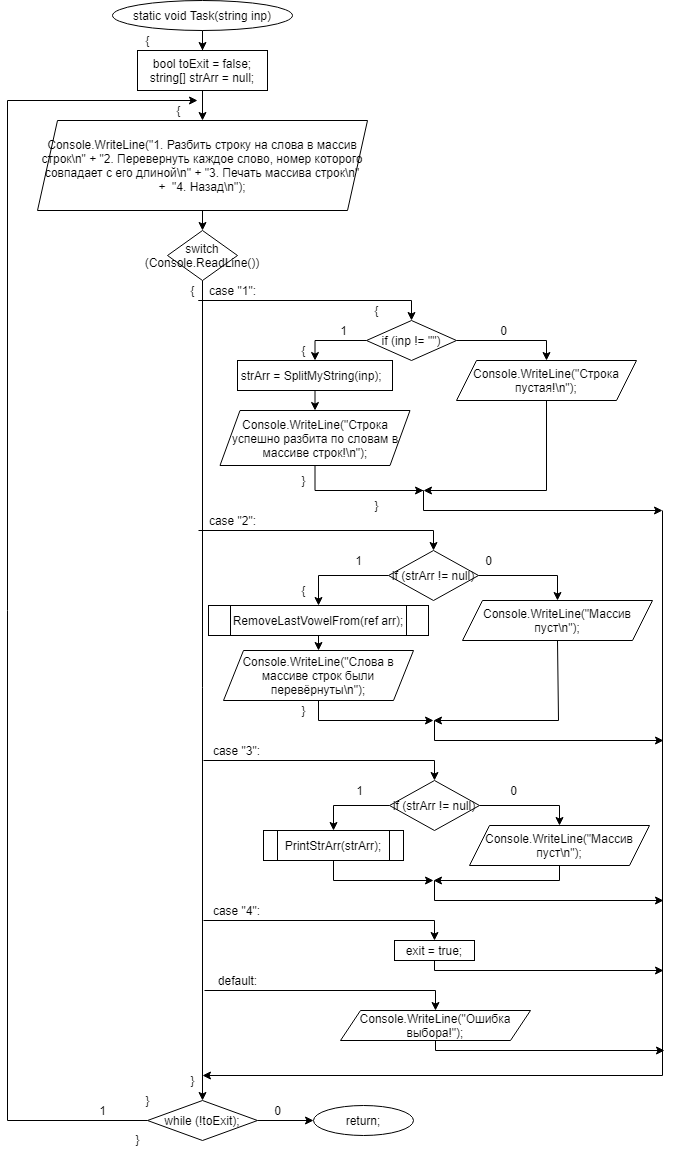


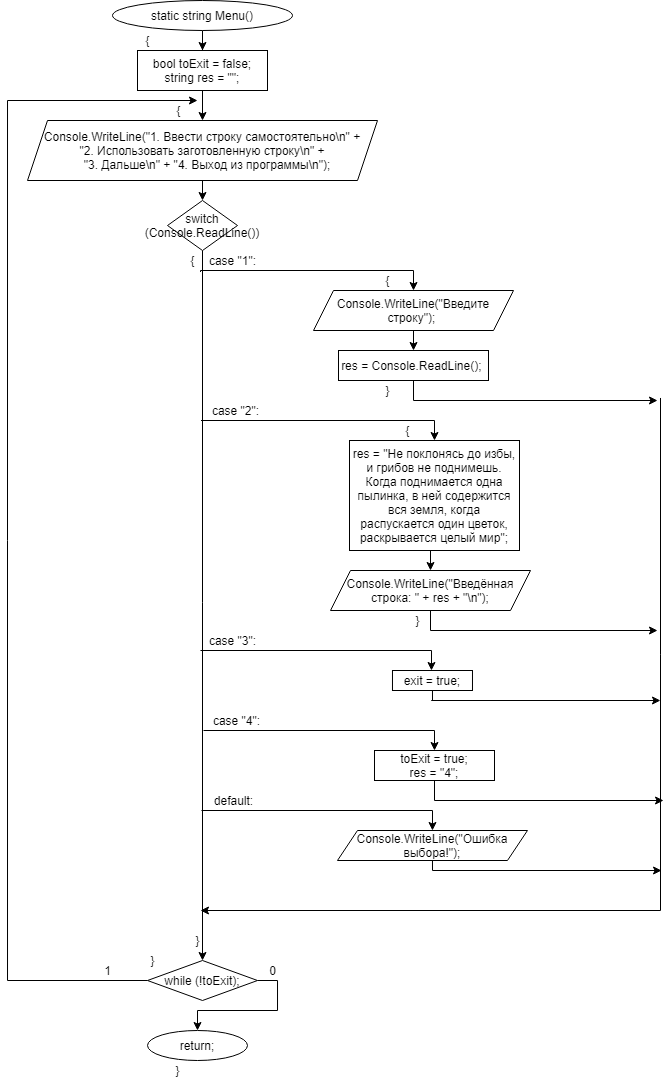


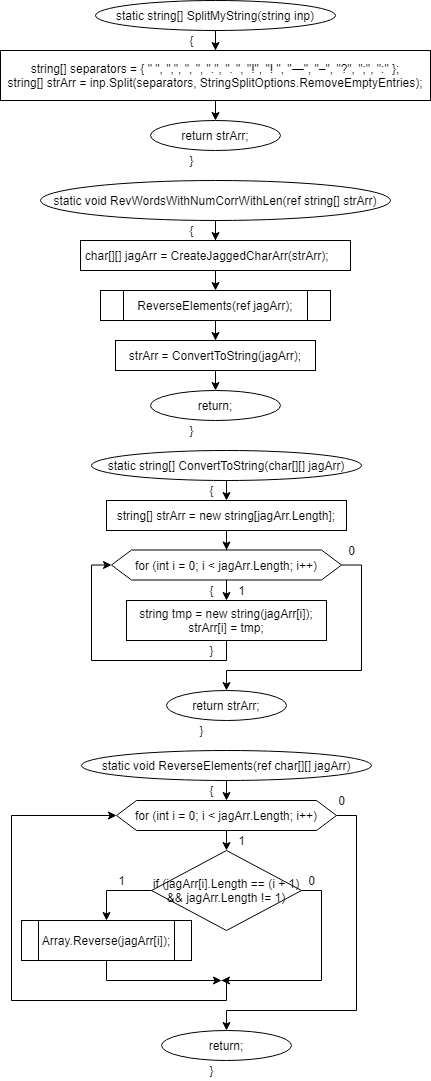


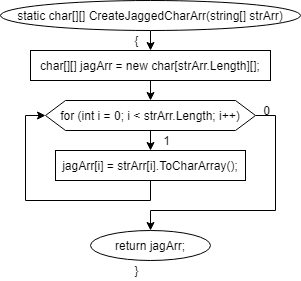
**Задача 2**





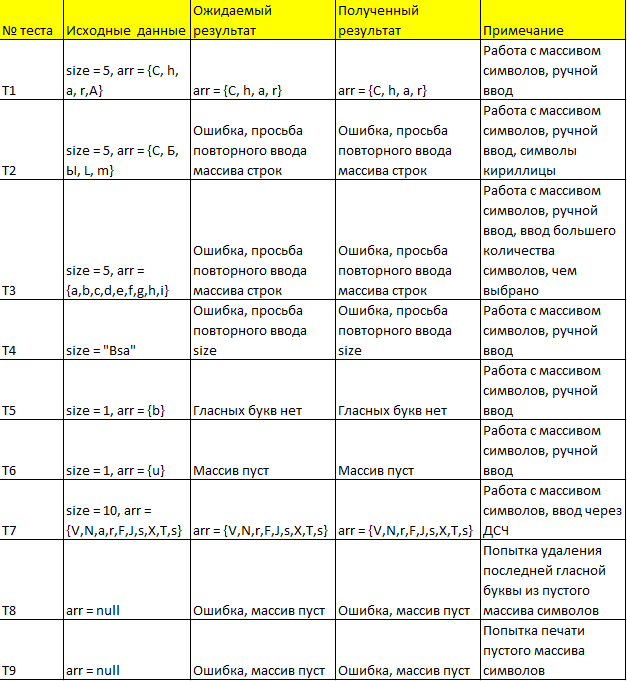


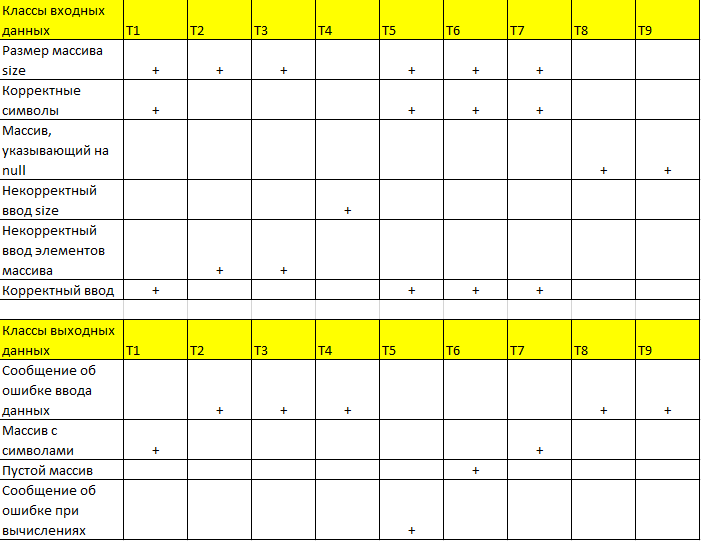




**Тесты**

**Задача 1**





**Задача 2**

