

Итоговая проверочная работа.

Данная работа необходима для проверки ваших знаний и навыков по итогу прохождения первого блока обучения на программе разработчик. Мы должны убедиться, что базовое знакомство с it прошло успешно.

Задача алгоритмически не самая сложная, однако для полноценного выполнения проверочной работы необходимо:

1. Создать репозиторий на GitHub
2. Нарисовать блок-схему алгоритма (можно обойтись блок-схемой основной содержательной части, если вы выделяете ее в отдельный метод)
3. Снабдить репозиторий оформленным текстовым описанием решения (файл README.md)
4. Написать программу, решающую поставленную задачу
5. Использовать контроль версий в работе над этим небольшим проектом (не должно быть так, что все залито одним коммитом. Как минимум, этапы 2, 3 и 4 должны быть расположены в разных коммитах)

Задача: Написать программу, которая из имеющегося массива строк формирует массив строк, длина которых меньше, либо равна 3 символа. Первоначальный массив можно ввести с клавиатуры, либо задать на старте выполнения алгоритма. При решении не рекомендуется пользоваться коллекциями, лучше обойтись исключительно массивами.

Примеры:

[“hello”, “2”, “world”, “:-)"] -> [“2”, “:-)"]

[“1234”, “1567”, “-2”, “computer science”] -> [“-2”]

[“Russia”, “Denmark”, “Kazan”] -> []

Код лежит в папке "Program"

Данная программа находит элементы массива, в которых не более трех символов.

1. Создаем метод WordSearch. Он принимает на вход два аргумента. Массив array и массив result. Пробегаясь циклом for по массиву array, с помощью условия if, проверяем длину элемента под индексом i. Если элемент удовлетворяет условию (в нем меньше 3-х символов), то отправляем его в массив result.
2. Создаем метод PrintArray, который будет принимать на вход полученный массив result, который мы получили после работы метода WordSearch. В методе используем цикл for, чтобы пробежаться по массиву и с каждой итерацией будем печатать все элементы по порядку. После того, как цикл пройдет по всему массиву, отправляем полученный массив в консоль.
3. В переменную symbols можем ввести то количество символов, которое будет считаться при работе условия if в первом методе.

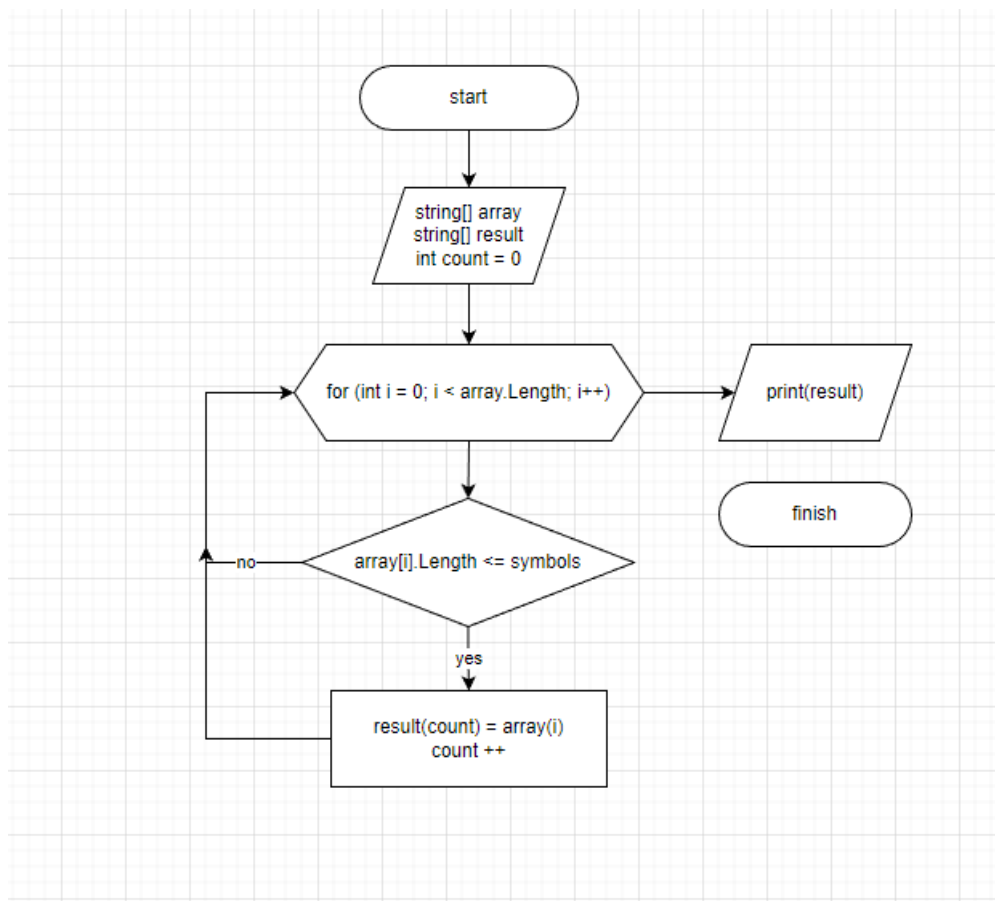


Рис 1. Блок-схема

```
C# Program.cs X
Program > C# Program.cs
1 Console.Clear();
2
3 string[] array = new string[] { "Russia", ":-)", "Denmark", "123", "4321" };
4 string[] result = new string[array.Length];
5 int symbols = 3;
6 int count = 0;
7
8 WordSearch(array, result);
9 PrintArray(result);
10
11 void WordSearch(string[] array, string[] result)
12 {
13     for (int i = 0; i < array.Length; i++)
14     {
15         if (array[i].Length <= symbols)
16         {
17             result[count] = array[i];
18         }
19         count++;
20     }
21 }
22
23 void PrintArray(string[] array)
24 {
25     for (int i = 0; i < array.Length; i++)
26     {
27         Console.Write($"{array[i]} ");
28     }
29     Console.WriteLine();
30 }
```

Рис 2. Программа