



МИНОБРНАУКИ РОССИИ

**Федеральное государственное бюджетное образовательное учреждение
высшего образования**

«МИРЭА - Российский технологический университет»

РТУ МИРЭА

**Институт перспективных технологий и индустриального программирования
(ИПТИП)**

ОТЧЕТ ПО ПРАКТИЧЕСКИМ РАБОТАМ

по дисциплине

«Технологии создания программного обеспечения»

Практическая работа №1

Выполнил студент группы ЭФМО-02-23

Мурадов Н.Н.

Москва 2023

СОДЕРЖАНИЕ

Задача.....	3
Решение.....	4
Результаты	8

Задача

1. Дана целочисленная последовательность, содержащая как положительные, так и отрицательные числа. Вывести ее первый положительный элемент и последний отрицательный элемент.
2. Даны цифра D (однозначное целое число) и целочисленная последовательность A . Вывести первый положительный элемент последовательности A , оканчивающийся цифрой D . Если требуемых элементов в последовательности A нет, то вывести 0.
3. Даны целое число $L (> 0)$ и строковая последовательность A . Вывести последнюю строку из A , начинающуюся с цифры и имеющую длину L . Если требуемых строк в последовательности A нет, то вывести строку «Not found».

Указание. Для обработки ситуации, связанной с отсутствием требуемых строк, использовать операцию ??.

4. Даны символ C и строковая последовательность A . Если A содержит единственный элемент, оканчивающийся символом C , то вывести этот элемент; если требуемых строк в A нет, то вывести пустую строку; если требуемых строк больше одной, то вывести строку «Error».

5. Даны символ C и строковая последовательность A . Найти количество элементов A , которые содержат более одного символа и при этом начинаются и оканчиваются символом C .

6. Дана целочисленная последовательность. Найти количество ее положительных двузначных элементов, а также их среднее арифметическое (как вещественное число). Если требуемые элементы отсутствуют, то дважды вывести 0 (первый раз как целое, второй - как вещественное).

7. Даны целое число $L (> 0)$ и строковая последовательность A . Строки последовательности A содержат только заглавные буквы латинского алфавита. Среди всех строк из A , имеющих длину L , найти наибольшую (в смысле лексикографического порядка). Вывести эту строку или пустую строку, если последовательность не содержит строк длины L .

8. Дана строковая последовательность. Найти сумму длин всех строк, входящих в данную последовательность.

9. Даны целое число D и целочисленная последовательность A . Начиная с первого элемента A , большего D , извлечь из A все нечетные положительные числа, поменяв порядок извлеченных чисел на обратный.

10. Дано целое число $K (> 0)$ и строковая последовательность A . Из элементов A , предшествующих элементу с порядковым номером K , извлечь те строки, которые имеют

нечетную длину и начинаются с заглавной латинской буквы, изменив порядок следования извлеченных строк на обратный.

11. Даны целые числа D и K ($K > 0$) и целочисленная последовательность A . Найти теоретико-множественное объединение двух фрагментов A : первый содержит все элементы до первого элемента, большего D (не включая его), а второй - все элементы, начиная с элемента с порядковым номером K . Полученную последовательность (не содержащую одинаковых элементов) отсортировать по убыванию.

12. Даны целое число K (> 0) и целочисленная последовательность A . Найти теоретико-множественную разность двух фрагментов A : первый содержит все четные числа, а второй - все числа с порядковыми номерами, большими K . В полученной последовательности (не содержащей одинаковых элементов) поменять порядок элементов на обратный.

13. Даны целое число K (> 0) и последовательность непустых строк A . Строки последовательности содержат только цифры и заглавные буквы латинского алфавита. Найти теоретико-множественное пересечение двух фрагментов A : первый содержит $3K$ начальных элементов, а второй - все элементы, расположенные после последнего элемента, оканчивающегося цифрой. Полученную последовательность (не содержащую одинаковых элементов) отсортировать по возрастанию длин строк, а строки одинаковой длины - в лексикографическом порядке по возрастанию.

14. Дана строковая последовательность A . Строки последовательности содержат только заглавные буквы латинского алфавита. Получить новую последовательность строк, элементы которой определяются по соответствующим элементам A следующим образом: пустые строки в новую последовательность не включаются, а к непустым приписывается порядковый номер данной строки в исходной последовательности (например, если пятый элемент A имеет вид «ABC», то в полученной последовательности он будет иметь вид «ABC5»). При нумерации должны учитываться и пустые строки последовательности A . Отсортировать полученную последовательность в лексикографическом порядке по возрастанию.

Решение

Листинг кода с 1 по 14 задание.

```
using System;
using System.Linq;
using System.Collections.Generic;
using System.Runtime.Versioning;

public class HelloWorld {
```

```

public static void Main() {
    Console.WriteLine("Hello World");
    num1();
    num2();
    num3();
    num4();
    num5();
    num6();
    num7();
    num8();
    num9();
    num10();
    num11();
    num12();
    num13();
    num14();
}

public static void num1() {
    Console.WriteLine("num1");
    int[] ints = { -2, 1, 4, 5, -5, 6, -1 };
    Console.WriteLine(ints.First(n => n > 0));
    Console.WriteLine(ints.Last(n => n < 0));
}

public static void num2() {
    Console.WriteLine("num2");
    int D = 7;
    int[] A = { -204, 178, 4345, 5435, -565, 676, -189 };
    Console.WriteLine(A.FirstOrDefault(n => n > 0 &&
n.ToString().EndsWith(D.ToString())));
}

public static void num3() {
    Console.WriteLine("num3");
    int L = 4, x = 0;
    string[] A = { "AXS1", "2TER", "B2GF", "LK3IK1" };
    Console.WriteLine(A.LastOrDefault(n => n.Length == L &&
int.TryParse(n.Substring(0, 1), out x)) ?? "Not Found");
}

public static void num4() {
    Console.WriteLine("num4");
    char C = '1';
    string[] A = { "AXS1", "2TER", "B2GF", "LK3IK1" };
    var rez = A.Where(n => n.EndsWith(C)).DefaultIfEmpty("");
    Console.WriteLine(rez.Count() > 1 ? "Error" : rez.ElementAt(0));
}

public static void num5() {
    Console.WriteLine("num5");
    char C = '1';

```

```

        string[] A = { "AXS1", "2TER", "B2GF", "LK3IK1", "1K3IK1", "1K4IK1" };
        Console.WriteLine(A.Count(n => n.Length > 1 && n.StartsWith(C) &&
n.EndsWith(C)));
    }

    public static void num6() {
        Console.WriteLine("num6");
        int[] A = { 0, 1, 22, 44, -56, 4 };
        var rez = A.Where(n => n > 9 && n < 100);
        int rezC = rez.Count();
        Console.WriteLine(rezC);
        Console.WriteLine(rezC > 1 ? rez.Average() : 0.0);
    }

    public static void num7() {
        Console.WriteLine("num7");
        int L = 5;
        string[] A = { "AXS", "TER", "BGF", "AAA", "XXX", "LKIK", "AKIK" };
        Console.WriteLine(A.Where(n => n.Length == L).OrderBy(n => n).LastOrDefault()
?? "");
    }

    public static void num8() {
        Console.WriteLine("num8");
        string[] A = { "AXS", "TER", "BGF", "AAA", "XXX", "LKIK", "AKIK" };
        Console.WriteLine(A.Sum(n => n.Length));
    }

    public static void num9() {
        Console.WriteLine("num9");
        int D = 7;
        bool b = false;
        int[] A = { -204, 3, 8, 7, 178, 4345, 5435, -565, 676, -189 };
        var rez = A.Where(n => (n > D || b) && (b = true));
        foreach (var n in rez.Where(n => n > 0 && n % 2 != 0).Reverse())
            Console.WriteLine(n);
    }

    public static void num10() {
        Console.WriteLine("num10");
        int K = 5;
        string[] A = { "aXs", "AXs", "AXS", "TER", "BGF", "AAA", "XXX", "LKIK",
"AKIK" };
        foreach (var el in A.Take(K).Where(n => n.Length % 2 != 0 && n[0] >= 'A' &&
n[0] <= 'Z').Reverse())
            Console.WriteLine(el);
    }

    public static void num11() {
        Console.WriteLine("num11");
        int K = 5, D = 6, L = K - 1;
        int[] A = { -204, 3, 8, 7, 178, 4345, 5435, -565, 676, -189 };

```

```

        foreach (var el in A.TakeWhile(n => n < D).Union(A.Where((n, i) => i >
L)).OrderByDescending(n => n))
            Console.WriteLine(el);
    }

    public static void num12() {
        Console.WriteLine("num12");
        int K = 5;
        int[] A = { -204, 3, 8, 7, 178, 4345, 5435, -565, 676, -189 };
        foreach (var el in A.Where(n => n % 2 == 0).Except(A.Where((n, i) => i >
K)).Distinct().Reverse())
            Console.WriteLine(el);
    }

    public static void num13() {
        Console.WriteLine("num13:");
        int K = 5;
        string[] A = { "AXS1", "2TER", "B2GF", "LK3IK1", "1K3IK1", "1K4IK1",
"B3GF", "B4GF" };
        var Ai = A.Select((s,i) => new { s, i }).DefaultIfEmpty(new { s = "", i = 0 });
        foreach (var el in A.Take(K).Except(A.Skip(Ai.Last(n =>
char.IsDigit(n.s[n.s.Length-1])).i + 1)).OrderBy(n => n.Length).ThenBy(n => n))
            Console.WriteLine(el);
        Console.WriteLine("Конец num13\n");
    }

    public static void num14() {
        Console.WriteLine("num14:");
        string[] A = { "aXs", "AXs", "AXS", "", "TER", "BGF", "AAA", "XXX", "",
"LKIK", "AKIK" };
        foreach (var el in A.Select((n, i) => n+i).Where((n, i) =>
!char.IsDigit(n[0])).OrderBy(n => n))
            Console.WriteLine(el);
        Console.WriteLine("Конец num14\n");
    }
}

```

Результаты

Результаты выполнения с 1 по 14 задачи представлены на Рисунке 1.

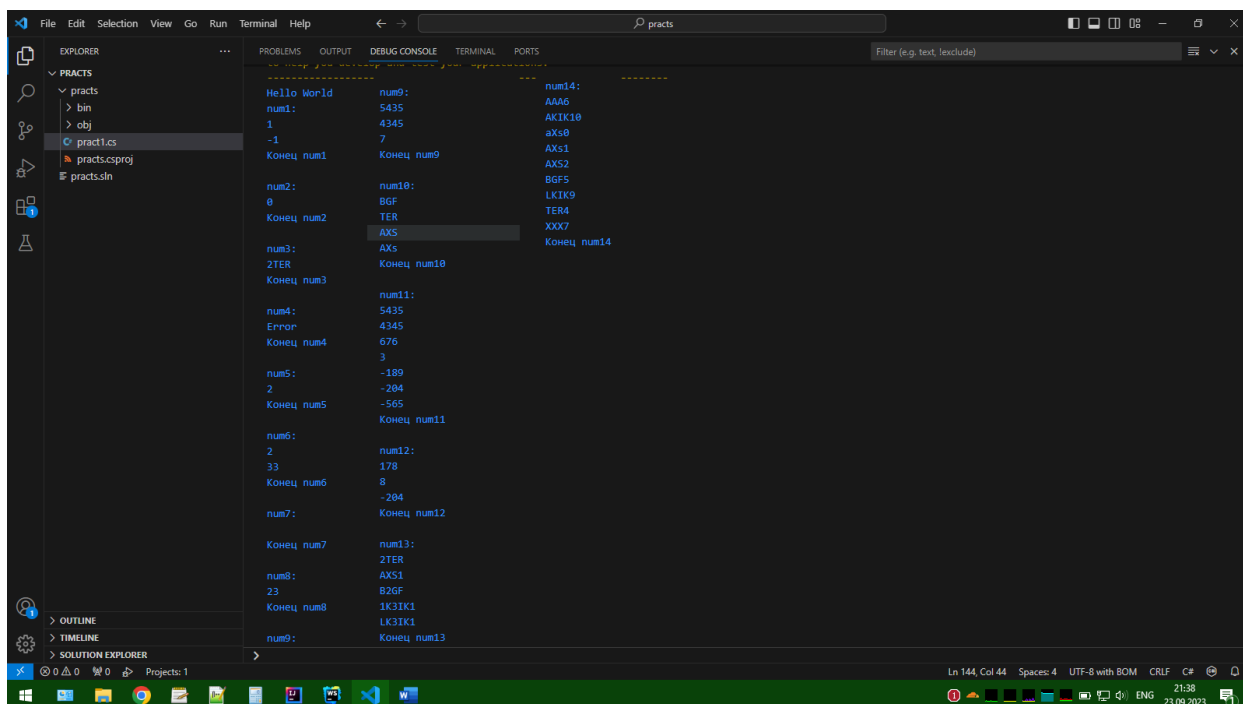


Рисунок 1 – Результаты компиляции с 1 по 14 заданий