Санкт-Петербургский Национальный Исследовательский Университет Информационных Технологий, Механики и Оптики

Факультет Прикладной Информатики

**Лабораторная работа №1**

Выполнил:

Зенин Д.Д.

Проверил Иванов С. Е.

Санкт-Петербург,

2025

Оглавление

[ВВЕДЕНИЕ 3](#_Toc192266537)

[1. Задание 1 4](#_Toc192266538)

[2. Задание 2 6](#_Toc192266541)

[ЗАКЛЮЧЕНИЕ 9](#_Toc192266543)

# ВВЕДЕНИЕ

Цель лабораторной работы №1 состояла в изучении алгоритмов хеширования, реализовать алгоритм Рабина-Карпа поиска подстроки в строке с применением хеширования.

**ЗАДАНИЕ НА ЛАБОРАТОРНУЮ РАБОТУ**

1.Реализовать поиск одинаковых строк. Дан список строк S[1..n], каждая длиной не более m символов. Требуется найти все повторяющиеся строки и разделить их на группы, чтобы в каждой группе были только одинаковые строки.

2. Реализовать алгоритм Рабина-Карпа поиска подстроки в строке за O (n).

# Задание 1

# В этом задании создам программу на языке С#, которая будет реализовывать алгоритм поиска одинаковых строк. Необходимо найти все повторяющиеся строки и разделить их на группы, чтобы в каждой группе были только одинаковые строки. Программа будет считывать строки

# 

Рисунок 1 – Программа на языке С#, считывающая строки

**Полиномиальное хеширование** строки: Хеш каждой строки вычисляется по формуле:

h=(s\_0⋅p^0+s\_1⋅p^1+s\_2⋅p^2+⋯+s\_(n−1)⋅p^(n−1)) % mod

где:

* s\_i — код символа строки,
* p — некоторое фиксированное число
* mod— модуль

Ниже представлена функция для вычисления степеней p и хеширования строки:

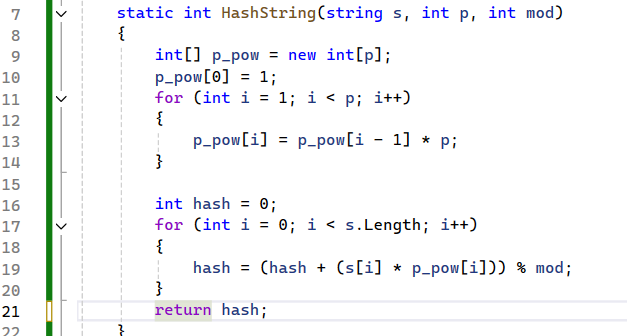


Рисунок 2 – Функция для вычисления степеней p и хэширования строки

Далее создаем словарь, где каждый хэш будет соответствовать своей строке. Потом проходимся по значениям словаря и если количество значений больше единицы(то есть нашлись одинаковые подстроки), то мы их выводим

Ниже представлен алгоритм для сортировки по хешам и вывод ответа:



Рисунок 4 – Листинг для сортировки по хешам и вывод ответа:

Ниже представлен список результатов: хеши и список групп:

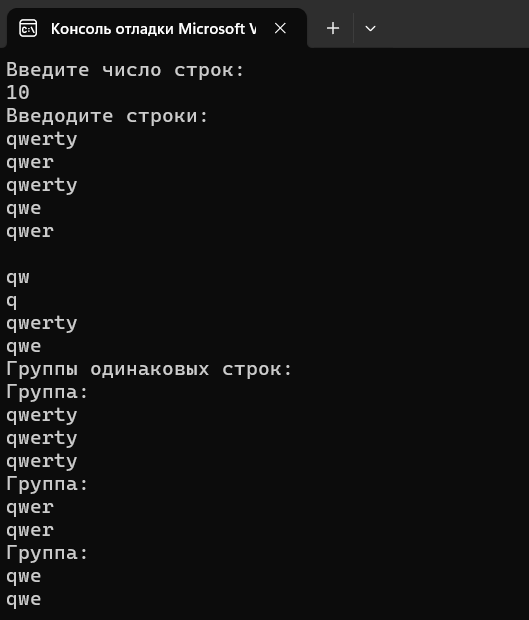


Рисунок 5 – Результат выполнения программы

# Задание 2

# В этом задании необходимо реализовать алгоритм Рабина-Карпа поиска подстроки в строке за O(N). Дана строка S и текст T, состоящие из маленьких латинских букв. Требуется найти все вхождения строки S в текст T за время O (|S| + |T|). Сначала также считаем степени P. Затем хеши подстрок текста и хеш строки. Ниже представлен листинг для вычисления степеней и хешей:

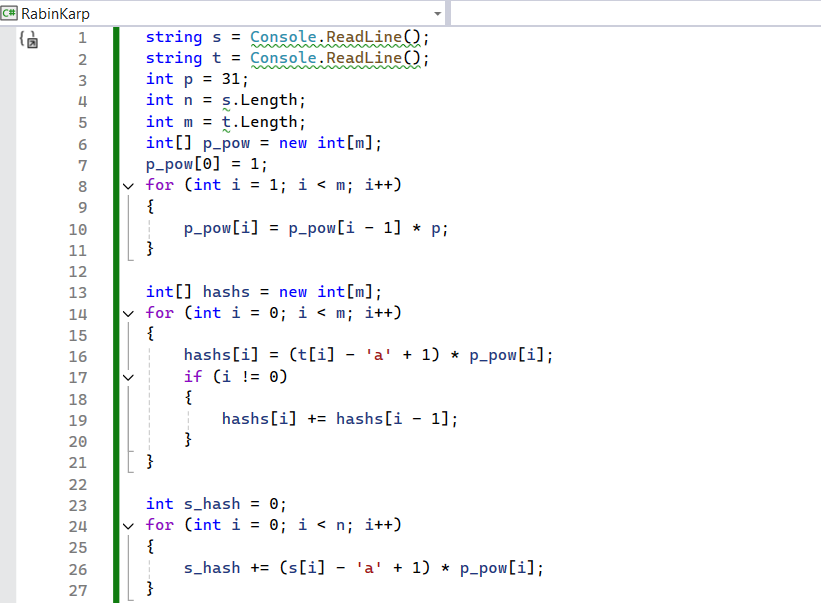


Рисунок 7 – Листинг для вычисления степеней и хешей

Используя формулу hash(s0..R) = hash(s0..L-1) + p Lhash(sL..R), найдем индексы, где в тексте появляется нужная строка. Ниже представлен листинг для вывода ответа:

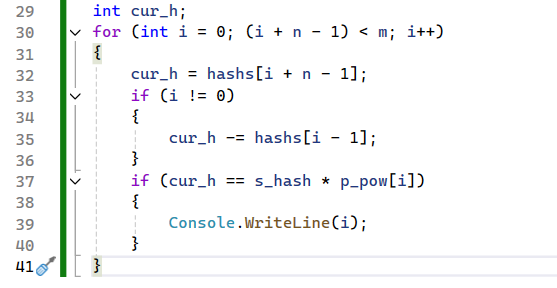


Рисунок 8 – Листинг для вывода ответа

Запустим программу:

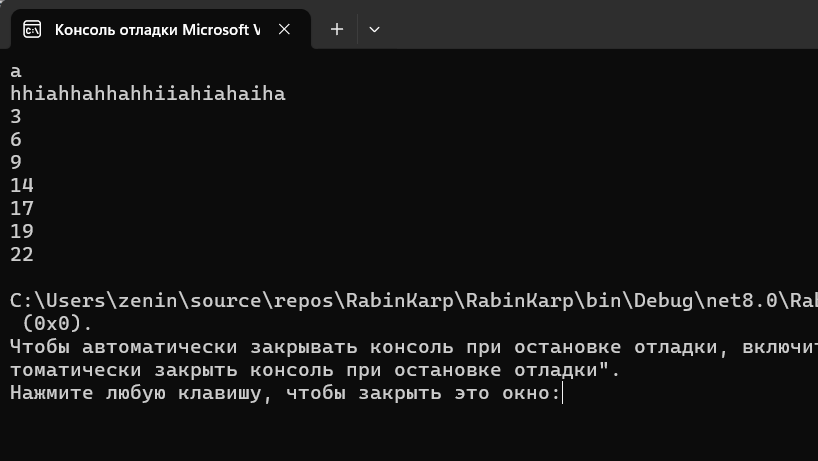


Рисунок 9 – Результат выполнения программы

# 

# ЗАКЛЮЧЕНИЕ

В ходе выполнения работы изучены и реализованы средствами ООП C# алгоритмы хеширования.