МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ

по лабораторной работе №4

по дисциплине «Построение и анализ алгоритмов»

Тема: КМП

Студент гр. 3388 _____ Беннер В.А.

Преподаватель _____ Жангиров Т.Р.

Санкт-Петербург

2025

Задание

Задача 1
Реализуйте алгоритм КМП и с его помощью для заданных шаблона P (|P| < 15000) и текста T (|T| < 5000000) найдите все вхождения P в T.

Вход:
Первая строка - P
Вторая строка - T

Выход:
Индексы начал вхождений P в T, разделенных запятой. Если P не входит в T, то вывести -1.

Задача 2
Заданы две строки A (|A| < 5000000) и B (|B| < 5000000).
Определить, является ли A циклическим сдвигом B (это значит, что A и B имеют одинаковую длину и A состоит из суффикса B, склеенного с префиксом B). Например, `defabc` является циклическим сдвигом `abcdef`.

Вход:
Первая строка - A
Вторая строка - B

Выход:
Если A является циклическим сдвигом B, индекс начала строки B в A, иначе вывести -1. Если возможно несколько сдвигов, вывести первый индекс.

Выполнение работы

Алгоритм префикс-функции — это ключевая часть алгоритма Кнута-Морриса-Пратта (КМП), предназначенная для эффективного поиска подстрок в тексте. Префикс-функция для строки S определяет массив π, где каждый элемент π[i] равен длине наибольшего собственного префикса подстроки S[0..i], совпадающего с её суффиксом. Например, для строки "ababc" префикс-функция принимает значения [0, 0, 1, 2, 0], так как у подстроки "aba" префикс "a" совпадает с суффиксом, а у "abab" совпадает "ab". Алгоритм вычисляет префикс-функцию за линейное время, используя динамическое программирование: на каждом шаге он сравнивает символы и либо увеличивает длину совпадения, либо откатывается к предыдущему совпадающему префиксу.

Префикс-функция вычисляется за O(M) для шаблона длины M.

Поиск вхождений в тексте длины N выполняется за O(N).

Общая сложность алгоритма — O(M + N), что оптимально для задачи поиска подстрок.

Память: O(M) для хранения префикс-функции шаблона.

Тестирование:

Input	Output
ab abab	0,2
IT TAITIT	2,4
dog cats	-1

Input	Output
defabc abcdef	3
xyzw yzwx	1
11123 11153	-1

Выводы:

Разработанный алгоритм корректно находит все вхождения шаблона в тексте, выводит их позиции в порядке возрастания и обрабатывает случаи отсутствия совпадений. Добавление отладочных выводов упрощает анализ работы программы. Решение демонстрирует высокую эффективность даже на больших данных, подтверждая теоретическую оценку сложности.