МИНОБРНАУКИ РОССИИ

Санкт-Петербургский государственный  
электротехнический университет  
«ЛЭТИ» им. В.И. Ульянова (Ленина)

Кафедра МО ЭВМ

ОТЧЁТ  
по лабораторной работе №1 (Вар. XX)  
по дисциплине «Построение и анализ алгоритмов»  
  
Тема: Поиск подстроки с использованием алгоритма Кнута-Морриса-Пратта (KMP)

Студентка гр. 3388  
Глебова В.С  
  
Преподаватель  
Жангиров Т.Р.

Санкт-Петербург  
2025

**Задание**

Реализовать алгоритм КМП и применить его для определения минимального циклического сдвига строки A, при котором она становится равна строке B.  
  
Входные данные: строки A и B одинаковой длины.  
Выход: смещение или -1.

**Описание алгоритма КМП**

КМП (Кнут-Моррис-Пратт) — алгоритм для поиска подстроки в строке за время O(n + m).  
Он состоит из двух частей: построения префикс-функции и основного поиска.

Построение префикс-функции (LPS-массив)

```python  
lps = [0] \* len(pattern)  
length = 0  
for i in range(1, len(pattern)):  
 if pattern[i] == pattern[length]:  
 length += 1  
 lps[i] = length  
 else:  
 if length != 0:  
 length = lps[length - 1]  
 else:  
 lps[i] = 0  
```  
Пояснение:  
LPS (Longest Prefix Suffix) показывает, какую часть строки можно переиспользовать при неудачном сравнении.  
Например, если первые 3 символа шаблона совпали, а 4-й не совпал, мы не начинаем сравнение с начала, а переходим к позиции, указанной в LPS.

Основной цикл поиска KMP

```python  
while i < len(text):  
 if pattern[j] == text[i]:  
 i += 1  
 j += 1  
 if j == len(pattern):  
 return i - j  
 else:  
 if j != 0:  
 j = lps[j - 1]  
 else:  
 i += 1  
```  
Пояснение:  
Когда символы совпадают, индексы i и j увеличиваются.  
Когда происходит несовпадение, j «откатывается» назад на lps[j-1], что позволяет избежать повторного сравнения уже проверенной части шаблона.

Проверка циклического сдвига

```python  
if len(A) == len(B):  
 result = kmp\_search(B, A + A)  
```  
Пояснение:  
Циклический сдвиг можно найти, если строка B содержится в строке A+A.  
Таким образом, если шаблон B входит в A+A — существует сдвиг, равный позиции вхождения.

Отладка

Код снабжён выводом текущих значений переменных и шагов поиска.  
Это позволяет отслеживать, как строится префикс-функция и как продвигается поиск.

Оценка сложности

Время: O(n + m) — линейное по длине текста и шаблона.  
Память: O(m) — массив префикс-функции.

**Выводы**

Был реализован и проанализирован алгоритм КМП. Он позволяет эффективно решать задачу поиска подстроки и применяется для задачи циклического сдвига.  
Использование отладочных сообщений помогает лучше понять ход выполнения алгоритма.