

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

**ОТЧЕТ**  
**по лабораторной работе №4**  
**по дисциплине «Построение и анализ алгоритмов»**  
**Тема: Алгоритм Кнута-Морриса-Пратта**

Студентка гр. 7304

\_\_\_\_\_

Нгуен Т.Т. Зуен

Преподаватель

\_\_\_\_\_

Филатов А.Ю.

Санкт-Петербург

2019

## Цель работы:

Исследование алгоритма Кнута-Морриса-Пратта (КМП) и его реализация на языке C++.

## Задание:

1. Реализуйте алгоритм КМП и с его помощью для заданных шаблона  $P$  ( $|P| \leq 15000$ ) и текста  $T$  ( $|T| \leq 5000000$ ) 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  $-1$ .

2. Заданы две строки  $A$  ( $|A| \leq 5000000$ ) и  $B$  ( $|B| \leq 5000000$ ).

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.

## Описание алгоритма

Алгоритм Кнута - Морриса - Пратта (КМП) - эффективный алгоритм, осуществляющий поиск подстроки в строке.

- Алгоритм КМП предварительно обрабатывает  $pat$  и создает вспомогательный  $lps$  размером  $m$  (такой же, как размер строки), который используется для пропуска символов при сопоставлении.

- $lps$  указывает на самый длинный правильный префикс, который также является суффиксом. Правильный префикс - это префикс с недопустимой целой строкой.

- Мы ищем  $lps$  в подстроке. Более четко мы сфокусируемся на подстроках строк, которые имеют префикс или суффикс.

Для каждого подстрока `pat[i]` ( $i = 0$  до  $m-1$ ), `lps[i]` хранит длину максимального соответствующего собственного префикса, который также является суффиксом подстрока `pat[i]`.

#### Описание функций:

`void computeLPSArray(string pat, int index, vector<int> &lps)` - вычислите значения префикс-функции (vector lps).

`void KMPsearch(string pat, string txt, vector<int> *res)` - алгоритм КМП для поиска позиции подстроки и содержание их в вектор результаты, иначе вывести  $-1$ .

`int KMPsearch(string pat, string txt)` - алгоритм КМП для поиска позиции начала строки `pat` в строку `txt` если строки `pat` является циклическим сдвигом `txt`, иначе вывести  $-1$ .

#### **Результаты:**

1.

```
ab
abdsabab
0,4,6
Press any key to continue .
```

2.

```
C:\Users\duyennh\source\repos\KMP
defabc
abcdef
3
Press any key to continue .
```

#### **Выводы:**

В результате работы программы была реализован алгоритм Кнута - Морриса - Пратта (КМП), который вычисляет позиции подстроки в строку, а также проверяет, что одна строка является ли циклическим сдвигом второй. Также реализован построения префикс-функции для поддержки алгоритма.