

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

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

Студент гр. 8304

Преподаватель

Воропаев А.О.

Размочаева Н.В.

Санкт-Петербург

2020

Вариант 3.

Цель работы.

Разработать программу, определяющую, является ли строка А циклическим сдвигом строки В

Основные теоретические положения.

А является циклическим сдвигом В, если А и В имеют одинаковую длину и А состоит из суффикса В, склеенного с префиксом В). Например, defabc является циклическим сдвигом abcdef

Описание алгоритма.

Для решения поставленной задачи была написана функция kmp, которая осуществляет работу алгоритма Кнута-Морисса-Пратта. Для начала мы высчитываем значения префикс функции для подстроки. Затем мы сравниваем очередные символы текста и подстроки, если они не совпали, мы сдвигаем образец на несколько символов и делаем еще одно сравнение. Если не совпало — опять сдвиг образца, пока символы не совпадут или не дойдем до начала образца.

Сложность алгоритма.

Сложность алгоритма $O(n+m)$, где n — это длина строки, которую мы ищем, а m — длина строки, которой производится поиск.

Сложность по памяти $O(2n+m)$.

Описание основных структур данных и функций.

1) `void prefix_function(const std::string& s, std::vector<int>& pi)` — функция для расчета значений длин префиксов для строки s

s — строка, по которой ведётся расчёт

pi — вектор, в который будет записан результат для каждой позиции в строке.

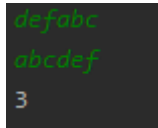
2) `void kmp(std::string& P, std::string& T, int& answer)` — функция реализующая алгоритм Кнута-Морисса-Пратта.

P — подстрока, которую необходимо найти

T — строка, в которой производится поиск

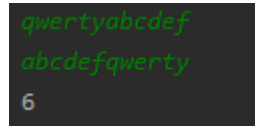
$answer$ — поле, для записи ответа.

Тестирование.



```
defabc
abcdef
3
```

Рисунок 1 – Результаты 1-ого теста



```
qertyabcdef
abcdefqerty
6
```

Рисунок 2 – Результаты 2-го теста

Вывод.

Был получен опыт в реализации алгоритма Кнута-Морисса-Пратта. Также в ходе работы была проанализирована сложность работы алгоритма.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

```
#include <iostream>
#include <string>
#include <vector>

void prefix_function (const std::string& s, std::vector<int>& pi) {
    pi[0] = 0;
    for (int i=1; i < s.length(); ++i) {
        int j = pi[i-1];
        while (j > 0 && s[i] != s[j])
            j = pi[j-1];
        if (s[i] == s[j])
            ++j;
        pi[i] = j;
    }
}

void kmp(std::string& P, std::string& T, int& answer){

    std::vector<int> pi(P.length());
    prefix_function(P, pi);

    for (int k = 0, i = 0; i < T.length(); ++i)
    {
        while ((k > 0) && (P[k] != T[i]))
            k = pi[k-1];

        if (P[k] == T[i])
            k++;

        if (k == P.length()) {
            answer = i - P.length() + 1;
            return;
        }
    }
    answer = -1;
}

int main() {

    int answer = -1;
    std::string text, substring;
    std::cin >> text >> substring;

    text += text;

    kmp(substring, text, answer);

    std::cout << answer;

    return 0;
}
```