# Лабораторная работа № 4 по курсу дискретного анализа: поиск подстроки в строке

Выполнил студент группы М8О-207-20 МАИ *Михеева Кристина*.

## Условие

1. Необходимо реализовать один из стандартных алгоритмов поиска образцов для указанного алфавита. Искомый образец задается на первой строке входного файла. Затем следует текст, состоящий из слов или чисел, в котором нужно найти заданные образцы. Никаких ограничений на длину строк, равно как на количество слов или числ в них, не накладывается. В выходной файл нужно вывести информацию о всех вхождениях искомых образцов вобрабатываемый текст: по одному вхождению на строчку. Для заданий, в которых требуется найти только один образец, следует вывести два числа через запятую: номер строки и номер слова в строке, с которого начинается найденный образец. Нумерация начинается с единицы. Номер строки в тексте должен отсчитываться от его реального начала (то есть, без учёта строк, занятых образцами). Порядок следования вхождений образцов несущественен.

2. Вариант алгоритма: Поиск одного образца при помощи алгоритма Бойера-Мура.

3. Вариант алфавита: Слова не более 16 знаков латинского алфавита (регистронезависимые).

## Метод решения

Поиск подстроки в строке осуществляется с помощью алгоритма Бойера-Мура, где сравнение образца с текстом происходит справо на лево. Гланой особеннстью данного алгоритма являются правило плохого символа и правило хорошего суффикса. С помощью них поиск происходит значительно бвстрее. Итак, что же из себя прдставляют правила хорошнго суффикса и плохого символа: ¯

1) Правило хорошего суффикса. После несовпадения сдвигаем образец так, чтобы совместить совпавшую подстроку T текста со следующим вхождением этой подстроки в образец. Если такой подстроки в образце больше нет, выбираем наибольший префикс образца, являющийся суффиксом совпавшей подстроки T. ¯
В сильном варианте дополнительно требуется, чтобы после сдвига в образце перед T встал другой символ.

2) Правило плохого символа.После несовпадения сдвигаем образец так,чтобы совместить несовпавший символ текста с таким же символом образца. (Если такого символа в образце нет, сдвигаем на всю длину образца.)

В слабом варианте для каждого символа алфавита запоминаем его крайнее правое вхождение в образец.

В сильном варианте для каждого символа алфавита запоминаем все его вхождения в образец.

## Описание программы

В данной программе содержится оин файл, где реализованы функции z-функция, правила плохого символа и хорошего суффикса, и сам алгоритм поиска подстроки в строке.

## Дневник отладки

Ошибки не выявлены.

## Тест производительности

Число операций: 1000, 10000, 1000000.

Алгоритм Боейра-Мура: 1099ms, 10765ms, 111911ms.

Наивный алгоритм: 2076ms, 12009ms, 136877ms.

Время отличается не значительно в алгоритме Бойера-Мура от Наивного алгоритма.

## Выводы

В данной лабораторной работе было предложено изуить некоторые виды алгоритмов поиска подстроки в строке. Мной был реализован алгоритм Боейра-Мура, а также правила хорошего суффикса и плохого символа.

Данный алгоритм является достаточно быстрвм, что позволяет за линейное время достаточно легко обрабатывать большие входные данные в течение нескольких секунд. В нем достаточно просто разобраться, однако при реализации алгоритма надо быть внимательным, чтобы не возникло никаких ошибок. Тема алгоритмов поиска подстроки в строке оказалась достаточно полезна, что полученные знания смогут пригодится в дальнейшей работе с поисками больших строк.