**Федеральное агентство связи  
Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования   
«Московский технический университет связи и информатики»**

Кафедра «Теории вероятностей и прикладной математики»

**Лабораторная работа №3**

По дисциплине «Структуры и алгоритмы обработки данных»   
по теме «Методы поиска подстроки в строке»  
Вариант №6

Выполнил: студент группы БСТ1955

Гердт Таисия

## Задание

Реализовать метод Кнута-Морриса-Пратта поиска подстроки в строке в соответствии с индивидуальным заданием. Для всех вариантов добавить реализацию добавления строк, ввода подстроки и поиска подстроки. Предусмотреть возможность существования пробела. Ввести опцию чувствительности / нечувствительности к регистру. Оценить время работы каждого алгоритма поиска и сравнить его со временем работы стандартной функции поиска, используемой в выбранном языке программирования.

Код программы:

#функция КМП

def kmp(pat, txt):

M = len(pat)

N = len(txt)

lps = [0]\*M

j = 0

lsp(pat, M, lps)

i = 0

flag = 0

while i < N:

if pat[j] == txt[i]:

i += 1

j += 1

if j == M:

print ("KMP: Found pattern at index " + str(i-j))

j = lps[j-1]

flag += 1

elif i < N and pat[j] != txt[i]:

if j != 0:

j = lps[j-1]

else:

i += 1

if flag == 0:

print("KMP: Pattern is not found")

#функция нахождения массива длиннейших префиксов являющихся суффиксами

def lsp(pat, M, lps):

len = 0

lps[0]

i = 1

while i < M:

if pat[i]== pat[len]:

len += 1

lps[i] = len

i += 1

else:

if len != 0:

len = lps[len-1]

else:

lps[i] = 0

i += 1

import time, random, string

flag = input("Enter 1 for case-sensitive search\nEnter 2 for non-case sensetive search\nEnter 0 for quit\n")

while(True):

if flag == "1":

txt = input("Enter string:\n")

pat = input("Enter pattern:\n")

t = time.time()

kmp(pat, txt)

t = time.time() - t

print("KMP execurtion time: ", t)

t = time.time()

m = txt.index(pat)

t = time.time() - t

print(".index(): Found pattern at index ", m)

print(".index() execurtion time: ", t)

elif flag == "2":

txt = input("Enter string:\n").lower()

pat = input("Enter pattern:\n").lower()

kmp(pat, txt)

elif flag == "0":

quit()

else:

print("Invalid command")

flag = input("Enter 1 for case-sensitive search\nEnter 2 for non-case sensetive search\nEnter 0 for quit\n")

Выполнение программы:





