

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

**ОТЧЕТ**  
**по лабораторной работе №3**  
**по дисциплине «Информатика»**  
**Тема: Машина Тьюринга**

Студент гр. 3344

Бажуков С.В.

Преподаватель

Иванов Д.В.

Санкт-Петербург

2023

## **Цель работы**

Получить навык работы с машиной Тьюринга, научиться создавать таблицы состояний.

## **Задание.**

### **Вариант 2**

На вход программе подается строка неизвестной длины. Каждый элемент является значением в ячейке памяти ленты Машины Тьюринга.

На ленте находится последовательность латинских букв из алфавита {a, b, c}.

Напишите программу, которая заменяет в исходной строке символ, идущий после последних двух встретившихся символов 'a', на предшествующий им символ (гарантируется, что это не пробел). Наличие в строке двух подряд идущих символов 'a' гарантируется.

Указатель на текущее состояние Машины Тьюринга изначально находится слева от строки с символами (но не на первом ее символе). По обе стороны от строки находятся пробелы.

Алфавит:

a

b

c

" " (пробел)

Соглашения:

1. Направление движения автомата может быть одно из R (направо), L (налево), N (неподвижно).
2. Гарантируется, что длина строки не менее 5 символов и не более 15.
3. В середине строки не могут встретиться пробелы.
4. При удалении или вставке символов направление сдвигов подстрок не принципиально (т. е. результат работы алгоритма может быть сдвинут по ленте в любую ее сторону на любое число символов).
5. Курсор по окончании работы алгоритма может находиться на любом символе.

## Выполнение работы

Вначале считываем начальную строку (ленту машины Тьюринга) в переменную *s* и преобразовываем эту строку в список. Далее создаём переменную текущего состояния *state* (изначально *state*="q1"), переменную *index*, содержащую текущий номер элемента (изначально *index*=0) и словарь состояний *states*.

В словаре *states* описываем каждое состояние:

q1 – начальное состояние, которое необходимо, чтобы найти начало строки.

q2 – состояние, которое необходимо, чтобы найти конец строки.

q3 – состояние, необходимое для поиска последнего символа "а".

q4 – состояние, необходимое для поиска предпоследнего символа "а".

q5 – состояние, необходимое для определения символа, предшествующего "аа".

q6 – состояние, необходимое для замены символа, идущего после "аа", на символ, идущий до "аа", если это символ "а".

q7 – состояние, необходимое для замены символа, идущего после «аа», на символ, идущий до "аа", если это символ "b".

q8 – состояние, необходимое для замены символа, идущего после «аа», на символ, идущий до "аа", если это символ "с".

q9 – конечное состояние.

После создания словаря состояний *states* было написано тело цикла *while*, который продолжался, пока значение переменной *state* не становилось "q9". Внутри цикла созданы переменная *token*, обозначающая текущий символ, переменная *step*, обозначающая шаг, на который необходимо сдвинуться по ленте. В переменные *token* и *step* записывались значения из словаря *states* в соответствии с текущим состоянием. В переменную *state* записывалось текущее состояние. Далее определённый символ в строке *s* изменялся.

После завершения цикла программа выводила на экран изменённую строку. Ниже Таблица 1 – Таблица состояний.

Таблица 1 - Таблица состояний

	« »	«a»	«b»	«c»
q1	" ",1,"q1"	"a",1,"q2"	"b",1,"q2"	"c",1,"q2"
q2	" ",-1,"q3"	"a",1,"q2"	"b",1,"q2"	"c",1,"q2"
q3		"a",-1,"q4"	"b",-1,"q3"	"c",-1,"q3"
q4		"a",-1,"q5"	"b",-1,"q3"	"c",-1,"q3"
q5		"a",1,"q6"	"b",1,"q7"	"c",1,"q8"
q6	"a",0,"q9"	"a",1,"q6"	"a",0,"q9"	"a",0,"q9"
q7	"b",0,"q9"	"a",1,"q7"	"b",0,"q9"	"b",0,"q9"
q8	"c",0,"q9"	"a",1,"q8"	"c",0,"q9"	"c",0,"q9"

### Тестирование.

Результаты тестирования представлены в табл. 2.

Таблица 2 - результаты тестирования.

№ п/п	Входные данные	Выходные данные	Комментарии
1.	асабсааб	асабсаас	-
2.	abbacabcbaa	abbacabcbaab	-

## **Выводы**

В ходе выполнения работы был изучен принцип работы машины Тьюринга. Была создана программа для обработки входной строки по определённому алгоритму по принципу машины Тьюринга.

## ПРИЛОЖЕНИЕ А

### ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: main.py

```
s=list(input())
state = "q1"
index = 0
states={
    "q1":{
        " ":[" ",1,"q1"],"a":["a",1,"q2"],"b":["b",1,"q2"],"c":["c",1,"q2"]
    },
    "q2":{
        " ":[" ",-1,"q3"],"a":["a",1,"q2"],"b":["b",1,"q2"],"c":["c",1,"q2"]
    },
    "q3":{"a":["a",-1,"q4"],"b":["b",-1,"q3"],"c":["c",-1,"q3"]},
    "q4":{"a":["a",-1,"q5"],"b":["b",-1,"q3"],"c":["c",-1,"q3"]},
    "q5":{"a":["a",1,"q6"],"b":["b",1,"q7"],"c":["c",1,"q8"]},
    "q6":{
        "a":["a",1,"q6"],"b":["a",0,"q9"],"c":["a",0,"q9"]," ":["a",0,"q9"]
    },
    "q7":{
        "a":["a",1,"q7"],"b":["b",0,"q9"],"c":["b",0,"q9"]," ":["b",0,"q9"]
    },
    "q8":{
        "a":["a",1,"q8"],"b":["c",0,"q9"],"c":["c",0,"q9"]," ":["c",0,"q9"]
    }
}

while state != "q9":
    token,step,state = states[state][s[index]]
    s[index] = token
    index+=step

print(''.join(s))
```