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 и конечные автоматы

Студент гр. 2381	 Долотов Н.А.
Преподаватель	 Шевская Н.В

Санкт-Петербург

2022

Цель работы.

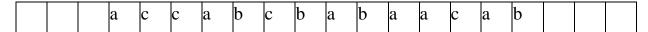
Изучить принцип работы машины Тьюринга и разработать программу для решения поставленной задачи, имитирующую механизм работы данной машины.

Задание.

Вариант 1

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

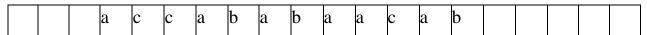
На ленте находится последовательность латинских букв из алфавита {a, b, c}.



Напишите программу, которая удаляет в исходной строке два символа, следующих за первым встретившимся символом 'b'. Если первый встретившийся символ 'b' — последний в строке, то удалить его. Если первый встретившийся символ 'b' — предпоследний в строке, то удалить один символ, следующий за ним, т. е. последний в строке. Если в строке символ 'b' отсутствует, то удалить самый первый символ строки. После удаления в строке не должно оставаться пробелов и пустых мест!

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.

Для примера выше лента будет выглядеть так:



Алфавит:

- a
- b
- C
- "" (пробел)

Соглашения:

- 1. Направление движения автомата может быть одно из R (направо), L (налево), N (неподвижно).
- 2. Гарантируется, что длинна строки не менее 5 символов и не более 13.
- 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.

Ваша программа должна вывести полученную ленту после завершения работы.

Выполнение работы.

Таблица состояний

	ʻa'	<i>'b'</i>	'c'	<i>、</i> ,
q1	'a', R, 'q2'	'b', R, 'q3'	'c', R, 'q2'	' ', R, 'q1'
<i>q</i> 2	'a', R, 'q2'	'b', R, 'q3'	'c', R, 'q2'	' ', <i>L</i> , 'q4'
<i>q</i> 3	'', R, 'q5'	'', R, 'q5'	'', R, 'q5'	' ', <i>L</i> , 'q5'
q4	'a', L, 'q4'	_	'c', L, 'q4'	' ', R, 'q5'
<i>q</i> 5	'', N, 'q0'	'', N, 'q0'	'', N, 'q0'	' ', N, 'q0'

Перемещения каретки:

- Влево L;
- Bправо R;
- Не передвигать N.

table — словарь таблицы состояний машины Тьюринга, *memory* — начальная строка, *state* — переменная, хранящая текущее состояние, *ind* — переменная, хранящая текущий номер ячейки на ленте.

Краткое описание состояний:

q1 — начальное состояние, которое необходимо, чтобы найти первую букву в строке; если первая буква — b, то машина переходит в состояние q3; если первая буква — a или c, то в состояние q2.

q2 — состояние, которое необходимо для обработки случая, когда буква b стоит не на первой позиции (после нахождения переходит в состояние q3) или её вообще нет в строке; в таком случае, машина пропускает все ячейки со

значениями a и c, доходит до конца буквенной строки и переходит в состояние q4.

q3 — состояние, которое необходимо для удаления первого символа, идущего сразу за первой найденной буквой b; если данный символ — '', то машина переходит в состояние q5.

q4 - состояние, которое необходимо для сдвига каретки с конца буквенной строки в начало; при нахождении символа ' сдвигается на первый символ и переходит в состояние q5.

q5 — состояние, которое необходимо для удаления второго символа, идущего после первой найденной буквы b, для удаления последнего символа в буквенной строке, либо для удаления первого символа в строке, если буквы b нет в строке; после обработки каждого символа машина переходит в состояние q0.

q0 – конечное состояние, завершение работы машины Тьюринга.

Выводы.

Был изучен принцип работы машины Тьюринга.

Разработана программа, имитирующая механизм работы машины Тьюринга и решающая поставленную задачу по удалению символов после нахождения первой буквы b.

приложение а

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: main.py

```
L, R, N = -1, 1, 0
     table = {
          'q1': {'a': ['a', R, 'q2'], 'b': ['b', R, 'q3'], 'c': ['c', R,
'q2'], ' ': [' ', R, 'q1']},

'q2': {'a': ['a', R, 'q2'], 'b': ['b', R, 'q3'], 'c': ['c', R, 'q2'], ' ': [' ', L, 'q4']},
          'q3': {'a': ['', R, 'q5'], 'b': ['', R, 'q5'], 'c': ['', R,
'q5'], ' ': [' ', L, 'q5']},
          'q4': {'a': ['a', L, 'q4'], 'c': ['c', L, 'q4'], ' ': [' ', R,
'q5']},
          'q5': {'a': ['', N, 'q0'], 'b': ['', N, 'q0'], 'c': ['', N,
'q0'],
        ' ': [' ', N, 'q0']},
     memory = list(input())
     state = 'q1'
     ind = 0
     while state != 'q0':
          act = table[state][memory[ind]]
          memory[ind] = act[0]
         ind += act[1]
          state = act[2]
     print(''.join(memory))
```