

**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**

**Сьомак Д.А.**

**Преподаватель**

**Иванов Д.В.**

**Санкт-Петербург**

**2023**

## **Цель работы**

Освоение работы с машиной Тьюринга. Получение навыков составления таблиц и словарей для машины Тьюринга.

## 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.

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 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.

## Выполнение работы

Была считана строка, преобразована в список и занесена в переменную `lenta`. Было задано начальное состояние `state = "qs"`, начальный индекс строки `index = 0`, словарь состояний машины Тьюринга `states`. Словарь содержит ключи в виде состояний и значения в виде словарей с действиями для каждого символа данного состояния.

Состояния:

`qs` - начальное состояние, поиск первого символа

`q1` - нахождение символа `b` проходом до конца строки

`q2` - символ `b` не найден, возвращение к началу строки

`q3`, `q4`, `q5a`, `q5c` - состояния, которые взаимодействуя друг с другом удаляют первый символ строки, путём замены первого символа на пробел и передвижения каждого символа на один влево, при встрече конца строки переходит к конечному состоянию

`q6` - символ `b` найден, если дальше него нет букв, то переход в `q7`

`q7` - замена символа `b` на пробел и переход к конечному состоянию

`q8` - если после `b` есть только один символ, то переход в `q9`

`q9` - замена любого символа на пробел и переход к конечному состоянию

`q10`, `q11`, `q12a`, `q12b`, `q12c` - состояния, которые взаимодействуя друг с другом удаляют второй символ после `b`, путём замены его на пробел и передвижения каждого символа на один влево, при встрече конца строки переходит в `q12sp`.

`q12sp` - пропускает пробел, который был сдвинут из середины строки

`q13` - возвращение к первому символу, который надо удалить

`q14`, `q15`, `q16a`, `q16b`, `q16c` - состояния, которые взаимодействуя друг с другом удаляют первый символ после `b`, путём замены его на пробел и передвижения каждого символа на один влево, при встрече конца строки переходит к конечному состоянию.

`qe` - конечное состояние, завершение работы программы

Был реализован цикл while, который работает до конечного состояния qe. На каждой итерации цикла в зависимости от текущего состояния происходило обновление symbol - символ ленты, step - шаг сдвига ленты, state - текущего состояния ленты. Далее происходит замена символа ленты на символ, полученный действиями текущего состояния. Обновление индекса текущим шагом step(значение от -1 до 1). После цикла отредактированный список lenta выводится на экран с помощью print(''.join(lenta)).

Таблица состояний представлена в табл. 1

Таблица 1 - Таблица состояний

	'a'	'b'	'c'	' '
qs	'a', 1, 'q1'	'b', 1, 'q6'	'c', 1, 'q1'	' ', 1, 'qs'
q1	'a', 1, 'q1'	'b', 1, 'q6'	'c', 1, 'q1'	' ', -1, 'q2'
q2	'a', -1, 'q2'		'c', -1, 'q2'	' ', 1, 'q3'
q3	' ', 1, 'q4'		' ', 1, 'q4'	' ', 1, 'q4'
q4	' ', -1, 'q5a'		' ', -1, 'q5c'	' ', 1, 'qe'
q5a				'a', 1, 'q3'
q5c				'c', 1, 'q3'
q6	' ', 1, 'q8'	' ', 1, 'q8'	' ', 1, 'q8'	' ', -1, 'q7'
q7		' ', 1, 'qe'		
q8	'a', 0, 'q10'	'b', 0, 'q10'	'c', 0, 'q10'	' ', -1, 'q9'
q9	' ', 1, 'qe'	' ', 1, 'qe'	' ', 1, 'qe'	' ', 1, 'qe'
q10	' ', 1, 'q11'	' ', 1, 'q11'	' ', 1, 'q11'	' ', 1, 'q11'
q11	' ', -1, 'q12a'	' ', -1, 'q12b'	' ', -1, 'q12c'	' ', -1, 'q12sp'
q12a				'a', 1, 'q10'
q12b				'b', 1, 'q10'

q12c				'c', 1, 'q10'
q12sp				'', -1, 'q13'
q13	'a', -1, 'q13'	'b', -1, 'q13'	'c', -1, 'q13'	'', 0, 'q14'
q14	'', 1, 'q15'	'', 1, 'q15'	'', 1, 'q15'	'', 1, 'q15'
q15	'', -1, 'q16a'	'', -1, 'q16b'	'', -1, 'q16c'	'', 1, 'qe'
q16a				'a', 1, 'q14'
q16b				'b', 1, 'q14'
q16c				'c', 1, 'q14'

Исходный код см. в приложении А

## Тестирование

Результаты тестирования представлены в табл. 2.

Таблица 2 – Результаты тестирования

№ п/п	Входные данные	Выходные данные	Комментарии
1.	асасасb	асасас	-
2.	ассасссbc	ассасссb	-
3.	ассасbbcасb	ассасbacb	-



## **Выводы**

Была освоена работа с машиной Тьюринга. Были получены навыки составления таблиц и словарей для машины Тьюринга. Была реализована машина Тьюринга в соответствии с условиями задания.

## ПРИЛОЖЕНИЕ А

### ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: Somak\_Demid\_lb3.py

```
lenta = list(input())
state = "qs"
index = 0
states = {'qs': {'a': ('a', 1, 'q1'), 'b': ('b', 1, 'q6'), 'c': ('c', 1, 'q1'), ' ': (' ', 1, 'qs')},
          'q1': {'a': ('a', 1, 'q1'), 'b': ('b', 1, 'q6'), 'c': ('c', 1, 'q1'), ' ': (' ', -1, 'q2')},
          'q2': {'a': ('a', -1, 'q2'), 'c': ('c', -1, 'q2'), ' ': (' ', 1, 'q3')},
          'q3': {'a': (' ', 1, 'q4'), 'c': (' ', 1, 'q4'), ' ': (' ', 1, 'q4')},
          'q4': {'a': (' ', -1, 'q5a'), 'c': (' ', -1, 'q5c'), ' ': (' ', 1, 'qe')},
          'q5a': {' ': ('a', 1, 'q3')},
          'q5c': {' ': ('c', 1, 'q3')},
          'q6': {'a': (' ', 1, 'q8'), 'b': (' ', 1, 'q8'), 'c': (' ', 1, 'q8'), ' ': (' ', -1, 'q7')},
          'q7': {'b': (' ', 1, 'qe')},
          'q8': {'a': ('a', 0, 'q10'), 'b': ('b', 0, 'q10'), 'c': ('c', 0, 'q10'), ' ': (' ', -1, 'q9')},
          'q9': {'a': (' ', 1, 'qe'), 'b': (' ', 1, 'qe'), 'c': (' ', 1, 'qe'), ' ': (' ', 1, 'qe')},
          'q10': {'a': (' ', 1, 'q11'), 'b': (' ', 1, 'q11'), 'c': (' ', 1, 'q11'), ' ': (' ', 1, 'q11')},
          'q11': {'a': (' ', -1, 'q12a'), 'b': (' ', -1, 'q12b'), 'c': (' ', -1, 'q12c'), ' ': (' ', -1, 'q12sp')},
          'q12a': {' ': ('a', 1, 'q10')},
          'q12b': {' ': ('b', 1, 'q10')},
          'q12c': {' ': ('c', 1, 'q10')},
          'q12sp': {' ': (' ', -1, 'q13')},
          'q13': {'a': ('a', -1, 'q13'), 'b': ('b', -1, 'q13'), 'c': ('c', -1, 'q13'), ' ': (' ', 0, 'q14')},
          'q14': {'a': (' ', 1, 'q15'), 'b': (' ', 1, 'q15'), 'c': (' ', 1, 'q15'), ' ': (' ', 1, 'q15')},
          'q15': {'a': (' ', -1, 'q16a'), 'b': (' ', -1, 'q16b'), 'c': (' ', -1, 'q16c'), ' ': (' ', 1, 'qe')},
          'q16a': {' ': ('a', 1, 'q14')},
          'q16b': {' ': ('b', 1, 'q14')},
          'q16c': {' ': ('c', 1, 'q14')}}

while state != "qe":
    symbol, step, state = states[state][lenta[index]]
    lenta[index] = symbol
    index += step

print("".join(lenta))
```