# 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(ка) гр. 3383	Логинова А. Ю.
Преподаватель	Иванов Д. В.

Санкт-Петербург

2023

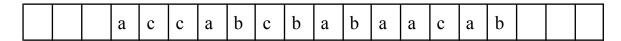
### Цель работы.

Написать программу, которая оборачивает исходную строку. Результат работы алгоритма - исходная последовательность символов в обратном порядке.

### Задание.

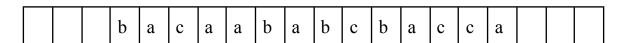
На вход программе подается строка неизвестной длины. Каждый элемент является значением в ячейке памяти ленты Машины Тьюринга.

На ленте находится последовательность латинских букв из алфавита {a, b, c}, которая начинается с символа 'a'.



Напишите программу, которая оборачивает исходную строку. Результат работы алгоритма - исходная последовательность символов в обратном порядке.

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 (можно расширять при необходимости):

- 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.
- 6. Нельзя использовать дополнительную ленту, в которую записывается результат.

Ваша программа должна вывести полученную ленту после завершения работы.

В отчет включите таблицу состояний. Отдельно кратко опишите каждое состояние, например:

q1 - начальное состояние, которое необходимо, чтобы обнаружить конец строки.

## Пример:

Input	Output	
abcabc	cbacba	
abacbbc	cbbcaba	

### Выполнение работы.

Для написания программы было необходимо составить таблицу состояний, с помощью которой будет перевернута пользовательская строка. Программа включает в себя словарь *table* и цикл, который проходится по элементам ленты *tape* с помощью индекса *i*. Каждый ключ в словаре *table* является кортежем, включающим в себя номер состояния и элемент ленты. Значением словаря является кортеж, состоящий из нового элемента (который необходимо записать), сдвиг (влево, вправо или на месте), и новое состояние. Так, встречая сдвиг вправо, индекс *i* будет увеличиваться, и встречая сдвиг влево, *i* будет уменьшаться.

Рассмотрим все состояния подробнее:

q1 - начальное состояние, которое ищет первую букву,

q2, q3, q4, q11 - состояния, которые записывают в ячейку a, b, c, npoбел соответственно, какой бы элемент в ней ни был, и сдвигаются влево,

q5, q6, q7 - состояния, которые записывают в ячейку a, b, c соответственно, какой бы элемент в ней ни был, и сдвигаются вправо,

q8 - состояние, которое записывает тот же самый элемент, что и в ячейке, и сдвигается вправо, q9 - состояние, которое записывает в ячейку пробел, какой бы элемент в ней ни был, и сдвигается влево, q10 - состояние, которое записывает тот же самый элемент, что и в ячейке, и сдвигается влево,

 $q12,\ q13,\ q14$  - состояния, которые записывают в пустые ячейки  $a,\ b,\ c$  соответственно.

 $q\theta$  - терминальное состояние.

Разработанный программный код см. в приложении А. Результаты тестирования представлены в табл. 1.

# Тестирование.

Таблица 1 – Результаты тестирования

№ п/п	Входные данные	Выходные данные	Комментарии
1	'abcabcabc '	'cbacbacba '	Программа работа корректно.
2	' abcabcabcbcabcabc '	' cbacbacbacbacbacba '	

# Выводы.

Была написана программа, обрабатывающая ленту с пользовательской строкой, которая переворачивает ее с помощью таблицы состояний. Был изучен принцип работы машины Тьюринга. Задача была решена с помощью словаря и цикла while.

### ПРИЛОЖЕНИЕ А

# ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: main.py

```
table = {
 ('q1', 'a'): ('a', 'R', 'q2'), ('q1', 'b'): ('b', 'R', 'q3'), ('q1',
'c'): ('c', 'R', 'q4'),
 ('q1', ''): ('', 'R', 'q1'),
 ('q2', 'a'): ('a', 'L', 'q5'), ('q2', 'b'): ('a', 'L', 'q6'), ('q2',
'c'): ('a', 'L', 'q7'),
 ('q2', ''): ('a', 'L', 'q9'),
 ('q3', 'a'): ('b', 'L', 'q5'), ('q3', 'b'): ('b', 'L', 'q6'), ('q3',
'c'): ('b', 'L', 'q7'),
 ('q3', ''): ('b', 'L', 'q9'),
 ('q4', 'a'): ('c', 'L', 'q5'), ('q4', 'b'): ('c', 'L', 'q6'), ('q4',
'c'): ('c', 'L', 'q7'),
 ('q4', ''): ('c', 'L', 'q9'),
 ('q5', 'a'): ('a', 'R', 'q8'), ('q5', 'b'): ('a', 'R', 'q8'), ('q5',
'c'): ('a', 'R', 'q8'),
 ('q6', 'a'): ('b', 'R', 'q8'), ('q6', 'b'): ('b', 'R', 'q8'), ('q6',
'c'): ('b', 'R', 'q8'),
 ('q7', 'a'): ('c', 'R', 'q8'), ('q7', 'b'): ('c', 'R', 'q8'), ('q7',
'c'): ('c', 'R', 'q8'),
 ('q8', 'a'): ('a', 'R', 'q2'), ('q8', 'b'): ('b', 'R', 'q3'), ('q8',
'c'): ('c', 'R', 'q4'),
 ('q8', ''): ('', 'R', 'q11'),
 ('q9', 'a'): (' ', 'L', 'q10'), ('q9', 'b'): (' ', 'L', 'q10'),
('q9', 'c'): (' ', 'L', 'q10'),
 ('q10', 'a'): ('a', 'L', 'q10'), ('q10', 'b'): ('b', 'L', 'q10'),
('q10', 'c'): ('c', 'L', 'q10'),
 ('q10', ''): ('', 'R', 'q8'),
 ('q11', 'a'): (' ', 'L', 'q12'), ('q11', 'b'): (' ', 'L', 'q13'),
('q11', 'c'): (' ', 'L', 'q14'),
 ('q11', ''): ('', 'N', 'q0'),
 ('q12', ' '): ('a', 'R', 'q8'), ('q13', ' '): ('b', 'R', 'q8'),
('q14', ''): ('c', 'R', 'q8')
tape = [i for i in input()]
state = 'q1'
i = 0
while state != 'q0':
```