

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. 3342

Лучкин М.А.

Преподаватель

Иванов Д.В.

Санкт-Петербург

2023

Цель работы

Написать программу, которая заменяет в исходной строке символ, идущий после последних двух встретившихся символов 'a', на предшествующий им символ, используя машину 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.

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.

В отчете предоставьте таблицу состояний. Отдельно кратко опишите каждое состояние, например:

q1 - начальное состояние, которое необходимо, чтобы найти первый встретившийся символ 'b'.

Выполнение работы

Код осуществляет обработку каждого из 12 состояний машины и переходов между ними, в результате изменяя входную ленту согласно условиям задачи.

1. Инициализация начального состояния $state = 'q1'$, установка коретки в начале ленты $head=0$.
2. Вход в цикл $while\ state \neq 'q12'$.
3. Извлечение символа и операции из таблицы переходов $d[state][tape[head]]$.
4. Обновление символа на ленте $tape[head]$ в соответствии с таблицей переходов.
5. Сдвиг коретки на нужное количество шагов в соответствии с операцией из таблицы переходов.
6. Изменение текущего состояния $state$ на новое состояние, указанное в таблице переходов.
7. Повторение шагов 3-6 до достижения конечного состояния $q9$.
8. Вывод измененной ленты после завершения работы Машины Тьюринга.

Тестирование

Результаты тестирования представлены в табл. 1.

Таблица 1 – Результаты тестирования

№ п/п	Входные данные	Выходные данные	Комментарии
1.	abcaabbcbc	ababbcbc	-
2.	aabbaacbaa	aabacbaa	-

Выводы

В результате работы было продемонстрировано применение Машины Тьюринга для решения задачи замены символа в строке.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: main.py

```
moves = {'R': 1, 'L': -1, 'N': 0}

d = {
    'q1': {'a': 'a;N;q2', 'b': 'b;N;q2', 'c': 'c;N;q2', ' ': ' ':
    ' ;R;q1'},
    'q2': {'a': 'a;R;q2', 'b': 'b;R;qX', 'c': 'c;R;q2', ' ': ' ':
    ' ;L;q10'},
    'qX': {'a': 'a;N;q3', 'b': 'b;N;q3', 'c': 'c;N;q3', ' ': ' ':
    ' ;L;qY'},
    'qY': {'b': ' ': ' ;N;q12'},
    'q3': {'a': 'a;RR;q4', 'b': 'b;RR;q4', 'c': 'c;RR;q4', ' ': ' ':
    ' ;N;q12'},
    'q4': {'a': 'a;LL;q6', 'b': 'b;LL;q7', 'c': 'c;LL;q8', ' ': ' ':
    ' ;LL;q9'},
    'q5': {'b': ' ': ' ;N;q12'},
    'q6': {'a': 'a;R;q3', 'b': 'a;R;q3', 'c': 'a;R;q3'},
    'q7': {'a': 'b;R;q3', 'b': 'b;R;q3', 'c': 'b;R;q3'},
    'q8': {'a': 'c;R;q3', 'b': 'c;R;q3', 'c': 'c;R;q3'},
    'q9': {'a': ' ': ' ;R;q3', 'b': ' ': ' ;R;q3', 'c': ' ': ' ;R;q3'},
    'q10': {'a': 'a;L;q10', 'b': 'b;L;q10', 'c': 'c;L;q10', ' ': ' ':
    ' ;R;q11'},
    'q11': {'a': ' ': ' ;N;q12', 'b': ' ': ' ;N;q12', 'c': ' ': ' ;N;q12'},
}

def turingMachine(tape):
    state = 'q1'
    head = 0
    while state != 'q12':
        symbol, directions, newState =
d[state][tape[head]].split(';')

        tape[head] = symbol
        for i in directions:
            head += moves[i]
        state = newState

tape = [' '] + list(input()) + [' '] + [' ']
turingMachine(tape)
print(''.join(tape[1:-1]))
```