

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

Белаид Фарук

Преподаватель

Иванов Д.В.

Санкт-Петербург

2023

Цель работы.

Изучить устройство и принцип работы машины Тьюринга, строение таблицы состояний и её использование для программирования машины. Научиться реализовывать машину на языке Python. На основе полученных сведений и навыков решить алгоритмическую задачу работы со строкой.

Задание.

Вариант №3

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, предшествующий первому встретившемуся символу 'c' на символ, следующий за первым встретившимся символом 'a'. Если первый встретившийся символ 'a' в конце строки, то используйте его в качестве заменяющего.

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.

Программа должна вывести полученную ленту после завершения работы.

Выполнение работы.

Таблица состояний реализованной машины Тьюринга в табл. 1.

Таблица 1 – Таблица состояний

Состояние	'a'	'b'	'c'	' '
q0	'a', R, q2	'b', R, q1	'c', R, q1	' ', R, q0
q1	'a', R, q2	'b', R, q1	'c', R, q1	' ', N, q12
q2	'a', L, q3	'b', L, q4	'c', L, q5	' ', L, q3
q3	'a', L, q3	'b', L, q3	'c', L, q3	' ', R, q6
q4	'a', L, q4	'b', L, q4	'c', L, q4	' ', R, q7
q5	'a', L, q5	'b', L, q5	'c', L, q5	' ', R, q8
q6	'a', R, q6	'b', R, q6	'c', L, q9	' ', N, q12
q7	'a', R, q7	'b', R, q7	'c', L, q10	' ', N, q12
q8	'a', R, q8	'b', R, q8	'c', L, q11	' ', N, q12
q9	'a', N, q12	'a', N, q12	'a', N, q12	'a', N, q12
q10	'b', N, q12	'b', N, q12	'b', N, q12	'b', N, q12
q11	'c', N, q12	'c', N, q12	'c', N, q12	'c', N, q12

Описание состояний:

- q0 – поиск начала строки, начальное состояние
- q1 – поиск первого вхождения символа 'a'
- q2 – “запоминание” символа после первого 'a'
- q3 – поиск начала строки для вставки 'a'
- q4 – поиск начала строки для вставки 'b'
- q5 – поиск начала строки для вставки 'c'

- q6 – поиск первого вхождения 'с' для вставки 'а'
- q7 – поиск первого вхождения 'с' для вставки 'b'
- q8 – поиск первого вхождения 'с' для вставки 'с'
- q9 – вставка 'а' перед первым встретившимся 'с'
- q10 – вставка 'b' перед первым встретившимся 'с'
- q11 – вставка 'с' перед первым встретившимся 'с'
- q12 – конечное состояние

В программе использованы следующие переменные:

- *table* – словарь, таблица состояний
- *L, N, R* – направление движения каретки и соответственно равны -1, 0, 1
- *line* – лента, список каждой ячейки ленты
- *state* – состояние автомата
- *position* – положение каретки

Сначала программа считывает ленту, после чего, меняя состояния и положение на ленте при помощи цикла *while* вплоть до q12, изменяет строку.

Разработанный программный код см. в приложении А.

Тестирование.

Результаты тестирования представлены в табл. 2.

Таблица 2 – Результаты тестирования

№ п/п	Входные данные	Выходные данные	Комментарии
1.	abcabc	abcabc	Ответ верный
2.	cabc	bcabc	Ответ верный
3.	bcca	acca	Ответ верный
4.	cabaac	bcabaac	Ответ верный
5.	bcac	ccac	Ответ верный
6.	abababa	abababa	Ответ верный (нет 'с')
7.	bcbcacb	bcbcacb	Ответ верный (нет 'а')

Выводы.

Были изучены устройство и принцип работы машины Тьюринга, а также возможности её реализации в Python.

Разработана программа, выполняющая считывание ленты с клавиатуры. Для хранения состояний машины использовался словарь, а для “автоматизации” цикл *while*, в теле которого изменялось состояние, положение каретки на ленте и значение ячейки в соответствии с таблицей состояний. Во избежание возникновения исключительных ситуаций при отсутствии искомых символов было добавлено состояние для поиска начала слова и обработка пробела для состояний, ищущих первое вхождение символа.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: main.py

```
L, N, R = -1, 0, 1
table = {
    'q0': {'a': ['a', R, 'q2'], 'b': ['b', R, 'q1'], 'c': ['c', R, 'q1'],
    ' ': [' ', R, 'q0']},
    'q1': {'a': ['a', R, 'q2'], 'b': ['b', R, 'q1'], 'c': ['c', R, 'q1'],
    ' ': [' ', N, 'q12']},
    'q2': {'a': ['a', L, 'q3'], 'b': ['b', L, 'q4'], 'c': ['c', L, 'q5'],
    ' ': [' ', L, 'q3']},
    'q3': {'a': ['a', L, 'q3'], 'b': ['b', L, 'q3'], 'c': ['c', L, 'q3'],
    ' ': [' ', R, 'q6']},
    'q4': {'a': ['a', L, 'q4'], 'b': ['b', L, 'q4'], 'c': ['c', L, 'q4'],
    ' ': [' ', R, 'q7']},
    'q5': {'a': ['a', L, 'q5'], 'b': ['b', L, 'q5'], 'c': ['c', L, 'q5'],
    ' ': [' ', R, 'q8']},
    'q6': {'a': ['a', R, 'q6'], 'b': ['b', R, 'q6'], 'c': ['c', L, 'q9'],
    ' ': [' ', N, 'q12']},
    'q7': {'a': ['a', R, 'q7'], 'b': ['b', R, 'q7'], 'c': ['c', L, 'q10'],
    ' ': [' ', N, 'q12']},
    'q8': {'a': ['a', R, 'q8'], 'b': ['b', R, 'q8'], 'c': ['c', L, 'q11'],
    ' ': [' ', N, 'q12']},
    'q9': {'a': ['a', N, 'q12'], 'b': ['a', N, 'q12'], 'c': ['a', N,
    'q12'], ' ': ['a', N, 'q12']},
    'q10': {'a': ['b', N, 'q12'], 'b': ['b', N, 'q12'], 'c': ['b', N,
    'q12'], ' ': ['b', N, 'q12']},
    'q11': {'a': ['c', N, 'q12'], 'b': ['c', N, 'q12'], 'c': ['c', N,
    'q12'], ' ': ['c', N, 'q12']}
}
line = list(input())
state = 'q0'
position = 0

while state != 'q12':
    line[position], shift, state = table[state][line[position]]
    position += shift

print(''.join(line))
```