

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. 3343

Поддубный В.А

Преподаватель

Иванов Д. В.

Санкт-Петербург

2023

Цель работы

Целью работы является реализация машины Тьюринга на Python для моделирования работы вычислительного устройства.

Задание

Вариант 2

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}.

			a	c	c	a	b	c	b	a	b	a	a	c	a	b			
--	--	--	---	---	---	---	---	---	---	---	---	---	---	---	---	---	--	--	--

Напишите программу, которая заменяет в исходной строке символ, идущий после последних двух встретившихся символов 'a', на предшествующий им символ (гарантируется, что это не пробел). Наличие в строке двух подряд идущих символов 'a' гарантируется.

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:

			c	c	c	a	b	c	b	a	b	a	a	c	a	b			
--	--	--	---	---	---	---	---	---	---	---	---	---	---	---	---	---	--	--	--

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 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.

Ваша программа должна вывести полученную ленту после завершения работы.

Выполнение работы

Таблица 1 - Таблица состояний

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

Описание состояний:

- q1 - начальное состояние, которое необходимо, чтобы найти первый встретившийся символ 'a'.
- q2 – состояние, которое определяет символ, следующий за первым встретившимся символом 'a'.
- q3 – состояние, которое определяет символ после двух символов 'a' и отправляет на соответствующий трек состояний.
- q4 - состояние для 'b', которое совершает сдвиг влево.
- q5 - состояние для 'b', которое совершает сдвиг влево.
- q6 – состояние, которое заменяет символ на 'b'.
- q7 - состояние для 'a', которое совершает сдвиг влево.
- q8 - состояние для 'a', которое совершает сдвиг влево.
- q9 – состояние, которое заменяет символ на 'a'.
- q10 - состояние для 'c', которое совершает сдвиг влево.

- q11 - состояние для 'с', которое совершает сдвиг влево.
- q12 – состояние, которое заменяет символ на 'с'.
Принцип работы Машины Тьюринга в коде:
- arr – введенная строка;
- states – таблица состояний, заданная словарем;
- state – текущее состояние, изначально q1;
- i – индекс текущей ячейки, изначально 0;
- С помощью цикла while и таблицы состояний строка изменяется согласно условию.

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.	abcaabc	abcaacc	Программа работает корректно
2.	aabbaa	aabbaab	Программа работает корректно

Выводы

Была реализована машины Тьюринга на Python для моделирования работы вычислительного устройства.

С помощью словаря была создана таблица состояний, а с помощью цикла while симитирована работа машины Тьюринга.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: main.py

```
L = -1
N = 0
R = 1

states = {
    # start parsing
    'q1': {'a': ['a', R, 'q2'], 'b': ['b', R, 'q1'], 'c': ['c', R,
'q1'], ' ': [' ', R, 'q1']},
    'q2': {'a': ['a', R, 'q3'], 'b': ['b', R, 'q1'], 'c': ['c', R,
'q1'], ' ': [' ', R, 'q1']},

    # founded
    'q3': {'a': ['a', L, 'q7'], 'b': ['b', L, 'q4'], 'c': ['c', L,
'q10'], ' ': [' ', L, 'q7']},

    # a
    'q7': {'a': ['a', L, 'q8']},
    'q8': {'a': ['a', L, 'q9']},
    'q9': {'a': ['a', R, 'qx'], 'b': ['a', R, 'qx'], 'c': ['a', R,
'qx'], ' ': ['a', R, 'qx']},
    # b
    'q4': {'a': ['a', L, 'q5']},
    'q5': {'a': ['a', L, 'q6']},
    'q6': {'a': ['b', R, 'qx'], 'b': ['b', R, 'qx'], 'c': ['b', R,
'qx'], ' ': ['b', R, 'qx']},
    # c
    'q10': {'a': ['a', L, 'q11']},
    'q11': {'a': ['a', L, 'q12']},
    'q12': {'a': ['c', R, 'qx'], 'b': ['c', R, 'qx'], 'c': ['c', R,
'qx'], ' ': ['c', R, 'qx']},
}

arr = list(input())[:-1]
state = "q1"
i = 0
space = list(' ')
arr = space + arr + space
while state != 'qx':
    current = arr[i];
    actions = states[state][current];
    arr[i] = actions[0]
    i += actions[1]
    if i == len(arr):
        arr = arr + space
        state = actions[2]
arr.pop(0)
arr.pop(len(arr) - 1)
print(*arr[:-1], sep='')
```