

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

Коршунов П.И.

Преподаватель

Иванов Д.В.

Санкт-Петербург

2023

Цель работы

Получить навык составления таблиц для машины Тьюринга, научиться работать с машиной Тьюринга.

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}. 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. 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.

Выполнение работы

Была считана входная строка, разбита на список и занесена в переменную *inp*. Было задано состояние *state=1*, начальный индекс *ind=0* и символ для замены *new_symbol=""*. Был создан словарь состояний *state* для машины Тьюринга. Ключами в это словаре являются состояния, а значениями словари с действиями состояния для каждого символа алфавита.

Состояния:

q1 – начальное состояние, находит начало строки

q2 – находит конец строки

q3 – находит первый с конца символ а

q4 – находит повтор символов а

q5 – считывает символ слева от повтора

q6 – заменяет символ справа от повтора на символ слева, если символ равен

“а”

q7 – заменяет символ справа от повтора на символ слева, если символ равен

“b”

q8 – заменяет символ справа от повтора на символ слева, если символ равен

“c”

q9 – конечное состояние

Был реализован цикл *while*, который работал до конечного состояния. На каждой итерации цикла проверялось: если состояние равнялось 4, то обновлялся *new_symbol* на значение слева от повтора, если состояние равнялось 5, то происходило обновление символа справа от повтора на символ слева от повтора. Также на каждой итерации цикла индекс ленты увеличивался или уменьшался согласно правилам словаря *states*. После цикла отредактированный список выводился на экран с помощью *print("".join(inp))*.

Таблица состояний представлена в табл. 1

Таблица 1 — Таблица состояний

	‘ ’	‘a’	‘b’	‘c’
q1	' ', 1, q1	'a', 1, q2	'b', 1, q2	'c', 1, q2
q2	' ', -1, q3	'a', 1, q2	‘b’, 1, q2	‘c’, 1, q2
q3		' a', -1, q4	‘b’, -1, q3	‘c’, -1, q3
q4		' a', -1, q5	‘b’, -1, q3	'c', -1, q3
q5		'a', 1, q6	'b', 1, q7	'c', 1, q8

q6	'a', 0, q9	'a', 1, q6	'a', 0, q9	'a', 0, q9
q7	'b', 0, q9	'a', 1, q7	'b', 0, q9	'b', 0, q9
q8	'c', 0, q9	'a', 1, q8	'c', 0, q9	'c', 0, q9

Тестирование.

Результаты тестирования представлены в табл. 1.

Таблица 1 – Результаты тестирования

№ п/п	Входные данные	Выходные данные	Комментарии
1.	abcaabc	abcaacc	-
2.	abcabcbaa	abcabcbaab	-

Выводы

Был получен навык составления таблиц для машины Тьюринга. Было получено знание о работе машины Тьюринга.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: Korshunov_Petr_lb3.py

```
inp = list(input())
state = 1
ind = 0
states = {
    1: {" ": [' ', 1, 1], "a": ['a', 1, 2], "b": ['b', 1, 2], "c": ['c',
1, 2]},
    2: {" ": [' ', -1, 3], "a": ['a', 1, 2], "b": ['b', 1, 2], "c": ['c',
1, 2]},
    3: {"a": ['a', -1, 4], "b": ['b', -1, 3], "c": ['c', -1, 3]},
    4: {"a": ['a', -1, 5], "b": ['b', -1, 3], "c": ['c', -1, 3]},
    5: {"a": ['a', 1, 6], "b": ['b', 1, 7], "c": ['c', 1, 8]},
    6: {" ": ['a', 0, 9], "a": ['a', 1, 6], "b": ['a', 0, 9], "c": ['a',
0, 9]},
    7: {" ": ['b', 0, 9], "a": ['a', 1, 7], "b": ['b', 0, 9], "c": ['b',
0, 9]},
    8: {" ": ['c', 0, 9], "a": ['a', 1, 8], "b": ['c', 0, 9], "c": ['c',
0, 9]},
}
while state != 9:
    symbol, add, state = states[state][inp[ind]]
    inp[ind] = symbol
    ind += add
print(''.join(inp))
```