

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №1
по дисциплине «Информатика»
Тема: Машина Тьюринга

Студент гр. 2381

Тищенко А. М.

Преподаватель

Шевская Н. В.

Санкт-Петербург

2022

Цель работы.

Изучить 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.

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.

Основные теоретические положения.

Состояние	a	b	c	“ “	Описание состояния
q0	a R q1	b R q2	c R q1	“ “ R q0	Начальное, находит начало слова
q1	a R q1	b R q2	c R q1	“ “ L q4	Находит первое <i>b</i>
q2	“ “ R q3	“ “ R q3	“ “ R q3	“ “ L q6	Удаляет 1-ый символ после <i>b</i>
q3	“ “ R q7	“ “ R q7	“ “ R q7		Удаляет 2-ой символ после <i>b</i>
q4	a L q4		c L q4	“ “ R q5	Возвращается в начало слова
q5	“ “ R q16		“ “ R q16		Удаляет первый символ слова
q6		“ “ N q16			удаляет <i>b</i> в конце слова
q7	“ “ L q8	“ “ L q12	“ “ L q14	“ “ N q16	Начинает сдвиг символа
q8				“ “ L q9	сдвиг автомата
q9				a R q10	Вставка <i>a</i>
q10				“ “ R q11	Возвращение автомата
q11				“ “ R q7	
q12				“ “ L q13	сдвиг автомата
q13				b R q10	Вставка <i>b</i>
q14				“ “ L q15	сдвиг автомата
q15				c R q10	Вставка <i>c</i>

q16 – Конечное состояние

Выполнение работы.

Переменные: целочисленные R , L , N , для перемещения автомата, словарь *table* для хранения таблицы состояний.

Функция *turing* в соответствии с программой обрабатывает ленту, так же как и машина Тьюринга.

Разработанный программный код см. в приложении А.

Выводы.

Была изучена машина Тьюринга.

Разработана функция, симулирующая её работу, создана таблица состояний в удаляющая символы в ленте в соответствии с заданием.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: main.py

```

R = +1
L = -1
N = 0
table = {
    'q0': {'a': ('a', R, 'q1'), 'b': ('b', R, 'q2'), 'c': ('c', R, 'q1'),
    ' ': (' ', R, 'q0')},
    'q1': {'a': ('a', R, 'q1'), 'b': ('b', R, 'q2'), 'c': ('c', R, 'q1'),
    ' ': (' ', L, 'q4')},
    'q2': {'a': (' ', R, 'q3'), 'b': (' ', R, 'q3'), 'c': (' ', R, 'q3'),
    ' ': (' ', L, 'q6')},
    'q3': {'a': (' ', R, 'q7'), 'b': (' ', R, 'q7'), 'c': (' ', R, 'q7'),
    ' ': (' ', N, 'ex')},
    'q4': {'a': ('a', L, 'q4'), 'c': ('c', L, 'q4'),
    ' ': (' ', R, 'q5')},
    'q5': {'a': (' ', N, 'ex'), 'c': (' ', N,
'ex')},},
    'q6': {
        'b': (' ', N, 'ex')},},
    'q7': {'a': (' ', L, 'q8'), 'b': (' ', L, 'q12'), 'c': (' ', L,
'q14'), ' ': (' ', N, 'ex')},
    'q8': {
    ' ': (' ', L, 'q9')},
    'q9': {
    ' ': ('a', R, 'q10')},
    'q10': {
    ' ': (' ', R, 'q11')},
    'q11': {
    ' ': (' ', R, 'q7')},
    'q12': {
    ' ': (' ', L, 'q13')},
    'q13': {
    ' ': ('b', R, 'q10')},
    'q14': {
    ' ': (' ', L, 'q15')},
    'q15': {
    ' ': ('c', R, 'q10')},
    'ex': 'exit'}
def turing(program: dict, tape: list, start: str) -> (list, str):
    cell = 0
    state = start
    res = ''
    while True:
        res += state + ' '
        rows = program[state]
        if rows == 'exit': break
        cur_row = rows[tape[cell]]
        tape[cell] = cur_row[0]
        cell += cur_row[1]
        state = cur_row[2]
    return tape, res
print(*turing(table, list(input()), 'q0')[0], sep='')

```