

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. 2381

Мавликаев И.А.

Преподаватель

Шевская Н.В.

Санкт-Петербург

2022

Цель работы.

Цель работы – изучить механизм работы машины Тьюринга, написать программу, решающую поставленную задачу при помощи машины 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.

Выполнение работы.

Таблица 1 – Программа для машины Тьюринга

состояние	a	b	c	' ' (пробел)
Start	a,N,find b	b,N,find b	c,N,find b	','R,start
find b	a,R,find b	b,R,first letter	c,R,find b	','L,none b
none b	a,L,none b	-	c,L,none b	','R,delete_symbol
delete_symbol	','N,end	','N,end	','N,end	','N,end
first letter	a,R,second letter	b,R,second letter	c,R,second letter	','L,delete_symbol
second letter	a,R,third letter	b,R,third letter	c,R,third letter	','L,delete_symbol
third letter	a,L,A	b,L,B	c,L,C	','L,delete_s
A	a,L,A1	b,L,A1	c,L,A1	-
A1	a,R,first letter	a,R,first letter	a,R,first letter	-
B	a,L,B1	b,L,B1	c,L,B1	-
B1	b,R,first letter	b,R,first letter	b,R,first letter	-
C	a,L,C1	b,L,C1	c,L,C1	-
C1	c,R,first letter	c,R,first letter	c,R,first letter	-
delete_s	','L,delete_symbol	','L,delete_symbol	','L,delete_symbol	-

Описание состояний:

- “start”: начальное состояние, до тех пор, пока курсор не дойдёт до символов, не являющихся пробелом.
- “find_b”: проход по строке, если символ равен ‘b’, то переход в состояние “first_letter”, если нет ни одного символа, то переход в состояние “none_b”.
- “none_b”: проход обратно из конца строки в начало, переход в состояние “delete_symbol” для удаления первого символа в строке.
- “delete_symbol”: замена текущего символа на пробел и переход в завершающее состояние “end”
- “first_letter”: курсор на первом символе после символа, который ‘b’ или уже записанного символа при удалении. В случае, что символ –

пробел, переход в состояние “delete_symbol” для удаления последнего символа в строке. Иначе курсор переходит на один символ вправо в состояние “second_letter”

- “second_letter”: в случае, если символ равен пробелу, переход в состояние “delete_symbol” для удаления последнего символа в строке. Если символ равен ‘а’, переход в состояние “А” и сдвиг курсора на символ влево. Если символ равен ‘b’, переход в состояние “В” и сдвиг курсора на символ влево. Если символ равен ‘с’, переход в состояние “С” и сдвиг курсора на символ влево.
- “А”: переход в состояние “А1” и сдвиг курсора на индекс влево.
- “А1”: запись в текущий символ ‘а’, сдвиг курсора вправо и переход в состояние “first_letter”
- “В”: переход в состояние “В1” и сдвиг курсора на индекс влево.
- “В1”: запись в текущий символ ‘b’, сдвиг курсора вправо и переход в состояние “first_letter”
- “С”: переход в состояние “С1” и сдвиг курсора на индекс влево.
- “С1”: запись в текущий символ ‘с’, сдвиг курсора вправо и переход в состояние “first_letter”
- "delete_s" – замена на пробел текущего символа и переход в состояние “delete_symbol” для удаления предыдущего символа.

В переменных R,L,N находится сдвиг по ленте, соответственно 1,-1,0. В переменной table лежит словарь словарей – таблица состояний. В переменной tape лежит массив символов – лента машины Тьюринга. В переменной state лежит текущее состояние. Изначально state равно “start”. В переменной ind лежит индекс символа в массиве tape. Изначально он равен нулю. В цикле while до тех пор, пока состояние state не равно “end”, согласно таблице для машины Тьюринга происходит переход в следующее состояние, делается шаг в переменной ind и изменение текущего символа в строке соответственно текущему состоянию.

Разработанный программный код см. в приложении А.

Тестирование.

Результаты тестирования представлены в табл. 1.

Таблица 1 – Результаты тестирования

№ п/п	Входные данные	Выходные данные	Комментарии
1.	abcabc	abbc	Удаляет 2 символа после первого 'b'
2.	acccbc	acccb	Удаляет последний символ так как первое вхождение 'b' – предпоследний символ
3.	acccccb	accccc	Удаляет последний символ так как первое вхождение 'b' – последний символ
4.	aaaaaaca	aaaaaaca	Удаляет первый символ в строке так как 'b' в строке нет

Выводы.

Были исследованы принципы работы машины Тьюринга. Разработана программа, имитирующая работу машины Тьюринга, выполняющая удаление двух символов в строке после первого символа 'b' (если 'b' нет, удаляется первый символ в строке, если первое вхождение 'b' последний символ в строке, он удаляется, если предпоследний – удаляется последний). Для выполнения программы были использованы словарь словарей и цикл for.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: main.py

```
R, N, L = 1, 0, -1
table = {
    "start": {'a': ['a', N, "find_b"], 'b' : ['b', N, "find_b"],
    'c' : ['c', N, "find_b"], ' ' : [' ', R, "start"]},
    "find_b" : {'a': ['a', R, "find_b"], 'b' : ['b', R,
"first_letter"], 'c' : ['c', R, "find_b"], ' ' : [' ', L, "none_b"]},
    "none_b" : {'a': ['a', L, "none_b"], 'c' : ['c', L, "none_b"],
    ' ' : [' ', R, "delete_symbol"]},
    "delete_symbol" : {'a': [' ', N, "end"], 'b' : [' ', N, "end"],
    'c' : [' ', N, "end"], ' ' : [' ', N, "end"]},
    "first_letter" : {'a': ['a', R, "second_letter"], 'b' : ['b',
R, "second_letter"], 'c' : ['c', R, "second_letter"], ' ' : [' ', L,
"delete_symbol"]},
    "second_letter" : {'a': ['a', R, "third_letter"], 'b' : ['b',
R, "third_letter"], 'c' : ['c', R, "third_letter"], ' ' : [' ', L,
"delete_symbol"]},
    "third_letter" : {'a': ['a', L, "A"], 'b' : ['b', L, "B"],
    'c' : ['c', L, "C"], ' ' : [' ', L, "delete_s"]},
    "A" : {'a': ['a', L, "A1"], 'b' : ['b', L, "A1"], 'c' : ['c',
L, "A1"]},
    "A1" : {'a': ['a', R, "first_letter"], 'b' : ['a', R,
"first_letter"], 'c' : ['a', R, "first_letter"]},
    "B" : {'a': ['a', L, "B1"], 'b' : ['b', L, "B1"], 'c' : ['c',
L, "B1"]},
    "B1" : {'a': ['b', R, "first_letter"], 'b' : ['b', R,
"first_letter"], 'c' : ['b', R, "first_letter"]},
    "C" : {'a': ['a', L, "C1"], 'b' : ['b', L, "C1"], 'c' : ['c',
L, "C1"]},
    "C1" : {'a': ['c', R, "first_letter"], 'b' : ['c', R,
"first_letter"], 'c' : ['c', R, "first_letter"]},
    "delete_s" : {'a': [' ', L, "delete_symbol"], 'b' : [' ', L,
"delete_symbol"], 'c' : [' ', L, "delete_symbol"]}
}
tape = list(input())
state = "start"
ind = 0
while state != "end":
    symbol = tape[ind];
    tape[ind] = table[state][symbol][0]
    ind += table[state][symbol][1]
    state = table[state][symbol][2]
print("".join(tape))
```