

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

ОТЧЕТ
по лабораторной работе №3
по дисциплине «Информатике»

Тема: Машина Тьюринга и конечные автоматы

Студент гр. 3384

Рудаков А.Л

Преподаватель

Иванов Д.В.

Санкт-Петербург

2023

Цель работы.

Изучить принцип работы машины Тьюринга. Применить навыки на практике, написав код, для решения поставленных задач.

Задание.

Вариант 4

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}, которая начинается с символа 'a'.

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

Напишите программу, которая оборачивает исходную строку.

Результат работы алгоритма - исходная последовательность символов в обратном порядке.

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:

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

Алфавит (можно расширять при необходимости):

⑩ a

⑩ b

⑩ c

⑩ " " (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 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.
6. Нельзя использовать дополнительную ленту, в которую записывается результат.

Ваша программа должна вывести полученную ленту после завершения работы.

В отчет включите таблицу состояний. Отдельно кратко опишите каждое состояние, например:

q1 - начальное состояние, которое необходимо, чтобы обнаружить конец строки.

Выполнение работы.

Для решения поставленной задачи был создан *state* - словарь, в котором ключи — состояния, в которых находится машина Тьюринга, а значения — словари, в которых ключи — символы, встречающиеся в строке, а значения — кортежи, содержащие в себе 3 значения: символ, который нужно записать в данную ячейку, направление сдвига и состояние, в которое должна перейти машина Тьюринга.

В алфавит были добавлены заглавные символы А, В и С. Смысл программы в том, что она проходит по слову от края до края, и меняет буквы местами, заменяя их заглавными. В конце, она меняет все заглавные на строчные.

	a	b	c	A	B	C	''
q1	A, r, q2						' ', r, q1
q2	a, r, q2	b, r, q2	c, r, q2				' ', l, q3
q3	A, l, q4	A, l, a5	A, l, q6	a, n, q22			
q4	a, l, q4	b, l, q4	c, l, q4	A, r, q7			
q5	a, l, q5	b, l, q6	c, l, q6	B, r, q7			
q6	a, l, q5	b, l, q6	c, l, q6	C, r, q7			
q7	a, r, q8	b, r, q9	c, r, q10	A, r, q20	B, r, q20	C, r, q20	
q8	a, r, q8	b, r, q9	c, r, q10	A, l, q11	B, l, a11	C, l, q11	
q9	a, r, q9	b, r, q9	c, r, q9	A, l, q12	B, l, q12	C, l, q12	
q10	a, r, q10	b, r, q10	c, r, q10	A, l, q13	B, l, q13	C, l, q13	
q11	A, l, q14	A, l, q15	A, l, q16				
q12	B, l, q14	B, l, q15	B, l, q16				
q13	C, l, q14	C, l, q15	C, l, q16				
q14	a, l, q14	b, l, q14	c, l, q14	A, r, q17	B, r, q17	C, r, q17	
q15	a, l, q15	b, l, a15	c, l, q15	A, r, q18	B, r, q18	C, r, q18	
q16	a, l, q16	b, l, q16	c, l, q16	A, r, q19	B, r, q19	C, r, q19	
q17	A, r, q7	A, r, q7	A, r, q7	A, r, q20	B, r, q20	C, r, q20	
q18	B, r, q7	B, r, q7	B, r, q7	A, r, q20	B, r, q20	C, r, q20	
q19	C, r, q7	C, r, q7	C, r, q7	A, r, q20	B, r, q20	C, r, q20	
q20				A, r, q20	B, r, q20	C, r, q20	' ', l, q21
q21				a, l, q21	b, l, q21	c, l, q21	' ', n, q22

q1 — начальное состояние, которое необходимо, чтобы дойти до начала строки и заменить первую букву а на А.

q2 — состояние, которое необходимо, чтобы дойти до конца строки.

q3 — состояние, которое необходимо, чтобы заменить последнюю букву на А, и считать букву, для выбора следующего состояния состояния.

q4 — состояние, которое необходимо, чтобы дойти до начала строки, если конечной была буква а и заменить ее на А.

q5 — состояние, которое необходимо, чтобы дойти до начала строки, если конечной была буква b и заменить ее на B.

q6 — состояние, которое необходимо, чтобы дойти до начала строки, если конечной была буква с и заменить ее на C.

q7 — состояние, которое необходимо, чтобы считать крайнюю левую не переставленную букву и выбрать благодаря этому следующее состояние, если же буква уже была считана и поменяна, то переход в состояние q20.

q8 — состояние, которое необходимо, чтобы дойти до крайней правой замененной букве и сделать шаг влево, если считанная буква была a.

q9 — состояние, которое необходимо, чтобы дойти до крайней правой замененной букве и сделать шаг влево, если считанная буква была c.

q10 — состояние, которое необходимо, чтобы дойти до крайней правой замененной букве и сделать шаг влево, если считанная буква была b.

q11 — состояние, которое необходимо, чтобы заменить крайнюю правую незамененную букву на A, если последняя считанная была a, и считать новую.

q12 — состояние, которое необходимо, чтобы заменить крайнюю правую незамененную букву на B, если последняя считанная была b, и считать новую.

q13 — состояние, которое необходимо, чтобы заменить крайнюю правую незамененную букву на C, если последняя считанная была c, и считать новую.

q14 — состояние, которое необходимо, чтобы дойти до крайней левой замененной букве и сделать шаг вправо, если считанная буква была a.

q15 — состояние, которое необходимо, чтобы дойти до крайней левой замененной букве и сделать шаг вправо, если считанная буква была c.

q16 — состояние, которое необходимо, чтобы дойти до крайней левой замененной букве и сделать шаг вправо, если считанная буква была b.

q17 — состояние, которое необходимо, чтобы заменить крайнюю левую незамененную букву на A, если последняя считанная была a, и считать новую, если встретилась считанная буква, то перейти в состояние q20.

q18 — состояние, которое необходимо, чтобы заменить крайнюю левую незамененную букву на В, если последняя считанная была b, и считать новую, если встретилась считанная буква, то перейти в состояние q20.

q19 — состояние, которое необходимо, чтобы заменить крайнюю левую незамененную букву на С, если последняя считанная была с, и считать новую, если встретилась считанная буква, то перейти в состояние q20.

q20 — состояние, которое необходимо, чтобы дойти до конца слова, после того как все буквы были заменены.

q21 — состояние, которое необходимо, чтобы дойти до начала предложения, заменяя при этом заглавные буквы на строчные.

После инициализации словаря состояний была реализована основа программы. В переменную s была считана строка. Переменной *tek* присвоено текущее состояние q1. Инициализирован список *new*, элементами которого являются символы считанной строки. Счетчику индекса *i* присвоено значение 0. Далее идет цикл *while*, выполняющийся до момента, пока состояние машины не станет q22. Внутри него переменной *c* присваивается символ текущей ячейки ленты. переменным *ch*, *vect*, *tek* присваивается значение символа, на который надо заменить текущий символ, направление сдвига по ленте и новое состояние соответственно. Значения берутся из словаря. В списке текущему элементы присваивается новое значение. Далее идет условный оператор, если направление сдвига *r*, то индекс сдвигается вправо, если 1, то влево. В конце выводится лента в виде строки.

Тестирование.

Результаты тестирования представлены в табл. 1.

Таблица 1 – Результаты тестирования

№ п/п	Входные данные	Выходные данные	Комментарии
1.	abcab	bacba	OK
2.	a	a	OK

3.	abc	cba	OK
----	-----	-----	----

Выводы.

Был написан код, решающий поставленную задачу. Функционал машины Тьюринга реализован благодаря словарю словарей кортежей. Программа работает корректно.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: main_lb3.py

```

state={'q1':{'a':('A','r','q2'),' ':(' ','r','q1')}, #дойти до букв
      'q2':{'a':('a','r','q2'),'b':('b','r','q2'),'c':('c','r','q2'),' ':(' ','l','q3')}, #дойти до конца
      'q3':{'a':('A','l','q4'),'b':('A','l','q5'),'c':('A','l','q6'),'A':('a','n','q22')}, #выбор пути по последней букве
      'q4':{'a':('a','l','q4'),'b':('b','l','q4'),'c':('c','l','q4'),'A':('A','r','q7')}, #дойти до начала по а
      'q5':{'a':('a','l','q5'),'b':('b','l','q5'),'c':('c','l','q5'),'A':('B','r','q7')}, #дойти до начала по b
      'q6':{'a':('a','l','q6'),'b':('b','l','q6'),'c':('c','l','q6'),'A':('C','r','q7')}, #дойти до начала по с
      'q7':{'a':('a','r','q8'),'b':('b','r','q9'),'c':('c','r','q10'),'A':('A','r','q20'),'B':('B','r','q20'),'C':('C','r','q20')}, #считывание буквы
      'q8':{'a':('a','r','q8'),'b':('b','r','q8'),'c':('c','r','q8'),'A':('A','l','q11'),'B':('B','l','q11'),'C':('C','l','q11')}, #дойти до заглавной справа по А
      'q9':{'a':('a','r','q9'),'b':('b','r','q9'),'c':('c','r','q9'),'A':('A','l','q12'),'B':('B','l','q12'),'C':('C','l','q12')}, #дойти до заглавной справа по b
      'q10':{'a':('a','r','q10'),'b':('b','r','q10'),'c':('c','r','q10'),'A':('A','l','q13'),'B':('B','l','q13'),'C':('C','l','q13')}, #дойти до заглавной справа по с
      'q11':{'a':('A','l','q14'),'b':('A','l','q15'),'c':('A','l','q16')}, #замена на А и считывание буквы справа
      'q12':{'a':('B','l','q14'),'b':('B','l','q15'),'c':('B','l','q16')}, #замена на В и считывание буквы справа
      'q13':{'a':('C','l','q14'),'b':('C','l','q15'),'c':('C','l','q16')}, #замена на С и считывание буквы справа
      'q14':{'a':('a','l','q14'),'b':('b','l','q14'),'c':('c','l','q14'),'A':('A','r','q17'),'B':('B','r','q17'),'C':('C','r','q17')}, #дойти до заглавной слева по а
      'q15':{'a':('a','l','q15'),'b':('b','l','q15'),'c':('c','l','q15'),'A':('A','r','q18'),'B':('B','r','q18'),'C':('C','r','q18')}, #дойти до заглавной слева по b
      'q16':{'a':('a','l','q16'),'b':('b','l','q16'),'c':('c','l','q16'),'A':('A','r','q19'),'B':('B','r','q19'),'C':('C','r','q19')}, #дойти до заглавной слева по с
  
```

'q17': {'a': ('A', 'r', 'q7'), 'b': ('A', 'r', 'q7'), 'c': ('A', 'r', 'q7'), 'A': ('A', 'r', 'q20'), 'B': ('B', 'r', 'q20'), 'C': ('C', 'r', 'q20')}, #замена на A и переход к считыванию буквы слева

'q18': {'a': ('B', 'r', 'q7'), 'b': ('B', 'r', 'q7'), 'c': ('B', 'r', 'q7'), 'A': ('A', 'r', 'q20'), 'B': ('B', 'r', 'q20'), 'C': ('C', 'r', 'q20')}, #замена на B и переход к считыванию буквы слева

'q19': {'a': ('C', 'r', 'q7'), 'b': ('C', 'r', 'q7'), 'c': ('C', 'r', 'q7'), 'A': ('A', 'r', 'q20'), 'B': ('B', 'r', 'q20'), 'C': ('C', 'r', 'q20')}, #замена на C и переход к считыванию буквы слева

'q20': {'A': ('A', 'r', 'q20'), 'B': ('B', 'r', 'q20'), 'C': ('C', 'r', 'q20'), ' ': (' ', 'l', 'q21')}, #дойти до конца справа

'q21': {'A': ('a', 'l', 'q21'), 'B': ('b', 'l', 'q21'), 'C': ('c', 'l', 'q21'), ' ': (' ', 'n', 'q22')} } #дойти до начала слева, заменяя заглавные на строчные буквы }

```
s=input()
tek='q1'
sost=[tek]
new=list(s)
i=0
while tek!='q22':
    c=new[i]
    ch,vect,tek=state[tek][c]
    new[i]=ch
    if vect=='r':
        i+=1
    elif vect=='l':
        i-=1
    sost.append(tek)
print(''.join(new))
```