**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 |  | Ханнанов А.Ф. |
| Преподаватель |  | Иванов Д.В. |

Санкт-Петербург

2023

## Цель работы.

Получить навыки составления таблицы машины Тьюринга и работы с машиной Тьюринга.

## Задание.

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'.**

**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:

Алфавит (можно расширять при необходимости):

* 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.

Составлена таблица состояний. После идёт цикл, который останавливается при достижении машиной конечного состояния. В переменные state, pos, symb записываются текущее состояние, позиция и символ, который нужно записать в ленту, соответственно.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | a | b | c | D | « » |
| q1 | a;N;q2 |  |  |  | « »;R;q1 |
| q2 | D;L;qa | D;L;qb | D;L;qc | D;R;q2 | « »;L;q4 |
| qa | a;L;qa | b;L;qa | c;L;qa | D;L;qa | a;R;q3 |
| qb | a;L;qb | b;L;qb | c;L;qb | D;L;qb | b;R;q3 |
| qc | a;L;qc | b;L;qc | c;L;qc | D;L;qc | c;R;q3 |
| q3 | a;R;q3 | b;R;q3 | c;R;q3 | D;R;q2 |  |
| q4 | a;N;q5 |  |  | « »;L;q4 |  |

q1 — Начальное состояние, нужно для нахождения начала строки

q2 — Заменяет найденный символ на временный «D» и переходит к состоянию, которое соответствует символу; если видит пробел, то переходит к состоянию q4

qa — Проходит влево и вставляет «a» на место первого найденного пробела

qb — Проходит влево и вставляет «b» на место первого найденного пробела

qc — Проходит влево и вставляет «c» на место первого найденного пробела

q3 — Проходит вправо до первого символа «D»

q4 — Удаляет символы «D»

q5 — Конечное состояние

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 |
|  | abcabc | cbacba | - |
|  | abacbbc | cbbcaba | - |

## Выводы.

Были получены навыки работы с машиной Тьюринга.

# Приложение А Исходный код программы

Название файла: main.py

table = {'q1': {'a': ('a', 0, 'q2'), ' ': (' ', 1, 'q1')},  
 'q2': {'a': ('D', -1, 'qa'), 'b': ('D', -1, 'qb'), 'c': ('D', -1, 'qc'), 'D': ('D', 1, 'q2'), ' ': (' ', -1, 'q4')},  
 'qa': {'a': ('a', -1, 'qa'), 'b': ('b', -1, 'qa'), 'c': ('c', -1, 'qa'), 'D': ('D', -1, 'qa'), ' ': ('a', 1, 'q3')},  
 'qb': {'a': ('a', -1, 'qb'), 'b': ('b', -1, 'qb'), 'c': ('c', -1, 'qb'), 'D': ('D', -1, 'qb'), ' ': ('b', 1, 'q3')},  
 'qc': {'a': ('a', -1, 'qc'), 'b': ('b', -1, 'qc'), 'c': ('c', -1, 'qc'), 'D': ('D', -1, 'qc'), ' ': ('c', 1, 'q3')},  
 'q3': {'a': ('a', 1, 'q3'), 'b': ('b', 1, 'q3'), 'c': ('c', 1, 'q3'), 'D': ('D', 1, 'q2')},  
 'q4': {'a': ('a', 0, 'q5'), 'D': (' ', -1, 'q4')}}  
  
strip = [' '] \* 15  
strip += [x for x in input()] # лента  
strip += [' '] \* 15  
state = 'q1' # состояние  
pos = 0 # позиция  
  
while state != 'q5':  
 symb, step, state = table[state][strip[pos]]  
 strip[pos] = symb  
 pos += step  
  
print(''.join(strip))