**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 и конечные автоматы**

|  |  |  |
| --- | --- | --- |
| Студент гр. 3344 |  | Бубякина Ю.В. |
| Преподаватель |  | Иванов Д.В. |

Санкт-Петербург

2023

## Цель работы

Изучение принципа работы машины Тьюринга и конечных автоматов. Применение машины Тьюринга на практике.

## 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):

* 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.**

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

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

## Выполнение работы

В первую очередь в переменную *program* был записан словарь, содержащий в себе каждое состояние машины Тьюринга в виде ключей и алгоритмы действий для каждого состояния в виде значений. Сами значения этих ключей – также словари, содержащие в себе алгоритмы действий для каждого возможного символа в ячейке при любом состоянии машины Тьюринга, такие как: символ, записываемый в ячейку, шаг по индексу (влево, вправо, остаться на месте), переход в следующее состояние. Переменная *tape* содержит в себе список, состоящий из символов входной строки (лента). Переменные *state* и *index* содержат в себе начальное состояние и начальный индекс, соответственно.

Немного о состояниях:

q0 – начальное состояние, находит символ «a», с которого начинается строка

qp – замена символа на «p»

q1 – возвращение к следующему не замененному символу

qa – запись «a» в начало строки, если замененный символ – «a»

qb – запись «b» в начало строки, если замененный символ – «b»

qc – запись «c» в начало строки, если замененный символ – «c»

q2 – удаление всех «p» после переворота строки

q3 – конечное состояние

Далее используется цикл *while*, который, используя данные о текущем состоянии машины Тьюринга, а именно состояния и индекса просматриваемой ячейки, в переменные *symbol, move\_to* и *state\_new* записывает новый символ, шаг по индексу, следующее состояние для машины, соответственно. При переходе в состояние q3 машина останавливается, выход из цикла. Таким образом, получается инвертированная строка, которая выводится программой.

Таблица состояний представлена в табл. 1

Таблица 1 — Таблица состояний

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | ‘ ’ | ‘a’ | ‘b’ | ‘c’ | p |
| q0 | ‘ ’; 1; ‘q0’ | ‘a’; 0; ‘qp’ | - | - | - |
| qp | ‘ ’; -1; ’q2’ | ‘p’; -1; ’qa’ | ‘p’; -1; ’qb’ | ‘p’; -1; ’qc’ | ‘p’; 1; ’qp’ |
| qa | ‘a’; 1; ’q1’ | ‘a’; -1; ’qa’ | ‘b’; -1; ’qa’ | ‘c’; -1; ’qa’ | ‘p’; -1; ’qa’ |
| qb | ‘b’; 1; ’q1’ | ‘a’; -1; ’qb’ | ‘b’; -1; ’qb’ | ‘c’; -1; ’qb’ | ‘p’;-1; ’qb’ |
| qc | ‘c’; 1; ‘q1’ | ‘a’; -1; ’qc’ | ‘b’; -1; ’qc’ | ‘c’; -1; ’qc’ | ‘p’; -1; ’qc’ |
| q1 | - | ‘a’; 1; ’q1’ | ‘b’; 1; ’q1’ | ‘c’; 1; ’q1’ | ‘p’; 1; ’qp’ |
| q2 | - | ‘a’; 0; ‘q3’ | - | - | ‘ ’; -1; ‘q2’ |

## Тестирование

Результаты тестирования представлены в табл. 2.

Таблица 2 – Результаты тестирования

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Входные данные | Выходные данные | Комментарии |
|  | abcabc | cbacba | - |
|  | abacbbc | cbbcaba | - |

## Выводы

Был освоен принцип работы машины Тьюринга. Был написан алгоритм для машины Тьюринга, инвертирующий входную строку.

# Приложение А Исходный код программы

Название файла: lb3.py

program={'q0': {'a': ('a', 0, 'qp'), ' ': (' ', 1, 'q0')},  
 'qp': {'a': ('p', -1, 'qa'), 'b': ('p', -1, 'qb'), 'c': ('p', -1, 'qc'), 'p': ('p', 1, 'qp'), ' ': (' ', -1, 'q2')},  
 'qa': {'a': ('a', -1, 'qa'), 'b': ('b', -1, 'qa'), 'c': ('c', -1, 'qa'), 'p': ('p', -1, 'qa'), ' ': ('a', 1, 'q1')},  
 'qb': {'a': ('a', -1, 'qb'), 'b': ('b', -1, 'qb'), 'c': ('c', -1, 'qb'), 'p': ('p', -1, 'qb'), ' ': ('b', 1, 'q1')},  
 'qc': {'a': ('a', -1, 'qc'), 'b': ('b', -1, 'qc'), 'c': ('c', -1, 'qc'), 'p': ('p', -1, 'qc'), ' ': ('c', 1, 'q1')},  
 'q1': {'a': ('a', 1, 'q1'), 'b': ('b', 1, 'q1'), 'c': ('c', 1, 'q1'), 'p': ('p', 1, 'qp')},  
 'q2': {'a': ('a', 0, 'q3'), 'p': (' ', -1, 'q2')}  
 }  
tape=list(' '\*15+input()+' '\*15)  
state='q0'  
index=0  
while state!='q3':  
 symbol,move\_to,state\_new=program[state][tape[index]]  
 tape[index]=symbol  
 index+=move\_to  
 state=state\_new  
print(''.join(tape))