**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 |  | Жаворонок Д.Н. |
| --- | --- | --- |
| Преподаватель |  | Иванов Д.В. |

Санкт-Петербург

2023

## Цель работы

Получить навык работы с машиной Тьюринга, научиться создавать таблицы состояний.

## Задание.

Вариант 2

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 заменяет в исходной строке символ, идущий после последних двух встретившихся символов 'a', на предшествующий им символ(гарантируется, что это не пробел). Наличие в строке двух подряд идущих символов 'a' гарантируется.

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 15.

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.

## Выполнение работы

Считываем *s* — ленту машины Тьюринга в список. Инициализируем текущее состояние *state* значением *"st1",* устанавливаем индекс элемента, на который указывает каретка *index = 0.* Описываем словарь состояний *states.*

st1 – начальное состояние, используемое для нахождения начала строки.

st2 – состояние, используемое для нахождения конца строки.

st3 – состояние, используемое для поиска последнего символа "a".

st4 – состояние, используемое для поиска предпоследнего символа "a".

st5 – состояние, используемое для определения символа перед "aa".

st6 – состояние, используемое для замены символа после "aa", на символ, стоящий перед "aa", в случае, если это символ "a".

st7 – состояние, используемое для замены символа после "aa", на символ, стоящий перед "aa", в случае, если это символ "b".

st8 – состояние, используемое для замены символа после "aa", на символ, стоящий перед "aa", в случае, если это символ "c".

st9 – терминирующее состояние.

Описываем цикл *while,* работающий до тех пор, пока состояние *state* не станет *st9.* Из *states* распаковываем символ для замены *symbol,* шаг индекса каретки *step* и новое состояние *state* в соответсвии с текущим положением*.*

После завершения цикла на экран выводим изменённую строку. Ниже Таблица 1 – Таблица состояний.

Таблица 1 - Таблица состояний

|  | «a» | «b» | «c» | « » |
| --- | --- | --- | --- | --- |
| st1 | "a", 1, "st2" | "b", 1, "st2" | "c", 1, "st2" | " ", 1, "st1" |
| st2 | "a", 1, "st2" | "b", 1, "st2" | "c", 1, "st2" | " ", -1, "st3" |
| st3 | "a", -1, "st4" | "b", -1, "st3" | "c", -1, "st3" |  |
| st4 | "a", -1, "st5" | "b", -1, "st3" | "c", -1, "st3" |  |
| st5 | "a", 1, "st6" | "b", 1, "st7" | "c", 1, "st8" |  |
| st6 | "a", 1, "st6" | "a", 0, "st9" | "a", 0, "st9" | "a", 0, "st9" |
| st7 | "a", 1, "st7" | "b", 0, "st9" | "b", 0, "st9" | "b", 0, "st9"] |
| st8 | "a", 1, "st8" | "c", 0, "st9" | "c", 0, "st9" | "c", 0, "st9" |

## Тестирование.

Результаты тестирования представлены в табл. 2.

Таблица 2 - результаты тестирования.

| № п/п | Входные данные | Выходные данные | Комментарии |
| --- | --- | --- | --- |
|  | abcbcbaaa | abcbcbaaaa | - |
|  | abcccaaaaaac | abcccaaaaaaa | - |

## Выводы

В ходе выполнения работы был изучена работа машины Тьюринга. Была написана программа для обработки входной строки по принципу машины Тьюринга. Был получен навык составления таблиц для машины Тьюринга.

# ПРИЛОЖЕНИЕ А ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: main.py

s = list(input())

state = "st1"

index = 0

states = {

"st1": {"a": ["a", 1, "st2"], "b": ["b", 1, "st2"], "c": ["c", 1, "st2"], " ": [" ", 1, "st1"]},

"st2": {"a": ["a", 1, "st2"], "b": ["b", 1, "st2"], "c": ["c", 1, "st2"], " ": [" ", -1, "st3"]},

"st3": {"a": ["a", -1, "st4"], "b": ["b", -1, "st3"], "c": ["c", -1, "st3"]},

"st4": {"a": ["a", -1, "st5"], "b": ["b", -1, "st3"], "c": ["c", -1, "st3"]},

"st5": {"a": ["a", 1, "st6"], "b": ["b", 1, "st7"], "c": ["c", 1, "st8"]},

"st6": {"a": ["a", 1, "st6"], "b": ["a", 0, "st9"], "c": ["a", 0, "st9"], " ": ["a", 0, "st9"]},

"st7": {"a": ["a", 1, "st7"], "b": ["b", 0, "st9"], "c": ["b", 0, "st9"], " ": ["b", 0, "st9"]},

"st8": {"a": ["a", 1, "st8"], "b": ["c", 0, "st9"], "c": ["c", 0, "st9"], " ": ["c", 0, "st9"]}

}

while state != "st9":

symbol, step, state = states[state][s[index]]

s[index] = symbol

index += step

print("".join(s))