

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

ОТЧЕТ
по лабораторной работе №2
по дисциплине «Информатика»
Тема: Моделирование работы Машины Тьюринга.

Студент гр. 1304

Шаврин А.П.

Преподаватель

Берленко Т.А.

Санкт-Петербург

2021

Цель работы.

Научиться моделировать работу Машины Тьюринга

Задание.

На вход программе подается строка неизвестной длины. Каждый элемент является значением в ячейке памяти ленты Машины Тьюринга.

На ленте находится троичное число, знак (плюс или минус) и троичная цифра.

		1	2	1	+	2			
--	--	---	---	---	---	---	--	--	--

Напишите программу, которая выполнит арифметическую операцию. Указатель на текущее состояние Машины Тьюринга изначально находится слева от числа (но не на первом его символе). По обе стороны от числа находятся пробелы. Результат арифметической операции запишите на месте первого числа. Для примера выше лента будет выглядеть так:

		2	0	0	+	2			
--	--	---	---	---	---	---	--	--	--

Ваша программа должна вывести полученную ленту после завершения работы.

Алфавит:

0

1

2

+

-

" " (пробел)

Соглашения:

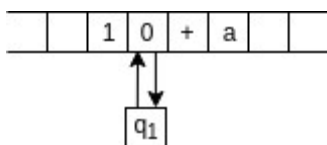
1. Направление движения автомата может быть одно из R (направо), L (налево), N (неподвижно).
2. Число обязательно начинается с единицы или двойки.
3. Числа и знак операции между ними идут непрерывно.
4. Гарантируется, что в результате операции вычитания не может получиться отрицательного числа.

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

q1 - начальное состояние, которое необходимо, чтобы найти первую цифру первого числа.

Основные теоретические положения.

Машина Тьюринга (МТ) состоит из двух частей: неподвижной бесконечной ленты (памяти) и автомата (процессора).



1. Лента используется для хранения информации. Она бесконечна в обе стороны и разбита на клетки, которые никак не нумеруются и не именуются. В

каждой клетке может быть записан один символ или ничего не записано. Память пассивна: она ничего не делает, просто хранит данные.

2. Алфавит ленты - конечное множество всех возможных символов ленты. Если предположить, что видимые символы - весь алфавит ленты из примера выше, то мы имеем следующий алфавит: $\{1, 0, +, 'a', ''\}$. Последний символ - пустой, означает пустое содержимое клетки.

3. Автомат – это активная часть Машины Тьюринга. В каждый момент он размещается под одной из клеток ленты и видит её содержимое; это видимая клетка, а находящийся в ней символ – видимый символ; содержимое же соседних и других клеток автомат не видит. Кроме того, в каждый момент автомат находится в одном из состояний, которые обычно обозначаются буквой q с номерами: q_0, q_1, q_2 и т.д. Существует конечное число таких состояний.

В каждом из состояний автомат выполняет какую-то конкретную операцию. Существует заключительное состояние, в котором автомат останавливается.

Автомат за один такт (шаг) может выполнить следующие действия :

1. считать видимый символ;
2. записывать в видимую клетку новый символ (в том числе пустой символ);
3. сдвигаться на одну клетку влево или вправо («перепрыгивать» сразу через несколько клеток автомат не может);
4. перейти в следующее состояние.
5. Один из способов записи программы для Машины Тьюринга является таблица следующего вида:
- 6.

	s_0	s_1	s_2	...	s_m
q_0					
q_1					
...					
q_n					

7.

$S_0, S_1, S_2, \dots, S_m$ - Алфавит ленты

q_0, q_1, \dots, q_n - Состояния автомата

8. В ячейках таблицы указываются тройка $\langle S', [L, R, N], q' \rangle$:

9.

10.1. S' - символ, который необходимо записать в видимую ячейку ленты

11.2. $[L, R, N]$ - одно из направлений, куда необходимо перейти на ленте: R - направо, L - налево, N - остаться на месте.

12.3. q' - состояние, в которое необходимо перейти автомату.

Выполнение работы.

В работе был использован алгоритм моделирования работы Машины Тьюринга при помощи функции *main* и представления таблицы состояний в виде словаря. Был изучен алгоритм сложения и вычитания в троичной системе счисления, согласно которому была составлена таблица состояний Машины Тьюринга. При проверке алгоритма на тестовых данных был получен верный ответ, что свидетельствует о корректном моделировании работы Машины Тьюринга.

Переменные:

tape(тип *list*) – моделирует ленту в Машине Тьюринга

table(тип *dict*) – является таблицей состояний

state(тип *str*) – характеризует состояние машины

i(тип *int*) – итерируемая переменная

symbol(тип *str*) – текущий символ на ленте

directions(тип *dict*) – словарь в котором ключ – *direction*, а значение число от -1 до 1

direction(тип *str*) – переменная, содержащая информация о том, в какую сторону перемещаться машине

В функции *main* реализована логика работы Машины Тьюринга.

Таблица состояний и описание состояний:

	‘ ‘	‘0’	‘1’	‘2’	‘+’	‘-’
q0	‘ ‘, ‘R’, ‘q0’	‘0’, ‘R’, ‘q0’	‘1’, ‘R’, ‘q0’	‘2’, ‘R’, ‘q0’	‘+’, ‘R’, ‘q1’	‘-’, ‘R’, ‘q1’
q1	-	‘0’, ‘N’, ‘q8’	‘1’, ‘L’, ‘q2’	‘2’, ‘L’, ‘q4’	-	-
q2	‘1’, ‘N’, ‘q8’	‘1’, ‘N’, ‘q8’	‘2’, ‘N’, ‘q8’	‘0’, ‘L’, ‘q2’	‘+’, ‘L’, ‘q2’	‘-’, ‘L’, ‘q3’
q3	-	‘2’, ‘L’, ‘q3’	‘0’, ‘L’, ‘q6’	‘1’, ‘N’, ‘q8’	-	-
q4	‘2’, ‘N’, ‘q8’	‘2’, ‘N’, ‘q8’	‘0’, ‘L’, ‘q2’	‘1’, ‘L’, ‘q2’	‘+’, ‘L’, ‘q4’	‘-’, ‘L’, ‘q5’
q5	-	‘1’, ‘L’, ‘q3’	‘2’, ‘L’, ‘q3’	‘0’, ‘L’, ‘q6’	-	-
q6	‘ ‘, ‘R’, ‘q7’	‘0’, ‘N’, ‘q8’	‘1’, ‘N’, ‘q8’	‘2’, ‘N’, ‘q8’	-	-
q7	‘0’, ‘N’, ‘q8’	‘ ‘, ‘R’, ‘q7’	‘1’, ‘N’, ‘q8’	‘2’, ‘N’, ‘q8’	‘+’, ‘L’, ‘q7’	‘-’, ‘L’, ‘q7’

q0 – начальное состояние, необходимое для поиска числа которое нужно прибавить/вычесть из данного

q1 – состояние определяющее какое число будет прибавляться/вычитаться

q2 – состояние прибавляющее единицу к числу(если же знак – то перевод в состояние q3)

q3 – состояние вычитающее единицу

q4 – состояние прибавляющее двойку к числу(если же знак – то перевод в состояние q5)

q5 – состояние вычитающее двойку

q6 – состояние проверяющее наличие ведущих нулей

q7 – состояние удаляющее ведущие нули(если же он был один, то запись его обратно)

q8 – конечное состояние завершающее работу Машины Тьюринга

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.	111+1	112+1	Успешный тест
2.	1-1	0-1	Успешный тест
3.	122+2	201+2	Успешный тест

4.	222+2	1001+2	Успешный тест
5.	202-2	200-2	Успешный тест

Выводы.

Были исследованы и изучены способы моделирования Машины Тьюринга.

Разработана программа, моделирующая работу Машины Тьюринга по сложению/вычитанию единицы/двойки из числа в троичной системе счисления.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: *main.py*

Моделирование работы Машины Тьюринга.

def main():

 # Начальное состояние Машины Тьюринга

global tape

directions = {'R': 1, 'L': -1, 'N': 0}

state = 'q0'

i = 0

while state != 'q8':

tape[i], direction, state = table[state][tape[i]]

i += directions[direction]

return tape

if __name__ == '__main__':

 # Считываем ленту.

tape = list(input())

 # Таблица состояний Машины Тьюринга.

table = {

 # Поиск цифры после знака.

'q0': {

' ': (' ', 'R', 'q0'), '0': ('0', 'R', 'q0'),

'1': ('1', 'R', 'q0'), '2': ('2', 'R', 'q0'),

'+' : ('+', 'R', 'q1'), '-' : ('-', 'R', 'q1')

},

 # Запоминаем число.

'q1': {

'0': ('0', 'N', 'q8'), '1': ('1', 'L', 'q2'),

'2': ('2', 'L', 'q4')

},

+1 или смена на -1 если будет знак "-".

'q2': {

' ': ('1', 'N', 'q8'), '0': ('1', 'N', 'q8'),

'1': ('2', 'N', 'q8'), '2': ('0', 'L', 'q2'),

'+' : ('+', 'L', 'q2'), '-' : ('-', 'L', 'q3')

},

-1.

'q3': {

'0': ('2', 'L', 'q3'), '1': ('0', 'L', 'q6'),

'2': ('1', 'N', 'q8')

},

+2 или смена на -2, если будет знак "-".

'q4': {

' ': ('2', 'N', 'q8'), '0': ('2', 'N', 'q8'),

'1': ('0', 'L', 'q2'), '2': ('1', 'L', 'q2'),

'+' : ('+', 'L', 'q4'), '-' : ('-', 'L', 'q5')

},

-2.

'q5': {

'0': ('1', 'L', 'q3'), '1': ('2', 'L', 'q3'),

'2': ('0', 'L', 'q6')

},

Проверка на наличие ведущего нуля.

'q6': {

' ': (' ', 'R', 'q7'), '0': ('0', 'N', 'q8'),

'1': ('1', 'N', 'q8'), '2': ('2', 'N', 'q8')

},

Удаление ведущего нуля.

'q7': {

' ': ('0', 'N', 'q8'), '0': (' ', 'R', 'q7'),

```
'1': ('1', 'N', 'q8'), '2': ('2', 'N', 'q8'),  
'+' : ('+', 'L', 'q7'), '-' : ('-', 'L', 'q7')  
}  
}  
print(''.join(main()))
```