In [1]:
```
### CSCI-3080 Discrete Structure
### OLA 6: Chapter 9 -- Finite-Sate Machine & Turing Machines
### Name: Ruth Dohrmann
### Student ID: M01577057
### Date: 11/25/2022
```

1. Please draw the state graph for the following finate state machine, and compute the output sequence for the given input sequence.

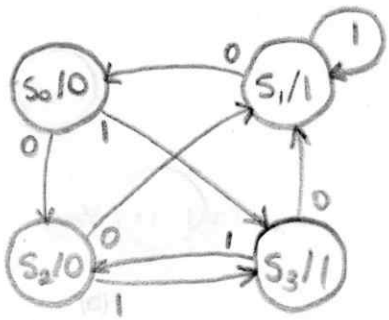
(a)

Input:0011

0011

Time	t_0	t_1	t_2	t_3	t_4
input	0	0	1	1	–
states	S_0	S_2	S_1	S_1	S_1
output	0	0	1	1	1

Present state	Next state		Output
	Present input		
	0	1	
s_0	s_2	s_3	0
s_1	s_0	s_1	1
s_2	s_1	s_3	0
s_3	s_1	s_2	1



In [ ]: Output: 0111

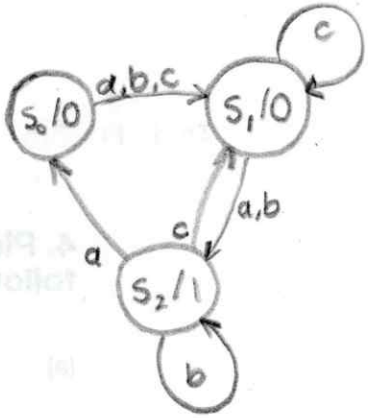
(b)

Input:acbbca

acbbca

Time	t_0	t_1	t_2	t_3	t_4	t_5	t_6
Input	a	c	b	b	c	a	–
States	S_0	S_1	S_1	S_2	S_2	S_1	S_2
Output	0	0	0	1	1	0	1

Present state	Next state			Output
	Present input			
	a	*b*	*c*	
s_0	s_1	s_1	s_1	0
s_1	s_2	s_2	s_1	0
s_2	s_0	s_2	s_1	1

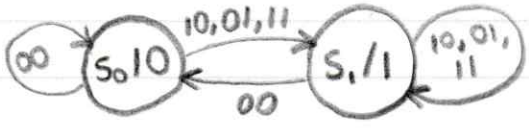


2. Output: 001101

(a) Please construct a finite-state machine that will compute the bitwise OR of two binary input string.

0 ∨ 0 = 0
1 ∨ 0 = 1
0 ∨ 1 = 1
1 ∨ 1 = 1

In [ ]:



(b) Write the output for the input sequence consisting of the strings 11011 and 10010 (read left to right)

In [ ]:

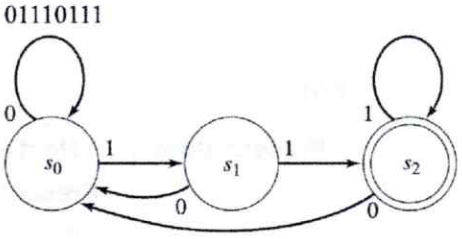
Time	t_0	t_1	t_2	t_3	t_4	t_5
Input	11	10	00	11	10	–
States	S_0	S_1	S_1	S_0	S_1	S_1
Output	0	1	1	0	1	1

Output: 11011

3. Determine whether the given machine recognizes the given input string.

(a)

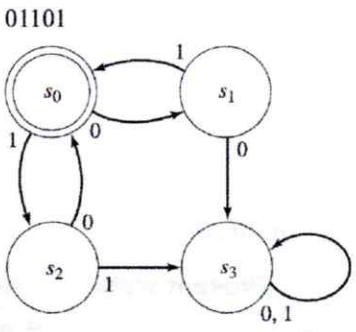
Input:01110111



In [ ]: Yes

(b)

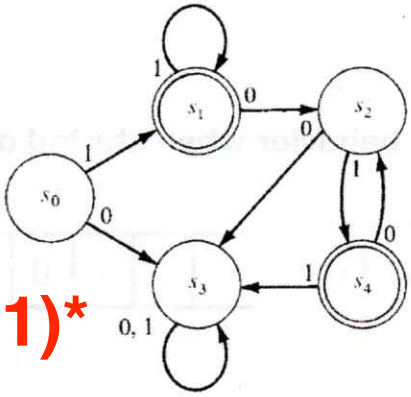
Input:01101



In [ ]: No

4. Please give a regular expression for the set recognized by the following finite-state machine.

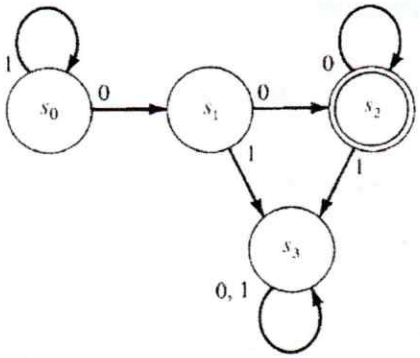
(a)



1(1*) v 1(1*)01(01)*

In [ ]: 1(1*) v 1(1*)01

(b)



In [ ]: (1*)00(0*)

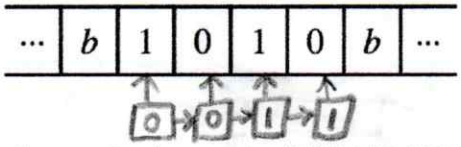
5. Consider the Turing Machine

(0, 1, 1, 0, R)
(0, 0, 0, 1, R)
(1, 1, 1, 1, R)
(1, b, 1, 2, L)
(2, 1, 1, 2, L)
(2, 0, 0, 2, L)
(2, b, 1, 0, R)

commands used:
(0,1,1,0,R)
(0,0,0,1,R)
(1,1,1,1,R)

(a). What is its behavior when started on the tape

The tape is left unchanged, ending with the state 1 and the symbol 0, which has no command.



5. Consider the Turing Machine

(0, 1, 1, 0, R)
(0, 0, 0, 1, R)
(1, 1, 1, 1, R)
(1, b, 1, 2, L)
(2, 1, 1, 2, L)
(2, 0, 0, 2, L)
(2, b, 1, 0, R)

(a). What is its behavior when started on the tape

···	*b*	1	0	1	0	*b*	···

...b 1 0 1 0 b ... (0, 1, 1, 0, R)

0

...b 1 0 1 0 b ... (0, 0, 0, 1, R)

0

...b 1 0 1 0 b ... (1, 1, 1, 1, R)

1

...b 1 0 1 0 b ... no command found, turing machine halts

1

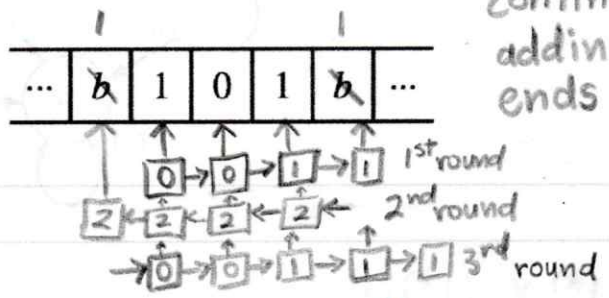
In [ ]:

(b). What is its behavior when started on the tape

The Turing machine continues forever, adding ones to the ends of the tape.

commands used:

(0,1,1,0,R)
(0,0,0,1,R)
(1,1,1,1,R)
(1,b,1,2,L)
(2,1,1,2,L)
(2,0,0,2,L)
(2,1,1,2,L)
(2,b,1,0,R)
(0,1,1,0,R)
(0,0,0,1,R)
(1,1,1,1,R)
(1,1,1,1,R)



6. Find a Turing machine that recognizes $\{0^n 1^{3n}, n \geq 0\}$.

Please take the reference from the lecgture we covered during the class:
https://lecture.yangxinmtsu.repl.co/3080/tm2.html
(https://lecture.yangxinmtsu.repl.co/3080/tm2.html)

In [ ]: State 6 is the only final state.

Turing machine:

(0,b,b,6,R) - recognize the empty tape
(0,0,X,1,R) - erase leftmost 0, move right
(1,0,0,1,R)
(1,1,1,1,R)
(1,b,b,2,L)
(1,X,X,2,L)
move right in state one until the end of the binary string is reached, move left in state 2
(2,1,X,7,L)
(7,1,X,8,L)
(8,1,X,3,L)
erase rightmost three ones, move left
(3,1,1,3,L) - move left over 1s
(3,0,0,4,L) - if there are more 0s, change to state 4
(3,X,X,5,R) - if there are no more 0s, change to state 5
(4,0,0,4,L) - move left over 0s
(4,X,X,0,R) - find end, begin again
(5,X,X,6,R) - no more 1s (being in state 5 means there are no more 0s), accept

5. Consider the Turing Machine

(0, 1, 1, 0, R)
(0, 0, 0, 1, R)
(1, 1, 1, 1, R)
(1, b, 1, 2, L)
(2, 1, 1, 2, L)
(2, 0, 0, 2, L)
(2, b, 1, 0, R)

(a). What is its behavior when started on the tape

···	*b*	1	0	1	0	*b*	···

...b 1 0 1 0 b ... (0, 1, 1, 0, R)

0

...b 1 0 1 0 b ... (0, 0, 0, 1, R)

0

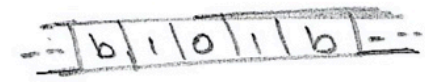
...b 1 0 1 0 b ... (1, 1, 1, 1, R)

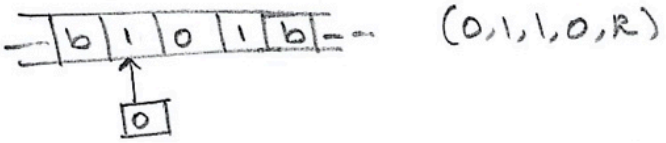
1

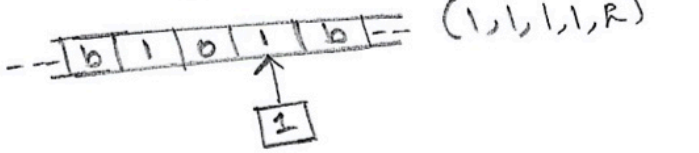
...b 1 0 1 0 b ... no command found, turing machine halts

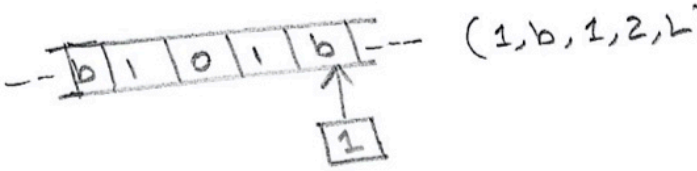
1

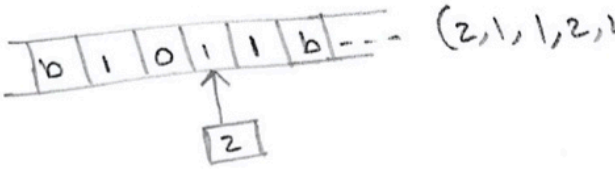
b.

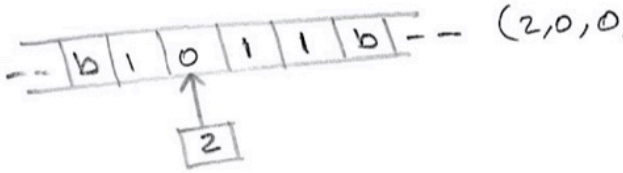


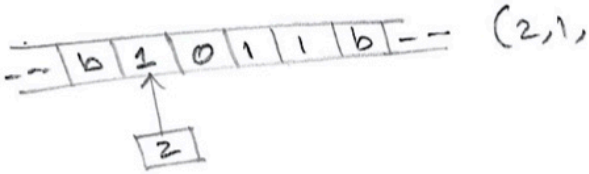


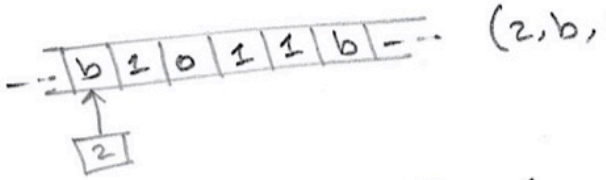


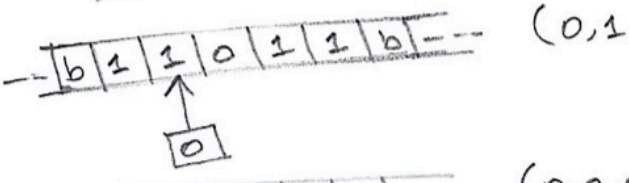


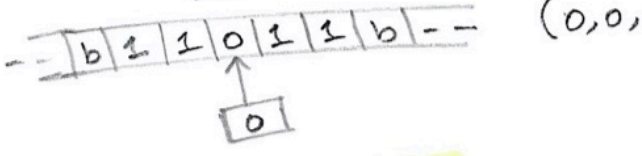












★ Turing machine doesn't halt and continuously add 1 on the right and left of the string on the tape.