Student Information

Full Name : Nazır Bilal Yavuz
Id Number : 2099471

Answer 1

a.

$S \to aSXaX$ —— (R1)
$S \to bSXbX$ —— (R2)
$S \to c$ —— (R3)
$X \to aX$ —— (R4)
$X \to bX$ —— (R5)
$X \to e$ —— (R6)

$(p, a, e), (p, a)$ for each $a \in \sum$ —— ($\triangle$1)

$(p, e, XaXSa), (p, S)$ —— ($\triangle$2)

$(p, e, XbXSb), (p, S)$ —— ($\triangle$3)

$(p, e, c), (p, S)$ —— ($\triangle$4)

$(p, e, Xa), (p, X)$ —— ($\triangle$5)

$(p, e, Xb), (p, X)$ —— ($\triangle$6)

$(p, e, e), (p, X)$ —— ($\triangle$7)

$(p, e, S), (p, e)$ —— ($\triangle$8)

b.

Table 1: Tracing abbcbabbaa

Step	State	Unread Input	Stack	Transition Used
0	p	abbcbabbaa	empty	
1	p	bbcbabbaa	a	1
2	p	bcbabbaa	ba	1
3	p	cbabbaa	bba	1
4	p	babbaa	cbba	1
5	p	babbaa	Sbba	4
6	p	babbaa	XSbba	7
7	p	abbaa	bXSbba	1
8	p	bbaa	abXSbba	1
9	p	bbaa	XabXSbba	7
10	p	bbaa	XbXSbba	5
11	p	bbaa	Sba	3
12	p	bbaa	XSba	7
13	p	baa	bXSba	1
14	p	aa	bbXSba	1
15	p	aa	XbbXSba	7
16	p	aa	XbXSba	6
17	p	aa	Sa	3
18	p	aa	XSa	7
19	p	a	aXSa	1
20	p	empty	aaXSa	1
21	p	empty	XaaXSa	7
22	p	empty	XaXSa	5
23	p	empty	S	2
24	p	empty	empty	8

Answer 2

a.

We can define the Turing Machine which computes $f(x)$ as $M = (K, \Sigma, \delta, s, H)$, where $K = \{s, e_0, o_0, e_1, e_2, e_3, o_1, o_2,$ $\Sigma = \{\sqcup, 1, \triangleright\}$, $H = \{h_1\}$ and the transition function is defined as follows;

$\delta_1(s, \sqcup) = (e_0, \sqcup, \rightarrow)$

$\delta_1(o_0, 1) = (e_0, x, \rightarrow)$
$\delta_1(o_0, \sqcup) = (o_1, \sqcup, \leftarrow)$
$\delta_1(o_1, x) = (o_1, x, \leftarrow)$
$\delta_1(o_1, \sqcup) = (o_2, \sqcup, \rightarrow)$
$\delta_1(o_2, x) = (o_2, 1, \rightarrow)$
$\delta_1(o_2, \sqcup) = (h_1, 1, \rightarrow)$

$\delta_1(e_0, 1) = (o_0, x, \rightarrow)$
$\delta_1(e_0, \sqcup) = (e_1, \sqcup, \leftarrow)$
$\delta_1(e_1, x) = (e_1, x, \leftarrow)$
$\delta_1(e_1, \sqcup) = (e_2, \sqcup, \rightarrow)$
$\delta_1(e_2, x) = (e_3, 1, \rightarrow)$
$\delta_1(e_3, x) = (e_3, x, \rightarrow)$
$\delta_1(e_3, \sqcup) = (e_4, \sqcup, \leftarrow)$
$\delta_1(e_4, x) = (e_5, \sqcup, \leftarrow)$
$\delta_1(e_5, x) = (e_5, x, \leftarrow)$
$\delta_1(e_5, 1) = (e_3, 1, \rightarrow)$
$\delta_1(e_3, \sqcup) = (h_1, \sqcup, \rightarrow)$

In this TM first go 1 from $\sqcup$ and check that it is even or odd while checking fill tape with x.

If it is odd go to the first x and make all x 1 and when you see $\sqcup$ make it 1 too.
This means $f(x) = x + 1$

If it is odd go to the first x make it 1 and go right until find $\sqcup$.
When you find $\sqcup$ go one left which is righmost x. After that, make this x to 1.
After that go leftmost x and make it 1.
Follow this pattern...
This means $f(x) = x/2$

Answer 3

If we can not go to the left of the Turing Machine that is not completely Turing Machine. We can not use full capability of Turing Machine so that this type of Turing Machine looks like DFA more than Turing Machine. This type of Turing Machines accept regular languages.

Answer 4

a.

We can define the Queue Based Turing Machine which computes
$f(x)$ as $M = (K, \Sigma, \delta, \top, \heartsuit, \clubsuit, \wp, \aleph, s, b, H)$, where;

K = set of states
Σ = input alphabet not containing $(\sqcup, \top, \clubsuit)$
δ = transition function
$\top$ = tape alphabet
$\heartsuit$ = queue symbol for at the start set rear and front
$\clubsuit$ = rear symbol
$\wp$ = front symbol
$\aleph$ = between left of head and right of head(explanation is in part d)
s = starting state, must be in K
b = blank symbol, must be in $\top$
H = set of final states

b.

Queue Based Turing Machine configuration is (q, w, u).
The current state is q.
w is the which $\wp_{(front\ symbol)}$ shows.
u is the which $\clubsuit_{(rear\ symbol)}$ shows.

c.

d.

In Queue based TM, front shows head of standard TM.
After that right of head comes.
After that $\aleph$.
After that left of head comes.
For example $abc\ d_{head}\ ef$ is in Queue based TM equals to $def\ \aleph\ abc$

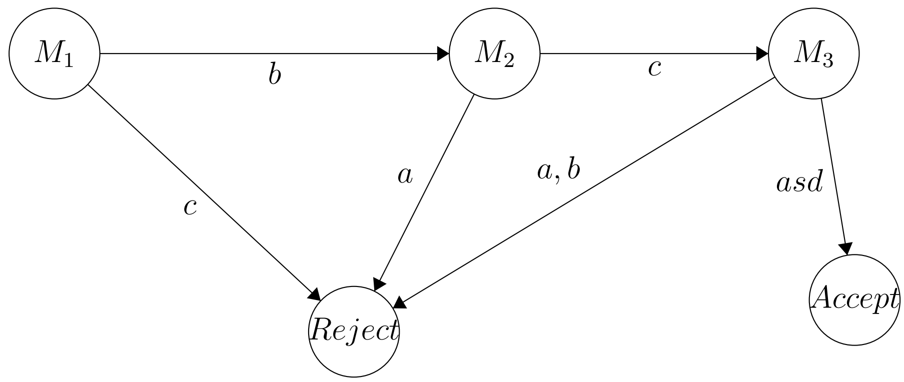
To make $d \to x, R$ dequeue front and enqueue x. At final state it looks like $ef\ \aleph\ abcx$.
To make $d \leftarrow x, R$ dequeue front and enqueue x. After that dequeue front and enqueue symbol which is dequeued. At final state it looks like $cxef\ \aleph\ ab$.

To access the rear element in the queue-based TM we can make $d \leftarrow d, L$
Front element in the queue-based TM is head of standard TM.

e.

Answer 5

a.



M_1, M_2, M_3 are TMs.
M_1 is for a^n
M_2 is for b^{2^n}.
M_3 is for c^{3n}.

b.

For M_1:
$S \to A \mid e$
$A \to AA \mid a \mid e$

For M_2:
$S \to QaZ$
$Q \to QM \mid e$
$Ma \to aaM$
$MZ \to Z$
$Z \to e$

For M_3
$S \to CCC \mid e \; CCC \to CCCCCC \mid ccc \mid e$

But I could not figure out how to combine them.

Answer 6

a.

Yes, there exist M_1, M_2, M_3, M_4, M_5 which represents L_1, L_2, L_3, L_4, L_5 respectively.

b.

c.

d.