

# AUTOMATA AND FORMAL LANGUAGES - (CS-208)

## TUTORIAL - 10

Q. 1 what do you understand by turing recognizability and turing decidability?

Ans:-

Turing recognizability is a property of any language if and only if there exist a turing machine such that

- when encountering a string in that language the machine terminates and accepts that string
- when encountering a string not in that language, the machine terminates the string or it rejects it or does not terminate at all.

Turing decidability is property of a language if and only if there exists a Turing machine such that

- when encountering a string in that language the machine terminates and accepts that string.
- when encountering a string not in that language the machine terminates and runs forever.

The "Turing decidability" is stronger criteria than "Turing recognisability" because if language is turing decidable then its turing machine does not run forever.

Q. 2

Design a Turing machine that accepts set of strings over  $\{0, 1\}$  with equal number of 0's and 1's.

Ans:-

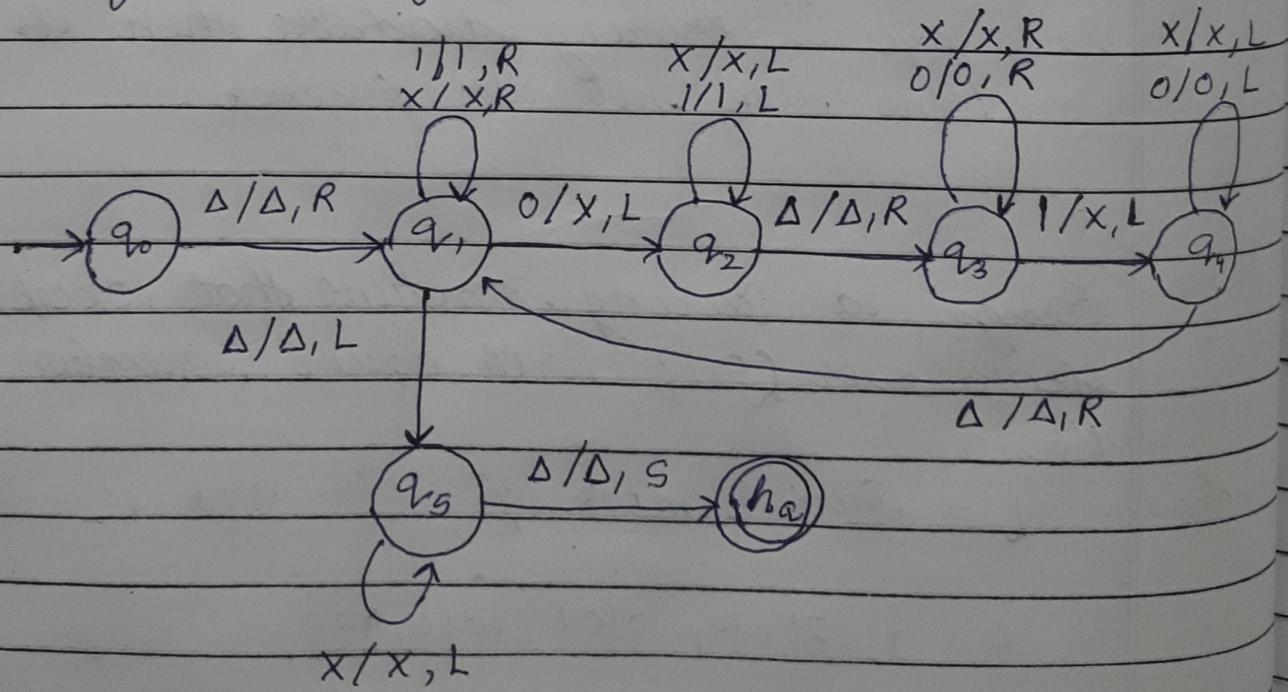
$L = \text{Same number of } 0's \text{ and } 1's$

$$L = \{ 01, 10, 0101, 1010, 1100, \dots \}$$

logic

string  
STRING → Δ T Δ

1. Keep moving Right, Read first leftmost 0, replace 0 with X ( $0 \rightarrow X$ )
2. Keep moving to left until you encounter Δ.
3. Move right, Read first leftmost 1, replace 1 with X ( $1 \rightarrow X$ )
4. Keep moving left until you encounter Δ.
5. Repeat the above steps until no more 0's and 1's left in string

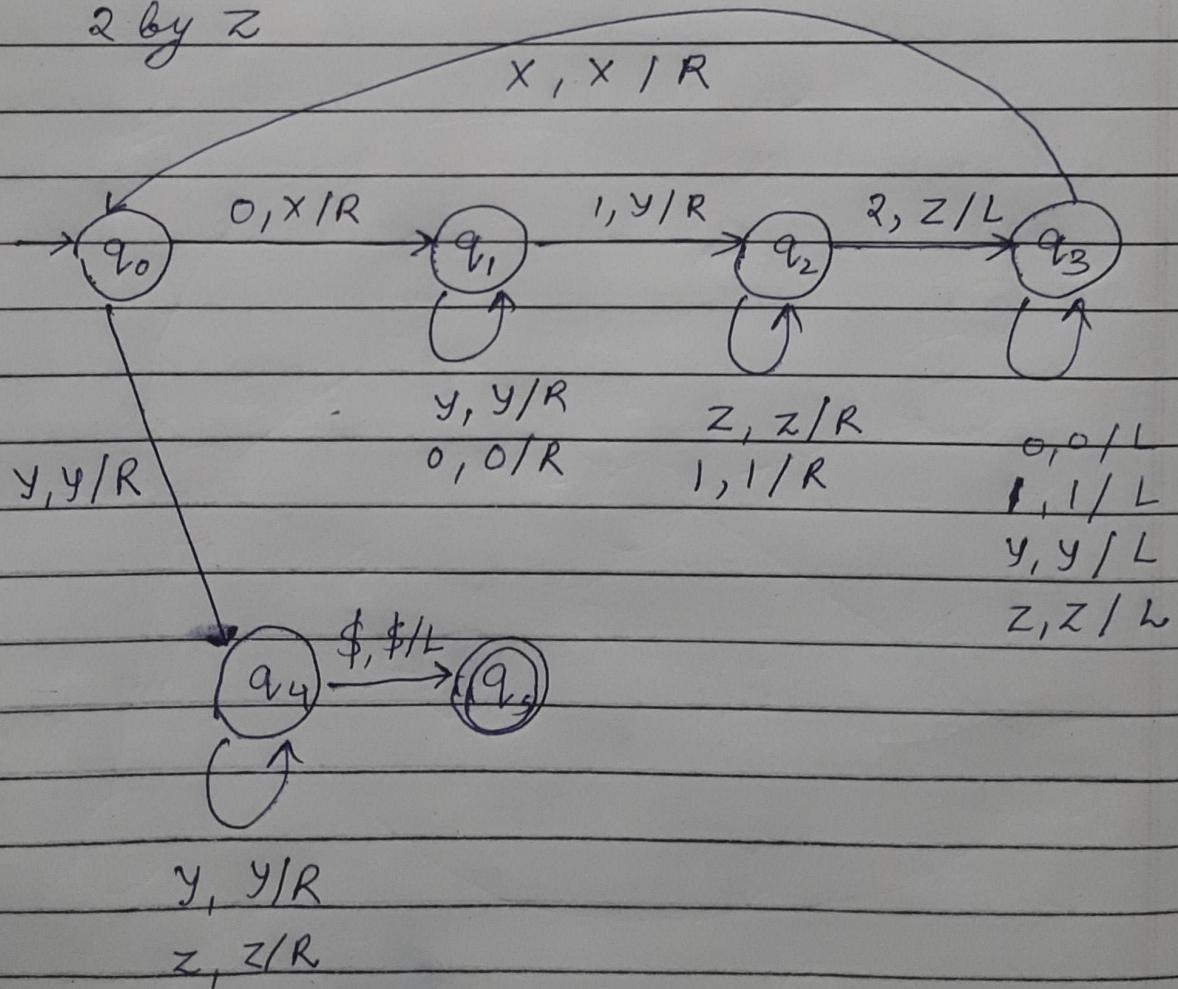


U19CS028

Q.3 Design a deterministic Turing Machine that decides language  $\{0^n 1^n 2^n \mid n \geq 1\}$

Ans:- Input e.g.: - 001122  $\rightarrow$  Accepted

Assumption  $\rightarrow$  we will replace 0 by X, 1 by Y and 2 by Z



U19CS028

Logic

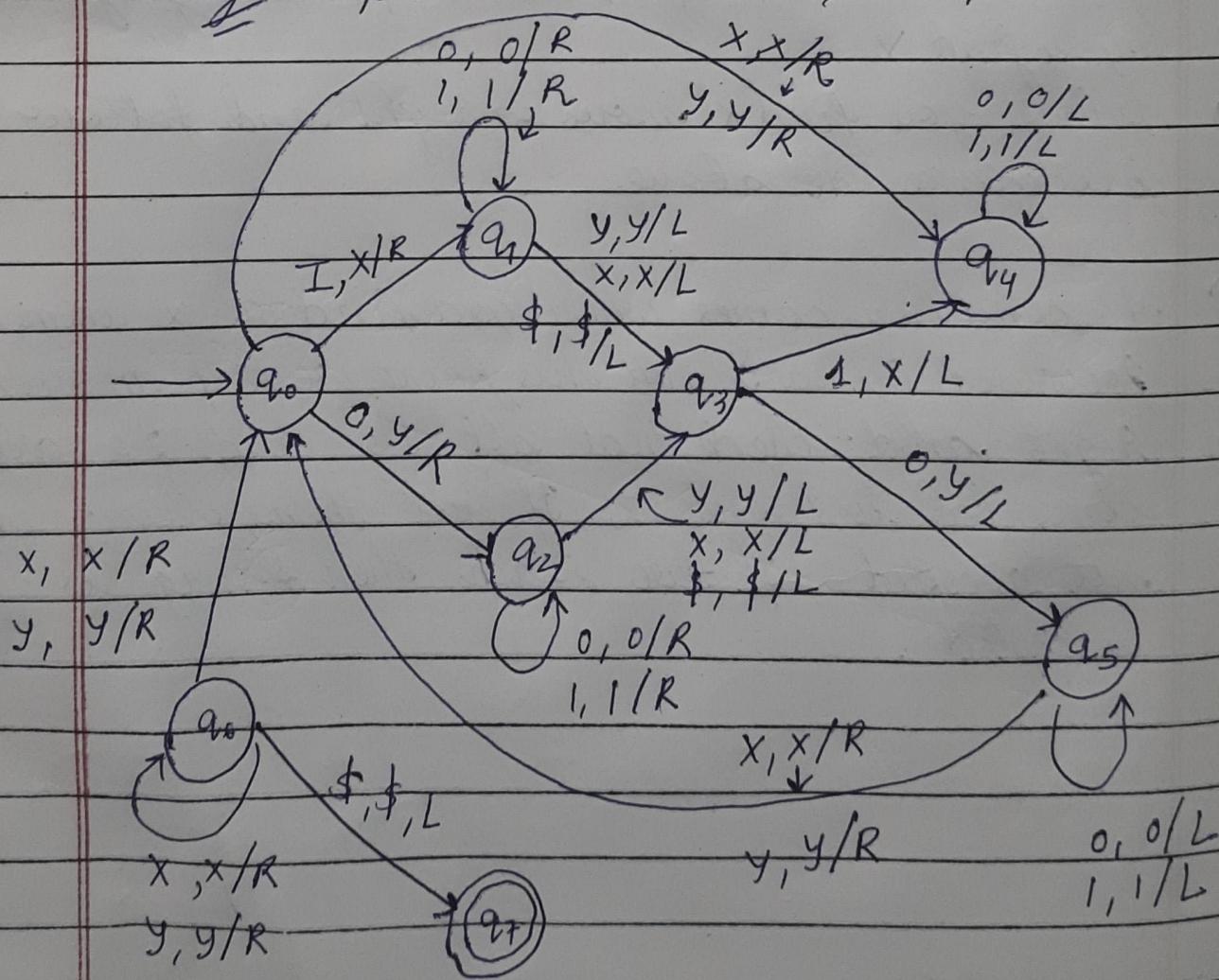
- first replace a zero from front by x.
- keep moving right till you find a 1 and replace this 1 by y.
- keep moving right till you find a 2 , replace it by z and move left , Now keep moving left till you find x
- when you find x move a right and follow same procedure as above .
- A condition comes when you find a x immediately followed by a y , At this point we keep moving right and check that all 1's and 2's are converted to y and z If not then string is not accepted , If we reach then string is accepted .

U19CS028

P-4 Design a Turing machine that accepts language of  $ww^R : w \in \{0,1\}$

Ans:- The language  $L = \{ww^R \mid w \in \{0,1\}\}$  represents a language where we use only 2 characters.

Eg. Input :- 00111100  $\Rightarrow$  Accepted



Assumption :- we will replace 0 by Y and 1 by X  
Logic :-

First check the first symbol, If it's 0 then replace it by Y and by X if it's 1, Then go to end of string, So last symbol is same as first we replace it also by X or Y depending on it.

Now again come back to position next to symbol replace from starting and repeat same process as told above.

One important thing is to note that since  $w^r$  is reverse of w of both of them will have equal number of symbols, Every time we replace  $n^{th}$  symbol from beginning of string replace a corresponding  $n^{th}$  symbol from end.

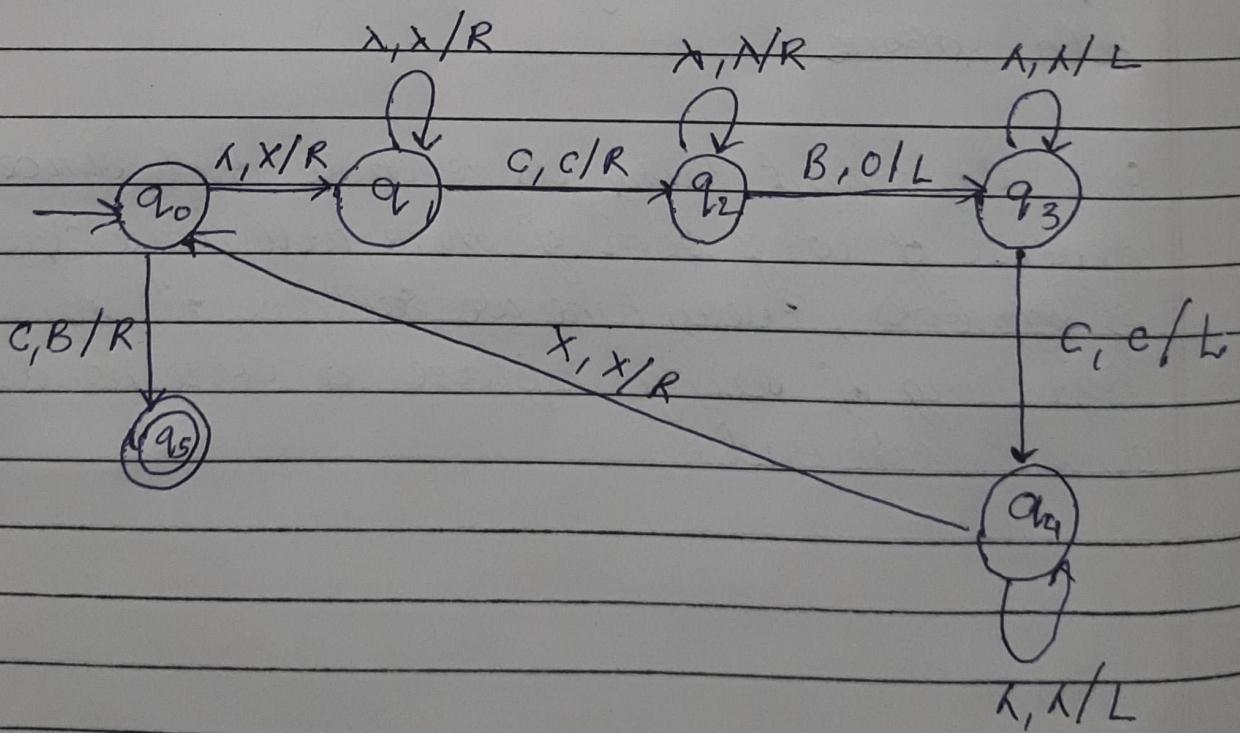
U19CS028

Q.5. Design a turing machine for addition of binary string separated by a 'c' considering TM follow unary format i.e. 0 or 1.

Input :- 1111c11      ∴ 4+2

Output :- 11111      ∴ 6.

Ans.- Let the Input character be  $\lambda$ . i.e. it can be 0 or 1.



U19CS028

Approach :-

Convert a  $\lambda$  (0 or 1) in first number to  $x$  and then traverse entire input and convert first blank encountered to  $\lambda$ , Then move left ignoring all  $1$ 's and " $c$ ", come the position just next to  $x$  and then repeat the same procedure till fine we get a " $c$ " instead of  $x$  on returning convert  $c$  into blank and addition is done.

Steps :-

1. Convert  $\lambda$  into  $x$  and goto step 2, If symbol is " $c$ " then convert it into blank (B), move right and goto step - 6.
2. Keep Ignoring  $1$ 's and move towards right Ignore " $c$ ", move right and go to step - 3.
3. Keep Ignoring  $0$ 's and move towards right convert a blank to  $\lambda$ , move left and go to step 4.

U19CS028

4. Keep Ignoring 'l's and move towards left  
Ignore "c" move left and go to step -3.
5. keep Ignoring 'l's and move towards left  
Ignore an 'x', move left and go to step -1
6. End.