**Fifth Semester B.E. Degree Examination, CBCS - Dec 2018 / Jan 2019**

**Automata Theory and Compatibility**

Time: 3 hrs. **Max. Marks: 80**

Note : *Answer any FIVE full questions, selecting ONE full question from each module.*

## Module - 1

**1. a. Define the following with example :**

**i) String ii) Language iii) Alphabet iv) DFSM. (08 Marks)**

**Ans.** i) The sequence of symbols obtained from the alphabets of a language is calle a string. Formally, a string is defined as a finite sequence of symbols from the alphabet $\Sigma$. Sequence of symbols from the alphabet $\Sigma$.

Ex : $\Sigma = \{0,1\}$

ii) A language can be defined as a set of strings obtained from $\Sigma^*$ where $\Sigma$ is set of alphabet of a particular language. In others words, a language is subset of $\Sigma^*$ which is denoted by $L \subseteq \Sigma^*$.

Ex : $\{\varepsilon, 0. 1, 01, 10, 1100, 0011 ....\}$

iii) A language consist of various symbols from which the words, statements etc., can be obtained. These symbols are called alphabets.

Ex : $\Sigma = \{a, b, ...., z, A, B, C, ...., Z, \#, \{, \}, (, ).....0, ....1\}$

iv) Deterministic Finite Automata (DFSM) is 5 - tuple or quintuple indicating five components $M = (Q, \Sigma, \delta, q_0, F)$

Where M is the name of machine

Q is non - empty finite set of states

$\Sigma$ is non - empty finite set of input alphabets

$\delta$ is transition function $Q \times \Sigma \rightarrow Q$

$q_0 \in Q$ is start state

$F \subseteq Q$ is accepting or final states

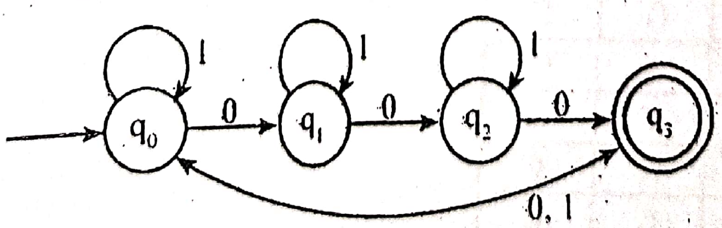
**b. Design a DFSM to accept each of the following languages :**

**i) $L = \{W \in \{0, 1\}^* : W$ has 001 as a substring$\}$**

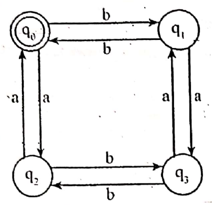
**ii) $L = \{W \in \{a, b\}^* : W$ has even number of a's and even number of b's$\}$.**

**(08 Marks)**

**Ans.** i)

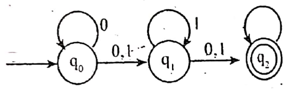



ii)



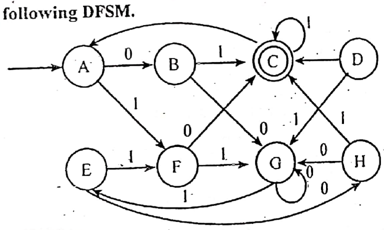

**OR**

**2. a. Define NDFSM. Convert the following NDFSM to its equivalent DFSM.** **(08 Marks)**




**Ans.** Refer Q.no.1(c) of June/July 2018.

**b. Minimize the following DFSM.** **(08 Marks)**




**Ans.**

| | 0 | 1 |
|---|---|---|
| →A | B | F |
| B | G | C |
| *C | A | C |
| D | C | G |
| E | H | F |
| F | C | G |
| G | G | E |
| H | G | C |







**Step 1 :**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| B | | | | | | | |
| C | X | X | | | | | |
| D | | X | | | | | |
| E | | X | | | | | |
| F | | X | | | | | |
| G | | X | | | | | |
| H | | X | | | | | |
| | A | B | C | D | E | F | G |

| δ | a | b |
|---|---|---|
| (A,B) | (B,G) | (F,C) |
| (A,D) | (B,C) | (F,G) |
| (A,E) | (B,H) | (F,F) |
| A,F) | (B,C) | (F,G) |
| (A,G) | (B,G) | (F,E) |
| (A,H) | (B,G) | (F,C) |
| (B,D) | (G,C) | (C,G) |
| (B,E) | (G,H) | (C,F) |
| (B,F) | (G,C) | (C,G) |
| (B,G) | (G,C) | (C,E) |
| (B,H) | (G,G) | (C,C) |
| (D,E) | (C,H) | (G,F) |
| (D,F) | (C,C) | (G,G) |
| (D,G) | (C,G) | (G,E) |
| (D,H) | (C,G) | (G,C) |
| (E,F) | (H,C) | (F,G) |
| (E,G) | (H,G) | (F,E) |
| (E,H) | (H,G) | (F,C) |
| (F,G) | (C,G) | (G,E) |
| (F,H) | (G,G) | (E,C) |
| (G,H) | (G,G) | (E,C) |

Step 2 :

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| B | X | | | | | | |
| C | X | X | | | | | |
| D | X | X | X | | | | |
| E | | X | X | X | | | |
| F | X | X | X | | X | | |
| G | | X | X | X | | X | |
| H | X | X | X | X | X | X | X |
| | A | B | C | D | E | F | G |

| | | |
|---|---|---|
| (A,E) | (B,H) | (F,F) |
| (A,G) | (B,G) | (F,E) |
| (B,H) | (G,G) | (C,C) |
| (D,F) | (C,C) | (G,G) |
| (E,G) | (H,G) | (F,E) |

Indistinguishable pairs : (A,E) , (,H) & (D,F)

Distinguishable pairs : C & G

**Minimize DFA :**

Step 1 :

(A,E), (B,H) & (D,F) Indistinguishable pair C,G distinguishable pair.

Step 2 :

States in minimized DFA

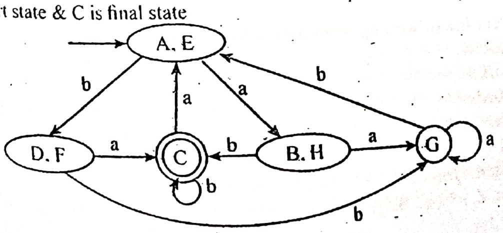
(A,E), (B,H), C; (D,F) , G

Step 3 :

| δ | 0 | 1 |
|---|---|---|
| (A,E) | (B,H) | (D,F) |
| (B,H) | G | C |
| C | (A,E) | C |
| (D,F) | C | G |
| G | G | (A,E) |

Step 4 :

(A,E) is start state & C is final state










## Module - 2

3. a. **Define Regular expression and write Regular expression for the following language.**

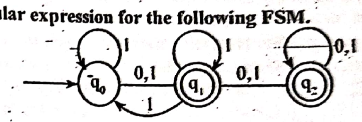
**i) L= $\{a^{2n} b^{2m} \mid n \geq 0, m \geq 0\}$**

**ii) L= $\{a^n b \mid m \geq 1, n \geq 1, nm \geq 3\}$.** **(08 Marks)**

**Ans.** Refer Q.no. 3(a) of June/ July 2018 for definition and (ii)

i. R . E = (aa)* (bb)*

b. **Obtain the Regular expression for the following FSM.** **(08 Marks)**




**Ans.** Since $q_2$ is

R . E = 1 01* 1(0 + 1)*

## OR

4. a. **Define a Regular grammar. Design regular grammars for the following languages.**

**i) Strings of a's and b's with at least one a.**

**ii) Strings of a's and b's having strings without ending with ab.**

**iii) Strings of 0's and 1 's with three consecutive 0's.** **(08 Marks)**

**Ans.** i) A grammer G is 4 - tople G = (V,T,P,S) where

V is set of variables or non - terminals

T is set of terminals

P is set of production

S is start symbol

i. V = {S,A}

T = {Q}

P = { S → aS

S → ε}

}

S is start symbol

ii. S → aA | bS

A → aA | bB

B → aA | bS | ε

iii. V = {S}

T = {0,1}

P = { S → A 000A

A → 0A | 1A | ε

}

S is the start symbol

**b. State and prove pumping theorem for regular languages.** (08 Marks)

**Ans.** Refer Q.no. 4(c) of June / July 2018.

## Module-3

**5. a. Define context free grammar. Design a context free grammar for the languages.**
**i) $L = \{0^m 1^m 2^n \mid m \geq 0, n \geq 0\}$ ii) $L = \{a^i b^j \mid i \neq j, i \geq 0, j \geq 0\}$**
**iii) $L = \{a^n b^{n-3} \mid n \geq 3\}$.** (08 Marks)

**Ans.** Refer Q.no. 5(a) of June/July 2018

i. $S \rightarrow AB$    $V = \{S,A,B\}$
$A \rightarrow 01 / )A1$    $T = \{0,1\}$
$B \rightarrow \varepsilon \mid 2B$    S is start symbol

ii. $V = \{S,A,B,C\}$
$T = \{a, b\}$
$P = \{$
$S \rightarrow aSb$
$S \rightarrow A$
$S \rightarrow B$
$A \rightarrow aA \mid a$
$B \rightarrow bB \mid b$
$\}$
S is start symbol

iii. $V = \{S,A\}$
$T = \{a,b\}$
$P = \{$
$S \rightarrow aaaA$
$A \rightarrow aAb \mid \varepsilon$
$\}$
S is start symbol

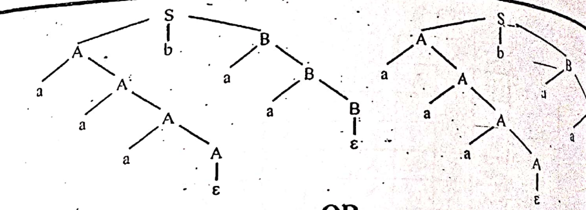
**b. Consider the grammar G with production.**
**$S \rightarrow AbB$**
**$A \rightarrow aA \mid \epsilon$**
**$B \rightarrow aB \mid bB \mid \epsilon$**
**Obtain leftmost derivation, rightmost derivation and parse tree for the string aaabab.** (08 Marks)

**Ans.**

| | |
|---|---|
| $S \Rightarrow A\beta B$ | $\Sigma \Rightarrow A\beta B$ |
| $\Rightarrow AbaB$ | $\Rightarrow aAbB$ |
| $\Rightarrow AbabB$ | $\Rightarrow aaAbB$ |
| $\Rightarrow Abab$ | $\Rightarrow aaaAbB$ |
| $\Rightarrow aAbab$ | $\Rightarrow aaabB$ |
| $\Rightarrow aaAbab$ | $\Rightarrow aaabaB$ |
| $\Rightarrow aaaAbab$ | $\Rightarrow aaababB$ |
| $\Rightarrow aaabab$ | $\Rightarrow aaabab$ |



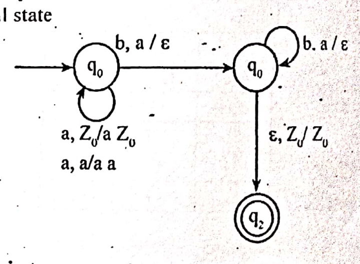





## OR

**6. a. Define a PDA. Obtain a PDA to accept- $L = \{a^n b^n \mid W \in \{a, b\}^*\}$. Draw the transition diagram.** (08

**Ans.** Refer Q.no. 6(a) of June / July 2018
$Q = \{q_0, q_1, q_2\}$
$\Sigma = \{a,b\}$
$\Gamma = \{a, Z_0\}$
$\delta : \{$
$\delta(q_0, a, z_0) = (q_0, aZ_0)$
$\delta(q_0, a, a) = (q_0, aa)$
$\delta(q_0, b, a) = (q_1, \varepsilon)$
$\delta(q_1, b, a) = (q_1, \varepsilon)$
$\delta(q_1, \varepsilon, Z_0) = (q_0, Z_0)$
$q_0 \in Q$ is the start state of machine
$Z_0 \in \Gamma$ is the initial symbol on the stack
$F = \{q_2\}$ is the final state




**b. Convert the following grammar into equivalent PDA.**
**$S \rightarrow aABC$**
**$A \rightarrow aB \mid a$**
**$B \rightarrow bA \mid b$**
**$C \rightarrow a.$** (08

**Ans. Step 1 :**
Push start symbol
$\delta(q_0, \varepsilon, Z_0) = (q, SZ_0)$
**Step 2 :**

| | |
|---|---|
| $S \rightarrow aABC$ | $\delta(q_1, a, S) = (q_1, ABC)$ |
| $A \rightarrow aB$ | $\delta(q_1, a, A) = (q_1, B)$ |
| $A \rightarrow a$ | $\delta(q_1, a, A) = (q_1, \varepsilon)$ |
| $B \rightarrow bA$ | $\delta(q_1, b, B) = (q_1, A)$ |
| $B \rightarrow b$ | $\delta(q_1, b, B) = (q_1, \varepsilon)$ |
| $C \rightarrow a$ | $\delta(q_1, a, c) = (q_1, \varepsilon)$ |
| | $\delta(q_1, \varepsilon, Z_0) = (q_f, Z_0)$ |

**Step 3 :**

$M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$

$Q = \{q_0, q, q_f\}$

$\Sigma = \{a,b\}$

$\Gamma = \{S, A, B, C, Z_0\}$

$\delta$ is transition in step 2

$q_0 \in Q$ is the start symbol

$Z_0 \in \Gamma$ is initial stack symbol

$F = \{q_f\}$ is final state

## Module-4

**7. a. State and prove pumping lemma for context free languages. Show that $L = \{a^n b^n c^n \mid n \geq 0\}$ is not context free.** **(10 Marks)**

**Ans.** **Statement :** Let L be the context free language and is infinite. Let Z be sufficiently long string and z L so that | z | n where n is some positive integer. If the string z can be decomposed into combination of strings z = uvwxy.

Such that |vwx| n. |vx| 1, then uv'w'y L for i=0,1,2,.......

**Proof of Pumping Lemma:**

By pumping lemma, it is assumed that string z L is finite and is context free language. We know that z is string of terminal which is derived by applying series of productions.

Case 1: To generate a sufficient long string z, one or more variables must be recursive. Let us assume that the language is finite, the grammar has a finite number of variables and each has finite length. The only way to derive sufficiently long string using such productions is that the grammar should have one or more recursive variables. Assume that no variable is recursive.

Since no non terminal is recursive, each variable must be defined. Since those variables are also non recursive, they to be defined in terms of terminal and other variables and so on.

From this we conclude that there is a limit length of the string that is generated from the start symbol S. this contradicts our assumption that the language is finite.

Therefore, the assumption that one or more variable are non recursive is incorrect. This means that this means that one or more variable are non recursive and hence the proof.

Case 2: The string z L implies that after applying some / all production some number of times, we get finally string of terminal and the derivation stops.





Let z L is sufficiently long string and so the derivation must have involved recursive use of some non terminal A and the derivation must have the form:

Note that any derivation should start from the start symbol S.

A DFA is a 5-tuple or quintuple $M=(Q, \Sigma, q_0, F)$

Q is non-empty, finite set of states.

I is non-empty, finite state set of input alphabet.

is transition function, which is mapping from Q. X $\Sigma$ to Q. for this transition function the parameters to be passed are state and input symbols. Based on the current state and input symbols, the machine may enter into another state. qo Q is the start state. F Q is a set of accepting or final state. Note: for each input symbol a, from a given state there is exactly one transition and we are sure to which state the machine enters. So the machine is called Deterministic Machine

Lanbncn = $\{a^n b^n c^n \mid n > 0\}$ is not a CFL.

Proof. Suppose $La^n b^n c^n$ is context-free. Let p be the pumping length.

- Consider $z = a^p b^p c^p \in La^n b^n c^n$.
- Since $|z| > p$, there are u, v, w, x, y such that z = uvwxy. $|vwx| \leq p$. $|vx| > 0$and $uv^iwx^iy \in L$ for all i >0.
- Since $|vwx| \leq p$, vwx cannot contain all three of the symbols a, b, c, because there are p bs. So vwx either does not have any as or does not have any bs or does not have any cs. Suppose, (w log) vwx does have any as. Then $uv^0wx^0y$ = uwy contains more as than either bs or cs. Hence uwy $\notin$ L.

**b. Explain Turing machine model.** **(06 Marks)**

**Ans.** Refer Q.no. 8(b) of Dec 2017/ Jan 2018.

**OR**

**8. a. Design a Turing machine to accept the language $L = \{0^n 1^n 2^n \mid n \geq 1\}$.** **(08 Marks)**

**Ans.** **Step-1:**

Replace 0 by X and move right, Go to state Q1.

**Step-2:**

Replace 0 by 0 and move right, Remain on same state

Replace Y by Y and move right, Remain on same state

Replace 1 by Y and move right, go to state Q2.

**Step-3:**

Replace 1 by 1 and move right, Remain on same state

Replace Z by Z and move right, Remain on same state

Replace 2 by Z and move right, go to state Q3.

**Step-4:**

Replace 1 by 1 and move left, Remain on same state

Replace 0 by 0 and move left, Remain on same state

Replace Z by Z and move left, Remain on same state

Replace Y by Y and move left, Remain on same state

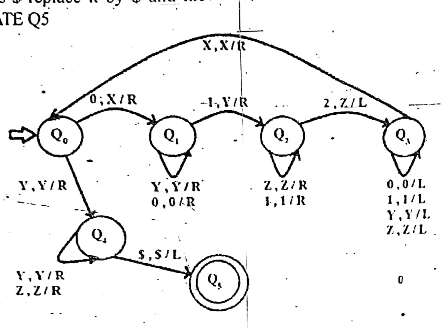
Replace X by X and move right, go to state QO.

**Step-5 :**

If symbol is Y replace it by Y and move right and Go to state Q4
Else go to step 1

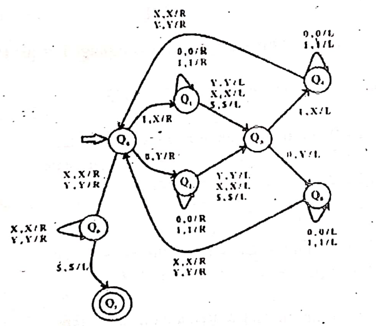
**Step-6:**

Replace Z by Z and move right, Remain on same state
Replace Y by Y and move right, Remain on same state
If symbol is $ replace it by $ and move left, STRING IS ACCEPTED, GO TO FINAL STATE Q5




**b. Design a Turing machine to accept strings of a's and b's ending with ab or ba.** (08 Marks)

**Ans.**







## Module-5

9\. a. **Explain the following :**
**i) Non deterministic Turing machine ii) Multi - tape Turing machine.** (06 M

Ans. REfer Q.no. 10 (a),(b) of June / July 2018.

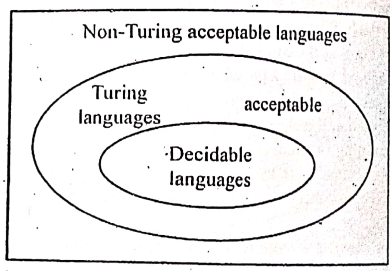
b. **Define the following :**
**i) Recursively enumerable language ii) Decidable language.** (06 M

Ans. **Recursive Enumerable (RE) or Type -0 Language**

RE languages or type-0 languages are generated by type-0 grammars. A language can be accepted or recognized by Turing machine which means it enter into final state for the strings f ~~language and~~ may or may not enter into rejecting state for the strings which are not part of the language. It means TM can loop for the strings which are not a part of the language. RE languages are also called Turing recognizable languages.

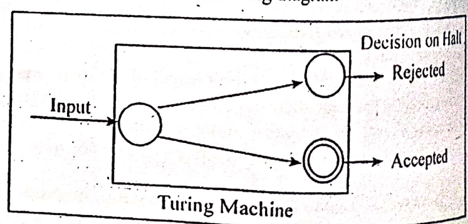
**Decidable language**

A language is called Decidable or Recursive if there is a Turing machine which accepts and halts on every input string w. Every decidable language is Turing Acceptable.




A decision problem P is decidable if the language L of all yes instances to P is decidable.

For a decidable language, for each input string, the TM halts either at the accept or the reject state as depicted in the following diagram -



c. **What is Post correspondence problem?** **(04 Marks)**

Ans. Refer Q.no. 10(d) of June/ July 2018.

**OR**

10. a. **What is Halting problem of Turing machine?** **(06 Marks)**

Ans. Refer Q.no. 10(b) of Dec 2017 / Jan 2018.

b. **Define the following : i) Quantum computer ii) Class NP.** **(06 Marks)**

Ans. **i) Quantum Computer :** Refer Q. no 9 c of June/July 2018

**ii) Class NP :** The class NP consists of those problems that are verifiable in polynomial time. NP is the class of decision problems for which it ~~is easy~~ to check the correctness of a claimed answer, with the aid of a little extra information. Hence, we aren't asking for a way to find a solution, but only to verify that an alleged solution really is correct. Every problem in this class can be solved in exponential time using exhaustive search.

c. **Explain Church Turing Thesis.** **(04 Marks)**

Ans.

- The Church-Turing thesis concerns an effective or mechanical method in logic and mathematics.
- A method, M, is called 'effective' or 'mechanical' just in case:
- M is set out in terms of a finite number of exact instructions (each instruction being expressed by means of a finite number of symbols)
- M will, if carried out without error, always produce the desired result in a finite number of steps
- M can (in practice or in principle) be carried out by a human being unaided by any machinery except for paper and pencil
- M demands no insight or ingenuity on the part of the human being carrying it out.
- They gave an hypothesis which means proposing certain facts.
- The Church's hypothesis or Church's turing thesis can be stated as:
- The assumption that the intuitive notion of computable functions can be identified with partial recursive functions.
- This statement was first formulated by Alonzo Church in the 1930s and is usually referred to as Church's thesis, or the Church-Turing thesis.