Semester: 5th
Branch: CSE/IT

Autumn Mid Semester Examination - 2018

School of Computer Engineering,
KIIT Deemed to be University,
Bhubaneswar-24

Formal Languages and Automata Theory

[Subject Code: CS-3003]

Time:1 hr 30 mins **Full Marks: 20**

(Answer any four questions including question number 1 which is compulsory)

Q1 Write your answers briefly. Provide proper justification to support your answer. **[1X5=5]**

a) Design an NFA without λ-transition and with a single final state that accepts the set $\{a\} \cup \{b^n : n \geq 1\}$

b) Let $L_1 = a^*$ and $L_2 = \{a^p : p \text{ is prime}\}$. Then, find the language represented by $L_1.L_2$.

c) Check whether the regular expressions $(a + ab + b)^*$ and $(b^*a^* + bba + \lambda)^*$ are equivalent or not. Justify.

d) Suppose M = {all binary strings ending with 1} and N = {all binary strings ending with 0}. Then, design finite automata with minimum number of states representing,

(i) $M \cup N$
(ii) $M \cap N$

e) Find δ^* (1, ba) in the following NFA,

δ	a	b	λ
1	{5}	-	{4}
2	{1}	-	-
3	-	{2}	-
4	-	{7}	{3}
5	-	-	{1}
6	-	{5}	{4}
7	{6}	-	-

Q2 Give regular expression for the following languages over {a, b}. **[5]**

a. Set of all strings containing neither **ab** nor **ba**.

b. $L = \{a^m b^n : m * n \text{ is even and } m + n \text{ is odd}\}$

c. Set of all strings not ending with **bba**.

d. Set of all strings having even number of **b's.**

e. $L = \{vwv : w, v \in (a + b)^* \text{ and } |v| = 2 \text{ and } |w| \geq 2\}$

3+2

Q3. Consider the regular expression $(\mathbf{a^*bab + ab^*ab})^*$. **[2+3=5]**

a. Design an NFA for the given regular expression.

b. Convert the NFA constructed in part (a) to the corresponding DFA.

Q4. Design DFA for the following languages, **[3+2=5]**

a. All strings over {1,2,3} whose sum of symbols (assuming symbols as integer) is a multiple of 6. (For example, 12333 is accepted as the sum of symbols is 12)

b. $L = \{1^x 2^y 3^z : x \geq 0,\ y \geq 0,\ z \geq 0\}$

Q5. Consider the following DFA, **[2.5X2=5]**

5

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

a. Minimize the above DFA.

b. Find a regular expression equivalent to the minimized DFA.