Printed Pages: 6 NMCA-214

(Following Paper ID and Roll No. to be filled in your Answer Books)

Paper ID : 214221 Roll No. | | | | | | | | | |

M.C.A.

Theory Examination (Semester-II) 2015-16

INTRODUCTION TO AUTOMATA THEORY & LANGUAGES

Time : 3 Hours *Max. Marks : 100*

Note: Attempt questions from all Sections as per directions.

Section-A

Attempt *all parts* of this section. Answer in brief. (2×10=20)

1. (a) Given the language L-{ab, aa, baa}, which of the following strings are in L*.

1) abaabaaabaa

2) aaaabaaaa

3) baaaaabaaaab

4) baaaaabaa

(A) 1,2 and 3 (B) 2,3 and 4

(C) 1,2 and 4 (D) 1,3 and 4

(b) Write the differences between DFA and NFA with example.

(c) Prove that regular sets are closed under union and complementation.

(d) Define universal Turing Machine, how it will be designed?

(e) Find the Language generated by G.

$S \rightarrow 0SA_1 2 \mid 012$

$2A_1 \rightarrow A_1 2$

$1A_1 \rightarrow 11$

Test whether (i) 00112 $\in$ L(G)

(ii) 001122 $\in$ L(G)

(f) Prove that the length of the shortest string NOT in the language (over $\Sigma = \{a,b\}$) of the following regular expression is a*b* (ba)* a*.

(g) Identify and remove the UNIT productions from the following Grammar.

S-> A/bb

A-> B/b

B-> S/a

(h) Prove $(a+b)^* = (a^*(ba^*))^*$

(i) What are the recursive and recursive enumerable language?

(j) What are the acceptance procedures for PDA? Give examples for each.

Section-B

Q2. Attempt *any five questions* from this section. (10×5=50)

a) Construct a Turing Machine for $L=\{ww^R / w=\{a, b\}^*\}$

b) Design DFA for the languages:-

a) $L=\{w \in (a,b)^* / n_a(w) \bmod 2 <> n_b(w) \bmod 3\}$.

b) $L=\{w \in (a,b)^* / (n_a(w) - n_b(w)) \bmod 3 = 0\}$

c) Design a deterministic finite automaton which can check the number of 'a' is divisible by 2 and the number of 'b' is divisible by 3. Minimize the number of states as much as possible.

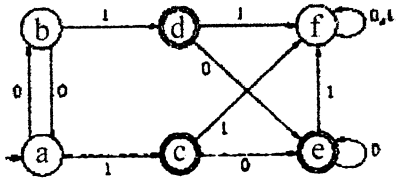
d) Find a grammar in GNF equivalent to the Grammar

E-> E+T / T

T-> T*F / F

F-> (E) / a

e) Construct a Minimization of DFA from an equivalent given transition diagram:



Present State	Next State			
	a = 0		a = 1	
	State	Output	State	Output
-> q0	q3	0	q1	1
q1	q0	1	q3	0
q2	q2	1	q2	0
q3	q1	0	q0	1

f) State and prove pumping Lemma for regular sets.

g) How PDA and CFG are equivalent? Explain the procedure to conversion of PDA to its equivalent CFG.

h) Construct a Turing Machine for checking the palindrome of the string of even length over {a, b}.

Section-C

Attempt *any two questions* from this section. **(15×2=30)**

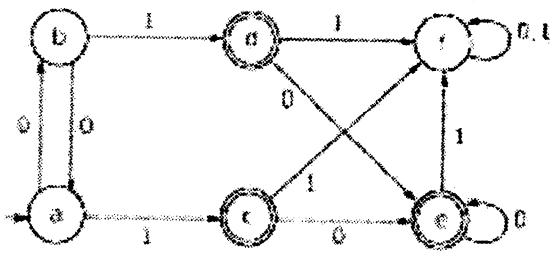
Q3. Design a bottom-up parser for the following grammar:-

E-> E+T / T

T-> T*F / F

F-> (E) / xl /x2.

Q4. (i) Construct a minimum state automaton equivalent to the finite automaton described by Fig.



(ii) Remove the € production from the given Grammar.

S -> ABAC

A -> aA/€

B -> bB/€

C -> c

Q5. (i) Simplify the following grammar by eliminating useless symbols and useless production :

S-> a / aA / B / C

A->aB / €

B->Aa

C->cCD

D-> dd.

Also find the Chomsky Normal Form of the simplified grammar.

(ii) State and prove Arden's theorem.