B5.2-R4 : AUTOMATA THEORY AND COMPILER DESIGN

NOTE :

1.	**Answer question 1 and any FOUR from questions 2 to 7.**
2.	**Parts of the same question should be answered together and in the same sequence.**

Time: 3 Hours **Total Marks: 100**

1. (a) Prove that for every two integers a and b, if a and b are odd, then ab is odd.

(b) Consider the CFG

$S \rightarrow S\,S\,+ \mid S\,S\,* \mid a$

Construct parse tree for string aa*a+

(c) Draw a DFA for (111+100)*0

(d) Generate quadruple table for given three address code.

t1 = b*c

t2 = a+t1

t3 = b*c

t4 = d/t3

t5 = t2-t4

(e) Prove the below given statement using mathematical induction.

$1 + 2 + 3 + \ldots + n = \frac{n\,(n+1)}{2}$

(f) What is DAG (Direct Acyclic graph) ? Find out the syntax tree and DAG for the following expression.

a+a*(b-c) + (b-c)*d

(g) What are the different phases of a Compiler ? **(7 × 4)**

2. (a) Construct a Syntax Directed Translation Scheme for a calculator that support + and * operations. Draw annotated parse tree for 3 + 5 * 4.

(b) Design a Turing machine to recognize all string given by (ab)*aba.

(c) List the roles of lexical analyzer. **(8 + 5 + 5)**

3. (a) Consider the given Mealy machine:

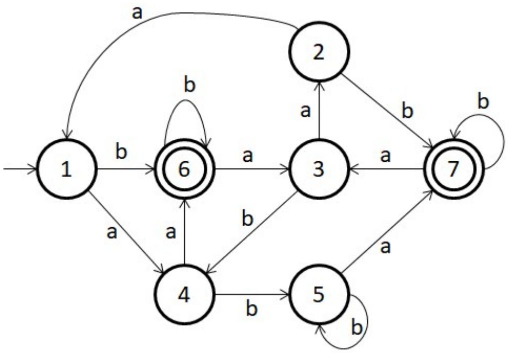
Present state	Next state			
	a=0		a=1	
	Next State	Output	Next state	Output
→ a	d	0	b	1
b	a	1	d	0
c	c	1	c	0
d	b	0	a	1

Construct Moore machine which is equivalent to it

(b) Construct Turing machine for $\{SS \mid S \in \{a,b\}^*\}$.

(c) Design PDA for $L = \{ WcW^R \mid W \in \{a,b\}^* , W^R$ is reverse of W }. **(6 + 6 + 6)**

4. (a) Check, whether following DFA is minimized or not. If not, then get the minimized one.



(b) Explain Left factoring and Left recursion.

(10 + 8)

5. (a) Obtain LR(1) item set and CLR(1) parsing table for given grammar
S → AA
A → aA | b

(b) Construct operator precedence table for given grammar
E → E + E | E – E | E * E | E / E | E ^ E | (E) | id

(10 + 8)

6. (a) Explain CNF. Convert given grammar in CNF.
S → AACD
A → aAb | ^
C → aC | a
D → aDa | bDb | ^

(b) Contract NFA for a^+ b (c | d) a* b using Kleene's theorem.

(c) Generate code for the following three-address sequence assuming that p and q are in memory locations:
y = *q
q = q + 4
*p = y
p = p + 4

(8 + 6 + 4)

7. (a) Define pumping lemma and prove that L={ 0^i1^j | j=i or j=2i } is not a regular language.

(b) Check given grammar is LL(1) or not.
S → i E t S S' |a
S' → e S | ∈
E → b

(c) Explain Input Buffering.

(6 + 6 + 6)