2015

M.Sc.

2nd Semester Examination

THEORY OF COMPUTATION & COMPILER

PAPER—COS-202

Full Marks : 50

Time : 2 Hours

The questions are of equal value.

The figures in the right-hand margin indicate full marks.

Candidates are required to give their answers in their own words as far as practicable.

Illustrate the answers wherever necessary.

Module—1

(*THEORY OF COMPUTATION*)

(*Marks : 20*)

Answer any *two* questions : 2×10

1. (a) Construct a Mealy machine which is equivalent to the Moore machine given by the following table : 5

Present State	Next State		Output
	a = 0	a = 1	
→ q_1	q_1	q_2	0
q_2	q_1	q_3	0
q_3	q_1	q_3	1

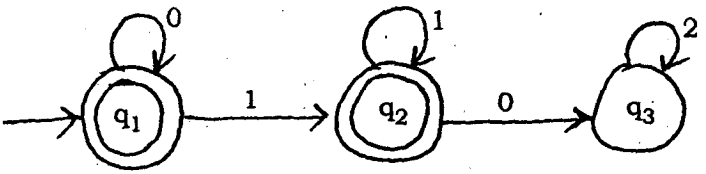
(b) Consider the following grammar G : 2

$$S \rightarrow aS \mid bS \mid a \mid b$$

Find the language L(G) generated by the grammar.

(c) Construct a grammar to generate all palindromes over {a, b}. 3

2. (a) Construct a regular expression corresponding to the state diagram described by the following transition diagram : 5



(b) Find a reduced grammar equivalent to the grammar G whose productions are : 5

$S \rightarrow AB \mid CA$

$B \rightarrow BC \mid AB$

$A \rightarrow a$

$C \rightarrow aB \mid b$

$D \rightarrow d$

3. (a) Find a grammar in Chomsky Normal Form equivalent to : 5

$S \rightarrow aAbB$
$A \rightarrow aA \mid a$
$B \rightarrow bB \mid b$

(b) Construct a grammar in Greibach Normal Form equivalent to the grammar : 5

$S \rightarrow AA \mid a$
$A \rightarrow SS \mid C$

4. (a) Construct a PDA accepting the Language

$L = \{a^n b^m c^m d^n \mid m, n \geq 1\}$

(b) Design a Turing Machine to recognize all strings over {0, 1} consisting of an odd number of 0's. 5

Internal Assessment : 5

Module—2

(*Compiler Design*)

(*Marks : 20*)

Answer any *two* questions.

1. (a) What do you mean by token and value of token ? Explain with suitable example. 5

(b) Consider the following grammar :

$S \rightarrow ABC$

$A \rightarrow Aald$

$B \rightarrow Bble$

$C \rightarrow Cd \mid f$

Eleminate the left recursion from the above grammar. 5

2. (a) Define LL(1) grammar.

   (b) Construct predictive parser for the following grammar :

   $E \rightarrow TE'$

   $E' \rightarrow TE' / \in$

   $T \rightarrow FT'$

   $T' \rightarrow *FT' / \in$

   $F \rightarrow (\in) / id$ 2+8

3. Construct LALR (1) parsers for the following grammar : 10

   $S \rightarrow L = R$

   $S \rightarrow R$

   $L \rightarrow *R$

   $L \rightarrow id$

   $R \rightarrow L$

4. Short notes (any *two*) :

   (a) LEX and YAAC ;

   (b) Loop optimization ;

   (c) Symbol table ;

   (d) Dependency graph.

Internal Assessment : 5
