Printed Pages: 4 **CA410/NMCAE-011/MCAE11**

(Following Paper ID and Roll No. to be filled in your Answer Books)	
Paper ID : 214410	**Roll No.**

M.C.A.

Theory Examination (Semester-IV) 2015-16

COMPILER DESIGN

Time : 3 Hours ***Max. Marks : 100***

Note : Attempt questions from all Sections as per directions.

Section-A

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

(a) What are the two parts of a compilation? Explain briefly.

(b) List the phases that constitute the front end of a compiler.

(c) Differentiate tokens, patterns, and lexeme.

(d) Mention the various notational shorthand's for representing regular expressions.

(e) Why lexical and syntax analyzers are separated out?

(f) Write the algorithm for FIRST and FOLLOW.

(g) Palindromes can't be recognized by any FSA why?

(h) What are kernel & non-kernel items?

(i) What is phrase level error recovery?

(j) List the advantages and disadvantages of operator precedence parsing.

Section-B

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

(a) What is the difference between S-attributed and L-attributed definitions?

(b) What is code motion? Also state the properties of optimizing compiler.

(c) Differentiate between top down and bottom up parser. Under which conditions predictive parsing can be constructed for a grammar?

(d) What are the various methods of implementing three address statements? What are the problems with top down parsing?

(e) Write the procedure to generate a*(b+c)/(d+e) TAC.

(f) Give the syntax-directed definition for if-else statement.

(g) Eliminate left recursion and left factor the following grammar. E→aba|abba|Eb|EbE

(h) Eliminate left recursion in more than one level.

S→Aa|b

A→Ac|Sd|ε

Section-C

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

3. Explain how the scope rules and the block structure of a programming language decide the structure of the symbol table? Construct the SLR parsing table for the following grammar:

E->E+ T

E->T

T->T * F

T->F

F-> id

L->L,E / E

4. What is the objective of intermediate code generation? What is the different form of intermediate code generated by intermediate code generation phase?

   Generate the three-address code for the following code segment:

   Main ( )

   { int a = 1; int b[10];

   while (a<= 10)

   b[a] = 2 ** a; }

5. Find the canonical collection of sets of LR (1) items:

   S->AaAb

   A->BbBa

   A-> ε B-> ε