INSTITUTE OF AERONAUTICAL ENGINEERING
(Autonomous)

Four Year B.Tech V Semester End Examinations(Regular) - November, 2019

Regulation: IARE – R16

COMPILER DESIGN

Time: 3 Hours   (Common to CSE | IT)   **Max Marks: 70**

Answer ONE Question from each Unit
All Questions Carry Equal Marks
All parts of the question must be answered in one place only

UNIT – I

1. (a) Explain the need for dividing the compilation process into various phases and explain its functions. **[7M]**

   (b) Write down the algorithm to eliminate left recursion and left factoring with example. **[7M]**

2. (a) Describe the error recovery schemes in the lexical phases of a compiler **[7M]**

   (b) Construct the predictive parser for the following grammar:
   S→ a / ↑ / (T)
   T→ T,S / S
   Show the behavior of the parser in the sentences,
   i) (a,(a,a))
   ii) (((a, a), ↑, (a), a)) **[7M]**

UNIT – II

3. (a) Define LR parsing and explain the parts of LR parser. **[7M]**

   (b) Construct CLR parsing table for the following grammar
   S→ CC
   C→ aC|d **[7M]**

4. (a) Explain conflicts during shift reduce parsing. **[7M]**

   (b) Explain error recovery for the following grammar
   E→E+E | E*E | (E) | id **[7M]**

UNIT – III

5. (a) What are three address codes? What is its type? How it is implemented. Explain with an example **[7M]**

   (b) Write down the translation scheme to generate three address code for the Boolean expression. **[7M]**

6. (a) Explain the evaluation of S-attributed and L-attributed definitions **[7M]**

   (b) Generate the three address code and draw the abstract tree for the following expressions.
   i) (x-y)*z+m-n
   ii) a+(b-c)+(b+c)*(a*e) **[7M]**

UNIT – IV

7. (a) Discuss in details about storage allocation strategies. **[7M]**

   (b) Explain in detail about the various operation and implementation forms of symbol table. **[7M]**

8. (a) Differentiate explicit and implicit allocation of memory to variables? **[7M]**

   (b) Illustrate the dynamic storage allocation techniques. **[7M]**

UNIT – V

9. (a) Explain in detail about the various representation loops in Flow Graph. **[7M]**

   (b) Draw the DAG for the following statements

   prod = 0 ;

   i = 1 ;

   do

   {

   prod = prod + a[i] x b[i] ;

   i = i + 1 ;

   } while (i <= 10) ; **[7M]**

10. (a) Explain code generation phase with simple code generation algorithm with example. **[7M]**

    (b) Discuss the principle sources of optimization with examples. **[7M]**