



P.E.S. College of Engineering, Mandya - 571 401

(An Autonomous Institution affiliated to VTU, Belagavi)

Sixth Semester, B.E. - Computer Science and Engineering

Semester End Examination; May / June - 2019

Compiler Design

Time: 3 hrs

Max. Marks: 100

Note: Answer FIVE full questions, selecting ONE full question from each unit.

UNIT - I

- 1 a. Explain the phases of compiler with example. 12
- b. Draw transition diagrams for the following tokens :
- i) Relop (Relational Operator)
- ii) Unsigned numbers in Pascal 8
- iii) Identifiers and Keywords
- How do you distinguish between Identifiers and Keyword?
- 2 a. Explain compiler constructor tools briefly. 5
- b. Briefly explain the following with example :
- i) Token ii) Pattern iii) Lexeme 6
- c. What is Input Buffering? Explain Input Buffering strategies used in lexical analysis phase. 9

UNIT - II

- 3 a. Construct a predictive parsing table for the following grammar : 12
- $$E \rightarrow TE' \quad E' \rightarrow +TE' / \epsilon \quad T \rightarrow FT' \quad T' \rightarrow *FT' / \epsilon \quad F \rightarrow id / (E)$$
- b. Write an algorithm for;
- i) Left factoring 8
- ii) Remove left recursion
- 4 a. Write algorithm to compute First and Follow sets. Also find First(X) where 'X' is grammar symbol and Follow(A) where 'A' is a non terminal for the following grammar : 12
- $$S \rightarrow iE + SS' / a \quad S' \rightarrow eS / \epsilon \quad E \rightarrow b$$
- b. Explain non-recursive predictive parsing algorithm with example. 8

UNIT - III

- 5 a. Construct canonical LR parsing table for the grammar, 16
- $$S \rightarrow cC \quad C \rightarrow cC / d$$
- b. Briefly explain the possible actions of shift reduce parser. 4
- 6 a. Construct LR(0) items for the grammar : 11
- $$E' \rightarrow E \quad E \rightarrow E + T / T \quad T \rightarrow T * F / F \quad F \rightarrow (E) / id$$
- b. Explain shift-reduce parser with example. 9

UNIT - IV

- 7 a. Construct DAG for the following expression : 3
 $a + a * (b - c) + (b - c) * d$
- b. Write grammar syntax directed definition for a simple desk calculator and show the annotated parse tree for the expression $(8+6) * (3+2)$. 12
- c. Define type-checking rules for coercion from integer to real. 5
- 8 a. Explain the following parameters parsing methods : 12
- i) Call by value ii) Call by reference
- iii) Call by name iv) Copy restore
- b. Explain the following with example : 8
- i) Inherited attribute
- ii) Synthesized attribute

UNIT - V

- 9 a. For the assignment statement $a = b * - c + b * - c$. Write sequence of;
 i) Three address code for the syntax tree 10
 ii) The address code for DAG
 iii) Give its Quadruple, Triple and Indirect representation
- b. Write grammar for control-flow statements and give syntax directed definition for flow of control statements. 10
- 10 a. Explain any five issues in the design of code-generation. 10
- b. Define basic block with example. Write an algorithm to partition a sequence of three address statements into basic blocks. Find the basic blocks in the following three address code : 10
- i) $prod := 0$
- ii) $i := 1$
- iii) $t_1 := 4 * i$
- iv) $t_2 := t_1 + 6$
- v) $prod := t_1 * t_2$
- vi) if $prod > 10$ goto (iii)

* * * *