Reg. No. : □□□□□□□□□□□□

Question Paper Code : 51143

B.E./B.Tech. DEGREE EXAMINATION, NOVEMBER/DECEMBER 2014.

Seventh Semester

Computer Science and Engineering

080230045 — PRINCIPLES OF COMPILER DESIGN

(Regulation 2008)

Time : Three hours Maximum : 100 marks

Answer ALL questions.

PART A — (10 × 2 = 20 marks)

1. List the functions of preprocessors.

2. Define Lexeme and token.

3. What is the use of symbol table in lexical analyzer?

4. What are the three parts in Lex program?

5. Eliminate the left recursion for the following grammar :

   $S \rightarrow AS \mid a, \quad A \rightarrow SA \mid b$

6. What is meant by handle and handle pruning?

7. Compare syntax directed definition and translation schemes.

8. Write the three address code for ternary representation $x := A[y]$.

9. What is called activation tree?

10. Define loop unrolling.

PART B — (5 × 16 = 80 marks)

11. (a) (i) Discuss in detail how symbol table is used in various phases of compiler. (8)

(ii) Write short notes on cousins of compiler. (8)

Or

(b) (i) Explain in detail about various phases of compiler with a neat block diagram. (12)

(ii) Briefly describe about the compiler construction tools. (4)

12. (a) Draw the minimized DFA for the regular expression $(a \mid b)^* abb(a \mid b)$. (16)

Or

(b) (i) Draw the minimized DFA directly from the augmented regular expression $(a \mid b)^* abb\#$ by drawing the syntax tree. (12)

(ii) Elaborate on various input buffering techniques in lexical analyzer. (4)

13. (a) (i) What is meant by ambiguous grammar? Explain it with an example. (4)

(ii) Design a predictive parser table for the following grammar : (12)

$S \to (L) \mid a,\ L \to L, S \mid a$.

Or

(b) Design an LALR parser for the following grammar and parse the string $id = *id\$$. (16)

$S \to L = R \mid R$

$R \to L$

$L \to {}^*R \mid id$

14. (a) (i) Draw the syntax tree, DAG, postfix notation and three address code for the expression $x = -(a+b) * (c+d) + (a+b)$. (8)

(ii) Explain in detail about the translation scheme for Boolean expressions and flow of control statements. (8)

Or

(b) (i) Draw the annotated parse tree and three address code for the statement $x = A[y, z]$ with the value 10 × 20, low1 = low2 = 1 and width = 4. (8)

(ii) Define Back patching. Write the translation scheme for procedure calls. (8)

15. (a) (i) Explain the two standard storage allocation strategies in run time storage management. (8)

(ii) Write the algorithm to partition a given three address code in to basic blocks. (8)

Or

(b) (i) Discuss in detail about loop optimisation with suitable example. (8)

(ii) Explain in detail the code generation algorithm. (8)