Reg. No. : ☐☐☐☐☐☐☐☐☐☐☐☐☐☐

Question Paper Code : 41162

B.E./B.Tech. DEGREE EXAMINATION, NOVEMBER/DECEMBER 2013.

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.   What is difference between compiler and assembler?

2.   What are cousins of the compilers?

3.   What is the role of Lexical analyser?

4.   Draw the NFA for regular expression ab $(a/b)^*$ ab *.

5.   Define context free grammar. Give an example.

6.   What are role of a parser?

7.   What are intermediate languages?

8.   Give some forms of intermediate languages.

9.   What is DAG?

10. What is meant by code optimization?

PART B — (5 × 16 = 80 marks)

11. (a)   What are the six phases of the compilers? Explain with an example.

Or

(b)   What are compiler construction tools? Describe about their features.

12. (a) Draw the DFA for the regular expression (a/b)*abb and find minimised DFA.

Or

(b) (i) Write in detail about tool for generating a Lexical Analyser. (8)

(ii) Give a lexical analyser specification for a model compiler. (8)

13. (a) Design a predictive parsing table for the grammar

E -> E+T/T

T -> T*F/F

F -> (E)/id

and show the moves for input string id * id + id.

Or

(b) Design an operator precedence parser for the grammar

E -> E+T/T

T -> T*F/F

F -> (E)/id

and draw a precedence function graph.

14. (a) Detail the translation scheme for flow control statements and precedure call.

Or

(b) Discuss about back patching of Boolean expression.

15. (a) What are the principle sources of optimization? Write notes on peephole optimization.

Or

(b) Describe a Simple Code Generator. Explain how it generates code by an example.

———————