Registration No. : | | | | | | | | | | |

Total number of printed pages – 2

B. Tech

PCCS4305(New)

Sixth Semester (Back) Examination – 2013

COMPILER DESIGN

BRANCH : CSE

QUESTION CODE : B 234

Full Marks – 70

Time : 3 Hours

*Answer Question No. **1** which is compulsory and any **five** from the rest.*
The figures in the right-hand margin indicate marks.

1. Answer the following questions : 2×10

   (a) Differentiate tokens, patterns, lexeme.

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

   (c) What is meant by handle pruning ?

   (d) What are the limitations of static allocation ?

   (e) What are the properties of optimizing compiler ?

   (f) What do you mean by porting of a compiler?

   (g) What is an ambiguous grammar ?

   (h) Write short notes on LEX ?

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

   (j) What are the properties of optimizing compiler?

2. (a) Explain in detail different phases of a compiler. Also write down the output for the following expression after each phase a:=b*c-d. 5

   (b) For regular expression (a |b) * abb draw DFA by minimizing number of states. 5

3. (a) Give the operator precedence parsing algorithm. 5

(b) Verify whether or not the following grammar is SLR(1) by constructing the parsing table : 5

$S \rightarrow Ab \mid B$

$A \rightarrow aB$

$B \rightarrow aA \mid a$

4. Explain how a symbol table can be integrated in to a compiler to perform semantic analysis. Specify how and when information should be stored and retrieved form to the symbol table. 10

5. (a) Explain the translation scheme of checking the types of statements. 5

(b) What are different storage allocation strategies ? Explain. 5

6. (a) What are compiler construction tools ? Explain. 5

(b) Explain why it's possible to design an independent lexical analyzer with an example. 5

7. (a) Explain how basic blocks are identified. 5

(b) Explain the use of DAG in code optimization. 5

8. Write short notes on any **two** of the following : 5×2

(a) Flow Graphs

(b) Quadruples

(c) LALR Parser

(d) Activation Record.