

Total No. of Questions : 6]

SEAT No. :

P4890

B.E./Insem.- 74

[Total No. of Pages : 2

B.E. (Computer Engg.)

**PRINCIPLES OF MODERN COMPILER DESIGN
(2012 Pattern) (Semester - I)**

Time : 1 Hour]

[Max. Marks : 30

Instructions to the candidates:

- 1) *Neat diagrams must be drawn wherever necessary.*
- 2) *Figures to the right indicate full marks.*
- 3) *Assume suitable data, if necessary.*

- Q1)** a) Explain need of symbol table with compiler. List different data structures for symbol table. [4]
b) What is garbage collection? [2]
c) What is LEX? Give format of LEX specification file. [4]

OR

- Q2)** a) Compare single pass and multi-pass design for compiler. [4]
b) What are lexeme, pattern and token in lexical analysis? [3]
c) Explain static Vs dynamic storage allocation. [3]

- Q3)** a) What are problems/ issues associated with top-down parser. [2]
b) What is type checking? [2]
c) Generate LR(1) parsing table for following grammar: [6]

$S \rightarrow BB$

$B \rightarrow cB$

$B \rightarrow d$

OR

- Q4)** a) Explain in brief: Recursive Descent parser [2]
b) Differentiate between syntax and semantic analysis by giving example. [2]
c) Check if following grammar is LL (1) [6]

$S \rightarrow iCtSS' \mid a$

$S' \rightarrow eS \mid \epsilon$

$C \rightarrow b$

P.T.O.

- Q5)** a) Explain advantages of intermediate code. [2]
b) Compare quadruple, triple and indirect triple. [4]
c) Generate intermediate code for following statement: [4]
a= b+c
(Specify syntax directed translation scheme)

OR

- Q6)** a) Explain need for intermediate code. [2]
b) Define: L-attributed grammar [2]
c) Generate intermediate code for following Boolean expression: [6]
p < q or a > b
(Specify syntax directed translation scheme)

