

Total No. of Questions : 10]

SEAT No. :

P3106

[5154]-672

[Total No. of Pages : 3

B.E.(Computer Engineering)
PRINCIPLES OF MODERN COMPILER DESIGN
(2012 Pattern) (Semester-I) (410442)

Time : 2½ Hours]

[Max. Marks : 70

Instructions to the candidates:

- 1) *Neat diagrams must be drawn wherever necessary.*
- 2) *figures to the right indicate full marks.*

Q1) a) Write down the regular expression for the following **[4]**

- i) Comment in C.
- ii) Floating point number.

b) Write a Syntax directed translation scheme for Boolean Expression. **[6]**

OR

Q2) a) Consider the statement: **[4]**

$X[i, j] := Y[i+j, k] + z.$

The maximum dimensions of X are [d1, d2] and of Y are [d3, d4].

Generate three address code.

b) What are synthesized and inherited attributes? What are Marker Non terminal symbols? Give example. **[6]**

Q3) a) Write a short note on I/P buffering used in Lexical Analyzer. **[4]**

b) Check whether the following grammar LL(1) or not. **[6]**

$E \rightarrow TE'$

$E' \rightarrow *TE' / \epsilon$

$T \rightarrow FT'$

$T' \rightarrow ^T / \epsilon$

$F \rightarrow (E) / id$

P.T.O.

OR

- Q4)** a) What is need of Semantic Analysis? Explain the position of Type Checker with diagram. [4]
- b) Show that the following grammar is not SLR (1) [6]
- $$S \rightarrow Aa Ab|B b Ba$$
- $$A \rightarrow \epsilon$$
- $$B \rightarrow \epsilon$$

- Q5)** a) Write a note on application of Directed Acyclic Graph (DAG) in code generation. [6]
- b) Write an algorithm for copy propogation. [6]
- c) Write a short note on Data flow equations and iterative data flow analysis. [6]

OR

- Q6)** a) Describe in detail about a simple code generator with the appropriate algorithm. [6]
- b) Discuss about the following: [6]
- i) Dead-code Elimination and
 - ii) Code motion.
- c) Show the steps involved on generating the code for the expression: [6]
- $$(x+y)/(p+q)$$

- Q7)** a) Discuss source language issues related to Object Oriented languages. [6]
- b) Explain code generation for control flow statements. [6]
- c) Explain Polymorphic typing with respect to Functional languages. [4]

OR

- Q8)** a) Explain following related to Haskell program. [6]
- i) Offside rule.
 - ii) Lists.

- b) Explain following with respect to Functional languages. [6]
i) Referential transparency.
ii) Lazy evaluation.
- c) What is activation record? Explain possible structure of an activation record? [4]
- Q9)** a) Discuss the issues in Tuple Space implementation. [6]
b) Write short notes on [6]
i) JIT
ii) nmake
- c) Explain following shared variable models [4]
i) Locks
ii) Monitors

OR

- Q10)**a) Explain cross compilation using XMLVM. [6]
b) Discuss following with respect to Parallel object oriented languages. [6]
i) Object location
ii) Object migration
- c) What is interpreter? Explain JVM interpreter. [4]

