

**UNIVERSITY OF PUNE**  
**[4364]-796**  
**B. E. (INFORMATION TECHNOLOGY) Examination 2013**  
**COMPILER DESIGN**  
**(2008 Pattern)**

**[Total No. of Questions:12]**  
**[Time : 3 Hours]**

**[Total No. of Printed pages :3]**  
**[Max. Marks : 100]**

***Instructions :***

- (1) *Answer three questions from each section.*
- (2) *Answers to the two Sections should be written in separate answer-books*
- (3) *Neat diagrams must be drawn wherever necessary.*
- (4) *Figures to right indicate full marks.*
- (5) *Assume suitable data, if necessary.*

---

---

**SECTION-1**

Q1) a) Explain the Role of Lexical Analyzer, Explain interaction between Lexical Analyzer and Parser, Define Lexeme, Token and Pattern with suitable example. [8]

b) Define passes and phases of Compiler. Explain different phases of Compiler in detail. [10]

**OR**

Q2) a) Explain the following terms [6]

- i) Cross Compiler
- ii) Bootstrapping
- iii) Incremental Compiler

b) Explain the role of Regular expression and DFA in Lexical Analyzer. [6]

c) Differentiate between Compiler and Interpreter. [6]

Q3) a) Discuss the term 'Ambiguity of Grammar'. Consider following grammar.  $S \rightarrow S + S \mid S * S \mid a \mid b$  [10]

Determine whether the grammar is ambiguous? If yes, show resultant parse trees for one example string.

b) Which are the conflicts may encounter during LR parsing? Explain with example. [6]

OR

Q4) a) Show that following grammar is LR (1) but not LALR. [8]

$S \rightarrow Aa \mid bAc \mid Bc \mid bBa$

$A \rightarrow d$

$B \rightarrow d$

b) Explain the following terms with suitable examples. [8]

i) Left Recursion                      ii) Left Factoring

Q5) a) Explain the following terms with suitable examples. [8]

i) S-attributed Grammar    ii) L-attributed Grammar

iii) Type Expression              iv) Abstract Syntax Tree

b) Suppose declarations are generated for Pascal Language by following grammar: [8]

$D \rightarrow id \ L$

$L \rightarrow , id \ L \mid : T$

$T \rightarrow int \mid real$

Construct a translation scheme to enter the type of each identifier into the symbol table

OR

Q6) Consider the following attributed grammar: [16]

Grammar Rule	Semantic Rule
$S \rightarrow XYZ$	$Y.s = S.s$
	$X.s = Y.i + Z.i$
	$S.i = X.i$
$X \rightarrow x$	$X.i = 3 * X.s$
$Y \rightarrow y$	$Y.i = Y.s$
$Z \rightarrow z$	$Z.i = 1$

i) Draw Parse tree for string "xyz".

ii) Draw annotated parse tree.

iii) Draw dependency graph for associating attributes and describe correct order of evaluation.

iv) If  $S.s = 4$  before evaluation, what is  $S.i$  after evaluation?

## SECTION –II

Q7)a)What is activation record? Explain its components with an example. [10]

b)Explain procedure call with an example. [6]

OR

Q8)a) Discuss Display Mechanism used by the Pascal compiler to handle access to non-local names with adequate illustration. [10]

b)Compare Static Scope and Dynamic Scope. Illustrate with suitable examples. [6]

Q9)a) Generate quadruple, triple and indirect triple representation for following  $a = -u * v / w^{x+y * z}$  [6]

b) Explain following code optimization techniques with example: [8]

i) Common subexpression Elimination

ii)Code Movement

iii) Strength Reduction

iv) Dead Code Elimination

c) Define the following terms with example [4]

i) Live Variable

ii) Available expression

OR

Q10) a) For the following fragment of code, generate three address code, AST and DAG.  $\text{do } \{ p = p + x[i] / y[i];$  [12]

$\} \text{ while } (i > 100);$

b) Explain the following terms with respect of simple code generation algorithm: [6]

i)Register Descriptor

ii)Address Descriptor

Q11)a)Is there any difference between Class-based languages and object-based languages? If yes, justify the answer. [6]

b) How can overloading and overriding of functions in object oriented programming languages handle by Compiler? Explain in detail. [10]

OR

Q12) a) Explain different types of inheritance with example. [8]

b) Discuss the features of OOP language and its benefits. [8]