

<b>A-9073</b>
---------------

<b>Sub. Code</b>
------------------

<b>4MCI3C1</b>
----------------

**M.Sc. DEGREE EXAMINATION, NOVEMBER 2019**

**Third Semester**

**Computer Science and Information Technology**

**PRINCIPLES OF COMPILER DESIGN**

**(CBCS – 2014 onwards)**

Time : 3 Hours

Maximum : 75 Marks

**Part A**

(10 × 2 = 20)

Answer **all** questions.

1. Define Syntax.
2. What is meant by Input buffering?
3. Define parsing.
4. Define recursion.
5. What is meant by translation?
6. Write any two differences between Abstract and concrete syntax.
7. What is meant by Storage organization?
8. What is meant by case statements?
9. What is Run time storage management?
10. What is a flow graphs?

**Part B****(5 × 5 = 25)**

Answer **all** the questions, choosing either (a) or (b).

11. (a) Write short note on Specification of Tokens in lexical analysis.

Or

- (b) Explain identifiers and keywords with examples.

12. (a) Differentiate between SLR parsing and LALR parsing.

Or

- (b) Give the differences between Top Down parsing and simple Bottom Up parsing.

13. (a) Explain in short on the Construction of syntax trees.

Or

- (b) Write short notes on Abstract and concrete syntax with examples.

14. (a) Write short notes on Intermediate languages.

Or

- (b) Discuss on Storage allocation strategies.

15. (a) Explain briefly on loop optimization in code generation phase.

Or

- (b) What is meant by

- (i) Run time storage management
- (ii) Targets of code generator.

**Part C** $(3 \times 10 = 30)$ Answer any **three** questions.

16. Discuss in detail the role of Lexical Analyzer.
  17. Give an account on bottom up and Shift reduce parsing in compiler design.
  18. Explain in detail the following:
    - (a) S-attributed
    - (b) L-attributed.
  19. Elaborate on intermediate code generation with an example.
  20. Discuss on issues in design of code generator.
-