Code No: RT31051 **R13** SET - 1

**III B.Tech I Semester Supplementary Examinations, October/November - 2019**

**COMPILER DESIGN**

(Computer Science and Engineering)

Time: 3 hours Max. Marks: 70

Note: 1. Question Paper consists of two parts (**Part-A** and **Part-B)**
2. Answering the question in **Part-A** is compulsory
3. Answer any **THREE** Questions from **Part-B**

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

**PART –A** **(22 Marks)**

| | | | |
|---|---|---|---|
| 1 | a) | What are the features of a Lexical analyzer? | [3M] |
| | b) | Explain in brief about left most and right most derivations. | [4M] |
| | c) | List out the rules for FIRST and FOLLOW. | [3M] |
| | d) | Describe in brief about types of LR parsers. | [4M] |
| | e) | What is common sub expression elimination? | [4M] |
| | f) | Discuss about Instruction Selection and Register allocation. | [4M] |

**PART –B** **(48 Marks)**

| | | | |
|---|---|---|---|
| 2 | a) | Define Compiler? Explain in brief about various language processing tools. | [4M] |
| | b) | Construct a Finite Automaton for the Regular Expression (00+11)*? | [8M] |
| | c) | Differentiate between NFA and DFA. | [4M] |
| 3 | a) | Discuss in brief about LL(1) Grammars. | [3M] |
| | b) | Differentiate between Top down and bottom up parsing techniques. | [8M] |
| | c) | Construct FIRST and FOLLOW for the Grammar: E→E+T/T, T→T*F/F, F→(E)/id. | [5M] |
| 4 | a) | Construct LALR Parsing table for the grammar S→L=R/R, L→*R/id, R→L. | [8M] |
| | b) | Define Ambiguous Grammar? Check whether the grammar S→aAB, A→bC/cd, C→cd, B→c/d, is Ambiguous or not? | [8M] |
| 5 | a) | Define Intermediate code generator. Explain in brief about different forms of Intermediate code generation. | [8M] |
| | b) | Explain in brief about Type checking and Type Conversion. | [8M] |
| 6 | a) | Differentiate between Static and Dynamic Storage allocation Strategies. | [8M] |
| | b) | What is dangling Reference in storage allocation? Explain with an Example. | [8M] |
| 7 | a) | Explain in brief about peephole optimization techniques. | [8M] |
| | b) | What is a Flow Graph? Explain how a given program can be converted in to a Flow graph? | [8M] |

*****
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~