FACULTY OF INFORMATICS
COMPILER CONSTRUCTION (Elective – I)

Time: 3 Hours ] [ Max. Marks: 75

Note: Answer all questions from Part – A. Answer any Five questions from Part – B.

PART – A

1. Differentiate between single pars and multipars compilers.
2. What is a crors compiler?
3. What is left recursion? Remove left recursion from exp → exp + term / term.
4. Write LR(o) items for the following grammar
   \[ S \rightarrow aAb / d \]
   \[ A \rightarrow AeS / S \]
5. What is a activation record? What are its contents?
6. Enumerate the ways, a symbol table be organized.
7. What is dynamic loading? What are its advantages?
8. What are attribute grammars?
9. Define FIRST and FOLLOW sets.
10. What is meant by dead code?

PART – B

11. (a) Explain about various data structures used in a compiler. Specify where they are used.
    (b) Minimize the following DFA.
    
    | State | Input | Symbol |
    |-------|-------|--------|
    | A     | a     | B      |
    | B     | b     | C      |
    | C     |       | B      |
    | *D    | b     | E      |
    | E     |       | C      |

    Where A is start state B and D are final states.

12. Construct LL(1) parsing table for the following grammar:
    \[ \text{exp} \rightarrow \text{exp addop term} / \text{term} \]
    \[ \text{addop} \rightarrow + / - \]
    \[ \text{term} \rightarrow \text{term mulop factor} / \text{factor} \]
    \[ \text{mulop} \rightarrow * \]
    \[ \text{factor} \rightarrow (\text{exp}) / \text{number} \]

(This paper contains 2 pages)
13. Construct SLR (1) parsing table for the following grammar:
   \[ S \rightarrow I \mid \text{other} \]
   \[ I \rightarrow \text{if } S \mid \text{if } S \text{ else } S \]

14. Write the attribute grammar for the following grammar and also draw the parse tree for the string \( w = \text{float } x, y \).
   \[ \text{decl} \rightarrow \text{type } \text{var-list} \]
   \[ \text{type} \rightarrow \text{int } \mid \text{float} \]
   \[ \text{var-list} \rightarrow \text{id } \mid \text{var-list } \text{id} \]

15. (a) Distinguish between static and dynamic storage allocations of a language.
   (b) Explain how a hash table can be used to implement a symbol table.

16. (a) Explain about various code optimization techniques with an example.
   (b) Write three address code and P-code for the following control statements:
   (i) if (E) \( S_1 \) else \( S_2 \)
   (ii) while (E) \( S \)

17. Write short notes on:
   (a) Error handling in top-down parsers.
   (b) Code generation from DAGs.
   (c) Semantic analysis.