PART – A (25 Marks)

1. List the major data structures used in a compiler.
2. What is the role of a lexical analyzer in a compiler?
3. Write the regular definition for the language of C identifiers.
4. Write the types of errors that occur in programming.
5. Remove the left recursion from \( E \rightarrow E + n/n \)
6. Define synthesized and inherited attributes.
7. Draw DAG for the following expression
   \[ a + a*(b-c) + (b-c)*d \]
8. What is short circuit code?
9. Write the design goals of a garbage collector.
10. List the techniques used for semantic preserving transformation.

PART – B (5x10 = 50 Marks)

11. (a) Explain the translation process with a neat diagram.
    (b) Give a brief description about lex.

12. (a) Compute FIRST and FOLLOW sets for the following grammar

\[
\begin{align*}
S & \rightarrow IEISS'/a \\
S' & \rightarrow eS/E \\
E & \rightarrow b. \\
T & \rightarrow T*F / F \\
F & \rightarrow (E) / id
\end{align*}
\]

(b) Write the algorithm for the construction of predictive parsing table.

13. Explain the process of shift-reduce parsing on the i/p string \( id_1 * id_2 \) for the following grammar.

\[
\begin{align*}
E & \rightarrow E + T / T \\
T & \rightarrow T*F / F \\
F & \rightarrow (E) / id
\end{align*}
\]

14. (a) Translate the assignment statement \( a[i]= b*c - b*d \) into quadruples and triples.
    (b) Using goto-avoiding translation scheme translate the expression
    
    \[
    \text{if}(a = b \& \& c = \text{"d"} \& \& \text{"e"} = f) \ x = 1.
    \]

15. (a) Discuss the issues in the design of a code generator.
    (b) Generate code for the following sequence assuming x,y,z are in memory locations.
        if \( x < y \) go to \( L_1 \)
        \( z = 0 \)
        go to \( L_2 \)
        \( L_1 : z = 1 \)

16. (a) Explain the data-flow analysis framework.
    (b) Explain the design of absolute loader.

17. Write short notes on:
    a) Parser generator YACC
    b) Conflicts in shift-reduce parsing.
    c) Rules for type checking.

****