2022

MCA

2nd Semester Examination

COMPILER DESIGN

PAPER—MCA-204

Full Marks : 100

Time : 3 Hours

The figures in the margin indicate full marks.

Candidates are required to give their answers in their own words as far as practicable.

Illustrate the answers wherever necessary.

Group—A

1. Answer any *five* questions : 5×2

(a) Define Ambiguous grammar.

(b) State the problems in Top Down Parsing.

(c) Differentiate tokens, patterns and lexeme.

(d) Write down the five properties of compiler.

(e) What is code optimization?

(f) What is the function of symantic analysis?

(g) What are the functions of error handler?

(h) Define annotated parse tree?

Group—B

Answer any *four* questions. 4×15

2. (a) Let G be a Context Free Grammar for which the production Rules are given below :

$$S \rightarrow aB \mid bA$$
$$A \rightarrow a \mid aS \mid bAA$$
$$B \rightarrow b \mid bS \mid aBB$$

Drive the string aaabbabbba using the above grammar (using Left Most Derivation and Right most Derivation).

(b) Consider the following Grammar :

$$A \rightarrow ABd \mid Aa \mid a$$
$$B \rightarrow Be \mid b$$

Remove left recursion.

(c) Do left factoring in the following grammar :

$$A \rightarrow aAB \mid aA \mid a$$
$$B \rightarrow bB \mid b$$

5+5+5

3. (a) Calculate FIRST and FOLLOW for the following grammar :

$$S \rightarrow xABC$$
$$A \rightarrow a \mid bbD$$
$$B \rightarrow a \mid \varepsilon$$
$$C \rightarrow b \mid \varepsilon$$
$$D \rightarrow c \mid \varepsilon$$

(b) Construct predictive parsing table for the given grammar in 3(a). Show that the given grammar is LL(1) or not.

(c) Show how the predictive parser parses the string "xbbcab". 5+5+5

4. (a) Consider the following grammar :

$$S \rightarrow Aa \mid bAc \mid Bc \mid bBa$$
$$A \rightarrow d$$
$$B \rightarrow d$$

Compute closure and goto..

(b) Construct SLR parsing table for the grammar given in 4(a).

(c) Show how the SLR parser parses the string (i) 'bdc' and (ii) 'bda'. 5+5+5

5. (a) What is intermediate code and write the two benefits of intermediate code generation ?

(b) Draw the syntax tree and DAG for the following expression :

$$(a * b) + (c - d) * (a * b) + b$$

(c) Write quadruples, triples and indirect triples for the expression :

$$-(a * b) + (c + d) - (a + b + c + d)$$

(2+2)+(2+3)+(2+2+2)

6. (a) What are basic blocks ? Write the algorithm for partitioning into Blocks.

(b) What is flow graph ? Give one example.

(c) What is common sub-expression and how to eliminate it ? Explain with example.

(d) Define Dead-code elimination with example.

(2+3)+(2+2)+3+3

7. (a) Explain the Type Checking with suitable examples.

(b) Differences between SLR, CLR, LALR parsers.

(c) Construct CLR Parsing table for the given grammar

$$S \rightarrow CC$$
$$C \rightarrow aC/d$$

5+5+5

8. (a) Explain syntax directed definition with simple examples.

(b) What is control and data flow analysis ? Explain with example.

(c) What are the properties of code generation phase ?

(d) Define cross compiler. 4+4+4+3

9. Write short note (any *three*) : 3×5

(a) Bootstrapping ;

(b) Handle pruning ;

(c) Dependency graph ;

(d) Symbol Table ;

(e) Phases of a compiler.

[Internal assessment - 30]