Question Paper

Exam Date & Time: 09-Jan-2024 (02:30 PM - 05:30 PM)

MANIPAL ACADEMY OF HIGHER EDUCATION

FIFTH SEMESTER B.TECH. DEGREE EXAMINATIONS - JANUARY 2024
SUBJECT: CSE 3151/CSE_3151 - COMPILER DESIGN

Marks: 50 **Duration: 180 mins.**

Answer all the questions.

1A) A lexical analyzer uses the following patterns P1, P2 and P3 to recognize three different tokens. (5)

P1: $(ab|bc|ac)^+a$
P2: $(a|b)^*bc(b|c)^+$
P3: $(ab)^+(ac|bc)$

Build the combined NFA and show the sequence of states entered while processing the input string ababbcbca$. What is the sequence of patterns that the lexical analyzer outputs? Justify your answer.

1B) Given, u, v, x, y, z are integer variables, determine the output of the different phases of compiler for the following expression: (3)

$(-u) + v / x / y - z \% 8$

Note: Use the mnemonics DIV and MOD in code generation phase to perform / and % operations respectively.

1C) Write and explain the structure of a Lex program. Discuss the various stages involved in generation of the tokens from the lex file. (2)

2A) Construct the LR(1) automaton for the given Grammar. $S \rightarrow S + S \mid S * S \mid (S) \mid num$ (5)

2B) When do we say that an error has been encountered in predictive parsing? With an example, explain the possible ways of selecting a synchronizing token set to recover from an error in panic mode recovery. (3)

2C) Develop a BISON program to check the validity of declaration grammar in C. (2)

program → data-type main () { declarations ; }
declarations→ data-type identifier-list; declarations | ∈
data-type → int | char
identifier-list → id | id, identifier-list
identifier-list → id [number] | id [number], identifier-list

3A) Construct LR(0) automata and SLR(1) parse table for the grammar given below. (5)

$S \rightarrow T \mid AR$
$T \rightarrow xTy \mid \varepsilon$
$R \rightarrow yRz \mid \varepsilon$
$A \rightarrow Ax \mid \varepsilon$

Show parsing action for **xxyz$.**

3B) For the given grammar, construct an annotated parse tree for evaluating the input expression 4+5*6, and write the semantic rules for each production in the grammar. (3)

$L \rightarrow En$

$E \rightarrow TE'$

$E' \rightarrow +TE' \mid \varepsilon$

$T \rightarrow FT'$

$T' \rightarrow *FT' \mid \varepsilon$

$F \rightarrow (E) \mid digit$

3C) With the help of an example, describe the role of Syntax Directed Translation (SDT) in compiler design. (2)

4A) Consider the following C fragment: (5)

```
for(j=0;i<n;j++)
{
 for(j=0;j<n;j++)
  {
   A[j]=A[j]+B[j];
  }
}
```

Write equivalent three address code assuming array elements are 4 bytes. Also write quadruple form for the above code.
(3+2 = 5 marks)

4B) Construct Abstract Syntax Tree (AST), Directed Acyclic Graph (DAG) and implement DAG using the value numbers method for the following expression (3)
((a + a)+a) + ((a + a)+a)

4C) How DAG improves efficiency? Justify your answer with an example. (2)

5A) Construct the three address code, find the leaders and draw flow graph by identifying basic blocks for the following C code. (5)

```
while (p< q) {
if(m< n)
r = s +t ;
else
r = s-t;}
```

5B) Show the steps for constructing Abstract Syntax Tree (AST) using Syntax Directed Definition (SDD) for the expression a= (a+b) * (c+d) + (a+b+c) and display AST. (3)

5C) Generate the assembly level code for the following code and calculate cost for each assembly level instruction. (2)

```
s = 0
i = 0
L1: if i > 10 goto L2
s = s + i
i = i + 1
goto L1
L2:
```

-----End-----