Nirma University

Institute of Technology

Semester End Examination (IR/RPR), Dec-2023

B.Tech. in Computer Science & Engineering, Semester -VII

2CS701-O Compiler Construction

Roll / Exam No. _____

Supervisor's Initial with Date _____

Time: 3 Hours | Max Marks: 100

Instructions:

1. Attempt all the questions.
2. Figures to right indicate full marks.
3. Use section-wise separate answer book.
4. Draw neat sketches wherever necessary.
5. Assume suitable data wherever required and note down in the answer book.

SECTION-I

Q-1	**Answer the following questions:**	**[18]**
(A) CLO3	Explain the various phases of compiler with following input: a = b + c * 20.0;	[6]
(B) CL03	Prepare recursive recent parser pseudo code for the following grammar, *bexpr → bexpr or bterm | bterm* *bterm → bterm and bfactor | bfactor* *bfactor → not bfactor | (bexpr) | true | false* Also parse the string - not (true or false).	[6]
	OR	
(B) CLO3	Prepare operator precedence table for grammar, S → aA A → Ab|c B → e|f Parse the string **"acb"** using it	[6]
(C) CLO3	Draw minimized DFA for given expression using syntax tree / followpos() method. (0+1)*1 0*10*#	[6]
	OR	

(C) CLO3 Eliminate Left recursion from given grammar. [6]

$S \rightarrow A$

$B \rightarrow bBc \mid f$

$A \rightarrow Ad \mid Ae \mid aB \mid ac$

Also perform left factoring on resultant grammar.

Q-2 **Answer the following questions:** **[16]**

(A) CLO1 "Error recovery is required during lexical and syntax analysis" State whether the given statement is true or false. Justify your answer with proper error recovery strategies and example. [8]

(B) CLO2 Construct LL(1) parsing table for the following grammar: [8]

$S' \rightarrow S \#$

$S \rightarrow qABC$

$A \rightarrow a \mid bbD$

$B \rightarrow a \mid \epsilon$

$C \rightarrow b \mid \epsilon$

$D \rightarrow c \mid \epsilon$

Q-3 **Answer the following questions:** **[16]**

(A) CLO3 Construct SLR parsing table for following grammar: [8]

$E \rightarrow BB$

$B \rightarrow cB \mid d$

(B) CL02 Justify with example: ambiguous, left recursive and non-left factored grammars cannot be LL(1) grammars. [8]

Section II

Q.4 **Do as directed** **[18]**

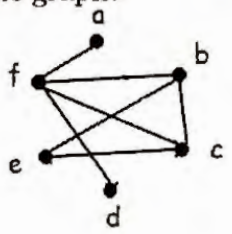
(A) CLO1 "If a S/R conflict occurs in LALR (1) table then definitely same S/R conflict occurred in CLR(1) table also". Whether the given statement is true or false. Justify your answer with proper reason and example. [9]

OR

(A) CLO1 Construct the DAG for the given below expression [9]

$((x + y) - ((x + y) / (x - y))) + ((x + y) * (x - y))$

(B) CLO3 Perform optimal registers allocation using graph coloring method for below given register interference graph. Show register name for each variable shown in the graph. How many minimum registers required for the registers given in the graph? [9]



OR

(B) CLO2 Convert the following C code into 3 address code: [6]

```
fact(x)
{
   int f = 1;
   for (i = 2; i <= x; i++)
      f = f * i;
   return f;
}
```

Q.5 **Do as directed** **[16]**

(A) CLO2 For the following grammar, write semantic rules to create Abstract Syntax tree and design an AST for the string 5 + 3 * 4. [8]

$E \rightarrow E_1 + T$
$E \rightarrow T$
$T \rightarrow T_1 * F$
$T \rightarrow F$
$F \rightarrow (E)$
$F \rightarrow$ **digit**

(B) CLO3 What do you mean by code optimization? Explain the following code optimization methods with suitable example. [8]

1. Loop Optimization
2. Peephole Optimization
3. Dead code Elimination
4. Common Sub expression elimination

Q.6 **Do as directed** **[16]**

(A) CLO3 Which are the intermediate representations used in compiler designs, write quadruples for the following code a = -b * c + d. [8]

(B) CLO4 Design CLR parser for the following grammar and trace the string: abab [8]

S → AaAb
S → BbBa
A → ε
B → ε