MARRI LAXMAN REDDY
INSTITUTE OF TECHNOLOGY AND MANAGEMENT (FN)
(AN AUTONOMOUS INSTITUTION)
(Approved by AICTE, New Delhi & Affiliated to JNTUH, Hyderabad)
Accredited by NBA and NAAC with 'A' Grade & Recognized Under Section2(f) & 12(B)of the UGC act,1956

III B.Tech I Sem Regular End Examination, December 2022
Automata Theory and Language Processors
(CSM)

Time: 3 Hours.   **Max. Marks: 70**

Note: 1. Question paper consists: Part-A and Part-B.
2. In Part – A, answer all questions which carries 20 marks.
3. In Part – B, answer any one question from each unit.
   Each question carries 10 marks and may have a, b as sub questions.

PART- A
                                             (10*2 Marks = 20 Marks)

|				
1. a)	Write the regular expression that represents Identifiers	2M	CO1	BL1
b)	Differentiate between NFA and DFA	2M	CO1	BL2
c)	Differentiate compiler and interpreter.	2M	CO2	BL2
d)	Compare various LR Parsers.	2M	CO2	BL2
e)	What is Ambiguous grammar? Give Example.	2M	CO3	BL1
f)	Differentiate synthesis and inherited attributes.	2M	CO3	BL2
g)	Why are quadruples preferred over triples in an optimizing compiler?	2M	CO4	BL1
h)	What is common sub expression elimination?	2M	CO4	BL1
i)	For the code given in 1.(J) generate the basic blocks and write the rules.	2M	CO5	BL1
j)	Generate three address code for the given pseudo code	2M	CO5	BL3

```
while(i<=100)
{ A=A/B*20; ++i;
print(A value) }
```

PART- B
                                             (10*5 Marks = 50 Marks)

2 a)	Given **1 (0 + 1)* 0** regular expression construct a NFA with and without epsilon transition?	5M	CO1	BL3
b)	Generate the regular expressions for Identifiers and Constants in C language and convert them to NFA?	5M	CO1	BL3

OR

3 a)	How to recognize various tokens of high level language program? Write the regular expressions and transition diagrams for each.	5M	CO1	BL1
b)	Convert the given NFA to DFA.	5M	CO1	BL3
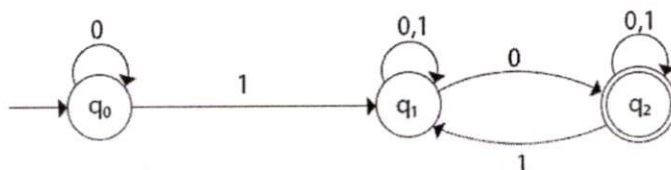
4 a)	Construct a Predictive parsing table for the Grammar E-> E+T/T ; T-> T*F/F; F-> (E)/id .	5M	CO2	BL3

b)	Construct FIRST and FOLLOW for the Grammar: E->E+T/T, T->T*F/F, F->(E)/id.	5M	CO2	BL3

OR

5	a)	Eliminate left recursion in the following grammar A → ABd | Aa | a, B → Be | b	5M	CO2	BL3
	b)	Differentiate between Top down and bottom up parsing techniques.	5M	CO2	BL2

6	a)	Explain in brief about Type checking and Type Conversion	5M	CO3	BL4
	b)	Give a grammar G	5M	CO3	BL3

Number	→	Sign List
Sign	→	+
	|	-
List	→	List Bit
	|	Bit
Bit	→	0
	|	1

Find the synthesized and inherited attributes of corresponding augmented grammar of G that compute the decimal value of signed binary number.

OR

7	a)	Explain about Chomsky hierarchy about different grammars with suitable examples	5M	CO3	BL4
	b)	Consider the grammer given in 6.(b) add the rules to compute the decimal value of signed binary number	5M	CO3	BL3

8	a)	Translate the expression –(a+b)*(c+d)+(a+b+c) in to quadruple, triple and indirect triple. And list advantages and disadvantages.	5M	CO4	BL5
	b)	What is a Flow Graph? Explain how a given program can be converted in to a Flow graph?	5M	CO4	BL4

OR

9	a)	Translate the given expression into Quadruples, triples and indirect triples (a+b)*(c+d)+(a*b/c)*b+60. And list advantages and disadvantages.	5M	CO4	BL5
	b)	Explain in brief about peephole optimization techniques.	5M	CO4	BL4

10	a)	How the intermediate code is transformed into target object code?	5M	CO5	BL1
	b)	Construct the DAG for below piece of code $t_1 = a + b$; $t_2 = t_1 + c$; $t_3 = t_1 \times t_2$	5M	CO5	BL3

OR

11	a)	Given a below flow graph find the minimum number of registers are required to execute the code.	5M	CO5	BL3
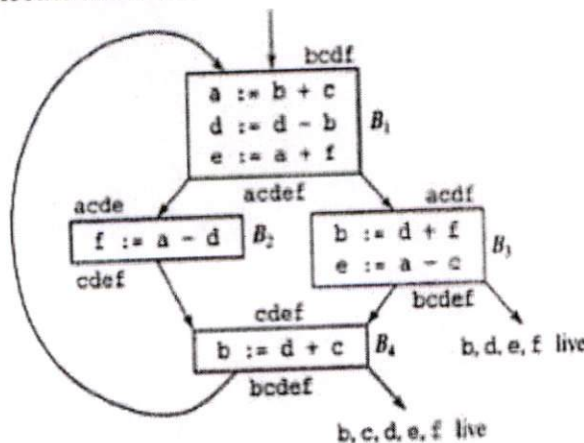
	b)	For the above graph generate the machine code using assigned registers.	5M	CO5	BL3

---oo0oo--

CO - Course Outcome

BL - Blooms Taxonomy Levels