| USN | 1 | M | S | 2 | 1 | C | I | 0 | 4 | 9 |
|---|---|---|---|---|---|---|---|---|---|---|

RAMAIAH
Institute of Technology

(Autonomous Institute, Affiliated to VTU)
(Approved by AICTE, New Delhi & Govt. of Karnataka)
Accredited by NBA & NAAC with 'A+' Grade

# SEMESTER END EXAMINATIONS – FEBRUARY 2024

| | | | | | |
|---|---|---|---|---|---|
| Program | : | B.E. – CSE (Artificial Intelligence and Machine Learning) / CSE (Cyber Security) | Semester | : | V |
| Course Name | : | Automata Theory and Compiler Design | Max. Marks | : | 100 |
| Course Code | : | CI53/CY53 | Duration | : | 3 Hrs |

**Instructions to the Candidates:**

- Answer one full question from each unit.

## UNIT - I

1. a) Design DFAs to accept the following languages. — CO1 (06)
   i. String beginning with 'bab' over $\Sigma = \{a,b\}$.
   ii. String not ending with 011 over $\Sigma = \{0,1\}$.
   iii. String starting with 'aa' and ending with 'bb'

   b) Assume you and your team wants to design a new computer programming language. For this proposed language, list out the phases of compilers in their logical sequence and explain their roles by taking a suitable example. — CO2 (08)

   c) Convert the following NFA to a DFA over $\Sigma = \{0,1\}$. — CO1 (06)

| NFA | € | 0 | 1 |
|---|---|---|---|
| → Q0 | φ | φ | Q1 |
| * Q1 | Q2 | Q1 | Q2 |
| Q2 | φ | φ | Q0 |

2. a) Describe the components of DFA and illustrate how to minimize DFA with an example. — CO1 (06)

   b) Consider the following mathematical statement, a= b*c + 50.0 (assume all variable are of integer type). List the phases of compiler and map the following statement to each phase and its processing. — CO2 (08)

   c) Design an DFA (Deterministic finite automata) to accept the set of strings of 0's and 1's that either — CO1 (06)
   (i) end in 010 and have 011 somewhere preceding, or
   (ii) end in 101 and have 100 somewhere preceding.

## UNIT - II

3. a) Consider the following DFA. — CO4 (08)
   E = ( Q , Σ , δ , q0 , F), Q ={ p1,p2,p3},Σ = { 0,1},q0 = p1 , F = {p3} and δ is as follows:
   δ(p1,0)=p2,δ(p1,1)=p1,δ(p2,0)=p3,δ(p2,1)=p1,δ(p3,0)=p3,δ(p3,1)=p2.
   Convert the above DFA to regular expression. Construct the transition table.

   b) Write a transition diagram to recognize the lexemes matching the relational operators. — CO2 (08)

   c) Describe the languages specified by regular expression:- — CO4 (04)
   i) a(a|b) * a ii) (a|b) * abb.

4. a) Design transition diagram for i) arithmetic operators ii) if , ifelse , nestedif, nestedfor — CO2 (08)

   b) Explain the role of lexical analyzer with a neat diagram. — CO2 (08)

   c) Write regular expression for the following Language: — CO4 (04)
   Strings of 0's and 1's with substring '0011'

## UNIT - III

5. a) Consider the following Context Free Grammar CO3 (09)

**S→SA |0 |ε**

**A→aS1 | a**

i. Compute the FIRST and FOLLOW sets for each non-terminal symbol.

ii. Construct the parsing table for a non-recursive predictive parser for the above grammar.

iii. Is the grammar LL (1)? Justify.

b) Examine the need of Augmentation in LR grammars and how the given grammar can be changed to an augmented grammar? CO3 (05)

c) Eliminate left recursion from the grammars given. CO3 (06)

i. S →aAbA | Abc | ScA

A →aAbab | Sb | c

ii. A →B x y | x

B→C D

C → A | c

D →d

6. a) Prove that the given Context Free Grammar is LR (1) by generating the parse table. CO3 (09)

**S→AaAb | BbBa**

**A→ε**

**B→ε**

Show the actions made by the parser on validating the input string **'ab$'**.

b) Write down the rules to check whether a grammar is LL (1) or not. Give suitable example of a LL (1) grammar. CO3 (06)

c) Construct CFG for accepting the strings containing alphabets a and b, which will start with exactly one 'a' followed by any number of 'b's. CO3 (05)

## UNIT- IV

7. a) Describe the language of a Turing Machine. Specify the Turing Machine and halting problem. CO4 (07)

b) Explain the rules for turning an L-attributed SDD to SDT. Convert the SDD given in Fig 7(b) below to SDT. CO5 (07)

| Production | Semantic Rules |
|---|---|
| D->B D' | D.val=D'.syn<br>D'.inh=B.val |
| D'->B $D_1$' | D1'.inh=D'.inh*2+B.val<br>D'.syn=D1'.syn |
| D'->ε | D'.syn=D'.inh |
| B->0 | B.val=0 |
| B->1 | B.val=1 |

Fig 7(b): SDD

c) Write a S-attributed definition for the given CFG to count the number of 1's in the given binary value. Draw an annotated parse tree for the same. CO5 (06)

**G: N→D**

**D→$D_1$ B**

**D→B**

**B→0**

**B→1**

8. a) Describe about the side effects involved in the translation of semantic rules. Indicate the possible ways to control the side effects. Identify the side effects involved in the translation given in Fig 8(a). CO5 (06)

| Input | Translation |
|---|---|
| 9 + 5 + 2 | S → T R |
| 9 + 5 + 2 | T → num {print(num.val);} |
| + 5 + 2 | R → + T {print('+');} R |
| 5 + 2 | T → num {print(num.val);} |
| + 2 | R → + T {print('+');} R |
| 2 | T → num {print(num.val);} |

Fig 8(a): Translation Scheme.

b) What is an attribute called if it is associated with the nonterminal symbol on the left of a CFG rule, and its value is calculated using the attributes of the symbols on the right of the CFG rule? Explain with an example. CO3 (06)

c) Write a L-Attributed SDD used for Simple desk calculator. Obtain the Semantic action, Annotated parse tree and dependency Graph for the string **3*5+4n**. CO5 (08)

## UNIT- V

9. a) Translate the following control flow statement with the help of semantic actions of one pass code generation. CO5 (08)

**if(a<b) a=1; else a=0;**

b) Construct 3AC and Annotated Parse Tree for CO5 (06)

i. **x=a[i]+b[j]**
ii. **x=c+a[1][2]**

c) Determine the types and relative address for the following: CO5 (06)

```
float x;
record{ int t;
        int y;
        int z;
}q;
```

10. a) Construct DAG, Syntax Tree, 3AC and Quadruple representation for CO5 (09)

i. **x=a+b*-c+b*-c**
ii. **x=a+b+b+a+a*b**
iii. **x=(a+b)*c+(d+a)-(a+b)**

b) Write the three address code and quadruple representation for the given statements: CO5 (06)

i. f=min(1,n-1,n+1)
ii. a=x[i]+b*c

c) Explain the implementation of getReg(I) function, for a three-address instruction, such as **I: a = b + c.** CO4 (05)

*********************************