Code No: 155FG **R18**

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
B. Tech III Year I Semester Examinations, March - 2024
FINITE AUTOMATA AND COMPILER DESIGN
(Common to CSE(IOT), CSE(N))

Time: 3 Hours **Max. Marks: 75**

Note: i) Question paper consists of Part A, Part B.
ii) Part A is compulsory, which carries 25 marks. In Part A, Answer all questions.
iii) 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

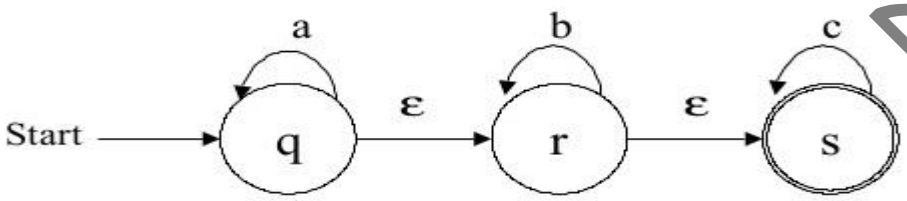
(25 Marks)

1.a)	How can an NFA be converted into a DFA?	[2]
b)	What is the key difference between lexical analysis and parsing?	[3]
c)	How does pruning affect the completeness and correctness of LR parsing?	[2]
d)	What is an Abstract Syntax Tree (AST) and its significance in compiler construction?	[3]
e)	Compare and contrast context-sensitive features with context-free features in Programming languages.	[2]
f)	Define type checking and its importance in programming language design.	[3]
g)	What is activation record? Write the various fields of Activation Record.	[2]
h)	Define basic blocks in the context of compiler optimization.	[3]
i)	Define object code and its role in the compilation process.	[2]
j)	What are the applications of DAG?	[3]

PART – B

(50 Marks)

2.a) Convert the following NFA with ε moves to DFA without ε moves.



b) Explain how input buffering helps lexical analyzer in compilation process. [5+5]

OR

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

S -> aB|bA

A -> a|aS|bAA

B -> b|bS|aBB

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

b) Construct LL(1) parse table for the given grammar: [5+5]

S→ ABC
A →a A| C
B → b
C → c

4.a) Discuss the components of a YACC specification file.

b) Construct a LALR parsing table for following grammar: [5+5]

S'->S
S->CC
C->cC/d

OR

5.a) Explain the concept of attribute grammars and their role in syntax-directed translation.

b) Write quadruples, triples and indirect triples for the expression: [5+5]

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

6.a) Explain the four types of grammars in the Chomsky hierarchy and provide examples for each.

b) Discuss how type equivalence is determined for simple and complex data types. [5+5]

OR

7.a) Explain how type conversion is handled in strongly typed versus weakly typed languages.

b) Explain how overloading resolution is performed during compilation. [5+5]

8.a) Discuss the static storage allocation strategy with example and its limitations.

b) Discuss the challenges associated with memory leaks and dangling pointers in dynamic Storage allocation. [5+5]

OR

9.a) Describe peephole optimization and its role in improving code quality.

b) Write pseudocode for finding sum of 'n' numbers and identify basic blocks then construct the flow graph for it. Explain the rules used for this. [5+5]

10.a) Explain the design issues for code generator phase.

b) Explain the importance of register allocation for optimizing program performance. [5+5]

OR

11.a) Discuss different forms of object code and their advantages in various scenarios.

b) Provide an example of how a DAG can be transformed into optimized code during code generation. [5+5]

---ooOoo---