College of Engineering and Technology (CEAT)
Department of Computer Science and Engineering

Assignment on DFA
Summer Semester 2025
Program: Bachelor of Computer Science and Engineering (BCSE)
Course Instructor: Md Nazir Ahmed

Course No.: CSC 397
Course Title: Theory of Computation **Full Marks: 100**
Section: I, J **Submission Date: 14/09/2025**

Target CO3, PO(b): Problem Analysis

CO3	Analyze and develop finite automata using formal computational models for a problem	
PO(b)	**K2, K3, K4**	**P1, P2, P3**

Note: 1) This assignment should be undertaken by the students individually. **2)** Assessment of this assignment will be done through the rubrics.

Objective:
To evaluate students' ability to design and implement a Deterministic Finite Automaton (DFA)–based access control system for a smart building, identifying realistic requirements, analyzing constraints, and developing a secure and deterministic solution.

Scenario:

A modern smart building requires a secure and automated **Access Control System** to regulate entry to different zones such as the Lobby, Laboratory, Server Room, Executive Lounge, Research Wing etc. The system must authenticate users through various input events such as card swipes, fingerprint scans, retina scans, face recognition, voice recognition, keypad PIN entries, biometric combinations, or administrative overrides. Each zone has a unique access policy that specifies a precise sequence of authentication steps before access can be granted.

The Access Control System will be used by building occupants, security personnel, and administrative staff. It must reject any incorrect or out-of-order authentication. Deterministic Finite

Automata (DFA) will be applied to model and implement this logic, ensuring predictable, secure, and verifiable transitions between authentication states.

Expected Features of the Assignment:

1. **Zone Management**
    - Specification of at least eight distinct access zones within a smart building
    - Assignment of unique access policies and authentication requirements to each zone
2. **Authentication Event Handling**
    - Inclusion of at least eight authentication event symbols (e.g., Card, Fingerprint, Retina Scan, PIN Entry, Admin Override, etc.)
3. **Access Policy Configuration**
    - Configuration of unique authentication sequences (minimum four steps) for each zone
    - Enforcement of deterministic state transitions for each input symbol
4. **Authentication Sequence Validation**
    - Detection of invalid or out-of-order authentication sequences
5. **Access Decision Program**
    - Implementation of a program that reads authentication sequences and outputs whether access is granted or denied

Considerations and Constraints

Requirement-Level Considerations and Constraints:

- **Unique authentication sequences for each zone:** Every zone (e.g., Lobby, Server Room, Research Wing) must have a distinct sequence of authentication events that cannot be reused by other zones.
- **Multi-factor authentication requirement:** Access policies must combine at least four or more authentication methods.
- **Immediate rejection of invalid sequences:** Any wrong or out-of-order authentication event should lead directly to a rejection state.

Technical-Level Considerations and Constraints:

- **DFA determinism:** The access control DFA must be fully deterministic—each input symbol leads to exactly one next state.
- **Variable-length sequence handling:** The DFA should handle authentication sequences of varying lengths, with a minimum of four events per accepted policy.

Task for Students:

1. **Consideration of Constraints during Problem Analysis**
    - Analyze both requirement-level and technical-level constraints.
    - Choose an approach to handle each identified constraint and justify your decision.
2. **DFA Design**
    - Define input symbols (authentication events) and states for your DFA.
    - Draw the DFA state diagram and prepare a transition table.
    - Clearly show how your DFA design addresses and resolves the identified constraints.
3. **Implementation in Programming Language**
    - Implement the main DFA features in any programming language (c, c++, java, python), which are reading authentication sequences, processing transitions, showing message for granting or denying access.
    - Ensure that your code addresses all the necessary considerations and resolves the constraints identified during analysis.
    - Include proper handling for invalid symbols and unexpected authentication sequences.
4. **Testing**
    - Test at least 5 real-world scenarios (e.g., correct sequence for a zone, incorrect sequence, partially correct sequence with a wrong final step, extra inputs after access is granted, invalid authentication symbols)
    - Record results in a table: **Test Case → Expected Output → Actual Output → Pass/Fail**.

5. **Documentation**
    - Submit a concise report including:
        - Introduction to DFA and the access control system
        - Problem Analysis & Constraint Resolution Approach
        - State Diagram & Transition Table of DFA
        - Code Implementation Overview
        - Test Results
        - Conclusion
        - Justification of CEP

Submission Requirements:

- Source code files
- State Diagram and Transition Table of DFA (image/PDF format)
- Report in PDF. **Copy Task lead to F**
- Submit all in a ZIP folder named: *StudentID_DFA_Section* and Upload to Google Classroom Drive with **HARD COPY**

Provide CEP Justification

Complex Engineering Criteria	Knowledge Profile	Your Justification by mentioning Section number/ line number of code/chapter number
	K2	
P1	K3	
	K4	
P2		
P3		

The following rubrics will be used for assessing the submission. Please see the next page:

Theory of Computation

Assessment Rubrics

Assessment Criteria	Sub-Criteria (KPA Level)	Weight (%)	Excellent 80-100	Good 60-79	Satisfactory 45-59	Insufficient Below 45
Problem Analysis	Application of computer and information science in problem analysis	10	Relevant computer and information science concepts are applied fully to understand the problem.	Some relevant concepts are applied, but not fully.	Few relevant concepts are applied, but partially.	Concepts are mostly missing or incorrectly applied.
	Formulation of Engineering Fundamentals in problem analysis	20	Engineering fundamentals are clearly formulated and applied to the analysis.	Fundamentals are reasonably formulated or applied.	Fundamentals are partly formulated or applied.	No clear application of engineering fundamentals.
	Application of Engineering Specialist knowledge in problem analysis	20	Engineering Specialist knowledge is clearly applied to	Specialist knowledge is reasonably applied.	Specialist knowledge is partly applied.	No clear application of engineering specialist

			the analysis.			knowledge.
	Identifying raised technical conflicts in problem analysis	50	All possible technical conflicts are identified	Some technical conflicts are identified	Few technical conflicts are identified	Very few or no conflicts are identified
DFA Design	Formulation of Engineering Fundamentals in DFA design	10	Engineering fundamentals are accurately applied to develop the DFA design.	Engineering fundamentals are fairly applied to develop the DFA design.	Engineering fundamentals are poorly applied to develop the DFA design.	Little or no use of fundamentals in DFA design.
	Application of Engineering Specialist knowledge in DFA design	10	Engineering specialist knowledge is accurately applied to develop the DFA design.	Engineering specialist knowledge is fairly applied to develop the DFA design.	Engineering specialist knowledge is poorly applied to develop the DFA design.	Little or no use of specialist knowledge in DFA design.
	Resolving raised technical conflicts in DFA design	20	All raised conflicts are effectively addressed in the design.	Some conflicts are addressed.	Few conflicts are addressed.	Very few or no conflicts are addressed.

	Standard of the proposed design	60	The proposed design is accurate and meets requirements.	The proposed design is somewhat accurate with minor issues.	The proposed design is to some extent accurate with major issues.	The proposed design is inaccurate or incomplete.
Implementation of Functionalities	Implement the features using the concepts of engineering fundamentals	10	All features are implemented correctly using relevant engineering fundamentals.	Some features are implemented correctly.	Few features are implemented correctly.	Very few or no features are implemented correctly.
	Implement the features using engineering specialist knowledge	10	All features are implemented correctly using relevant engineering specialist knowledge.	Some features are implemented correctly.	Few features are implemented correctly.	Very few or no features are implemented correctly.
	The raised conflicts are resolved by implementing	80	All raised conflicts are resolved	Some raised conflicts	Few conflicts are resolved.	Very few or no conflicts are

	the functionalities.		by the implemented functionalities.	are resolved.		resolved by the implemented functionalities or no relevant functionalities are found
Error Handling	Handling the error situations like invalid input symbols or out-of-order authentication sequences	100	All error situations are managed effectively.	Some error situations are managed.	Few error situations are managed.	Very few error situations are managed.
Documentation	Completeness and clarity of documentation	100	Documentation is complete, clear, and well-organized	Documentation is satisfactorily complete but lacks clarity.	Documentation is poorly complete or lacks clarity.	Documentation is incomplete or missing.