MODULE 2: MATHEMATICAL NOTION AND TERMINOLOGY
WEEK 2

Learning Outcomes:

After completing this course you are expected to demonstrate the following:

1. Classify and explain the basic mathematical object, tools, and notation that use in automata theory.

A. Engage

Recall the previous module to refresh their mind about the topic. Ask some questions that are related to the previous topic.

Example question:

1. Who are the two neurophysiologists who presented the paper about finite automata in 1943?
2. What differentiate computability from complexity theories?
3. What is the major objective of automata theory?
4. What are the four major families of automaton?
5. It describe as decidable?

B. Explore

Video Title: **Mathematical Logic**
YouTube Link: https://www.youtube.com/watch?v=OjLhYlh-OrU
Module Video Title: **Week 2 - Mathematical Logic**

C. Explain

The main ideas from mathematics that will be required. While intuition will frequently be our guide in exploring ideas, the conclusions we draw will be based on rigorous arguments. This will involve some mathematical machinery, although the requirements are not extensive.

D. Elaborate

Automata – What is it?

The term **"Automata"** is derived from the Greek word **"αὐτόματα"** which means "self-acting". An automaton (Automata in plural) is an abstract self-propelled computing device which follows a predetermined sequence of operations automatically.

An automaton with a finite number of states is called a **Finite Automaton** (FA) or **Finite State Machine** (FSM).

Related Terminologies

1. **Alphabet**

Definition − An **alphabet** is any finite set of symbols.

Example: ∑ = {a, b, c, d} is an **alphabet set** where 'a', 'b', 'c', and 'd' are **symbols**.

2. **String**

Definition − A **string** is a finite sequence of symbols taken from ∑.

Example: 'cabcad' is a valid string on the alphabet set ∑ = {a, b, c, d}

3. **Length of a String**

Definition − It is the number of symbols present in a string. (Denoted by **|S|**).

Examples:

- If S = 'cabcad', |S|= 6
- If |S|= 0, it is called an **empty string** (Denoted by **λ** or **ε**)

4. **Kleene Star**

Definition − The Kleene star, **∑***, is a unary operator on a set of symbols or strings, **∑**, that gives the infinite set of all possible strings of all possible lengths over **∑** including **λ**.

Representation − $\sum^* = \sum_0 \cup \sum_1 \cup \sum_2 \cup \ldots\ldots$ where $\sum_p$ is the set of all possible strings of length p.

Example: If ∑ = {a, b}, ∑* = {λ, a, b, aa, ab, ba, bb,………..}

5. **Kleene Closure/Plus**

Definition − The set $\boldsymbol{\sum^+}$ is the infinite set of all possible strings of all possible lengths over ∑ excluding λ.

Representation − $\sum^+ = \sum_1 \cup \sum_2 \cup \sum_3 \cup \ldots\ldots$

$\sum^+ = \sum^* - \{ \lambda \}$

Example: If ∑ = { a, b } , $\sum^+$ = { a, b, aa, ab, ba, bb,………..}

6. **Language**

Definition − A language is a subset of ∑* for some alphabet ∑. It can be finite or infinite.

Example: If the language takes all possible strings of length 2 over ∑ = {a, b}, then L = { ab, aa, ba, bb }

Finite Automaton can be classified into two types:

1. Deterministic Finite Automaton (DFA)
2. Non-deterministic Finite Automaton (NDFA / NFA)

Deterministic Finite Automaton (DFA)

In DFA, for each input symbol, one can determine the state to which the machine will move. Hence, it is called **Deterministic Automaton**. As it has a finite number of states, the machine is called **Deterministic Finite Machine** or **Deterministic Finite Automaton.**

Formal Definition of a DFA

A DFA can be represented by a 5-tuple $(Q, \Sigma, \delta, q_0, F)$ where −

1. **Q** is a finite set of states.
2. **∑** is a finite set of symbols calling the alphabet.
3. **δ** is the transition function where $\delta: Q \times \Sigma \rightarrow Q$
4. $\mathbf{q_0}$ is the initial state from where any input is processed ($q_0 \in Q$).
5. **F** is a set of final state/states of Q ($F \subseteq Q$).

Graphical Representation of a DFA

A DFA is represented by digraphs called **state diagram**.

1. The vertices represent the states.
2. The arcs labeled with an input alphabet show the transitions.
3. The initial state is denoted by an empty single incoming arc.
4. The final state is indicated by double circles.
5. The states that not meet or connect in the final state called DEAD states.

Example #1:

Let a deterministic finite automaton be →

1. $Q = \{a, b, c\}$,
2. $\Sigma = \{0, 1\}$,
3. $q_0 = \{a\}$,
4. $F = \{c\}$, and

Transition function δ as shown by the following table –

Present State	Next State for Input 0	Next State for Input 1
a	a	b
b	c	a

Its graphical representation would be as follows:

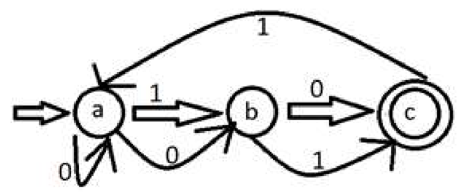

Figure 2.1

Example #2:

Let L = { 101 } as shown in DFA Representation is accepatble or not.

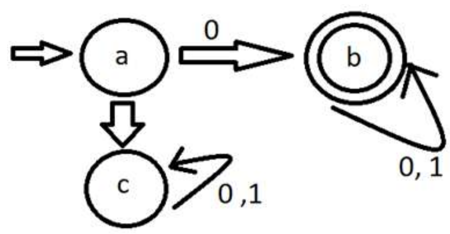

Figure 2.2 DFA Representation

Answer: *L = {101} is NOT ACEPTABLE because of the DEAD state.*

To satisfy the answer, there are two ways to prove it.

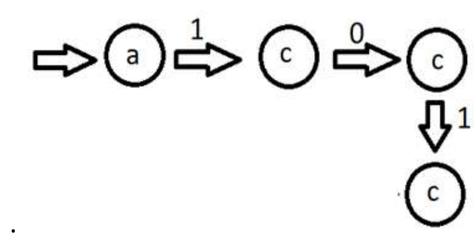

Figure 2.3 Solution Number 1

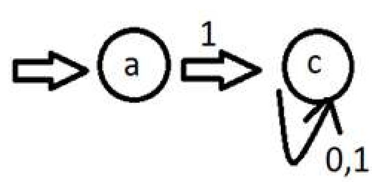

Figure 2.4 Solution Number 2

E. Evaluate

ASSESSMENT:

Instructions: You may write your answer on the Activity Sheet (ACTS) provided in this module.

CONTENT FOR ASSESSMENT:
Activity Number 1 (20-points).

Problem:

a. **Let L = {001}** as shown in DFA Representation is acceptable or not.

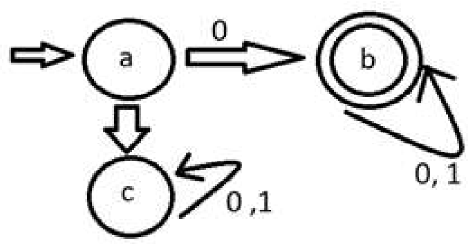

Figure 2.5 DFA Representation

References

1. *An Introduction to Formal Languages and Automata, 5th edition by PETTER LINZ*
2. *YouTube link : https://www.youtube.com/watch?v=OjLhYlh-OrU*

Facilitated By:		
Name	:	
MS Teams Account (email)	:	
Smart Phone Number	:	