Mid Assessment│Summer 2023

Md. Shafayet Hossain

CSE - 21st Batch │ **Course Title:** Theory of Computing

Course Code: CSI - 317 │ **ID:** 2121210071

Answer to the Question no- 1

(a)

Automata

An automaton (automata in plural) is an abstract self-propelled computing device which follows a predetermined sequence of operations automatically.

(b)

Differences between DFA and NFA

DFA	NFA
Deterministic Finite Automata (DFA)	Nondeterministic Finite Automata (NFA)
Building a DFA is difficult	An NFA is simple to construct
NFAs are the source of all DFAs	All NFAs are not DFAs
There is a need for more space allocation	There is less need for space requirement
The DFA transition function is $\delta: Q \times \Sigma \rightarrow Q$	The NFA transition function is $\delta: Q \times \Sigma \rightarrow 2^Q$

(c)

Turing Machine:

Alan Mathison Turing proposed the Turing machine in 1936, a computer model capable of simulating all computational behaviors. A Turing machine is a fictitious machine. The Turing machine was instrumental in the development of the modern computer.

- A Turing machine is a fictitious machine.
- Despite its simplicity, the machine is capable of simulating any computer algorithm, no matter how complex.
- For the sake of simplicity, we will assume that this machine can only process 0, 1 and blank.
- The machine has a head that is positioned over one of the tape's squares.
- The head can carry out the three basic operations listed below:
    1. **Read** the symbols of the square of the head.
    2. **Edit** the symbol by replacing it with a new symbol or erasing it.
    3. **Move** the tape to the left or right by one square so that the machine can read and edit the symbol on the next square.

Answer to the Question no- 2

(a)

The Pigeonhole and letterbox Principle:

Suppose that a flock of 20 pigeons flies into a set of 19 pigeonholes to roost. Because there are 20 pigeons but only 19 pigeonholes, at least one of these 19 pigeonholes must have at least two pigeons in it. To see why this is true, note that if each pigeonhole had at most one pigeon in it, at most 19 pigeons, one per hole, could be accommodated. This illustrates a general principle called the pigeonhole principle, which states that if there are more pigeons than pigeonholes, then there must be at least one pigeonhole with at least two pigeons in it.

Theorem –

I) If "A" is the average number of pigeons per hole, where A is not an integer then

- At least one pigeon hole contains **ceil[A]** (smallest integer greater than or equal to A) pigeons
- Remaining pigeon holes contains at most **floor[A]** (largest integer less than or equal to A) pigeons

II) We can say as, if n + 1 objects are put into n boxes, then at least one box contains two or more objects. The abstract formulation of the principle: Let X and Y be finite sets and let be a function.

- If X has more elements than Y, then f is not one-to-one.
- If X and Y have the same number of elements and f is onto, then f is one-to-one.
- If X and Y have the same number of elements and f is one-to-one, then f is onto.

Pigeonhole principle is one of the simplest but most useful ideas in mathematics. We will see more applications that proof of this theorem.

<u>(b)</u>

Explaination of the notations M= (Q,∑,δ,q0,F):

A Deterministic Finite Automata (DFA) is defined by the 5-tuple

M=(Q, Σ, δ, q0, F) M is the 5 tuple-

where ,

Q - The finite set of internal states.

Σ - The finite set of symbols , the input alphabet.

δ - Transition function δ: Q×Σ→Q.

q0 - An initial state, q0∈Q.

F - A set of final states or accept states, F⊆Q.

The DFA starts in state q0 and reads the first symbol on the input tape. The machine consumes the input symbol and moves to a state specified by the transition function. When the machine hits end input string, it is in one of the final states - input accepted, or it is not in a final state -input rejected.

(c)

Design a state diagram and state table for the grammar "aabaab"

Creating a state diagram and state table for the given grammar "aabaab" involves designing a finite automaton that can recognize this pattern. The grammar seems to describe a specific sequence of characters. I'll assume you're looking to design a finite automaton that accepts strings following the pattern "aabaab" while ensuring that the characters are read in the correct order.

Here's how you can represent the state diagram and state table:

State Diagram:

```
+---a---+
|       |
V       |
--> q0 ---> q1 --> q2 --> q3 --> q4 --> q5
    |       |
    +---b----+
```

In this diagram:

- q0 is the initial state.
- q5 is the accepting (final) state, as it corresponds to the complete "aabaab" pattern.

State Table:

State	Input 'a'	Input 'b'
q0	q1	q0
q1	q2	q0
q2	q3	q0
q3	q3	q4
q4	q3	q5
q5	q5	q5

Here's how to interpret the state table:

- When in state q0 and reading an input 'a', transition to state q1.
- When in state q0 and reading an input 'b', stay in state q0.
- When in state q1 and reading an input 'a', transition to state q2.
- When in state q1 and reading an input 'b', stay in state q0.
- And so on for the other transitions.

Answer to the Question no- 3

(a)

DFA has been widely used in various applications such as:

- Language recognition
- Compiler Design
- Text Processing
- Network Protocols
- Lexical Analysis
- Cybersecurity
- Pattern matching
- Automated testing
- Formal verification

(b)

Union of Sets Definition-

The union of two sets X and Y is equal to the set of elements that are present in set X, in set Y, or in both the sets X and Y. This operation can be represented as;

$X \cup Y = \{a: a \in X \text{ or } a \in Y\}$

Let us consider an example, say; set A = {1, 3, 5} and set B = {1, 2, 4} then;

$A \cup B = \{1, 2, 3, 4, 5\}$

Proof of Union Theorem

If set A contains 13 elements, set B contains 8 elements and the intersection of these two sets contains 5 elements, then find the number of elements in A union B.

Solution:

Given,

Number of elements in set A = n(A) = 13

Number of elements in set B = n(B) = 8

Number of elements in A intersection B = n(A ∩ B) = 5

We know that,

n(A U B) = n(A) + n(B) – n(A ∩ B)

= 13 + 8 – 5

= 21 – 5

= 16

Therefore, the number of elements in A union B = n(A U B) = 16.

(c)

No, it's not true that every integer "a" can be written in the form "a = 2km" where "k" and "m" are integers. The equation you've mentioned represents an integer as a product of two integers: "2" and "k", where "m" is another integer. This equation implies that every integer is an even multiple of "k".

However, not all integers are even multiples of another integer. For example, odd integers cannot be represented in this form since they cannot be divided evenly by 2. Also, prime numbers are not expressible in this form because they only have two divisors: 1 and the prime number itself.

So, while the equation is true for even integers and some special cases, it doesn't hold for all integers.