1&2 questions ki answers notebook lo chusukondi Examples problem

1. conversion of NFA to DFA (problems)

2. conversion of ∈-NFA to without ∈-NFA (problems.)

3. a) Explain NFA & DFA

b) Explain procedure for conversion of NFA to DFA.

4. construct FA to $L = \{w \mid w \% 7 \equiv 0,\ w \in \{0,1,2\}\}$.

5. construct FA, which accepts even number of 0's over i/p alphabet $\Sigma = \{0,1\}$.

DFA (DETERMINISTIC FINITE AUTOMATA):

It can determine exactly what is the next state reading a particular input symbol from a particular state then that finite automata is called DFA.

* A DFA is a finite state machine where the pair of current state and current input symbol, there is a unique state.

REPRESENTATION OF DFA:

mathematically DFA Represented as 5 tuples like

$$\boxed{M = (Q, \Sigma, q_0, \delta, F)}$$

Q = set of states, Q is a finite and non-empty set of states

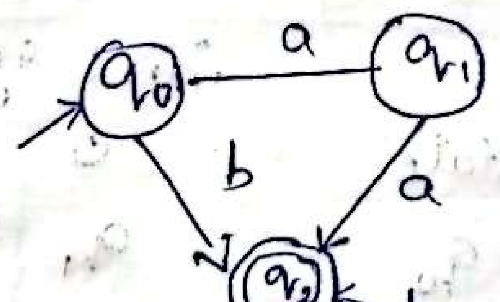
Σ = It is finite and non-empty set of input symbols.

δ = It is a transition function

q_0 = It is initial state

F = It is finite set of final states.

Example:



ACCEPTANCE OF A STRING BY DFA: Consider the DFA 'M' and the string 'w' over input symbol Σ. now the String 'w' is accepted by 'M' if and only if

$$\boxed{L(M) = \{ w \mid w \in \Sigma, \delta(q_0, w) = q_F \}}$$

* method for check whether the given String is accepted (or) not by DFA. There are three methods

1. using sequence Diagram
2. using extended transition function
3. using V dash function

DESIGN OF DFA:

PROCEDURE:

1. understanding the language which is for Designing a DFA.
2. Determine minimum length string in the language.
3. Draw the DFA for minimum length.
4. Determine Initial, Intermediate, Dead and final states for DFA
5. Apply each input symbol on every state of DFA.

Design a DFA for the language which consists of set of all strings of 0's over $\Sigma = \{0\}$

Let M be is a ~~string~~ DFA with 5 tuples like $M = \{Q, \Sigma, q_0, \delta, F\}$

Given input $\Sigma = \{0\}$

$$L(M) = \{\epsilon, 0, 00, 000, ---\}.$$

Language of particular machine

The minimum string is $= \{\epsilon\}$

The next min string is $= \{0\}$

∴ DFA = 

$Q = \{q_0\}$ $\quad$ $F = \{q_0\}$

$\delta =$

State	input symbol
	0
q_0	q_0

NON-DETERMINISTIC FINITE AUTOMATA: (NFA)

we cannot determine the next state exactly after reading an input symbol from a particular state then that finite automata is called NFA.

NFA is finite state machine whenever each pair of current state and particular input state symbol it has more than one next state.

The elements are:

1. State
2. Input symbol
3. Initial state
4. Final state
5. Transition.

REPRESENTATION OF NFA:

mathematically. NFA is 5 tuples like,

$$\boxed{M = \{Q, \Sigma, q_0, \delta, F\}}$$

where Q = finite and non-empty set of states

Σ = input alphabet (or) input symbols

δ = It is a transition function which is defined as $\boxed{Q \times \Sigma \Rightarrow Q^Q}$

q_0 = Initial state

F = final state

DESIGN OF NFA:

* understanding the language which is for Designing of NFA
* Determine minimum string ~~with~~ in the language
* Draw the NFA for minimum length string
* Determine initial, Intermediate, Dead and final States of NFA.
* apply the each input symbol on initial & final states of NFA.

Design NFA that accept all strings over $\Sigma = \{0,1\}$ that have atleast two consecutive 0's or 1's.

Let M be a NFA

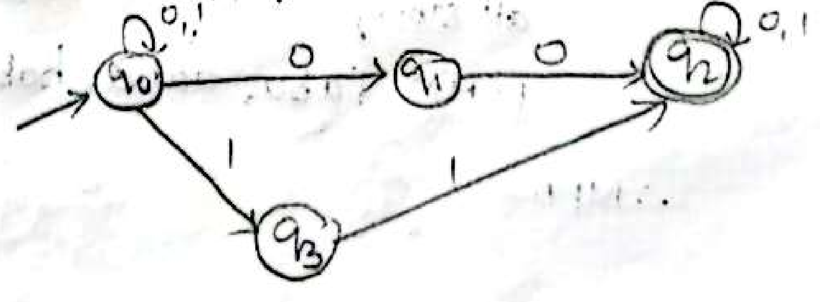
$$M = \{Q, \Sigma, q_0, \delta, F\}$$

$$\Sigma = \{0,1\}$$

each string has atleast two consecutive 0's or 1's. i.e, $(0+1)^* (11+00)(0+1)^*$

$$L(M) = \{00, 11, 100, 000, ---\}$$

$\therefore$ NFA =



$\delta =$

STATE	I/p SYMBOL 0	1
q_0	$\{q_0, q_1\}$	$\{q_3, q_0\}$
q_1	q_2	–
q_2	q_2	q_2
q_3	–	q_2

$Q = \{q_0, q_1, q_2, q_3\}$ $q_0 = \{q_0\}$ $F = \{q_2\}$

CONVERSION OF NFA TO DFA :

ALGORITHM :

Step 1: let D be a DFA

Step 2: let N be a NFA, This algorithm is called power set (or) sub set construction algorithm. because for NFA with 'n' states then the corresponding DFA can have 2^n states.

SUBSET CONSTRUCTION ALGORITHM :

STEP 1: construct the start state q_0, consisting of q_0 and all the states of NFA that can be reached from q_0 by one (or) more transitions. Mark q_0 as unfinished.

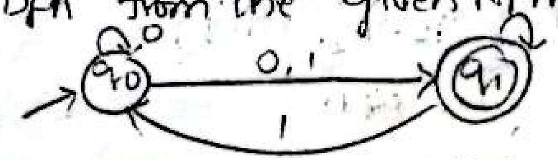
STEP 2: while there are unfinished states

1. Take an unfinished state 'S'
2. for each $A \in \Sigma$, $\delta(S), A = T$ either finished (or) unfinished states.
3. Mark S as finished

STEP 3: mark all states that contain a final state from N as final states of D.

Example :

construct DFA from the given NFA,



Let M be a NFA

$$M = \{Q, q_0, \Sigma, \delta, F\}$$

$$\Sigma = \{0, 1\}$$

$$q_0 = \{q_0\}$$

Transition table

$\delta =$

STATE	I/P SYMBOL 0	1
$\rightarrow q_0$	$\{q_0, q_1\}$	$\{q_1\}$
q_1	-	$\{q_0, q_1\}$

let us consider the DFA D is like

$$M = \{Q', \Sigma', q_0', \delta', F'\}$$

Apply subset construction algorithm.

Based upon Initial State we define the new state

Transition table:

STATE	I/P SYMBOL 0	1
$\rightarrow q_0$	$\{q_0, q_1\}$	$\{q_1\}$
$\{q_0, q_1\}$	$\{q_0, q_1\}$	$\{q_0, q_1\}$
$\{q_1\}$	ϕ	$\{q_0, q_1\}$
ϕ	ϕ	ϕ

$\delta' =$ the table above

$\delta(\{q_0, q_1\}, 0) =$

$\delta(q_0, 0) \cup \delta(q_1, 0)$

$= \{q_0, q_1\} \cup \{\phi\}$

$= \{q_0, q_1\}$

$\delta(\{q_0, q_1\}, 1) =$

$\delta(q_0, 1) \cup \delta(q_1, 1)$

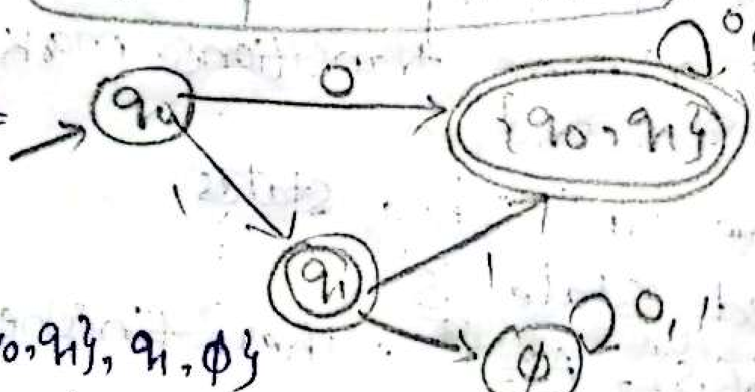
$= \{q_1\} \cup \{q_0, q_1\}$

$= \{q_0, q_1\}$

$\delta(q_1, 0) = \phi$

$\delta(q_1, 1) = \{q_0, q_1\}$

$\therefore$ DFA =



$Q' = \{q_0, \{q_0, q_1\}, q_1, \phi\}$

The Initial State in NFA will be the Initial State in DFA

$\therefore q_0' = \{q_0\}$

The final state of DFA will be the combination of final states in NFA

$\therefore F' = \{\{q_0, q_1\}, \{q_1\}\}$

Construct DFA from given NFA

② construct a DFA for the language $L = \{w / |w| = \text{mod } 7\}$ where string is created as ternary number (0,1,2)

$\Sigma_1 = \{0,1,2\}$

Given $\Sigma = \{0,1,2\}$

$L(M) = \{$ all strings divisible by 7 $\}$

∴ The possible remainders are 0,1,2,3,4,5,6

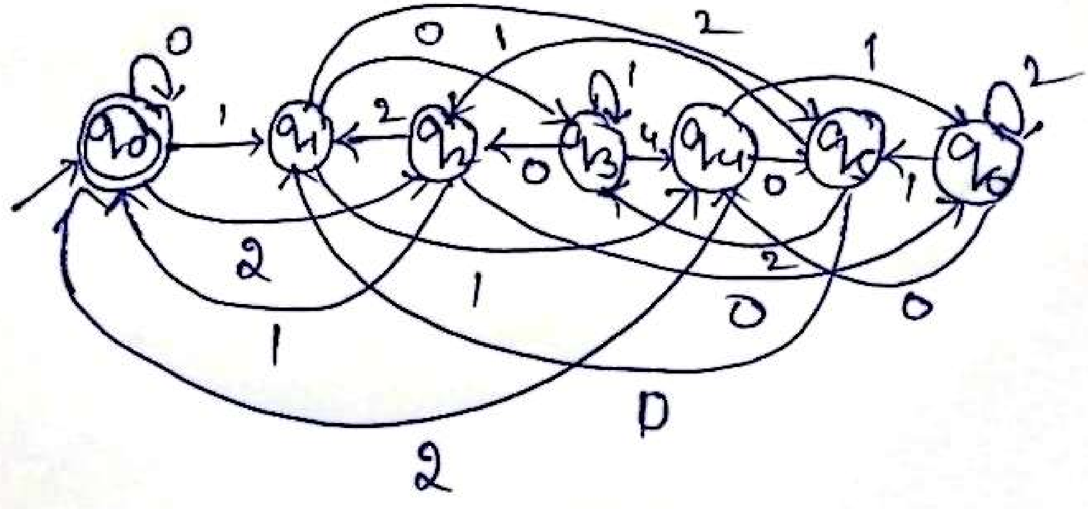
∴ The states are $q_0, q_1, q_2, q_3, q_4, q_5, q_6$

Initial states ~~are~~ is q_0

$\delta =$

STATE	Input symbol		
	0	1	2
$\rightarrow q_0$	q_0	q_1	q_2
q_1	q_3	q_4	q_5
q_2	q_6	q_0	q_1
q_3	q_2	q_3	q_4
q_4	q_5	q_6	q_0
q_5	q_1	q_2	q_3
q_6	q_4	q_5	q_6

∴ DFA =



$Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6\}$ $q_0 = \{q_0\}$ $F = \{q_0\}$

2. Construct FA, which accepts even number of 0's over i/p alphabet $\Sigma = \{0,1\}$

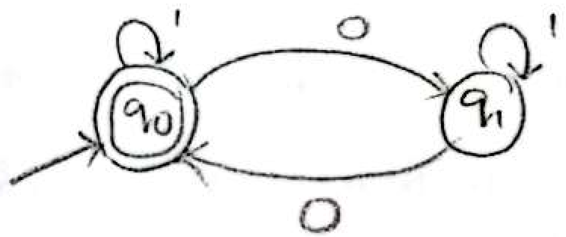
Let M be a FA

$M = \{Q, \Sigma, q_0, \delta, F\}$

Given $\Sigma = \{0,1\}$

$L(M) = \{00, 001, 0011, 1001, \ldots\ldots\ldots\}$

$\therefore$ FA =



5

$Q = \{q_0, q_1\}$

$q_0 = \{q_0\}$

$F = \{q_0\}$

δ =

STATE	INPUT SYMBOL	
	0	1
$\rightarrow q_0$	q_1	q_0
q_1	q_0	q_1