REGULAR LANGUAGES

20 / 03 / 2003

Computational Model → An idealised computer.

↳ Mathematical theories are set on these as it is not possible to do so directly on real computers.

Finite State Machine

We use several different computational model based on features we want.

Simplest Model → Finite state machine or Finite automaton.

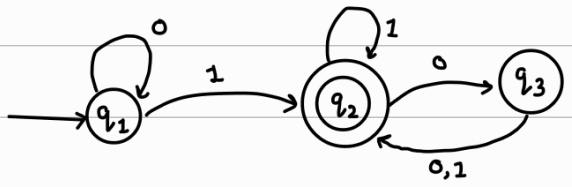
Finite Automata:

- good comp. models with extremely limited memory.
- The designing of devices with low memory such as thermostat, dish washers, digital watches requires the designer to keep methodology and terminology of finite automata in mind.

Markov Chains

- Finite Automata & their probabilistic counterpart 'Markov Chains' are useful tools when we're attempting to recognize patterns in data.
- These devices are used in speech processing & optical character recognization.

Mathematical Theory of Finite Automa :



Finite automaton M_1 with 3 states.

- State diagram
- States
- Start / accept state
- transitions (→)

- The above fig. is called state diagram for M_1.
- There are three states q_1, q_2, q_3
- q_1 → Start state
- q_2 → accept state. (indicated with double circle).

→ Arrows going from one state to other are transitions.

- When the above automaton receives a string input it produces an o/p which is either accept or reject.

determining o/p. → * After reading the last symbol in string, if M_1 is in the 'accept state' the o/p is accept else reject.

Finite Automaton:

Reasons for formal defn:

- Formal defn is precise
- It resolves any uncertainities about what's allowed in finite automaton.
- Good notation helps you think & express your thoughts clearly.

/ /

Parts of finite Automaton

1) Set of states and rules for going from one state to another, depending on i/p symbol.

2) It has an i/p alphabet indicating allowed input symbols.

3) It has a start state and a set of accept states.

Definition:

A finite-automaton is a 5-tuple $(Q, \Sigma, \delta, q_0, F)$, where

$Q \rightarrow$ finite set called the states

$\Sigma \rightarrow$ finite set called the alphabet.

$\delta \rightarrow$ $Q \times \Sigma \longrightarrow Q$ is the transition function

$q_0 \rightarrow$ $q_0 \in Q$ is the start state

$F \rightarrow$ $F \subseteq Q$ is the set of accept states.

- If 'A' is a set of strings that machine 'M' accepts

* Accept states are sometimes called finite states.