Module 2 Continued.

Pumping Lemma for Regular Languages.

Theorem:

Let $M = (Q, \Sigma, \delta, q_0, F)$ be a Finite Automata and has n number of states. Let L be the regular language accepted by M. Let for every string x in L, there exists a constant n such that $|x| \geq n$. Now, if the string x can be broken into three substrings u, v and w such that $x = uvw$

satisfying the following constraints:

(i) $v \neq \varepsilon$ ie $|v| \geq 1$

(ii) $|uv| \leq n$

then uv^iw in L for $i \geq 0$

Proof:

Let $M = (Q, \Sigma, \delta, q_0, F)$ be an Finite Automata and let L is the language accepted by DFA and is regular. Let $x = a_1 a_2 \ldots a_m$ where $m \geq n$ and each a_i is in Σ. Here, n represent the states of DFA.

Since there are m input symbols, we should have $m+1$ states in the sequence $q_0 q_1 \ldots q_m$

where q_0 will be the start state and q_m will be the final state as shown below:-

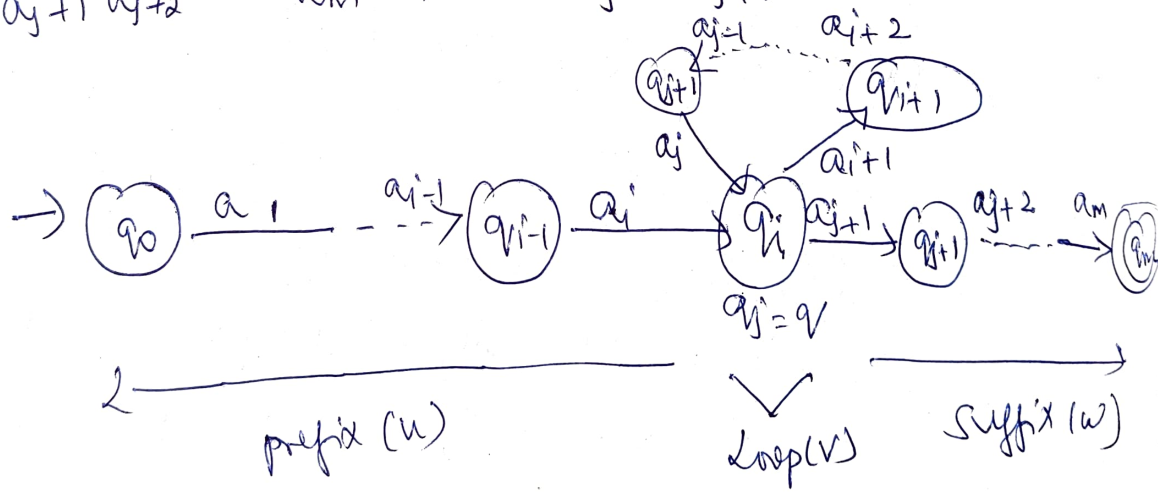
Based on Pigeon Hole principle,

let the string x is divided into three substrings as shown below:.

· The first group is the string prefix from $a_1 a_2 \cdots a_i$ i.e $u = a_1 a_2 \cdots a_i$

The second group is the loop string from $a_{i+1} a_{i+2} \cdots a_{j-1} a_j$ i.e $v = a_{i+1} a_{i+2} \cdots a_{j-1} a_j$

The third group is the string suffix from $a_{j+1} a_{j+2} \cdots a_m$ i.e $w = a_{j+1} a_{j+2} \cdots a_m$



From the above figure, the prefix string u takes the machine from q_0 to q_i, the loop string v takes the machine from q_i to q_i and

suffix string w takes the machine from q_j to q_f.
The minimum string that can be accepted by the above finite automata is uw with $i=0$.

when $i=1$, the string uvw is accepted
when $i=2$, the string $uvvw$ is accepted.

so if $i \geq 0$, the machine goes from q_0 to q_i on input u, circles from q_i to q_i based on values of i and goes to accepting state on input w.

In general, the string x is split into substring uvw, then for all $i \geq 0$,

$$uv^iw \in L$$

This can be shown as:-

$$\delta(q_0, a_1 a_2 \cdots a_{i-1} a_i a_{j+1} a_{j+2} \cdots a_m)$$

$$= \delta(\delta(q_0, a_1 a_2 \cdots a_{i-1} a_i), a_{j+1} a_{j+2} \cdots a_m)$$

$$= \delta(q_r, a_{j+1} a_{j+2} \cdots a_m)$$

$$= \delta(q_k, a_{k+1} a_{k+2} \cdots a_m)$$

$$= (q_m) = q_m$$

Hence, the machine enters the final state.