Theory of Automata and Formal Language

Lecture-20

Dharmendra Kumar
(Associate Professor)
Department of Computer Science and Engineering
United College of Engineering and Research, Prayagraj

May 4, 2021

Kleene's Theorem

Statement: Let r be a regular expression. Then there exists some non-deterministic finite automata that accepts L(r). Consequently, L(r) is a regular language.

Proof:

We will prove this theorem by using induction method.

Apply the induction method on the no. of operators used in the regular expression.

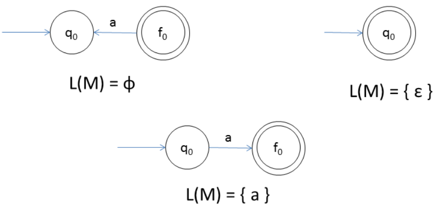
Basis step: In this step, we will prove for the regular expressions which does not contain any operators.

In this case, regular expressions are

ϕ, ϵ and a, where a$\in \Sigma$.

Regular Expression

The finite automata corresponding to the above regular expressions are:-



Clearly, there exists a finite automata for every regular expressions which not contain any operator. Therefore, theorem is true for this case.

Induction step:

Suppose the theorem is true for the regular expressions which contain n operators. We will prove the theorem for the regular expressions that contain n+1 operators. The $(n+1)^{th}$ operator in the regular expressions may be one of the following:-

(1) + (Union operator) (2) . (Concatenation operator) (3) * (Kleene closure operator)

Case 1 Union operator($r_1 + r_2$):

Suppose the $(n+1)^{th}$ operator is $+$.

Consider the regular expressions r_1 and r_2 of n operators.

Since the theorem is true for r_1 and r_2, therefore there exists finite automata M_1 and M_2 corresponding to r_1 and r_2. Let these finite automatas are

$M_1 = (Q_1, \Sigma_1, \delta_1, q_1, \{f_1\})$

$M_2 = (Q_2, \Sigma_2, \delta_2, q_2, \{f_2\})$

Now, we construct the finite automata corresponding to $r_1 + r_2$. This is constructed as following:-

$M = (Q, \Sigma, \delta, q_0, \{f_0\})$

Where, Q $= Q_1 \cup Q_2 \cup \{q_0, f_0\}$

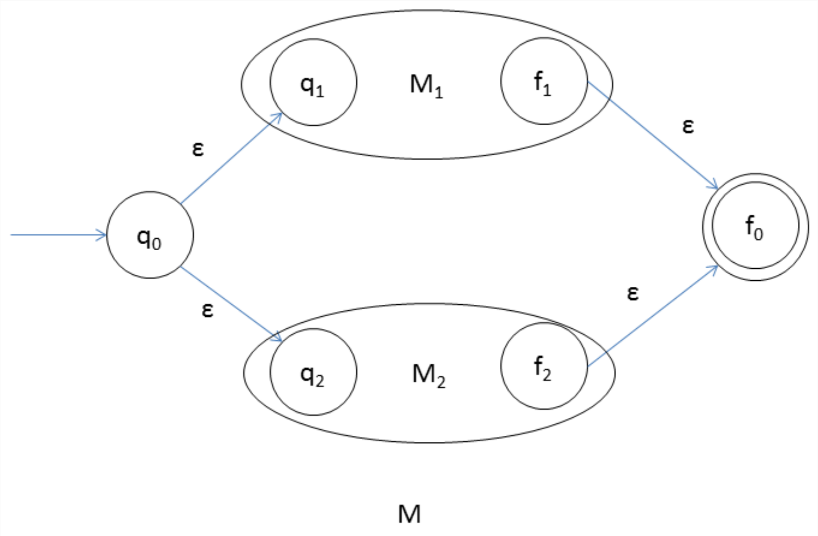
$\Sigma = \Sigma_1 \cup \Sigma_2$

$$\delta(q,a) = \begin{cases} \delta_1(q,a), & if q \in Q_1 \ and \ a \in \Sigma_1 \\ \delta_2(q,a), & if q \in Q_2 \ and \ a \in \Sigma_2 \end{cases} \tag{1}$$

$\delta(q_0, \epsilon) = \{q_1, q_2\}$

$\delta(f_1, \epsilon) = \{f_0\} = \delta(f_2, \epsilon)$

Regular Expression



Now,we have to show that

L(M) $= L(M_1) \cup L(M_2)$ ....(1)

Let x$\in$ L(M) $\Leftrightarrow$ $\epsilon x \epsilon \in L(M)$

$\Leftrightarrow$ x$\in$ L(M_1) or x$\in$ L(M_2)

$\Leftrightarrow$ x$\in$ $L(M_1) \cup \in L(M_2)$

Therefore, L(M) $= L(M_1) \cup L(M_2)$

Therefore M is a finite automata which accepts the set $L(r_1) \cup L(r_2)$.

Therefore M is a finite automata for the regular expression $r_1 + r_2$.

Therefore theorem is true for this case.

Case 2 Concatenation operator($r_1.r_2$):

Suppose the $(n+1)^{th}$ operator is . .

Consider the regular expressions r_1 and r_2 of n operators.

Since the theorem is true for r_1 and r_2, therefore there exists finite automata M_1 and M_2 corresponding to r_1 and r_2. Let these finite automatas are

$M_1 = (Q_1, \Sigma_1, \delta_1, q_1, \{f_1\})$

$M_2 = (Q_2, \Sigma_2, \delta_2, q_2, \{f_2\})$

Now, we construct the finite automata corresponding to $r_1.r_2$. This is constructed as following:-

$M = (Q, \Sigma, \delta, q_1, \{f_2\})$

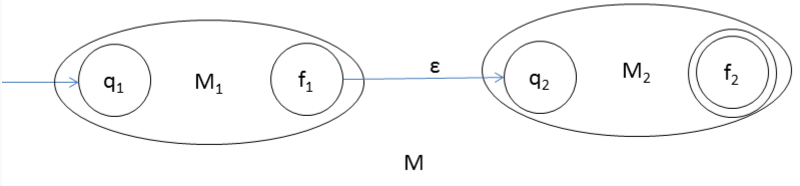
Where, Q $= Q_1 \cup Q_2$

$\Sigma = \Sigma_1 \cup \Sigma_2$

$$\delta(q,a) = \begin{cases} \delta_1(q,a), & ifq \in Q_1 \; and \; a \in \Sigma_1 \\ \delta_2(q,a), & ifq \in Q_2 \; and \; a \in \Sigma_2 \end{cases} \tag{2}$$

$\delta(f_1, \epsilon) = \{q_2\}$

Regular Expression



Regular Expression

Now,we have to show that

L(M) = $L(M_1).L(M_2)$ ....(2)

Let x$\in$ L(M) $\Leftrightarrow$ $x_1 \epsilon x_2 \in L(M)$ (Let x=$x_1.x_2$)

$\Leftrightarrow$ $x_1 \in$ L(M_1) and $x_2 \in$ L(M_2)

$\Leftrightarrow$ $x_1.x_2 \in L(M_1).L(M_2)$

$\Leftrightarrow$ x $\in L(M_1).L(M_2)$

Therefore, L(M) = $L(M_1).L(M_2)$

Therefore M is a finite automata which accepts the set $L(r_1).L(r_2)$.

Therefore M is a finite automata for the regular expression $r_1.r_2$.

Therefore theorem is true for this case.

Case 3 Kleene closure operator(r^*):

Suppose M is a finite automata corresponding to regular epression r.

$$M = (Q, \Sigma, \delta, q_0, \{f_0\})$$

Now, we construct finite automata corresponding to r^*. This is constructed as following:-

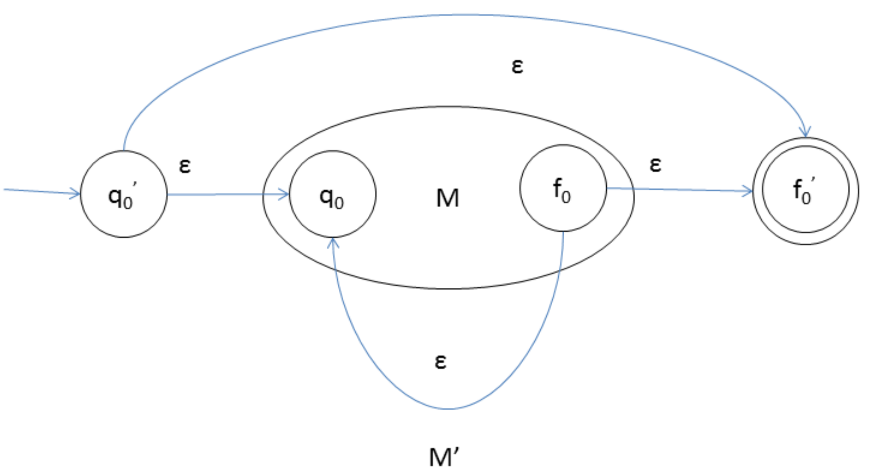
$M' = (Q', \Sigma, \delta', q_0', \{f_0'\})$

Where, Q $= Q \cup \{q_0', f_0'\}$

$$\delta'(q, a) = \begin{cases} \delta(q, a), & if q \in Q \ and \ a \in \Sigma \end{cases} \tag{3}$$

$\delta'(q_0', \epsilon) = \{q_0, f_0'\} = \delta'(f_0, \epsilon)$

Regular Expression



Regular Expression

Now,we have to show that

L$(M') = (L(M))^*$ ....(3)

Let x$\in$ L$(M') \Leftrightarrow \ \epsilon x \epsilon \in L(M')$ (Let x=$x_1.x_2$)

$\Leftrightarrow \ \epsilon x_1 \epsilon x_2 \epsilon.........\epsilon x_n \epsilon \in L(M')$ (Let x=$x_1.x_2........x_n$)

$\Leftrightarrow \ x_i \in L(M), \ \ \forall i = 1, 2, ..., n$

$\Leftrightarrow \ x_1.x_2........x_n \in (L(M))^n$

$\Leftrightarrow \ x \in (L(M))^n$

$\Leftrightarrow \ x \in (L(M))^*$ (since $(L(M))^n) \subseteq (L(M))^*$

Therefore, L(M) = $(L(M))^*$

Therefore M is a finite automata which accepts the set $L(r_1).L(r_2)$.

Therefore M is a finite automata for the regular expression $r_1.r_2$.

Therefore theorem is true for this case.

Since in all the three cases, theorem is true, therefore theorem is true for induction step also.

Therefore, for any regular expression, there exists a finite automata.

Now the theorem is proved.