Construct a Regular Expression from Finite Automata

- Example
- R_{12} is a regular expression from s_1 to s_2.
- We construct table to find the regular expression.
- As we need multiple columns in a table, we use a superscript in naming regular expression. For example R_{12}^{1}, R_{64}^{2},... R_{ij}^{k}, R_{ij}^{k-1} are just names of different regular expressions.
- The following tables gives the regular expression for different values of k, where k denotes the maximum number of states from state i to state j.

Construct a Regular Expression from Finite Automata

Column n is built from starting state to all possible final states.

	K=0	K=1	K=2	.....	K=n
R_{11}^{k}	R_{11}^{0}	R_{11}^{1}	R_{11}^{2}	.....	R_{11}^{n}
R_{12}^{k}	R_{12}^{0}	R_{12}^{1}	R_{12}^{2}	.....	R_{12}^{n}
R_{21}^{k}	R_{21}^{0}	R_{21}^{1}	R_{21}^{2}	.....	R_{21}^{n}
R_{22}^{k}	R_{22}^{0}	R_{22}^{1}	R_{22}^{2}	.....	R_{22}^{n}

Construct a Regular Expression from Finite Automata

- Finally the regular expression is union of all the expressions in column n
- In the above table there are n rows and n+1 columns
- All pairs of numbers from state1 to state n will constitute n rows
- Now, build the table entries for k=0 columns
- $$R_{ij} = \begin{cases} \{a \mid \delta(q_1, a) = q_j\}, & i \neq j \\ \{a \mid \delta(q_1, a) = q_j\} + \varepsilon, & i = j \end{cases}$$
- $R_{i,j}^{k}$ could be Φ, ε, a, 0+1 or a+ ε
- Epsilon(ε) – if there is transition over the same state
- Phi(Φ) – If there is no transition between two different states

Minimization rules for regular expression

Minimization rules for regular expression

Rule No.	Expression	Rule No.	Expression
1	$\Phi+R=R$	11	$(P+Q)^* = (P^*Q^*)^* = (P^*+Q^*)^*$
2	$\Phi R = R\,\Phi = R$	12	$(P+Q)R = PR + QR,\ R(P+Q) = RP + RQ$
3	$\varepsilon\, R = R\,\varepsilon = R$	13	$R^*+ \varepsilon = R^*$
4	$\varepsilon^* = \varepsilon$ and $\Phi^* = \varepsilon$	14	$(R + \varepsilon)^* = R^*$
5	$R + R = R$	15	$R^*(a+b) + (a+b) = R^*\,(a+b)$
6	$R^*R^* = R^*$	16	$R^*R+R = R^*R$
7	$RR^* = R^*R$	17	$(R+ \varepsilon)R^* = R^*(R+ \varepsilon) = R^*$
8	$(R^*)^* = R^*$	18	$(R+ \varepsilon)(R+ \varepsilon)^*(R+ \varepsilon) = R^*$
9	$\varepsilon + RR^* = R^* = \varepsilon + R^*R$	19	$\Phi+ \varepsilon = \varepsilon$
10	$(PQ)^*P = P(QP)^*$		

Construct a Regular Expression from Finite Automata

- Note that there are no Kleene star or concatenation in this column. To build successive column corresponding to k=1, k=1,2,..n the recursive formula is used.

$$R_{ij}^{k} = R_{ij}^{k-1} + R_{ik}^{k-1} \left(R_{kk}^{k-1} \right) R_{kj}^{k-1}$$

OR

$$R_{ij}^{k} = R_{ij}^{k-1} \cup R_{ik}^{k-1} \left(R_{kk}^{k-1} \right) R_{kj}^{k-1}$$

- Finally the regular expression is $R_{ip}^{n} + R_{iq}^{n} + R_{1r}^{n}$ where p,q,r are final states and 1 is the initial state where n represents number of states in FA.