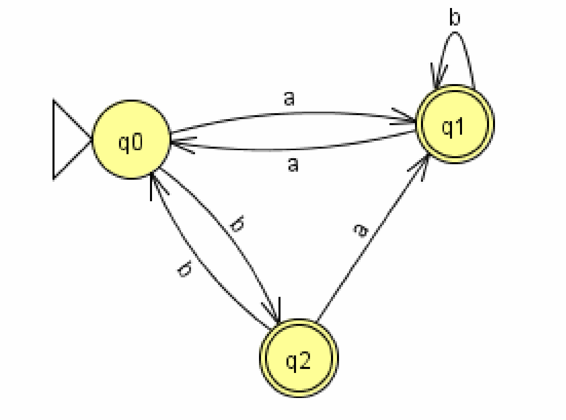
CSE355 Class Notes - 2/14/2008

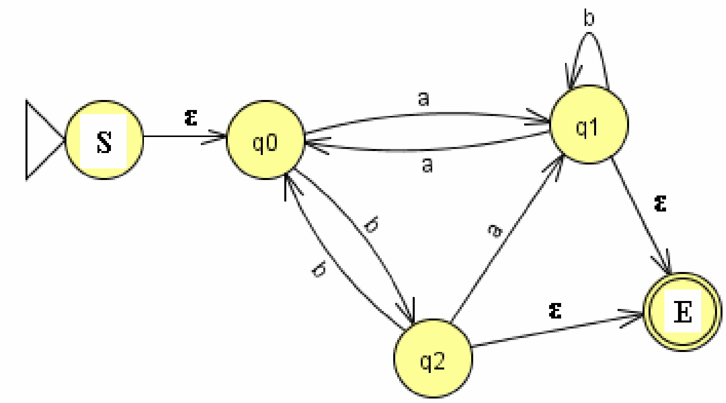
Converting DFA to Regular Expression:

Example: Write the Regular Expression for the following DFA:



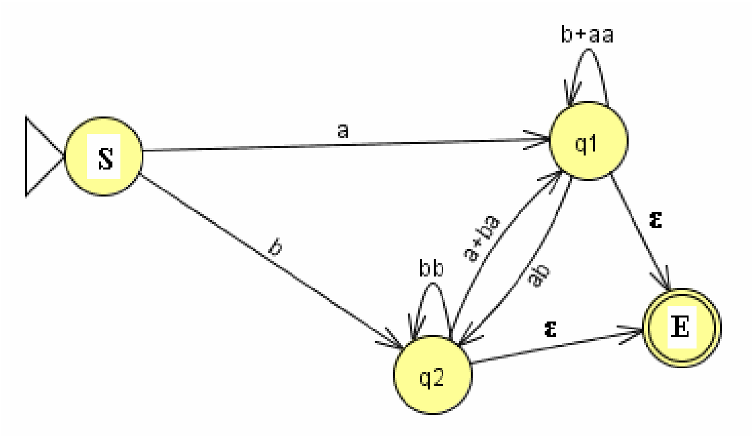
Solution:

Step 1: Add a new initial state (S) and a common final state (E) using ε-transition:



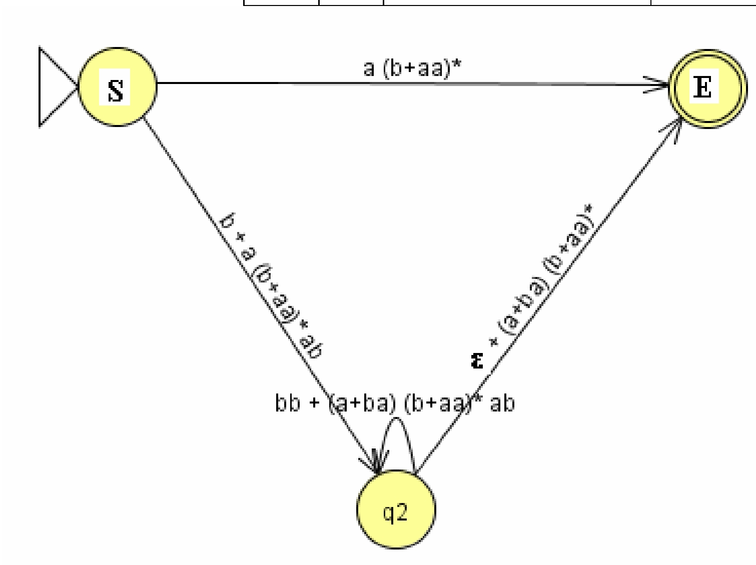
Step 2: Remove state q0:

	S	q1	q2
S	--	a	b
q1	--	aa	ab
q2	--	ba	bb



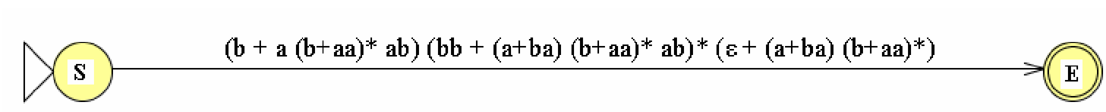
Step 3: Remove state q1:

	S	q2	E
S	--	a (b+aa)* ab	a (b+aa)*
q2	--	(a+ba) (b+aa)* ab	(a+ba) (b+aa)*
E	--	--	--



Step 4: Remove state q2:

	S	E
S	--	(b + a (b+aa)* ab) (bb + (a+ba) (b+aa)* ab)* (ε + (a+ba) (b+aa)*)
E	--	--

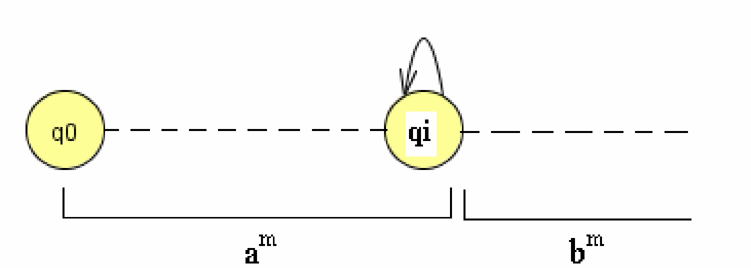


Algebraic Laws for Regular Expressions:

$L + M = M + L$
$(L + M) + N = L + (M + N)$
$L(MN) = (LM)N$
$\varepsilon L = L = L\varepsilon$
$\Phi L = \Phi = L\Phi$
$L(M + N) = LM + LN$
$(M + N)L = ML + NL$
$(L^*)^* = L^*$
$\varepsilon^* = \varepsilon$
$\Phi^* = \varepsilon$

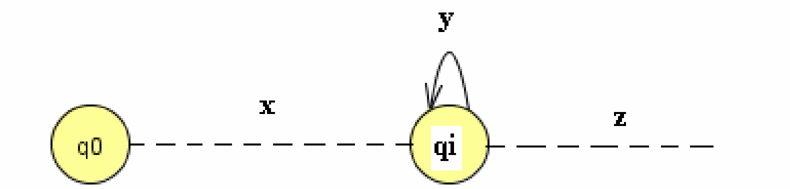
Show that $L = \{a^n b^n, n \geq 0\}$ is not a regular language

Proof:
Suppose to the contrary and it is regular
Then there must be a DFA for L
Suppose that the DFA has m number of states
This DFA must accept the string $a^m b^m$
While processing a^m through the DFA it will go through some state at least twice
It will look something like:



Now by looping more number of times at qi, we can have a string $a^p b^m$ where $p > m$ that is accepted by the DFA
This contradicts our assumption that the DFA was for language L.
Hence, out initial assumption that L is regular language must be wrong.

Pumping Lemma:



For any regular language L, there is an integer m such that all strings in L whose length is $\geq$ m can be written as xyz such that:

(i) $xy^ix \in L$
(ii) $|xy| \leq m$
(iii) $y \neq \varepsilon$