Lem. If a language is regular, then it is described by a regular expression.

**IDEA:** (1) DFA $\rightarrow$ GNFA

* (2) GNFA $\rightarrow$ regular expression

**Sketch of proof (step 2):**

"By construction".

Let $M$ be the DFA for language $A$; we first convert $M$ to a GNFA $G$.

Then, run $\text{Convert}(G)$:

1. Let $k$ be the # of states in $G$
2. If $k=2$, return the expression $R$
3. If $k>2$, select any state $q_{\text{rip}} \in Q$ different from $q_{\text{start}}$ and $q_{\text{accept}}$.
   Delete $q_{\text{rip}}$ as in Fig 1 to obtain a new GNFA $G'$ (with $k-1$ states).
4. Recursively call $\text{Convert}(G')$. 

**Fig 1**