**Theorem** A language is regular iff an NFA accepts it.

**Proof** We need to show two things.

1. \( \forall A \) regular \( \exists \) NFA \( N \) such that \( L(N) = A \)
2. \( \forall \) NFA \( N \), \( L(N) \) is regular.

For (1), \( A \) regular \( \Rightarrow \exists \) DFA \( D \) such that \( L(D) = A \).

But all DFAs are NFAs, so \( D \) is an NFA with \( L(D) = A \), and we’re done.

For (2), let \( N = (Q, \Sigma, S, q_0, F) \) be an NFA. It suffices to construct a DFA \( D \) such that \( L(D) = L(N) \) (i.e. \( D \) simulates \( N \)).

Let \( D = (Q', \Sigma, S', q'_0, F') \) be a DFA such that

- \( Q' = 2^Q \)
- \( q'_0 = \pi(q_0) \)
- \( F' = \{ q' \in Q' \mid \exists q \in F : q \subseteq q' \} \)
- \( \delta'(q', a) = R(\pi(q), S(p, a)) \)

where \( R(S) \) is the set of all states reachable from states in \( S \) via no input.

Briefly we explicitly show \( L(D) = L(N) \) via a lemma.

**Definition** We define iterated transition function \( \delta^* : Q \times \Sigma^* \rightarrow Q \) to be for \( \omega \in \Sigma^* \) and \( a \in \Sigma \)

\[ \delta^*(q, \epsilon) = q \]

\[ \delta^*(q, aw) = \delta(\delta^*(q, w), a) \]

Similarly for \( \delta^* : \pi(Q) \times \Sigma^* \rightarrow \pi(Q) \)

\[ \delta^*(q, \epsilon) = R(q) \]

\[ \delta^*(q, aw) = R(\bigcup_{p \in \delta^*(q, w)} \delta(p, a)) \]

where \( R \) is as defined above.
Lemma  For any input $w \in \Sigma^*$, $S^*(q_0, w) = S^*(q_0', w)$.

First notice that if this lemma is true, then the set of all possible states $N$ could be in after processing $w$ is exactly the single state $D$ is in. This would imply $\forall q \in F, q \in S^*(q_0, w)$ iff $q \in S^*(q_0', w)$, hence $N$ accepts $w$ iff $D$ accepts $w$, thus $L(N) = L(D)$.

It only remains to prove the lemma.

$\text{Proof}$  This is clearly true by inspection/construction/definition.

Cor  For every NFA $N$, there is a DFA $D$ such that $L(N) = L(D)$.

As an aside, notice that we can remove the non from nondeterministic via an exponential increase in memory. This will be important later.