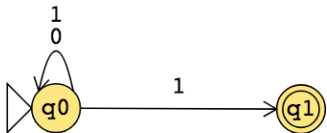


Week2 friday

Nondeterministic finite automaton $M = (Q, \Sigma, \delta, q_0, F)$	
Finite set of states Q	Can be labelled by any collection of distinct names. Default: q_0, q_1, \dots
Alphabet Σ	Each input to the automaton is a string over Σ .
Arrow labels Σ_ϵ	$\Sigma_\epsilon = \Sigma \cup \{\epsilon\}$.
Transition function δ	Arrows in the state diagram are labelled either by symbols from Σ or by ϵ . $\delta : Q \times \Sigma_\epsilon \rightarrow \mathcal{P}(Q)$ gives the set of possible next states for a transition from the current state upon reading a symbol or spontaneously moving.
Start state q_0	Element of Q . Each computation of the machine starts at the start state.
Accept (final) states F	$F \subseteq Q$.
M accepts the input string	if and only if there is a computation of M on the input string that processes the whole string and ends in an accept state.
Page 53	

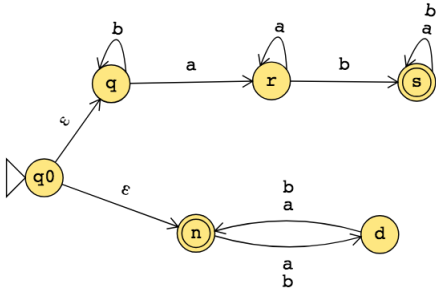
The formal definition of the NFA over $\{0, 1\}$ given by this state diagram is:



The language over $\{0, 1\}$ recognized by this NFA is:

Change the transition function to get a different NFA which accepts the empty string.

The state diagram of an NFA over $\{a, b\}$ is below. The formal definition of this NFA is:



The language recognized by this NFA is: