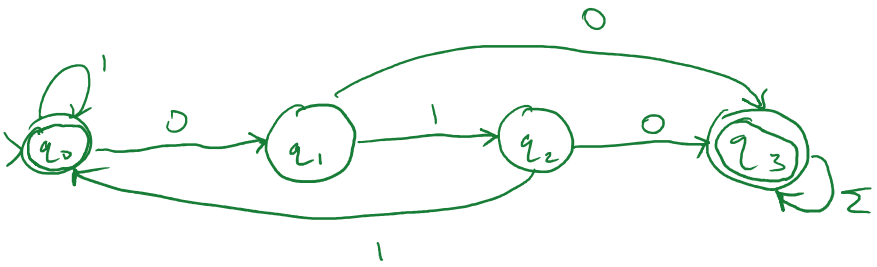
Non-Deterministic Finite Automata

Sunday, October 25, 2020 1:22 PM

Tabular DFA's

Can represent DFA's as table!

ex.



	0	1
* q_0	q_1	q_0
q_1	q_3	q_2
q_2	q_3	q_0
* q_3	q_3	q_3

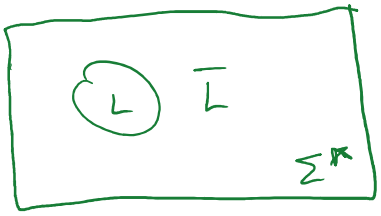
Regular Languages

Defn. A language L is called a regular language if there exists a DFA D where $\mathcal{L}(D) = L$.

If L is a language and $\mathcal{L}(D) = L$, we say that D recognizes L.

Language Complement. Given a language $L \subseteq \Sigma^*$, the complement $\bar{L}$ is the language of all strings in Σ^* but not in L.

i.e., $\bar{L} = \Sigma^* - L$



If L is a regular language, then $\bar{L}$ is also a regular language.

NFAs

Determinism vs Nondeterminism

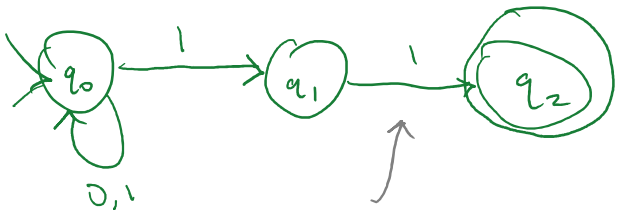
A model of computation is <u>deterministic</u> if at every point in the computation, there is exactly one choice it can make.

(accepts if set of choices leads to accepting state)

<u>Nondeterministic</u> if at one point there are zero or multiple decisions to choose from.

(accepts if <u>any</u> series of choices leads to accepting state)

Ex.



if no appropriate transition exists, the automaton <u>dies</u> and the path does not accept.

Language. The language of an NFA is

$$\mathcal{L}(N) = \{w \in \Sigma^* \mid N \text{ accepts } w\}$$

Special type of transition

ε-transitions NFA can follow any # of Σ-transitions at any time without consuming input

NFA's are not required to take ε-transitions