Lecture 3:
- Finish regex
- DFA
  - formal def'n
  - examples

\[(\varepsilon + 0) (1 + 10)^* \quad \varepsilon = \text{no 2 consecutive 0's} \quad 000 \]

Language (English) \rightarrow \text{reg ex}

1. strings with 001 or 100 as substring
   \[(0 + 1)^* (001 + 100) (0 + 1)^* \]

2. strings with an even # of 1's
   \[(0 + 11)^* \quad \varepsilon \]
   \[(0 + 10^* 1)^* \quad 0000 \checkmark \]
   \[(0^* 10^* 10^*)^* \quad 0110 \]

3. strings that do not contain 011 as substring
   \[1^* (0 + (1 + \varepsilon))^* \quad \frac{011}{01} \]

Deterministic Finite Automata
aka DFA's
Finite State Machines (FSMs)
Recognizing languages

Given $L$ and $w$

$w \in L$?

$f_L (w) = \begin{cases} 0 & \text{if } w \notin L \\ 1 & \text{if } w \in L \end{cases}$

Transitions: one per character

One pass over input string
A DFA accepts \( w \) if (unique) walk defined by \( w \) ends in an accepting state.

For the given DFA:

- \( M \) does not accept 10110.
- \( M \) accepts 01.
- \( M \) accepts 01.
TCP State Machine