Review of Deterministic Finite Automata (DFA)
Example DFA

Binary numbers divisible by 3

Reversing a regular set

Reverse(L) = \{w | Reverse(W) \in L\}

Idea:
- Old final states are all start states (more than one start state?)
- Old start state is new final state
- Arrows go backward (more than one arrow with same symbol leaving state?)

Example
- Reverse of: every 0 is immediately followed by at least 2 ones
  - Which means: every 0 is immediately preceded by at least 2 ones
Non-deterministic Finite Automata (NFA)

There can be many paths through the NFA
- If any one path ends in a final state (after consuming all the input), accept the string
- Differences from Deterministic Finite Automata (DFA):
  - Can have moves on $\epsilon$
  - Can have multiple edges with the same symbol leaving a state
  - Need not have edges for all symbols from a state (if can't leave a state, that path fails)

Example
- Binary string contains 101101

Example

Binary strings divisible by 4
Union using NFA

Binary strings ending in 00 or with an even number of 1’s

Union using DFA

Binary strings ending in 00 or with an even number of 1’s
Regular sets

A set is regular if some Finite State Automata (FSA) exists that recognizes it.
- Regular sets are closed under:
  - Union
  - Intersection
  - Complement
  - Concatenation

Converting an NFA to an equivalent DFA

Idea: in the DFA, keep track of the set of states that the NFA could be in.
More formal description

Given an NFA, $N$, create a DFA whose:
- states are the powerset of the NFA
- start state is the set of all states reachable from $N$'s start state
- final states are those states containing $N$'s final states
- arrows are of form: from a state in DFA, on a symbol go to all states that can be reached from that symbol in set of states in NFA (make sure to follow $\epsilon$-arrows)

- If desired, remove all unreachable states
  - (Or, create the states on an as-needed basis)

Concatenating two regular sets

A string from the first set followed by a string from the second set
- Idea: guess where the first string ends

- Example: $L$ = binary strings divisible by 4, $M$ = every 0 immediately followed by 2 1s