from NFAs to DFAs
converting nfas to dfas

• Can convert NFAs to deterministic finite automata (DFAs)
  • No choices — never a need to “split” pointers
• Initial idea: simulate NFA for all possible inputs, any time there is a new configuration of pointers, create a state to capture it
  • Pointers at states 1, 3 and 4 → new state \{1, 3, 4\}
• Trying all possible inputs is impractical; instead, for any new state, explore all possible next states (that can be reached with a single character)
  • Process ends when there are no new states found
• This is an example of a fixed-point algorithm (we’ll see many more of these in the future)
• Convert the following into a DFA