DEFINITIONS

- A Deterministic Finite State Automaton (DFA), also called a Finite State Automaton (FSA) *A* consists of 5 objects
  - A set *I* called the input alphabet, of input symbols
  - A set *S* of states the automaton can be in;
  - A designated state s_0 called the initial state;
  - A designated set of states $F \subseteq S$ called the set of accepting states, or final states;
  - A next-state function $N: S \times I \rightarrow S$ that associates a "next-state" to each ordered pair consisting of a "current state" and "current input". For each state s in *S* and input symbol *m* in *I*, $N(s,m)$ is called the state to which *A* goes if *m* is input to *A* when *A* is in state *s*.
- The operation of an FSA is commonly described by a diagram called a (state-)transition diagram. In a transition diagram, states are represented by circles, and accepting states by double circles. There is one arrow that points to the initial state and other arrows between states as follows: There is an arrow from state *s* to state *t* labeled *m* ($\in$I) iff $N(s,m)=t$.
- The next-state table is a tabular representation of the next-state function. In the annotated next-state table, the initial state is indicated by an arrow and the accepting states by double circles.
- The eventual-state function of *A* is the function $N^*: S \times I^* \rightarrow S$ defined as: for any state *s* of *S* and any input string *w* in I^*, $N^*(s,w)$ = the state to which *A* goes if the symbols of *w* are input into *A* in sequence starting when *A* is in state *s*.

AUTOMATA AND REGULAR LANGUAGES

- Let *A* be a FSA with set of input symbols *I*. Let *w* be a string of I^*. Then *w* is accepted by *A* iff $N^*(s_0,w)$ is an accepting state.
- The language accepted by *A*, denoted $L(A)$, is the set of all strings that are accepted by *A*. $L(A) = \{w \in I^* \mid N^*(s_0,w) \text{ is an accepting state of } A\}$
- Kleene's Theorem: A language is accepted by an FSA iff it can be described by a regular expression. Such a language is called a regular language.
- Theorem 1: Some languages are not regular.
- Theorem 2: The set of regular languages over an alphabet *I* is closed under the complement, union, intersection and concatenation operators. (Union and concatenation already handled by definition).

NON-DETERMINISM

- For any set S, P(S), the power set of S, is the set of all possible subsets of S
- A Non-Deterministic Finite State Automaton (NFA) *A* consists of 5 objects
  - A set *I* called the input alphabet, of input symbols
  - A set *S* of states the automaton can be in;
  - A designated state s_0 called the initial state;
  - A designated set of states $F \subseteq S$ called the set of accepting states, or final states;
  - A next-state function *N*: $S \times (I \cup \{\varepsilon\}) \rightarrow P(S)$ that associates a subset of S to each ordered pair consisting of a "current state" and "current input". For each state s in S and input symbol *m* in $I \cup \{\varepsilon\}$, *N(s,m)* is called the set of possible next states to which *A* goes if *m* is input to *A* when *A* is in state *s*.
- The eventual-state function of *A* is the function $N^*: S \times I^* \rightarrow P(S)$ defined as: for any state *s* of *S* and any input string *w* in I^*, $N^*(s,w)$ = the set of states to which *A* can go if the symbols of *w* are input into *A* in sequence starting when *A* is in state *s*.
- Let *A* be a NFA with set of input symbols *I*. Let *w* be a string of I^*. Then *w* is accepted by *A* iff there is one possible state in $N^*(s_0,w)$ which is an accepting state.
- The language accepted by *A*, denoted *L(A)*, is the set of all strings that are accepted by *A*. $L(A) = \{w \in I^* \mid N^*(s_0,w) \text{ contains an accepting state of } A\}$
- NFAs can contain spontaneous (or epsilon) transitions: these are transitions between two states that occur when reading the null string ε. In other words the change in state can occur without reading any symbols.

EQUIVALENT AUTOMATA

- Let A and A' be automata (deterministic or not) with the same input alphabet I. A is said to be equivalent to A' iff L(A) = L(A')
- Theorem:
  - Every DFA is equivalent to some NFA
  - Every NFA is equivalent to some DFA.
- Corollary: a language is a regular language iff it is accepted by an NFA.