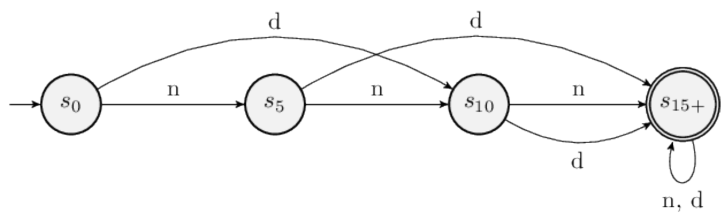
CSCI 301 - Lecture 23:
DFAs and Regular Languages

What strings put this machine in an accept state?



nnd
dd
nd
nnn
dn
nnnn

Definition: Let $M = (Q, \Sigma, \delta, q, F)$ be a finite automaton and let $w = w_1 w_2 w_3 \ldots w_n$ be a string over Σ. Define a sequence of states $r_0, r_1, \ldots r_n$ as follows:

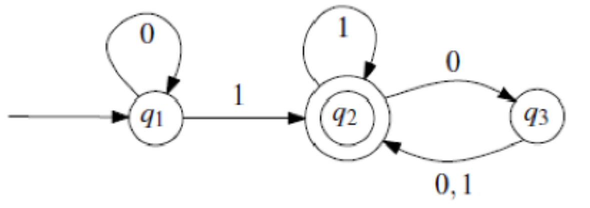
- $r_0 = q$ (the start state)
- $r_{i+1} = \delta(r_i, w_i + 1)$ for $i = 0, 1, \ldots, n-1$

If $r_n \in F$, then M **accepts** w.

If $r_n \notin F$, then M **rejects** (or does not accept) w.

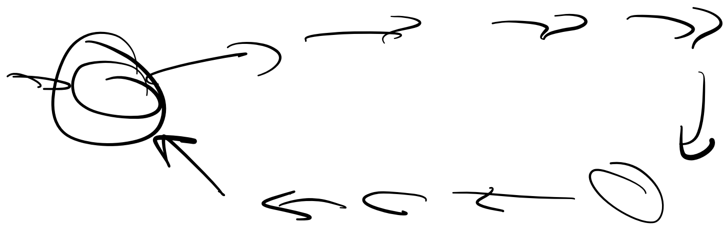
The **language accepted** by a machine M is the set of all strings accepted by the machine:

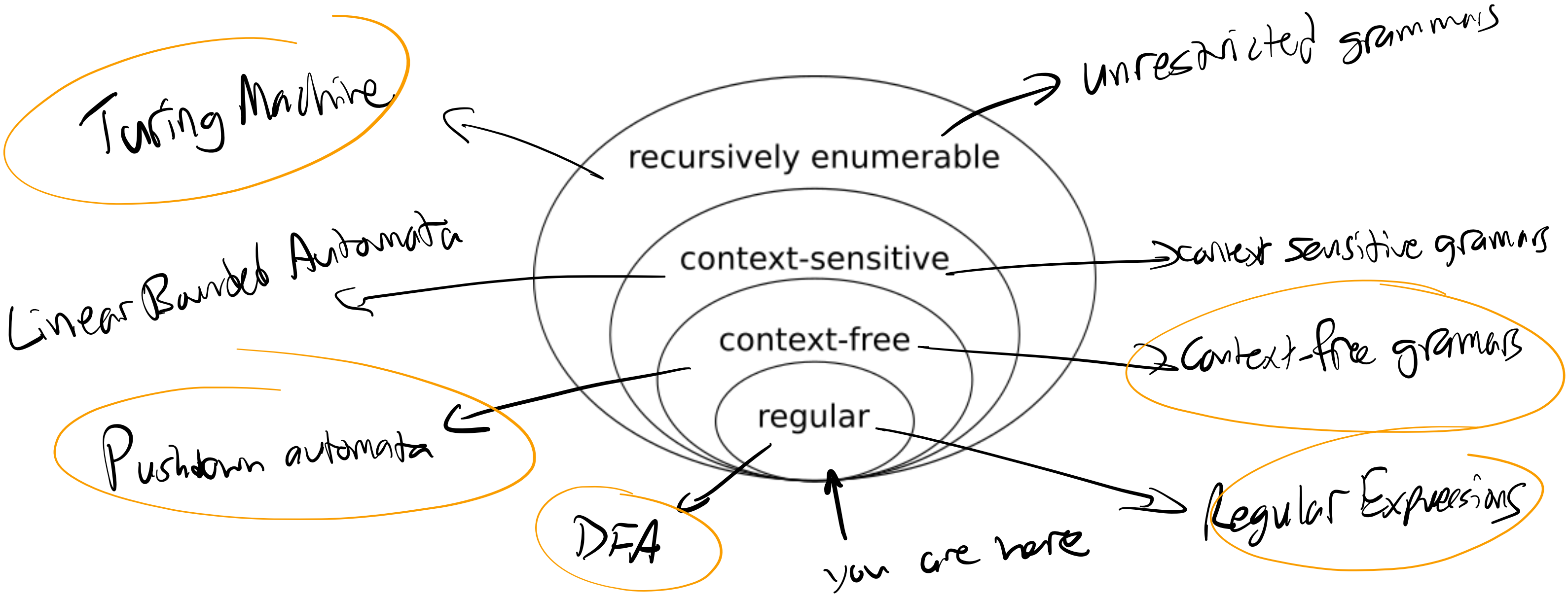
$$L(M) = \{w : \text{ w is a string over } \Sigma \text{ and } M \text{ accepts } w\} \tag{1}$$



What languages can be accepted by DFAs?

$$L(M) = \{\varepsilon,\ ab,\ aabb,\ aaabbb,\ \dots\}$$





Regular Languages

A language A is regular if there is a DFA M such that $L(M) = A$

Known Regular Languages:

- $L = \emptyset$
- $L = \{\varepsilon\}$
- $L = \{s\}$ for any $s \in \Sigma$

Regular Operations

- The **union** of two languages A and B is defined as $A \cup B = \{w : w \in A \text{ or } w \in B\}$.
- The **concatenation** or **product** of two languages A and B is defined as $AB = \{ww' : w \in A \text{ and } w' \in B\}$.
- The **closure** or **star** (or Kleene closure) of a language A is defined as: $A^* = \{u_1u_2 \dots u_k : k \geq 0 \text{ and } u_i \in A \text{ for all } i = 1, 2, \dots, k\}$

Equivalently:

- $A^0 = \{\varepsilon\}$
- $A^k = AA^{k-1}$
- $A^* = \bigcup_{i \geq 0}^{\infty} A^i$

Thm: The regular languages are closed under the regular operations. That is, if A and B are regular, then:

- $A \cup B$ is regular
- AB is regular
- A^* is regular

Proof: Not yet!