Nonregular languages

Sipser 1.4 (pages 77-82)

Nonregular languages?

- We now know:
  - Regular languages may be specified either by
    - regular expressions
    - DFAs or NFAs
- What if we can't find a regular expression or finite state automaton for a language?
- How do we show a language is not regular?

Limited memory

- Since finite state automata cannot back up when reading an input, they are allowed only a bounded amount of memory
- What about the language

$\{0^n 1^n \mid n \geq 0\}$?

PAUSE

Hmm… how can we prove a language is not regular?

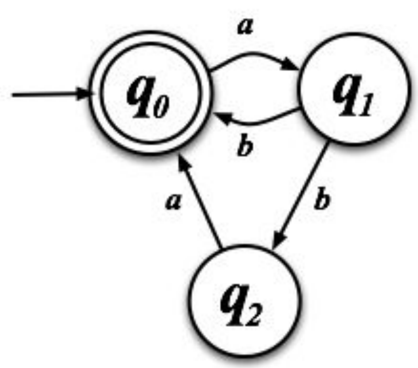
- What about
  - $\{w \mid w$ *has an equal number of 0s and 1s*$\}$
  - $\{w \mid w$ *has an equal number of occurrences of the substrings 01 and 10*$\}$

Try a different perspective

- Are there properties of a language that imply it is regular?
- What if the language is finite?

Try a different perspective

- How can a regular language be infinite?
  - Regular expression must have a star
  - Star operators correspond to cycles in the finite state automaton

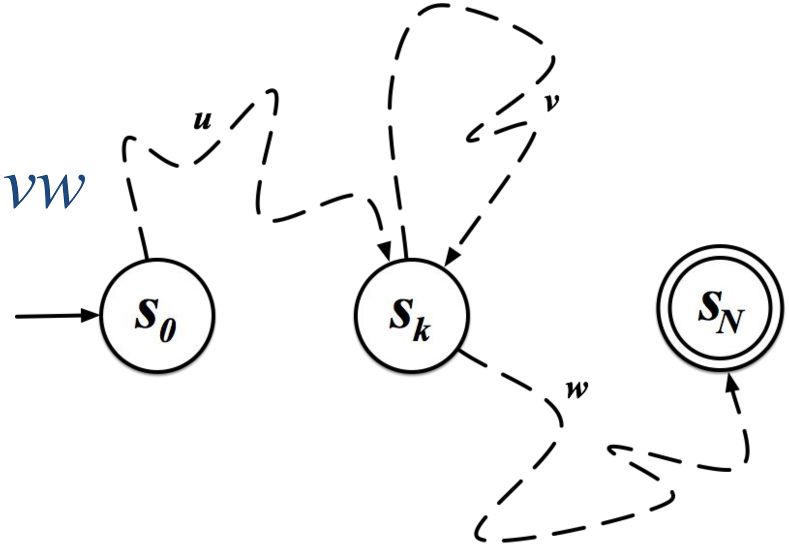


Pigeonhole principle

- Let M be a finite state machine with N states recognizing an infinite language
- Let $x \in L(M)$ *with* $|x| = N$
- Then there exists a sequence of states $s_0, s_1, s_2, \ldots, s_N$
- So: $N+1$ pigeons into N holes…
  - Some hole must have at least 2 pigeons!
  - I.e., at least two of the states must be the same, so there must be a cycle

Machine cycles

- Let s_k be the first repeated state; that is, $s_k = s_{k+c}$ for some c, $0 \leq k < k+c \leq N$.
- Where
  - $x = a_1 a_2 \ldots a_k \ldots a_{k+c} \ldots a_N = uvw$
  - $u = a_1 a_2 \ldots a_k$
  - $v = a_{k+1} \ldots a_{k+c}$
  - $w = a_{k+c+1} \ldots a_N$



- We conclude: $uv^i w \in L(M)$ for all $i \geq 0$.

The pumping lemma

- Theorem 1.70: If A is a regular language, then there is a number p where,

  if x is any string of length at least p, then $x = uvw$, such that

  1. For each $i \geq 0$, $uv^iw \in A$,
  2. $|v| > 0$, and
  3. $|uv| \leq p$.

So now...

- Is $L = \{0^n 1^n : n \geq 0\}$ regular?
- Let's prove it!
  - Suppose, for a contradiction, L is regular... then we can apply the pumping lemma
  - Let p be pumping length given by P.L.

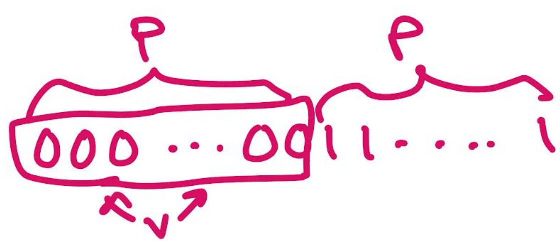
    Consider $x =$ _________ $\in L$.

    By P.L., $x = uvw$ such that $uv^iw \in L$ for all $i \geq 0$.
    - What does v look like?
    - Can we pump v to get a contradiction?
    - uv—w

Complete proof

$L = 0^n 1^n$

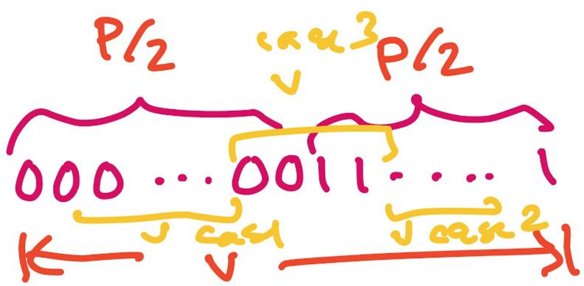
Pf Suppose, for a contradiction, L is regular. Let p be the pumping length given by P.L. Consider the string $x = 0^p 1^p \in L$. Since $|x| \geq p$, $x = uvw$ such that the conditions of the P.L. hold. By conditions 2 + 3, $v = 0^k$, where $k > 0$. Then, by condition 1, $uv^2w = 0^{p+k} 1^p \in L$. But $p + k \neq p$ since $k > 0$; thus $uv^2w \notin L$, giving the contradiction.



Choose x wisely

$L = 0^n 1^n$

Pf Suppose, for a contradiction, L is regular. Let p be the pumping length given by P.L. Consider the string $x = \boxed{0^{p/2} 1^{p/2}} \in L$. Since $|x| \geq p$, $x = uvw$ such that the conditions of the P.L. hold. By conditions ② + ③, $v =$, where . Then, by condition ①, $uv^{\boxed{0}}w = 0^{p-k} 1^{p} \in L$. But $p-k \neq p$ since $k>0$; thus $uv^2w \notin L$, giving the contradiction.



v could be

case 1: all 0s ($v = 0^k$)

case 2: all 1s ($v = 1^k$)

case 3: 0s & 1s ($v = 0^j 1^k$)

1 before a 0

$uv^0w = 0^{p-k} 1^{p-k} \in L$ $\quad j = k$

Reuse!

- Is $C = \{\, w \mid w$ *has an equal number of 0s and 1s*$\}$ a regular language?

Pf Suspose, for a contradiction, that $C = \{w \mid w$ has an equal # of 0s + 1s$\}$ were regular. We know that 0^*1^* describes a regular language A. Consider the language $A \cap C$; this is 0^n1^n. Since regular lang. are closed under $\cap$, $A \cap C$ is reg. But we just showed $0^n1^n = A \cap C$ is not reg., giving the contradiction.

Picking the substring to pump

- Is $PAL = \{w \in \{0, 1\}^* : w \text{ is a palindrome}\}$ a regular language?