- Questions, hw11, hw12
- FSM vs (right-regular) grammars vs. regular expressions
- The Marriage Problem
- Intro to Graphs
  - def'n
  - terms
  - representations

A FSM for recognizing floats.

Grammar
reg. exp.

DD^* or $DD^*.DD^*$

digit   0 or more digits

or $-DD^*$ or $-DD^*.DD^*$

$x = 789;$

$x = .3;$

$x = -.3;$

$([+-])DD^*(.DD^*)$

"one of"   optional

$\boxed{([+-])D^*(.DD^*)}$

A Grammar for floats

Let G be the grammar

$S \to AB$
$A \to aA$
$A \to aa$
$B \to b$

$S \overset{*}{\to} aaab$, $S \to AB \to Ab \to aab$

$S \to AB \to Ab$
$\to aAb \to aaAb \to aaaAb$
$\to aaaaab$

Let G' be:

$S \to aaS$
$S \to ab$
$S \to b$

aaa^*b is a regular expression corresponding

$L(G) = \{aab, aaab, aaaab, \dots\}$

What is a grammar generating all strings match the reg. exp.

+

$$\underbrace{([+-])}_{G}\underbrace{\overbrace{D^*}^{N}\overbrace{(.DD^*)}^{F}}_{\text{"V"}}$$

$S \to V$

$S \to GV$

$G \to +$

$G \to -$

$V \to N$

$V \to NF$

$N \to \varepsilon$

$N \to DN$

$F \to .DN$

$D \to 0$

$D \to 1$

$D \to 2$

$D \to 3$

$\vdots$

$D \to 9$

Th'm: Given any r.e. R, there exists a Grammar G such that $L(G) = L(R)$

Def'n: A right-regular Grammar is a grammar where each rule:

- has a single non-terminal on l.h.s.
- the r.h.s. has only at most <u>one</u> nonterminal, which occurs right-most.

Th'm: For any reg. exp R, there is a FSM M such that $L(M)$ (all sequences of input which lead to a designated "end state") $= L(R)$.

Btw- Both th'ms can go the other direction

Graphs (Rosen chpt. 9)

(chpt. 8 in 5ed.)

- def'n
- terms
- basic notions

Def'n: A Graph $G = \langle V, E \rangle$ where

V is a set of vertices,
and $E \subseteq V \times V$

Ex: $V = \{ROA, CLT, LAX, ORD\}$

$E = \{ \langle ROA, CLT \rangle, \langle ROA, ORD \rangle, \langle LAX, ORD \rangle, \ldots \}$

A graph is undirected if $\forall x, y \in V$

$\langle x, y \rangle \in E \rightarrow \langle y, x \rangle \in E$.

Def'n: "out- degree of a vertex" in a graph G:

For any $v \in V$,

$\text{out-deg}(v) = |\{ w \mid \langle v, w \rangle \in E \}|$

Def'n: A path in $G = \langle V, E \rangle$ is

a sequence of vertices

$v_1, v_2, v_3, \ldots, v_n$

such that $\forall i \in \mathbb{N}, i < n, \langle v_i, v_{i+1} \rangle \in E$

$\langle v_1, v_2 \rangle \in E$

$\langle v_2, v_3 \rangle \in E$

$\ldots$

$\langle v_{n-1}, v_n \rangle \in E$

Next time:

- algorithms on finding a path;
- spanning tree
- topological sort

Def'n: A cycle is a path

$v_1, v_2, \ldots, v_n, v_1$

Def'n: A Tree is a graph with no cycles.

The marriage problem:
(n men, n women)

Q: Terminate?

Yes —
proof: Each man makes at most n moves, so at most n^2 moves total, so terminates in $O(n^2)$ steps.

Def'n: a matching is optimal for k if they are matched with k^*, and k^* is their highest choice in any stable matching.

Th'm: The suitor algorithm gives each male their optimal choice.

Proof: Consider the first step when a guy M moves away from W, (his optimal choice)

This happened because he was bumped by some other guy N.
But (because W is optimal for M), there is some stable matching S which includes ⟨M,W⟩, and ⟨N, X⟩.

W: ...., N, ... M, ... (since M got bumped)

N: ...., X, ...

- W can't occur before X, in N's list: else S isn't stable.
- W can't occur after X, in N's list: this is the first round where a guy leaves their optimal choice

Lemma: If M is matched with their optimal choice W, then M is W's pessimal (least-preferred) choice, of any stable matching.

Proof: Consider a matching S which is stable, but W is matched with N. Then, W: ... N, M, ...
If W preferred M to N, then S wouldn't be stable: