CSC236 fall 2016

automata and languages

aka FSA aka FSM

Danny Heap
heap@cs.toronto.edu / BA4270 (behind elevators)
http://www.cdf.toronto.edu/~csc236h/fall/
416-978-5899

Using Introduction to the Theory of Computation, Chapter 7

Outline

FSAs (finite state automata)

notes

turnstile finite-state machine

what are the rules for turnstiles?

q_0 – locked
q_1 – unlocked

- no bikes!

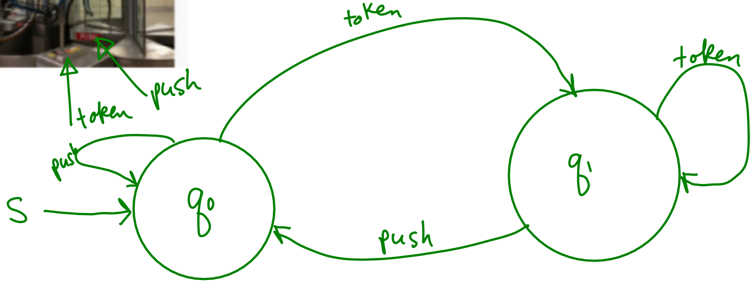
if locked
- token → unlocked

already unlocked → token → unlocked.

- push → locked

token

push



float machine

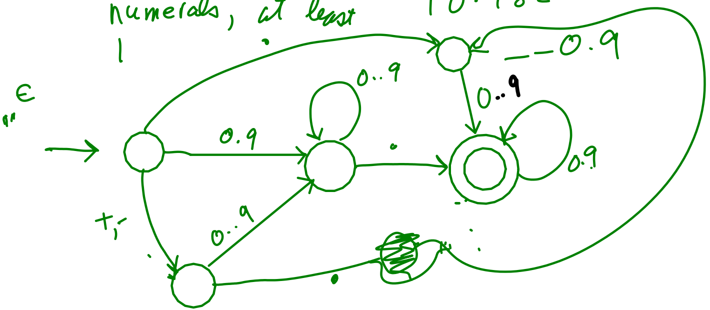
5 5.0 0.4 .4 5.

01.2 0..0.0 00...6

what strings are floats in Python?

rules must have . and numerals, at least 1

+0.482 −934.56



states needed to classify a string

what state is a stingy vending machine in, based on coins?
accepts only nickles, dimes, and quarters,
no change given, and everything costs 30 cents...
here's a useful toy (you'll need JRE)

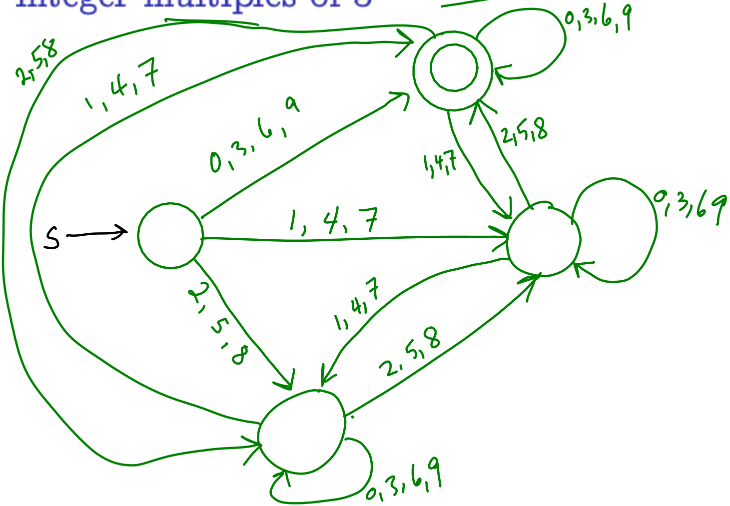
draw this

δ	0	5	10	15	20	25	≥ 30
n	5	10	15	20	25	≥ 30	≥ 30
d	10	15	20	25	≥ 30	≥ 30	≥ 30
q	25	≥ 30	≥ 30	≥ 30	≥ 30	≥ 30	≥ 30

integer multiples of 3

base 10

36



build an automaton with formalities...

quintuple: $(Q, \Sigma, q_0, F, \delta)$

Q is set of states, Σ is finite, non-empty alphabet, q_0 is start state

F is set of accepting states, and $\delta : Q \times \Sigma \mapsto Q$ is transition function

We can extend $\delta : Q \times \Sigma \mapsto Q$ to a transition function that tells us what state a **string** s takes the automaton to:

$$\delta^* : Q \times \Sigma^* \mapsto Q \qquad \delta^*(q, s) = \begin{cases} q & \text{if } s = \varepsilon \\ \delta(\delta^*(q, s'), a) & \text{if } s' \in \Sigma^*, \\ & \quad a \in \Sigma, s = s'a \end{cases}$$

String s is accepted if and only iff $\delta^*(q_0, s) \in F$, it is rejected otherwise.

example — an odd machine

devise a machine that accepts strings over $\{a, b\}$ with an odd number of as

Formal proof requires inductive proof of invariant:

$$\delta^*(E, s) = \begin{cases} E & \text{if } s \text{ has even number of } a\text{s} \\ O & \text{if } s \text{has odd number of } a\text{s} \end{cases}$$

more odd/even

L is the language of binary strings
with an odd number of as, but even length
Devise a machine for L

notes