CSC236 fall 2016

regular languages, regular expressions

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

regular expressions, regular languages

notes

they're equivalent:

$L = L(M)$ for some DFSA M $\Leftrightarrow$ $L = L(M')$ for some NFSA M' $\Leftrightarrow$
$L = L(R)$ for some regular expression R

step 1: convert $L(R)$ to $L(M')$

start with $\emptyset$, ε, $a \in \Sigma$

equivalence. . .

step 1.5: convert $L(R)$ to $L(M')$:

union, concatenation, stars

equivalence. . .

step 2: convert $L(M')$ to $L(M)$

use subset construction (also see example in week 11)

there could be $2^{|Q|}$ subsets to consider..

they're equivalent:

$L = L(M)$ for some DFSA $M \Leftrightarrow L = L(M')$ for some NFSA $M' \Leftrightarrow$
$L = L(R)$ for some regular expression R

step 3: convert $L(M)$ to $L(R)$, eliminate states

equivalence. . .

state elimination recipe for state q

1. $s_1 \ldots s_m$ are states with transitions **to** q, with labels $S_1 \ldots S_m$
2. $t_1 \ldots t_n$ are states with transitions **from** q, with labels $T_1 \ldots T_n$
3. Q is any self-loop on q
4. Eliminate q, and add (union) transition label $S_i Q^* T_j$ from s_i to t_j.

regular languages closure

Regular languages are those that can be denoted by a regular expression or accept by an FSA. In addition:

- L regular $\Rightarrow \overline{L}$ regular
- L regular $\Rightarrow Rev(L)$ regular

pumping lemma (see course notes, page 234)

If $L \subseteq \Sigma^*$ is a regular language, then there is some $n_L \in \mathbb{N}$ (n_L depends on L) such that if $x \in L$ and $|x| \geq n_L$ then:

- $\exists u, v, w \in \Sigma^*, x = uvw$
- $|v| > 0$
- $|uv| \leq n_L$
- $uv^k w \in L \forall k \in \mathbb{N}$

consequences of regularity

How about $L = \{1^n 0^n | n \in \mathbb{N}\}$

How about $L = \{w \in \Sigma^* \mid |w| = p, p \text{ is prime}\}$

notes