CS3311
Oct. 8, 2012
Monday 1

Definition 2.1.5

Let u be a string in Σ^*. The reversal of u, denoted u^R, is defined as follows:

i) basis: if length$(u) = 0$, then $u = \lambda$ and $\lambda^R = \lambda$

ii) recursive step: if $\underset{= n}{\text{length}(u)} > 0$

then $u = wa$ for some string w and $a \in \Sigma$, and length$(w) = n-1$ and $u^R = aw^R$

$u = \underbrace{abc}_{w}\underbrace{d}_{a}$

$d \; w^R$

$ab\underline{c}$

$c \; (ab)^R$

$b \; (a)^R$

$a \; \underbrace{\lambda^R}_{\lambda}$

$a^2 \quad (abc)^2 \quad abcabc$

$(ab)^R$ ↳ reverse

~~a~~ * big picture: We are trying to define a language to define languages.

$L \subseteq \Sigma^*$

set of strings that have some rules.

How to define rules:

1. recursive definitions : pseudo code like
2. regular sets — limited set operations to define languages
3. regular expressions — similar to regular sets (no set braces)

∞ → set is infinite

$$a^* = \bigcup_{i=0}^{\infty} a^i = \{\lambda, a, aa, \dots\}$$

set members are finite length strings

$a^* \quad b^* \quad a^* \quad b^* \quad a^*$

would not work.

$a^* \; (b \cup b^*) \; a^* \; (b \cup b^*) \; a^*$

are the above expressions different?

No. $b \cup b^* = b^*$

does not require a "b"

also does not require a "b"

but $b.b^* \neq b^*$

$\{b\} \; \{\epsilon, b, bb, bbb, \ldots\}$

$\{b, bb, bbb, \ldots\}$

Example: The set of strings containing an even number of "b"s.
Over $\Sigma = \{a, b\}$

$a^* (b a^* b)^* a^*$

$a^* (a^* b a^* b a^*)^* a^*$

$\underbrace{b a^* b}\ \underbrace{b a^* b}$

$a^* (a^* b a^* b a^*)^*$

λ	in	✓
a	$\in L$	X
b	out	✓
aa		
ab		
ba		
bb		

$a^* (b a^* b)^* a^*$

$a^* (b a^* b \mid a)^*$

$a^* (b a^* b a^*)^*$