Find strings of length < 4 matching $(a+b)^*b(a+ab)^*$. 9/12/18

You found out this includes every string with a 'b'.

Outline of an argument:

1) Any string matching must have the middle 'b'

2) Any string ending in a 'b' can be constructed using $\underbrace{(a+b)^*}_{\text{Constructs any string}}\ b$ (last b)

3) Any string with a 'b' ending in an 'a' can use the $(a+ab)^*$ to construct all the final a's.

(Regular) Grammars

Another way of specifying a language.

E.g. one of these

variables: A, B, C

letters (in our alphabet): x, y.

start variable: A

production rules:

$A \longrightarrow AB$

$AB \longrightarrow xBC$

$BC \longrightarrow yxC$

$C \longrightarrow AC$

$A \longrightarrow \lambda$

$C \longrightarrow y$

A string that is generated by this grammar:

$A \rightarrow AB \longrightarrow xBC \longrightarrow xyxC \longrightarrow xyxy$

a = xBC, b = $xyxC$

example of what symbols next page mean

$c = x \quad d = BC \quad e = \lambda$

$f = yxC$

Formally: A grammar ~~is~~ $G = (V, \Sigma, S, P)$ where

V is a finite set (variables)

Σ is our alphabet (also a finite set)

$S \in V$ is our start variable (or initial variable)

$P \subseteq (V \cup \Sigma)^* \times (V \cup \Sigma)^*$ (productions)

where we require that $(\lambda, w) \notin P$ for all w.

Some people make this V^*, requiring everything on LHS of a production be a variable.

Def: Let $a, b \in (V \cup \Sigma)^*$, $a \neq \lambda$. We say a produces/derives b in one step if there exist $c, d, e, f \in (V \cup \Sigma)^*$ such that $a = cde$, $b = cfe$, $(d, f) \in P$.

Def: Let $a, b \in (V \cup \Sigma)^*$, $a \neq \lambda$. We say a produces b if there exist a number k and strings $w_0 = a, w_1, w_2, \ldots, w_k = b$ such that w_i produces w_{i+1} in one step for all i.

Def: A grammar $G=(V, \Sigma, S, P)$ generates/produces/derives $w \in \Sigma^*$ if ~~it produces w from S~~. S produces w.

Def: A grammar is left-regular / right-regular if all the productions are of the form

$$X \longrightarrow Yw \quad \text{or} \quad X \longrightarrow w$$
$$X \longrightarrow wY \quad \text{or} \quad X \longrightarrow w$$

where X, Y are single variables, $w \in \Sigma^*$ is a string of letters (no variables)

E.g. ~~S~~

$$\left.\begin{array}{l} S \longrightarrow aX \\ X \longrightarrow bS \\ S \longrightarrow \lambda \end{array}\right\} = (ab)^*$$

$S \to \lambda$

or $S \to aX \to abS \to ab$

or $S \to aX \to abS \to abaX \to ababS \to abab$

etc.

An equivalent grammar

$$S \rightarrow abS$$
$$S \rightarrow \lambda$$

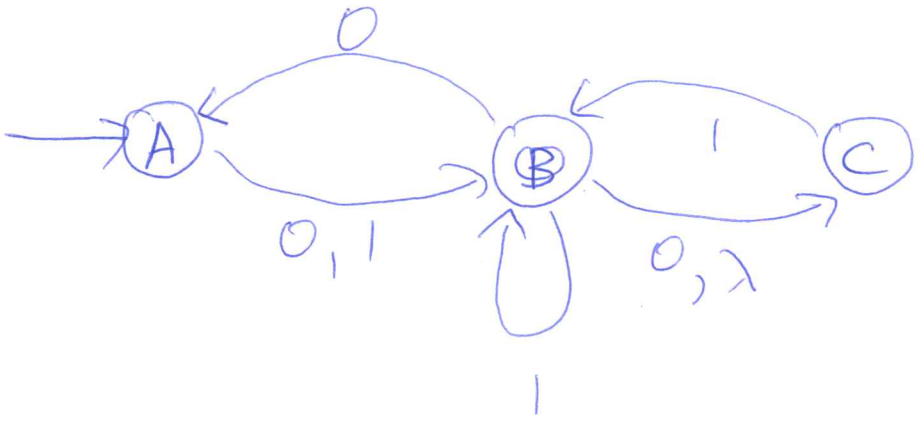
Sometimes abbreviated as

$$S \rightarrow abS \mid \lambda$$

"or"

Converting an NFA to a (right-regular) grammar

E.g.



A is my start variable

$$A \rightarrow 0B \mid 1B$$

$$B \rightarrow 0A \mid 1B \mid 0C \mid C \mid \lambda$$

$$C \rightarrow 1B$$

Since B is a final state.

000 is accepted 

$A \to 0B \to 00A \to 000B \to 000$

No daily HW for Friday