



1. (5 points) Construct a regular grammar for the language on  $\Sigma = \{a, b\}$  consisting of all strings not containing the substring  $aba$ .

♣ This is a regular grammar, so every sentential form will have a prefix. We first choose variables which describe that prefix, and then impose rules which enforce that description.

$V_{ab}$ : prefix ends in  $ab$  and contains no  $aba$ .

$V_b$ : prefix ends in  $b$  but not  $ab$ , and contains no  $aba$ .

$V_a$ : prefix ends in  $a$  and contains no  $aba$ .

$$\begin{aligned} G : S &\rightarrow \lambda \mid aV_a \mid bV_b \mid a \mid b \\ V_a &\rightarrow aV_a \mid bV_{ab} \mid a \mid b \\ V_b &\rightarrow aV_a \mid bV_b \mid a \mid b \\ V_{ab} &\rightarrow bV_b \mid b \quad \clubsuit \end{aligned}$$

2. (5 points) Construct a regular grammar for the language of all strings on  $\Sigma = \{a, b\}$  which are of odd length and contain exactly two  $b$ 's.

♣ Same method of design as the previous.

$E_0$ : prefix of even length, contains no  $b$ 's.  $E_0$  will play the role of the start symbol.

$E_1$ : prefix of even length, contains 1  $b$ .

$E_2$ : prefix of even length, contains 2  $b$ 's.

$O_0$ : prefix of odd length, contains no  $b$ 's.

$O_1$ : prefix of odd length, contains 1  $b$ .

$O_2$ : prefix of odd length, contains 2  $b$ 's. These prefixes are what we want in the language.

$$\begin{aligned} G : E_0 &\rightarrow aO_0 \mid bO_1 \\ O_0 &\rightarrow aE_0 \mid bE_1 \\ E_1 &\rightarrow aO_1 \mid bO_2 \\ O_1 &\rightarrow aE_1 \mid bE_2 \\ E_2 &\rightarrow aO_2 \\ O_2 &\rightarrow aE_2 \mid \lambda \end{aligned}$$

Note: If you don't want  $\lambda$  rules, you would write it in this form:

$$\begin{aligned} G : E_0 &\rightarrow aO_0 \mid bO_1 \\ O_0 &\rightarrow aE_0 \mid bE_1 \\ E_1 &\rightarrow aO_1 \mid bO_2 \mid b \\ O_1 &\rightarrow aE_1 \mid bE_2 \\ E_2 &\rightarrow aO_2 \mid a \\ O_2 &\rightarrow aE_2 \end{aligned}$$