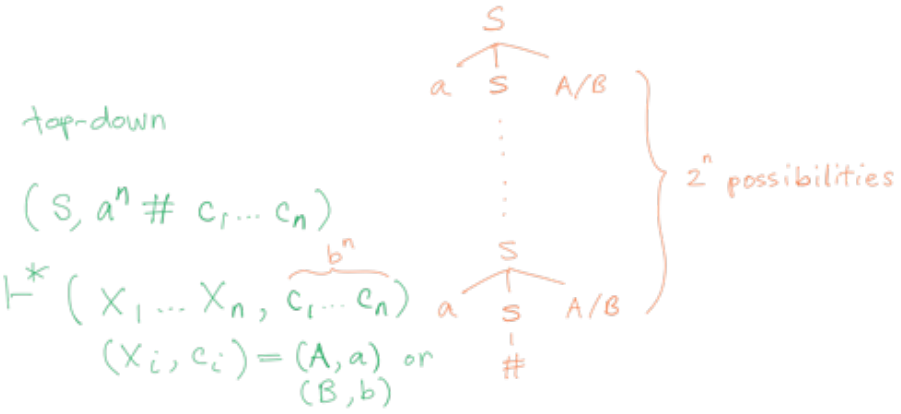
## 5/11 Mathematical Linguistics

Inefficiency of stack-based recognizers

$$S \to aSA \mid aSB \mid \#$$
$$A \to a \qquad B \to b$$

$a^n \# \{a,b\}^n$




$$L(G) = \{ w \in \Sigma^* \mid S \Rightarrow^* w \}$$

top-down ≈ leftmost derivation

$$\Rightarrow^* \subseteq (N \cup \Sigma)^* \times (N \cup \Sigma)^*$$

bottom-up ≈ rightmost

binary relation between strings

$$\text{holds-of} = \Rightarrow^* \cap (N \times \Sigma^*)$$
$$L(G) = \{ w \in \Sigma^* \mid S \text{ holds-of } w \}$$

CFG production

$A \to v_0 B_1 v_1 \cdots v_{\ell-1} B_\ell v_\ell$

$\ell = 0$
$A \to v_0$
A holds-of $v_0$

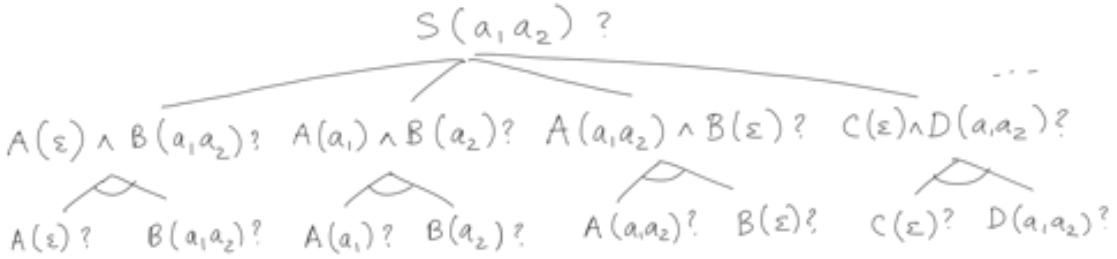
$A$ holds-of $v_0 x_1 v_1 \cdots v_{\ell-1} x_\ell v_\ell$
if $B_1$ holds-of $x_1 \wedge \cdots \wedge B_\ell$ holds-of $x_\ell$

$A(v_0 x_1 v_1 \cdots v_{\ell-1} x_\ell v_\ell) \leftarrow B_1(x_1), \ldots, B_\ell(x_\ell)$

logic programming

$\forall x_1 \cdots \forall x_\ell \left( (B_1(x_1) \wedge \cdots \wedge B_\ell(x_\ell)) \to A(v_0 x_1 v_1 \cdots v_{\ell-1} x_\ell v_\ell) \right)$

$S \to A\ B$ $\quad$ $S \to C\ D$




AND/OR tree

alternating Turing machine

input : $a_1 a_2 \cdots a_n$

$A(a_{i+1} \cdots a_j)$ $\quad$ $A(i,j)$

logarithmic space

LOGCFL $\subseteq$ P

# Dynamic Programming

$A(i, j)$

nonterminal — (inter-symbol) input position

## Datalog representation

rules P

$S(i,k) \leftarrow NP(i,j), VP(j,k)$ — $S \rightarrow NP\ VP$

$Aux(i,j) \leftarrow does(i,j)$ — $Aux \rightarrow does$

input D

$a_1(0,1), a_2(1,2), \ldots, a_n(\overline{n-1}, \overline{n})$ — $a_1 a_2 \cdots a_n$


Derive theorems from $P \cup D$

Naive bottom-up evaluation

rule

$$\frac{P \leftarrow Q_1, \ldots, Q_\ell \qquad Q_1\theta \quad \cdots \quad Q_\ell\theta}{P\theta}$$

ground fact

closed atomic formula

$\theta$: ground substitution