Theory of Computation

Dr Samayveer Singh

Context-Free Language

Context-Free Grammar

- A grammar is context-free if every production is of the form

$$A \rightarrow \alpha$$

where $A \in V_N$ *and* $\alpha \in (V_N \cup \Sigma)^*$.

Example: Context Free Grammar

- Give the context free grammar which generates a string containing only a's.

P :

$S \rightarrow a$

$S \rightarrow aS$

$CFG = (\{S\}, \{a\}, P, S)$

Example: Context Free Grammar

- Give the context free grammar to generate a string containing a, b in any sequence.

P: $S \rightarrow aS$

$S \rightarrow bS$

$S \rightarrow a$

$S \rightarrow b$

$CFG = (\{S\}, \{a, b\}, P, S)$

Example: Context Free Grammar

- Give the context free grammar which generating an alternating sequence of 0's and 1's.

$S \to 0T$

$S \to 1U$

$T \to 1U$

$U \to 0T$

$T \to 1$

$U \to 0$

$CFG = (\{S, T, U\}, \{0, 1\}, P, S)$

Example: Context Free Grammar

- Give the context free grammar for generating a string in which no consecutive b's can occur but only a's can be consecutive.

Sol^n

$S \to a$

$S \to b$

$S \to aS$

$S \to bT$

$T \to aS$

$T \to a$

$CFG = (\{S, T\}, \{a, b\}, P, S)$

Example: Context Free Grammar

- Write the context free grammar for the string which don't contain 3-consecutive b's.

P:

$S \to a$

$S \to b$

$S \to aS$

$S \to bT$

$T \to bU$

$U \to aS$

$T \to a$

$T \to b$

$U \to a$

$CFG = \{ \{S, T, U\}, \{a, b\}, P, S \}$

Example: Context Free Grammar

- To generate the string write the context free grammar which contains at least one occurrence of aaa.

P:

$S \to TaaaT$

$T \to \Lambda$

$T \to aT$

$T \to bT$

$CFG = (\{S, T\}, \{a, b\}, P, S)$

Example: Context Free Grammar

- Give the context free grammar which generates a string containing at least 2a's.

P:

$S \to TaTaT$

$T \to aT$

$T \to bT$

$T \to \Lambda$

$CFG = (\{S, T\}, \{a, b\}, P, S)$

Example: Context Free Grammar

$$L=\{\mathbf{a}^n\ \mathbf{b}^n \mid n\geq 0\}$$

Productions are:

$S \rightarrow \varepsilon \mid \mathbf{a}\ S\ \mathbf{b}$

$S \rightarrow aSb$

$S \rightarrow \lambda$

Formally: $G = (\{S\}, \{\mathbf{a},\mathbf{b}\},$

$\{S \rightarrow \varepsilon, S \rightarrow \mathbf{a}\ S\ \mathbf{b}\}, S)$

Example: Context Free Grammar

$P \rightarrow (P)$
$\rightarrow (PP)$
$\rightarrow (()())$

- Design a CFG which accepts all strings of balanced parentheses

$$\mathbf{P} \rightarrow \underline{\varepsilon} \mid \underline{(\ \mathbf{P}\ )} \mid \underline{\mathbf{P}\ \mathbf{P}}$$

( )( )( )( )( )( )( )

Formally: G = ({P}, {**(**,**)**},

{P → ε | **(** P **)** | P P}, P)

bc

$P \rightarrow (P)$
$\rightarrow ( )$

$S \rightarrow aSc$ $S \rightarrow X$
$S \rightarrow aXc$
$X \rightarrow \Lambda$

- L={$\mathbf{a}^m\ \mathbf{b}^n\ \mathbf{c}^{m+n} \mid m,n \geq 0$}

Rewrite as {$\mathbf{a}^m\ \mathbf{b}^n\ \mathbf{c}^n\ \mathbf{c}^m \mid m,n \geq 0$}: $X \rightarrow bXc$

$P \rightarrow PP$

S → A | **a** S **c,** A → ε | **b** A **c** $S \rightarrow \Lambda$

$\rightarrow (P)(P)$

Formally: G = ({S, A}, {**a, b, c**},

$\rightarrow ( )( )$

{S → A | **a** S **c,** A → ε | **b** A **c**}, S)

Example: Context Free Grammar

- Design a CFG for all binary strings with an even number of 0's.

$$S \rightarrow 1S \mid 0A0S \mid \varepsilon,$$

$$A \rightarrow 1A \mid \varepsilon$$

$$S \rightarrow 0A0S$$

$$\rightarrow 0A00A0S$$

$$0000$$

Example: Context Free Grammar

- A CFG for the regular language corresponding to the RE $00*11*$. The language is the concatenation of two languages: all strings of zeroes with all strings of ones.

$S \longrightarrow CD$

$C \longrightarrow 0C \mid 0$

$D \longrightarrow 1D \mid 1$

$RE = 00^* 11^*$