

Lecture 12 – Examples of Context-Free Grammars

COSE215: Theory of Computation

Jihyeok Park



2023 Spring

- A context-free grammar (CFG):

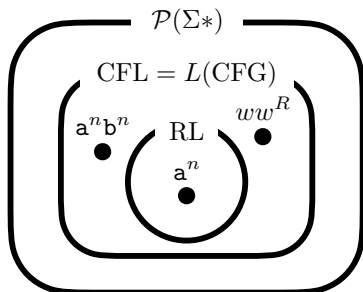
$$G = (V, \Sigma, S, P)$$

- The language of a CFG G :

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

- A language L is a context-free language (CFL):

$$\exists \text{ CFG } G. L(G) = L$$



1. Examples of Context-Free Grammars

Example 1: Regular Languages

Example 2: $b^n a^m b^{2n}$

Example 3: Well-Formed Brackets

Example 4: Equal Number of a's and b's

Example 5: Unequal Number of a's and b's

Example 6: Arithmetic Expressions

Example 7: Regular Expressions

Example 8: Simplified Scala Syntax

Theorem (RLs are CFLs)

If a language L is a regular language (RL), then L is a CFL.

Proof) For a given RE R , construct a CFG G such that $L(G) = L(R)$.

RE R	CFG G
\emptyset	$S \rightarrow S$
ϵ	$S \rightarrow \epsilon$
$a \in \Sigma$	$S \rightarrow a$
$R_1 \mid R_2$	$S \rightarrow S_1 \mid S_2$
$R_1 \cdot R_2$	$S \rightarrow S_1 S_2$
R_1^*	$S \rightarrow \epsilon \mid S_1 S$
(R_1)	$S \rightarrow S_1$

where S_1 and S_2 are start variables of CFGs G_1 and G_2 such that $L(G_1) = L(R_1)$ and $L(G_2) = L(R_2)$, respectively.

Example 1: Regular Languages

For a given RE R , construct a CFG G such that $L(G) = L(R)$.

- $R = \epsilon | ab | ba$

$$\begin{array}{llll} S \rightarrow F | D & A \rightarrow a & C \rightarrow AB & E \rightarrow \epsilon \\ & B \rightarrow b & D \rightarrow BA & F \rightarrow E | C \end{array}$$

Its simplified version:

$$S \rightarrow \epsilon | ab | ba$$

- $R = (\epsilon | a)^*$

$$S \rightarrow \epsilon | AS \quad A \rightarrow \epsilon | a$$

- $R = (0 | 1(01^*0)^*1)^*$

$$\begin{array}{lll} S \rightarrow \epsilon | AS & A \rightarrow 0 | 1B1 & C \rightarrow 0D0 \\ & B \rightarrow \epsilon | CB & D \rightarrow \epsilon | 1D \end{array}$$

Example 2: $b^n a^m b^{2n}$

Construct a CFG for the language:

$$L = \{b^n a^m b^{2n} \mid n, m \geq 0\}$$

$$S \rightarrow A \mid bSbb$$

$$A \rightarrow \epsilon \mid aA$$

A derivation for bbaaabbbb:

$$\begin{aligned} S &\Rightarrow bSbb && \Rightarrow bbSbbbb && \Rightarrow bbAbbbb \\ &\Rightarrow bbaAbbbb && \Rightarrow bbaaAbbbb && \Rightarrow bbaaaAbbbb \\ &\Rightarrow bbaaabbbb \end{aligned}$$

Example 3: Well-Formed Brackets

Construct a CFG for the language:

$$L = \{w \in \{ (,), \{, \}, [,] \}^* \mid w \text{ is well-formed} \}$$

$$S \rightarrow \epsilon \mid (S) \mid \{S\} \mid [S] \mid SS$$

A derivation for $(\{\})\{ }\{ () []]$:

$$\begin{aligned} S &\Rightarrow SS && \Rightarrow SSS && \Rightarrow (S)SS \\ &\Rightarrow (\{S\})SS && \Rightarrow (\{\})SS && \Rightarrow (\{\})\{S\}S \\ &\Rightarrow (\{\})\{S\}S && \Rightarrow (\{\})\{S\}[S] && \Rightarrow (\{\})\{S\}[SS] \\ &\Rightarrow (\{\})\{S\}[(S)S] && \Rightarrow (\{\})\{S\}[()S] && \Rightarrow (\{\})\{S\}[() [S]] \\ &\Rightarrow (\{\})\{S\}[() []] \end{aligned}$$

Example 4: Equal Number of a's and b's

Construct a CFG for the language:

$$L = \{w \in \{a, b\}^* \mid N_a(w) = N_b(w)\}$$

where $N_a(w)$ and $N_b(w)$ are the number of a's and b's in w , respectively.

$$S \rightarrow \epsilon \mid aSb \mid bSa \mid SS$$

The left-most derivation for $abbaaabb$:

$$\begin{aligned} S &\xrightarrow{\text{lm}} aSb && \xrightarrow{\text{lm}} aSSb && \xrightarrow{\text{lm}} abSaSb \\ &\xrightarrow{\text{lm}} abbSaaSb && \xrightarrow{\text{lm}} abbaaSb && \xrightarrow{\text{lm}} abbaaaSbb \\ &\xrightarrow{\text{lm}} abbaaabb \end{aligned}$$

Example 5: Unequal Number of a's and b's

Construct a CFG for the language:

$$L = \{w \in \{a, b\}^* \mid N_a(w) \neq N_b(w)\}$$

where $N_a(w)$ and $N_b(w)$ are the number of a's and b's in w , respectively.

$$S \rightarrow P \mid N$$

$$P \rightarrow ZP \mid aP \mid aZ$$

$$N \rightarrow ZN \mid bN \mid bZ$$

$$Z \rightarrow \epsilon \mid aZb \mid bZa \mid ZZ$$

The right-most derivation for aabbbaaab:

$$\begin{array}{lll} S & \xrightarrow{\text{rm}} & P & \xrightarrow{\text{rm}} & ZP & \xrightarrow{\text{rm}} & ZaZ \\ & \xrightarrow{\text{rm}} & ZaaZb & \xrightarrow{\text{rm}} & Zaab & \xrightarrow{\text{rm}} & ZZaab \\ & \xrightarrow{\text{rm}} & ZbZaaab & \xrightarrow{\text{rm}} & Zbaaab & \xrightarrow{\text{rm}} & aZbbaaab \\ & \xrightarrow{\text{rm}} & aaZbbbaaab & \xrightarrow{\text{rm}} & aabbbaaab & & \end{array}$$

Example 6: Arithmetic Expressions

An **arithmetic expression** is defined with the following CFG:

$$\begin{aligned} S &\rightarrow N \mid X \mid S+S \mid S*S \mid (S) \\ N &\rightarrow 0 \mid \dots \mid 9 \mid 0N \mid \dots \mid 9N \\ X &\rightarrow a \mid \dots \mid z \end{aligned}$$

The left-most derivation for $13*(2+x)$:

$$\begin{array}{llll} S & \xRightarrow{\text{lm}} & S*S & \xRightarrow{\text{lm}} & N*S & \xRightarrow{\text{lm}} & 1N*S \\ & \xRightarrow{\text{lm}} & 13*S & \xRightarrow{\text{lm}} & 13*(S) & \xRightarrow{\text{lm}} & 13*(S+S) \\ & \xRightarrow{\text{lm}} & 13*(N+S) & \xRightarrow{\text{lm}} & 13*(2+S) & \xRightarrow{\text{lm}} & 13*(2+X) \\ & \xRightarrow{\text{lm}} & 13*(2+x) & & & & \end{array}$$

Example 7: Regular Expressions

Is the following language regular? or context-free?

$$L = \{w \in \{\emptyset, \varepsilon, a, b, |, *, (,)\}^* \mid w \text{ is a regular expression over } \{a, b\}\}$$

We can prove that L is not regular using the pumping lemma.

(Hint: consider a word $(^n\varepsilon)^n$ for a given $n > 0$)

The language L is context-free:

$$S \rightarrow \emptyset \mid \varepsilon \mid a \mid b \mid S \mid S \mid SS \mid S^* \mid (S)$$

The right-most derivation for $(\varepsilon \mid ab)^*$:

$$\begin{aligned} S &\xrightarrow{\text{rm}} S^* && \xrightarrow{\text{rm}} (S)^* && \xrightarrow{\text{rm}} (S \mid S)^* \\ &\xrightarrow{\text{rm}} (S \mid SS)^* && \xrightarrow{\text{rm}} (S \mid Sb)^* && \xrightarrow{\text{rm}} (S \mid ab)^* \\ &\xrightarrow{\text{rm}} (\varepsilon \mid ab)^* \end{aligned}$$

We can define a CFG for a simplified version of Scala syntax¹:

(Scala Program)	$S \rightarrow E \mid E ; S$
(Expressions)	$E \rightarrow N \mid X \mid E + E \mid E - E \mid E * E \mid E / E$ $\mid \text{val } X : T = E$ $\mid \text{def } X(P) : T = E$ $\mid E(A)$ $\mid \text{if } (E) E \text{ else } E$ $\mid \text{trait } T(P)$ $\mid \text{case class } T(P)$ $\mid E \text{ match } \{ C \}$
(Numbers)	$N \rightarrow 0 \mid \dots \mid 9 \mid 0N \mid \dots \mid 9N$
(Variables)	$X \rightarrow Y \mid YX$ $Y \rightarrow _ \mid a \mid \dots \mid z \mid A \mid \dots \mid Z$
(Types)	$T \rightarrow X \mid T [T] \mid T \Rightarrow T$
(Parameters)	$P \rightarrow \epsilon \mid X : T \mid P , X : T$
(Arguments)	$A \rightarrow \epsilon \mid E \mid A , E$
(Cases)	$C \rightarrow \text{case } E \Rightarrow E \mid C ; \text{case } E \Rightarrow E$

¹<https://docs.scala-lang.org/scala3/reference/syntax.html>

```
def sum(n: Int): Int = n match { case 0 => 0; case n => n + sum(n - 1) }
```

The left-most derivation for this program:

$$\begin{aligned}
 S &\xRightarrow{\text{lm}} \text{def } X(P): T = E && \xRightarrow{\text{lm}}^* \text{def sum}(P): T = E \\
 &\xRightarrow{\text{lm}}^* \text{def sum}(X: T): T = E && \xRightarrow{\text{lm}}^* \text{def sum}(n: \text{Int}): \text{Int} = E \\
 &\xRightarrow{\text{lm}}^* \text{def sum}(n: \text{Int}): \text{Int} = E \text{ match } \{ C \} \\
 &\xRightarrow{\text{lm}}^* \text{def sum}(n: \text{Int}): \text{Int} = n \text{ match } \{ C \} \\
 &\xRightarrow{\text{lm}}^* \text{def sum}(n: \text{Int}): \text{Int} = n \text{ match } \{ \text{case } E \Rightarrow E ; C \} \\
 &\xRightarrow{\text{lm}}^* \text{def sum}(n: \text{Int}): \text{Int} = n \text{ match } \{ \text{case } 0 \Rightarrow 0 ; C \} \\
 &\xRightarrow{\text{lm}}^* \text{def sum}(n: \text{Int}): \text{Int} = n \text{ match } \{ \text{case } 0 \Rightarrow 0 ; \text{case } E \Rightarrow E \} \\
 &\xRightarrow{\text{lm}}^* \text{def sum}(n: \text{Int}): \text{Int} = n \text{ match } \{ \text{case } 0 \Rightarrow 0 ; \text{case } n \Rightarrow E \} \\
 &\xRightarrow{\text{lm}}^* \text{def sum}(n: \text{Int}): \text{Int} = n \text{ match } \{ \text{case } 0 \Rightarrow 0 ; \text{case } n \Rightarrow E + E \} \\
 &\xRightarrow{\text{lm}}^* \text{def sum}(n: \text{Int}): \text{Int} = n \text{ match } \{ \text{case } 0 \Rightarrow 0 ; \text{case } n \Rightarrow n + E \} \\
 &\xRightarrow{\text{lm}}^* \text{def sum}(n: \text{Int}): \text{Int} = n \text{ match } \{ \text{case } 0 \Rightarrow 0 ; \text{case } n \Rightarrow n + \text{sum}(n - 1) \}
 \end{aligned}$$

1. Examples of Context-Free Grammars

Example 1: Regular Languages

Example 2: $b^n a^m b^{2n}$

Example 3: Well-Formed Brackets

Example 4: Equal Number of a's and b's

Example 5: Unequal Number of a's and b's

Example 6: Arithmetic Expressions

Example 7: Regular Expressions

Example 8: Simplified Scala Syntax

- Parse Trees and Ambiguity

Jihyeok Park
jihyeok_park@korea.ac.kr
<https://plrg.korea.ac.kr>