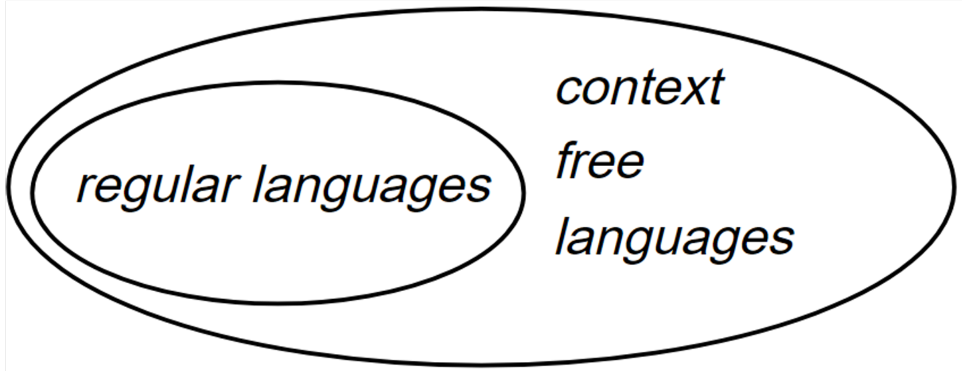
COS210 - Theoretical Computer Science
Context-Free Languages (Part 1)

Context-Free Languages

Context-free languages are a **superclass** of the regular languages:



The following non-regular language is an example of a **context-free language**:

$$L = \{0^n 1^n : n \geq 0\}$$

We have seen that regular languages can be described by finite automata

We will now see that context-free languages can be described by **context-free grammars**

Context-Free Grammars

In practice, context-free grammars are used to define the syntax of **programming languages**

A context-free grammar is defined by a set of **substitution rules** used to generate a context-free language

Example set of rules:

1. $S \rightarrow AB$
2. $A \rightarrow a$
3. $A \rightarrow aA$
4. $B \rightarrow b$
5. $B \rightarrow bB$

Here, S, A, and B are **variables**. S is a special variable namely the **start variable**. Both a and b are **terminals**

Context-Free Grammars

Substitution rules can be used to **generate strings** over terminals

1. $S \rightarrow AB$
2. $A \rightarrow a$
3. $A \rightarrow aA$
4. $B \rightarrow b$
5. $B \rightarrow bB$

How to use rules:

I) Begin with an **initial string** equal to the start variable S

II) Use a rule to **replace a variable** in the current string

III) **Repeat step II** until the string consists of terminals only

Context-Free Grammars: Example

We assume given is the same grammar (set of rules) as before:

1. $S \rightarrow AB$
2. $A \rightarrow a$
3. $A \rightarrow aA$
4. $B \rightarrow b$
5. $B \rightarrow bB$

Derive a string based on the rules:

$$
\begin{aligned}
S &\Longrightarrow AB && \text{(rule 1)} \\
&\Longrightarrow aAB && \text{(rule 3)} \\
&\Longrightarrow aAbB && \text{(rule 5)} \\
&\Longrightarrow aaAbB && \text{(rule 3)} \\
&\Longrightarrow aaAbb && \text{(rule 4)} \\
&\Longrightarrow aaabb && \text{(rule 2)}
\end{aligned}
$$

Consequently, the string *aaabb* is an element of the context-free language defined by the context-free grammar above

Context-Free Grammars: Example

1. $S \to AB$
2. $A \to a$
3. $A \to aA$
4. $B \to b$
5. $B \to bB$

The **language of a grammar** is the set of all strings that

- can be **derived** from the start variable and
- only contain **terminals**

The language of the grammar above is:

$$L = \{a^m b^n : m \geq 1, n \geq 1\}$$

Context-Free Grammar: Formal Definition

Definition (Context-Free Grammar)

A context-free grammar is a **4-tuple** $G = (V, \Sigma, R, S)$, where

- V is a finite set, whose elements are called **variables**,
- Σ is a finite set, whose elements are called **terminals**,

  $V \cap \Sigma = \emptyset$ (sets of variables and terminals have no shared elements),
- R is a finite set, whose elements are called **substitution rules**. Each rule has the form $A \to w$, where $A \in V$ and $w \in (V \cup \Sigma)^*$,
- S is an element of V; it is called the **start variable**.

Context-Free Grammar: Formal Definition

Our earlier example can be formally defined as the context-free grammar $G = (V, \Sigma, R, S)$, where

- $V = \{S, A, B\}$
- $\Sigma = \{a, b\}$
- $R = \{S \to AB, \quad A \to a, \quad A \to aA, \quad B \to b, \quad B \to bB\}$

and the following conditions hold:

- $V \cap \Sigma = \emptyset$
- $S \in V$ is the start variable

Context-Free Grammar: Derivation

General form of a substitution rule is $A \rightarrow w$ where A is a variable and w is a string over variables and/or terminals

$A \rightarrow w$ can be used to **make a derivation** from any string that contains A, i.e. a string of the form uAv where u and v are arbitrary sub strings

Given uAv, we can use the rule $A \rightarrow w$ to derive the new string uwv

Context-Free Grammar: Derivation

General form of a substitution rule is $A \to w$ where A is a variable and w is a string over variables and/or terminals

$A \to w$ can be used to **make a derivation** from any string that contains A, i.e. a string of the form uAv where u and v are arbitrary sub strings

Given uAv, we can use the rule $A \to w$ to derive the new string uwv

Definition (Derived-In-One-Step-From: $\implies$)

Let $G = (V, \Sigma, R, S)$ be context-free grammar. Let A be a variable in V and let u, v, and w be strings in $(V \cup \Sigma)^*$ such that $A \to w$ is a rule in R. Then we say that the string uwv can be **derived in one step from** the string uAv, and write this as

$$uAv \implies uwv$$

In our example we have $aaAbb \implies aaaAbb$, since $A \to aA$ is a rule in R

Context-Free Grammar: Derivation

We can extend the single-step derivation to **arbitrary many** derivation steps:

Definition (Derived-From: $\overset{*}{\Longrightarrow}$)

Let $G = (V, \Sigma, R, S)$ be context-free grammar. Let u and v be strings in $(V \cup \Sigma)^*$. Then we say that v can be **derived from** u, written as

$$u \overset{*}{\Longrightarrow} v$$

if one of the following two conditions holds:

- $u = v$ **(zero-step derivation)** or
- there exists a sequence of strings $u_1, u_2, \ldots, u_k$ with $k \geq 2$ such that
  - $u_1 = u$,
  - $u_k = v$, and
  - $u_1 \Longrightarrow u_2 \Longrightarrow \ldots \Longrightarrow u_k$.

  (single- or multi-step derivation)

Context-Free Grammar: Derivation

$u \overset{*}{\Longrightarrow} v$ means that by starting with string u and applying rules **zero or more times**, we obtain string v.

In our example we have that

$$aAbB \overset{*}{\Longrightarrow} aaAbb \quad (aaAbb \text{ can be derived from } aAbB)$$

since

$$aAbB \underbrace{\Longrightarrow}_{\text{rule 3: } A \to aA} aaAbB \underbrace{\Longrightarrow}_{\text{rule 4: } B \to b} aaAbb$$

Language of a Context-Free Grammar

Definition (Language of a Grammar)

Let $G = (V, \Sigma, R, S)$ be a context-free grammar. The **language** of G is defined as the set of all strings in Σ^* that can be **derived from** the start variable S:

$$L(G) \;=\; \{w \in \Sigma^* : S \overset{*}{\Longrightarrow} w\}$$

Now we can also give a formal definition of a context-free language:

Definition (Context-Free Language)

A language L is called **context-free**, if there exists a context-free grammar G such that $L(G) = L$.

Context-Free Grammar: Shorthand Notation for Rules

Rules of a context-free grammar can be shortened from

$$R = \{\underbrace{S \rightarrow AB,}_{\text{rule 1}} \quad \underbrace{A \rightarrow a,}_{\text{rule 2}} \quad \underbrace{A \rightarrow aA,}_{\text{rule 3}} \quad \underbrace{B \rightarrow b,}_{\text{rule 4}} \quad \underbrace{B \rightarrow bB}_{\text{rule 5}}\}$$

to

$$R = \{\underbrace{S \rightarrow AB,}_{\text{rule 1}} \quad \underbrace{A \rightarrow a|aA,}_{\text{rules 2,3}} \quad \underbrace{B \rightarrow b|bB}_{\text{rules 4,5}}\}$$

where **|** is shorthand for **or**

For instance, the rule $A \rightarrow a|aA$ indicates that the variable A can be substituted by a **or** by aA.

Context-Free Grammar: Properly Nested Brackets Example

Consider the following context-free grammar $G = (V, \Sigma, R, S)$

- $V = \{S\}$
- $\Sigma = \{a, b\}$
- $R = \{S \to \underbrace{\epsilon}_{1} \mid \underbrace{aSb}_{2} \mid \underbrace{SS}_{3}\}$

For instance, we can derive:

$$
\begin{aligned}
S &\implies aSb && (2)\\
&\implies aaSbb && (2)\\
&\implies aaSSbb && (3)\\
&\implies aaSaSbbb && (2)\\
&\implies aaaSbaSbbb && (2)\\
&\implies aaabaSbbb && (1)\\
&\implies aaababbb && (1)
\end{aligned}
$$

Now assume $a =$ "(" is a **left-bracket** and $b =$ ")" is a **right-bracket**

Context-Free Grammar: Properly Nested Brackets Example

The language $L(G)$ consists of all strings of properly nested brackets. For example:

$$aaababbb \; = \; ((()()))$$

Any string of properly nested brackets is either

- **empty** (derived using rule $S \to \epsilon$),
- consists of a **left-bracket**, followed by an arbitrary string of **properly nested brackets**, followed by a **right-bracket** (derived using rule $S \to aSb$), or
- consists of an arbitrary string of **properly nested brackets**, followed by an arbitrary string of **properly nested brackets** (derived using rule $S \to SS$).

Context-Free Grammar: Non-Regular Language Example

Consider the following language:

$$L_1 = \{0^n 1^n : n \geq 0\}$$

In a previous lecture we have proven that L_1 is **non-regular**

It is easy to show that L_1 is in fact **context-free**

Context-Free Grammar: Non-Regular Language Example

Consider the following language:

$$L_1 = \{0^n 1^n : n \geq 0\}$$

In a previous lecture we have proven that L_1 is **non-regular**

It is easy to show that L_1 is in fact **context-free**

Observation: each $w \in L_1$ either of the form

- $w = \epsilon$, or
- w consists of a 0, followed by an arbitrary $v \in L_1$, followed by a 1

Example:

$$w = 0\,\underbrace{0011}_{v}\,1 \in L_1 \quad \text{and} \quad v \in L_1$$

Context-Free Grammar: Non-Regular Language Example

$$L_1 = \{0^n1^n : n \geq 0\}$$

The following grammar takes these two cases into account:

- $G_1 = (V_1, \Sigma_1, R_1, S_1)$
- $V_1 = \{S_1\}$
- $\Sigma_1 = \{0, 1\}$
- $R_1 = \{S_1 \to \epsilon | 0S_11\}$

Context-Free Grammar: Non-Regular Language Example

$$L_1 = \{0^n 1^n : n \geq 0\}$$

The following grammar takes these two cases into account:

- $G_1 = (V_1, \Sigma_1, R_1, S_1)$
- $V_1 = \{S_1\}$
- $\Sigma_1 = \{0, 1\}$
- $R_1 = \{S_1 \to \epsilon | 0S_1 1\}$

Exercise: Which grammar would define the following language?

- $L = \{0^n 1^n : n \geq 0\} \cup \{1^n 0^n : n \geq 0\}$

Context-Free Grammar: Non-Regular Language Example

Solution:

$$L = \underbrace{\{0^n1^n : n \geq 0\}}_{L_1} \cup \underbrace{\{1^n0^n : n \geq 0\}}_{L_2}$$

- $G_1 = (V_1, \Sigma_1, R_1, S_1)$
- $V_1 = \{S_1\}$
- $\Sigma_1 = \{0, 1\}$
- $R_1 = \{S_1 \rightarrow \epsilon | 0S_11\}$

- $G_2 = (V_2, \Sigma_2, R_2, S_2)$
- $V_2 = \{S_2\}$
- $\Sigma_2 = \{0, 1\}$
- $R_2 = \{S_2 \rightarrow \epsilon | 1S_20\}$

- $G = (V, \Sigma, R, S)$
- $V = \{S, S_1, S_2\}$
- $\Sigma = \{0, 1\}$
- $R = \{S \rightarrow S_1 | S_2, S_1 \rightarrow \epsilon | 0S_11, S_2 \rightarrow \epsilon | 1S_20\}$