CS304: Automata and Formal Languages

Lec 19

Pumping Lemma for CFLs

Rachit Nimavat

October 3, 2025

Outline

Defn: CFG

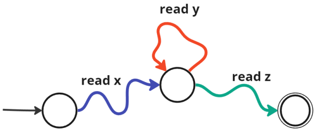
A CFG is a set of recursive rules used to generate patterns of strings. Think of it as a blueprint for a language.

Definition 1.1

A Context-Free Grammar is a 4-tuple $G = (V, T, P, S)$, where:

- **V**: Finite set of **variables**. think: syntactic categories or placeholders. (e.g., S, E, T in previous examples)
- **T**: Finite set of **terminals** or **tokens**. actual symbols or words of the language. (e.g., $\mathsf{a}, \mathsf{b}, \mathsf{c}, +, *$ in previous examples)
- **P**: Finite set of **production rules**. Rules have the form $A \to \alpha$, where $A \in V$ and $\alpha \in (V \cup T)^*$, i.e., α is a string consisting of variables and terminals. specifies how to replace variables with strings of variables and terminals. (e.g., $S \to \mathsf{a}S\mathsf{b}$ and $E \to T$) in previous examples
- **S**: The **start symbol** $(S \in V)$.

Pumping Lemma for Regular Languages



Lemma 1.2

For each regular language L, there is a constant p (<u>the pumping length</u>) such that any string $s \in L$ with $|s| \geq p$ can be divided into three parts, $s = xyz$, such that

i. $|y| > 0$ *("looped-string" y is not empty)*

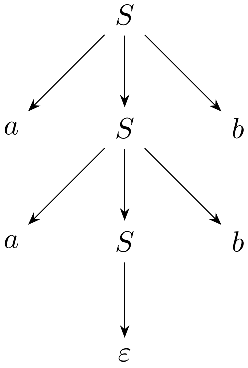
ii. $|xy| \leq p$ *(the loop starts within the first p characters)*

iii. $\forall i \geq 0$, *the string* $xy^iz \in L$ *(we can pump the loop zero, one, or many times, and the resulting string must still be accepted)*

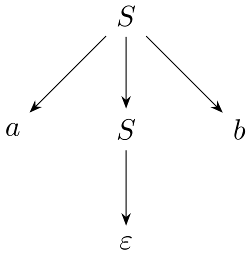
Visualizing CFG Derivations: Parse Trees

Similar to DFA, wer have **Parse Trees** to visualize derivation in CFGs.
Consider $L = \{a^n b^n \mid n \geq 0\}$ over $\Sigma = \{a, b\}$ and CFG $S \to \varepsilon \mid \mathrm{a}S\mathrm{b}$.

The parse tree for $aabb$:

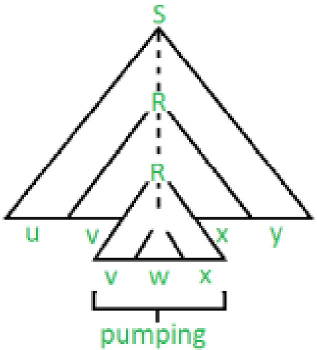


Parse tree for ab:



Parse tree for ε:

Pumping Lemma for CFLs



Lemma 1.3

For each CFL L, there is a constant p (the pumping length) such that any string $s \in L$ with $|s| \geq p$ can be divided into five parts, $s = uvwxy$, such that

i. $|vx| > 0$ *(at least 1 of v and x is not empty)*

ii. $|vwx| \leq p$ *(middle portion is not too long)*

iii. $\forall i \geq 0$, *the string $uv^iwx^iy \in L$ (we can pump v and x, one, or many times, and the resulting string is still in L)*

How to use Pumping Lemma?

Goal: Prove that a language L is not a CFL.
Assume that L is a CFL.

Pumping Lemma: Player-1

Us: Player-2

1. Provides 'pumping length' p
2. Cleverly choose $s \in L$ such that $|s| \geq p$
3. Provides a partition $s = uvwxy$ such that $|vx| > 0$ and $|vwx| < p$
4. WIN by choosing $i \geq 0$ such that $uv^iwx^iy \notin L$

If $uv^iwx^iy \notin L$, then we have a contradiction. Thus, our assumption that L is context-free must be false. Therefore, L is not a CFL.

Note: For this to work, we must have a winning strategy for each p and for each partition $s = uvwxy$ (that satisfies $|vx| > 0$ and $|vwx| < p$).

How to use the Pumping Lemma?

Prove that $L = \{0^n 1^n 2^n \mid n \geq 0\}$ is not context-free

Assume for contradiction that L is context-free

From pumping lemma, there is a constant p with "looping property"

Let's pick a string $s = 0^p 1^p 2^p$. Notice that $s \in L$

Consider its decomposition $s = uvwxy$ given by the pumping lemma

$0 < |vx| \leq |vwx| \leq p$ implies at least one of $\{0, 1, 2\}$ is not in vwx (since vwx is of length at most p, it cannot cover all three segments of s)

Choose $i = 0$. From pumping lemma, $s_0 = uv^0wx^0y = uwy$, which has unequal number of 0s, 1s and 2s.

Contradiction!

How to use the Pumping Lemma?

Prove that $L = \{0^i 1^j 2^k \mid 0 \leq i \leq j \leq k\}$ is not context-free

Assume for contradiction that L is context-free

From pumping lemma, there is a constant p with "looping property"

Let's pick a string $s = 0^p 1^p 2^p$. Notice that $s \in L$

Consider its decomposition $s = uvwxy$ given by the pumping lemma

$0 < |vx| \leq |vwx| \leq p$ implies at least one of $\{0, 1, 2\}$ is not in vwx (since vwx is of length at most p, it cannot cover all three segments of s)

We need to do case-analysis based on which symbols are in vwx:

Case 1: vwx do not contain any 0. Pump down!
Case 2: vwx do not contain any 1 but there are 0s. Pump up!
Case 3: vwx do not contain any 1 but there are 2s. Pump down!
Case 4: vwx do not contain any 2. Pump up

In each case, we get a contradiction!

How to use the Pumping Lemma?

HW: Prove that $L = \left\{ww \mid w \in \{0,1\}^*\right\}$ is not context-free

Closure Properties

Closure Properties: A set is closed under an operation if applying that operation to its elements results in elements that are also in the set.

- CFLs are closed under union (create a new start symbol with $S \mapsto S_1 \mid S_2$)
- CFLs are closed under concatenation (create a new start symbol with $S \mapsto S_1 S_2$)
- If L is a CFL, so is L^* (add rule $S \mapsto SS \mid \varepsilon$)
- Reversal of a CFL is context-free Reverse all production rules
- CFLs are closed under intersection **NO!** (Use $L_1 = \{a^i b^i c^j\}$ and $L_2 = \{a^j b^i c^i\}$)
- CFLs are closed under complementation **NO!** (Use DeMorgan's law)