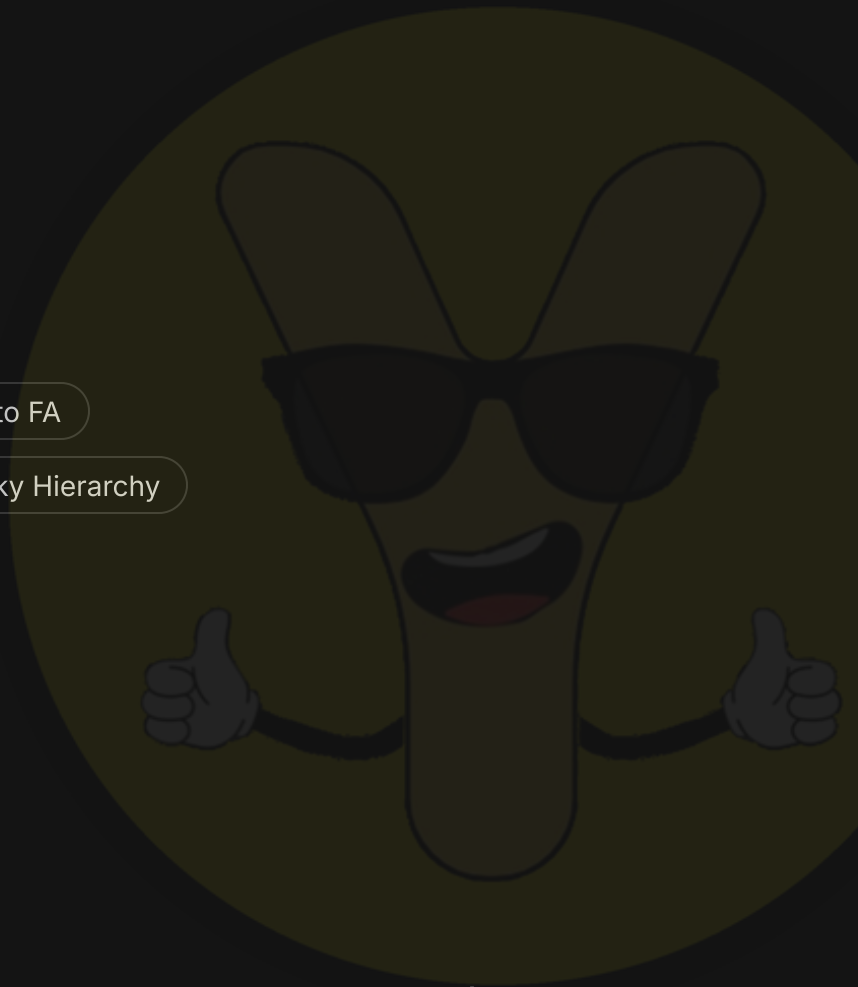
AKTU · 2ND YEAR · SEMESTER 4 · BCS402

Theory of Automata and Formal Languages

Unit 3 — Regular and Non-Regular Grammars

- Context Free Grammar(CFG)-Definition, Derivations, Languages, Derivation Trees and Ambiguity
- Regular Grammars-Right Linear and Left Linear grammars
- Conversion of FA into CFG and Regular grammar into FA
- Simplification of CFG
- Normal Forms- Chomsky Normal Form(CNF), Greibach Normal Form (GNF)
- Chomsky Hierarchy
- Programming problems based on the properties of CFGs



DECODED BY

Chirag Jain

Founder, Ycotes · Gold Medalist in Artificial Intelligence, IIT Madras · taught 10,000+ students

Context-Free Grammar (CFG)

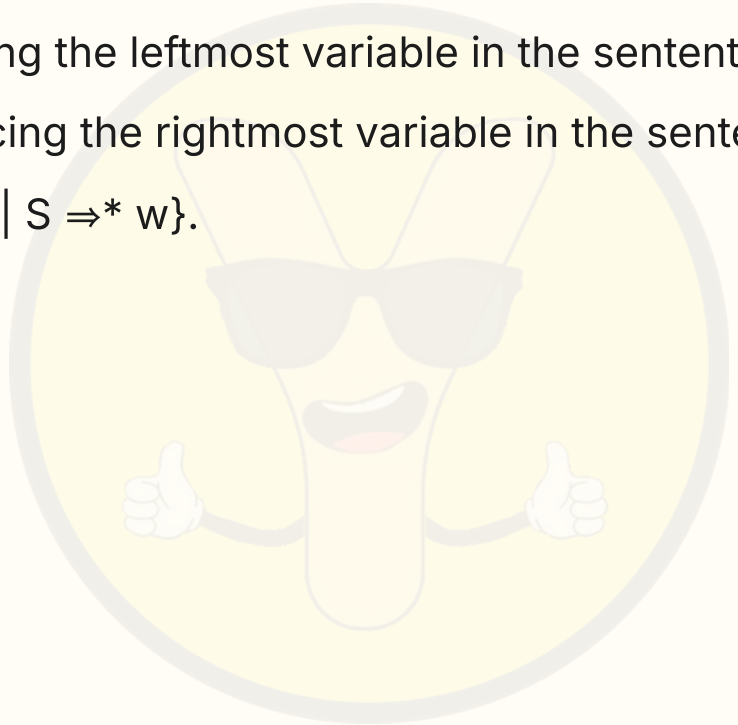
CFG DEFINITION

A CFG is a 4-tuple $G = (V, \Sigma, R, S)$ where V is a finite set of variables, Σ is a set of terminals, R is a set of production rules, and S is the start symbol.

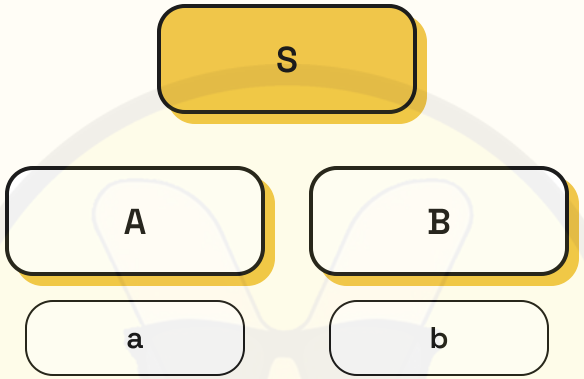
- Production rules are of the form $A \to \alpha$, where $A \in V$ and $\alpha \in (V \cup \Sigma)^*$.
- CFGs are used to define the syntax of programming languages.
- They are more powerful than Regular Grammars.

Derivations and Languages

- Derivation: A sequence of rule applications to transform the start symbol into a string of terminals.
- Leftmost Derivation: Always replacing the leftmost variable in the sentential form.
- Rightmost Derivation: Always replacing the rightmost variable in the sentential form.
- Language of a CFG: $L(G) = \{w \in \Sigma^* \mid S \Rightarrow^* w\}$.



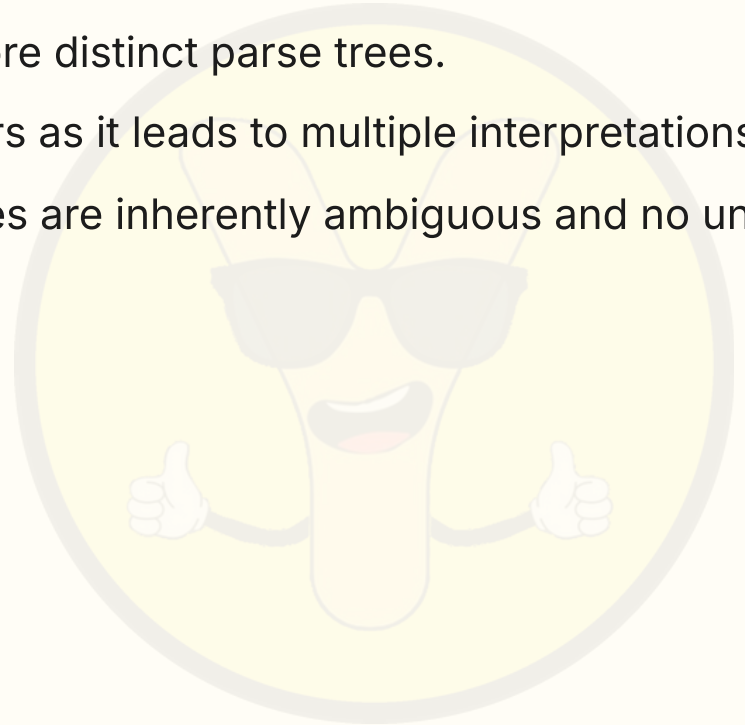
Derivation Tree (Parse Tree)



A graphical representation of the derivation process.

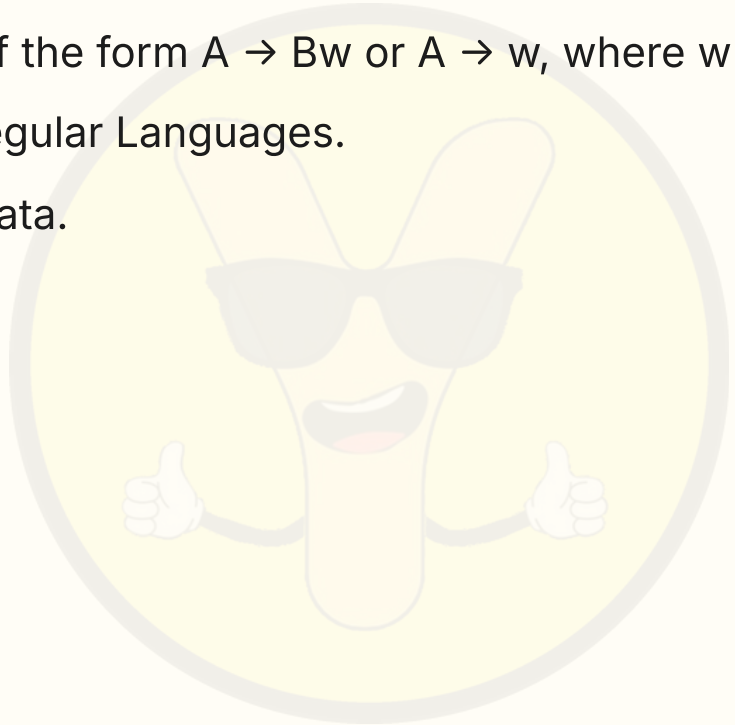
Ambiguity in CFG

- A CFG is ambiguous if there exists at least one string in L(G) that has more than one leftmost derivation.
- Equivalently, a string has two or more distinct parse trees.
- Ambiguity is a problem for compilers as it leads to multiple interpretations of code.
- Inherent Ambiguity: Some languages are inherently ambiguous and no unambiguous grammar exists for them.



Regular Grammars

- Right Linear Grammar: All rules are of the form $A \to wB$ or $A \to w$, where $w \in \Sigma^*$ and $A, B \in V$.
- Left Linear Grammar: All rules are of the form $A \to Bw$ or $A \to w$, where $w \in \Sigma^*$ and $A, B \in V$.
- Regular grammars generate only Regular Languages.
- They are equivalent to Finite Automata.

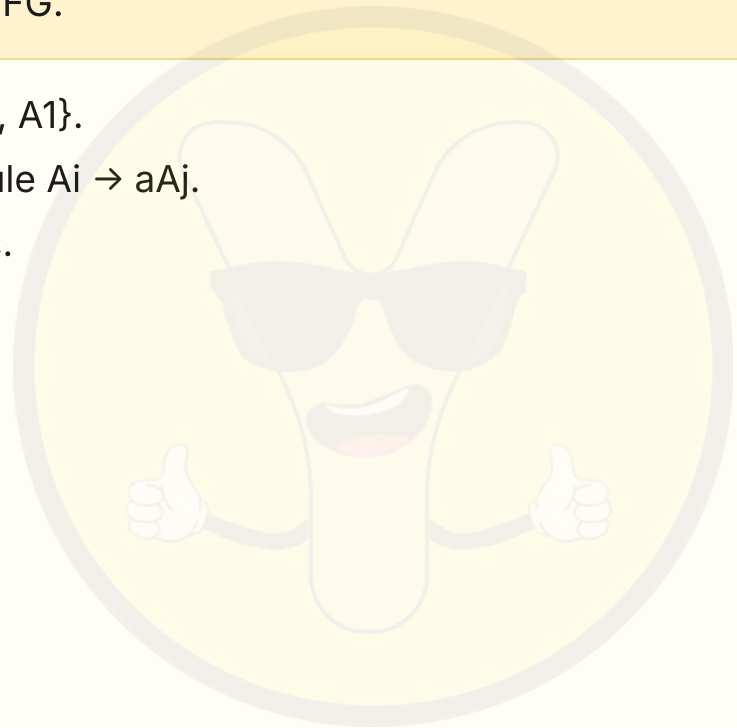


Conversion: FA to CFG

Convert an FA with states {q0, q1} to a CFG.

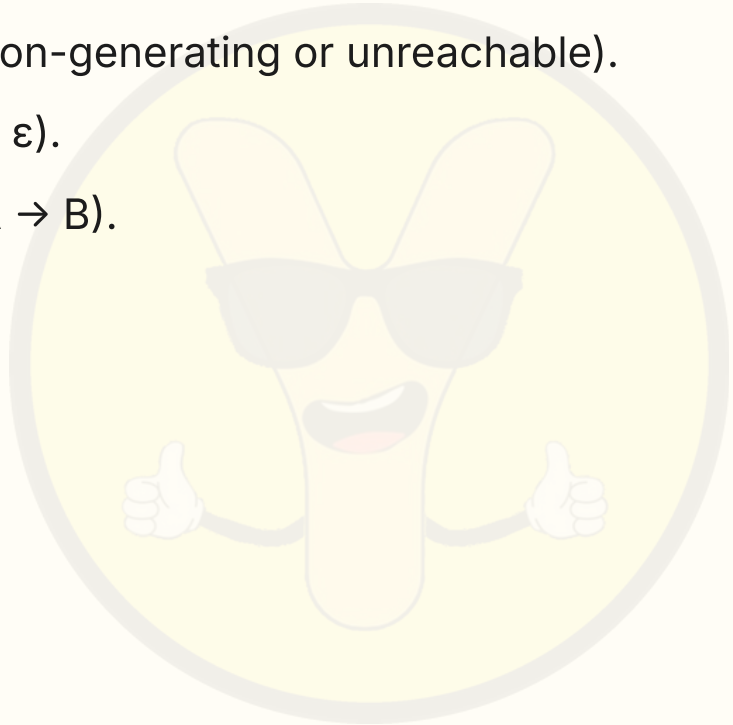
1. Create a variable for each state: V = {A0, A1}.
2. For each transition δ(qi, a) = qj, create rule Ai → aAj.
3. For each final state qf, create rule Af → ε.
4. Start symbol is the initial state variable.

Rules: A0 → aA1, A1 → bA0, A1 → ε



Simplification of CFG

- Goal: Remove redundant symbols and rules to make the grammar efficient.
- Step 1: Remove Useless Symbols (non-generating or unreachable).
- Step 2: Remove ε-productions (A → ε).
- Step 3: Remove Unit Productions (A → B).

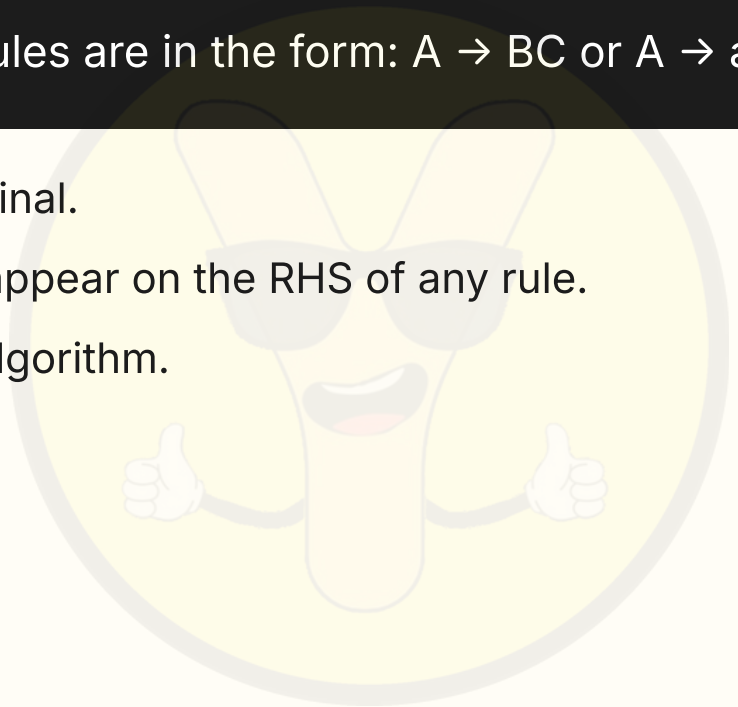


Chomsky Normal Form (CNF)

CNF DEFINITION

A CFG is in CNF if all production rules are in the form: A → BC or A → a.

- A, B, C are variables and a is a terminal.
- S → ε is allowed only if S does not appear on the RHS of any rule.
- CNF is useful for the CYK parsing algorithm.

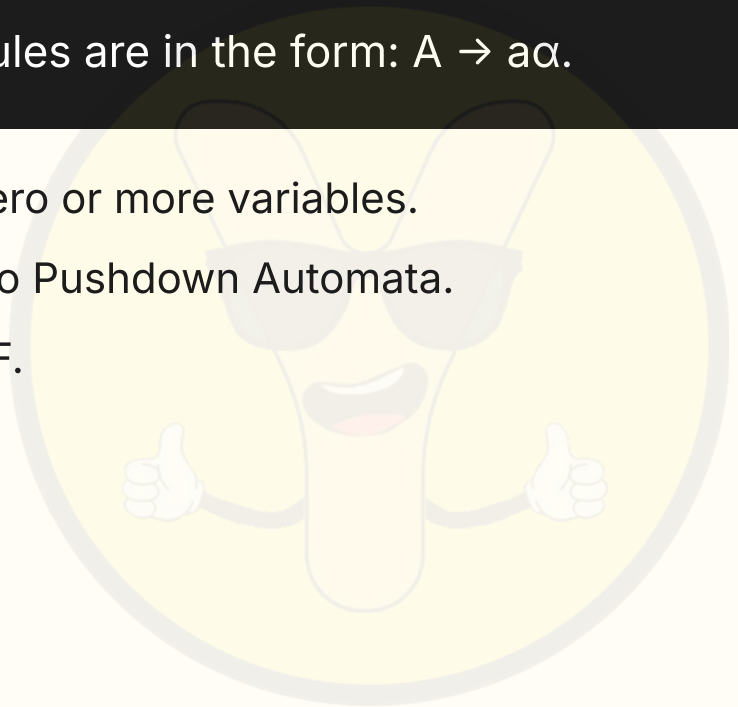


Greibach Normal Form (GNF)

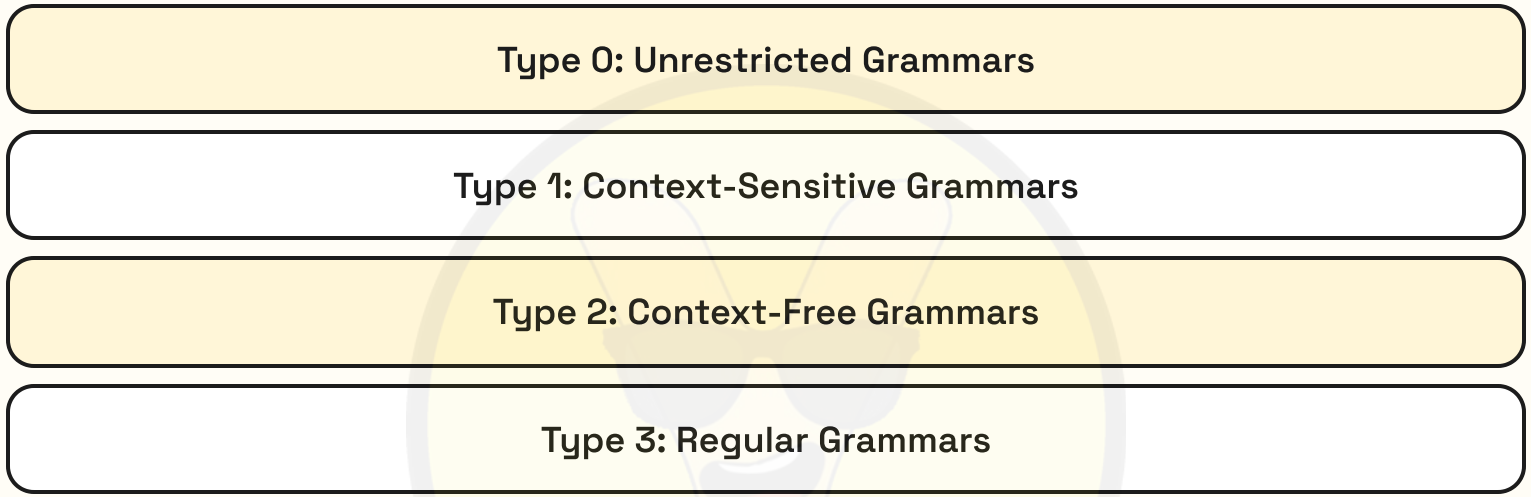
GNF DEFINITION

A CFG is in GNF if all production rules are in the form: A → aα.

- a is a terminal and α is a string of zero or more variables.
- GNF is useful for converting CFGs to Pushdown Automata.
- Every CFG can be converted to GNF.



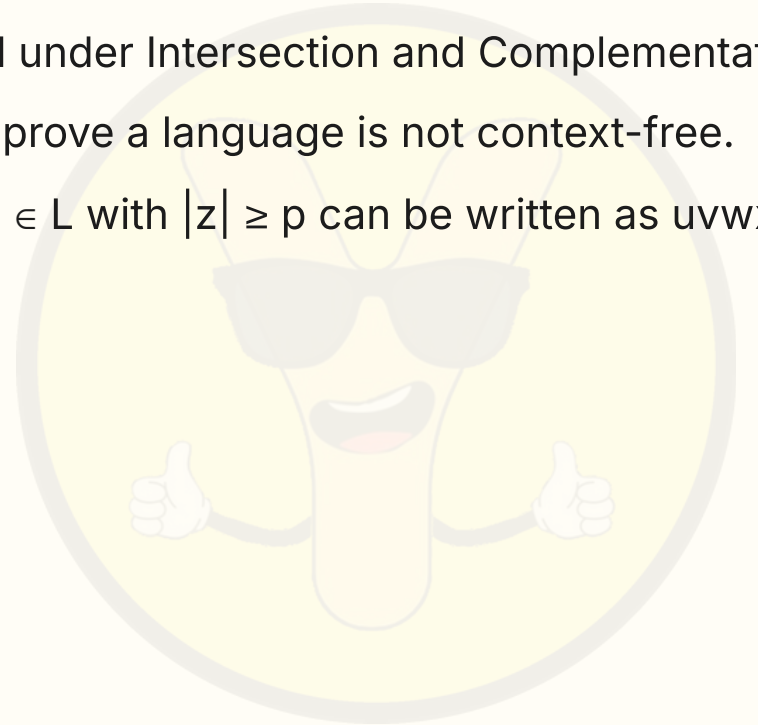
Chomsky Hierarchy



Hierarchy showing containment of language classes.

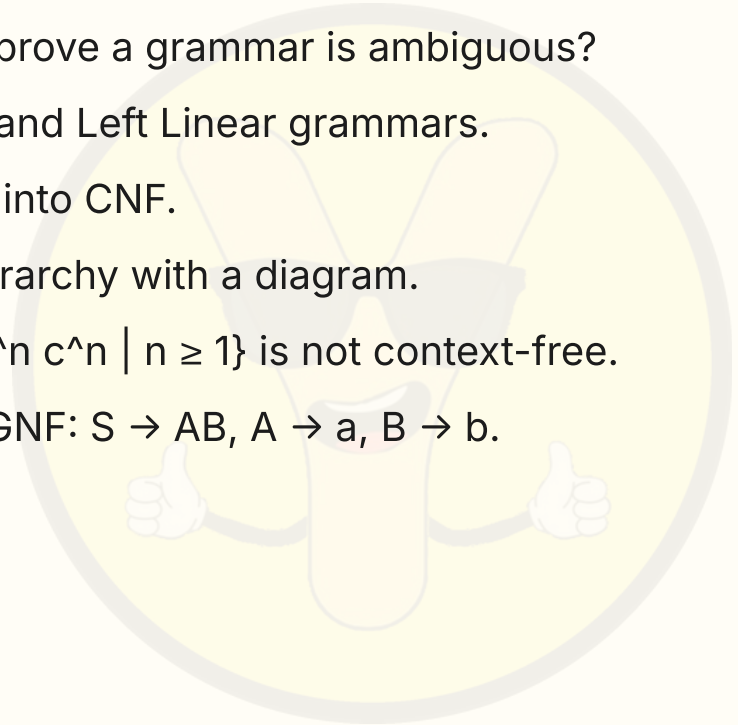
Properties of CFGs

- Closure Properties: CFGs are closed under Union, Concatenation, and Kleene Closure.
- Non-Closure: CFGs are NOT closed under Intersection and Complementation.
- Pumping Lemma for CFLs: Used to prove a language is not context-free.
- Condition: If L is a CFL, any string $z \in L$ with $|z| \geq p$ can be written as uvwxy.



Exam corner

- **Q1** Define CFG and explain its components with an example.
- **Q2** What is ambiguity in CFG? How to prove a grammar is ambiguous?
- **Q3** Differentiate between Right Linear and Left Linear grammars.
- **Q4** Explain the steps to convert a CFG into CNF.
- **Q5** State and explain the Chomsky Hierarchy with a diagram.
- **Q6** Prove that the language L = {a^n b^n c^n | n ≥ 1} is not context-free.
- **Q7** Convert the following grammar to GNF: S → AB, A → a, B → b.



Unit 3 done. You are exam-ready.

Revise the Exam corner once more, then open Unit 4 on theycotes.com. Every AKTU subject, unit by unit, free. Notes by Chirag Jain, Ycotes.

