Parsing can be optimized by setting an upper bound on the length of the sentential forms being derived. This can be done by removing the unit and empty productions.

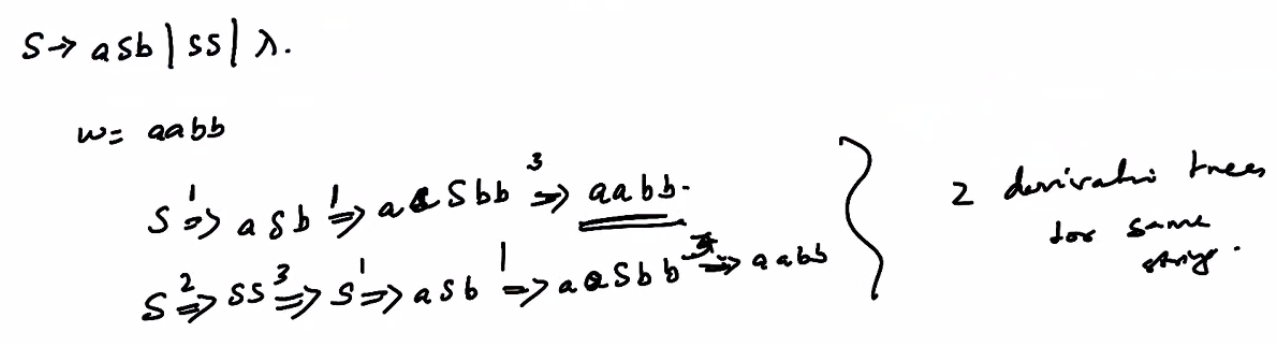
Ambiguity

There should not be 2 or more ways to derive the same expression. If there are 2 or more left-most derivations that can be constructed for a given string, then the grammar is said to be ambiguous.

A CFG G is said to be ambiguous if there exists some w belonging to L(G) that has atleast 2 distinct derivation trees.

Ambiguous grammars are not fit to be used in programming languages.

Eg



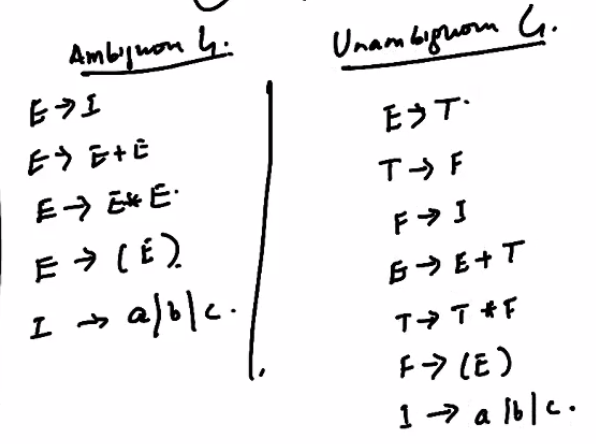
Here both the derivations are left-most derivations, so the grammar is ambiguous.

Disambiguation of grammars

If there are multiple derivation trees for the same input string, there are multiple ways to interpret the string. This can be removed by:

1. Adding new variable
2. Introducing precedence rules of operations

Eg of removing ambiguity from a given grammar:



Here we introduce new variables T and F. Now we create productions for these new variables in such a way that we create the precedence rules we need. This is because for a grammar to be ambiguous we need a left-most or a right-most derivation, i.e. we go from left to right or right to left order. This happens because the higher precedence operator will go further to the right of the production rule. By introducing new production rules for the variables, we can create an order in which these production rules will be applied for creating the left-most or right-most derivations.

Now there will be no ambiguity, as there will exist only 1 way to create a left-most derivation.

If a CFL L has a CFG that is unambiguous, the language L is said to be unambiguous.

There may be multiple CFGs representing the language L. If none of the CFGs representing L are unambiguous (all are ambiguous), then the language L is said to be inherently ambiguous.

Inherent ambiguity (due to the language itself) cannot be removed. If even any one of the grammars is unambiguous, we can convert the rest of the grammars to be unambiguous, and so the ambiguity caused by grammars may be removed.

Standard forms for grammar

Chomsky Normal Form (CNF)

Greibach Normal Form (GNF)