Software Specifications
Ambiguity

Cain Susko

Queen's University
School of Computing

February 7, 2022

Ambiguity

given the example:

$$\diamond \rightarrow \diamond + \diamond \mid \diamond \times \diamond \mid (\diamond) \mid a.$$

Where this grammar's terminal string is:

$$a + a \times a.$$

Upon examination of the string one may find that the string as 2 valid parse trees:

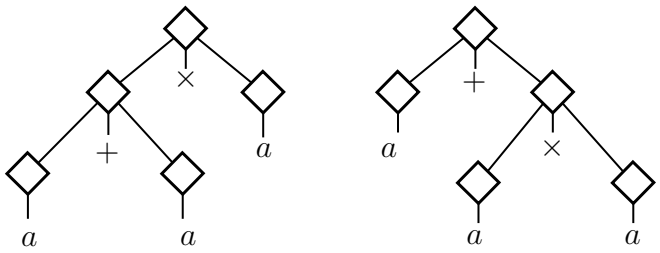


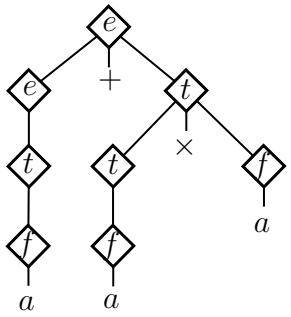
Figure 1: the two parse trees of $a + a \times a$

Thus this means that this terminal string is **ambiguous**. However, the tree on the right is the *desired* parse tree as multiplication has higher precedence than addition (we read the trees from the bottom up).

We can then use an 'ad-hoc' process to transform this grammar example into an equivalent *unambiguous* grammar. This new grammar is as follows. Note that $\diamond = expression$

$$\begin{aligned} expression &\rightarrow expressiom + term \mid term \\ term &\rightarrow term \times factor \mid factor \\ factor &\rightarrow (expression) \mid a. \end{aligned}$$

Thus the terminal string $a + a \times a$ in this grammar is unambiguous:



Thus this terminal string has a unique parse tree is unambiguous. To Prove a Grammar is unambiguous one must first show that all terminal strings within the grammar are unambiguous.