Context-Free Grammars

This exercise is concerned with context-free grammars (CFGs). Please read Section 7.4 in the tutorial which explains how to model CFGs as inductive definitions. Our particular example is about defining valid sequences of parentheses.

Two grammars

The most natural definition of valid sequences of parentheses is this:

$$S \quad \rightarrow \quad \varepsilon \quad | \quad '('\ S\ ')' \quad | \quad S\ S$$

where ε is the empty word.

A second, somewhat unusual grammar is the following one:

$$T \quad \rightarrow \quad \varepsilon \quad | \quad T\ '('\ T\ ')'$$

Model both grammars as inductive sets S and T and prove $S = T$.

A recursive function

Instead of a grammar, we can also define valid sequences of parentheses via a test function: traverse the word from left to right while counting how many closing parentheses are still needed. If the counter is 0 at the end, the sequence is valid.

Define this recursive function and prove that a word is in S iff it is accepted by your function. The $\Longrightarrow$ direction is easy, the other direction more complicated.