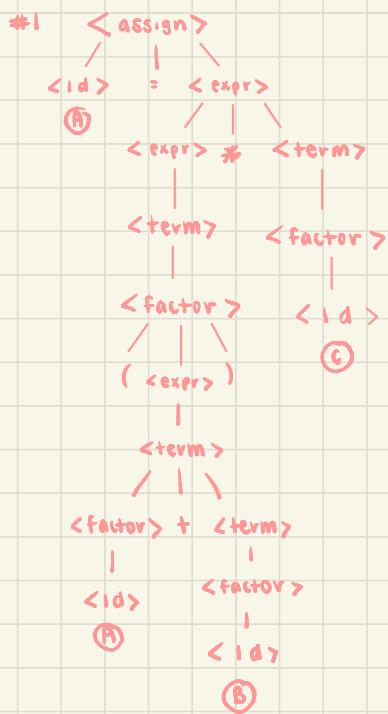


## CS 381 - Homework 3 Syntax

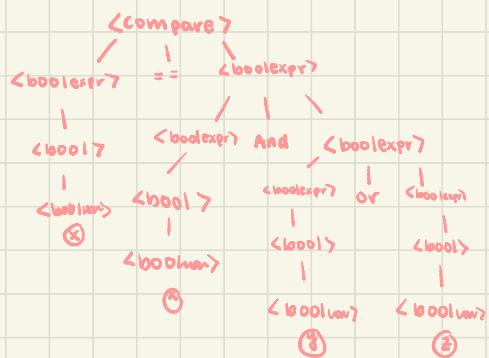
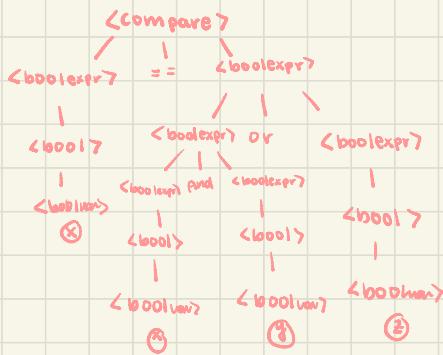


$\text{<assign>} \rightarrow \text{<id>} = \text{<expr>}$   
 $\rightarrow A = \text{<expr>}$   
 $\rightarrow A = \text{<expr>} * \text{<term>}$   
 $\rightarrow A = \text{<term>} * \text{<term>}$   
 $\rightarrow A = \text{<factor>} * \text{<term>}$   
 $\rightarrow A = (\text{<expr>}) * \text{<term>}$   
 $\rightarrow A = (\text{<expr>}) * \text{<term>}$   
 $\rightarrow A = (\text{<factor>} + \text{<term>}) * \text{<term>}$   
 $\rightarrow A = (A + \text{<term>}) * \text{<term>}$   
 $\rightarrow A = (A + \text{<factor>}) * \text{<term>}$   
 $\rightarrow A = (A + \text{<id>}) * \text{<term>}$   
 $\rightarrow A = (A + B) * \text{<term>}$   
 $\rightarrow A = (A + B) * \text{<factor>}$   
 $\rightarrow A = (A + B) * \text{<id>}$   
 $\boxed{\rightarrow A = (A + B) * C}$

#2

$\text{<assign>} \rightarrow \text{<id>} = \text{<expr>}$   
 $\text{<expr>} \rightarrow \text{<expr>} * \text{<term>}$   
 $| \text{<term>}$   
 $\text{<term>} \rightarrow \text{<factor>} + \text{<term>}$   
 $| \text{<factor>}$   
 $\text{<factor>} \rightarrow (\text{<expr>})$   
 $| \text{<id>}$   
 $| \text{<id>} ++$   
 $| \text{<id>} --$   
 $\text{<id>} \rightarrow A$   
 $| B$   
 $| C$

井3



The grammar is ambiguous because it can be defined in two ways

# 4 a grammar  $G_1$  for language  $L$  can be defined as follows:

↳ S → 1T | 01T

$L \leftarrow T \rightarrow OT | IT | E$

Here, S is the start symbol while T is the non-terminal symbol representing the binary of odd ints. greater than 4, 1 and 0 are terminals while  $\epsilon$  is an empty string. The parse tree for the string "101" would look like this:

↳  $S \wedge T \wedge \neg T \wedge \neg \pi \wedge \neg \tau$

For " $\{10\}$ ":

↳  $S \wedge IT \wedge IT \wedge OT \wedge IE$

However it is important to note that the string "11" would not be in L because it represents the binary value for 3 so it is odd int. That's less than 4 and does not fit :-