Problem 1: With the grammar show the parse tree and the leftmost derivation

A = B *(C * (A + B))

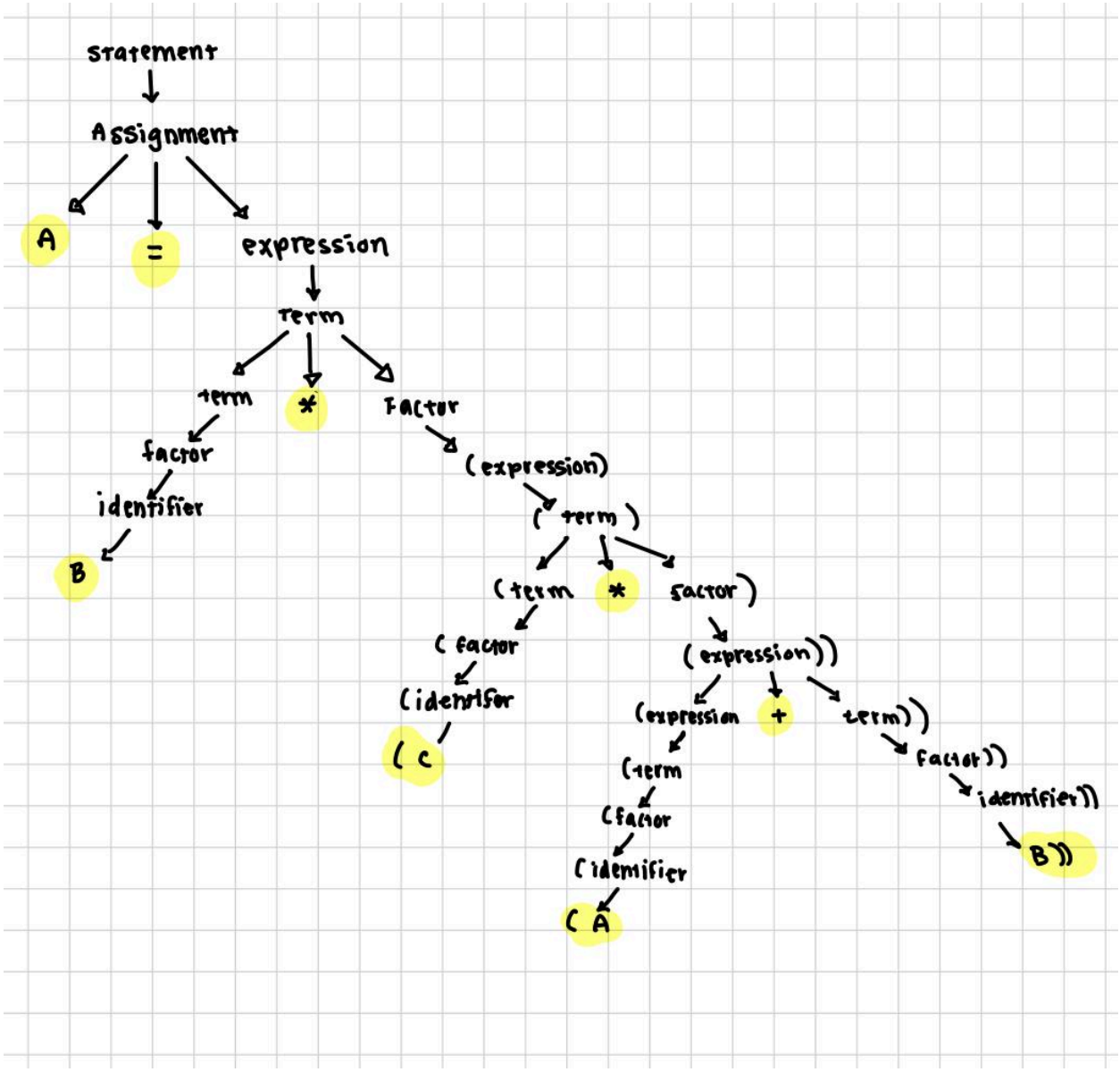
Grammar:

Statement → Assignment | Expression
Assignment → Identifier = Expression
Expression → Expression + Term | Term
Term → Term * Factor | Factor
Factor → (Expression) | Identifier
Identifier → A | B | C

Answer:

Statement → Assignment
Assignment → Identifier = Expression
Assignment→ A = Expression
Assignment→ A = Term
Assignment→ A = Term * Factor
Assignment→ A = Factor * Factor
Assignment→ A = Identifier * Factor
Assignment→ A = B * Factor
Assignment→ A = B * (Expression)
Assignment→ A = B * (Term)
Assignment→ A = B * (Term * Factor)
Assignment→ A = B * (Factor * Factor)
Assignment→ A = B * (Identifier * Factor)
Assignment→ A = B * (C * Factor)
Assignment→ A = B * (C * (Expression))
Assignment→ A = B * (C * (Expression + Term))
Assignment→ A = B * (C * (Term + Term))
Assignment→ A = B * (C * (Factor + Term))
Assignment→ A = B * (C * (Identifier + Term))
Assignment→ A = B * (C * (A + Term))
Assignment→ A = B * (C * (A + Factor))
Assignment→ A = B * (C * (A + Identifier))
Assignment→ A = B * (C * (A + B))

Problem 1 Parse Tree:



Problem 2: Consider the following pseudo code

a. Using static scope what is written on the screen?

add(v): u=u+v+z -> u=42+69+69=180
bar(add): u=w=17
foo(u,13): u=42 => v=42
Main: u=42, v=69, w=17

180 is printed to the screen

b. Using Dynamic Scope with Deep binding what is printed on the screen?

add(v): u+v+z => 42+42+42=126
bar(add): v=42 u=42
foo(u,13): u=42 => v=42 w=13
Main: u=42, v=69, w=17

126 is printed to the screen

c. Using Dynamic Scope with Shallow Binding

add(v): u+v+z= 42+ 13+ 42= 97
bar(add): v=42 u=13
foo(u,13): u=42 => v=42 w=17
Main: u=42, v=69, w=17

97 is printed to the screen