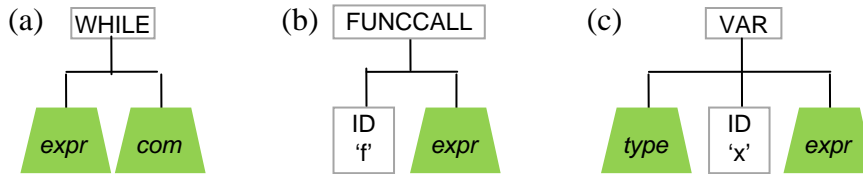# Exercises 7 (Contextual analysis)

7A. *(Type checking)*

The course notes (slide 7-11) outline type checking on some Fun AST fragments. In similar style, outline type checking on the following AST fragments:

(a)   (b)   (c)

7B. *(Symbol tables)*

The structure of a compiler's symbol tables (e.g., the type table) depends on the source language's block structure. What can you say about the structure of a symbol table in a compiler for each of the following source languages?

(a) In early versions of Cobol, all variables were global.

(b) In early versions of Fortran, all variables were either global or local. A local variable could have the same identifier as a global variable (but then the local variable would "hide" the global variable).

(c) In C, the program consists of declarations of global variables and functions. A function body is a block ("{ … }"). Any block may contain nested blocks. Any block may contain declarations of local variables. A local variable can have the same identifier as a global variable or even a variable declared in an enclosing block (but then the local variable would "hide" the other variable).