Due: beginning of class, Thursday, 3/1

1. [20] Use the following BNF grammar to

   (a) develop a leftmost derivation for the assignment, \( X = A \ast (B + C) \).
   (b) develop a rightmost derivation for the assignment in (a).
   (c) draw a parse tree for the derivation in (a).
   (d) draw an abstract syntax tree for (c).

\[
assign \rightarrow id = expr \\
id \rightarrow A | B | C | \ldots | Y | Z \\
expr \rightarrow \ expr \ + \ term \ | \ term \\
term \rightarrow \ term \ \ast \ factor \ | \ factor \\
factor \rightarrow \ ( \ expr \ ) \ | \ id
\]

2. [22] For each of the lines labeled A, B, C, and D below, list the label followed by each variable (in scope at that point) and its value as an ordered pair (e.g., C: \(<x, 0> <y, 0>\) in the order that the program is executed. Assume all variables are passed by value.

```c
int a = 10;

int f (int y) {
    int x = 0;
    int z = 3;
    // A
    z -= 1;
    a -= 1;
    return a;
}

void main () {
    int x = 2;
    int y = 3;
    int z = 4;
    int a = 5;
    // B
    y = f(x);
    // C
    a += 1;
    x = f(z);
    // D
}
```
3. [18] Show each of the following infix expressions in (a) Polish Prefix, (b) Polish Postfix, and (c) Cambridge Prefix notation, assuming standard mathematical precedence and associativity:

a. \( a \times b + c / d \times 3 \)
b. \( (a + b + c) \times c - d \times (c - 2) \)
c. \( a - b \times (c + (d - 1) / (e + 7)) \times 2 \)

4. [8] Consider the following statement in C:

\[
c = a ! = 0 \& \& b / a > -1 ? ++b * 2 : a--; \]

a. What will be the values of a, b, and c after evaluating this statement if a=0 and b=3?
b. What will be the values of a, b, and c after evaluating this statement if a=-2 and b=-6?

5. [20] Consider the following pseudocode, assuming nested subroutines and static scope:

```
procedure main
  g : integer

  procedure B(a : integer)
    x : integer

    procedure A(n : integer)
      g := n

    procedure R(m : integer)
      write_integer(x)
      x /= 2 -- integer division
      if x > 1
        R(m + 1)
      else
        A(m)

    -- body of B
    x := a \times a
    R(1)

    -- body of main
    B(3)
    write_integer(g)
```

a. What does this program print?
b. Show the frames on the stack when A has just been called. For each frame, show the static and dynamic links.
c. Explain how A finds the declaration of variable g.
6. [12] Consider the following pseudocode:

```plaintext
x : integer       -- global

procedure set_x(n : integer)
  x := n

procedure print_x
  write_integer(x)

procedure first
  set_x(1)
  print_x

procedure second
  x : integer
  set_x(2)
  print_x

set_x(0)
first()
print_x
second()
print_x
```

a. What does this program print if static scoping is used?
b. What does this program print if dynamic scoping is used?