CS 381 Homework 6 – Scope and Parameters

Please Note:

- You may work in groups of up to three people on this assignment. You will need to sign-up for a HW6_Group in Canvas at least 24 hours before the due date of the assignment.
- Submit your solution as a plain ASCII txt file.

Problem 1. Runtime Stack

Consider the following block. Assume static scoping and call-by-value parameter passing.

```
1   { int x;
2     int z;
3     z := 1;
4     { int f(int x) {
5         if x=0 then {
6            z := 1 }
7         else  {
8            z := f(x-1)*z+1 };
9         return z;
10      };
11      x := f(2);
12    };
13  }
```

Demonstrate the computations that take place during the evaluation of this block, that is, give a sequence of lines each showing the complete runtime stack with all activation records after each statement or function call. For recursive calls use one stack onto which a new activation record is pushed on for each recursive function call.

Problem 2. Static and Dynamic Scope

Consider the following block. Assume call-by-value parameter passing.

```
1   { int x;
2     int y;
3     int z;
4     x := 3;
5     y := 7;
6     { int f(int y) { return x*y };
7       int y;
8       y := 11;
9       { int g(int x) { return f(y) };
10        { int y;
11          y := 13;
12          z := g(2);
13        };
14      };
15    };
16  }
```

a) Draw the runtime stack after each line executes under **static scoping**. What value assigned to z in line 12?

b) Draw the runtime stack after each line executes under **dynamic scoping**. What value assigned to z in line 12?

Problem 3. Parameter Passing

Consider the following block. Assume dynamic scoping.

```
1   { int y;
2     int z;
3     y := 7;
4     { int f(int a) {
5         y := a+1;
6         return (y+a)
7       };
8       int g(int x) {
9         y := f(x+1)+1;
10        z := f(x-y+3);
11        return (z+1)
12      }
13      z := g(y*2);
14    };
15  }
```

a) Draw the runtime stack after each line executes given that both parameters **a** and **x** are passed using **Call-by-Name**. What are the values of **y** and **z** after line 13 executes?

b) Draw the runtime stack after each line executes given that both parameters **a** and **x** are passed using **Call-by-Need**. What are the values of **y** and **z** after line 13 executes?