
CS 381, Spring 2018

Homework 4
(Runtime Stack, Scoping, Parameter Passing)

Please note:

1. Hand in a *printed* copy of your solution before class on May, 24, or make otherwise sure that the TAs receive the copy before the deadline. (**Note: No online submission this time.**)
2. Submit *only one* solution per team. (Each team can have up to 5 members.) Make sure to list the names of all team members.

Late submissions will **not** be accepted. Do **not** send solutions by email.

Exercise 1. Runtime Stack

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

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

Illustrate the computations that take place during the evaluation of this block, that is, draw a sequence of pictures each showing the complete runtime stack with all activation records after each statement or function call.

Note. Do **not** use the alternative model of “temporary stack evaluation” that was briefly illustrated on slides 20 and 25 to explain the implementation given in `FunStatScope.hs` and `FunRec.hs`. Rather use one stack onto which a new activation record is pushed on each recursive function call.

Exercise 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) Which value will be assigned to *z* in line 12 under static scoping?
- (b) Which value will be assigned to *z* in line 12 under dynamic scoping?

It might be instructive to draw the runtime stack for different times of the execution, but it is not strictly required.

Exercise 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 }
```

What are the values of *y* and *z* at the end of the above block under the assumption that both parameters *a* and *x* are passed:

- (a) Call-by-Name
- (b) Call-by Need

It might be instructive to draw the runtime stack for different times of the execution, but it is not strictly required.