

Important Reminders!

1. Upload your solution as a single ASCII text file (ending in .txt) to Canvas. (*Note:* No pdf; don't use Pensieve.)
2. You must do all homework assignments by yourself, without the help of others. Also, you must not use services such as Chegg or Course Hero. If you need help, simply ask on Canvas, and we will help!
3. You can work in teams of *up to four* students to create and submit a common homework solution. To this end, first create an alphabetically sorted (by last name) list of the team members, and add this information at the beginning of the submitted text file, like so.

GROUP: Helena Eage, Dylan George, Mark Scout

The other group members **must not** submit a solution. All group members will receive the same grade.

Exercise 1. Runtime Stack

Consider the following block.

```
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, create a sequence of stacks, each showing the complete runtime stack with all activation records after each statement or function call. You should use the simplified model of runtime stacks where each activation record contains just a single binding. Function results can be represented as bindings “res=v” on top of the stack, right before the activation record is removed.

Moreover, to better keep track of recursive function calls, it might be useful to indent stacks, for example, like so:

```

...
4  [f={}, y=1, ...]
11 >>
    5  [x=..., f={}, ...]
    8  >>
        5  [x=..., x=..., f={}, ...]
        ...
        <<
        8  [x=..., ...]
        9  [res=..., x=..., ...]
        <<
    8  [x=..., ...]
    9  [res=..., ...]
    <<
11 [ ... ]

```

Exercise 2. Static and Dynamic Scope

Consider the following block.

```

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 }

```

- Which value will be assigned to `z` in line 12 under static scoping?
- 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

What are the values of `y` and `z` at the end of the following block under the assumption that both parameters `x` are passed:
 (i) call by value, (ii) call by reference, and (iii) call by value-result.

```
1  { int y;  
2    int z;  
3    y := 7;  
4    { int f(int x) {  
5        x := x+1;  
6        y := x;  
7        x := x+1;  
8        return y  
9    };  
10   int g(int x) {  
11       y := f(x)+1;  
12       x := f(y)+3;  
13       return x  
14   };  
15   z := g(y)  
16 }  
17 }
```