

# Advanced Programming

## Homework Assignment 6

Due Wednesday, February 26, at 3PM

### Preminimaries

1. Read chapter 10 of Bird (photocopied handout). It is safe to skim or skip the parts about “monad laws.”
2. Grab the file `lec9.hs` from the course web site (it is called Handout 9) and rename it `YourName(s)6.hs`.
3. If you have chosen to work with a partner on this assignment, you should do the whole thing “shoulder to shoulder,” both of you working at the same screen at the same time. If you’re working alone, feel free to talk to others in the class about the basic concepts, but write your code by yourself.

### Main assignment

1. Do exercises 10.2.1 and 10.2.3 from Bird. Push these changes through all of the evaluators in the provided code, including the basic evaluators, monadic evaluators, and monad transformers.
2. Do exercises 10.4.1 and 10.4.2 from Bird. Use the extended versions of `STT` and `EXC` from the previous exercises.
3. Build an evaluator that uses *all* of `EXC`, `OUT`, and `STT`.
4. *Optional:* Adapt your lexer and parser from last week’s homework to this week’s `Term` datatype and hook them up to your evaluator to give a complete interpreter for this small language.
5. *Optional:* We saw in last week’s homework how a list monad can be used to model computations with multiple possible results. Extend the `Term` datatype with a constructor `Flip`, whose evaluation yields *both* 0 and 1. Write a simple evaluator for this extended language that returns a list of possible results, then a monadic version of the same evaluator. Finally, can you package this functionality into a monad transformer `MULTI` that transforms an arbitrary monad into one returning multiple results and use it to build a monadic interpreter with exceptions, output, state, and multiple results?

### Submission instructions

- Submit your code in a file `YourName(s)6.hs`.  
Put your name(s) in a comment at the top of the file. Also, please put the approximate number of hours that you spent on this assignment. Give separate numbers for time spent reading and time spent programming.
- Your submission should define a module `Main` that includes an action `main` demonstrating all of your extensions to the original `lec9.hs`.
- Email the file to both `jschorr@seas.upenn.edu` and `bcpierce@cis.upenn.edu`.