

# CS 571 Programming Assignment 5 Instructions

**Due: Friday, April 26, 2024**

## 1 AST Manipulation

**10 points**

The file `ast.hs` contains a data structure `Expr` that encodes an abstract syntax tree for expressions. In the same file you should implement the functions `eval` that evaluates an input expression and `equals` that computes structural equality between two input expressions.

The file `ast.hs` also contains test cases for both of these functions.

## 2 List Manipulation

**10 Points**

In the file `listManip.hs`, implement the function `prefix` – which computes whether or not one list is a prefix of another – and the function `sublist` – which computes a list containing all sublists of the input list. Note that order is not important, so `sublist [1,2,3]` can output `[[], [1], [2], [3], [1, 2], [2, 3], [1, 3], [1, 2, 3]]` or `[[3, 2, 1], [2, 3], [1, 3], [2, 3], [3], [2], [1], []]`.

## 3 Stateful Data Structures

**15 Points**

In the file `bank.hs`, implement an interface for a bank account. Your interface should support the following operations:

- `deposit`, which takes in an input dollar amount and adds it to the bank account.
- `withdraw`, which takes in an input requested withdrawal amount and returns the actual amount. The `withdraw` operation should allow a bank account to be overdrawn up to \$100.
- `getBalance`, which returns the current balance on the account.
- `getOverdrawn`, which returns whether or not the account is overdrawn.

Your implementation should support composing operations with `do` notation, and you should define an auxiliary function `runBankOp` that runs a (sequence of) operations. Test cases are available in the file `bank.hs`.