

**Discrete Math & Functional Programming— CSCI 054— Spring 2024**

**Instructor: Osborn**

Homework 4 - groupwork (1 point(s))

Due: 10:00PM on Friday

Some things to keep in mind:

- The first question is meant in part to help you start reviewing for the checkpoint. These are evaluated based on effort+participation and not correctness, so the only thing you need to turn in for that first question is what you decided to work on. Please decide as a group how you want to use the hour that you have with your TA!
  - This week's problem set gives you the opportunity to practice using higher-order functions. Note that this is going to be our last purely-programming problem set, so enjoy it! (The rest of the assignments this semester will be either only written or a combination of written and coding.)
1. **[1 point(s)] Recap** The only thing that you need to turn in on Gradescope for this problem is what your group decided to work on.

(a) What are the types of the following functions?

```
f1 a b c = if a < b then c else False
f2 (a, b, c) = if a < b then c else False
f3 a b c = if (a b) < (a c) then c else False
```

(b) Write a function `exists :: (a -> Bool) -> [a] -> Bool` which takes a predicate and a list and returns `True` if and only if at least one element in the list satisfies the predicate. Depending on what you want to practice/review, you could try this problem in a number of ways. Can you implement using list recursion? filter? list comprehension? pattern matching? guards? if-then-else? folds? anonymous functions? etc? What makes the problem seem like a better fit for some of these than others?