

COS418 Precept 1 - Programming in Go

02/02/24

Welcome to the first COS418 precept in Spring 2024!

Because all the assignments in this class use Go, there will be an emphasis on programming in Go in the first few weeks of precepts. In general, topics covered in precepts will be centered around the assignments.

The best way to learn Go is to work through examples and exercises. The golang tour (<https://tour.golang.org/list>) is a great resource for this and will help you significantly with assignment 1.1 due next week on 2/9.

Getting started

Go to <https://play.golang.org/> and run the hello world example. This will be your code dumping ground for this precept. For assignment 1.1 and beyond, however, you should install Go on your laptop and set up a development environment. I personally use IntelliJ as my IDE but sublime, vim and emacs are also great. Goland is also a good IDE to try (you can use it for free as a student).

Exercises

Note: You don't have to do *all* of them! Just focus on a few that you can manage within the duration of the precept.

Easy

1. Print the first 10 squared numbers.
2. Print the first 10 fibonacci numbers.
3. Fizz-buzz: replace multiples of 3 with *Fizz* and multiples of 5 with *Buzz*, and multiples of both with *FizzBuzz*. Print the first 10 numbers in this sequence.
4. Write a function that reverses a slice.
5. Write a function that returns the number of unique items in a slice.

Medium

6. Implement a binary tree in which each node contains a number. Then write a function that sums all the numbers in the tree.
7. Write a function that launches n goroutines to square all entries in a slice in parallel, where n is provided by the caller. Your function should block until all goroutines terminate.

A little harder

8. Given an n by n matrix, print all the entries in spiral order. Now do it in both directions (clockwise and anti-clockwise).
9. Implement merge-sort using goroutines. If the size of the input slice is n , how many goroutines are launched in total?