CS41 Lab 8

October 26, 2020

The lab this week focuses on dynamic programming. The purpose of this lab is to gain practice using dynamic programming to solve problems. Do **not** expect to solve both problems. Consider lab a success if you solve one of the problems and make some progress on the other problem.

General Hints:

- Focus on the **choice** you might make to construct an optimal solution.
- In each problem below, you will need to keep track of some extra information. Try to be deliberate about what extra information you might need to keep track of, and why this extra information might be required to efficiently get the dynamic programming to work.
- Initially focus on the first two steps of the dynamic programming process. Don't stress about pseudocode until after you've solved all lab problems.

1. **Making Change With Strange Coins.** In Homework 6, you considered the problem of making change for n cents out of the fewest number of coins. You previously designed a greedy algorithm that gave optimal solutions for US coins, and for coins that are used in a country called Algorithmland. You also saw that the greedy algorithm (always use the largest denominations possible) will not work for all possible coin denominations.

   Design a dynamic programming algorithm that takes as input the amount of change to make, n, and a list of coin denominations, $c_1, \dots c_k$. If it is possible to make n cents in change using denominations $c_1, \dots, c_k$, your algorithm should output the smallest number of coins required. Otherwise, your algorithm should output `FAIL`.

2. **Subset Sum.** In this problem, you are given an integer *weight threshold* $W > 0$ and a list of n items $\{1, \dots, n\}$ each with nonnegative weight w_i. Your task is to output a subset of items $S \subseteq \{1, \dots, n\}$ such that $\sum_{i \in S} w_i$ is as large as possible, subject to $\sum_{i \in S} w_i \leq W$.

   Design an analyze a dynamic program to solve **Subset Sum**. Your algorithm should run in $O(nW)$ time.