Name: _____ Wisc id: _____

Dynamic Programming Proof Structure

1. **Subproblem definition**
   - Describe the meaning of subproblem in words. Subproblem(s) is often suffix, prefix, substring, combinations, etc...
   - Define all variables you use, and their valid ranges.
2. **Base cases**
   - State solutions for all independent subproblems where Bellman equations break down.
3. **Bellman equation**
   - Relate subproblem solutions recursively.
   - Make sure all cases are accounted for.
4. **Original problem**
   - State the order in which you solve the subproblems (e.g., increasing order of $j - i$).
   - Show how to compute solution to the original problem from solutions to subproblem(s).
5. **Time and space analysis**
   - Time of original problem = (# subproblems) $\cdot$ (time/problem)
   - Space of original problem = Size of solution matrix + Size of input encoding

Problems

Ascending the Iron Throne

King Robert Baratheon has set his eyes on the Iron Throne, an intimidating structure with n steep steps leading up to it. The king, being robust yet not as nimble as he once was, can ascend the throne by climbing either 1 or 2 steps at a time.

Maester Pycelle, ever the curious scholar, wonders in how many unique ways King Robert can ascend these n steps to claim his seat on the Iron Throne. Your task is to help him find the answer.

Valyrian Steel in the Valyrian peninsula

CS 6.006, Spring 2008, MIT. Dragon Steel, known for its effectiveness against White Walkers, has been discovered in various parts of the doomed Valyrian peninsula. The peninsula is mapped out as an $n \times n$ grid, and each cell in the grid contains a certain amount of Valyrian Steel shards. Daenerys Targaryen needs to collect as many Valyrian Steel shards as possible to arm her troops.

You start at the entrance of peninsula (the bottom-left corner of the grid) and need to make your way to Elyria (the top-right corner of the grid). At each step, you can move either one cell up or one cell to the right. Every time you step on a cell (i, j), you can collect all the Valyrian Steel shards $c(i, j)$ located there.

Help Daenerys by finding the path that allows you to collect the maximum amount of dragon glass. Describe an $O(n^2)$ algorithm to find the maximum number for Valyrian Steel shards you can collect.

The Royal Treasury of King's Landing

Queen Daenerys Targaryen has successfully claimed the Iron Throne and now sits in the Red Keep. To solidify her reign, she plans to build several charitable institutions across King's Landing. Each project has a cost that needs to be paid in exact amounts, using Gold Dragons—the currency of the Seven Kingdoms.

The Royal Treasury has k denominations of coins. Daenerys aims to use the fewest coins possible for each payment to conserve the treasury's resources. Queen Daenerys tasks Tyrion Lannister to help her make these payments efficiently.

Tyrion is given coin denominations d_i for each coin i ($1 \leq i \leq k$), and an integer M representing the exact amount needed to fund a specific project. His task is to tell his queen the minimum number of coins needed to fund the project. If it's not possible to fund the exact amount using the available coin denominations, Tyrion must tell her so.

The Night's Watch Exam Preparation

Jon Snow, a member of the Night's Watch, has to prepare for the exams to advance in rank. However, *winter is coming*, and the weather is getting colder in the North. Jon prefers to be outside sharpening his sword, training, or scouting during the time where the weather is more favorable. Specifically, if the temperature rises to t integer units above freezing, Jon's morale will increase by t for that day (or decrease if t is negative).

In the next n days leading up to his exams, Jon has to decide whether to spend each day studying in the library with Samwell Tarly or training outside. To balance his responsibilities, he vows not to spend more than two consecutive days outside.

Given Maester Aemon's weather forecast estimating the temperatures for the next n days, your task is to describe an $O(n)$-time dynamic programming algorithm to determine which days Jon should study and which days he should train outside to maximize his overall morale.