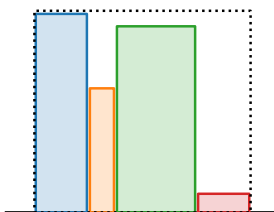


Homework 5

CS 331

Due Thursday, February 22

1. Suppose you have an $n \times n$ matrix M of numbers where each entry can be either positive or negative. You want to move from the top left corner to the bottom right corner in a way so that the sum of the numbers that you traverse is as large as possible. You are allowed to move left, right, or down (but not up) and you can never visit the same square twice. Give an $O(n^2)$ time dynamic programming algorithm that finds the value of the best path from top left to bottom right.
2. (Optional) You have n rectangles, each of which has a height h_i and width w_i , $h_i \geq w_i$. You would like to place them next to each other on a line, such that the area of the minimum enclosing axis-aligned rectangle is minimized. You may rotate the rectangles, but they must be placed with one edge flush against the given line. For example, this is a valid but suboptimal solution:



It is suboptimal because it would be better to rotate the red rectangle so the enclosing rectangle can be less wide. Give an $O(n \log n)$ time algorithm to compute the area of the minimum enclosing rectangle.

Hint (rot13): Fhccbfr gur fbyhgvba jrer gb “gbccyr” x bs gur erpgnatyrf; juvpu barf jbyhq or xabpxrq bire? Pna lbh gura gel nyy cbffvoyr inyhrf bs x?