Zihao Shao Assignment-1

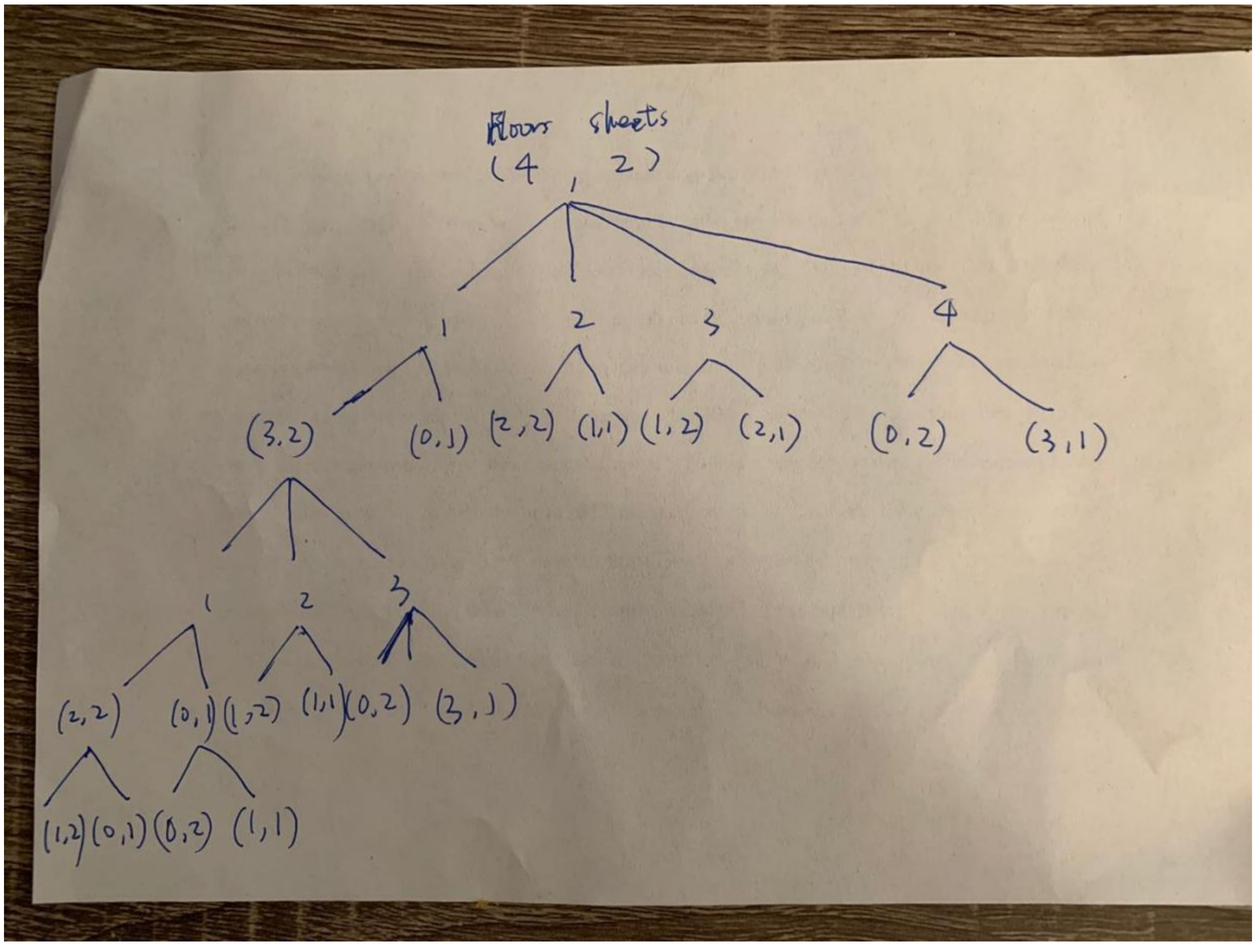
Falling Class

(a) Describe the optimal substructure/recurrence that would lead to a recursive solution

1. Set up the base cases for the algorithm where when we have less than 1 floors or if we have 1 sheet left.

2. Create a value to represent the largest value possible comparing this with the biggest value between the result of 1 less floor and sheet and the result of an x number of floors less and returning the least value found from all this.

(b) Draw recurrence tree for given (floors = 4, sheets = 2)



(c) Code your recursive solution under GlassFallingRecur(int n numFloors, int m numGlass)

Posted

(d) How many distinct subproblems do you end up with given 4 floors and 2 sheets?

4*2 = 8 distinct subproblems

(e) How many distinct subproblems for n floors and m sheets?

n*m distinct subproblems

(f) Describe how you would memoize GlassFallingRecur

1. Create a cache to store temporary results.

2. Check whether the calculation has been done, if it is done, use the store result.

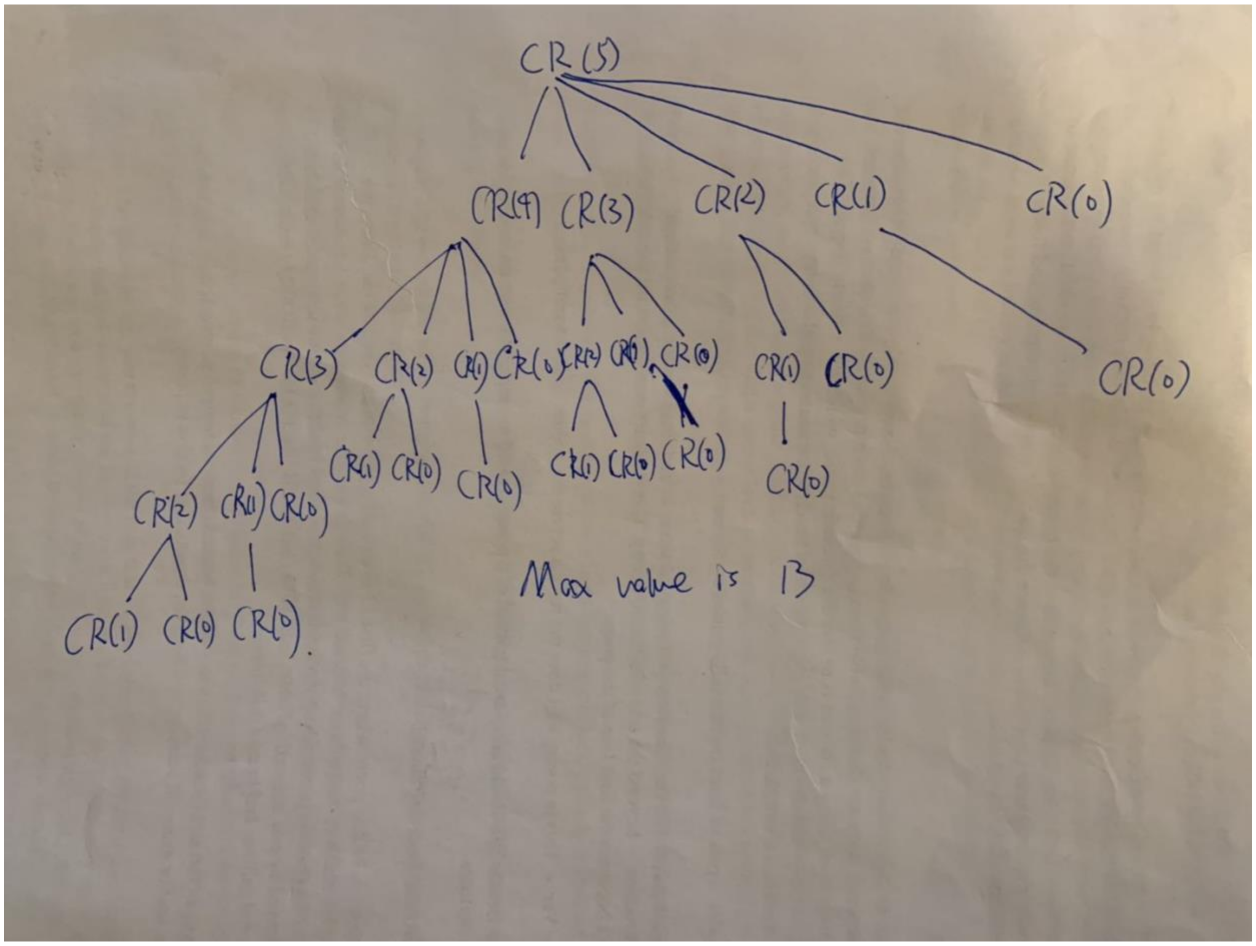
3.Check if the value is first time calculate, store the results.

(g) Code a bottom-up solution GlassFallingBottomUp(int n numFloors, int m numGlass)

Posted

Rod Cutting

(a) Draw the recursion tree for a rod of length 5


CR(5)
CR(4) CR(3) CR(2) CR(1) CR(0)
CR(3) CR(2) CR(1) CR(0) CR(2) CR(1) CR(0) CR(1) CR(0) CR(0)
CR(1) CR(0) CR(0) CR(1) CR(0) CR(0) CR(0)
CR(2) CR(1) CR(0)
CR(1) CR(0) CR(0)
Max value is 13

(b) On page 370: answer 15.1-2 by coming up with a counterexample, meaning come up with a situation / some input that shows we can only try all the options via dynamic programming instead of using a greedy choice.

Let p1=1, p2 = 5, p3 = 8, p4 = 9, n=4

1/1=1, 5/2=2.5, 8/3=2.667, 9/4=2.25

Max density is 8/3, the greedy first cut off a piece of length of 3, remain 1. 8+1=9 total profit.

But if cut the rob with length of will give 5+5=10 is greater than 9.

(c) Code the memoized recursive version in RodCutting.java under rodCuttingRecur

Posted

(d) Code the bottom-up solution in RodCutting.java under rodCuttingBottomUp

Posted