

Glass Falling

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

Ans-Given: # of floors= n

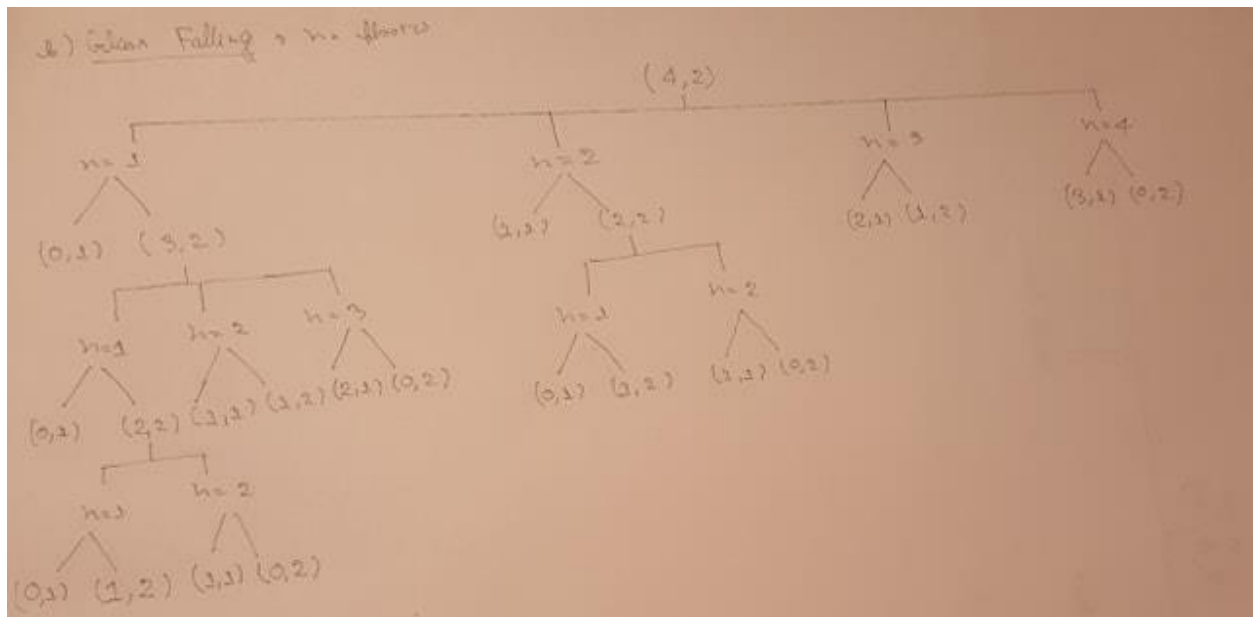
of glass sheet= m

Assume the trial is being held on the i -th floor. There are 2 outcomes from the trial.

Outcome 1- The glass sheet breaks. This will result in $(m-1)$ glass sheets and $(i-1)$ remaining trial floors. [Since we know if we go higher the glass sheets will definitely break]

Outcome 2- The glass sheet does not break. This will result in (m) glass sheets with (n-i) remaining trial floors.[Since we know the lower floors will not result in glass break, we have to go higher]

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



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

Ans-8 distinct subproblems.

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

Ans- $n*m$ distinct subproblems.

(f) Describe how you would memoize GlassFallingRecur

Ans-To memorize GlassFallingRecur follow the following steps

Step 1- Create a table/array for [floors, sheets] where the result of base case recursive calls are stored.

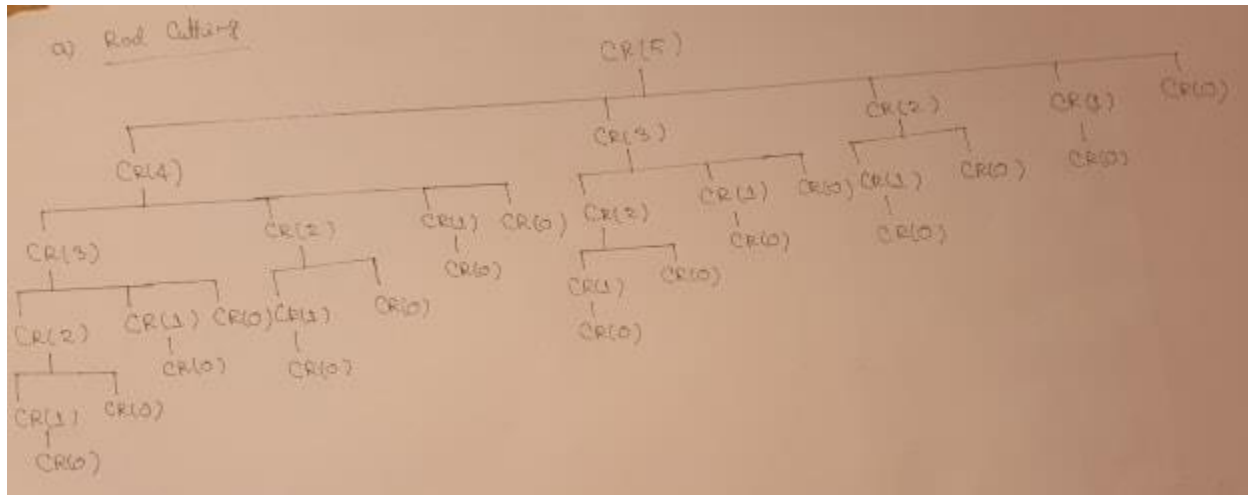
Step 2- The code will follow recursive method but before each call it will have to check the table for existing values of [floor, sheets],

Step 3- If the value exists, the code will then use the value from the table and continue to the next recursive call

Step 4- If the value does not exist, the code will store the value of the current recursive call.

Rod cutting

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



(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.

Ans-

Counterexample: Rod Length-1, 2, 3, 4,

Price- 1, 10, 27, 32,

Density (Price/Length)-1, 5, 9, 8

To acquire rod of length 4, the greedy algorithm will pick length 3 first since it has the highest density. This will result in a rod with 2 pieces one length 3 and other length 1. And their value will summed up to be $(27+1) = 28$. But in reality that is not the optimal solution, since picking rod length will give a value of $32 > 28$. Hence why Dynamic Programming will provide the optimal solution for this problem.