

Nathaniel Abreu

CS323 Assignment 1 - Glass Falling

A). We do not know the strength of the glass sheet we are testing. Therefore, whenever we drop the sheet there could only happen 1 of 2 outcomes. Either the glass breaks or it does not. For the first possible outcome, if the glass breaks ~~on~~ on a floor, we are then only concerned with checking the remaining lower floors with the remaining glass. So, the sub problem becomes that floor - 1 and # of Sheets - 1.

For the second possible outcome, if the sheet does not break then we are only concerned with checking the floors above the floor tested. So the subproblem becomes floors - # of floor tested.

Moreover, since we need the minimum # of trials we need to check the maximum of each possible outcome, ~~the~~. Since that would be the worst case scenario. So, for the outcome where the glass does not break the worst case would be floors - 1st floor continuing up each floor. For the outcome where the glass breaks, it would be floors - 1 and Sheets - 1.

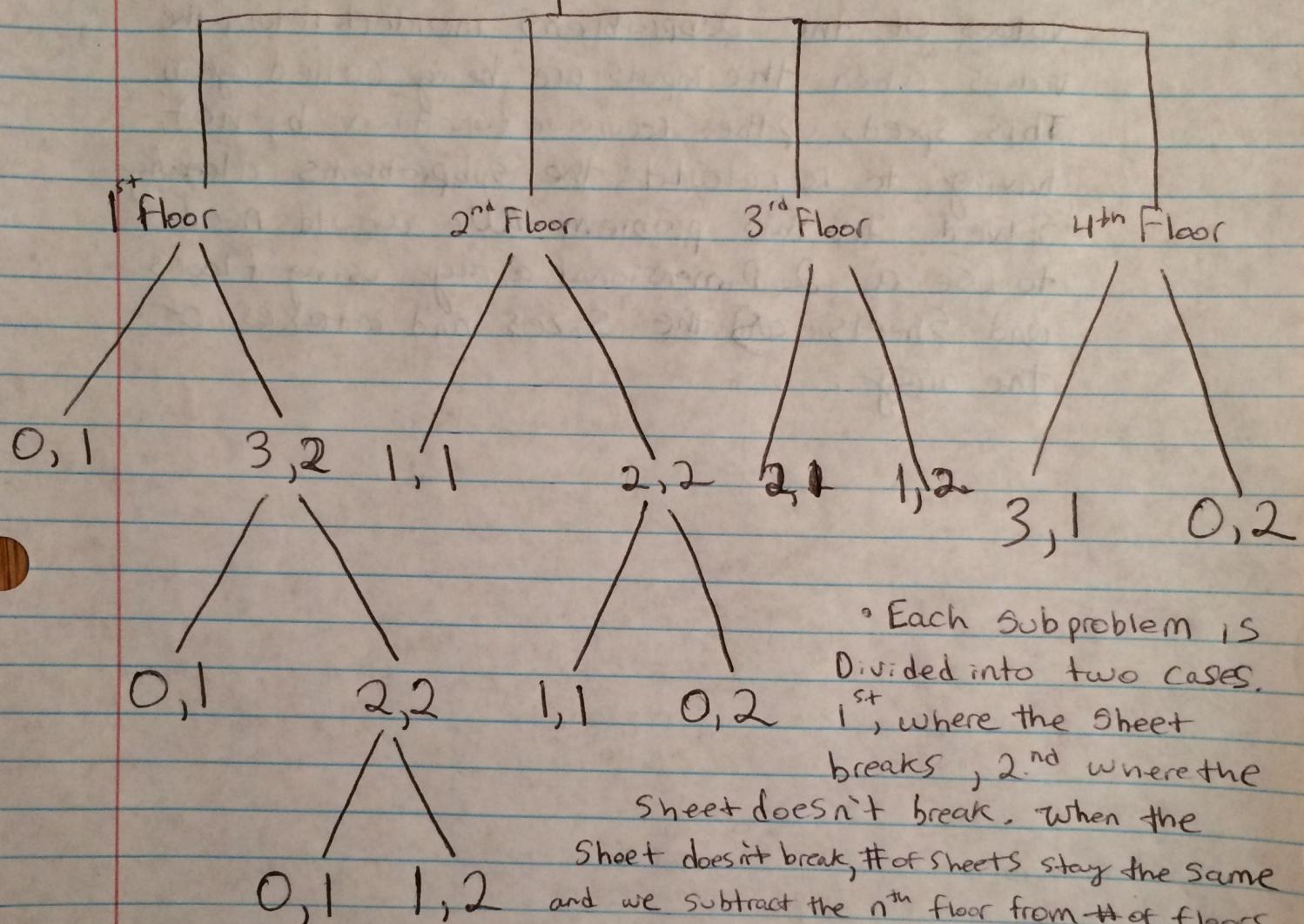
Reference

break

no
break

Floors sheets
(4, 2)

B). Recurrence Tree:



- Each subproblem is divided into two cases.

- 1st, where the sheet breaks ; 2nd where the sheet doesn't break.

- When the sheet doesn't break, # of sheets stay the same and we subtract the n^{th} floor from # of floors.

- Since when floor = 0, 1 or Sheets = 1 floors is returned and ~~the~~ function does not go further.

D). There are 8 distinct Subproblems for 4 Floors and 2 Sheets.

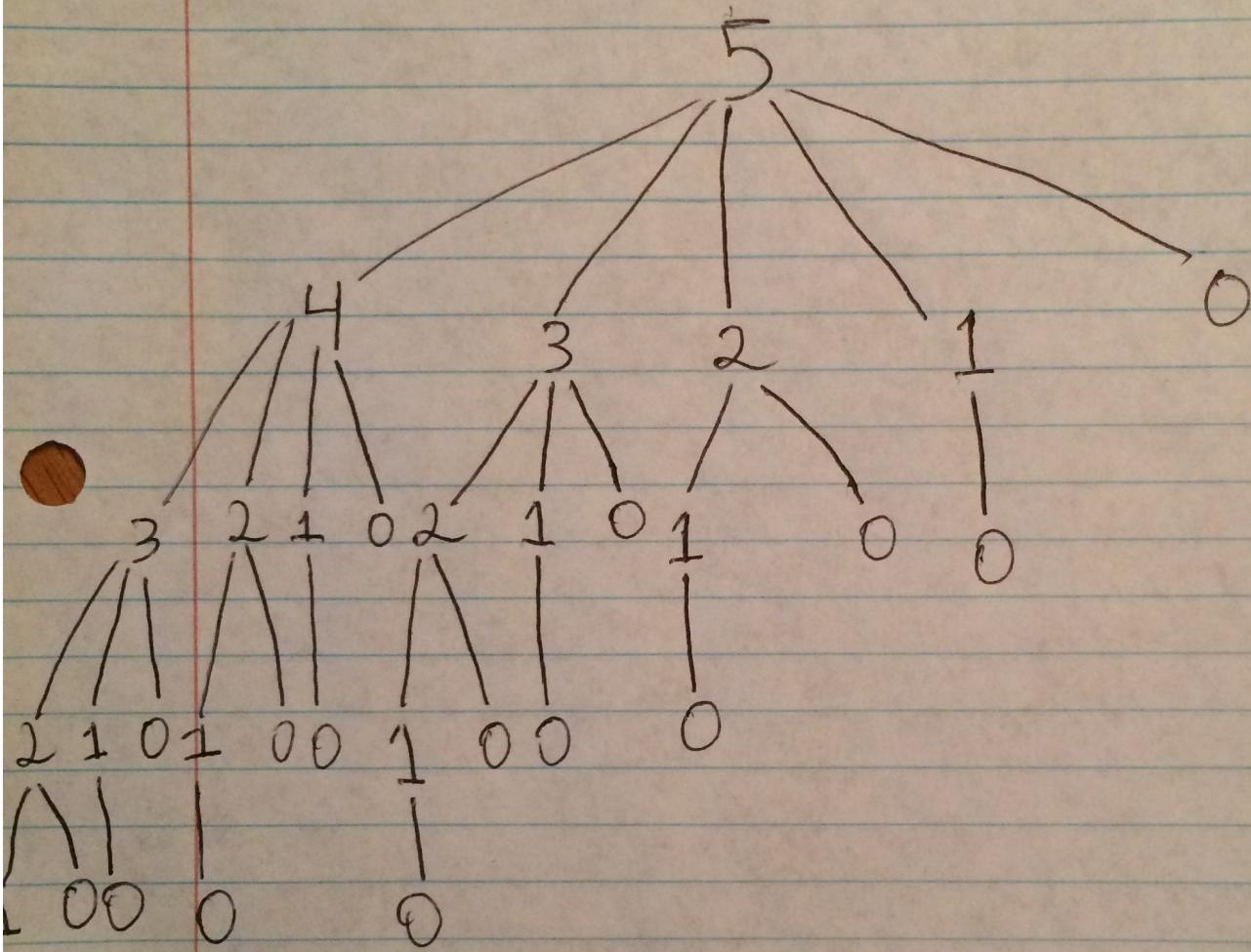
e). There are 2^n distinct subproblems for n floors and m sheets.

f). To memoize GlassFallingRecur an array needs to be created in order to store the values of the Subproblems in order to return the values when the inputs are being called again. This speeds up the recursive function by not having to re calculate the subproblems already solved. In this problem we would need to use a 2 Dimensional array using floors and Sheets and the sizes and indexes of the array.

Nathaniel fibrew

CS323 Assignment 1 - Rod Cutting

A). Recursion Tree



B). Let's say we have a rod length of 5. The values for the densities of each length cut is as follows $1=1, 2=5, 3=6, 4=7, 5=8$. Using the greedy solution we use length 5 since its the highest density. So the remaining is $5-5=0$ and 8 is the output. However, the optimal would actually be lengths 2 + 3. Since we would get 11 as the total value.