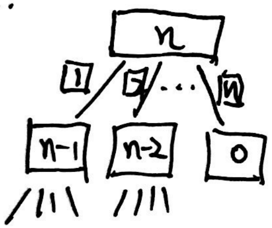
Part 1: we can assume you have rod of n length.

So, for every point $i = 1 \cdots n$, you can cut or not cut.
So, you have 2^n different cut method.
calculate the cost need n time,
so, the running time is $O(2^n \cdot n)$.

Part 2:



Define C_i as the ~~min cost~~ max amount of money of a diamond of weight i.

we can have, $y = 2$.

$$C_n = \max\{P_i - y + C_{n-i}, \text{ for } i = 1 \cdots n\}.$$

general:

$$\begin{cases} C_i = \max\{P_j - y + C_{i-j}, \text{ for } j \le i\}. \\ C_0 = 0 \end{cases}$$

The run time of this DP algorithm is $O(n^2)$

Part 3:

$C_n = \{P_i - y + C_{n-i}, \text{ for } i = 1 \cdots n\}$. $y = 2$.

Given $\underbrace{C_n}_{\text{is the max amount of money}}$ for dimand of weight n with first cut i,

$C_n = P_i - y + a$,

a is the amount of money of dimand of weight $n-i$.

Prove, a is the max amount of money of dimand of weight $n-i$.

Proof: Assume: a is not the max amount of money.

$\Rightarrow \exists b$ is the cut stratgy that $rev(b) > rev(a)$.

$rev(b) + P_i - y > rev(a) + P_i - y = C_n$

it is contradiction with the information that C_n is the max amount of money, So, there does not exist b and a is the max amount of money for subproblem.

Scanned with CamScanner