[DPV] Problem 6.2 (hotel stops)

(1a) Define the entries of your table in words. E.g., $T(i)$ is ..., or $T(i,j)$ is ....

$P(i)$ = minimum penalty from a trip starting at 0, ending at i & allowed to stop at locations $a_1, a_2, \ldots, a_i$

(1b) State the recurrence for the entries of your table in terms of smaller subproblems.

$T(0) = 0$

let $a_0 = 0$

$$T(i) = \min_{0 \le j \le i-1} \left\{ T(j) + \left(200 - (a_i - a_j)\right)^2 \right\}$$

Penultimate stop at a_j: Penalty up to a_j Penalty for last day from $a_j \to a_i$

(1c) Write pseudocode for your algorithm to solve this problem.

$T(0) = 0$

for $i = 1 \rightarrow n$

$T(i) = (200 - a_i)^2$

for $j = 1 \rightarrow i-1$

if $T(i) > T(j) + (200 - (a_i - a_j))^2$

then $T(i) = T(j) + (200 - (a_i - a_j))^2$

Return$(T(n))$

(1d) Analyze the running time of your algorithm.

2 nested for $\underbrace{\text{loops}}_{}$ of size $O(n)$ & $O(1)$ time inside

$\Rightarrow O(n^2)$ total time.

[DPV] Problem 6.3 (Yuckdonald's)

(1a) Define the entries of your table in words. E.g., $T(i)$ is ..., or $T(i,j)$ is ....

$T(i)$ = max profit from restaurants at a subset of $m_1, \ldots, m_i$ but including m_i

(1b) State the recurrence for the entries of your table in terms of smaller subproblems.

$T(0) = 0$

$T(i) = \max_j \{ p_i + T(j) : 0 \le j \le i-1, m_j \le m_i - k \}$

(1c) Write pseudocode for your algorithm to solve this problem.

$T(0) = 0$

for $i = 1 \rightarrow n$

$\quad T(i) = p_i$

$\quad$ for $j = 1 \rightarrow i-1$

$\quad\quad$ if $m_j \leq m_i - k$ & $T(i) < p_i + T(j)$

$\quad\quad\quad$ then $T(i) = p_i + T(j)$

$max = 0$

for $i = 1 \rightarrow n$

$\quad$ if $T(i) > T(max)$

$\quad\quad$ then $max = i$

Return$(T(max))$

(1d) Analyze the running time of your algorithm.

2 nested for loops of size $O(n)$

& $O(1)$ time inside

$\Rightarrow O(n^2)$ total time.