Task 2: Gym Badges (`gymbadges`)

Authored and prepared by: Teow Hua Jun

Subtask 1

Limits $N \leq 10$

Iterate through all permutations of gyms and simulate going to the gyms in order and challenge them if possible. The maximum gym badges you can obtain among all permutations will be the answer.

Time Complexity: $O(N!)$

Subtask 2

Limits: L is Constant

Since L is constant the only difference between gyms is the level gained. Sort the gym by L_i and greedily choose the smallest X_i until sum of X chosen is greater than L.

Time Complexity: $O(N \log N)$

Subtask 3

$N \leq 5000$

Limits: $N \leq 5000$

Note that there exist an optimal solution, $a_1, a_2, a_3, ..., a_k$ where a_i is the i^{th} gym challenged, such that $X_{a_i} + L_{a_i} \leq X_{a_{i+1}} + L_{a_{i+1}}, \forall i < k$.

Proof: Assume exist optimal solution, $a_1, a_2, a_3, ..., a_k$ where exist adjacent gyms such that $X_{a_i} + L_{a_i} > X_{a_{i+1}} + L_{a_{i+1}}$.

$$L_{cur} = \sum_{n=1}^{i-1} X_{a_i}$$

$$L_{cur} + X_{a_i} \le L_{a_{i+1}}$$
$$L_{cur} + X_{a_i} + X_{a_{i+1}} \le L_{a_{i+1}} + X_{a_{i+1}} < L_{a_i} + X_{a_i}$$
$$\therefore L_{cur} + X_{a_{i+1}} < L_{a_i}$$

Thus, we can challenge gym a_{i+1} before a_i, swapping the order of these two adjacent gyms, note that gyms after the a_{i+1} gym are not affected as the gyms a_i and a_{i+1} still make Wabbit gain $X_{a_i} + X_{a_{i+1}}$ levels.

Therefore, from any optimal solution we can keep swapping adjacent gyms to make the solution sorted with respect to $X_i + L_i$

Using this observation, we can sort the gyms and process them in order.

Let $dp(a, b)$ = Minimum level Wabbit will be at after challenging a gyms after processing b gyms.

We can form a simple transition of

$$dp(a,b) = \begin{cases} \min(dp(a-1,b-1) + X_a, dp(a-1,b)), & \text{if } dp(a-1,b-1) \le L_a \\ dp(a-1,b), & \text{otherwise} \end{cases}$$

State: $O(N^2)$ Transition: $O(1)$

Time Complexity: $O(N^2)$

Subtask 4

Limits: No further constraints

Using the observation made in subtask 3, we keep the gyms sorted by $X_i + L_i$

Let $S_x = \{a_1, a_2, ..., a_k\}$ be an optimal set of gyms challenged to obtain the maximum number of badges with the minimum level gain from the first x gyms and $G_x = \sum_{i=1}^{|S_x|} X_{a_i}$, the total level gain after challenging the gyms in S_x.

Lemma 1: If $L_{n+1} < G_n$ and exist $y \in S_n$ such that $X_y > X_{n+1}$, then we can replace gym y with gym $n+1$.

Let gym k be the last gym in S_n. Note that k can be the same as y

$$\begin{aligned} G_n - X_k \le L_k \implies G_n &\le X_k + L_k \\ &\le X_{n+1} + L_{n+1} \\ &< X_y + L_{n+1} \end{aligned}$$

$$\therefore G_n - X_y < L_{n+1}$$

Thus, we can swap gym y with gym $(n+1)$

Solution:

Assume we have the optimal solution for the first n^{th} gyms with the maximum number of badges with minimum level gain and currently processing our $(n+1)^{th}$ gym.

If $L_{n+1} \geq G_n$, then $S_{n+1} = S_n \cup \{n+1\}$:

Namely, we add gym $n+1$ to the current solution set of gyms.

This can be proven by contradiction as if exist another set of gyms with greater number of badges or less level gain. Then there will exist a set of gyms in the first n gyms that have greater number of badges or less level gain than S_n

Let the gym will the maximum level gain in S_n be gym y

Else if $L_{n+1} \geq G_n$ and $X_y > X_{n+1}$, then $S_{n+1} = S_n \setminus \{y\} \cup \{n+1\}$:

Namely, we swap gym y for gym $(n+1)$ to have a lower level gain.

We can prove this is optimal as we cannot have $|S_n| + 1$ gym badges as $L_{n+1} \geq G_n$ and the level gain is minimised as otherwise S_n will have a smaller level gain. From lemma 1 we can also see that this swap is possible.

Else $S_{n+1} = S_n$:

We can prove this is optimal using similar logic as the above case.

Hence, as we have shown how to construct the optimal solution of $n+1$ gyms from n, we can inductively generate the answer. The actual implementation can be done using a priority queue or set to maintain the set of gyms.

Time Complexity: $O(N \log N)$

Alternative solution

The alternative solution is based on the following observations:

Observation 1: suppose gym i has the smallest value of X_i among all gyms. Then there exists an optimal solution which challenges gym i at some point in time.

Proof of observation 1: Consider the first gym challenged in any optimal solution which does not use gym i. Change the first gym to gym i.

Observation 2: If we know that there exists an optimal solution which challenges gym i, then we can calculate the answer by doing the following:

- Remove gym i
- For all gyms $j \neq i$, if $X_j + L_j > X_i + L_i$, replace L_j by $L_j - X_i$ (if L_j becomes negative due to this operation, we can delete gym j)
- Calculate the answer to this new problem, and increase the answer by 1

Proof of observation 2: suppose we have a sequence of gym challenges that uses gym i. Now consider what happens when we delete gym i from the sequence. For every gym j challenged after i, the entering level when gym j is challenged is decreased by X_i.

Similarly, if we have a solution to the new problem, we can attempt to insert gym j at the latest possible opportunity, and observe that gyms which satisfy $X_j + L_j > X_i + L_i$ are affected.

These two observations immediately give us an $O(N^2)$ solution. We can speedup this to $O(N \log N)$ by using lazy propagation segment tree.