Documentation for Russian Doll Envelopes

Intuition

The problem of Russian Doll Envelopes requires identifying the maximum number of envelopes that can fit into one another based on their dimensions. For one envelope to fit inside another, its width and height must be strictly smaller than those of the other envelope. This creates a two-dimensional ordering challenge. The task can be simplified by recognizing that once the envelopes are sorted in a specific way, the problem reduces to finding the longest increasing Sequence (LIS), a well-known algorithmic problem.

Sorting and Transformation

To address the two-dimensional nature of the problem, we sort the envelopes. First, the envelopes are sorted by their widths in ascending order. For envelopes with the same width, they are sorted by their height in descending order. The descending order for heights in the case of ties ensures that envelopes with the same width cannot be nested, as they would conflict during the LIS computation. After sorting, the height of the envelopes becomes the key to solving the problem, as the width order is already fixed by the sorting step.

Longest Increasing Subsequence (LIS) Application

Once the envelopes are sorted, the task of finding the maximum number of nested envelopes reduces to finding the LIS based on the heights of the envelopes. The LIS problem involves finding the longest subsequence where each element is greater than the previous one. Here, the sequence is the list of heights extracted from the sorted envelopes. Using LIS on the heights ensures that we find the maximum nesting sequence while respecting the constraints of the problem.

Efficient Computation of LIS

To compute the LIS efficiently, we use a greedy algorithm combined with binary search. A dynamic list (lis) is maintained, which represents the smallest ending values of increasing subsequences of all possible lengths encountered so far. For each height in the sequence, a binary search is performed to determine its position in the lis list. If the height is greater than all current elements, it is appended to the list. Otherwise, the smallest element in lis that is greater than or equal to the height is replaced with the current height. This ensures that lis remains optimized for future computations.

Key Insights from Sorting

The descending sort of heights for envelopes with the same width plays a critical role in the algorithm. Without this step, envelopes with equal widths could interfere with the LIS computation, leading to incorrect results. By sorting heights in descending order for equal widths, we ensure that only valid sequences are considered during LIS computation, as envelopes with the same width cannot form a valid nesting sequence.

Complexity and Scalability

The time complexity of the solution is O(nlogn), where n is the number of envelopes. This complexity arises from the sorting step, which takes O(nlogn), and the LIS computation, which also takes O(nlogn) due to the binary search. The space complexity is O(n), as we need additional space to store the lis array. This efficient combination of sorting and binary search makes the solution scalable, and capable of handling the maximum constraints of the problem.

Conclusion and Broader Applications

The Russian Doll Envelopes problem demonstrates the power of reducing complex multi-dimensional problems into simpler one-dimensional forms. By leveraging sorting and LIS techniques, the solution efficiently handles the constraints and produces the desired result. This approach can be applied to similar problems in computational geometry and optimization, where relationships between elements must be maximized under specific constraints. The problem highlights the importance of structured problem-solving and algorithmic transformation in achieving optimal solutions.