ZOJ 1025

daseinpwt

The problem can be interpreted as to find the minimum number of sequences of the sticks, where every two sticks i and j $(i < j)$ in each sequence meet the condition $l_i \leq l_j$ and $w_i \leq w_j$.

One step to make the problem simpler is to first sort the sticks, using l as the first key (ascending) and w the second (ascending). Let S denotes the sorted list. Now we can instead find the minimum number of non-decreasing subsequences of S, where every two sticks i and j $(i < j)$ in each sequence meet the condition $w_i \leq w_j$. Note that the condition is simplified because that $l_i \leq l_j$ always holds.

Next, we proof that the answer (the minimum number of non-decreasing subsequences) is equal to the length of the longest strictly decreasing subsequence (in terms of w) of S.

Proof. Let M denotes the length of the longest strictly decreasing subsequence, and P denotes the minimum number of non-decreasing subsequences.

(i) For every S_i and S_j $(i \neq j)$ in the longest strictly decreasing subsequence, S_i and S_j are not in the same non-decreasing subsequence. Thus $P \geq M$.

(ii) We introduce M empty queues q_1, q_2, ..., q_M. Then we move the elements from S into the M queues one by one. For each element S_i, we first check with q_1. If q_1 is empty or the tail element $tail_1$ in q_1 satisfies that $w_{tail_1} \leq w_i$ then S_i is appended to q_1. If not, we check with q_2, q_3 and more until S_i is appended into some queue. This will certainly happen because that if we can not find a place for S_i after checking all M queues, we find a strictly decreasing subsequence of length $M + 1$, which is a contradiction. Finally, we will get less than or equal to M non-decreasing subsequences (which are all the non-empty queues). Thus $P \leq M$.

In conclusion, $P = M$. □

The problem is now transformed to find the length of the longest strictly decreasing subsequence. A classic $\mathcal{O}(N^2)$ algorithm is sufficient.