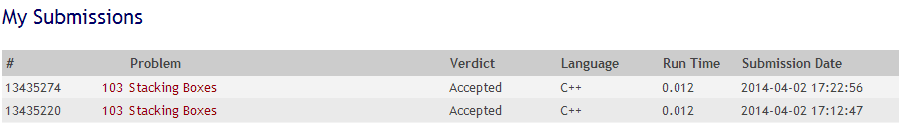
Stacking Boxes

UVA Online Judge

DASALGO Problem Code: S1 Status: Accepted

UVA Problem 103 – Stacking Boxes Runtime: 0.012 seconds



(Source code with comments at the last page)

Discussion

Box 1 can be stacked to Box 2 if all of the dimensions of Box 1 are lower than Box 2. This is the condition for boxes to be stackable.

First, two numbers are asked for input, these numbers corresponds to the number of boxes and the dimensions it has respectively. Immediately after the dimensions of the box are inputted, it is sorted. After which, all boxes are then compared, to see which one is higher than the other, this sorts the boxes in terms of highest to lowest dimensions. Box 1 can be stacked to Box 2 if all of the dimensions of Box 1 are lower than Box 2. To get the maximum number of boxes that can be stacked from the given input, two arrays are created to store the information about the boxes. First array contains the stackable boxes. Second array contains the indexes of the stackable boxes. With this information, we can get the most number of stackable boxes and their indexes. The first array gets its values changed when the box can be stacked on by other boxes, and the boxes before it have lower ability to stack. The second array gets its values by getting the indexes of the boxes that can be stacked. After the values are changed, it is time to put the stackable boxes in order. To do this, we get the index of the highest value in the first array, meaning it is the highest possible stacks we can have. After getting the index of the highest, we use this index in the second array to get the index of the box that can have the highest stacks. By replacing the recent value of the second array to the index of the next box, we can recursively find the boxes to be placed in order until a value is repeated. We then use the gotten indexes to find the values in the original array. The values in the original array are then pushed into a stack. The output is the maximum number of boxes that can be stacked, and the corresponding order by getting the values from the stack.

References

"programming contest problems and solutions." : ACM. N.p., n.d. Web. 2 Apr. 2014. <http://acm-solution.blogspot.com/2010/11/acm-uva-103-stacking-boxes.html>.