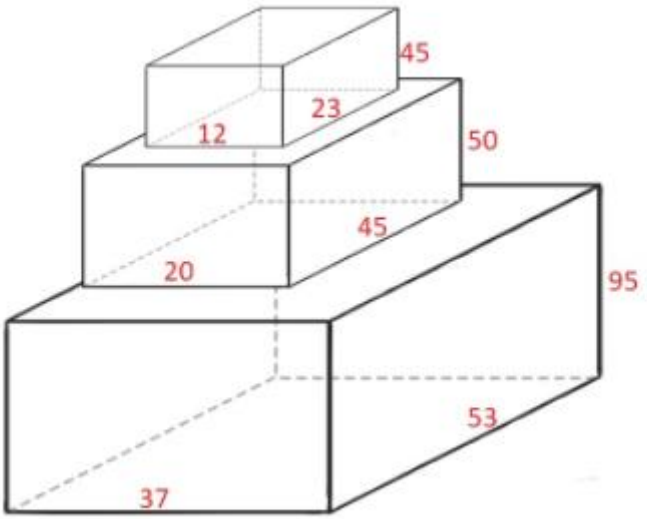
Maximum Height by Stacking Blocks

Given `n` `blocks` where the dimensions of the `i`th block is `blocks[i] = [width`$_i$`, length`$_i$`, height`$_i$`]` (**0-indexed**). Choose a **subset** of `blocks` and place them on each other.

You can place block `i` on block `j` if `width`$_i$ `<= width`$_j$ and `length`$_i$ `<= length`$_j$ and `height`$_i$ `<= height`$_j$. You can rearrange any blocks's dimensions by rotating it to put it on another block.

Return *the* ***maximum height*** *of the stacked* `blocks`.

Example 1:



Input: blocks = [[50,45,20],[95,37,53],[45,23,12]]

Output: 190

Explanation:

Block 1 is placed on the bottom with the 53x37 side facing down with height 95.

Block 0 is placed next with the 45x20 side facing down with height 50.

Block 2 is placed next with the 23x12 side facing down with height 45.

The total height is 95 + 50 + 45 = 190.

Example 2:

```
Input: blocks = [[38,25,45],[76,35,3]]

Output: 76

Explanation:

You can't place any of the blocks on the other.

We choose block 1 and rotate it so that the 35x3 side is facing down and its
height is 76.
```

Example 3:

```
Input: blocks = [[7,11,17],[7,17,11],[11,7,17],[11,17,7],[17,7,11],[17,11,7]]

Output: 102

Explanation:

After rearranging the blocks, you can see that all blocks have the same dimension.

You can place the 11x7 side down on all blocks so their heights are 17.

The maximum height of stacked blocks is 6 * 17 = 102.
```

Constraints:

- `n == blocks.length`
- `1 <= n <= 100`
- `1 <= width_i, length_i, height_i <= 100`