AlgoExpert Quad Layout Python 12px Sublime Monokai 00:00:00

Prompt Scratchpad Our Solution(s) Video Explanation Run Code

Solution 1

```
1 \, # Copyright @ 2020 AlgoExpert, LLC. All rights reserved.
 3 \# O(n^2) time | O(n) space
 4 def diskStacking(disks):
        disks.sort(key=lambda disk: disk[2])
         heights = [disk[2] for disk in disks]
         sequences = [None for disk in disks]
         maxHeightIdx = 0
 8
         for i in range(1, len(disks)):
 9
10
             currentDisk = disks[i]
             for j in range(0, i):
11
                 otherDisk = disks[j]
12
13
                 if areValidDimensions(otherDisk, currentDisk):
                      if heights[i] <= currentDisk[2] + heights[j]:</pre>
14
15
                          heights[i] = currentDisk[2] + heights[j]
                          sequences[i] = j
16
             if heights[i] >= heights[maxHeightIdx]:
17
18
                 maxHeightIdx = i
19
         return buildSequence(disks, sequences, maxHeightIdx)
20
21
    def areValidDimensions(o, c):
22
23
          \textbf{return} \ \texttt{o[0]} \ \texttt{<} \ \texttt{c[0]} \ \textbf{and} \ \texttt{o[1]} \ \texttt{<} \ \texttt{c[1]} \ \textbf{and} \ \texttt{o[2]} \ \texttt{<} \ \texttt{c[2]} 
24
25
26 def buildSequence(array, sequences, currentIdx):
27
         sequence = []
28
         while currentIdx is not None:
29
             sequence.append(array[currentIdx])
30
             currentIdx = sequences[currentIdx]
31
         return list(reversed(sequence))
32
```