AlgoExpert

Solution 1 Solution 2

28

29

return abs(a - b)

\_\_\_\_

**Quad Layout** 

Python

14рх

Sublime

Monokai

00:00:

Run Code

Our Solution(s)

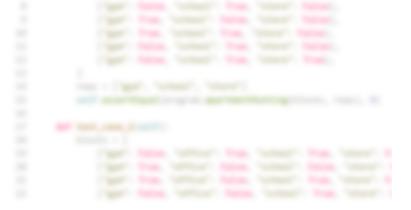
```
Run Code
```

**Your Solutions** 

Solution 3

```
1 # Copyright © 2020 AlgoExpert, LLC. All rights reserved.
   # O(b^2*r) time | O(b) space - where b is the number of blocks and r
   def apartmentHunting(blocks, reqs):
       maxDistancesAtBlocks = [float("-inf") for block in blocks]
       for i in range(len(blocks)):
           for req in reqs:
               closestReqDistance = float("inf")
9
                for j in range(len(blocks)):
                   if blocks[j][req]:
10
                       closestReqDistance = min(closestReqDistance, dist
11
               maxDistancesAtBlocks[i] = max(maxDistancesAtBlocks[i], cl
12
13
        return getIdxAtMinValue(maxDistancesAtBlocks)
14
15
16 def getIdxAtMinValue(array):
17
        idxAtMinValue = 0
       minValue = float("inf")
18
19
        for i in range(len(array)):
20
           currentValue = array[i]
21
           if currentValue < minValue:</pre>
22
               minValue = currentValue
23
               idxAtMinValue = i
24
        return idxAtMinValue
25
26
27 def distanceBetween(a, b):
```

```
Solution 1 Solution 2
1 def apartmentHunting(blocks, reqs):
      # Write your code here.
      pass
```



Run or submit code when you're ready.