AlgoExpert

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

Prompt | Scratchpad | Our Solution(s) | Video Explanation | Run Code

Solution 1 | Solution 2

```python
# Copyright © 2020 AlgoExpert, LLC. All rights reserved.

# O(b^2*r) time | O(b) space - where b is the number of blocks and r is the number of requirements
def apartmentHunting(blocks, reqs):
    maxDistancesAtBlocks = [float("-inf") for block in blocks]
    for i in range(len(blocks)):
        for req in reqs:
            closestReqDistance = float("inf")
            for j in range(len(blocks)):
                if blocks[j][req]:
                    closestReqDistance = min(closestReqDistance, distanceBetween(i, j))
            maxDistancesAtBlocks[i] = max(maxDistancesAtBlocks[i], closestReqDistance)
    return getIdxAtMinValue(maxDistancesAtBlocks)


def getIdxAtMinValue(array):
    idxAtMinValue = 0
    minValue = float("inf")
    for i in range(len(array)):
        currentValue = array[i]
        if currentValue < minValue:
            minValue = currentValue
            idxAtMinValue = i
    return idxAtMinValue


def distanceBetween(a, b):
    return abs(a - b)

```