

AlgoExpert

Quad Layout

C++

12px

Sublime

Monokai

00:00:00

PromptScratchpadOur Solution(s)Video Explanation

Run Code

Solution 1Solution 2

1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 #include <vector>
4 #include <unordered_map>
5 #include <climits>
6 #include <algorithm>
7 #include <cmath>
8
9 using namespace std;
10
11 int getIdxAtMinValue(vector<int> array);
12 int distanceBetween(int a, int b);
13
14 // O(b^2*r) time | O(b) space - where b is the number of blocks and r is the
15 // number of requirements
16 int apartmentHunting(vector<unordered_map<string, bool>> blocks,
17 vector<string> reqs) {
18 vector<int> maxDistancesAtBlocks(blocks.size(), INT_MIN);
19 for (int i = 0; i < blocks.size(); i++) {
20 for (string req : reqs) {
21 int closestReqDistance = INT_MAX;
22 for (int j = 0; j < blocks.size(); j++) {
23 if (blocks[j][req]) {
24 closestReqDistance = min(closestReqDistance, distanceBetween(i, j));
25 }
26 }
27 maxDistancesAtBlocks[i] =
28 max(maxDistancesAtBlocks[i], closestReqDistance);
29 }
30 }
31 return getIdxAtMinValue(maxDistancesAtBlocks);
32 }
33
34 int getIdxAtMinValue(vector<int> array) {
35 int idxAtMinValue = 0;
36 int minValue = INT_MAX;
37 for (int i = 0; i < array.size(); i++) {
38 int currentValue = array[i];
39 if (currentValue < minValue) {
40 minValue = currentValue;
41 idxAtMinValue = i;
42 }
43 }
44 return idxAtMinValue;
45 }
46
47 int distanceBetween(int a, int b) { return abs(a - b); }
48

