Solution 1 Solution 2

Your Solutions

Solution 1 Solution 2 Solution 3

Run Code

Our Solution(s) Run Code

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
4 #include <vector>
 5 #include <unordered_map>
 6 #include <climits>
7 #include <algorithm>
8 #include <cmath>
10 using namespace std;
11
12 vector<int> getMinDistances(vector<unordered_map<string, bool>> blocks
13
                               string req);
14 vector<int> getMaxDistancesAtBlocks(vector<unordered_map<string, bool>
15
                                      vector<vector<int>> minDistancesFr
16 int getIdxAtMinValue(vector<int> array);
17
   int distanceBetween(int a, int b);
18
19 \// O(br) time \| O(br) space - where b is the number of blocks and r is
20 // number of requirements
21 int apartmentHunting(vector<unordered_map<string, bool>> blocks,
                        vector<string> reqs) {
23
     vector<vector<int>> minDistancesFromBlocks;
24
     for (string req : reqs) {
25
      minDistancesFromBlocks.push_back(getMinDistances(blocks, req));
26
27
     vector<int> maxDistancesAtBlocks =
28
         getMaxDistancesAtBlocks(blocks, minDistancesFromBlocks);
29
     return getIdxAtMinValue(maxDistancesAtBlocks);
30 }
31
32 vector<int> getMinDistances(vector<unordered_map<string, bool>> blocks
33
                               string req) {
```

the figure of the latest service of the late



Run or submit code when you're ready.