```swift
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

class Program {
    // O(b^2 * r) time | O(b) space
    func apartmentHunting(_ blocks: [[String: Bool]], _ requirements: [String]) -> Int {
        var maxDistancesAtBlocks = Array(repeating: -Int.max, count: blocks.count)
        for i in 0 ..< blocks.count {
            for requirement in requirements {
                var closestReqDistance = Int.max

                for j in 0 ..< blocks.count {
                    if let requirementAvailable = blocks[j][requirement], requirementAvailable {
                        closestReqDistance = min(closestReqDistance, distanceBetween(i, j))
                    }
                }

                maxDistancesAtBlocks[i] = max(maxDistancesAtBlocks[i], closestReqDistance)
            }
        }

        return getIndexAtMinValue(maxDistancesAtBlocks)
    }

    func getIndexAtMinValue(_ array: [Int]) -> Int {
        var indexAtMinValue = 0
        var minValue = Int.max

        for i in 0 ..< array.count {
            let currentValue = array[i]

            if currentValue < minValue {
                minValue = currentValue
                indexAtMinValue = i
            }
        }

        return indexAtMinValue
    }

    func distanceBetween(_ a: Int, _ b: Int) -> Int {
        return abs(a - b)
    }
}

```