

The function takes in two arguments:

- 'blocks', an array of objects representing the various blocks in a neighborhood. Each object has keys gym, school, and store, which are boolean values indicating whether the respective facilities are available in the block.
- 'reqs', an array of strings representing the required facilities.

The function returns an integer, the index of the block in the blocks array that is the optimal choice for the apartment.

The function first initializes two variables:

- 'minDistance', set to infinity, which will be used to keep track of the minimum distance from an apartment to the farthest required facility.
- 'optimalBlock', set to -1, which will be used to keep track of the optimal block with the minimum distance.

The function then iterates through each block in the blocks array. For each block, it initializes a variable 'maxDistance' to 0, which will be used to keep track of the farthest distance from the apartment to a required facility.

The function then iterates through each required facility in the 'reqs' array. If the current block has the required facility, it sets the distance variable to 0. If the current block does not have the required facility, it sets the distance variable to infinity and then searches the blocks ahead and behind the current block to find the nearest block with the required facility. It updates the distance variable with the minimum distance to the nearest block with the required facility.

The function then updates the 'maxDistance' variable with the maximum of 'maxDistance' and distance.

After iterating through all the required facilities, the function compares 'maxDistance' to 'minDistance' and updates 'minDistance' and 'optimalBlock' if necessary.

Finally, the function returns the value of 'optimalBlock' as expected.