

INTELLIGENT AUTOMATION

Prospective Developer Assessment

1. Background



On a suburban street in Sydney, a number of houses stand built in a row.

On a given week, the occupancy of each household can be in one of two states. These are either:

[1] Home, or [0] Away

Each week, the occupancy of each household may change. These changes are based on the following rules:

1. If both neighbours of a household are away or both are at home for the current week, then that household will be at **home** the following week.
2. If one neighbour is away, and the other is at home for the current week, then that household will be **away** the following week.
3. If a household only has 1 neighbour (i.e., the first or last house in the street), then it can be assumed the other "unseen" neighbour is **always away** for the current week.

2. Task Description

Write a function "calculateOccupancy" that takes an initial occupancy state for a row of houses and then returns the final occupancy state after a given number of weeks. (The interim occupancy state is not required, only the final state).

Inputs:

1. An array "initialOccupancy" containing the occupancy state of each house for that week.

initialOccupancy = [1,1,0,0,1,0,1]

2. An integer "numberOfWeeks" containing the number of weeks after which we would like to know the occupancy.

numberOfWeeks = 3

Output:

An array containing the occupancy state after the given number of weeks.

finalOccupancy = [0,1,0,0,0,1,0]

Examples:

Test Case	Input 1 - initialOccupancy	Input 2 - numberOfWeeks	Expected Output - finalOccupancy
1	[0,0,1,0,1]	1	[1,0,1,1,1]
2	[1,1,0,0,1,0,1]	3	[0,1,0,0,0,1,0]

3. Assessment Structure

Analyse the problem and outline a solution. You may use diagrams, flow charts, pseudocode or code your solution in any language of your choice.

Send you solution as an attachment to the below email:

alvaro.malmierca@smartgroup.com.au

We are interested in your approach, organisation and thought process as much as your result.