573. Squirrel Simulation

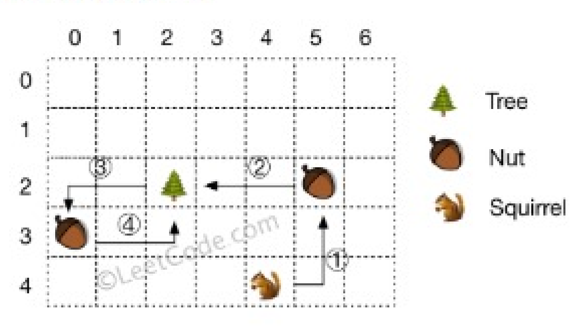
May 6, 2017 | 6.2K views

Average Rating: 5 (8 votes)

There's a tree, a squirrel, and several nuts. Positions are represented by the cells in a 2D grid. Your goal is to find the **minimal** distance for the squirrel to collect all the nuts and put them under the tree one by one. The squirrel can only take at most **one nut** at one time and can move in four directions - up, down, left and right, to the adjacent cell. The distance is represented by the number of moves.

Example 1:

```
Input:
Height : 5
Width : 7
Tree position : [2,2]
Squirrel : [4,4]
Nuts : [[3,0], [2,5]]
Output: 12
Explanation:
```



Note:

1. All given positions won't overlap.
2. The squirrel can take at most one nut at one time.
3. The given positions of nuts have no order.
4. Height and width are positive integers. 3 <= height * width <= 10,000.
5. The given positions contain at least one nut, only one tree and one squirrel.

Solution

Approach 1: Simple Solution

Algorithm

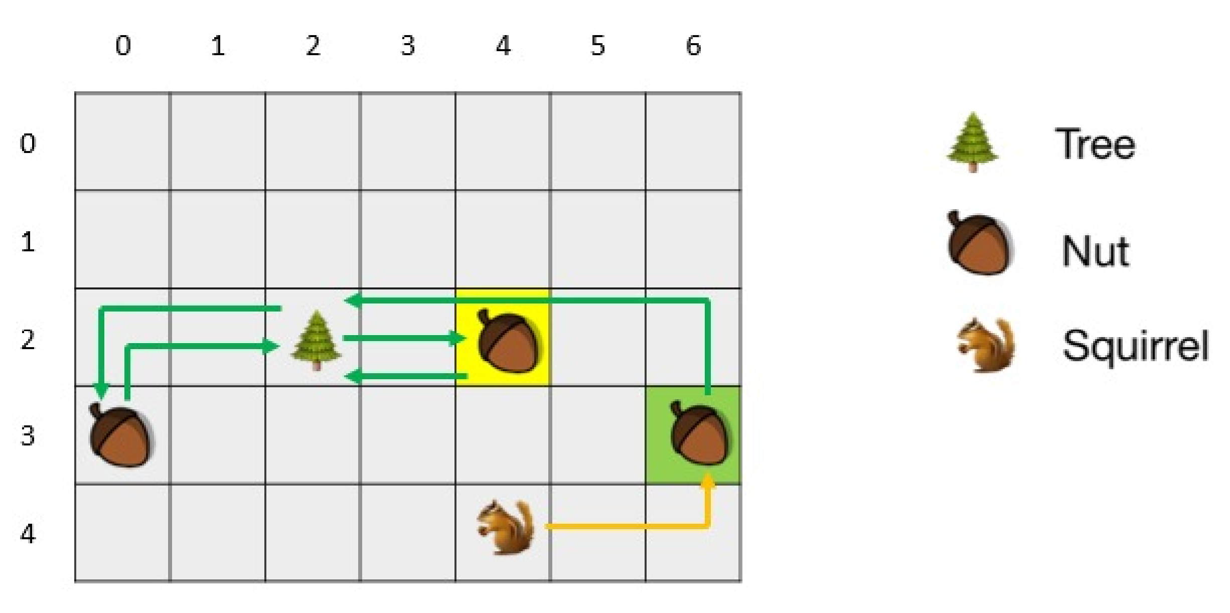
We know, the distance between any two points(tree, squirrel, nut) is given by the absolute difference between the corresponding x-coordinates and the corresponding y-coordinates.

Now, in order to determine the required minimum distance, we need to observe a few points. Firstly, the order in which the nuts are picked doesn't affect the final result, except one of the nuts which needs to be visited first from the squirrel's starting position. For the rest of the nuts, it is mandatory to go from the tree to the nut and then come back as well.

For the first visited nut, the saving obtained, given by d, is the difference between the distance between the tree and the current nut & the distance between the current nut and the squirrel. This is because for this nut, we need not travel from the tree to the nut, but need to travel an additional distance from the squirrel's original position to the nut.

While traversing over the $nuts$ array and adding the to-and-fro distance, we find out the saving, d, which can be obtained if the squirrel goes to the current nut first. Out of all the nuts, we find out the nut which maximizes the saving and then deduct this maximum saving from the sum total of the to-and-fro distance of all the nuts.

Note that the first nut to be picked needs not necessarily be the nut closest to the squirrel's start point, but it's the one which maximizes the savings.



🟨 nut is closer to the squirrel's start point but the savings are maximized if it goes to 🟩 nut first

Java

```java
public class Solution {
    public int minDistance(int height, int width, int[] tree, int[] squirrel, int[][] nuts) {
        int tot_dist = 0, d = Integer.MIN_VALUE;
        for (int[] nut: nuts) {
            tot_dist += (distance(nut, tree) * 2);
            d = Math.max(d, distance(nut, tree) - distance(nut, squirrel));
        }
        return tot_dist - d;
    }
    public int distance(int[] a, int[] b) {
        return Math.abs(a[0] - b[0]) + Math.abs(a[1] - b[1]);
    }
}
```

Complexity Analysis

- Time complexity : $O(n)$. We need to traverse over the whole $nuts$ array once. n refers to the size of $nuts$ array.
- Space complexity : $O(1)$. Constant space is used.

Analysis written by: @vinod23

Rate this article:

Previous Next

Comments: 3

Sort By

Type comment here... (Markdown is supported)

Preview Post

sha256pki 515 August 7, 2017 12:55 AM Report

Thanks,
Here is my understanding -

d1 - distance between nut 1 and tree
d2 - distance between nut 2 and tree

Read More

2 Share Reply

SHOW 1 REPLY

vinod23 425 August 7, 2017 12:43 AM Report

@sha256pki Hi

Yes, you can say that we'll achieve the maximum profit by picking up a nut which is farther from the tree but closer to the squirrel, as the first nut. This is because, the only travel distance which we can save is from the tree to the nut, but to achieve this saving, we need to go from the squirrel's start

Read More

1 Share Reply

sha256pki 515 August 6, 2017 11:58 AM

Hey Vinod,

Why is the saving d is the difference between distance(tree,nut) and distance(squirrel,nut)? So the difference is actually how farther nut is from tree compared to how farther it is from squirrel. So is the algorithm to pick the nut which is farther from tree but closer to squirrel?

Read More

0 Share Reply

SHOW 1 REPLY