

Rock Climbing

18th May 2016 Advance
Problem

Nayan Ostwal

Rock climbing

There is a man who wants to climb a rock from a starting point to the destination point. Given a map of the rock mountain which N = height, M = width. In the map, character '-' is the possible foot place spot (where can climb).

He can freely move up/down at vertical spots which '-' exists sequentially. It's **impossible** to move horizontally in case there is more than one space between '-' in the same height level.

Depending on how high/low he moves towards the upper or lower direction at one time, the level of difficulty of rock climbing gets determined.

The maximum height of moving from the starting point to destination point is the level of difficulty of rock climbing .

The total distance of movement is not important. There are more than one path from the starting point to destination point. => Output: The minimum level of difficulty of all rock climbing paths level.

Hint: Start with difficulty level 0 and then keep increasing it one by one.

Sotong

- The sample test is present in Sotong, CoBY.
- <http://sotong.sec.samsung.net/sotong/cp/cpContestProbShow.do?contestId=AVTNeGq1677VIdEa&contestProbId=AVTIOFaFGX7VIdEa>

Approaches

- DFS with recursion
- BFS can also be used with increase in code complexity
- Few people solved it with Backtracking as well (not advised though)
- If the visited array is marked as “steps_count” under consideration, then we need not initialize the visited array with 0 for every “step_count”.
- Attached is DFS solution with TC generation as well as timing calculation



rock_climbing.txt

Errors/Bugs

- Executed DFS from source and destination both at the same “step_count”.
 - This can lead to error as it can now jump $2 * \text{step_count}$.
- Did not change the visited array properly after incrementing the “step_count”
- Tried to jump only the current “step_count” in consideration.
 - You should consider all the steps from 0 to “step_count” in every loop.