

Problem 1138: Alphabet Board Path

Problem Information

Difficulty: Medium

Acceptance Rate: 51.73%

Paid Only: No

Tags: Hash Table, String

Problem Description

On an alphabet board, we start at position `(0, 0)` , corresponding to character `board[0][0]` .

Here, `board = ["abcde", "fghij", "klmno", "pqrst", "uvwxyz", "z"]` , as shown in the diagram below.

We may make the following moves:

* ``U`` moves our position up one row, if the position exists on the board; * ``D`` moves our position down one row, if the position exists on the board; * ``L`` moves our position left one column, if the position exists on the board; * ``R`` moves our position right one column, if the position exists on the board; * ``!`` adds the character `board[r][c]` at our current position `(r, c)` to the answer.

(Here, the only positions that exist on the board are positions with letters on them.)

Return a sequence of moves that makes our answer equal to `target` in the minimum number of moves. You may return any path that does so.

Example 1:

Input: target = "leet" **Output:** "DDR!UURRR!!DDD!"

Example 2:

****Input:**** target = "code" ****Output:**** "RR!D!RR!UUL!R!"

****Constraints:****

* `1 <= target.length <= 100` * `target` consists only of English lowercase letters.

Code Snippets

C++:

```
class Solution {
public:
    string alphabetBoardPath(string target) {
        }
    };
}
```

Java:

```
class Solution {
    public String alphabetBoardPath(String target) {
        }
    }
}
```

Python3:

```
class Solution:
    def alphabetBoardPath(self, target: str) -> str:
```