Problem 1263: Minimum Moves to Move a Box to Their Target Location

Problem Information

Difficulty: Hard
Acceptance Rate: 49.26%
Paid Only: No
Tags: Array, Breadth-First Search, Heap (Priority Queue), Matrix

Problem Description

A storekeeper is a game in which the player pushes boxes around in a warehouse trying to get them to target locations.

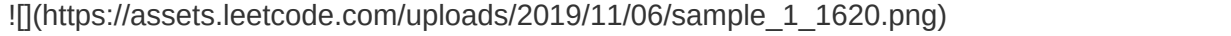
The game is represented by an `m x n` grid of characters `grid` where each element is a wall, floor, or box.

Your task is to move the box `'B'` to the target position `'T'` under the following rules:

* The character `'S'` represents the player. The player can move up, down, left, right in `grid` if it is a floor (empty cell). * The character `'.'` represents the floor which means a free cell to walk. * The character `'#'` represents the wall which means an obstacle (impossible to walk there). * There is only one box `'B'` and one target cell `'T'` in the `grid`. * The box can be moved to an adjacent free cell by standing next to the box and then moving in the direction of the box. This is a **push**. * The player cannot walk through the box.

Return _the minimum number of**pushes** to move the box to the target_. If there is no way to reach the target, return `-1`.

Example 1:



Input: grid = [["#","#","#","#","#","#"], ["#","T","#","#","#","#"], ["#",".",".","B",".","#"], ["#",".","#","#",".","#"], ["#",".",".",".","S","#"], ["#","#","#","#","#","#"]] **Output:** 3 **Explanation:** We return only the number of times the box is pushed.

Example 2:

Input: grid = [["#","#","#","#","#","#"], ["#","T","#","#","#","#"], ["#",".",".","B",".","#"], ["#","#","#","#",".","#"], ["#",".",".",".","S","#"], ["#","#","#","#","#","#"]] **Output:** -1

Example 3:

Input: grid = [["#","#","#","#","#","#"], ["#","T",".",".","#","#"], ["#",".","#","B",".","#"], ["#",".",".",".",".","#"], ["#",".",".",".","S","#"], ["#","#","#","#","#","#"]] **Output:** 5 **Explanation:** push the box down, left, left, up and up.

Constraints:

* `m == grid.length` * `n == grid[i].length` * `1 <= m, n <= 20` * `grid` contains only characters `'.'`, `'#'`, `'S'`, `'T'`, or `'B'`. * There is only one character `'S'`, `'B'`, and `'T'` in the `grid`.

Code Snippets

C++:

```
class Solution {
public:
int minPushBox(vector<vector<char>>& grid) {

}
};
```

Java:

```
class Solution {
public int minPushBox(char[][] grid) {

}
}
```

Python3:

```
class Solution:
def minPushBox(self, grid: List[List[str]]) -> int:
```