Problem 1033: Moving Stones Until Consecutive

Problem Information

Difficulty: Medium
Acceptance Rate: 51.83%
Paid Only: No
Tags: Math, Brainteaser

Problem Description

There are three stones in different positions on the X-axis. You are given three integers `a`, `b`, and `c`, the positions of the stones.

In one move, you pick up a stone at an endpoint (i.e., either the lowest or highest position stone), and move it to an unoccupied position between those endpoints. Formally, let's say the stones are currently at positions `x`, `y`, and `z` with `x < y < z`. You pick up the stone at either position `x` or position `z`, and move that stone to an integer position `k`, with `x < k < z` and `k != y`.

The game ends when you cannot make any more moves (i.e., the stones are in three consecutive positions).

Return _an integer array_`answer` _of length_`2` _where_ :

* `answer[0]` _is the minimum number of moves you can play, and_ * `answer[1]` _is the maximum number of moves you can play_.

Example 1:

Input: a = 1, b = 2, c = 5 **Output:** [1,2] **Explanation:** Move the stone from 5 to 3, or move the stone from 5 to 4 to 3.

Example 2:

Input: a = 4, b = 3, c = 2 **Output:** [0,0] **Explanation:** We cannot make any moves.

Example 3:

Input: a = 3, b = 5, c = 1 **Output:** [1,2] **Explanation:** Move the stone from 1 to 4; or move the stone from 1 to 2 to 4.

Constraints:

* `1 <= a, b, c <= 100` * `a`, `b`, and `c` have different values.

Code Snippets

C++:

```
class Solution {
public:
vector<int> numMovesStones(int a, int b, int c) {

}
};
```

Java:

```
class Solution {
public int[] numMovesStones(int a, int b, int c) {

}
}
```

Python3:

```
class Solution:
def numMovesStones(self, a: int, b: int, c: int) -> List[int]:
```