# 2037. Minimum Number of Moves to Seat Everyone

There are n seats and n students in a room. You are given an array seats of length n, where seats[i] is the position of the ith seat. You are also given the array students of length n, where students[j] is the position of the jth student.

You may perform the following move any number of times:

Increase or decrease the position of the ith student by 1 (i.e., moving the ith student from position x to x + 1 or x - 1)

Return the minimum number of moves required to move each student to a seat such that no two students are in the same seat.

Note that there may be multiple seats or students in the same position at the beginning.

## SOLUTION IN JAVA

class Solution {

public int minMovesToSeat(int[] seats, int[] students) {

Arrays.sort(seats);

Arrays.sort(students);

int moves = 0;

for (int i = 0; i < seats.length; i++) {

moves += Math.abs(seats[i] - students[i]);

}

return moves;

}

}