Experiment – 2.2

Student Name: Subhadip Patra **UID: 20BCS2543**

Branch: CSE **Section/Group: 707/B**

Subject Name: Competitive Coding II **Subject Code: 20CSP-351**

Aim: Find the Difference

Objective:

You are given two strings s and t.

String t is generated by random shuffling string s and then add one more letter at a random position.

Return the letter that was added to t.

Example 1:

Input: s = "abcd", t = "abcde"

Output: "e"

Explanation: 'e' is the letter that was added.

Example 2:

Input: s = "", t = "y"

Output: "y"

Constraints:

0 <= s.length <= 1000

t.length == s.length + 1

s and t consist of lowercase English letters.

Code:

```
class Solution {
    public char findTheDifference(String s, String t) {
        char c = 0;
        for(char cs : s.toCharArray()) c ^= cs;
        for(char ct : t.toCharArray()) c ^= ct;
        return c;
    }
}
```

Output:

Java Auto

```
class Solution {
    public char findTheDifference(String s, String t) {
        char c = 0;
        for(char cs : s.toCharArray()) c ^= cs;
        for(char ct : t.toCharArray()) c ^= ct;
        return c;
    }
}
```

Testcase Result

Accepted Runtime: 0 ms

Case 1 Case 2

Input

s =
"abcd"

t =
"abcde"

Output

"e"

Expected

"e"

Console Run Submit

Aim: Predict the Winner

Objective:

You are given an integer array nums. Two players are playing a game with this array: player 1 and player 2.
Player 1 and player 2 take turns, with player 1 starting first. Both players start the game with a score of 0. At each turn, the player takes one of the numbers from either end of the array (i.e., nums[0] or nums[nums.length - 1]) which reduces the size of the array by 1. The player adds the chosen number to their score. The game ends when there are no more elements in the array.
Return true if Player 1 can win the game. If the scores of both players are equal, then player 1 is still the winner, and you should also return true. You may assume that both players are playing optimally.

Example 1:
Input: nums = [1,5,2]
Output: false
Explanation: Initially, player 1 can choose between 1 and 2.
If he chooses 2 (or 1), then player 2 can choose from 1 (or 2) and 5. If player 2 chooses 5, then player 1 will be left with 1 (or 2).
So, final score of player 1 is 1 + 2 = 3, and player 2 is 5.
Hence, player 1 will never be the winner and you need to return false.

Example 2:
Input: nums = [1,5,233,7]
Output: true
Explanation: Player 1 first chooses 1. Then player 2 has to choose between 5 and 7. No matter which number player 2 choose, player 1 can choose 233.
Finally, player 1 has more score (234) than player 2 (12), so you need to return True representing player1 can win.

Constraints:

1 <= nums.length <= 20
0 <= nums[i] <= 107

Code:

```
public class Solution {
    public boolean PredictTheWinner(int[] nums) {
        if(nums.length == 1) return true;
        int su = 0;
        for (int i: nums) su += i;
        int res = ans(nums, 0, nums.length - 1);
        if (res < (su - res)) return false;
        return true;
    }
    public static int ans(int[] nums, int left, int right) {
        if (left > right) return 0;
        int choice1 = nums[left] + Math.min(ans(nums, left + 2, right),
ans(nums, left + 1, right - 1));
        int choice2 = nums[right] + Math.min(ans(nums, left + 1, right - 1),
ans(nums, left, right - 2));
        return Math.max(choice1, choice2);
    }
}
```

Output:

Java Auto

```java
public class Solution {
    public boolean PredictTheWinner(int[] nums) {
        if(nums.length == 1) return true;
        int su = 0;
        for (int i: nums) su += i;
        int res = ans(nums, 0, nums.length - 1);
        if (res < (su - res)) return false;
        return true;
    }
    public static int ans(int[] nums, int left, int right) {
        if (left > right) return 0;
        int choice1 = nums[left] + Math.min(ans(nums, left + 2, right), ans(nums, left + 1, right - 1));
        int choice2 = nums[right] + Math.min(ans(nums, left + 1, right - 1), ans(nums, left, right - 2));
        return Math.max(choice1, choice2);
    }
}
```

Testcase Result

Accepted Runtime: 0 ms

• Case 1 • Case 2

Input

nums =
[1,5,2]

Output

false

Expected

false

Console Run Submit