Sum of odd numbers in array using Recursion

Problem Statement:

Write a recursive function sum(n) that calculates the sum of all odd numbers in an array arr up to index n.

Example:

Input: [5, 2, 6, 1, 3]

Odd Numbers: 5, 1, 3

Output: 9

Concepts:

Recursion: Repeatedly check whether `arr[n]` is **odd**, and add it only if true `true`.

Base Case: If n == 0, `return arr[0]` if it's **odd**, otherwise `0`.

Recursive Case: `Return (arr[n] if odd) + sum(n - 1)`.

Approach:

Check if `arr[n]` is **odd**.

If **yes**, add it to `recursive result` of `sum(n-1)`.

Else, skip it and continue **recursion**.

Time & Space Complexity:

Time Complexity: O(n)

Space Complexity: O(n) `recursive call stack`

JavaScript Python Java C++ C C#

```
let arr = [5, 2, 6, 1, 3];

function sum(n) {
    let isOdd = arr[n] % 2 !== 0;
    if (n === 0) return isOdd ? arr[0] : 0;
    return (isOdd ? arr[n] : 0) + sum(n - 1);
}
console.log(sum(arr.length - 1)); // Output: 9
```

Video Course Discuss doubts Contribute Certificate

Sum of odd numbers in array using Recursion - DSA Notes

Sum of odd numbers in array using Recursion