Problem 1982: Find Array Given Subset Sums

Problem Information

Difficulty: Hard
Acceptance Rate: 49.38%
Paid Only: No
Tags: Array, Divide and Conquer

Problem Description

You are given an integer `n` representing the length of an unknown array that you are trying to recover. You are also given an array `sums` containing the values of all `2n` **subset sums** of the unknown array (in no particular order).

Return _the array_`ans` _of length_`n` _representing the unknown array. If**multiple** answers exist, return **any** of them_.

An array `sub` is a **subset** of an array `arr` if `sub` can be obtained from `arr` by deleting some (possibly zero or all) elements of `arr`. The sum of the elements in `sub` is one possible **subset sum** of `arr`. The sum of an empty array is considered to be `0`.

Note: Test cases are generated such that there will **always** be at least one correct answer.

Example 1:

Input: n = 3, sums = [-3,-2,-1,0,0,1,2,3] **Output:** [1,2,-3] **Explanation:**[1,2,-3] is able to achieve the given subset sums: - []: sum is 0 - [1]: sum is 1 - [2]: sum is 2 - [1,2]: sum is 3 - [-3]: sum is -3 - [1,-3]: sum is -2 - [2,-3]: sum is -1 - [1,2,-3]: sum is 0 Note that any permutation of [1,2,-3] and also any permutation of [-1,-2,3] will also be accepted.

Example 2:

Input: n = 2, sums = [0,0,0,0] **Output:** [0,0] **Explanation:** The only correct answer is [0,0].

Example 3:

Input: n = 4, sums = [0,0,5,5,4,-1,4,9,9,-1,4,3,4,8,3,8] **Output:** [0,-1,4,5]
Explanation: [0,-1,4,5] is able to achieve the given subset sums.

Constraints:

* `1 <= n <= 15` * `sums.length == 2n` * `-104 <= sums[i] <= 104`

Code Snippets

C++:

```
class Solution {
public:
vector<int> recoverArray(int n, vector<int>& sums) {

}
};
```

Java:

```
class Solution {
public int[] recoverArray(int n, int[] sums) {

}
}
```

Python3:

```
class Solution:
def recoverArray(self, n: int, sums: List[int]) -> List[int]:
```