Tug of War

Given a set of n integers, divide the set in two subsets of n/2 sizes each such that the difference of the sum of two subsets is as minimum as possible. If n is even, then sizes of two subsets must be strictly n/2 and if n is odd, then size of one subset must be (n-1)/2 and size of other subset must be (n+1)/2. For example, let given set be {3, 4, 5, -3, 100, 1, 89, 54, 23, 20}, the size of set is 10. Output for this set should be {4, 100, 1, 23, 20} and {3, 5, -3, 89, 54}. Both output subsets are of size 5 and sum of elements in both subsets is same (148 and 148).

Let us consider another example where n is odd. Let given set be {23, 45, -34, 12, 0, 98, -99, 4, 189, -1, 4}. The output subsets should be {45, -34, 12, 98, -1} and {23, 0, -99, 4, 189, 4}. The sums of elements in two subsets are 120 and 121 respectively.

The following solution tries every possible subset of half size. If one subset of half size is formed, the remaining elements form the other subset. We initialize current set as empty and one by one build it. There are two possibilities for every element, either it is part of current set, or it is part of the remaining elements (other subset). We consider both possibilities for every element. When the size of current set becomes n/2, we check whether this solutions is better than the best solution available so far. If it is, then we update the best solution.

```python
import sys


def tugOfWar(arr):
    idx = 0
    set1 = []
    set2 = []
    sum1 = 0
    sum2 = 0
    minDiff = [sys.maxsize]
    ans = [0,0]
    tugOfWarUtil(arr, idx, set1, set2, sum1, sum2, minDiff, ans)
    return ans


def tugOfWarUtil(arr, idx, set1, set2, sum1, sum2, minDiff, ans):
    if idx == len(arr):
        if abs(sum1 - sum2) < minDiff[0]:
            minDiff[0] = abs(sum1 - sum2)
            temp1 = []
            for el in set1:
                temp1.append(el)
            # ans.append(temp1)
```

```python
            temp2 = []
            for el in set2:
                temp2.append(el)
            ans[0] = temp1
            ans[1] = temp2
        return

    if len(set1) < (len(arr) + 1) // 2:
        set1.append(arr[idx])
        tugOfWarUtil(arr, idx + 1, set1, set2, sum1 + arr[idx], sum2,
minDiff, ans)
        set1.pop()

    if len(set2) < (len(arr) + 1) // 2:
        set2.append(arr[idx])
        tugOfWarUtil(arr, idx + 1, set1, set2, sum1, sum2 + arr[idx],
minDiff, ans)
        set2.pop()


arr = [23, 45, -34, 12, 0, 98, -99, 4, 189, -1, 4]
print(tugOfWar(arr))
```

Ans : [[23, 0, -99, 4, 189, 4], [45, -34, 12, 98, -1]]