1497. Check If Array Pairs Are Divisible by k

Given an array of integers `arr` of even length `n` and an integer `k`.

We want to divide the array into exactly `n / 2` pairs such that the sum of each pair is divisible by `k`.

Return *True* If you can find a way to do that or *False* otherwise.

Example 1:

```
Input: arr = [1,2,3,4,5,10,6,7,8,9], k = 5
Output: true
Explanation: Pairs are (1,9),(2,8),(3,7),(4,6) and (5,10).
```

Example 2:

```
Input: arr = [1,2,3,4,5,6], k = 7
Output: true
Explanation: Pairs are (1,6),(2,5) and(3,4).
```

Example 3:

```
Input: arr = [1,2,3,4,5,6], k = 10
Output: false
Explanation: You can try all possible pairs to see that there is no way to
divide arr into 3 pairs each with sum divisible by 10.
```

Example 4:

```
Input: arr = [-10,10], k = 2
Output: true
```

Example 5:

```
Input: arr = [-1,1,-2,2,-3,3,-4,4], k = 3
Output: true
```

```
def canArrange(self, arr: List[int], k: int) -> bool:
        freq = collections.defaultdict(int)
        for ele in arr:
            rem = ele%k
```

```
            if rem in freq:
                freq[rem] = freq[rem]+1
            else:
                freq[rem] = 1

        for ele in arr:
            rem = ele%k
            if rem==0:
                if freq[rem]%2!=0:
                    return False
            elif 2*rem==k:
                if freq[rem]%2!=0:
                    return False
            else:
                if freq[k-rem]!=freq[rem]:
                    return False
        return True
```