target sum pair (consider duplicate pair)

target = 50

40 20 30 10 -5 60 55 10 40

T : nlogn

-5 10 10 20 30 40 40 55 60

j i

-5, 55

10, 40

10, 40

20, 30

```
if (A[i] + A[j] == tar) {
        syso (A[i] + A[j]);
        i++ ; j--;
}
else if(A[i] + A[j] < tar) {
        i++;
}
else {
        j--;
}
```

2 Sum - Target Sum Unique Pairs

40 20 30 10 -5 60 55 10 40

target = 50

sort: -5 10 10 20 30 40 40 55 65

j i

-5, 55

10, 40

20, 30

```
if (i!=0 && A[i] == A[i-1]) {
    i++;
    continue;
}
```

ts – unique pairs

```
while(i < j) {
    //cosider unique pairs only
    if(i != 0 && arr[i] == arr[i-1]) {
        i++;
        continue;
    }

    if(arr[i] + arr[j] == target) {
        ArrayList<Integer>res = new ArrayList<>();
        res.add(arr[i]);
        res.add(arr[j]);
        ans.add(res);

        i++;
        j--;
    }
    else if(arr[i] + arr[j] < target) {
        i++;
    }
    else {
        j--;
    }
}
```

10, 40

12, 38

20, 30

arr : 20 30 60 10 12 38 40

10 20 50 38 30

target = 50

10 10 12 20 20 30 30 38 38 40 50 60

j

i

3 Sum - Target Sum Unique Triplet

target : 50

arr : 20 10 10 15 10 5 15 20 40 -10

40 20 30 20

-10, 20, 40

5, 15, 30

10, 10, 30

10, 20, 20

k = 40

sort : -10 5 10 10 10 15 15 20 20 20 20 30 40 40

i

18. 4Sum - target sum unique quadrapels

```
[1,0,-1,0,-2,2]
```

target = 0

0 - (-2) = 2

-2 -1 0 0 1 2

i

-2 0 0 2

-2 -1 1 2

K Sum - Target Sum Unique Set

```java
public static List<List<Integer>> KSum(int[]arr,int target,int k,int si) {
    if(k == 2) {
        return tsUniquePairs(arr,si,target);
    }

    List<List<Integer>>ans = new ArrayList<>();

    for(int i = si; i < arr.length - k + 1;i++) {
        if(i != si && arr[i] == arr[i-1]) {
            continue;
        }

        List<List<Integer>>res = KSum(arr,target-arr[i],k-1,i+1);

        for(List<Integer>list : res) {
            list.add(arr[i]);
            ans.add(list);
        }
    }

    return ans;
}
```

k = 4

0 10 10 10 20 30 30 40