```java
while(l<r){
    long currSum = (long)(arr[l] + arr[r]);

    if(currSum == remainingTarget){
        List<Integer> currQuad = new ArrayList<>();
        currQuad.add(arr[i]);
        currQuad.add(arr[j]);
        currQuad.add(arr[l]);
        currQuad.add(arr[r]);
        res.add(currQuad);

        int prevL = arr[l];
        while(l<r && arr[l] == prevL) l++;

        int prevR = arr[r];
        while(r>l && arr[r] == prevR) r--;

    }else if(currSum > remainingTarget){
        int prevR = arr[r];
        while(r>l && arr[r] == prevR) r--;//decrement r until it comes onto new value
    }else if(currSum < remainingTarget){
        int prevL = arr[l];
        while(l<r && arr[l] == prevL) l++;
    }
}
```

Meeting Chat

Anurag Vishwakarma to Everyone 5:25 PM

i j l r

target 16

At ⇒ 13

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

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

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

[0,0,0,0,0,0,0]

[0, 0, 0, 0, 0, 0, 0]

(0,0,0,0)

2 4 3 9 6 2 3

2.0 4+2=6 2=6-4

235115030

l r

1 1 3 3 4 5

mx=2

K=100 0 1 2 3 sp

10 5 2 5 ep

10

10 5

5

5 2

5 2 5

2

2 5

5

prod = 50

{10}

{10 5}

5

{5 2}

2

5 2 5

2 5

5

101 001 10001 11100 11 sp ep

while(ep<n){

if(arr[ep]==0) zc++ sp → ep → ans = 3

K = 2

while(zc>K){

if arr[sp]==0) zc--; zc = 2

sp++

}

ans = max(ans, ep-sp+1);

ep++

}

add

invalid

shrink

consider ans

expand

introduce new person

shrink until invalid

consider the window for ans

expand