Brute:

def team(skill: [int], n: int) -> int:

    count=0

    for i in range(n):

        for j in range(i+1,n):

            if skill[i]>2\*skill[j]:

                count+=1

    return count

Optimal:

def merge(skill,s,e,mid):

    i=s

    j=mid+1

    count=0

    while(i<=mid and j<=e):

        if skill[i]>2\*skill[j]:

            count+=mid-i+1

            j+=1

        else:

            i+=1

    arr3=[]

    i=s

    j=mid+1

    while(i<=mid and j<=e):

        if skill[i]<=skill[j]:

            arr3.append(skill[i])

            i+=1

        else:

            arr3.append(skill[j])

            j+=1

    while(i<=mid):

        arr3.append(skill[i])

        i+=1

    while(j<=e):

        arr3.append(skill[j])

        j+=1

    for i in range(s,e+1):

        skill[i]=arr3[i-s]

    return count

def mergesort(skill,s,e):

    cnt=0

    if s>=e:

        return cnt

    mid=(s+e)//2

    cnt+=mergesort(skill,s,mid)

    cnt+=mergesort(skill,mid+1,e)

    cnt+=merge(skill,s,e,mid)

    return cnt

def team(skill: [int], n: int) -> int:

    # Write your code here.

    return mergesort(skill,0,n-1)