```
T.C.  o(nlog(n))
S.C.  o(1)
```

Leetcode
Daily Challenge

24/03/2022

problem
Boats to Save People

pre-requisites
sorting, 2 pointer

difficulty
Medium

est. time
10-15 min

Let's build Intuition

can be asked in...



50%
Accuracy

Statement

Description

- You are given an array people where people[i] is the weight of the ith person, and an infinite number of boats where each boat can carry a maximum weight of limit. Each boat carries at most two people at the same time, provided the sum of the weight of those people is at most limit.
- Return the minimum number of boats to carry every given person.

i/p	o/p
people = [3,2,2,1], limit = 3	3

explanation

3 boats (1, 2), (2) and (3)

Observation

What are we given

- people[] (weights of n people)
- limit (limit of boat)
- we can put at max 2 people i,j in boat if people[i]+people[j] <= limit
- return min boats required to carry all of them

Let's solve test cases...

consider this ex.

people[] = {3,8,1,6} limit = 9

min boats required = 2 ({3,6}, {1,8})

I would encourage you to solve few more test cases before jumping to any solid conclusion

from above test case let's get some observation

next slide ---------->

Observation

people[] = {3,8,1,6} limit = 9 test case #1

there can be 3 ways with which we can pair 2 people in a boat

1. 2 heavy weight people
2. 2 light weight people
3. 1 heavy + 1 light

1 2 heavy weight people

their combined weight can exceed boat limit.

from test case #1 8 + 6 > limit (14 > 9)

2 2 light weight people

their combined weight may be way less than boat limit & thus their would be vacant limit that we are just wasting(which we could have used).

from test case #1 3 + 1 < limit (4 < 9)

3 1 heavy + 1 light weight people optimal

their combined weight would be closest to fill the max boat limit, hence the wastage of boat limit can be avoided.

from test case #1 1 + 8 <= limit (9 <= 9)

Observation

from previous example we concluded to pair

1 light + 1 heavy weight person.

how can we get 1 light + 1 heavy person ?

by SORTING

we can sort the people[] in asc. so left ones are light weight & right ones are heavy

now we want 1 person from left & 1 from right, what we can use ? 2 POINTER

1 pointer starts from extreme left & other from extreme right

final conclusion

we will sort the people[] array & use 2 pointer technique to traverse the array

Algortihm

people[] = {3,8,1,6,4} limit = 9

int cnt = 0 (cnt = count of boats required)

- sort people[]

  people[] = {1,3,4,6,8}

- take 2 pointers

  l = 0, h = n-1 (n = people.size())

- if(people[l] + people[h] <= limit)

  it means we can put these 2 people in a boat so

  cnt++;

  l++;

  h--;

- else

  it means their weight exceeds boat limit so heavy person(h) can't be paired with anyone, so he will occupy separate boat but light one can be paired so only cnt++;

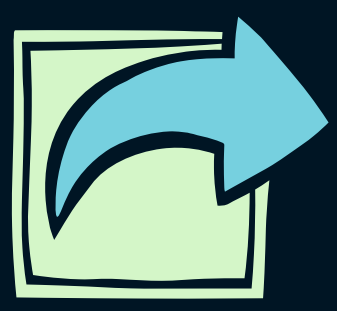
  h--;

  see here we haven't done l++ as l(th) person(light weight) still has chance to be paired up with someone, & we want to pair as many as possible to use least boats

```cpp
class Solution {
public:
    int numRescueBoats(vector<int>& people, int limit) {

        sort(people.begin(), people.end());

        int cnt = 0;
        int l = 0;
        int h = people.size() - 1;

        while(l <= h) {

            if(people[l] + people[h] <= limit) {
                l++;
            }

            h--;
            cnt++;
        }

        return cnt;
    }
};
```

Leave a Like

Comment if you love posts like this, will motivate me to make posts like these



Share, with friends