Term Project

ALGORITHMS DESIGN

Ravee Virojsirasak 6215204

TOPIC

Leetcode 881.
Boats to Save People

Difficulty : Medium

Problem

You are given an array people where people[i] is the weight of the i^{th} 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.

Given Example

Example 1

Example 1:

```
Input: people = [1,2], limit = 3
Output: 1
Explanation: 1 boat (1, 2)
```

Person 1 and 2's weight added together added doesn't go over limit which is 3. So the minimum boat is 1

Example 2

Example 2:

```
Input: people = [3,2,2,1], limit = 3
Output: 3
Explanation: 3 boats (1, 2), (2) and (3)
```

Only person weight 2 and 1 add together doesn't go over 3 (limit). So there will be 3 boats as other two person will get their won boats

Example 3

Example 3:

```
Input: people = [3,5,3,4], limit = 5
Output: 4
Explanation: 4 boats (3), (3), (4), (5)
```

There is no weights from input that is added together and doesn't go over 5 (limit). So, each person will het there own boats which the answer is 4

Analysis 1

What problem state?

01

Given Variable

- Array of people weight
- Limit weight of one boat

02

Limitation

- One boat cannot carry more than 2 people at the same time
- 2 people weight added together must not exceed Limit weight

Analysis 2

What should we do?

01

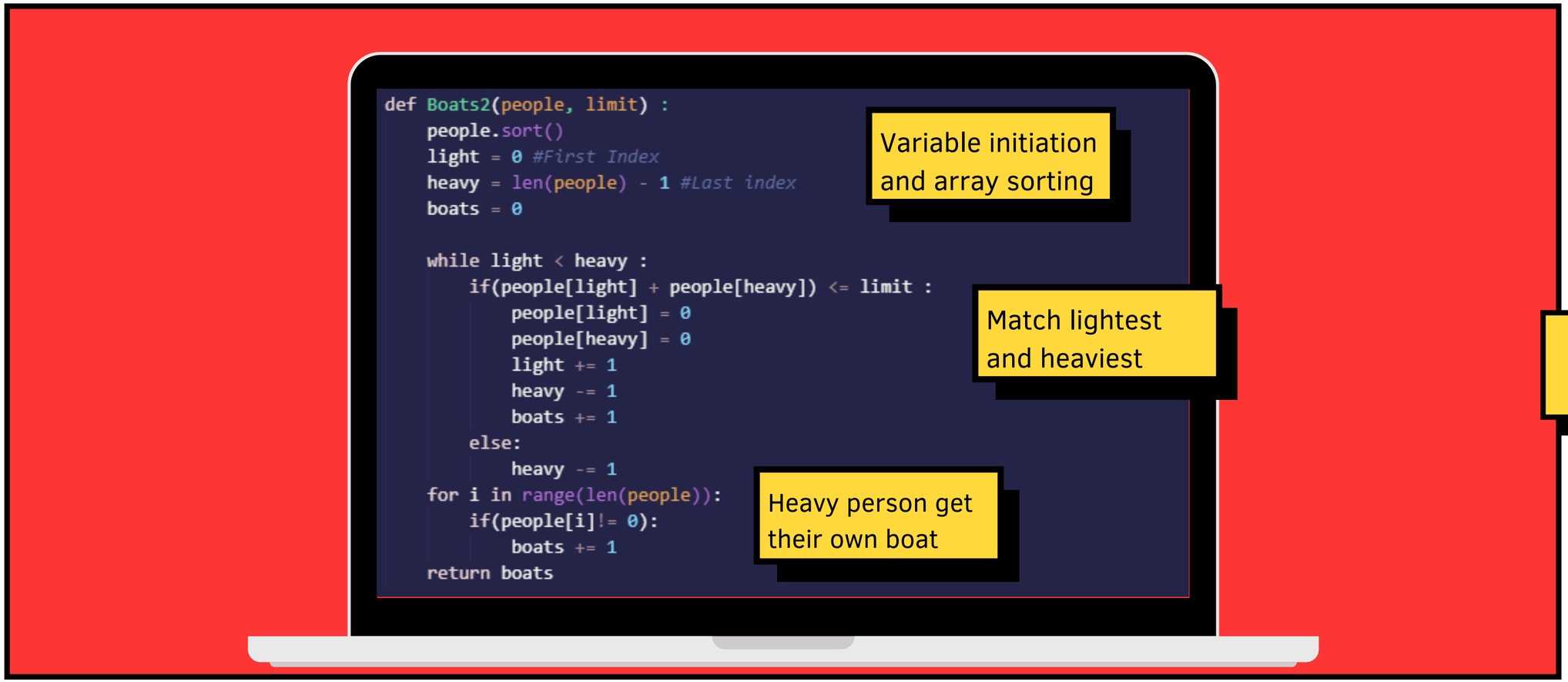
- The lightest person will go with the heaviest person.
- Or lightest person go with who have heavier weight but not heaviest.

02

- Add their weight added together
- It must not exceed the limit.

03

If that is not possible, then the heaviest person will get their own boat.



First Approach

- People = 0 in order to count any leftover person with weight that cannot go with anyone (Calculation done in for loop)

First Approach Submission

Runtime = 460 ms, Memory = 19.6 MB

Success Details ›

Runtime: 460 ms, faster than 50.97% of Python3 online submissions for Boats to Save People.

Memory Usage: 19.6 MB, less than 97.56% of Python3 online submissions for Boats to Save People.

Time Submitted	Status	Runtime	Memory	Language
09/21/2021 20:30	Accepted	460 ms	19.6 MB	python3

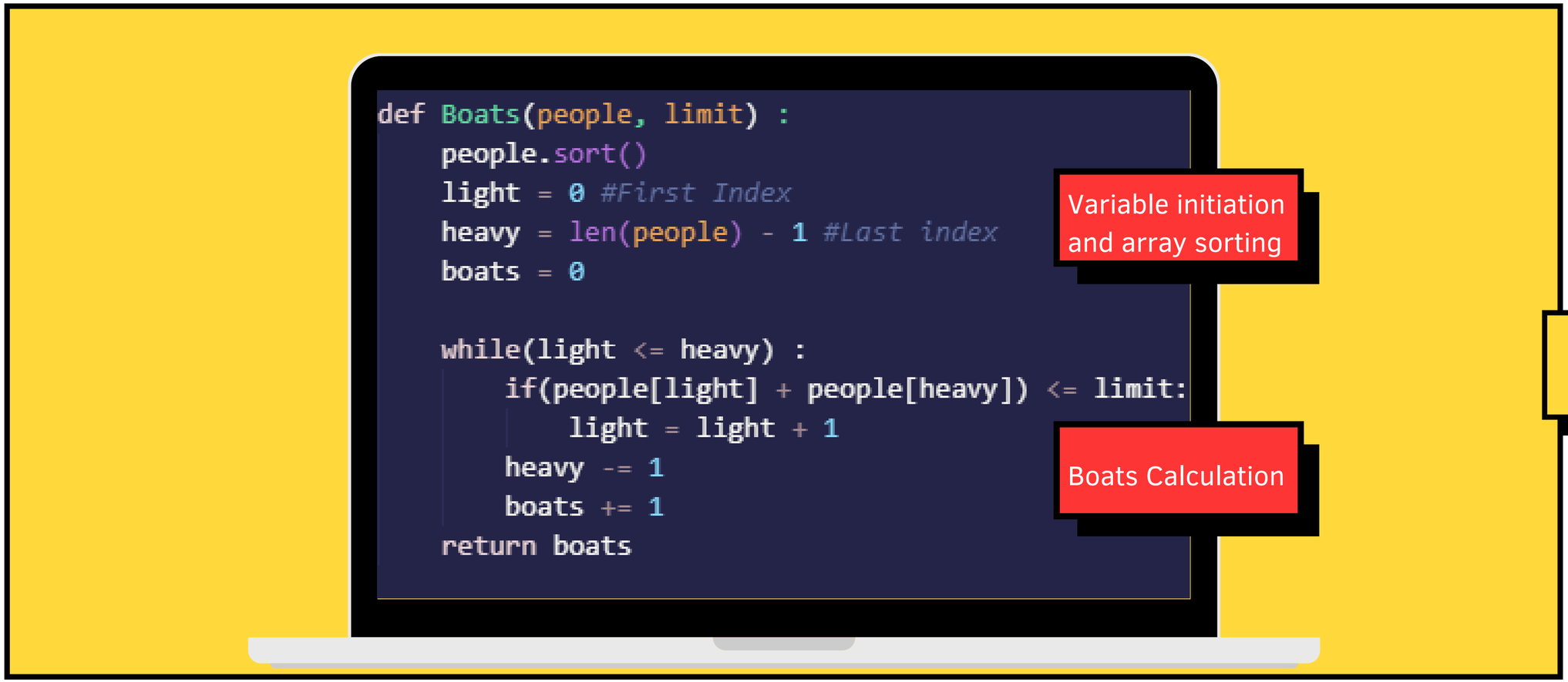
Reduce Runtime

```python
def Boats2(people, limit) :
    people.sort()
    light = 0 #First Index
    heavy = len(people) - 1 #Last index
    boats = 0

    while light < heavy :
        if(people[light] + people[heavy]) <= limit :
            people[light] = 0
            people[heavy] = 0
            light += 1
            heavy -= 1
            boats += 1
        else:
            heavy -= 1
    for i in range(len(people)):
        if(people[i]!= 0):
            boats += 1
    return boats
```

Making changes

- Changing light < heavy to light <= heavy to calculate the last person as well
- delete else statement and move indent of heavy -= 1 and boats += 1
- delete the for loop



Second Approach

If the heaviest person add with lighest person is greater than the limit, their get their own boat, and the index move down to second heaviest.
(The weight are sorted in ascending order)

Second Approach Submission

Runtime = 448 ms, Memory = 20.9 MB

Success Details ›

Runtime: 448 ms, faster than 72.09% of Python3 online submissions for Boats to Save People.

Memory Usage: 20.9 MB, less than 93.00% of Python3 online submissions for Boats to Save People.

Time Submitted	Status	Runtime	Memory	Language
09/22/2021 11:32	Accepted	448 ms	20.9 MB	python3

THANK YOU!

Ravee Virojsirasak 6215204

References

01 **Greedy Algorithms**

https://www.geeksforgeeks.org/greedy-algorithms/

02 **Leetcode 881**

https://leetcode.com/problems/boats-to-save-people/

03 **Github**

https://github.com/grandyang/leetcode/issues/881