When More Is Less: Why 100 Lines Can Beat 2

Do you think that is it always the length of code which determines how efficient it is? Let us bust that myth with an example which explains how a well thought and logical lengthy code can beat a traditional, 2 liner code in terms of efficiency.

## Problem:

You are given 9 balls and one of those balls is heavier than the rest of the group. Your task is to find out the odd ball in the least possible comparisons.

Arr = [8, 8, 8, 8, 8, 8, 8, 9, 8]

## Solution 1 (Short code):

The traditional way to solve this problem is by comparing the weights of all the balls starting from the first ball and finding out which ball has a higher weight.  
The below code snippet demonstrates this solution.  
  
//Code Snippet

As you can see this is a small code with just few lines to write but when we dig deep into the working of this code, you’ll find out that in the worst case scenario this code will iterate through all the balls to return the answer, resulting in a total of **8** comparisons.

## Solution 2 (Lengthy Code):

To find the heavier ball we can divide the 9 balls into 3 groups with 3 balls in each group.

Now, we can determine the sum of all the balls within each group:

G1: arr[0] + arr[1] + arr[2];

G2: arr[3] + arr[4] + arr[5];

G3: arr[6] + arr[7]+ arr[8];

Next, compare the sum in G1 and G2 (group 1 and group 2).

If G1 = G2, this means that there is no heavy ball in these groups and hence our heavy ball is in G3. In this way with just 1 comparison, we’ve eliminated 6 balls out of the total 9 balls.

Now, we can compare the remaining 3 balls from G3

If arr[6] > arr[7] => arr[6] is the heavy ball

Else if arr[6] < arr[7] => arr[7] is the heavy ball

Else arr[8] is the heavy ball

If you pay attention and count the number of comparisons in the above code you’ll find that the code only has 3 comparisons at max for any given case.

## Conclusion:

This example explains that how a code which appears to be lengthy and time consuming might actually be very efficient and faster that a code which appears to be smaller.

Click [here](https://code-efficiency-time-complexity.hashnode.dev/when-more-is-less-why-100-lines-can-beat-2) to read the article on hashnode.