

## Task 8 : Fake Box

### 8.1 Problem Description

If you have 50 boxes that contains 50 pieces of metal all of the same known weight. one of these boxes contains fake metal pieces that weigh 1 kilogram less than the pieces in the rest of the boxes. You can use a digital scale only once to find this fake box.

Design a brute force algorithm to solve this problem.

### 8.2 Detailed Assumptions

1. If we assume that each real box weigh [ a kilogram ] , then fake box weigh [ a-1 kilograms ]
2. Then each real metal piece in real boxes weight  $\frac{a}{50}$
3. Then each real metal piece in real boxes weight  $\frac{a}{50}$

### 8.3 Detailed Solution

#### 8.3.1 Description of Solution

1. we give each box a unique number from 1 to 50
2. we take a number of metal pieces from each box depending on box's number  
Example: 1 piece of box 1 and 2 pieces of box 2 and so on until we take 50 pieces of box 50
3. we weigh the gathered pieces and store the result
4. by computing 50 probabilities of expected results and storing them
5. comparing the result the digital scale with computed results, we find the fake box

### 8.3.2 Pseudo Code

findFakeBox(array boxes,float digitalScaleReading ,double realMetalPieceWeigt, double fakeMetalPieceWeigt)

```
// boxes: array of number of pieces taken from each box
float sum = 0;
arrayOfProbablities[50]
for i <- 1 to 50
    sum = 0;
    for j <- 0 to 49
        if j+1 == i
            sum += boxes[j] * fakeMetalPieceWeigt;
        else
            sum += boxes[j] * realMetalPieceWeigt;
    endfor
    arrayOfProbablities[i-1] = sum;
endfor
for i <- 0 to 49
    if digitalScaleReading == arrayOfProbablities[i]
        return i+1;
endfor
```

### 8.4 Complexity Analysis for the Algorithm

$$T(n) = \sum_{1 \leq i \leq 50} (\sum_{0 \leq j \leq 49} 1) + \sum_{0 \leq i \leq 49} 1 = O(1)$$

### 8.5 Sample Output of the Solution

```
Enter Real box weight: 50
Enter digital Scale Reading: 1274.8
fake box is box number 10
```

```
Enter Real box weight: 50
Enter digital Scale Reading: 1274.26
fake box is box number 37
```

### 8.6 Comparison with Another Algorithm

TABLE 1

	Brute Force	Divide & Conquer
Time Complexity	$O(n^2)$	$O(n^2)$
Space Complexity	$O(1)$	$O(1)$
Readability	Easy	Easy

### 8.7 Conclusion

In conclusion, we used two approaches to solve the problem

The brute force approach compare the scaled weight with each possibility of fake box number

On the other hand, the divide & conquer approach sort the possibilities and use binary search to find the fake box

But regardless of algorithms used all of them will lead to same efficiency of Brute force so brute force is better because of it is simplicity