Nearest Diet Challenge - Problem and Solution

Gautham Kolluru

27-01-2019

Problem Statement:

There is a family that eats biscuits for meal and have their sitting arrangements depend on the number of biscuits, like, A person 'a' of the family will be supposedly seated on the left of the person 'b' if he/she ate lesser number of biscuits than person 'b' and to the right of person 'b' if he/she ate more number of biscuits than person 'b'.

Contd. . . :

Suppose if a new member 'z' comes into the family who ate 'X' number of biscuits, find out the existing family member with whom the new member 'z' will be sitting.

Understand the problem:

What to find out is, the person with most minimal difference between the existing family member.

Approach:

- ▶ You'll be given a list / array of integers containing the number of biscuits ate by each family member and the count of biscuits ate by the new member.
- ▶ Create a list / array to contain the differences between the new member's count with the list / array of integers
- ▶ Find the most minimal number of the differences list
- ▶ Find out the index of the most minimal number of the differences list and the element corresponding to that index from the given list of integers.

Solution in Python:

```python
def nearestDiet(arr, nm):
    difference_arr = [abs(new_mem - i) for i in arr]
    return arr[difference_arr.index(min(difference_arr))]


arr = [100,50,150,250,20,30,130]

new_mem = 140

print(nearestDiet(arr, new_mem))
```

Solution in Java:

```java
import java.math.*;
public class NearestDiet {
    public static void main(String[] args) {
        int[] arr1 = {100,50,150,20,30,250,130};
        int new_mem = 140;
        nearestDiet(arr1, new_mem);
    }
    public static void nearestDiet(int[] arr, int nm) {
        int min = Math.abs(nm - arr[0]), min_ind = 0;
        for (int i = 0; i < arr.length; i++) {
            if (Math.abs(nm - arr[i]) < min) {
                min = Math.abs(nm - arr[i]);
                min_ind = i;
            }
        }
        System.out.println(arr[min_ind]);
    }
}
```