Short Project – Hashing

Problem 2) Solution :

MostFrequentNumber.java implements two methods to find the most frequently occurring element in an array.

Method 1 : mostFrequentUsingMap(int[] array)

Finds the most frequently occurring element in the array using HashMap

HashMap is used to store the array elements in key value pair such that the array element is the key and the count of that number is the value. The hashmap is then sorted and the key value with maximum value is returned by the function.

The Time Complexity in this case is O(n)

Method 2 : mostFrequentUsingArraySort(int[] array)

Array.sort() method is used to sort the array and then the most frequent element is found by traversing the sorted array.

The Time Complexity in this case id O(nlogn)

Input :

The input array size was varied from 10 elements to 100000000 elements. The input array was initialized with random values between 1-99

The observations are as below :

Array size : 10

Output :

```
*** Arary.sort() implementation ***
Max frequent element by Array Sort is : 65
Time Required by Array.sort() : 1 millisecs
*** Map implementation ***
Most frequent element by Map is : 65
Time Required by Map implementation : 1 millisecs
```

Array size : 10000

```
*** Arary.sort() implementation ***
Max frequent element by Array Sort is : 52
Time Required by Array.sort() : 6 millisecs
*** Map implementation ***
Most frequent element by Map is : 52
Time Required by Map implementation : 7 millisecs
```

Array size : 1000000

```
*** Arary.sort() implementation ***
Max frequent element by Array Sort is : 10
Time Required by Array.sort() : 98 millisecs
*** Map implementation ***
Most frequent element by Map is : 10
Time Required by Map implementation : 48 millisecs
```

Array size : 100000000

```
*** Arary.sort() implementation ***
Max frequent element by Array Sort is : 36
Time Required by Array.sort() : 4015 millisecs
*** Map implementation ***
Most frequent element by Map is : 36
Time Required by Map implementation : 1993 millisecs
```

Observations:

From the outputs it can be seen that, as the size of array goes on increasing, the performance of HashMap is better than Array.sort(). For small size arrays the performance of Array.sort() is better than HashMap.

On repeated running the program over different sizes of the input array, it was observed that for array.lenght > 300000,the performance of HashMap was better than Array.sort().