Lecture 3 : Hashmaps
Deadline
Oct 5, 2020, 11:59 PM

Lecture

0.0%

Score 0/200

- Introduction to Hashmaps
- Inbuilt Hashmap
- Remove Duplicates
- Code : Maximum Frequency Number 0/40
- Code : Print Intersection 0/80
- Code : Pair Sum to 0 0/80
- Iterators
- Bucket Array and hash function
- Collision Handling
- Hashmap Implementation - Insert
- Hashmap Implementation - Delete a...
- Time complexity and Load factor
- Rehashing

Assignment

You need at least 80% in Lecture 3 : Hashmaps (Lecture) to unlock this

Problem | Result

Code : Print Intersection

Send Feedback

Given two random integer arrays, print their intersection. That is, print all the elements that are present in both the given arrays.

Input arrays can contain duplicate elements.

Note : Order of elements are not important
Input format :

Line 1 : Integer N, Array 1 Size
Line 2 : Array 1 elements (separated by space)
Line 3 : Integer M, Array 2 Size
Line 4 : Array 2 elements (separated by space)

Output format :

Print intersection elements in different lines

Constraints :

0 <= M, N <= 5 * 10^7

Sample Input 1 :

```
6
2 6 8 5 4 3
4
2 3 4 7
```

Sample Output 1 :

```
2
4
3
```

Sample Input 2 :

```
4
2 6 1 2
5
1 2 3 4 2
```

Sample Output 2 :

```
2
2
1
```

```
int highestFrequency(int *input, int n){

/* Don't write main().
    * the input array is already passed as function argument.
    * Taking input and printing output is handled automatically.
    */



}
```