Same Number Same Frequency

Problem | Submissions | Leaderboard | Discussions

Take an **Integer N** as input and then take **N integers** input from **Geeku**.

Geeku wants to print all those integers whose frequency is **exactly same** as the integer's absolute value.

You have to help **Geeku** in doing so.

Note: 0 is excluded

Sample Input 0

```
10
4 5 -3 8 -3 4 4 -3 6 4
```

Sample Output 0

```
-3
4
```

Hashmap.

4 - 4

5 - 1

-3 - 3

8 - 1

6 - 1

|key| == val

↳ key.

eg1. |4| == 4 ↳ 4

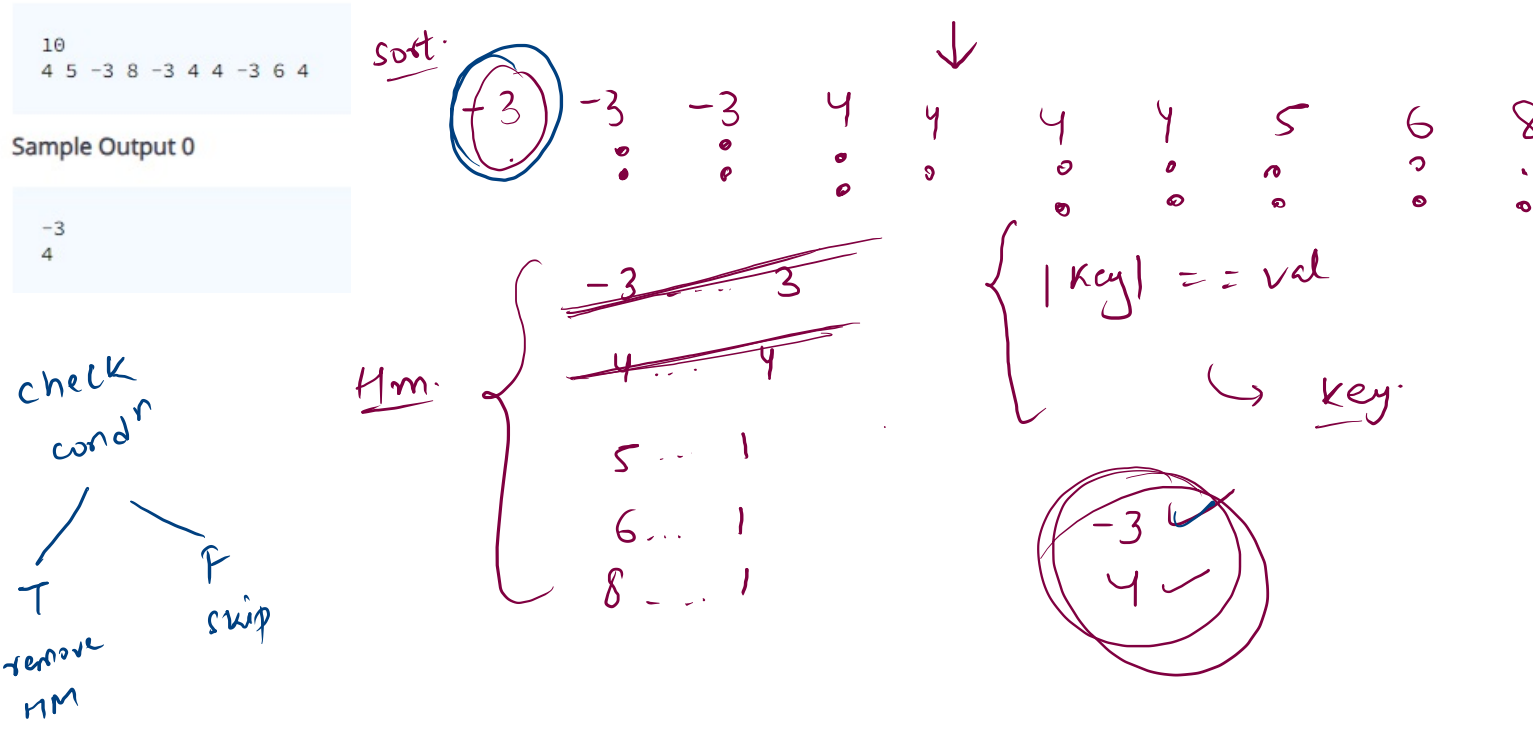
eg2. |-3| = 3 ↳ -3

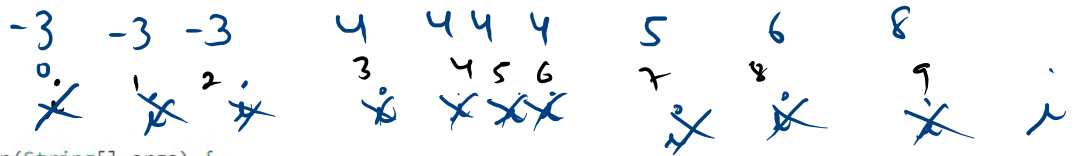
Sample Input 0

```
10
4 5 -3 8 -3 4 4 -3 6 4
```

Sample Output 0

```
-3
4
```

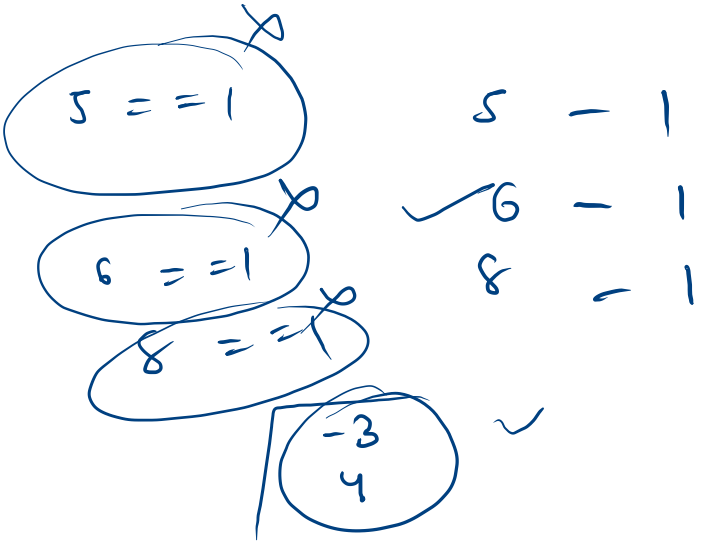





```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int [] A = new int[n];
    for(int i = 0; i < n; i++){
        A[i] = scn.nextInt();
    }

    //sort -- printing in appropriate order
    Arrays.sort(A);
    // freq map
    HashMap<Integer, Integer> hm = new HashMap<>();
    for(int i = 0; i < n; i++){
        hm.put(A[i], hm.getOrDefault(A[i], 0) + 1);
    }
    for(int i = 0; i < n; i++){
        if(hm.containsKey(A[i])){
           if(Math.abs(A[i]) == hm.get(A[i])){
               System.out.println(A[i]);
               hm.remove(A[i]);
           }
        }
    }

}
}
```



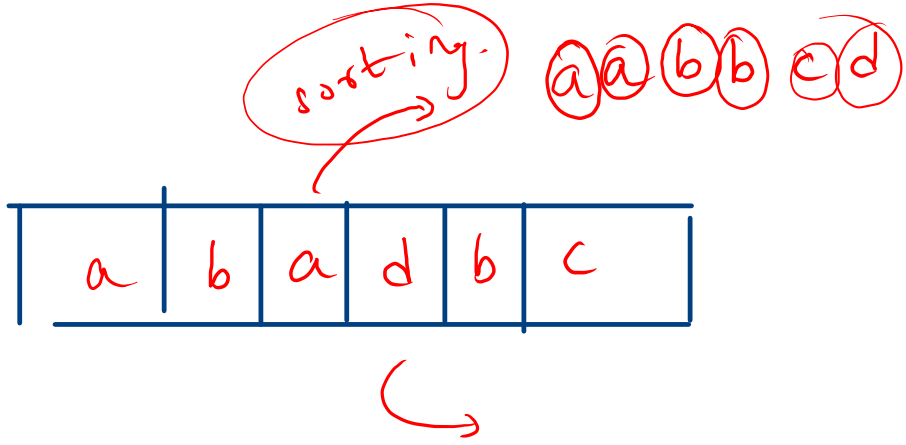
Character and it's Frequency

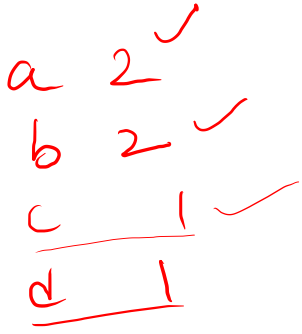
Sample Input 0

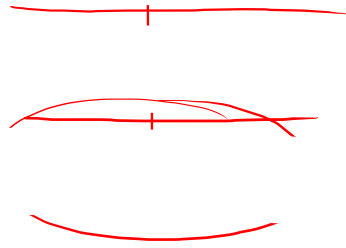
```
6
a b a d b c
```

Sample Output 0

```
a 2
b 2
c 1
d 1
```







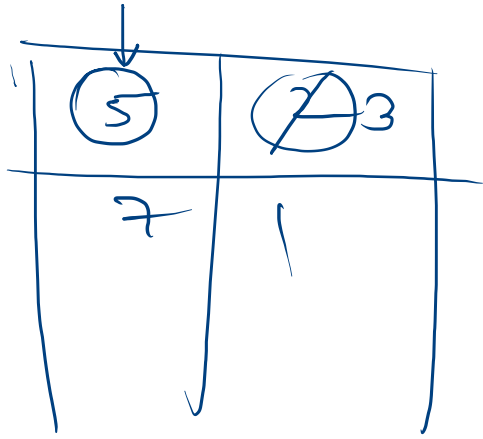
1. i/p
2. sort i/p
3. freq map
4. print & remove

```java
import java.io.*;
import java.util.*;

public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int n = scn.nextInt();
        char [] A = new char[n];
        for(int i = 0; i < n; i++){
            A[i] = scn.next().charAt(0);
        }
        Arrays.sort(A);
        HashMap<Character, Integer> hm = new HashMap<>();
        for(int i = 0; i < n; i++){
            if(hm.containsKey(A[i])){
                hm.put(A[i], hm.get(A[i]) + 1);
            }else{
                hm.put(A[i], 1);
            }
        }
        for(int i = 0; i < n; i++){
            if(hm.containsKey(A[i])){
                System.out.println(A[i] + " " + hm.get(A[i]));
                hm.remove(A[i]);
            }
        }

    }
}
```



employee management

Problem | Submissions | Leaderboard | Discussions

You are tasked with developing an employee management system for a company. To efficiently store employee data, you decide to use a **HashMap<>**. In this HashMap, the keys represent unique employee IDs, and the values are ArrayLists of employee details as strings, including the employee's name, job title, and department.

you will be getting **T** queries which includes:

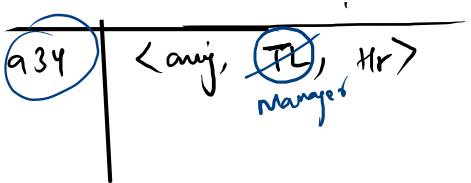
1. case-1 (add) -> add employee with details.
2. case-2 (update) -> update job title of a given employee.
3. case-3 (delete) -> remove the employee.
4. case-4 (show) -> print details of a given employee else print -1.

Sample Input 0

```
5
add a21 Akhil Developer Tech
add a34 anuj TeamLead Hr
update a34 Manager
delete a21
show a34
```

Sample Output 0

```
anuj Manager Hr
```



```java
HashMap<String, ArrayList<String>> hm = new HashMap<>();

int t = scn.nextInt();
for(int i = 0; i < t; i++){
    String opr = scn.next();
    if(opr.equals("add")){
        String id = scn.next();
        String name = scn.next();
        String title = scn.next();
        String dept = scn.next();
        ArrayList<String> details = new ArrayList<>();
        details.add(name);
        details.add(title);
        details.add(dept);
        hm.put(id, details);
    }
```

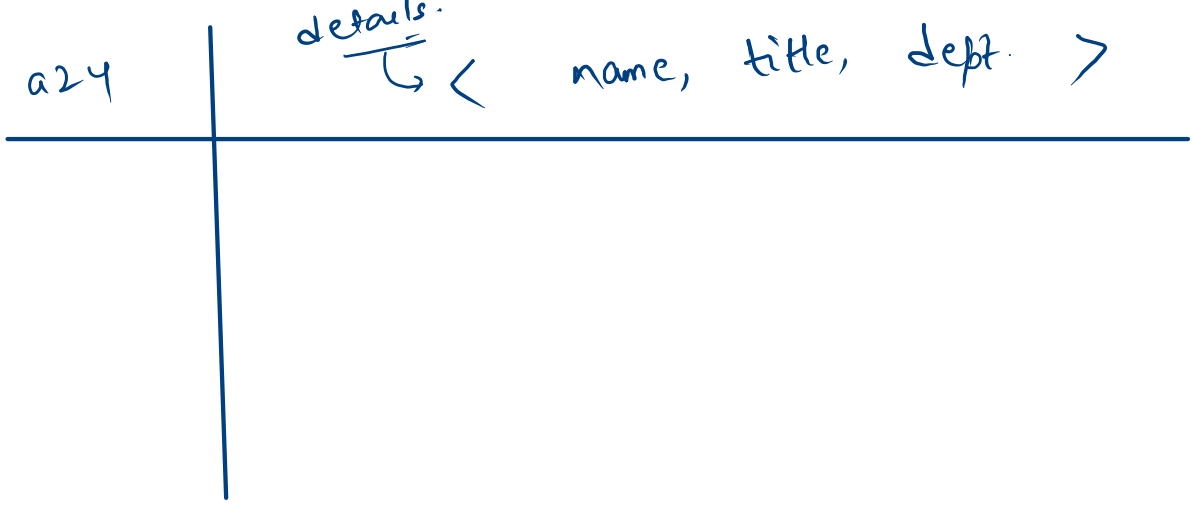
add a24 Aman Instructor DSA

id = a24

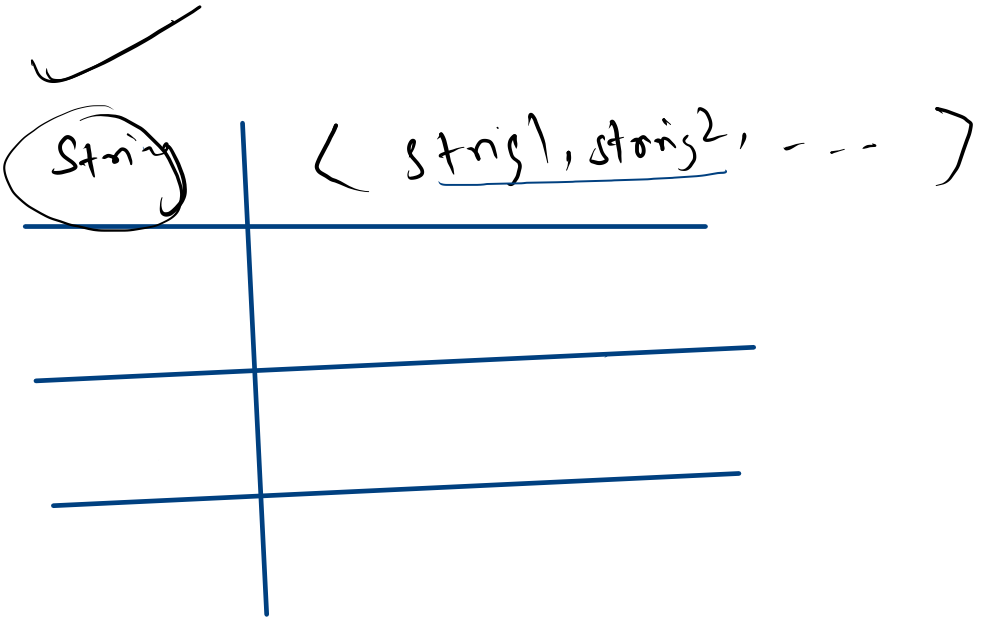
name = Aman

title = Instructor

dept = DSA



```
9    HashMap<String, ArrayList<String>> hm = new HashMap<>();
```



Hm.

x	a21	⟨"Akhil", "Developer", "Tech"⟩
	a34	⟨"Aman", "Manager Hr", "HRD"⟩

add a21 Akhil Developer Tech

```
if(opr.equals("add")){
    String id = scn.next();
    String name = scn.next();
    String title = scn.next();
    String dept = scn.next();
    ArrayList<String> details = new ArrayList<>();
    details.add(name);
    details.add(title);
    details.add(dept);
    hm.put(id, details);
}
```

```
update a34 Rahul
delete a21
```

```
}else if(opr.equals("delete")){
    String id = scn.next();
    hm.remove(id);
```

id

show a21

```
}else if(opr.equals("show")){
    String id = scn.next();
    if(hm.containsKey(id)){
        //show
        ArrayList<String> details = hm.get(id);
        for(String s : details){
            System.out.print(s + " ");
        }
        System.out.println();
    }else{
        System.out.println("-1");
    }
}
```

id = a21

for(int i=0 ; i< details.size(); i++)

{ String s = details.get(i)

details = < "Akhil" , "Developer" , "Tech" >

Akhil Developer Tech

update a34 Manager

```
}
else if(opr.equals("update")){
    // update a34 Manager
    String id = scn.next();
    String newTitle = scn.next();
    ArrayList<String> details = hm.get(id);
    details.set(1, newTitle);

}else if(opr.equals("delete")){
```

```java
import java.io.*;
import java.util.*;

public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);

        HashMap<String, ArrayList<String>> hm = new HashMap<>();

        int t = scn.nextInt();
        for(int i = 0; i < t; i++){
            String opr = scn.next();
            if(opr.equals("add")){
                String id = scn.next();
                String name = scn.next();
                String title = scn.next();
                String dept = scn.next();
                ArrayList<String> details = new ArrayList<>();
                details.add(name);
                details.add(title);
                details.add(dept);
                hm.put(id, details);
            }
            else if(opr.equals("update")){
                // update a34 Manager
                String id = scn.next();
                String newTitle = scn.next();
                ArrayList<String> details = hm.get(id);
                details.set(1, newTitle);

            }else if(opr.equals("delete")){
                String id = scn.next();
                hm.remove(id);
            }else if(opr.equals("show")){
                String id = scn.next();
                if(hm.containsKey(id)){
                    //show
                    ArrayList<String> details = hm.get(id);
                    for(String s : details){
                        System.out.print(s + " ");
                    }
                    System.out.println();
                }else{
                    System.out.println("-1");
                }
            }
        }

    }
}
```

Valid Anagram 5

Problem | Submissions | Leaderboard | Discussions

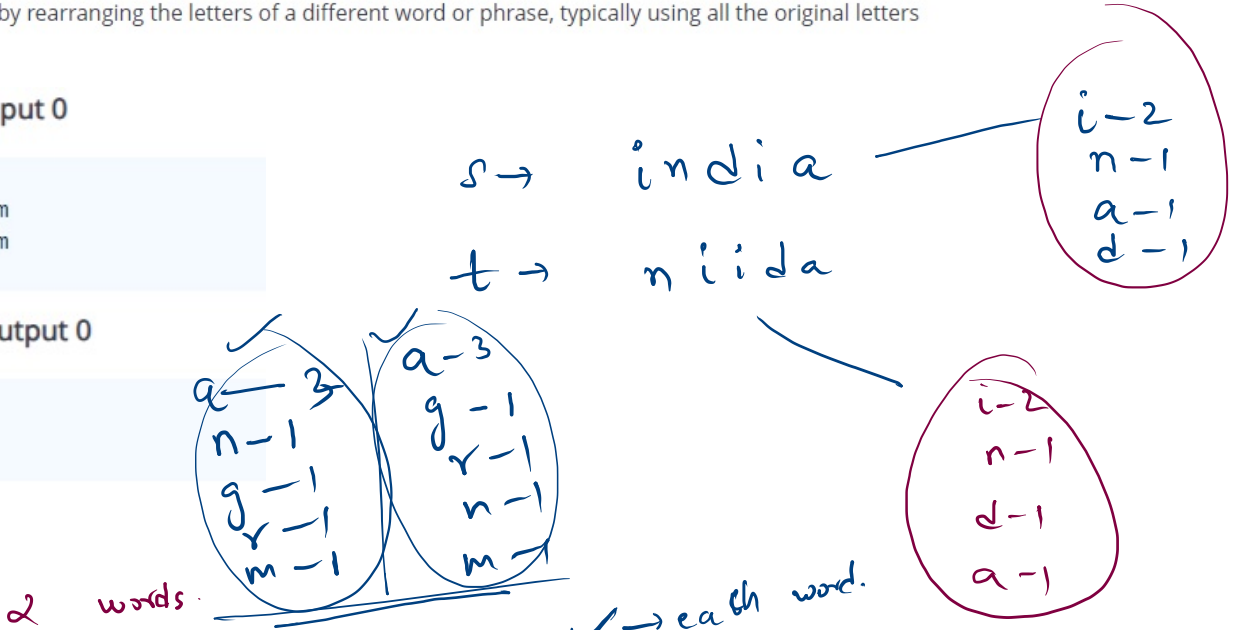
Given two strings s and t, return true if t is an anagram of s, and false otherwise. An Anagram is a word or phrase formed by rearranging the letters of a different word or phrase, typically using all the original letters exactly once.

Sample Input 0

```
anagram
nagaram
```

Sample Output 0

```
true
```



```java
import java.io.*;
import java.util.*;

public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        String s = scn.next();
        String t = scn.next();
        //1. freq map
        HashMap<Character, Integer> hm1 = new HashMap<>();
        for(int i = 0; i < s.length(); i++){
            hm1.put(s.charAt(i), hm1.getOrDefault(s.charAt(i), 0) + 1);
        }
        HashMap<Character, Integer> hm2 = new HashMap<>();
        for(int i = 0; i < t.length(); i++){
            hm2.put(t.charAt(i), hm2.getOrDefault(t.charAt(i), 0) + 1);
        }

        System.out.println(hm1.equals(hm2));

    }
}
```