**Find unique element**

Submissions: [1563](https://practice.geeksforgeeks.org/problem_submissions.php?pid=2645)  Accuracy:

45.69%

   Difficulty: [Basic](https://practice.geeksforgeeks.org/Basic/0/0/)   Marks: 1

Show Topic Tags   

Given an array which contains all elements occurring k times, but one occurs only once. Find that unique element.

**Input:**  
The first line consists of an integer **T** i.e number of test cases. The first line of each test case consists of two integers **n** and **k**. The next line consists of **n** spaced integers.

**Output:**  
Print the required output.

**Constraints:**   
1<=T<=100  
2<=k<n<=100  
1<=a[i]<=10000

**Example:  
Input:**  
2  
7 3  
6 2 5 2 2 6 6   
5 4  
2 2 2 10 2

**Output:**  
5  
10

\*\* For More Input/Output Examples Use ['Expected Output'](https://practice.geeksforgeeks.org/problems/find-unique-element/0#ExpectOP) option \*\*

Contributor: Saksham Ashtputre  
[Author: saksham0751](https://auth.geeksforgeeks.org/user/saksham0751/practice/)

<https://practice.geeksforgeeks.org/problems/find-unique-element/0>

#include <iostream>

#include <map>

#include <stdio.h>

using namespace std;

int ObtenerElem(int arr[], int n)

{

map<int,int> hash;

int candidato = 0;

for (int i = 0; i < n; i++)

{

hash[arr[i]]++;

}

for(std::map<int, int>::iterator it = hash.begin(); it != hash.end(); it++) {

if(it->second == 1) {

return it->first;

}

}

return -1;

}

int main() {

int t;

scanf("%d", &t);

while(t--) {

int n, k;

scanf("%d %d", &n, &k);

int arr[n];

for(int i =0; i<n; i++) {

scanf("%d", &arr[i]);

}

cout << ObtenerElem(arr, n) << endl;

}

return 0;

}