2-G-Cookies Problem

Aim:

Assume you are an awesome parent and want to give your children some cookies. But, you should give each child at most one cookie.

Each child i has a greed factor g[i], which is the minimum size of a cookie that the child will be content with; and each cookie j has a size s[j]. If s[j] >= g[i], we can assign the cookie j to the child i, and the child i will be content. Your goal is to maximize the number of your content children and output the maximum number.

**Example 1:**

**Input:**

3

1 2 3

2

1 1

**Output:**

1

Explanation: You have 3 children and 2 cookies. The greed factors of 3 children are 1, 2, 3.

And even though you have 2 cookies, since their size is both 1, you could only make the child whose greed factor is 1 content.

You need to output 1.

**Constraints:**

1 <= g.length <= 3 \* 10^4

0 <= s.length <= 3 \* 10^4

1 <= g[i], s[j] <= 2^31 - 1

Algorithm:

 Read the integer n and array a of size n.

 Read the integer m and array b of size m.

 Iterate through each element of b, and for each element b[i], check if it exists in a and is not equal to the previously found value s.

 If such an element is found, update s to that element.

 Print the value of s which is the first common element found in a and b.

Code:

#include <stdio.h>

int main()

{

int n,m,s=0;

scanf("%d",&n);

int a[n];

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

{

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

}

scanf("%d",&m);

int b[m];

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

{

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

}

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

{

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

{

if(a[i]==a[j] && a[i]!=s)

{

s=a[i];

}

}

}

printf("%d",s);

}

Output:

|  | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | 2  1 2  3  1 2 3 | 2 | 2 |  |

Passed all tests!

**Correct**

Marks for this submission: 1.00/1.00.

Result:

The expected output was obtained