3.b. 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:

```
function main() {
    initialize n // number of children
```

```
read n

initialize greed array of size n // array to hold children's greed factors

// read greed factors for each child
for i from 0 to n-1 {
    read greed[i] from user
}

initialize c // number of cookie sizes
read c from user

initialize csize array of size c // array to hold cookie sizes

// read cookie sizes
for j from 0 to c-1 {
    read csize[j] from user
}

initialize count to 0 // counter for satisfied children

// check each child's greed against available cookie sizes
for i from 0 to n-1 {
    for j from 0 to c-1 {
        if csize[j] is greater than or equal to greed[i] {
            increment count by 1 // child is satisfied
            break // exit inner loop after satisfying this child
        }
    }
}

print count // output the total count of satisfied children
```

}

Program:

```
#include<stdio.h>
#include<string.h>

int main(){
    int n;
    scanf("%d",&n);
    int greed[n];
    for(int i=0;i<n;i++){
        scanf("%d ",&greed[i]);
    }

    int c;
    scanf("%d",&c);
    int csize[c];
    for(int i=0;i<c;i++){
        scanf("%d ",&csize[i]);
    }

    int count=0;
    for(int i=0;i<n;i++){
        for(int j=0;j<c;j++){
            if (csize[j]>=greed[i]){
                count++;
                break;
            }
```

```
        }
    }
    printf("%d",count);


}
```

Output:

	Input	Expected	Got	
✔	2 1 2 3 1 2 3	2	2	✔