



## USACO 2022 FEBRUARY CONTEST, SILVER PROBLEM 3. EMAIL FILING

[Return to Problem List](#)

Contest has ended.

[Log in to allow submissions in analysis mode](#)

English (en) ▼

Farmer John has fallen behind on organizing his inbox. The way his screen is organized, there is a vertical list of folders on the left side of the screen and a vertical list of emails on the right side of the screen. There are  $M$  total folders, numbered  $1 \dots M$  ( $1 \leq M \leq 10^4$ ). His inbox currently contains  $N$  emails numbered  $1 \dots N$  ( $1 \leq N \leq 10^5$ ); the  $i$ th email needs to be filed into folder  $f_i$  ( $1 \leq f_i \leq M$ ).

FJ's screen is rather small, so he can only view  $K$  ( $1 \leq K \leq \min(N, M)$ ) folders and  $K$  emails at once. Initially, his screen starts out displaying folders  $1 \dots K$  on the left and emails  $1 \dots K$  on the right. To access other folders and emails, he needs to scroll through these respective lists. For example, if he scrolls down one position in the list of folders, his screen will display folders  $2 \dots K + 1$ , and then scrolling down one position further it will display folders  $3 \dots K + 2$ . When FJ drags an email into a folder, the email disappears from the email list, and the emails after the one that disappeared shift up by one position. For example, if emails  $1, 2, 3, 4, 5$  are currently displayed and FJ drags email  $3$  into its appropriate folder, the email list will now show emails  $1, 2, 4, 5, 6$ . FJ can only drag an email into the folder to which it needs to be filed.

Unfortunately, the scroll wheel on FJ's mouse is broken, and he can only scroll downwards, not upwards. The only way he can achieve some semblance of upward scrolling is if he is viewing the last set of  $K$  emails in his email list, and he files one of these. In this case, the email list will again show the last  $K$  emails that haven't yet been filed, effectively scrolling the top email up by one. If there are fewer than  $K$  emails remaining, then all of them will be displayed.

Please help FJ determine if it is possible to file all of his emails.

### INPUT FORMAT (input arrives from the terminal / stdin):

The first line of input contains  $T$  ( $1 \leq T \leq 10$ ), the number of subcases in this input, all of which must be solved correctly to solve the input case. The  $T$  subcases then follow. For each subcase, the first line of input contains  $M, N$ , and  $K$ . The next line contains  $f_1 \dots f_N$ .

It is guaranteed that the sum of  $M$  over all subcases does not exceed  $10^4$ , and that the sum of  $N$  over all subcases does not exceed  $10^5$ .

### OUTPUT FORMAT (print output to the terminal / stdout):

Output  $T$  lines, each one either containing either YES or NO, specifying whether FJ can successfully file all his emails in each of the  $T$  subcases.

### SAMPLE INPUT:

```
6
5 5 1
1 2 3 4 5
5 5 1
1 2 3 5 4
5 5 1
1 2 4 5 3
5 5 2
1 2 4 5 3
3 10 2
1 3 2 1 3 2 1 3 2 1

3 10 1
1 3 2 1 3 2 1 3 2 1
```

### SAMPLE OUTPUT:

```
YES
```

YES  
NO  
YES  
YES  
NO

**SCORING:**

- In inputs 2-10, the sum of  $M$  over all subcases does not exceed  $10^3$ .
- In inputs 11-12, no additional constraints.

Problem credits: Brian Dean

Contest has ended. No further submissions allowed.

---