**Seats Wanted!**

**Author:**

**Time limit:** 1 second

**Memory limit:** 256 megabytes

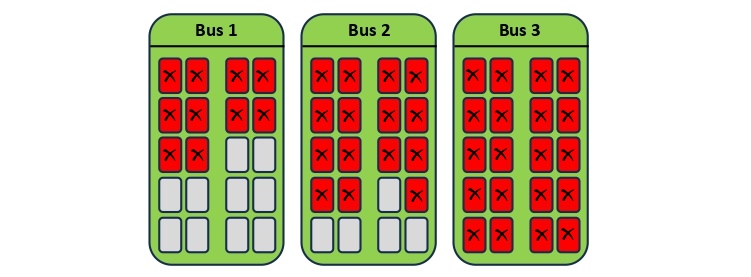
Jashim uncle is a kind individual at our university, IIUC, who oversees the transportation division. Whenever a student faces a problem with transportation, Jashim uncle genuinely shows concern and tries to help.

Today, Samia and her friends have planned to go to a restaurant to have lunch together after their classes using the university bus. Their group consists of ***n*** people, including Samia. To travel together, they need a bus with at least ***n*** seats.

IIUC has ***b*** buses in total, and each bus has exactly ***s*** seats. Checking each bus individually for available seats is time-consuming and tedious for Samia and her friends. Therefore, they asked Jashim uncle for help. Jashim uncle maintains a list of the information for all the buses. For ***ith*** bus ***(1 <= i <= b)***, he knows the number of seats already taken, denoted as ***taken\_seatsi*** ***(0 <= taken\_seatsi <= s)***. He reviewed the list.

Write a program to determine if there is a bus with at least n seats available.

The visual representation of the third test case is given below:



**Input**

The input begins with an integer ***t (1 ≤ t ≤ 1000)***, representing the number of test cases. Each test case consists of two lines.

For each test case:

* The first line contains three space-separated integers: ***n (1 ≤ n ≤ 50)*** representing the total number of group members, ***b (1 ≤ b ≤ 100)*** representing the total number of buses, and ***s (1 ≤ s ≤ 100)*** representing the number of seats on every bus.
* The second line contains b space-separated integers denoting the number of seats already taken for each bus. The number of seats taken for the ***ith*** bus is represented as ***taken\_seatsi (0 ≤ taken\_seatsi ≤ s)***, where ***1 ≤ i ≤ b***.

**Output**

After reviewing the list, if there is a bus with at least ***n*** seats available, print "Yes". Otherwise, print "No".

**Examples**

|  |  |
| --- | --- |
| **Input** | **Output** |
| **3**  **7 10 40**  **40 37 35 39 34 40 33 30 35 39**  **20 5 30**  **15 20 30 25 20**  **5 3 20**  **10 15 20** | **Yes**  **No**  **Yes** |