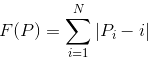
**KOD-O-SIS Final Question**

**Question #1**

Let's say **P= [P1, P2 . . . PN]** is a permutations of all positive integers less than or equal to **N**. A function called "special count" **F(P)** is defined as:



Now Ram gives a **N** and **K** to Ravan and asks if **K** is a possible special count for any permutation of first **N** positive integers.

**Input**  
First line contains **C**, the number of test cases. Each test case consists of **n** and **k** in one line.

**Output**  
For each test case, print "**YES**" or "**NO**", denoting whether **k** is a possible special count or not.

**Constraints**  
1 ≤ **C** ≤ 100  
20% testdata: 1 ≤ **n** ≤ 10  
20% testdata: 1 ≤ **n** ≤ 20  
60% testdata: 1 ≤ **n** ≤ 40  
0 ≤ **k** ≤ 2000

Sample Input

2

2 1

3 2

Sample Output

NO

YES

Explanation

First test case:  
None of the two permutations [1, 2] and [2, 1] have special count 1.  
  
Second test case:  
Permutations [1, 3, 2] and [2, 1, 3] have special count 2.

**Question #2**

Dhal gives to Rohan a bag of numbers **[1, 2, 3 ... n]** and then he removes **L** numbers **A1, A2 ... AL** from the bag. He now asks Rohan to find the **P**'th smallest number in the bag.

**Input**  
First line contains **X**, the number of test cases. Each test case consists of **c**, **k** and **p** followed by **k** integers in next line denoting the array **A**.

**Output**  
For each test case, print **P**'th smallest number in the bag. If no such number exists output **-1**.

**Constraints**  
1 ≤ **X** ≤ 10  
20% testdata: 1 ≤ **n** ≤ 103  
20% testdata: 1 ≤ **n** ≤ 105  
60% testdata: 1 ≤ **n** ≤ 109  
0 ≤ **k** ≤ **min(n, 105)**  
1 ≤ **p** ≤ **n**

Sample Input

2

4 1 2

1

5 2 4

1 3

Sample Output

3

-1

Explanation

Test case 1  
Remaining numbers are [2, 3, 4]. 3 is the 2nd smallest remaining numbers.  
  
Test case 2  
Remaining numbers are [2, 4, 5]. 4th smallest remaining number doesn't exist.