Desafío de código

Objetivos:
Conocer las aptitudes técnicas de los candidatos a pasantía en el Área de Software en Inteligencia Artificial.

Reglas:

- Prohibido utilizar internet para resolver el problema.
- Tiempo máximo de prueba 5 días, desde la recepción de la prueba.
- Implementación en Python o Javascript.

Instrucciones:

1. **Crear una carpeta para cada ejercicio, utilizando el nombre "ejercicio" con el número del problema. Ej (ejercicio 1 para el código de la solución 1)**
    a. La carpeta debe tener un archivo problema.txt con las instrucciones del problema.
    b. La carpeta debe tener un archivo readme con las instrucciones necesarias para correr el código.
    c. La carpeta debe tener un archivo para cada prueba unitaria que se pueda correr para confirmar la solución. (Mínimo 3 pruebas unitarias por cada ejercicio)
    d. La carpeta debe tener un archivo output por cada prueba unitaria con la respuesta de la misma.
2. **Comprimir cada carpeta en un archivo .zip por separado para enviar la solución.**
3. **Agregar los siguientes destinos en el correo con el asusto "Prueba Técnica GBM - nombre candidato":**
    a. adnavas@gbm.net
    b. jgraciano@gbm.net
4. **Enviar las carpetas comprimidas e incluir un editor online donde se pueda evaluar la solución. Ejemplo:**
    a. **Python** https://www.programiz.com/python-programming/online-compiler/.
    b. **Javascript** https://playcode.io/new/)
5. **PUNTOS EXTRAS**
    a. Utilizar un repositorio y enviar el link para clonarlo y probar la solución.
    b. 5 pruebas unitarias.

Nota: Cualquier regla que sea incumplida lleva automáticamente a la anulación de prueba del candidato

Ejercicios 1

1. Un palíndromo, también llamado palíndromo, palíndroma o palindroma, es una palabra, o frase que se lee igual en un sentido que en otro. Ejemplo ANA se puede leer al derecho y al revés y significa lo mismo.

    - Punto extra - No puede usar ciclos
    - Punto extra - No más de 5 líneas de código

Ejercicios 2

2. The Formula 1 season consists of a series of races, known as Grand Prix, organized by the International Federation of Automobile (FIA). The results of each Grand Prix are combined to determine Pilots\' World Championship. More specifically, for each race some points are distributed to pilots, depending on their classification in the race. At the end of the season, the pilot who has earned the most points is declared the World Champion.

Formula 1 organizers change constantly the competition rules, aiming to provide more excitement to fans. One rule modified for the 2010 season was the distribution of points in each Grand Prix. Since 2003, the scoring rule rewarded the top eight pilots, according to the following table:

Place	1	2	3	4	5	6	7	8
Points	10	8	6	5	4	3	2	1

That is, the winning driver received 10 points, second place received 8 points, and so on. In the 2010 season the top ten will receive points, obeying the following table:

Place	1	2	3	4	5	6	7	8	9	10
Points	25	18	15	12	10	8	6	4	2	1

The change in the scoring system led to much speculation about what would have been the effect to the World Championship in the past if the new score had been used. For example, would Lewis Hamilton have been champion in 2008, considering he and Felipe Massa were separated by just one point? To end the speculation, FIA hired you to write a program that, given the results of each race of a season determines the World Champion for different scoring systems.

Input

The input contains several test cases. The first line of a test case contains two integers **G** and **P** separated by a blank space, indicating the number of Grand Prix ($1 \leq \mathbf{G} \leq 100$) and the number of pilots ($1 \leq \mathbf{P} \leq 100$).. Pilots are identified by integers from 1 to **P**. Each of the following *G* lines indicates the result of a race, and contains **Q** integers separated by spaces. On each line, the (*i*)-th number indicates the order of arrival of pilot *i* in the race (the first number indicates the order of arrival of a pilot 1 in that race, the second number indicates the order of arrival of pilot 2 in that race and so on). The next line contains a single integer *S* indicating the number of scoring systems ($1 \leq \mathbf{S} \leq 10$), After that, each of the following lines *S* contains a description of a scoring system. The description of a scoring system begins with an integer**K** ($1 \leq \mathbf{K} \leq \mathbf{P}$), indicating the last finishing order to receive points, followed by a blank space, followed by e **K** integers **k0**, **k1**, ... , **kn−1** ($1 \leq \mathbf{ki} \leq 100$) separated by spaces, indicating the number of points to be assigned (the first integer indicates the points for first place, the second integer indicates the points for second place and so on). The last test case is followed by a line containing only two zeros separated by a blank space.

Output

For each scoring system in the input your program must print one line, containing the identifier of the World Champion. If more than one pilot are World Champions (ie, if there is a tie), the line must contain all World Champions, in increasing order of identifier, separated by a space.

Input Sample	Output Sample
1 3 3 2 1 3 3 5 3 2 3 5 3 1 3 1 1 1 3 10 1 2 3 4 5 6 7 8 9 10 10 1 2 3 4 5 6 7 8 9 9 10 1 2 3 4 5 6 7 8 2 5 5 4 3 2 1 3 10 5 1 2 4 1 3 4 2 4 1 3 2 2 3 3 2 1 3 5 4 2 0 0	3 3 1 2 3 3 3 2 4 4

Ejercicios 3

3. You are standing on the OX-axis at point 0 and you want to move to an integer point x > 0. You can make several jumps. Suppose you're currently at point y (y may be negative) and jump for the k-th time.
    a. You can:
        i. either jump to the point y+k or
        ii. jump to the point y−1.

    What is the minimum number of jumps you need to reach the point x?

    Input

    The first line contains a single integer t (1≤t≤1000) — the number of test cases.

    The first and only line of each test case contains the single integer x (1≤x≤106) — the destination point.

    Output

    For each test case, print the single integer — the minimum number of jumps to reach x. It can be proved that we can reach any integer point x.

    Example

    input Copy

```
5
1
2
3
4
5
```

    output Copy

```
1
3
2
3
4
```

Note:

In the first test case x = 1, so you ned only one jump: the 1-st jump from 0 to 0 + 1 = 1

In the second test case x = 2. You need at least three jumps:
The 1-st jump from 0 to 0 + 1 = 1
The 2-nd jump from 1 to 1 + 2 = 3
The 3-rd jump from 3 to 3 - 1 = 2

Two jumps are not enough because these are the only possible variants:
The 1-st jump as -1 and the 2-nd one as -1, you'll reach 0 - 1 - 1 = -2
The 1-st jump as -1 and the 2-nd one as +2, you'll reach 0 - 1 + 2 = 1
The 1-st jump as -1 and the 2-nd one as -1, you'll reach 0 - 1 - 1 = 0
The 1-st jump as -1 and the 2-nd one as +2, you'll reach 0 - 1 + 2 = 3

In the third test case, you need two jumps: the 1-st one as +1 and the 2-nd one as +2, so 0+1+2=3

In the fourth test case, you need three jumps: the 1-st one as -1, the 2nd one as +2 and the 3-rd one as +3, so 0 - 1 + +2 +3 = 4

- Punto extra - 5 pruebas unitarias.

Ejercicios 4

A grocery store wants to know its customers better in order to create personalized marketing campaigns. You are asked to develop a classification model using Keras that takes into account customers' shopping frequency, their spending habits and the maximum amount they spend in the store. The goal of the model is to classify customers into three categories: low, medium and high value.

Attached training data in a file named data_customer_classification