

## Formando equipos impares

Contribución de Lautaro Lasorsa

### Descripción del problema

Una cierta empresa, que se organiza de forma jerárquica, desea armar equipos para llevar a cabo diferentes proyectos.

La empresa tiene  $N$  empleados, numerados desde 0 hasta  $N - 1$  inclusive. Está dirigida por un gerente general, que es supervisor directo de algunos empleados, que son a su vez supervisores directos de otros empleados de menor jerarquía, y así siguiendo hasta llegar a aquellos empleados sin subordinados.

En esta empresa cada empleado (excepto el gerente general) tiene **exactamente un supervisor directo**, y puede supervisar a 0 o más empleados. Todos los empleados son directa o indirectamente supervisados por el gerente general.

Un equipo deberá estar formado por:

- Un empleado designado como *director* del equipo.
- Opcionalmente, puede haber más empleados además del director, que sean directa o indirectamente supervisados por él.
- Es un requisito que por cada miembro del equipo que no sea el director del equipo, **su supervisor directo también forme parte del equipo**.
- Para evitar empates en las votaciones y así facilitar la toma de decisiones, los equipos deben estar conformados por **una cantidad impar** de empleados.

Siguiendo estas reglas, se debe calcular cuántas maneras diferentes existen de formar un equipo eligiendo algunos empleados de la empresa.

Como este número puede ser muy grande, se debe indicar el resto al dividirlo por  $10^9 + 7$

### Detalles de implementación

Debes implementar la función `impares(subordinados)`, que recibe un único parámetro `subordinados`: un arreglo de  $N$  arreglos de enteros. El arreglo `subordinados[i]` con  $0 \leq i < N$  indica los *subordinados* del empleado  $i$ , es decir, aquellos empleados que tienen al  $i$  como supervisor directo. La función debe retornar un entero con el valor solicitado.

El gerente general es el empleado 0.

### Evaluador local

El evaluador lee de la entrada estándar:

- Una línea con el entero  $N$
- $N$  líneas, que representan los arreglos `subordinados[i]` en orden, para  $i$  entre 0 y  $N - 1$  inclusive. Cada línea comienza con un número  $k$  que indica el tamaño del correspondiente arreglo `subordinados[i]`, seguido de los  $k$  elementos del arreglo.

Escribe en la salida estándar una línea con el resultado retornado por la llamada `impares(subordinados)`.

### Restricciones

- $1 \leq N \leq 200.000$

**Ejemplos**

Para la siguiente entrada:

```
6
5 1 2 3 4 5
0
0
0
0
0
```

La respuesta correcta es:

```
21
```

En cambio para:

```
5
1 1
1 2
1 3
1 4
0
```

La respuesta es:

```
9
```

Y para:

```
15
2 1 2
2 3 4
2 5 6
2 7 8
2 9 10
2 11 12
2 13 14
0
0
0
0
0
0
0
0
```

La respuesta es:

```
380
```

**Subtareas**

1.  $N \leq 10$  (12 puntos)
2. `subordinados[i]` tiene como máximo un elemento para  $0 \leq i < N$  (12 puntos)
3. `subordinados[0]` tiene tamaño 2, y `subordinados[i]` tiene como máximo un elemento para  $1 \leq i < N$  (6 puntos)
4. `subordinados[0]` tiene hasta 20 elementos, `subordinados[i]` tiene a lo más un elemento si  $1 \leq i < N$  (15 puntos)
5. `subordinados[i]` tiene hasta dos elementos para  $0 \leq i < N$  (15 puntos)
6. Sin más restricción (40 puntos)