

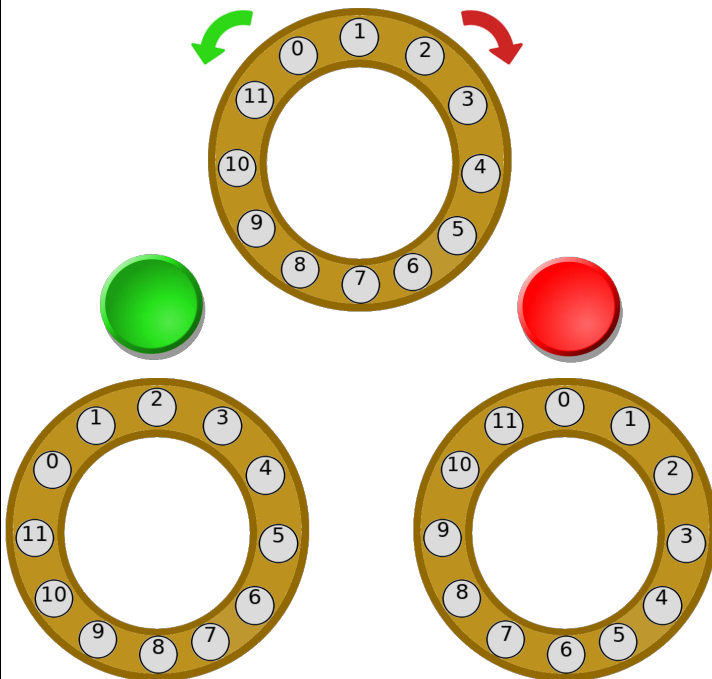
Analizando muestras

Contribución de Agustín Santiago Gutiérrez

Descripción del problema

En un enorme laboratorio de análisis clínicos se deben analizar n muestras de sangre de numerosos pacientes. Cada muestra está contenida en un tubo de ensayo.

Los tubos se encuentran ubicados en una enorme gradilla circular giratoria con exactamente n posiciones donde ubicar los tubos. Esta gradilla se controla mediante dos botones. Cada botón permite **rotar la gradilla en exactamente una posición**, de modo que cada tubo quede ubicado en donde estaba su vecino correspondiente. Uno de los botones permite girar en sentido horario, y el otro en sentido antihorario.



Por fuera de la gradilla, frente a cada una de las n posiciones donde están los tubos, se encuentran estacionadas máquinas de análisis. Las distintas máquinas no son todas iguales: algunas pueden realizar tipos de análisis que otras no.

Por cada una de las n muestras, se conoce un listado que indica cuáles de las n máquinas son capaces de analizarla.

El tiempo de análisis de una muestra es despreciable, y se pueden estar analizando varias muestras al mismo tiempo (necesariamente en distintas máquinas), pero el tiempo de rotación de la máquina es sumamente lento.

Por esta razón, tu tarea consiste en calcular la mínima cantidad de pulsaciones de botón que es necesario utilizar para que **todas** las muestras sean analizadas por **alguna** máquina.

Las muestras se numeran de 0 a $n - 1$ en sentido horario, al igual que las máquinas analizadoras. Inicialmente, la máquina i se encuentra frente al tubo i .

Detalles de implementación

Debes implementar la función:

- `analizar(muestras):`
 - `muestras`: arreglo de n arreglos de enteros. El arreglo `muestras[i]` contiene los índices de todas las máquinas capaces de analizar la muestra i .
 - Debe retornar un entero: la mínima cantidad de pulsaciones de botón necesarias para poder analizar todas las muestras.

Cotas

- $2 \leq n \leq 100.000$
- La suma de las longitudes de todos los arreglos de `muestras` será a lo sumo 300.000.
- Ninguno de los arreglos `muestras[i]` contiene elementos repetidos.
- $0 \leq \text{muestras}[i][j] < n$
- Todos los arreglos `muestras[i]` contienen al menos un elemento.

Evaluador local

El evaluador local lee de la entrada estándar:

1. Una línea con el entero n
2. n líneas, que contienen primero un entero k_i con la cantidad de números de un arreglo $muestras[i]$, y luego los k_i números del arreglo $muestras[i]$.

Escribe por la salida estándar el resultado de aplicar la función `analizar(muestras)`.

Ejemplo

Si el evaluador local recibe:

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

Con una solución correcta, imprime:

```
1
```

Si en cambio recibe:

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

Con una solución correcta, imprime:

```
3
```

Subtareas

1. $n = 2$ (7 puntos)
2. $n = 3$ (13 puntos)
3. Todos los arreglos $muestras[i]$ contienen **exactamente** un elemento (29 puntos)
4. Sin más restricción (51 puntos)