

Los viejos teclados T9



Los teclados T9, también conocidos como teclados de marcación alfanumérica, fueron una característica común en los teléfonos móviles antes de la era de los *smartphones*. Este tipo de teclado simplificaba la entrada de texto en dispositivos con pantallas pequeñas y sin teclados completos *QWERTY*. Funcionaban asignando varias letras a cada tecla numérica, como se muestra en la siguiente figura:



1	2 ABC	3 DEF
4 GHI	5 JKL	6 MNO
7 PQRS	8 TUV	9 WXYZ
0		

Por ejemplo, la tecla 2 podría representar las letras A, B y C. Al presionar repetidamente una tecla, el sistema ciclaba a través de las letras asociadas a esa tecla, permitiendo a los usuarios seleccionar la letra deseada para formar palabras. Por ejemplo, para teclear PERA, había que pulsar la tecla 7 una vez para obtener la letra P, luego la tecla 3 dos veces para obtener la E, luego la tecla 7 tres veces para la R y, por último, pulsar la tecla 2 una sola vez para obtener la letra A. En total, eran necesarias 7 pulsaciones para teclear la palabra entera.

Los teclados T9 son eficientes para escribir mensajes cuando el alfabeto del idioma tiene un número limitado de letras. Pero, ¿qué pasaría si el alfabeto tuviera un número arbitrario de letras distintas? Para simplificar, vamos a suponer que asignamos un código numérico a cada letra del alfabeto. Además, cada tecla contiene un número k indicando que, al pulsar esa tecla, se imprimirá la letra cuyo código es k . Si se pulsa la misma tecla otra vez, en su lugar se imprimirá la letra cuyo código es $k + 1$, y así sucesivamente. Por ejemplo, dado el siguiente teclado:

1	3	5	9	11
---	---	---	---	----

La tecla 1 permite escribir las letras con código 1 y 2. La tecla 5 permite escribir las letras con código 5, 6, 7 y 8, mientras que la tecla 11 permite escribir las letras cuyo código es mayor o igual a 11. De este modo, si queremos escribir la secuencia [3, 10, 15, 5] tendremos que pulsar una vez la tecla 3, dos veces la tecla 9, cinco veces la tecla 11 y, por último, una vez la tecla 5. En total, necesitamos nueve pulsaciones.

También cabe la posibilidad de que haya mensajes que no se puedan teclear con nuestro teclado. Por ejemplo, si quisiéramos escribir la secuencia [3, 0], no es posible escribir la letra con código 0, ya que nuestro teclado solamente permite teclear letras cuyo código es mayor o igual a 1.

1. Añade a la clase `SetTree<T>` un método `find_le(const T &elem)` que devuelva un `pair<T, bool>`. Este método recibe un valor `elem` y busca en el conjunto `this` el mayor elemento x tal que $x \leq elem$. Si tal elemento existe, la segunda componente del par devuelto es `true`, y la primera componente es el x buscado. Por el contrario, si todos los elementos del conjunto son estrictamente mayores

que `elem`, la segunda componente del par es `false` y la primera componente puede contener cualquier valor arbitrario.

La especificación de `find_1e` es la siguiente:

$$\text{find_1e}(S : \text{Set}\langle T \rangle, \text{elem} : T) \rightarrow \langle \text{res} : T, b : \text{bool} \rangle \\ \{(b \Rightarrow \text{res} = \text{máx}\{z \in S \mid z \leq \text{elem}\}) \wedge (\neg b \Rightarrow \neg \exists z : (z \in S \wedge z \leq \text{elem}))\}$$

Puedes definir los métodos privados que sean necesarios dentro de la clase `SetTree<T>`.

2. Indica y justifica el coste del método `find_1e` y de los métodos privados que necesites. Para ello, puedes suponer que el árbol binario correspondiente está equilibrado. Si alguno de los métodos es recursivo, indica la recurrencia de coste.
3. Escribe un programa que procese varios casos de prueba. Cada caso contiene la descripción de un teclado, y una secuencia de códigos que se quiere escribir. El programa ha de determinar el número mínimo de pulsaciones necesarias para escribir la secuencia dada con ese teclado.
4. Indica y justifica el coste de tratar un caso de prueba en función del número de teclas en el teclado y de la longitud de la secuencia de letras que se quiere escribir.

Entrada

La entrada contiene varios casos de prueba. Cada caso de prueba comienza con dos números N y M que indican, respectivamente, cuántas teclas hay en el teclado y la longitud de la secuencia de letras que se quiere escribir. La línea siguiente contiene N números que indican los códigos que hay en las teclas. La línea siguiente contiene M números con los códigos de las letras que se van a escribir. Se cumple que $1 \leq N \leq 10^5$ y que $0 \leq M \leq 10^5$. Los códigos son números contenidos entre 0 y 10^6 .

La entrada finaliza con dos ceros (0 0), que no se procesan.

Salida

Para cada caso de prueba debe imprimirse una línea con el número mínimo de pulsaciones necesarias para escribir la secuencia de letras dada. Si no es posible escribir la secuencia con el teclado indicado, debe imprimirse `NO SE PUEDE`.

Entrada de ejemplo

```
5 4
9 3 11 5 1
3 10 15 5
1 3
10
15 20 25
1 3
10
10 9 11
0 0
```

Salida de ejemplo

```
9
33
NO SE PUEDE
```

Créditos

Autor: Manuel Montenegro

Imagen teléfono móvil: Multicherry, distribuido con licencia CC BY-SA 4.0 Deed.