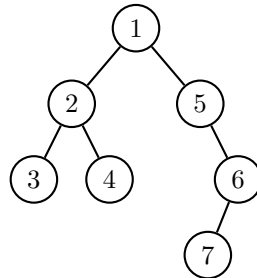


Reconstrucción de un árbol binario

El semestre pasado estuvimos diseñando árboles binarios, con números enteros en los nodos. Los más bonitos eran aquellos que tenían todos los valores distintos, como este:



Cuando acabamos con ellos, guardamos los más bonitos en un archivo, serializados según su recorrido en preorden (ya sabes: raíz, hijo izquierdo, hijo derecho). Por aquello de la tolerancia a fallos, los guardamos en un segundo archivo, pero ahí serializados según su recorrido en inorden (hijo izquierdo, raíz, hijo derecho). *Y ¡menos mal!* Creemos que con uno solo de los recorridos no habríamos podido ahora reconstruir los árboles y tampoco estamos seguros de poderlo hacer teniendo los dos. ¿Nos ayudas?

Requisitos de implementación.

Se implementará una función que reciba los recorridos en preorden e inorden de un árbol (por ejemplo en dos vectores) y devuelva un objeto de la clase `bintree` que represente un árbol que tenga esos dos recorridos.

Entrada

La entrada está formada por una serie de casos. Cada caso consta de dos líneas: la primera contiene el recorrido en preorden de un árbol binario (cuyos nodos tenían valores entre 1 y 10^6) y la segunda el recorrido en inorden de ese mismo árbol.

Salida

Para cada caso, se escribirá el recorrido en postorden del árbol reconstruido.

Entrada de ejemplo

```
1 2 3 4 5 6 7
3 2 4 1 5 7 6
2 4 6
2 4 6
1 2 4 8 5 3 6 7
8 4 2 5 1 6 3 7
```

Salida de ejemplo

```
3 4 2 7 6 5 1
6 4 2
8 4 5 2 6 7 3 1
```

Autor: Alberto Verdejo.