

# Elemento mínimo de un árbol

Dado un árbol binario no vacío, cuyos nodos almacenan valores que se pueden ordenar (se supone que tienen definido el operador  $<$ ), queremos saber el elemento menor.

*Requisitos de implementación.*

El problema puede resolverse de diferentes maneras: extendiendo la clase `bintree` con un método público que devuelva el menor elemento; implementando una función externa a la clase recursiva, que explore el árbol buscando el menor; utilizando iteradores. En cualquier caso, la función tiene que ser genérica, es decir, no puede conocer el tipo `T` de los elementos almacenados en el árbol. Además, el coste de la operación debe ser lineal en el número de nodos del árbol.

## Entrada

Cada caso de prueba ocupa dos líneas. En la primera aparecerá una `N` si el árbol es de números enteros positivos, o una `P` si el árbol es de palabras. En la segunda línea aparecerá la descripción del árbol: primero la raíz, después la descripción del hijo izquierdo y después la descripción del hijo derecho. Si el árbol es de números, se utilizará `-1` para indicar el árbol vacío; si es de palabras, se utilizará `#`.

## Salida

Para cada árbol se escribirá una línea con el valor menor almacenado en sus nodos. En el caso de un árbol de palabras se entiende que el orden es el que proporciona el operador  $<$  del tipo `string`.

## Entrada de ejemplo

```
N
5 4 1 -1 -1 3 -1 -1 2 -1 -1
P
ramon luis maria # # carlos # # sara # #
```

## Salida de ejemplo

```
1
carlos
```

**Autor:** Alberto Verdejo.