
Mínimo y máximo de un BST

Y96513_es

Dado un árbol binario de búsqueda (BST) de enteros no vacío, implementa dos funciones que devuelvan el valor mínimo y el valor máximo del árbol, respectivamente.

```
/**
 * @brief Devuelve el valor mínimo de un BST.
 *
 * @param t Un árbol binario de búsqueda.
 * @pre `t` no está vacío.
 * @returns El valor mínimo de `t`.
 */
int bst_minim(BinTree<int> t);

/**
 * @brief Devuelve el valor máximo de un BST.
 *
 * @param t Un árbol binario de búsqueda.
 * @pre `t` no está vacío.
 * @returns El valor máximo de `t`.
 */
int bst_maxim(BinTree<int> t);
```

Entrada

La entrada consiste en una secuencia de árboles binarios de búsqueda de enteros no vacíos, en formato visual.

Salida

Para cada árbol, una línea con el valor mínimo y el valor máximo, separados por un espacio.

Observación

Los ficheros públicos (icono del gatito) son: la clase `BinTree` (fichero `bintree.hh`), la entrada/salida de `BinTree` (`bintree-io.hh`) y el programa principal. También hay un `Makefile` y el directorio `.vscode` con la configuración para compilar y depurar con VSCode.

Debes implementar `bst_minim` y `bst_maxim` en un **fichero .cc nuevo**, compilar (está preparado para poder compilar y depurar con VSCode), y finalmente **enviar solo el fichero con las dos funciones**.

Ejemplo de entrada

```
visual
10
|-- 5
|   |-- 2
|   '--- 7
```

```
|  '--- 15
|      |-- 12
|      '--- 20
|
|  3
|
|  50
```

```
|-- 25
|   |-- 10
|   |   |-- 5
|   |   |-- #
|   |-- 30
|-- 75
    |-- #
    |-- 100
```

Ejemplo de salida

```
2 20
3 3
5 100
```

Información del problema

Autoría: Pau Fernández

Traducción: Pau Fernández

Generación: 2026-03-12T15:58:05.303Z

© *Jutge.org*, 2006–2026.

<https://jutge.org>