

---

## Mínim i màxim d'un BST

Y96513\_ca

---

Donat un arbre binari de cerca (BST) d'enters no buit, implementa dues funcions que retornin el valor mínim i el valor màxim de l'arbre, respectivament.

```
/**
 * @brief Retorna el valor mínim d'un BST.
 *
 * @param t Un arbre binari de cerca.
 * @pre `t` no és buit.
 * @returns El valor mínim de `t`.
 */
int bst_minim(BinTree<int> t);

/**
 * @brief Retorna el valor màxim d'un BST.
 *
 * @param t Un arbre binari de cerca.
 * @pre `t` no és buit.
 * @returns El valor màxim de `t`.
 */
int bst_maxim(BinTree<int> t);
```

### Entrada

L'entrada consisteix en una seqüència d'arbres binaris de cerca d'enters no buits, en format visual.

### Sortida

Per a cada arbre, una línia amb el valor mínim i el valor màxim, separats per un espai.

### Observació

Els fitxers públics (icona del gatet) són: la classe `BinTree` (`fitxerbintree.hh`), l'entrada/sortida de `BinTree` (`bintree-io.hh`) i el programa principal. També hi ha un `Makefile` i el directori `.vscode` que té la configuració per compilar i debuggar amb VSCode.

Has d'implementar `bst_minim` i `bst_maxim` en un **fitxer .cc nou**, compilar (està preparat per poder compilar i debuggar amb VSCode), i finalment **enviar només el fitxer amb les dues funcions**.

### Exemple d'entrada

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

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

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

### Exemple de sortida

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

### Informació del problema

Autoria: Pau Fernández

Generació: 2026-03-12T15:58:02.584Z

© *Jutge.org*, 2006–2026.

<https://jutge.org>