
Arbre binari enfilat. Recorregut en inordre

X69521_ca

Donada la classe *Abin* que permet gestionar arbres binaris enfilats usant memòria dinàmica, cal implementar el mètode

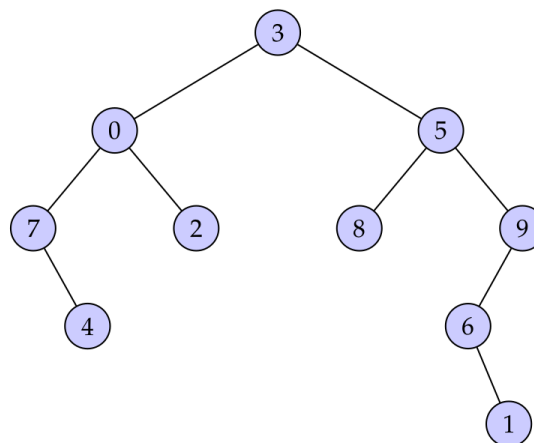
```
list <T> inordre() const;
```

que retorna una llista amb els elements de l'arbre en inordre **sense utilitzar recursivitat ni estructures de dades addicionals**, aprofitant que l'arbre binari ja està enfilat.

Cal enviar a jutge.org només la implementació del mètode *inordre*. Indica dins d'un comentari a la capçalera del mètode el seu cost en funció del nombre d'elements *n* de l'arbre binari. Per testejar la solució, jutge.org ja té implementats la resta de mètodes de la classe *Abin* i un programa principal que llegeix un arbre binari, després crida el mètode *inordre* i finalment imprimeix els elements de la llista.

Entrada

L'entrada consisteix en la descripció d'un arbre binari d'enters (el seu recorregut en preordre amb les fulles marcades amb un -1). Per exemple, l'arbre (mira el PDF de l'enunciat)



es descriuria amb

```
3 0 7 -1 4 -1 -1 2 -1 -1 5 8 -1 -1 9 6 -1 1 -1 -1 -1
```

Sortida

Una línia amb els elements de l'arbre en inordre i separats per un espai.

Observació

Cal enviar la solució (el fitxer *solution.cpp*) comprimida en un fitxer *.tar*:

```
tar cvf solution.tar solution.cpp
```

Només cal enviar la implementació del mètode *inordre* i el seu cost en funció del nombre d'elements *n* de l'arbre binari. Segueix estrictament la definició de la classe de l'enunciat.

Exemple d'entrada 1

3 0 7 -1 4 -1 -1 2 -1 -1 5 8 -1 -1 9 6 -1 1 -1 -1 -1

Exemple de sortida 1

7 4 0 2 3 8 5 6 1 9

Exemple d'entrada 2

3 0 7 -1 -1 2 -1 4 -1 6 -1 -1 5 8 9 1 -1 -1 -1 -1 -1

Exemple de sortida 2

7 0 2 4 6 3 1 9 8 5

Exemple d'entrada 3

3 -1 -1

Exemple de sortida 3

3

Exemple d'entrada 4

-1

Exemple de sortida 4

Informació del problema

Autoria: Jordi Esteve

Generació: 2026-01-25T21:22:28.355Z

© *Jutge.org*, 2006–2026.

<https://jutge.org>