
Arbre binari. Calcula arbre amb factors de pes**X53343_ca**

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

```
void arbre_factors_pes ();
```

que modifica el contingut de l'arbre per tal de guardar a cada node el factor de pes (diferència entre la quantitat de nodes del fill esquerra i la quantitat de nodes del fill dret).

Cal enviar a jutge.org la següent especificació de la classe *Abin* i la implementació del mètode dins del mateix fitxer. Indica dins d'un comentari a la capçalera del mètode el seu cost en funció del nombre d'elements *n* de l'arbre.

```
include <cstdlib>
```

```
#include <iostream>
```

```
using namespace std;
```

```
typedef unsigned int nat;
```

```
template <typename T>
```

```
class Abin {
```

```
public:
```

```
    Abin(): _arrel (NULL) {};
```

```
    // Pre: cert
```

```
    // Post: el resultat és un arbre sense cap element
```

```
    Abin(Abin<T> &ae, const T &x, Abin<T> &ad);
```

```
    // Pre: cert
```

```
    // Post: el resultat és un arbre amb un element i dos subarbres
```

```
    // Les tres grans
```

```
    Abin(const Abin<T> &a);
```

```
    ~Abin();
```

```
    Abin<T>& operator=(const Abin<T>& a);
```

```
    // operador « d'escriptura
```

```
    template <class U> friend std::ostream& operator<<(std::ostream&, const Abin<U> &a);
```

```
    // operador » de lectura
```

```
    template <class U> friend std::istream& operator>>(std::istream&, Abin<U> &a);
```

```
    // Modifica el contingut de l'arbre per tal de guardar a cada node el factor
```

```
    // de pes (diferència entre quantitat nodes fill esquerra i quantitat nodes fill dret).
```

```
    void arbre_factors_pes ();
```

```
private:
```

```
    struct node {
```

```
        node* f_esq;
```

```
        node* f_dret;
```

```
        T info;
```

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 i després crida el mètode *arbre_factors_pes*.

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

Només cal enviar la classe requerida i la implementació del mètode *arbre_factors_pes* amb el seu cost en funció del nombre d'elements n de l'arbre. Pots ampliar la classe amb mètodes privats. Segueix estrictament la definició de la classe de l'enunciat.

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

$$\begin{array}{r}
 [7] \\
 \backslash _ [8] \\
 | \quad \backslash _ [4] \\
 | \quad | \quad \backslash _ [3] \\
 | \quad | \quad | \quad \backslash _ . \\
 | \quad | \quad | \quad \backslash _ . \\
 | \quad | \quad \backslash _ [6] \\
 | \quad | \quad \backslash _ . \\
 | \quad | \quad \backslash _ . \\
 | \quad \backslash _ [9] \\
 | \quad \backslash _ . \\
 | \quad \backslash _ . \\
 \backslash _ [5] \\
 \quad \backslash _ . \\
 \quad \backslash _ . \\
 \\
 [-4] \\
 \backslash _ [-2] \\
 | \quad \backslash _ [0]
 \end{array}$$

```

|      |      \__[0]
|      |      | \__.
|      |      | \__.
|      |      \__[0]
|      |      | \__.
|      |      | \__.

```

Exemple d'entrada 2

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

```

|      \__[0]
|      | \__.
|      | \__.
\__[0]
| \__.
| \__.

```

Exemple de sortida 2

```

\__[5]
| \__[7]
| \__.
| \__[6]
| \__[1]
| \__.
| \__.
| \__.
| \__[4]
| \__.
| \__.
\__[0]
| \__[2]
| \__.
| \__.
\__[7]
| \__[4]
| \__.
| \__.
\__.

[-1]
\__[-2]
| \__[2]
| \__.
| \__[-1]
| \__[0]
| \__.
| \__.
| \__[0]
| \__.
| \__.
\__[1]
| \__[0]
| \__.
| \__.
\__[-1]
| \__[0]
| \__.
| \__.
\__.

```

Exemple d'entrada 3

-1

Exemple de sortida 3

```

.
.

```

Exemple d'entrada 4

$$\begin{matrix} 3 & -1 & -1 \end{matrix}$$

Exemple de sortida 4

[3]
 __.
 __.

$$\begin{array}{l} [0] \\ \backslash _ . \\ \backslash _ . \end{array}$$

Exemple d'entrada 5

$$3 \quad 2 \quad -1 \quad -1 \quad -1$$

Exemple de sortida 5

$$\begin{array}{l} [3] \\ \backslash _ \\ \backslash _ [2] \\ \quad \backslash _ \\ \quad \backslash _ \end{array}$$

```
[1]
 \___.
 \___[0]
      \___.
      \___.
```

Exemple d'entrada 6

$$3 \quad -1 \quad 2 \quad -1 \quad -1$$

Exemple de sortida 6

$$\begin{array}{c} [3] \\ \diagdown \quad \diagup [2] \\ | \qquad \qquad \diagdown \quad \diagup \\ | \qquad \qquad \diagdown \quad \diagup \\ \diagdown \quad \diagup \end{array}$$
$$\begin{array}{c} [-1] \\ \backslash ___ [0] \\ | \qquad \backslash ___ . \\ | \qquad \backslash ___ . \\ \backslash ___ . \end{array}$$

Exemple d'entrada 7

-3 -2 -1 -1 -4 -1 -1

Exemple de sortida 7

$$\begin{array}{r} [-3] \\ \backslash _ [-4] \\ | \quad \backslash _ . \\ | \quad \backslash _ . \\ \backslash _ [-2] \\ \quad \backslash _ . \\ \quad \backslash _ . \end{array}$$
$$\begin{array}{l} [0] \\ \backslash _ [0] \\ | \quad \backslash _ . \\ | \quad \backslash _ . \\ \backslash _ [0] \quad \backslash _ . \\ \quad \quad \backslash _ . \end{array}$$

Informació del problema

Autoria: Jordi Esteve

Generació: 2026-01-25T16:50:27.174Z

© *Jutge.org*, 2006–2026.
<https://jutge.org>

