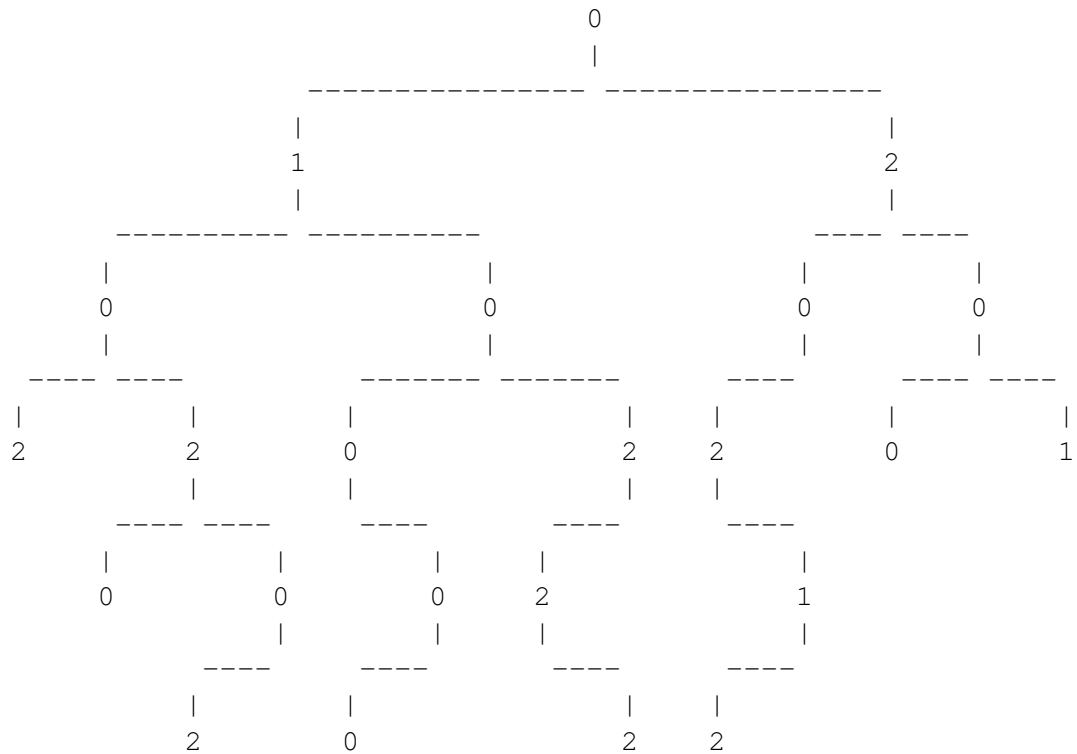

(Arbre) Nombre de fulles amb mateix valor que l'arrel X91879_ca

En aquest exercici afegirem un nou mètode `numLeavesLikeRoot` a la classe `Arbre`, que es cridarà només quan l'arbre sigui no buit, per a calcular el nombre de fulles que tenen el mateix valor que l'arrel.

Per exemple, suposeu que aquest és l'arbre representat per la variable `a` de tipus `Arbre`:



Llavors, la crida `a.numLeavesLikeRoot()` ha de retornar 3.

D'entre els fitxers que s'adjunten en aquest exercici, trobareu `Arbre.hh`, a on hi ha una implementació de la classe genèrica `Arbre`. Haureu de buscar dins `Arbre.hh` les següents línies i implementar el mètode que s'hi indica (afegint també algun mètode auxiliar si cal):

```

// Pre: El paràmetre implícit representa un arbre no buit.
// Post: Retorna el nombre de fulles de l'arbre representat pel paràmetre imp
//       que tenen el mateix valor que l'arrel.
// Descomenteu les següents dues línies i implementeu el mètode:
// int numLeavesLikeRoot() const{
// }
private:

// Afegiu alguna funció auxiliar si ho considereu oportú.

```

Podeu suposar que el tipus genèric `T` de la classe té predefinida la operació de comparació `==`. De fet, es testejarà la vostra implementació amb el tipus `T=int`. Ara bé, una solució que

no sigui genèrica es considerarà incorrecta i serà invalidada a posteriori, encara que superi els jocs de proves.

D'entre els fitxers que s'adjunten a l'exercici també hi ha `main.cc` (programa principal), i el podeu compilar directament, doncs fa `include` de `Arbre.hh`. Només cal que pugueu `Arbre.hh` al jutge.

Entrada

L'entrada conté un nombre arbitrari d'arbres. Cada cas consisteix en una descripció d'un arbre binari d'enters. La descripció consisteix en un recorregut en preordre dels nodes de l'arbre, amb marques on hi anirien els arbres buits. Fixeu-vos en que el programa que us oferim ja s'encarrega de llegir aquestes entrades. Només cal que implementeu els mètodes abans esmentats.

Sortida

Per a cada cas, la sortida conté la corresponent avaluació de l'arbre. Fixeu-vos en que el programa que us oferim ja s'encarrega d'escriure aquesta avaluació. Només cal que implementeu els mètodes abans esmentats.

Exemple d'entrada 1

0 1 0 # # 2 # # 2 # 0 # #	1 2 2 0 # # 1 # # 1 # # 0 # 0 0 # # 2 # #
2 2 0 0 # # # 0 # # 2 # #	0 2 2 # # 2 # # 0 0 1 2 # # # 1 # 2 # # 2 1 # # 0 0 # #
1 0 1 # # # 1 # 0 1 # # #	2 2 1 # # 1 # # 0 0 0 1 # # 0 # # 2 2 # # 1 # # 2 2 0 0 # #
2 # 0 0 1 # 2 # 0 # # 1 0 # # 1 0 # # # 2	0 2 # # #
0 # #	1 1 1 1 2 # # 2 # # 2 # # # 2 # #
2 0 1 # # 0 # # 2 2 0 # 0 # # 1 2 # # 1 #	2 2 # # 1 # #
1 0 0 1 # 0 # # 0 # 2 # # 1 # 1 2 # # 2 #	1 # 0 2 # # #
0 0 2 0 # # # 2 0 0 # # 0 # # # 0 2 1 1 #	0 2 # 0 # # 1 # #
2 0 2 # # 0 # # #	# 1 2 1 # 2 # # 0 # # 1 # # # #
2 # 2 0 # # 1 # #	2 0 2 # # 1 2 # # 0 2 # # 0 2 # #
0 # 1 1 # # #	0 1 0 2 # # 0 # # 0 # #
2 # #	1 1 2 0 # # 1 # # 0 0 1 2 # # 0 # # 2 # # # 0 0 # # 2 # #
2 2 # # 2 # #	1 # 2 # #
1 2 # 2 1 # # 2 # # 0 # #	0 0 0 2 # # 2 # # 1 # # #
1 2 # # 1 # #	2 2 1 0 # # 2 1 # # 0 # # # 1 2 # # 0 2 # # 1 2 # # #
2 0 1 # # 0 # # 0 2 2 0 # 1 # # 0 # # 2 0	0 0 2 # # 0 # # 2 2 # # 0 # #
0 2 # 2 2 # # 2 # #	0 0 2 0 # # # 1 0 # # # 0 0 0 # # 0 # # 0 # #
0 2 1 2 # 0 # # 0 # # # 0 2 2 # # # 2 # #	# 1 # # 1 2 2 0 # # 2 # #
1 1 # # 0 # #	0 2 # # 1 # #
2 1 # # 2 0 # 0 # # 0 1 2 0 # # 0 # # 2 #	2 2 # # 2 0 2 0 2 # # # 1 # 1 # # 2 # # 2 # #
0 2 # 1 1 1 2 # # 1 # # 1 1 # # 2 # # 1 #	0 0 0 # # 1 # # 1 1 # 0 2 # # 2 # # 1 0 # # 1 2 # #
1 2 # # 0 # 2 # #	0 2 # # 1 0 0 # # 2 # #
0 2 0 0 1 # # 0 # # 1 1 # # # 2 # # 1 2 2	2 0 2 2 1 # # # # 1 2 # 1 # # 0 2 # # 2 # # 0 1 1 # 0 2
1 # 1 0 # # 2 # #	2 2 # # 2 # #
2 2 0 0 1 # # # 1 2 # # # 0 1 1 # # 1 # #	2 1 1 # # 0 # # 0 2 1 # # 1 # # 0 2 # # #
2 2 # # 2 # #	2 2 0 0 0 0 1 # # 0 2 # # 2 0 # # 2 # # 1 1 1 0
1 # 1 1 2 # # 2 # # 0 1 # # 0 # #	0 2 2 # # 2 # # 1 # #
2 0 2 1 # # 2 # # 2 2 # # # 0 # 1 1 # # #	1 # 2 2 1 # # 1 # # #
2 0 # 1 2 0 2 # # 1 # # 1 0 # # 2 # # 2 0	0 1 # # 0 # #
2 0 # # 2 # #	0 2 # # # #
1 0 1 2 0 # # 2 2 # # 0 # # 0 1 1 # # 0 #	0 0 # # 0 1 # #
2 1 0 2 2 # # 2 0 # # # 2 0 # # 0 # # 0 #	# 0 # # 0 # # 0 1 2 0 # # # 1 # # 2 0 2 0 # # 1 # # #
0 1 2 # 0 # # 2 2 # # 1 # # 1 2 # 2 # # #	2 0 # # 1 # # 2 # # 1 2 0 # # 0 # # 2 1 # # 0 # #
2 0 0 # # 1 # # 1 1 2 # # 1 # # 2 # 1 # #	1 0 2 1 # 0 0 # # 2 # # # 1 0 0 2 # # # 2 # 1 # # 2 # #
2 0 # # 1 0 2 0 # # # #	1 # 0 # #
	0 1 2 # # # 2 0 1 2 # # 1 # # 2 # # #

Exemple de sortida 1

2
1
2
0 # # 1 # #
1
1 # #
2
1
4
1
0
0
1
2
1
1
2
0
2
1
1
0
0
1
0
6
2
1
2
2
1
3
2
1
1
0
2
1
3
0
0
1
0
0
1
4
2
1
0
0
3
2
5
1
0
4
2
1
1
3

1	0
1	0
0	0
0	0
2	2
1	1
0	1
1	1
2	1
2	1
3	1

Observació

Avaluació sobre 10 punts: (Afegiu comentaris si el vostre codi no és prou clar)

- Solució lenta: 5 punts.
- solució ràpida: 10 punts.

Entenem com a solució ràpida una que és correcta, de cost lineal i capaç de superar els jocs de proves públics i privats. Entenem com a solució lenta una que no és ràpida, però és correcta i capaç de superar els jocs de proves públics.

Una solució que no sigui genèrica (per a qualsevol tipus T amb $==$) serà invalidada i rebrà nota 0, encara que superi els jocs de proves.

Informació del problema

Autoria: PRO2

Generació: 2026-01-27T18:57:10.912Z

© *Jutge.org*, 2006–2026.

<https://jutge.org>