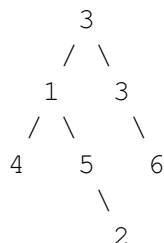

Màxima suma d'un camí descendent.**X58333_ca**

Preliminars

Considereu el següent arbre binari de naturals positius:



Com podeu observar, hi ha tres fulles amb valors 4, 2 i 6. Si agafem el camí descendent que ens porta des de l'arrel fins a la fulla amb un 4, i anem sumant tots els valors que trobem pel camí, obtenim $3+1+4 = 8$. En canvi, si agafem el camí que ens porta des de l'arrel fins a la fulla amb un 2, la suma és $3+1+5+2 = 11$. Pel cas de la fulla amb un 6, la suma obtinguda és $3+3+6 = 12$. Per tant, 12 és la suma màxima que podem obtenir en un camí descendent des de l'arrel fins a alguna fulla.

Fi de preliminars

Implementeu una funció **RECURSIVA** que, donat un arbre binari de naturals positius, retorna la suma màxima que es pot obtenir sumant els valors dels nodes que es troben en un camí descendent des de l'arrel fins a alguna fulla. En el cas de l'exemple d'arbre anterior, el resultat de la funció seria 12. Aquesta és la capçalera:

```
/**
 * @pre t conté naturals positius en els seus nodes.
 * @post Retorna la suma màxima que es pot obtenir sumant els valors que
 * es troben en un camí descendent des de l'arrel fins a alguna fulla.
 */
int maxSumDescPath(BinTree<int> t);
```

L'enunciat d'aquest exercici ja ofereix uns fitxers que haureu d'utilitzar per a compilar: `main.cc`, `BinTree.hh`, `maxSumDescPath.hh`. Us falta crear el fitxer `maxSumDescPath.cc` amb els corresponents `includes` i implementar-hi la funció anterior. Només cal que pugueu `maxSumDescPath.cc` al jutge.

Entrada

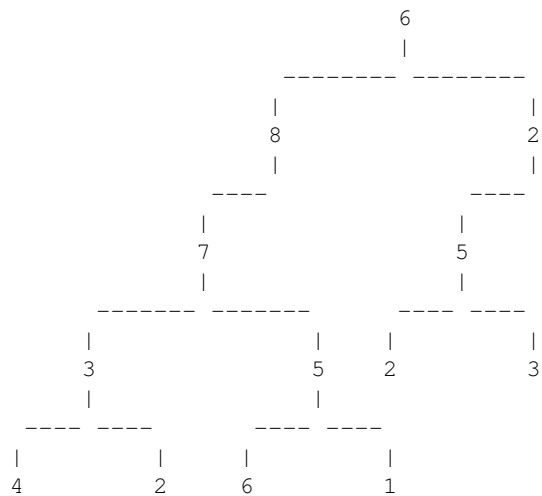
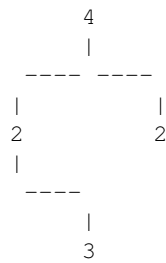
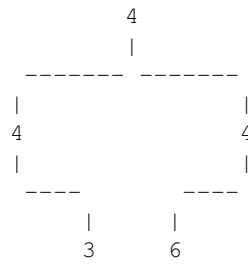
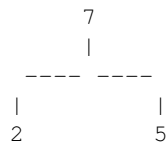
La primera línia de l'entrada descriu el format en el que es descriuen els arbres, o bé `INLINEFORMAT` o bé `VISUALFORMAT`. Després venen un nombre arbitrari de casos. Cada cas consisteix en una descripció d'un arbre binari d'enters. El programa que us oferim ja s'encarrega de llegir aquestes entrades. Només cal que implementeu la funció abans esmentada.

Sortida

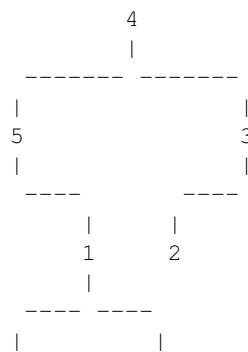
Per a cada cas, la sortida conté el corresponent resultat de cridar a la funció. El programa que us oferim ja s'encarrega d'escriure aquest resultat. Només cal que implementeu la funció `maxSumDescPath`.

Exemple d'entrada 1

VISUALFORMAT

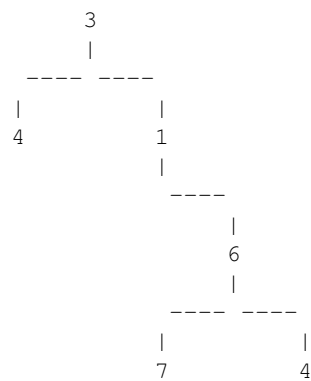
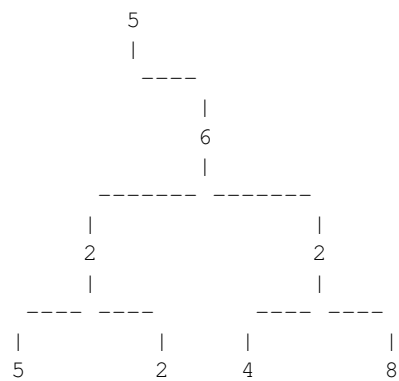


3

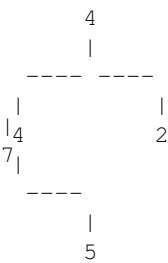


3

4



4



Exemple de sortida 1

12
14
9
32

3
14
21
17
4
13

Exemple d'entrada 2

```
INLINEFORMAT
7 (2, 5)
4 (4 (, 3), 4 (6, ))
4 (2 (, 3), 2)
6 (8 (7 (3 (4, 2), 5 (6, 1)), ), 2 (5 (2, 3), 7))
3
4 (5 (, 1 (3, 4)), 3 (2, ))
5 (, 6 (2 (5, 2), 2 (4, 8)))
3 (4, 1 (, 6 (7, 4)))
4
4 (4 (, 5), 2)
5 (1 (4 (1 (6, ), 8), 3 (6 (3, 4), 1)), 4 (7, 3 (, 2 (5, 7)) (3, 1))
5 (7 (3 (3 (, 6 (5, 7)), 8 (, 5)), ), 3 (6 (2 (, 5), 2 (3 (, 6 (7, 1 (, 4))), ))
8 (2 (7, ), )
6 (8, 5)
3 (3 (, 7), 6 (3, ))
5 (1 (4 (2, ), 8 (2 (4, ), 2 (7, ))), 1 (, 6 (7 (, 5), 2 (4, 3 (6, ))
5 (6 (5, 5), 2 (, 8))
5 (6 (6 (7, 3), 1 (5, 1 (, 3 (, 2))) ), 6)
3
2 (5 (7, 4), 5 (1, ))
```

Exemple de sortida 2

12
14
9
32
3
14
21
17
4
13
21
31
14
13
24
31
24
3
14

Observació

La vostra funció i subfuncions que creeu han de treballar només amb arbres. Heu de trobar una solució **RECURSIVA** del problema. Avaluació sobre 10 punts:

- 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.

Informació del problema

Autoria: PRO2

Generació: 2026-01-25T17:07:03.220Z

© Jutge.org, 2006–2026.

<https://jutge.org>