
Avaluar expressions**Z82639_ca**

En aquest exercici considerarem arbres que representen expressions Booleanes sobre els operadors `and`, `or`, i `not`, amb els operands 0 (fals) i 1 (cert). Per exemple, el següent arbre

```
or
|-- and
|   |-- 1
|   '-- 0
|-- or
|   |-- not
|   |   '-- 0
|   |-- 1
|   '-- 0
'-- 0
```

representa l'expressió $(0 \wedge 1) \vee ((\neg 0) \vee 1 \vee 0) \vee 0$. Els nodes d'aquest arbre poden ser enters o operadors. Quan són operadors, els fills del node són els operands. Quan són operands, no tenen fills. La negació sempre té només un operand.

Implementeu, doncs, la funció següent:

```
/**
 * @brief Avalua un arbre no buit que representa una expressió Booleana.
 *
 * L'expressió és sobre l'1 (true) i el 0 (fals) i els operadors
 * 'and', 'or', i 'not'.
 *
 * @pre L'arbre és no buit i l'expressió és correcta, és a dir, els operands
 * 'and' i 'or' tenen més d'un operand, i l'operador 'not' en té només 1.
 *
 * @param t Arbre que representa l'expressió.
 * @return Resultat de l'avaluació de l'expressió.
 */
bool evaluate(Tree<string> t);
```

Observació

Els fitxers públics (icona del gatet) contenen:

<code>tree.hh</code>	la classe <code>Tree</code>
<code>tree-io.hh</code>	l'entrada/sortida de <code>Tree</code>
<code>eval.hh</code>	la declaració de la funció a implementar
<code>main.cc</code>	el programa principal

També hi ha un `Makefile` i el directori `.vscode` que té la configuració per compilar i depurar amb VSCode.

Cal implementar `evaluate` en un fitxer `.cc` nou, compilar, i finalment enviar només el fitxer amb la funció.

Entrada

Cada cas consisteix en una representació textual d'una expressió del tipus definit. (Aquesta lectura ja la fa el programa principal.)

Sortida

Per a cada cas, la sortida és el resultat d'avaluar l'expressió, cada resultat en una línia separada. (Això també ho fa el programa principal.)

Exemple d'entrada

```
and
|-- 1
'-- 1

and
|-- 0
|-- 0
'-- and
    |-- 0
    '-- 0

and
|-- 1
|-- 0
|-- 0
'-- 1

or
|-- and
|   |-- 1
|   |-- 0
|   '-- 0
|-- or
|   |-- 0
|   |-- 1
|   |-- 0
|   '-- 0
'-- 0

or
|-- 0
|-- or
|   |-- 1
|   |-- 1
|   |-- 1
|   '-- 0
|-- 0
'-- 1
```

Exemple de sortida

```
1
0
0
1
1
```

Informació del problema

Autor : Pau Fernández

Generació : 2025-03-20 18:03:43

© Jutge.org, 2006–2025.

<https://jutge.org>