
Árbol de múltiplos

T56798_es

Decimos que un árbol binario de números naturales T es un árbol "de múltiplos" cuando, para todo nodo que no sea una hoja, los valores de las raíces de los subárboles izquierdo y derecho, E y D , si no son vacíos, son múltiplos del valor en la raíz de T . Consideraremos que un múltiplo de n es un natural $m = nk$ tal que $k \geq 1$. Por ejemplo, el siguiente árbol es de múltiplos:

```
1
|- 2
|  |- 4
|  ' - #
|- 3
   |- 9
   ' - 6
      |- 6
      ' - 12
```

Haz una función `arbre_de_multiples` con cabecera:

```
/**
 * @brief Determina si un árbol es "de múltiplos"
 *
 * @param t Un árbol binario de naturales
 *
 * @returns `true` si `t` es un árbol de múltiplos
 *          `false` en caso contrario.
 */
bool arbre_de_multiples(BinTree<int> t);
```

Observación

Los archivos públicos (icono del gatito) contienen:

<code>main.cc</code>	el programa principal, con la entrada/salida ya hecha
<code>bintree.hh</code>	la clase <code>BinTree<T></code>
<code>bintree-io.hh</code>	la entrada/salida de <code>BinTree<T></code>
<code>bintree-inline.hh</code>	la entrada/salida "inline" de <code>BinTree<T></code>
<code>Makefile</code>	para compilar con <code>make</code>
<code>.vscode</code>	para compilar y debuggar con VSCode

Hay que implementar `arbre_de_multiples` en un **archivo .cc nuevo**, compilar, y finalmente **enviar solo el archivo con la función** y funciones auxiliares si son necesarias.

Entrada

(Esto ya lo hace el programa principal dado). La entrada comienza con "visual" o "inline" para indicar el formato de los árboles de entrada. Después viene una secuencia de árboles en el formato indicado.

Salida

(Esto también lo hace el programa principal dado). La salida son los strings resultantes de llamar a la función `arbre_de_multiples`, un resultado por línea.

Ejemplo de entrada 1

```
visual
5

1
|-- 2
'-- 1

1
|-- 3
'-- 5

2
|-- 3
'-- 1

1
|-- 2
'-- 3

3
|-- 3
'-- 1
    |-- 3
    '-- 2

2
|-- 6
|   |-- 30
|   '-- 12
'-- 4
```

Ejemplo de salida 1

```
si
si
si
no
si
no
si
```

Ejemplo de entrada 2

```
inline
5
1(2,1)
1(3,)
2(3,1)
1(2,3)
3(3,1(3,2))
2(6(30,12),4)
```

Ejemplo de salida 2

```
si
si
si
no
si
no
si
```

Información del problema

Autoría: M^a Lluïsa Bonet i Pau Fernández

Generación: 2026-03-24T13:59:10.597Z

© *Jutge.org*, 2006–2026.

<https://jutge.org>