

---

**Subseqüències en camins d'un arbre (MakePRO2)**

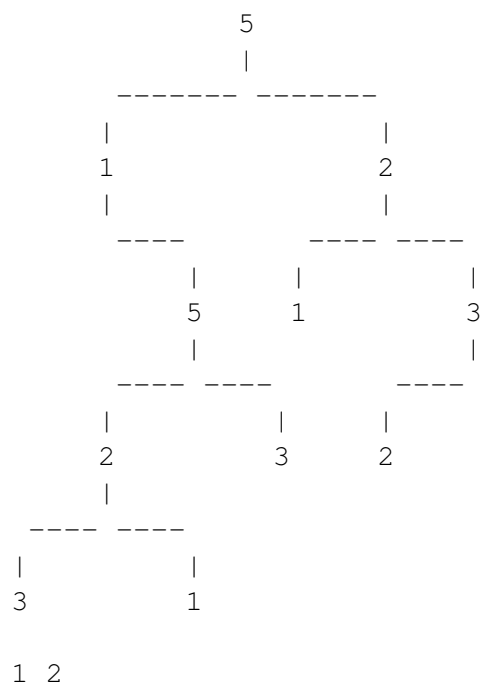
**X16746\_ca**

---

Heu d'implementar un programa que llegirà un arbre d'enters  $t$  i també diverses seqüències d'enters. Per a cada seqüència  $s$ , el programa haurà de calcular quantes fulles hi ha a  $t$  tals que els elements de  $s$  es troben en el camí des de l'arrel fins a aquella fulla, en el mateix ordre, i a on possiblement hi pot haver també altres valors. Recordeu que una fulla és un arbre amb un únic node, i que per tant els seus dos fills directes són arbres buits.

Per exemple, considereu el següent arbre i seqüència:

Entrada:



Sortida:

2

En total hi ha 2 fulles tals que en el camí des de l'arrel fins a qualsevol d'aquelles fulles ens hi apareix la seqüència  $s$  (en el mateix ordre, i també enmig d'altres valors). Mostrem a continuació quins serien els 2 camins, que indiquem amb els valors dels nodes visitats:

```
5, 1, 5, 2, 3
5, 1, 5, 2, 1
```

Fixeu-vos que l'enunciat d'aquest exercici ja ofereix uns fitxers que haureu d'utilitzar per a compilar: `Makefile`, `BinTree.hh`. Us falta crear el fitxer `program.cc` amb els corresponents `includes` i implementar-hi el programa que hem descrit. Quan pugueu la vostra solució al jutge, només cal que pugueu un tar construït així:

```
tar cf solution.tar program.cc
```

De cara a llegir el format d'entrada i l'arbre, us recomanem aquestes línies:

```
string format;
getline(cin, format);
BinTree<int> t;
t.setInputOutputFormat (format=="INLINEFORMAT"? BT::INLINEFORMAT : BT::VI
cin >> t;
cin.ignore();
```

## Entrada

La primera línia de l'entrada descriu el format en el que es descriuen els arbres, o bé IN-  
LINEFORMAT o bé VISUALFORMAT.

Després ve la descripció d'un únic arbre binari d'enters, l'arbre t.

Cadascuna de les següents línies conté una seqüència d'enters. La podeu llegir i emmagatzemar com considereu convenient. Però penseu bé com ho feu (*queue*, *stack*, *list*, *vector*), doncs hi haurà maneres que faran més fàcil implementar un programa eficient.

## Sortida

Per a cadascuna de les seqüències hi ha un valor de sortida en una línia, el nombre de fulles de t que compleixen la condició abans esmentada.

### Exemple d'entrada 1

VISUALFORMAT

```

          1
          |
      -----
      |           |
      5           2
      |
      -----
      |           |
      3           0
      |
      -----
      |
      3
      |
      -----
      |
      6

1 0
1 5
3 3
1 2
1 3 3
2
1
```

### Exemple de sortida 1

```
1
2
1
1
1
1
1
3
```

### Exemple d'entrada 2

INLINEFORMAT

```
6(2(0(1(5(4(,2(8,)),4(10(10,3),7(10,))),4(7(0,7)),),6),7(5(4(3,10),1),9(,10(6(1,),3(10(6,),))))))
6 2 1 10 3
```

```
6
6 2 0
7 5 1
(7 0 7),),6),7(5(4(3,10),1),9(,10(6(1,),3(10(6,),))))))
```

6 10  
6  
6 7  
6  
  
5  
6 0  
6 2  
7 4  
6 1  
6 2  
6 2 6  
6  
6 10  
9

## Exemple de sortida 2

1  
12  
6  
1  
2  
7  
12  
7  
12  
12  
7  
6  
7  
2  
8  
7  
1  
12  
7  
2

### Observació

Aquest exercici requereix d'una bona optimització per tal de superar els jocs de proves privats.

### Informació del problema

Autor : PRO2

Generació : 2023-10-17 02:31:59

© *Jutge.org*, 2006–2023.

<https://jutge.org>