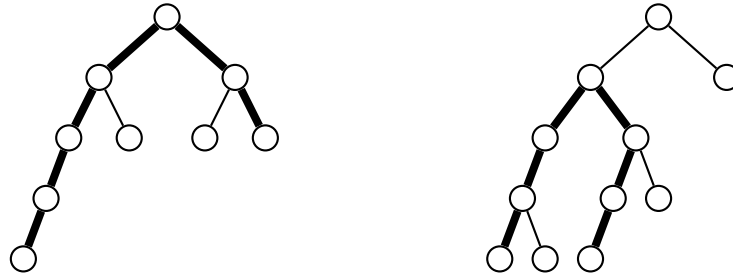


Diámetro de un árbol binario

Definimos un *camino* en un árbol binario como una secuencia de nodos $n_1 n_2 \dots n_k$ sin repeticiones (por cada nodo del árbol se pasa como mucho una vez) tal que para todo par de nodos consecutivos $n_i n_{i+1}$ ($1 \leq i < k$) uno de ellos siempre es padre del otro (n_i es padre de n_{i+1} o n_{i+1} es padre de n_i). Definimos la *longitud* de un camino $n_1 n_2 \dots n_k$ como el número de nodos que lo forman, k . Y definimos el *diámetro* de un árbol como la longitud del camino más largo del árbol.

Por ejemplo, los dos árboles siguientes tienen diámetro 7 y un camino de esa longitud aparece resaltado con trazo más grueso en los árboles.



Dado un árbol binario queremos averiguar su diámetro.

Requisitos de implementación.

Se implementará una función externa a la clase `bintree` que explore el árbol averiguando su diámetro. Esta función debe tener un coste lineal con respecto al número de nodos del árbol.

Entrada

La entrada comienza indicando el número de casos de prueba que vendrán a continuación. Cada caso consiste en una cadena de caracteres con la descripción de un árbol binario: el árbol vacío se representa con un '.'; un árbol no vacío se representa con un '*' (que denota la raíz), seguido primero de la descripción del hijo izquierdo y después de la descripción del hijo derecho.

Salida

Para cada caso, se escribirá una línea con el diámetro del árbol correspondiente.

Entrada de ejemplo

```
5
.
*..
*.*..
*****.....**.....
*****.*.....**.....*..
```

Salida de ejemplo

```
0
1
2
7
7
```

Autor: Alberto Verdejo.