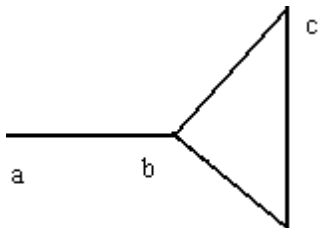


Proyecto # 3: Un Bosque de Árboles

Descripción del objeto de estudio

Un grafo es un conjunto de puntos o vértices $V(G)$, sumado a un conjunto de aristas $E(G)$, donde cada elemento de $E(G)$ consiste en la conexión de un par de puntos de $V(G)$.

Por ejemplo, sea G un grafo donde $V(G) = \{a, b, c, d\}$ y $E(G) = \{(a, b), (b, c), (c, d), (d, b)\}$.



Notemos que G contiene el ciclo $\{(b, c), (c, d), (d, b)\}$. Un grafo desprovisto de ciclos se le llama árbol.

Un camino, en un grafo G , consiste en una secuencia alternativa de vértices y aristas (iniciando y finalizando con un vértice) tal que todos los vértices del camino son distintivos. En el grafo del ejemplo, $\{a, (a, b), b, (b, c), c, (c, d), d\}$ es un camino.

Cada dos vértices de un grafo están enlazados por un único camino. Un grafo se le denomina conexo si cada par de vértices están enlazados por un camino. El grafo del ejemplo es conexo. Si un grafo no es conexo, entonces está conformado por subgrafos que son así. A cada uno de estos subgrafos se les conoce como componente conexa del grafo G , y se les llama **árbol**.

Un grafo compuesto por árboles, se le conoce como un **bosque**. Existen casos en que los árboles componentes del grafo tienen vértices, pero no aristas que los enlacen. A estos árboles, que lucen como puntos aislados, se les llama **bellotas**.

El problema

Sea un bosque, debes implementar una aplicación informática desarrollada en el Lenguaje Rust, que cuente el número de árboles y bellotas que lo componen.

Para lograr este objetivo, **no se permite el uso de librerías o elementos externos complementarios al lenguaje de programación**. Solo se podrá utilizar las librerías nativas del lenguaje que trae la instalación estándar.

Especificación de la entrada y salida

La entrada de datos viene dada mediante un archivo de texto plano (tree.in) con el siguiente formato:

- Una lista de vértices del grafo, separados por comas (una línea de, máximo, 26 vértices representados por letras mayúsculas, A-Z).
- La lista de aristas del grafo (una por línea, entre paréntesis y separadas por comas, finalizada por un grupo de asteriscos).

```
A, B, C, D, E, F, G, H, I, J, K, L, M, N
(A, B)
(B, C)
(B, D)
(D, E)
(E, F)
(B, G)
(G, H)
(G, I)
(J, K)
(K, L)
(K, M)
*****
```

La salida será por consola (salida estándar) y consistirá en una frase única que expresará el número de árboles y bellotas detectadas. Por ejemplo, "Hay **X** árbol(es) y **Y** bellota(s).", donde X e Y son los números de árboles y bellotas, respectivamente.

```
Hay 2 árbol(es) y 1 bellota(s)
```

Entrega

- La entrega se realizará por medio de la plataforma Google Classroom.
- Se consignarán el pseudocódigo (pro3.pse) y el código fuente en el lenguaje de programación Rust (.rs), ambos **autodocumentados** y con la identificación del participante (nombre, apellido y cédula).
- NO SE ACEPTARÁN ARCHIVOS COMPRIMIDOS

Primero que funcione, luego se mejora!!!
@_Nasch_

Programar es divertido hasta que llega el día en que la solución NO está en StackOverflow