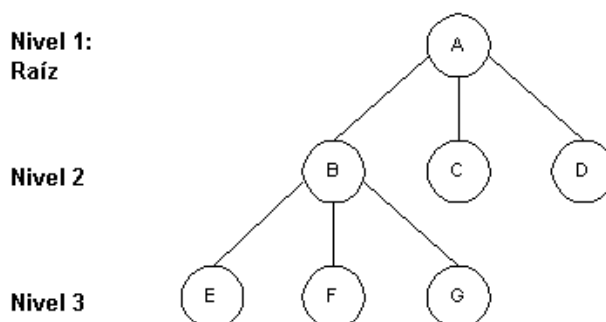


### 4.2.1 El enfoque jerárquico

Un DBMS jerárquico utiliza jerarquías o árboles para la representación lógica de los datos. Los archivos son organizados en jerarquías, y normalmente cada uno de ellos se corresponde con una de las entidades de la base de datos. Los árboles jerárquicos se representan de forma invertida, con la raíz hacia arriba y las hojas hacia abajo (*Figura 4.7*).

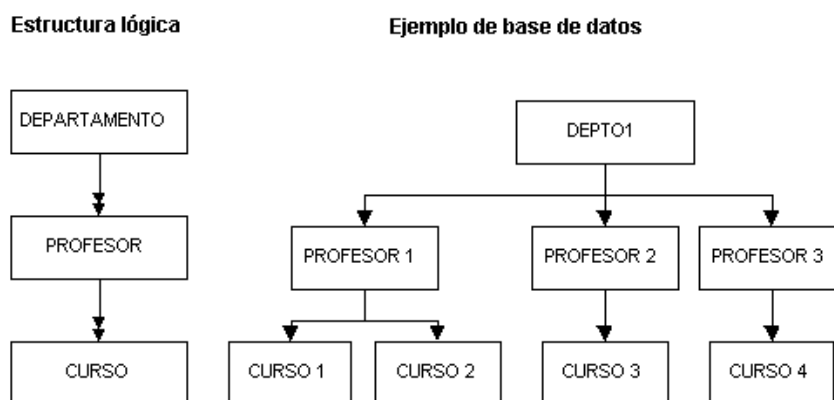


**Figura 4.7** Estructura de un árbol jerárquico

Un DBMS jerárquico recorre los distintos nodos de un árbol en un *preorden* que requiere tres pasos:

1. Visitar la raíz.
2. Visitar el hijo más a la izquierda, si lo hubiera, que no haya sido visitado.
3. Si todos los descendientes del segmento considerado se han visitado, volver a su padre e ir al punto 1.

Cada nodo del árbol representa un tipo de registro conceptual, es decir, una entidad. A su vez, cada registro o segmento está constituido por un número de campos que los describen – las propiedades o atributos de las entidades. Las relaciones entre entidades están representadas por las ramas. En la *Figura 4.8*, cada departamento es una entidad que mantiene una relación de uno a muchos con los profesores, que a su vez mantienen una relación de uno a muchos con los cursos que imparten.



**Figura 4.8** Base de datos jerárquica. Estructura lógica y ejemplo

A modo de resumen, enumeramos las siguientes características de las bases de datos jerárquicas:

1. Los segmentos de un archivo jerárquico están dispuestos en forma de árbol.
2. Los segmentos están enlazados mediante relaciones uno a muchos.
3. Cada nodo consta de uno o más campos.
4. Cada ocurrencia de un registro padre puede tener distinto número de ocurrencias de registros hijos.
5. Cuando se elimina un registro padre se deben eliminar todos los registros hijos (integridad de los datos).
6. Todo registro hijo debe tener un único registro padre excepto la raíz.

Las reglas de integridad en el modelo jerárquico prácticamente se reducen a la ya mencionada de eliminación en cadena de arriba a abajo. Las relaciones muchos a muchos no pueden ser implementadas de forma directa. Este modelo no es más que una extensión del modelo de ficheros.

Como ejemplos de DBMSs basados en este enfoque podemos citar el IMS de IBM Corporation y el SYSTEM 2000 de Intel Corporation.

[Anterior](#) | [Siguiente](#) | [Índice capítulo 4](#) | [Índice General](#)