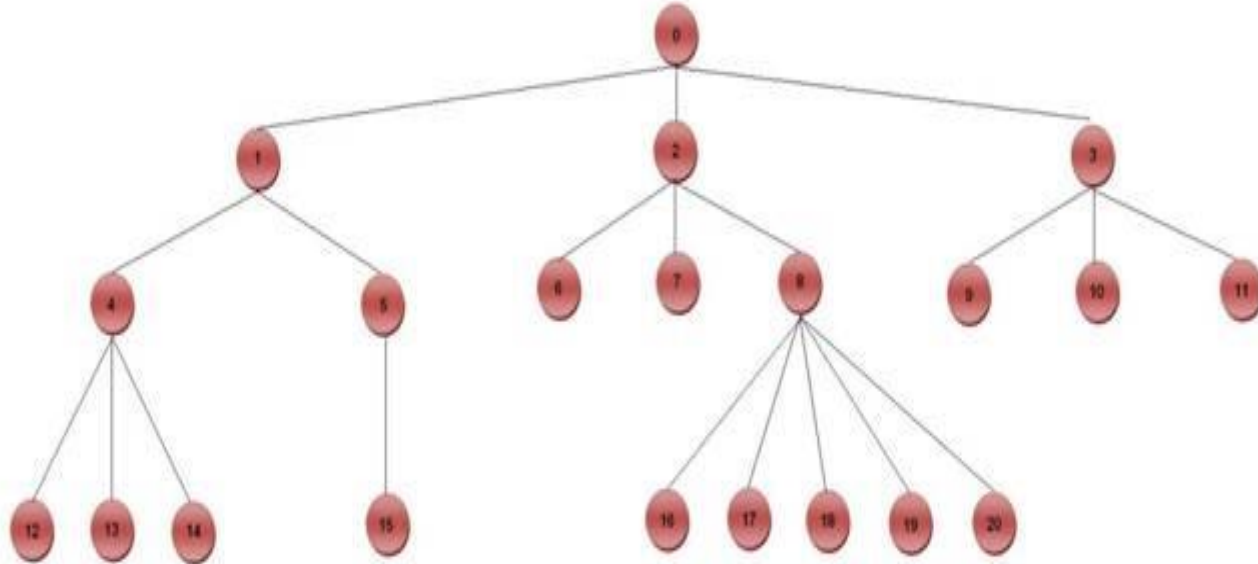


ARBOLES



Arboles: Introducción

- Las listas enlazadas son estructuras lineales
- Son flexibles pero son secuenciales, un elemento detrás de otro
- Los árboles junto con los grafos son estructuras de datos no lineales. En especial los árboles son estructuras **Jerárquicas**.
- Superan las desventajas de las listas, en cuanto a la búsqueda de elementos.
- Sus elementos se pueden recorrer de distintas formas, no necesariamente uno detrás de otro
- Son **muy útiles** para la búsqueda y recuperación de información

ÁRBOLES: Características

- SON ESTRUCTURAS JERÁRQUICAS
- SUS ELEMENTOS SON HOMOGÉNEOS Y SE DENOMINAN NODOS.
- CADA ELEMENTO SE RELACIONA CON CERO O MÁS NODOS LLAMADOS HIJOS
- SI EL ÁRBOL NO ESTÁ VACÍO HAY UN ÚNICO ELEMENTO LLAMADO RAÍZ QUE NO TIENE PADRE (ANTECESOR)
- TODO ELEMENTO POSEE UN ÚNICO PADRE Y ES DESCENDIENTE DE LA RAÍZ.

ARBOLES:Definiciones

- LOS NODOS SIN **HIJOS** SE LLAMAN **HOJAS**
- **CAMINO** ES LA SECUENCIA LINEAL DE NODOS, DONDE CADA UNO DE ELLOS ES EL PADRE DEL PRÓXIMO EN LA SECUENCIA.
- **RAMA** ES EL CAMINO QUE EXISTE DESDE LA RAÍZ HASTA UNA HOJA
- **SURÁRBOL** ES UN NODO CON TODOS SUS DESCENDIENTES.

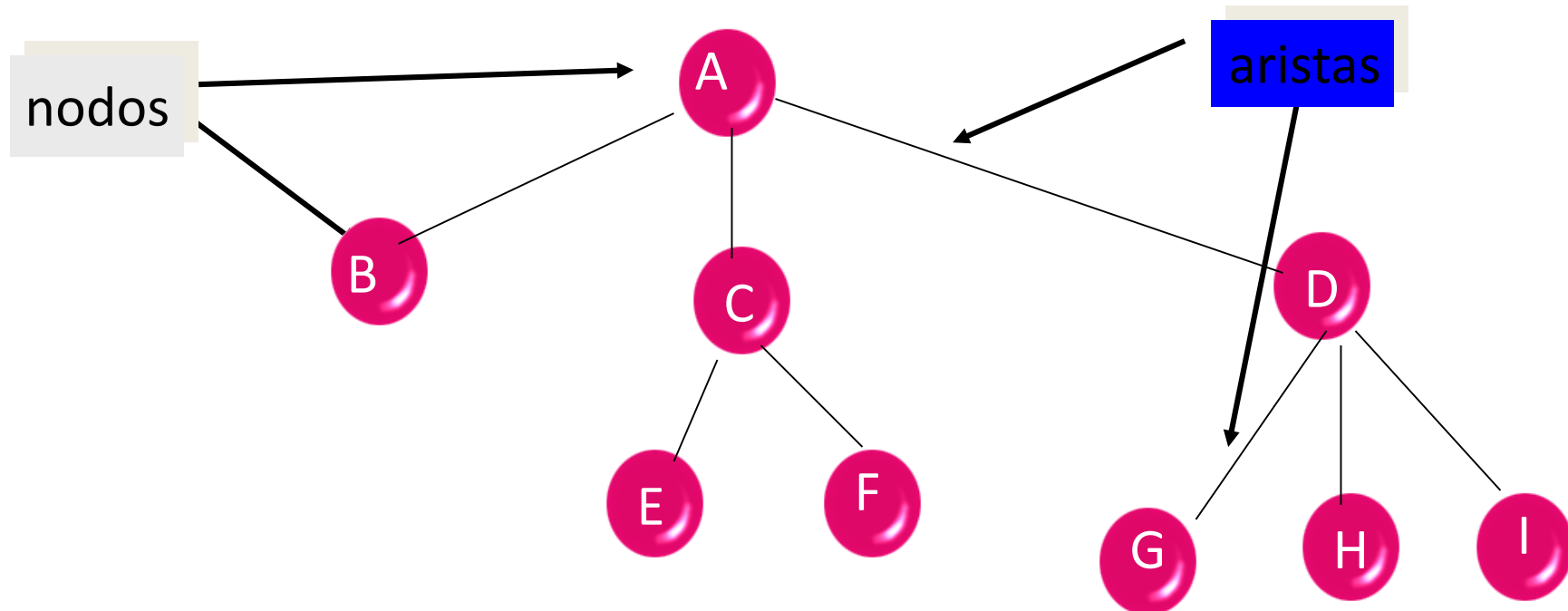
ÁRBOLES: Definiciones

- **NIVEL DE UN NODO** ES EL NÚMERO DE NODOS QUE HAY SOBRE EL ÚNICO CAMINO QUE LO UNE A LA RAÍZ.
- **LA ALTURA DEL ARBOL** ES EL MÁXIMO ENTRE LOS NIVELES DE LOS NODOS DEL ÁRBOL.
- LOS NODOS SON **HERMANOS** SI TIENE EL MISMO PADRE.
- **ANTEPASADO** DE UN NODO ES SU **PADRE** Y TODOS LOS ANTEPADOS DE ESTE
- **DESCENDIENTE** DE UN NODO, SON SUS HIJOS Y LOS DESCENDIENTES DE ESTOS

Repasemos conceptos:

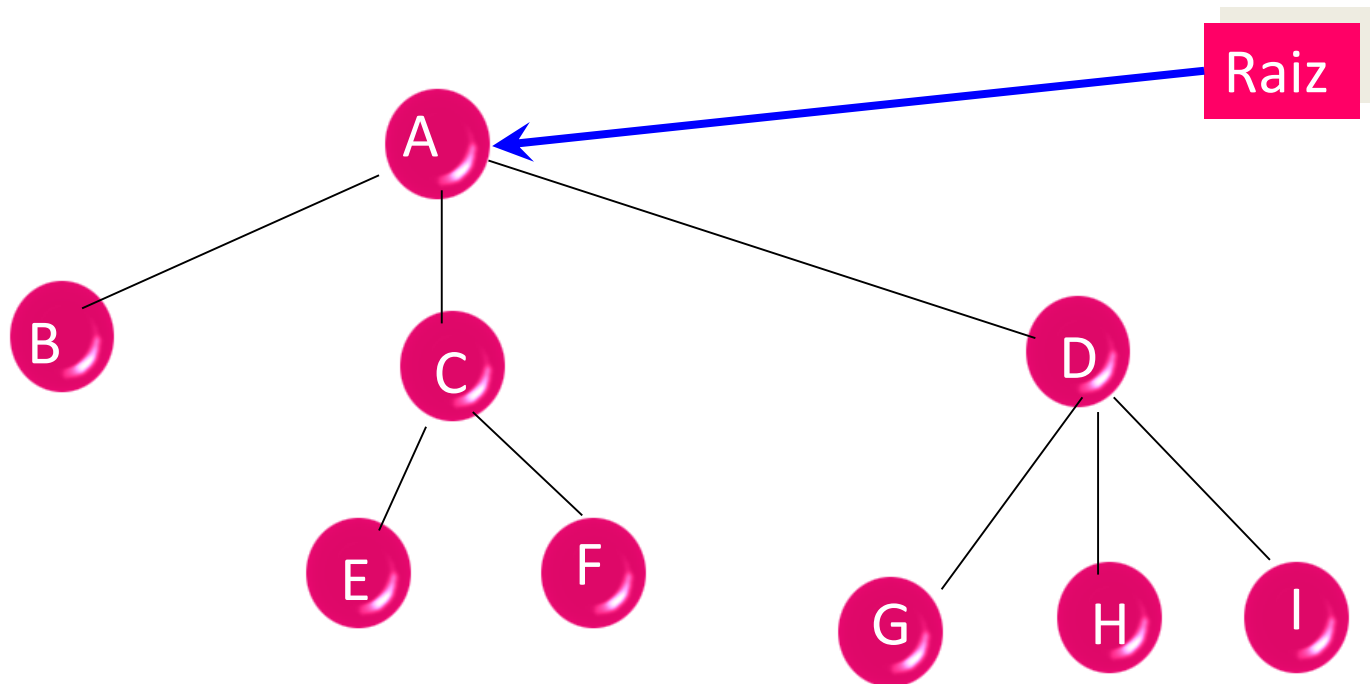
¿Qué es un árbol?

- Un árbol es una estructura de datos formada por nodos los cuales están conectados por aristas.



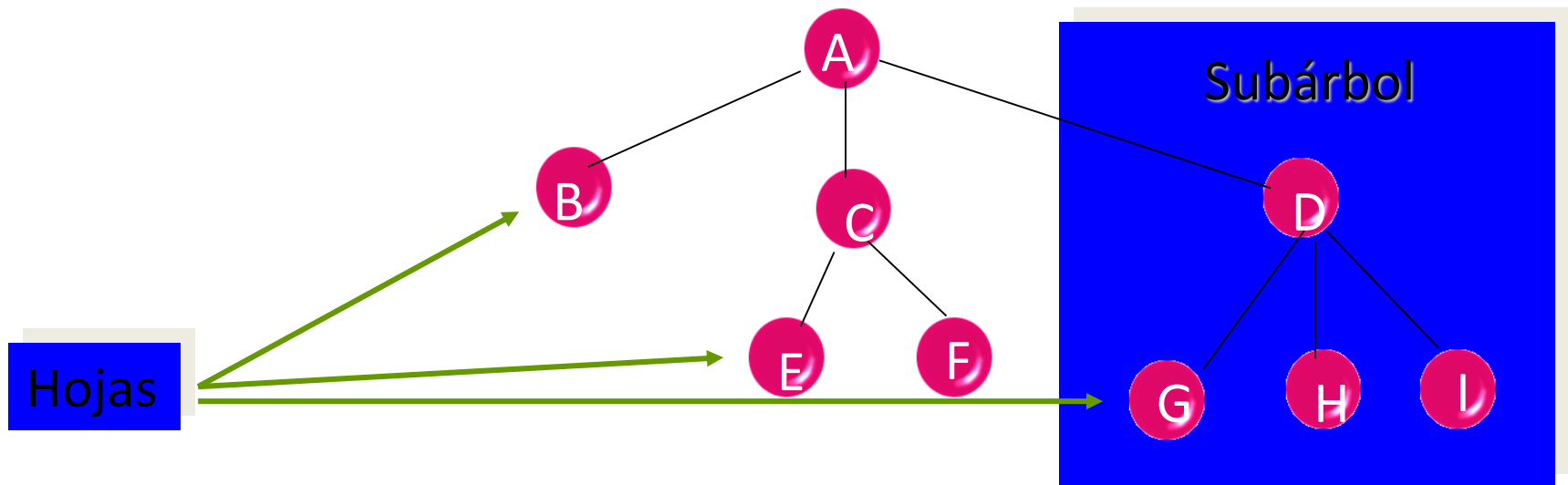
Conceptos Básicos

- Un árbol puede estar vacío; es decir no contener ningún nodo.
- Hablaremos entonces del Árbol Vacío.
- **Raíz:** es el nodo que está al tope del árbol. Un árbol solo tiene una raíz.



Conceptos Básicos

- **Hojas:** son aquellos nodos que no tienen hijos. En un árbol solo puede haber una raíz pero pueden haber muchas hojas. Ejemplo: B,E,F,G,H,I son hojas.
- **Subárbol:** Cualquier nodo se puede considerar como la raíz de un subárbol.



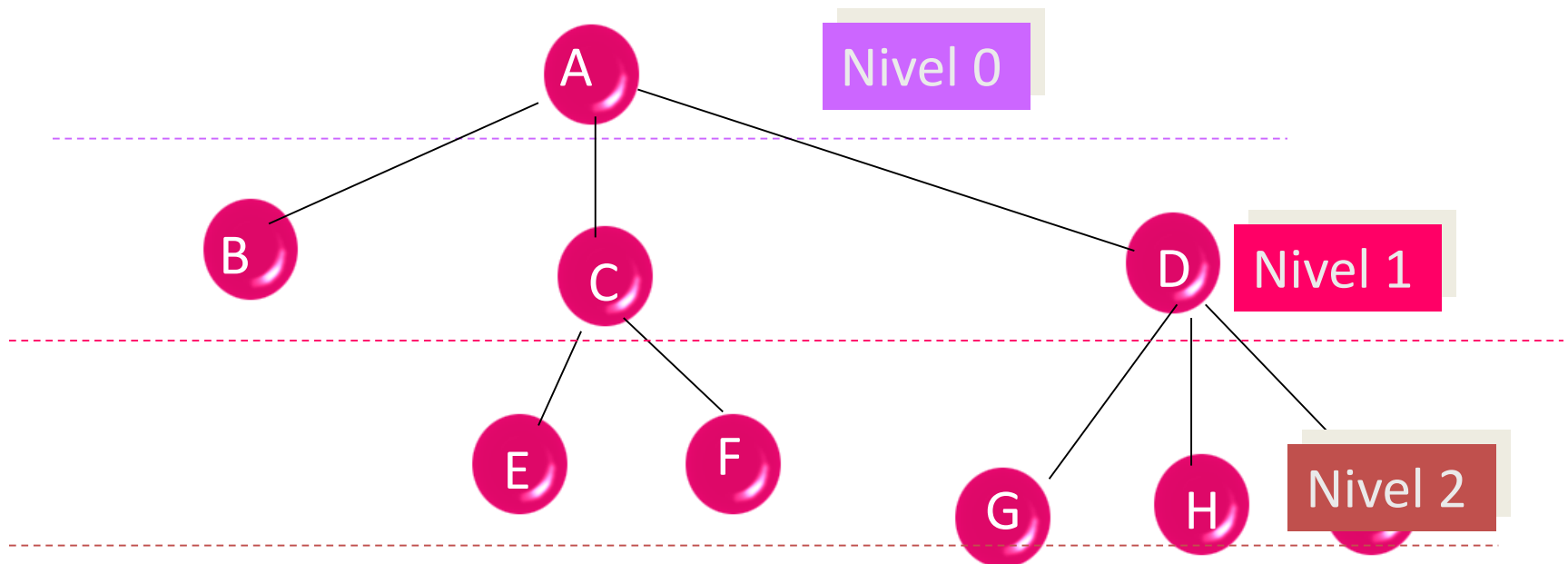
Conceptos Básicos

- **Nivel**: el nivel de un nodo es el numero de generaciones que hay desde la raíz hasta él.

El nivel de la raíz es cero.

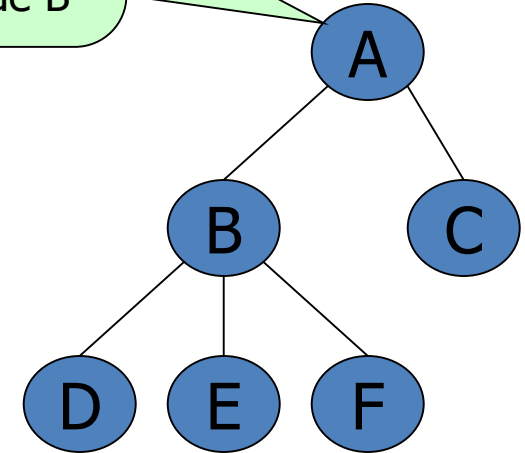
- **Profundidad o altura = Nivel**: es la longitud del camino mas largo desde la raíz hasta una hoja.

La profundidad de este árbol es 2. La raíz tiene profundidad 0.

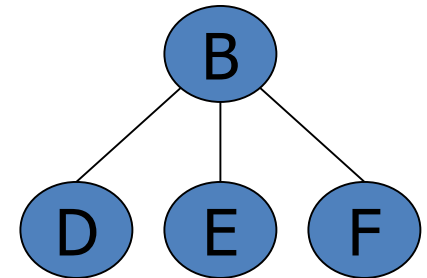


Conceptos Básicos

A es Padre
B y C hijos de A:
hermanos
B es Padre
D, E, F hijos de B



- Estructura que organiza sus elementos formando jerarquías: PADRES E HIJOS
 - Los elementos de un árbol se llaman **nodos**
 - Si un nodo **p** tiene un enlace con un nodo **m**,
 - **p** es el padre y **m** es el hijo
 - Los hijos de un mismo padre se llaman: **hermanos**
 - Todos los nodos tienen al menos un padre, menos la raíz: A
 - Si no tienen hijos se llaman **hoja**: D, E, F y C
 - Un subárbol de un árbol
 - Es cualquier nodo del árbol junto con todos sus descendientes

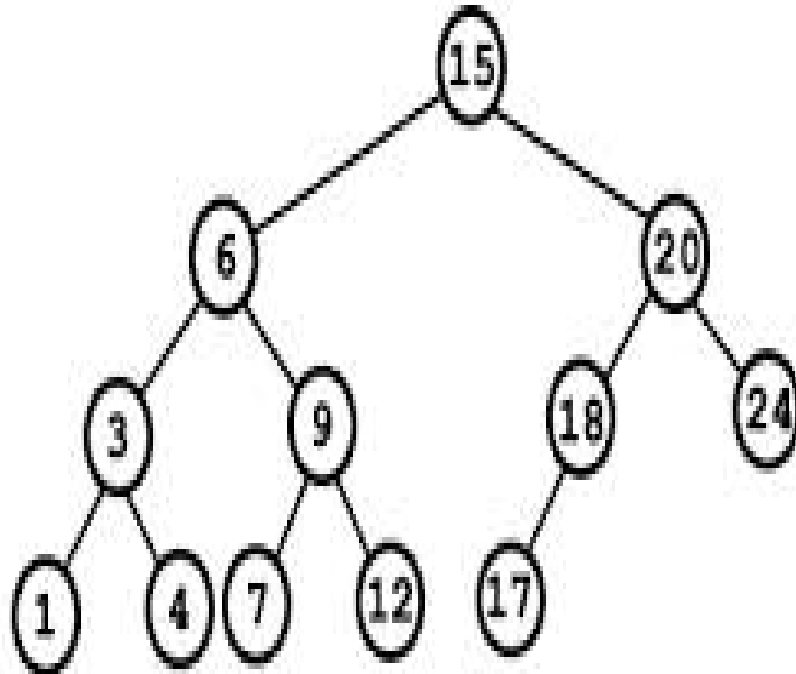


TAD ARBOL : Características

Operaciones: Dependen del tipo de árbol, pero en general tenemos:

- Inicializar el árbol
- Eliminar un árbol
- Saber si un árbol esta vacío
- Insertar un nodo en el árbol
- Borrar un nodo en el árbol
- Saber si un valor está en el árbol
- Recorrer un árbol

ARBOLES BINARIOS DE BÚSQUEDA



ARBOLES BINARIOS DE BÚSQUEDA:

Características

- Cada nodo tiene a lo sumo dos hijos (Hijo izquierdo e Hijo Derecho)
- En campo clave del Hijo Izquierdo es menor que el de su padre.
- El campo clave del Hijo Derecho es mayor o igual que el de su padre

ARBOLES BINARIOS DE BÚSQUEDA

SUBARBOL IZQUIERDO DE UN NODO: ES SU HIJO IZQUIERDO Y TODOS SUS DESCENDIENTES

SUBARBOL DERECHO DE UN NODO: ES SU HIJO DERECHO Y TODOS SUS DESCENDIENTES

NIVEL DE UN NODO: ES LA DISTANCIA (CANTIDAD DE NODOS) QUE EXISTEN EN SU CAMINO HACIA LA RAÍZ.

EL MÁXIMO NÚMERO DE NODOS DE UN NIVEL N ES 2^N

OPERACIONES BÁSICAS SOBRE ABB

- CREAR EL ÁRBOL
- INSERTAR UN NODO EN EL ÁRBOL
- BORRAR UN NODO EN EL ÁRBOL
- BUSCAR UN NODO EN EL ÁRBOL
- RECORRER EL ÁRBOL
- VACIAR EL ÁRBOL
- SABER SI UN ÁRBOL ESTA VACÍO

CREAR UN ÁRBOL

PROCEDURE (VAR RAÍZ: TPUNARBOL)

BEGIN

Raíz:= Nill

End;

PROCEDURE INSERTAR

PROCEDURE INSERTAR(var Raíz: tpunarb; valor:tinfo);

Var

Nuevonodo, P, ant: tpuntnodo;

Begin

New (Nuevonodo); Nuevonodo.^IZQ:= NIL; Nuevonodo.^DER:=
NIL;Nuevonodo.^info:= valor;

(*buscar el lugar de inserción*)

P:=RAÍZ; ANT:=NIL;

While P<> NIL do

 Begin

 ANT:=P;

 IF P.^Info> valor then P:= P.^IZQ
 else P:=P.^DER

END (*WHILE*)

IF ANT =NILL THEN RAIZ:= Nuevonodo;

 ELSE IF ANT.^INFO> valor THEN ANT.^HIZ:= Nuevonodo

 ELSE ANT.^HDER:= Nuevonodo

END

PROCEDURE INSERTAR RECURSIVO

Procedure insertarr (Var P: tpuntar; dato: telem);

Begin

If P= nil then begin

 New(P); P.^elem:= dato;

 P.^Izq:= nil; P.^Der:= nil

 end;

else If dato < P.^elem then Insertarr(P.^Izq, dato)

 Else Insertarr(P.^Der, dato)

End;

Ejercicios

Generar los árboles binarios de búsqueda que se generarán si se ingresan los datos:

- a) DBFACEG
- b) BADCGFE
- c) ABCDEFG
- d) GFEDCBA