



Universidad  
de Huelva

---

## **ALGORITMOS Y ESTRUCTURAS DE DATOS II**

Ingeniería Técnica en Informática de Gestión

Ingeniería Técnica en Informática de Sistemas

**CURSO 2002/03**

---

---

### **Práctica 3**

Diseño y construcción de un TAD basado en un  
árbol binario de búsqueda (ABB)

---

## Objetivos

---

- Definir e implementar las operaciones más habituales de los ABB

## Enunciado

---

Con idea de acelerar el proceso de lectura y escritura de los ficheros de datos, la empresa ha decidido utilizar un árbol binario de búsqueda como estructura intermedia para gestionar las fichas de los empleados antes de crear los correspondientes ficheros de los últimos puestos ocupados por los empleados.

## Se pide

---

1. Especificar e implementar el TAD *TABBF* en un módulo (unidad) independiente. El fichero se guardará con el nombre **tabbf.pas**.

2. Un pequeño programa usuario, llamado **gestionF.pas**, que lea el fichero **datos.dat** y cree dos ficheros (**ABBP.dat** y **ABBS.dat**) a partir de los árboles que contienen los datos más recientes de cada empleado, las cuales se construyen mediante la operación de *inserción* del TAD *TABBF*. Debido a que puede haber errores en los datos del fichero, cada ficha leída deberá ser validada y creada mediante la operación *crearFicha* del TAD *TFicha*.

- Una vez creados los ficheros de salida (**ABBP.dat** y **ABBS.dat**), insertar las siguientes fichas:

F1 (29000000,100,254564), F2 (37000000,200,234543), F3 (41999999,300,284321)

y generar un fichero **resABB.txt**, que será un fichero de texto (Text) con la siguiente información:

- Tiempo que tardó la inserción de cada ficha (F1, F2 y F3)
- Tiempo medio de inserción de estas tres mismas fichas.

- Liberar los ABB creados

3. Utilizando el TAD *ListaF* de la práctica 2, procesar el fichero **datos.dat**, generando las correspondientes listas LP y LS. A continuación insertar las fichas F1, F2 y F3, y generar un fichero **resLista.txt**, que será un fichero de texto (Text) con la misma información que el **resABB.txt**.

## Descripción del TAD

---

### TAD *TABBF*:

El TAD *TABBF* está formado por el tipo **ABBF**, basado en un árbol binario de búsqueda mediante punteros. Las características y operaciones del TAD son las siguientes:

- El tipo **ABBF** se implementará mediante un registro formado por 2 campos: un puntero apuntando a la raíz del árbol y un natural que indica el número de elementos que tiene el árbol. La **clave** de búsqueda es el DNI de la ficha.

**procedimiento** crearABBF(**var** A: ABBF);

*crea un ABBF vacío*

**función** esVacio(A: ABBF): booleano;

*devuelve verdad si A está vacío y falso en caso contrario*

**función** longitud(A: ABBF): entero;

*devuelve el número de fichas que hay en el ABBF A*

**función** esta(vdni: entero largo, A: ABBF): booleano;

*devuelve verdad si existe un ficha con un determinado dni en A y falso en caso contrario*

**función** inserta(**var** AP,AS: ABBF; F: Ficha): natural;

*inserta la ficha F en el ABBF AP o en el ABBF AS según el siguiente criterio:*

*(a) si el DNI de F no se encuentra en AP, se inserta F en AP*

*(b) si el DNI de F se encuentra en AP y su código es mayor que el encontrado en AP, se sustituye la ficha de AP por la nueva ficha F. La ficha sustituida de AP se intenta insertar en AS, siguiendo este mismo criterio.*

*(c) si el DNI de F se encuentra en AP pero su código es menor que el encontrado en AP, entonces se intenta insertar F en AS. Si también se encuentra en AS y el código sigue siendo menor que el encontrado en AS, la ficha no se inserta en ningún ABBF.*

*las inserciones se realizarán, según la clave, siguiendo el criterio de búsqueda*

*la función devuelve "1" si la ficha se ha insertado en AP, "2" si se inserta en AS, y "0" si no se inserta en ningún ABBF*

**función** borrarDNI(vdni: entero largo; **var** A: ABBF): booleano;

*elimina del ABBF A una ficha con un determinado dni. Devuelve verdad si se ha eliminado y falso en caso contrario*

**función** borrarTlf(tlf: entero largo; **var** A: ABBF): entero;

*elimina del ABBF todas las fichas con un determinado teléfono. Devuelve el número de fichas que han sido eliminadas*

**acción** creaFichero(A: ABBF; nf: string);

*crea un fichero de fichas con el contenido del ABBF A*

Además, habrá que implementar las operaciones: igualdad, asignación y liberación

- **función** iguales(A1,A2: ABBF): booleano;
- **procedimiento** asigna(**var** A1: ABBF; A2: ABBF);
- **procedimiento** libera(**var** A: ABBF);

## Tiempo estimado de realización

---

3 clases (se entregará en la semana del 6 al 10 de enero en la clase de prácticas)