

ADP - PRÁCTICA 12

LISTAS MULTIENTLACE

1. Escribir un programa que lea una secuencia de números positivos hasta que se ingresa el 0, el cual no debe procesarse. Los números leídos deben ser almacenados en orden ascendente en una lista doblemente enlazada. Los números se leen en orden, por ejemplo: 1, 2, 7, 9, 28, 37, 0. **Finalizada la lectura**, imprimir en pantalla los números en forma ascendente (de menor a mayor) y descendente (de mayor a menor).
2. Modificar el programa del inciso 1) de modo que haga exactamente lo mismo pero los números ahora se leen sin orden, por ejemplo: 28, 2, 9, 37, 51, 1, 7, 0.
3. Escribir un programa que lea la información de personas, de cada persona se lee: DNI, Apellido y Nombre. La lectura finaliza cuando llega la persona con DNI 0, la cual no debe procesarse. Almacene a las personas en una lista doblemente enlazada ordenada por DNI de forma ascendente. Las personas se leen desordenadas por DNI.
 - a. **Una vez finalizada la carga:**
 - i. Escriba un módulo para actualizar la información de una persona. Para ello el módulo debe leer un DNI, buscar dicho DNI en la lista y si existe actualizar el Apellido y Nombre, leyendo nuevos datos desde teclado.
 - ii. Escriba un módulo que imprima en pantalla la lista de forma ascendente y en forma descendente.
4. Una empresa dispone de una lista de empleados, de un empleado se conoce: Número de Empleado, Apellido, Nombre, Profesión, Código de Departamento al que pertenece, Sueldo básico y años antigüedad. Dicha lista está ordenada por Número de Empleado (orden principal); pero también puede accederse en forma ordenada por código de departamento (orden secundario). **Se pide:**
 - a. Realizar un módulo que reciba un nuevo empleado y lo incorpore a la lista de empleados de la empresa (manteniendo ambos órdenes).
 - b. Implementar un módulo que elimine de la lista todos los empleados que pertenezcan al departamento 4 o al departamento 10 (estos departamentos pueden no existir).
 - c. Implementar un módulo que reciba un departamento como parámetro y retorne un empleado de dicho departamento que sea "Contador". Si existe más de uno, debe retornar el de mayor antigüedad (tener en cuenta que puede no existir ningún empleado con dichas características e incluso puede no existir el departamento).
5. Un colegio dispone de una estructura donde almacena la información de las donaciones que ha recibido. De cada donación se conoce: código de donación, país que la donó, año en el cual se efectuó la donación (1992..2010), monto de la donación y nombre de la entidad que efectuó la donación. Esta información se encuentra accesible por dos órdenes diferentes: un orden es el código de donación de manera ascendente y el otro orden es el país que la donó de manera también ascendente. **Se pide:**
 - a. Realizar un módulo para informar los códigos de las donaciones efectuadas por "España". Tener en cuenta que puede haber 0,1 o más donaciones efectuadas por España.
 - b. Realizar un módulo que calcule e Informe el año en el cual se realizaron más donaciones.
 - c. Realizar un módulo para informar el nombre de la entidad que efectuó la donación con código 1092. Tener en cuenta que dicho código puede no existir.

Nota: se debe implementar el programa principal donde se realicen los llamados a los módulos.

Ejercicio adicional

6. Realice un programa que lea números hasta que llegue el -1. Por cada número el programa debe indicar si es capicúa o no. Para resolver este problema utilice estructuras dinámicas.