# Parcial programación en C

## Enunciado

La aseguradora de autos Seguros SA tuvo un inconveniente de corte de luz y las transacciones de las cuotas que deben abonar los usuarios quedaron desactualizadas.

Cuando sucede este tipo de problemas el sistema genera un archivo parche.dat donde se guardan todas las cuotas a pagar por los diferentes clientes.

**Archivo parche.dat**

|  |  |  |
| --- | --- | --- |
| **código** | (10 caracteres) | Código del vehículo (9 Patente (XXX 999) |
| **cuota** | Numero de cuota entero | 1-10 |
| **monto** | Valor real monto de la cuota | - |

También poseen un archivo mensual llamado **autos.dat** el cual posee la siguiente estructura.

**Archivo autos.dat**

|  |  |  |
| --- | --- | --- |
| **código** | (10 caracteres) | Código o patente del vehículo |
| **modelo** | (45 caracteres) | Marca y modelo del vehículo |
| **importe** | (Real) | Total a abonar por el vehículo. |

A parte de estos dos archivos la empresa cuenta con una estructura de datos cargada con un índice al archivo *autos.dat* con el **código de auto** y ***el número de registro*** donde se encuentran los datos del vehículo.

Con el paso del tiempo algunos clientes se dieron de baja del sistema por lo tanto la empresa necesita saber quiénes se dieron de baja para no enviar la factura con el importe a cobrar que figura en el archivo *parche.dat* para establecer esto el número de registro del índice figura con un valor -1. En este caso (NO DEBE ACTUALIZARSE EL ARCHIVO) *autos.dat.* Debe por una cuestión de seguridad borrar del índice dicho elemento.

El índice es un árbol binario de búsqueda.

* Se solicita realizar la función **buscar clave** en el índice y **borrar elementos** con el número de registro -1.
* Actualizar el Archivo autos.dat con la información del archivo parche.dat.
* Crear una lista con las cuotas que NO pudieron cobrarse debido a que los clientes se dieron de baja.

**Funciones Pedidas:**

**actualizarAutomores** (Para actualizar el archivo autos.dat).

**borrarElemento** (Para borrar el nodo elegido del árbol binario)

**ponerEnLista** (Para insertar un elemento t\_cueta en la lista doblemente enlazada).

**buscarClave** (Busca por clave en el índice y devuelve la información completa y un entero informando si lo encontró o no (1 / 0)).

**NOTA:**

**Cualquier función adicional que el alumno necesite realizar puede hacerlo las mismas no serán evaluadas ni tenidas en cuenta. Esto es solo por comodidad del alumno en el desarrollo del examen.**

**NO SE PUEDEN UTILIZAR VARIABLES GLOBALES.**

La información de las funciones utilizadas y las que tienen que generar se encuentra en el archivo ./include/header.h del proyecto.