

### Programa C

Una empresa que se dedica al armado y reparación de PCs, necesita actualizar sus existencias de componentes. Para ello, le encargó desarrollar un sistema que realice dicha actualización.

La estructura del archivo binario de **Componentes (Componentes.dat)** es la sig:

codigo: Texto(20), descripcion: Texto(50), stock: Entero, precioUnitario: Real

Los componentes se actualizan en base a un archivo binario donde se registran los armados y reparaciones efectuadas, donde se detallan los códigos, las cantidades y el precio unitario de los componentes utilizados en cada operación.

La estructura del archivo de **Armados y Reparaciones (ArmadosYReparaciones.dat)** es la sig:

nroOp: Entero, codigo: Texto(20), cantidad: Entero, precioUnitario: Real

El sistema actualizador correrá todas las noches, luego del cierre de la empresa. **Se encuentra parcialmente desarrollado**, Ud. sólo deberá desarrollar las funciones que se le indiquen, pudiendo utilizar éstas cualquier función que también deberá desarrollar.

Al archivo de **Componentes** debe accederse en forma **directa** a través de un **Índice implementado en un árbol binario de búsqueda**. El índice existe como un archivo binario **que tiene el mismo nombre que el archivo de Componentes, pero su extensión es idx**. Ud. deberá cargar este archivo de índices en un **árbol**. Tenga en cuenta que **el árbol debe estar balanceado**, por lo que una carga leyendo el archivo **en forma secuencial NO es posible**.

El tamaño que se espera que tenga a futuro el archivo de **Componentes**, **hace inviable su carga en memoria. Sus registros NO pueden accederse más de una vez**.

La persona encargada de este proyecto antes que Ud. desarrolló una función que carga todo el contenido de los armados/reparaciones en un **TDA Lista con implementación en lista doblemente enlazada**. Ud. deberá utilizar esa función. El problema es que esta función carga un elemento en la lista por cada registro del archivo, por lo que existe más de un elemento con el mismo código de componente. Si se intenta actualizar los componentes con los elementos de la lista como están, se incurrirá en más de un acceso por registro. Por lo que **deberá eliminar los duplicados, fusionando sus informaciones**.

**Debe entregar solamente el archivo main.c. NO modifique ni entregue otro archivo.**



## Programa C++

Desarrollar una clase Punto, con coordenadas x e y de tipo double.

Desarrollar una clase Recta, conformada por 2 puntos de la clase anterior.

Debe resolver las clases de manera que compile y ejecute correctamente la función main provista.

### Ecuación de la distancia de un punto a una recta:

Dada una recta que pasa por 2 puntos  $P_1 = (x_1, y_1)$  y  $P_2 = (x_2, y_2)$ , la distancia del punto  $(x_0, y_0)$  a la recta está dada por:

$$\frac{|(y_2 - y_1)x_0 - (x_2 - x_1)y_0 + x_2y_1 - y_2x_1|}{\sqrt{(y_2 - y_1)^2 + (x_2 - x_1)^2}}$$

### Ecuación de la intersección de 2 rectas:

Dada la recta  $R1 = [(x_1, y_1), (x_2, y_2)]$  y la recta  $R2 = [(x_3, y_3), (x_4, y_4)]$ , la intersección, el punto  $P = (x, y)$  está dada por:

$$(x, y) = \frac{(x_1y_2 - y_1x_2)(x_3 - x_4) - (x_1 - x_2)(x_3y_4 - y_3x_4)}{(x_1 - x_2)(y_3 - y_4) - (y_1 - y_2)(x_3 - x_4)}, \frac{(x_1y_2 - y_1x_2)(y_3 - y_4) - (y_1 - y_2)(x_3y_4 - y_3x_4)}{(x_1 - x_2)(y_3 - y_4) - (y_1 - y_2)(x_3 - x_4)}$$

**Aquí debe entregar el proyecto completo, sin los archivos generados por la compilación.**

Los programas deben compilar y ejecutar correctamente para obtener 4 o más. Debe eliminar los duplicados de la lista y actualizar correctamente el archivo Componentes.dat, mediante el índice cargado en el árbol, junto con el punto de C++ correcto para obtener de 4 a 6. Sumado a lo anterior, deberá cargar correctamente el árbol para obtener 7 o más.

EVALUACIÓN TOMADA EN MODALIDAD NO PRESENCIAL

**NOTA** No utilice ni suponga variables globales.  
Ante cualquier duda, consulte a los docentes.