

## Trabajo Práctico Nivel 6

*Tipo de Dato Arreglo de Registros: Concepto y Operaciones*

*Fecha de Inicio 08/10/2025 - Fecha de Fin 20/10/2025*

**Ejercicio 1:** Crear un programa que permita el ingreso de una lista de N países, de cada País se guarda la siguiente información: "Código del País, Nombre del país, Capital del país y Continente al que pertenece."

Se debe realizar la carga de la lista de países y posteriormente ordenar la lista por Código del país, luego utilizando un menú poder realizar las siguientes operaciones:

- Añadir un nuevo País preservando el orden.
- Dado un Código de un país, modificar la Capital que se tiene registrada.
- Dado un Código de un país, eliminar dicho registro.
- Dada una Capital, mostrar el País al que pertenece.

**Ejercicio 2:** Un banco mantiene la información de sus clientes en un array de registros (Lista de clientes). De cada Cliente se almacena la siguiente información: "Número de cuenta, Apellido y Nombre del cliente,

Domicilio del cliente, Saldo de la cuenta. "

Se solicita escribir un programa que permita realizar la carga de las cuentas de N Clientes de un banco y luego mediante un menú realizar las siguientes operaciones:

- Permitir actualizar el domicilio de un cliente, dado su número de cuenta
- Realizar un depósito (es decir, dado un monto a depositar y una cuenta, actualizar el saldo).
- Realizar un retiro (es decir, dado un monto a retirar y una cuenta, actualizar el saldo si es que el saldo es mayor o igual al importe a retirar).
- Realizar una transferencia entre cuentas (es decir, dado un monto a transferir, una cuenta origen y una cuenta destino, actualizar el saldo destino si es que el saldo origen es mayor o igual al importe a transferir).

**Ejercicio 3:** Realice los cambios apropiados en el Ejercicio 2, para impedir la carga de un número de cuenta ya existente en la lista y que la lista de clientes se ordene por el número de cuenta a medida que se ingresan los clientes.

**Ejercicio 4:** Una academia de danzas tiene una lista de alumnos, registrando de cada alumno los siguientes datos: "DNI del alumno, Apellido y Nombre del alumno, Código de curso de verano ([1] folclore, [2] español, [3] contemporáneo)".

Crear un programa que permita cargar la lista de alumnos, inicializando el campo Código de curso en 0. Luego ordene la lista por DNI del alumno utilizando el método de la Burbuja. Posteriormente, a través de un menú realizar las siguientes tareas:

- Agregar un alumno a la lista.
- Dado el DNI de un alumno inscribirlo a uno de los 3 cursos que se dictan, teniendo en cuenta que el cupo es de 10 personas para cada curso. Un alumno sólo puede estar inscripto en un curso a la vez.
- Permitir dar de baja a un alumno de un curso ingresando su DNI. (Cambiar Código a 0)
- Eliminar de la lista todos los alumnos que no se encuentren inscriptos en algún curso.
- Dado un Código de curso, mostrar la lista de inscriptos.

**Ejercicio 5:** Se necesita crear un programa que maneje una Agenda telefónica de amigos, que almacena los siguientes datos de cada persona: "Apellidos, Nombres, Dirección, Celular, Fecha de nacimiento". Realizar un programa que permita cargar la lista de amigos y que la misma se ordene alfabéticamente por Apellidos y Nombres a medida que cargan los registros. Utilizando un menú permita realizar las siguientes operaciones:

- Añadir un nuevo contacto, considerando que cada contacto es único. (Mantener ordenada la lista por apellido y nombres).
- Borrar un contacto. (Mantener ordenada la lista por apellido y nombres).
- Dado el apellido y el nombre, modificar su dirección o su teléfono.

- d)
- e) Dado un apellido y el nombre, mostrar sus datos.
- f) Mostrar la agenda telefónica completa.
- g) Dado un apellido, mostrar todos los datos de las personas con ese apellido.
- h) Dado un mes, mostrar todas las personas que cumplen años en ese mes.

*Nota: Diseñar un módulo agrega un registro nuevo a la lista existente, que se pueda reutilizar en la operación carga la lista y en la operación agrega nuevo registro.*

**Para los ejercicios posteriores, las operaciones de Carga una lista y Alta de un nuevo elemento, deben utilizar el módulo Agrega Nuevo Registro**

**Ejercicio 6:** Dada una lista de Repuestos de autos con los siguientes datos: “Código del repuesto, Nombre del repuesto, Nombre del Proveedor, Cantidad en Stock, Precio del repuesto”. Se pide crear un programa que permita cargar la lista de N repuestos y ordenarla a medida que se realiza la carga por Código del repuesto. Además, realizar un menú que permita:

- a) Agregar un nuevo repuesto.
- b) Dado un Código de repuesto, eliminar de la lista sus datos.
- c) Dada una lista precios nuevos con la siguiente información Código del repuesto y Nuevo precio. Actualizar los precios.
- d) Dado un Proveedor generar y mostrar una nueva lista con los repuestos provisto por él. e) Mostrar la lista completa.

*Nota: El ingreso y el alta no deben permitir Código de repuestos repetidos.*

**Ejercicio 7:** modificar el programa anterior de manera que permita realizar la venta de una cantidad no determinada de repuestos.

#### **Ejercicio 8:**

Crear una estructura de datos de array de registros para almacenar los datos de los vehículos del estacionamiento. Los datos a almacenar son: “Nro. de Patente, Tipo de Vehículo (auto, camioneta, moto), Hora y Minutos de Ingreso, Hora y Minutos de Egreso, Importe Pagado” Crear 2 listas de vehículos:

- La primer lista llamada INGRESOS, es para almacenar los vehículos que ingresan, en esta lista la hora y minutos de egreso y el importe pagado no se debe cargar.
- La segunda lista llamada EGRESOS, es para almacenar los datos de los vehículos que se retiraron, en donde se modificará la hora y minutos de egreso y el importe pagado.

Se pide realizar un programa que permita cargar la primer lista INGRESOS de N vehículos y luego a través de un menú permita las siguientes operaciones:

- a) Agregar un nuevo vehículo a la lista INGRESOS.
- b) Retirar un vehículo (Para esta acción se solicita el Nro. de Patente del vehículo, se buscará el vehículo en INGRESOS y se deben copiar los datos del vehículo a la lista EGRESOS, para cargar la información de salida. Posteriormente se debe eliminar el vehículo de la lista INGRESOS.
- c) Eliminar de la lista EGRESOS todos aquellos vehículos de un Tipo determinado por el usuario.

**Ejercicio 9:** Retomar el Ejercicio 2 (Un banco mantiene la información de sus clientes en un array de registros...) agregando las siguientes operaciones:

Que la lista luego de ser ingresada se ordene por Número de cuenta y al menú de opciones las operaciones de:

- Mostrar la lista de clientes.
- Generar y mostrar una nueva lista con los clientes cuyo Saldo sea menor o igual a cero.
- Mostrar cuántos clientes tiene un saldo mayor a un valor ingresado por teclado

*Nota: Realizar los cambios necesarios para que se adecue el programa previamente realizado, a los nuevos requerimientos.*

**Ejercicio 10:** Una entrenadora de ciclismo lleva un registro con información de N entrenamientos realizados por un ciclista. Para cada entrenamiento, se registra la Velocidad del ciclista, el Tiempo transcurrido (en segundos), La fecha de realización (Dia/Mes/Año), y un Número de identificación para cada sesión. Se pide crear un programa que permita cargar la lista de N entrenamientos, además realizar un menú que permita:

- a) Agregar un nuevo entrenamiento.
- b) Eliminar un entrenamiento: Dado un Número de identificación, eliminar de la lista la información correspondiente a ese entrenamiento.
- c) Dado un mes y un año eliminar todas las actividades realizadas.
- d) Dado un día y un número de identificación. Mostrar su actividad mostrar las actividades correspondientes.
- e) Filtrar entrenamientos por rango de tiempo: Dado un tiempo T, generar y mostrar una nueva lista con los entrenamientos cuyo tiempo transcurrido se encuentre en el intervalo  $[T-30 \text{ segundos y } T+30 \text{ segundos}]$ .
- f) Mostrar las listas:
  1. Dado un rango de dos Fecha 1 y Fecha 2, mostrar todas las actividades realizadas en ese rango.
  2. Mostrar toda la información registrada de los entrenamientos realizados.