

## PRÁCTICA 1 LISTAS SIMPLE-LISTAS DOBLES Y CIRCULARES

*Nota:* Para todos los ejercicios definir las estructuras utilizadas y modularizar como corresponda. Realizar los ejercicios de manera eficiente.

1.- Escriba un programa que lea una secuencia de números enteros hasta que llegue el número 999, el cual no debe procesarse, y genere una estructura lista simple almacenando cada número. Una vez generado la estructura lista, informe:

- Dos máximos entre los números leídos.
- La cantidad de números pares y la cantidad de números impares.
- La cantidad de dígitos que tiene el número que tiene más cifras.

2.- Se **dispone** de una lista, con la información de los clientes que se acercan a pagar impuestos. De cada cliente se conoce: DNI, Apellido y Nombre, Código de impuesto a pagar (de A a K) y el monto a pagar. Se pide:

- Simular la atención de los clientes hasta que se haya recaudado al menos \$500.000 o se hayan procesado todos los clientes de la lista.

Al finalizar la atención de los clientes, se desea informar:

- El monto total recaudado por cada código de impuesto.
- La cantidad de clientes que quedaron sin atender.

3.- Un congreso de nutrición **dispone** de una lista con la información de sus disertantes. De cada disertante, se conoce: Apellido y Nombre, Sexo, Título de la charla que dará y País de procedencia.

Se desea procesar la información para:

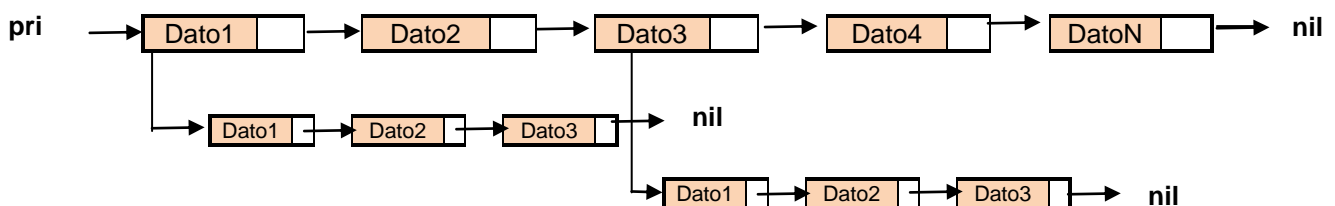
- Informar los nombres de las participantes "Mujeres", con país de procedencia "Argentina". Además, la cantidad total de participantes que cumplen con tal condición.
- Calcular e informar el porcentaje de participantes que provienen de países que no son "Argentina".
- Generar una nueva lista ordenada por País de procedencia.
- ¿Se podría haber resuelto el inciso c, sin haber creado una nueva lista?. Explique, no implemente.

4.- Tres sucursales de una farmacia deben crear una lista de las ventas realizadas. De cada venta se conoce código del producto y cantidad vendida. Las ventas deben quedar ordenadas por código de producto.

Un producto puede ser vendido 0, 1 ó mas veces en la misma sucursal y además puede ser vendido en las dos sucursales. Se pide realizar un programa que:

- Generar las tres listas (una para cada sucursal) de manera ordenada, cada lista debe ser creada y cargada con ventas hasta que llegue el producto con código 0. Tener en cuenta que la lectura de las ventas desde teclado no se realiza de manera ordenada y un código de producto puede aparecer más de una vez y en cualquier orden.
- Luego de que todas listas estén creadas y cargadas generar una nueva lista que totalice la cantidad de ventas de cada producto.

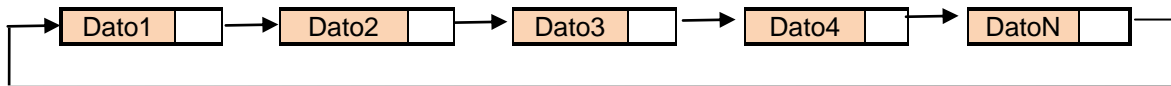
5. -Defina con un type la estructura de una lista de listas.



6.- Se dispone de una lista con los datos personales (nombre y apellido, dni, código de vendedor) y las ventas realizadas (día, producto vendido, monto y tipo de venta -contado o crédito-) por vendedores de productos de belleza.

- a. Generar una lista con los vendedores sin ventas realizadas (ordenada por código de vendedor).
- b. Imprimir para cada vendedor el total facturado por el mismo (distinguiendo las ventas al contado de las realizadas a crédito).

7.- Defina el type para la estructura de una lista circular



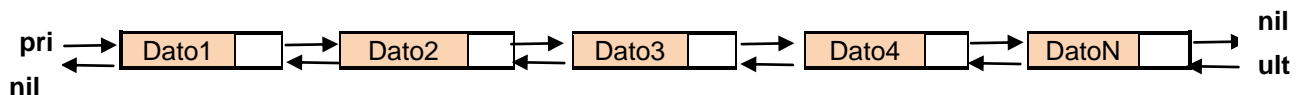
8.- Implemente las operaciones de creación, inserción y borrado de un elemento en este tipo de lista. Considere una lista circular donde el último elemento se “engancha” físicamente con el primero.

9.- Se quiere implementar el juego “ronda de suerte”. El mismo consiste en un grupo de jugadores colocados en ronda, donde se van eliminando de a uno. Para eliminar un jugador se tira un dado y el jugador que se encuentra ubicado en la posición que indique el dado es eliminado. El juego concluye cuando queda un único jugador el cual es el ganador.

Desarrolle un programa que simule el juego en forma completa. Utilice los módulos y la estructura de datos que considere adecuados para el juego.

**Nota:** La primera vez se empieza a contar desde el primer jugador. Las veces sucesivas se cuenta a partir del jugador siguiente al jugador eliminado. Suponga que existe una función tirar dado que devuelve un número entre 1 y 6.

10.-Defina el type de la estructura de una lista doblemente enlazada.

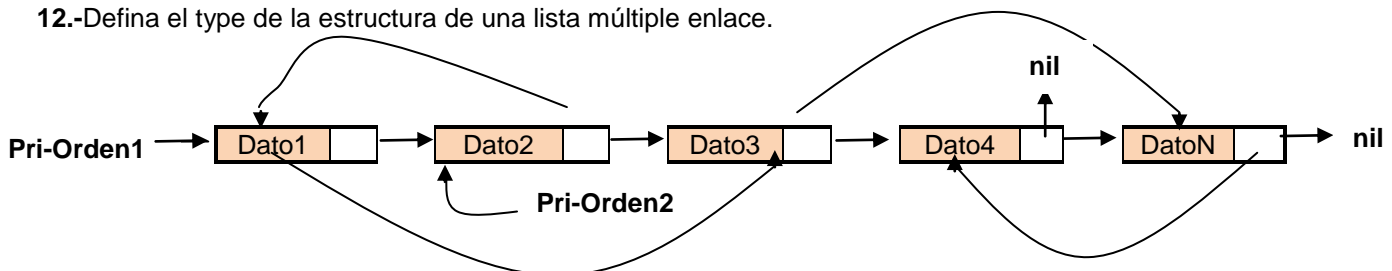


11.- a) Generar una lista ordenada por código de producto doblemente enlazada con los datos de los productos comprados por un comedor escolar, de dos maneras:

- i. suponiendo que los datos se leen ordenados por código de producto
- ii. suponiendo que los datos se leen sin orden.

b) Dada la lista generada obtener dos listados: uno en forma ascendente y otro en forma descendente.

12.-Defina el type de la estructura de una lista múltiple enlace.



13.- Una empresa mantiene una lista de empleados, de un empleado se conoce su Nro. de Empleado, apellido y nombre, profesión, código de departamento al que pertenece, sueldo básico y antigüedad.

Dicha lista está ordenada por Nro. de Empleado (orden principal); pero también puede accederse en forma ordenada por código de departamento (orden secundario). Suponiendo que la lista ya existe 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).

**14.-**Se desea desarrollar un programa que maneje la información de los alumnos de la Facultad de Informática de la UNLP. De cada alumno regular se conoce: N° de legajo, DNI, Nombre y Apellido y una lista con todas sus materias aprobadas con nota final (1..10). La información está ordenada por dos criterios: N° de legajo y Apellido. Además, la Facultad **dispone** de una estructura con el detalle de las 150 materias que dicta en sus carreras. De cada materia se conoce el código, el nombre y el docente titular.

Se pretende contar con un software que permita cargar toda la información de todos los alumnos regulares de la Facultad, contemplando las siguientes operaciones:

- a. Insertar los datos de un nuevo alumno manteniendo el orden en ambos criterios.
- b. Eliminar los datos de un determinado alumno con N° de legajo leído de teclado.
- c. Agregar una nueva materia a un determinado alumno.
- d. Listar en pantalla el Legajo y Nombre y Apellido de aquellos alumnos que aprobaron la materia "Programación II".