

## Ejercicios Adicionales

### Imperativo

**Ejercicio 1.** Implementar un programa que procese la información de los participantes a un concurso de preguntas y respuestas. De cada participante se lee el código de participante, código de ciudad de origen y edad. El ingreso de los participantes finaliza cuando se lee el código -1.

Implementar un programa que:

- a) Genere un ABB a partir de la información leída, ordenado por código de participante.
- b) Contenga un módulo que reciba el árbol generado en a) y un código de ciudad y retorne una lista con los participantes de esa ciudad.
- c) Invoque al módulo de b) y luego muestre el contenido de la lista resultante utilizando un módulo recursivo.
- d) Muestre la cantidad de participantes cuyos códigos están comprendidos entre dos valores determinados. Para ello implementar un módulo que reciba el árbol generado en a) y dos valores y devuelva la cantidad pedida.
- e) Informe la edad promedio de los participantes del concurso.

**Ejercicio 2.** Una familia de 4 integrantes llevan registro de sus gastos (tipo de gasto y monto gastado). Se pide:

- a) Generar una estructura que almacene los gastos de los integrantes de la familia, agrupados según el integrante. Por cada integrante, se leen gastos hasta tipo de gasto 'zzz'. Los gastos de cada integrante deben ordenarse por tipo de gasto.
- b) Imprimir la estructura generada en a.
- c) A partir de la estructura generada en a, realice un merge para obtener los montos totales gastados por cada tipo de gasto. Genere un ABB con monto total y tipo de gasto, ordenado por monto total.
- d) A partir del ABB, busque e imprima el tipo de gasto con mayor monto total.

**Ejercicio 3.** Una perfumería dispone en papel de la información de las ventas de sus productos. De cada venta conoce: código de producto, tipo de producto, fecha, cantidad de unidades vendida y monto.

Se pide realizar un programa que:

- a) A partir de la información de ventas ingresadas por teclado, genere un ABB que almacene para cada código de producto: su tipo, la cantidad total de unidades vendidas y el monto total vendido (entre todas las ventas realizadas). El ABB debe ordenarse por código de producto. El ingreso de información finaliza cuando se lee cantidad vendida 0.
- b) Luego de generado el ABB, informar (implementar un módulo para cada inciso):
  - i - El tipo de los productos cuyo código está comprendido entre los valores 1500 y 2200.
  - ii - La cantidad de productos con monto total vendido comprendido entre 10000 y 50000.

### Objetos

**Ejercicio 1.** Un Laboratorio realiza experimentos para evaluar la eficacia de un nuevo fármaco para la diabetes y para ello conformó dos grupos de pacientes: Grupo Alfa y Grupo Beta.

Ambos grupos registran la información de a lo sumo 10 pacientes (identificados de 1 a 10). De cada paciente se guarda: nombre, último resultado de glucosa (double) y última dosis recibida de fármaco (double). Sin embargo, los grupos difieren en la forma de aplicar el fármaco a los pacientes (*esto se detalla más adelante*)

- 1) Genere las clases necesarias, cada una con los constructores, estado, getters y setters adecuados. Tenga en cuenta que los grupos inicialmente no tienen pacientes.
- 2) Agregue a la clases que corresponda los métodos necesarios para:
  - a) Agregar un paciente *P* al grupo y retornar su número identificador en el grupo (ID).

- b) Obtener un paciente del grupo dado un ID válido (1 a 10).
- c) Aplicar una dosis a un paciente. Se recibe una dosis  $D$  (double) y se debe modificar su última dosis recibida a  $D$  y disminuir la glucosa en un valor aleatorio entre 0 y 1.
- d) Aplicar una dosis  $D$  (double) de fármaco a los pacientes del grupo, teniendo en cuenta que: en el Grupo Alfa se le aplica la dosis  $D$  a todos los pacientes; en el Grupo Beta se le aplica la dosis  $D$  a los pacientes cuya glucosa supera el valor 2.5.

*Ayuda: para implementar este inciso deberá usar los métodos incisos b) y c).*

- e) Obtener la representación string del grupo, la cual se compone por el ID,. nombre, última glucosa y última dosis de todos los pacientes del grupo

- 3) Realice un programa que instancie un Grupo Alfa y un Grupo Beta. Llene cada grupo con pacientes (el primero con 3 y el segundo con 4). Aplique una dosis  $D$  de fármaco (leída por teclado) a los pacientes de cada grupo. Imprima la representación string de cada grupo.

### Concurrente

**Ejercicio 1.** Implemente el siguiente juego. Existe un área delimitada por las esquinas (10,10) y (20,20) compartida entre tres robots jugadores. Además existe un robot fiscalizador.

Cada robot jugador debe tratar de juntar la mayor cantidad de flores posible, para esto tiene tres intentos. En cada intento se posiciona en una esquina del área determinada al azar, junta todas las flores de esa esquina y vuelve a su esquina original.

El robot fiscalizador determinará qué robot juntó más flores, y su cantidad juntada.

Los robots se posicionan inicialmente en (1,1), (2,1) y (3,1) y el robot fiscalizador en la esquina (4,1).

**Ejercicio 2.** Modifique el ejercicio 1 para que: el robot jugador que juntó más flores, las deposite en la esquina (21,21).