

# Programación Avanzada

## PRIMER PARCIAL 2015

Por favor siga las siguientes indicaciones:

- Escriba con lápiz.
- Escriba su nombre y número de documento en todas las hojas que entregue.
- Numere las hojas e indique el total de hojas en la primera de ellas.

### Problema 1 (15 puntos)

La compañía *Muebles Modernos* se dedica al desarrollo de muebles armados en base a componentes estándar disponibles en el mercado. Existe un departamento que diseña los productos, que se identifican por su nombre; además se debe saber el año de diseño del producto y su diseñador. Cada producto tendrá un proyecto asociado (inicialmente puede no tener ninguno), del que se conoce su código (que lo identifica) y su proyectista. Un proyecto comprende varios componentes, identificados por un número y que pueden ser de tipo estructural o de cerramiento. Un proyecto es llevado a cabo por uno o varios constructores, que deben tener habilidad en la manipulación de los componentes que comprende el proyecto. Para cada constructor se sabe su grado de habilidad en cada componente, representado como un número entero no negativo en el rango 0..5; se considera que un constructor tiene habilidad sobre un componente si su respectivo valor asociado es mayor o igual a 2. Tanto los diseñadores, como los proyectistas y constructores, son identificados por su número de documento de identidad.

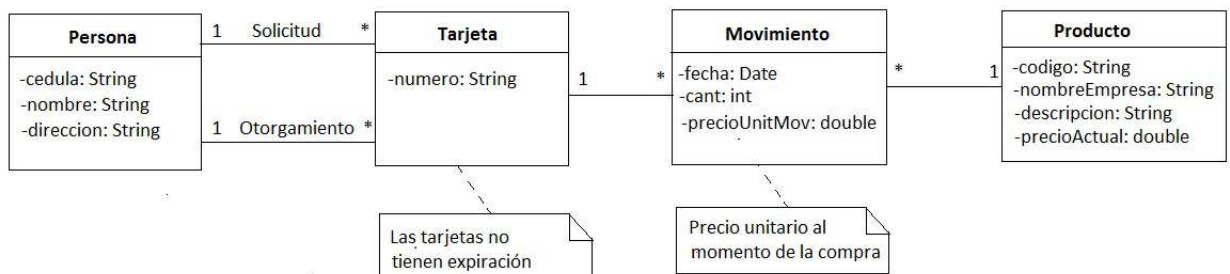
SE PIDE:

Construir el **Modelo de Dominio** de toda la realidad (incluyendo el caso de uso presentado) y presentarlo en un diagrama utilizando UML. Las **restricciones** deben ser expresadas en lenguaje natural.

### Problema 2 (15 puntos)

Considere un sistema de gestión de solicitudes de tarjetas y registro de movimientos usando las mismas.

A continuación se presenta el Modelo de Dominio y los Casos de Usos construidos en la etapa de análisis para dicho sistema:



Restricciones:

- La cédula identifica a una persona.
- El número identifica a una tarjeta.
- El código de producto y el nombre de la empresa identifica a un producto.

Caso de Uso:	Otorgamiento de tarjeta	Actor:	Empleado
Descripción:	El empleado indica al sistema el otorgamiento de una tarjeta conocida por su número a la persona identificada por su cédula de identidad. El sistema chequea la existencia de una solicitud de dicha tarjeta por la persona. En caso de que exista, se elimina la solicitud y se registra el otorgamiento. En caso contrario, la operación no debe tener efecto alguno sobre el sistema.		

Caso de Uso:	Ingreso de movimientos	Actor:	Empleado
Descripción:	El caso de uso comienza cuando el empleado quiere ingresar un conjunto de movimientos de compras indicando la tarjeta y la cédula de identidad de la persona que hizo la compra para que el sistema realice el chequeo de que la tarjeta ha sido otorgada a esa persona. Luego, el empleado agrega uno a uno los movimientos, indicando la fecha del movimiento, la cantidad de productos, el precio unitario, el código del producto, la descripción del producto y el nombre de la empresa. Durante este proceso, si el producto al cual se refiere el movimiento no existe, se agrega al sistema. Finalmente, el empleado le indica al sistema el fin del ingreso de movimientos. El sistema le devuelve la cantidad de movimientos ingresados.		

Caso de Uso:	Baja de tarjeta	Actor:	Empleado
Descripción:	El empleado indica al sistema la baja de una tarjeta dando el número de la misma. El sistema elimina el registro de solicitud u otorgamiento, según corresponda. Por otro lado, todos los movimientos de la tarjeta son eliminados. La persona y los productos involucrados son mantenidos en el sistema.		

**SE PIDE:**

- Realice los **Diagramas de Secuencia del Sistema (DSS)** para los Casos de Uso: *Otorgamiento de tarjeta, Ingreso de movimientos y Baja de tarjeta.*
- Expresar en lenguaje natural las **pre-condiciones y post-condiciones de los contratos** correspondientes a las operaciones de los DSS realizados en la parte a).

**Problema 3 (10 puntos)**

Implementar en C++ el datatype *Fecha* (archivos .h y .cpp), que representa a una fecha que incluye día, mes y año. Las operaciones que se deben considerar son: constructores (por defecto y común), setters, getters y destructor.

**Observaciones:**

- El constructor por defecto deberá crear una fecha correspondiente al 01/01/2015.
- El constructor común deberá crear una fecha a partir de los enteros que se reciben como parámetros (día, mes y año respectivamente).
- Asuma que los parámetros recibidos son correctos.
- No es necesario incluir directivas de preprocesador en el código.