

**Tecnólogo Informático – San José**  
**Programación Avanzada**  
**Examen Julio 2015**

- *Completar TODAS las hojas con el nombre y el número de cédula. Numerarlas y escribir el total en la primer hoja.*
- *Escriba las hojas de un solo lado.*
- *No se puede utilizar material de ningún tipo. **Apagar celulares.***
- *Solo se contestan dudas acerca de la letra de los ejercicios.*
- *El parcial dura 2 horas y media.*

**Problema 1 [25 puntos]**

- a) Cuándo se habla de re definición de operaciones y de sobrecarga de operaciones, ¿se habla del mismo concepto?. Explique y ejemplifique su respuesta.
- b) ¿Qué utilidad tienen las pre y post condiciones de un contrato de software?
- c) Explique el concepto de tipo asociativo. De un ejemplo.
- d) En el contexto del Diseño, describa los tipos de visibilidad que conoce.

**Problema 2 [35 puntos]**

Se le ha encomendado el análisis de un sistema de venta de entradas a eventos (e.g. recitales y conciertos) brindados por artistas del ambiente musical. Para ello se cuenta con la siguiente descripción de la realidad.

“El objetivo es registrar todos los eventos de los cuales la empresa vende entradas. De cada artista se conoce su nombre que lo identifica y una descripción del mismo. Los artistas pueden ser músicos o grupos. Si es un músico se conoce su fecha de nacimiento. Si es un grupo se debe conocer cada uno de los músicos que lo integran y el año de formación. De cada evento se conoce su nombre que lo identifica, el artista que participa, la fecha en la que se realiza el evento (e.g. 10 de setiembre 21.00 hs) y el lugar donde se realiza (e.g. Velódromo Municipal). Para cada lugar se conoce su nombre que lo identifica y su dirección. Cada lugar tiene diferentes tipos de localidades (e.g. Platea Alta, Campo VIP) cada una con un precio diferente para el evento. De cada tipo de localidad se conoce su nombre (que lo identifica en el lugar) y su capacidad (cantidad de asientos). Para un evento es necesario conocer el precio de las entradas y cuantas localidades se vendieron para cada tipo de localidad. Esto último se hace con la finalidad de llevar un registro de las ganancias generadas por la venta de entradas y además para verificar que no haya una sobreventa de las mismas.”

Adicionalmente se plantea el siguiente caso de uso:

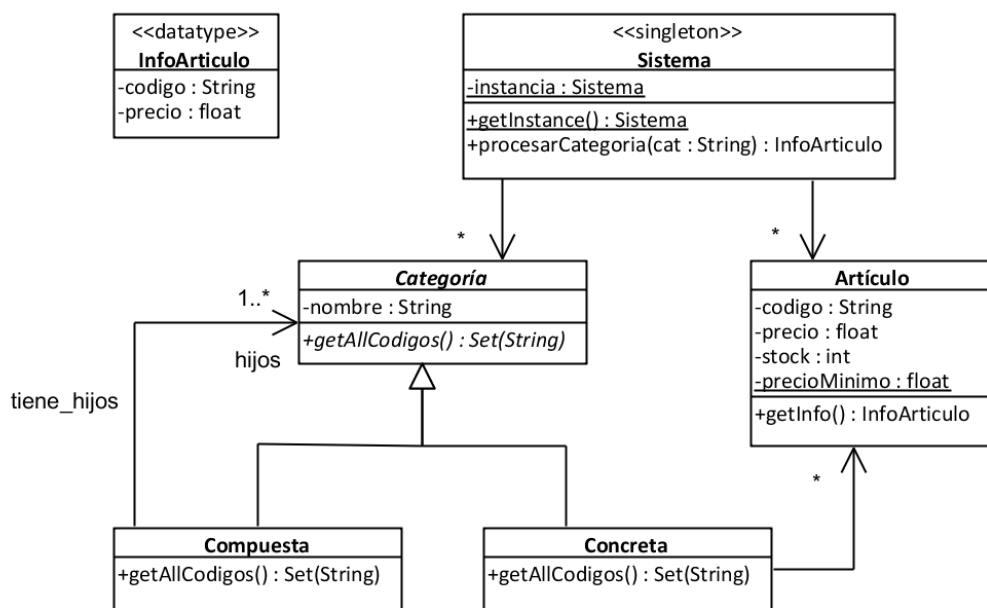
Nombre:	Alta artista
Actores:	Usuario
Sinopsis:	El usuario da de alta un nuevo artista en el sistema. Para ello ingresa el nombre y la descripción. En caso de ser músico ingresa su fecha de nacimiento, mientras que si es un grupo ingresa los nombres de cada uno de sus integrantes y el año de formación.

Se pide:

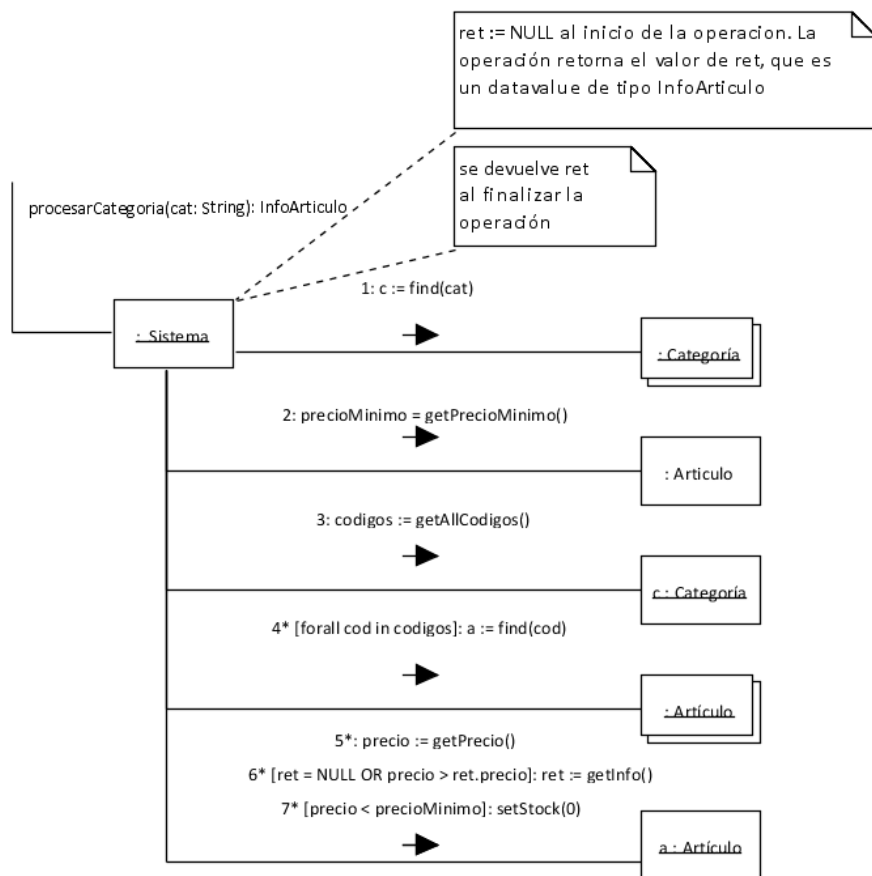
- Realizar el modelo de dominio en UML y expresar sus restricciones en lenguaje natural.
- Realizar el Diagrama de Secuencia del Sistema para el caso de uso Alta artista.

### Problema 3 [40 puntos]

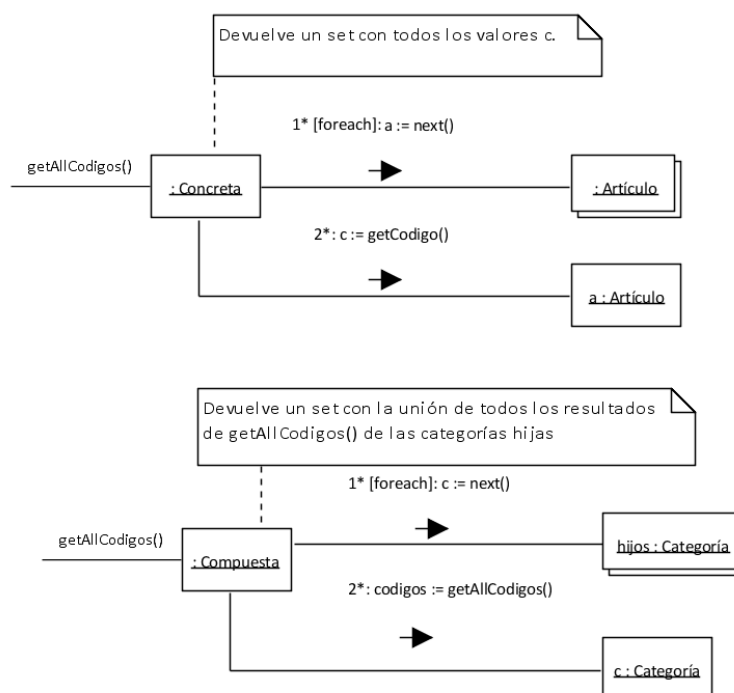
Se desea implementar un sistema que gestiona el inventario de artículos de una tienda. Los artículos tienen un código, precio y se conoce su stock. Además, los artículos se catalogan en distintas categorías que pueden estar anidadas dentro de otras categorías. A continuación se presenta un diagrama de clases de diseño parcial de la solución diseñada.



La operación `procesarCategoria(cat: String)` reduce a 0 el stock de todos los artículos de la categoría `cat` cuyo precio es menor a `Articulo::precioMinimo` y devuelve la información del artículo de precio mayor que esté asociado a dicha categoría usando la operación `Categoria::getAllCodigos()`. Se tiene el siguiente diagrama de comunicación para la operación `procesarCategoria()`.



La operación `procesarCategoria()` tiene como precondición que exista una categoría de nombre igual a `cat` en el sistema y en caso contrario lanza una excepción de tipo `std::invalid_argument`. La operación abstracta `getAllCodigos()` devuelve un conjunto con todos los códigos de los artículos asociados a la misma categoría (si es Concreta) o asociados recursivamente a través de la asociación `tiene_hijos` (si es Compuesta). A continuación los diagramas de comunicación correspondientes a dicha operación.



Se pide:

- i. Implementar en C++ los .h de las clases Sistema, Artículo, Categoría y Compuesta.
- ii. Implementar en C++ los .cpp de las clases Sistema y Compuesta.

Notas:

- Deben verificarse las precondiciones de las operaciones.
- No debe incluirse get y set de atributos.
- Suponga que se tiene una implementación de la clase String que implementa
- ICollectible e IKey.
- Suponga que se tiene una implementación de ICollection llamada Col e IDictionary
- llamada Dict.
- No incluya constructores ni destructores de las clases.
- No incluya las directivas del preprocesador.