

Programación Avanzada

EXAMEN DICIEMBRE 2008

19/12/2008

Nombre y Apellido

C.I.

Problema 0 (Total: 2 puntos)

Por favor siga las siguientes indicaciones:

- Este examen contiene un total de **3 páginas**.
- Escriba con **lápiz**.
- Escriba las hojas de **un solo lado**.
- 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.
- Comience cada ejercicio en una hoja nueva.
- El total máximo de puntos del examen es **100 puntos**.
- El examen se aprueba con **60 puntos**.

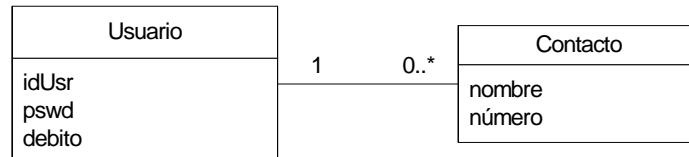
Problema 1 (Total: 28 puntos) [Teórico]

Responda las siguientes preguntas:

1. ¿Qué significan los conceptos *Herencia*, *Full Descriptor* y *Segment Descriptor* y cómo se relacionan? ¿Cómo se construye el **Full Descriptor** de un objeto?
2. Durante el curso se utilizaron algunos diagramas de UML que permiten expresar interacciones dinámicas. Describa cada uno explicando:
 - i. Para qué y cuándo (en qué etapa del proceso de desarrollo) se usa.
 - ii. En qué artefactos previos se basa para su realización.
 - iii. Qué elementos están presentes, y cuál es su uso.
3. Describa el algoritmo usado para construir el *Diagrama de Clases de Diseño*. Nombre los artefactos previos que utiliza como entrada de este algoritmo (no es necesario definirlos).
4. ¿Qué significa *Clase Abstracta* y qué elementos de C++ son necesarios para implementar este tipo de Clases? Escriba un ejemplo, en C++, de una **Clase Abstracta**, y de otra **Concreta**, que le permita explicar todos los conceptos involucrados. No es necesario realizar la explicación, salvo que lo considere muy necesario.

Problema 2 (Total: 35 puntos) [Práctico]

Está en construcción un sistema de telefonía por internet que permite a los usuarios registrados realizar llamadas a teléfonos fijos, cuyo cobro se realiza por medio de una tarjeta de crédito. Además, el sistema permite que los usuarios mantengan una lista de contactos personales a los efectos de facilitar la realización de llamadas. Un modelo de dominio preliminar es el que se muestra a continuación.



El principal caso de uso es el que permite realizar una llamada, especificado a continuación:

Nombre	Realizar una Llamada
Actores	Usuario
Descripción	El caso de uso comienza cuando el usuario se conecta al sistema indicando su nombre de usuario (que lo identifica) y contraseña, tras lo cual el sistema lo valida. Si el usuario es válido, puede realizar tantas llamadas como desee. Para realizar una llamada, el usuario ingresa de a uno los dígitos del teléfono al cual desea llamar y el sistema verifica que el numero que se está ingresando esa valido, informando de ello. Si el ingreso es invalido, el cliente puede consultar opcionalmente los contactos (nombre y teléfono) que posee en su lista de direcciones antes de intentar realizar una nueva llamada. Si el ingreso es válido y el usuario terminó de ingresar el número de teléfono (se supone que el número existe), el usuario indica al sistema que llame a lo cual el sistema inicia la conversación. Cuando el cliente decide terminar la misma, avisa de esto al sistema y éste devuelve los datos de la llamada realizada (duración, número llamado y costo de la llamada), pudiendo luego realizar una nueva llamada. Cuando el usuario no desea realizar más llamadas, se desconecta del sistema. El sistema devuelve el monto total de las llamadas realizadas e ingresa en la cuenta del usuario el monto a cobrar.

- i. Realice un único Diagrama de Secuencia del Sistema para el caso de uso anterior, incluyendo toda la información contenida en el mismo.

Para el almacenamiento de los contactos de un usuario se definió el siguiente caso de uso, el cual, puede ser llevado a cabo por una única operación *altaContacto()*.

Nombre	Alta de un Contacto
Actores	Usuario
Descripción	El caso de uso comienza cuando el usuario ingresa su nombre de usuario y pide al sistema agregar en su lista de contactos uno nuevo. Para ello especifica un nombre para el contacto (que lo identifica) y el número de teléfono del mismo. Si el contacto no se encuentra en la lista de contactos de ese usuario, el sistema da de alta al contacto en la lista. El sistema devuelve <i>true</i> si el contacto se da de alta, <i>false</i> en caso contrario.

- ii. Especifique las pre- y post- condiciones de la operación *altaContacto()*.

Problema 3 (Total: 35 puntos) [Práctico]

Se desea construir un sistema que permita comunicar aplicaciones entre sí a través del envío y la recepción de mensajes. Las aplicaciones colocan los mensajes que desean enviar en canales. Posteriormente, otras aplicaciones podrán consumir los mensajes de estos canales completando así la comunicación.

Las aplicaciones poseen un identificador numérico único. Cada una puede consumir y enviar mensajes desde y hacia muchos canales, pero no puede realizar ambas tareas en un mismo canal. Previo al envío y/o consumo de mensajes, una aplicación debe registrarse con el canal a interactuar, siendo de interés almacenar la fecha en que se registro. Por último, para cada aplicación se debe conocer el conjunto de mensajes que fueron enviados o consumidos desde cada canal.

Los canales poseen un nombre que los identifica. Además deben mantener el conjunto de mensajes que fueron enviados a través de ellos. Los canales pueden ser de dos tipos: punto a punto (p2p) o publicación-subscripción (pub/sub).

Los canales p2p se caracterizan por permitir que cada mensaje enviado sólo pueda ser consumido por una única aplicación de las muchas consumidoras que pueda tener el canal.

Los canales pub/sub se caracterizan por permitir la subscripción de aplicaciones para avisar la llegada de los mensajes a ser consumidos. Para que una aplicación pueda subscribirse a un canal pub/sub debe haberse registrado previamente como consumidora del mismo. Las subscripciones deben mantener su fecha de creación.

De cada mensaje se conoce un identificador numérico único, la aplicación que lo envió, los canales a los que fueron enviados y su contenido (texto).

Se pide:

Construya un Diagrama de Modelo e Dominio UML para la realidad descripta. Las restricciones deben ser expresadas en lenguaje natural. Modele exclusivamente en base a la información presente en la descripción.

Nota: suponga dado el datatype Fecha con los operadores de comparación definidos:

