# Universidad Interamericana de Panamá Facultad de Ingeniería, arquitectura y diseño Escuela de Ingeniería y Sistemas.

Materia: Programación de Computadoras II Cuatrimestre: IQ\_2020

Código: 301-00070 Facilitador: Leonardo Esqueda

Nombre completo: Adrián Quintero

Carrera: Licenciatura en Ingeniería en Sistemas

# Parcial #1

ID:

## Instrucciones: La prueba parcial está enfocada en dos (2) pates: Teórico y Práctico; constando con 20 problemas. Lea detenidamente antes de responder.

1. **Teoría:**
   1. **¿Qué diferencia un examen de estructura lineal a una estructura orientada a objetos (POO)? 5ptos.**

Una estructura lineal se caracteriza porque sus elementos están en secuencia, uno luego del otro. A esto se le conoce como programación estructurada, en el que los algoritmos se desarrollan por procedimientos (funciones) y el diseño del programa es modular. Cada función es un procedimiento y se escribe de forma descendente; mientras que en una estructura orientada a objetos la unidad de programación es la clase, y se agrupa el código en forma de objetos, los cuales son una abstracción de elementos de la vida real. Encapsula los datos y las funciones (ahora llamándome atributos y métodos) en objetos que se relacionan entre sí.

* 1. **¿Dónde se utiliza una estructura lineal y dónde una estructura orientada a objetos (POO)? 5ptos.**

Se puede utilizar una estructura lineal en programas pequeños y sencillos (como una calculadora básica), mientras que cuando desarrollamos programas grandes y extensos en los que intervienen una gran variedad de elementos, es importante utilizar una estructura orientada a objetos. De esa forma es más fácil identificar cada funcionalidad del programa y detectar errores.

* 1. **Mencione 8 tipos de variables y sus definiciones y en explique en qué casos se utilizan 5ptos.**

**int:** son variables que almacenan un número positivo o negativo entero (no decimal). Puede almacenar números desde -231 hasta 231 – 1.

Los enteros se suelen utilizar para hacer iteraciones (ciclos for), contadores, constantes o para almacenar cualquier otro tipo de dato que no necesite parte fraccionaria, como la edad de una persona.

**float:** la variable tipo float representa números reales (con decimales), es decir con parte fraccionaria. Pero estos no se representan de manera exacta, sino que son aproximaciones. Por ejemplo, el número 0.7 se representa internamente como 0.69999999999999996. Su tamaño en memoria es de 4 bytes.

Se suele utilizar para hacer facturas y otro tipo de cálculos.

**double:** Al igual que float, almacena números con coma flotante, mantisa y exponente; pero utilizan una mayor precisión. Su tamaño en memoria es de 8 bytes.

Se utiliza en casos en los que se necesita una mayor precisión con los decimales.

**boolean:** Es un tipo de dato lógico que solo permite dos valores, verdadero o falso. En la computadora se representan como 0 (falso) y 1 (verdadero). Ocupa 1 byte.

Se utiliza para hacer verificaciones, o para hacer condicionales (if).

**long:** Al igual que int, almacena un número positivo o negativo no decimal, con la diferencia de que su tamaño en memoria es más grande y por lo tanto el rango de números que puede almacenar es mayor. El rango es de -263 hasta 263 – 1.

Se utiliza cuando se necesita una mayor cantidad de números de la permitida por int.

**char:** El tipo de variable char se utiliza para representar caracteres individuales Unicode. Cada caracter tiene un entero equivalente, por lo que se puede considerar como una clase entera pequeña. Su tamaño en memoria es de 2 bytes. Cada variable almacena un solo caracter, pero se puede almacenar palabras o contraseñas en forma de arreglos.

**String:** String es técnicamente un arreglo de caracteres, que también se le conoce como cadena de caracteres.

Se utiliza para almacenar palabras, frases, textos, etc. Son una sucesión de caracteres.

**Byte:** Es un tipo entero de 8 bits con complemento a dos, que toma valores entre -128 y 127.

Es útil debido al poco espacio en memoria que ocupa. No reconoce decimales.

* 1. **¿Qué es un paradigma? 5ptos.**

Un paradigma es un modelo o un patrón que se sigue en una determinada situación, o un conjunto de normas y reglas para resolver un problema.

* 1. **¿A qué se refiere un paradigma a nivel de programación? 5ptos.**

A nivel de programación, un paradigma es el estilo que se utiliza para desarrollar un programa. Indica la manera en que se estructuran y organizan las tareas que el programa debe llevar a cabo. Un ejemplo de paradigma es la programación orientada a objetios.

* 1. **¿Qué es un objeto? 5ptos.**

Un objeto es un componente del programa que posee atributos y métodos. Sus atributos son sus propiedades o características que lo definen, y sus métodos son sus funciones o acciones que puede realizar. La mejor forma de entenderlo es comparandolo con un objeto en la vida real, por ejemplo, un carro. Sus atributos serían su color, marca, matrícula, etc. Y sus métodos serían acelerar, frenar, encender el motor, etc.

* 1. **¿Cuáles son los pilares de la programación orientada a objetos? 5ptos.**

Los pilares de la programación orientada a objetos son:

**Abstracción:** las características específicas del objeto que lo distinguen los demás objetos. Es decir, sus atributos y su comportamiento.

**Encapsulamiento:** se organizan los datos y métodos de una estructura evitando el acceso a datos por cualquier otro medio distinto a los especificados, para garantizar la integridad de los datos.

**Herencia:** permite reutilizar código usando las características y comportamientos de una clase superior en las clases que deriven de esta.

**Polimorfismo:** permite definir varios métodos o comportamientos de un objeto bajo el mismo nombre de una forma que sea posible modificar los parámetros del método o reescribir su funcionamiento. En pocas palabras, permite que una entidad pueda tomar diferentes formas.

* 1. **¿A qué se refieren las características de un objeto? 5ptos.**

Las características de un objeto se refieren a sus atributos que lo distingue de los otros objetos, por ejemplo, los atributos marca o modelo del objeto automóvil.

* 1. **¿Cuál es el método para transformar una variable de tipo entero a uno de tipo doble o flotante y cómo se utiliza?**

Hay dos métodos:

1. Crear una variable de tipo float o double y asignarle la variable entera añadiéndole un paréntesis con el tipo de variable a que queremos convertir, por ejemplo:

float miFloat = (float) miEntero;

o,

double miDouble = (double) miEntero;

1. Crear un objeto de la clase Integer, instanciándola con el entero a convertir y utilizar su método .floatValue() que devolverá el mismo valor pero de tipo flotante. Ejemplo:

Integer intg = new Integer(miEntero);

float miFloat = intg.floatValue();

* 1. **¿Cuál es la diferencia entre librerías y paquetes en programación?**

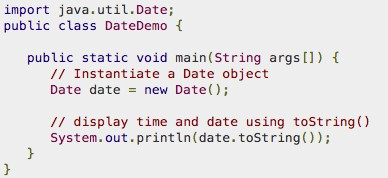
Un paquete es una agrupación de clases que tienen una funcionalidad en común, lo que permite agrupar las distintas partes de un programa. Estos paquetes también proporcionan un método de control de acceso, para evitar que otros paquetes puedan acceder a las clases que son de uso exclusivo para una funcionalidad del programa.

Las librerías son un conjunto de clases que tienen una serie de métodos y atributos que podemos utilizar para reutilizar código y no tener que implementar nosotros esas funcionalidades. Por lo general, las clases que nos proporciona las librerías ya están compiladas, por lo que no podemos modificar su código, sino únicamente utilizar sus métodos y atributos. Estas clases se agrupan en un paquete, pero la librería a veces puede estar conformada por muchos paquetes.

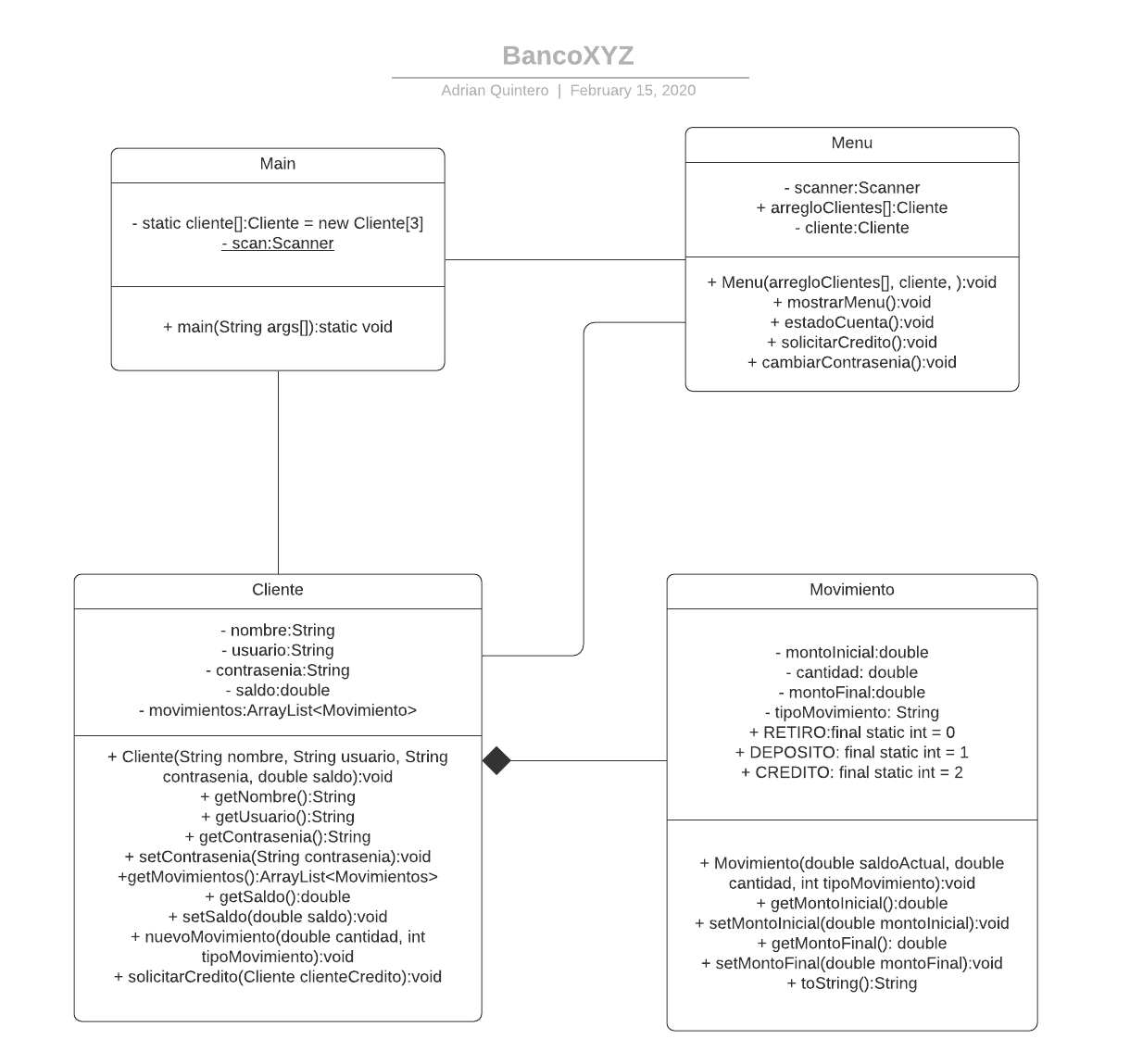
En conclusión, un paquete es una forma de agrupar clases con una funcionalidad en común y una librería es un conjunto de clases o paquetes compilados que nos proporcionan sus métodos para utilizar en nuestro código una funcionalidad ya implementada.

1. **Práctica:**
2. El banco XYZ necesita desarrollar un sistema online en el cual sus usuarios puedan revisar su estado de cuenta por internet. Para esto, se necesita desarrollar la lógica inicial de la siguiente manera en POO:
   1. Los usuarios (ej. Cliente1 y Cliente2) deberán ingresar sus respectivas credenciales (usuario y contraseña) qué, inicialmente se las otorga el propio sistema del banco. (Se debe tomar en consideración que la institución bancaria, solo otorga este sistema a los usuarios que mantengan un depósito mínimo de $1,500 en sus cuentas).
   2. Una vez el cliente haya ingresado por primera vez en la cuenta, el sistema debe mostrar la fecha y hora de ingreso. Y, a su vez, le debe pedir que actualice (cambie) la contraseña.
   3. Luego de haber cambiado la contraseña, el sistema debe mostrar al cliente un menú con el siguiente formato:
      1. Ver estado de la cuenta: se debe mostrar los movimientos de saldo que ha tenido el cliente. (Ej. el monto inicial y el monto final de la cuenta).
      2. Solicitar un Crédito: El banco está en la potestad de utilizar hasta un 90% del saldo actual de una 3ra cuenta para prestar al cliente. (Ej. Puede utilizar hasta un 90% del saldo del cliente2 para prestárselo al cliente1) y debe mostrar que el banco cobrará un 30% del valor prestado como comisión.
      3. Cambiar contraseña: El cliente puede cambiar su contraseña de acceso.
      4. Salir: Termina el proceso y debe mostrar la fecha y hora en la que el cliente cerró la sesión.

## Nota1\*: En este problema puede utilizar: Scanner, Date, if, else, switch, for, while, do-while y arrays según considere.

**Nota2\*\*: uso de Date ** **15ptos**

1. Desarrolle el problema práctico #1 en Diagrama de Flujos y a nivel de pseudocódigo 10ptos



1. Desarrolle un algoritmo que pueda resolver el siguiente problema: Un camión puede transportar, como máximo, 12 Tm. por viaje. En cierto viaje se desea transportar al menos, 5 Tm. de la mercancía A y un peso de la mercancía B que no sea inferior a la mitad del peso que se transporte de A. Sabiendo que cobra 4 céntimos por kilo de mercancía A y 3 céntimos. por kilo de mercancía B transportadas, ¿cómo se debe cargar el camión para obtener la ganancia máxima? 10ptos
2. Crear una clase Libro que contenga los siguientes atributos: 10ptos

* ISBN
* Titulo
* Autor
* Número de páginas

Crear sus respectivos métodos get y set correspondientes para cada atributo. Crear el método toString() para mostrar la información relativa al libro con el siguiente formato: “El libro con ISBN creado por el autor tiene páginas”

En el fichero main, crear 2 objetos Libro (los valores que se quieran) y mostrarlos por pantalla.

Por último, indicar cuál de los 2 tiene más páginas.

1. Haz una clase llamada Persona que siga las siguientes condiciones:

Sus atributos son: nombre, edad, DNI, sexo (H hombre, M mujer), peso y altura. No queremos que se accedan directamente a ellos. Piensa que modificador de acceso es el más adecuado, también su tipo. Si quieres añadir algún atributo puedes hacerlo.

Por defecto, todos los atributos menos el DNI serán valores por defecto según su tipo (0 números, cadena vacía para String, etc.). Sexo sera hombre por defecto, usa una constante para ello.

Se implantaran varios constructores:

Un constructor por defecto.

Un constructor con el nombre, edad y sexo, el resto por defecto. Un constructor con todos los atributos como parámetro.

Los métodos que se implementaran son:

calcularIMC(): calculara si la persona esta en su peso ideal (peso en kg/(altura^2 en m)), si esta fórmula devuelve un valor menor que 20, la función devuelve un -1, si devuelve un número entre 20 y 25 (incluidos), significa que esta por debajo de su peso ideal la función devuelve un 0 y si devuelve un valor mayor que 25 significa que tiene sobrepeso, la función devuelve un 1. Te recomiendo que uses constantes para devolver estos valores.

esMayorDeEdad(): indica si es mayor de edad, devuelve un booleano.

comprobarSexo(char sexo): comprueba que el sexo introducido es correcto. Si no es correcto, sera H. No sera visible al exterior.

toString(): devuelve toda la información del objeto.

generaDNI(): genera un número aleatorio de 8 cifras, genera a partir de este su número su letra correspondiente. Este método sera invocado cuando se construya el objeto. Puedes dividir el método para que te sea más fácil. No será visible al exterior.

Métodos set de cada parámetro, excepto de DNI. Ahora, crea una clase ejecutable que haga lo siguiente:

Pide por teclado el nombre, la edad, sexo, peso y altura.

Crea 3 objetos de la clase anterior, el primer objeto obtendrá las anteriores variables pedidas por teclado, el segundo objeto obtendrá todos los anteriores menos el peso y la altura y el último por defecto, para este último utiliza los métodos set para darle a los atributos un valor. Para cada objeto, deberá comprobar si está en su peso ideal, tiene sobrepeso o por debajo de su peso ideal con un mensaje.

Indicar para cada objeto si es mayor de edad.

Por último, mostrar la información de cada objeto.

Puedes usar métodos en la clase ejecutable, para que os sea más fácil. 15p5os