

# Programación 2

## Segundo Parcial - 2do Semestre 2016

El parcial se compone en 3 partes, dos individuales (una en papel y otra en computadora) y una grupal (en papel).

### Cronograma del Parcial

- 18:15: Comienzo del parcial. Inicio de parte teórica individual.
- 19:10: Finaliza la parte teórica individual.
- 19:10 - 19:20: Recreo.
- 19:20: Realización de parte práctica individual.
- 20:05: Finalización de parte individual (entrega solamente via webasignatura).
- 20:10: Inicio de prueba grupal.
- 20:50: Entrega de la prueba grupal.

## PARTE TEÓRICA INDIVIDUAL

### Pregunta 1

Corrija el siguiente código para cumplir con los principios OCP y SRP.

```
public class Instagram {
    public Bitmap Comprimir20(Bitmap imagen){/** Comprime la imagen 20% y la devuelve **/}
    public void SubirFotoInst(Bitmap imagen){/** Sube la foto a Instagram **/}
}

public class Facebook {
    public Bitmap Comprimir10(Bitmap imagen){/** Comprime la imagen 10% y la devuelve **/}
    public void SubirFotoFB(Bitmap imagen){/** Sube la foto a Facebook **/}
}

public class RedSocial {
    private String GetExtensionArchivo (String archivo) {/** Extensión del archivo **/}
    private Bitmap PngToBitmap (String fotoPng) {/** Convierte la foto png a bitmap **/}
    private Bitmap JpgToBitmap (String fotoJpg) {/** Convierte la foto jpg a bitmap **/}
    public void SubirFoto (String archivo, int tipoRed) {
        String ext = GetExtensionArchivo (archivo);
        Bitmap imagen = null;
        if (ext == ".png")
            imagen = PngToBitmap(archivo);

        if (ext == ".jpg")
            imagen = JpgToBitmap(archivo);

        if (tipoRed == 0) {
            Facebook face = new Facebook();
            imagen = face.Comprimir10(imagen);
            face.SubirFotoFB(imagen);
        }
        if (tipoRed == 1) {
            Instagram inst = new Instagram();
            imagen = inst.Comprimir20(imagen);
            inst.SubirFotoInst(imagen);
        }
    }
}
```

## Pregunta 2

```
public class BaseDeDatos {
    private Boolean Conectar () { /** Se conecta a la base de datos */ }
    private Boolean Desconectar () { /** Se desconecta de la base de datos */ }
    public void CerrarConexion() { /** Cierra la conexión */ }
    public int AlmacenarInformacion(String info) {
        If (! Conectar()) {
            return -1;
        }
        If (info == null) {
            return -2;
        }
        /** Se realiza el almacenamiento de la info */
        Desconectar();
        return 0;
    }
}

public class Program {
    public static void Main (String[] args) {
        String msg = "";
        BaseDeDatos bd = new BaseDeDatos();
        int res = db.AlmacenarInformacion("Prueba");
        if (res == -1) {
            msg = "Hubo un error al conectarse";
        }
        if (res == -2) {
            msg = "La información no puede ser vacía";
        }
        if (res == 0) {
            msg = "Operación realizada con éxito.";
        }
        Console.WriteLine (new Date().ToString() + msg);
    }
}
```

Modifica el código anterior para realizar dos excepciones (ErrorConexionExcepcion (lanzada cuando existe un error en la conexión) y InfoVacíaExcepcion (lanzada cuando es vacía la información) y recodar que siempre debe mostrarse el mensaje (junto con la fecha/hora) de la operación al momento de almacenar la información a través de la base de datos.

## Pregunta 3

Se pide realizar un **modelo de clases** para el siguiente problema.

Un videoclub necesita gestionar el alquiler de sus películas por parte de sus clientes. Todas las películas tienen un código numérico, un nombre y un precio base de alquiler por día (fijo para todas las películas). Un cliente alquila una película por un cierto tiempo (días) y el precio final del alquiler depende de dicho tiempo y la categoría de la película. Las tarifas de las películas difieren según su tipo: infantil, normal o estreno. Los precios de los alquileres son:

- Película normal. Si el alquiler es por más de 2 días, se cobra la mitad el resto de los días.
- Película estreno. El alquiler es 3 veces más del precio base por la cantidad de días que la alquile.
- Película infantil. El alquiler es la mitad del precio base por los días alquilados.

El programa deberá realizar:

- Agregar películas al stock.
- Agregar clientes al sistema.
- Proceso del alquiler de películas por parte de los clientes.

**Nota:** Para la parte individual práctica, deberán proveer un programa en consola que realice todas las funciones anteriormente solicitadas, así como también un Main con un claro juego de datos que ejemplifique su uso.