

DATOS PERSONALES		FIRMA
Nombre:	DNI:	
Apellidos:		

  

ESTUDIO	ASIGNATURA	CONVOCATORIA
MÁSTER UNIVERSITARIO EN INGENIERÍA MATEMÁTICA Y COMPUTACIÓN (PLAN 2016)	4391010005.- MÉTODOS AVANZADOS DE PROGRAMACIÓN CIENTÍFICA Y COMPUTACIÓN	Ordinaria

  

FECHA	MODELO	CIUDAD DEL EXAMEN
19-21/03/2021	Modelo - C	

  

Etiqueta identificativa
-------------------------

## INSTRUCCIONES GENERALES

1. La duración del examen es de **2 horas**.
2. Escribe únicamente con **bolígrafo/esfero azul o negro**.
3. No está permitido utilizar más hojas de las que te facilita la UNIR (puedes utilizar folios para hacerte esquemas u organizarte pero **se entregarán junto al examen**).
4. **El examen PRESENCIAL supone el 60%** de la calificación final de la asignatura. Es necesario aprobar el examen, para tener en cuenta la evaluación continua, aunque esta última sí se guardará para la siguiente convocatoria en caso de no aprobar.
5. No olvides **rellenar EN TODAS LAS HOJAS los datos del cuadro** que hay en la parte superior con tus datos personales.
6. El **DNI/NIE/PASAPORTE debe estar sobre la mesa** y disponible para su posible verificación.
7. **Apaga y retira del alcance los teléfonos móviles**.
8. **Retirar del alcance y visibilidad el smartwatch**.
9. Las preguntas se contestarán en **CASTELLANO**.
10. El profesor tendrá muy en cuenta las **faltas de ortografía** en la calificación final.
11. Los gráficos y el código los puede realizar a mano e insertar en el documento una foto de lo realizado. En este caso, también debe ceñirse al espacio indicado para cada pregunta, no pudiendo reducirse la foto para que parezca que se ha ocupado menos espacio.

## Puntuación

### PREGUNTAS TEORICO PRÁCTICAS

- Puntuación máxima 3,00 puntos

### PROGRAMACIÓN Y DISEÑO ORIENTADO A OBJETOS

- Puntuación máxima 3,00 puntos

### PROGRAMACIÓN CONCURRENTES

- Puntuación máxima 4,00 puntos

**NO UTILIZAR ESTA  
PARTE DE LA HOJA**

DATOS PERSONALES		FIRMA
Nombre:	DNI:	
Apellidos:		

- Cada una de las tres preguntas debe ser desarrollada como máximo en una cara.
- Cada pregunta vale 1 punto.

1. Defina las relaciones de composición y agregación. ¿Qué diferencias existen entre ellas?

Según la siguiente especificación de clases:

```
class Pagina{
protected:
    Imagen* miListaImágenes;
    Texto txt;
    //...
public:
    Pagina(Imagen*, Texto*);
}
```

```
class Documento {
protected:
    Pagina* portada;
    Pagina contraportada;
    //...
}
```

Muestre su representación en UML e indique la forma de implementación. Justifique la respuesta.

(1 punto)

NO UTILIZAR ESTA  
PARTE DE LA HOJA



DATOS PERSONALES		FIRMA
Nombre:	DNI:	
Apellidos:		

**2.**

Explique e implemente el patrón Singleton.

Construya el diagrama de clases correspondiente, ponga un ejemplo de uso e impleméntelo

**NO UTILIZAR ESTA  
PARTE DE LA HOJA**



DATOS PERSONALES		FIRMA
Nombre:	DNI:	
Apellidos:		

3. Revise el siguiente código en Java, explique su funcionamiento y el problema que presenta.

```
class Interna{

    protected boolean _cond=false;

    synchronized void metodoConGuarda(){

        while (!cond)

            try{wait();} catch (InterruptedException e){}

        // etc...

    }

    synchronized void metodoQueLiberaColaWait(boolean c){

        _cond=c;

        notifyAll();

    }

}

class Externa{

    protected Interna _interna = new Interna();

    synchronized void utilizar(){

        _interna.metodoConGuarda();

    }

    synchronized void liberar(boolean c){

        _interna.metodoQueLiberaColaWait(c);

    }

}
```

(1 punto)

**NO UTILIZAR ESTA  
PARTE DE LA HOJA**





DATOS PERSONALES		FIRMA
Nombre:	DNI:	
Apellidos:		

1. Se quiere gestionar las obras de una empresa de construcción. Las tareas que se realizan en una obra se organizan por partidas (conjunto de tareas). Cada obra requerirá de una serie de partidas para su consecución. A cada partida se le puede asignar una cantidad limitada de material necesaria para cumplir con las tareas que la componen. Cada tarea requiere de uno o más materiales. Cada material tiene un único proveedor y un precio. Algún proveedor puede encargarse de más de un material
- Realice el diseño de clases en UML e identifique, al menos, dos atributos y dos métodos para cada clase. Implemente las clases. Los métodos no es necesario implementarlos, es suficiente con la cabecera.
  - Identifique las relaciones y su cardinalidad. Impleméntelas y explique y justifique la implementación.
  - Haga un esbozo de la implementación de la función obtener\_precio de una partida (mostrando el uso de objetos relacionados con la clase que representa una partida) y de como la usaría el programa principal.

**(3 puntos)**

**NO UTILIZAR ESTA  
PARTE DE LA HOJA**



DATOS PERSONALES		FIRMA
Nombre:	DNI:	
Apellidos:		

1. Diseñe una clase Buffer que implemente un buffer limitado mediante un array estándar del lenguaje Java. Su comportamiento debe ser el siguiente:

- a. No se debe permitir la inserción y la extracción simultánea.
- b. No se debe permitir más de un número fijo de hilos  $N$ , que deben esperar para insertar un elemento cuando el buffer esté lleno. Si ya se tiene  $N$  hilos esperando para insertar, el siguiente hilo recibirá una notificación de rechazo mediante una excepción.
- c. El número de hilos que esperan para la extracción en un buffer vacío puede ser ilimitado.

Simule el funcionamiento creando hilos para inserción y extracción. Explique y justifique todo lo implementado.

**(4 puntos)**

**NO UTILIZAR ESTA  
PARTE DE LA HOJA**

DATOS PERSONALES		FIRMA
Nombre:	DNI:	
Apellidos:		



DATOS PERSONALES		FIRMA
Nombre:	DNI:	
Apellidos:		

**NO UTILIZAR ESTA  
PARTE DE LA HOJA**



DATOS PERSONALES		FIRMA
Nombre:	DNI:	
Apellidos:		

**B O R R A D O R**  
**PÁGINA NO VÁLIDA PARA RESPONDER**

**B O R R A D O R**  
**PÁGINA NO VÁLIDA PARA RESPONDER**