

PracticaSeptiembre2020.pdf



Hippo



Programación Orientada a Objetos



2º Grado en Ingeniería Informática



Escuela Superior de Ingeniería
Universidad de Cádiz

Máster Online en Ciberseguridad

Nº1 en España según El Mundo



Hasta el 46%
de beca



Mejor Máster
según el
Ranking de
El MUNDO

Para ser el mejor hay que aprender
de los mejores.

IME
Smart Education

Deloitte

Infórmate

Consigue Empleo o Prácticas

Matrícúlate en IMF y accede sin coste a nuestro servicio de Desarrollo Profesional con más de 7.000 ofertas de empleo y prácticas al mes.



IMF
Smart Education

Ejercicio 1 septiembre (2 puntos)

Sea una clase llamada **Binario** que almacena un número binario de longitud arbitraria, representado por la secuencia de bits que lo componen (dígitos 0 y 1).

La secuencia de bits de un **Binario** se almacena de menos a más significativo en el atributo llamado **bits**, un vector de bajo nivel de **unsigned int**. Cada elemento del vector **bits** contiene hasta **sizeof(unsigned int)** bytes de la secuencia. Los bits más significativos del último elemento del vector que sobren valdrán 0 y se ignorarán. El tamaño del vector (**m**) depende de la longitud en bits (**n**) del número binario y viene dado por la expresión $m = (n + \text{bits_elto} - 1) / \text{bits_elto}$, donde la constante **bits_elto** corresponde al número de bits de **unsigned int**.

Ejemplo: Representación de un **Binario** de $n = 57$ bits, suponiendo que **sizeof(unsigned int) = 2 bytes = 16 bits = bits_elto**, almacenado en el atributo **bits**, un vector de bajo nivel de $m = (n + \text{bits_elto} - 1) / \text{bits_elto} = (57 + 16 - 1) / 16 = 4$ **unsigned int** en el que sobran los 7 bits más significativos.

Binario = 1 0101 0011 0011 1010 0000 1000 1010 0111 0000 0000 0111 0000 1010 0011

3	2	1	0	
0000 0001 0101 0011	0011 1010 0000 1000	1010 0111 0000 0000	0111 0000 1010 0011	bits
bits sobrantes				

```
class Binario {
public:
    // ...
private:
    static const size_t bits_elto = CHAR_BIT * sizeof(unsigned); // CHAR_BIT = bits por byte
    size_t n; // longitud en bits
    size_t m; // longitud del vector de enteros
    unsigned* bits; // vector de enteros
};
```

Declare e implemente los mínimos métodos imprescindibles para que las siguientes instrucciones proporcionen los resultados descritos en los comentarios. Si es válido el comportamiento por defecto de alguno de esos métodos, explique por qué.

```
Binario b1(12);        // b1 = 000000000000 (12 dígitos de valor 0)
Binario b2;            // b2 = 0 (1 dígito de valor 0)
b2 = ~b1;             // b2 = 111111111111 (12 dígitos de valor 1)
b2 = std::move(b1);    // b2 = 000000000000 (12 dígitos de valor 0)
```

Instrucciones de entrega:

- Escriba a mano su respuesta en papel.
- Coloque su DNI o tarjeta universitaria sobre cada página escrita (sin que tape el texto) y capture una imagen nítida.
- Renombre el fichero de cada imagen con su nombre y apellidos y añada un número correlativo de página.
- Suba cada fichero de imagen.

¿Quieres conocer todos los servicios?



WUOLAH

Ejercicio 2 septiembre (1,5 puntos)

El CAC (Centro de Atención al Cliente) de cierta empresa recibe llamadas telefónicas de sus clientes. Una llamada es inicialmente asignada a un operador; si este no puede resolver la cuestión, la transfiere a un compañero especialista, el cual procede de la misma forma: atiende al cliente y si no puede resolver el asunto, transfiere la llamada a otro operador. Las llamadas se transfieren entre operadores hasta que son atendidas satisfactoriamente.

La dirección del CAC está interesada en mantener un histórico de todas las llamadas recibidas por su plantilla de operadores. Un operador atenderá diversas llamadas durante su jornada laboral, quedando registradas tanto la fecha (*string*), hora de inicio (*string*) y duración de la atención del operador a cada llamada. Así mismo, se registrarán los operadores que atienden cada llamada.

Implemente las clases **Operador** y **Llamada** con los atributos necesarios para implementar la relación entre ellas, así como los siguientes métodos:

- **recibir()**: un operador recibe una llamada.
- **asignar()**: una llamada se asigna a un operador.
- **llamadas()**: llamadas atendidas por un operador.
- **asistente()**: operadores que atendieron una llamada.

Instrucciones de entrega:

- Escriba a mano su respuesta en papel.
- Coloque su DNI o tarjeta universitaria sobre cada página escrita (sin que tape el texto) y capture una imagen nítida.
- Renombre el fichero de cada imagen con su nombre y apellidos y añada un número correlativo de página.
- Suba cada fichero de imagen.

Tiempo máximo 20 minutos

EXAMEN SORPRESA

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos Holandés con una garantía de hasta 100.000 euros por depositante. Consulta más información en Ing.es

1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

¿Qué te pide la Cuenta NoCuenta?

- ✓ a) Nada
- ✓ b) Nada de nada

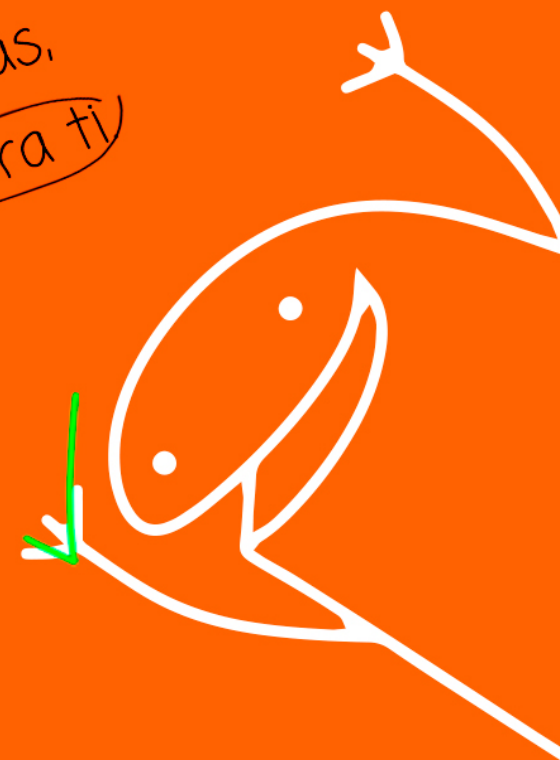
¿Qué comisiones* tiene?

- ✓ a) 0 %
- ✓ b) 0,00000000 %

¿Cuál es su permanencia?

- ✓ a) Ninguna
- ✓ b) Si ya sabes la respuesta pa qué preguntas

* Si has acertado
todas las respuestas,
esta cuenta es (para ti)



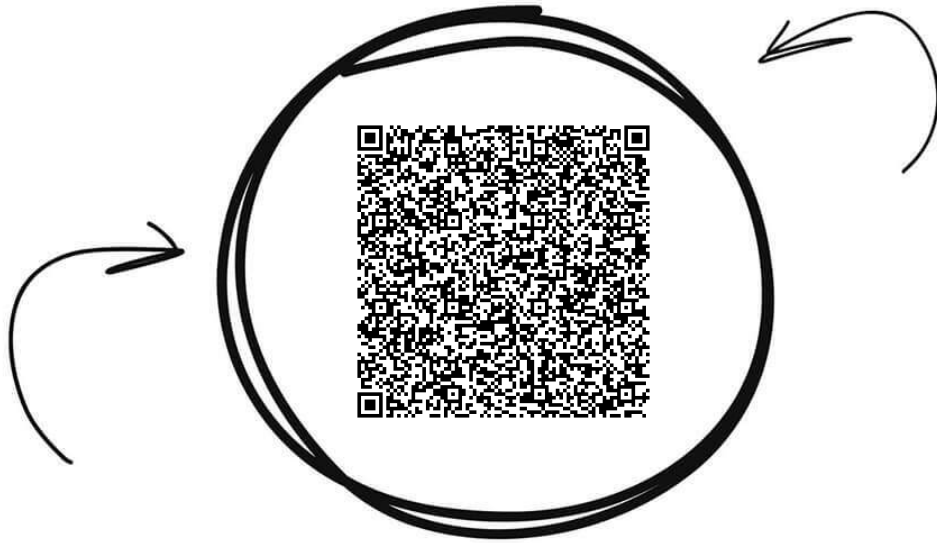
¡Me apunto!

* TIN 0 % y TAE 0 %.



do your thing

Programación Orientada a Obj...



Banco de apuntes de la

WUOLAH



**Comparte estos flyers en tu clase y
consigue más dinero y recompensas**

- 1** Imprime esta hoja
- 2** Recorta por la mitad
- 3** Coloca en un lugar visible para que tus compis puedan escanar y acceder a apuntes
- 4** Llévate dinero por cada descarga de los documentos descargados a través de tu QR



Ejercicio 3 septiembre (1,5 puntos)

Sea **Instrumento** una clase abstracta de la que derivan las clases **Bombo**, **Guitarra**, **Clarinete** y otras. Supongamos que existe una función para tocar instrumentos definida como sigue:

```
#include <typeinfo>

void tocar(Instrumento& i)
{
    if(typeid(i) == typeid(Bombo)) {
        // hacer sonar el bombo
        cout << "tocar bombo" << endl;
    }
    else if(typeid(i) == typeid(Guitarra)) {
        // hacer sonar la guitarra
        cout << "tocar guitarra" << endl;
    }
    else if(typeid(i) == typeid(Clarinete)) {
        // hacer sonar el clarinete
        cout << "tocar clarinete" << endl;
    }
    // y otras
}
```

a) Escriba unas líneas de código, como ejemplo de uso de la función `tocar()`, que permitan apreciar su comportamiento polimórfico con diversos tipos de instrumentos. **(0,3 p)**

b) ¿Cree que ésta es la mejor forma de implementar la acción de hacer sonar los diferentes instrumentos? Razone la respuesta. En caso negativo, describa cómo mejorarla, escriba todo el código necesario y modifique el ejemplo anterior para mostrar la diferencia de uso. **(1,2 p)**

Instrucciones de entrega:

- Escriba a mano su respuesta en papel.
- Coloque su DNI o tarjeta universitaria sobre cada página escrita (sin que tape el texto) y capture una imagen nítida.
- Renombre el fichero de cada imagen con su nombre y apellidos y añada un número correlativo de página.
- Suba cada fichero de imagen.

Tiempo máximo 20 minutos

DEL DIRECTOR DE ADÚ

MIGUEL
HERRÁN

SUSANA
ABAITUA

ALEXANDRA
MASANGKAY

Ejercicio 4 septiembre (2 puntos)

Dada la clase paramétrica

```
template <class T1, class T2> class par {  
public:  
    par(const T1& x = T1(), const T2& y = T2())  
        : prime(x), segun(y) {}  
    T1 primero() const { return prime; }  
    T1& primero() { return prime; }  
    T2 segundo() const { return segun; }  
    T2& segundo() { return segun; }  
protected:  
    T1 prime;  
    T2 segun;  
};
```

a) Sobrecargue `operator <<` para la plantilla de clase `par`. (0,5 p)

b) Defina un tipo `racional` como una especialización de `par` para representar números racionales con el signo en el numerador. Sobrecargue el operador suma de números racionales. (0,75 p)

c) Utilizando la clase paramétrica `par` defina una clase `complejo` con partes real e imaginaria de tipo `double`. Sobrecargue el operador de autosuma (`+=`) de números complejos. (0,75 p)

Instrucciones de entrega:

- Escriba a mano su respuesta en papel.
- Coloque su DNI o tarjeta universitaria sobre cada página escrita (sin que tape el texto) y capture una imagen nítida.
- Renombre el fichero de cada imagen con su nombre y apellidos y añada un número correlativo de página.
- Suba cada fichero de imagen.

Tiempo máximo 25 minutos

WUOLAH