

# Prueba Técnica EmasT

Duración total: 2 horas

## Instrucciones Generales

- Lea cuidadosamente todas las instrucciones antes de comenzar.
- Gestione su tiempo adecuadamente. Se proporcionan tiempos sugeridos para cada sección.
- Puede usar el lenguaje de programación de su preferencia para la sección de codificación.
- Si tiene alguna pregunta, levante la mano y un supervisor se acercará a usted.

### *Parte 1: Ejercicios de Lógica de Programación (40 minutos)*

Complete los siguientes ejercicios de programación. Puede utilizar el lenguaje de su elección.

#### **2.1 Secuencia de Fibonacci (10 minutos)**

Implemente una función que genere los primeros n números de la secuencia de Fibonacci.

#### **2.2 Palíndromo (10 minutos)**

Escriba una función que determine si una palabra o frase es un palíndromo (se lee igual de izquierda a derecha y de derecha a izquierda, ignorando espacios y signos de puntuación).

#### **2.3 FizzBuzz (10 minutos)**

Implemente la función FizzBuzz que imprima números del 1 al n, pero para múltiplos de 3 imprima "Fizz", para múltiplos de 5 imprima "Buzz", y para múltiplos de ambos imprima "FizzBuzz".

#### **2.4 Anagramas (10 minutos)**

Escriba una función que determine si dos palabras son anagramas (contienen exactamente las mismas letras, pero en diferente orden).

## Parte 2: Ejercicio de Codificación Principal (70 minutos)

Desarrolle una aplicación simple de gestión de una biblioteca utilizando principios de Programación Orientada a Objetos (POO) y una base de datos relacional.

Requisitos:

1. Cree una clase base ``Item`` y al menos dos clases derivadas (por ejemplo, ``Libro`` y ``Revista``).
2. Implemente una clase ``Biblioteca`` que gestione una colección de ítems.
3. Utilice encapsulación para proteger los datos de los ítems.
4. Implemente al menos un método polimórfico (por ejemplo, ``mostrarInfo()``).
5. Diseñe un esquema de base de datos simple para almacenar los ítems y los préstamos.
6. Implemente métodos para agregar ítems, prestar ítems y devolver ítems.
7. Escriba al menos tres consultas SQL para:
  - Listar todos los ítems disponibles.
  - Encontrar los ítems prestados actualmente.
  - Calcular el ítem más prestado.

Entregables:

- Código fuente de las clases implementadas.
- Esquema de la base de datos (puede ser un diagrama o sentencias CREATE TABLE).
- Consultas SQL solicitadas.

## Parte 3: Diseño de Sistema (20 minutos)

Diseñe un sistema simple de comercio electrónico. Incluya:

1. Un diagrama de clases UML mostrando al menos 5 clases relevantes (por ejemplo, Usuario, Producto, Carrito, Orden, Pago).
2. Una breve descripción de cómo implementaría la funcionalidad de "agregar al carrito".
3. Un esquema básico de la base de datos para soportar este sistema.

## Criterios de Evaluación

Se evaluará:

- Corrección y eficiencia del código.
- Aplicación adecuada de principios de POO.
- Diseño apropiado de la base de datos.
- Claridad y estructura del diseño del sistema.
- Capacidad de razonamiento lógico.

¡Buena suerte!