

Informática
Nivel medio
Prueba 2

Lunes 6 de noviembre de 2017 (mañana)

1 hora

Instrucciones para los alumnos

- No abra esta prueba hasta que se lo autoricen.
- Conteste todas las preguntas de una de las opciones.
- La puntuación máxima para esta prueba de examen es **[45 puntos]**.

Opción	Preguntas
Opción A — Bases de datos	1 – 3
Opción B — Modelos y simulaciones	4 – 6
Opción C — Ciencia de la Web	7 – 9
Opción D — Programación orientada a objetos	10 – 12

Opción A — Bases de datos

1. (a) Defina el término *transacción de bases de datos*. [2]
- (b) Explique la importancia de la durabilidad en una transacción de bases de datos. [2]
- (c) Identifique **dos** tipos diferentes de relaciones en una base de datos. [2]
- (d) Describa la naturaleza del diccionario de datos. [4]
- (e) Identifique **cuatro** responsabilidades de un administrador de bases de datos. [4]

2. Un grupo de alumnos de arte ha realizado tres exámenes en el año académico. Cada uno de estos tres exámenes tiene una puntuación máxima posible de 100. La puntuación total de cada alumno en el año académico es la suma de las puntuaciones que haya obtenido en los tres exámenes.

Los alumnos aprueban, salvo que sus puntuaciones cumplan al menos una de las siguientes condiciones (o ambas):

- Lograr menos de 30 puntos en **cualquiera** de los tres exámenes.
- Lograr una puntuación total que sea inferior a 150.

Considere la siguiente relación que ha creado el profesor de este grupo de alumnos.

TABLA_CLASE

ID_Alumno	Nombre_Alumno	Examen_Uno	Examen_Dos	Examen_Tres	Total
A1209	Ines Prest	30	39	30	99
A1422	Eva Maren	67	29	100	196
C4567	Joe Skrin	100	70	42	212
X0986	Perry Blare	54	68	30	152
B0078	Mia Lorres	26	100	100	226

- (a) Con referencia a la relación TABLA_CLASE, distinga entre datos e información. [3]
- (b) (i) Defina el término *entidad*. [1]
- (ii) Indique la entidad de este ejemplo de relación. [1]
- (c) Identifique un tipo de datos adecuado para ID_Alumno. [1]
- (d) Explique la función de la validación de datos y la verificación de datos. [4]
- (e) Explique cómo podría validarse Total. [2]
- (f) Describa los pasos de una consulta que dé como salida
 - (i) los nombres de todos los alumnos que obtuvieran la puntuación máxima en Examen_Dos. [2]
 - (ii) las ID_Alumno de todos los alumnos que aprobasen. [4]

(La opción A continúa en la página siguiente)

(Opción A: continuación)

3. (a) (i) Indique el significado de datos redundantes en las bases de datos. [1]
- (ii) Explique **un** problema que podrían causar los datos redundantes en una base de datos. [2]

Considere el siguiente tipo de relación. Contiene datos de un número de profesores y alumnos de diferentes colegios que realizan labores de voluntariado para ayudar a la comunidad local en determinados días.

TABLA_VOLUNTARIOS_COLEGIOS

(Nombre_Colegio, Codigo, Direccion, Fecha, Num_Voluntarios)

Nombre_Colegio	<u>Codigo</u>	Direccion	<u>Fecha</u>	Num_Voluntarios
Riverbank HS	AB1234	Goldmar Dr 15	10/07/2017	32
East High	EE1324	East Street 20	10/07/2017	15
East High	EE1324	East Street 20	20/10/2017	19
Oceanview HS	BG3445	Long Road 123	10/07/2017	14
Oceanview HS	BG3445	Long Road 123	20/10/2017	26
Blue Sky HS	SB9008	West Street 19	20/10/2017	37

Los atributos clave están subrayados.

- (b) Identifique **tres** características de la primera forma normal (1FN) que sean evidentes en esta relación. [3]
- (c) Explique por qué se utiliza una clave compuesta para la relación TABLA_VOLUNTARIOS_COLEGIOS. [2]

A continuación se muestra la relación normalizada TABLA_VOLUNTARIOS_COLEGIOS:

TABLA_COLEGIOS
Codigo, Nombre_Colegio, Direccion

TABLA_VOLUNTARIOS
Codigo, Fecha, Num_Voluntarios

- (d) Discuta si estas relaciones están en la tercera forma normal (3FN). [5]

Fin de la opción A

Opción B — Modelos y simulaciones

4. (a) Distinga entre un modelo informático y una simulación por computadora. [4]
- (b) Identifique **dos** razones por las cuales algunos sistemas son difíciles de modelar adecuadamente. [2]

Las simulaciones informáticas se utilizan a menudo en situaciones en las que, por alguna razón, no es posible realizar experimentos prácticos. Una de estas razones puede ser una cuestión ética.

- (c) (i) Con una clara referencia a la cuestión ética, describa **un** ejemplo en el que la experimentación práctica no sea posible debido a razones éticas. [3]
- (ii) Aparte de las razones éticas, indique otras **tres** ventajas de crear un modelo informático como simulación en lugar de construir un modelo físico. [3]

5. Una empresa fabricante que crea varios productos va a utilizar un software de hojas de cálculo para modelar sus finanzas. Esto incluye cálculos que estimarán diferentes cantidades, entre ellas los beneficios que tendrá la empresa en futuros años.

El modelo implica el uso de un software de hojas de cálculo que se organizará empleando distintas hojas para distintas áreas de las finanzas de la empresa. Anteriormente se utilizaban métodos menos sofisticados para realizar un seguimiento de los costos y las ventas.

- (a) Incluyendo ejemplos donde corresponda, describa una estructura básica para este modelo. [6]
- (b) Sugiera cómo podría comprobarse la confiabilidad del modelo. [2]

La empresa ha establecido determinados objetivos de beneficios que desea alcanzar en los próximos tres años.

- (c) Explique cómo puede utilizarse este modelo para investigar distintas estrategias que permitan lograr dichos objetivos. [4]

(La opción B continúa en la página siguiente)

(Opción B: continuación)

6. Los meteorólogos utilizan modelo informático que son capaces de simular futuros patrones meteorológicos. Al principio, estas predicciones estaban limitadas al futuro cercano. Sin embargo, ahora los sistemas modernos permiten producir predicciones a largo plazo.
- (a) Sugiera **dos** razones por las cuales estas simulaciones han mejorado tanto en precisión como en alcance a futuro. [4]
 - (b) Sugiera por qué las predicciones son menos precisas cuanto más a futuro se realizan. [4]
 - (c) Discuta si los datos históricos permiten predecir con precisión las condiciones meteorológicas futuras. [5]
- La simulación de los modelos de predicción meteorológica genera datos específicos a los que se puede dar salida de diversas formas.
- (d) (i) Defina el término *visualización*. [2]
 - (ii) Con la ayuda de ejemplos, discuta cómo el desarrollo de la manera en que se visualizan los datos ha hecho que los resultados de estas simulaciones sean más accesibles para la población general. [6]

Fin de la opción B

Opción C — Ciencia de la Web

7. *BuildYourWebSite* es una empresa en línea que proporciona varias plantillas comunes para que sus usuarios construyan su propio sitio web. Cada plantilla incluye un archivo HTML, un archivo CSS, una carpeta de imágenes web, y una carpeta de efectos sonoros especiales.

- (a) Identifique **dos** características del HTML. [2]
- (b) Discuta las ventajas y desventajas de que la plantilla incluya un archivo CSS además de la página HTML. [6]

Cada plantilla se puede descargar mediante un navegador web como un único archivo comprimido.

- (c) Explique por qué TCP/IP es un protocolo confiable en relación con las operaciones de descarga. [3]
- (d) Evalúe la compresión con pérdidas y la compresión sin pérdidas cuando se utilizan para descargar archivos. [4]

8. Una comunidad de desarrolladores ha hecho disponible un código abierto que se actualiza frecuentemente. El código puede descargarse de manera gratuita, pero los usuarios deben inscribirse en el sitio web, y su acceso se debe autenticar.

- (a) Evalúe el uso de scripts de servidor para proporcionar el mecanismo de inscripción. [4]

La autenticación se basa en que el usuario acceda mediante una cuenta en una tercera parte establecida; por ejemplo, una cuenta existente de correo electrónico o en una red social. Dicha tercera parte verifica al usuario, y le da acceso al sitio web que contiene el código abierto.

- (b) Explique cómo puede mantenerse la privacidad del usuario mientras se utiliza este método de autenticación. [4]

El URL de este sitio web es www.OpenSourceDev.org. Todo nuevo código que los desarrolladores hacen disponible pasa a formar parte de los recursos del sitio web. Un script genera semanalmente notificaciones automáticas del nuevo código que hay disponible en el sitio, y envía dicha notificación a los usuarios por correo electrónico (email).

- (c) (i) Resuma, con un ejemplo, cómo se generará el URL del nuevo código. [2]
- (ii) Resuma los pasos que podría realizar el script para enviar estas notificaciones. [3]

(La opción C continúa en la página siguiente)

(Continuación: opción C, pregunta 8)

A continuación se muestran un fragmento de un script y un formulario web.
En el script no se implementan algunas funciones, y solo se proporciona su especificación.

```

<?php
  //incluir una base de datos de direcciones URL
  include('url_db.php');
  $url = $short = "";

  if ($_SERVER["REQUEST_METHOD"] == "POST")
  { $url = $_POST['url'];
    $short = make_short($url);
    function make_short($u)
    { $x = make_alpha_string($u);
      $y = first4_last4($x);
      $z = limits($x);
      $v = $y. ".".$z; //concatenación de cadenas
      return $v;
    }

    function make_alpha_string($u)
    { // Esto elimina, en este orden, las subcadenas que corresponden
      // a nombres de protocolos, la subcadena www, y todos los caracteres
      // excepto las letras.
    }

    function first4_last4($u)
    { // Retorna la cadena compuesta de los cuatro primeros caracteres
      // seguida por los últimos cuatro caractres de $u
    }

    function limits($u)
    { // Retorna la cadena compuesta del primer y
      // el último carácter de $u
    }

    mysql_query("
      INSERT INTO url_db(orig_url, short_url, url_ip) VALUES
      ( '".$_POST['url']."',
        '$short.',
        '".$_SERVER['REMOTE_ADDR']."'
      )
    ");
  }
?>

<form method="post" action="">
URL:
<input type="text" name="url" />
<br><br>
<input type="submit" name="Submit" value="Submit" />
</form>

```

- (d) Describa el procesamiento que sucede cuando en el formulario se introduce el URL <https://www.the2nd.org/bin.php?id=70> que el servidor descubre que está asociado con la dirección IP 172.16.254.1.

[3]

(La opción C continúa en la página 9)

Página en blanco

(Opción C: continuación)

9. Un museo tiene un catálogo en línea de sus piezas. No se han adquirido piezas nuevas en los últimos 20 años, y el museo ha corrido riesgo de cierre debido a la falta de financiación. Su sitio web consiste solamente en páginas web estáticas.

Se ha indicado que la falta de éxito del museo puede estar relacionada, en parte, con sus páginas web estáticas.

- (a) Identifique **tres** diferencias entre una página web estática y una página web dinámica. [3]
- (b) Sugiera **dos** servicios, y las ventajas que aportarían, si el museo rediseñara su sitio web para hacerlo dinámico. [4]
- (c) Explique cómo la prestación de servicios en el sitio web del museo puede hacer que este ascienda en su clasificación en los motores de búsqueda. [4]

También se sugiere que el catálogo básico del museo se revise y se actualice mediante la asignación de etiquetas meta a todas las piezas del museo.

- (d) Sugiera cómo el uso de etiquetas meta puede ser pertinente para la optimización para los motores de búsqueda. [3]

Fin de la opción C

Opción D — Programación orientada a objetos

Un restaurante utiliza un programa orientado a objetos para gestionar el costo de la comida y la bebida que consumen sus clientes. Cada vez que se ocupa una mesa, se crea una instancia de un objeto `Payment` que contendrá detalles de los productos que se pidan. A medida que se pide cada producto, se agrega un objeto `FoodItem` o `DrinkItem` al objeto `Payment` según corresponda.

```
public class Payment
{
    private FoodItem[] fi = new FoodItem[100];
    private int fiCount;
    private static double foodTax = 0.2; // 20% de impuestos se agrega a
                                        // todos los precios de comidas
    private DrinkItem[] di = new DrinkItem[100];
    private int diCount;
    private static double drinkTax = 0.1; // 10% de impuestos se agrega a
                                        // todos los precios de bebidas

    public Payment()
    {
        fiCount = 0;
        diCount = 0;
    }

    public DrinkItem getDi(int x)
    {
        return di[x];
    }

    // están incluidos todos los otros métodos accesor y mutador

    // addFoodItem() - este método añade un nuevo objeto FoodItem
    // addDrinkItem() - este método añade un nuevo objeto DrinkItem

    public static double findPrice(Item[] pl, String c)
    { //código omitido }

    // calculateBill() - Este método devuelve la cuenta (el valor total de
    // los artículos consumidos por una mesa en particular)
}

public class FoodItem
{
    private String itemCode;
    private int quantity;

    public FoodItem(String x, int y)
    {
        itemCode = x;
        quantity = y;
    }

    // están incluidos todos los métodos accesor y mutador
}
```

La clase `DrinkItem` se define de manera parecida.

(La opción D continúa en la página siguiente)

(Opción D: continuación)

10. Cuando se crea una instancia de un objeto `Payment`, las variables `fiCount` y `diCount` se inicializan a 0 mediante el código que se encuentra en el constructor.

- (a) Resuma un método alternativo de inicializar estas variables que no requiera el uso del código que se encuentra en el constructor. [2]
- (b) Indique la implicación del uso del término `static` en la clase `Payment`. [2]
- (c) Con referencia a **dos** ejemplos de las clases que figuran en la página 10, explique las ventajas que aporta el uso de distintos tipos de datos. [4]
- (d) Describa el propósito de la siguiente instrucción:

```
private FoodItem[] fi = new FoodItem[100]; [3]
```

En la clase `Payment`, al método `addFoodItem()` se le pasa un objeto `FoodItem` como parámetro.

- (e) Elabore el método `addFoodItem()`. [3]

(La opción D continúa en la página siguiente)

(Opción D: continuación)

11. La variable global `tables` se declara de la siguiente forma:

```
Payment[] tables = new Payment[50];
```

Los índices de esta matriz representan el número de mesa, de modo que `tables[1]` es un objeto de `Payment` para los clientes que ocupan la mesa número 1.

La clase principal contiene el siguiente código. **Nota:** Puede suponer que todos los métodos accesos y mutador correspondientes se han incluido en sus respectivas clases.

```
tables[1] = new Payment();
tables[2] = new Payment();
FoodItem a = new FoodItem("f102", 2);
FoodItem b = new FoodItem("f100", 1);
DrinkItem c = new DrinkItem("d102", 3);
tables[1].addFoodItem(a);
tables[1].addFoodItem(b);
tables[2].addDrinkItem(c);
tables[2].addDrinkItem(new DrinkItem("d103", 1));
System.out.println(tables[1].getFiCount());
System.out.println(Payment.getFoodTax());
System.out.println(tables[2].getDi(1).getItemCode());
```

- (a) Indique la salida después de que se ejecute este código. [3]
- (b) Elabore instrucciones, en código, que impriman lo siguiente:
 - (i) El número de bebidas que ha pedido la mesa 40. [1]
 - (ii) El código de producto de la tercera comida que ha pedido la mesa 2. [1]

(La opción D continúa en la página siguiente)

(Continuación: opción D, pregunta 11)

El precio de cada producto se almacena en un objeto de la clase `Item`.
La clase se resume a continuación:

```
public class Item
{
    private String code;    // código de producto
    private String name;   // nombre de producto
    private double price;  // precio por unidad antes de impuestos
    // se incluyen todos los métodos accesor, mutador y constructor
}
```

Todos los objetos de esta clase se guardan en la matriz global `pl` de acuerdo con la siguiente declaración: `Item[] pl = new Item[200];`

Nota: El número de objetos que se encuentran en esta matriz cambiará de semana en semana.

El método `findPrice(Item[] pl, String c)` de la clase `Payment` busca y devuelve el precio del producto con el código `c`.

- (c) Elabore el método `findPrice()`. Puede suponer que el producto existe en la matriz. [6]

Cuando un cliente quiere pagar la cuenta, se llama al método `calculateBill()`. Si la cuenta fuera para la mesa 10, entonces se realizaría la siguiente llamada:

```
double finalBill = tables[10].calculateBill(Item[] pl);
```

- (d) Elabore el método `calculateBill()`. Deberá utilizar cualquiera de los métodos definidos anteriormente. [7]

12. (a) Elabore un diagrama que muestre las relaciones entre las clases `Payment`, `FoodItem`, `DrinkItem`, e `Item` classes. No hay necesidad de incluir los nombres de atributos o métodos. [3]

- (b) Haciendo referencia cualquiera de las clases anteriores, describa **dos** ventajas que podría obtener un equipo de programación al utilizar la encapsulación. [6]

La empresa propietaria de este restaurante también posee hoteles, tiendas y un negocio de alquiler de coches. Los programas que gestionan las finanzas de estos negocios tienen clases parecidas a las que se han mostrado en este cuestionario de examen.

- (c) Explique cómo podría utilizarse la herencia tanto para mejorar la claridad del diseño como para reducir la cantidad de código que debe escribirse. [4]

Fin de la opción D
