



88107020



**INFORMÁTICA**  
**NIVEL SUPERIOR**  
**PRUEBA 2**

Miércoles 17 de noviembre de 2010 (mañana)

2 horas 15 minutos

---

**INSTRUCCIONES PARA LOS ALUMNOS**

- No abra esta prueba hasta que se lo autoricen.
- Responda a todas las preguntas.

Responda a **todas** las preguntas.

1. (a) En el contexto de las *estructuras de datos*, explique qué se entiende por

(i) *cola*; [2 puntos]

(ii) *pila*. [2 puntos]

(b) Indique **una** aplicación informática en que resulte adecuada una estructura de datos de tipo cola. [1 punto]

Considere la siguiente clase.

```
class Nodo
{
    public int elemento;
    public Nodo siguiente;

    public Nodo(int d)
    {
        elemento = d;
        siguiente = null;
    }

    public void mostrarNodo()
    {
        output(elemento + " ");
    }
}
```

(c) La instrucción `Nodo x = new Nodo(5);` crea un objeto de tipo `Nodo`. Indique la salida producida por la llamada `x.mostrarNodo();`. [1 punto]

*(Esta pregunta continúa en la siguiente página)*

(Pregunta 1: continuación)

Examine la siguiente implementación de una cola mediante una lista enlazada.

```
class MiCola
{ private Nodo primero;
  private Nodo ultimo;
  public MiCola() { primero = null; ultimo = null; }
  public boolean estaVacio() { return primero == null; }

  public void ponerEnCola(int x)
  { Nodo nuevoNodo = new Nodo(x);
    if (estaVacio())
      { primero = nuevoNodo; }
    else
      { ultimo.siguiete = nuevoNodo; }
    ultimo = nuevoNodo;
  }

  public int quitarDeLaCola()
  {
    // Falta el código que elimina un nodo de la cola
  }

  public void mostrarCola()
  { if (primero == null)
    { output(";La cola está vacía!"); }
    else
    { Nodo temp = primero;
      while (temp != null)
      { temp.mostrarNodo();
        temp = temp.siguiete;
      }
    }
  }
}
```

(d) La instrucción `MiCola x = new MiCola();` crea una cola vacía.

(i) Indique la salida generada después de la ejecución de la siguiente instrucción.

```
x.mostrarCola();
```

[1 punto]

(ii) Construya el código para el método `quitarDeLaCola()`. El método debe eliminar un elemento de `x` y devolver el valor del elemento eliminado.

[4 puntos]

(iii) Indique la salida generada después de la ejecución de las siguientes instrucciones.

```
x.ponerEnCola(2);
x.ponerEnCola(4);
int y = x.quitarDeLaCola();
output("Elemento eliminado: " + y);
x.ponerEnCola(1);
x.ponerEnCola(7);
output("Elementos en la cola: ");
x.mostrarCola();
```

[3 puntos]

(e) Explique cómo se puede invertir el orden de los elementos de una cola que no está vacía usando una pila.

[6 puntos]

2. Una empresa de automóviles vende cinco modelos diferentes de coches y da empleo a cuatro vendedores. Un registro de las ventas de cada mes se puede representar mediante una tabla. La primera fila de la tabla contiene el número de ventas de cada modelo conseguidas por el vendedor 1; la segunda fila contiene el número de ventas de cada modelo conseguidas por el vendedor 2, y así sucesivamente.

	Modelo 1	Modelo 2	Modelo 3	Modelo 4	Modelo 5
Vendedor 1	12	0	0	5	6
Vendedor 2	11	1	3	1	3
Vendedor 3	10	11	5	3	0
Vendedor 4	9	8	5	4	5

- (a) (i) Calcule las ventas totales del vendedor 2. [1 punto]
- (ii) Calcule las ventas totales del modelo 3. [1 punto]

Los datos de ventas listados más arriba se introducen en la matriz bidimensional `Ventas`, declarada como `int[][] Ventas = new int[4][5]`; que se puede representar lógicamente como se muestra a continuación.

Ventas		[0]	[1]	[2]	[3]	[4]
	[0]	12	0	0	5	6
	[1]	11	1	3	1	3
	[2]	10	11	5	3	0
	[3]	9	8	5	4	5

Examine el código siguiente.

```
public void misterio(int[][] Ventas)
{
    for (int n = 0; n < 4; n = n + 1)
    {
        int total = 0;
        for (int m = 0; m < 5; m = m + 1)
        {
            total = total + Ventas[n][m];
        }
        output("Las ventas totales para el vendedor "
            + (n + 1) + " ascienden a " + total);
    }
}
```

- (b) Indique la salida de la llamada al método `misterio(Ventas)`. [4 puntos]

*(Esta pregunta continúa en la siguiente página)*

*(Pregunta 2: continuación)*

- (c) Construya el método que imprima el número total de ventas para cada uno de los cinco modelos de coche. *[6 puntos]*

Asuma que los precios de los coches se almacenan en la siguiente matriz unidimensional.

```
double[] PrecioModelo = {10288.00, 12999.99, 18456.00, 20345.00, 45799.00}
```

Por ejemplo:

`PrecioModelo[1]` contiene el valor 12999.99.

Esto significa que el precio del Modelo 2 es 12999,99.

- (d) Construya un método que reciba como parámetro la matriz unidimensional `PrecioModelo` y la matriz bidimensional `Ventas`. El método deberá determinar el mejor vendedor (aquél que ha conseguido el mayor monto vendido total) y dar como salida el mejor vendedor y el mayor monto vendido total. Asuma que al determinar el mejor vendedor no habrá empate. *[8 puntos]*

- 3. (a) (i) Defina el término *fichero*. [2 puntos]
- (ii) Identifique **dos** razones por las que es adecuado usar ficheros en aplicaciones que requieran grandes conjuntos de datos. [2 puntos]
- (b) Normalmente, un fichero está compuesto por una cantidad de *registros*. Cada registro está formado por *campos*.
  - (i) Defina el término *campo clave*. [2 puntos]
  - (ii) En relación con los *ficheros totalmente indexados*, explique la diferencia entre la *organización lógica* y la *organización física del fichero*. [4 puntos]
- (c) Una pequeña tienda usa un computador para mantener un control diario de sus niveles de existencias. La información relacionada con las ventas de productos se registra en un *fichero secuencial de transacciones* llamado FicheroActualización.

Al final del día el fichero maestro FicheroExistencias se actualiza con la información del fichero de transacciones FicheroActualización y se almacena en un nuevo fichero maestro llamado NuevoFicheroExistencias. Cada fichero se ordena para que el campo del número de elemento esté ordenado en forma ascendente.

Por ejemplo, si los ficheros FicheroExistencias y FicheroActualización contienen los datos siguientes antes de la actualización:

FicheroExistencias				FicheroActualización	
núm. elemento	descripción	cantidad	...otros campos...	núm. elemento	cantidad
1100	Choc/blanco	52		1100	14
1200	Infusión de hierbas	35		1206	4
1206	Té verde	23			

Después de la actualización, el contenido del nuevo fichero maestro NuevoFicheroExistencias será:

NuevoFicheroExistencias			
núm. elemento	descripción	cantidad	...otros campos...
1100	Choc/blanco	<b>38</b>	
1200	Infusión de hierbas	35	
1206	Té verde	<b>19</b>	

- (i) Esboce por qué los ficheros FicheroExistencias y FicheroActualización se deberían ordenar siguiendo el mismo criterio. [2 puntos]
- (ii) Esboce el algoritmo para actualizar los registros de FicheroExistencias con la información de FicheroActualización para crear NuevoFicheroExistencias. [8 puntos]

4. Para esta pregunta se requiere el estudio de un caso.

- (a) La Terminal 5 tiene varias áreas que requieren que el acceso sea autorizado. La dirección del aeropuerto está considerando la posibilidad de introducir distintos métodos de seguridad en todo el aeropuerto.

El personal del aeropuerto puede usar *tarjetas inteligentes*, *tarjetas con banda magnética* o *códigos de barras* como medios de acceder a cualquier área segura dentro del aeropuerto. Las tarjetas inteligentes no contienen información biométrica.

- (i) Esboce por qué el uso de tarjetas inteligentes es una medida de seguridad eficaz. [2 puntos]

Para mejorar la seguridad las tarjetas inteligentes pueden asociarse con sistemas de identificación biométrica.

- (ii) Indique **dos** tipos de sistemas de identificación biométrica. [2 puntos]

- (iii) Describa cómo funcionan los sistemas de identificación biométrica. [4 puntos]

- (iv) Compare los distintos niveles de seguridad que se pueden aplicar al personal y los pasajeros. [4 puntos]

- (b) El **sistema operativo** controla el funcionamiento de cada uno de los sistemas informáticos del aeropuerto. Los sistemas operativos pueden ser monousuario o **multiusuario** y permiten la **multitarea**.

- (i) Defina el término *sistema operativo*. [1 punto]

- (ii) Defina el término *multiusuario*. [1 punto]

- (iii) Defina el término *multitarea*. [1 punto]

- (iv) Identifique **tres** razones para usar la multitarea en el control del tráfico aéreo. [3 puntos]

- (v) Explique cómo se usa un sistema multiusuario en el sistema de facturación. [2 puntos]

(Esta pregunta continúa en la siguiente página)

(Pregunta 4: continuación)

- (c) Esboce las principales características de los siguientes modos de funcionamiento.
    - (i) *procesamiento por lotes* [2 puntos]
    - (ii) *procesamiento en tiempo real* [2 puntos]
    - (iii) *procesamiento en línea* [2 puntos]
  
  - (d) Identifique un método adecuado de procesar datos para
    - (i) el control del tráfico aéreo; [1 punto]
    - (ii) el sistema de facturación. [1 punto]
  
  - (e) Los sistemas informáticos de un aeropuerto moderno se conectan a través de una *red*.
    - (i) En relación con los sistemas del aeropuerto, esboce el significado de los términos *cliente* y *servidor*. [4 puntos]
    - (ii) Indique **un** dispositivo de hardware necesario para la interconexión de una red y esboce su función. [2 puntos]
    - (iii) Explique la importancia de los *protocolos de red* en los sistemas del aeropuerto. [2 puntos]
  
  - (f) Explique cómo una lista enlazada podría ser una estructura de datos adecuada para el sistema del control del tráfico aéreo. [4 puntos]
-