



88107022



INFORMÁTICA
NIVEL MEDIO
PRUEBA 2

Miércoles 17 de noviembre de 2010 (mañana)

1 hora 30 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. Se ha escrito un programa para procesar los datos de los estudiantes en un colegio. Para cada estudiante, se almacenan su nombre y su fecha de nacimiento. La clase `Estudiante` se ha diseñado para almacenar estos detalles.

```
class Estudiante
{
    private String nombre;
    private int anoDeNacimiento; // Año en que nació el estudiante
    private int mesDeNacimiento; // 1-12 corresponde a Ene-Dic
    private int diaDeNacimiento; // Día 1-31 del mes

    public Estudiante()
    {
        // constructor
    }

    public String obtenerNombre()
    {
        return nombre;
    }

    // otros métodos
    // ...
}
```

- (a) En relación con la clase `Estudiante`, describa las diferencias entre los identificadores *public* y *private*. [4 puntos]
- (b) Construya un método `obtenerAnoDeNacimiento()` que devuelva el año en que nació el estudiante. [2 puntos]

(Esta pregunta continúa en la siguiente página)

(Pregunta 1: continuación)

El objeto `Estudiante` sólo almacena la fecha de nacimiento del estudiante y, por tanto, su edad debe calcularse en función de la fecha actual.

- (c) (i) Asumiendo que la fecha actual es 17 de noviembre de 2010, calcule la edad, en años, de cada estudiante que aparece en la tabla siguiente. [2 puntos]

Estudiante	Año de nacimiento	Mes de nacimiento	Día de nacimiento
Estudiante A	2000	08	15
Estudiante B	2000	11	16
Estudiante C	2000	11	18

- (ii) Explique cómo se puede determinar si ha pasado la fecha del cumpleaños de un estudiante en el año actual. [2 puntos]

- (iii) Las variables de tipo `int` `anoActual`, `mesActual` y `diaActual` almacenan el año, el mes y el día actuales, respectivamente. Usando estas variables, construya un método `obtenerEdad()` que devuelva la edad de un estudiante en años. [5 puntos]

- (d) Los datos que el colegio tiene del estudiante se almacenan en un fichero. El método `obtenerSiguiente()` lee el fichero y devuelve un objeto `Estudiante` cada vez que es llamado. Cuando se llega al final del fichero, `obtenerSiguiente()` devuelve un objeto `Estudiante` que contiene el nombre "XXX".

Construya un fragmento de programa que imprima una lista con los nombres de todos los estudiantes que tengan 12 años. Debe usar el método `obtenerEdad()`. [5 puntos]

2. Un programa contiene una matriz de tipo `String` con nombres de estudiantes y una matriz de tipo `float` que almacena los promedios de las notas de cada uno.

En cada matriz hay `int numEstudiantes` elementos.

Por ejemplo:

	0	1	2	3	4	...
nombreEstudiantes	Mary	John	Paul	Luke	Trina	
	0	1	2	3	4	...
promedioNotas	78	62	71	89	52	

El programa tiene que ordenar los estudiantes en función de los promedios de sus notas.

- (a) (i) Esboce las ventajas de almacenar los nombres de los estudiantes y los promedios de notas en matrices. [2 puntos]
- (ii) Construya el método `mejorEstudiante()`, iniciado más abajo, que devuelve el nombre del estudiante con la nota promedio más alta.

```
String mejorEstudiante(String[] nombres, float[] promedios,
                       int numEstudiantes)
{
    // Insertar código aquí
}
```

[4 puntos]

- (b) (i) Explique por qué la matriz `promedioNotas` no se debería ordenar en forma descendente sin cambiar también la matriz `nombreEstudiantes`. [2 puntos]
- (ii) Construya un método `rankingEstudiantes()`, iniciado más abajo, que utilice el algoritmo de la burbuja para ordenar las dos matrices en orden descendente en función de la nota media.

```
void rankingEstudiantes(String[] nombres, float[] promedios,
                        int numEstudiantes)
{
    // Insertar código aquí
}
```

[8 puntos]

- (c) Explique cómo es posible evitar la dificultad de mantener alineados los elementos correspondientes de las dos matrices usando *objetos*. [4 puntos]

3. 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 pueden permitir 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 3: 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) Los sistemas informáticos de un aeropuerto moderno se conectan a través de una *red*. En relación con los sistemas del aeropuerto, esboce el significado de los términos *cliente* y *servidor*. [4 puntos]
-