



22067022

INFORMÁTICA
NIVEL MEDIO
PRUEBA 2

Miércoles 3 de mayo de 2006 (mañana)

1 hora y 30 minutos

INSTRUCCIONES PARA LOS ALUMNOS

- No abra el examen hasta que se le indique.
- Responda a todas las preguntas.

1. La matriz de cadenas de caracteres `nombres` se ha declarado tal como se muestra a continuación:

```
String[ ] nombres = new String [100];
```

La variable global de tipo entero `entradas` almacena el número actual de nombres en la matriz.

Por ahora, la matriz sólo contiene los seis elementos siguientes:

	0	1	2	3	4	5
nombres	sara	surita	juan	alvaro	nicole	peter

p.ej. `nombres[0] = sara`, `entradas = 6`

El método `borrar()`, escrito parcialmente más abajo, borra el elemento de la matriz cuyo índice es `posición` y mueve todos los elementos con índice mayor que `posición`, una posición a la izquierda.

p.ej. si `posición = 2`, el elemento `juan` se eliminará de la matriz de nombres, y `nombres[2]` pasaría a contener `alvaro`, `nombres[3]` contendría `nicole` y así sucesivamente.

```
// los datos cuyo índice sea 'posición' se borrarán de la matriz de nombres
public void borrar(String[ ] nombres, int posición) {

    int i= posición + 1;
    while (i < entradas) {
        nombres [i - 1] = nombres [i];
        i = i + 1;
    }
    entradas = entradas - 1;
}
```

- (a) Explique qué nos dice acerca del método `borrar()` la información en la firma del método. [2 puntos]
- (b) Explique qué se entiende por ámbito de una variable, haciendo referencia a la variable `i` del método `borrar()`. [3 puntos]
- (c) Complete la tabla rastreo para el algoritmo en función de la llamada al método `borrar(nombres, 3)`. [3 puntos]

posición	i	nombres[0]	nombres[1]	nombres[2]	nombres[3]	nombres[4]	nombres[5]
3							

(Esta pregunta continúa en la siguiente página)

(Pregunta 1: continuación)

- (d) Esboce cómo el uso de la variable `entradas` puede evitar la visualización o la impresión de nombres duplicados de esta matriz. *[2 puntos]*

Un teléfono móvil usa las **dos** matrices paralelas `nombres` y `números` para almacenar un máximo de 100 entradas en su agenda.

p.ej. la primera entrada se podría almacenar como:
`nombres[0] = 'sara', números[0] = '68594753'`

Si la agenda no está completa, las posiciones vacías se ubican al final de las matrices.

- (e) Construya el método `agregarEntrada()` que añada una nueva entrada (nombre y número) a la agenda, si aún no está llena. Asuma que la variable global `entradas` almacena el número actual de entradas de la agenda. *[4 puntos]*

Cuando el teléfono móvil recibe una llamada, se llama al método `buscarNombre()`, que busca en la agenda el número de teléfono de la persona que llama. Si lo encuentra, el método devuelve el nombre de dicha persona. Si no se encuentra el número, se devuelve un mensaje adecuado.

- (f) Construya el método `buscarNombre()`. *[6 puntos]*

2. Una compañía de ingeniería está diseñando un nuevo sistema de señalización para una sección de la línea de ferrocarriles.

El equipo de análisis está usando una aproximación mediante prototipos como parte del proceso de diseño.

- (a) Explique la ventaja de usar esta aproximación para
 - (i) los usuarios finales (la compañía de ferrocarriles). [2 puntos]
 - (ii) el equipo de análisis de sistemas. [2 puntos]

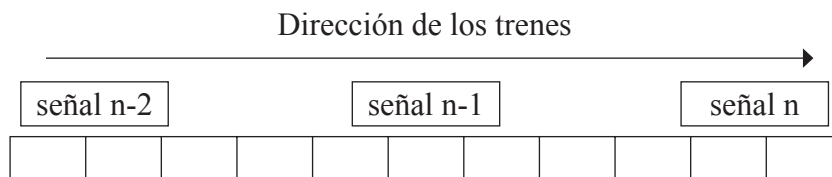
La implementación incluirá la instalación de una gran pantalla en el centro de control de señalización, en la que se podrá visualizar la red de ferrocarriles junto con la posición de todos los trenes y el estado de todas las señales. El centro también actualiza una pantalla similar en la oficina del jefe de cada estación. El sistema incluye programas instalados centralmente que controlan y establecen las señales en las vías.

(b) Describa **una** forma en que los sensores pueden captar la posición de cada tren. [2 puntos]

Los datos tomados por los sensores se almacenan en una cinta magnética, para que pueda volver a ejecutarse si es necesario.

- (c) Explicar qué tipo de procesamiento se necesita
 - (i) para controlar la señalización. [2 puntos]
 - (ii) para el almacenamiento y recuperación de los datos de los sensores. [2 puntos]

Considere el tramo de vías que se muestra a continuación. Los trenes se mueven sólo en la dirección que se muestra y las señales están numeradas por orden, empezando por el 0.



El algoritmo que controla el estado de las señales necesita dos estructuras de datos:

Una registra cada señal, n, en el caso de que el tren haya pasado dicha señal pero no haya pasado aún la siguiente. (i.e. se encuentra entre la primera señal y la siguiente).

La otra registra para cada señal el estado de la misma: **0** para señal en rojo, **1** para amarillo, y **2** para verde.

(Esta pregunta continúa en la siguiente página)

(Pregunta 2: continuación)

- (d) Indique **dos** estructuras de datos adecuadas. *[2 puntos]*

Cuando un tren pase la señal n , dicha señal cambiará a rojo, la señal anterior ($n-1$) a amarillo y la señal ($n-2$) a verde.

- (e) Complete el fragmento de algoritmo que se inicia más abajo, que controla si un tren ha pasado cada una de las señales, y establece las señales según corresponda. *[4 puntos]*

```
// totalSeñales representa el número de señales en esta
// sección de la vía
for (int n = 0; n < totalSeñales; n++) {
    // faltan una o más líneas
}
```

- (f) Discuta posibles estrategias de copia de seguridad que podrían mantener activo total o parcialmente el sistema ferroviario, en caso de que se produjera un fallo de los sistemas en el centro de control. *[4 puntos]*

Para esta pregunta se requiere el estudio de un caso.

- 3. (a) Explique la diferencia fundamental entre la forma en que el formato MIDI almacena los datos en comparación con los archivos de música almacenados en CD y cintas de cassette. [2 puntos]

En un piano, la nota DO central se mantiene pulsada 5 segundos y posteriormente se suelta.

- (b) Compare los requisitos de almacenamiento de un archivo MIDI con los de un archivo de música almacenado en un CD, haciendo referencia al ejemplo dado (no se requieren valores de almacenamiento reales). [3 puntos]

Una banda recién creada desea hacer una grabación promocional de su música. Sin embargo, como aún no son unos músicos expertos, encuentran dificultades al ejecutar algunas piezas.

- (c) Explique **dos** formas en que un secuenciador informático podría ayudar a mejorar la calidad de su música grabada. [4 puntos]
- (d) La instrucción de voz del canal MIDI más común es **NOTE ON**, que está formada por un byte de estado seguida de dos bytes de datos.

El formato general de una instrucción **NOTE ON** es el siguiente:

Byte de estado	Byte de datos #1 (tono/nota)	Byte de datos #2 (volumen)
1001nnnn	0xxxxxxx	0xxxxxxx

donde (nnnn + 1) representa el número de canal y xxxxxxxx representa un valor decimal entre 0 y 127.

Por ejemplo: 10011010 00111100 00000001

representa **NOTE ON** en el canal 11 de la nota #60 (DO CENTRAL) con volumen 1 (el volumen más bajo posible).

- (i) Indique qué número de canal selecciona el byte de estado 10010011. [1 punto]
- (ii) Describa cómo el dispositivo receptor (p.ej. un módulo de sonido) distingue entre un byte de estado y un byte de datos. [2 puntos]
- (iii) Indique el número de notas diferentes que se pueden reproducir usando este formato. [1 punto]
- (iv) Identifique los bytes que corresponderían a la instrucción siguiente: **NOTE ON** usando el canal 12, pulsando DO central al máximo volumen. [2 puntos]

(Esta pregunta continúa en la siguiente página)

(Pregunta 3: continuación)

(e) Los bytes de las instrucciones suelen escribirse en hexadecimal. Como en el ejemplo, el byte 10011010 (binario), podría escribirse como 9A (hex), donde cada dígito hexadecimal sustituye a un grupo de cuatro bits binarios.

(i) Indique por qué las instrucciones se suelen escribir en hexadecimal. *[1 punto]*

(ii) El ejemplo de **NOTE ON** anterior de la parte (d) era:

10011010 00111100 00000001

Teniendo en cuenta que el byte de estado para **NOTE OFF** se representa mediante 1000nnnn, identifique, mediante notación hexadecimal, la instrucción que apagará esta nota. *[3 puntos]*

(f) En las aplicaciones cotidianas se encuentran archivos MIDI de pequeño tamaño, ajenos a su uso original en el estudio de grabación. Esboce **dos** de estas aplicaciones. *[4 puntos]*

(g) Explique **dos** características de los módulos de sonido de MIDI que permiten a los músicos crear sonidos más completos y más variados. *[4 puntos]*

(h) Discuta si el canto se podría manejar eficazmente usando la tecnología MIDI. *[3 puntos]*
