

Examen de Fundamentos de Computadores y Lenguajes

Examen Parcial. Febrero 2005

Primera parte: Cuestiones (5 cuestiones, 5 puntos en total)

- 1) Escribir una clase para almacenar los datos de inventario de un producto en un almacén. La clase tendrá los siguientes atributos *públicos*, así como un constructor que servirá para dar valor inicial a cada atributo, usando los datos que se le pasen como parámetros:
 - identificador (entero)
 - nombre (texto)
 - cantidad (entero)
 - precio (número real)
- 2) Escribir un método que retorne y calcule el resultado de la siguiente expresión aritmética que indica la cantidad de calor Q que fluye en una barra de aluminio con sus extremos sometidos a dos temperaturas diferentes T_A y T_B (°K).

$$Q = K \frac{T_A - T_B}{d} A t$$

Donde d es la longitud de la barra (m), A la sección (m²), y t es el tiempo transcurrido (s). En la clase de este método estaría ya declarada la constante $K=209.3\text{W/m}^\circ\text{K}$ que indica el coeficiente de transmisión de calor. El resto de los datos se le pasarán al método mediante parámetros.

- 3) Se dispone de la siguiente clase, de la que se muestra sólo una parte:

```
/**
 * Clase que almacena los datos de un barco hundido
 * y calcula datos para reflotarlo.
 */
public class BarcoHundido
{
    // atributos privados
    ...

    /**Constructor. Masa en Tm, Superficie en m2, profundidad en m */
    public BarcoHundido(double masa, double superficie,
                       double profundidad)
    { ... }

    /** Retorna el volumen de aire a inyectar en el barco (m3) */
    public double volumen() { ... }

    /** Retorna el tiempo de llenado (seg), dado el caudal en m3/seg */
    public double tiempo(double caudal) { ... }
}
```

Escribir un fragmento de programa (no un programa entero) con las declaraciones e instrucciones que nos permiten:

- Crear dos objetos uno para un barco de 830Tm, de 280m² de superficie y hundido a 235 m de profundidad, y otro de 2856Tm, 532m², y 37 m.
- Mostrar en pantalla el volumen de aire necesario para reflotar cada uno de los dos barcos.
- Mostrar en pantalla los tiempos de llenado para ambos barcos.

- 4) Indicar en qué lugar de un programa Java se declaran cada uno de los siguientes datos, e indicar su principal utilidad, remarcando para cada uno de ellos lo que le distingue de los demás. Contestar con un máximo de tres líneas por cada dato:
- atributo
 - variable
 - parámetro
- 5) Escribir la secuencia de instrucciones necesarias en un intérprete de órdenes de Linux para copiar (con todo su contenido) un directorio llamado `problema1` y situado en un directorio de usuario de nombre completo `/home/pedro`; queremos copiarlo dentro de un directorio llamado `copias` ya existente en el mismo directorio de usuario. Además, es preciso borrar en el directorio original todos los ficheros acabados en `".class"` y `".txt"` (sin las comillas). El directorio de partida es `/tmp`.

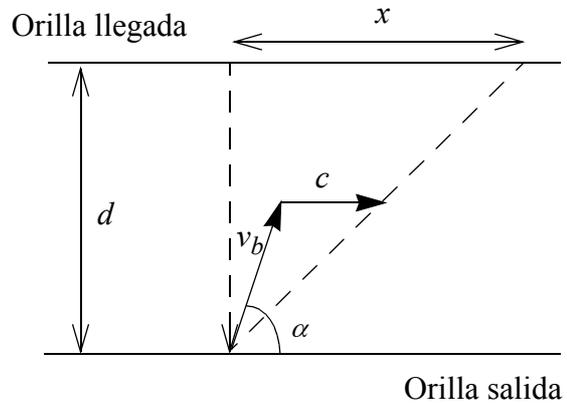
Examen de Fundamentos de Computadores y Lenguajes

Examen Parcial. Febrero 2005

Segunda parte: Problema (5 puntos)

Se desea escribir un programa para calcular el tiempo empleado por un bote en cruzar un río, así como predecir el punto de la orilla opuesta al que se llegará. La figura de la derecha muestra los datos del problema, que son:

- c : velocidad del agua (m/s)
- v_b : velocidad relativa del bote respecto al agua (m/s)
- d : anchura del río (metros)
- α : ángulo de la dirección hacia la que se dirige la barca (suponiendo el agua quieta) (grados)
- x : desviación del punto de llegada respecto a la perpendicular con el de salida (metros)
- t : tiempo (seg)



El programa constará de dos clases: la clase `Rio` que guarda los datos del río y del bote y que tiene métodos para calcular el tiempo y punto de llegada del cruce del río; y la clase `CruceRio` que contiene el método principal (`main`).

La clase `Rio` está compuesta por:

- *atributos privados*: c , v_b , d (explicados arriba)
- *métodos públicos*:
 - un constructor al que se le pasan como parámetros los valores iniciales de los atributos, para copiarlos.
 - `desviacionLlegada`: método al que se le pasa como parámetro el ángulo α y que retorna la desviación del punto de llegada, x , obtenida como:

$$x = \frac{d(v_b \cos \alpha + c)}{v_b \sin \alpha}$$

- `tiempoLlegada`: método al que se le pasa como parámetro el ángulo α y que retorna el tiempo de llegada a la orilla, t , obtenido como:

$$t = \frac{d}{v_b \sin \alpha}$$

La clase `CruceRio` tiene el método `main` que hace lo siguiente:

- Lee de teclado (mediante un objeto de la clase `Lectura`) los datos necesarios para calcular el paso del río: c , v_b , d , α .
- Crea un objeto de la clase `Rio` usando los datos leídos del teclado.

- Calcula mediante los métodos de la clase `Rio` el tiempo de llegada y la desviación del punto de llegada al cruzar el río, y los almacena en sendas variables.
- Muestra en una ventana de la clase `Escritura` los dos resultados obtenidos, así como la velocidad media de la barca, calculada como el cociente entre la distancia total recorrida (hipotenusa del triángulo cuyos catetos son d y x) y el tiempo de llegada.