



Universidade de Vigo

ESCOLA SUPERIOR DE  
ENXEÑERÍA INFORMÁTICA

## Programación Avanzada – Febrero de 2003 –

### Examen de Teoría

NOTA

Apellidos \_\_\_\_\_

Nombre \_\_\_\_\_

PLAN: • Nuevo • Viejo

DNI \_\_\_\_\_

---

#### IMPORTANTE:

- El tiempo para la realización del examen es de 2 horas 30 minutos.
- Se deben cubrir los datos del encabezado y firmar la parte de atrás de esta hoja, que será grapada junto con las hojas de respuestas que deberán ir numeradas.
- En el momento de la entrega del examen se deberá firmar una hoja de entrega, que estará a disposición del alumno.

#### REVISIÓN DE EXÁMENES:

**Viernes (21 de febrero) [8:30-13:30] y [15:30-19:30].**

---

☐ **1.- Generalidades sobre Java:**

- 1.1.- ¿Qué diferencia existe entre un JDK y un JRE?.
- 1.2.- Cuando el compilador o el intérprete de Java necesitan localizar una clase, ¿cuál es la secuencia de pasos que siguen estos programas hasta encontrarla?.
- 1.3.- ¿Qué significado tiene el incluir un punto (.) en el contenido de la variable CLASSPATH?.

☐ **2.- Paquetes en Java:**

- 2.1.- ¿Qué es un paquete de Java?.
  - 2.2.- ¿Cuál es el paquete que se importa automáticamente en todas las aplicaciones Java?.
  - 2.3.- ¿Cómo afecta esta organización a los modificadores de acceso a los miembros de una clase?.
- Tabla.

☐ **3.- Miembros (atributos y métodos) de clase e instancia en Java:**

- 3.1.- ¿Cuál es su diferencia?.
- 3.2.- ¿Por qué el método *main* es un método de clase?.
- 3.3.- ¿Para qué sirve un *iniciador estático*?.

☐ **4.- Atributos, métodos y clases finales. ¿Qué ocurre cuando aplicamos el modificador *final* en la declaración de estas entidades en Java?.**

☐ **5.- Excepciones en Java:**

- 5.1.- ¿Cuál es diferencia entre excepciones *comprobadas* y *no comprobadas* en Java?.
- 5.2.- Forma general de un bloque de gestión de excepciones en Java. Explicar todas sus cláusulas.

☐

**6.- Seguridad en Java:**

6.1.- Sintaxis de una *entrada garantizada*.

6.2.- Diferencia entre una *entrada garantizada* y una *entrada de permiso*.

6.3.- ¿Dónde se definen la *entradas garantizadas* y las *entradas de permiso* en Java?.

☐

**7.- Desarrollo de aplicaciones distribuidas en Java:**

7.1.- En el lado servidor, ¿qué ficheros se necesitan para construir una clase que exporte métodos mediante RMI?.

7.2.- ¿Cuál es la utilidad de la herramienta *rmic* de Java y de los ficheros que genera?.

☐

**8.- Programación multihilo en Java:**

8.1.- ¿De qué dos formas se puede crear un hilo en Java?.

8.2.- ¿Cuáles son los dos niveles de bloqueo de un recurso en Java?.

8.3.- ¿Cómo podemos garantizar el acceso exclusivo a objetos de una clase que no fue diseñada para acceso multihilo?.

☐

**9.- Gestión de Entrada / Salida en Java:**

9.1.- Diferencia entre las clases Reader/Writer e InputStream/OutputStream.

9.2.- ¿Qué es la serialización?.

9.3.- ¿Para qué sirve declarar un atributo con el modificador *transient*?.

☐

**10.- Generación dinámica de contenidos con Java:**

10.1.- ¿Cuál es la principal diferencia entre un Servlet y una página JSP?.

10.2.- La especificación JSP requiere que todas las implementaciones de JSP soporten salida con buffer, ¿por qué?.

10.3. El protocolo HTTP carece de estado, ¿qué dos posibilidades ofrece Java para mantener el estado de una sesión?.

---

En Ourense, a 13 de febrero de 2003

Fdo.:

---