



PROGRAMACIÓN AVANZADA

[DICIEMBRE / FC]
2007

ESCOLA SUPERIOR DE
ENXEÑERÍA INFORMÁTICA

NOTA

UNIVERSIDADE
DE VIGO

APELLIDOS _____ NOMBRE _____

DNI _____ FIRMA _____

IMPORTANTE:

- Cada pregunta tiene una única respuesta correcta.
- Se deberá firmar la hoja del examen y una hoja de control de firmas en el momento de la entrega.
- La aportación a la puntuación por el tipo de respuesta es la siguiente:

Correcta	+ 0,4	puntos
No contestada	0	puntos
Incorrecta	- 0,2	puntos
- La revisión de TEORÍA y PRÁCTICAS se llevará a cabo en el siguiente horario:
18 de diciembre de 2007 (martes) de [09:00h - 13:00h] y [16:00h - 19:00h]
- Tiempo de realización: 40 minutos.
- Sólo se corregirán las respuestas que aparezcan marcadas en la tabla que sigue a continuación:

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20	C21	C22	C23	C24	C25
A																									
B																									
C																									
D																									

1. Características del lenguaje Java, ¿qué afirmación es la correcta?

- A) Java soporta la utilización de estructuras y uniones para la definición de estructuras de datos complejas.
B) Aunque se desaconseja su uso, Java dispone de métodos para liberar la memoria ocupada por aquellos objetos que se salen de ámbito.
C) Java da soporte a la instanciación de variables y objetos en tiempo de compilación (ligadura estática).
D) En Java pueden existir clases de usuario sin constructores públicos.

2. Prioridad y asociatividad de los operadores en Java, ¿qué afirmación es la correcta?

- A) La asociatividad del operador ternario de Java (?:) es de "izquierda a derecha".
B) El operador de asignación en Java (=) tiene más prioridad que el operador de incremento (++).
C) Los operadores de multiplicación (*), división (/) y módulo (%) tienen la misma prioridad.
D) Las afirmaciones A y B son correctas.

3. Paquetes en Java, ¿qué afirmación es la correcta?

- A) Un paquete en Java puede contener clases, interfaces y otros paquetes.
B) Todas las clases Java (incluidas las de usuario) pertenecen siempre a algún paquete.
C) En Java no se puede aplicar ningún modificador de acceso (*public*, *protected*, *private* o *package*) a un paquete.
D) Todas las afirmaciones anteriores son verdaderas.

4. Sentencia *import* en Java, ¿qué afirmación es la correcta?

- A) Mediante una única sentencia *import* se pueden importar clases pertenecientes a distintos paquetes.
B) En un programa Java pueden aparecer cualquier número de sentencias *import*.
C) Al importar un paquete en Java se importan automáticamente todos los subpaquetes que este contenga.
D) Las afirmaciones B y C son verdaderas.

5. Iniciadores estáticos en Java, ¿qué afirmación es la correcta?

- A) Los iniciadores estáticos son método anónimos.
B) Los iniciadores estáticos no tienen parámetros y además no retornan ningún valor.
C) Los iniciadores estáticos son invocador automáticamente por el intérprete de Java.
D) Todas las afirmaciones anteriores son verdaderas.

6. ¿Qué modificadores de acceso se pueden aplicar a una clase (no interna) en Java?

- A) *public*, *protected*, *private* y *package*.
B) *protected* y *public*.
C) *private* y *package*.
D) *public* y *package*.

7. Modificador *static* en Java, ¿qué afirmación es la correcta?

- A) Para acceder a un atributo declarado como *static* se puede utilizar un objeto de la clase o bien el nombre de la clase.
B) Los atributos declarados como *static* deben ser inicializados en el momento de la declaración.
C) Los métodos estáticos tienen acceso a los objetos de su clase a través de la palabra reservada *this*.
D) Los métodos estáticos no pueden ser sobrecargados.

8. Modificador *final* en Java, ¿qué afirmación es la correcta?

- A) Cuando una clase se declara como *final*, todos sus métodos se convierten automáticamente en *final*.
B) El modificador *final* puede aplicarse en la declaración de una referencia.
C) El modificador *final* puede aplicarse en la declaración de parámetros de un método.
D) Todas las afirmaciones anteriores son verdaderas.

9. Constructores en Java, ¿qué afirmación es la correcta?

- A) Los constructores en Java son, por defecto, sincronizados.
B) Cuando en una clase se declara como usuario no se especifica ningún constructor, el compilador de Java añade uno automáticamente que no recibe ningún parámetro.
C) Desde código fuente es posible invocar a un constructor simplemente poniendo su nombre (como si de un método más se tratara).
D) Los constructores no pueden devolver ningún valor de retorno, por lo que deben ser declarados con el modificador *void*.

10. Modificador *abstract* en Java, ¿qué afirmación es la correcta?

- A) De una clase abstracta no se puede heredar.
B) Todos los métodos de una clase abstracta deben ser abstractos.
C) En cuanto una clase define algún método como abstracto, la clase debe ser abstracta.
D) En Java no tiene sentido declarar una referencia a una clase abstracta.

11. Redefinición de métodos en Java, ¿qué afirmación es la correcta?

- A) Cuando en una subclase se redefine un método de una superclase con distinto tipo o número de parámetros, el método de la superclase no se oculta, sino que se comporta como una sobrecarga de ese método.
B) En el caso de que el método heredado por una subclase sea abstracto en la superclase, es obligatorio redefinirlo, de lo contrario la subclase debería ser declarada también abstracta.
C) Se puede redefinir un método cambiando únicamente su valor de retorno.
D) Las afirmaciones A y B son verdaderas.

12. Conversión de objetos en Java, ¿qué afirmación es la correcta?

- A) Se puede utilizar una referencia de una superclase para acceder a un objeto de la subclase de forma directa, es decir, sin necesidad de hacer un casting explícito.
- B) Para la conversión entre objetos de distintas clases, Java exige que dichas clases estén relacionadas por herencia.
- C) Utilizando una referencia de una superclase se puede acceder a las variables exclusivas de la subclase.
- D) Todas las afirmaciones anteriores son verdaderas.

13. Interfaces en Java, ¿qué afirmación es la correcta?

- A) Las clases internas estáticas pueden implementar una determinada interfaz están obligadas a proporcionar una definición para todos los métodos definidos por esa interfaz y las interfaces de las que esta hereda.
- B) Las interfaces sólo admiten los modificadores de acceso *public* y *package*.
- C) Las constantes definidas en una interfaz son implícitamente *public*, *static* y *final*.
- D) Todas las afirmaciones anteriores son verdaderas.

14. Clases e interfaces internas en Java, ¿qué afirmación es la correcta?

- A) Las clases internas estáticas pueden acceder a todos los atributos de la clase contenedora.
- B) Las clases internas miembro puede ser declaradas utilizando los modificadores de acceso *private* y *protected*.
- C) Las clases internas locales sólo tienen acceso a las variables y parámetros pertenecientes al método donde han sido declaradas, no pudiendo acceder a los atributos de la clase contenedora.
- D) Todas las afirmaciones anteriores son falsas.

15. Gestión de Excepciones en Java, ¿qué afirmación es la correcta?

- A) Utilizando la sentencia *throw* no es posible hacer que un programa lance una subclase de excepción propia (definida por el programador).
- B) Siempre que un método pueda lanzar una excepción que no gestiona él mismo, es obligatorio declarar este hecho mediante *throws* para que los métodos que lo llaman puedan protegerse frente a esa excepción.
- C) El bloque *finally* debe ir detrás de todos los bloques *catch* considerados, pero puede existir una cláusula *try* sin ningún *catch* y con un *finally*.
- D) Las afirmaciones B y C son verdaderas.

16. Ficheros .jar en Java, ¿qué afirmación es la correcta?

- A) Los ficheros *.jar* únicamente pueden contener ficheros de clases (*.class*).
- B) A partir del JDK 1.2 los ficheros *.jar* firmados digitalmente se ejecutan en la JVM con todos los permisos.
- C) Un fichero *.jar* sólo puede ser firmado digitalmente por un único usuario con el fin de garantizar su procedencia.
- D) Todas las afirmaciones anteriores son falsas.

17. Applets en Java, ¿qué afirmación es la correcta?

- A) Un applet en Java puede invocar a métodos del navegador sobre el que se está ejecutando.
- B) Los métodos *init*, *start*, *main*, *stop* y *destroy* de un applet son llamados automáticamente por el navegador en distintos momentos de su ciclo de vida.
- C) Los applets no pueden recibir parámetros desde la página HTML desde la que son invocados.
- D) Las afirmaciones A y C son falsas.

18. Programación multihilo en Java, ¿qué afirmación es la correcta?

- A) Los métodos estáticos no pueden ser declarados como *synchronized*.
- B) Los métodos *wait*, *notify* y *notifyAll* únicamente pueden ser invocados desde un método o un bloque sincronizado.
- C) Cuando un hilo despierta de su sueño, retoma la CPU y continúa ejecutándose justo después de la llamada al método *sleep*.
- D) En Java se pueden crear hilos heredando de la clase *Runnable* o implementando la interfaz *Thread*.

19. Gestión de E/S en Java, ¿qué afirmación es la correcta?

- A) Las clases *ObjectInputStream* y *ObjectOutputStream* permiten leer y escribir objetos utilizando streams (flujos) que hereden de *Reader* o *Writer*.
- B) Mediante la clase *File* un programa puede crear un fichero en disco y escribir o leer de él su contenido.
- C) Los métodos *mark* y *reset* no están disponibles en todos los streams que heredan de *InputStream*.
- D) Todas las afirmaciones anteriores son falsas.

20. Programación para la red en Java, ¿qué afirmación es la correcta?

- A) Las clases *DatagramPacket* y *DatagramSocket* hacen uso del conjunto de protocolos TCP/IP.
- B) A partir de un mismo *Socket* en Java se pueden crear dos flujos distintos: uno para leer y otro para escribir caracteres.
- C) Mediante la utilización de la clase *URLConnection* se puede acceder al contenido de un archivo remoto identificado por su dirección URL.
- D) Las afirmaciones B y C son verdaderas.

21. Desarrollo de aplicaciones distribuidas en Java, ¿qué afirmación es la correcta?

- A) Para poder invocar localmente un método de un objeto remoto en Java, sus parámetros deben ser serializables.
- B) Para poder invocar localmente un método de un objeto remoto en Java, es necesario que el fichero *.class* del objeto remoto sea accesible por el objeto local a través de la variable CLASSPATH.
- C) RMI permite la codificación de objetos remotos en otros lenguajes de programación distintos a Java, siempre y cuando proporcionen una interfaz Java de sus métodos al objeto local.
- D) Todas las afirmaciones anteriores son falsas.

22. Programación de Servlets en Java, ¿qué afirmación es la correcta?

- A) Un Servlet puede ser ejecutado escribiendo su URL directamente en la barra de direcciones del navegador.
- B) Un Servlet puede ser ejecutado desde una página HTML.
- C) Un Servlet puede ser ejecutado por otro Servlet (instanciando su clase a través del operador *new* e invocando a los métodos apropiados).
- D) Todas las afirmaciones anteriores son verdaderas.

23. Programación de páginas JSP en Java, ¿qué afirmación es la correcta?

- A) La especificación JSP obliga a que todas las implementaciones de JSP soporten salida con búfer.
- B) El contenedor JSP exporta varios objetos internos para su uso directo (sin declaración previa) desde las páginas *.jsp*.
- C) Las páginas *.jsp* pueden ser ejecutadas directamente mediante la utilidad *appletviewer* del JDK de Java.
- D) Las afirmaciones A y B son verdaderas.

24. Sobre CLASSPATH y PATH, ¿qué afirmación es la correcta?

- A) Ambas son variables de entorno del Sistema Operativo.
- B) Para poder cargar el controlador JDBC desde una aplicación Java, se debe establecer su ruta correctamente en la variable PATH.
- C) El JRE (entorno de ejecución de Java) a diferencia del JDK (kit de desarrollo de Java) no hace uso de la variable CLASSPATH.
- D) Todas las afirmaciones anteriores son falsas.

25. Sobre la práctica, ¿qué afirmación es la correcta?

- A) El servidor de SERENO da servicio a múltiples clientes conectados al mismo tiempo a través de programación multihilo.
- B) El servidor de AUTENTIFICACIÓN y LOGS no tiene porque ser ejecutado en la misma máquina que el resto de aplicaciones (cliente y servidor de SERENO).
- C) La implementación propuesta hace uso del protocolo UDP.
- D) Las afirmaciones A y B son verdaderas.