

Apellidos: \_\_\_\_\_ Nombre: \_\_\_\_\_ DNI: \_\_\_\_\_

Centro Asociado en el que está MATRICULADO: \_\_\_\_\_

**INSTRUCCIONES:** Complete sus datos personales en la cabecera de esta hoja, y **ENTRÉGUELA OBLIGATORIAMENTE** con el resto de hojas de su examen. **Cualquier examen que no venga acompañado de esta hoja de enunciados no será corregido.** Complete **TODOS** los datos que se piden en la hoja de lectura óptica o **en caso contrario su examen no será corregido.** El examen consta de 20 preguntas. Cada respuesta acertada tiene un valor de 0.5 puntos y cada respuesta incorrecta de -0.25 puntos. Para superar el examen es necesario obtener una puntuación de al menos 5 puntos.

**Test :** Conteste exclusivamente en una HOJA DE LECTURA ÓPTICA, no olvidando marcar que su tipo de examen es B.

1. Para poder ejecutar un applet denominado TicTacToe.class empaquetado en formato jar hay que incluir en la página HTML que va a contener al applet la siguiente información:

- A) `<applet class=TicTacToe.class archive=TicTacToe.jar width=200 height=100> </applet>`
- B) `<applet code=TicTacToe.jar archive=TicTacToe.class width=200 height=100> </applet>`
- C) `<applet code=TicTacToe.class archive=TicTacToe.jar width=200 height=100> </applet>`
- D) Ninguna de las anteriores.

2. Para empaquetar un Java Bean es necesario:

- A) Utilizar la herramienta jar de la misma forma que para el empaquetamiento de una clase normal.
- B) Escribir un archivo de manifiesto para indicar que el objeto a empaquetar es un Java Bean y, a continuación, utilizar la herramienta jar.
- C) Escribir un archivo de manifiesto para indicar el nombre de la clase principal del Bean y, a continuación, recurrir a la herramienta jarbean para el empaquetamiento de objetos del tipo Java Bean.
- D) Ninguna de las anteriores.

3. Dado el siguiente código se puede afirmar que

```
public void miMetodo(int uno) {  
    return(uno+uno);  
}
```

- A) Es correcto y devolverá la suma de uno más uno.
- B) Es incorrecto el método nunca se puede definir del tipo public.
- C) Es incorrecto, si se usa el tipo void no se usa la palabra clave return.
- D) Ninguna de las anteriores.

4. Señale cuál es resultado de la ejecución del método *main* de la clase siguiente:

```
public class Ejemplo2 {  
    public static void main(String[] args) {  
        int y=0;  
        {  
            System.out.print(y+" ");  
            int y=1;  
            System.out.print(y);  
        }  
    }  
}
```

- A) 0 0
- B) 0 1
- C) 1 1

D) Se obtiene error, ya que aunque los bloques pueden estar anidados, no se puede declarar una variable con el mismo nombre que otra que está en un ámbito exterior.

5. Indique cuál de las siguientes afirmaciones es verdadera:

- A) El rango de valores del tipo simple (o primitivo) entero *int* no está estrictamente definido en Java, sino que depende de la arquitectura de la plataforma. Por ejemplo, un *int* tiene como rango  $[-2^{31}, 2^{31}-1]$  en el caso de ordenadores de 32 bits, y tiene como rango  $[-2^{15}, 2^{15}-1]$  en el caso de ordenadores de 16 bits. Esto garantiza que no se produce pérdida de rendimiento en ningún entorno.
- B) Java contiene tipos simples (o primitivos) enteros cuyos rangos de valores no contienen números negativos. En particular, los rangos de valores de los tipos enteros *byte* y *short* no contienen números negativos.
- C) El tipo *char* de Java tiene un tamaño de 16 bits, ya que Java utiliza *Unicode*.
- D) Todas las afirmaciones anteriores son falsas.

6. Señale cuál es resultado de la ejecución del método *main* de la clase siguiente:

```
public class Ejemplo1 {
    public static void main(String args[]) {
        int x=2^3; // ^ es el operador OR exclusivo a nivel de bit
        System.out.print("Resultado: ");
        System.out.println((int) (2.6*x));
    }
}
```

- A) Resultado: 3
- B) Resultado: 2
- C) Resultado: 20
- D) Ninguna de las anteriores.

7. Señale cuál de los siguientes métodos *main* proporciona el resultado siguiente: 0 11 20

- A) 

```
public class Ejemplo3 {
    public static void main(String[] args) {
        int x=0, y=10, z=20;
        if (x==1 && y++<10) z=100;
        System.out.println(x+" "+y+" "+z);
    }
}
```
- B) 

```
public class Ejemplo3 {
    public static void main(String[] args) {
        int x=0, y=10, z=20;
        if (x==1 & y++<10) z=100;
        System.out.println(x+" "+y+" "+z);
    }
}
```
- C) 

```
public class Ejemplo3 {
    public static void main(String[] args) {
        int x=0, y=10, z=20;
        if (x=1 & ++y<11) z=100;
        System.out.println(x+" "+y+" "+z);
    }
}
```
- D) Ninguno de los anteriores.

8. Indique cuál es el resultado de la ejecución del método *main* de la clase *Ejemplo4*:

```
public class NumeroEntero {
    int i;
    NumeroEntero(int i) { this.i = i; }
```

```

void incrementaA(int i) { this.i += i; }
void incrementaB(int i, int j) { i += j; }
void incrementaC(NumeroEntero n, int j) { n.i += j; }
void incrementaD(int j) { i += j; }
}

public class Ejemplo4 {
    public static void main(String[] args) {
        NumeroEntero n = new NumeroEntero(1);
        for (int k=0; k<=3; k++) {
            switch(k) {
                case 0: n.incrementaA(2); break;
                case 1: n.incrementaB(n.i,2); break;
                case 2: n.incrementaC(n,2); break;
                case 3: n.incrementaD(2); break;
            }
            System.out.print(n.i+" ");
        }
    }
}

```

- A) 3 5 7 9                      B) 3 3 5 7                      C) 3 5 5 7                      D) Ninguno de los anteriores.

9. Debido al sistema de seguridad implícito de Java, un applet que se descarga a través de la red en un computador, que denominaremos cliente, para ser ejecutado en un navegador:
- A) No puede leer o escribir archivos del computador cliente en el que se ejecuta.
  - B) No puede comunicarse con otras aplicaciones utilizando sockets.
  - C) Necesita de una instalación previa del JRE para poder ejecutarse en el navegador.
  - D) Ninguna de las anteriores.
10. En un applet, el método *start()* se diferencia del método *init()* en que:
- A) Sólo se invoca cuando se carga el applet mientras que el método *init()* se puede invocar todas las veces que sea necesario.
  - B) Se puede invocar las veces que sea necesario mientras que el método *init()* se invoca una vez durante toda la vida del applet.
  - C) El método *start()* se utiliza antes que el método *init()* para proceder a la inicialización de toda la interfaz gráfica.
  - D) Ninguna de las anteriores.
11. Si se desea efectuar el posicionamiento absoluto de los objetos en la interfaz gráfica de un applet o aplicación hay que recurrir al método:
- A) `setLayout (new Layout(null))`.
  - B) `setLayout (null)`.
  - C) `setLayout (new NullLayout())`.
  - D) Ninguna de las anteriores.
12. ¿Cuál de las siguientes palabras claves hace referencia a la herencia entre clases en Java?
- A) `implements`
  - B) `inherits`

- C) extends
- D) imports

13. ¿Cuál de las siguientes afirmaciones es correcta?

- A) La clase *Error*, está relacionada con errores de programación. El usuario debe preocuparse en capturarlos y tratarlos.
- B) La clase *Exception*, está relacionada con excepciones que el usuario debe tener en cuenta y chequear si se producen.
- C) La clase *RuntimeException* está relacionada con errores de compilación.
- D) Ninguna de las anteriores.

14. ¿Cuál de las siguientes afirmaciones es correcta?

- A) Una clase abstracta es aquella que posee al menos un método abstracto;
- B) Una clase abstracta debe tener todos sus métodos abstractos;
- C) Una clase abstracta debe tener todos sus métodos y miembros abstractos.
- D) Ninguna de ellas.

15. ¿Cuál de las siguientes afirmaciones NO es cierta?

- A) Una clase puede heredar solamente de otra clase.
- B) Una clase puede heredar de muchas clases a la vez sin restricciones.
- C) Una clase puede heredar de una clase e implementar una interfaces.
- D) Una clase puede implementar varias interfaces sin restricciones.

16. ¿Cuál de las siguientes afirmaciones es cierta?

- A) En Java todos los archivos están orientados a bytes.
- B) En Java todos los archivos están orientados a caracteres.
- C) En Java todos los archivos están orientados a sockets.
- D) Ninguna de ellas.

17. ¿Para qué se utiliza la palabra clave imports?

- A) Informa a la máquina virtual que se van a usar todas las clases del API de Java
- B) Informa a la máquina virtual que se va a usar un fichero de gráficos
- C) Informa al compilador para que compile todas las clases de la API de Java.
- D) Permite referirse a las clases de la API de Java de una forma más breve y por tanto clarificando el código del programa.

18. ¿Cual de las siguientes sentencias es incorrecta?

- (1) `persona[25] personas;`
  - (2) `persona[] personas;`
  - (3) `persona personas[] = new persona 25;`
  - (4) `persona personas[25] = new persona [25]`
- A) 1 y 3.

- B) 1 y 4.
- C) 2
- D) Ninguna de las anteriores

19. ¿Cual sería el resultado del siguiente código JAVA ?

```
InetAddress addr=InetAddress.getByName(null);
```

- A) Anular el envío de datos através de un socket.
- B) Produce la InetAddress de la dirección IP del bucle local.
- C) Produce la InetAddress de la dirección IP de un servidor remoto desconocido.
- D) Ninguna de las anteriores

20. ¿Cuál de las siguientes afirmaciones es cierta?

- A) Un programa Java termina cuando acaban todos sus threads.
- B) Un programa Java termina cuando acaban todos sus threads o cuando los únicos que quedan están clasificados como *daemons*.
- C) Un programa Java termina cuando acaban todos sus threads si estos han sido mediante una implementación de *Runnable*.
- D) Ninguna de las anteriores