

Apellidos: _____ Nombre: _____ DNI: _____

Centro Asociado en el que está MATRICULADO: _____ Especialidad: _____

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 C.

1. Si se desea conocer la dirección IP real de un computador que se encuentra conectado a Internet, no la dirección conocida como *localhost*, una de las posibles formas de obtenerla es mediante:
 - A) `InetAddress myAddress = InetAddress.getLocalHost()`
 - B) `InetAddress myAddress = InetAddress.getIPAddress()`
 - C) `InetAddress myAdress = InetAddress.getByName(null)`
 - D) Ninguna de las anteriores.
2. Uno de los objetos específicos que proporciona el lenguaje Java para el almacenamiento de las direcciones IP y para, posteriormente, poder construir un objeto de la clase Socket es:
 - A) `java.net.ServerSocket`
 - B) `java.net.Address`
 - C) `java.net.InetAddress`
 - D) Ninguna de las anteriores.
3. Cuando se pretende que un computador que realiza las tareas de cliente establezca una conexión mediante un socket con el computador que hace de servidor, es necesario que el programa cliente o applet:
 - A) Conozca la dirección IP del servidor, así como el puerto en el que éste permanece a la escucha para atender las conexiones entrantes.
 - B) Conozca la dirección IP del servidor ya que el número de puerto del servidor lo obtendrá el cliente posteriormente mediante el método `Socket.getPort()` una vez que el servidor haya aceptado la conexión.
 - C) Conozca únicamente el nombre del servidor para enviarle la dirección IP y el puerto desde el que está solicitando el establecimiento de la conexión.
 - D) Ninguna de las anteriores.
4. El método `accept()` perteneciente al objeto `ServerSocket` se utiliza para:
 - A) Que los programas que realizan tareas de cliente puedan iniciar la conexión con el servidor.
 - B) Que un programa que realiza tareas de servidor permanezca a la escucha y genere un socket mediante el que intercambiar datos con el cliente.
 - C) Única y exclusivamente programar servidores concurrentes, es decir, que puedan atender a varios clientes simultáneamente.
 - D) Ninguna de las anteriores.
5. La construcción de una aplicación que tenga la capacidad de atender varios clientes de forma simultánea a través de sockets, es decir, un servidor concurrente:
 - A) No conlleva la realización de ningún tipo especial de operación ya que el objeto `ServerSocket` crea un socket para cada una de las conexiones entrantes.
 - B) Implica la creación en el servidor de tantos threads como clientes existan de forma que en cada hilo se instancie un objeto `ServerSocket` con el método `accept()` para atender la petición de conexión de un cliente concreto.
 - C) Implica la utilización de threads con la finalidad de crear múltiples objetos que pueda atender en exclusiva los canales de comunicación establecidos con los múltiples clientes.
 - D) Ninguna de las anteriores.

6. ¿Cuál sería el resultado de la ejecución del siguiente fragmento de código?

```
class Prueba1{
    int x;
    char y;
}

class Prueba2 extends Prueba1{
    int z;
    int x=3;
}

public class Prueba extends Prueba2{
    public Prueba() {
        int x=5;
    }
    public static void main(String[] args) {
        Prueba1 AAA = new Prueba();
        AAA.x=1;
        System.out.println(AAA.x);
    }
}
```

- A) Imprime en consola el entero 1.
 - B) Ninguna. No se puede asignar un manejador de una clase a un objeto de otra.
 - C) Imprime en consola el entero 5.
 - D) Imprime en consola el entero 3.
7. ¿Qué método de la interfaz AppletContext recibe como argumento un objeto URL y muestra en una ventana de un navegador el recurso asociado con la URL que se le indica?
- A) showURL
 - B) getDocument
 - C) showDocument
 - D) getURL
8. La palabra reservada *synchronized* se utiliza para:
- A) Evitar los accesos simultáneos a un recurso cuando varios thread pueden acceder a él.
 - B) Ordenar el acceso a un recurso.
 - C) Sincronizar las llamadas al método con la prioridad del thread.
 - D) Evitar que un método se quede bloqueado al acceder a un recurso compartido.
9. ¿Cuál sería el resultado del siguiente código JAVA ?
- ```
InetAddress addr=InetAddress.getByName(null);
```
- A) Anular el envío de datos a través 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.
10. Indicar qué afirmación es correcta:
- A) La variable de entorno CLASSPATH indica dónde se encuentra los ficheros ejecutables del SDK
  - B) La variable de entorno CLASSPATH indica dónde se encuentran los ficheros objeto (.class)
  - C) La variable de entorno CLASSPATH indica dónde se encuentra los ficheros fuente (.java)
  - D) Ninguna de las anteriores

11. ¿Cuál sería el resultado de la ejecución del siguiente fragmento de código?

```
import java.io.*;

public class Fichero {
 public Fichero() {
 }
 public static void main(String[] args) throws IOException{
 String str;
 String NombreArchivo="";
 BufferedReader br=new BufferedReader(new InputStreamReader(System.in));

 System.out.print("Nombre del archivo : ");
 try {
 NombreArchivo = br.readLine();
 }
 catch (IOException ex) {
 }

 int i;
 FileInputStream fin;

 try{
 fin=new FileInputStream(NombreArchivo);

 do{
 i=fin.read();
 if(i != -1){
 System.out.print((char) i);
 }
 }
 while (i!=-1);

 }
 catch (FileNotFoundException e) {
 System.out.println("Archivo no encontrado");
 }
 }
}
```

- A) Copia en sí mismo el fichero cuyo nombre se encuentra en la variable NombreArchivo
- B) Imprime en consola el fichero cuyo nombre se encuentra en la variable NombreArchivo
- C) No hace nada debido a un error en la programación de las excepciones.
- D) No hace nada debido a un error en los flujos.

12. ¿Cuál sería el resultado de la ejecución del siguiente fragmento de código?

```
import java.io.Serializable;

class Persona implements Serializable, Cloneable
{String nombre;
String apellidos;
int edad;

Persona(String x,String y,int z){
 nombre=x;
 apellidos=y;
 edad=z;
}

public Object clone(){
 Persona p=null;
 try{
 p=(Persona)super.clone();
 p.nombre=new String(nombre);
 p.apellidos=new String(apellidos);
 }
 catch(Exception e){
 }
 return p;
}

public class Driver {
 public Driver() {
 }
 public static void f(Persona p){
 p.edad=50;
 System.out.println(p.edad);
 }
 public static void main(String[] args) {
 Driver driver1 = new Driver();
 Persona x=new Persona("Juan", "Garcia", 20);
 System.out.println(x.edad);
 f((Persona)x.clone());
 System.out.println(x.edad);
 }
}
```

- A) Imprime en consola 20  
20  
20
- B) Imprime en consola 20  
20  
50
- C) Imprime en consola 20  
50  
20
- D) Imprime en consola 20  
50  
50

13. Señale cuál es el resultado de la ejecución del método main de la clase siguiente. Las líneas se han numerado con el propósito de facilitar la referencia a las mismas.

```
1 public class Ejemplo1 {
2 public static void main(String[] args) {
3 int precision=2;
4 double pi=0;
5 if (precision==2) { double pi = 3.14; }
6 if (precision==4) { double pi = 3.1415; }
7 System.out.println(pi);
8 }
9 }
```

- A) Se obtiene error en las líneas 5 y 6, debido a que la variable pi ya ha sido definida en main.
- B) 3.14
- C) 3.1415
- D) Ninguna de las anteriores.

14. Señale cuál es el resultado de la ejecución del método main de la clase Ejemplo3. Las líneas se han numerado con el propósito de facilitar la referencia a las mismas.

```
1 public class Ejemplo3_A {
2 int i = 1;
3 void letreroA() { System.out.print(" A"+i); }
4 }
5
6 public class Ejemplo3_B extends Ejemplo3_A {
7 int i = 2;
8 void letreroA() {System.out.print(" AA"+i); }
9 void letreroB() {System.out.print(" B"+i); }
10 }
11
12 public class Ejemplo3 {
13 public static void main(String[] args) {
14 Ejemplo3_B e1 = new Ejemplo3_A();
15 e1.letreroA();
16 e1.letreroB();
17 }
18 }
```

- A) A1 B2
- B) AA1 B2
- C) Se obtiene error en la línea número 14.
- D) Ninguno de las anteriores.

15. Indique cuál es el resultado de la ejecución del método main de la clase siguiente. Las líneas se han numerado con el propósito de facilitar la referencia a las mismas.

```
1 public class Ejemplo4 {
2 static double a=10;
3 static {
4 int a = 100;
5 System.out.print(duplica(a)+" ");
6 }
7 static int duplica(double a) { return (int)(a *= 2); }
8 public static void main(String[] args) {
9 System.out.print(duplica(a)+" ");
10 }
11 }
```

- A) 20 200
- B) 200 20
- C) Se obtiene error en la línea número 5.
- D) Ninguno de las anteriores.

16. Indique cuál es el resultado de la ejecución del método main de la clase siguiente.

```
public class Ejemplo5 {
 public static void main(String[] args) {
 int x[] = { 1, 2, 3}, y=0; double z;
 try {
 z = x[x.length-1] / y; System.out.print(" P1");
 } catch (ArrayIndexOutOfBoundsException e1) {
 System.out.print(" P2"); y = 1;
 } catch (Exception e2) {
 System.out.print(" P3");
 }
 System.out.print(" P4");
 }
}
```

- A) P2 P4
- B) P3 P4
- C) P2 P1 P4
- D) Ninguna de las anteriores.

17. Indique cuál es el resultado de la ejecución del método main de la clase Ejemplo6. Las líneas se han numerado con el propósito de facilitar la referencia a las mismas.

```
1 public class Cuadrado {
2 double longitudLado;
3 Cuadrado(double L) { longitudLado = L; }
4 public String toString() { return "longitud del lado "+longitudLado; }
5 }
6
7 public class Ejemplo6 {
8 public static void main(String[] args) {
9 Cuadrado c1 = new Cuadrado(2);
10 String s = "Dimensión: "+c1;
11 System.out.println(s);
12 }
13 }
```

- A) Se obtiene error en la línea 4, ya que la clase Cuadrado no puede sobrescribir el método toString()
- B) Se obtiene error en la línea 10, ya que el objeto c1 no es String.
- C) Las dos respuestas anteriores son verdaderas.
- D) Dimensión: longitud del lado 2.0

18. Indicar qué afirmación es correcta:

- A) El nombre de una variable puede empezar con cualquier carácter
- B) El nombre de una variable está limitado a 18 caracteres
- C) El nombre de una variable no debe tener espacios en blanco
- D) Ninguna de las anteriores

19. Indicar cuál de las siguientes declaraciones es correcta:

- A) char A='A';
- B) byte peso=300+1;
- C) byte peso=(byte)(300+1);
- D) Ninguna de las anteriores.

20. Cuál de las siguientes afirmaciones es falsa:

- A) Los argumentos no basados en tipos primitivos de datos son pasados por referencia.
- B) Los argumentos pasados por valor nunca quedan modificados después de la ejecución de los métodos.
- C) Los argumentos pasados por referencia pueden ser modificados en los métodos llamados.
- D) Todas son falsas.