

INSTRUCCIONES:

- El EXAMEN TEÓRICO se divide en dos parciales. El PRIMER PARCIAL corresponde a las unidades 1 a 5. El SEGUNDO PARCIAL corresponde a las unidades 6 a 9.
- Aquellos alumnos que aprobaron el examen de FEBRERO no están obligados a realizar este PRIMER PARCIAL.

LAS RESPUESTAS A TODAS LAS PREGUNTAS SE DARÁN EN LA TABLA DE RESPUESTAS. NO SE TENDRÁ EN CUENTA NINGUNA RESPUESTA FUERA DE ESTA TABLA.

EVALUACIÓN PRIMER PARCIAL

La puntuación total del examen TEÓRICO + PRÁCTICO será de 10 puntos. Parte teórica: 3.5 puntos. Parte Práctica: 6.5 puntos.

- El examen teórico consta de 25 preguntas tipo test (2.5 puntos) más una pregunta de conocimientos prácticos (1 punto).
- Cada pregunta correcta del tipo test vale 0.1 puntos y cada pregunta incorrecta -0.05 puntos.

RESPUESTAS A LAS PREGUNTAS TIPO TEST

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
d	b	a	d	b	d	b	b	a	c	c	b	b	d	a	b	b	b	c	a	b	c	b	b	d

RESPUESTA A LA PREGUNTA PRÁCTICA

26	SOLUCIÓN: 6
----	-------------

EVALUACIÓN SEGUNDO PARCIAL

La puntuación total del examen TEÓRICO + PRÁCTICO será de 10 puntos. Parte teórica: 3.5 puntos. Parte Práctica: 6.5 puntos.

- El examen teórico consta de 25 preguntas tipo test (2.5 puntos) más cuatro preguntas de conocimientos prácticos (1 punto).
- Cada pregunta correcta del tipo test vale 0.1 puntos y cada pregunta incorrecta -0.05 puntos.

RESPUESTAS A LAS PREGUNTAS TIPO TEST

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
b	c	b	c	b	b	a	a	c	a	a	c	b	c	c	b	b	c	b	c	d	c	c	c	b

RESPUESTAS A LAS PREGUNTAS PRÁCTICAS

26a	<code>protected abstract double getSalario();</code>
26b	<code>public static final double SALARIO_MINIMO=635.0;</code>
27a	<code>this(nombre);</code> <code>this.sueldoBase = sueldo</code>
28a	<code>super(nombre, sueldo);</code> <code>this.complementoSueldo = complemento;</code>



Módulo: PROGRAMACIÓN

I.E.S.: AGUADULCE

C.F.G.S. : DESARROLLO DE APLICACIONES WEB

Fecha: 13 DE JUNIO DE 2012

Hora de Comienzo: 15:30 h

Examen: TEÓRICO JUNIO

Duración: 2:30 h

Nombre:

D.N.I.:

Centro en el que se realiza el examen:

28b	<code>Empleado cajero1 = new Cajero("Pepico Pérez", 1001.0, 300);</code>
29a	<code>String imprime = super.toString();</code>



PRIMER PARCIAL

- Un lenguaje compilado recibe también el nombre de _____ y debe ser traducido a un código que pueda entender la máquina.
 - Lenguaje ensamblador
 - Lenguaje interpretado
 - Lenguaje de bajo nivel
 - Lenguaje de alto nivel**
- ¿Qué orden debemos introducir en línea de comandos para poder obtener un archivo .class, si tuviéramos un archivo llamado programa.java?
 - javac.exe
 - javac programa.java**
 - java programa.java
 - ./java programa.class
- ¿Cuál de los siguientes entornos no es de pago?
 - BlueJ.**
 - JBuilder.
 - JDeveloper.
 - IntelliJ IDEA.
- Si la solución a un problema se hace en un tiempo mínimo y de manera óptima, decimos que esta solución es _____, por el uso correcto de los recursos del sistema.
 - Fiable.
 - Eficaz.
 - Correcto.
 - Eficiente.**
- ¿En cuál de las fases de la programación se realiza la compilación del programa?
 - En la fase de resolución del problema.
 - En la fase de implementación.**
 - En la fase de explotación.
 - En la fase de mantenimiento.
- El operador _____ se usa para la toma de decisiones.
 - ?:
 - :
 - ?;
 - ?:**
- Indica los valores de x y z después de las siguientes sentencias:
int x = 12;
int z = ++x%7;
 - x es 13 y z es 5.
 - x es 13 y z es 6.**
 - x es 12 y z es 6.
 - x es 12 y z es 5.
- Señala cuál no es un tipo primitivo en Java
 - short.
 - string.**
 - double.
 - boolean.



9. En la definición de una clase debemos tener en cuenta que:
- Se deben incluir los atributos comunes del conjunto de objetos y los métodos que operan sobre ellos.**
 - Crearemos la clase con la palabra reservada `classes`.
 - El archivo de la clase debe tener el mismo nombre que el método que contenga dicha clase.
 - Todas son ciertas.
10. De las siguientes afirmaciones referidas a los métodos, señala cuál es la correcta:
- Los atributos de instancia junto con los métodos de instancia reciben el nombre de miembros de clase.
 - Cualquier método puede no devolver un valor, en cuyo caso se indica sin utilizar ninguna palabra reservada.
 - La lista de parámetros de un método debe coincidir con la lista de argumentos con los que es llamado.**
 - Todas son correctas.
11. De las siguientes afirmaciones referidas a los métodos, señala cuál es la correcta:
- Los atributos de instancia junto con los métodos de instancia reciben el nombre de miembros de clase.
 - Cualquier método puede no devolver un valor, en cuyo caso se indica sin utilizar ninguna palabra reservada.
 - La lista de parámetros de un método debe coincidir con la lista de argumentos con los que es llamado.**
 - Todas son correctas.
12. Una variable local almacena un valor temporal y se declara dentro de
- Una clase.
 - Un método.**
 - Un tipo de datos.
 - Un bloque de código entre corchetes.
13. Las cadenas de caracteres se representan mediante la clase
- Array.
 - String.**
 - Scanner.
 - Math.
14. Señala cuál es la correcta de las siguientes definiciones referidas a clases y objetos:
- Toda clase es una instancia de un único objeto.
 - Un programa orientado a objetos es una colección estructurada de objetos que definen los distintos tipos de clases que van a intervenir en la resolución del problema.
 - Toda clase que forma parte del programa tiene, en un instante dado, uno o más objetos que son instancia de ella.
 - Un programa orientado a objetos está compuesto por un conjunto de objetos que son representaciones del mundo real y que interactúan entre sí para la resolución de un problema.**
15. Indica cuál es equivalente al operador condicional de Java:
- if o if-else.**
 - switch.
 - if, if-else y switch.
 - Todas las respuestas son correctas.



16. ¿Qué es necesario incluir en cada conjunto de sentencias asociadas a los posibles valores que pueden tomarse en un switch?
- Puntos y comas.
 - Una sentencia break.**
 - Una sentencia continue.
 - Una cláusula default.
17. ¿En qué bucle se lleva a cabo la inicialización de una variable en su cabecera?
- En el bucles for/in.
 - En el bucle for.**
 - En el bucle while.
 - En el bucle do-while.
18. ¿Qué tipo de estructura no lleva a cabo ningún tipo de comprobación lógica?
- Las estructuras de selección.
 - Las secuencias.**
 - Las estructuras de iteración.
 - Las secuencias repetitivas.
19. Para declarar una nueva clase se utiliza la palabra reservada:
- new.
 - object.
 - class.**
 - clasdef.
20. `if (numero % 2 == 0) System.out.print("El número es par /n");`
- Muestra el mensaje por pantalla cuando el número almacenado en la variable número es par.**
 - Muestra el mensaje por pantalla cuando al dividir el valor de la variable número entre 2 obtenemos cero como resultado.
 - No muestra el mensaje por pantalla, ya que la condición del if nunca se cumplirá.
 - Ninguna respuesta es correcta.
21. ¿Qué tipo es devuelto por un constructor?
- void.
 - No devuelve ningún tipo (ni siquiera void).**
 - Depende de si el constructor está sobrecargado o no.
 - El mismo tipo que el atributo principal de la clase.
22. Los métodos especiales que permiten la creación de un objeto y que tienen el mismo nombre que la clase a la que pertenecen son conocidos como...
- Iniciadores.
 - Compiladores.
 - Constructores.**
 - Extractores.
23. ¿Es posible utilizar una return en cualquier punto de un método?, con lo que éste finalizará en el lugar donde se encuentre dicho return. ?
- No, siempre debe ir al final del método.
 - Sí y hará que éste finalice en el lugar donde se encuentre el return.**
 - Sí y podemos añadir tantos return como necesitemos.
 - No, return sólo se incluirá en aquellos métodos que devuelven void.



24. ¿Qué modificadores habría que añadir a un método que se desea que sea accesible desde fuera de la clase y que no haga falta que existan objetos de la clase para poder ser utilizado?
- a. protected final.
 - b. public static.**
 - c. private static.
 - d. protected.
25. ¿Qué palabra reservada hay que utilizar para llamar a un constructor de una clase?
- a. build.
 - b. create.
 - c. free.
 - d. new.**

26. Dado el siguiente código en Java, indica el resultado imprimido en pantalla. (1 punto)

```
public class Sumas{
    public static void main(String[] args){
        int suma;
        for (int i=1;i<=10;i++){
            suma = 0;
            for (int j=i-1; j>=1; j--){
                if (i%j==0){
                    suma=suma+j;
                }
            }
            if (suma==i){
                System.out.print(i+" ");
            }
        }
    }
}
```



SEGUNDO PARCIAL

1. Para averiguar la codificación que posee un fichero, podemos usar:
El método Unicode().
El método getEncoding().
El método getCodification().
El método getCharacter().
2. Para averiguar si un elemento es un directorio usamos el método:
a. mkdirs.
b. mkdir.
c. isDirectory.
d. isFolder.
3. Al método setLayout()
a. Se le puede indicar si se desea que la aplicación sea portable o no.
b. Se le pasa como argumento un objeto del tipo de Layout que se quiere establecer.
c. Se le pasa como argumento un String para indicarle la forma del layout que se quiere establecer.
d. Ninguna es cierta.
4. Un botón JButton y un JToggleButton
a. Son lo mismo.
b. Son botones, pero el primero se comporta como un interruptor de dos posiciones.
c. Son botones, pero el segundo se comporta como un interruptor de dos posiciones.
d. Todas son correctas.
5. Swing es
a. Un componente de SWT.
b. una librería de Java para la generación del GUI en aplicaciones.
c. Una librería de NetBeans.
d. Ninguna afirmación es correcta.
6. Si al intentar acceder a un fichero, no existe se generará una:
a. RMIException.
b. IOException.
c. SQLException.
d. FileNotFoundException.
7. Las casillas de verificación en Swing están implementadas para Java por la clase:
a. JCheckBox
b. JScrollPane
c. JSeparator
d. JButton
8. El componente Swing que dibuja una línea horizontal en el menú es:
a. JSeparator
b. JSeparator
c. JDistinct
d. JHorLine
9. La capacidad de las estructuras denominadas dinámicas...
a. es infinita.
b. se establece en el momento de la creación.
c. crece conforme insertamos nuevos elementos.
d. depende de los elementos que se inserten.



10. ¿Cuál de las siguientes expresiones encajan con la expresión regular "[A-Z0-9]+0-9"?
- "AZ090-9"
 - "0AZZ-9"
 - "AAA09"
 - "ABC9+0"
11. A continuación, se muestra un listado de métodos que permiten comprobar, a través de la clase `Matcher`, si una cadena encaja con un patrón, ¿cuál de ellos debe usarse para hacer uso de los métodos `start` y `end`, también disponibles en la clase `Matcher`? (Imagina que `m` es una instancia de la clase `Matcher`.)
- `m.find()`**
 - `m.search()`
 - `m.lookingAt()`
 - `m.matches()`
12. Dado el array `int j[]={1,2,3,4,5,6};`, ¿cuál es el elemento en la posición 3?
- 2
 - 3
 - 4**
 - La inicialización del array es incorrecta.
13. ¿Cuál de las siguientes afirmaciones sobre el método `toString()` es falsa?
- Está disponible en cualquier clase de Java.
 - Los tipos de datos primitivos, `int`, `long`, etc. pueden pasarse a cadena con este método directamente.**
 - Sirve para convertir un objeto a cadena, es especialmente útil en las clases envoltorio de los datos primitivos.
 - Este método está disponible en la clase `String`.
14. ¿Cuáles de las siguientes especificaciones de formato para el método `format` genera un número con dos decimales?
- `System.out.println(String.format("%.2d",2));`
 - `System.out.println(String.format("%2s",2));`
 - `System.out.println(String.format("%.2f",2));`**
 - `System.out.println(String.format("%2b",2));`
15. En Programación Orientada a Objetos, ¿con qué nombre es conocido el mecanismo que permite crear clases basadas en otras existentes?
- Polimorfismo.
 - Derivación.
 - Herencia.**
 - Encapsulación.
16. ¿Para qué estructura existe la herencia múltiple en Java?
- Para clases.
 - Para interfaces.**
 - Para clases que implementen la interfaz `Multiple`.
 - En ningún caso.
17. ¿Qué modificadores incluyen implícitamente los métodos de una interfaz en Java y por tanto no es necesario indicarlos?
- `protected` y `final`.
 - `public` y `abstract`.**
 - `public` y `final`.
 - `protected` y `abstract`.



18. Cuando una clase está definida dentro de otra, ¿qué tipo de relación se suele decir que existe entre esas dos clases?
- Herencia.
 - Derivación.
 - c. Anidación.**
 - Composición.
19. ¿Con qué nombre son conocidas aquellas clases cuya única función es la de ser superclase en una jerarquía, sin que llegue a haber nunca instancias de ellas?
- clases básicas.
 - b. clases abstractas.**
 - clases jerárquicas.
 - Ese tipo de clases no tienen sentido y no existen en Java.
20. ¿Cuál es la palabra reservada que se utiliza para indicar la herencia múltiple de clases en Java?
- extendsMultiple.
 - inherits.
 - c. Java no soporta la herencia múltiple de clases.**
 - isSubClass.
21. ¿Qué palabra reservada hay que utilizar en Java para referirse a la superclase de la clase actual?
- superclass.
 - that.
 - this.
 - d. super.**
22. ¿Qué hay que hacer en Java para crear un objeto polimórfico?
- Utilizar la palabra reservada polymorphic.
 - Declarar una variable como referencia a un objeto de una clase determinada y posteriormente asignar a esa variable referencias a objetos de otras clases diferentes.
 - c. Declarar una variable como referencia a un objeto de una clase determinada que tenga clases derivadas y así posteriormente se podrán asignar a esa variable referencias a objetos de subclases de la clase referencia inicial.**
 - En Java no es posible el polimorfismo.
23. Dada la expresión regular "[A-Z]*([a-z]*)([0-9]+)", al usar el método find() de la clase Matcher sobre la cadena "AABBccdd1234", ¿cómo puedes extraer las letras en minúsculas? (Imagina que m es la instancia de la clase Matcher).
- No es posible, dado que la cadena no encaja con el patrón dado.
 - m.group(1)
 - c. m.group(2)**
 - m.get(1)
24. ¿Cuáles de los siguientes métodos nos permiten insertar elementos de un TreeSet?
- append()
 - insert()
 - c. add()**
 - offer()
25. ¿Cuál de las siguientes afirmaciones sobre documentos XML DOM es falsa?
- Tienen un único elemento raíz.
 - b. Los atributos pueden estar dentro de comentarios (clase Comment) y dentro de elementos (clase Element).**
 - Puede haber elementos (clase Element) dentro de otros elementos.
 - Un documento XML DOM es una estructura jerárquica donde todos los elementos extienden la clase Node.



26. Dado el siguiente código en Java:

```
abstract class Empleado{
    private String nombre;
    private double sueldoBase;

    public Empleado(String nombre) {
        this.nombre=nombre;
        sueldoBase=0;
    }
    public String toString(){
        String imprime = "Nombre: " + this.nombre + "Sueldo base: " + this.sueldoBase;
        return imprime;
    }
}

class Cajero extends Empleado{
    private int complementoSueldo;
}

class Interventor extends Empleado{
    private double comisionVentas;
}
```

- a. **Escribe la sentencia que defina un método abstracto para la clase Empleado, que se llame getSalario, que devuelve el tipo double, y que sea accesible solo por las subclases que pertenezcan al mismo paquete que la clase Empleado.**
- b. **Escribe la sentencia que defina una constante de clase pública llamada SALARIO_MINIMO, de tipo double inicializada a 635.0.**

27. Dada la siguiente definición de la cabecera de otro constructor para la clase Empleado:

```
public Empleado(String nombre, double sueldo) {
    //cuerpo del constructor
}
```

- a. **Escribe las sentencias necesarias en el cuerpo para completar la definición, utilizando para ello una llamada al primer constructor de la clase.**



28. Dada la siguiente definición de la cabecera del constructor para la clase Cajero:

```
public Cajero(String nombre, double sueldo, int complemento) {  
    //cuerpo del constructor  
}
```

- a. Escribe las sentencias necesarias en el cuerpo para completar la definición, utilizando para ello una llamada al segundo constructor de la clase padre.
- b. Utiliza el constructor que acabas de completar, para crear un objeto de tipo *Cajero* llamado *cajero1* cuyo nombre es *Pepico Pérez*, cuyo sueldo base *1001.0* y cuyo complemento es *300*. El objeto creado será referenciado por una variable de tipo *Empleado*.

29. Necesitamos que el método `toString` de la clase *Cajero* heredado de la clase *Empleado* devuelva también el valor del atributo `complementoSuelto` dentro de la cadena de caracteres. Dada la siguiente definición de la cabecera y el cuerpo del método:

```
public String toString(){  
    // Sentencia que falta  
    imprime = imprime + "Complemento del sueldo: " + this.complementoSuelto;  
    return imprime;  
}
```

- a. Escribe la sentencia necesaria que falta en el cuerpo del método que declare la variable `imprime` de tipo `String` y a la que se le asigne la cadena de caracteres devuelta en la llamada al método `toString` de la clase *Empleado*.

