

# Prueba escrita 9 de Octubre de 2025

## Unidad 2: Condicionales

RA1. Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.

- d) Se han identificado los distintos tipos de variables y la utilidad específica de cada uno.
- e) Se ha modificado el código de un programa para crear y utilizar variables.
- i) Se han introducido comentarios en el código.

RA2. Escribe y prueba programas sencillos, reconociendo y aplicando los fundamentos de la programación orientada a objetos.

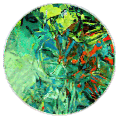
- b) Se ha identificado la sintaxis del lenguaje de programación.

RA3. Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.

- a) Se ha escrito y probado código que haga uso de estructuras de selección.
- e) Se han creado programas ejecutables utilizando diferentes estructuras de control.

RA5. Realiza operaciones de entrada y salida de información, utilizando procedimientos específicos del lenguaje y librerías de clases.

- a) Se ha utilizado la consola para realizar operaciones de entrada y salida de información.



## 1. Posible solución

(2,5 puntos) Ejercicio 1: Idear un programa que solicite al usuario un número comprendido entre 1 y 7, correspondiente a un día de la semana. Se debe mostrar el nombre del día de la semana al que corresponde. Por ejemplo, el número 1 corresponde a «lunes» y el 6 a «sábado».

```
package examen;
import java.util.Scanner;
public class Ejercicio1 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        // Solicita un número del 1 al 7 y muestra el día correspondiente
        System.out.print("Introduce un número (1-7): ");
        int n = sc.nextInt();
        String dia;
        switch (n) {
            case 1 -> dia = "lunes";
            case 2 -> dia = "martes";
            case 3 -> dia = "miércoles";
            case 4 -> dia = "jueves";
            case 5 -> dia = "viernes";
            case 6 -> dia = "sábado";
            case 7 -> dia = "domingo";
            default -> dia = "Valor fuera de rango (debe ser 1-7)";
        }
        System.out.println(dia);
        sc.close();
    }
}
```

(2,5 puntos) Escribe un programa en Java que solicite al usuario su **edad** y determine si puede **entrar a una sala de cine**.

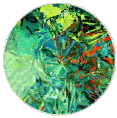
- Si tiene **menos de 13 años**, no puede entrar solo.
- Si tiene **entre 13 y 17 años**, puede entrar acompañado.
- Si tiene **18 o más**, puede entrar sin restricciones.

El programa debe mostrar un **mensaje personalizado** según el caso.

Debes usar:

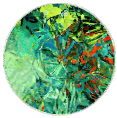
- **Una constante literal de texto** para el nombre del cine.
- **Variables** para almacenar la edad y el mensaje.
- **Estructuras condicionales** para decidir la salida.

```
package examen;
import java.util.Scanner;
public class Ejercicio2 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        // Variables
        // Nombre del cine
        final String CINE = "Cine Estrella";
```



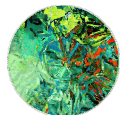
```
String mensaje;
int edad;
System.out.print("Introduce tu edad: ");
edad= sc.nextInt();

if (edad < 0) {
    mensaje = "Edad no válida.";
} else if (edad < 13) {
    // Menores de 13 años deben venir con su representante legal.
    mensaje = "No puedes entrar solo.";
} else if (edad <= 17) {
    // Entre 13 y 17 años pueden entrar acompañados (por un amigo
    o adulto).
    mensaje = "Puedes entrar acompañado.";
} else {
    // Mayores de edad: sin restricciones.
    mensaje = "Puedes entrar sin restricciones.";
}
System.out.println("[ " + CINE + " ] " + mensaje);
sc.close();
}
```



**(5 puntos) Ejercicio 3: Selecciona la respuesta correcta**

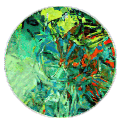
<p><b>2.1. Los operadores lógicos operan con valores booleanos, resultando:</b></p> <p>a) Valores enteros. b) Valores enteros y booleanos. c) Otros tipos de valores. <b>d) Sólo valores booleanos.</b></p>	<p><b>2.2. La evaluación de una expresión relacional puede generar un valor de tipo:</b></p> <p>a) Entero. b) Real. <b>c) Booleano.</b> d) Todos los anteriores.</p>		
<p><b>2.3. La expresión</b> <b>3 == 3 &amp;&amp; 2 &lt; 3 &amp;&amp; 1 != 2</b> <b>resulta:</b> <b>a) Cierto.</b> b) Falso. c) No se puede evaluar. d) No genera un booleano, ya que la expresión es aritmética.</p>	<p><b>2.4. La siguiente expresión, donde interviene la variable booleana a:</b> <b>3 != 3    a    1 &lt; 2</b> <b>resulta:</b> <b>a) Dependerá del valor a.</b> b) Cierto. c) Falso. d) No se puede evaluar.</p>		
<p><b>2.5. Elige los valores de las variables enteras (a, b y c) que permiten que la evaluación de la siguiente expresión sea cierta:</b> <b>a &lt; b &amp;&amp; b != c &amp;&amp; b &lt;= c</b> a) a = 1, b = 1, c = 2. b) a = 1, b = 2, c = 2. <b>c) a = 1, b = 2, c = 3.</b> d) a = 2, b = 2, c = 3.</p>	<p><b>2.6. El bloque de instrucciones de una sentencia if se ejecutará:</b> a) Siempre. b) Nunca. <b>c) Dependerá de la evaluación de la expresión utilizada.</b> d) Todas las respuestas anteriores son correctas.</p>		
<p><b>2.7. En una sentencia if-else los bloques de instrucciones (bloque true y bloque false) pueden ejecutarse:</b> a) Simultáneamente. b) Siempre. c) Si es posible, dependiendo de la condición utilizada, que no se ejecute ninguno. <b>d) Sólo se ejecuta al menos uno y son excluyentes.</b></p>	<p><b>2.8. Realiza una traza del siguiente fragmento de código y selecciona el valor que toma finalmente la variable a:</b></p> <table><tr><td><pre>a = 0; switch (a) {     case 0:         a = 1;     case 1:         a = 3;     case 2:         a++;         break;     case 3:         a--;         break; }</pre></td><td><p>a) 1 b) 2 c) 3 <b>d) 4</b></p></td></tr></table>	<pre>a = 0; switch (a) {     case 0:         a = 1;     case 1:         a = 3;     case 2:         a++;         break;     case 3:         a--;         break; }</pre>	<p>a) 1 b) 2 c) 3 <b>d) 4</b></p>
<pre>a = 0; switch (a) {     case 0:         a = 1;     case 1:         a = 3;     case 2:         a++;         break;     case 3:         a--;         break; }</pre>	<p>a) 1 b) 2 c) 3 <b>d) 4</b></p>		
<p><b>2.9. La cláusula default en la sentencia switch es:</b> a) Obligatoria y tiene que ser la última que aparezca. b) Obligatoria, pero puede aparecer en cualquier lugar. c) Opcional y tiene que ser la última que aparezca. <b>d) Opcional y puede usarse en cualquier lugar.</b></p>			



**C.F.G.S. Desarrollo de Aplicaciones Web/Multiplataforma**

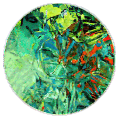
**MÓDULO: Programación**

IES CAMAS Curso 2025/26



## 2. Notas

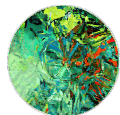
	2'5 puntos							2'5 puntos							5 puntos										
	Ejercicio 1							Ejercicio 2							Ejercicio 3									Total	
	a	b	c	d	e	f	g	a	b	c	d	e	f	g	1	2	3	4	5	6	7	8	9		
ialcrod1802	1	1	0,5	1	1	1	0	1	1	0,5	1	1	1	0	1	0	1	0	1	1	1	0	1	7,6	ialcrod1802
jblagon1804	0,5	0,75	0,5	0,5	0,5	0,5	0	0	0	0	0	0	0	0	1	0	1	1	1	1	0	1	0	4,6	jblagon1804
jcalgar480	1	0,75	1	1	1	1	0	1	0,75	1	1	1	1	0	1	0	1	0	1	1	1	0	0	7,0	jcalgar480
lcascan0310																								NP	lcascan0310
acobfra1611	1	0,75	0,5	1	1	1	1	1	0,75	0,5	1	1	1	1	1	1	1	0	1	1	1	1	0	8,4	acobfra1611
mdiajim407	1	0,75	0,5	1	1	1	0	1	0,75	0,5	1	1	1	0	1	0	1	0	1	1	1	0	0	6,8	mdiajim407
adorcum696	0,5	0,5	0,5	0,5	0,5	0,5	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	2,8	adorcum696
rfrabar2206	1	0,75	0,5	1	1	1	1	0,75	0,75	0,5	1	1	1	1	0	0	1	1	1	0	1	0	0	6,5	rfrabar2206
ggarlob532	0,5	0,75	0,5	1	1	1	1	0,5	0,75	0,5	1	1	1	1	1	0	1	0	1	1	1	1	1	7,6	ggarlob532
sherace1405	1	0,75	0,5	1	1	1	0	1	0,75	0,5	1	1	1	0	0	0	1	1	1	1	1	0	1	7,3	sherace1405
mhuecob081	1	1	1	1	1	1	0	1	1	1	1	1	1	0	1	1	1	0	1	1	1	0	0	7,8	mhuecob081
gllepen645	0	0	0,5	1	1	1	0	0	0	0,5	1	1	1	0	1	1	1	1	1	1	1	0	0	5,6	gllepen645
amacpac0204	0,5	0,5	0	1	0,75	0,5	0	0,5	0,5	0	1	0,75	0,5	0	1	0	1	0	1	1	1	0	0	5,2	amacpac0204
amarloz3112	1	0,75	0,5	1	1	1	0	1	0,75	0,5	1	1	1	0	0	0	1	0	1	1	0	0	0	5,7	amarloz3112
amarboc																								NP	amarboc
rrodsan855	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	1	0	2,8	rrodsan855
asanesp2907	0	0,5	0,5	1	1	1	1	0	0,5	0,5	1	1	1	1	0	0	1	1	1	1	1	0	0	5,5	asanesp2907
jsersan1204	1	1	1	1	1	1	0	1	1	1	1	1	1	0	1	0	1	0	1	1	1	0	0	7,3	jsersan1204
msieufa855	1	1	1	1	1	1	0	1	0,75	0,5	1	1	1	0	1	0	1	1	1	1	0	0	0	7,0	msieufa855
ptormar1403	1	0,5	0,5	1	1	1	0	1	1	0,5	1	1	1	0	1	1	1	1	1	1	1	0	0	7,9	ptormar1403



sfad

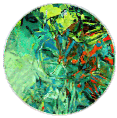
### 3. Comentarios

RA1.i). -> Hay que comentar el código RA2.b) Falta el identificador del nombre de la clase. En Java, tras la palabra clave <code>class</code> debe ir un nombre válido (identificador), y si la clase es <code>public</code> debe coincidir con el nombre del archivo.	ialcrod1802
RA1.i). -> Hay que comentar el código. RA2.b) Falta el identificador del nombre de la clase. En Java, tras la palabra clave <code>class</code> debe ir un nombre válido (identificador), y si la clase es <code>public</code> debe coincidir con el nombre del archivo. 0.5 RA3. e:). -> Se sugiere guardar el texto en una variable intermedia (por ejemplo, <code>mensaje</code> ) y mostrarlo al final, en un único punto de salida, para mejorar la claridad y mantenimiento del código	jblagon1804
RA1.i). -> Hay que comentar el código. RA3. e:). -> Se sugiere guardar el texto en una variable intermedia (por ejemplo, <code>mensaje</code> ) y mostrarlo al final, en un único punto de salida, para mejorar la claridad y mantenimiento del código	jcalgar480
	lcascan0310
RA2.b) Falta el identificador del nombre de la clase. En Java, tras la palabra clave <code>class</code> debe ir un nombre válido (identificador), y si la clase es <code>public</code> debe coincidir con el nombre del archivo. RA3. e). -> Se sugiere guardar el texto en una variable intermedia (por ejemplo, <code>mensaje</code> ) y mostrarlo al final, en un único punto de salida, para mejorar la	acobfra1611

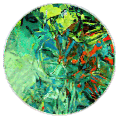


claridad y mantenimiento del código	
<p>RA2.b) Falta el identificador del nombre de la clase. En Java, tras la palabra clave <code>class</code> debe ir un nombre válido (identificador), y si la clase es <code>public</code> debe coincidir con el nombre del archivo.</p> <p>RA3. e). -&gt; Se sugiere guardar el texto en una variable intermedia (por ejemplo, <code>mensaje</code>) y mostrarlo al final, en un único punto de salida, para mejorar la claridad y mantenimiento del código</p> <p>RA1.i). -&gt; Hay que comentar el código.</p>	mdiajim407
<p>RA5.a) → En el ejercicio 1 no lees desde teclado, y no finaliza imprimiendo el resultado.</p> <p>RA1.i). -&gt; Hay que comentar el código.</p>	adorcum696
<p>RA2.b) -&gt;Falta el identificador del nombre de la clase. En Java, tras la palabra clave <code>class</code> debe ir un nombre válido (identificador), y si la clase es <code>public</code> debe coincidir con el nombre del archivo</p> <p>RA3. e: ). -&gt; Se sugiere guardar el texto en una variable intermedia (por ejemplo, <code>mensaje</code>) y mostrarlo al final, en un único punto de salida, para mejorar la claridad y mantenimiento del código.</p>	rfrabar2206
<p>RA2.b) Falta el identificador del nombre de la clase. En Java, tras la palabra clave <code>class</code> debe ir un nombre válido (identificador), y si la clase es <code>public</code> debe coincidir con el nombre del archivo.</p> <p>RA3. e). -&gt; Se sugiere guardar el texto en una variable intermedia (por ejemplo, <code>mensaje</code>) y mostrarlo al final, en un único punto de salida, para mejorar la claridad y mantenimiento del código</p> <p>En el ejercicio 1, dentro del <code>switch</code>, faltan las sentencias <code>break</code> ; en los distintos <code>case</code>. Esto puede provocar que el flujo de ejecución continúe hacia el siguiente caso de forma no intencionada</p> <p>Lo correcto sería</p>	ggarlob532

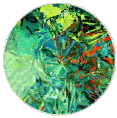




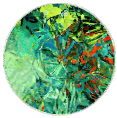
<pre>switch (dia) { case 1:     mensajeDiaSemana="Lunes";     break; case 2:     mensajeDiaSemana ="Martes";     break; ... default:     mensajeDiaSemana ="Valor fuera de rango"; } System.out.println(mensajeDiaSemana)</pre>	
<p>RA2.b) -&gt;Falta el identificador del nombre de la clase. En Java, tras la palabra clave <code>class</code> debe ir un nombre válido (identificador), y si la clase es <code>public</code> debe coincidir con el nombre del archivo</p> <p>RA3. e). -&gt; Se sugiere guardar el texto en una variable intermedia (por ejemplo, <code>mensaje</code>) y mostrarlo al final, en un único punto de salida, para mejorar la claridad y mantenimiento del código.</p> <p>RA1.i). -&gt; Hay que comentar el código</p>	sherace140 5
<p>RA1.i). -&gt; Hay que comentar el código</p>	mhuecob08 1
<p>RA2.b) RA3.a RA3.e) -&gt; Has puesto:</p> <pre>if (n &gt; 1) &amp;&amp; (n&lt;=7)     Switch(n) {</pre> <p>- Evita redundancia en el código:</p> <p>Usa solo una estructura de selección para la misma lógica: o bien <code>if/else</code>, o bien <code>switch</code>, pero no ambas. Esto mejora la claridad,</p> <p>O usas una estructura u otra</p> <p>RA2.b) Falta el identificador del nombre de la clase. En Java, tras la palabra clave <code>class</code> debe ir un nombre válido (identificador), y si la clase es <code>public</code> debe coincidir con el nombre del archivo.</p>	gllepen645



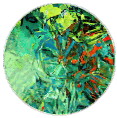
RA1.i). -> Hay que comentar el código.	
<p>RA1.i). -&gt; Hay que comentar el código</p> <p>RA2.b) Falta el identificador del nombre de la clase. En Java, tras la palabra clave <code>class</code> debe ir un nombre válido (identificador), y si la clase es <code>public</code> debe coincidir con el nombre del archivo.</p> <p>RA2.b) Has escrito:</p> <pre>... }else(num 3){ String.println("es martes");</pre> <p>No es correcto. Recuerda que la <b>condición</b> sólo puede evaluarse como "<i>cierta</i>" o "<i>falsa</i>".</p> <p>Si no se cumple, el programa puede evaluar otra condición mediante <code>else if</code>, y si ninguna se cumple, ejecutará el bloque <code>else</code></p> <pre>if(&lt;condición&gt;){ }else if(&lt;condición&gt;){ }else{ }</pre> <p>has escrito</p> <pre>case 1: a&gt;13    a &lt;=17 ...</pre> <p>No es correcto. Usa solo una estructura de selección para la misma lógica: o bien <code>if/else</code>, o bien <code>switch</code>, pero no ambas. Esto mejora la claridad, Lo correcto sería</p> <pre>switch (dia) { case 1:     syso("Lunes");     break; case 2:     syso("Martes");     break; default:     syso("Valor fuera de rango"); }</pre>	amacpac02 04



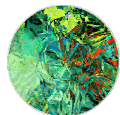
<p>RA2.b) Falta el identificador del nombre de la clase. En Java, tras la palabra clave <code>class</code> debe ir un nombre válido (identificador), y si la clase es <code>public</code> debe coincidir con el nombre del archivo.</p> <p>RA1.i). -&gt; Hay que comentar el código</p> <p>RA3. e ) -&gt; Se sugiere guardar el texto en una variable intermedia (por ejemplo, <code>mensaje</code>) y mostrarlo al final, en un único punto de salida, para mejorar la claridad y mantenimiento del código.</p>	amarloz311 2
	amarboc
<p>RA2.b) -&gt;Falta el identificador del nombre de la clase. En Java, tras la palabra clave <code>class</code> debe ir un nombre válido (identificador), y si la clase es <code>public</code> debe coincidir con el nombre del archivo.</p> <p>RA3.a) -&gt; Recuerda <code>&lt;condición&gt;</code> solo puede ser <code>'true'</code> o <code>'false'</code>.</p> <pre>if(&lt;condición&gt;){ }else if(&lt;condicion&gt;){ }else{ }</pre> <p>RA3. e). -&gt; Se sugiere guardar el texto en una variable intermedia (por ejemplo, <code>mensaje</code>) y mostrarlo al final, en un único punto de salida, para mejorar la claridad y mantenimiento del código</p>	rodsan855
<p>RA2.b) -&gt;Falta el identificador del nombre de la clase. En Java, tras la palabra clave <code>class</code> debe ir un nombre válido (identificador), y si la clase es <code>public</code> debe coincidir con el nombre del archivo.</p> <p>RA3. e). -&gt; Se sugiere guardar el texto en una variable intermedia (por ejemplo, <code>mensaje</code>) y mostrarlo al final, en un único punto de salida, para mejorar la claridad y mantenimiento del código</p> <p>Has escrito</p> <pre>if { edad &lt; 13 ){ ...</pre>	asanesp290 7



<pre>if { (edad &gt; 13 &amp;&amp; edad&lt;17);</pre> <p>No es correcto-</p> <p>Recuerda que la <b>condición</b> sólo puede evaluarse como “<i>cierta</i>” o “<i>falsa</i>”. Si no se cumple, el programa puede evaluar otra condición mediante <code>else if</code>, y si ninguna se cumple, ejecutará el bloque <code>else</code></p> <pre>if(&lt;condición&gt;){ }else if(&lt;condición&gt;){ }else{ }</pre>	
<p>RA1.i). -&gt; Hay que comentar el código</p>	jsersan1204
<p>RA3.a) RA2.b) -&gt; has escrito</p> <pre>if(switch (dia)) .. case 1 (syso ("Lunes")); case 2 (syso("Martes"));</pre> <p>No es correcto. Lo correcto sería</p> <pre>switch (dia) { case 1:     syso("Lunes");     break; case 2:     syso("Martes");     break; default:     syso("Valor fuera de rango"); }</pre> <p>Has escrito</p> <pre>if { edad &lt; 13 ; ... else if { (edad &gt; 13);</pre> <p>No es correcto-</p> <p>Recuerda que la <b>condición</b> sólo puede evaluarse como “<i>cierta</i>” o “<i>falsa</i>”. Si no se cumple, el programa puede evaluar otra condición mediante <code>else if</code>,</p>	msieufa855



<p>y si ninguna se cumple, ejecutará el bloque else</p> <pre>if(&lt;condición&gt;){   }else if(&lt;condición&gt;){   }else{   }</pre> <p>RA3. e). -&gt; Se sugiere guardar el texto en una variable intermedia (por ejemplo, mensaje) y mostrarlo al final, en un único punto de salida, para mejorar la claridad y mantenimiento del código.</p> <p>RA1.i). -&gt; Hay que comentar el código</p>	
<p>RA2.b) -&gt; Falta el identificador del nombre de la clase. En Java, tras la palabra clave <code>class</code> debe ir un nombre válido (identificador), y si la clase es <code>public</code> debe coincidir con el nombre del archivo.</p> <p>RA3.a) RA2.b) -&gt; has escrito</p> <pre>int numDia = sc.nextInt(); switch(numDia){   case 1 -&gt; "Lunes";   case 2 -&gt; "Martes";   ...   default -&gt; "Error"; };</pre> <p>No es correcto porque no lo guardas en ninguna variable. Lo correcto sería</p> <pre>int numDia = sc.nextInt(); String <b>nombreDia</b> = switch (numDia) {   case 1 -&gt; "Lunes";   case 2 -&gt; "Martes";   ...   default -&gt; "Valor fuera de rango"; };</pre> <p>RA1.i). -&gt; Hay que comentar el código</p>	<p>ptormar140 3</p>



## Ponderaciones

Reparto de 2,5 puntos (ajustado proporcionalmente):	Peso Ejercicio 1 y 2		Escala de logro
RA3.a: Se ha escrito y probado código que haga uso de estructuras de selección.	0,75	a	1 Excelente Cumple totalmente el criterio con precisión y claridad.
RA3. e: ) Se han creado programas ejecutables utilizando diferentes estructuras de control.	0,5	b	0,75 Adecuado Cumple de forma correcta, con mínimos errores.
RA2.b) Se ha identificado la sintaxis del lenguaje de programación.	0,25	c	0,5 Básico Cumple parcialmente o con fallos leves.
RA5.a) Se ha utilizado la consola para realizar operaciones de entrada y salida de información.	0,25	d	0 Insuficiente No alcanza el criterio o presenta errores graves.
RA1.d) Se han identificado los distintos tipos de variables y la utilidad específica de cada uno.	0,25	e	
RA1.e) Se ha modificado el código de un programa para crear y utilizar variables.	0,25	f	
RA1.i) Se han introducido comentarios en el código.	0,25	g	