

PRUEBAS DE ACCESO A LOS CICLOS FORMATIVOS DE GRADO SUPERIOR

Convocatoria de 18 de junio de 2025 (Resolución de 26 de marzo de 2025)

DATOS DEL ASPIRANTE		CALIFICACIÓN
Apellidos:		
Nombre:	D.N.I.:	

GRADO SUPERIOR – PARTE ESPECÍFICA OPCIÓN “B3” – PROGRAMACIÓN Y GESTIÓN DE DATOS

Mantenga su **DNI en lugar visible** durante la realización de la prueba.
Lea detenidamente los **enunciados** de las cuestiones.
 Cuide la presentación (orden, claridad y limpieza). **Destaque las soluciones.**
Duración de la prueba: 2 HORAS

Ejercicio 1

Supongamos que en cierto lenguaje de programación escribimos las siguientes sentencias (aparecen numeradas según el orden de ejecución):

1.	$a = 2$
2.	$b = 2*a+1$
3.	$a = a + b +1$
4.	$a = 2*a + b$

→ **[0,5 puntos]** Completa la siguiente tabla indicando los valores de las variables **a** y **b** tras la ejecución de cada una de las líneas de código:

Línea	Valor de a	Valor de b
1.		
2.		
3.		
4.		

Ejercicio 2

Supongamos que en cierto lenguaje de programación se cuenta con la instrucción **print**, que permite mostrar mensajes e información en pantalla. Por ejemplo, se pueden escribir expresiones que tendrán una salida en pantalla como las de los ejemplos:

Código	Salida en pantalla
<code>print("Hola")</code>	Hola
<code>a = 25 print(a)</code>	25
<code>b = 10 print("El valor de la variable b es ", b, ". Fin.")</code>	El valor de la variable b es 10. Fin.

En los que se muestra que se pueden escribir en pantalla cadenas de caracteres, el valor de variables y combinaciones de ambas separándolas mediante "comas".

Supongamos que contamos con la orden **input**, que permite la entrada de valores desde el teclado y guardarlos en una variable. Por ejemplo, con una orden como

```
x = input("Introduce un número")
```

aparecería en pantalla el mensaje **Introduce un número**. Si introducimos a través del teclado, por ejemplo, **12**, el resultado es que la variable **x** toma el valor **12**.

➔ **[1,25 puntos]** Con ayuda de las órdenes descritas en los párrafos anteriores, se quiere escribir un programa que permita calcular la longitud y la superficie de una circunferencia de radio dado. Para poder hacerlo, se recuerda que la longitud de una circunferencia es $2 \cdot \pi \cdot r$ y que la superficie es $\pi \cdot r \cdot r$. Con todo ello, se pide escribir un programa en el que (1) se defina una variable de nombre **Pi** con un valor de **3.14159**, (2) el programa muestre en pantalla un mensaje en el que se pida que se introduzca el radio de la circunferencia, (3) el número que se introduzca a través del teclado se guarde en la variable **Radio**, (4) el programa calcule la longitud y la superficie con las fórmulas dadas y guarde los resultados en las variables **Longitud** y **Superficie**, respectivamente. Finalmente, (5) el programa ha de mostrar en pantalla dos mensajes: uno que indique **La longitud de la circunferencia es: xxxxx**, donde **xxxxx** representa el valor guardado en la variable **Longitud** y otro mensaje semejante para indicar el valor resultante del cálculo de la superficie.

Ejercicio 3

Supongamos que en cierto lenguaje de programación se cuenta con las siguientes **sentencias de control**:

Forma 1	Forma 2	Forma 3
<pre>if (condición) { sentencia1 sentencia2 ... }</pre>	<pre>if (condición) { sentencia1 sentencia2 ... } else { sentencia3 sentencia4 ... }</pre>	<pre>if (condición1) { sentencia1 sentencia2 ... } else if (condición2) { sentencia3 sentencia4 ... } else { sentencias }</pre>

Qué significa en la **Forma 1** que, si se cumple la **condición**, se ejecutarán las sentencias encerradas entre llaves (en el ejemplo: **sentencia1**, **sentencia2** y las siguientes --si las hay--). En la **Forma 2**, que, si se cumple la **condición**, se ejecutará el grupo de sentencias encerradas entre el primer par de llaves y, si no se cumple, se ejecutarán las sentencias encerradas en las llaves tras el **else**. En la **Forma 3** se muestra cómo se pueden enlazar condiciones sucesivas excluyentes en este lenguaje.

Para escribir las condiciones, el lenguaje utiliza los signos: mayor >, menor <, el operador de igualdad == y el operador distinto !=. También puede hacer uso de los operadores mayor o igual >= y menor o igual <=.

➔ **[0,5 puntos]** De acuerdo con la descripción de la sentencia de control y teniendo en cuenta las instrucciones **print**, indica cómo será la salida en pantalla del siguiente programa:

Código	Salida en pantalla
<pre> a = 2 b = 10 if (a>b) { print("Divido la variable a entre b") c = a/b } else { print("Divido la variable b entre a") c = b/a } print("La división ha dado: ", c) </pre>	

➔ **[1,25 puntos]** Escribe un programa que realice las siguientes acciones: (1) muestre un mensaje en pantalla indicando que se introduzca un número, (2) lo lea y guarde en una variable y (3), haciendo uso de la una secuencia de condicionales en la **Forma 3**, escriba un mensaje en pantalla indicando si el número es menor que diez, igual a diez o mayor que diez.

Ejercicio 4

El lenguaje permite trabajar con listas de datos (también llamadas arrays). Por ejemplo, una variable que contiene una lista de números puede ser:

Lista = [1, 2, 4, 9]

Para acceder a los elementos de una variable de tipo lista se escribe el nombre de la variable y el número que indica la posición del elemento en la lista entre corchetes. Por ejemplo, **Lista[3]** toma el valor **4** porque el tercer elemento de la lista es un 4. Para cambiar el valor de un elemento de la lista se usan expresiones como, por ejemplo, **Lista[2] = 10**, que hace que el segundo elemento de la lista ahora valga **10**. Una orden como **print(Lista)** muestra en pantalla la lista completa incluyendo los corchetes de comienzo y final, es decir, para la variable **Lista** del ejemplo la salida en pantalla es: **[1, 10, 4, 9]**.

➔ **[0,75 puntos]** De acuerdo con las instrucciones sobre cómo operar con listas y teniendo en cuenta las instrucciones **print**, indica cómo será la salida en pantalla del siguiente programa:

Código	Salida en pantalla
<pre> A = [8, 7, 5, 2, 6] A[2] = 2 print(A) A[3] = A[3] + A[2] * A[5] print(A) print(A[1]+A[4]) </pre>	

Ejercicio 5

Bucles y repeticiones

El lenguaje permite escribir instrucciones que controlan la repetición de partes del programa, como, por ejemplo, la orden **for**, cuya sintaxis es:

```
for i from inicio to final by salto {  
  instrucciones a repetir  
}
```

Donde **i** es la variable que controla la repetición, **inicio** el primer valor que toma la variable dentro de la secuencia de repeticiones, **final** es el último valor y **salto** es la cantidad en que se va incrementando la variable **i** del ejemplo. Seguido, se muestra un pequeño programa que hace uso de la instrucción y su salida en pantalla.

Código	Salida en pantalla
for a from 2 to 11 by 3 { print(a) }	2 5 8 11

La orden **for** se puede usar en una forma abreviada si la variable de control se incrementa de uno en uno:

```
for i from inicio to final {  
  instrucciones a repetir  
}
```

- ➔ [1,25 puntos] Haciendo uso de la orden **for**, escribe un programa que sume los cuadrados de los números pares entre uno y cien (el cien incluido). Tras calcular la suma, el programa debe indicar un mensaje en pantalla indicado el resultado.

Ejercicio 6

El lenguaje también permite controlar el flujo del programa mediante la orden **while**, cuya sintaxis es:

```
while (condicion) {  
  instrucciones  
}
```

Donde **condicion** es una expresión condicional. Si la condición se cumple, el programa ejecuta las instrucciones entre llaves; si no, se las salta. Seguido, se muestra un pequeño programa que hace uso de la instrucción y su salida en pantalla.

Código	Salida en pantalla
i = 0 while(i<=3) { print(i) i = i + 1 }	0 1 2 3

El siguiente programa pide al usuario que adivine un número que tiene guardado en una variable. El usuario tiene que introducir un número repetidas veces, hasta que encuentre el que se busca. Para cada número que introduce, el programa le indica si el número que se busca es mayor, menor o igual que el que ha introducido. Pero el programa que se presenta no está del todo terminado...

1.	print("Encuentra el número entre 1 y 100 en el que estoy pensando")
2.	objetivo = 27
3.	seguir = 1
4.	while (seguir==1) {
5.	x = input("Introduce el número candidato: ")
6.	if(x==objetivo) {
7.	print("Has encontrado el número, efectivamente, es ", x)
8.	} else if (x<objetivo) {
9.	print("El número que buscas es mayor que ", x)
10.	} else {
11.	print("El número que buscas es mayor que ", x)
12.	}
13.	}
14.	print("Fin del programa.")

➔ Responde a las siguientes cuestiones:

- [0,5 puntos]** Observa las líneas 9 y 11. ¿Hay que hacer algún cambio en una de ellas? Si es así, ¿qué cambio es? Explicalo.
- [1 punto]** Indica la secuencia de mensajes en pantalla que va a mostrar el programa desde el momento en que se pone en marcha si el usuario introduce los siguientes números, en este orden: 70, 50, 25, 27. Responde a este apartado teniendo en cuenta la respuesta que has dado en el anterior.
- [0,5 puntos]** Puede que el programa esté mal diseñado y le falte una línea de código, porque parece que no acaba nunca. Indica qué código le añadirías y, haciendo uso de la numeración a la izquierda de cada línea, en qué posición.

Ejercicio 7

➔ **[0,5 puntos]** Indica si las siguientes afirmaciones son verdaderas o falsas. Si son falsas, explica brevemente qué sería lo correcto.

- En una base de datos hay más tablas que campos.
- Una clave primaria puede contener valores nulos.
- Dos tablas pueden tener el mismo nombre si están en bases de datos distintas.
- Los campos de una tabla pueden tener nombres repetidos.
- Un campo de tipo FECHA puede almacenar la hora exacta de un evento.

Ejercicio 8

Observa las siguientes dos tablas relacionadas en una base de datos:

Tabla Clientes

IdCliente	Nombre	FechaNacimiento
1	Ana	1990-06-21
2	Luis	1985-03-11

Tabla Inscripciones

IdInscripcion	IdCliente	Actividad
101	1	Spinning
102	2	Yoga
103	1	Pilates

→ **[2 puntos]** Responde a las siguientes preguntas:

- ¿Cuál es la clave primaria de la tabla Clientes? ¿Y la de la tabla Inscripciones?
- ¿Qué campo en la tabla Inscripciones actúa como clave externa? ¿A qué tabla y campo se refiere?
- ¿Puede el campo IdCliente en la tabla Inscripciones tener valores repetidos? ¿Por qué?
- ¿Puede la tabla Inscripciones existir sin que exista la tabla Clientes? ¿Y al revés? Justifica tus respuestas.
- ¿En qué actividades está inscrita Ana? Explica cómo lo has deducido.