

| Ejercicio 0102 | | |
|---|------------------------|---|
| ¿Cuáles de estos adjetivos se aplican a un algoritmo? | | |
| 1. | Finito | X |
| 2. | Corto | |
| 3. | Valorado | |
| 4. | Ambigüo | |
| 5. | Ese concepto no existe | |

| Ejercicio 0204 | | |
|---------------------------------------|--|---|
| ¿Las funciones devuelven información? | | |
| 1. | Sí, devuelven siempre 1 valor y se lo puedo asignar a una variable | |
| 2. | No, solo reciben parámetros y realizan una acción, nunca devuelven un resultado | |
| 3. | Sí, devuelven todos los valores que queramos, independientemente de los parámetros que reciban | X |
| 4. | No, no devuelven valores, salvo que reciban parámetros, en ese caso, devuelven 1 o más valores | |
| 5. | Sí, devuelven todos los que queramos, siempre y cuando hayan recibido por lo menos 1 parámetro | |

| Ejercicio 0304 | | |
|--|--|---|
| ¿Cuáles de los siguientes son programas válidos si se quiere mostrar por pantalla el valor de v y w? | | |
| 1. | <pre>v = 1 w = 2 algo(v, w) print("V vale", v) print("W vale", v)</pre> | |
| 2. | <pre>def algo(v, w): print("V vale " + str(v)) print("W vale " + str(w)) v = 6 w = 8 algo(v)</pre> | |
| 3. | <pre>def algo(v, w): print("V vale " + str(v)) print("W vale " + str(w)) v = 6 w = 8 algo(v, v)</pre> | |
| 4. | <pre>ddef aglo(v, w): print("V vale " + str(v)) print("W vale " + str(w)) algo(2, 4)</pre> | |
| 5. | <pre>def algo(): print("V vale " + str(v)) print("W vale " + str(w)) v = 6 w = 6 algo(v, w)</pre> | |
| 6. | Ninguno es válido | X |

| Ejercicio 0401 | | |
|-----------------------------------|--|---|
| Las estructuras if sirven para... | | |
| 1. | Iterar, repitiendo un bloque de código múltiples veces hasta cumplir con una condición | |
| 2. | Encapsular un bloque de código, al cual puedo pasarle parámetros y obtener una respuesta | |
| 3. | Determinar qué camino elegir entre muchos dependiendo de una condición | X |
| 4. | 1 y 2 son correctas | |
| 5. | 2 y 3 son correctas | |
| 6. | 1 y 3 son correctas | |
| 7. | Ninguna es correcta | |

| Ejercicio 0505 | | |
|---|---|---|
| ¿Cuál de los siguientes bloques de código no tiene una notación correcta? | | |
| 1. | <pre>if txt[0]=='a' and txt[0]=='a': print('El texto empieza con a') elif txt[0]=='a' and txt[0]=='a': print('El texto empieza con a') else: print('El texto no empieza con a')</pre> | |
| 2. | <pre>if txt[0]=='a' and txt[0]=='a': print('El texto empieza con a') elif txt[0]=='a' and txt[0]=='a': print('El texto empieza con a') elif txt[0]=='a' and txt[0]=='a': print('El texto empieza con a') elif txt[0]=='a' and txt[0]=='a': print('El texto empieza con a') else: print('El texto no empieza con a')</pre> | |
| 3. | <pre>if txt[0]=='a' and txt[0]=='a': print('El texto empieza con a') else if txt[0]=='a' and txt[0]=='a': print('El texto empieza con a') else: print('El texto no empieza con a')</pre> | X |
| 4. | <pre>if txt[0]=='a' and txt[0]=='a': print('El texto empieza con a') elif txt[0]=='a' and txt[0]=='a': print('El texto empieza con a') else: print('El texto no empieza con a')</pre> | |

| Ejercicio 0606 | | |
|--|--|---|
| ¿Cuál de los siguientes problemas es más acorde para el uso de una estructura iterativa? | | |
| 1. | Quiero asignarle un mismo valor a tres variables distintas | |
| 2. | Quiero imprimir varias veces la misma palabra | X |
| 3. | Quiero pedirle al usuario que ingrese una palabra | |
| 4. | Quiero asignarle un valor a una variable sólo si se cumple una condición | |

| Ejercicio 0706 | | |
|---|--|---|
| ¿Cuál es el valor de i después de ejecutar el siguiente bloque de código? | | |
| <pre>i=1 for j in range(8): i = i + 1</pre> | | |
| 1. | Ese bloque de código está mal sintácticamente, va a dar un error antes de ejecutarlo | |
| 2. | 8 | |
| 3. | 9 | X |
| 4. | Nunca va a parar de imprimir | |

| Ejercicio 0801 | | |
|--|--------------------------------|---|
| Si tengo la siguiente lista: | | |
| <pre>lista = [2,4,6]</pre> | | |
| Y quiero agregar el elemento "8" al final. | | |
| ¿Cuál de los siguientes bloques de código inserta correctamente el elemento? | | |
| 1. | <pre>lista[3] = 8</pre> | |
| 2. | <pre>lista.add(8)</pre> | |
| 3. | <pre>lista = lista + [8]</pre> | X |
| 4. | <pre>lista[4] = 8</pre> | |

| Ejercicio 0901 | | |
|---|--|---|
| Se quiere hacer una función que devuelva true si la lista recibida contiene al menos un elemento par. ¿Cuál de las siguientes funciones hace lo pedido? | | |
| 1. | <pre>def contiene_par(lista): tiene_par = False for elemento in lista: if elemento % 2 == 0: tiene_par = True return tiene_par</pre> | X |
| 2. | <pre>def contiene_par(lista): tiene_par = False for elemento in lista: if elemento % 2 != 0: tiene_par = True return tiene_par</pre> | |
| 3. | <pre>def contiene_par(lista): tiene_par = True for elemento in lista: if elemento % 2 != 0: tiene_par = False return tiene_par</pre> | |
| 4. | <pre>def contiene_par(lista): tiene_par = True for elemento in lista: if elemento % 2 == 0: tiene_par = False return tiene_par</pre> | |

| Ejercicio 1008 | | |
|--|--|---|
| ¿Cuál de los siguientes problemas podría ser resuelto usando la función map? | | |
| 1. | Quiero eliminar todos los elementos mayores a cien de una lista de números | |
| 2. | Quiero conseguir el promedio de una lista de números | |
| 3. | Quiero restarle cinco a todos los elementos de una lista de números | X |
| 4. | Quiero sumarle uno al primer elemento de una lista | |

| Ejercicio 1105 | | |
|---|--------------------------------------|---|
| <p>Dado el siguiente fragmento de código</p> <pre>def funcion_misteriosa(palabra): if(len(palabra)<=4): return palabra.upper() return palabra lista = ['Hola!','Cómo','están','todos?'] print(list(map(funcion_misteriosa ,lista)))</pre> <p>¿Qué se imprimirá por pantalla cuando se ejecute?</p> | | |
| 1. | ['Hola!', 'CÓMO', 'están', 'todos?'] | X |
| 2. | ['Hola!', 'CÓMO', 'ESTÁN', 'todos?'] | |
| 3. | ['HOLA!', 'CÓMO', 'están', 'todos?'] | |
| 4. | ['HOLA!', 'CÓMO', 'ESTÁN', 'todos?'] | |

| Ejercicio 1204 | | |
|---|---|---|
| ¿Cuál de los siguientes problemas podría ser resuelto usando la función filter? | | |
| 1. | Quiero multiplicar por dos todos los elementos de una lista de temperaturas pero solo si son menores a 20 | |
| 2. | Quiero obtener el mínimo elemento de una lista de edades | |
| 3. | Quiero quedarme sólo con los elementos mayores a 18 de una lista de edades | X |
| 4. | Quiero borrar solo el primer elemento de una lista de nombres | |

| | | |
|---|--|---|
| Ejercicio 1304 | | |
| Dado el siguiente fragmento de código | | |
| <pre>def filtro(persona): return (persona[1] % 2 != 0) and len(persona[0]) >= 4 lista=[('Logan', 75), ('Kendall', 40), ('Roman', 34), ('Shiv', 30), ('Tom', 34), ('Connor', 54)] lista_filtrada=list(filter(filtro,lista)) print(lista_filtrada)</pre> | | |
| ¿Qué se imprimirá por pantalla cuando se ejecute? | | |
| 1. | [('Logan', 75), ('Kendall', 40), ('Roman', 34), ('Shiv', 30), ('Tom', 34), ('Connor', 54)] | |
| 2. | [('Logan', 75)] | X |
| 3. | [('Roman', 34), ('Shiv', 30), ('Tom', 34)] | |
| 4. | [('Logan', 75), ('Shiv', 30), ('Tom', 34)] | |
| 5. | [] | |