

**PENSAMIENTO COMPUTACIONAL (90)**

**UBAXXI  
TEMA 3**

EXAMEN: PRIMER PARCIAL

APELLIDO:	CALIFICACIÓN:
NOMBRE:	
DNI (registrado en SIU Guaraní):	
E-MAIL:	DOCENTE (nombre y apellido):
TEL:	
AULA:	

Duración del examen: 1:30h.

- ✓ Escribir claramente el nombre en todas las páginas.
- ✓ El examen consta de 10 preguntas de opción múltiple.
- ✓ Cada pregunta tiene una y sólo una respuesta correcta.
- ✓ Las respuestas seleccionadas deben consignarse en la siguiente matriz de opciones.
- ✓ **Sólo se considerarán las respuestas anotadas en la matriz.**
- ✓ Las preguntas de la 1 a la 7 inclusive permiten acumular 1 punto (si son correctas), de la 8 a la 10 cada una acumula 2 puntos o 0.
- ✓ La nota final se calcula de acuerdo a la siguiente función:

<b>Puntos</b>	<b>1 o 2</b>	<b>3 o 4</b>	<b>5 o 6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>	<b>11</b>	<b>12</b>	<b>13</b>
<b>Nota</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>

**Matriz de Respuestas**

	Ej 1 1 Pto	Ej 2 1 Pto	Ej 3 1 Pto	Ej 4 1 Pto	Ej 5 1 Pto	Ej 6 1 Pto	Ej 7 1 Pto	Ej 8 2 Ptos	Ej 9 2 Ptos	Ej 10 2 Ptos	
<b>1</b>											<b>1</b>
<b>2</b>											<b>2</b>
<b>3</b>											<b>3</b>
<b>4</b>											<b>4</b>

**¡ATENCIÓN!** Las respuestas sólo se considerarán válidas si se encuentran en la matriz. De haber diferencias entre la opción seleccionada en el ejercicio y en la matriz, se considerará como válida esta última.

<b>0103 - 1 Pto</b>			
Si <b>nom='marta'</b> y <b>permiso=False</b> , Cuál programa evalúa menos condiciones?:			
<b>1</b>	<pre> if nom=='marta':     print('Primero') if permiso:     print('Segundo') if nom=='juan' and permiso:     print('Tercero') if nom=='juan' and not permiso:     print('Cuarto') if nom!='marta':     print('Quinto')                     </pre>		<b>1</b>
<b>2</b>	<pre> if nom=='marta':     print('Primero') else:     if permiso:         print('Segundo')     elif nom=='juan' and permiso:         print('Tercero')     elif nom=='juan':         print('Cuarto')     elif nom!='marta':         print('Quinto')                     </pre>	X	<b>2</b>
<b>3</b>	<pre> if nom!='marta':     print('Primero') if permiso:     print('Segundo') if nom=='juan' and permiso:     print('Tercero') if nom=='juan' and not permiso:     print('Cuarto') if nom=='marta':     print('Quinto')                     </pre>		<b>3</b>
<b>4</b>	<pre> if nom=='marta':     if permiso:         print('Primero')     else:         print('Segundo') if nom=='juan':     if permiso:         print('Tercero')     else:         print('Cuarto') if nom!='marta' and nom!='juan':     print('Quinto')                     </pre>		<b>4</b>

<b>0203 - 1 Pto</b>			
¿Cuál de las siguientes condiciones <b>no</b> es equivalente al resto? Sugerencia: Probá para los siguientes valores... <b>dia=31 , mes=4 , anio=2023</b>			
<b>1</b>	(mes in (4,6,9) and dia in range(1,31)) or mes in (1,3,5,7,8) or anio>2030		<b>1</b>
<b>2</b>	mes in range(1,8,2) or mes==8 or anio>=2031 or (mes in (4,6,9) and dia in range(1,31))		<b>2</b>
<b>3</b>	mes in (4,6,9) or dia in range(1,31) or mes in (1,3,5,7,8) or anio>2030	X	<b>3</b>
<b>4</b>	mes in (8,7,5,3,1) or (mes in (9,6,4) and dia>0 and dia<31) or anio>=2031		<b>4</b>

<b>0303 - 1 Pto</b>			
<p>¿Cuál de los siguientes programas <b>no</b> deja la lista <b>c</b> de la siguiente manera?  <b>c=[1, 2, 1, 2, 1, 2, 1, 2, 1, 2]</b></p> <p><b>Notas:</b></p> <p>El método <b>extend()</b> le agrega al final una lista a otra  <b>Ej:</b>  <b>a=[1,0]</b>  <b>a.extend([2,3]) -&gt; [1,0,2,3]</b></p> <p>El operador <b>+</b> concatena listas, respetando el orden  <b>Ej:</b>  <b>a=[1,0,2,1] b=[5,5]</b>  <b>a+b -&gt; [1,0,2,1,5,5]</b> y <b>b+a -&gt; [5,5,1,0,2,1]</b></p> <p>El operador <b>*</b> repite listas  <b>Ej:</b>  <b>a=[1,0,2]</b>  <b>a*2 -&gt; [1,0,2,1,0,2]</b></p>			
<b>1</b>	<pre>a=[1,2] b=[0,6,7] c=[] for num in b:     c.extend(a) for num in a:     c.extend(a)</pre>		<b>1</b>
<b>2</b>	<pre>a=[1,2] b=[0,6,7] c=a*len(b+a)</pre>		<b>2</b>
<b>3</b>	<pre>a=[1,2] b=[0,6,7] d=len(a)+len(b) c=a*d</pre>		<b>3</b>
<b>4</b>	<pre>a=[1,2] b=[0,6,7] d=len(a)+len(b) c=[] for num in a:     c=c+b</pre>	<b>X</b>	<b>4</b>

0403 - 1 Pto		
<p>¿Cuál de los siguientes códigos muestran por pantalla exactamente 4 asteriscos?</p> <p><b>Nota:</b></p> <p>El método <b>isupper()</b> devuelve <b>True</b> si el texto tiene letras y todas ellas son mayúsculas</p> <p><b>Ej:</b></p> <p>'9'.isupper() -&gt; False y 'T'.isupper() -&gt; True</p>		
1	<pre>i=1 for letra in ('a','A','e','E','i'):     print('*')     i+=1 while i&gt;0:     print('*')     i-=1</pre>	1
2	<pre>for letra in ('a','A','e','E','i'):     if letra.isupper():         for i in range(2):             print('*')</pre>	X 2
3	<pre>for i in range(4):     for j in range(2):         print('*') print('*')</pre>	3
4	<pre>for i in range(4,1,-3):     for j in range(2,2):         print('*')</pre>	4

0503 - 1 Pto		
<p>¿Cuál será la salida por pantalla del siguiente programa?</p> <pre>traductor={'A':0,'Á':5,'E':1,'É':6,'I':2,'Í':7} cambia='aeiouáéíóú' txt='TANTO VA EL CÁNTARO A LA FUENTE QUE AL FINAL SE ROMPE' txtEditado=txt for mayusc in traductor:     txtEditado=txtEditado.replace(mayusc,cambia[traductor[mayusc]]) print(txtEditado)</pre>		
1	aaeáaaaeaeae	1
2	TaNTO Va eL CÁNTaRO a La FUeNTe QUe aL FiNaL Se ROMPe	X 2
3	TNTO V L CNTRO L FUNT QU L FNL S ROMP	3
4	tOnto v0 l1 c5nt0ro 0 l0 fu1nt1 qu1 0l f2n0l s1 romp1	4

0603 - 1 Pto		
<p>¿Qué devuelve <b>fun()</b> si recibe la siguiente lista: ['carne','Pasta','pollo','Sushi']?</p> <pre>def fun(lista):     txt=''     for plato in lista:         if plato[0] in 'pP':             txt+=plato.lower()             txt+=' '     return txt</pre>		

<b>1</b>	'Sushi Carne'		<b>1</b>
<b>2</b>	' '		<b>2</b>
<b>3</b>	'pasta pollo'	<b>X</b>	<b>3</b>
<b>4</b>	'CARNE PASTA POLLO SUSHI'		<b>4</b>

<b>0703 - 1 Pto</b>			
¿Cuál de los códigos <b>no</b> es equivalente al siguiente?			
<pre> i=1 nombres=['lara', 'juan', 'PABLO', 'Ema'] while i&lt;=len(nombres):     print(nombres[i-1].capitalize())     i+=1 </pre> <p><b>Nota:</b> El método <i>reverse()</i> invierte una lista</p> <p><b>Ej:</b> <b>a=[1,0]</b> <b>a.reverse() -&gt; a=[0,1]</b></p>			
<b>1</b>	<pre> nombres=['lara', 'juan', 'PABLO', 'Ema'] for i in range(len(nombres)):     print(nombres[i].capitalize()) </pre>		<b>1</b>
<b>2</b>	<pre> nombres=['lara', 'juan', 'PABLO', 'Ema'] nombres.reverse() for i in range(3, len(nombres)):     print(nombres[i-1].upper()) </pre>	<b>X</b>	<b>2</b>
<b>3</b>	<pre> nombres=['lara', 'juan', 'PABLO', 'Ema'] for nom in nombres:     print(nom.capitalize()) </pre>		<b>3</b>
<b>4</b>	<pre> for nom in ['lara', 'juan', 'PABLO', 'Ema']:     print(nom.capitalize()) </pre>		<b>4</b>

0803 - 1 Pto			
<p>¿Qué función <b>edita</b> hay que usar para que el siguiente programa modifique un texto reemplazando letra <b>l</b> por / y letra <b>p</b> por *?</p> <pre>def edita(...):     -     -     -  #PPa1 texto='Luis es el primo de Pedro' antes='PpLl' queda='**//' print('Frase original:',texto) nuevoTexto=edita(texto,antes,queda) print('Frase editada:',nuevoTexto)</pre> <p>El programa debería mostrar por pantalla lo siguiente:</p> <p>Frase original: Luis es el primo de Pedro Frase editada: /uis es e/ *rimo de *edro</p> <p><b>Nota:</b> El método <b>replace()</b> cambia las ocurrencias del primer argumento por el segundo <b>Ej:</b> <b>'Hola qué tal!'.replace('a', 'e') -&gt;'Hole qué tel!'</b></p>			
1	<pre>def edita(a,q):     for letra in t:         if letra in a:             t.replace(letra,q[i])     return t</pre>		1
2	<pre>def edita(t,a,q):     for i in range(len(a)):         if a[i] in t:             t=t.replace(a[i],q[i])     return t</pre>	X	2
3	<pre>def edita(t,a,q):     for letra in t:         if letra in a:             t=t.replace(letra,'*')     t=t.replace('*',q[0])</pre>		3
4	<pre>def edita():     for i in range(len(a)):         if a[i] in t:             t.replace(q[i],a[i])     return t</pre>		4

0903 - 1 Pto			
<p>¿Qué muestra por pantalla el siguiente programa?</p> <pre>a=2.5 b=10 c=0 d=2 calculo= b*(d+a)//(b-d)-c print(calculo)</pre>			
1	0.0		1

2	7.2		2
3	2.0		3
4	5.0	X	4

<b>1003 - 1 Pto</b>			
Indique cuál es la salida correcta del siguiente programa:			
<pre>def terminaCons(pal):     ultima=pal[-1]     return ultima.lower() not in 'aeiou'  golosinas=['Alfajor', 'CAMELO', 'BOMBÓN', 'chupetín', 'chicle'] i=0 while i&lt;len(golosinas):     if not terminaCons(golosinas[i]):         golosinas.pop(i)     else:         i+=1 print(golosinas)</pre>			
<p><b>Nota:</b> El método <b>pop()</b> elimina el elemento de la lista que está en la posición indicada en el argumento</p> <p><b>Ej:</b> <b>a=[1,2,3,4,5]</b> <b>a.pop(2) -&gt; a=[1,2,4,5]</b></p> <p>Un índice negativo <b>-i</b> apunta a un elemento <b>i</b> posiciones antes del final</p> <p><b>Ej:</b> <b>a=[1,2,3,4,5]</b> <b>a[-2] -&gt; 4</b></p>			
1	['caramelo', 'chicle']		1
2	['alfajor', 'caramelo', 'bombón', 'chupetín', 'chicle']		2
3	[]		3
4	['Alfajor', 'BOMBÓN', 'chupetín']	X	4