

**PENSAMIENTO COMPUTACIONAL (90)**

**.UBAXXI**

**TEMA 1**

EXAMEN: EXAMEN DICIEMBRE

APELLIDO:	CALIFICACIÓN:
NOMBRE:	
DNI (registrado en SIU Guaraní):	
E-MAIL:	NOTA Y FIRMA DOCENTE (no rellenar)
TEL:	
AULA:	

Duración del examen: 1:30h.

- ✓ Escribir claramente el nombre en todas las páginas.
- ✓ El examen consta de 9 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 5 inclusive permiten acumular 1 punto (si son correctas), de la 6 a la 9 cada una acumula 2 puntos o 0.
- ✓ La nota final se calcula de acuerdo a la siguiente función:

Puntos	1 o 2	3 o 4	5 o 6	7	8	9	10	11	12	13
<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 2 Ptos	Ej 7 2 Ptos	Ej 8 2 Ptos	Ej 9 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.



**Talón de Control para el Alumno (Tema ....)**

	Ej 1 1 Pto	Ej 2 1 Pto	Ej 3 1 Pto	Ej 4 1 Pto	Ej 5 1 Pto	Ej 6 2 Ptos	Ej 7 2 Ptos	Ej 8 2 Ptos	Ej 9 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>

<b>Ejercicio 1 – Tema 1</b>		<b>1 Pto</b>	
¿En cuál de los siguientes bucles el cuerpo se ejecuta exactamente 5 veces <b>siempre</b> ?			
<b>1</b>	<pre>a=3 i=1 while a!=5:     a=int(input('Número (5 para salir): '))     i+=1</pre>		<b>1</b>
<b>2</b>	<pre>i=1 while i!=5:     a=int(input('Número (5 para salir): '))     i-=1</pre>		<b>2</b>
<b>3</b>	<pre>b=[9,6,7,1,0] for elem in b:     a=input('Número (5 para salir): ')</pre>	<b>X</b>	<b>3</b>
<b>4</b>	<pre>for elem in [0,1,2,3,4,5]:     a=int(input('Número (5 para salir): '))</pre>		<b>4</b>

<b>Ejercicio 2 – Tema 1</b>		<b>1 Pto</b>	
¿Qué muestra el siguiente programa?			
<pre>a=1 b=True c=(1,0,2) if 0 in c and not b:     print('Hay 0') elif len(c)&gt;2:     if b and a&gt;0:         if a in c:             print('Está a')         else:             print('nada')     elif a in c:         print('Está ok') else:     print('chau')</pre>			
<b>1</b>	Está ok chau		<b>1</b>
<b>2</b>	nada		<b>2</b>
<b>3</b>	Está a	<b>X</b>	<b>3</b>
<b>4</b>	Hay 0 chau		<b>4</b>

Ejercicio 3 – Tema 1		1 Pto	
Para el siguiente Dataframe			
	plato	Yemas Claras	Azúcar Harina
0	Flan	8.0 2.0	100 NaN
1	Arroz Con Lecha	NaN NaN	150 NaN
2	Dulce de Leche	NaN NaN	250 NaN
3	Pio Nono	4.0 4.0	150 100.0
4	Merengue	NaN 6.0	270 90.0
¿Qué instrucción produce el siguiente resultado?			
	plato	Yemas Claras	Azúcar Harina
1	Arroz Con Lecha	NaN NaN	150NaN
2	Dulce de Leche	NaN NaN	250NaN
4	Merengue	NaN 6.0	27090.0
1	menu [menu [ 'Yemas ' ] . isnull ( ) ]		X 1
2	menu . head ( 2 )		2
3	menu [ [ 'Claras ' , 'Yemas ' , 'plato ' ] ]		3
4	menu [menu [ 'Azúcar ' ] > 100 ]		4

Ejercicio 4 – Tema 1		1 Pto	
¿Qué muestra por pantalla el siguiente programa?			
<pre>txt1='tostadora' txt2='sandwichera' nuevo1=txt1[5].upper()+txt1[6:] nuevo2=txt2[0].upper()+txt2[1:4]+txt2[-2]+txt2[-1] nuevoTxt=nuevo1+' y '+nuevo2 print(nuevoTxt)</pre>			
<b>Notas:</b>			
Un índice negativo referencia un elemento descontando del largo de la string. O desde el final hacia el inicio			
<b>Ej:</b>			
'hola'[-1] -> 'a'			
Se puede seleccionar una porción de una string con <b>[desde:hasta]</b> , (recordá que hasta no entra)			
<b>Ej:</b>			
'hola, qué tal?':[4] -> 'hola'			
1	sandradora		1
2	Dora y Sandra		X 2
3	TOSTADORAYSANDWICHERA		3
4	T Y S		4

Ejercicio 5 – Tema 1		1 Pto	
¿Cuál programa valida correctamente que se ingrese un número entre <b>0 y 100</b> ? Debe insistir hasta conseguir un <b>entero</b> entre esos valores			
1	<pre>pide=False while pide:     try:         num=int(input('Entero entre 0 y 100 inclusive '))     except ValueError:         print('Debe ser un entero')     pide=False</pre>		1
2	<pre>try:     num=int(input('Entero entre 0 y 100 inclusive ')) except ValueError:     print('Debe ser un entero')</pre>		2
3	<pre>pide=True while pide:     try:         num=int(input('Entero entre 0 y 100 inclusive '))         if num in range(101):             pide=False         else:             print('Fuera de rango')     except ValueError:         print('Debe ser un entero')</pre>	X	3
4	<pre>try:     num=int(input('Entero entre 0 y 100 inclusive ')) except ValueError:     print('Debe ser un entero') while num not in range(1,100):     num=int(input('Entero entre 0 y 100 inclusive '))</pre>		4

Ejercicio 6 – Tema 1		2 Ptos	
¿Qué muestra por pantalla el siguiente programa?			
<pre>def recorta (n):     return n[:6]  alturas=[3123,267,2004,1252] calles=['San Martín', 'La Plata', 'Independencia', 'Cabildo'] dcc=list(map(recorta,calles)) visitas=[] for i in range(len(dcc)):     if alturas[i]%2==0:         visitas.append([dcc[i],alturas[i]]) print(visitas)</pre>			
<b>Nota:</b> Se puede seleccionar una porción de una string con <b>[desde:hasta]</b> , (recordá que <b>hasta</b> no entra)			
<b>Ej:</b> 'hola, qué tal?'[4:] -> ', qué tal?'			
1	[[ 'Indepe', 2004], [ 'Cabild', 1252]]	X	1
2	[]		2
3	['San Martín', 'La Plata', 'Independencia', 'Cabildo']		3
4	['San Martín', 3123, 'La Plata', 267, 'Independencia', 2004, 'Cabildo', 1252]		4
Ejercicio 7 – Tema 1		2 Ptos	
Para el siguiente programa			
<pre>def apertura (archivo, modo):     arch=open(archivo,modo)     return arch</pre>			

```

nomArch=apertura('datos1.txt',...) #1
nomArch.write('Archivo nuevo que contiene texto genérico\n')
nomArch.write('Línea 2\n')
nomArch.close()

entrada=apertura('datos1.txt',...) #2
lineas=entrada.readlines()

salida=apertura('datos2.txt',...) #3
salida.write(lineas[1])
entrada.close()
salida.close()
    
```

¿Qué secuencia de modos de apertura debe usarse?

1	'r' 'w' 'a'		1
2	'w' 'r' 'w'	X	2
3	'r' 'r' 'r'		3
4	'w' 'w' 'w'		4

Ejercicio 8 – Tema 1		2 Ptos	
<p>¿Qué contenido debe tener el diccionario <b>precios</b> para que el siguiente programa funcione adecuadamente?</p> <pre> precios={...}  items=['','azafrán','pimentón','canela','anís','vainilla'] print('Seleccioná para ver precio') for i in range(1,len(items)):     print(i,'-',items[i]) sel=int(input()) while sel not in range(1,len(items)):     sel=int(input()) print('Precio de',items[sel],': \$',precios[items[sel]])                     </pre> <p>Ejemplo de una corrida:</p> <pre> Seleccioná para ver precio 1 - azafrán 2 - pimentón 3 - canela 4 - anís 5 - vainilla 2 Precio de pimentón : \$ 2670                     </pre>			
1	precios={}		1
2	precios={['canela', 'anís', 'pimentón', 'vainilla', 'azafrán']: [2500, 1700, 2670, 6000, 21000]}		2
3	precios={'canela':2500, 'anís':1700, 'pimentón':2670, 'vainilla':6000, 'azafrán':21000}	X	3
4	precios={2500: 'CANELA', 1700: 'ANÍS', 2670: 'PIMENTÓN', 6000: 'VAINILLA', 21000: 'AZAFRÁN'}		4

Ejercicio 9 – Tema 1		2 Ptos	
<p>¿Qué versión de la función <b>combina no funciona</b> adecuadamente para el siguiente programa?</p> <pre>def combina(...):     -     -     impar=[1,3,5]     par=[2,4,6]     total=combina(impar,par)     print(total)</pre> <p>La salida final debe ser: [1, 2, 3, 4, 5, 6]</p> <p><b>Notas:</b> El método <b>insert(dde,qué)</b> agrega un elemento a una lista en una posición específica El método <b>remove(elem)</b> elimina la primer ocurrencia de <b>elem</b> La función <b>min()</b> devuelve el menor de una lista <b>Ejs:</b> [0,1,2,3,4].insert(2,5) -&gt; [0,1,5,2,3,4] [0,0,1,0,1,1].remove(1) -&gt; [0,0,0,1,1] min([1,2,0]) -&gt; 0</p>			
1	<pre>def combina(x,y):     for i in range(len(y)):         x.insert(i*2+1,y[i])     return x</pre>		1
2	<pre>def combina(x,y):     nueva=[]     largo=len(x)+len(y)     while len(nueva)&lt;largo:         if len(x)&gt;0:             menx=min(x)         else:             menx=99         if len(y)&gt;0:             meny=min(y)         else:             meny=99         if menx&lt;meny:             nueva.append(menx)             x.remove(menx)         else:             nueva.append(meny)             y.remove(meny)     return nueva</pre>		2
3	<pre>def combina(nueva):     nueva=[]     for num in y:         nueva.append(num)     for num in x:         nueva.append(num)     return nueva</pre>	X	3
4	<pre>def combina(x,y):     nueva=[]     for i in range(len(x)):         nueva.append(x[i])         nueva.append(y[i])     return nueva</pre>		4

