

PENSAMIENTO COMPUTACIONAL (90)

EXAMEN: EXAMEN DICIEMBRE

UBA XXI

TEMA 4

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
Nota	1	2	3	4	5	6	7	8	9	10

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	
1										1
2										2
3										3
4										4

¡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	
1										1
2										2
3										3
4										4

Ejercicio 1 – Tema 4			1 Pto
¿En cuál de los siguientes bucles el cuerpo no siempre se ejecuta exactamente 5 veces?			
Nota: string[1:] recorta string y se queda con el segundo caracter en adelante			
1	<pre>b=1 i=5 while i>0 and b==1: a=input('Número (5 para salir): ') i-=1</pre>		1
2	<pre>a=1 i=0 while a!=5 and i<5: a=input('Número (5 para salir): ') if a=='5': a=1 i+=1</pre>		2
3	<pre>txt='letra' while txt!='': a=input('Número (5 para salir): ') if a=='5': txt=txt+'letra' txt=txt[1:]</pre>	X	3
4	<pre>for i in range(len('letra'),0,-1): a=input('Número (5 para salir): ')</pre>		4

Ejercicio 2 – Tema 4			1 Pto
¿Qué muestra el siguiente programa?			
<pre>a='ana' b=('juan','pedro') c='a' if c in a and a in b: print('Hay a') elif len(c)<2: if a in b and len(a)>0: if a in c: print('Está a') else: print('Nada') elif c in a: print('Está ok') print('Chau')</pre>			
1	<pre>Está ok Chau</pre>	X	1
2	<pre>Chau</pre>		2
3	<pre>Está a Chau</pre>		3
4	<pre>Nada</pre>		4

Ejercicio 3 – Tema 4					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	Harina	Azúcar			
0	Flan	NaN	100			
1	Arroz Con Lecha	NaN	150			
2	Dulce de Leche	NaN	250			
3	Pio Nono	100.0	150			
4	Merengue	90.0	270			
1	menu[['plato', 'Harina', 'Azúcar']]				X	1
2	menu.iloc[2:3]					2
3	menu.tail(4)					3
4	menu[menu['Yemas']>3]					4

Ejercicio 4 – Tema 4			1 Pto	
¿Qué muestra por pantalla el siguiente programa?				
<pre>exp1='2+5' exp2=' (1+1) * (exp1) ' nuevo1=exp2[:7]+exp1+') ' nuevoTxt=nuevo1+' = 14' print(nuevoTxt)</pre>				
Notas: Se puede seleccionar una porción de una string con [desde:hasta] , (recordá que hasta no entra) Ej: 'hola, qué tal?'[2:] -> 'la, qué tal?'				
1	(1+1) *EXP1=14			1
2	2+5 = (1+1) * (2+5)			2
3	(1+1) * (2+5) = 14			X 3
4	14			4

Ejercicio 5 – Tema 4			1 Pto
¿Cuál programa valida correctamente que se ingrese un valor para día entre 1 y 31 ? Debe insistir hasta conseguir un entero entre esos valores			
1	<pre>dia=int(input('Día: ')) while dia>31 or dia<1: dia=int(input('Día: '))</pre>		1
2	<pre>try: dia=int(input('Día: ')) if dia not in range(1,32): print('No es un día válido') except ValueError: print('Debe ser un entero') while dia>31 or dia<1: dia=int(input('Día: '))</pre>		2
3	<pre>while False: try: dia=int(input('Día: ')) ok=True except ValueError: print('Debe ser un entero')</pre>		3
4	<pre>ok=False while not ok: try: dia=int(input('Día: ')) if dia in range(1,32): ok=True except ValueError: print('Debe ser un entero')</pre>	X	4

Ejercicio 6 – Tema 4			2 Ptos
¿Qué muestra por pantalla el siguiente programa?			
<pre>def muchos(artic): return artic[1]>10 compraBazar=[['copas',15],['tazas',5],['velas',11], ['platos',4],['paneras',1]] nuevos=list(filter(muchos,compraBazar)) print(nuevos)</pre>			
1	<code>['copas', 'tazas', 'velas', 'platos', 'paneras']</code>		1
2	<code>[['copas', 15], ['velas', 11]]</code>	X	2
3	<code>[[15], [5], [11], [4], [1]]</code>		3
4	<code>[['COPAS', 15], ['TAZAS', 5], [['VELAS', 11], ['PLATOS', 4], ['PANERAS', 1]]</code>		4

Ejercicio 7 – Tema 4			2 Ptos
Para el siguiente programa <pre>def apertura (archivo, modo): arch=open (archivo,modo) return arch nomArch=apertura('datos2.txt',...) #1 lin=nomArch.readline() nomArch.close() nuevo=apertura('datos1.txt',...) #2 nuevo.write(lin) nuevo.close() nuevo=apertura('datos1.txt',...) #3 lineas=nuevo.readlines() print(len(lineas)) nuevo.close()</pre> ¿Qué secuencia de modos de apertura debe usarse?			
1	'a' 'r' 'r+'		1
2	'r+' 'r' 'w'		2
3	'w' 'r' 'w'		3
4	'r' 'w' 'r'	X	4

Ejercicio 8 – Tema 4		2 Ptos	
¿Qué contenido debe tener el diccionario <i>herramientas</i> para que el siguiente programa funcione adecuadamente?			
<pre>herramientas={...} print('Seleccioná para ver precio') for cl in herramientas: print(cl, '-', herramientas[cl][0]) sel=input() while sel not in herramientas: sel=input() print('Precio de ', herramientas[sel][0], ': \$ ', herramientas[sel][1])</pre>			
Ejemplo de una corrida:			
<pre>Seleccioná para ver precio ma001 - martillo carpintero ma010 - martillo redondo to025 - tornillo rosca de 0.25 cl129 - clavo autoperforante de 0.5 ma055 - martillo zapatero ma001 Precio de martillo carpintero : \$ 12500</pre>			
1	<pre>herramientas=[['MA001', 'martillo carpintero', 12500], ['MA010', 'martillo redondo', 11750], ['TO025', 'tornillo rosca de 0.25', 3245], ['CL129', 'clavo autoperforante de 0.5', 980], ['MA055', 'martillo zapatero', 19300]]</pre>		1
2	<pre>herramientas={ 'ma001', 'ma010', 'to025', 'cl129', 'ma055' }</pre>		2
3	<pre>herramientas={ 'ma001': ['martillo carpintero', 12500], 'ma010': ['martillo redondo', 11750], 'to025': ['tornillo rosca de 0.25', 3245], 'cl129': ['clavo autoperforante de 0.5', 980], 'ma055': ['martillo zapatero', 19300]}</pre>	X	3
4	<pre>herramientas={ 12500: 'martillo carpintero', 11750: 'martillo redondo', 3245: 'tornillo rosca de 0.25', 980: 'clavo autoperforante de 0.5', 19300: 'martillo zapatero' }</pre>		4

Ejercicio 9 – Tema 3			2 Ptos
¿Qué versión de la función separa no funciona adecuadamente para el siguiente programa? <pre>def separa (...): - - numeros=[1,27,10,2,35,3,5,88] pares=[] impares=[] separa (numeros,impares,pares) print (pares) print (impares)</pre> La salida final debe ser: [2, 10, 88] [1, 3, 5, 27, 35] Notas: El método sort() ordena una lista La función min() devuelve el menor de una lista Ej: [0,0,1,0,1,1].sort() -> [0,0,0,1,1,1] min([1,2,0]) -> 0			
1	<pre>def separa (num,imp,par): while len (num)>0: men=min (num) if men%2==0: par.append (men) else: imp.append (men) num.remove (men)</pre>		1
2	<pre>def separa (num,imp,par): for n in num: if n%2==0: par.append (n) else: imp.append (n) par.sort () imp.sort ()</pre>		2
3	<pre>def separa (num,imp,par): num.sort () for n in num: if n%2==0: par.append (n) else: imp.append (n)</pre>		3
4	<pre>def separa (par,num): while len (num)>0: men=num[0] if men%2==0: imp.append (men) else: par.append (men) num.remove (men) return num,imp,par</pre>	X	4