

PENSAMIENTO COMPUTACIONAL (90)

EXAMEN: PRIMER PARIAL – PRIMER CUATRIMESTRE 2025

UBA XXI

TEMA 7

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

Consignar con una X (cruz) la respuesta elegida en cada ejercicio en la siguiente matriz:

	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	
A										
B										
C										
D										



Talón de Control para el Alumno (Tema 7)

	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	
A										
B										
C										
D										

Ejercicio 1 – Tema 7		1 Pto	
¿Cuál de los siguientes programas muestra por pantalla caro ?			
A	<pre>producto='birome' unidad='caja' precio=60 if (producto in ('lápiz color','birome', 'boligoma')): if precio<100 and unidad=='caja' and unidad=='kg': print('carito') elif precio<100 and unidad=='unidad': print('caro') elif producto in ('plasticola','pincel','hoja'): print('barato')</pre>		A
B	<pre>producto='birome' unidad='caja' precio=60 if producto in ('block A4','etiqueta'): if (precio>100 and (unidad=='caja' or unidad=='docena')): print('carito') elif precio<100 and unidad=='unidad': print('maso') elif producto in ('plasticola','hoja','birome'): if unidad in ('ciento','docena','bolsa'): print('barato') else: print('caro')</pre>	X	B
C	<pre>producto='birome' unidad='caja' precio=60 if producto in ('birome','etiqueta','marcador'): if (precio>100 and (unidad=='caja' or unidad=='docena')): print('precio medio') elif precio>100 and unidad=='unidad': print('caro') print('barato')</pre>		C
D	<pre>producto='birome' unidad='caja' precio=60 if producto in ('lápiz color','marcador','birome'): if (precio>100 or unidad=='caja' or unidad=='kg'): print('precio medio') elif precio>100 and unidad=='unidad': print('caro') else: print('barato')</pre>		D

Ejercicio 2 – Tema 7		1 Pto	
¿Qué muestra el siguiente programa? <pre>a=0 b=1 c=2 d=-2 cant=d-b*b+c*2/b+4 print('&'* int(cant))</pre> Nota: El operador * aplicado a una string y un número entero es repetición Ej: 'hola'*2-> 'holahola'			
A	3		A
B	&&&&&	X	B
C	&&		C
D	&&&		D

Ejercicio 3 – Tema 7		1 Pto	
¿Con qué contenido acabará la lista final en el siguiente programa? <pre>indicador=[1,2,3,4,5,6] numeros=[1,2,3,4,5,6] final=[0]*6 for i in range(len(numeros)): final[-(indicador[i])]=numeros[i]</pre> Notas: El operador * aplicado a una lista es repetición Ej: [1, 3]*2 -> [1, 3, 1, 3] Se puede referenciar de atrás para adelante un elemento de una lista con un índice negativo Ej: a=[1,2,3,4] a[-2] -> 3 y a[-1] -> 4			
A	[0, 0, 0, 6, 5, 4]		A
B	[6, 5, 4, 3, 2, 1]	X	B
C	[0, 0, 0]		C
D	[1, 2, 3, 4, 5, 6]		D

Ejercicio 4 – Tema 7		1 Pto	
¿Cuál programa carga ordenadamente en forma descendente las listas enteros y fraccionarios con los elementos de numeros que corresponda? En enteros debe quedar [10, 2, -3] y en fraccionarios [0.25, -3.36] Notas: El método <i>reverse()</i> invierte una lista Ej: lis=[-1,2,0] lis.reverse() lis-> [0,2,-1] El método <i>sort()</i> ordena una lista Ej: lis=[6,-1,0] lis.sort() -> lis=[-1, 0, 6] El método <i>copy()</i> genera una copia de la lista Ej: lis=[6,0] lis1=lis.copy() -> hay dos listas que en ese momento tienen los mismos datos El método <i>remove()</i> quita la primer ocurrencia de un elemento en una lista Ej: lis=[6,-1,0,5,6] lis.remove(6) -> lis=[-1, 0, 5, 6]			
A	<pre>numeros=[-3.36,10,2,0.25,-3] enteros=[] fraccionarios=[] for num in numeros: if num==int(num): fraccionarios.append(num) else: fraccionarios.append(num) enteros.sort() fraccionarios.reverse()</pre>		A
B	<pre>numeros=[-3.36,10,2,0.25,-3] enteros=numeros.copy() fraccionarios=numeros.copy() for num in enteros: if num!=int(num): enteros.remove(num) enteros.sort() fraccionarios.sort()</pre>		B
C	<pre>numeros=[-3.36,10,2,0.25,-3] enteros=[] fraccionarios=[] for num in numeros: if num==int(num): enteros.append(num) else: fraccionarios.append(num) enteros.sort() enteros.reverse() fraccionarios.sort() fraccionarios.reverse()</pre>	X	C
D	<pre>numeros=[-3.36,10,2,0.25,-3] enteros=[] fraccionarios=[] for num in enteros: if num==int(num): numeros.append(num) else: fraccionarios.append(num) numeros.reverse() fraccionarios.reverse()</pre>		D

Ejercicio 5 – Tema 7			1 Pto
¿Qué devuelve la función fun en este programa? <pre>def fun(t): vocales='AEIOUÁÉÍÓÜ' repes=False repetidas='' for letra in t: if letra in vocales and t.count(letra)>0: repes=True if letra not in repetidas: repetidas+=letra return repetidas.lower() txt='Estaba LA RANA cantando debajo del AGUA' print(fun(txt))</pre> Nota: El método count() cuenta cuántas apariciones de su argumento hay en una string Ej: 'hola, qué tal?'.count('a') -> 2			
A	'eau'	X	A
B	False		B
C	True		C
D	'EAAAAUA'		D

Ejercicio 6 – Tema 7			2 Ptos
¿Qué muestra por pantalla el siguiente programa? <pre>impares='13579' for i in range(len(impares)): linea='' for j in range(len(impares)*2): if j%2!=0: linea+=impares[j//2] else: linea+=' ' print(linea)</pre> Notas: El operador * aplicado a una string es repetición Ej: 'hola'*3 -> 'holaholahola' El operador // es el cociente entero Ej: 7//2 -> 3			
A	1 2 3 4 5 6 7 8 9		A
B	1 3 5 7 9 1 3 5 7 9 1 3 5 7 9 1 3 5 7 9 1 3 5 7 9	X	B
C	123456789 987654321		C
D	1 1 3 1 3 5 1 3 5 7 1 3 5 7 9		D

Ejercicio 7 – Tema 7		2 Ptos	
¿Cuál programa muestra el siguiente texto como salida?			
LAS ABEJAS VAALAN Y LAS OVEJAS BALAN			
Notas: El método replace() permite cambiar todas las apariciones de una string por otra dentro de un texto Ej: 'llueve a las doce. Qué llueva, qué llueva!'.replace('llue','yu') -> "yuve a las doce. Qué yuva, qué yuva!" Se puede seleccionar una porción de una string con [desde:hasta], (recordá que hasta no entra) Ej: 'hola, qué tal?'[4:] -> ', qué tal?'			
A	<pre>frase='Las abejas vuelan y las ovejas balan' animal1='abejas' animal2='ovejas' frase1=frase[:18] frase2=frase[18:] frase1=frase1.replace(animal1,animal2) frase2=frase2.replace(animal2,animal1) frase1=frase1.replace('vuelan','no vuelan') frase2=frase2.replace('balan','no balan') frase=frase1+frase2.upper() print(frase)</pre>		A
B	<pre>frase='Las abejas vuelan y las ovejas balan' animal1='abeja' animal2='oveja' frase=frase.replace(animal1,'ABEJA') frase=frase.replace(animal2,'OVEJA') for voc in 'EIOU': frase=frase.replace(voc,'A') print(frase)</pre>		B
C	<pre>frase='Las abejas vuelan y las ovejas balan' animal1='abejas' animal2='ovejas' frase=frase.replace(animal1,animal2) frase=frase.replace(animal2,animal1) frase=frase.replace('vuelan','no vuelan') frase.replace('balan','no balan') print(frase)</pre>		C
D	<pre>frase='Las abejas vuelan y las ovejas balan' animal1='abeja' animal2='oveja' frase=frase.replace(animal1,'ABEJA') frase=frase.replace(animal2,'OVEJA') for voc in 'eiou': frase=frase.replace(voc,'A') print(frase.upper())</pre>	X	D

Ejercicio 8 – Tema 7		2 Ptos	
¿Cuál es la versión correcta de la función edita? El programa debe mostrar argumentar CAOS en cualquier situación no es COMPROMISO <pre>def edita(...) : - - - txt='argumentar caos en cualquier situación no es compromiso' resalta=['compromiso', 'éxito', 'objetivos', 'caos', 'ganancias'] print(edita(txt,resalta))</pre> Notas: El método <i>split()</i> genera una lista con las partes de una string separadas por blancos Ej: txt='hola qué tal a todos' lis= txt.split() lis-> ['hola', 'qué', 'tal', 'a', 'todos'] El método <i>join()</i> genera una string con los caracteres que tiene como elementos una lista separándolos por el texto al inicio Ej: lis= ['h', 'o', 'l', 'a'] txt='*'.join(lis)-> txt='h**o**l**a'			
A	<pre>def edita(t,palabras): lista=t.split() for i in range(len(lista)): if lista[i] in palabras: lista[i]=lista[i].lower()</pre>		A
B	<pre>def edita(t): t.lower() lista=t.split() for i in range(len(lista)): if lista[i] in palabras: lista[i]=lista[i] return ' '.join(palabras)</pre>		B
C	<pre>def edita(t,palabras): t=t.lower() lista=t.split() for i in range(len(lista)): if lista[i] in palabras: lista[i]=lista[i].upper() return ' '.join(lista)</pre>	X	C
D	<pre>def edita(t,palabras): t=t.lower() lista=t.split() for i in range(len(t)): if t[i] in palabras: t[i]=t[i].upper() t=' '.join(palabras) return t.upper()</pre>		D

Ejercicio 9 – Tema 7		2 Ptos	
¿Cuál condición es candidata a ser equivalente a la siguiente?			
(num not in ent and num not in fracc and ((num==int(num) and num<10)or(num!=int(num) and num<1)))			
Atención!!			
Te recomendamos que pruebes las condiciones con los siguientes valores:			
ent=[-2,21,16]			
fracc=[33.5,-0.21,12.05]			
num=-2 y num=2.05			
Nota:			
Una condición (proposición) p es equivalente a otra q , si sus tablas de verdad coinciden. En otras palabras: si siempre que p da V , q da V y siempre que p da F , q da F			
A	((int(num)!=num or num==int(num) or num<=9) and (num not in fracc or num not in ent))		A
B	(int(num)!=num or num==int(num) or num not in fracc or num not in ent)		B
C	((int(num)!=num and num<1)or (num==int(num) and num<=9)) and num not in fracc and num not in ent)	X	C
D	num in ent and num==int(num) and num<1		D