

PENSAMIENTO COMPUTACIONAL (90)

.UBAXXI

TEMA 2

EXAMEN: EXAMEN FEBRERO 2025

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 2)

	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 2		1 Pto
<p>¿Cuál programa produce la siguiente salida?</p> <pre>C C C C A A A A M M M M A A A A</pre> <p>Notas: Se puede seleccionar una parte de una string con [:] Ej: 'hola y chau'[1:4] -> 'ola'</p> <p>El operador * aplicado a una string es repetición Ej: 'mi'*3 -> 'mimimi'</p>		
1	<pre>mostrar=' '*2+'MA' for i in range(len('CAMA')): print(mostrar)</pre>	1
2	<pre>linea='CAMA' while len(linea)>0: print(linea) linea=linea[1:]</pre>	2
3	<pre>for i in range(len('CAMA')): print(('CAMA'[i]+' ')*len('cama'))</pre>	X 3
4	<pre>for i in range(len('cama')//2): print('CAMA'[:len('cama')//2]) for i in range(len('cama')//2,len('cama')): inicio=' '*len('cama')//2 fin='CAMA'[len('cama')//2:] print(inicio+fin)</pre>	4

Ejercicio 2 – Tema 2		1 Pto
¿Cuál programa produce la siguiente salida? amigo de la escuela		
1	<pre> nombre='JUAN' apellido='CESCA' apodo='PACHI' if len(nombre)==4 and 'C' in apellido: if apodo in ('PUCHO','PACHI','POCHO'): print('amigo de fútbol') elif apodo=='RUSO': print('amigo del trabajo') elif 'U' in nombre or 'U' in apellido: if len(apellido)==9: print('amigo de la escuela') elif 'A' in apodo and 'A' in apellido: if len(apodo) in range(1,9,2): print('mi mejor amigo') </pre>	1
2	<pre> nombre='JUAN' apellido='CESCA' apodo='PACHI' if len(nombre)==4 and 'U' in apellido: if apodo in ('PUCHO','PACHI','POCHO'): print('amigo de la escuela') elif apodo=='RUSO': print('amigo del trabajo') elif 'U' in nombre and 'U' in apellido and 'U' in apodo: if len(apellido)==9: print('mi mejor amigo') else: print('un amigo de mi hermano') else: print('no es amigo') </pre>	2
3	<pre> nombre='JUAN' apellido='CESCA' apodo='PACHI' if len(nombre)==4 and 'U' in apellido: if apodo in ('PUCHO','PACHI','POCHO'): print('amigo de la escuela') elif apodo=='RUSO': print('amigo del trabajo') elif 'U' in nombre or 'U' in apellido or 'U' in apodo: if len(apellido)==9: print('amigo de fútbol') elif 'A' in apodo and 'A' in apellido and 'A' in nombre: if len(apodo) in range(1,9,2): print('mi mejor amigo') else: print('no es amigo') </pre>	3
4	<pre> nombre='JUAN' apellido='CESCA' apodo='PACHI' if len(nombre)==4 and 'U' in apellido: if apodo in ('PUCHO','PACHI','POCHO'): print('amigo de fútbol') elif apodo=='RUSO': print('amigo del trabajo') elif 'U' in nombre or 'U' in apellido or 'U' in apodo: if len(apellido) in (1,3,5,7): print('amigo de la escuela') elif 'A' in apodo or 'A' in apellido or 'A' in nombre: if len(apodo) in range(1,9,2): print('mi mejor amigo') else: print('no es amigo') </pre>	X 4

Ejercicio 3 – Tema 2		1 Pto	
¿Cuál programa valida correctamente que se ingrese una fecha (día y mes)? No se consideran años bisiestos (sólo se aceptan 28 días para febrero). Debe insistir hasta conseguir una fecha válida			
1	<pre>while False: try: dia=int(input('Día: ')) except ValueError: print('Día debe ser entero') try: mes=int(input('Mes: ')) except ValueError: print('Mes debe ser entero') if ((mes in(1,3,5,7,8,10,12) and dia in range(1,32)) or (mes in (4,6,9,11) and dia in range(1,31)) or(mes==2 and dia in range(1,29))): print(dia,'/',mes) else: print(dia,'/',mes,'no es una fecha válida')</pre>		1
2	<pre>dia=0 mes=0 while dia==0: try: dia=int(input('Día: ')) mes=int(input('Mes: ')) except ValueError: print('Día y Mes deben ser enteros') dia=0 mes=0 if ((mes in(1,3,5,7,8,10,12) and dia in range(1,32)) or (mes in (4,6,9,11) and dia in range(1,31)) or(mes==2 and dia in range(1,29))): print(dia,'/',mes) else: print(dia,'/',mes,'no es una fecha válida') dia=0 mes=0</pre>	X	2
3	<pre>try: dia=int(input('Día: ')) mes=int(input('Mes: ')) except ValueError: print('Día y Mes deben ser enteros') dia=0 mes=0 if ((mes in(1,3,5,7,8,10,12) and dia in range(1,32)) and(mes in (4,6,9,11) and dia in range(1,31))): print(dia,'/',mes) else: print(dia,'/',mes,'no es una fecha válida')</pre>		3
4	<pre>while dia==0: try: dia=int(input('Día: ')) mes=int(input('Mes: ')) except ValueError: print('Día y Mes deben ser enteros') if ((mes in(1,3,5,7,8,10,12) and dia in range(1,32)) and(mes in (4,6,9,11) and dia in range(1,31)) and(mes==2 and dia in range(1,29))): print(dia,'/',mes) else: print(dia,'/',mes,'no es una fecha válida') mes=0</pre>		4

Ejercicio 4 – Tema 2		1 Pto	
<p>En el siguiente programa:</p> <pre>amigos=['mono','teffi','lau','chachi','lauta','santi'] zodiaco=... print('Signos de mis amigos') for nom in amigos: if nom in zodiaco: print(nom.upper(),'es',zodiaco[nom][1],'del',zodiaco[nom][0])</pre> <p>Cuál es la definición correcta de la estructura zodiaco para que funcione adecuadamente y genere la siguiente salida:</p> <p>Signos de mis amigos MONO es piscis del 26/02 TEFFI es géminis del 23/05 CHACHI es virgo del 18/09 LAUTA es piscis del 10/03 SANTI es géminis del 03/06</p>			
1	<pre>zodiaco={'joaco':['25/03','aries'], 'lauta':['10/03','piscis'], 'teffi':['23/05','géminis'], 'mono':['26/02','piscis'], 'mati':['12/10','libra'], 'santi':['03/06','géminis'], 'chachi':['18/09','virgo']}</pre>	X	1
2	<pre>zodiaco=[['joaco','25/03','aries'], ['lauta','10/03','piscis'], ['teffi','23/05','géminis'], ['mono','26/02','piscis'], ['mati','12/10','libra'], ['santi','03/06','géminis'], ['chachi','18/09','virgo']]</pre>		2
3	<pre>zodiaco=['joaco','25/03','aries','lauta', '10/03','piscis','teffi','23/05', 'géminis','mono','26/02','piscis', 'mati','12/10','libra','santi','03/06', 'géminis','chachi','18/09','virgo']</pre>		3
4	<pre>zodiaco={('25/03','aries'):'joaco', ('10/03','piscis'):'lauta', ('23/05','géminis'):'teffi', ('26/02','piscis'):'mono', ('12/10','libra'):'mati', ('03/06','géminis'):'santi', ('18/09','virgo'):'chachi'}</pre>		4

Ejercicio 5 – Tema 2		1 Pto	
<p>¿Qué salida produce el siguiente programa?</p> <pre> a=[1,6,10,3] b=[4,5,6,2,9,11] c=[] if len(a)<len(b): hasta=len(a) else: hasta=len(b) for i in range(hasta): num=abs(a[i]-b[i]) if num>0 and num%2==0: c.append(num) if hasta==len(a) and hasta!=len(b): for i in range(hasta,len(b)): num=abs(b[i]-1) if num>0 and num%2==0: c.append(num) elif hasta==len(b) and hasta!=len(a): for i in range(hasta,len(a)): num=abs(a[i]-1) if num>0 and num%2==0: c.append(num) print(c) </pre> <p>Nota: La función abs() devuelve el valor absoluto de su argumento Ej: abs(-2) -> 2</p>			
1	[1, 6, 10, 3]		1
2	[-3, 1, 4, 1]		2
3	[-11, 9, -2, 6, -5, 4]		3
4	[4, 8, 10]	X	4

Ejercicio 6 – Tema 2		2 Ptos	
<p>Para el siguiente DataFrame oficinas</p> <pre> regiónprovincia empleados ventas 0 cuyo san juan 3 72500.0 1 litoral santa fé 10 115000.0 2 cuyo mendoza 5 91350.9 3 patagonia chubut 2 44000.5 4 cuyo san luis 11 96725.0 5 pampa buenos aires 8 NaN </pre> <p>¿Qué instrucción produce la siguiente salida?</p> <p>44000.5</p>			
1	oficinas[oficinas['empleados']<10]['ventas'].min()	X	1
2	oficinas.groupby('región')['ventas'].sum()		2
3	oficinas.sample(3)		3
4	oficinas[['ventas','provincia']]		4

Ejercicio 7 – Tema 2		2 Ptos	
Para el siguiente programa:			
<pre> def apertura (archivo, modo): arch=open (archivo,modo) return arch nomArch=apertura('datos1.txt',...) #1 nomArch.write('Archivo nuevo que contiene texto genérico\n') nomArch.close() nomArch=apertura('datos1.txt', ...) #2 for lin in nomArch: print(lin.strip('\n')) print() nomArch.close() nomArch=apertura('datos1.txt', ...) #3 nomArch.write('Agrego esta línea') nomArch.close() nomArch=apertura('datos1.txt', ...) #4 lineas=nomArch.readlines() for lin in lineas: print(lin.strip('\n')) nomArch.close() </pre>			
¿Qué secuencia de modos de apertura debe usarse para que produzca la siguiente salida? Considerando que <i>datos1.txt</i> no existe antes de correr el programa			
Archivo nuevo que contiene texto genérico			
Archivo nuevo que contiene texto genérico Agrego esta línea			
1	'r' 'r' 'w' 'r'		1
2	'r' 'a' 'r' 'w'		2
3	'w' 'r' 'a' 'r'	X	3
4	'a' 'a' 'r' 'r'		4

Ejercicio 8 – Tema 2		2 Ptos	
En el siguiente programa:			
<pre>def separa(t): return t[-1] in 'AEIOUÁÉÍÓÚ' def todoIgual(t): return t.upper() nombres=['juan', 'Juana', 'ANDRÉS', 'ANalía', 'aleJO'] α print(finVocal)</pre>			
¿Qué secuencia de código es la correcta para reemplazar α de modo que el programa produzca la siguiente salida?			
['JUANA', 'ANALÍA', 'ALEJO']			
Nota: Se puede seleccionar un elemento de una string de atrás para adelante con el signo –			
Ej: 'mi muñeca linda'[-2] -> 'd'			
1	<code>finVocal=list(filter(separa,nombres))</code> <code>unificado=list(map(todoIgual,nombres))</code>		1
2	<code>unificado=list(map(todoIgual,nombres))</code> <code>finVocal=list(map(separa,unificado))</code>		2
3	<code>unificado=list(map(todoIgual,nombres))</code> <code>finVocal=list(filter(separa,unificado))</code>	X	3
4	<code>unificado=list(map(separa,nombres))</code> <code>finVocal=list(filter(todoIgual,unificado))</code>		4

Ejercicio 9 – Tema 2		2 Ptos	
<p>¿Qué salida produce el siguiente programa?</p> <pre>nombres=['juan', 'mario', 'pedro', 'inés'] picada=['queso', 'crudo', 'aceituna', 'maní'] txt='una table de '.capitalize()+picada[0].upper() for nom in picada[1:]: txt+=", " txt+=nom.upper() txt+=' y otros para los que estábamos: ' txt+=nombres[-1].capitalize() for i in range(2,len(nombres)): txt+=", "+nombres[-i].capitalize() txt+=' y '+nombres[0].capitalize() print(txt)</pre> <p>Notas: Se puede referenciar una porción de una string con [:] Ej: 'hola a todos'[:2] -> 'ho'</p> <p>Se puede referenciar caracteres de una string de atrás para adelante con índices negativos Ej: 'mucho ruido y pocas nueces'[-2]-> 'e'</p> <p>El método capitalize() devuelve todas las letras de una string en minúsculas, exceptuando el primer carácter que es mayúscula (si es letra) Ej: 'listo de una vez'.capitalize()->'Listo de una vez'</p>			
1	QUESO, CRUDO, ACEITUNA y MANÍ para los que estábamos: JUAN, MARIO, PEDRO, INÉS		1
2	Una table de QUESO, CRUDO, ACEITUNA, MANÍ y otros para los que estábamos: Inés, Pedro, Mario y Juan	X	2
3	una table de Queso, Crudo, Aceituna, Maní Y OTROS PARA LOS QUE ESTÁBAMOS: inés, pedro, mario y juan		3
4	Junto a Inés, Juan, Mario y Pedro pedimos una tabla de maní, aceituna, crudo y QUESO		4