

PENSAMIENTO COMPUTACIONAL (90)

.UBAXXI

TEMA 1

EXAMEN: RECUPERATORIO 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:

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 1 Pto	Ej 7 1 Pto	Ej 8 2 Ptos	Ej 9 2 Ptos	Ej 10 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.

0101 – 1 Pto			
¿Cuál es la línea de código que debe ir en la línea punteada?			
<pre>print('Superficie del cuadrado') #Línea a remplazar sup=lado*lado print('La superficie del cuadrado de lado',lado,'cm es',sup)</pre>			
1	lado=float(input('Lado en cm: '))	X	1
2	lado=int(print('Lado en cm: '))		2
3	a=input('Lado en cm: ').lower()		3
4	lado == input('Lado en cm: ') and lado > 0		4

0201 – 1 Pto			
¿Qué se muestra en la pantalla?			
<pre>a=95 b=5 c=10 print(int(a/c)+2*(b-1))</pre>			
1	1.5		1
2	0		2
3	17	X	3
4	-46.99		4

0301 – 1 Pto			
¿Cuál de los siguientes programas evalúa menos condiciones si <i>flor='rosa'</i> y <i>mes='marzo'</i> ?			
1	<pre>if mes in ('enero','febrero','marzo','abril'): if flor=='cala': print('No hay') elif flor in ('rosa','gerbera','margarita'): print('Hay') if mes in ('octubre','septiembre'): print('seguro') else: print('Mmmm, puede ser ...')</pre>		1
2	<pre>if mes=='enero' and flor in ('rosa','gerbera','margarita'): print('Hay') elif mes=='febrero' and flor in ('rosa','gerbera','margarita'): print('Hay') elif mes=='abril' and flor in ('rosa','gerbera','margarita'): print('Hay') elif mes=='marzo' and flor in ('rosa','gerbera','margarita'): print('Hay') if mes in ('octubre','septiembre'): print('seguro') else: print('Mmmm, puede ser ...')</pre>		2
3	<pre>if mes=='enero' and flor in ('rosa','gerbera','margarita'): print('Hay') elif mes=='febrero' and flor in ('rosa','gerbera','margarita'): print('Hay') elif mes=='abril' and flor in ('rosa','gerbera','margarita'): print('Hay') else: if mes in ('octubre','septiembre'): print('seguro')</pre>		3
4	<pre>if mes in ('enero','febrero','marzo','abril'): if flor in ('rosa','gerbera','margarita'): print('Hay') elif flor=='cala': print('No hay') elif mes in ('octubre','septiembre'): if flor=='rosa' or flor=='gerbera': print('seguro') else: print('Mmmm, puede ser ...')</pre>	X	4

0401 – 1 Pto			
¿Cuál condición NO es equivalente a las otras tres? Sugerencia: probalas con <i>flor='margarita'</i> , <i>mes='marzo'</i> y <i>color='blanco'</i>			
1	<pre>(flor in ('margarita','jarzmin') or (color=='rojo' and mes=='marzo') or (flor=='rosa' and mes=='abril'))</pre>		1
2	<pre>(flor in ['rosa'] and mes=='abril' and flor in ['margarita','jarzmin'] and color=='rojo' and mes=='marzo')</pre>	X	2
3	<pre>((flor in ['rosa'] and mes=='abril') or flor=='margarita' or flor=='jarzmin' or (color=='rojo' and mes=='marzo'))</pre>		3
4	<pre>((flor in ['rosa'] and mes=='abril') or (flor in ['margarita','jarzmin'] and (color=='blanco' or color!='blanco')) or (color=='rojo' and mes=='marzo'))</pre>		4

0501 – 1 Pto			
<p>¿Cuál de los siguientes programas no dibuja lo siguiente?</p> <p>LUNAR U A N N A U RANUL</p> <p>Notas: Se puede invertir una secuencia (una string es un tipo de secuencia) usando el siguiente tip: Ej: a='Hola' b=a[::-1] -> 'aloH'</p> <p>El operador * aplicado a una string la repite n veces Ej: 'una'*2 -> 'unauna'</p>			
1	<pre> palabra='LUNAR' print(palabra) for i in range(1,4): print(palabra[i]+' '*3+palabra[4-i]) print(palabra[::-1]) </pre>		1
2	<pre> palabra='LUNAR' alReves=palabra[::-1] hueco=' ' # 3 blancos print(palabra) for i in range(1,4): print(palabra[i]+hueco+alReves[i]) print(palabra[::-1]) </pre>		2
3	<pre> palabra='LUNAR' i=0 while i<len(palabra): if i not in (0,4): cabecera=palabra[i] cola=palabra[4-i] print(cabecera+' '+cola) # hay 3 blancos elif i==0: print(palabra) else: palabra=palabra[::-1] print(palabra) i+=1 </pre>		3
4	<pre> palabra='LUNAR' for i in range(len(palabra)): palabra=palabra[::-1] print(palabra) </pre>	X	4

0601 – 1 Pto			
¿Cuál de los siguientes programas deja a <i>lista</i> de la siguiente manera?			
lista=[1, 4, 2, 0, 3]			
Notas:			
El método count() cuenta la cantidad de ocurrencias del argumento en una secuencia			
Ej:			
a='casa'			
a.count('a') -> 2			
El método remove() saca de la lista la primer ocurrencia del argumento			
Ej:			
lista=[1, 2, 3, 4, 2, 1, 5, 2]			
lista.remove(2) -> lista=[1, 3, 4, 2, 1, 5, 2]			
El método reverse() da vuelta una lista (la invierte)			
Ej:			
lista=[1,2,3]			
lista.reverse() -> lista=[3,2,1]			
1	<pre>lista=[1,4,2,0,1,1,3,2,1] for num in lista: lista.reverse() while lista.count(num)>1: lista.remove(num) lista.reverse()</pre>	X	1
2	<pre>lista=[1,4,2,0,1,1,3,2,1] for num in lista: if lista.count(num)==1: lista.remove(num)</pre>		2
3	<pre>lista=[] for num in range(1,10): lista.append(num)</pre>		3
4	<pre>lista=[] for num in range(1,10): lista.append(num) lista.reverse() elem=lista[0] lista.remove(elem)</pre>		4

0701 – 1 Pto			
¿Qué muestra el siguiente programa?			
<pre>def maximo(meses, let): mayor=0 m='no hay' for mes in meses: total=mes.count(let) if total>mayor: mayor=total m=mes return m meses=['enero', 'febrero', 'marzo', 'abril', 'mayo', 'junio', 'julio', 'agosto', 'septiembre', 'octubre', 'noviembre', 'diciembre'] letras='eorki' ganadores={} for let in letras: ganadores[let]=maximo(meses, let) print(ganadores)</pre>			
1	{3: 'e', 2: 'o', 2: 'r', 0: 'k', 2: 'i'}		1
2	{'e': 'septiembre', 'o': 'agosto', 'r': 'febrero', 'k': 'no hay', 'i': 'diciembre'}	X	2
3	{'i': 2, 'k': 0, 'r': 2, 'o': 2, 'e': 3}		3
4	[3, 2, 2, 0, 2]		4

0801 – 2 Ptos			
¿Cuál código muestra exactamente 3 *?			
1	for i in range(1,5,2): print('*')		1
2	for i in range(6,6): print('*') for car in 'tal': print('*') i=5 while i>6: print('*') i+=1	X	2
3	for i in range(1,4): for j in range(2): print('*') print('*')		3
4	for i in range(10,4,-4): print('*')		4

0901 – 2 Ptos			
¿Qué muestra el siguiente programa?			
<pre>def llena(a): nombres='' for nom in a: nombres+=nom.lower() lis=[] voc='aeiou' for v in voc: lis.append(nombres.count(v)) return lis a=['ANA', 'BETTY', 'Alina', 'elena'] b=llena(a) print(b)</pre>			
1	['ANA', 'BETTY', 'Alina', 'elena']		1
2	['a', 'e', 'i', 'o', 'u']		2
3	[5, 3, 1, 0, 0]	X	3
4	[5, 5, 5, 5, 5]		4

1001 – 2 Ptos			
En el siguiente programa:			
<pre>def ubica(...): - - - #Ppal dias=['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday'] iniciales={} for d in dias: ubica(iniciales,d)</pre>			
¿Cuál es la función ubica() adecuada para que el diccionario iniciales quede de la siguiente manera?			
<pre>iniciales={'M': ['Monday'], 'T': ['Tuesday', 'Thursday'], 'W': ['Wednesday'], 'F': ['Friday'], 'S': ['Saturday', 'Sunday']}</pre>			
1	<pre>def ubica(iniciales,d): if d[0] in iniciales: iniciales[d[0]].append(d) else: iniciales[d[0]]=d</pre>	X	1
2	<pre>def ubica(iniciales): iniciales[d]=0</pre>		2
3	<pre>def ubica(iniciales,d): iniciales[d]=d</pre>		3
4	<pre>def ubica(): if d in iniciales: iniciales[d].append(d[0]) else: iniciales[d]=[d[0]]</pre>		4



Talón de Control para el Alumno

	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	
1												1
2												2
3												3
4												4