

APELLIDO:	CALIFICACIÓN:
NOMBRE:	
DNI (registrado en SIU Guaraní):	
E-MAIL:	
TEL:	DOCENTE (nombre y apellido):
AULA:	

Duración del examen: 1:20h. Completar con **letra clara, mayúscula e imprenta**. El examen consta de 11 preguntas de opción múltiple. Cada pregunta tiene una y sólo una respuesta correcta.

Las respuestas deben completarse en la siguiente matriz:

Opción	EJ. 1	EJ. 2	EJ. 3	EJ. 4	EJ. 5	EJ. 6	EJ. 7	EJ. 8	EJ. 9	EJ. 10	EJ. 11
1											
2											
3											
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 la de la matriz.

Ejercicio 0102 - 1 punto		
Si <code>dip='ia'</code> y <code>txt='criada y cambiada'</code> ¿Cuál condición da Verdadero?		
1	<code>dip in txt and txt.count('o')>0 and len(dip)%2!=0</code>	1
2	<code>(dip in txt) or (txt.count('o')==0 and len(dip)%2==0)</code>	X 2
3	<code>dip.upper() in txt and txt.count('o')>=0 and len(dip)%2==1</code>	3
4	<code>(dip in txt.upper() and txt.count('o')>=0) or (len(dip)%2==3)</code>	4

Ejercicio 0202 - 1 punto		
¿Qué imprime el siguiente programa?		
<pre>txt='habemus condiciones'.upper() if txt==txt.lower() and txt.count(' ')<1: print('una') elif txt==txt.lower() and txt.count('')==1: print('dos') elif txt.count(' ')<1: print('tres') else: print('cuatro')</pre>		
1	una	1
2	dos	2
3	tres	3
4	cuatro	X 4

Ejercicio 0302 - 1 punto			
<p>¿Cuántos * imprime el siguiente programa?</p> <pre> for j in range(10,0,-3): print('*') cant=3 while j<=cant: for i in range(3,60,20): print('*') j+=2 </pre>			
1	10	X	1
2	2		2
3	8		3
4	15		4

Ejercicio 0401 - 1 punto			
<p>¿Qué imprime el siguiente programa si los datos ingresados por el usuario (n1 y n2) son 46 y 78?</p> <pre> n1=input('Entero: ') n2=input('Entero: ') result=int(n2+n1) print('Tomando los dos valores: '+str(result)) </pre>			
1	Tomando los dos valores: 46 78		1
2	Tomando los dos valores:		2
3	Tomando los dos valores: 124.00		3
4	Tomando los dos valores: 7846	X	4

Ejercicio 0502 - 1 punto			
<p>¿Qué muestra en el segundo print() el siguiente código?</p> <pre> def paraLlenar(txt,rellenar,l): for i in range(len(txt)): if txt[i]==l: rellenar[i]=l return ' '.join(rellenar) pal='conmiseración' completar=['_']*len(pal) print('Ahorcado') print(paraLlenar(pal,completar,'n')) </pre>			
1	CONMISERACIÓN		1
2	C_N_____N		2
3	__n_____n	X	3
4	__N_____		4

Ejercicio 0602 - 1 punto		
<p>¿Qué contenido de <i>lista</i> mostrará el siguiente programa?</p> <pre> lista=[0]*3 nombres=['ANA', 'luis', 'GERónimo', 'Lucrecia', 'dante', 'ema'] for nom in nombres: if nom==nom.lower(): lista[0]+=1 elif nom==nom.upper(): lista[1]+=1 else: lista[2]+=1 elem=lista.pop(0) lista.append(elem) print(lista) </pre>		
1	[3, 1, 2, 3]	1
2	[1, 2, 3]	X 2
3	[3, 1]	3
4	[1, 1, 1]	4

Ejercicio 0702 - 1 punto		
<p>¿Qué contenido tendrán <i>totales</i> y <i>pierde</i> al correr el siguiente programa?</p> <pre> juego=[[[1,0,11,6], 'juan'], [3,0,22,1], 'elena'], [6,16,11,4], 'inés'] totales=[] for jug in juego: totales.insert(0, [sum(jug[0]), jug[1]]) pierde=min(totales) print(totales) print(pierde) </pre>		
1	[[37, 'inés'], [26, 'elena'], [18, 'juan']] [18, 'juan']	X 1
2	[18, 'juan'] [[37, 26, 18]]	2
3	[['inés', 37], ['elena', 26], ['juan', 18]] [26, 'ELENA']	3
4	[[18, 'JUAN'], [26, 'ELENA'], [37, 'INÉS']] 18	4

Ejercicio 0802 - 1 punto		
<p>¿Qué expresión debería ir en los ... para que el siguiente programa pueda obtener el total a pagar de una venta realizada, teniendo los ítems código y cantidad vendidos en la lista <i>venta</i> y los precios unitarios vinculados a cada artículo, por su código en el diccionario <i>precios</i>? <i>venta</i> contiene listas con código de artículo y cantidad vendida <i>precios</i> contiene como claves códigos de artículos y como valor su precio unitario</p> <pre> venta=[['aguja01', 25], ['hilox25', 1], ['hiloC72', 3]] precios={'botonXS003':10.25, 'botonXL105':320, 'aguja01':10.25, 'cierreDES50':780, 'hilox25':130, 'hiloC72':120} totalAPagar=0 for i in range(len(venta)): clave=venta[i][0] cant=venta[i][1] totalAPagar=... print('A pagar: \$', totalAPagar) </pre>		
1	cant*precios[i]	1
2	totalAPagar + cant*precios[clave]	X 2
3	totalAPagar + cant + precios[cant]	3
4	totalAPagar*cant + precios[clave]	4

Ejercicio 0902 - 1 punto			
<p>¿Cuál será la salida del programa?</p> <pre> propietarios={'AC 233 FQ':'Ignacio González', 'AE 002 MC':'Arturo Paz', 'AC 105 OP':'Isabel Paz', 'IKF 937':'Isabel Andújar'} patOrd=sorted(proprietarios) for p in patOrd: print(proprietarios[p],p) </pre>			
1	<p>AE 002 MC - Arturo Paz AC 233 FQ - Ignacio González IKF 937 - Isabel Andújar AC 105 OP - Isabel Paz</p>		1
2	<p>Isabel Paz AC 105 OP Ignacio González AC 233 FQ Arturo Paz AE 002 MC Isabel Andújar IKF 937</p>	X	2
3	<p>Arturo Paz Ignacio González Isabel Andújar Isabel Paz</p>		3
4	<p>AC 105 OP I Paz AC 233 FQ I González AE 002 MC A Paz IKF 937 I Andújar</p>		4

Ejercicio 1002 - 2 puntos			
<p>¿Qué dibuja el siguiente código?</p> <pre> pal=['O', 'T', 'O', 'Ñ', 'O'] tope=len(pal)-1 print(' '.join(pal)) for i in range(1,tope): print(pal[i]+' '*7+pal[i]) print(' '.join(pal)) </pre>			
1	<p>OTOÑO OÑOTO</p>		1
2	<p>00000000</p>		2
3	<p>OTOÑO T T O O Ñ Ñ OTOÑO</p>	X	3
4	<p>O OT OTO OTOÑ OTOÑO</p>		4

Ejercicio 1102 - 2 puntos			
<p>¿Cuál función trabaja correctamente en el programa para informar cuántos valores están por debajo del promedio, cuántos por arriba y cuántos coinciden con el promedio?</p> <pre> def mitades (...): - - - numeros=[-10,20,55,0,35] prome=sum(numeros)/len(numeros) resul=mitades(numeros,prome) print('Hay',resul[0],'por debajo de',prome,',',',', resul[1],'por arriba y',resul[2],'coinciden con él')</pre>			
1	<pre> def mitades(l,pr): totales=[0,0,0] for num in l: if num<pr: totales[0]+=1 elif num==pr: totales[2]+=1 else: totales[1]+=1 return totales</pre>	X	1
2	<pre> def mitades(pr): totales=[] for num in l: if num<pr: totales[0]+=1 elif num==pr: totales[1]+=1 else: totales[2]+=1 return totales</pre>		2
3	<pre> def mitades(pr,l): totales=[] for num in l: if num<pr: totales.append(1) elif num==pr: totales+= [1] else: totales.append(1) return totales[0],totales[2]</pre>		3
4	<pre> def mitades(l): totales=[] for num in l: if num<sum(l)/3: totales.append(1) elif num==sum(l)/2: totales+= [1] else: totales.append(1) return sum(totales)</pre>		4