

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

Duración del examen: 1:20h. Completar con **letra clara, mayúscula e imprenta**. El examen consta de 10 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
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 0103 - 1 punto

Dado el siguiente condicional:

if a in c:
 print('De una')
elif (a.upper()>'E' and b) or (a<'m') or (not b and a=='*'):
 print('Sale segunda')
elif a>'c' and a<'u':
 print('Última opción')
Qué valores de **a,b** y **c** permiten que se imprima: **Última opción** ?

1	a='E', b=False, c='dale'		1
2	a='o', b=False, c='una'	X	2
3	a='e', b=True, c='dale'		3
4	a='m', b=True, c='una'		4

Ejercicio 0203 - 1 punto

¿Qué muestra por pantalla el siguiente programa?

j=0
for i in range(3):
 j+=10
while i<5:
 j-=1
 m=j-10
 for k in range(10,m):
 j-=1
 i+=1
print(j)

1	80		1
2	18	X	2
3	0		3
4	-75		4

Ejercicio 0303 - 1 punto

¿Con qué contenido acabará el diccionario *contCons* cuando se ejecute el siguiente programa?

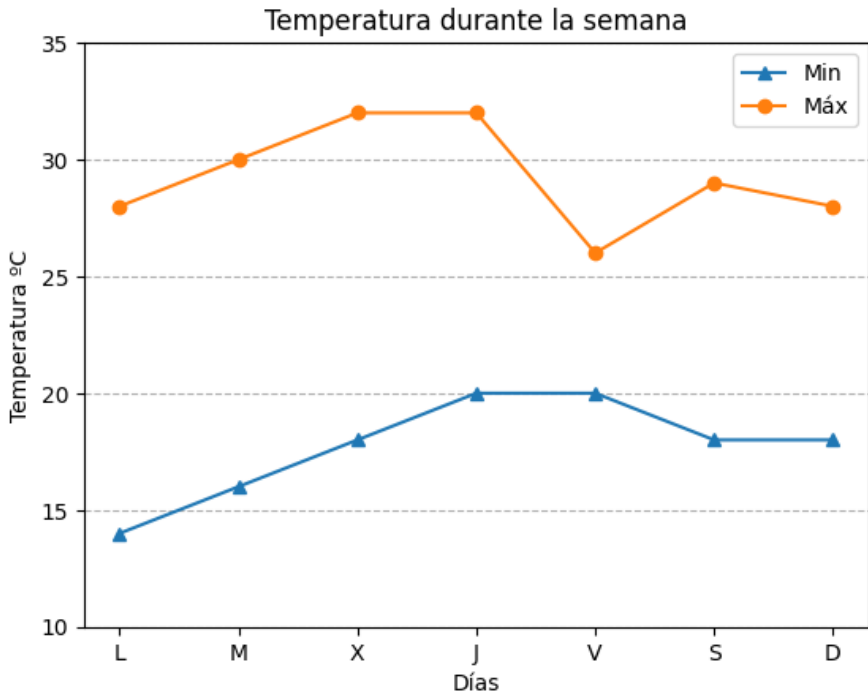
<pre>parrafo='eventualmente la ciudad DESPERTÓ DE SU LETARGO estival.' contCons={} for let in parrafo: if let.isalpha() and let.lower()not in 'aeiouáéíóú': if let.lower() in contCons: contCons[let.lower()] +=1 else: contCons[let.lower()]=1</pre> Nota: El método <i>isalpha()</i> devuelve True si el argumento es una letra, False en caso contrario			
1	{'v': 2, 'n': 1, 't': 5, 'l': 4, 'm': 1, 'c': 1, 'd': 4, 's': 3, 'p': 2, 'r': 2, 'g': 1}		1
2	{26:'aeiouáéíóú}		2
3	{'E': 9, 'U': 3, 'A': 5, 'I': 2, 'Ó': 1, 'O': 1}		3
4	{'v': 2, 'n': 2, 't': 5, 'l': 4, 'm': 1, 'c': 1, 'd': 4, 's': 3, 'p': 1, 'r': 2, 'g': 1}	X	4

Ejercicio 0403 - 1 punto			
Si <i>lis</i>=[1,22,4,1,6,5,1,10], cuál código NO logra traer todos los pares al principio de la lista? <i>lis</i> debería quedar [22, 4, 6, 10, 1, 1, 5, 1] Es decir, del mismo tamaño pero con todos los pares que tenga al inicio, y los impares al fondo en el mismo orden original.			
1	<pre>def par(n): return n%2==0 lis=... pares=list(filter(par,lis)) for n in lis: if n%2==1: pares.append(n) lis=pares</pre>		1
2	<pre>def par(n): return n%2==0 lis=... original=lis.copy() pares=list(filter(par,lis)) for n in lis: if n%2!=1: original.insert(0,n) lis=original</pre>	X	2
3	<pre>pares=[] impares=[] for n in lis: if n%2==0: pares.append(n) else: impares.append(n) lis=pares+impares</pre>		3
4	<pre>def par(n): return n%2==0 def impar(n): return n%2==1 lis=... pares=list(filter(par,lis)) impares=list(filter(impar,lis)) lis=pares+impares</pre>		4

Ejercicio 0503 - 1 punto			
¿Qué versión de la función <i>negativo</i> funcionará correctamente en el siguiente programa para obtener un ingreso por teclado de un número negativo? La función debe insistir hasta que el número ingresado sea efectivamente un valor negativo. <pre>def negativo(...): - - - a=negativo() b=negativo() print(a-b)</pre>			
1	<pre>def negativo(): dale=True while dale: try: num=input('Número negativo: ') num=float(num) if num>=0: dale=True else: print('Ingrese un número negativo') dale=False except ValueError: print('Ingrese un número negativo')</pre>		1
2	<pre>def negativo(): while True: try: num=float(input('Número negativo: ')) if num<0: return num else: print('Ingrese un número negativo') except ValueError: print('Ingrese un número negativo')</pre>	X	2
3	<pre>def negativo(): dale=False while dale: try: num=input('Número negativo: ') num=float(num) if num<0: dale=False else: print('Ingrese un número negativo') dale=True except ValueError: print('Ingrese un número negativo')</pre>		3
4	<pre>def negativo(): try: num=input('Número negativo: ') num=float(num) if num<0: return num return 0 except ValueError: print('Ingrese un número negativo')</pre>		4

Ejercicio 0603 - 1 punto			
¿Qué muestra el siguiente programa? <pre>def letras(num): let='totales' return let[abs(num)%7] lista=[2,22,6,10,-8] nueva=list(map(letras,lista)) print(nueva)</pre>			
1	[]		1
2	['0', '0']		2
3	['t', 'o', 's', 'a', 'o']	X	3

4	['T', 'O', 'T', 'A', 'L']		4
---	---------------------------	--	---

Ejercicio 0703 - 1 punto																																											
Temperatura durante la semana <table><caption>Temperatura durante la semana (°C)</caption><tr><th>Día</th><th>Min (°C)</th><th>Máx (°C)</th></tr><tr><td>L</td><td>14</td><td>28</td></tr><tr><td>M</td><td>16</td><td>30</td></tr><tr><td>X</td><td>18</td><td>32</td></tr><tr><td>J</td><td>20</td><td>32</td></tr><tr><td>V</td><td>20</td><td>26</td></tr><tr><td>S</td><td>18</td><td>29</td></tr><tr><td>D</td><td>18</td><td>28</td></tr></table> ¿Cuál de las siguientes líneas de código NO SE CORRESPONDE con la figura? <table><tr><td>1</td><td><code>ax.grid(axis = 'y', linestyle = 'dashed')</code></td><td></td><td>1</td></tr><tr><td>2</td><td><code>ax.set_title("Temperatura durante la semana")</code></td><td></td><td>2</td></tr><tr><td>3</td><td><code>ax.plot(dias, temperaturas_min, marker = '^', label = 'Min')</code> <code>ax.scatter(dias, temperaturas_max, marker = 'o', label = 'Máx')</code></td><td>X</td><td>3</td></tr><tr><td>4</td><td><code>ax.set_ylabel("Temperatura °C")</code></td><td></td><td>4</td></tr></table>				Día	Min (°C)	Máx (°C)	L	14	28	M	16	30	X	18	32	J	20	32	V	20	26	S	18	29	D	18	28	1	<code>ax.grid(axis = 'y', linestyle = 'dashed')</code>		1	2	<code>ax.set_title("Temperatura durante la semana")</code>		2	3	<code>ax.plot(dias, temperaturas_min, marker = '^', label = 'Min')</code> <code>ax.scatter(dias, temperaturas_max, marker = 'o', label = 'Máx')</code>	X	3	4	<code>ax.set_ylabel("Temperatura °C")</code>		4
Día	Min (°C)	Máx (°C)																																									
L	14	28																																									
M	16	30																																									
X	18	32																																									
J	20	32																																									
V	20	26																																									
S	18	29																																									
D	18	28																																									
1	<code>ax.grid(axis = 'y', linestyle = 'dashed')</code>		1																																								
2	<code>ax.set_title("Temperatura durante la semana")</code>		2																																								
3	<code>ax.plot(dias, temperaturas_min, marker = '^', label = 'Min')</code> <code>ax.scatter(dias, temperaturas_max, marker = 'o', label = 'Máx')</code>	X	3																																								
4	<code>ax.set_ylabel("Temperatura °C")</code>		4																																								
Ejercicio 0803 - 2 puntos																																											
¿Qué versión de <i>elige</i> funcionará correctamente en este programa? <pre>def elige(...): - - - salsas={'f': 'Filetto', 'b': 'Boloñesa', 'c': 'Crema', 'p': 'Pesto', 's': 'Sin salsa'} pastas={'f': 'Fideos cinta', 'r': 'Ravioles de calabaza', 'm': 'fideos Moñito'} plato=[] print('Armá tu plato de Pasta, hasta 2 salsas') plato.append(elige(pastas, 'Qué pasta?')) for i in range(1,3): plato.append(elige(salsas, 'Salsa '+str(i))) print('Mi plato de pastas es',plato[0], 'con',plato[1], 'y',plato[2])</pre>																																											
1	<pre>def elige(dicci, cartel): print('Elegí tu opción con la letra resaltada') print(cartel) for opc in dicci: print(dicci[opc]) sel=input().lower() while sel not in dicci: sel=input().lower() return dicci[sel]</pre>	X	1																																								
2	<pre>def elige(dicci): print('Elegí tu opción con la letra resaltada') print('cartel') for opc in dicci: print(opc) sel=input().upper() while sel in dicci: sel=input().upper() return dicci[sel]</pre>		2																																								

3	<pre>def elige(cartel,dicci): print('Elegí tu opción con la letra resaltada') print(cartel) for opc in dicci: print(dicci[opc]) sel=input().upper() return dicci[sel] while sel not in dicci: sel=input()</pre>		3
4	<pre>def elige(cartel): print('Elegí tu opción con la letra resaltada') print(cartel) for opc in dicci: print(dicci[opc]) sel='f' return dicci[sel]</pre>		4

Ejercicio 0903 - 2 puntos			
¿Qué valor para el argumento <i>modo</i> debería pasársele a la función <i>abrir</i> (en la primera y en la segunda invocación) para permitir que el programa invierta el orden de las palabras del archivo <i>ar.txt</i>? <pre>def abrir(arch,modo): archivo=open(arch,modo,encoding='utf-8') return archivo verbos=abrir('ar.txt',...) #Primera lineas=verbos.readlines() verbos.close() lineas.reverse() verbos=abrir('ar.txt',...) #Segunda for lin in lineas: verbos.write(lin) verbos.close()</pre>			
1	Primera: 'a+' Segunda: 'w'		1
2	Primera: 'r' Segunda: 'w+'	X	2
3	Primera: 'r' Segunda: 'a'	X	3
4	Primera: 'r' Segunda: 'w'	X	4

Dado que el enunciado es ambiguo y se puede interpretar como agregar o reemplazar, las opciones marcadas son las correctas.

Ejercicio 1003 - 2 puntos					
Para el siguiente DataFrame <i>arts</i> :					
	Cgo	Desc	Pr	Unit	Exist
0	LIB001	Lápiz negro HB	450.0	55.0	
1	PAP112	Res Ledesma 70gr AA	2700.0	NaN	
2	PAP099	Blck cuad Rivadavia A4	958.5	2.0	
3	MAP070	Planisferio nro 5	210.0	13.0	
4	LIB015	Marc negro Sylvapen	675.0	11.0	
5	LIB118	Regl Pizzini 30cm	387.0	NaN	
¿Qué instrucción produce la siguiente salida?					
	Cgo	Desc	Pr	Unit	Exist
0	LIB001	Lápiz negro HB	450.0	55.0	
1	PAP112	Res Ledesma 70gr AA	2700.0	NaN	
2	PAP099	Blck cuad Rivadavia A4	958.5	2.0	
4	LIB015	Marc negro Sylvapen	675.0	11.0	
5	LIB118	Regl Pizzini 30cm	387.0	NaN	
1	arts[arts['Pr Unit'].between(300,2700)]				X 1
2	arts['Exist'].sum()				2
3	arts[['Pr Unit', 'Desc', 'Cgo']]				3

4	arts[(arts['Pr Unit']<900)&(arts['Exist']>10)]		4
---	--	--	---