

Ejercicio 1 – Tema 4			1 Pto
¿Qué muestra por pantalla el siguiente programa? <pre>nivel='secundario' orientacion='química' materia='matemáticas' anio=4 if nivel not in ('secundario','terciario') and nivel[0]=='s': print('****') elif nivel in ('primario','secundario'): print('*') if materia in ('química orgánica','lengua','matemáticas'): print('*') if orientacion!='electrónica' and anio>=4: print('*') elif anio<3: print('*') else: print('***') print('*')</pre>			
A	* * * *	X	A
B	*** *		B
C	* **		C
D	*****		D

Ejercicio 2 – Tema 4		1 Pto	
¿Cuál programa produce la siguiente salida?			
2*2=4			
Nota: El método replace() permite remplazar una substring (dentro de otra) por un texto diferente Ej: txt='hola, hola Chola !!' txt.replace('hola','chau') -> 'chau, chau Cchau !!'			
A	<pre>exp='DOS x DOS es cuatro' digitos=('cero','uno','dos','tres','cuatro', 'cinco','seis','siete','ocho','nueve') exp=exp.lower() for i in range(len(exp)): car=exp[i] if car in digitos: exp=exp.replace(car.upper(),car) exp=exp.replace(' x ','x') exp=exp.replace(' = ','es') print(exp)</pre>		A
B	<pre>exp='DOS x DOS es cuatro' digitos=('uno','tres','cuatro','dos', 'siete','ocho','cinco','seis', 'nueve','cero') exp=exp.lower() for i in range(len(digitos)): if digitos[i] in exp: exp=exp.replace(digitos[i],str(i)) exp=exp.replace(' * ','x') exp=exp.replace('es','=') print(exp.upper())</pre>		B
C	<pre>exp='DOS x DOS es cuatro' digitos=('cero','uno','dos','tres','cuatro', 'cinco','seis','siete','ocho','nueve') exp=exp.lower() for i in range(len(digitos)): if digitos[i] in exp: exp=exp.replace(digitos[i],str(i)) exp=exp.replace(' x ','*') exp=exp.replace(' es ','=') print(exp)</pre>	X	C
D	<pre>exp='DOS x DOS es cuatro' digitos=('cero','uno','dos','tres','cuatro', 'cinco','seis','siete','ocho','nueve') for dig in digitos: if dig in exp: exp=exp.replace(dig,dig.lower()) exp=exp.replace('X','x') exp=exp.replace(' es ','=') exp=exp.lower() print(exp)</pre>		D

Ejercicio 3– Tema 4		1 Pto	
¿Cuál programa valida correctamente que se ingrese un horario (hora y minutos) de mañana correcta? Se considera mañana entre 0:1 y 12:0, ambos inclusive Debe insistir hasta conseguir un horario matutino válido			
A	<pre>pide=True while pide: try: hora=int(input('Hora (0-12): ')) min=int(input('Minutos (0-59): ')) if ((hora==0 and min in range(1,60)) or (hora==12 and min==0) or (hora in range(1,12) and min in range(0,60))): pide=False else: print('Horario inválido') except ValueError: print('Deben ser valores enteros') print(hora,':',min)</pre>	X	A
B	<pre>try: hora=int(input('Hora (0-12): ')) min=int(input('Minutos (0-59): ')) if (hora not in range(0,13) or (hora==12 and min>0)or (hora==0 and min==0)or min not in range(0,60)): print('Horario inválido') except ValueError: print('Deben ser valores enteros') print(hora,':',min)</pre>		B
C	<pre>pide=False while not pide: hora=int(input('Hora (0-12): ')) min=int(input('Minutos (0-59): ')) if ((hora==0 and min in range(1,60)) or (hora==12 and min==0) or (hora in range(1,12) and min in range(0,60))): print('Ok') else: print('Horario inválido') try: print(hora,':',min) except ValueError: print('Deben ser valores enteros')</pre>		C
D	<pre>pide=True while not pide: try: hora=int(input('Hora (0-12): ')) min=int(input('Minutos (0-59): ')) if ((hora==0 and min in range(1,60)) or (hora==12 and min==0) or (hora in range(1,12) and min in range(0,60))): pide=False else: print('Horario inválido') except ValueError: print('Deben ser valores enteros') print(hora,':',min)</pre>		D

Ejercicio 4 – Tema 4			1 Pto
En el siguiente programa: <pre>def arma(b,stk): bSemanal=[] for tipo in b: cant=b[tipo] tipo=tipo[:2] for i in range(cant): bSemanal.append(stk[tipo][i]) return bSemanal bolsonTipo= ... #seleccionar la estructura adecuada stock={'TO':['cherry','perita','redondo'], 'ZA':['zucchini','calabaza','cabutia'], 'HO':['mantecosa','espinaca','criolla']} bolson=arma(bolsonTipo,stock) print('Detalle del Bolsón de Verduras semanal') for verd in bolson: print(verd)</pre> Cuál es la definición correcta de la estructura <i>bolsonTipo</i> para que funcione adecuadamente y genere la siguiente salida: Detalle del Bolsón de Verduras semanal mantecosa espinaca zucchini calabaza cherry Nota: Se puede seleccionar una porción de una string con [:] Ej: txt='Hola a Todos' txt[3:] -> 'a a Todos' y txt[1:4] -> 'ola'			
A	<code>bolsonTipo=[2,2,1]</code>		A
B	<code>bolsonTipo={'HOJAS':2,'ZAPALLOS':2,'TOMATES':1}</code>	X	B
C	<code>bolsonTipo={2:['HOJAS','ZAPALLOS'],1:['TOMATES']}</code>		C
D	<code>bolsonTipo={'tomates':1,['cherry','perita','redondo'], 'hojas':2,['mantecosa','espinaca','criolla'], 'zapallos':2,['zucchini','calabaza','cabutia']}</code>		D

Ejercicio 5 – Tema 4			1 Pto
¿Qué muestra por pantalla el siguiente programa? <pre>colores=['rosa','azul','amarillo','verde'] tonos=['claro','oscuro','medio'] tonalidades=[] for i in range(len(colores)): tonalidades.append(colores[-(i+1)]) for i in range(len(colores)): tonalidades.append(tonos[-(i%3+1)]) print(tonalidades)</pre> Nota: Se puede seleccionar elementos de una lista de atrás para adelante con un índice negativo Ej: lis=['a','b','c'] lis[-2] -> 'b'			
A	['verde','amarillo','azul','rosa','medio','oscuro','claro','medio']	X	A
B	[['verde','medio'],['amarillo','oscuro'], ['azul','claro'], ['rosa','medio']]		B
C	['verde','claro','amarillo','oscuro','azul','medio']		C
D	['verde','medio','amarillo','oscuro','azul','claro','rosa','medio']		D

Ejercicio 6 – Tema 4			2 Ptos	
Para el siguiente DataFrame <i>mundialClubes</i>				
	nomClub	pts	partJug	octavos
0	Atlético Madrid	0	3	False
1	Bayern Munich	3	3	False
2	Benfica	0	1	False
3	Boca Juniors	2	2	True
4	Botafogo	0	2	True
5	Chelsea	3	3	False
6	Inter Milan	1	1	True
7	Inter Miami	2	2	True
8	Manchester City	3	3	True
9	Paris Saint-Germain	9	3	False
10	Real Madrid	3	3	False
11	River Plate	3	3	True
¿Qué instrucción produce la siguiente salida?				
	nomClub	pts	partJug	octavos
3	Boca Juniors	2	2	True
4	Botafogo	0	2	True
6	Inter Milan	1	1	True
7	Inter Miami	2	2	True
8	Manchester City	3	3	True
11	River Plate	3	3	True
A	print(mundialClubes[['nomClub','octavos']])			A
B	print(mundialClubes[mundialClubes['octavos']==True])			X B
C	print(mundialClubes.tail(3))			C
D	print(mundialClubes.head(6))			D

Ejercicio 7 – Tema 3		2 Ptos	
¿Qué versión de la función edita() será la adecuada para eliminar los acentos?			
<pre>def edita (...): - - - txt='condición: cúrcuma con anís es más fácil' vocCon='áéíóú' vocSin='aeiou' print(edita(txt,vocCon,vocSin))</pre>			
La salida debe ser: CONDICION: CURCUMA CON ANIS ES MAS FACIL			
Nota: El método replace() permite cambiar todas las ocurrencias de la primera substring por la segunda Ej: txt='sala de la cAsa' txt=txt.replace('a','e') txt -> 'sele de le cAse'			
A	<pre>def edita(t,sin,con): t=t.lower() for i in range(len(con)): t.replace(con[i],sin[i]) return t</pre>		A
B	<pre>def edita(t,con,sin): t=t.lower() for i in range(len(con)): t=t.replace(con[i],sin[i]) return t.upper()</pre>	X	B
C	<pre>def edita(t,con,sin): for i in range(len(con)): t.replace(con[i],sin) return t.lower()</pre>		C
D	<pre>def edita(t): t=t.lower() for i in range(len(con)): t=t.replace(con[i],sin[i])</pre>		D

Ejercicio 8– Tema 4		2 Ptos	
¿Qué salida produce el siguiente programa?			
<pre>def edita(naip): n1=naip[0][0].upper()+' de ' n2=naip[1][0].upper()+' de ' n3=naip[2][0].upper()+' de ' return [(n1,naip[0][1]),(n2,naip[1][1]),(n3,naip[2][1])] def numeros(naip): neg=('REY de ','CABALLO de ','SOTA de ') return naip[0][0] not in neg or naip[1][0] not in neg or naip[2][0] not in neg mano=[(('2','Oros'),('3','Oros'),('7','Copas')), (('2','Espadas'),('4','Oros'),('5','Copas')), (('Rey','Copas'),('4','Copas'),('6','Copas')), (('7','Espadas'),('Caballo','Espadas'),('2','Bastos'))] naipes=list(map(edita,mano)) tieneNum=list(filter(numeros,naipes)) if len(tieneNum)>0: print('Hay NÚMEROS en la mano!') for naipes in tieneNum: salida=naipes[0][0]+naipes[0][1]+' - ' salida+=naipes[1][0]+naipes[1][1]+' - ' salida+=naipes[2][0]+naipes[2][1] print(salida)</pre>			
A	Hay NÚMEROS en la mano!		A
B	Hay NÚMEROS en la mano! 2 de Oros - 3 de Oros - 7 de Copas 2 de Espadas - 4 de Oros - 5 de Copas REY de Copas - 4 de Copas - 6 de Copas 7 de Espadas - CABALLO de Espadas - 2 de Bastos	X	B
C	Hay NÚMEROS en la mano! REY de Copas - 4 de Copas - 6 de Copas 7 de Espadas - CABALLO de Espadas - 2 de Bastos		C
D	Hay NÚMEROS en la mano! 2 de Oros - 3 de Oros - 7 de Copas 2 de Espadas - 4 de Oros - 5 de Copas		D

Ejercicio 9– Tema 4			2 Ptos
¿Qué contendrá el archivo datosAlumnos.txt al final de este programa? El archivo inicialmente no existe <pre>def guarda (d,nom,dcc): a=open('datosAlumnos.txt','w') linea=str(d)+' - '+nom[d][1].upper()+' - '+nom[d][0]+'\\n' a.write(linea) a.close() a=open('datosAlumnos.txt','w') linea=str(d)+' - '+dcc[d][0].upper()+' - '+dcc[d][1]+'\\n' a.write(linea) a.close() direcciones={45667718:['Condarco 5325','gabiorg@org.com'], 42164671:['Medrano 1205 4to B','tamiorg@org.com'], 43301718:['Caracas 3532 2do A','lucasorg@org.com'], 39002166:['Malabia 1316 3ro A','anaorg@org.com']} nombres={39002166:['Ana Inés','Lempert'], 45667718:['Gabriel','Urriza'], 43301718:['Lucas Iván','Pérez Cort'], 42164671:['Tamara','Tenti']}</pre> <pre>for dni in nombres: guarda(dni,nombres, direcciones)</pre>			
A	39002166 - LEMPERT - Ana Inés 39002166 - MALABIA 1316 3RO A - anaorg@org.com 45667718 - URRIZA - Gabriel 45667718 - CONDARCO 5325 - gabiorg@org.com 43301718 - PÉREZ CORT - Lucas Iván 43301718 - CARACAS 3532 2DO A - lucasorg@org.com 42164671 - TENTI - Tamara 42164671 - MEDRANO 1205 4TO B - tamiorg@org.com		A
B	45667718 - URRIZA - Gabriel 45667718 - CONDARCO 5325 - gabiorg@org.com 42164671 - TENTI - Tamara 42164671 - MEDRANO 1205 4TO B - tamiorg@org.com		B
C	42164671 - MEDRANO 1205 4TO B - tamiorg@org.com	X	C
D	39002166 - Ana Inés - Lempert;Malabia 1316 3ro A - anaorg@org.com 43301718 - Lucas Iván - Pérez Cort - Caracas 3532 2do A - lucasorg@org.com		D