

PENSAMIENTO COMPUTACIONAL (90)

UBAXXI

TEMA 4

EXAMEN: PRIMER PARCIAL

| | |
|----------------------------------|---------------------------------------|
| 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 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 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

| | | | | | | | | | | |
|----------|---------------|---------------|---------------|---------------|---------------|----------------|----------------|----------------|----------------|----------|
| | 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 |

| | | | |
|--|--------------|----------|----------|
| 0104 – 1 Pto | | | |
| ¿Qué resultado muestra el siguiente programa? | | | |
| a=3 b=0 c=2.5 d=3.5 print(d-1*a+(b+a)*c) | | | |
| 1 | 8.0 | X | 1 |
| 2 | -18.5 | | 2 |
| 3 | 18.5 | | 3 |
| 4 | 5 | | 4 |

| 0204 – 1 Pto | | | |
|--|---|---|---|
| ¿Qué programa evalúa menos condiciones en total? | | | |
| 1 | <pre> animal='cacaatúa' clase='ave' subclase='loro' if animal in ('sapo','caballo','cacaatúa'): print('Las mascotas de mi vecino') else: if animal in ('guacamayo','pitón'): print('Papu') elif clase=='ave' or clase=='reptil': print('Leloir o Anita') </pre> | X | 1 |
| 2 | <pre> animal='cacaatúa' clase='ave' subclase='loro' if animal in ('sapo','caballo') or clase=='ave': if subclase=='loro' and animal=='guacamayo': print('Aníbal') elif subclase=='persa' or subclase=='tucán' or subclase=='loro': if animal=='gato' and clase=='otro': print('Cacho') elif animal=='gato' and clase=='felino': print('Teddy') else: print('No lo conozco') elif animal in ('guacamayo','cacaatúa','pitón'): print('Papu') else: print('No lo conozco') else: print('No lo conozco') </pre> | | 2 |
| 3 | <pre> animal='cacaatúa' clase='ave' subclase='loro' if clase in ('anfibio','mamífero','ave'): print('Aníbal') elif animal in ('guacamayo','pitón'): print('Papu') else: print('No lo conozco') </pre> | X | 3 |
| 4 | <pre> animal='cacaatúa' clase='ave' subclase='loro' if animal in ('sapo','caballo','cacaatúa'): print('Las mascotas de mi vecino') if animal in ('guacamayo','pitón'): print('Papu') elif clase=='ave' or clase=='reptil': print('Leloir o Anita') </pre> | | 4 |

Por error de enunciado, las respuestas 1 y 3 son correctas.

| 0304 – 1 Pto | | | |
|---|---|---|---|
| <p>¿Qué muestra por pantalla el siguiente programa?</p> <pre> palabras=['y', 'tomar', 'pastos'] silabas=['nada', 'una sílaba', 'una sílaba', 'una o dos sílabas', 'dos o tres sílabas', 'dos, tres o cuatro sílabas', 'uff!', 'uff!'] for pal in palabras: print(pal, silabas[len(pal)]) </pre> | | | |
| 1 | uff! pocas ninguna | | 1 |
| 2 | Y TOMAR PASTOS | | 2 |
| 3 | y una sílaba tomar dos, tres o cuatro sílabas pastos uff! | X | 3 |
| 4 | 6 uff! 5 dos, tres o cuatro sílabas 1 una sílaba | | 4 |

| 0404 – 1 Pto | | | |
|---|--------------------------|---|---|
| <p>¿Qué muestra por pantalla el siguiente programa?</p> <pre> nom1='Juan' nom2='Elena' nom3='fausto' nom3=nom3.upper() texto=nom2+' '+nom1[2:]+ 'y '+nom3[0] print(texto) </pre> <p>Notas: Recordá que los caracteres de las string se numeran de 0 en adelante Se puede seccionar una string con [:] Ejs `marca'[2:] -> `rca` - `marca`[:3] -> `mar` y `marca`[1:4] -> `arc`</p> <p>Se puede invertir una string con el siguiente tip a[::-1] Ej `hola`[::-1] -> `aloh`</p> <p>Se puede referenciar un elemento (carácter) con un índice positivo o negativo. Positivo toma el n elemento desde la izquierda; negativo toma el n elemento desde la derecha (antes del final) Ejs `notas`[1] -> `o` y `notas`[-1] -> `s` o `notas`[-3] -> `t`</p> | | | |
| 1 | Elena, any F | X | 1 |
| 2 | Elena, Juan Fausto | | 2 |
| 3 | ELENA, JUANyFAUSTO | | 3 |
| 4 | E, ju y otsuaf | | 4 |

| 0504 – 1 Pto | | | |
|---|-------------------------|---|---|
| <p>¿Qué muestra por pantalla el siguiente programa?</p> <pre>pal='COLOR' for i in range(3): if i%2!=0: print(pal) else: print(pal[0]+' '*3+pal[-1])</pre> <p>Notas: El operador * repite n veces una string Ej 'una'*2 -> 'unauna'</p> <p>Se puede referenciar un elemento (carácter) con un índice positivo o negativo. Positivo toma el n elemento desde la izquierda; negativo toma el n elemento desde la derecha (antes del final) Ejs 'notas'[1] -> 'o' y 'notas'[-1] ->'s' o 'notas'[-3]->'t'</p> | | | |
| 1 | C R COLOR C R | X | 1 |
| 2 | COLOR COLOR COLOR | | 2 |
| 3 | ROLOC ROLOC ROLOC | | 3 |
| 4 | COLOR C R COLOR | | 4 |

| 0604 – 2 Ptos | | | |
|---|-----------------|---|---|
| <p>¿Qué muestra el siguiente programa?</p> <pre>a=[4,3,1,2] b=[3,1,2,1,2] c=[] for num in b: c.append(a[len(a)-num]) print(c)</pre> | | | |
| 1 | [4, 3, 1, 2] | | 1 |
| 2 | [3, 2, 1, 2, 1] | X | 2 |
| 3 | [1, 1] | | 3 |
| 4 | [0] | | 4 |

| 0704 – 2 Ptos | | | |
|---|--|---|---|
| <p>¿Qué programa ejecuta más veces el cuerpo del bucle?</p> | | | |
| 1 | for i in range(1,len('mamarracho'),-4): j=0 | | 1 |
| 2 | for car in 'mar': i=0 | | 2 |
| 3 | i=10 j=1 while i>j: i-=1 | X | 3 |
| 4 | for i in range(5,50,10): j=1 | | 4 |

| | | | |
|---|---|----------|----------|
| 0804 – 2 Ptos | | | |
| ¿Qué función es adecuada para el siguiente programa? <pre>def calcula(...): - - - term1=[5,0,3] term2=[3,6,2] for i in range(len(term2)): print(calcula(term2[i],term1[i]))</pre> Debe mostrar la siguiente salida: 8 6 5 | | | |
| 1 | <pre>def calcula(term2[i]): resultado=n1+n2 return term2</pre> | | 1 |
| 2 | <pre>def calcula(): return 1*2</pre> | | 2 |
| 3 | <pre>def calcula(n1,n2): resultado=n1+n2 return resultado</pre> | X | 3 |
| 4 | <pre>def calcula(n1,n2): resultado=n1*8 return 0</pre> | | 4 |

| | | | |
|--|--|----------|----------|
| 0904 – 2 Ptos | | | |
| ¿Qué muestra por pantalla el siguiente programa? <pre>for i in range(3): for j in (2,1): print('a'*j)</pre> Nota: El operador * aplicado a una string repite n veces la misma Ej: a='mi'*2 -> a='mimi' | | | |
| 1 | aaaaaaaaa | | 1 |
| 2 | aa aaa aaaa aa aaa aaaa | | 2 |
| 3 | a a a a | | 3 |
| 4 | aa a aa a aa a | X | 4 |

