

PENSAMIENTO COMPUTACIONAL (90)

UBA XXI

TEMA 6

|                                  |                                 |
|----------------------------------|---------------------------------|
| EXAMEN: SEGUNDO PARCIAL          |                                 |
| APELLIDO:                        | CALIFICACIÓN:                   |
| NOMBRE:                          |                                 |
| DNI (registrado en SIU Guaraní): |                                 |
| E-MAIL:                          | DOCENTE<br>(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<br>1 Pto | Ej 2<br>1 Pto | Ej 3<br>1 Pto | Ej 4<br>1 Pto | Ej 5<br>1 Pto | Ej 6<br>1 Pto | Ej 7<br>1 Pto | Ej 8<br>2 Ptos | Ej 9<br>2 Ptos | Ej 10<br>2 Ptos |   |
|---|---------------|---------------|---------------|---------------|---------------|---------------|---------------|----------------|----------------|-----------------|---|
| 1 |               |               |               |               |               |               |               |                |                |                 | 1 |
| 2 |               |               |               |               |               |               |               |                |                |                 | 2 |
| 3 |               |               |               |               |               |               |               |                |                |                 | 3 |
| 4 |               |               |               |               |               |               |               |                |                |                 | 4 |

**iATENCIÓN!** 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.

|   |   |   |   |
|---|---|---|---|
| <b>0106 – 1 Pto</b>   |   |   |   |
| <p>¿Cuál de los siguientes códigos valida adecuadamente que ingrese un nombre finalizado con vocal? Debe detectar el error y garantizar un dato válido</p> <p><b>Ejs válidos: elena, Pedro, SANDRA</b></p> <p><b>Ejs inválidos: javier, Inés, 9ema</b></p> <p><b>Nota:</b></p> <p>Un índice negativo (-i) aplicado a una secuencia hace referencia al elemento de la posición len(secuencia)-i</p> <p><b>Ej:</b></p> <p><b>a=[1,23,56,7]</b></p> <p><b>a[-2] -&gt; 56</b></p> |   |   |   |
| 1   | <pre>def vocFin(n):<br/>    voc='aeiouáéíóú'<br/>    return n[-1] not in voc<br/><br/>#PPal<br/>nom=input('Nombre finalizado en vocal: ')<br/>while not vocFin(nom) or not nom.isalpha():<br/>    nom=input('Nombre finalizado en vocal: ')</pre>   |   | 1 |
| 2   | <pre>def vocFin(n):<br/>    voc='aeiouáéíóú'<br/>    return n[len(n)-1] in voc<br/><br/>#PPal<br/>nom=input('Nombre finalizado en vocal: ')<br/>if vocFin(nom):<br/>    if not nom.isalpha():<br/>        print('Ok')<br/>    else:<br/>        print('Ingreso Inválido, hasta pronto!')<br/>else:<br/>    print('Ingreso Inválido, hasta pronto!')</pre> |   | 2 |
| 3   | <pre>def vocFin(n):<br/>    voc='aeiouáéíóú'<br/>    return n[-1].lower() in voc<br/><br/>#PPal<br/>nom=input('Nombre finalizado en vocal: ')<br/>while not vocFin(nom) or not nom.isalpha():<br/>    nom=input('Nombre finalizado en vocal: ')</pre>   | X | 3 |
| 4   | <pre>def vocFin(n):<br/>    voc='aeiouáéíóú'<br/>    return n[-1].upper() in voc<br/><br/>#PPal<br/>nom=input('Nombre finalizado en vocal: ')<br/>if vocFin(nom) and nom.isalpha():<br/>    print('Ok')<br/>else:<br/>    print('Ingreso Inválido, hasta pronto!')</pre>  |   | 4 |

|   |   |   |   |
|---|---|---|---|
| <b>0206 – 1 Pto</b>   |   |   |   |
| ¿Cuál es la salida correcta del siguiente programa?   |   |   |   |
| <pre>def nomDia(d):     dias={1:'domingo',2:'lunes',3:'martes',4:'miércoles',           5:'jueves',6:'viernes',7:'sábado'}     return dias[d]  #PPal dias=[1,5,2,6,7] nombres=list(map(nomDia,dias)) print(nombres)</pre> |   |   |   |
| 1   | [1, 5, 2, 6, 7] [3, 4]                              |   | 1 |
| 2   | ['domingo', 'jueves', 'lunes', 'viernes', 'sábado'] | X | 2 |
| 3   | 'SABADO'  |   | 3 |
| 4   | ['SAB', 'VIE', 'JUE', 'MIE', 'MAR', 'LUN', 'DOM']   |   | 4 |

|   |               |   |   |
|---|---------------|---|---|
| <b>0306 – 1 Pto</b>   |               |   |   |
| ¿Cuál es la salida del siguiente programa?  |               |   |   |
| <pre>def mes30(m):     de30dias=(4,6,9,11)     return m in de30dias  #PPal meses=[1,5,6,2,7] de30=list(filter(mes30,meses)) print(de30)</pre> |               |   |   |
| 1   | [4, 6, 9, 11] |   | 1 |
| 2   | 'noviembre'   |   | 2 |
| 3   | [6]           | X | 3 |
| 4   | [1, 7]        |   | 4 |

0406 – 1 Pto

¿Qué contenido tendrá el archivo **destacados.txt** al finalizar la ejecución del programa si el archivo **deportistas.txt** tiene el siguiente contenido?

Contenido de **deportistas.txt**:

```
lionel,messi,1
lucha,aymar,3
julián,álvarez,1
manu,ginobili,5
```

Programa a ejecutarse:

```
deportes={1:'Futbol',2:'Patín',3:'Hokey',
           4:'Tenis',5:'Basquet',6:'Ciclismo'}
arch=open('deportistas.txt','r')
lista=arch.readlines()
arch.close()
nom=[]
for dep in lista:
    d=dep.split(',')
    nom.append(d[0].upper()+' es de '+deportes[int(d[2])]+\n')
arch=open('destacados.txt','w')
arch.writelines(nom)
arch.close()
```

**Nota:**  
El método **split()** devuelve una lista con las partes de un texto tomando como separador el argumento

**Ej:**  
**'yo soy argentina'.split(' ') -> ['yo', 'soy', 'argentina']**

|   |  |   |   |
|---|--|---|---|
| 1 | ginobili es de BASQUET   |   | 1 |
| 2 | messi lionel,FUTBOL<br>aymar lucha,HOKEY<br>álvarez julián,FUTBOL<br>ginobili manu,BASQUET |   | 2 |
| 3 | lionel,messi,1<br>lucha,aymar,3<br>julián,álvarez,1<br>manu,ginobili,5                     |   | 3 |
| 4 | LIONEL es de Futbol<br>LUCHAQ es de Hokey<br>JULIÁN es de Futbol<br>MANU es de Basquet     | X | 4 |

0506 – 1 Pto

¿Qué debería ser **estructura** para que la siguiente instrucción se ejecute sin problemas?

```
estructura.sort()
```

**Nota;**  
El método **sort()** ordena los elementos

**Ej:**  
**a=[2,1,6]**  
**a.sort -> a queda [1,2,6]**

|   |                |   |   |
|---|----------------|---|---|
| 1 | Un diccionario |   | 1 |
| 2 | Una tupla      |   | 2 |
| 3 | Una lista      | X | 3 |

|   |            |  |   |
|---|------------|--|---|
| 4 | Una string |  | 4 |
|---|------------|--|---|

|   |   |   |   |
|---|---|---|---|
|   | <b>D606 – 1 Pto</b>   |   |   |
|   | <p>Para averiguar la posición o número de orden de un alumno en una lista de nombres. ¿Cuál función impide que aborte la ejecución del mismo si se ingresa el nombre <b>Sanz, Ignacio</b> ?</p> <pre>def donde (...):<br/>    -<br/>    -<br/>    -<br/><br/>alumnos=['Andreoli, Julieta','Vargas, Ulises','Fiquet, Paulo']<br/>alum=input('Nombre: ')<br/>pos=donde(alumnos,alum)<br/>print(alum,pos)</pre> <p><b>Nota:</b><br/>El método <b>index()</b> devuelve la posición de un elemento en una lista. Si el elemento no está, se aborta la ejecución del programa</p> <p><b>Ej:</b><br/><b>[1,2,3].index(3) -&gt; 2</b></p> |   |   |
| 1 | <pre>def donde(alumnos,alum):<br/>    indice=alumnos.index(alum)<br/>    cartel='está en la posición '+str(indice+1)<br/>    return cartel</pre>  |   | 1 |
| 2 | <pre>def donde(alumnos,alum):<br/>    valido=False<br/>    while not valido:<br/>        indice=alumnos.index(alum)<br/>        if indice in range(len(alumnos)):<br/>            cartel='está en la posición '+str(indice+1)<br/>            valido=True<br/>    return cartel</pre>   |   | 2 |
| 3 | <pre>def donde(alumnos):<br/>    if:<br/>        indice=alumnos.index(alum)<br/>        cartel='está en la posición '+str(indice+1)<br/>    elif:<br/>        cartel='NO ESTÁ EN LA LISTA'<br/>    return cartel</pre>  |   | 3 |
| 4 | <pre>def donde(alumnos,alum):<br/>    try:<br/>        indice=alumnos.index(alum)<br/>        cartel='está en la posición '+str(indice+1)<br/>    except:<br/>        cartel='NO ESTÁ EN LA LISTA'<br/>    return cartel</pre>  | X | 4 |

|  |      |         |      |                 |             |   |                 |
|--|------|---------|------|-----------------|-------------|---|-----------------|
| 0706 – 1 Pto   |      |         |      |                 |             |   |                 |
| Para el DataFrame <b>vue</b> de pandas, que contiene:  |      |         |      |                 |             |   |                 |
|  | fila | asiento | zona | pax             |             |   |                 |
| 0  | 5    | A       | 4    | tripulación     |             |   |                 |
| 1  | 16   | A       | 2    | None            |             |   |                 |
| 2  | 16   | B       | 2    | Marcelo Uriondo |             |   |                 |
| 3  | 16   | C       | 2    | Delsy Anchorena |             |   |                 |
| 4  | 23   | D       | 1    | None            |             |   |                 |
| 5  | 23   | B       | 1    | Juana Nazar     |             |   |                 |
| <b>Nota:</b>   |      |         |      |                 |             |   |                 |
| <b>None</b> es la constante nula. Si aparece <b>None</b> en algún campo de un DataFrame es porque ese campo está vacío, no tiene contenido (equivale a <b>NaN</b> ). |      |         |      |                 |             |   |                 |
| ¿Qué contendrá <b>vue</b> después de la siguiente operación?   |      |         |      |                 |             |   |                 |
| vue.loc[1, 'pax']='disponible'   |      |         |      |                 |             |   |                 |
| 1  | fila | asiento | zona | pax             | X           | 1 |                 |
|  | 0    | 5       | A    | 4               |             |   | tripulación     |
|  | 1    | 16      | A    | 2               |             |   | disponible      |
|  | 2    | 16      | B    | 2               |             |   | Marcelo Uriondo |
|  | 3    | 16      | C    | 2               |             |   | Delsy Anchorena |
|  | 4    | 23      | D    | 1               |             |   | None            |
| 5  | 23   | B       | 1    | Juana Nazar     |             |   |                 |
| 2  | fila | asiento | zona | pax             |             | 2 |                 |
|  | 0    | 5       | A    | 4               | tripulación |   |                 |
|  | 1    | 16      | A    | 2               | None        |   |                 |
| 3  | fila | asiento | zona | pax             |             | 3 |                 |
|  | 0    | 5       | A    | 4               | tripulación |   |                 |
| 4  | zona | fila    |      |                 |             | 4 |                 |
|  | 1    | 2       | 16   |                 |             |   |                 |
|  | 2    | 2       | 16   |                 |             |   |                 |
|  | 3    | 2       | 16   |                 |             |   |                 |

|   |                     |      |   |
|---|---------------------|------|---|
| 0806 – 2 Ptos   |                     |      |   |
| ¿Cuáles modos de apertura deben emplearse con los archivos <b>datos.txt</b> y <b>otro.txt</b> en el siguiente programa para que no salte error?   |                     |      |   |
| <pre>arch=open('datos.txt',...) filas=arch.readlines() for fil in filas:     print(fil) arch.close()  arch=open('otro.txt',...) linea=arch.readline() print(linea) arch.write('Fin\n') arch.close()</pre> |                     |      |   |
| 1   | open() de datos.txt | 'w+' | 1 |
|   | open() de otro.txt  | 'a'  |   |
| 2   | open() de datos.txt | 'w'  | 2 |
|   | open() de otro.txt  | 'w'  |   |
| 3   | open() de datos.txt | 'a'  | 3 |
|   | open() de otro.txt  | 'r'  |   |
| 4   | open() de datos.txt | 'r'  | X |
|   | open() de otro.txt  | 'r+' |   |

0906 – 2 Ptos

Para el DataFrame **vue** de pandas, que contiene:

|   | fila | asiento | zona | pax             |
|---|------|---------|------|-----------------|
| 0 | 5    | A       | 4    | tripulación     |
| 1 | 16   | A       | 2    | None            |
| 2 | 16   | B       | 2    | Marcelo Uriondo |
| 3 | 16   | C       | 2    | Delsy Anchorena |
| 4 | 23   | D       | 1    | None            |
| 5 | 23   | B       | 1    | Juana Nazar     |

Nota:

None

es la constante nula. Si aparece **None** en algún campo de un DataFrame es porque ese campo está vacío, no tiene contenido (equivale a **NaN**).

¿Qué operación produce el siguiente resultado?

| fila |   |
|------|---|
| 5    | A |
| 16   | C |
| 23   | D |

|   |                                      |   |   |
|---|--------------------------------------|---|---|
| 1 | vue[['pax', 'zona']]                 |   | 1 |
| 2 | vue[vue['pax'].isnull()]             |   | 2 |
| 3 | vue.head(4)                          |   | 3 |
| 4 | vue.groupby('fila')['asiento'].max() | X | 4 |

|  |  |   |   |
|--|--|---|---|
| 1006 – 2 Ptos  |  |   |   |
| En el siguiente programa:  |  |   |   |
| <pre>multas={'AC 335 FQ':[], 'AD 358 DD':[15644,33207],         'AB 124 EG':[], 'AB 358 FH':[89000]} pat=input('Ingrese patente: ').upper() if pat in multas:     ..... # línea a completar  print('El dominio',pat,'tiene \$',total,       'en concepto de deuda por infracciones')</pre> |  |   |   |
| Que calcula el importe total de multas adeudadas de una patente. ¿Cuál debería ser la línea faltante en el código?   |  |   |   |
| La salida final, para un ingreso <b>ad 358 dd</b> debería ser:   |  |   |   |
| El dominio AD 358 DD tiene \$ 48851 en concepto de deuda por infracciones  |  |   |   |
| 1  | <code>total=sum(multas[pat])</code>    | X | 1 |
| 2  | <code>total=multas[1]+multas[2]</code> |   | 2 |
| 3  | <code>sum(multas[pat][0])</code>       |   | 3 |
| 4  | <code>total=multas[pat][0]</code>      |   | 4 |



**Talón de Control para el Alumno**

|          | Ej 1<br>1 Pto | Ej 2<br>1 Pto | Ej 3<br>1 Pto | Ej 4<br>1 Pto | Ej 5<br>1 Pto | Ej 6<br>1 Pto | Ej 7<br>1 Pto | Ej 8<br>2 Ptos | Ej 9<br>2 Ptos | Ej 10<br>2 Ptos |          |
|----------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|----------------|----------------|-----------------|----------|
| <b>1</b> |               |               |               |               |               |               |               |                |                |                 | <b>1</b> |
| <b>2</b> |               |               |               |               |               |               |               |                |                |                 | <b>2</b> |
| <b>3</b> |               |               |               |               |               |               |               |                |                |                 | <b>3</b> |
| <b>4</b> |               |               |               |               |               |               |               |                |                |                 | <b>4</b> |