

**ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**  
**FUNDAMENTOS DE PROGRAMACIÓN**

**2020 – 2S**

**TAREA 2**

1. Escriba una función que reciba un número n como parámetro y que muestre por consola una pirámide de n pisos.

Pista: La expresión "A" \* 5 resulta en "AAAAA"

Ejemplos de salida:

N = 2	N = 5
<pre>&gt;&gt;&gt;dibujarPiramide(2) * **</pre>	<pre>&gt;&gt;&gt;dibujarPiramide(5) * ** *** **** *****</pre>

2. Un cono sigue el siguiente modelo  $z(x,y) = \sqrt{x^2 + y^2}$ , escriba una función en Python que retorne el valor de z para cualquier x o y.

Pista: La expresión math.sqrt(4) retornaría 2. Recuerde importar "math".

Ejemplos de salida:

x = 2.5 ; y = 4.6	x = 7.55 ; y = 2.66
<pre>&gt;&gt;&gt; print(z(2.5,4.6)) 5.235456045083369</pre>	<pre>&gt;&gt;&gt; print(z(7.55,2.66)) 8.0048797617453</pre>

3. Escriba una función que reciba una listaA con elementos repetidos, que retorne una nueva lista con elementos únicos.

Pista: La expresión “a” in [“a”, “b”] retornaría True.

Ejemplos de salida:

<code>lista = ['a', 'b', 'c', 'd', 'a', 'c', 'g', 'z', 'a']</code>
<pre>&gt;&gt;&gt;print(elementosUnicos(['a', 'b', 'c', 'd', 'a', 'c', 'g', 'z', 'a'])) ['a', 'b', 'c', 'd', 'g', 'z']</pre>

4. Escriba una función que reciba una listaA con elementos repetidos y una listaB con los respectivos elementos únicos. Retorne una nueva lista paralela a listaB con la respectiva frecuencia de cada elemento.

Pista: La expresión ['a','b','a','a'].count('a') retorna 3

Ejemplos de salida:

<code>listaA = ['a', 'b', 'c', 'd', 'a', 'c', 'g', 'z', 'a']</code>
<code>listaB = ['a', 'b', 'c', 'd', 'g', 'z']</code>
<pre>&gt;&gt;&gt;print(contarElementos(listaA, listaB)) [3, 1, 2, 1, 1, 1]</pre>

5. Escriba una función que reciba 2 listas y qué retorne la cantidad de elementos en común. Tenga en cuenta que los elementos en ambas listas pueden estar repetidos.

Pista: Puede combinar los elementos de las 2 listas, iterarlo y luego identificar si el elemento iterado pertenece a la combinación.

Ejemplos de salida:

<code>listaA = ['g', 't', 'c', 's', 'd', 'a', 't', 'y', 'a']</code>
<code>listaB = ['r', 'b', 'e', 'r', 'd', 's', 'd', 'l', 'd', 'p']</code>
<pre>&gt;&gt;&gt;print(elementosEnComun(listaA, listaB)) 2 &gt;&gt;&gt;print(elementosEnComun(listaB, listaA)) 2</pre>

6. Escriba una función en Python que reciba un arreglo de números y que muestre por pantalla cada elemento en orden ascendente. Para este método no use listas ni sus funciones.

Pista: La función `np.array([3,1,2]).argsort()` retorna `[1, 2, 0]`

Ejemplos de salida:

```
arr = np.array([12, 45, 58, 86, 0, 84, 48, 15, 3, 3])
```

```
>>> imprimirEnOrden(arr)
0
3
3
12
.
.
.
```

7. Se tienen 2 arreglos paralelos. Escriba una función que recibe un arregloA que representa las edades de un grupo de persona y un arregloB de float que representan el aporte que han realizado al estado en el último año. Retorne el valor promedio de las aportaciones anuales de las personas entre 18 y 40 años. No itere los arreglos. Use funciones de numpy.

Pista: La expresión `vec = np.array([1,2,4]); vec[vec >= 2]`, retorna `array([2, 4])`

Ejemplos de salida:

```
arregloA = np.array([67, 27, 73, 60, 78, 60, 66, 45, 21, 23])
```

```
arregloB = np.array([43.0, 11.1, 21.6, 35.8, 11.8, 14.7, 39.8, 35.4, 46.0, 47.7])
```

```
>>> print(obtenerPromedio(arr1,arr2))
34.933333333333334
```

8. Escriba una función que reciba un arreglo que representan calificaciones de estudiantes. Estas calificaciones están sobre 10, usted debe generar un nuevo arreglo para ponderar la nota sobre 8. A partir del nuevo arreglo muestre por pantalla los aprobados. Los aprobados con aquellos con una nota mayor o igual a 5.75. No itere el arreglo. Use funciones de numpy.

Pista: La expresión `np.where(np.array([2,4,1]) == 4, "A", "B" )`, retorna `array(['B', 'A', 'B'], dtype='<U1')`

Ejemplos de salida:

```
arregloA = np.array([8.75, 8.46, 4.98, 5.21, 5.53, 4.52, 5.35, 6.86, 6.15, 8.76])
```

```
>>> mostrarAprobados(arregloA)
['AP' 'AP' 'RP' 'RP' 'RP' 'RP' 'RP' 'AP' 'AP' 'AP']
```

## Tabla de conocimiento

Tabla que relaciona cada ejercicio con los temas de la materia que debe de conocer para llegar a una posible solución. Recuerde que puede existir más de una solución que aplique una X más o menos que la propuesta.

[illegible]