

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN

CCPG1001 - FUNDAMENTOS DE PROGRAMACIÓN

PRIMERA EVALUACIÓN - I TÉRMINO 2018-2019/ Junio 29, 2018

Nombre: \_\_\_\_\_ Matrícula: \_\_\_\_\_ Paralelo: \_\_\_\_\_

COMPROMISO DE HONOR: Al firmar este compromiso, reconozco que el presente examen está diseñado para ser resuelto de manera individual, que puedo usar un lápiz o esferográfico; que sólo puedo comunicarme con la persona responsable de la recepción del examen; y, cualquier instrumento de comunicación que hubiere traído, debo apagarlo y depositarlo en la parte anterior del aula, junto con algún otro material que se encuentre acompañándolo. Además no debo usar calculadora alguna, consultar libros, notas, ni apuntes adicionales a los que se entreguen en esta evaluación. Los temas debo desarrollarlos de manera ordenada.

Firmo el presente compromiso, como constancia de haber leído y aceptado la declaración anterior. "Como estudiante de ESPOL me comprometo a combatir la mediocridad y actuar con honestidad, por eso no copio ni dejo copiar".

\_\_\_\_\_  
Firma

**TEMA 1 (40 PUNTOS)**

La compañía ACME S.A. está desarrollando un nuevo método para detectar especies en base a su ADN. Para representar una especie por su ADN se utiliza una secuencia **S** compuesta únicamente de las letras A, C, G y T.

Se tienen como datos:

Una lista **L** de secuencias **S** y una cadena de referencia **R** que identifica de forma única a la especie buscada. **R no tiene letras repetidas.**

Implemente un programa en Python que muestre todas las secuencias **S** que *pertenecen a la especie buscada* y los índices en la inversa de **S (INV)** donde aparece la cadena de referencia **R**.

Para realizar esta tarea, por cada secuencia **S** en **L**:

1. Forme la cadena inversa (**INV**) de la secuencia **S**.
2. Si la cadena **R** aparece exactamente dos veces en la segunda mitad de **INV** y al menos 3 veces en total, la secuencia **S pertenece a la especie buscada**.
3. Si **S pertenece a la especie buscada**, muestre la secuencia **S** y los índices.

Ejemplo:

**L** = [ 'ATTGCTTGCTATTTAAACCGGTTATGCATAGCGC', 'ATTAGCCGCTATCGA', ... ]

**R** = 'CG'

Salida:

Secuencia: ATTGCTTGCTATTTAAACCGGTTATGCATAGCGC

Índices: [0, 2, 25, 29]

Secuencia: ...

Índices: ...

NO COPIAR

## TEMA 2 (50 PUNTOS)

A Ud. le dan una matriz **M** con las estadísticas de las buenas características de los jugadores del Mundial, por equipos:

	Portugal		Brasil		...		Argentina	
	Cristiano Ronaldo		Neymar Jr.		...		Lionel Messi	
Goles anotados	32	...	24	...	...	...	45	...
...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...
% posesión del balón	90	...	84	...	...	...	91	...
% de pases acertados	54	...	60	...	...	...	78	...
Tiros directos al arco	187	...	239	...	...	...	112	...

También posee una lista con los nombres de los países

```
países = ['Portugal', 'Brasil', ... , 'Argentina']
```

Y listas con los nombres de los jugadores por país:

```
prtgl = ['Cristiano Ronaldo', ...]
```

```
brsl = ['Neymar Jr.', ... ]
```

```
...
```

```
argntn = ['Lionel Messi', ...]
```

Además, tiene la lista de jugadores de cada país en otra lista (lista de listas):

```
jugadores_países = [prtgl, brsl, ..., argntn] #lista de listas
```

Además, tenemos la métrica:

$$\text{Efectividad} = \text{Goles anotados} / \text{Tiros directo al arco}$$

Entonces escriba sentencias de Python para:

1. Determinar el país con el número de goles promedio más alto. El promedio de goles se calcula como **goles anotados del país / número de jugadores del país**.
2. Contar cuántos jugadores alemanes tienen una efectividad mayor que la efectividad promedio de Alemania.
3. Mostrar la lista con los nombres de los jugadores que tienen más del 91% de **pases acertados**.
4. Mostrar el nombre del jugador con mayor posesión del balón, y su país.
5. Calcular el **promedio mundial** por cada una de las características ( "Goles anotados", "...", "% posesión del balón", "% de pases acertados", "Tiros directos al arco" ). Una vez obtenido el promedio mundial, determine si cada uno de los valores de las características de Cristiano Ronaldo están por encima del correspondiente promedio mundial. Cristiano Ronaldo es el primer jugador de la matriz. Luego de verificar todas las características del jugador, muestre el mensaje:

**"Cristiano Ronaldo está/no está por encima del promedio mundial"**

dependiendo si todas sus características superan el promedio mundial o no.

### TEMA 3 (10 PUNTOS)

¿Qué imprime el siguiente código? Justifique su respuesta

```
import numpy as np

arr = np.array([8,3,7,1,5,2,6,4])
d = "vwxyz"
p = ''
for a in range(arr[arr < 5].size):
    p += d[a] * a
print(p)
```

---//---

#### Cheat Sheet. Funciones y propiedades de referencia en Python.

Librería Numpy para <b>arreglos</b> :	para <b>listas</b> :	para <b>cadena</b> s:
<code>np.array((numRows,numCols),dtype= )</code> <code><b>arreglos</b>.shape</code> <code><b>arreglos</b>.reshape()</code> <code>numpy.sum(<b>arreglos</b>)</code> <code>numpy.mean(<b>arreglos</b>)</code> <code><b>arreglos</b>.sum(axis=1)</code>	<code><b>listas</b>.append(...)</code> <code><b>listas</b>.extend(...)</code> <code><b>listas</b>.count(...)</code> <code><b>listas</b>.index(...)</code> <code><b>listas</b>.pop()</code> <code><i>elemento</i> in <b>listas</b></code>	<code><b>cadena</b>s.islower()</code> <code><b>cadena</b>s.isupper()</code> <code><b>cadena</b>s.lower()</code> <code><b>cadena</b>s.upper()</code> <code><b>cadena</b>s.split(...)</code> <code><b>cadena</b>s.find(...)</code> <code><b>cadena</b>s.count(...)</code>