

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN  
CCPG1001 - FUNDAMENTOS DE PROGRAMACIÓN  
TERCERA EVALUACIÓN - II TÉRMINO 2018-2019/ Febrero 15, 2019

**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 (30 PUNTOS)**

Asuma que tiene una lista **T** de términos. Un término puede ser una palabra o uno de los siguientes tres símbolos: punto (.), coma (,) y guión (-). Desarrolle un programa que forme un texto usando las siguientes reglas:

- El texto debe estar compuesto de 73 términos seleccionados aleatoriamente
- El primer término debe ser una palabra
- No se puede seleccionar dos símbolos de manera consecutiva. Si eso pasa, seleccione un nuevo término aleatoriamente hasta que sea una palabra
- Dos palabras seguidas deben estar separadas por un espacio. Ejemplo: palabra1 palabra2
- La coma debe estar pegada a la palabra a su izquierda y separada por un espacio de la palabra a su derecha. Ejemplo: palabra1, palabra2
- El guión debe estar pegado a sus dos palabras. Ejemplo: palabra1-palabra2
- El punto debe estar pegado a la palabra de la izquierda y seguido de un salto de línea. Ejemplo: palabra1.
- No elimine términos de la lista T.

Escriba este texto resultante en un archivo de nombre literatura.txt.

## TEMA 2 (60 PUNTOS)

Asuma que se tiene un archivo por cada partido jugado en el Sudamericano Sub-20. Cada archivo tiene información con datos de los jugadores que participaron en el partido con el siguiente formato:

Pais,Jugador,Tarjetas\_Amarillas,Tarjetas\_Rojas,Goles,Minutos,KM\_recorridos

Ejemplo:

```
Pais,Jugador,Tarjetas_Amarillas,Tarjetas_Rojas,Goles,Minutos,KM_recorridos
...
Ecuador, José Cifuentes,1,0,0,75,6.3
Uruguay, Sebastián Cáceres,2,1,0,90,7
Ecuador, Leonardo Campana,0,0,1,87,10
...
```

Implemente las siguientes funciones:

1. **[15 puntos] actualizaDiccionario(nomArchivo, dic)** que recibe el nombre de un archivo con datos del partido y actualiza el diccionario de **totales por jugador** que tiene el siguiente formato:

```
dic = {'Ecuador':{'Leonardo Campana':{'TA':3, 'TR':0, 'Goles':6, 'Minutos':800, 'KM':75},
...
},
'Argentina':{...},
...
}
```

2. **[9 puntos] buenDeportista(jugador, dic)** que recibe el nombre de un jugador y el diccionario de totales. La función retorna True o False dependiendo si ese jugador puede ser catalogado como un **"buen deportista"**. Un jugador se considera un **"buen deportista"** si ha recibido menos de 2 tarjetas por **cada** 270 minutos de juego.
3. **[9 puntos] jugadorAtleta(jugador, dic)** que recibe el nombre de un jugador y el diccionario de totales. Si el jugador ha corrido como mínimo el promedio de lo que han corrido los jugadores de su país y ha anotado al menos un gol, retorna **True**. En caso contrario, retorna **False**.
4. **[9 puntos] paisBuenasPracticas(pais, dic)** que recibe el nombre de un país y el diccionario de totales. La función retorna True o False dependiendo si ese país puede ser nominado para el **"Best Practices award"**. Un país puede ser nominado a este premio si **TODOS** los jugadores del país pueden ser catalogados como **"buen deportista"**.

Escriba un programa que:

5. **[2 puntos]** Forme el **diccionario de totales** a partir de una lista con los nombres de los archivos de datos de los partidos. Asuma que tiene la lista  $L = ['br-ur.csv', \dots, 'ec-vn.csv']$  para esta tarea.
6. Muestre las siguientes estadísticas **por país**:
  - a. **[5 puntos]** Porcentaje de jugadores atletas, es decir el número de jugadores atletas dividido para el total de jugadores del país.
  - b. **[5 puntos]** Goles por Km recorrido, es decir el número de goles del país dividido para el total de Km recorridos por todos sus jugadores
7. **[3 puntos]** Muestre los países nominados para el **"Best Practices award"**.
8. **[3 puntos]** Muestre la nómina de jugadores atletas con su respectivo país.

### TEMA 3 (10 PUNTOS)

¿Qué imprime el siguiente código? Justifique su respuesta

letras

E	B	A	O
G	P	L	A
I	J	K	S
R	V	U	S

numeros

93	58	63	21
31	60	44	72
29	51	50	32
85	11	15	67

```
import numpy as np
V1 = numeros[numeros >= 60]
V2 = letras[numeros >= 60]
V2 = V2[np.argsort(V1)]
for letra in V2:
    print(letra, end='')
print('!!')
```

**Asuma que este tema NO tiene errores de compilación. Si usted cree que hay algún error de compilación, consúltelo inmediatamente con su profesor.**

---//---

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

Librería Numpy para <b>arreglos</b> :	para <b>listas</b> :	para <b>cadena</b> s:
np.array([elementos],dtype= ) np.unique( <b>arreglo</b> ) np.sum( <b>arreglo</b> ) np.mean( <b>arreglo</b> ) <b>arreglo</b> .shape <b>arreglo</b> .size <b>arreglo</b> .sum()	<b>listas</b> .append(...) <b>listas</b> .extend(...) <b>listas</b> .count(...) <b>listas</b> .index(...) <b>listas</b> .pop() elemento in <b>listas</b>	<b>cadena</b> s.islower() <b>cadena</b> s.isupper() <b>cadena</b> s.lower() <b>cadena</b> s.upper() <b>cadena</b> s.split(...) <b>cadena</b> s.find(...) <b>cadena</b> s.count(...)