

FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN  
 CCPG1001 - FUNDAMENTOS DE PROGRAMACIÓN  
 PRIMERA EVALUACIÓN - III TÉRMINO 2018-2019/ Marzo 22, 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 ESPOI me comprometo a combatir la mediocridad y actuar con honestidad, por eso no copio ni dejo copiar".

\_\_\_\_\_  
 Firma

**TEMA 1 (40 PUNTOS)**

La cédula ecuatoriana está formada por 10 dígitos que se descomponen de la siguiente manera:

- Los dos primeros dígitos corresponden a la provincia donde fue expedida.
- El tercer dígito es un número menor a 6.
- Los siguientes hasta el noveno dígito forman un número que sigue el orden de expedición.
- El décimo dígito es el dígito verificador, creado mediante el "Módulo 10".

01 Azuay	13 Manabí
02 Bolívar	14 Morona Santiago
03 Cañar	15 Napo
04 Carchi	16 Pastaza
05 Cotopaxi	17 Pichincha
06 Chimborazo	18 Tungurahua
07 El Oro	19 Zamora Chinchipe
08 Esmeraldas	20 Galápagos
09 Guayas	21 Sucumbíos
10 Imbabura	22 Orellana
11 Loja	23 Santo Domingo de los Tsachilas
12 Los Ríos	24 Santa Elena

El algoritmo "Módulo 10" utilizado multiplica los 9 dígitos por 2 o 1 según su posición (índices pares multiplica por 2, impares multiplica por 1). Si el resultado de alguna multiplicación es mayor o igual a 10, entonces se le resta 9. Después se suman todos los resultados y se toma el último dígito de dicha suma. Finalmente, se obtiene el índice verificador al restar la respuesta obtenida de 10. Por ejemplo, para la cédula 0903686962, el proceso del Módulo 10 es el siguiente:

cédula	0	9	0	3	6	8	6	9	6	② → Último dígito
coeficientes	2	1	2	1	2	1	2	1	2	
resultado de multiplicación	0	9	0	3	12	8	12	9	12	
mayores de 10, se resta 9	0	9	0	3	3	8	3	9	3	
se suman los resultados	0 + 9 + 0 + 3 + 3 + 8 + 3 + 9 + 3 = 38									
el último dígito de la suma se resta de 10										10 - 8
dígito verificador:										10 - 8 = ② → Verificador obtenido

\*Si el resultado es 10, el dígito verificador es cero (0)

Realice un programa en Python que solicite el ingreso de una cédula y que luego de haberla verificado, muestre en pantalla la provincia a la que pertenece. Si la cédula ingresada no es válida, se debe mostrar en pantalla algún motivo por la cual no se la acepta. Ejemplos:

Ingrese cédula: 0903686962 La cédula pertenece a la provincia de Guayas
Ingrese cédula : 0983686965 Tercer dígito no valido
Ingrese cédula : 098368 Cédula debe tener 10 dígitos
Ingrese cédula : 09XYZ86962 Cédula debe tener solamente números

Ingrese cédula : 0903686968 Dígito verificador no valido
Ingrese cédula : 2903686968 Cédula no pertenece a ninguna provincia
Ingrese cédula : 098368643453962 Cédula debe tener 10 dígitos

## TEMA 2 (60 PUNTOS)

Se tiene un registro mensual del nivel de contaminación del aire, el cual es almacenado en archivos con el siguiente formato:

"Contaminacion\_mm\_aaaa.txt"

```
dia1:C1,C2, ..., Cn
dia2: C1,C2, ..., Cn
.....
diaN: C1,C2, ..., Cn
```

donde C1,C2, ..., Cn representan la cantidad de partículas por millón de un componente medido en el aire. Asuma que siempre se miden los mismos componentes en el mismo orden.

1. Defina la función **crearReporteAnual(año)**, que recibe como parámetro el año del que se desea crear el reporte. La función crea el archivo Contaminacion\_aaaa.txt, que contiene el siguiente formato:

"Contaminacion\_aaaa.txt"

```
mm: mC1,mC2, ..., mCn
.....
.....
mm: mC1,mC2, ..., mCn
```

donde mm representa el mes (01-12) y mC1,mC2, ..., mCn representan la cantidad total mensual de los componentes. Asuma que ya existen todos los archivos mensuales requeridos.

2. Defina la función **cargarDatos(listaAños)**, que recibe una lista de años y retorna una lista de arreglos que contiene los valores de contaminación de esos años, con el siguiente formato:

[[aC1 aC2 ... aCn] ... [aC1 aC2 ... aCn]]

donde aC1 aC2 ... aCn representan la cantidad total anual de los componentes para cada año ingresado.

Se ha implementado un programa en Python en el cual se crean las siguientes listas:

años = [2018, 2017, ... ]

aire2018, aire2017, ... = cargarDatos(años )

nombreComponentes = ["CO2","O3", ... , "NO"] #Contiene n nombres de componentes

tipoComponente=["Contaminante","Neutro",..., "Contaminante"] #Contiene n tipos de componentes

Escriba el código necesario para mostrar por pantalla la siguiente información:

3. El promedio anual de cada componente.
4. El nombre del componente "Neutro" con mayor promedio anual.
5. ¿Cuántos componentes tienen un promedio mayor a 0.20?
6. ¿Qué tipo de componente tiene una mayor cantidad de emisiones promedio?
7. Solicitar el ingreso por teclado del nombre de un componente y mostrar en qué año se registró la menor cantidad de dicho componente.

---//---

### 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(data,dtype= ) np.zeros((nFilas,nCols),dtype= ) <b>arreglos</b> .shape <b>arreglos</b> .reshape() numpy.sum( <b>arreglos</b> ) numpy.mean( <b>arreglos</b> )	<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(...)