

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN  
CCPG1001 - FUNDAMENTOS DE PROGRAMACIÓN  
SEGUNDA EVALUACIÓN - I TÉRMINO 2018-2019/ Agosto 31, 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)**

Ud. ha sido contratado por la Policía Nacional, para analizar la criminalidad. Para esto, la Policía lleva un registro en un archivo, de la criminalidad en el país por ciudad y tipo de delito. A Ud. le proveen el módulo **criminalidad** con la función **cargarDatos()** que lee dicho archivo y retorna un diccionario con la cantidad de veces que se ha reportado cada tipo de delito en las ciudades del país. El diccionario tiene el siguiente formato:

```
{'Guayaquil': {'robo vehiculo':605, 'asalto':6345,...},  
'Cuenca': {'robo vehiculo':123, 'asalto':676,...},  
...  
}
```

Su trabajo consiste en importar el módulo **criminalidad** e implementar lo siguiente:

- (10 puntos)** Una función **ciudadesCrimenes(diccionario)** que recibe el diccionario de arriba y retorna una tupla de dos elementos. El primer elemento es una tupla con los nombres de las ciudades (sin duplicados) y el segundo es otra tupla con los tipos de crímenes (sin duplicados).
- (20 puntos)** Una función **crearMatriz()** que use la información del diccionario de arriba y retorne la siguiente matriz de Numpy con las cantidades de delitos por ciudades del país:

	Robo vehiculo	Asalto	Escandalo via publica	...
Quito	540	4523	24	...
Guayaquil	605	6345	5	...
Cuenca	123	676	133	...
Machala	67	1234	412	...
...	..	..	..	...

- (10 puntos)** Una función **ciudadesMenosSeguras(matriz, ciudades, delitos, tipoDelito, poblacion)** que recibe la matriz del numeral dos, una tupla con los nombres de todas las ciudades, una tupla con los nombres de todos los delitos, un nombre de delito específico y un vector de Numpy con la población de cada ciudad del país. **ciudades** y **población** aparecen en el mismo orden que las filas en **matriz**. A su vez, **delitos** aparece en el mismo orden que las columnas en **matriz**. La función retorna una tupla con los nombres de las tres ciudades que tienen el mayor índice de criminalidad per cápita para **tipoDelito** en el país.

$$\text{índice per cápita} = \frac{\text{número de incidentes reportados}}{\text{población de la ciudad}}$$

## TEMA 2 (50 PUNTOS)

Una compañía de miles de empleados con distintas sucursales paga sus salarios considerando horas regulares, horas fuera de horario normal, feriados y fines de semana.

**La asistencia de los empleados se registra en archivos individuales por cada mes del año.** En las cinco primeras líneas del archivo de datos, se registra el valor hora a pagar (VH) y los factores que se utilizan para: pagar por horas regulares (HR), horas extras regulares (HER), horas fines de semana (HFDS) y horas feriado (HF). A continuación de esas cinco líneas se encuentra la información sobre las horas trabajadas por cada empleado, incluyendo una cabecera. En el ejemplo se muestran tres registros correspondientes a los empleados **FG849901** y **GH907603** que trabajaron 1 hora el 10 de agosto que fue feriado (Sí) y cayó viernes que es el (5) día de la semana, Se muestra otro ejemplo en el cual **FG849901** trabajó 9 horas el día 9 de agosto y generó por tanto 8 horas regulares (HR) y 1 hora extra en día regular (HER) . **Las horas extras en días regulares se calculan después de la 8va. hora de trabajo:**

```
VH,10,Valor hora en en esta compañía
HR,1, Factor Hora regulares
HER,1.21, Factor Horas extras en dias regulares (lunes-viernes)
HFDS,1.37, Factor Horas en fin de semana (sabado o domingo)
HF,1.39, Factor Horas en feriado
fecha,dia,feriado,ID, nombre, sucursal,ciudad,horas-trabajadas
...
10-Agosto-2018,5,Sí,FG849901,Fabricio Granados,River Mall,Cuenca,1
10-Agosto-2018,5,Sí,GH907603,Segunda Vez Zambrano,River Mall,Cuenca,1
09-Agosto-2018,4,No,FG849901,Fabricio Granados,River Mall,Cuenca,9
```

Si alguien trabaja en un día que es al mismo tiempo **feriado** y **fin de semana**, la tarifa que se aplica es la de **feriado**.

Dada esta información usted debe escribir un programa para calcular lo que la compañía debe pagar a cada empleado usando las siguientes funciones:

1. **(12 puntos)** La función **calcularHoras(línea)** que recibe una línea del archivo previamente descrito. La función determina el número de horas trabajadas en cada categoría y retorna una tupla con el identificador del empleado, el nombre de la ciudad y las horas trabajadas regulares, extras regulares, fines de semana y feriado. Por ejemplo:

```
calcularHoras("09-Agosto-2018,4,No,FG849901,Fabricio Granados,River Mall,Cuenca,9")
retorna ("FG849901","Cuenca",8,1,0,0)
```

2. **(25 puntos)** La función **leerData(nomA)** que recibe el nombre del archivo **nomA** y retorna una tupla de tres elementos. El primer elemento es un diccionario con los **totales en dólares** de HR, HER, HFDS y HF trabajados por cada empleado en las ciudades del país. El segundo elemento de la tupla es el mes de la nómina y el tercer elemento es el año. El diccionario tiene la siguiente estructura:

```
datos= {'Cuenca': { 'FG849901': { 'HR': 530.0,'HER': 36.30, 'HFDS': 0.0,'HF': 1.39},
                  'GH907603': { 'HR': 425.0,'HER': 48.30, 'HFDS': 13.70,'HF': 13.90},
                  ...},
        'Quito' : {...},
        ...
}
```

3. **(13 puntos)** La función **generaReporte(nomA)** que recibe el nombre del archivo con los registros de los empleados (ver ejemplo arriba) y genera un archivo de nómina **para cada ciudad**. El formato del nombre del archivo de salida es ciudadMes-Año.txt y contiene la siguiente información (incluyendo la cabecera):

```
idEmpleado,total$PorHorasRegulares,total$PorHorasExtras (HER+HFDS+HF)
```

### TEMA 3 (10 PUNTOS)

Indique la salida por pantalla del siguiente código. Justifique su respuesta.

```
letters = np.array(['A','Z','P','B','E','R','O','M','B','A','C','D','Q','O'])
indexes = np.array([54, 23, 60, 13, 29, 65, 31, 23, 30, 99, 19, 89, 10, 56])
nl = letters[indexes >= 30]
res = {}
for x in nl:
    res[x] = res.get(x, 0) + 1
print(res)
```

---//---

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

Librería Numpy para <b>arreglos</b> :	para <b>listas</b> :	para <b>diccionarios</b> :	para <b>conjuntos</b> :
<code>np.array((nFilas,nCols),dtype= )</code> <code>np.zeros((nFilas,nCols),dtype= )</code> <code>arreglos.shape</code> <code>arreglos.reshape()</code> <code>numpy.sum(arreglos)</code> <code>numpy.mean(arreglos)</code> <code>arreglos.sum(axis=1)</code>	<code>listas.append(...)</code> <code>listas.extend(...)</code> <code>listas.count(...)</code> <code>listas.index(...)</code> <code>listas.pop()</code> <code>elemento in listas</code>	<code>dicc.items()</code> <code>dicc.keys()</code> <code>dicc.values()</code> <code>dicc.get(clave, valor)</code> <code>dicc.update(dicc2)</code>	<code>c.add(item)</code> <code>c.update(c2)</code> <code>c.intersection(c2)</code> <code>c.union(c2)</code> <code>c.difference(c2)</code>