Programación—Certamen 2 - Jueves 28 de Noviembre de 2013

Nombre: ______________________ Rol: __________ – __

1. [25 %] Realice el ruteo de los siguientes programas e indique qué es lo que imprimen. Cada vez que el valor de una variable cambie, escríbalo en una nueva fila de la tabla. Recuerde que si una variable es de tipo string, coloque su valor entre comillas simples ' '. Si una variable almacena una función, coleque el nombre de ésta como valor (sin comillas).

***Importante**: La tabla tiene suficientes filas.*

```
def f1(x,y):
  s = 0
  for i in x:
    if i%2==0:
      s+=y(i)
    else:
      s+=f4(i)
  return s
```

```
def f2(y,x):
  if x%2==0:
    return f1(y,f3)
  return f1(y,f4)
```

```
def f3(x,y=3):
  i,p = 0,1
  while i<y:
    p*=x
    i+=1
  return p
```

```
def f4(x):
  f=1
  for i in range(x,0,-1):
    f*=i
  return f
```

```
a = [2,3]
c = 0
while (c<len(a)-1):
  print f2(a,c)
  c+=1
```

global		f1				f2		f3				f4		
a	c	x	y	s	i	x	y	x	y	i	p	x	f	i

Programación—Certamen 2 - Jueves 28 de Noviembre de 2013

Nombre: _____ Rol: _____ – _____

2. [35 %] Se desea construir una aplicación para recomendar series de televisión. Para ello se cuenta con la información de todas la series en una lista de tuplas en donde cada tupla tiene la siguiente estructura: `nombre, pais_de_origen, rating, generos`. A su vez generos es una lista con todos los géneros a los cuales pertenece dicha serie. Por ejemplo, la serie 24 es de acción y suspenso.

```
series = [
('game of thrones','USA',9.4,['ficcion']),
('24','USA',8.4,['accion','suspenso']),
('orange is the new black', 'USA', 8.5, ['comedia','drama']),
('sherlock','UK',9.2,['policial','drama','suspenso']),
('whitecollar','USA',8.2,['comedia','drama','suspenso']),
('heroes','USA',7.7,['ficcion','accion']),
('mistfit','UK',8.4,['accion','drama','ficcion'])
# ...
]
```

a) Desarrolle la función `series_por_pais(pais, series)` que recibe el nombre de un `pais` y la lista de `series`, y que permita obtener la lista con las series realizadas en dicho país. Cada elemento de esta lista es una tupla con el nombre de la serie y el rating.

```
>>> series_por_pais('UK',series)
[('sherlock', 9.2), ('mistfit', 8.4)]
```

b) Desarrolle la función `series_por_generos(generos, series)` que reciba una lista con los `generos` a elección y la lista de `series`. Esta función debe retornar un conjunto con todas las series que cumplan con alguno de los géneros pasados por parámetro. Cada elemento de este conjunto es una tupla con el nombre de la serie y el rating.

```
>>> series_por_generos(['drama','ficcion'],series)
set([('game of thrones', 9.4), ('orange is the new black', 8.5),
('sherlock', 9.2), ('whitecollar', 8.2), ('heroes', 7.7),
('mistfit', 8.4)])
```

c) Desarrolle la función `recomendaciones(pais, generos, series)` que, a partir de un `pais` de origen, una lista de `generos` y la lista de `series`, permita obtener el nombre de la serie con mejor rating que cumpla tanto con el país de origen como con alguno de los géneros pasados por parámetro. Puede utilizar las funciones anteriores.

```
>>> recomendaciones('UK',['drama','ficcion'],series)
'sherlock'
```

Programación—Certamen 2 - Jueves 28 de Noviembre de 2013

Nombre: _____ Rol: _____ – _

3. **[40 %]** Un profesor bastante peculiar de la UTFSM realiza viajes académicos cada vez que los alumnos tienen vacaciones (o paros!). El Sansa-Aeropuerto ha notado su afición por viajar y le ha solicitado crear una serie de funciones que ayudarán en su funcionamiento. El profesor aprovechará el Certamen 2 de Programación para pedir a sus alumnos que las realicen por él, y así tener más tiempo para vacacionar. Los datos con los que se trabajará se encuentra en una lista de tuplas y dos diccionarios, todos ellos variables globales, como los siguientes **ejemplos**.

Una lista de tuplas con los datos de `salida` de los vuelos, en donde cada tupla tiene la siguiente estructura, una tupla con la fecha de salida del vuelo, el código del vuelo, la ciudad de origen del vuelo y el estado actual del vuelo, estos últimos campos en `string`.

```
salidas = [ ((2013,11,2), 'LAN123','NewYork','EMBARQUE'),
            ((2013,4,28), 'MX201', 'Cancun', 'ARRIBADO'),
            #...
            ]
```

Un diccionario de `vuelos` que tiene por llave el código del vuelo y por valor un conjunto con todos los ruts de los pasajeros que se encuentran en él.

```
vuelos = {  'LAN123': {'16740623-7', '1111111-1', '555555-5'},
            'ALGO00': {'444444-4'},
            'MX201': {'777777-7'},
            # ...
        }
```

Finalmente, un diccionario de `personas` que tiene como clave el rut de un pasajero y como valor una tupla con el nombre del pasajero, ciudad de origen, fecha de nacimiento y cantidad de millas en su cuenta.

```
personas = {'16740623-7':('OEncina', 'NewYork', (1987, 7, 22), 62000),
            '444444-4':('Edwar Lopez', 'Miami', (1900, 3, 11), 120000),
            '777777-7':('Jorge Perez', 'Santiago', (1989, 2, 17), 1000),
            '555555-5':('Daniela Perez', 'Roma', (1991, 8, 17), 12000),
            '1111111-1':('Sandra Lazo', 'Ibiza', (1970, 4, 14), 10000),
            # ...
            }
```

Se pide:

a) Desarrollar la función `estado_pasajero(nombre)` que a partir del `nombre` del pasajero retorne una tupla con su rut, ciudad de origen y estado del vuelo. Si el pasajero no se encuentra en el sistema se deberá retornar **`None`**.

```
>>> estado_pasajero('OEncina')
('16740623-7', 'NewYork', 'EMBARQUE')
```

b) Desarrollar la función `cambia_de_vuelo(rut, nuevo_vuelo, millas)` que mueva al pasajero de su actual vuelo y lo agregue al `nuevo_vuelo`. Adicionalmente se le sumarán la cantidad de `millas` entregada como parámetro a su cuenta. Esta operación retorna **`True`** si existe el pasajero y **`False`** en caso contrario.

```
>>> cambia_de_vuelo('1111111-1','ALGO00', 5000)
True
```

c) Desarrollar la función `filtro_nac(fecha, estado)` que retorne un conjunto con todos los pasajeros nacidos después de la `fecha` proporcionada y que su vuelo esté en el `estado` indicado.

```
>>> filtro_nac((1980,1,1), 'EMBARQUE')
set(['555555-5, '16740623-7'])
```