

Programación—Certamen recuperativo, jueves 19 de julio de 2012

Nombre:

Rol: -

1. [30%] Indique qué es lo que imprimen los siguientes programas.

```
x = 'y'
y = 'x'
print 'x', 'y'
```

```
x = 1 + 2 * 3
y = 8 > x
z = 0 < x
print y and z
```

```
a = 1452
b = a % 100
c = a / 100
print b + c
```

```
texto = 'a-aa-aaa'
cosas = texto.split('-')
print '\n'.join(cosas)
```

```
s = [(15, 'cm'), (12, 'km'),
      ( 9, 'm'), (17, 'cm')]
t = ''
for c, u in s:
    t += u
print t
```

```
d = {
    'd': {'d': 'x', 's': 'a'},
    's': {'p': 'd', 'm': 'n'},
    'x': {'s': 'x', 'd': 'q'},
}
print d[d['d']]['d']['d']
```

```
a = 100
b = 200
c = 700
while a != b + c:
    if a < b:
        a += b
    if a < c:
        c -= a
print a, b, c
```

```
def f(a, b):
    return a
def g(x, y):
    return y
a, b, x, y = (1, 2, 3, 4)
print g(f(a, x), f(x, b))
```

Programación—Certamen recuperativo, jueves 19 de julio de 2012

Nombre: Rol: -

2. [35 %] La empresa Recuperativix necesita desarrollar un programa para manipular listas de clientes. Cada lista de clientes siempre contiene tuplas con el nombre, el apellido, la fecha de nacimiento y el email de cada cliente. La fecha de nacimiento es una tupla con el año, el mes y el día, en ese orden.

```
clientes = [  
    ('Pedro', 'Perez',      (1970, 12, 31), 'pperez@hutmail.cl'),  
    ('Antonio', 'Nobel',    (1983,  5, 14), 'anobel@llahu.de'),  
    ('Marcela', 'Martinez', (1990,  5, 31), 'm.m@jemail.es'),  
    ('Pepe', 'Ternero',     (1995,  3, 12), 'pe.ter.95@ab.xyz.qwe.es'),  
    ('Alfonsina', 'Lee',    (1992, 11, 28), 'alee@hutmail.cl'),  
    ('Luisa', 'Gonzalez',   (1984,  2, 29), 'lugon@alumnos.usm.cl'),]
```

La parte de la dirección de email que va después del último punto se llama TLD, y está asociada al país de origen del cliente. Por ejemplo, el TLD del email de Luisa González es `cl`, e indica que ella es chilena.

El programa que se está desarrollando tiene dos variables globales: el diccionario `países`, que asocia a cada TLD el país correspondiente, y la lista `meses`, que tiene los nombres de los doce meses del año.

```
países = {  
    'es': 'España',  
    'uk': 'Reino Unido',  
    'de': 'Alemania',  
    'bw': 'Botswana',  
    'cl': 'Chile',  
}  
  
meses = ['enero', 'febrero', 'marzo', 'abril',  
         'mayo', 'junio', 'julio', 'agosto',  
         'septiembre', 'octubre', 'noviembre', 'diciembre']
```

- a) Escriba la función `contar_paises(clientes)` que retorne un diccionario que asocie a cada país la cantidad de clientes que provienen de él.

```
>>> contar_paises(clientes)  
{'España': 2, 'Alemania': 1, 'Chile': 3}
```

- b) Escriba la función `persona_mas_joven(clientes)` que retorne el nombre completo del cliente más joven de la lista.

```
>>> persona_mas_joven(clientes)  
'Pepe Ternero'
```

- c) Al comienzo de cada mes, Recuperativix desea enviar un email felicitando a todas las personas que tienen cumpleaños durante el mes.

Escriba la función `obtener_mails_cumpleaños(clientes, mes)` que retorne el conjunto de las direcciones de email de los clientes que tienen cumpleaños en el mes indicado.

```
>>> obtener_mails_cumpleaños(clientes, 'mayo')  
set(['anobel@llahu.de', 'm.m@jemail.es'])
```

Note que el valor del parámetro debe ser el nombre del mes, no su número.

Programación—Certamen recuperativo, jueves 19 de julio de 2012

Nombre:

Rol: -

3. [35 %] En un torneo de golf, cada jugador debe intentar meter la bola en cada uno de los 18 hoyos del campo usando la menor cantidad de golpes posible.

Cada uno de los hoyos tiene asignado un *par*, que es la cantidad de golpes en que se espera que un jugador consiga el hoyo. El par de un hoyo suele ser 3, 4 o 5, pero podría ser cualquier valor.

Al final del día, cada jugador debe entregar una tarjeta indicando cuántos golpes hizo en cada hoyo. Esta tarjeta antes se hacía en papel, pero hoy en día los jugadores simplemente envían por correo electrónico un archivo de texto de 18 líneas, cada una de las cuales tiene un número entero. El nombre del archivo debe ser `golpes-nombre-apellido.txt`, reemplazando con el nombre y el apellido del jugador en minúsculas.

La organización del torneo usa dos archivos adicionales para calcular el marcador del juego. El archivo `pares.txt` también tiene 18 líneas, cada una de las cuales tiene el par del hoyo correspondiente. El archivo `jugadores.txt` tiene los nombres de los jugadores que participan del torneo.

Archivo `jugadores.txt`:

```
Tiguerio Gutierrez
Arnoldo Palmeras
Jack Nicolas
```

Para calcular el puntaje final de un jugador, se suman las diferencias entre el par de cada hoyo y el número de golpes que necesitó para completarlo. Un puntaje negativo es mejor que uno positivo, porque significa que necesitó menos golpes en total.

- a) Escriba la función `leer_valores(nombre_archivo)`, que recibe como parámetro el nombre de un archivo que tiene 18 enteros (uno por línea), y retorne un arreglo con esos valores.

```
>>> leer_valores('pares.txt')
array([5, 3, 3, 4, 3, 5, 3, 3, 5, 4, 3, 3, 4, 4, 3, 5, 4, 4])
>>> leer_valores('golpes-arnoldo-palmeras.txt')
array([6, 3, 3, 4, 3, 6, 2, 2, 6, 4, 3, 3, 3, 5, 3, 6, 4, 4])
```

(De este ejemplo podemos deducir que el puntaje final de Arnoldo Palmeras fue +2).

- b) Escriba la función `obtener_marcador()`, que retorne un diccionario que asocie a cada jugador su puntaje final.

```
>>> obtener_marcador()
{'Jack Nicolas': -1, 'Arnoldo Palmeras': 2, 'Tiguerio Gutierrez': 0}
```

- c) Escriba la función `obtener_ganador()`, que retorne el nombre del ganador del torneo.

```
>>> obtener_ganador()
'Jack Nicolas'
```

Si hay varios jugadores empatados en el primer lugar, elija a cualquiera (¡pero sólo a uno!) de ellos.