## Programación—Certamen Recuperativo - 21 de Agosto de 2018

Nombre			
Rol		Paralelo	

**1.** [20%] Realice el ruteo del siguiente programa e indique qué es lo que imprime. 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, su valor debe ir entre comillas simples ''.

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

```
def s(c, p):
    c1 = ''
    c2 = ''
    i = 0
    while i < len(c):
        if i < p:
            c1 += c[i]
        else:
            c2 = c2 + c[i]
        i = 1 + i
        return c1, c2

x = 'girasol'
x = s(x, 4)
print x[1]
print x[0]</pre>
```

global			S		
X	С	p	c1	c2	i

## Análisis de Algoritmo

Analice la función **s** anterior y determine, **en menos de 30 palabras**, lo que realiza. No debe usar más del espacio indicado.

## Programación—Certamen Recuperativo - 21 de Agosto de 2018

Nombre	
Rol	Paralelo Paralelo

**2.** [40 %] **NOTA**: Puede hacer uso de las funciones definidas en la pregunta 1.

Pythonmed es un centro médico donde se puede solicitar una hora de consulta con distintos especialistas. Considere la lista de tuplas medicos, donde cada tupla de la lista contiene el rut, nombre y especialidad de un médico.

Por otra parte, los horarios de atención se almacenan en el diccionario agenda. Cada llave de esta estructura corresponde al rut de un médico y el valor es un conjunto de tuplas con información de los horarios de atención. Considere la información de horarios de atención como tuplas con un string de la fecha y hora en el formato AAAAMMDDHHMM y un string que indica disponibilidad (O: libre, X: ocupado). Los puntos suspensivos en los ejemplos de las estructuras presentadas, señalan que pueden existir muchos más registros, por lo que **asuma que hay muchos más datos que los presentados**.

```
agenda = {'13594298-4': {('201807231400', 'X'),('201808111500', 'X'), ('201807231430', 'O'), ...}, '17302713-1': {('201807211400', 'X'), ('201807261100', 'O'), ('201808161430', 'O')}, ...}
```

a) Escriba la función agenda\_por\_dia(agenda, rut, fecha) que reciba la estructura agenda, un string con el rut de un médico y una fecha en el formato AAAAMMDD. La función debe retornar una lista de tuplas con los horarios de atención para el rut y fecha ingresados como parámetro. Esta lista deberá estar ordenada por hora en forma ascendente. Asuma que el rut del médico ingresado, siempre existirá y estará correcto.

```
>>> agenda_por_dia(agenda, "13594298-4", "20180723")
[..., ('201807231400', 'X'), ('201807231430', 'O'),
('201807231500', 'X'), ...]
```

b) Escriba la función solicitar hora (agenda, rut, horario) que reciba como parámetros la estructura agenda, un string con el rut del médico y un string con el horario de atención en el formato AAAAMMDDHHMM. La función debe verificar si hay disponibilidad de atención del médico para el horario ingresado y retornar la agenda actualizada con el horario de atención ocupado. Si no existe el horario ingresado o está ocupado debe retornar False. Asuma que el rut del médico ingresado, siempre existirá y estará correcto.

```
>>> solicitar_hora(agenda, "13594298-4", "201807231400")
False
>>> solicitar_hora(agenda, "13594298-4", "201807231430")
{'13594298-4': set([('201807231400', 'X'), ('201808111500', 'X'), ('201807231430', 'X'), ...]), ...)}
```

Nombre	
Rol	Paralelo Paralelo
[40 %] Pytho	on airlines mantiene un exhaustivo recuento de todos sus vuelos y pilotos. Cada uno
	os tiene un identificador único y al igual que cada piloto. La empresa ha almacenado
	vo itinerario2018.txt todos los vuelos realizados el año anterior, donde cada
linea tiene e	el formato id_vuelo:origen/destino/kilometros:
	itinerario2018.txt
NAC23:San	tiago/Iquique/1600
INT12:Sac	Paulo/Paris/9401
	on/Paris/400
_	gota/Santiago/6383
_	gota/San Andres/1209
_	rique/Punta Arenas/3600
NAC25:Pun	ta Arenas/Santiago/2000
	· · ·
VP2018.tx	tt donde cada línea tiene el formato id_piloto-mes:id_vuelo1;id_vuelo2,  VP2018.txt
VP2018.tx	tt donde cada línea tiene el formato id_piloto-mes:id_vuelo1;id_vuelo2,  VP2018.txt  INT12;NAC22
VP2018.tx 12-marzo: 11-marzo:	INT12; NAC22 NAC23; INT33; NAC49
VP2018.tx 12-marzo: 11-marzo:	tt donde cada línea tiene el formato id_piloto-mes:id_vuelo1;id_vuelo2,  VP2018.txt  INT12;NAC22
VP2018.tx 12-marzo: 11-marzo:	VP2018.txt  INT12; NAC22 NAC23; INT33; NAC49
12-marzo: 11-marzo: 3-noviemb  Se pide que realizar un redel mes y e	VP2018.txt  INT12; NAC22 NAC23; INT33; NAC49 Ore: NAC34; NAC25  desarrolle la función resumen_piloto (mes) que reciba el mes para el cual se quiere resumen. La función debe escribir un archivo resumen_X.txt donde X es el nombre
12-marzo: 11-marzo: 3-noviemb Se pide que realizar un redel mes y etotales reconsolo deben mes pedido función con	VP2018.txt  INT12; NAC22 NAC23; INT33; NAC49 Pre: NAC34; NAC25  desarrolle la función resumen_piloto (mes) que reciba el mes para el cual se quiere resumen. La función debe escribir un archivo resumen_X.txt donde X es el nombre en cada linea de archivo debe contener el id_piloto y la cantidad de kilómetros en cada linea de archivo debe contener el id_piloto y la cantidad de kilómetros en cada linea de archivo debe contener el id_piloto y la cantidad de kilómetros en cada linea de archivo debe contener el id_piloto y la cantidad de kilómetros en cada linea de archivo debe contener el id_piloto y la cantidad de kilómetros en cada linea de archivo debe contener el id_piloto y la cantidad de kilómetros en cada linea de archivo debe contener el id_piloto y la cantidad de kilómetros en cada linea de archivo debe contener el id_piloto y la cantidad de kilómetros en cada linea de archivo debe contener el id_piloto y la cantidad de kilómetros en cada linea de archivo debe contener el id_piloto y la cantidad de kilómetros en cada linea de archivo debe contener el id_piloto y la cantidad de kilómetros el cada linea de archivo debe contener el id_piloto y la cantidad de kilómetros el cada linea de archivo debe contener el id_piloto y la cantidad de kilómetros el cada linea de archivo debe contener el id_piloto y la cantidad de kilómetros el cada linea de archivo debe contener el id_piloto y la cantidad de kilómetros el cada linea de archivo debe contener el id_piloto y la cantidad de kilómetros el cada linea de archivo debe contener el cada linea de archivo debe c