



## Ciclo 1 Fundamentos de programación

### Reto 4

**Descripción del problema:** La dependencia de cartera de la empresa de servicios públicos gas de la ciudad AXY requiere realizar un informe de la general del servicio prestados según las lecturas generadas durante un periodo. Para cada predio se tiene la siguiente información: **id\_predio** el cual es una llave único además de los valores del diccionario que son los siguientes:

Nombre	Tipo	Descripción
toma_lectura	list	Lista que contiene la información de la toma de lectura
estrato	int	Categorización de los predios (1,2,3,4,5,6)
estado	str	Estado del predio ('activo','inactivo')

Adicionalmente, cada uno de los elemento de la lista toma de lectura es un diccionario que contiene la información de consumo, con la siguiente estructura:

Nombre	Tipo	Descripción
lec_anterior	int	Lectura anterior del predio
lec_actual	int	Lectura actual del predio

### Diccionario Plan Tarifario Estático:

Nombre	Tipo	Descripción
cargo_basico	float	Valor cargo básico del periodo correspondiente.
consumo	float	Valor consumo por cada metro.
escala_sud	int	Escala para cobro por consumo según el subsidio de cada predio según su categorización

**Requerimiento:** Escriba una función que reciba dos diccionario que contiene la información previamente especificada. Retornar una tupla en la que el primer elemento es una lista que



contiene una tupla con el id\_predio, total liquidado por predio el cual deben ser (redondeados a dos dígitos), un segundo elemento de la tupla debe ser el total a pagar de todos los predios el cual deben ser (redondeados a dos dígitos), el tercer elemento de la tupla debe ser una lista con el total subsidio por consumo y total subsidio por cargo fijo de todos los predios el cual deben ser (redondeados a dos dígitos).

La forma de liquidar cada uno de los predios es el siguiente:

- Estrato 1 tiene un subsidio del 45 por ciento en el cargo fijo y consumo teniendo en cuenta la escala de subsidio.
- Estrato 2 tiene un subsidio del 35 por ciento en el cargo fijo y consumo teniendo en cuenta la escala de subsidio.
- Estrato 3 tiene un subsidio del 10 por ciento en el cargo fijo y consumo teniendo en cuenta la escala de subsidio.
- Los estratos del 4 al 6 tiene un contribución del 40 por ciento en el cargo fijo y consumo.

**Esqueleto:**

```
def inforServicio(lectura : dict, tarifa : dict)-> tuple:  
    pass
```

**Ejemplo:**

dict	dict
<pre>{   '501001190001': {     'toma_lectura': [       {         'lec_anterior': 1232,         'lec_actual': 1304,       }     ],     'estrato': 1,     'estado': 'activo'   },   '501002190324': {     'toma_lectura': [       {         'lec_anterior': 1203,         'lec_actual': 1230,       }     ],     'estrato': 4,     'estado': 'activo'   } }</pre>	<pre>tarifa = {   'cargo_basico': 13405.45,   'consumo': 1100.80,   'escala_sub': 15 }</pre>



return

(((('501001190001', 79200.2), ('501002190324', 60377.87)), 139578.07, [7430.4, 6032.45]))

dict

dict

```
{
  '501001190001' :{
    'toma_lectura': [
      {
        'lec_anterior': 1232,
        'lec_actual': 1304,
      }
    ],
    'estrato': 1,
    'estado': 'inactivo'
  }
}
```

```
tarifa = {
  'cargo_basico': 13405.45,
  'consumo': 1100.80,
  'escala_sub': 24
}
```

return

Sin lecturas

dict

dict

```
{
  '201501001' :{
    'toma_lectura': [
      {
        'lec_anterior': 12,
        'lec_actual': 60,
      }
    ],
    'estrato': 1,
    'estado': 'activo'
  },
  '201501002' :{
    'toma_lectura': [
      {
        'lec_anterior': 2,
        'lec_actual': 6,
      }
    ],
    'estrato': 2,
    'estado': 'activo'
  },
  '201501003' :{
    'toma_lectura': [
      {
        'lec_anterior': 23,
        'lec_actual': 43,
      }
    ],
    'estrato': 3,
    'estado': 'activo'
  },
  '201501004' :{
    'toma_lectura': [
      {
        'lec_anterior': 90,
        'lec_actual': 120,
      }
    ],
    'estrato': 1,
    'estado': 'activo'
  },
  '201501005' :{
    'toma_lectura': [

```

```
tarifa = {
  'cargo_basico': 13405.45,
  'consumo': 1100.80,
  'escala_sub': 20
}
```



```
{
  'lec_anterior': 1,
  'lec_actual': 9,
}
],
'estrato': 1,
'estado': 'inactivo'
},
'201564006': {
  'toma_lectura': [
    {
      'lec_anterior': 10,
      'lec_actual': 20,
    }
  ],
  'estrato': 6,
  'estado': 'activo'
}
}
```

**return**

```
((('201501001', 52781.0), ('201501002', 11575.62), ('201501003', 32429.71), ('201501004', 32966.6), ('201564006', 34178.83)), 163931.76, [31262.72, 18097.36])
```

### Salida:

Tipo del retorno	Descripción
tuple	<code>(([<i>id_predio</i>, <i>total_predio</i>]), <i>total</i>, [<i>total_sub_consumo</i>, <i>total_sub_cargofijo</i>])</code>

**Validaciones:** tenga en cuenta que si el **estado: 'inactivo'** debe ser eliminado del informe a presentar, además si no hay ningún predio activo se debe enviar un mensaje 'Sin lecturas'.

**Observaciones:** Tenga en cuenta la utilización de las funciones (`filter`, `reduce`, `zip`, `lambda`).