



RETO 4 – FUNDAMENTOS DE PROGRAMACIÓN

VARIANTE 3

Una reconocida asociación bancaria (Donde usted trabaja en el área de Ingeniería informática) en cada sede cuenta con tres colas (O filas):

1. General: Cola donde los clientes de la entidad bancaria reclaman un ficho para pasar a una de las otras filas.
2. Caja: Cola donde los clientes de la entidad bancaria pueden retirar o consignar la cantidad que ellos desean (En el caso de retiro, solo pueden retirar como máximo la cantidad disponible en sus cuentas bancarias).
3. Información: Cola donde los clientes de la entidad bancaria pueden solicitar cualquier tipo de información.

La entidad necesita un software que, dada una cola General, la distribuya en las otras dos colas (Suponga que ninguna persona se va sin antes ser atendido), y además calcule:

- La suma total de todos los retiros a realizar.
- La suma total de todas las consignaciones a realizar.
- La edad mínima de las personas que harán retiro de dinero.
- La edad mínima de las personas que pedirán información.
- La edad mínima de las personas que harán una consignación de dinero.

Usted es el desarrollador a cargo de esta nueva funcionalidad requerida por la entidad.

Para ello, se define que se usará POO (Programación Orientada a Objetos) para representar a los clientes de la entidad bancaria; la clase cliente cuenta con los siguientes atributos:

- Nombre
- Edad
- Cantidad de dinero en la cuenta bancaria
- Fila de interés (Caja o información)
- Tipo de transacción a realizar (Retiro, consignación, o ninguna)
- Cantidad para retirar (Si el cliente **NO** va a retirar, la cantidad es 0)
- Cantidad para consignar (Si el cliente **NO** va a consignar, la cantidad es 0)





TAREAS

- Mediante la programación de la función requerida (Descrita en los siguientes numerales), realizar una función en Python que permita a la entidad financiera distribuir los clientes de la cola General a las demás colas (Teniendo en cuenta el orden de llegada) y calcular los datos requeridos anteriormente.

Usted no debe pedir datos a través de la función *input*, solo asegurarse que la función *sede_bancaria* retorne los valores solicitados a continuación usando los datos que recibe como parámetro.

FORMATO DE ENTRADA

La función *sede_bancaria(cola_general)* recibe como parámetro:

- cola_general**: Lista de Python (Objeto de la clase **list**) que contiene la información de cada persona que está haciendo fila en la cola general, cada persona está representada como un objeto de la clase cliente (Esta clase ya viene creada, **NO** modificarla).

La clase persona está implementada de la siguiente manera:

```
1 class Cliente:
2     def __init__(self, nombre, edad, dinero_cuenta_bancaria, fila_interes, transaccion, cantidad_retirar, cantidad_consignar):
3         self.nombre = nombre
4         self.edad = edad
5         self.dinero_cuenta_bancaria = dinero_cuenta_bancaria
6         self.fila_interes = fila_interes
7         self.transaccion = transaccion
8         self.cantidad_retirar = cantidad_retirar
9         self.cantidad_consignar = cantidad_consignar
```

Estas son las posibles filas de interés de las que puede escoger el cliente:

CADENA DE TEXTO	OPERACIÓN
"caja"	Desencola de la fila general, y encola en cola_caja
"info"	Desencola de la fila general, y encola en cola_info





Estos son los tipos de transacciones permitidas de las que puede escoger el cliente:

CADENA DE TEXTO
"retirar"
"consignar"
"ninguna"

Ejemplo:

```
pepito = cliente('Pepe', 24, 0, 'info', 'ninguna', 0, 0)
```

Esto quiere decir que el cliente Pepe, tiene 24 años, no dispone de dinero en su cuenta bancaria (0), va a ir a la fila de información, no hará ninguna transacción, por lo que las cantidades a consignar y retirar son 0.





FORMATO DE SALIDA

La función `sede_bancaria(cola_general)` debe hacer los siguientes retornos en ese orden:

VARIABLE A RETORNAR	TIPO DEL VALOR QUE SE GUARDA	DESCRIPCIÓN
<code>cola_caja</code>	Lista de Python (Objeto de la clase <code>list</code>)	Contiene los nombres de todas las personas que harán cola en caja
<code>cola_info</code>	Lista de Python (Objeto de la clase <code>list</code>)	Contiene los nombres de todas las personas que harán cola en información
<code>suma_retiros</code>	Número entero (Objeto de la clase <code>int</code>)	Contiene la suma total de dinero de todos los retiros a realizar en el conjunto de personas dado
<code>suma_consignaciones</code>	Número entero (Objeto de la clase <code>int</code>)	Contiene la suma total de dinero de todas las consignaciones a realizar en el conjunto de personas dado
<code>edad_minima_retiro</code>	Número entero (Objeto de la clase <code>int</code>)	Contiene la edad de la persona más joven de las personas que van a realizar retiro de dinero
<code>edad_minima_info</code>	Número entero (Objeto de la clase <code>int</code>)	Contiene la edad de la persona más joven de las personas que van a pedir información
<code>edad_minima_consignacion</code>	Número entero (Objeto de la clase <code>int</code>)	Contiene la edad de la persona más joven de las personas que van a realizar consignación de dinero





En caso tal de no haber personas que hagan retiro, consignación o pidan información, la edad mínima será -1 (Es decir, como este valor es imposible, permitirá identificar que no hubo gente que hizo retiro, consignaciones o haya pedido información).

EJEMPLOS

ENTRADA:

```
cola_general = [  
    cliente("Matt",21,235000,"caja","retirar",100000,0),  
    cliente("Dan",32,658000,"caja","retirar",98000,0),  
    cliente("Diana",29,87000,"info","ninguna",0,0)  
]
```

SALIDA:

```
cola_caja = ["Matt","Dan"]  
cola_info = ["Diana"]  
suma_retiros = 198000  
suma_consignaciones = 0  
edad_minima_retiro = 21  
edad_minima_info = 29  
edad_minima_consignacion = -1
```

