

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ  
FACULTAD DE CIENCIAS E INGENIERÍA

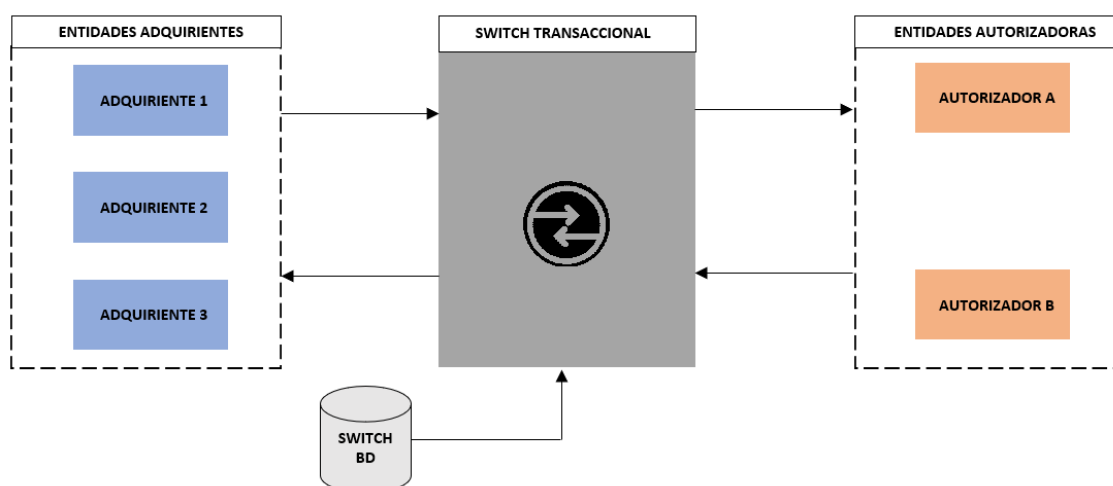
[1IEE14] ARQUITECTURA DE COMPUTADORAS

Laboratorio 8  
(Segundo semestre 2023)

Indicaciones generales:

- Debe enviar su solución, siguiendo el formato indicado en cada pregunta.
- **La hora de entrega es hasta culminadas las 3 horas de evaluación de práctica.**
- Puede utilizar apuntes de clase

Un switch transaccional tiene la función de rutear transacciones financieras provenientes de distintas entidades adquirentes hacia una determinada entidad autorizadora.



Las entidades autorizadoras A y B requieren hacer un reporte de las transacciones financieras recibidas en el presente año, pero un fallo en sus sistemas de base de datos ha ocasionado que la información almacenada se corrompa. Por ello, han solicitado a la empresa GLOBAL FINANCIAL S.A.C, desarrolladora del software que utilizan como Switch Transaccional, reenviar las transacciones procesadas durante dicho periodo de tiempo.

Convenientemente, GLOBAL FINANCIAL S.A.C posee una base de datos que almacena las transacciones procesadas anualmente, en la cual se dispone del archivo `transacciones-2023.txt` con la información de interés. Se muestran las primeras líneas del archivo `.txt`:

Interfaz Adquiriente --- Transacción codificada --- Campos de la Transacción

```
transacciones-2023.txt
1  izipy000 --- 7104552265 TZ1D3BAF0008 00000000000000000000000000000000 --- B002 LEN [16] PAN
2  izipy000 --- 7104552265 TZ1D3BAF0008 00000000000000000000000000000000 --- B010[531000]
3  izipy000 --- 7104552265 TZ1D3BAF0008 00000000000000000000000000000000 --- B011[418196]
4  izipy000 --- 7104552265 TZ1D3BAF0008 00000000000000000000000000000000 --- B012[111339]
5  izipy000 --- 7104552265 TZ1D3BAF0008 00000000000000000000000000000000 --- B013[0504]
6  izipy000 --- 7104552265 TZ1D3BAF0008 00000000000000000000000000000000 --- B018[5942]
7  izipy000 --- 7104552265 TZ1D3BAF0008 00000000000000000000000000000000 --- B020[301060]
8  izipy000 --- 7104552265 TZ1D3BAF0008 00000000000000000000000000000000 --- B022[051]
9  izipy000 --- 7104552265 TZ1D3BAF0008 00000000000000000000000000000000 --- B033[418196]
10 izipy000 --- 7104552265 TZ1D3BAF0008 00000000000000000000000000000000 --- B040[501001]
11 *****
12 kasnt000 --- 2309694410 NH1D2CJG0003 00000000000000000000000000000000 --- B002 LEN [16] PAN
13 kasnt000 --- 2309694410 NH1D2CJG0003 00000000000000000000000000000000 --- B010[000440]
14 kasnt000 --- 2309694410 NH1D2CJG0003 00000000000000000000000000000000 --- B011[812060]
15 kasnt000 --- 2309694410 NH1D2CJG0003 00000000000000000000000000000000 --- B012[151016]
16 kasnt000 --- 2309694410 NH1D2CJG0003 00000000000000000000000000000000 --- B013[0609]
17 kasnt000 --- 2309694410 NH1D2CJG0003 00000000000000000000000000000000 --- B018[5982]
18 kasnt000 --- 2309694410 NH1D2CJG0003 00000000000000000000000000000000 --- B020[707080]
19 kasnt000 --- 2309694410 NH1D2CJG0003 00000000000000000000000000000000 --- B022[000]
20 kasnt000 --- 2309694410 NH1D2CJG0003 00000000000000000000000000000000 --- B033[418196]
21 kasnt000 --- 2309694410 NH1D2CJG0003 00000000000000000000000000000000 --- B040[501001]
22 *****
23 niubz000 --- 6406617633 OP1F6HUI0005 00000000000000000000000000000000 --- B002 LEN [16] PAN
24 niubz000 --- 6406617633 OP1F6HUI0005 00000000000000000000000000000000 --- B010[070711]
25 niubz000 --- 6406617633 OP1F6HUI0005 00000000000000000000000000000000 --- B011[302176]
26 niubz000 --- 6406617633 OP1F6HUI0005 00000000000000000000000000000000 --- B012[095120]
27 niubz000 --- 6406617633 OP1F6HUI0005 00000000000000000000000000000000 --- B013[2007]
28 niubz000 --- 6406617633 OP1F6HUI0005 00000000000000000000000000000000 --- B018[5942]
29 niubz000 --- 6406617633 OP1F6HUI0005 00000000000000000000000000000000 --- B020[205050]
30 niubz000 --- 6406617633 OP1F6HUI0005 00000000000000000000000000000000 --- B022[051]
31 niubz000 --- 6406617633 OP1F6HUI0005 00000000000000000000000000000000 --- B033[418196]
32 niubz000 --- 6406617633 OP1F6HUI0005 00000000000000000000000000000000 --- B040[207002]
33 *****
```

La información asociada a los campos de la transacción se especifican en la siguiente tabla:

Campo	Información
B010	Código de tipo de transacción
B012	Hora de envío de la transacción (HHMMSS)
B013	Fecha de envío de transacción (DDMM)
B020	Código de entidad adquiriente
B040	Código de entidad autorizadora

Los códigos de las entidades autorizadoras de interés también son conocidos:

Autorizador	Código (B040)
A	501001
B	207002

Se le pide lo siguiente:

- 1) Implementar una comunicación cliente-servidor (considerar `SIZE_BUFFER = 128`) entre la base de datos y el switch transaccional, respectivamente, con la finalidad de transferir el archivo `transacciones-2023.txt` **(4 p)**
  - Para ello debe crear un par de scripts: `BD_switch.py` y `switch_server.py`, los cuales deben estar en diferentes directorios. **(2.5 p)**
  - Medir el tiempo de lectura, envío, recepción y escritura e imprimir dichos valores en los respectivos scripts. **(1.5 p)**
- 2) Implementar una comunicación cliente-servidor (considerar `SIZE_BUFFER = 128`) entre el switch transaccional y cada uno de los autorizadores, respectivamente. Para ello, considerar: **(11 p)**
  - Implementar el script `switch_client.py`, el cual procesa `transacciones-2023.txt`, y escribe los archivos `transacciones-2023_autA.txt` y `transacciones-2023_autB.txt`, cada uno de los cuales contiene únicamente las transacciones cuyo destino era AUTORIZADOR A y al AUTORIZADOR B, respectivamente. **(4p)**

**NOTA: Debido a que los autorizadores desean optimizar la capacidad de almacenamiento de sus servidores, han solicitado que solo se envíen los campos B010, B012, B013, B020 y B040 de cada transacción.**

- Transferir el archivo `transacciones-2023_autA.txt` al AUTORIZADOR A a través de una comunicación cliente-servidor (considerar `SIZE_BUFFER = 128`). Para ello, es necesario implementar el script `autorizador_A.py` (servidor). **(1.5 p)**
- Transferir el archivo `transacciones-2023_autB.txt` al AUTORIZADOR B a través de una comunicación cliente-servidor (considerar `SIZE_BUFFER = 128`). Para ello, es necesario implementar el script `autorizador_B.py` (servidor). **(1.5 p)**
- Modificar los scripts `switch_client.py`, `autorizador_A.py` y `autorizador_B.py`, de tal manera de medir exclusivamente los tiempos de envío y recepción (omitir lectura/escritura), para un tamaño de buffer variable (`SIZE_BUFFER = {64, 128, 256, 512, 1024}`). Obtener una gráfica

de Tiempo vs N en cada caso y analizar sus resultados (brindar un comentario en los scripts correspondientes). **(4 p)**

- 3) Los autorizadores requieren hacer un reporte en donde se especifique el la cantidad de transacciones de cada tipo recibida desde cada adquirente distinto. Para ello, se brinda la siguiente información: **(5 p)**

<b>B010</b>	<b>Tipo de transacción</b>
000440	Retiro
230060	Depósito
531000	Consulta de movimientos
070711	Pago de préstamo
<b>B020</b>	<b>Entidad Adquiriente</b>
301060	Izipay
707080	Globokas
205050	Niubiz

- Modificar el script `autorizador_A.py` de tal manera que procese la información en `transacciones-2023_autA.txt` y escriba el archivo `reporte.txt`, el cual contiene el reporte de interés con el siguiente formato: **(2 p)**

```
autorizador_A > ≡ reporte.txt
1  IZIPAY
2  000440 = 0
3  230060 = 0
4  531000 = 16
5  070711 = 0
6  KASNET
7  000440 = 16
8  230060 = 24
9  531000 = 8
10 070711 = 8
11 NIUBIZ
12 000440 = 16
13 230060 = 8
14 531000 = 0
15 070711 = 0
```

- Modificar el script `autorizador_B.py` de tal manera que procese la información en `transacciones-2023_autB.txt` y escriba el archivo `reporte.txt` (considerar el formato anterior) **(1 p)**
- Medir el tiempo que toma generar (procesar/escribir) los archivos `reporte.txt` en cada autorizador e imprimir la velocidad asociada a cada autorizador. ¿Cuál autorizador es más rápido? ¿A qué se deben dichos resultados? **(2 p)**

Se brinda la siguiente sugerencia para poder organizar los archivos a implementarse:

