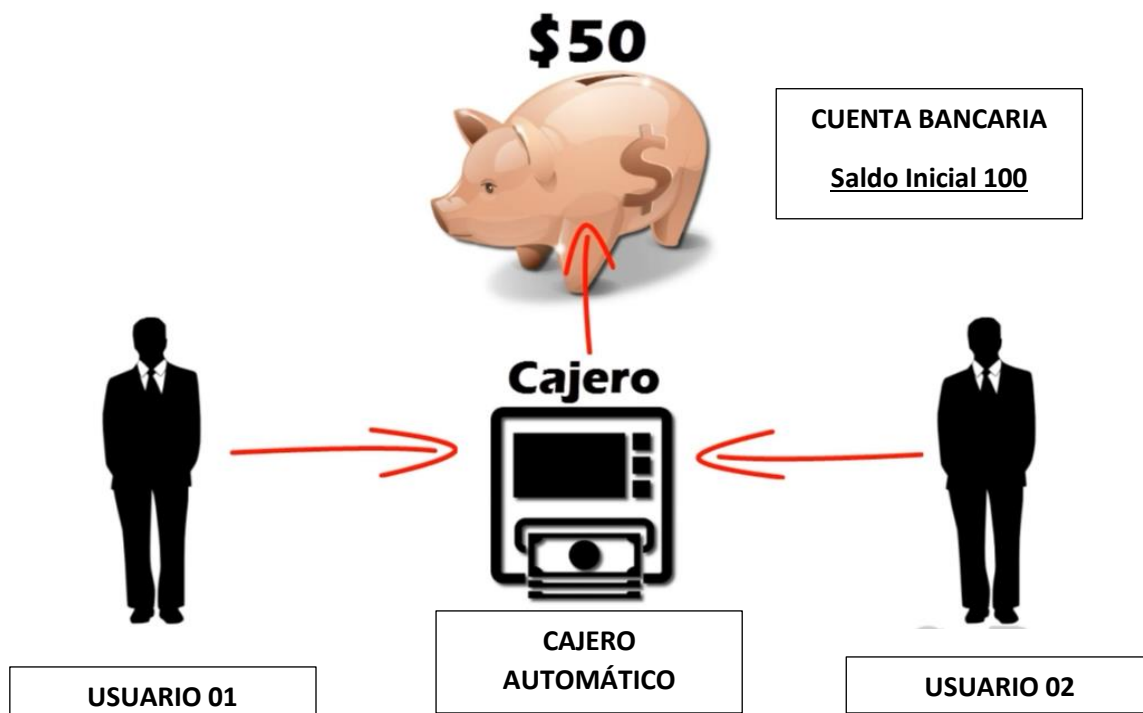


EJERCICIO PRÁCTICO DE HILOS

Implementar y Diseñar las clases con atributos y métodos necesarios, para desarrollar el siguiente caso, aplicando los conceptos de Thread en Java

Ejercicio 1

- El caso consiste en **simular una cuenta bancaria** que es compartida por dos hermanos (**2 usuarios**, las cuales se encuentran en ciudades distintas) dichos hermanos pueden acceder a la cuenta y van a poder hacer retiradas de efectivo con base a un **saldo inicial de 100 euros en cuenta**.
- Dicha cuenta, es mantenida por su papuchi, que les hace **un ingreso periódico de 100 euros, con un tiempo que puede variar entre 1 y 3 meses (serán segundos en este caso)**.
- Cada usuario puede hacer transacciones y el saldo inicial se debe de actualizar **cada vez que se realiza un retiro**) por parte de cualquiera de los hermanos.
- Las transacciones a la cuenta bancaria (ingresos o retiros) por parte de los usuarios se pueden realizar mediante el cajero automático, pero debe de bloquearse al otro usuario si desea realizar una transacción al mismo tiempo.
- La aplicación finalizará cuando entren en descubierto (0 o menos euros a deber en cuenta).



Ejercicio 2

Se pide resolver el problema de los productores y los consumidores. Se va a definir 3 clases: el **hilo Productor**, el **hilo Consumidor**, y el **objeto Recipiente** donde el productor deposita el valor producido, y de donde el consumidor extrae los datos.

- El productor se ejecuta durante 10 iteraciones y en cada una de ellas deposita en el recipiente el número de la iteración actual. Entre iteración e iteración se quedará durmiendo durante un tiempo aleatorio entre 1 y 2 segundos.
- El consumidor se ejecuta el mismo número de iteraciones que el productor, pero en cada una de ellas saca el valor almacenado en el recipiente y lo muestra por pantalla. Entre cada iteración duerme también un tiempo aleatorio entre 1 y 2 segundos.
- El recipiente proporciona los métodos produce y consume para depositar un dato en él y para sacarlo de él respectivamente.

El programa mostrará cuando el productor produce un valor y cuando el consumidor lo consume. El funcionamiento correcto debería ser que el consumidor consuma exactamente los mismos valores que el productor ha producido, sin saltarse ninguno ni repetirlos. Se pide:

- a. Implementar el código necesario en los métodos produce y consume para sincronizar el acceso a ellos. El comportamiento debería ser el siguiente:
 - Si queremos producir y todavía hay datos disponibles en el recipiente, esperaremos hasta que se saquen, si no produciremos y avisamos a posibles consumidores que estén a la espera.
 - Si queremos consumir y no hay datos disponibles en el recipiente, esperaremos hasta que se produzcan, si no consumimos el valor disponible y avisamos a posibles productores que estén a la espera.

Implemente los métodos necesarios para la sincronización. No olvidar de manejar las excepciones.

Ejercicio 3

Ingresa los nombres de tres mascotas y luego simular una carrera de 30 metros entre ellos e indicar, quien llega primero a la meta. Se debe recoger la prioridad y el avance de cada una.

Este ejercicio es para comprobar como se maneja las prioridades. Mostramos un ejemplo de salida:

```
tortuga avanza 16 metros
liebre avanza 12 metros
tortuga avanza 17 metros
liebre avanza 13 metros
tortuga avanza 18 metros
liebre avanza 14 metros
tortuga avanza 19 metros
liebre avanza 15 metros
tortuga avanza 20 metros
liebre avanza 16 metros
tortuga avanza 21 metros
liebre avanza 17 metros
tortuga avanza 22 metros
tortuga avanza 23 metros
liebre avanza 18 metros
tortuga avanza 24 metros
liebre avanza 19 metros
tortuga avanza 25 metros
liebre avanza 20 metros
tortuga avanza 26 metros
tortuga avanza 27 metros
liebre avanza 21 metros
tortuga avanza 28 metros
liebre avanza 22 metros
tortuga avanza 29 metros
liebre avanza 23 metros
conejo avanza 1 metros
liebre avanza 24 metros
conejo avanza 2 metros
liebre avanza 25 metros
conejo avanza 3 metros
liebre avanza 26 metros
conejo avanza 4 metros
liebre avanza 27 metros
conejo avanza 5 metros
liebre avanza 28 metros
liebre avanza 29 metros
conejo avanza 6 metros
liebre llegó a meta
tortuga llegó a meta
conejo avanza 7 metros
conejo avanza 8 metros
conejo avanza 9 metros
conejo avanza 10 metros
conejo avanza 11 metros
conejo avanza 12 metros
conejo avanza 13 metros
conejo avanza 14 metros
conejo avanza 15 metros
conejo avanza 16 metros
conejo avanza 17 metros
conejo avanza 18 metros
conejo avanza 19 metros
conejo avanza 20 metros
conejo avanza 21 metros
conejo avanza 22 metros
conejo avanza 23 metros
conejo avanza 24 metros
conejo avanza 25 metros
conejo avanza 26 metros
conejo avanza 27 metros
conejo avanza 28 metros
conejo avanza 29 metros
conejo llegó a meta
```