**Кейс 1. Банковский перевод**

При выполнении задачи нельзя использовать какие-либо сторонние библиотеки и фреймворки либо стоит использовать их по минимуму, объяснив необходимость. По возможности реализуйте методы только на Core Java любой версии.

Есть банк, представленный классом Bank, в котором есть счета (класс Account) с двумя полями:

**public** **class** Account {

…

**private** long money;

**private** String accNumber;

…

}

Все счета хранятся внутри банка в какой-то Java-коллекции (можете выбрать на свой вкус). Клиенты банка (их много) могут делать переводы между счетами, запрашивать баланс по своему счёту. Всё это происходит в highly concurrent (многопоточной) среде.

При этом транзакции на крупные суммы (> 50 000) отправляются на проверку службе безопасности. Можно считать, что таких транзакций не более 5% от всех.

За эту проверку отвечает отдельный и уже реализованный кем-то метод Bank.isFraud. Служба безопасности в банке всего одна, работает медленно и не может обрабатывать более одной транзакции одновременно. Проверка транзакции занимает у неё 1 000 мс.

Если в ходе проверки служба безопасности выявила мошенничество, то необходимо заблокировать оба счёта, то есть запретить любые изменения остатков.

**public** **class** Bank {

…

**private** **final** Random random = **new** Random();

**public synchronized boolean** isFraud(String fromAccountNum, String toAccountNum, **long** amount) **throws** InterruptedException {

Thread**.**sleep(1000);

**return** random**.**nextBoolean();

}

…

}

**Реализуйте:**

1. **Метод transfer класса Bank**, который переводит деньги с одного счёта на другой:

**public** **void** transfer(String fromAccountNum, String toAccountNum, **long** amount)

Если сумма транзакции больше 50 000, то после совершения транзакции она отправляется на проверку службе безопасности — вызывается метод isFraud. Если возвращается true, то счета блокируются (как — на ваше усмотрение).

1. **Метод getBalance класса Bank**, который возвращает остаток на счёте:

**public** **long** getBalance(String accountNum)

Классы Account и Bank можно дорабатывать для решения задачи как угодно.