

## № 7 Многопоточные приложения

### Задание

1. Разработать многопоточное консольное приложение. Можно использовать `synchronized`.

**Ремонт дороги.** В Минске на улице Белорусской идет ремонт дороги и работает только одна полоса. В обоих направлениях собралось много машин. Обеспечьте безопасное движение машин (информируйте о прибытии машины и направлении движения, ожидании, проезде и о количестве машин на каждой стороне). Машина ожидает проезда определенное время, после чего уезжает.

**Автостоянка.** Доступно  $n$  машиномест. На одном месте может находиться только один автомобиль. Если все места заняты, то автомобиль не станет ждать больше определенного времени и уедет на другую стоянку.

2. Используя возможности, предоставляемые пакетом `java.util.concurrent` (не использовать `synchronized`) вспомогательные классы управления потоками

**Call center.** В организации работает  $n$  операторов. Оператор может обслуживать только одного клиента, остальные должны ждать своей очереди. Клиент может ждать, положить трубку, если превышено время ожидания и перезвонить еще раз через некоторое время. Промоделировать работу для  $m$  клиентов ( $m > n$ ).

**Библиотека.** В библиотеке доступны для чтения  $n$  книг. Одинаковых книг в библиотеке нет. Некоторые выдаются на руки, некоторые только в читальный зал. Читатель может брать на руки и в читальный зал несколько книг. Если книга на руках – читатель уходит. Если в читальном зале – ждет некоторый промежуток времени, после чего уходит.

Все объекты, желающие получить доступ к ресурсу, должны быть отдельными потоками. Написать демонстрацию. Все выполняемые действия и результаты должны быть выведены в лог (консоль, можно файл).

### Вопросы:

1. Каким образом можно создать поток?
2. Что такое монитор?
3. Как работают методы `wait` и `notify/notifyAll`?
4. Чем отличается работа метода `wait` с параметром и без параметра?
5. Как работает метод `Thread.yield()`? Чем отличаются методы `Thread.sleep()` и `Thread.yield()`?
6. Как работает метод `Thread.join()`?
7. Что такое `dead lock`?
8. Что значит приоритет потока?
9. Что такое потоки - демоны в джава?

10. Что значит усыпить поток?

11. В каких состояниях может быть поток в джава? Как вообще работает поток?

12. Различия между CyclicBarrier и CountdownLatch?

13. Как работает Semaphore?

1. Разработать многопоточное консольное приложение.

Можно использовать возможности, предоставляемые пакетом `java.util.concurrent`.

Не использовать слово `synchronized`.

Использовать типы на основе ООП. Все объекты, желающие получить доступ к ресурсу, должны быть отдельными потоками.