

Домашнее задание №5. Многопоточность

- Deadline: 18.12.2020, 23:59.
- В задаче нужно реализовать:
 - unit-тесты, а также JavaDoc для нетривиальных методов и классов,
 - сборку кода в JAR-файл с помощью Maven.
 - автоматический запуск сборки и тестирования после push в master-ветку (используйте GitLab CI для этого). Jar-файл должен быть доступен в качестве артефакта по завершении pipeline GitLab CI.

Для корректного запуска CI-job выбирайте Runner с тегом docker-atp. Подробнее про теги в GitLab-CI можно прочитать [здесь](#).

Задача. Корабли

Есть транспортные корабли, которые плывут от “генератора” к причалам для погрузки разного рода товаров. Они проходят через узкий туннель где одновременно могут находиться только 5 кораблей. Прохождение через туннель занимает некоторое время (пусть 1 секунда для каждого корабля. Так как одновременно в тоннеле могут находиться 5 кораблей, это означает, что при одновременном выпуске с генератора 5 кораблей, они пройдут через тоннель за одну секунду).

Каждый корабль представляет из себя отдельный поток java (Thread). Соответственно генератор кораблей генерирует объекты, которые реализуют интерфейс Runnable, это можно делать через ExecutorPool.

Существует 3 Типа кораблей (с хлебом, с бананами и с одеждой) и три вида вместительности 10, 50, 100 шт. товаров. 3 типа кораблей * 3 вида вместительности = 9 разных видов кораблей. Далее есть 3 вида причалов для погрузки кораблей — Хлеб, Банан и Одежда. Каждый причал берет или подзывает к себе необходимый ему корабль и начинает его загружать. За одну секунду причал загружает на корабль 10 ед. товара. То есть если у корабля вместительность 50 шт., то причал загрузит его за 5 секунд своей работы. Причал может загружать только по одному кораблю.

Требования к задаче:

- Правильно разбить задачу на параллельность.
- Синхронизировать потоки, сохранить целостность данных. Ведь ограничить доступ потоков к общему ресурсу дело не сложное, а заставить их работать согласованно уже намного сложнее. Проверку синхронизации потоков можно сделать инструментами логирования (по желанию можно использовать класс Logger из java.util, либо обычный system.out)
- Работа генератора кораблей не должна зависеть от работы причалов и наоборот.
- Общий ресурс должен быть Thread Safe
- Потоки не должны быть активными если нет задач.
- Потоки не должны держать mutex если нет задач.