

Задание 3  
**Отказоустойчивость, выбор лидера**  
крайний срок - 29 ноября

Модификация задания 2 — в системе появляется отказоустойчивость.

**1) Архитектура системы**

Два клиента, *федерация координаторов*, узел — симулятор коммуникационной среды,  $k$  узлов-реплик. Число  $k$  — параметр. У каждой реплики — один и тот же код.

**2) Отказоустойчивость**

В каждый момент времени в федерации координаторов (не менее 4х процессов с одинаковым кодом) один является главным. Через него идёт работа с запросами клиентов. Остальные отслеживают его состояние, в случае отказа — организуют выборы лидера с помощью кольцевого алгоритма Чанга-Робертса.

**3) Работа системы**

На мастере и репликах хранятся переменные  $x$ ,  $y$ ,  $z$ .

Клиент командами вида 'add  $x$  10', 'mul  $x$  3' может модифицировать переменные. Команды клиент отправляет мастеру.

Мастер, получив команду, модифицирует свою копию соответствующей переменной, присваивает номер обновлению, передаёт его узлу-симулятору.

Симулятор моделирует работу неидеальной коммуникационной среды — накапливает обновления, пока их не станет  $N$  ( $N$  - параметр), затем рассылает репликам в разном порядке (порядок для каждой реплики выстраивается путём случайного перемешивания).

Каждая реплика при обновлении своей копии переменной ориентируется на номера обновлений. Например, если уже применено обновление 4, а вновь пришедшее обновление имеет номер 7 — оно откладывается, пока не придут обновления 5 и 6.

Каждая реплика после очередного обновления своих копий переменных распечатывает значения всех переменных.

**4) Коммуникации**

RabbitMQ