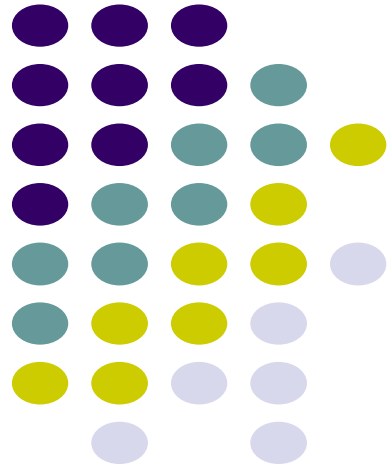
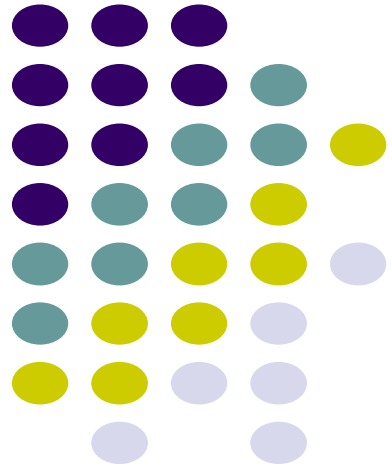
Course
Parallel Programming and Distributed Systems in Java

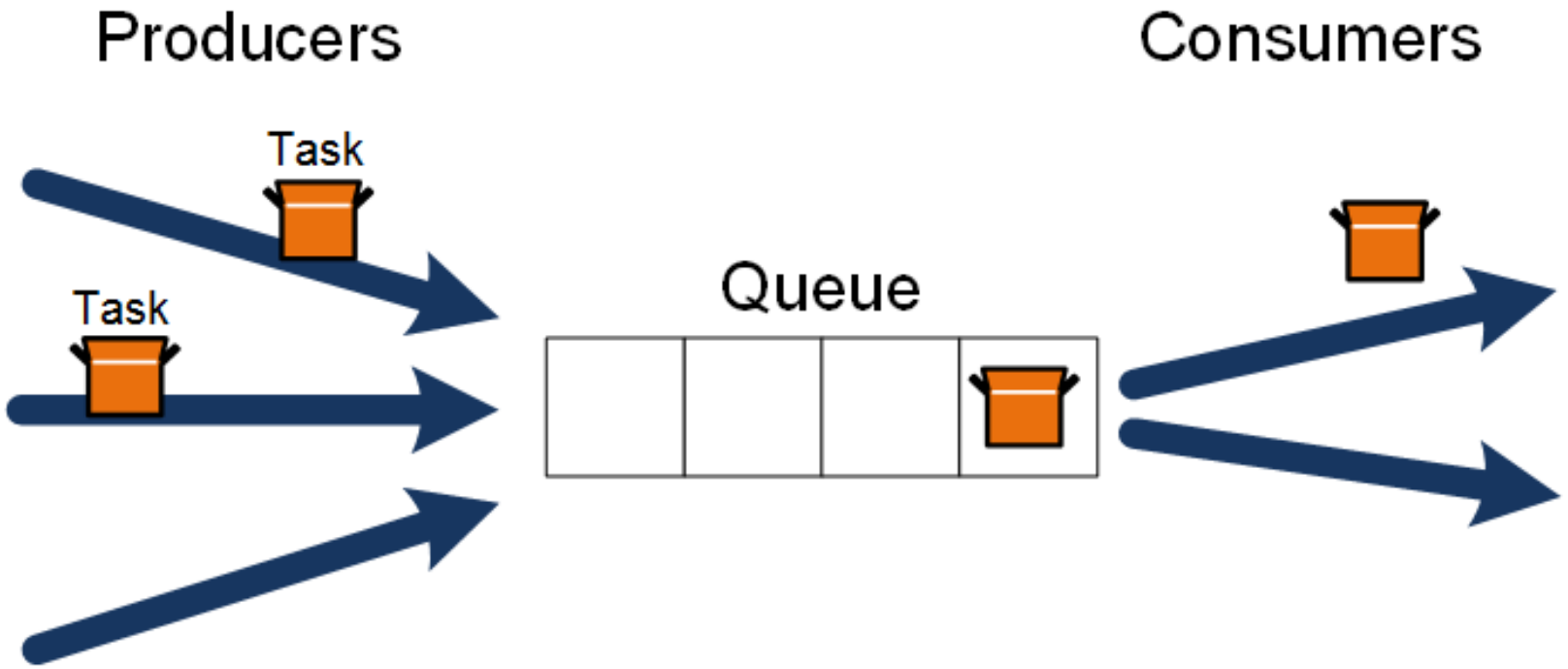


Лекция 3.1

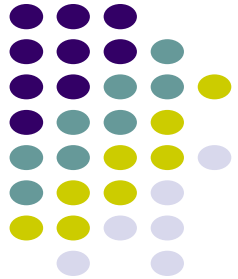
Взаимодействие потоков Producer Consumer



Модель Producer / Consumer

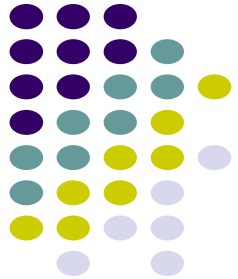


Эффективная обработка задач



- Количество потоков?
- Размер и структура очереди
- Когда задач слишком много

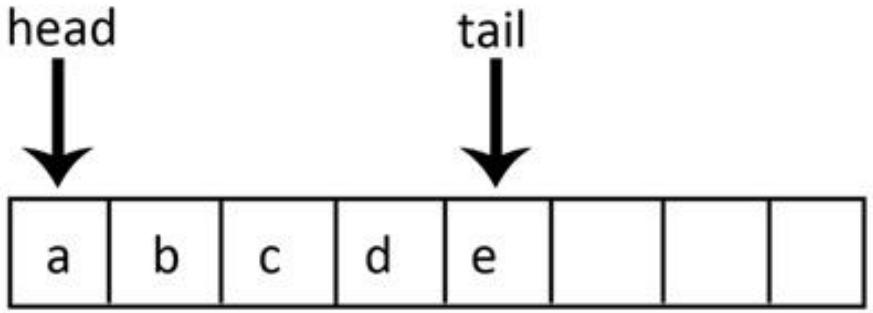
Количество потоков



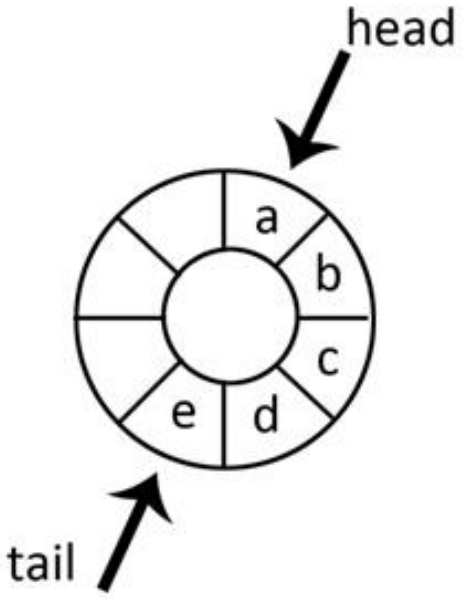
- Один поток – это не всегда мало
- От чего зависит оптимальное количество потоков?

Размер и структура буфера

- Очередь

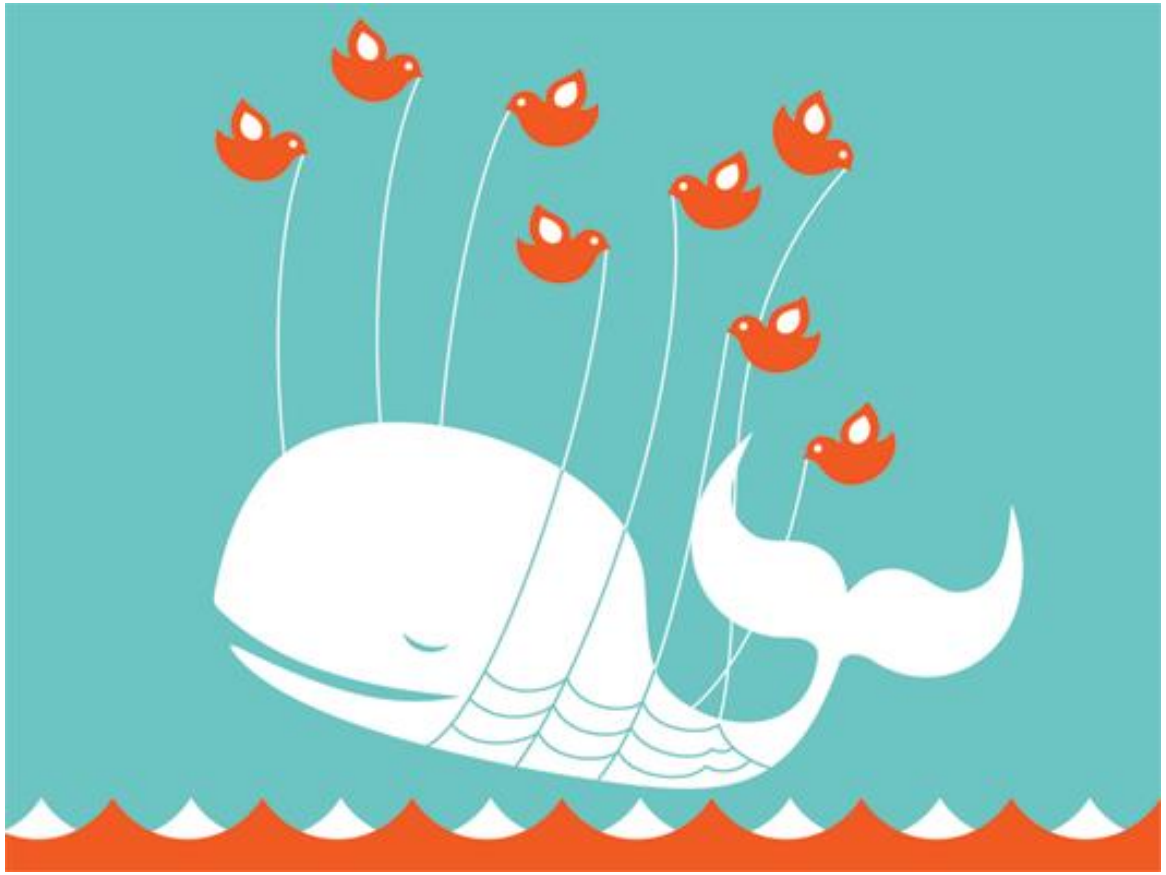


Кольцо

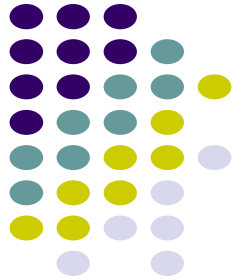


Переполнение

- Не обрабатывать новые
- Не обрабатывать старые
- Блокировать Producer’a
- Заставить поток Producer’a исполнить задачу

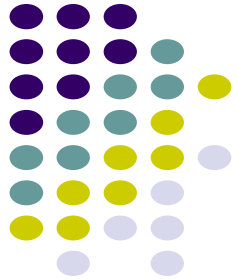


Примеры задач



- Перевод денег со счёта на счёт
- Обновление графика курсов валют
- Обработка списка изображений
- Получение картинки от веб-сервера

Meanwhile in Java world



```
ExecutorService executor = new ThreadPoolExecutor(
      5,  // core pool size
      10, // max pool size
    10000L, TimeUnit.MILLISECONDS, // keep alive
    new LinkedBlockingQueue<Runnable>(1000),    // queue
    new ThreadPoolExecutor.CallerRunsPolicy()); // handler
```