

## Задание 12

### Условие

Основная цель переработать Задание 11 так, чтобы ввод данных осуществлялся по сети.

Необходимо разработать асинхронный сервер, работающий с протоколом аналогичным консоли, принимающий построчно команды. Дальнейшая обработка остается без изменений - вывод на консоль и в файлы.

В качестве библиотеки сетевого взаимодействия можно использовать `boost.asio` (можно использовать любую предоставляющую асинхронные возможности).

Поскольку не предполагается отправка данных основное взаимодействие будет вокруг методов `async_read` и `async_acept`.

Порядок запуска:

```
# bulk_server <port> <bulk_size>
```

где

- *port* – номер tcp порта для входящих соединений. Соединения должны обслуживаться с произвольных интерфейсов.
- *bulk\_size* – уже знакомый размер блока команд.

Опционально взять за основу вместо однопоточной версии из Задания 7 многопоточную реализацию из Задания 10.

Не должно быть искусственных ограничений на количество одновременно подключенных клиентов. Команды полученные из разных соединений должны смешиваться между собой, команды окружённые { и } обрабатываются без смешивания.

Запуск команд:

```
# bulk_server 9000 3
```

и

```
# seq 0 9 | nc localhost 9000
```

(что означает формирование 10 строк с числами от 0 до 9 на каждой строке и отправкой их по сети на локальный порт 9000) должны привести к появлению в консоли:

```
bulk: 0, 1, 2
bulk: 3, 4, 5
bulk: 6, 7, 8
bulk: 9
```

и формированию четырех файлов с содержимым аналогичным каждой строке.

Одновременный запуск

```
# seq 0 9 | nc localhost 9000  
# seq 10 19 | nc localhost 9000
```

К появлению соответственно:

```
bulk: 0, 1, 10  
bulk: 2, 3, 11  
bulk: 4, 5, 12  
bulk: 6, 7, 13  
bulk: 8, 9, 14  
bulk: 15, 16, 17  
bulk: 18, 19
```

Как именно смешаются команды из разных источников зависит от условий времени исполнения и может отличаться от запуска к запуску.

## Требования к реализации

Результатом работы должен стать исполняемый файл `bulk_server` и находиться в пакете `bulk_server`.

Специальных усилий по демонизации сервера выполнять не следует, это значит, что сразу после запуска сервер не возвращает управление до принудительного завершения.

Результат работы должен быть опубликован на [bintray](#).

## Проверка

Задание считается выполненным успешно, если после установки пакета и запуска с тестовыми данными вывод соответствует описанию Задания 7.

Будет отмечена способность не терять команды полученные непосредственно перед закрытием соединения клиентом.