Задание повышенной сложности к модулю Multithreading

# Общее

Наша задача – разработка клиента и сервера для простого корпоративного чата. Основные характеристики нашего решения:

* Сервер и клиент реализованы как обычные консольные или GUI приложения (на ваш выбор)
* Взаимодействие между клиентами и сервером осуществляется посредствам Named Pipes (System.IO.Pipes) или Sockets (System.Net.Sockets) – также на ваш выбор. Для простоты настройки можно хранить все параметры подключения в коде.
* Клиент представляет собой бот, который после запуска выполняет циклически:
  + Подключается с новым именем к серверу
  + Отправляет несколько сообщений серверу (сообщения выбираются случайно из готового списка, количество отправляемых сообщений и паузы между ними также задаются случайно)
  + Принимает все сообщения от сервера, которые выдают на экран и/или сохраняет в файл
  + Отключается от сервера

Цикл повторяется до тех пор, пока пользователь не завершит работу клиента или не возникнет ошибка работы с сервером

* Сервер:
  + Принимает подключение от клиента. При подключении узнает имя подключенного клиента.
  + Принимает от клиентов строки сообщений и рассылает их остальным подключенным клиентам
  + Хранит историю из N последних сообщений, которые рассылает клиентам при первом подключении
  + При завершении приложения рассылает клиентам уведомление и корректно закрывает все подключения

# Задание 1

Реализуйте клиент и сервер используя для сервера схему «Для каждого клиента – свой поток обработки»

Чтение и запись можно делать синхронными операциями

# Задание 2

Перепишите клиент и сервер используя (на выбор или совместно):

* Классические асинхронные операции (BeginXXX/EndXXX) и пул потоков для операций, которые вы инициируете сами
* Task Parallel Library