ПЕРМСКИЙ ФИЛИАЛ ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО  
АВТОНОМНОГО ОБРАЗОВАТЕЛЬНОГО УЧРЕЖДЕНИЯ ВЫСШЕГО  
ОБРАЗОВАНИЯ

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Факультет социально-экономических и компьютерных наук

Берсенёв Илья Иванович

**Лабораторная работа №4**

Теория и алгоритмы высокопроизводительных вычислений

г. Пермь, 2025 год

**Задание**

1. Убедитесь, что у вас установлена служба очередей сообщений. Для этого в «Панели управления» выберите раздел «Программы и компоненты» выберите пункт «Включение и отключение компонентов Windows». В открывшемся дереве найдите вершину «Сервер очереди сообщений Майкрософт». Если данный компонент не установлен, то доустановите его.

2. Для просмотра имеющихся на компьютере очередей сообщений в «Панели управления» перейдите в пункт «Администрирование», выберите ярлык «Управление компьютером». В открывшемся окне (в дереве навигации) щелкните по вершине «Службы и приложения» и выберите пункт «Очередь сообщений». Изучите какие очереди сообщений уже имеются на вашей машине.

3. Изучите разработанное приложение. Запустите исполняемый файл сервера и клиента на одной машине, затем перенесите один из компонентов распределенного приложения на другой вычислительный узел (лучше на ноутбук, т.к. администратор ЛВС вуза может запретить обмен сообщения между различными узлами ЛВС) и попробуйте его запустить. Объясните, почему приложение перестало работать? Запустите несколько клиентов на одной машине с сервером и попробуйте отправить сообщения серверу. В процессе изучения (тестирования) приложения просмотрите очереди сообщений компьютера и убедитесь, что сообщения действительно приходят в очередь.

4. Запустите несколько серверов на одной машине. Отправьте им сообщения от нескольких клиентов. Объясните, почему приложение перестало корректно работать.

5. Модифицируйте приложение так, чтобы существовала возможность на сервере идентифицировать клиентов не по имени вычислительного узла, а по нику/логину пользователя.

6. Модифицируйте приложение так, чтобы получился полноценный чат. Клиент может отправлять сообщения всем клиентам, участвующим в беседе. Для этого каждый клиент должен иметь возможность просмотра всех сообщений от всех клиентов, а сервер должен содержать список клиентов, которые хотят участвовать в беседе, чтобы каждый раз выполнять им рассылку сообщений.

**Решение**

**Задание 3**

Приложение не работает т.к. по умолчанию создается private очередь, чей скоуп ограничен локальным компьютером.

**Задание 4**

При отправке сообщений с одного клиента на 2 сервера сообщение отправлялось попеременно на оба сервера. Судя по всему чтение из очереди блокирующее, а потому шедулится ОС.

**Задания 5 - 6**

Интерфейс и логика аналогичны лабораторной работе с mailslots (https://github.com/ressiwage/LEARN-HSE-HPCTA-2-3), за исключением АПИ для отправки и получения сообщений. Неожиданностью стала простота работы с MSMQ, Однако недостатком является отсутсвие event-driven подхода (невозможно повесить listener на event получения сообщения в очереди, приходится в цикле запрашивать данные). Почти вся логика от работы с pipes после небольших доработок хорошо сочетается с очередями.

Для приема сообщений клиент создает свою очередь формата DNS\_HOSTNAME\private$\NICKNAME, затем начинает в цикле прослушивать новые сообщения. Сервер же слушает сообщения от клиентов, регистрирует клиентов и отправляет всем зарегистрированным клиентам сообщение.

Код доступен по адресу <https://github.com/ressiwage/LEARN-HSE-HPCTA-4-1>

Демонстрация работы доступна по адресу <https://disk.yandex.ru/d/JunK6I6JRWJ4GQ>

