



# Отчет по лабораторной работе №567 по курсу

## Операционные системы

Студент группы: М8О-201Б-22, Парфенов Михаил Максимович, № по списку: 21, Контакты

mishaslsk@gmail.com Работа выполнена: «    » \_\_\_\_\_ 202\_\_ г.

Преподаватель: Миронов Евгений, Входной контроль знаний с

оценкой \_\_\_\_\_

Отчет сдан «    » \_\_\_\_\_ 202\_\_ г., итоговая оценка \_\_\_\_\_

Подпись преподавателя \_\_\_\_\_

1. **Тема:** Очередь сообщений
2. **Цель работы:** : приобретение практических навыков в управлении серверами сообщений, применении отложенных вычислений, интеграции программных систем друг с другом
3. **Задание (вариант № 22):** : Реализовать распределенную систему по асинхронной обработке запросов. В данной распределенной системе должно существовать 2 вида узлов: «управляющий» и «вычислительный». Необходимо объединить данные узлы в соответствии с той топологией, которая определена вариантом. Связь между узлами необходимо осуществить при помощи технологии очередей сообщений. Также в данной системе необходимо предусмотреть проверку доступности узлов в соответствии с вариантом. При убийстве («kill -9») любого вычислительного узла система должна пытаться максимально сохранять свою работоспособность, а именно все дочерние узлы убитого узла могут стать недоступными, но родительские узлы должны сохранить свою работоспособность.
4. **Оборудование**  
*Оборудование ПЭВМ студента, если использовалось:*  
Процессор AMD Ryzen 5 5500U с ОП 8192 Мб, ТТН 256 GB. Мониторы Dell Monitor, 1920 x 1080, 14 дюймов.
5. **Программное обеспечение (лабораторное):**  
*Программное обеспечение ЭВМ студента, если использовалось:*  
Операционная система семейства GNU/Linux, наименование Arch Linux версия 5.19.7 интерпретатор команд zsh версия 5.8.  
Система программирования \_версия \_  
Редактор текстов \_ версия \_  
Утилиты операционной системы \_  
Прикладные системы и программы \_  
Местонахождение и имена файлов программ и данных на домашнем компьютере \_
6. **Идея, метод, алгоритм** решения задачи (в формах: словесной, псевдокода, графической [блок-схема, диаграмма, рисунок, таблица] или формальные спецификации с пред- и постусловиями)

Управляющий узел получает на вход команду со стандартного входа, обрабатывает входные данные и заворачивает их в структуру данных вида "номер команды, номер вычислительного узла, строка, подстрока", и отправляет на все свои дочерние вычислительные узлы. Каждый узел является subscriber относительно своего родительского узла. Вычислительный узел обрабатывает команду, если номер вычислительного узла совпал с личным номером, в противном случае отправляет дальше по иерархии.

7. **Сценарий выполнения работы** [план работы, первоначальный текст программы в черновике (можно на отдельном листе) и тесты либо соображения по тестированию]. *Пункты 1-7 отчета составляются строго до начала лабораторной работы.*
- 1) Изучить библиотеку ZeroMQ для работы с очередями сообщений
  - 2) Реализовать главный управляющий узел, который принимает команды со стандартного ввода и отправляет их на вычислительные узлы
  - 3) Реализовать вычислительный узел, который принимает команды от кпрауляющего узла либо родительского вычислительного узла, перенаправляет команды на дочерние узлы и выполняет логику программы
  - 4) Реализовать алгоритм поиска подстроки в строке (в данном случае алгоритм Кнутта-Морриса-Пратта с помощью префикс-функции)
  - 5) Протестировать работу программы
8. **Вывод:** По ZeroMQ нашел хорошую статью на habr, что позволило быстро понять, что нужно в сделать в работе. Но тем не менее код получился очень объемный. Тема асинхронных систем мне понравилась, так как из нее вытекает создание клиент-серверных приложений.