| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |
| --- | --- |

ФАКУЛЬТЕТ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

КАФЕДРА \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ОТЧЕТ**

**Отчет по лабораторной работе №3 по курсу «Компьютерные сети»**

Вариант \_\_\_\_\_16\_\_\_\_\_\_

Студент ИУ7-73Б \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Сукочева А.

(Группа) (Подпись, дата) (И.О.Фамилия)

Преподаватель  \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Рогозин Н.О.

(Подпись, дата) (И.О.Фамилия)

2021 г.

**Часть I**

Разработать клиент-серверное приложение с использованием сетевых сокетов и протокола TCP для передачи бинарных файлов от сервера клиентам. Сервер должен позволять подключение нескольких клиентов. Имя файла задается клиентом при подключении в виде текстовой строки. Сервер должен отобразить, используя пользовательский интерфейс, названия всех файлов в папке с исполняемым файлом сервера, предоставив клиенту выбор.

**Часть II**

Разработать http сервер для обработки GET-запросов и предоставления статической информации клиенту. Разработать http-клиент для проверки данного сервера с помощью GET-запросов. Использовать системные сокеты и транспортный протокол TCP. Язык С/С++. Возможно использование ранее разработанного TCP-сервера в качестве основы.

Использование фреймворков не допускается.

1)Наименьшее количество баллов выставляется за сервер с последовательной обработкой запросов, без выполнения доп. задачи

2)Наибольшее количество баллов выставляется за многопоточный сервер с обработкой соединений через использование динамически распределяемого пула потоков + выполнение доп. задачи

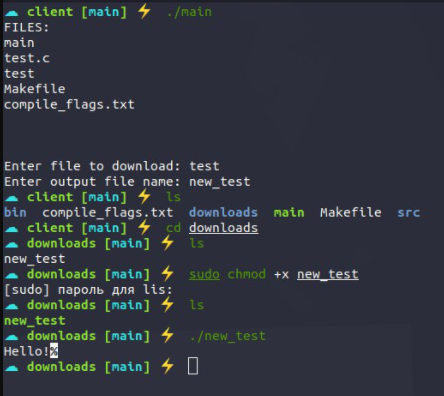
Вариант доп. задачи:

Сохранять статистику о различных форматах файлов, к которым обращался тот или иной пользователь.

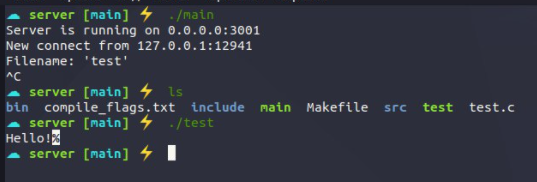
**Часть 1**

Сервер работает следующим образом. При отправке строки ls возвращает содержимое директории с файлами. При отправки другой строки пытается найти файл с таким именем и если находит – отправляет файл и разрывает соединение.

Пример работы программы:

Клиент:  


Сервер:



Клиент указывает название желаемого файла и название файла, в который запишется содержимое. Как видно на приведенных рисунках, бинарный файл успешно передался и запустился.

**Часть 2**

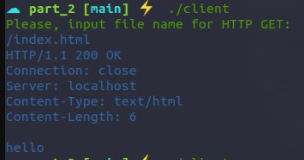
Весь код с поясняющими комментариями приведен во вложениях.

Пример работы программы:

Содержимое папки. К этим файлам у клиента есть доступ.  

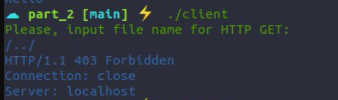

Запускаем клиента. Будем выполнять GET-запросы.

Вводим название желаемого файла (index.html) и получаем следующий ответ.

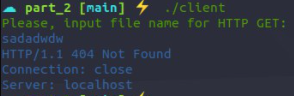


Пытаемся получить доступ к файлу, который находится в папке выше.

Доступ к запрошенному ресурсу запрещен.



Пытаемся получить доступ к несуществующему файлу.



Файл со статистикой.

Статистика представлена в следующем виде.

Формат файла (расширение).

Далее идет информации о пользователе и времени, когда он обращался к данному файлу.

