



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 ИУ «Информатика и системы управления»

КАФЕДРА ИУ-7 «Программное обеспечение ЭВМ и информационные технологии»

РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
К КУРСОВОЙ РАБОТЕ
НА ТЕМУ:

«Разработка статического сервера»

Студент ИУ7-7Б _____ Ковель А.Д.

Руководитель курсовой работы _____

2023 г.

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)

УТВЕРЖДАЮ

Заведующий кафедрой ИУ-7

И. В. Рудаков

«16» сентября 2023 г.

ЗАДАНИЕ
на выполнение курсовой работы

по теме

«Разработка статического сервера»

Студент группы **ИУ7-76Б**

Ковель Александр Денисович

Направленность КР

учебная

Источник тематики

Курсовая работа кафедры

График выполнения КР: 25% к 6 нед., 50% к 9 нед., 75% к 12 нед., 100% к 15 нед.

Техническое задание

Разработать статический веб-сервер для отдачи контента с диска. В качестве мультиплексора использовать poll. Сервер должен реализовывать многопоточную обработку запросов с использованием пула потоков.

Оформление научно-исследовательской работы:

Расчетно-пояснительная записка на **12-20** листах формата А4.

Дата выдачи задания «16» сентября 2023 г.

Руководитель курсовой работы

(Подпись, дата)

(Фамилия И. О.)

Студент

(Подпись, дата)

Ковель А. Д.
(Фамилия И. О.)

СОДЕРЖАНИЕ

| | |
|---|-----------|
| ВВЕДЕНИЕ | 4 |
| 1 Аналитические раздел | 5 |
| 1.1 Thread pool | 5 |
| 1.2 Сокет poll | 5 |
| 2 Конструкторская часть | 8 |
| 2.1 Разработка алгоритмов | 8 |
| 3 Технологическая часть | 11 |
| 3.1 Средства реализации | 11 |
| 4 Исследовательская часть | 12 |
| 4.1 Технические характеристики | 12 |
| 4.2 Демонстрация работы программы | 12 |
| 4.3 Вывод | 12 |
| ЗАКЛЮЧЕНИЕ | 13 |
| СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ | 14 |

ВВЕДЕНИЕ

Цель работы — разработка написать статический сервер для отдачи файлов с диска на основе пула потоков и сокета poll.

Для достижения поставленной цели, необходимо решить следующие задачи:

- 1) формализовать задачу;
- 2) определить структуры, связанные с поставленной задачей;
- 3) разработать алгоритм poll сокета;
- 4) реализовать программное обеспечение;
- 5) провести исследование скорости работы статического сервера.

1 Аналитические раздел

В данном разделе рассмотрены технологии thread pool и сокет poll.

1.1 Thread pool

Параллельные вычисления — это тип вычислений, при котором одновременно выполняется множество операций или процессов. Пул потоков (Thread pool) [1] — это фиксированный набор потоков, одновременно выполняющих независимые друг от друга задачи, помещенные в массив. Массив задач представляется в виде очереди. Ключевым аспектом логики пула потоков является факт того, что все потоки запускаются один раз.

Число операций, которые можно поставить в очередь в пуле потоков, ограничено только доступной памятью. Однако пул потоков имеет ограничение на число потоков, которые можно активировать в процессе одновременно. Если все потоки в пуле заняты, дополнительные рабочие элементы помещаются в очередь и ожидают их освобождения.

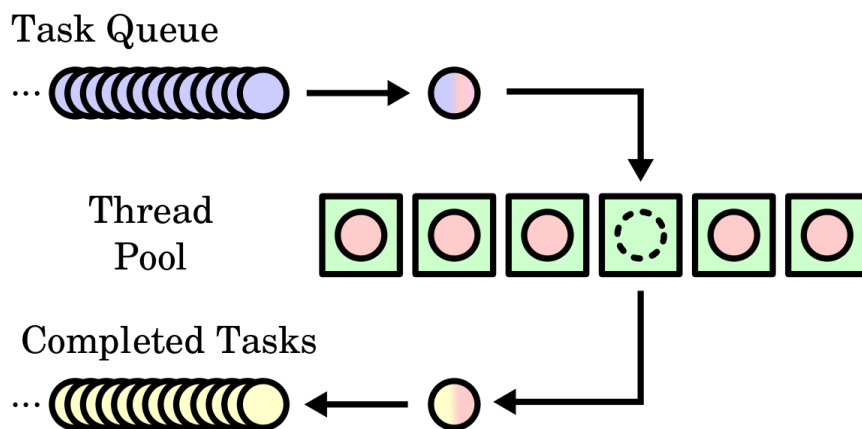


Рисунок 1 – Thread pool

1.2 Сокет poll

Сокеты — название программного интерфейса для обеспечения обмена данными между процессами. Процессы при таком обмене могут исполняться как на одной ЭВМ, так и на различных, связанных между собой сетью. Сокет — это абстрактный объект, представляющий конечную точку соединения.

Каждый сокет имеет свой адрес. ОС семейства UNIX могут поддерживать много типов адресов, но обязательным являются INET-адрес и UNIX-адрес.

Poll — это метод опроса сокетов, созданный после того как ресурс select оказался недостаточен. Преимуществами poll сокетов являются:

- 1) нет никакого лимита количества наблюдаемых дескрипторов;
- 2) не модифицируется структура pollfd, что дает возможность ее переиспользования между вызовами poll();
- 3) для идентификации отключения клиента, нет необходимости чтения данных из сокета.

Сокеты poll используют блокировку ввода-вывода с мультиплексированием, алгоритм работы представлен на рисунке 2.

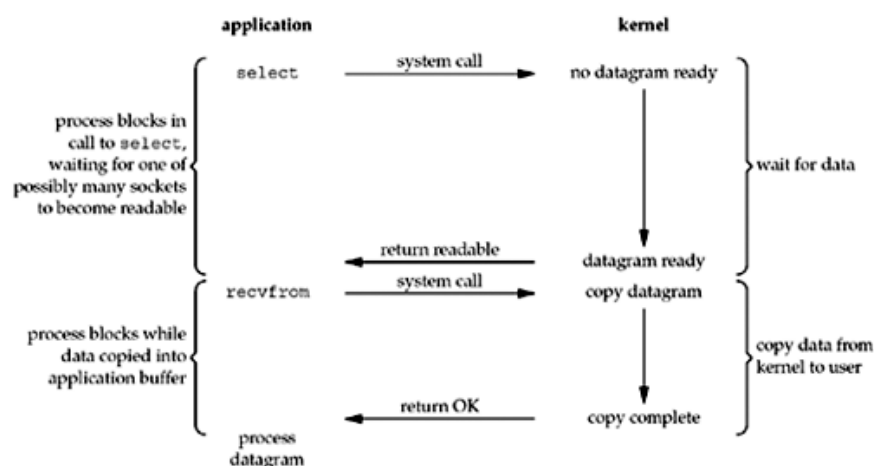


Рисунок 2 – Ввод-вывод с мультиплексированием

Основной функцией сокетов является:

Листинг 1 – Функция poll

```
#include <poll.h>

int poll (struct pollfd *fdarray, unsigned long nfds, int timeout);

/* Returns: count of ready descriptors, 0 on timeout, -1 on error */
```

Первым аргументом этой функции является структура pollfd:

Листинг 2 – Структура pollfd

```
struct pollfd {
    int      fd;          /* descriptor to check */
    short    events;      /* events of interest on fd */
    short    revents;     /* events that occurred on fd */
};
```

Проверяемые условия задаются членом `events`, а функция возвращает статус дескриптора в соответствующим члене `revents`. Такая структура данных (с двумя переменными на дескриптор, одна из которых является значением, а другая результатом) позволяет избежать аргументов «значение-результат». Каждые из этих двух членов состоит из одного или нескольких битов, которые задают определенное условие.

Вывод

В данном разделе дано определение пула потоков и сокета `poll`. Также был представлен алгоритм их работы.

2 Конструкторская часть

В данном разделе представлены схемы алгоритмов сокетов poll и thread poll.

2.1 Разработка алгоритмов

На рисунке представлена схемка алгоритма сокетов poll 3.

На рисунке 4 представлена схема алгоритма thread poll.

Вывод

В данном разделе разработаны и представлены схемы алгоритмов сокетов poll и пул потоков.

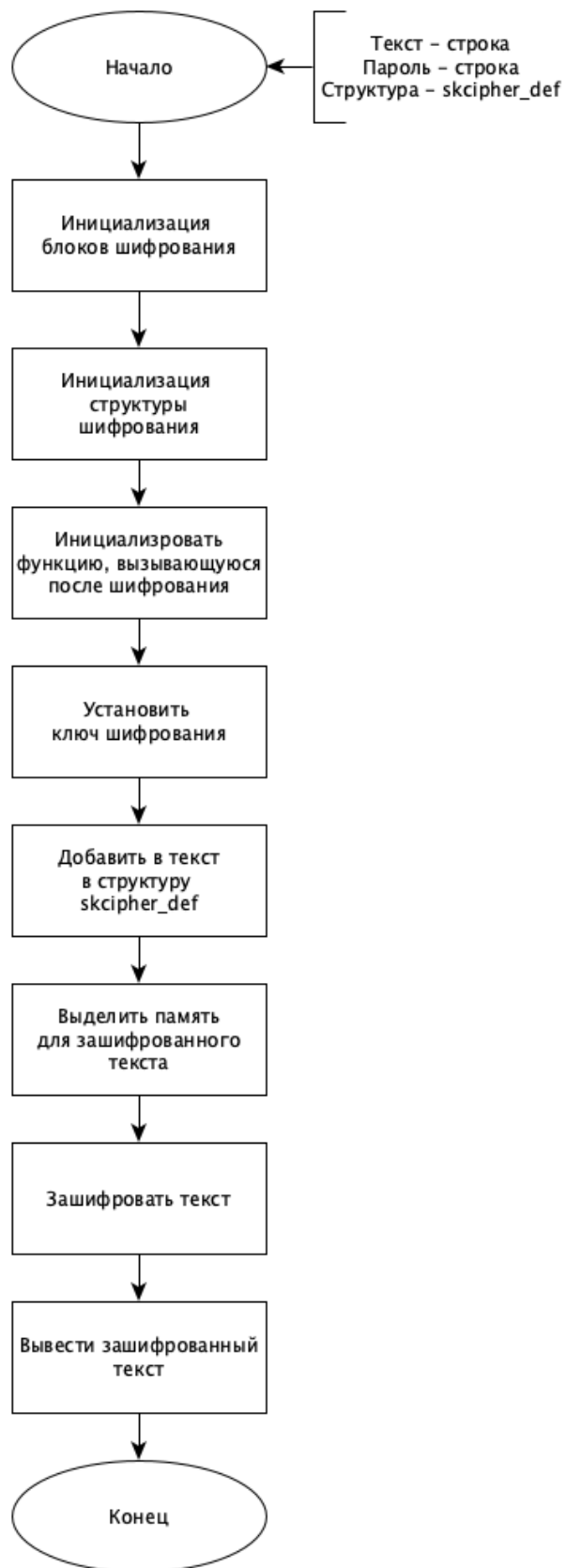


Рисунок 3 – Сокет poll

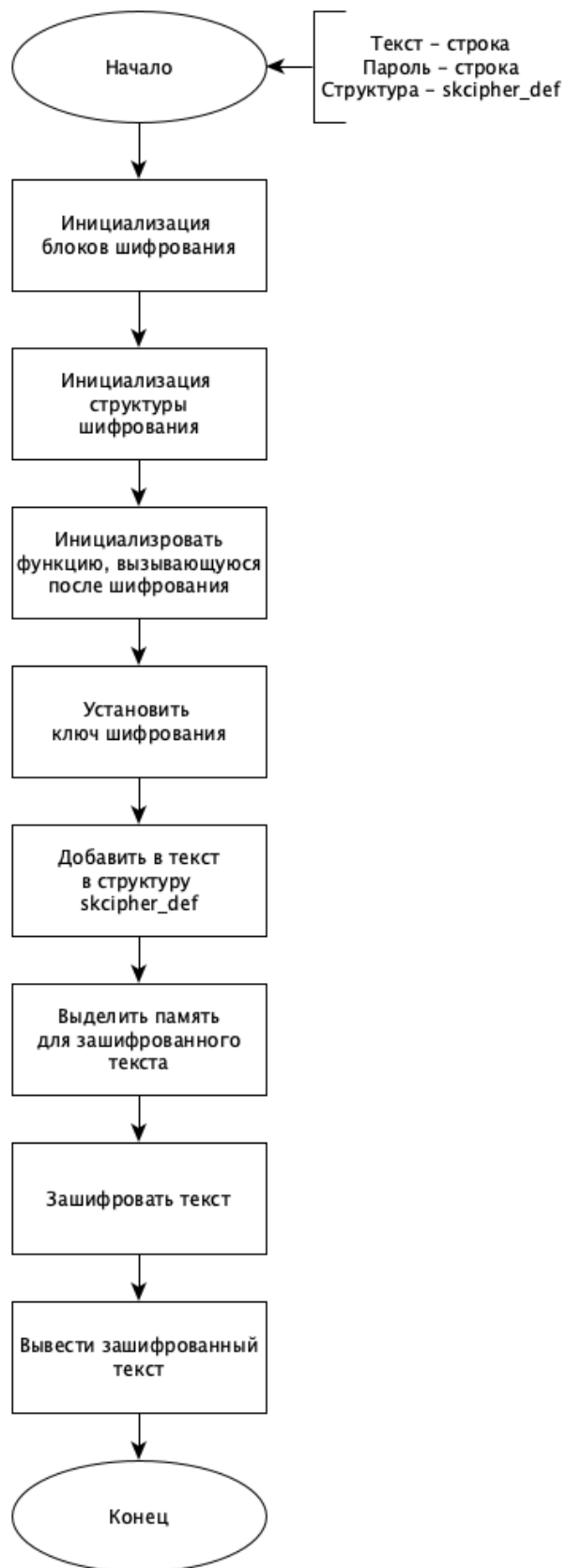


Рисунок 4 – Пул потоков

3 Технологическая часть

3.1 Средства реализации

Для реализации ПО был выбран язык С [2]. В данном языке есть все требующиеся инструменты для данной курсовой работы. В качестве среды разработки была выбрана среда VS code [3].

4 Исследовательская часть

4.1 Технические характеристики

Тестирование выполнялось на устройстве со следующими техническими характеристиками:

- Операционная система Pop!_OS 22.04 LTS [4] Linux [5];
- Оперативная память 32 Гбайт;
- Процессор AMD® Ryzen 7 2700 eight-core processor × 16 [6].

4.2 Демонстрация работы программы

4.3 Вывод

В данном разделе был приведен анализ работы статического сервера.

ЗАКЛЮЧЕНИЕ

В ходе выполнения данной работы были выполнены следующие задачи:

- 1) формализована задачу;
- 2) определены структуры, связанные с поставленной задачей;
- 3) разработан алгоритм poll сокета;
- 4) реализовано программное обеспечение;
- 5) проведено исследование скорости работы статического сервера.

Поставленная цель достигнута.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Thread Pools [Электронный ресурс]. Режим доступа: <https://www.iitk.ac.in/esc101/05Aug/tutorial/essential/threads/pool.html#:~:text=A%20thread%20pool%20is%20a,executing%20a%20collection%20of%20tasks> (дата обращения: 03.12.2023).
2. Язык программирования C [Электронный ресурс]. Режим доступа: <https://learn.microsoft.com/ru-ru/cpp/c-language/?view=msvc-170>. дата обращения: 03.12.2023.
3. Vscode [Электронный ресурс]. <https://code.visualstudio.com/>. дата обращения: 03.12.2023.
4. Pop OS 22.04 LTS [Электронный ресурс]. Режим доступа: <https://pop.system76.com> (дата обращения: 03.12.2023).
5. Linux – Документация [Электронный ресурс]. Режим доступа: <https://docs.kernel.org> (дата обращения: 03.12.2023).
6. Процессор AMD® Ryzen 7 2700 eight-core processor × 16 [Электронный ресурс]. Режим доступа: <https://www.amd.com/en/products/cpu/amd-ryzen-7-2700> (дата обращения: 03.12.2023).