Практическая работа №5 (5 баллов)

**ЯП и технологии: C, OpenMP.**

**Задание:** Анализ логов веб-сервера с использованием OpenMP.

**Цель задания:**

Разработать параллельную программу на языке C с использованием OpenMP для анализа лог-файла веб-сервера.

Ваша задача – подсчитать количество HTTP-запросов различных типов (например, GET, POST, PUT и др.) в файле логов. Программа должна эффективно использовать многопоточность для ускорения обработки файлов логов.

**Описание:**

Лог-файл веб-сервера содержит записи о запросах клиентов. Каждая строка лога содержит информацию о запросе, включая IP-адрес клиента, дату и время запроса, метод HTTP (например, GET, POST), путь к ресурсу, код ответа.

**Пример строки лога:**

127.0.0.1 - - [08/Oct/2024:10:15:32] GET "/index.html HTTP/1.1" 200

192.168.1.1 - - [08/Oct/2024:10:16:10] POST "/submit HTTP/1.1" 200

**Подробная постановка задачи:**

1. На любом языке программирования написать скрипт для генерации логов.
2. Сгенерировать логи веб-сервера.
3. Написать программу, которая читает лог-файл веб-сервера.
4. Программа должна распознавать HTTP-методы (GET, POST, PUT, DELETE, HEAD).
5. Использовать OpenMP для распараллеливания процесса обработки логов.
6. Программа должна корректно синхронизировать доступ к глобальным данным. Для этого используйте синхронизацию (critical, lock или иные механизмы OpenMP) для предотвращения состояния гонки при обновлении глобальных переменных.
7. Программа должна корректно обрабатывать строки логов разной длины и формата.
8. Если метод запроса не распознается, он должен игнорироваться.
9. Вывод программы должен содержать количество запросов для каждого HTTP-метода

**Пример работы программы:**

Входной файл лога:

127.0.0.1 - - [08/Oct/2024:10:15:32] GET "/index.html HTTP/1.1" 200

192.168.1.1 - - [08/Oct/2024:10:16:10] POST "/submit HTTP/1.1" 200

127.0.0.1 - - [08/Oct/2024:10:16:45] GET "/images/logo.png HTTP/1.1" 200

**Ожидаемый вывод программы:**

Results:

GET: 2

POST: 1

PUT: 0

DELETE: 0

HEAD: 0

**Советы по выполнению**:

Для обработки строк лог-файла используйте функции стандартной библиотеки C, такие как fgets() для чтения строк и strstr() для поиска подстрок.

При параллельной обработке строки можно считывать в локальный буфер для каждого потока.

При обновлении глобальных переменных необходимо использовать механизмы синхронизации.