Национальный исследовательский университет «Высшая школа экономики»

Домашнее задание №5

Архитектура вычислительных систем

Пояснительная записка

Исполнитель

студент группы БПИ206

Андрющенко Михаил Андреевич

2021 г.

Задание

Вариант: 1

Задача о читателях и писателях. (С++/Go)

1. Читатели, читают базу данных. Если нет писателей, читатели могут читать бд параллельно.
2. Писатели, пишут в базу данных. Право владения исключительное.
3. Реализовать паттерн читатели и писатели, используя такой примитив синхронизации, как семафор

* g++ -f <input.txt> - ввод из файла

input file:

1. Первая строка: кол-во элементов в базе данных.
2. Далее идут строки базы данных.

* g++ -n <count\_row> - ввод с использованием рандомного генератора

1. Компиляция С++

* просьба использовать 11 и выше версию стандарта языка, а также (-phtread ), чтобы подключить pthreads библиотеку
* пример (g++): g++ -std=c++17 -pthread main.cpp

Семафор

Семафор - примитив синхронизации работы процессов и потоков, в основе которого лежит счётчик, над которым можно делать операции: увеличение и уменьшение значения на единицу, при этом операция уменьшения для нулевого значения счётчика представляется блокирующейся.

Используется для синхронизации параллельно функционирующих задач, для защиты передачи данных через разделяемую память, для защиты критических секций, а также для управления доступом к аппаратному обеспечению.

Чуть проще говоря, семафор помогает ограничить доступ потоков к объекту синхронизации по их количеству.

Первоначально семафор инициализируется и ему передается количество потоков, которые к нему могут обратиться. Дальше при любом обращении к ресурсу его счетчик уменьшается. Когда счетчик понизится до 0 к ресурсу обратиться уже нельзя. При отсоединении потока от семафора его счетчик увеличивается, что позволяет иным потокам обратиться к нему. Когда счетчик равен 0, семафор считается сброшенным.

Модель читатели и писатели.

Есть область памяти, допускающая чтение и запись. Несколько потоков имеют к ней доступ, при этом одновременно могут читать сколько угодно потоков, но писать — только один.

Важно:

В данной реализации преимуществом обладают читатели.

Характеристики программы

Интерфейсные модули: 0

Модули реализации: 1

Общий размер результатов тестов: 0 Кб (вывод в консоль)

Размер исполняемого кода: 4.71 Кб

Заключение

Разработали консольное приложение на языке С++ с использованием библиотеки POSIX Threads языка программирования C и на Go без использования библиотек (В Go многопоточность встроена). Разобрались в идее семафоров, а также в модели «читатели и писатели». Программу на Golang писать приятнее, быстрее и эффективнее.