

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №3
по дисциплине «Параллельные алгоритмы»
Тема: Реализация параллельной структуры данных с тонкой
блокировкой.

Студент гр. 9304

Преподаватель

Атаманов С.Д.

Сергеева Е.И.

Санкт-Петербург

2022

Цель работы.

Реализовать корректную работу потоков используя шаблон “производитель-потребитель”.

Задание.

Обеспечить структуру данных из лаб.2 как минимум тонкой блокировкой (*сделать lock-free). Протестировать доступ в случае нескольких потоков-производителей и потребителей. Сравнить производительность со структурой с грубой синхронизацией (т.е. с лаб.2).

В отчёте сформулировать инвариант структуры данных.

Выполнение работы.

Для выполнения работы была разработана lock-free очередь. Очередь содержит указатель на голову и хвост очереди. Указатели были реализованы с помощью атомарных переменных типа `std::atomic<Node*>`, где `Node` - узел для хранения матрицы и указателя на следующий элемент.

Для работы с очередью были написаны методы `Push` и `Pop`. Методы реализованы с помощью атомарных операций типа `CAS` (`Compare and Swap`). Оба метода работают в бесконечном `CAS` цикле. `Push` - добавляет элемент в очередь, `Pop` - возвращает из очереди данные и сдвигает очередь.

Исследование зависимости между количеством потоком, размерами входных данных и параметрами вычислительной системы.

Исследование для одного потребителя и производителя.

Таблица 1 - Сравнение размера входных данных и времени вычисления для одного потока при грубой синхронизации структуры:

Время вычисления(миллисек.)	Размер входных данных
228	1000 x 1000
5472	5000 x 5000
22648	10000 x 10000

Таблица 2 - Сравнение размера входных данных и времени вычисления для одного потока при тонкой синхронизации структуры:

Время вычисления(миллисек.)	Размер входных данных
252	1000 x 1000
6423	5000 x 5000
36654	10000 x 10000

Исследование для нескольких потребителей и производителей.

В таблице 3 представлено общее время работы программы в зависимости от кол-ва производителей и потребителей при грубой синхронизации структуры:

Один производитель и	Два производителя и один	Два производителя и два потребителя (миллисек.)	Размер генерируемых данных
---------------------------------	-------------------------------------	--	---

два потребителя (милисек.)	потребитель (милисек.)		
34212	27193	20743	6 * (5000*5000)
68287	63298	45634	12 * (5000*5000)

В таблице 4 представлено общее время работы программы в зависимости от кол-ва производителей и потребителей при тонкой синхронизации структуры:

Один производитель и два потребителя (милисек.)	Два производителя и один потребитель (милисек.)	Два производителя и два потребителя (милисек.)	Размер генерируемых данных
24643	15865	16010	6 * (5000*5000)
43494	25953	28425	12 * (5000*5000)

Из проведенных измерений можно сделать вывод, что решение задачи вышло не слишком эффективным.

Выводы.

В ходе выполнения лабораторной работы была реализована программа на языке программирования C++ для попарного сложения матриц, использующая в качестве структуры данных *lock-free* очередь.