

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

ОТЧЕТ
по лабораторной работе №2
по дисциплине «Параллельные алгоритмы»
Тема: Реализация потокобезопасных структур данных с
блокировками.

Студент гр. 0304

Алексеев Р.В.

Преподаватель

Сергеева Е.И.

Санкт-Петербург

2023

Цель работы.

Изучить работу с блокировками для создания потокобезопасных структур. Реализовать различные виды структур.

Задание.

Реализовать итерационное (потенциально бесконечное) выполнение подготовки, обработки и вывода данных по шаблону “производитель-потребитель” (на основе лаб. 1 (части 1.2.1 и 1.2.2)).

Обеспечить параллельное выполнение потоков обработки готовой порции данных, подготовки следующей порции данных и вывода предыдущих полученных результатов.

Использовать механизм “условных переменных”.

2.1

Использовать очередь с “грубой” блокировкой.

2.2

Использовать очередь с “тонкой” блокировкой

Выполнение работы.

1. Для реализации структур создан класс *Handler*, который включает в себя методы *push* и *pop* для вставки и удаления элементов очереди. Также реализован метод *handle*, который обрабатывает данные: производит умножение матриц и вывод результата.

В реализации грубой блокировки класс имеет один мьютекс, который блокирует всю очередь при выполнении операций вставки и удаления. Очередь реализована с помощью встроенной библиотеки *queue*.

В тонкой блокировке класс имеет два мьютекса: для вставки и для удаления. Очередь реализована при помощи односвязного списка, первый и последний узлы которого сохраняются в качестве полей класса.

В обоих случаях обработка данных и их вывод происходит после разблокировки мьютекса для экономии ресурсов.

Для реализации добавления производителей и потребителей созданы функции *pusher* и *popper* соответственно.

2. Были произведены замеры времени в зависимости от количества производителей и потребителей для тонкой и грубой блокировок. Во всех случаях количество задач — 10000. Результаты представлены в таблицах 1 и 2.

Таблица 1 — Время выполнения программы с грубой блокировкой.

Количество производителей/потребителей	Время, мс
2/2	142
10/2	178
2/10	236
50/50	264
1000/50	263
50/1000	465
1000/1000	395

Таблица 2 — Время выполнения программы с тонкой блокировкой.

Количество производителей/потребителей	Время, мс
2/2	170
10/2	174
2/10	211
50/50	258
1000/50	275

50/1000	385
1000/1000	393

По таблицам видно, что в ряде случаев реализация с тонкой блокировкой превосходит по времени реализацию с грубыми, но при малом количестве потоков, например 2/2, когда потоки меньше конкурируют за мьютекс, реализация с грубой блокировкой оказывается быстрее. При большем количестве потоков борьба за мьютекс возрастает, но она все ещё невелика, поэтому реализации с грубыми и тонкими блокировками работают примерно за равное время. При большом равном количестве потоков — 1000/1000, обе реализации работают примерно за одинаковое время. В случае, когда количество производителей больше — 1000/50, программам требуется также примерно одинаковое время, т. к. в данном случае большинство потоков конкурирует за мьютекс хвоста, который использует и при вставке, и при удалении. Когда количество потребителей больше — 50/1000, реализация с тонкими блокировками требует меньше времени, т. к. при использовании тонких блокировок потоком не надо ожидать разблокировку мьютекса хвоста для начала обработки данных.

Таким образом, реализация очереди с тонкими блокировками имеет преимущество по времени при большом общем количестве потоков и, если количество потребителей больше количества производителей. В противных случаях время требуемое реализациями с тонкими и грубыми блокировками примерно равное.

Выводы.

В ходе работы были изучены способы реализации потокобезопасных структур данных с блокировками.

Была реализована очередь с тонкими и грубыми блокировками.

Была исследована зависимость количества и отношения производителей и потребителей ко времени выполнения программы. Было установлено, что реализация с тонкими блокировками имеет преимущество по времени в случае, когда количество потоков достаточно велико и количество производителей меньше количества потребителей — 50/1000. В остальных случаях время примерно равно со временем работы реализации с грубыми блокировками.