**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МОЭВМ**

отчет

**по лабораторной работе №5**

**по дисциплине «Параллельные алгоритмы»**

Тема: Оптимизация доступа к памяти в модели OpenCL

|  |  |  |
| --- | --- | --- |
| Студент гр. 0303 |  | Парамонов В.В. |
| Преподаватель |  | Сергеева Е. И. |

Санкт-Петербург

2023

## Цель работы.

Реализовать умножение матриц в OpenCL с оптимизированным доступом к памяти (использованием локальной памяти). Сравнить полученную реализацию с умножением на CPU с использованием алгоритма Штрассена.

**Постановка задачи.**

1. Реализовать умножение матриц на OpenCL (с использованием локальной памяти).
2. Произвести оценку производительности с реализацией на CPU с использованием алгоритма Штрассена.

**Выполнение задач.**

1. **Реализация программы:**
2. В отличие от кода лабораторной 5 в opencv\_handler.cpp и opencv\_handler.hpp был выделен отдельный класс OpenCVHandler, занимающейся всей работой с библиотекой OpenCL. Так же был изменен код ядра kernel на умножение матриц.
3. В отличие от кода лабораторной 4 в реализацию умножения матриц на CPU с использованием алгоритма Штрассена был добавлен пул потоков, управляемый библиотекой TBB, чтобы увеличить шансы данной реализации.
4. **Оценка производительности:**

* Зависимость времени выполнения от размера перемножаемых квадратных матриц см. в таблице 1:

Таблица 1 – Время выполнения от числа итераций вычисления фрактала

|  |  |  |
| --- | --- | --- |
| Размер матрицы (N, N) | OpenCL, GPU (сек) | Штрассен, CPU (сек) |
| 128 | 0.181732 | **0.042008** |
| 256 | **0.057057** | 0.075035 |
| 512 | **0.057902** | 0.246533 |
| 1024 | **0.065808** | 1.23141 |
| 2048 | **0.127959** | 7.66022 |
| 4096 | **0.520505** | 57.2085 |
| 8192 | **3.17413** | 444.024 |

Исходя из данных таблицы 1 реализация перемножения матриц на GPU, OpenCL с использованием локальной памяти устройства показывает гигантское увеличение скорости вычислений по сравнению с реализацией Штрассен, CPU.

**Заключение.**

В ходе работы было реализовано умножение матриц с использованием локальной памяти на OpenCL. Так же был слегка оптимизирован реализация умножения матриц Штрассена с использованием пула потоков. Как показала оценка производительности GPU, OpenCL гораздо производительнее, чем Штрассен, CPU.