

Теория параллелизма

Отчёт

Решение уравнения теплопроводности
оптимизированными методами

Выполнила Рыбинцева Светлана 23930
05.05.25

Цель работы: Реализовать оптимизированное решение уравнения теплопроводности (разностная схема – пятиточечный шаблон) на двумерной области на равномерных сетках (128^2 , 256^2 , 512^2 , 1024^2).

Компилятор — g++

Профилировщик — “Nsight Systems”

Замер времени проводил используя библиотеку `std::chrono`

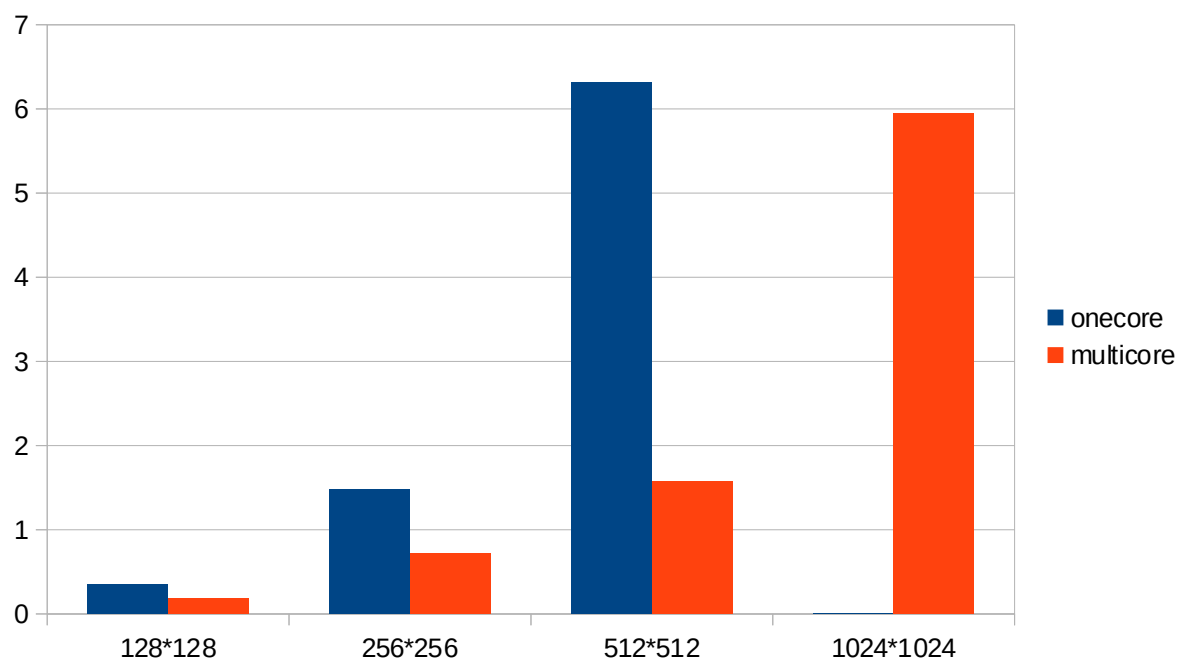
CPU-onecore

Размер сетки	Время выполнения(с)	Точность	Количество итераций
128*128	0.352	0.000001	8167
256*256	1.48	0.000001	8495
512*512	6.32	0.000001	8913

CPU-multicore

Размер сетки	Время выполнения(с)	Точность	Количество итераций
128*128	0.183	0.000001	8167
256*256	0.72	0.000001	8495
512*512	1.572	0.000001	8913
1024*1024	5.941	0.000001	8983

Диаграмма сравнения время работы CPU-one и CPU-multi



Выполнение на GPU

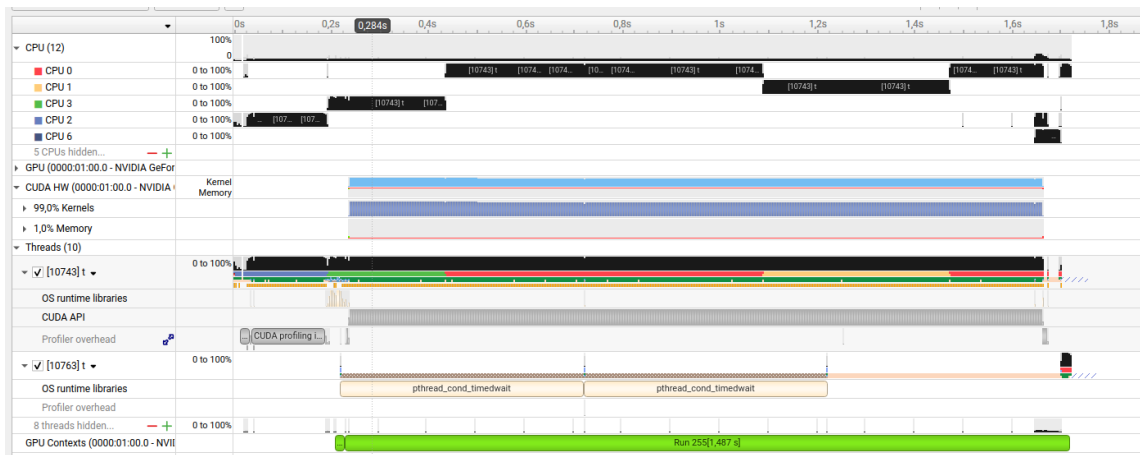
Этапы оптимизации на сетке 512*512

Этап №	Время выполнения	Точность	Максимальное количество итераций	Комментарии
1	15.37	10^{-6}	1_000_000	Не оптимизированная
2	1.44	10^{-6}	1_000_000	Убрано лишнее копирование
3	0.47	10^{-6}	1_000_000	Убрано лишнее копирование между матрицами

1)



2)



3)

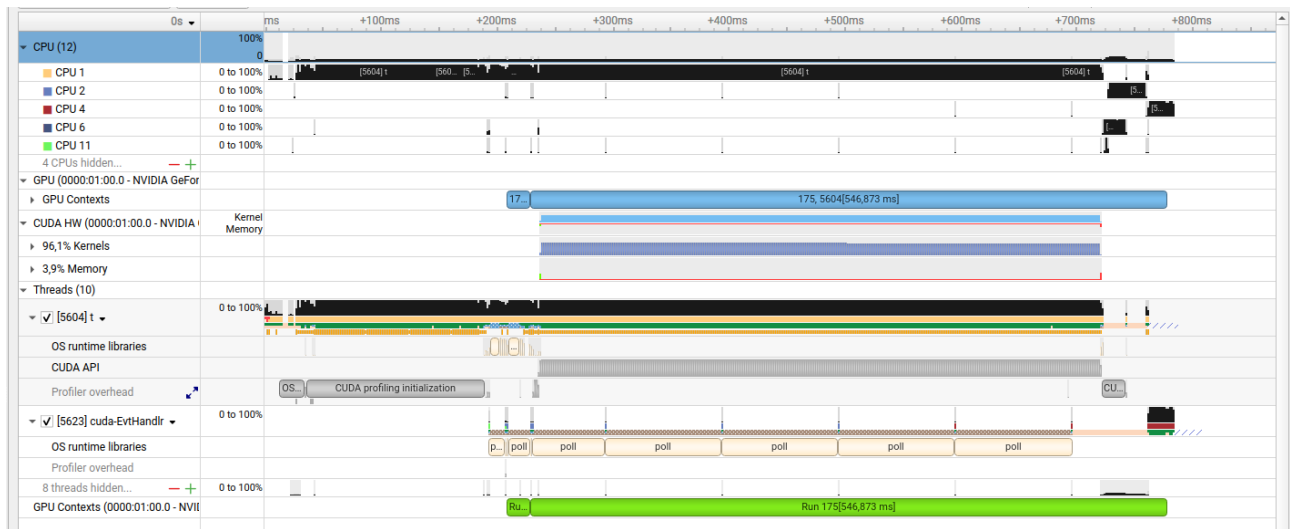
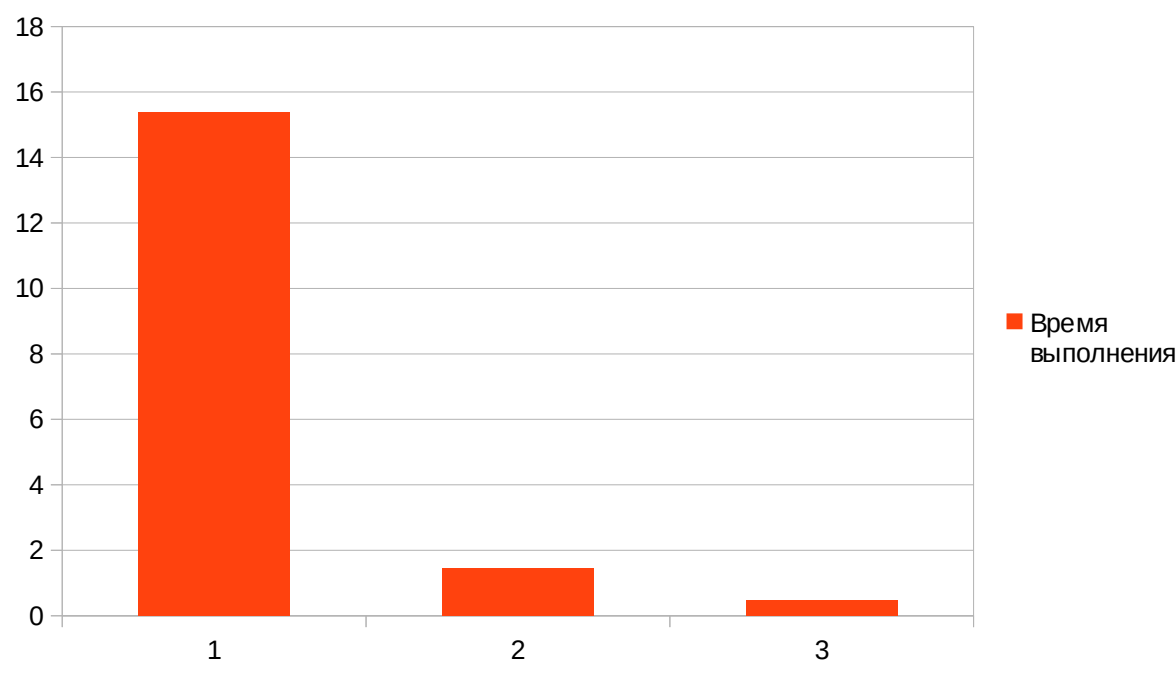


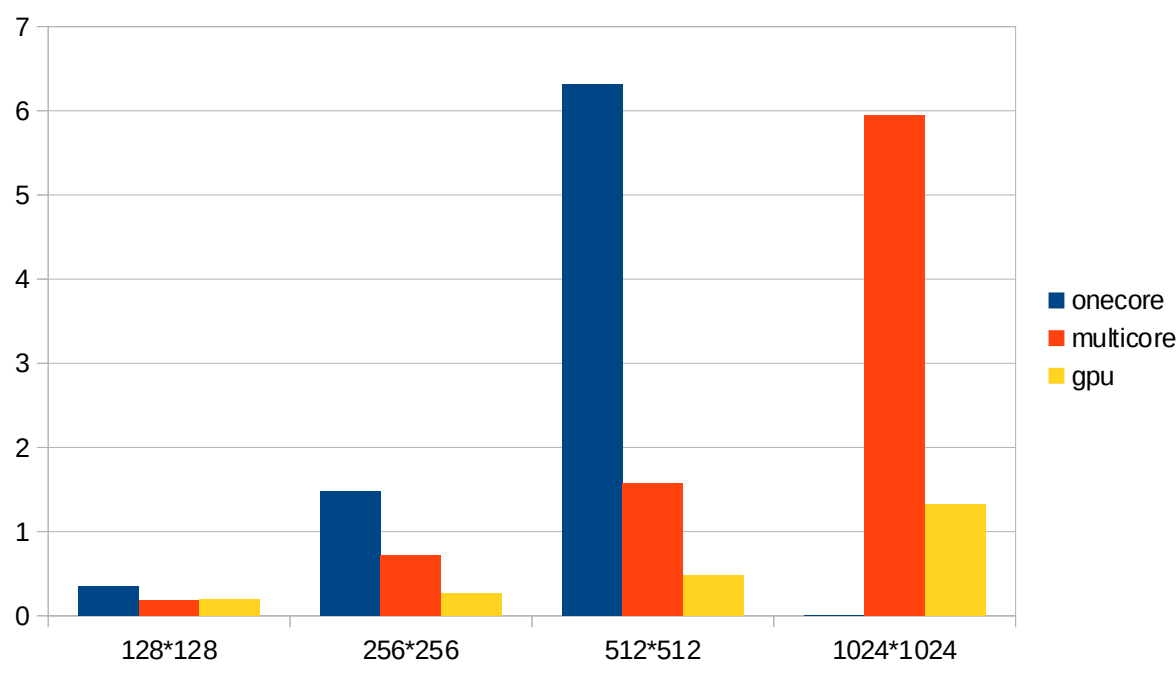
Диаграмма оптимизации



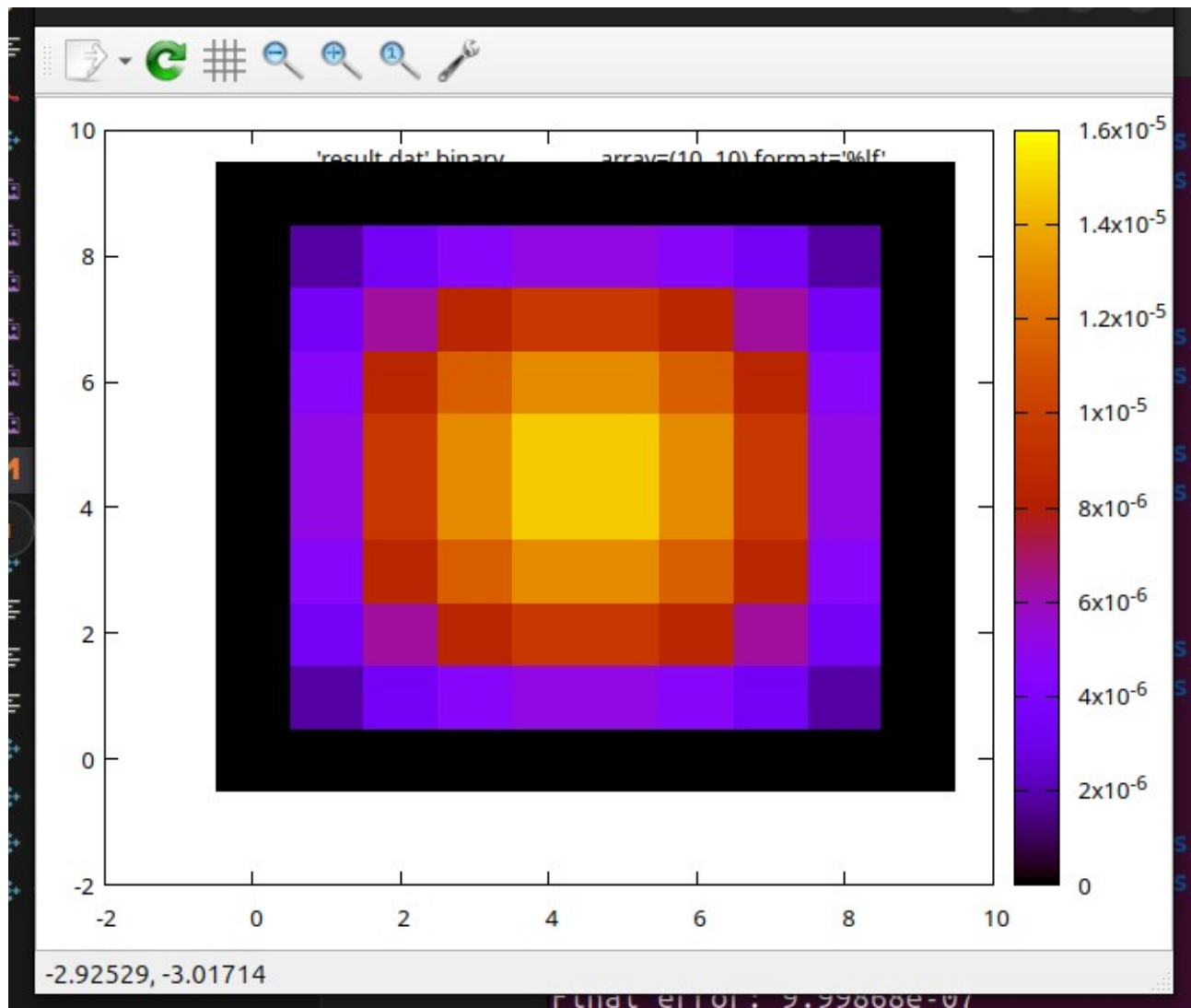
GPU – оптимизированный вариант

Размер сетки	Время выполнения	Точность	Количество итераций
128 * 128	0.20	10^-6	8167
256 * 256	0.27	10^-6	8495
512* 512	0.48	10^-6	8913
1024 * 1024	1.33	10^-6	8983

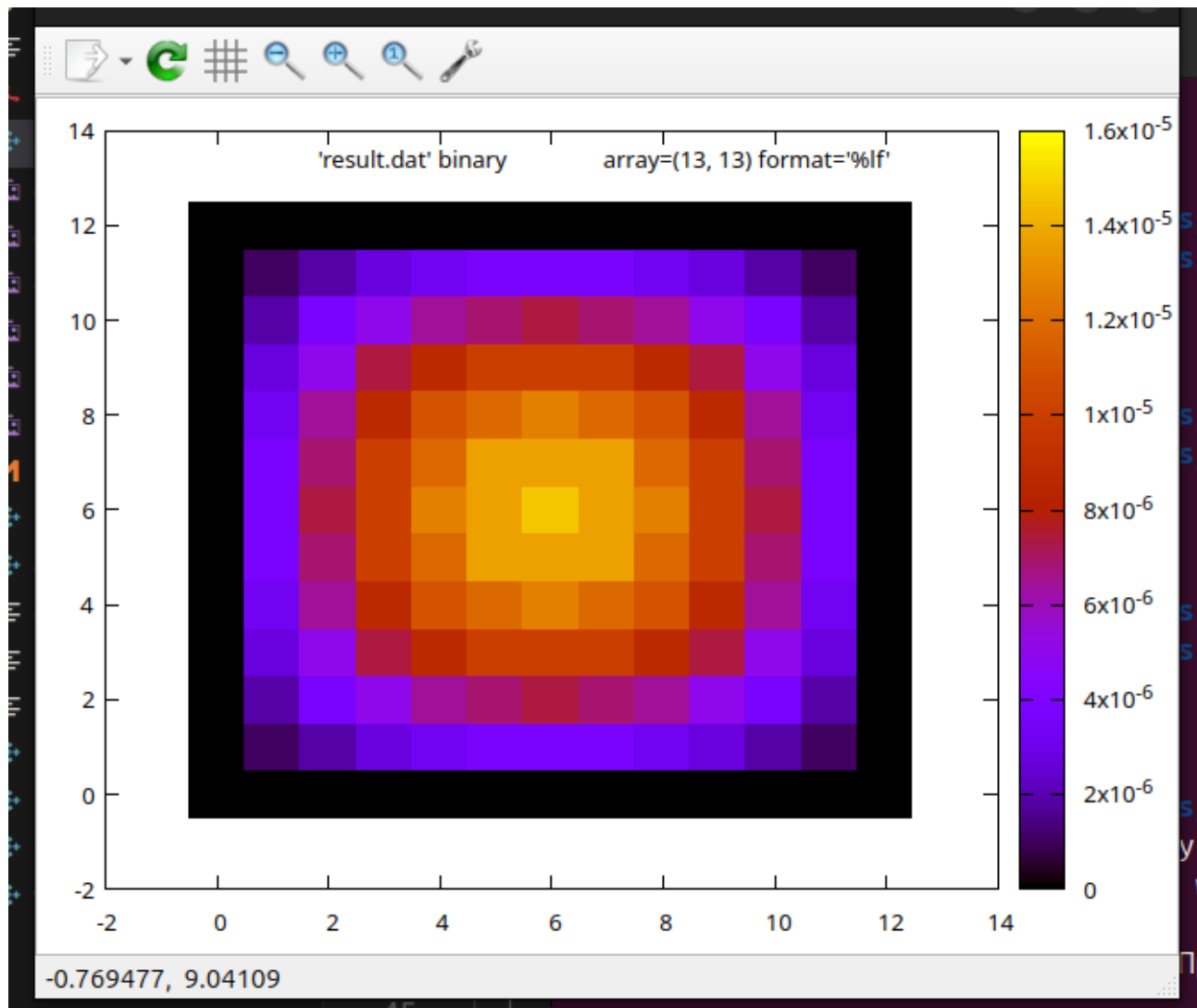
Onecore, multicore and gpu сравнение



10x10



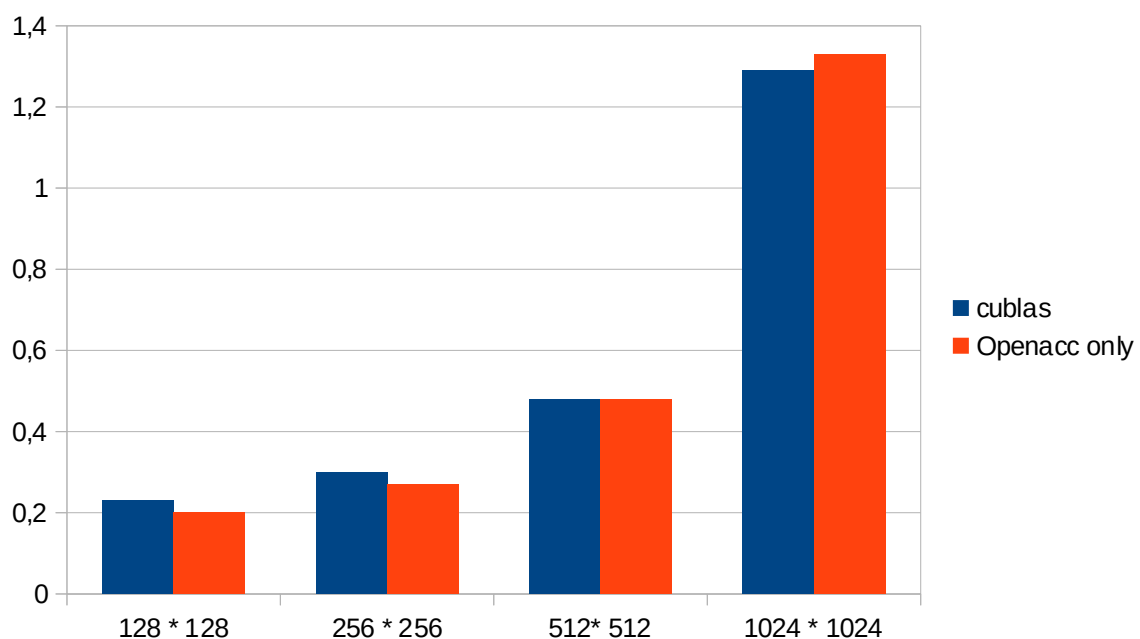
13x13



Оптимизированная версия программы показывает лучшую производительность при сохранении той же точности результатов. Основные улучшения достигнуты за счет эффективного использования параллелизма на GPU.

Cublas:

Размер сетки	Время выполнения	Точность	Количество итераций
128 * 128	0.23	10^{-6}	8167
256 * 256	0.3	10^{-6}	8495
512 * 512	0.48	10^{-6}	8913
1024 * 1024	1.29	10^{-6}	8983



Реализация через cuBLAS работает примерно также как и на orepass, с увеличением размера матрицы начинает работать чуть быстрее