

**МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)**

**Институт №8 «Информационные технологии и прикладная математика»
Кафедра 806 «Вычислительная математика и программирование»**

**Лабораторная работа №4
по курсу «Программирование графических процессоров»**

Работа с матрицами. Метод Гаусса.

Выполнил: Д.В. Коростелев

Группа: 8О-408Б

Преподаватели: К.Г. Крашенинников,
А.Ю. Морозов

Москва, 2021

Условие

Цель работы:

Использование объединения запросов к глобальной памяти. Реализация метода Гаусса с выбором главного элемента по столбцу. Ознакомление с библиотекой алгоритмов для параллельных расчетов Thrust.

Вариант 2. Метод расстояния Махаланобиса.

На первой строке задано число n -- размер матрицы. В следующих n строках, записано по n вещественных чисел -- элементы матрицы.

Программное и аппаратное обеспечение

- Графический процессор NVIDIA GeForce GTX 1050

Графическая память	2 Gb
Разделяемая память на блок	48 Mb
Константная память	64 Mb
Количество регистров на блок	65536
Максимальное количество блоков на процессор	32
Максимальное количество потоков на блок	1024
Количество мультипроцессоров	5

- Процессор Intel Core i7-7700HQ 4x 2.808ГГц

Количество ядер	4
Количество потоков	8
Базовая тактовая частота	2.80 GHz
Максимальная тактовая частота	3.80 GHz
Кеш-память	6 Mb

- Оперативная память DDR4-SODIMM

Объем памяти	8 Gb
Частота	2400 MHz
Форм-фактор	SODIMM
Количество плашек	2

- SSD и HDD накопители

Объем SSD накопителя	128 Gb
Объем HDD накопителя	1 Tb

- Программное обеспечение

Операционная система	Windows 10 Pro
Средство разработки на CUDA (IDE)	Microsoft Visual Studio 2019
Компиляторы	MSVC 2019
Версия CUDA Toolkit	11.4.2
Дополнительный текстовый редактор	Notepad++

Метод решения

Требуется реализовать многопоточную программу, которая будет способна вычислять определитель матриц любого размера. Для того, чтобы получить определитель матрицы достаточно привести матрицу к верхнему диагональному виду, на главной диагонали должны быть единицы. Запомнив все коэффициенты, на которые делятся строки во время прохода метода и перемножив их можем получить определитель матрицы.

Описание программы

Производим ввод матрицы. Матрицу храним в транспонированном виде, так как в будущем при выполнении метода Гаусса потребуется брать набор лидирующих элементов каждой строки при помощи одного вызова копирования. Далее начинаем выполнять алгоритм Гаусса выполняя большую часть вычислений на видеокарте. Ядра устраиваем таким образом, чтобы работал механизм объединений потоков, следим за тем, чтобы не было гонки потоков. После всех вычислений перемножаем коэффициенты таким образом, чтобы с наибольшей вероятностью исключить переполнения.

Результаты

Замеры времени работы ядер с различными конфигурациями (размерность теста показывает размерность квадратной матрицы).

CUDA конфигурация / Размерность теста		100	1000	2000	10000
Одномерные ядра	Двумерные ядра				
32;32	8x8;8x8	6 мс	281 мс	1 682 мс	90 100 мс
64;128	16x16;16x16	9 мс	279 мс	1 401 мс	63 127 мс
128;256	16x16;32x32	26 мс	418 мс	1 624 мс	51 345 мс
256;512	32x32;32x32	96 мс	852 мс	2 553 мс	40 234 мс

Сравнение вычислений на CPU с вычислениями на GPU

Размерность входных данных	CPU	32;32 8x8;8x8	128;256 16x16;32x32
100	5 мс	6 мс	26 мс
1000	1 253 мс	281 мс	418 мс
10000	~10 мин	90 100 мс	51 345 мс

Выводы

Полученная программа позволяет за минимальное время вычислять определители огромных матриц. Исходя из полученных бенчмарков замечен колоссальный выигрыш в скорости вычислений, так для вычисления на CPU определителя матрицы потребовалось около 10 минут, а на GPU это заняло чуть меньше 1ой минуты.