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: Г.Н. Хренов

Группа: 8О-407Б

Преподаватели: К.Г. Крашенинников,

А.Ю. Морозов

Москва, 2021

**Условие**

1. 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.
2. Вариант 6. Нахождение ранга матрицы.

**Программное и аппаратное обеспечение**

GPU name: NVIDIA GeForce RTX 2060

compute capability 7:5

totalGlobalMem: 6442450944

sharedMemPerBlock: 49152

totalConstMem: 65536

regsPerBlock: 65536

maxThreadsDim: 1024 1024 64

maxGridSize: 2147483647 65535 65535

multiProcessorCount: 30

CPU name: AMD Ryzen 7 3750H with Radeon Vega Mobile Gfx

MaxClockSpeed: 2300

NumberOfCourse: 4

RAM: 8

SSD: 256, HDD: 1024

OS: Windows10

Compiler: nvcc

**Метод решения**

Реализуем метод Гаусса, каждая итерация распараллеливается на gpu, максимальный элемент ищется с помощью метода из библиотеки thrust. Ранг определяется как количество “ступенек” в приведенной матрице. Ранг считается сразу в итерационном процессе приведения матрицы для экономии времени.

**Описание программы**

lab4.cu:

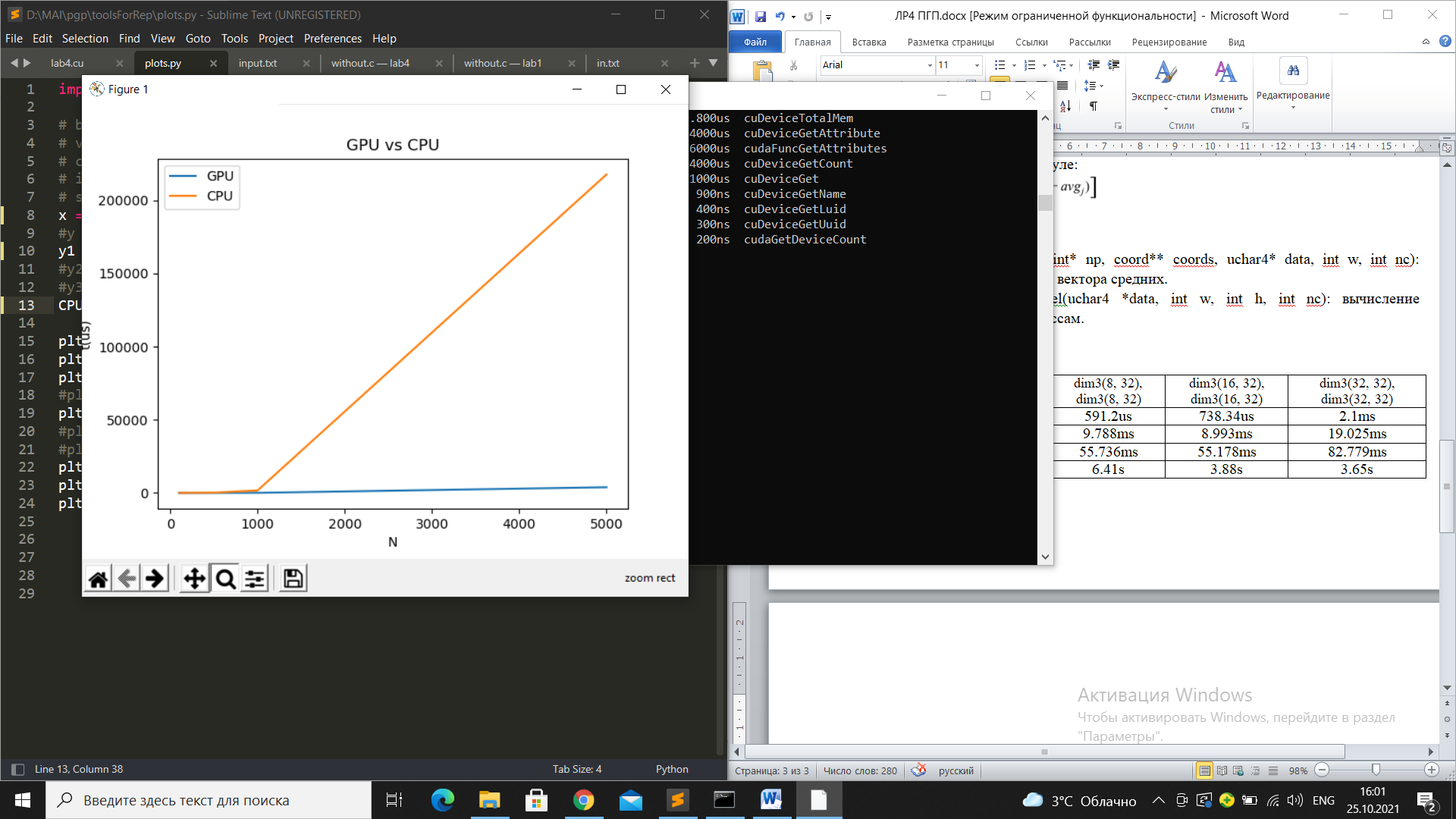
\_\_global\_\_ void kernel\_gaussian(double\* a, int w, int h, int i, int j) – итерация метода Гаусса

\_\_global\_\_ void kernel\_swap(double\* dev\_data, int w, int h, int j, int i, int ind) – меняет i-ю и j-ю строчку местами

thrust::max\_element(col\_beg+i, col\_beg+w, comp) – максимальный элемент в указанной строке

**Результаты**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | dim3(8, 16), dim3(8, 16) | dim3(8, 32), dim3(8, 32) | dim3(16, 32), dim3(16, 32) | dim3(32, 32), dim3(32, 32) |
| 100\*100 | 489.98us | 591.2us | 738.34us | 2.1ms |
| 500\*500 | 9.76ms | 9.788ms | 8.993ms | 19.025ms |
| 1000\*1000 | 63.488ms | 55.736ms | 55.178ms | 82.779ms |
| 5000\*5000 | 6.3s | 6.41s | 3.88s | 3.65s |



**Выводы**

Метод Гаусса имеет сложность O(n3), но его преимущество заключается в том, что он легко распараллеливается. Библиотека thrust очень удобна в применении и экономит время, позволяя не писать примитивные ядра для реализации простых методов. Объем глобальной памяти самый большой из всех типов памяти, но в тоже время эта память самая медлительная. Однако роста производительности можно добиться с помощью объединения запросов к глобальной памяти. Для этого нужно расположить данные в памяти таким образом, чтобы соседние потоки получали доступ к последовательным местоположениям в глобальной памяти.