МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

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

Лабораторная работа №1
по курсу «Программирование графических процессоров»

Освоение программного обеспечения для работы с технологией CUDA.

Примитивные операции над векторами.

Выполнил: П.А. Гамов
Группа: 8О-407Б
Преподаватели: К.Г. Крашенинников,
А.Ю. Морозов

Москва, 2021

Условие

Ознакомление и установка программного обеспечения для работы с программно-аппаратной архитектурой параллельных вычислений (Cuda). Реализация одной из примитивных операций над векторами.

Вариант 2. Вычитание векторов.

Программное и аппаратное обеспечение

Nvcc 7.0

Ubuntu 14.04 LTS

Compute capability	6.1
Name	GeForse GTX 1050
Total Global Memory	2096103424
Shared Mem per block	49152
Registers per block	65534
Max thread per block	(1024,1024,64)
Max block	(2147483647, 65535, 65535)
Total constant memory	65536
Multiprocessor’s count	5

Метод решения

Выделим 3 массива типа double для записи в них чисел. Далее с помощью cudaMalloc выделяем память на карте. Используя cudaMemcpy скопируем наши векторы на векторы карты для дальнейшего использования.

Описание программы

Программа состоит из 1 файла и 1 ядра.

```
__global__ void sub (const double *v1, const double *v2, double *res, int n) {
   for (int i = threadIdx.x + blockIdx.x * blockDim.x; i < n; i += gridDim.x * blockDim.x) {
      res[i] = v1[i] - v2[i];
   }
}
```

Результаты

	10^4	10^6	10^8
<<<1,32>>>	0.000239	0.020254	1.99469
<<<32,32>>>	0.000042	0.001424	0.141055
<<<256,256>>>	0.000029	0.000707	0.070564
<<<256,512>>>	0.000033	0.000704	0.070367
<<<512,512>>>	0.000043	0.000701	0.070502
<<<512,1024>>>	0.000090	0.000751	0.069963

<<<1024,1024>>>	0.000156	0.000808	0.06992
C++	0.000087	0.004305	0.435336

Выводы

Данный алгоритм является демонстрационным, реализация может быть полезна в векторной математике, например для вычитания больших векторов, но выделение памяти и копирование не дает существенного выигрыша. Только на огромных значениях, многопоточная реализация показывает существенный выигрыш до 10 раз.