GPU Computing CSE 560 (Winter 2022) - Lab 2

Aditya Singh Rathore - 2018007

February 4, 2022

Constant Memory : Task

```
// initialize1.cu
__global__ void initialize1(float* C){
    int i = blockDim.x*blockIdx.x + threadIdx.x;
    if(i < LENGTH){
        C[i] = A[i] + B[i];
    }
}
//...
```

```
// initialize2.cu
__global__ void initialize2(float *C){
    int i = blockDim.x*blockIdx.x + threadIdx.x;
    if(i < LENGTH){
        C[i] = A[blockIdx.x] + B[blockIdx.x];
    }
//...
```

Code was tested on Nvidia 1050 Ti	
Code	Execution Time
initialize1.cu	0.00989895 ms
initialize2.cu	0.00919106 ms

initialize2.cu is faster.

The constant memory is cached. In *initialize2.cu*, all threads in the warp will be reading same A[blockIdx.x] and B[blockIdx.x] as blockIdx.x is same for all of them, which is available in cache.

In *initialize1.cu*, each threads will be reading different A[i] and B[i] which will lead to cache misses and lead comparatively larger execution time.