Name : Kiran K. Patil

ID : 211070904

Branch : Computer Engineering.

Assignment 01

Aim : Write a program (CUDA program) for Matrix addition, Matrix Transpose and Matrix Multiplication.

Theory :-

1) CUDA - Compute Unified Device Archirechture is a parallel computing platform and application programming Interface (API) model created by NVIDIA. It allows software developers to use CUDA - enabled graphics processing unit (GPU)

2) The Main Idea behind CUDA programming is to use the massive parallism of GPUs to accelatate the computation of certain types of data / tasks. A CUDA enabled GPU contains many small processors, called CUDA cores, that can work together in parallel to perform calculations.

CUDA programming involves writing code that runs on the GPU as well as on the CPU. The code that runs on the GPU is called a kernal, and is executed by many thread in parallel. Each thread performs the same operation on a diffrent peice of data, and the results are combined to produce the final output.

The CUDA programming model consists of two main components: host code and device code. The host code runs on the CPU and its responsible for allocating memory on the GPU, and launching kernals and transfering data between CPU and GPU. The Device code runs on the GPU and contains the kernal that perform the computation

The kernal is lauached with a grid of blocks where each block contains a number of threads. The grid block and diamensions are specified as argument to the kernal launch, and they determine the number of threads that will be executing the kernal. Each Thread has unique ID that can be used to determine its position whithin the grid block.

Threads within a block can synchronize and share data through shared memory, which is a fast and efficient memory space that is shared among threads within the same block.

Why we need CUDA:

1) GPU designed to perform high speed parallel computation to display graphics suchas games

2) use available CUDA resources. More than 100million GPU's are already deployed

3) It provides 30-100x speed up over Microprocessors for some application.

4) It has very small ALU compared to CPU. This allows many parrallel calculations, such as cal. the color for each pixel.

working of CUDA:

1) GUP runs one kernal at a time.

2) Each kernal consists of blocks which are independent groups of ALU's

3) Each block contains thread which are level of computation.

4) The thread in each block typically works together to calculate a value.

5) Threads in same block can share memory

6) The CUDA, sending information form CPU to GPU is often the most typical part of the computation.

7) For each thread local memory is fastest followed by shared memory, global, static and texture slowest

CUDA applications:

1) computational finance.
2) safty and security
3) Deep learning & machine learning
4) Manufacturing
5) Data science and analytics.
6) climate, weather and ocean monitoring
7) Reasearch

Overall, CUDA programming requires a good understanding of parallel computing consepts and a deep knowledge of CUDA programming model, as well as the specific features of the target GPU. However it can be a powerful tool for accelerating the computation of certain types of tasks and is widly used in feids such as scientific computing Image computing and machine leaning.

conclusion: Hence from this experiment we learned about CUDA programming and learned how to execute it of ~~gotgal~~ google colab platform and we also perform matrix additi subtraction and Transpose and matrix multiplicatio in CUDA programming.