

Class: Final Year (Computer Science and Engineering)

Year: 2021-22 **Semester:** 1

Course: High Performance Computing Lab

Practical No.

Exam Seat No:2018BTECS00100

1. Exam Seat Number - Prakash Singh

Problem Statement 1: Write a CUDA C program to perform the simple matrix-matrix multiplication. Perform code optimization and profiling of existing CUDA C program. (Attach Snapshot of execution before optimization and after optimization)

Screenshot #:

Before Optimization:

```
prax@praxx-ideapad:~/Desktop/HPC/8$ nvcc 1.cu
prax@praxx-ideapad:~/Desktop/HPC/8$ ./a.out
The elapsed time is 0.419739 seconds
prax@praxx-ideapad:~/Desktop/HPC/8$ s
```

After Optimization:

```
prax@praxx-ideapad:~/Desktop/HPC/8$ nvcc 1b.cu
prax@praxx-ideapad:~/Desktop/HPC/8$ ./a.out
The elapsed time is 0.305848 seconds
prax@praxx-ideapad:~/Desktop/HPC/8$
```

Profiling of optimized program

```
prax@praxx-ideapad:~/Desktop/HPC/8$ nvcc 1b.cu
prax@praxx-ideapad:~/Desktop/HPC/8$ nvprof --print-gpu-trace ./a.out
==21749== NVPROF is profiling process 21749, command: ./a.out
The elapsed time is 0.383789 seconds
==21749== Profiling application: ./a.out
==21749== Profiling result:
  Start Duration      Device      Grid Size      Block Size      Regs*      SSMem*      DSMem*      Size      Throughput      SrcMemType      D
stMemType
252.68ms 581.98us      Device      Context      Stream      Name      -      -      -      976.56KB      1.6003GB/s      Pageable
  Device NVIDIA GeForce      1      7 [CUDA memcpy HtoD]
253.41ms 587.10us      Device      Context      Stream      Name      -      -      -      976.56KB      1.5863GB/s      Pageable
  Device NVIDIA GeForce      1      7 [CUDA memcpy HtoD]
254.00ms 231.78ms      (500 500 1)      (500 1 1)      24 1.9531KB      0B      -      -      -
  - NVIDIA GeForce      1      7 matrix_multiply(int*, int*, int*) [116]
485.78ms 576.06us      Device      Context      Stream      Name      -      -      -      976.56KB      1.6167GB/s      Device
Pageable NVIDIA GeForce      1      7 [CUDA memcpy DtoH]

Regs: Number of registers used per CUDA thread. This number includes registers used internally by the CUDA driver and/or t
ools and can be more than what the compiler shows.
SSMem: Static shared memory allocated per CUDA block.
DSMem: Dynamic shared memory allocated per CUDA block.
SrcMemType: The type of source memory accessed by memory operation/copy
DstMemType: The type of destination memory accessed by memory operation/copy
```

Problem Statement 2:

Write a CUDA C program to demonstrate the use of different GPU memories.

- Use of private memory.
- Use of shared memory.
- Use of global memory.

Screenshot #:

Use of private memory:

```
prax@praxx-ideapad:~/Desktop/HPC/8$ nvcc 2a.cu
prax@praxx-ideapad:~/Desktop/HPC/8$ ./a.out

Six elements of first array: 0 1 2 3 4 5
Six elements of second array: 1 2 3 4 5 6
Sum of two arrays:
1      3      5      7      9      11      prax@praxx-ideapad:~/Desktop/HPC/8$
```

Use of shared memory:

```
prax@praxx-ideapad:~/Desktop/HPC/8$ nvcc 2c.cu
prax@praxx-ideapad:~/Desktop/HPC/8$ ./a.out
Elements in array:0 100 200 300 400 500 600 700 800 900
0 100 200 300 400 500 600 700 800 900
prax@praxx-ideapad:~/Desktop/HPC/8$
```

Use of global memory:

```
prax@praxx-ideapad:~/Desktop/HPC/8$ nvcc 2b.cu
prax@praxx-ideapad:~/Desktop/HPC/8$ ./a.out

Six elements of first array: 0 1 2 3 4 5
Six elements of second array: 1 2 3 4 5 6
Sum of two arrays:
1      3      5      7      9      11      prax@praxx-ideapad:~/Desktop/HPC/8$
```

Github Link:<https://github.com/praxx1/HPC-LAB/tree/master/8>

