**Class:** Final Year (Computer Science and Engineering)

**Year:** 2022-23 **Semester:** 1

**Course:** High Performance Computing Lab

**Practical No. 6**

**Exam Seat No:**

2019BTECS00033 – Teknath Krishna Jha

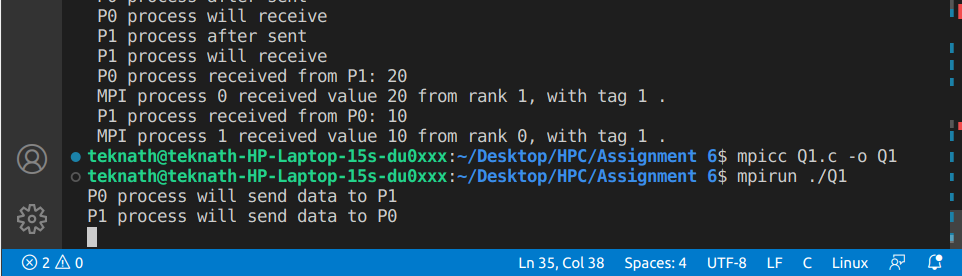
**Title of practical:**

MPI programming

**Problem Statement 1:**

Q1: Implement a MPI program to give an example of Deadlock.

**Output 1:**

****

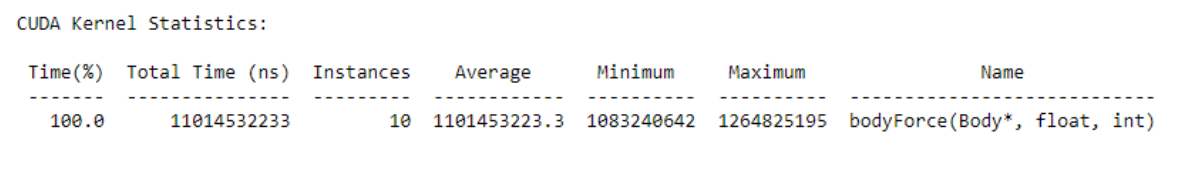
**COMMENT : Here Both process p0 and p1 sending data to p1 and p0 respectively at same time and due to this deadlock is created .**

**Problem Statement 2:**

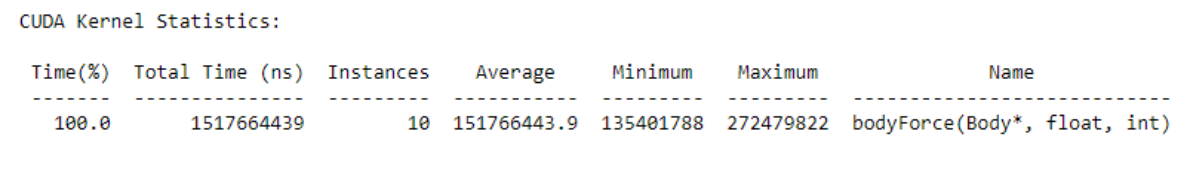
Q2. Implement blocking MPI send & receive to demonstrate Nearest neighbor exchange of data in a ring topology.

**Screenshot 2:**

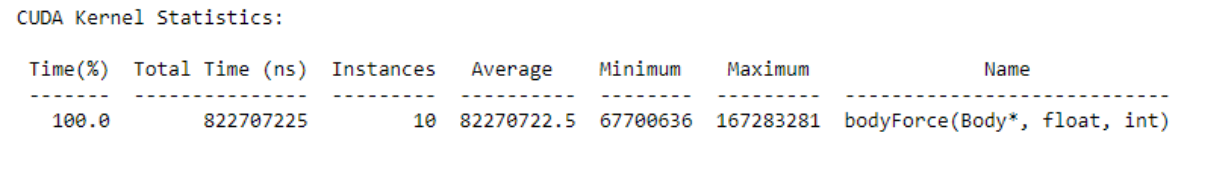
**<<<1,1>>>**

****

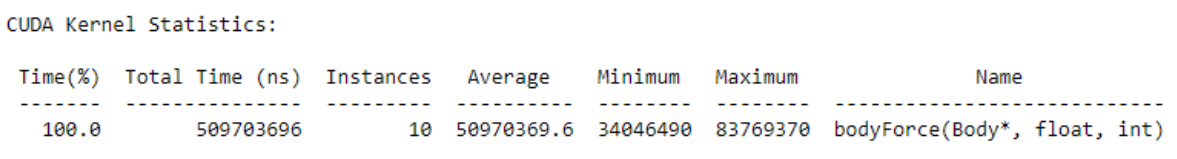
**<<<1,8>>>**

****

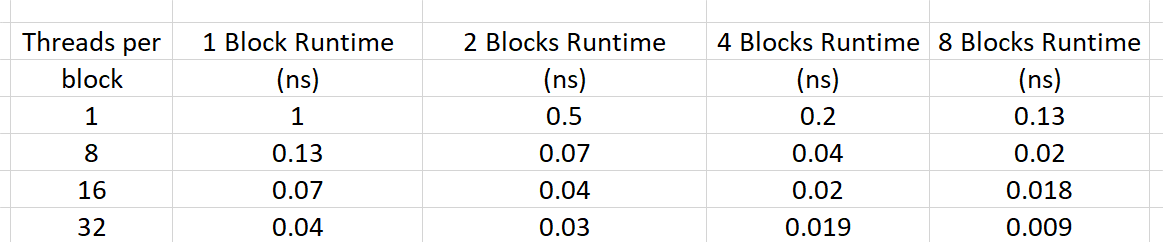
**<<<1,16>>>**

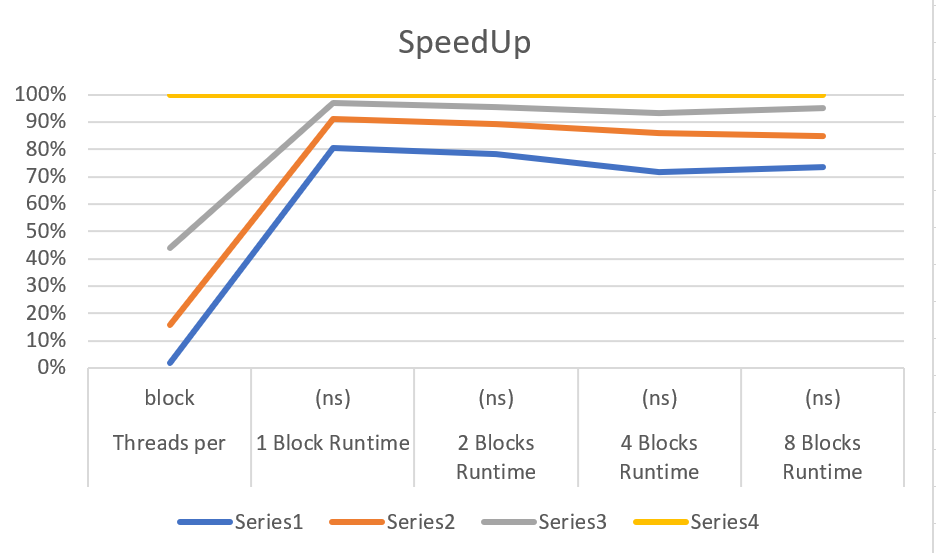
****

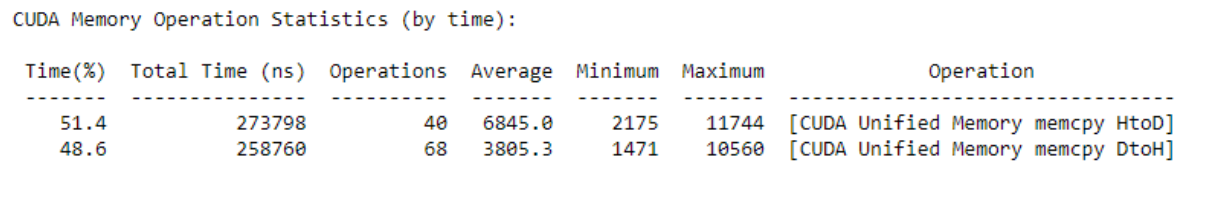
**<<<1,32>>>**

****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Threads per | 1 Block Runtime | 2 Blocks Runtime | 4 Blocks Runtime | 8 Blocks Runtime |
| block | (ns) | (ns) | (ns) | (ns) |
| 1 | 11014532233 | 5563155974 | 2868550980 | 1498958754 |
| 8 | 1517664439 | 830366506 | 484274927 | 308540581 |
| 16 | 822707225 | 489260562 | 298343463 | 203863012 |
| 32 | 509703696 | 331265419 | 210476249 | 106152724 |
| SpeedUp calculation of above observation table : |  |  |  |  |



****

****

**Execution time:**

As expected, the execution time of kernel decreases as the total threads (blocks \* threads) increases. Total time taken by program also follows a similar trend.

**Memory allocation and deallocation:** All instances seem to follow random fluctuations since the same amount of memory is being allocated outside any parallel region. But is pretty much constant

**Github Link:**

<https://github.com/Teknath-jha/HPC-LAB-2019BTECS00033/tree/main/Assignment-9>.

Q3. Write a MPI program to find the sum of all the elements of an array A of size n. Elements of an array can be divided into two equals groups. The first [n/2] elements are added by the first process, P0, and last [n/2] elements the by second process, P1. The two sums then are added to get the final result.