General Instructions

- **Timing 3PM – 5PM**
- **Answer both the questions**
- Write your Registration Number, Name, Roll Number, Semester, Section in **first page.**
- Write roll number and sign **at the top of every page.**
- Upload a legible handwritten single file with your **Section_roll number_name as name of the file e.g., D_12_xxx**

1) Write an efficient MPI program using **4** processes(including root(process 0)), where **process 0** reads a matrix **A** of size **4X4** and displays the resultant matrix **RES** of size **4X4**. Each element of resultant matrix is obtained by adding a key value. The key value for the first row is the bitwise AND of the elements in the last row. The key value for the remaining rows is the bitwise AND of the elements from the previous row of the Matrix **A**. Key values obtained have to be sent to the required processes using point to point standard send function. For remaining operations use any collective communication function except MPI_Bcast.

Note : Do not calculate the Bitwise AND manually.

Sample input:

```
$ mpirun -n 4 ./a.out

Enter Matrix A:
1  2  3  4
2  3  4  5
3  4  5  6
4  5  6  7

Output Matrix B:
5  6  7  8
2  3  4  5
3  4  5  6
4  5  6  7
```

2) Write **CUDA host and kernel code** that reads a string **S** with unequal length words. The host code should find **MAXLEN** which is the length of the word with maximum size. Kernel should produce a character matrix **A** as follows: Every character of matrix **A** should be generated in parallel. It generates matrix **A** by considering each word of **S** as an independent row. In the matrix **A** where the words that have length smaller than MAXLEN, put '*****' in the remaining places as shown below. Change the case of all vowels in the character matrix **A**. Kernel should accept only the required elements. Resultant matrix should be printed in the host code.

Input:	**Output A:**
Enter a string(S): PCAP ONLINE LAB EXAMS	P C a P * * o N L i N e L a B * * * E X a M S *