NAME : RAJVARDHAN REDDY
REG NO : 180905093
SEC :B -B1 , ROLL NO : 19

1)

```
#include <mpi.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

void main (int argc, char * argv[])
{
   int rank, size;
   MPI_Init(&argc, &argv);
   MPI_Comm_rank(MPI_COMM_WORLD, &rank);
   MPI_Comm_size(MPI_COMM_WORLD, &size);

   long fact;
   long i, n;
   long rec;
   long arr[100], facts[100];

   if (rank == 0)
   {
      n = size;
      printf("Enter the numbers: \n");

      for (i = 0; i < n; ++i)
      {
         scanf("%ld", &arr[i]);
      }
   }

   MPI_Scatter(arr, 1, MPI_LONG, &rec, 1, MPI_LONG, 0,
MPI_COMM_WORLD);
   printf("Process [%d] received = %ld.\n", rank, rec);

   fact = 1;

   for (i = 2; i <= rec; ++i)
   {
      fact *= i;
   }
```

```
   MPI_Gather(&fact, 1, MPI_LONG, facts, 1, MPI_LONG, 0,
MPI_COMM_WORLD);




 if (rank == 0)
   {
      printf("Sum of factorials = ");
      long sum = 0;

      for (i = 0; i < n; ++i)
      {
         sum += facts[i];
         printf("%ld %s", facts[i], (i != n-1)?"+ ":" ");
      }

      printf(" = %ld\n", sum);
   }

   MPI_Finalize();
}
```

```
student@lplab-Lenovo-Product:~/180905093_PCAP Lab/lab 3$ mpicc w3_p1.c
student@lplab-Lenovo-Product:~/180905093_PCAP Lab/lab 3$ mpirun -np 5 ./a.out
Enter the numbers:
1
2
3
4
5
Process [0] received = 1.
Process [1] received = 2.
Process [2] received = 3.
Process [3] received = 4.
Process [4] received = 5.
Sum of factorials = 1 + 2 + 6 + 24 + 120    = 153
student@lplab-Lenovo-Product:~/180905093_PCAP Lab/lab 3$
```

2)

```
#include <mpi.h>
#include <stdio.h>
#include <math.h>


int main(int argc, char** argv) {
   MPI_Init(NULL, NULL);
```

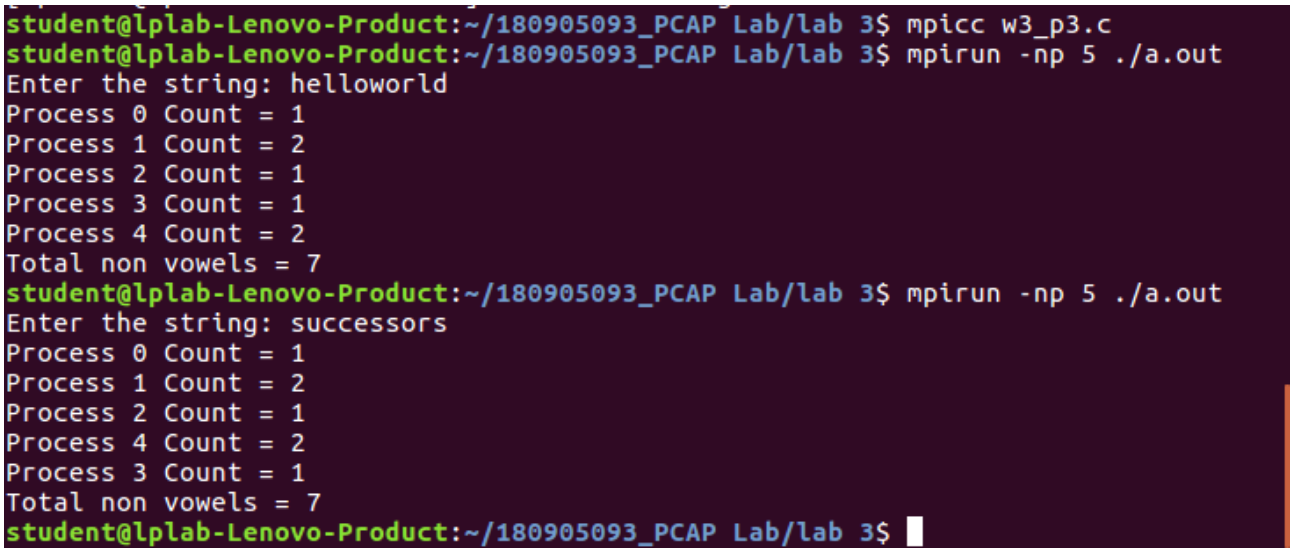
```
    int size;
    MPI_Comm_size(MPI_COMM_WORLD, &size);
    int rank;
    MPI_Comm_rank(MPI_COMM_WORLD, &rank);

    int A[100] , m = 3;
    float B[100];
    if(rank == 0){
        //printf("Enter M: ");
        //scanf(" %d" , &m);
        printf("Enter %d size array: \n", size *m);
        for(int i = 0; i < size * m; i++){
            scanf(" %d" , &A[i]);
        }
    }
    int c[m] ;
    float avg = 0.0;
    MPI_Scatter(A , m ,MPI_INT, &c , m, MPI_INT, 0 , MPI_COMM_WORLD);
    for(int i = 0; i < m; i++){
        avg += c[i];
    }
    avg /= m;
    printf("Process %d outputs %.1f\n" , rank , avg);
    MPI_Gather(&avg , 1 ,MPI_FLOAT, B , 1, MPI_FLOAT, 0 ,
MPI_COMM_WORLD);
    if(rank == 0){
        float tavg = 0.0;
        for(int i = 0; i < size; i++){
            tavg += B[i];
        }
        tavg = tavg / size;
        printf("The total average is : %.1f\n" , tavg);

    }
    MPI_Finalize();
}
```

```
student@lplab-Lenovo-Product:~/180905093_PCAP Lab/lab 3$ mpicc w3_p2.c
student@lplab-Lenovo-Product:~/180905093_PCAP Lab/lab 3$ mpirun -np 2 ./a.out
Enter 6 size array:
3
4
5
6
7
8
Process 0 outputs 4.0
Process 1 outputs 7.0
The total average is : 5.5
student@lplab-Lenovo-Product:~/180905093_PCAP Lab/lab 3$
```

3)

```
#include <mpi.h>
#include <stdio.h>
#include <string.h>

int main(int argc, char* argv[])
{

int rank,size;
int count = 0;
int b[100] = {0};

int i, n, l;

char str[100], c[100];

MPI_Init(&argc, &argv);

MPI_Comm_rank(MPI_COMM_WORLD, &rank);
MPI_Comm_size(MPI_COMM_WORLD, &size);

if (rank == 0)
{

n = size;

printf("Enter the string: ");
scanf("%s", str);

l = strlen(str) / n;

}

MPI_Bcast(&l, 1, MPI_INT, 0, MPI_COMM_WORLD);
MPI_Scatter(str, l, MPI_CHAR, c, l, MPI_CHAR, 0, MPI_COMM_WORLD);

count = 0;

for (i = 0; i < l; ++i)
{
if(c[i] =='a' || c[i] == 'e' || c[i] == 'i' || c[i] == 'o' || c[i] == 'u')
continue;
count+=1;
}
```

```
printf("Process %d Count = %d\n", rank, count);
fflush(stdout);

MPI_Gather(&count, 1, MPI_INT, b, 1, MPI_INT, 0, MPI_COMM_WORLD);

if (rank == 0)
{

int tcount = 0;

for (i = 0; i < n; i++)
tcount += b[i];

printf("Total non vowels = %d\n", tcount);
fflush(stdout);

}

MPI_Finalize();
}
```



4)

```
#include <mpi.h>
#include <stdio.h>
#include <string.h>

int main(int argc, char* argv [])
```

```
{

	int rank, size;

	float avg = 0;
	char b[100], str1[100], str2[100], c1[100], c2[100], concatted[100];

	int i, j, m;

	MPI_Init(&argc, &argv);

	MPI_Comm_rank(MPI_COMM_WORLD, &rank);
	MPI_Comm_size(MPI_COMM_WORLD, &size);

	if (rank == 0)
	{

		printf("Enter string 1: ");
		scanf("%s", str1);

		printf("Enter string 2: ");
		scanf("%s", str2);

		m = strlen(str1) / size;
	}

	MPI_Bcast(&m, 1, MPI_INT, 0, MPI_COMM_WORLD);

	MPI_Scatter(str1, m, MPI_CHAR, c1, m, MPI_CHAR, 0,
MPI_COMM_WORLD);
	MPI_Scatter(str2, m, MPI_CHAR, c2, m, MPI_CHAR, 0,
MPI_COMM_WORLD);

	int t = 0;









	for (t = 0; t <= 2 * m; t += 2)
	{
		concatted[t] = c1[t/2];
		concatted[t+1] = c2[t/2];
```

```
	}

	concatted[2*m] = '\0';

	MPI_Gather(concatted, 2*m, MPI_CHAR, b, 2*m, MPI_CHAR, 0,
MPI_COMM_WORLD);

	if (rank == 0)
	{
		b[m*size*2] = '\0';
		printf("Concatted:%s\n", b);
	}

	MPI_Finalize();
}
```

```
student@lplab-Lenovo-Product:~/180905093_PCAP Lab/lab 3$ mpicc w3_p4.c
student@lplab-Lenovo-Product:~/180905093_PCAP Lab/lab 3$ mpirun -np 6 ./a.out
Enter string 1: string
Enter string 2: length
Concatted:slternigntgh
student@lplab-Lenovo-Product:~/180905093_PCAP Lab/lab 3$ mpirun -np 5 ./a.out
Enter string 1: hello
Enter string 2: world
Concatted:hweolrllod
student@lplab-Lenovo-Product:~/180905093_PCAP Lab/lab 3$
```