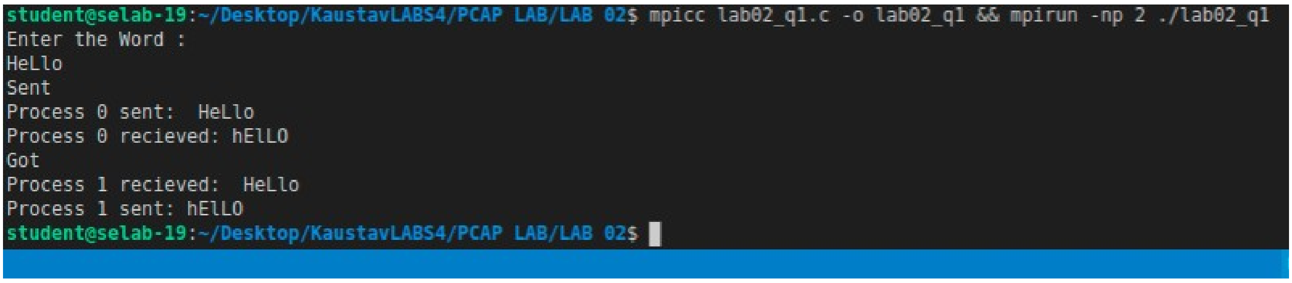
Q1

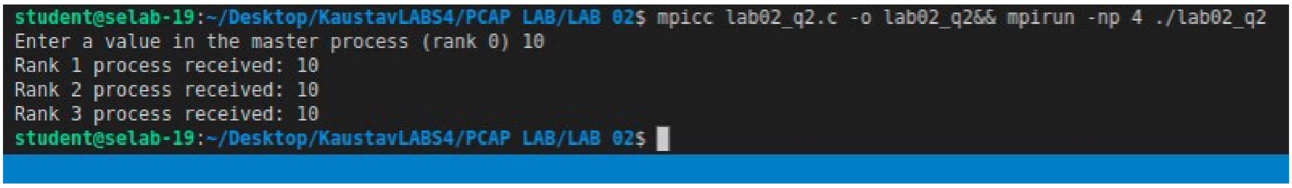
```
#include <stdio.h>
#include <string.h>
#include <mpi.h>

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

    if (rank == 0)
    {
        char a[25];

        fprintf(stdout, "Enter the Word : \n");
        fflush(stdout);
        scanf("%s", a);

        n = strlen(a);

        MPI_Ssend(&n, 1, MPI_INT, 1, 0, MPI_COMM_WORLD);
        printf("Sent\n");
        MPI_Ssend(a, n + 1, MPI_CHAR, 1, 1, MPI_COMM_WORLD);
        fprintf(stdout, "Process %d sent:  %s\n", rank, a);
        fflush(stdout);
        MPI_Recv(a, n + 1, MPI_CHAR, 1, 2, MPI_COMM_WORLD, &status);
        fprintf(stdout, "Process %d recieved: %s\n", rank, a);
        fflush(stdout);
    }
    else if (rank == 1)
    {
        char b[25];

        MPI_Recv(&n, 1, MPI_INT, 0, 0, MPI_COMM_WORLD, &status);
        printf("Got\n");
        MPI_Recv(b, n + 1, MPI_CHAR, 0, 1, MPI_COMM_WORLD, &status);
        fprintf(stdout, "Process %d recieved:  %s\n", rank, b);
        fflush(stdout);

        for (int i = 0; i < n; i++)
        {
            if (b[i] >= 'a' && b[i] <= 'z')
            {
                b[i] = b[i] - 32;
            }
            else
            {
                b[i] = b[i] + 32;
            }
        }
        MPI_Ssend(b, n + 1, MPI_CHAR, 0, 2, MPI_COMM_WORLD);
        fprintf(stdout, "Process %d sent: %s\n", rank, b);
        fflush(stdout);
    }
    MPI_Finalize();
    return 0;
}
```



```
student@selab-19:~/Desktop/KaustavLABS4/PCAP LAB/LAB 02$ mpicc lab02_q1.c -o lab02_q1 && mpirun -np 2 ./lab02_q1
Enter the Word :
HeLlo
Sent
Process 0 sent:  HeLlo
Process 0 recieved: hElLO
Got
Process 1 recieved:  HeLlo
Process 1 sent: hElLO
student@selab-19:~/Desktop/KaustavLABS4/PCAP LAB/LAB 02$ 
```

Q2

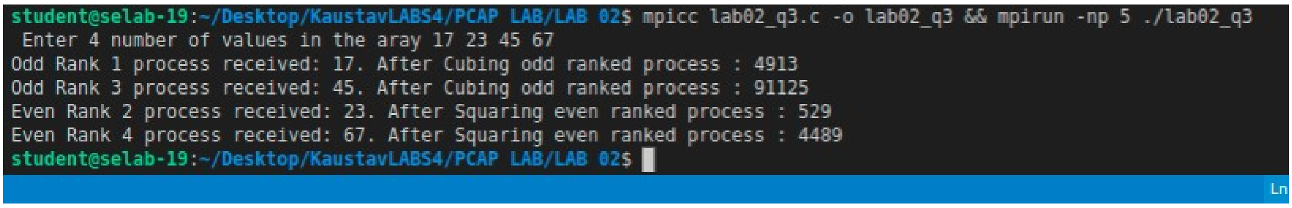
```
#include "mpi.h"
#include <stdio.h>

int main(int argc, char *argv[])
{
    int a, b, c;
    int rank, size;
    MPI_Status status;

    MPI_Init(&argc, &argv);
    MPI_Comm_rank(MPI_COMM_WORLD, &rank);
    MPI_Comm_size(MPI_COMM_WORLD, &size);

    if (rank == 0)
    {
        printf("Enter a value in the master process (rank %d) ", rank);
        scanf(" %d", &a);
        for (int i = 1; i < size; i++)
            MPI_Send(&a, 1, MPI_INT, i, 1, MPI_COMM_WORLD);
    }
    else
    {
        MPI_Recv(&b, 1, MPI_INT, 0, 1, MPI_COMM_WORLD, &status);
        fprintf(stdout, "Rank %d process received: %d \n", rank, b);
    }

    MPI_Finalize();
    return 0;
}
```



```
student@selab-19:~/Desktop/KaustavLABS4/PCAP LAB/LAB 02$ mpicc lab02_q2.c -o lab02_q2&& mpirun -np 4 ./lab02_q2
Enter a value in the master process (rank 0) 10
Rank 1 process received: 10
Rank 2 process received: 10
Rank 3 process received: 10
student@selab-19:~/Desktop/KaustavLABS4/PCAP LAB/LAB 02$ 
```

Q3

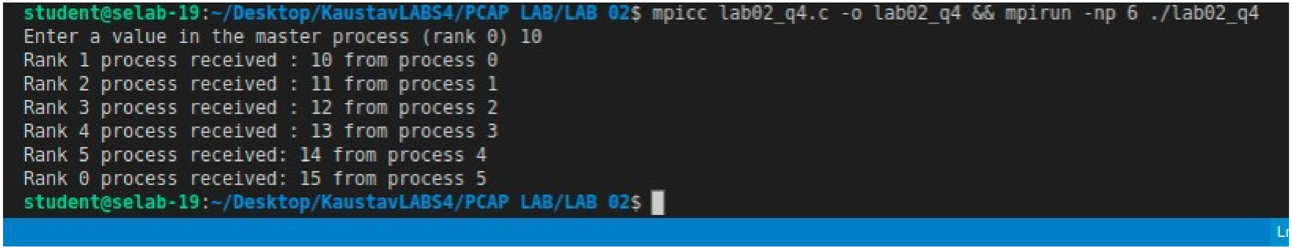
```c
#include "mpi.h"
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[])
{
    int x;
    int rank, size;
    MPI_Status status;

    MPI_Init(&argc, &argv);
    MPI_Comm_rank(MPI_COMM_WORLD, &rank);
    MPI_Comm_size(MPI_COMM_WORLD, &size);

    if (rank == 0)
    {
        int *arr = (int *)calloc(size, sizeof(int));
        printf(" Enter %d number of values in the aray ", size - 1);
        for (int i = 1; i < size; i++)
            scanf(" %d", &arr[i]);

        for (int i = 1; i < size; i++)
            MPI_Send(&arr[i], 1, MPI_INT, i, i, MPI_COMM_WORLD);
    }
    else
    {
        MPI_Recv(&x, 1, MPI_INT, 0, rank, MPI_COMM_WORLD, &status);

        if (rank % 2 == 0)
            fprintf(stdout, "Even Rank %d process received: %d. After Squaring
even ranked process : %d  \n", rank, x, x * x);
        else
            fprintf(stdout, "Odd Rank %d process received: %d. After Cubing odd
ranked process : %d  \n", rank, x, x * x * x);
        fflush(stdout);
    }

    MPI_Finalize();
    return 0;
}
```



```
student@selab-19:~/Desktop/KaustavLABS4/PCAP LAB/LAB 02$ mpicc lab02_q3.c -o lab02_q3 && mpirun -np 5 ./lab02_q3
 Enter 4 number of values in the aray 17 23 45 67
Odd Rank 1 process received: 17. After Cubing odd ranked process : 4913
Odd Rank 3 process received: 45. After Cubing odd ranked process : 91125
Even Rank 2 process received: 23. After Squaring even ranked process : 529
Even Rank 4 process received: 67. After Squaring even ranked process : 4489
student@selab-19:~/Desktop/KaustavLABS4/PCAP LAB/LAB 02$
```

Q4

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

int main(int argc, char *argv[])
{
    int a, b, c;
    int rank, size;
    MPI_Status status;

    MPI_Init(&argc, &argv);
    MPI_Comm_rank(MPI_COMM_WORLD, &rank);
    MPI_Comm_size(MPI_COMM_WORLD, &size);

    if (rank == 0)
    {
        printf("Enter a value in the master process (rank %d) ", rank);
        scanf(" %d", &a);
        MPI_Send(&a, 1, MPI_INT, 1, size, MPI_COMM_WORLD);

        MPI_Recv(&a, 1, MPI_INT, size - 1, size, MPI_COMM_WORLD, &status);
        a = a + 1;
        fprintf(stdout, "Rank %d process received: %d from process %d  \n",
rank, a, size - 1);
        fflush(stdout);
    }
    else if (rank != size - 1)
    {
        MPI_Recv(&b, 1, MPI_INT, rank - 1, size, MPI_COMM_WORLD, &status);

        fprintf(stdout, "Rank %d process received : %d from process %d \n",
rank, b, rank - 1);
        fflush(stdout);

        b = b + 1;

        MPI_Send(&b, 1, MPI_INT, rank + 1, size, MPI_COMM_WORLD);
    }
    else
    {
        MPI_Recv(&c, 1, MPI_INT, size - 2, size, MPI_COMM_WORLD, &status);
        fprintf(stdout, "Rank %d process received: %d from process %d \n", rank,
c, rank - 1);
        MPI_Send(&c, 1, MPI_INT, 0, size, MPI_COMM_WORLD);
        fflush(stdout);
    }

    MPI_Finalize();
    return 0;
}
```



```
student@selab-19:~/Desktop/KaustavLABS4/PCAP LAB/LAB 02$ mpicc lab02_q4.c -o lab02_q4 && mpirun -np 6 ./lab02_q4
Enter a value in the master process (rank 0) 10
Rank 1 process received : 10 from process 0
Rank 2 process received : 11 from process 1
Rank 3 process received : 12 from process 2
Rank 4 process received : 13 from process 3
Rank 5 process received: 14 from process 4
Rank 0 process received: 15 from process 5
student@selab-19:~/Desktop/KaustavLABS4/PCAP LAB/LAB 02$ 
```