

LAB-10

-Aayush kumar Singh

19BCE1113

1.

CODE

```
lab-10_1.c
1 #include <mpi.h>
2 #include <stdio.h>
3 int main(int argc, char **argv)
4 {
5     int rank;
6     int a,b,send_data;
7     const int root = 0;
8     MPI_Init(&argc, &argv);
9     MPI_Comm_rank(MPI_COMM_WORLD, &rank);
10    int world_size;
11    MPI_Comm_size(MPI_COMM_WORLD, &world_size);
12    if (rank != root)
13    {
14        b=2;
15        MPI_Recv(&a, 1, MPI_INT, rank-1, 0, MPI_COMM_WORLD, MPI_STATUS_IGNORE);
16        printf("Process [P%d]: received data %d\n", rank, a);
17        send_data=b;
18    }
19    else{
20        a=1;
21        send_data=a;
22    }
23    printf("Process p[%d]: sent data %d\n",rank,send_data);
24    MPI_Send(&send_data, 1, MPI_INT, (rank+1)%world_size, 0, MPI_COMM_WORLD);
25    if(rank==root){
26        MPI_Recv(&b, 1, MPI_INT, 1, 0, MPI_COMM_WORLD, MPI_STATUS_IGNORE);
27        printf("Process [P%d]: received data %d\n", rank, b);
28    }
29    MPI_Finalize();
30    return 0;
31 }
32
```

OUTPUT

```
2. 192.168.56.102 (aayush)
aayush@aayush-VirtualBox:~$ mpicc -o lab-10_1 lab-10_1.c
aayush@aayush-VirtualBox:~$ mpirun -np 2 ./lab-10_1
Process p[0]: sent data 1
Process [P1]: received data 1
Process p[1]: sent data 2
Process [P0]: received data 2
aayush@aayush-VirtualBox:~$
```

2.

CODE

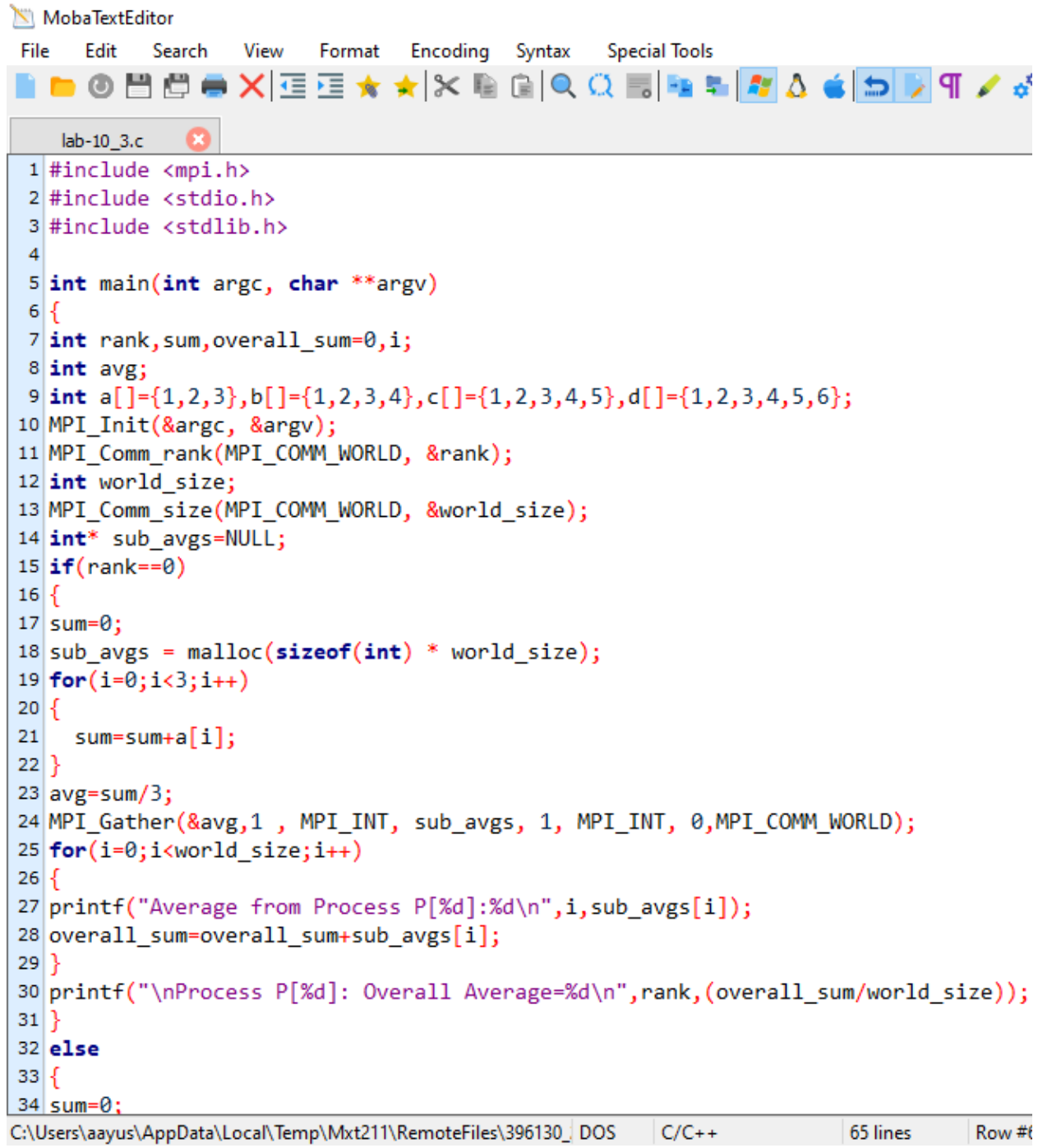
```
lab-10_2.c
1 #include <mpi.h>
2 #include <stdio.h>
3 int main(int argc, char **argv)
4 {
5     int rank, world_rank, rank1;
6     MPI_Init(&argc, &argv);
7     MPI_Comm_rank(MPI_COMM_WORLD, &world_rank);
8     int world_size;
9     MPI_Comm_size(MPI_COMM_WORLD, &world_size);
10    if (world_rank != 0) {
11        MPI_Recv(&rank1, 1, MPI_INT, world_rank - 1, 0, MPI_COMM_WORLD,
12        MPI_STATUS_IGNORE);
13        rank=world_rank;
14        printf("Process %d received rank %d from process %d\n", world_rank, rank1, world_rank - 1);
15        printf("sum of ranks=%d\n", (world_rank+rank1));
16    } rank=world_rank;
17    MPI_Send(&rank, 1, MPI_INT, (world_rank + 1) % world_size, 0, MPI_COMM_WORLD);
18    if (world_rank == 0) {
19        rank=world_rank;
20        MPI_Recv(&rank1, 1, MPI_INT, world_size - 1, 0, MPI_COMM_WORLD,
21        MPI_STATUS_IGNORE);
22        printf("Process %d received rank %d from process %d\n", world_rank, rank1, world_size - 1);
23        printf("sum of ranks=%d\n", (world_rank+rank1));
24    }
25    MPI_Finalize();
26 }
27
```

OUTPUT

```
2. 192.168.56.102 (aayush)
aayush@aayush-VirtualBox:~$ mpicc -o lab-10_2 lab-10_2.c
aayush@aayush-VirtualBox:~$ mpirun -np 3 ./lab-10_2
Process P1 received rank 0 from process P0
sum of ranks=1
Process P2 received rank 1 from process P1
sum of ranks=3
Process P0 received rank 2 from process P2
sum of ranks=2
aayush@aayush-VirtualBox:~$
```

3.

CODE



The screenshot shows the MobaTextEditor interface with a menu bar (File, Edit, Search, View, Format, Encoding, Syntax, Special Tools) and a toolbar. The active window is 'lab-10_3.c'. The code is a C program that calculates the average of an array 'a' across multiple processes using MPI. It includes headers for MPI, stdio, and stdlib. The main function initializes MPI, gets the world size, and calculates the average for each process. The status bar at the bottom shows the file path, DOS encoding, C/C++ language, 65 lines of code, and the current row number.

```
1 #include <mpi.h>
2 #include <stdio.h>
3 #include <stdlib.h>
4
5 int main(int argc, char **argv)
6 {
7     int rank, sum, overall_sum=0, i;
8     int avg;
9     int a[]={1,2,3}, b[]={1,2,3,4}, c[]={1,2,3,4,5}, d[]={1,2,3,4,5,6};
10    MPI_Init(&argc, &argv);
11    MPI_Comm_rank(MPI_COMM_WORLD, &rank);
12    int world_size;
13    MPI_Comm_size(MPI_COMM_WORLD, &world_size);
14    int* sub_avgs=NULL;
15    if(rank==0)
16    {
17        sum=0;
18        sub_avgs = malloc(sizeof(int) * world_size);
19        for(i=0; i<3; i++)
20        {
21            sum=sum+a[i];
22        }
23        avg=sum/3;
24        MPI_Gather(&avg, 1, MPI_INT, sub_avgs, 1, MPI_INT, 0, MPI_COMM_WORLD);
25        for(i=0; i<world_size; i++)
26        {
27            printf("Average from Process P[%d]:%d\n", i, sub_avgs[i]);
28            overall_sum=overall_sum+sub_avgs[i];
29        }
30        printf("\nProcess P[%d]: Overall Average=%d\n", rank, (overall_sum/world_size));
31    }
32    else
33    {
34        sum=0;
```

C:\Users\aaayus\AppData\Local\Temp\Mxt211\RemoteFiles\396130_ DOS C/C++ 65 lines Row #

```

lab-10_3.c
33 {
34 sum=0;
35 if(rank==1)
36 {
37 for(i=0;i<4;i++)
38 {
39 sum=sum+b[i];
40 }
41 avg=sum/4;
42 MPI_Gather(&avg,1 , MPI_INT, sub_avgs, 1, MPI_INT, 0,MPI_COMM_WORLD);
43 }
44 if(rank==2)
45 {
46 for(i=0;i<5;i++)
47 {
48 sum=sum+c[i];
49 }
50 avg=sum/5;
51 MPI_Gather(&avg,1 ,MPI_INT, sub_avgs, 1, MPI_INT, 0,MPI_COMM_WORLD);
52 }
53 if(rank==3)
54 {
55 for(i=0;i<6;i++)
56 {
57 sum=sum+c[i];
58 }
59 avg=sum/6;
60 MPI_Gather(&avg,1 , MPI_INT, sub_avgs, 1, MPI_INT, 0,MPI_COMM_WORLD);
61 }
62 }
63 MPI_Finalize();
64 }
65

```

OUTPUT

```

192.168.56.102 (aayush)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Quick connect...
/home/aayush/
Name
forkexample1.c
forkexample2.c
forksleep.c
guided
guided.c
lab-10_1
lab-10_1.c
lab-10_2
lab-10_2.c
lab-10_3
lab-10_3.c
lab-6_1
lab-6_1.c
lab-6_2
lab-6_2.c
lab-6_3
lab-6_3.c
lab-7_1

aayush@aayush-VirtualBox:~$ mpicc -o lab-10_3 lab-10_3.c
aayush@aayush-VirtualBox:~$ mpirun -np 3 ./lab-10_3
Average from Process P[0]:2
Average from Process P[1]:2
Average from Process P[2]:3
Process P[0]: Overall Average=2
aayush@aayush-VirtualBox:~$

```