**HRITIK BANSAL**
**CSE A 15**
**180905105**
**PP LAB**
**WEEK 4**

**Q1)**

```c
#include "mpi.h"
#include <stdio.h> #include <string.h>
int fact(int n)
{
                if (n <= 1) return 1;
                else return n * fact(n - 1);
}
int main(int argc, char *argv[])
{
                 int rank, size; int i = 0;
                int k = 0, fac = 1, ans[1000], sum = 0; int n;
                MPI_Init(&argc, &argv);
                MPI_Comm_rank(MPI_COMM_WORLD, &rank);
                // Set the error handler to MPI_ERRORS_RETURN
                MPI_Errhandler_set(MPI_COMM_WORLD, MPI_ERRORS_RETURN);
                MPI_Comm_size(MPI_COMM_WORLD, &size);
                //Get the error code on broadcasting; purposely fail this int error;
                error = MPI_Bcast(&fac, 1, MPI_INT, 4, MPI_COMM_WORLD);
if (error != MPI_SUCCESS)
{
                char s[100]; int len, class1;
                MPI_Error_string(error, s, &len); MPI_Error_class(error,
                &class1); fprintf(stderr, "Error description is %s", s);
                fflush(stderr);
                fprintf(stderr, "Error class is %d", class1); fflush(stderr);
}
for (i = 1; i <= rank + 1; i++)
{

                fac = fac * i;
    }
MPI_Scan(&fac, &k, 1, MPI_INT, MPI_SUM, MPI_COMM_WORLD);
 if (rank == size - 1)
{

                fprintf(stdout, "%d\n", k); fflush(stdout);
}

MPI_Finalize();
return 0;
}
```

**Output:**

```
hb@LAPTOP-70G8NNKV: ~/LabsSem6/PCAP/WEEK4                                    —   □   X

hb@LAPTOP-70G8NNKV:~/LabsSem6/PCAP/WEEK4$ mpicc -o p1 prg1.c
hb@LAPTOP-70G8NNKV:~/LabsSem6/PCAP/WEEK4$ mpirun -np 7 ./p1
--------------------------------------------------------------------
WARNING: Linux kernel CMA support was requested via the
btl_vader_single_copy_mechanism MCA variable, but CMA support is
not available due to restrictive ptrace settings.

The vader shared memory BTL will fall back on another single-copy
mechanism if one is available. This may result in lower performance.

  Local host: LAPTOP-70G8NNKV
--------------------------------------------------------------------
5913
[LAPTOP-70G8NNKV:02584] 6 more processes have sent help message help-btl-vader.txt / cma-permission-d
enied
[LAPTOP-70G8NNKV:02584] Set MCA parameter "orte_base_help_aggregate" to 0 to see all help / error mes
sages
hb@LAPTOP-70G8NNKV:~/LabsSem6/PCAP/WEEK4$
```

**Q2)**

```c
#include "mpi.h"

 #include <stdio.h>

int main(int argc, char *argv[])
{
int ierr,errclass,resultlen;

   char err_buffer[MPI_MAX_ERROR_STRING];
    int  rank,size,i;
    float rect ,pi;
MPI_Init(&argc,&argv);

MPI_Comm_rank(MPI_COMM_WORLD,&rank);
MPI_Comm_size(MPI_COMM_WORLD,&size);
MPI_Errhandler_set(MPI_COMM_WORLD,MPI_ERRORS_RETURN);
float x = (rank+0.5)/size; float x2 = x*x;
rect = (4/(1+x2))*(1/(float)size);
        ierr =MPI_Reduce(&rect,&pi,1,MPI_FLOAT,MPI_SUM,0,MPI_COMM_WORLD );
         if (ierr != MPI_SUCCESS) {
          MPI_Error_class(ierr,&errclass);
         if (errclass== MPI_ERR_RANK) {
         fprintf(stderr,"Invalid rank used in MPI send call\n");
MPI_Error_string(ierr,err_buffer,&resultlen);      printf("%s\n",err_buffer);
    MPI_Abort(MPI_COMM_WORLD,ierr);
    }
  }

       if(rank==0){
        printf("Val of pi  = %f\n",pi);
}

MPI_Finalize();
        return 0; }
```

**Output:**

```
hb@LAPTOP-70G8NNKV:~/LabsSem6/PCAP/WEEK4$ mpicc -o p2 prg2.c
hb@LAPTOP-70G8NNKV:~/LabsSem6/PCAP/WEEK4$ mpirun -np 8 ./p2
--------------------------------------------------------------------------
WARNING: Linux kernel CMA support was requested via the
btl_vader_single_copy_mechanism MCA variable, but CMA support is
not available due to restrictive ptrace settings.

The vader shared memory BTL will fall back on another single-copy
mechanism if one is available. This may result in lower performance.

   Local host: LAPTOP-70G8NNKV
--------------------------------------------------------------------------
Val of pi = 3.142895
```

**Q3)** #include "mpi.h" #include
<stdio.h>

```c
int main(int argc, char *argv[])
{
int ierr,errclass,resultlen;

    char err_buffer[MPI_MAX_ERROR_STRING];
int rank,size,i,j,a[3][3],b[3],key,count,countsum; MPI_Init(&argc,&argv);

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

        if(rank==0){ printf(" Enter values in 3x3 matrix:\n"); for(int
                i=0;i<3;++i){ for(int j=0;j<3;++j){ scanf("%d",&a[i][j]);
                        }
                }
                printf(" Enter ele to search:\n");
                scanf("%d",&key);
}
        ierr = MPI_Bcast(&key, 1, MPI_INT, 0, MPI_COMM_WORLD);
        if (ierr != MPI_SUCCESS) {      MPI_Error_class(ierr,&errclass);        if (errclass==
MPI_ERR_RANK) {
          fprintf(stderr,"Invalid rank used in MPI send call\n");
MPI_Error_string(ierr,err_buffer,&resultlen);      printf("%s\n",err_buffer);
    MPI_Abort(MPI_COMM_WORLD,ierr);
    }
  }
        ierr=MPI_Scatter(a,3,MPI_INT,b,3,MPI_INT,0,MPI_COMM_WORLD);
        if (ierr != MPI_SUCCESS) {      MPI_Error_class(ierr,&errclass);
         if (errclass== MPI_ERR_RANK) {
         fprintf(stderr,"Invalid rank used in MPI send call\n");
MPI_Error_string(ierr,err_buffer,&resultlen);
        printf("%s\n",err_buffer);
    MPI_Abort(MPI_COMM_WORLD,ierr);
    }
  }


        for(int i=0;i<3;++i){ if(key==b[i]){
                        ++ count;
                }
}
```

```
        ierr=MPI_Reduce(&count,&countsum,1,MPI_INT,MPI_SUM,0,MPI_COMM_WORLD );
         if (ierr != MPI_SUCCESS) {
          MPI_Error_class(ierr,&errclass);
          if (errclass== MPI_ERR_RANK) {
          fprintf(stderr,"Invalid rank used in MPI send call\n");        MPI_Error_string(ierr,err_buffer,&resultlen);
          printf("%s\n",err_buffer);
     MPI_Abort(MPI_COMM_WORLD,ierr);
      }
  }


        if(rank==0){
         printf("total no of occurances of %d  = %d\n",key,countsum);
}


        MPI_Finalize(); return 0; }
```

**Output:**

```
hb@LAPTOP-70G8NNKV:~/LabsSem6/PCAP/WEEK4$ mpicc -o p3 prg3.c
hb@LAPTOP-70G8NNKV:~/LabsSem6/PCAP/WEEK4$ mpirun -np 6 ./p3
--------------------------------------------------------------------------
WARNING: Linux kernel CMA support was requested via the
btl_vader_single_copy_mechanism MCA variable, but CMA support is
not available due to restrictive ptrace settings.

The vader shared memory BTL will fall back on another single-copy
mechanism if one is available. This may result in lower performance.

  Local host: LAPTOP-70G8NNKV
--------------------------------------------------------------------------
 Enter values in 3x3 matrix:
[LAPTOP-70G8NNKV:02657] 5 more processes have sent help message help-btl-vader.txt / cma-permission-d
enied
[LAPTOP-70G8NNKV:02657] Set MCA parameter "orte_base_help_aggregate" to 0 to see all help / error mes
sages
1
2
3
4
5
6
7
8
9
 Enter ele to search:
5
total no of occurances of 5 = 1
```

**Q4)** #include "mpi.h"

```c
#include <stdio.h>
int main(int argc, char *argv[]) {
int ierr,errclass,resultlen;

  char err_buffer[MPI_MAX_ERROR_STRING];
 int rank,size,i,j,a[4][4],b[4],c[4],count=0,countsum;

 MPI_Init(&argc,&argv);

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

        if(rank==0){
        printf(" Enter values in 4x4 matrix:\n");
        for(int i=0;i<4;++i){ for(int j=0;j<4;++j){ scanf("%d",&a[i][j]);
                        }
                } }



        ierr = MPI_Scatter(a,4,MPI_INT,b,4,MPI_INT,0,MPI_COMM_WORLD);
        if (ierr != MPI_SUCCESS) {
         MPI_Error_class(ierr,&errclass);
        if (errclass== MPI_ERR_RANK) {
          fprintf(stderr,"Invalid rank used in MPI send call\n");
MPI_Error_string(ierr,err_buffer,&resultlen);
         printf("%s\n",err_buffer);
    MPI_Abort(MPI_COMM_WORLD,ierr);
    }
  }

ierr=MPI_Scan(b,c,4,MPI_INT,MPI_SUM,MPI_COMM_WORLD );
if (ierr != MPI_SUCCESS) {     MPI_Error_class(ierr,&errclass);
 if (errclass== MPI_ERR_RANK) {
 fprintf(stderr,"Invalid rank used in MPI send call\n");
 MPI_Error_string(ierr,err_buffer,&resultlen);
  printf("%s\n",err_buffer);
   MPI_Abort(MPI_COMM_WORLD,ierr);
    }
  }
        if(rank==0){
        printf(" output 4x4 matrix:\n");
}
printf(" process %d :",rank); for(int i=0;i<4;++i){
 printf("%d ",c[i]);
}
printf("\n");


MPI_Finalize();
return 0;
}
```

**Output:**

```
hb@LAPTOP-70G8NNKV:~/LabsSem6/PCAP/WEEK4$ mpicc -o p4 prg4.c
hb@LAPTOP-70G8NNKV:~/LabsSem6/PCAP/WEEK4$ mpirun -np 4 ./p4
--------------------------------------------------------------------
WARNING: Linux kernel CMA support was requested via the
btl_vader_single_copy_mechanism MCA variable, but CMA support is
not available due to restrictive ptrace settings.

The vader shared memory BTL will fall back on another single-copy
mechanism if one is available. This may result in lower performance.

  Local host: LAPTOP-70G8NNKV
--------------------------------------------------------------------
 Enter values in 4x4 matrix:
2
34
4
45
5
6
44
3

[LAPTOP-70G8NNKV:02686] 3 more processes have sent help message help-btl-vader.txt / cma-permission-d
enied
[LAPTOP-70G8NNKV:02686] Set MCA parameter "orte_base_help_aggregate" to 0 to see all help / error mes
sages
2
1
1
2
33
34
4
3
 output 4x4 matrix:
 process 0 :2 34 4 45
 process 1 :7 40 48 48
 process 3 :42 75 53 53
 process 2 :9 41 49 50
```