Introduction to MPI - Worksheet 5 – Chapter 6
CHAPTER 6, NOT CHAPTER 5!
Collective Communications in MPI

1. Collective communication routines use the tag mechanism for associating calls in the same way as point to point Send and Receive.
   TRUE/FALSE

2. The user must ensure that (ALL/ONLY THE ROOT) _____ processor(s) execute the same collective communication calls and execute them in the same order.

3. Explain what is happening in section 6.2 demonstrating MPI_Bcast. What is the broadcast operation is doing here.

4. In the example program from section 6.2,
   A. What is the rank of the root process? _____
   B. Which process(es) see the MPI_Bcast call? _____
   C. Is there a "tag" parameter in the MPI_Bcast call? ____
   D. Which process(es) send the parameter 23? ____
   E. Which process(es) receive the parameter 23? _____

5. Describe the operation of MPI_Reduce

6. Why are there two buffers in the MPI_Reduce routine? (what is the purpose of each buffer)

7. What is the purpose of the fifth parameter, which can have values such as MPI_SUM, MPI_MAX, and MPI_PROD?

8. Run the example program from section 6.3. Does your output agree with

PE:7 MPI_PROD result is 3628800

NOTE: HOW MANY PROCESSES THIS IS RUN WITH (10!)

Explain the output of this program. What is this program doing?

9. MPI_Gather is a(n) "all-to-1" or "1-to-all" operation? _____

10. When MPI_GATHER is called, each process - EXCEPT the root process - sends the contents of its send buffer to the root process. TRUE/FALSE

11. Describe the difference between MPI_Gather and MPI_Allgather.

12. Explain what is happening with MPI_Scatter in Figure 6.5.

13. In the example of MPI_Scatter, section 6.5,
A. What process acts as "root"?
B. How many values are stored in the array?
C. What is the sendcount, how many values are "scattered" to each other process?
D. How many processors are needed here to send out all the values in the array?

14. What is the difference between MPI_Reduce and MPI_Allreduce?

15. We want to do a simple broadcast of variable abc[7] in processor 0 to the same location in all other processors of the communicator. What is the correct syntax of the call to do this broadcast?
A. MPI_Bcast (&abc[7], 1, MPI_REAL, 0, comm)
B. MPI_Bcast (&abc, 7, MPI_REAL, 0, comm)
C. MPI_Broadcast (&abc[7], 1, MPI_REAL, 0, comm)

16. Each processor has a local array a with 50 elements. Each local array is a slice of a larger global array. We wish to compute the average (mean) of all elements in the global array.

Our preferred approach is to add all of the data element-by-element onto the root processor, sum elements of the resulting array, divide, and broadcast the result to all processes. Which sequence of calls will accomplish this? Assume variables are typed and initialized appropriately.

A.
```
start = 0;
final = 49;
count = final - start + 1;
mysum = 0;
for ( i=start; i<=final; ++i ) mysum += a[i];
MPI_Reduce ( &mysum, \x{2211}, 1, MPI_REAL,
                    MPI_SUM, root, comm );
total_count = nprocs * count;
if ( my_rank == root ) average = sum / total_count;
MPI_Bcast ( &average, 1, MPI_REAL, root, comm );
```

B.
```
start = 0;
final = 49;
count = final - start + 1;
MPI_Reduce ( a, sum_array, count,
                    MPI_REAL, MPI_SUM, root, comm );
sum = 0;
for ( i=start, i<=final; ++i )
          sum += sum_array[i];
total_count = nprocs * count;
if ( my_rank == root ) average = sum / total_count;
MPI_Bcast ( &average, 1, MPI_REAL, root, comm );
```

C.
```
start = 0;
final = 49;
count = final - start + 1;
mysum = 0;
for ( i=start; i<=final; ++i ) mysum += a[i];
my_average = mysum / count;
MPI_Reduce ( &my_average, \x{2211}, 1, MPI_REAL,
                    MPI_SUM, root, comm );
if ( my_rank == root ) average = sum / nprocs;
MPI_Bcast ( &average, 1, MPI_REAL, root, comm );
```

17. Consider a communicator with 4 processes. How many total MPI_Send()'s and MPI_Recv()'s would be required to accomplish the following:
MPI_Allreduce (&a, &x, 1, MPI_REAL, MPI_SUM, comm);

A. 3
B. 4
C. 12
D. 16