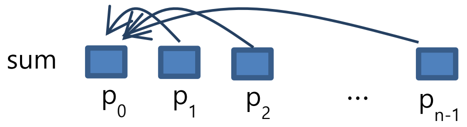
Lab6(1) – MPI(p2p)

e.g. greeting from p_0

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

main(int argc, char* argv[])
{
    int np, pid, dest;
    int tag = 0;
    char message[100];
    MPI_Status status;

    char processor_name[80];
    int name_len;

    MPI_Init(&argc, &argv);
    MPI_Comm_size(MPI_COMM_WORLD, &np);
    MPI_Comm_rank(MPI_COMM_WORLD, &pid);

    MPI_Get_processor_name(processor_name, &name_len);
    printf("%s, rank %d₩n", processor_name, pid);

    if (pid == 0) {
        sprintf(message, "Greeting from process %d", pid);
        for (dest = 1; dest < np; dest++) {
            MPI_Send(message, strlen(message)+1, MPI_CHAR, dest,
                    tag, MPI_COMM_WORLD);
        }
    }
    else {
        MPI_Recv(message, 100, MPI_CHAR, 0, tag,
                MPI_COMM_WORLD, &status);
        printf("%d: %s₩n", pid, message);
    }

    MPI_Finalize();
}
```

ex. Add all numbers (1, 2, .., N) onto p_0

1. Copy /home/course/hpc/sum.c into your working directory.
2. Complete sum.c to parallelize using MPI.
3. Test “mpiexec –n #processes sum N” assuming the executable file name is *sum*.
   For example, if N=100, sum prints “sum: 5050.0”.



After done, submit your program(sum.c).