

The objective of this exercise is to demonstrate the most fundamental MPI calls used in almost any MPI program. You are asked to write an SPMD (Single Process, Multiple Data) program where, again, each process checks its rank, and decides if it is the master (if its rank is 0), or a worker (if its rank is 1 or greater).

```
#include <stdio.h>
#include <string.h>
#include "mpi.h"
int main(int argc, char *argv[])
{
    int my_rank;    /* rank of process */
    int p;          /* number of processes */
    int source;     /* rank of sender */
    int dest;       /* rank of receiver */
    int tag = 0;    /* tags for messages */
    char message[100]; /* storage for message */
    MPI_Status status; /* return status for receive */
    /* start up MPI */
    MPI_Init(&argc, &argv);
    /* find out process rank */ MPI_Comm_rank(MPI_COMM_WORLD, &my_rank); /* find
out number of processes */
    MPI_Comm_size(MPI_COMM_WORLD, &p);
    if (my_rank != 0)
    {
        /* create message */
        sprintf(message, "Process %d: Hello World!", my_rank);
        dest = 0;
        /* use strlen+1 so that '\0' get transmitted */
        MPI_Send(message, strlen(message) + 1, MPI_CHAR,
            dest, tag, MPI_COMM_WORLD);
    }
    else
    {
        printf("Process %d: Hello World!\n ", p);
        for (source = 1; source < p; source++)
        {
            MPI_Recv(message, 100, MPI_CHAR, source, tag, MPI_COMM_WORLD, &status);
            printf("%s\n", message);
        }
    }
    /* shut down MPI */
    MPI_Finalize();
    return 0;
}
```

```
Jacobs-MacBook-Pro:Parallel-and-Distributed-Computing jacobjohn$ $HOME/opt/usr/local/bin/mpirun -np 12 ./mock_fat
Process 12: Hello World!
Process 1: Hello World!
Process 2: Hello World!
Process 3: Hello World!
Process 4: Hello World!
Process 5: Hello World!
Process 6: Hello World!
Process 7: Hello World!
Process 8: Hello World!
Process 9: Hello World!
Process 10: Hello World!
Process 11: Hello World!
Jacobs-MacBook-Pro:Parallel-and-Distributed-Computing jacobjohn$ █
```