



## Hpc program

1 message

Satyam Nalkar <satyamnalkar95@gmail.com>

Thu, 28 Aug, 2025 at 11:44 pm

To: satyamnalkar20@gmail.com <satyamnalkar20@gmail.com>

```
#include <iostream>
#include <mpi.h>
#include <vector>
using namespace std;

int main(int argc, char *argv[])
{
    MPI_Init(&argc, &argv);
    MPI_Status status;
    int rank, size;
    MPI_Comm_rank(MPI_COMM_WORLD, &rank);
    MPI_Comm_size(MPI_COMM_WORLD, &size);

    int n = 100, part = n / size, rem = n % size;
    part += (rank < rem ? 1 : 0);

    if (rank == 0)
    {
        // cout << "Part : " << part << endl;
        // cout << "Remainder : " << rem << endl;

        vector<int> A;
        for (int i = 1; i <= n; i++) // generating number from 1 to 100
            A.push_back(i);

        int offset = part; // index from which data to be sent

        // cout << "start location: " << offset << endl;
        // cout << endl;

        for (int i = 1; i < size; i++)
        {
            int count = part;
            if(rem != 0)
                count -= (i >= rem ? 1 : 0); // calculating the size of the data to be sent
            MPI_Send(A.data() + offset, count, MPI_INT, i, 1, MPI_COMM_WORLD);
            offset += count; // incrementing index of sending location
        }

        cout << endl;

        // debugging purpose
        // cout << "Rank " << rank << endl;
        // for (int i = 0; i < part; i++)
        //     cout << A[i] << " ";

        // cout << endl;

        int lSum = 0;
```

```

for (int i = 0; i < part; i++) // calculating local sum for 0
lSum += A[i];

cout << "Local Sum on 0: " << lSum << endl;
cout << endl;

for (int i = 1; i < size; i++)
{
int sum = 0;
MPI_Recv(&sum, 1, MPI_INT, i, 2, MPI_COMM_WORLD, &status);
lSum += sum;
}

cout << "Total sum is " << lSum << endl;
}
if (rank != 0)
{
vector<int> B(part);
MPI_Recv(B.data(), part, MPI_INT, 0, 1, MPI_COMM_WORLD, &status);

// cout << endl;
// cout << "Rank " << rank << endl;
// for (int i = 0; i < part; i++)
// cout << B[i] << " ";

// cout << endl;

int lSum = 0;
for (int i = 0; i < part; i++) // calculating local sum for other processes
lSum += B[i];

cout << "Local Sum on " << rank << " : " << lSum << endl;
cout << endl;

MPI_Send(&lSum, 1, MPI_INT, 0, 2, MPI_COMM_WORLD);
}
MPI_Finalize();
return 0;
}

```