Weicong Feng

Professor Feng Gu

CSC 429

03/07/2019

Lab2-II

// bcastSum.c

#include <mpi.h>

#include <stdio.h>

main(int argc, char\*\* argv)

{

int rank,size,i,j,k,localsum=0, globalsum;

int arr[1000];

MPI\_Init(&argc,&argv);

MPI\_Status status;

MPI\_Comm\_rank(MPI\_COMM\_WORLD, &rank);

MPI\_Comm\_size(MPI\_COMM\_WORLD,&size);

int brr[1000/size];

if(rank==0){

for (i=0;i<1000;i++)

{

arr[i]= i+1;

}

}

MPI\_Bcast(arr, 1000, MPI\_INT, 0, MPI\_COMM\_WORLD);

for(i=(1000/size)\*rank; i<(1000/size)\*(rank+1);i++)

{

localsum+=arr[i];

}

MPI\_Reduce(&localsum, &globalsum, 1, MPI\_INT, MPI\_SUM, 0, MPI\_COMM\_WORLD);

if(rank==0){

printf("The final sum is %d.",globalsum);

}

else

{

printf("localsum from processor %d is %d\n",rank,localsum);

}

MPI\_Finalize();

}

// mpi.sh

#!/bin/bash

#BATCH --job-name slurm\_mpi

#SBATCH --nodes 2

#SBATCH --ntasks 10

#SBATCH --ntasks-per-node 16

#SBATCH --mem 20000

#SBATCH --partition partedu

cd $SLURM\_SUBMIT\_DIR

echo 0 > cap

/opt/openmpi/bin/mpirun -np 10 ./bcastSum

