CS406-531 Homework 3 Deadline: May 30, 2021 – 23:59

In this homework, you will implement a basic version of the all-to-all broadcast operation of MPI by using pairwise communication primitives MPI_Send and MPI_Receive. Please check the first 20-25 slides of the accompanying slide set.

Your function will be used to broadcast a given number of integers and the number of processors will always be a power of 2. You can find the function signature and the testing environment in the accompanying source code file **main.cpp.**

Your performance will be taken into account while your homework will be graded (extra bonus points will be given to the top performance).

Here is a sample compilation and execution on gandalf.sabanciuniv.edu.

```
[kaya@gpu02 ~]$ module load gcc/8.2.0
[kaya@gpu02 ~]$ module load mvapich2-psm-x86_64
[kaya@gpu02 ~]$ mpicc main.cpp  -O3 -lm

[kaya@gpu02 ~]$ mpirun -np 8 -host localhost ./a.out 4 1
MPI: elapsed time is 0.000078 - 112.000000 = 112.000000
0 0 0 0 1 1 1 1 2 2 2 2 3 3 3 3 4 4 4 4 5 5 5 5 6 6 6 6 7 7 7 7
MPI: elapsed time is 0.000001 - 0.000000 = 112.000000
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

[kaya@gpu02 ~]$ mpirun -np 8 -host localhost ./a.out 50000 0
MPI: elapsed time is 0.001044 - 1400000.000000 = 1400000.000000
MPI: elapsed time is 0.000001 - 0.000000 = 1400000.000000

[kaya@gpu02 ~]$ mpirun -np 128 -host localhost ./a.out 50000 0
MPI: elapsed time is 2.940348 - 406400000.000000 =
406400000.000000
MPI: elapsed time is 0.020993 - 0.000000 = 406400000.000000
```

The program runs with the number of integers (message sizes) as the first parameter and a debug flag (0/1) as the second. The flag is there for you to test if your algorithm works correctly.

If the debug flag is 1, the program outputs the last processes receive buffer. If the debug flag is 0 then for both algorithms (the default one and your

implementation), the elapsed time with the sum of the integers in the received buffer is printed along with the value it should be.

You need to submit an extra (.pdf) file that describes the approach you used (at least a few sentences about how you implemented the protocol). The file also should report the timings you get with sequential 4, 16, 32, and 128 processors and various message sizes (number of integers).

So in short, be careful about the performance and do not only submit a cpp file as your HW. Your file name should be in the form **surname_name.zip** (e.g., **sinir_zihni.zip**, please DO NOT submit a rar file). A report with the details above must be in your compressed submission file.

Good Luck.

Kamer Kaya