

## CS336 : Parallel & Distributed processing

### Project 9 Report

#### Summary of tasks

For this project the main aim was to write straight-forward MPI (message passing interface) code and run it on NSCC to count the number of 3s appearing in the given array.

#### Tasks

2. I created count\_3s which allocated an array of randomly generated integers between 0 and 9 in the root process, scattered them to the rest of the processors, then counted the number of 3's in the array in each process, and then reduced the counts from the different processes.

3. Running my code with a size of 160, I got this result:

```
[mkgamedz@n22 ~/proj09]$ /usr/lib64/openmpi/bin/mpirun -np 16 --hostfile ~/hostfile.txt count_3s 160
** startup time 0.000004
starting problem details: size=160, numtasks=16
** filling array, took 0.000007s

** startup time 0.000001 rank=0,time=0.000000s
** clean-up time 0.000001 rank=1,time=0.000002s
** MPI total run time 0.316554
** startup time 0.000001 rank=2,time=0.000002s
** clean-up time 0.000001 rank=3,time=0.000002s
** MPI total run time 0.314700
** startup time 0.000001 rank=4,time=0.000002s
** clean-up time 0.000001 rank=5,time=0.000002s
** MPI total run time 0.311092
** startup time 0.000002 rank=6,time=0.000002s
** clean-up time 0.000001 rank=7,time=0.000002s
** MPI total run time 0.309054
** startup time 0.000001 rank=8,time=0.000002s
** clean-up time 0.000001 rank=9,time=0.000002s
** MPI total run time 0.307203
** startup time 0.000001 rank=12,time=0.000002s
** clean-up time 0.000002 rank=14,time=0.000002s
** MPI total run time 0.305248
** startup time 0.000001 rank=15,time=0.000002s
** clean-up time 0.000001 rank=10,time=0.000002s
** MPI total run time 0.301771
** startup time 0.000001 rank=11,time=0.000002s
** clean-up time 0.000001 rank=13,time=0.000003s
** MPI total run time 0.298266
** startup time 0.000002
** final count time on root process (rank0), took 0.000000s
count=12, time=0.000542s
sequential count took 0.000002s
actual count should be=12
** clean-up time 0.000001
** MPI total run time 0.318865
** clean-up time 0.000001
** MPI total run time 0.303779
** startup time 0.000001
** clean-up time 0.000001
** MPI total run time 0.298266
** startup time 0.000002
** clean-up time 0.000001
** MPI total run time 0.294662
[mkgamedz@n22 ~/proj09]$
```

4. Timing the different sections of the code, this was my result:

problem size	time (s)		filling array	rank count	final count	clean up time	sequention count	MPI total run time
	startup							
16	0.000001	0.000002	0.000002	0.000002	0	0.000002	0	0.296579
160	0.000001	0.000006	0.000006	0.000002	0	0.000001	0.000001	0.297034
1600	0.000004	0.000056	0.000056	0.000003	0	0.000001	0.000008	0.30877
16000	0.000006	0.000628	0.000628	0.000008	0	0.000001	0.000089	0.304078
160000	0.000012	0.005894	0.005894	0.000079	0	0.000001	0.000809	0.32565
1600000	0.000014	0.045339	0.045339	0.0006	0	0.000017	0.005829	0.374719
16000000	0.000016	0.411588	0.411588	0.005068	0	0.000255	0.058955	0.829152
160000000	0.000021	4.0138	4.0138	0.050804	0	0.001813	0.587462	5.248026
1600000000	0.000019	40.169775	40.169775	0.507872	0	0.016269	5.875443	49.687415

The MPI run-time does increase significantly till a couple orders of magnitude higher.

Extensions

None

Collaborators

I worked alone. Thanks to stephanie for help debugging.

.