Astronomy 598 Topics in Theoretical Astrophysics (Scientific Supercomputing)

Autumn 2015 Problem Set 5

Due Date, Nov 4, 2015

Use a command line terminal on a Linux machine or a mac machine to do this problem. Use vi for all your text editing in order to get familiar with vi. On hyak, you can get an interactive session using qsub -I.

Do all your work in a directory called

```
abcxyz_hw5
```

(replace abc with your first name and xyz with your last name). The directory should contain at least 4 files: sumrange, reducesumrange, myscript.pbs, README. (Do not include output files or text editor backup files). Use below command to tar the files and e-mail me your tar file with e-mail subject line ASTRO 598 HW5:

```
tar -cvf abcxyz_hw5.tar abcxyz_hw5
```

This problem is about distributing work to all the cores in a "map" phase and then collecting all the work in a "reduce" phase. This is a very common case in many applications of parallel computing. (In real life, you would also put checkpointing in the code. However, to keep things simple, we do not consider checkpointing here since we already covered checkpointing in HW4.)

(a) Write a program `sumrange` which takes three parameters: a starting number `n` and an ending number `m` for the summation and an output file number `k`. For example, below command should add all the numbers from `n` to `m` (inclusive) and then put the final sum in file `out.k`. Here `n, m, k` are positive integers.

```
sumrange n m k
```

(b) Write a program `reducesumrange` which sequentially opens files `out.p` to `out.q`, reads the number, sums all the numbers and produces a output file `out.final` which contains the total sum.

(c) Write a PBS script `myscript.pbs` like below. Change abc to your hyak group, change xyz to your userid. (Change ppn=16 to ppn=12 or to ppn=8 if your group has only 12 core nodes or only 8 core nodes. If you change ppn to 12 or 8 then make the appropriate changes to below script so that you still add all the numbers from 1 to 16000) Here the 16 calls to `sumrange` are the "map" phase and the call to `reducesumrange` is the "reduce" phase.

```
#!/bin/sh
#PBS -N "hw_job"
#PBS -d /gscratch/abc/xyz
#PBS -l nodes=1:ppn=16,mem=22gb,walltime=00:10:00
#PBS -M xyz@u.washington.edu
#PBS -m abe

sumrange 1 1000 1 &
sumrange 1001 2000 2 &
sumrange 2001 3000 3 &
sumrange 3001 4000 4 &
sumrange 4001 5000 5 &
sumrange 5001 6000 6 &
sumrange 6001 7000 7 &
sumrange 7001 8000 8 &
sumrange 8001 9000 9 &
sumrange 9001 10000 10 &
sumrange 10001 11000 11 &
sumrange 11001 12000 12 &
sumrange 12001 13000 13 &
sumrange 13001 14000 14 &
sumrange 14001 15000 15 &
sumrange 15001 16000 16 &
wait
reducesumrange 1 16
```

(d) Submit your PBS script using below command:

```
qsub myscript.pbs
```

(e) After the job completes, verify that the output in `out.final` is correct

(f) The `wait` command in the PBS script is an example of a barrier in parallel computing. Why do we need to issue the `wait` command before calling the `reducesumrange` program? Write the answer in your README file.

(g) Write instructions on how to run `sumrange`, `reducesumrange`, `myscript.pbs` in your README file.