Prerequisites and System Information for our Parallel Programming Course

Tim Mattson

Intel Corp.

timothy.g.mattson@ intel.com

All materials are on github. To download them:

git clone https://github.com/tgmattso/ParProgCourse.git

1

Materials and programming languages

- All materials (slides and exercises) are on github. Download the course materials onto your laptop:

  git clone https://github.com/tgmattso/ParProgCourse.git

- The course will be taught in C.
  - If you do not know C, we have a file (learningC.c) in the Exercises directory that you can use to learn the small subset of C we will use in this course.
  - Compile and run this learningC program as:

    gcc learningC.c

    ./a.out

- You need a compiler that includes OpenMP and a version of MPI that works with that compiler.
  - We recommend the gnu compilers and MPIch or OpenMPI.

Note: Depending on time, we might cover MPI with lectures, not hands-on. So don't worry if you can't load MPI onto your system.

Systems

- You can do this course on your laptop if you have C compilers with OpenMP and MPI on your laptop.
  - I prefer using my laptop … so everything I need to continue learning is available after the course.
- Or you can use your laptop to log in to the provided cluster
  - This has the advantage of a better CPU for running parallel programs and actual distributed memory nodes to run MPI programs on.

Warning: Xcode may rename gcc to Apple's clang compiler.

You may need to load a real, gcc compiler onto your laptop.

To test, put the line #include <omp.h>
in a C program and see if it will compile with the –fopenmp command line option

OpenMP Compilers on Apple Laptops: MacPorts

- To use OpenMP on your Apple laptop:
  - Download Xcode. Be sure to choose the command line tools that match your OS.
  - Install MacPorts (if you haven't already … use the installer for your OS from macports.org).

```
sudo port selfupdate
```

Update to latest version of MacPorts

```
sudo port install gcc9
```

Grab version 9 gnu compilers (5-10 mins)

```
port select --list gcc
```

List versions of gcc on your system

```
sudo port select --set gcc mp-gcc9
```

Select the mp enabled version of the most recent gcc release

```
gcc -fopenmp hello.c
```

Test the installation with a simple program

These directions are for gcc version 9. You should use the most recent stable release … all gcc compilers have OpenMP, there is nothing special about version 9.

MPIch library on Apple Laptops: MacPorts

- To use MPI on your Apple laptop:
  - Download Xcode. Be sure to choose the command line tools that match your OS.
  - Install MacPorts (if you haven't already … use the installer for your OS from macports.org).

```
sudo port selfupdate
```

Update to latest version of MacPorts

```
sudo port install mpich-gcc9
```

Graph the library that matches the version of your gcc compiler.

```
mpicc hello.c
mpiexec -n 4 ./a.out
```

Test the installation with a simple program

Preliminaries: Using the PSFC GPU cluster

- Connect to your MIT VPN account
- Logon to the GPU cluster:

  ssh <<login_name>>@ gpu.psfc.mit.edu

- Copy the Exercises to your home directory:

  cp –r Exercises /home<</home/login_name>>

- Login to one of the cluster nodes … pick your node based on the first letter of your first name:

  ssh <<node>>

Note: we may need to do additional load balancing among the nodes once we see the actual distribution

<<node>>	First letter of last name
gpu-v100s-01	A, B, C, D
gpu-v100s-03	E, F, G, H
gpu-v100s-05	I, J, K, L
gpu-rtx6000-02	M, N, O
gpu-rtx6000-04	P Q, R, S
gpu-rtx6000-06	T, U, V, W, X, Y, Z