# High Performance Computing
# Winter 2017-2018

### Programming Assignment 0

**Objective:** The goal of this assignment is to set up the computational environment, get you familiar with programming under the Linux environment, and to get you started writing MPI code.

In this assignment you are to write the first MPI hello world program under linux platform. Follow the following steps and post any problems you get on the course group page.

1) In case you are new to Linux, open a shell window and create a folder where you will be working (ask lab assistance if you are new to this)

```
$ mkdir cs480
$ mkdir assignment1
$ cd assignment1
```

2) It is very likely that you have everything you need already installed, check that by writing on the command line:

```
$ mpic++
```

if you get command not found then it is either not in your path or you need to do some installation as follows. **If the command is found then skip to step 5**

The following instruction is for redhat based linux (redhat, centos, ..etc). For debian just do the same but google the name of the packages.

a) Check if the compiler gcc is installed

```
Redhat:  rpm -q gcc-c++
```

if not installed, do ( while connected to the internet)

```
Redhad: yum install gcc-c++
```

b) Check if openmpi is installed

```
$ rpm -q openmpi
```

if not do

```
redhad: sudo  yum install openmpi
        sudo yum install openmpi-devel

ubuntu: sudo apt-get install sudo apt-get install openmpi-bin libopenmpi-dev
```

c) Make sure the wrapper is in your path

```
$ which mpic++
```

if not add it to your path, by adding to your ~/.bashrc the line

```
PATH=$PATH:/usr/lib64/openmpi/bin/
```

save the file and sources it

```
$ source ~/.bashrc
```

now try

```
$ mpic++
```

if you still get command not found, please post your problem on the facebook group page

3) Using your favorite editor - I strongly suggest to use either vi or emacs, open a new file under the name hello.cpp. If emacs is not installed, install it by using

```
$ sudo yum install emacs
```

4) Copy the following hello world mpi program to the hello.cpp

```
#include <iostream>
#include <mpi.h>

using namespace std;
int main(int argc, char **argv) {
        MPI_Init(&arg,  &argv);
        cout << "Hello World " << endl;
        MPI_Finalize();
        return 0;
}
```

5) Compile the program, from the line command do

```
mpic++ -o hello hello.cpp
```

6) Fix any compilation errors you get until you get an executable (there is one error), try run the executable

```
    $ mpirun -n 2 ./hello
    $ mpirun -n 4 ./hello
```

7) If you get a shared library error when trying to run the executable, update LD_LIBRARY_PATH in your ~/.bashrc to include the path to the shared library.

8) Add code, so your program prints the following when you run it

```
$ mpirun -n2 ./hello
hello world from process 0 / 2
hello world from process 1 / 2
```

```
Use;   MPI_Comm_rank(MPI_COMM_WORLD, &rank);
        MPI_Comm_size(MPI_COMM_WORLD, &size);
```

9) Add code, so your program prints the processor name as well
Use: MPI_Get_processor_name(name, &len);

10) Add code, so that processes with even rank say hello world rank, while processes with odd ranks say Yello world. Like,

```
$ mpirun -n 4 ./hello
hello world from 0
Yello world from 1
hello world from 2
Yello world from 0
```