Name: Cicelia Siu
CS 219 Assignment #11
Purpose: Become more familiar with concurrency methods and issues
Points: 100 pts → Part A 20 pts
Part B 40 pts
Assignment:
Part A:
Answer the following questions: (20 pts, 10 pts each)

- Given a 100-processors multiprocessor system and an application that has 15% of the code parallelized, what is the approximate effective speed-up?

  SpeedUp(.15, 100) = 1/(1-.15)+(.15/100)
  = 1/(.85)+(.0015)
  = 1/.8515
  = **1.174**
- Given a 10-processors multiprocessor system and an application that has 85% of the code parallelized, what is the approximate effective speed-up?

  SpeedUp(.85, 10) = 1/(1-.85)+(.85/10)
  = 1/ (.15)+(.085)
  = 1/ .235
  = **4.255**

Part B:
The value of π can be calculated by numerical integration of the function, f(x)=4/(1+x^2)

The value represents the area under the curve of the function for an interval of 0.0 to 1.0. To find an approximation of the area, the interval is divided into some number of n subintervals and each are in the subinterval in calculated. Then all of the areas are added up to give the final result. So, the larger the value of n, the more accurate the value of π.

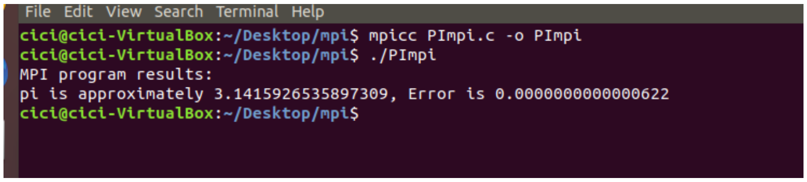
In order to speed the calculations, it is possible to split the computations into groups and send each group to a different processor or core. Thus, the computations can be performed in parallel. When done, the individual group sums can be combines into a final answer.

We will look at two different ways to accomplish the parallel computations; message passing (MPI) and shared memory (threading).

Compile and execute the provided programs; PImpi.c and PIpth.cpp. Compilation instructions are included in the comments. Review the code and examine the results.

For the PImpi.c program.
Answer the following questions (no more than 1-2 sentences each):

- include a copy of the results (cut-and-paste)



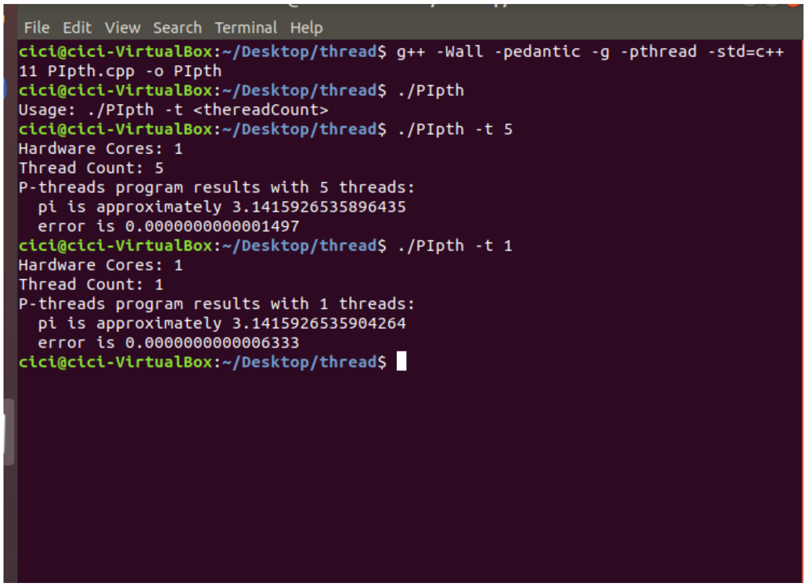
```
cici@cici-VirtualBox:~/Desktop/mpi$ mpicc PImpi.c -o PImpi
cici@cici-VirtualBox:~/Desktop/mpi$ ./PImpi
MPI program results:
pi is approximately 3.1415926535897309, Error is 0.0000000000000622
cici@cici-VirtualBox:~/Desktop/mpi$
```

- summarize basic approach to parallelism and the typical configuration
    - Approach to Parallelism: Distributed Processing
        - This programs runs by distributed processing, where parallel Computation occurs on different computers. Distributed processing uses Message Passing Interface (MPI library and function calls) to communicate the tasks and results between the computers.
- list several benefits of the distributed approach
    - No race conditions.
    - Varying capabilities with different computers
- list several dangers of the distributed approach
    - Communication Speed between computers is limited
    - relies on network (if slow, the process is slow)

For the PIpth.cpp program.

Answer the following questions (no more than 1-2 sentences each):

- include a copy of the results (cut-and-paste)



```
cici@cici-VirtualBox:~/Desktop/thread$ g++ -Wall -pedantic -g -pthread -std=c++
11 PIpth.cpp -o PIpth
cici@cici-VirtualBox:~/Desktop/thread$ ./PIpth
Usage: ./PIpth -t <thereadCount>
cici@cici-VirtualBox:~/Desktop/thread$ ./PIpth -t 5
Hardware Cores: 1
Thread Count: 5
P-threads program results with 5 threads:
  pi is approximately 3.1415926535896435
  error is 0.0000000000001497
cici@cici-VirtualBox:~/Desktop/thread$ ./PIpth -t 1
Hardware Cores: 1
Thread Count: 1
P-threads program results with 1 threads:
  pi is approximately 3.1415926535904264
  error is 0.0000000000006333
cici@cici-VirtualBox:~/Desktop/thread$ 
```

- summarize basic approach to parallelism and the typical configuration
    - Approach to Parallelism: Threaded Processing
        - This program runs on threaded processing where, parallel Computation occur on different cores of the same computer. Multiple threads, typically pthreads, communicate with each other to complete the task.
- list several benefits of the pthread approach
    - Interprocess communication speed is faster.
    - Lower overhead means light weight processes
- list several dangers of the pthread approach

- There is a possibility of a race conditions.
- Limited number of Cores available