Problem 1
Write a C/C++ program that will determine the time (in \(\mu\text{sec}\)) it takes to create (and then join) a Pthread. The thread should be an empty function (i.e., it just returns). The main routine should measure the time in \(\mu\text{sec}\) it takes to execute \(N\) calls of the sequence \texttt{pthread_create()} followed by \texttt{pthread_join()} and then compute the average time to do one sequence.

Run the experiment three (3) times, and submit:

a) What host and OS did you use for the measurements?

b) Your source code.

c) A single table showing your measurement results for each run and for the average for \(N = 100, 1000,\) and 10000. (That’s two columns (run, \(N\)) and nine rows). Note that these times should not include any I/O time.

d) What can you conclude about thread creation time versus process creation time (Homework 2)?

Note that on a CEC Solaris host, you will need to include the standard header \texttt{<pthread.h>} and use something like one of the following commands for compiling (i.e., include the library \texttt{pthread}):

```
gcc -o foo foo.c -lpthread
```

```
g++ -o foo foo.C -lpthread
```

Problem 2
Write a C/C++ program that will determine the time (in \(\mu\text{sec}\)) it takes to context switch between two threads that synchronize via a mutex. The main routine should create \(M\) threads and then wait for them to complete using a conditional signal. Each thread should loop \(N\) times around an empty critical section protected by a mutex. The last thread to finish should signal the main routine that it is done.

The main routine should measure the time \(T\) in \(\mu\text{sec}\) elapsed between the first call to \texttt{pthread_create()} and the last thread to finish and then compute the average time \(T/N\).

Run the experiment three (3) times, and submit:

a) What host and OS did you use for the measurements?

b) Your source code.

c) A single table showing your measurement results for each run, \(M\), and \(N\) where \(M = 2, 4, 8\) and \(N = 100, 1000, 10000\). (That’s three columns (run, \(M, N\)) and 27 rows.) Note that these times should not include any I/O time.

Problem 3
Stallings, Problem 4.2.

Problem 4
Stallings, Problem 4.4.