Course Mini-Project #2: Programming with Multiple Threads
Due date: March 1

1. Introduction

The objective of this mini-project is to implement a C/C++ module that uses multiple threads to compress an input data stream. It aims to help you gain hands-on experience of multithread programming, which could very much help your job hunting. Compression can be done by calling the standard zlib library (http://zlib.net), which is the core of the widely used GZIP and is available on almost all the major OS distributions. Each 4KB block in the input data stream should be compressed by zlib individually (i.e., all the 4KB blocks are compressed independently from each other). All the compressed blocks are in-order written to one output file (i.e., the 1st 4KB in the input stream is compressed and written to the output file as the 1st block, the 2nd 4KB in the input stream is compressed and written to the file right after the 1st block, etc.). You can use a big file (e.g., tens of GBs) as the source of the input data stream. Your code should use one thread to read data from the file, dispatch 4KB blocks to other worker threads for compression, receive compressed blocks from other worker threads, and write compressed blocks (in the correct order) to the output file.

2. Requirement

Your implementation should be able to support a configurable number of worker threads. In addition to the source code, your Github site should contain
   (1) Readme that clearly explains the structure/usage of your code
   (2) Experimental results that show the performance of your multi-threaded compression module under different number of worker threads
   (3) Analysis and conclusion

3. Additional Information

C/C++ does not have built-in support for multithreaded application, hence must rely on the underlying operating systems. Linux provides the pthread library to support multithreaded programming. You can find a very nice tutorial on pthread at https://computing.llnl.gov/tutorials/pthreads/. Microsoft also provides support for multi-thread programming (e.g., see https://docs.microsoft.com/en-us/windows/win32/procthread/multiple-threads). You are highly encouraged to program on Linux since Linux-based programming experience will help you most on the job market.