COMP 3500 Project #2

Maximum Points Possible: 100
Team Assignment (1..2 members per team)

There should be no collaboration among students (teams). A student/team shouldn't share any project code with any other student/team. Collaborations among students in any form will be treated as a serious violation of the University's academic integrity code.

Objectives: To learn the following.

1. Thread management
   a. Creation of threads.
   b. Synchronization of threads using semaphores.
2. Usage of POSIX PThread library.
3. Implement solution to Second Readers-Writers problem.
4. Read/Write text files using a C Program.
5. Passing command line parameters to a program

Instructions:

1. This project can be submitted individually, or in teams of two members only.
2. Program must be written in C / C++ programming language.
3. Your program must take the following as command line parameters.
   a. # of reader threads
   b. # of writer threads
4. Your program will be tested with varied number of reader/writer threads.
5. Make no assumptions regarding the number/type of threads.
6. Perform necessary error checking of command line parameters.

Part – 1 (45 Points)

Write a C program that includes following functionality:

1. (5 Points) Creates two types of threads: reader threads, and writer threads
2. (5 Points) All threads shared one integer variable. Exclusive access to the shared variable is implemented using semaphores.
3. (25 Points) Reader thread repeats the following 10 times. Note: There can be any number of reader threads simultaneously accessing the shared variable. However, no new reader thread must be allowed access if a write thread is waiting for access to the shared variable. Note: This is the **second readers-writers problem** discussed in class.
    a. Reads the shared integer value.
    b. Writes it to the output file.
    c. Sleeps for one second.
4. (10 Points) Writer thread repeats the following 10 times.
    a. Reads the shared integer value.
    b. Increments its value.
    c. Writes the old value and the new value to the output file.
    d. Sleeps for one second

Part – 2 (55 Points)

Provide a writeup including the following information:

1. (10 Points) What functionality needs to be included in the critical sections of reader and writer threads?
2. (10 Points) Briefly describe the solution for synchronization problem.
3. (10 Points) Describe the entry section and exit section of reader and writer threads.
4. (5 Points) What semaphore(s) are implemented?
5. (5 Points) What are the initialization value(s) for semaphore(s). Why? Provide reasoning.
6. (5 Points) Briefly describe the purpose of each semaphore (s) used?
7. (5 Points) Are there any additional shared variables used? If yes, describe their purpose.
8. (5 Points) Are there any situations when the implemented solution approach doesn’t work? Briefly describe with an example.