48450 Real Time Operating Systems

Assignment 2 (20 marks)

Deadline for submission: 23:59 PM, 12 April 2024

1. Introduction

This assignment involves the development of application programs with rea-time file reading and writing. You are required to create a program that applies several key concepts from the subject of 48450 Real-time Operating Systems. Submission will be marked based on their merits and may receive a mark less than the total of 20 if of modest quality. It is necessary to follow the technical report template and submit the assignment online by the specified due date.

All programs are implemented in C language.

This assignment is marked out of 20 and comprises 20% of the total score for this course

Please download a C file template from Canvas. You are required to use the downloaded the **template** as your baseline to develop the programs for your assignment. Failure to do so will result in 50% penalty score applied before the assignment is marked.

Note: This assignment is to be completed in groups with a maximum of two students. Alternatively, one student can also be a self-formed group. If a group consists of two students, both will receive the same mark. Each student is allowed to register once for joining a group only once. Any changes to the group arrangement should be approved by the subject coordinator.

2. Assignment details

Topic: Pipe, Threads, Semaphore and Shared Memory for Real Time File Reading/Writing and Screen Display – Mark 20

The program implementation involves using the concept of threads and pipes. Your program is required to create three threads (A, B and C) for reading data from one file (data.txt”) and writing the processed data to another file (“output.txt”).

Your program is required to achieve the following operation goals:

1. To read the “data.txt” and write the result into the “output.txt”. Your program needs to give these file names in the commend line such as “./you_porgram data.txt output.txt”.
2. Thread A writes one line of data from a given “data.txt” file to a pipe. Note: As shown in Figure 1, the “data.txt” text file contains two parts, the file header region and the content region.
3. Thread B reads data from the pipe and passes it to Thread C. **Note:** There are multiple ways to pass the data for the operation of Thread C. and you are required to find a suitable solution.
4. Thread C reads the data (line by line) from Thread B and determines whether the characters are from the **File header region** or from the **Content region** (see Figure 1). If the characters are not from the **File header region**, they should be written into a “output.txt” file. Otherwise, the characters should be discarded. **Note:** To detect the end of file header region, in Thread C, you are also required to identify the “end header” – **find a smart solution to address this issue!**

The three threads (A, B, C) run line by line sequentially (only one thread runs at a time while other two threads are blocked) and iteratively (process one line of data per each loop). When Thread A reaches to the end of the “data.txt” file, your program needs to handle and complete Threads B and C to finish the A->B->C sequential loop.

There is a global variable “int sum = 1” declared. When your program runs in Thread A, it will perform the calculation ‘sum = 2*sum’ five times. When running in Thread B, the program will pick up the sum Thread A and continue the with the calculation ‘sum = 3*sum’ three time. When running in Thread C, the program will pick up the sum from Thread B and proceed with the calculation ‘sum = sum – 5’ four times. Your program is required to print out the final ‘sum’ value on the screen.

Your program is required to include pipe, thread and semaphore whereby necessary. Please note: The three threads A, B and C must adopt a mutually exclusive strategy to achieve your program. Again, only the content region of data as shown in Figure 1 is written into the “output.txt” through the program. It is also required to use a ‘struct’ to pass the parameters to the threads. The “data.txt” file can be downloaded from Lab 4 folder of 48450 Module section on Canvas.

In addition to the above tasks, you are required to write a program that reads the information from shared memory and output the information from the memory to your PC/Laptop monitor.

You are required to do an experiment: If you do not apply semaphore, what will be the results by only using the three threads and the pipe? **You are required to summarise your analysis and observation** in your assignment report.

3. Assignment Deadline and Submission

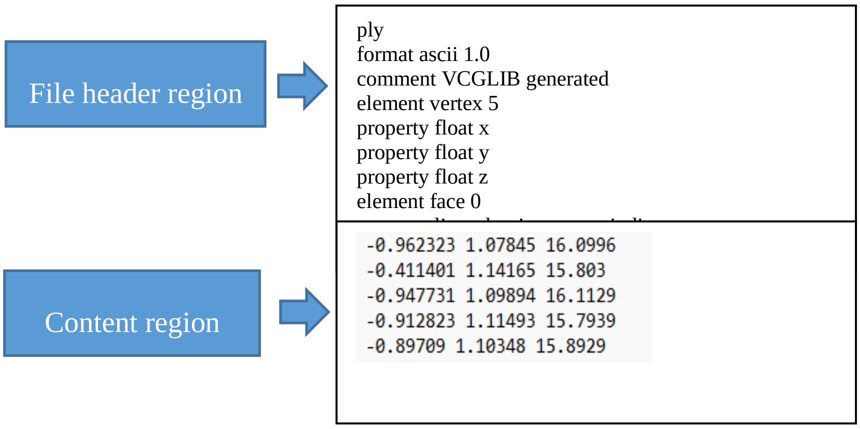
The deadline to submit this assignment is 23:59 PM, 12 April 2024

You are required to submit two formats of the assignment:

1. Upload your full assignment report.
2. Upload your ‘C’ code file

If you use Makefile for compiling your program, you are required to upload the Makefile files as well

Figure 1. The “data.txt” file



```
ply
format ascii 1.0
comment VCGLIB generated
element vertex 5
property float x
property float y
property float z
element face 0
```

```
-0.962323 1.07845 16.0996
-0.411401 1.14165 15.803
-0.947731 1.09894 16.1129
-0.912823 1.11493 15.7939
-0.89709 1.10348 15.8929
```