

# 48450 Real Time Operating Systems

## Assignment 2

**Deadline for submission: 23:59 PM, 6 May 2020**

### 1. Introduction

This assignment will involve the development of some application programs with real time file reading and writing. You are required to create a program that applies several key concepts of 48450 (RTOS) subject. A submission will be marked based on its merits and may be awarded a mark that is less than the total mark, if it's of modest quality.

All programs are implemented in C language.

**This assignment is currently marked out of 20 and comprises 20% of the total score for this course. However, there is a plan to change assessment weightings and group work (Max members = 2 students) that isn't yet approved**

### 2. Assignment details

#### Topic: Pipe, Threads and Semaphore for Real Time File Reading/Writing

The program implementation will involve 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 and writing the data to another file through the pipe-line concept. This is similar to an internet file transmission.

Your program is required to achieve the following operation goals:

1. Thread A writes one line of data from a given "data.txt" file to a pipe. Note: The "data.txt" text file contains two parts, the file header region and the content region. Each region is written with several lines.
2. Thread B reads data from the pipe. You need to consider a way to save the data in a 2D-array in the Thread B for the next operation in Thread C.
3. Thread C reads the data (line by line) from a 2D-array in Thread B and detects whether the line of data is from the file header region (see Figure 1). This assignment requires to write the data from the content region of "data.txt" into the "output.txt" only. Therefore, if the line of data is from the file header, it will be discarded.

The three threads (A, B, C) run 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, Threads B and C will then finish once the current loop is completed.

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 UTSONline.

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, 6 May 2020**

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

<pre>ply format ascii 1.0 comment VCGLIB generated element vertex 5 property float x property float y property float z element face 0 property list uchar int vertex_indices end_header</pre>	File header region
<pre>-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</pre>	Content region