

# 48450 Real Time Operating Systems

## Assignment 2 (20 marks)

**Deadline for submission: 23:59 PM, 10 May 2024**

### 1. Introduction

This assignment will involve developing application programs with real-time CPU scheduling and Memory management. You are required to create a program that applies several key concepts of 48450 (RTOS) subject. Submissions will be marked based on their merits and may receive a score of less than 25 score if they demonstrate modest quality.

This assignment is individually assessed. Students are not allowed to share their code and report. Please refer to the Academic Integrity section on page 10 of the subject outline for 48450

It is marked out of **25 points** and comprises 25% of the total score for this course. All programs are implemented in C language.

Please download the C file templates for Program\_1 and Program\_2 from Canvas. You must use the downloaded templates as your baseline for developing the programs for your assignment. Failure to do so will result in 50% penalty score applied before the assignment is marked.

### 2. Assignment details

#### CPU Scheduling, FIFOs (named Pipe), Memory management and Signals.

The program implementation will utilise the concepts of CPU scheduling, FIFOs and Signal concepts. Your program should consist of two parts: **Program\_1.c** and **Program\_2.c** files.

##### 2.1 Program\_1.c (13 points):

In Program\_1.c file, you are required to incorporate **CPU scheduling** and **FIFOs** (named Pipe). It should include two threads, **Threads 1 and 2**.

- **Thread 1:** This thread simulates CPU scheduling using **Round Robin (RR) with a time quantum of 4 milliseconds**. The burst time is measured in millisecond. Your program should calculate the *average waiting time and turn-around time* during **CPU scheduling**. After completion, it should define a FIFO and write these *average waiting time and turn-around time* to CPU memory through the FIFO. The input data for CPU scheduling are as follows:

Process ID	Arrive time	Burst time
1	8	10
2	10	3
3	14	7
4	9	5
5	16	4
6	21	6
7	26	2

- **Thread 2:** In this thread, your program is required to read the *average waiting time and turn-around time* from the memory through the **FIFOs** (named Pipe) as defined in **Thread 1**. Subsequently, your program should write this retrieved (received) data to a text file named "output.txt".

**Note:**

**1) You are required to draw a Gantt Chart to illustrate the schedules of processes 1 to 7 in your report.**

**2) Only the Round Robin scheduling is applied for Program\_1.c**

**3) For Program\_1.c:** the time quantum of RR and the name of output file ("output.txt") needs to be provided via the command line.

-----**Compile Program 1.c and run it**-----

## **2.2 Program\_2.c (9 points):**

In Program\_2.c file, you are required to incorporate **Memory Management and Signal**. Your program should simulate page-replacement for virtual memory management by using **First-In-First-Out (FIFO)** Algorithm.

In the **Program\_2.c**, your program needs to create **4 frames**, and the reference string (please refer to slide 28 of the online lecture) is as follows:

**7,0,1,2,0,3,0,4,2,3,0,3,0,3,2,1,2,0,1,7,0,1,7,5**

At each point when a reference string enters the frames, your program should check for page faults and output the current frame state of the frames, including the page fault number, on your console (screen), following the format illustrated in slide 28 of recorded lecture [1]. Once the entire reference string entering process is completed, your program should handle a "ctrl-c" signal. Therefore, upon running your program, it should wait for you to press the combined "ctrl-c" keys. Upon doing so, your program will print out the total number of fault pages on your console (Screen).

**Note: For Program\_2.c:** the number of frames need to be provided via command line.

-----**Compile Program 2.c and run it**-----

**2.3 Report (3 points):** You are required to write a report to summarise your observation.

## **3. Assignment Deadline and Submission**

**The deadline to submit assignment 3 is 23:59 PM, 10 May 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 need to upload the Makefile file

## **4. Reference**

[1] L7-Chapters 8 Main Memory & 9 Virtual Memory-1.pptx