



United International University
Course: Operating Systems (CSE 4509), Summer 2025
Class Test 2, Set A
Total Marks: 20, Time: 40 minutes

Name		ID		Section	
------	--	----	--	---------	--

1. Answer the following questions briefly and precisely. Each question contains 2 marks.
[5x2 = 10]

- a) In what kind of workload does STCF perform poorly, and why?
- b) Does STCF guarantee fairness among processes? Why or why not?
- c) What is the main motivation behind using multiple queues in MLFQ scheduling?
- d) What problem does MLFQ try to solve that simple priority scheduling cannot?
- e) Is it necessary for threads in a process to have separate copies of the program Executable?

2. Consider The **Reader–Writer problem** that deals with synchronizing access to a shared resource (e.g., a database) where multiple readers can read simultaneously, but **writers need exclusive access**. In this version, if there are waiting readers and a writer, the **readers are given priority**. Multiple readers can read at the same time. A writer can only write when **no reader is reading** and **no new reader is waiting**. You are given with the pseudocode of the reader thread, you need to implement the **writer thread**. **[10 marks]**

```
mutex          // global lock
readCount = 0   // number of active readers
writeCount = 0  // 0 or 1, indicates writer is active
okToWrite      // condition variable for writers
okToRead       // condition variable for readers
```

Reader Thread	Writer Thread
<pre> reader() { lock(mutex) // Wait if a writer is active while (writeCount > 0) { wait(okToRead, mutex) } readCount = readCount + 1 unlock(mutex) // ---- Critical Section ---- readDatabase() lock(mutex) readCount = readCount - 1 // If this was the last reader → allow writer if (readCount == 0) { signal(okToWrite) } unlock(mutex) } </pre>	