CS 314 Operating Systems
Spring 2024
Quiz #7 on March 19, 2024 (SOLUTIONS)

Your Last Three Digits: _____

(please do NOT write all of your student ID or your name)

Grade: _____

****

(1) How do threads reduce the high context switching overhead in processes?

Threads reduce the context-switching overhead by switching only TCBs (Thread Control Blocks) when a context-switching from a thread to another is performed.

The overhead is reduced since TCBs are smaller than PCBs (since TCBs contain only the information that is unique to each thread). Thus, switching TCBs (threads) is faster than switching PCBs (processes).

(2) As we discussed in the classroom, "threads" are introduced after many system programmers were using "processes" for multi-tasking (we even discussed that "threads" were introduced to avoid two problems in "processes"). After all, while "processes" and "threads" have many things in common (and "threads" seem to be better than "processes"). Then why do we still use "processes" (mention at least one reason)?

Since processors (or operating systems) are not able to prevent accessing any memory addresses within a process, threads in a process do not have a good data protection processes have (if they try to access any memory addresses within the same process) Therefore, threads gave weaker protection against illegal/harmful memory accesses (i.e., weaker data protections).

(3) What is the primary role of a "kernel mode (of processors)"?

The primary role of "the kernel mode" in a processor is to let processors to perform any activities/operations without any restriction (no restriction on what a processor performs/executes).

It is primarily for operating systems to access/manages any (literally "any") operation (e.g., accessing hardware resources, memory addresses).

(4) What are the advantages in the kernel-mode threads?

There are two primary advantages:

- Fair-thread scheduling is possible, <u>since preemptive thread scheduling is possible</u> (i.e., operating systems will manage threads better).

- <u>More robust</u> thread executions. Especially if a thread in a process lost control (e.g., infinite loops), operating systems will be able to assign a processor to other (still functioning) threads in the same process.

(5) What are the advantages in the user-mode threads?

There are two primary advantages:

- Since context-switching to an operating system will not occur when a context-switching from a thread to another in the same process (e.g., the mode of the processor remains the USER-MODE, the whole PCB (of a user process) to the PCB of the operating system is not needed to be switched, and even some more), context-switching between <u>threads will be faster</u>.

- Program source code files are <u>more portable</u>.