Operating Systems

Homework 2

Question 1

An advantage of using a ‘mini’ context switch to handle interrupts from input devices, is that since only some of the registers are being saved to memory, the context switch can be performed faster.

A disadvantage of using the ‘mini’ context switches, is that the code that handles the interrupt would need to be created using as few registers as possible, hence would likely be required to be written in assembly since most programming languages do not try to optimise the number of registers used. Additionally, this would cause the interrupt handler to require lots of memory accesses, since it is using as few registers as possible

Question 2

Context switches for processes are much more intensive than for threads. In both cases the registers, stack pointer and program counter must be stored and loaded. But when switching processes, the address spacing and accounting information must be swapped as well.

Question 3

- I would have to consider if my program’s different sections need to share memory. Sharing memory between threads is typically easier than sharing memory between processes. If they don’t need to share memory, then processes makes it safer so that each process cannot access another’s memory.
- If context switches would occur often, then threads have an advantage over processes, because it is faster to perform a context switch between threads than processes.
- Whether or not the operating system provides support for threads.