**Asynchronous interrupt / Synchronous interrupt (i.e., exception/fault/trap).** Circle only one of A or S.

1. A / S Divide by zero

2. A / S CPU timer interrupt

3. A / S Completion of I/O operation

4. A / S System call that invokes the kernel

5. A / S Response to attempted execution of privileged instruction in user mode

**True / false.** Circle only one of T or F.

6. T / F A guest OS runs in a virtual machine provided by a host OS.

7. T / F On a multiprocessor, an interrupt causes all available processors to respond.

**Fill in the blanks.**

8. The generic response to an interrupt is:

save the \_\_\_\_\_\_\_\_PC\_\_\_\_\_\_\_\_\_\_\_\_ and the \_\_\_\_\_\_\_\_\_PSR (contains mode bit and permissions)\_\_\_\_\_\_\_\_\_\_\_

change execution mode to \_\_\_\_\_kernel\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

disable or restrict further \_\_\_\_\_\_interrupts\_\_\_\_\_\_\_\_\_\_\_\_\_\_

load the new \_\_\_\_\_\_\_\_PC\_\_\_\_\_\_\_\_\_\_\_ from the \_\_\_\_\_\_IVT (interrupt table)\_\_\_\_\_\_\_\_\_\_\_\_\_\_

9. An interrupt return (iret) instruction needs to restore the \_\_\_\_\_PSR\_\_\_\_\_\_\_\_\_\_\_ and the \_\_\_\_\_\_PC\_\_\_\_\_\_\_\_\_\_ to provide restartable execution.

**Short Answer.**

10. What is the purpose of a hardware timer?

To interrupt the execution and return control to the OS kernel periodically.

11. What bad thing could happen if the user had access to the hardware timer and could change the value?

Malware could keep adding time to the timer and prevent the processor from breaking out of an infinite loop in user code.

12. Why should the hardware allow the masking of interrupts?

The kernel will need to protect small sequences of code with interrupts off to prevent corruption of kernel data.

13. Identify the two major differences between a jump-to-subroutine instruction (i.e., procedure call instruction) and a syscall instruction (i.e., software interrupt or trap instruction).

Syscall has to enter the OS kernel only at entry points defined in the IVT.

Syscall will change execution mode to kernel mode since it is an intentional interrupt.

14. Why would one process need both a user stack and a kernel stack?

15. Why should the operating system copy the system call parameters before checking their validity?