Word Bank. Write one of the words or terms from the following list into the blank appearing to the left of the appropriate definition. Note that there are more words and terms than definitions. (1 pt. each)

<table>
<thead>
<tr>
<th>asynchroneous I/O</th>
<th>hardware timer</th>
<th>privileged instruction</th>
<th>running list</th>
<th>efficiency</th>
<th>host OS</th>
<th>process</th>
<th>thread</th>
</tr>
</thead>
<tbody>
<tr>
<td>exception</td>
<td>interrupt handler</td>
<td>protection</td>
<td>thread context switch</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>finished list</td>
<td>interrupt masking</td>
<td>ready list</td>
<td>throughput</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>green threads</td>
<td>microkernel</td>
<td>resource</td>
<td>virtualization</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>guest OS</td>
<td>monolithic kernel</td>
<td>response time</td>
<td>waiting list</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. ______________________ An operating system running in a virtual machine.
2. ______________________ The lack of overhead in implementing an abstraction.
3. ______________________ A kernel procedure invoked when an interrupt occurs.
4. ______________________ The ability to temporarily defer any hardware interrupts.
5. ______________________ Instruction available in kernel mode but not in user mode.
6. ______________________ The time for a task to complete, from when it starts until it is done.
7. ______________________ A physical or virtual entity that can be assigned to a user or application.
8. ______________________ A hardware device that can cause a processor interrupt after some delay.
9. ______________________ A single execution sequence that represents a separately schedulable task.
10. ______________________ The set of threads that are ready to be run but which are not currently running.
11. ______________________ Provide an application with the illusion of resources that are not physically present.
12. ______________________ An OS design where most of the OS functionality is linked together inside the kernel.
13. ______________________ Suspend execution of a currently-running thread and resume execution of some other thread.
14. ______________________ A hardware event caused by the user program behavior that causes a transfer of control to a kernel handler.
15. ______________________ An operating system that provides the abstraction of a virtual machine, to run another operating system as an application.
16. ______________________ The set of threads that are complete but not yet de-allocated, e.g., because a join may read the return value from the TCB.
17. ______________________ The execution of an application program with restricted rights – the abstraction for protection provided by the operating system kernel.
18. ______________________ When a process issues an I/O request, the system call returns immediately. The process later on receives a notification when the I/O completes.
19. ______________________ The isolation of potentially misbehaving applications and users so that they do not corrupt other applications or the operating system itself.
20. ______________________ A thread system implemented entirely at user-level without any reliance on operating system kernel services, other than those designed for single-threaded processes.
Kernel mode / User mode. Circle **one or both** of K and U, as applies. (2 pts. each)

21. K / U  A value from a user stack can be loaded in this mode.
22. K / U  A store instruction is allowed to execute in this mode.
23. K / U  All physical memory locations can be accessed in this mode.
24. K / U  An iret (interrupt return) instruction is allowed to execute in this mode.
25. K / U  For port-mapped I/O, I/O instructions are allowed to execute in this mode.

Process / Thread. Circle **one or both** of P or T, as applies. (2 pts. each)

26. P / T  Has an associated control block.
27. P / T  Has an associated data segment.
28. P / T  Has an associated code segment.
29. P / T  Has an associated heap segment.
30. P / T  Has an associated stack segment.
31. P / T  Has an associated scheduling priority.
32. P / T  Has an associated PC (program counter).
33. P / T  Has an associated PSR (processor status register).

True / False. Circle **only one** of T or F. (1 pt. each)

34. T / F  An operating system kernel can use internal threads.
35. T / F  Linux is an example of the microkernel design approach.
36. T / F  Interrupt handlers are scheduled by the thread scheduler.
37. T / F  A command interpreter on a Unix or Linux system is also known as a shell.
38. T / F  Polling for I/O completions is typically more efficient than using interrupts.
39. T / F  Threads are less expensive for the operating system kernel to create than processes.
40. T / F  When an I/O device completes, each processor in a multiprocessor system is interrupted.
41. T / F  When a user attempts to execute a privileged instruction in user mode the CPU should halt.
42. T / F  An OS can provide services to user applications using a server process that runs in kernel mode.
43. T / F  The Unix process management API (e.g., fork() and exec()) has been stable over several decades.
44. T / F  The interrupt vector table should be held in kernel memory and not arbitrarily changed by users.
45. T / F  A loadable device driver means that the kernel does not have to be recompiled to use the device.
46. T / F  The OS should be invoked by a syscall or trap instruction rather than a normal jump to subroutine.
47. T / F  An OS typically provides a single, global waiting list for all the threads that are in the waiting state.
48. T / F  In a typical modern OS, each schedulable unit of execution needs to have its own protection domain.
49. T / F  fork() in UNIX creates a new thread, which then executes the function that is passed as an argument.
50. The four generic actions that hardware performs in response to an interrupt are: (2 pts. each)

Short Answer. To receive credit, your answers must be specific and go beyond aspirational/good-feeling words or terms such as efficiency, performance, protection, reliability, and security. (3 pts. each)

51. Why is an OS said to be interrupt driven?

52. Give at least one function provided in the seL4 microkernel.

53. Give at least one way to cause a transfer from user mode to kernel mode.

54. Give at least one way to cause a transfer from kernel mode to user mode.
55. Why should an OS copy system call parameters before checking their validity?

56. Give at least one distinct action, feature, or service provided by the OS in the role of glue.

57. Give at least one distinct action, feature, or service provided by the OS in the role of referee.

58. Give at least one distinct action, feature, or service provided by the OS in the role of illusionist.

59. Give at least one action, event, or system call that causes a thread to enter the Running state.

60. List at least one way in which an OS can return results after a call to perform asynchronous I/O.