1. In the dining philosophers problem, the philosophers spend their lives alternating between thinking and
   a. working.
   b. eating.
   c. computing.

2. The shared resource(s) in the dining philosophers problem is(are)
   a. food.
   b. seats at a circular table.
   c. forks.

3. The processes in the dining philosophers problem correspond to the behavior of
   a. the food supply.
   b. the philosophers.
   c. the benefactor supplying the food.

4. Suppose there are 7 philosophers in the dining philosophers problem. At most how many of these could be eating at the same time?
   a. 2
   b. 3
   c. 7

5. Suppose there are 8 philosophers in the dining philosophers problem. What is the minimum number of these that may be eating at the same time?
   a. 0
   b. 5
   c. 1

6. When all that is impossible is eliminated, then whatever remains, however improbable,
   a. must be the truth.
   b. must be eliminated.
   c. will occur with great frequency.

7. One way to find potential problems with a proposed solution to an IPC problem is to
   a. consider all possible execution sequences and eliminate those that are not causing problems.
   b. run the proposed solution on a system with one processor for each process.
   c. rewrite the solution using a different IPC mechanism.
8. One sequence of actions that must not be allowed in the dining philosophers problem is
   a. for the last philosopher to eat, then the next to the last philosopher to eat, and so forth.
   b. for each philosopher to pick up the fork to their left and then, while holding it, wait for the fork to their right.
   c. for all philosophers to be thinking at the same time.

9. A good solution to the dining philosophers problem must provide a reasonable measure of assurance that
   a. once a philosopher has indicated a desire to eat that he or she will be eventually allowed to do so.
   b. no more than three other philosophers will be allowed to eat before a philosopher that is hungry will be allowed to eat.
   c. no more than two other philosophers will be allowed to eat before a philosopher that is hungry will be allowed to eat.

10. Suppose in a proposed solution to the dining philosopher problem that a philosopher has indicated it is hungry, but it is forever denied the opportunity to eat. In this case, we say that the solution exhibits
    a. indefinite postponement.
    b. temporal instability.
    c. none of the other answer choices.

11. In the solution to the dining philosophers problem presented, a third state is associated with each philosopher. This additional state is called
    a. hungry.
    b. right fork.
    c. full.

12. How many classes of processes are there in the readers/writers problem?
    a. more than 4
    b. 2
    c. 3

13. Each process in the readers/writers problem is involved with reading or writing what?
    a. three objects: one unique to the readers, one unique to the writers, and one shared by the readers and the writers
    b. a shared object, like a database
    c. one or more objects, none of which may be accessed by more than one reader at a time

14. How many reader processes can be accessing an object in the readers/writers problem at the same time?
    a. as many as there are writer processes
    b. as many as desired
    c. at most two
15. How many writer processes can be accessing an object in the readers/writers problem at the same time?
   a. at most one
   b. at most two
   c. the maximum of NW - NR and 1, where NW is the number of writer processes and NR is the number of reader processes

16. In total, how many reader and writer processes may be accessing an object in the readers/writers problem at the same time?
   a. An arbitrarily large number of readers and at most one writer may be accessing an object at the same time.
   b. An equal number of readers and writers may be accessing an object at the same time.
   c. Readers and writers may not both access the shared object at the same time.

17. Which of the following are common variants of the reader/writer problem?
   a. reader priority and writer priority
   b. static writer and dynamic writer
   c. equal readers and writers, and unequal readers and writers

18. Assume we allow reader processes to continually gain access to the shared object in the reader/writer problem, even if writers are waiting to access it. Which of the following is true?
   a. Indefinite postponement of writers could occur.
   b. Indefinite postponement of readers and writers could occur.
   c. None of the other answer choices is correct.

19. The solution presented for the reader/writer problem uses two semaphores. What resources have their access controlled by these semaphores?
   a. the variable that indicates the number of active reader processes, and the variable that indicates the number of active writer processes
   b. the shared object and the variable that indicates which type of processes are currently accessing the shared object
   c. the variable that indicates the number of active reader processes, and the shared object

20. Assume there is a semaphore \( DB \) used to control access to the shared object in the reader/writer problem. Also assume there are four readers currently accessing the shared object. How many of those readers did a down operation on the \( DB \) semaphore to obtain their current access to the shared object?
   a. one or none
   b. three
   c. four
21. Assume there is a semaphore used to control access to the shared object in the reader/writer problem. Also assume five write operations have been performed on the shared object. How many successful down operations were performed by writer processes on the semaphore?
   a. at least one
   b. at least two but no more than five
   c. five

22. The term *batch processing* historically refers to
   a. the processing of a set of jobs that were all submitted as a group.
   b. the processing of requests from a batch of users at terminals.
   c. the processing of a batch of jobs submitted all at once from an interactive terminal.

23. Batch processing today is most likely referred to as
   a. embedded processing.
   b. group processing.
   c. background processing.

24. Jobs processed in batches
   a. are always run using the shortest-job first scheduling algorithm.
   b. must all have been submitted from the same interactive terminal.
   c. usually do not perform I/O on terminals or other interactive devices.

25. *Interactive jobs*
   a. can only perform input/output using the local keyboard, display, and mouse.
   b. utilize terminals, mice, and other devices designed for human input and output.
   c. can use only text-mode devices, as opposed to devices capable of doing multimedia input/output.

26. Interactive processes
   a. typically require timely response from the system after user input occurs.
   b. must always be started from the terminal directly connected to the system, never through a network terminal.
   c. cannot directly perform network input/output, but must instead start a batch process to handle the network input/output.

27. The term *scheduling* refers to
   a. all activities of a system associated with determining when various actions are to be performed.
   b. the processing of requests from processes in an order that is strictly the same as the order in which the requests were submitted.
   c. None of the other answer choices is correct.
28. If a typical process is executed several times with the same data
   a. it should execute exactly the same sequence of instructions.
   b. it will produce output that is dependent on the number of times the same code is being executed; that is, how many processes are executing the same text editor, for example.
   c. None of the other answer choices is correct.

29. Regardless of the number of processes being executed, or the code that processes are executing
   a. each process will receive the same amount of CPU time.
   b. each possible execution sequence of each process is finite.
   c. None of the other answer choices is correct.

30. The term *job scheduling* refers to
   a. the detailed scheduling of the individual instructions of individual processes.
   b. the determination of when individual jobs are submitted to a pool, or queue, of jobs from which individual jobs may be selected for execution by the CPU.
   c. the scheduling of a job, or sequence of individual sequential steps.

31. Job scheduling is done using
   a. the number of times the job has been run over the past several time intervals in which scheduling decisions are made.
   b. a special hardware device that estimates the length of time a process will take to execute.
   c. information about the resources required by the entire job, and the currently available system resources.

32. Which of the following would be characterized as *high-level* scheduling?
   a. job scheduling
   b. instruction scheduling
   c. all of the above can be characterized as high-level scheduling.

33. Which of the following would be characterized as *low-level* scheduling?
   a. thread scheduling
   b. job scheduling
   c. process scheduling

34. To what does the term *turnaround time* apply?
   a. the time required to switch a processor (CPU) from one process to another.
   b. the time between submitting a non-interactive, or batch, job and receiving the results.
   c. the time at which a half-duplex communication line has the direction of communication changed.
35. Which of the following is not a goal of scheduling?
   a. give interactive users timely responses to their actions.
   b. keep the CPU as idle as possible, reflecting effective use of the processor.
   c. Each of the above is a goal of scheduling.

36. When preemptive scheduling is used
   a. a process can continue to use the processor as long as it wishes, or is capable of using it.
   b. a running process can have the CPU taken away from it, and it can be returned to the
      ready state.
   c. None of the other answer choices is correct.

37. The term non-preemptive scheduling is also called
   a. interactive scheduling.
   b. run-to-completion scheduling.
   c. batch scheduling.

38. A clock in a computer system operates by
   a. providing periodic interrupts.
   b. measuring the rate at which the CPU is executing processes.
   c. effectively sending what appears to be input from a terminal with a constant, but
      programmable, interval.

39. Which of the following actions might be taken each time the primary system clock ticks?
   a. If any actions are scheduled to occur at the current time, then steps are taken to start those
      actions.
   b. Every process in the blocked state is checked to see if the resource (or resources) for
      which it is waiting is now available.
   c. None of the other answer choices is correct.

40. Suppose there are only three processes in a system, and they are compute-bound (that is,
    they do no input/output at all). Assume the processes are using round-robin scheduling with
    a quantum size of 10 milliseconds. Suppose each process needs 1 second (1000
    milliseconds) of CPU time to complete its work. Ignoring the time required to switch the
    CPU between processes, after how long will the first process complete its work?
    a. 1020 milliseconds
    b. 2990 milliseconds
    c. 2980 milliseconds

41. The maximum amount of time a process may use the CPU before it is returned to the ready
    state in a round-robin system is called
    a. the quantum size of the process.
    b. the recycle period of the system.
    c. the user limit for the system.
42. If the quantum size in a system is very small, then most of the time in the system is spent
   a. performing context switches between processes.
   b. executing the idle process.
   c. performing input/output.

43. If the quantum size in a system is very large,
   a. memory utilization will be very efficient.
   b. interactive processes will experience very good response times to user input.
   c. the system effectively appears to be doing non-preemptive scheduling.

44. In a system that uses priority scheduling
   a. the process currently using the CPU is one of those that has the highest priority.
   b. it is not also possible to use round-robin scheduling.
   c. a high priority process has its priority reduced if and when it becomes blocked.

45. In a system that uses priority scheduling
   a. processes with the same priority are allowed only if the system has multiple processors.
   b. processes with the same priority can be run in a round-robin fashion.
   c. processes with the same priority are not allowed.

46. In a UNIX system, when a process blocks for an input/output operation,
   a. the system assumes it is I/O bound (that is, spending most of its time performing
      input/output), and its priority is increased.
   b. its priority is changed to the lowest of all ready processes.
   c. None of the other answer choices is correct.

47. Suppose a system is using the inverse remainder of quantum variant of round-robin
    scheduling. A process uses 30 percent of its quantum before blocking. When it is eventually
    returned to the ready queue,
   a. its priority is increased by 30 percent.
   b. it is placed at the rear of the ready queue.
   c. it is placed behind 30 percent of the processes in the ready queue, not at the rear.

48. The batch scheduling algorithm called shortest job first always selects the job with the
    smallest execution time to run next. This has the result of
   a. always yielding the smallest average turnaround time of any job scheduling algorithm.
   b. yielding the smallest standard deviation of the job turnaround times of any job
      scheduling algorithm.
   c. None of the other answer choices is correct.
49. Since the shortest job first algorithm cannot be used directly on interactive processes, a variation called **aging** is used. In this algorithm
   a. the estimated execution time for each process is increased by a fixed percentage each time the process enters the running state.
   b. the estimated execution time for a process is a weighted sum of the previous execution time estimate and the last execution time.
   c. None of the other answer choices is correct.

50. For many systems, the only criteria associated with the correct execution of a process is whether the process yields the correct results. Systems in which it is also required that actions be taken in a specified amount of time or at specified times are called
   a. batch systems.
   b. priority-based systems.
   c. real-time systems.

51. A system in which failing to meet a deadline can result in total system failure or death is
   a. called a **hard real-time system**.
   b. permitted only if multiple processors are available.
   c. called a **distributed system**.

52. Suppose a real-time system is going to process only two types of events. The first type of event occurs once every 10 milliseconds and requires 4 milliseconds of CPU time to process. The second event occurs once every 20 milliseconds. What is the most CPU time each 20-millisecond event can require if the system is to be considered schedulable?
   a. 8 milliseconds
   b. 12 milliseconds
   c. 10 milliseconds

53. In the rate-monotonic scheduling algorithm
   a. processes are run to completion once they are selected for execution.
   b. each process gets a priority that is proportional to the frequency of occurrence of its triggering event.
   c. each process gets a priority that is inversely proportional to the frequency of occurrence of its triggering event.

54. The term **swapping** describes a system that
   a. moves the memory content of processes between primary memory and secondary storage if insufficient memory is available.
   b. gives each process two priorities which are swapped when the process moves between the ready/running states and the blocked state.
   c. None of the other answer choices is correct.
55. Which of the following statements regarding the terms *policy* and *mechanism* is correct?
   
a. A policy is a set of rules, or parameters associated with a set of rules, that are implemented using the available mechanisms in a system.

b. Policies are tools available in an operating system to implement various mechanisms that may be desired.

c. None of the other answer choices is correct.