Operating Systems & Networking MCQs

This document contains **30 multiple-choice questions** (MCQs) on operating systems and networking, followed by their respective answers.

Questions

1. In a paged memory system, increasing the page size generally:
   - A. Increases internal fragmentation but reduces page table size
   - B. Decreases internal fragmentation and increases page table size
   - C. Has no impact on page table size
   - D. Eliminates page faults completely
2. Which of the following is least likely to cause a page fault?
   - A. Accessing a page for the first time
   - B. Accessing a page already in main memory
   - C. Accessing a swapped-out page
   - D. Accessing a memory location not mapped to any page
3. In a system using demand paging, the effective access time (EAT) is:
   - A. Always equal to main memory access time
   - B. Weighted sum of memory access time and page fault service time
   - C. Sum of main memory access time and page fault service time
   - D. Main memory access time multiplied by hit ratio
4. The “working set” model helps in:
   - A. Preventing segmentation faults
   - B. Estimating the minimum number of frames needed to avoid thrashing
   - C. Reducing internal fragmentation
   - D. Improving CPU scheduling
5. Belady’s anomaly occurs in:
   - A. FIFO page replacement
   - B. LRU page replacement
   - C. Optimal page replacement
   - D. All of the above
6. Shared memory IPC is generally:
   - A. Slower than message passing
   - B. Faster than message passing
   - C. Always OS-dependent and slower than sockets
   - D. Not possible in Linux
7. In UNIX System V shared memory, `shmat()` is used for:
   - A. Creating a shared memory segment
   - B. Attaching a existing shared memory segment
   - C. Removing a shared memory segment
   - D. Locking shared memory
8. The biggest challenge with shared memory IPC is:

- A. High latency
- B. Synchronization between processes
- C. Lack of portability
- D. Expensive system calls per read/write

9. In POSIX shared memory, `shm_open()` returns:
   - A. A pointer to the shared memory
   - B. A file descriptor
   - C. A memory-mapped address immediately
   - D. A semaphore handle
10. A binary semaphore differs from a mutex because:
    - A. Mutex is faster
    - B. Binary semaphore can be signaled by a thread that didn't acquire it
    - C. Mutex can be acquired multiple times by the same thread without blocking
    - D. Binary semaphore has no synchronization use
11. A counting semaphore initialized to N allows:
    - A. Exactly N processes to be in the critical section simultaneously
    - B. Unlimited processes in critical section
    - C. At most N processes waiting
    - D. Exactly N processes to wait
12. The P and V operations on semaphores:
    - A. Increment and decrement the semaphore count respectively
    - B. P $\rightarrow$ wait (decrement), V $\rightarrow$ signal (increment)
    - C. Both increment
    - D. Both decrement
13. Priority inversion in semaphore usage occurs when:
    - A. Low priority thread holds semaphore needed by a high priority thread
    - B. High priority thread holds semaphore needed by a low priority thread
    - C. Semaphore count is negative
    - D. Semaphore value overflows
14. Which socket type guarantees message boundaries?
    - A. TCP
    - B. UDP
    - C. RAW
    - D. None of these
15. The `listen()` call in TCP server sockets:
    - A. Actively starts connection establishment
    - B. Passively waits for incoming connections
    - C. Sends data to client
    - D. Binds the socket to a port
16. In TCP, the TIME_WAIT state is important because:
    - A. It reduces latency for new connections
    - B. It ensures old duplicate packets don't get misinterpreted

- C. It frees ports quickly
- D. It's only for servers

17. Which is NOT true about `select()`?
    - A. Can monitor multiple sockets
    - B. Blocks until at least one socket is ready or timeout expires
    - C. Returns immediately if no socket is ready
    - D. Can monitor sockets for read, write, and exceptions
18. In preemptive scheduling:
    - A. A running process cannot be interrupted
    - B. Scheduler can stop a process to run another
    - C. Only batch jobs run
    - D. No context switch occurs
19. Which scheduling algorithm may cause starvation?
    - A. Round Robin
    - B. Shortest Job First (SJF)
    - C. Priority Scheduling
    - D. FCFS
20. Context switch time is:
    - A. Counted as CPU utilization
    - B. Pure overhead
    - C. Always zero
    - D. Negative in some cases
21. The scheduler that adjusts priorities dynamically to prevent starvation is:
    - A. FCFS
    - B. Multi-level feedback queue
    - C. SJF
    - D. Lottery scheduling
22. The main cost of handling a page fault is:
    - A. OS scheduler run time
    - B. Disk I/O to load the page
    - C. CPU cache invalidation
    - D. TLB flush
23. In demand paging, the initial access to a large array:
    - A. Has uniform access time for all elements
    - B. May cause multiple page faults until the array is paged in
    - C. Never causes page faults if array is allocated
    - D. Causes only one page fault
24. Copy-on-write is used to:
    - A. Avoid race conditions
    - B. Delay copying of a page until it is modified
    - C. Improve mutex performance
    - D. Avoid swapping
25. Thrashing occurs when:
    - A. Disk is faster than memory
    - B. Processes spend more time paging than executing
    - C. The system has high CPU utilization

- D. Page size is too small

26. TLB miss penalty is reduced by:
    - A. Increasing page size
    - B. Increasing associativity of TLB
    - C. Disabling demand paging
    - D. Using FCFS scheduling
27. In producer-consumer using shared memory, semaphores must:
    - A. Ensure mutual exclusion only
    - B. Ensure synchronization and mutual exclusion
    - C. Be avoided for performance
    - D. Be replaced with sleep calls
28. Non-blocking sockets are primarily useful for:
    - A. Low-latency event-driven servers
    - B. High-throughput batch jobs
    - C. Reducing memory fragmentation
    - D. Eliminating TIME_WAIT
29. The "dirty bit" in a page table entry is used to:
    - A. Indicate page is in memory
    - B. Indicate page has been modified since being loaded
    - C. Indicate page has been swapped
    - D. Indicate page is locked in memory
30. When two processes communicate via sockets on the same machine:
    - A. Data always goes through the NIC
    - B. OS may optimize by using loopback interface and kernel memory
    - C. Performance is same as shared memory
    - D. It's not possible

Answers with Explanations

1. **A – Increases internal fragmentation but reduces page table size** Larger pages waste more memory inside each page (internal fragmentation), but fewer pages are needed, so page table shrinks.

2. **B – Accessing a page already in main memory** Page fault happens only when the required page is not in memory. If it's already loaded, no fault.

3. **B – Weighted sum of memory access time and page fault service time** Effective Access Time (EAT) = (1 − p) × memory access + p × page fault time, where `p` = page fault probability.

4. **B – Estimating the minimum number of frames needed to avoid thrashing** The working set tracks actively used pages; ensuring enough frames for this avoids thrashing.

5. **A – FIFO page replacement** Belady's anomaly = increasing frames

can increase page faults, seen in FIFO, not LRU/Optimal.

6. **B – Faster than message passing** Shared memory is direct and avoids kernel involvement per message, unlike message passing.
7. **B – Attaching an existing shared memory segment** `shmat()` = attach; `shmget()` = create; `shmctl()` = control/remove.
8. **B – Synchronization between processes** Multiple processes can overwrite shared memory simultaneously → race conditions.
9. **B – A file descriptor** `shm_open()` returns a file descriptor, which can be mapped with `mmap()`.
10. **B – Binary semaphore can be signaled by a thread that didn't acquire it** Mutex must be released by the same owner; binary semaphore doesn't enforce ownership.
11. **A – Exactly N processes to be in the critical section simultaneously** Counting semaphore initialized to `N` gives up to `N` concurrent accesses.
12. **B – P → wait (decrement), V → signal (increment)** Classic Dijkstra operations: P (Proberen = test) decrements, V (Verhogen = increment) signals.
13. **A – Low priority thread holds semaphore needed by a high priority thread** High-priority thread gets blocked by a lower one → inversion occurs.
14. **B – UDP** UDP preserves message boundaries; TCP gives a stream without boundaries.
15. **B – Passively waits for incoming connections** `listen()` puts the server in a passive wait state; `accept()` completes connection.
16. **B – It ensures old duplicate packets don't get misinterpreted** TIME_WAIT ensures delayed/duplicate packets from old connections don't interfere.
17. **C – Returns immediately if no socket is ready** `select()` blocks unless timeout = 0 (polling mode).
18. **B – Scheduler can stop a process to run another** Preemption = forcibly stopping a process for better scheduling.
19. **C – Priority Scheduling** Low-priority processes may never get CPU → starvation risk.
20. **B – Pure overhead** Context switching saves/restores states, consuming CPU time but doing no useful work.
21. **B – Multi-level feedback queue** Adjusts priorities dynamically → avoids starvation by eventually promoting waiting jobs.

22. **B – Disk I/O to load the page** Disk access dominates page fault handling time (ms vs ns).
23. **B – May cause multiple page faults until the array is paged in** First access to each new page of the array causes a fault (demand paging).
24. **B – Delay copying of a page until it is modified** Copy-on-write: processes share the same page until one writes to it.
25. **B – Processes spend more time paging than executing** Thrashing = excessive paging activity, CPU utilization drops.
26. **B – Increasing associativity of TLB** Higher associativity reduces collisions in TLB lookups, lowering miss penalty.
27. **B – Ensure synchronization and mutual exclusion** Need both: mutex for buffer access, semaphores for producer/consumer coordination.
28. **A – Low-latency event-driven servers** Non-blocking sockets let servers handle many connections without blocking I/O.
29. **B – Indicate page has been modified since being loaded** Dirty bit = page must be written back to disk before replacement.
30. **B – OS may optimize by using loopback interface and kernel memory** Local sockets don't go through NIC, OS routes via loopback.
