

PAPER 1

Chapters 13-20: Memory and Addressing

Multiple Choice Questions (MCQs)

1. **What is the main purpose of address translation in an operating system?**
 - a) To allocate memory to processes
 - b) To map virtual addresses to physical addresses
 - c) To enable direct memory access for devices
 - d) To ensure CPU scheduling
 2. **Which of the following is a characteristic of paging?**
 - a) Eliminates external fragmentation
 - b) Uses variable-sized blocks of memory
 - c) Requires contiguous memory allocation
 - d) Reduces internal fragmentation
 3. **What happens if a virtual address exceeds the bounds of a segment?**
 - a) A page fault occurs
 - b) A segmentation fault occurs
 - c) The address wraps around
 - d) The system ignores it
 4. **In a hierarchical page table, what does the second-level table map?**
 - a) Virtual addresses to physical frames
 - b) Page numbers to frame numbers
 - c) Virtual pages to segments
 - d) Logical addresses to offsets
 5. **Which of the following is NOT a goal of memory virtualization?**
 - a) Process isolation
 - b) Efficient memory usage
 - c) Direct hardware access
 - d) Simplified memory management
-

True/False

6. Paging eliminates the need for contiguous memory allocation. *(True/False)*
7. A page fault occurs when a requested page is already in memory. *(True/False)*
8. Virtual memory uses disk space as an extension of RAM. *(True/False)*
9. Multi-level page tables reduce memory usage by only allocating entries for used address spaces. *(True/False)*
10. Swapping allows processes to execute that are larger than physical memory. *(True/False)*

Fill in the Blanks

11. A _____ fault occurs when a process tries to access a page that is not in memory.
 - a) Page
 - b) Segmentation
 - c) Protection
 - d) Access
 12. In segmentation, each segment has its own _____ and bounds register.
 - a) Page table
 - b) Base
 - c) Offset
 - d) Frame
 13. Virtual memory uses _____ space as an extension of physical memory.
 - a) Cache
 - b) Disk
 - c) Page table
 - d) RAM
-

Chapters 26-30: Concurrency and Synchronization

Multiple Choice Questions (MCQs)

14. **What does a mutex ensure in a multi-threaded program?**
 - a) Multiple threads execute the critical section simultaneously
 - b) Threads have exclusive access to shared resources
 - c) Threads are preempted during execution
 - d) Threads communicate via message passing
15. **Which synchronization primitive is best for read-heavy workloads?**
 - a) Spinlock
 - b) Read-Write Lock
 - c) Binary Semaphore
 - d) Reentrant Lock
16. **What happens during a deadlock?**
 - a) Threads execute the critical section out of order
 - b) Threads wait indefinitely on each other to release resources
 - c) Threads fail to acquire locks but proceed with execution
 - d) Threads are preempted by the operating system
17. **What is the primary role of a condition variable?**
 - a) Prevent race conditions
 - b) Allow threads to wait for specific conditions to be met

- c) Manage shared resources
- d) Signal a semaphore to release threads

18. Which function is used to increment a semaphore?

- a) sem_wait()
- b) sem_post()
- c) pthread_cond_signal()
- d) pthread_mutex_lock()

19. Which of the following is a benefit of threads? *(Multiple Correct Answers)*

- a) Lightweight compared to processes
- b) Faster context switching
- c) Separate memory spaces for each thread
- d) Parallel execution on multi-core CPUs

True/False

- 20. A binary semaphore allows multiple threads to access shared resources simultaneously. *(True/False)*
- 21. Condition variables must be used with mutexes to avoid race conditions. *(True/False)*
- 22. Deadlock can occur if threads acquire locks in different orders. *(True/False)*
- 23. Spinlocks are efficient for short critical sections. *(True/False)*
- 24. Race conditions occur when threads access shared resources without synchronization. *(True/False)*

Fill in the Blanks

- 25. A _____ is a lock that continuously checks its availability without putting threads to sleep.
 - a) Binary Semaphore
 - b) Spinlock
 - c) Reentrant Lock
 - d) Mutex
- 26. Condition variables are often used in the _____ problem to synchronize producers and consumers.
 - a) Deadlock
 - b) Thrashing
 - c) Producer-Consumer
 - d) Swapping
- 27. A _____ lock allows the same thread to acquire it multiple times without causing deadlock.
 - a) Spinlock

- b) Reentrant
- c) Read-Write
- d) Condition

PAPER 2

Chapters 13-22: Memory and Addressing

Multiple Choice Questions (MCQs)

1. **What is the purpose of the memory management unit (MMU) in an operating system?**
 - a) To manage the CPU scheduling
 - b) To map virtual addresses to physical addresses
 - c) To schedule processes in memory
 - d) To allocate disk space
2. **Which of the following is a feature of paging?**
 - a) Eliminates external fragmentation
 - b) Uses variable-sized memory blocks
 - c) Requires contiguous memory allocation
 - d) Reduces internal fragmentation
3. **Which of the following is true about a page fault?**
 - a) It occurs when a virtual page is accessed that is not in memory
 - b) It is a hardware error
 - c) It occurs when a page is too large to fit into memory
 - d) It happens only during the first access to a page
4. **What does a segmentation fault indicate?**
 - a) A page was swapped out of memory
 - b) A process accessed a memory address outside its allowed bounds
 - c) The memory management unit failed
 - d) A stack overflow occurred
5. **In a multi-level page table, what does the second-level table do?**
 - a) Maps virtual addresses to physical addresses
 - b) Maps page numbers to frame numbers
 - c) Maps segments to frames
 - d) Maps page numbers to segment numbers
6. **What is the main disadvantage of segmentation?**
 - a) Internal fragmentation
 - b) External fragmentation
 - c) Memory leaks
 - d) Slower address translation
7. **Which of the following is an advantage of paging over segmentation?**
 - a) Elimination of external fragmentation
 - b) Better support for shared memory
 - c) Allows variable-sized blocks of memory
 - d) Simpler memory management with larger blocks

8. **How is memory divided in paging?**
 - a) Into pages and segments
 - b) Into frames and offsets
 - c) Into blocks and segments
 - d) Into frames and pages
 9. **What happens if a process tries to access a page that is not currently in memory?**
 - a) The process is killed
 - b) A page fault occurs, and the page is loaded into memory
 - c) The process waits for the page to be swapped in
 - d) The page is allocated to the process immediately
 10. **What is a major disadvantage of multi-level paging?**
 - a) It requires large page tables
 - b) It can lead to longer translation times
 - c) It results in more fragmentation
 - d) It eliminates the need for a page table
-

True/False

11. **In paging, the page size is always fixed.** *(True/False)*
 12. **Each segment in segmentation has its own base and bounds register.** *(True/False)*
 13. **A page fault occurs when the requested page is already in memory.** *(True/False)*
 14. **External fragmentation occurs when free memory is scattered in small chunks, making allocation difficult.** *(True/False)*
 15. **In a two-level page table, the second-level table maps virtual pages to physical addresses.** *(True/False)*
-

Fill in the Blanks

16. A page fault occurs when a process tries to access a page that is not in _____.
 - a) RAM
 - b) Disk
 - c) Cache
 - d) Memory
17. In a _____ table, each entry maps a virtual page to a physical frame.
 - a) Page
 - b) Segmentation
 - c) Frame
 - d) Translation

18. The _____ is a data structure used to map virtual addresses to physical addresses in a paging system.
- a) Page Table
 - b) Frame Table
 - c) Segment Table
 - d) Address Table
19. The _____ algorithm replaces the page that will not be used for the longest period.
- a) FIFO
 - b) LRU
 - c) Optimal
 - d) Random
20. Virtual memory uses _____ to provide the illusion of a large continuous memory space.
- a) Paging
 - b) Segmentation
 - c) Disk storage
 - d) Cache memory
-

Chapters 26-30: Concurrency and Synchronization

Multiple Choice Questions (MCQs)

21. **What is the purpose of a mutex in a multi-threaded program?**
- a) To allow multiple threads to access shared resources concurrently
 - b) To provide exclusive access to shared resources
 - c) To schedule threads for execution
 - d) To synchronize thread creation
22. **What is a spinlock?**
- a) A lock that causes threads to wait by busy-waiting
 - b) A lock that blocks threads until a condition is met
 - c) A lock that allows multiple threads to acquire it concurrently
 - d) A lock that works only with a single thread
23. **What happens when a thread calls `pthread_mutex_lock()` on a mutex that is already locked?**
- a) The thread enters a waiting state until the mutex is unlocked
 - b) The thread is terminated
 - c) The thread continues executing
 - d) A deadlock occurs
24. **Which of the following is used to allow multiple threads to read shared data but only one thread to write it?**
- a) Mutex
 - b) Spinlock

- c) Read-Write Lock
- d) Condition Variable

25. **What does `pthread_cond_wait()` do?**

- a) Blocks a thread and releases the mutex
- b) Signals all threads waiting on a condition variable
- c) Unlocks the mutex
- d) Terminates a thread

26. **Which of the following can be a consequence of a deadlock?**

- a) Threads block each other indefinitely
- b) Threads execute the critical section out of order
- c) Threads are preempted and restarted
- d) The system crashes

27. **What is the key difference between a binary semaphore and a mutex?**

- a) A binary semaphore cannot be locked by a thread more than once
- b) A mutex is a signaling mechanism, while a binary semaphore is for mutual exclusion
- c) A binary semaphore is used to synchronize threads, while a mutex is used for resource sharing
- d) There is no difference

28. **Which of the following is NOT a type of synchronization mechanism?**

- a) Mutex
- b) Semaphore
- c) Spinlock
- d) Process

29. **What is the typical use case for condition variables?**

- a) Synchronizing threads in a multi-core CPU
- b) Allowing threads to wait for specific conditions before proceeding
- c) Allocating memory between threads
- d) Managing CPU scheduling

30. **Which of the following is true about threads and processes?**

- a) Threads share memory, while processes do not
- b) Processes can share memory, while threads cannot
- c) Threads and processes cannot share memory
- d) Both threads and processes share the same memory space

True/False

31. **Condition variables must always be used in conjunction with mutexes. (True/False)**

32. **Race conditions occur when threads access shared resources in a synchronized manner. (True/False)**

33. **Deadlocks can be prevented by ensuring that threads acquire resources in a consistent order.**
(True/False)
34. **A reentrant lock allows the same thread to acquire the lock multiple times.** (True/False)
35. **Spinlocks are effective for long critical sections where threads will be blocked for a long time.**
(True/False)
-

Fill in the Blanks

36. A _____ is a synchronization primitive that ensures only one thread can access a resource at a time.
a) Semaphore
b) Mutex
c) Spinlock
d) Condition variable
37. A _____ occurs when multiple threads or processes wait indefinitely for resources to be released.
a) Deadlock
b) Starvation
c) Race condition
d) Context switching
38. A _____ lock allows multiple threads to read concurrently, but only one thread to write at a time.
a) Mutex
b) Spinlock
c) Read-Write
d) Reentrant
39. Threads can be synchronized using _____, which allow threads to wait for a certain condition before continuing.
a) Mutex
b) Condition variables
c) Semaphores
d) Spinlocks
40. A _____ is a lock that repeatedly checks its availability rather than going to sleep.
a) Spinlock
b) Mutex
c) Read-Write Lock
d) Condition Variable
-

Question Paper 3

Multiple Choice Questions (MCQs)

1. **[Chapters 13-22]** What is the primary role of the TLB in address translation?
 - a) To manage page faults
 - b) To cache page table entries for faster translation
 - c) To replace the page table entirely
 - d) To prevent segmentation faults
2. **[Chapters 26-30]** Which of the following can be used to prevent race conditions?
 - a) Spinlock
 - b) Mutex
 - c) Semaphore
 - d) All of the above
3. **[Chapters 13-22]** What happens during a TLB miss?
 - a) The page table is accessed to fetch the frame number
 - b) A segmentation fault occurs
 - c) The TLB is cleared
 - d) The operating system terminates the process
4. **[Chapters 26-30]** What does the `sem_init` function do?
 - a) Waits for a semaphore to be available
 - b) Initializes a semaphore with a specific value
 - c) Signals a semaphore
 - d) Locks a mutex
5. **[Chapters 13-22]** What is the primary advantage of hierarchical page tables?
 - a) Faster access to memory
 - b) Reduced memory usage for page tables
 - c) Eliminates internal fragmentation
 - d) Avoids the need for segmentation
6. **[Chapters 26-30]** A deadlock-free solution ensures which of the following conditions is avoided?
 - a) Mutual exclusion
 - b) Hold and wait
 - c) Preemption
 - d) Multiprogramming

7. **[Chapters 13-22]** What is internal fragmentation?
 - a) Wasted memory within an allocated region
 - b) Free memory scattered across the system
 - c) Memory lost due to swapping
 - d) Misaligned memory access
 8. **[Chapters 26-30]** Which thread state occurs if a thread is waiting for a condition variable?
 - a) Ready
 - b) Running
 - c) Blocked
 - d) Terminated
 9. **[Chapters 13-22]** Which of the following best describes swapping?
 - a) Moving processes between different segments
 - b) Moving memory pages between disk and RAM
 - c) Defragmenting physical memory
 - d) Copying processes from one core to another
 10. **[Chapters 26-30]** What does a reentrant lock allow?
 - a) Multiple threads to access a critical section simultaneously
 - b) The same thread to acquire the lock multiple times
 - c) Multiple processes to synchronize using a lock
 - d) Recursive function calls without locking
-

True/False

11. **[Chapters 13-22]** Virtual memory enables processes to use more memory than is physically available. *(True/False)*
 12. **[Chapters 26-30]** Mutexes are slower than spinlocks in scenarios involving short critical sections. *(True/False)*
 13. **[Chapters 13-22]** Paging eliminates both external and internal fragmentation. *(True/False)*
 14. **[Chapters 26-30]** A condition variable can be used without a mutex. *(True/False)*
 15. **[Chapters 13-22]** The effective memory access time depends on the TLB hit rate. *(True/False)*
-

Fill in the Blanks

16. **[Chapters 13-22]** The _____ algorithm replaces the page that has been in memory the longest.
 - a) FIFO
 - b) LRU
 - c) Optimal
 - d) Random

17. **[Chapters 26-30]** Threads within a process share the same _____.
a) Memory space
b) Thread IDs
c) CPU registers
d) Semaphore
18. **[Chapters 13-22]** A _____ is the smallest unit of memory that can be swapped between disk and RAM.
a) Page
b) Frame
c) Block
d) Segment
19. **[Chapters 26-30]** A _____ lock continuously checks availability, wasting CPU cycles if not available.
a) Spinlock
b) Binary Semaphore
c) Read-Write Lock
d) Condition Variable
20. **[Chapters 13-22]** A _____ fault occurs when a process tries to access memory outside its segment bounds.
a) Page
b) TLB
c) Segmentation
d) Access
-

Question Paper 4

Multiple Choice Questions (MCQs)

1. **What is the role of the base and bounds registers in segmentation?**
 - a) To map pages to frames
 - b) To define the start and size of a segment
 - c) To manage virtual memory translation
 - d) To eliminate page faults
2. **What happens if a process exceeds the bounds of its segment?**
 - a) A page fault occurs
 - b) A segmentation fault occurs
 - c) The system reallocates the segment
 - d) The segment wraps around
3. **Which of the following is NOT an advantage of paging?**
 - a) Elimination of external fragmentation
 - b) Simplified allocation of memory
 - c) Efficient use of memory
 - d) Variable-sized memory allocation
4. **What is the purpose of a valid/invalid bit in a page table?**
 - a) To indicate if a page is in memory
 - b) To determine access permissions for a page
 - c) To track page reference counts
 - d) To signal a segmentation fault
5. **What is a major drawback of larger page sizes?**
 - a) Increased page table size
 - b) Higher internal fragmentation
 - c) Slower page replacement algorithms
 - d) Increased TLB misses
6. **What is the main difference between a segmentation fault and a page fault?**
 - a) Segmentation faults occur due to invalid page permissions
 - b) Page faults occur due to exceeding segment bounds
 - c) Segmentation faults occur due to memory access outside allowed bounds
 - d) Page faults occur only in systems without segmentation
7. **Which replacement algorithm guarantees the minimum number of page faults?**
 - a) FIFO
 - b) LRU
 - c) Optimal
 - d) Random
8. **What is the function of the dirty bit in a page table entry?**
 - a) Indicates if the page is currently in memory

- b) Tracks if the page has been modified
- c) Shows if the page is mapped to disk
- d) Determines access permissions for the page

9. **Which of the following describes a soft page fault?**

- a) A page fault where the page is found in a swap space
- b) A page fault where the page is already in memory
- c) A page fault due to invalid permissions
- d) A page fault due to segmentation errors

10. **What is the role of demand paging?**

- a) Preloads all pages into memory before execution
- b) Loads pages into memory only when accessed
- c) Allocates memory dynamically during execution
- d) Maps virtual addresses directly to disk

True/False

- 11. Larger page sizes increase the chances of internal fragmentation. *(True/False)*
- 12. Virtual memory allows each process to have its own isolated address space. *(True/False)*
- 13. Swapping moves entire processes between disk and memory. *(True/False)*
- 14. TLB entries are replaced using page replacement algorithms. *(True/False)*
- 15. Multi-level page tables use additional levels to reduce memory usage. *(True/False)*

Fill in the Blanks

- 16. A _____ fault occurs when a process accesses a page not currently in memory.
 - a) Segmentation
 - b) Page
 - c) Access
 - d) TLB
- 17. The _____ bit in a page table entry indicates whether the page has been modified.
 - a) Valid
 - b) Dirty
 - c) Reference
 - d) Access
- 18. In virtual memory, _____ is the process of moving pages between RAM and disk.
 - a) Swapping
 - b) Paging
 - c) Segmentation
 - d) Thrashing

19. A _____ replacement algorithm evicts the least recently used page.

- a) FIFO
- b) Optimal
- c) LRU
- d) Random

20. A _____ table maps virtual page numbers to physical frame numbers.

- a) Frame
- b) Page
- c) Segment
- d) Translation