Become a Tech-Savvy Entrepreneur

Sanjula Nadeeshani

අධ්‍යයන පොදු සහතික පත්‍ර (උසස් පෙළ) විභාගය, 2026 අගෝස්තු

General Certificate of Education (Adv. Level) Examination, August 2026

තොරතුරු හා සන්නිවේදන තාක්ෂණය I, II
Information & Communication Technology I, II

20 E MCQ

Unit Revising Test 5.1

(1) A university IT technician is setting up a new lab with dual boot capabilities, where each computer needs to run both Windows and Linux. During the configuration process, the technician decides to split each hard disk into multiple logical sections. Which utility operation is most critical at this stage, and what potential consequence must be considered before proceeding?

A. Disk formatting to enable secure OS installation, risking performance issues due to fragmented files
B. File compression to reduce file size, potentially corrupting critical installation files
C. Disk partitioning to create separate logical drives, which could lead to data loss if done on an active disk without backup
D. System profiling to manage OS startup sequences, risking incompatibility with disk types
E. Task manager to allocate processor cores for each OS, possibly causing dual boot failure

(2) During a routine health check of the company's desktop systems, a system administrator uses a tool to monitor for hardware issues and identify software faults that are causing system instability. This tool, part of the operating system, provides detailed diagnostics. What type of utility is this and what function does it primarily serve?

A. File compression utility – reduces the space required to store crash logs
B. Task manager utility – terminates background services consuming RAM
C. System diagnosis utility – scans and reports hardware/software anomalies
D. Clipboard utility – logs memory usage from recent commands
E. Screensaver utility – prevents screen burn-in while analyzing data

(3) A user notices their computer has become unusually slow while opening a large project file. Upon investigation, it is revealed that the file is split across different sectors of the hard drive. Which concept best explains this behavior and what utility can help resolve the issue?

A. Internal fragmentation; resolved using disk partitioning
B. External fragmentation; resolved using file compression
C. File corruption; resolved using antivirus software
D. File fragmentation; resolved using disk defragmentation
E. Cache overflow; resolved using system profiling

(4) An IT student installs a fresh copy of an operating system but is unable to save files to the newly created partition on the hard disk. Which of the following best explains the issue and what must be done before the disk becomes usable?

A. The disk lacks antivirus protection and must be scanned
B. The partition is fragmented and must be defragmented
C. The partition is not formatted and requires a file system

D. The disk has external fragmentation and must be repartitioned
E. The system lacks clipboard synchronization with the partition

(5) An organization with high security standards decides to implement a utility that automatically scans all incoming files and identifies patterns associated with malware. This utility must actively eliminate threats before execution. Which utility does this best describe, and why is it preferred?

A. Backup software – because it restores infected files to a previous state
B. Anti-virus software – because it detects and removes malicious content
C. File compression – because it reduces executable file size to prevent viruses
D. Task manager – because it halts high-CPU usage from infected programs
E. Network utility – because it isolates packets carrying malware

(6) A student is preparing a research project that involves moving large datasets between school and home using a USB flash drive. The drive needs to be emptied before reuse, but some files are corrupt. Which process should the student apply to the drive, and what is a critical caution before doing so?

A. Disk defragmentation; it will recover the corrupt files
B. Disk scanning; it will back up the entire drive contents
C. Formatting; it will erase all data and make the drive reusable
D. Partitioning; it will divide the data for easier access
E. Synchronization; it will merge datasets without loss

(7) A software developer installs a fixed-block memory management system where all processes are given uniform memory blocks. However, many applications are smaller than the allocated memory, leading to wasted space. What phenomenon is being observed and how is it characterized?

A. External fragmentation – caused by gaps between large memory allocations
B. Internal fragmentation – caused by unused memory within fixed-sized blocks
C. Disk fragmentation – caused by scattered pieces of file storage
D. File compression – caused by shrinking file sizes too much
E. Memory synchronization – caused by delay in data backup

(8) In a networked environment, an admin needs to ensure system logs and performance metrics are consistent between the server and backup systems. Which utility would support this requirement and how does it function?

A. Task manager – by showing active processes on each node
B. Data synchronization software – by maintaining identical datasets across systems
C. Disk partitioning software – by separating logs from user files
D. Clipboard utility – by storing log data temporarily
E. Defragmentation tool – by aligning file sectors for faster log access

(9) A user installs several heavy applications that frequently modify and update their saved files. Over time, system performance degrades, and file access becomes delayed. Which of the following scenarios best describes the underlying issue, and what process is necessary to restore performance?

A. Disk formatting has occurred repeatedly, leading to OS degradation
B. External fragmentation has filled the RAM, requiring system profiling
C. File fragmentation from continuous editing, solvable by defragmentation
D. System diagnosis failed to remove corrupted processes
E. Disk partitioning caused overlapping data in memory segments

(10) A technician attempts to run a program that requires 50KB of memory, but although the system has 55KB available, the program fails to execute. The memory blocks are non-contiguous. What memory issue is this, and what would be a viable solution?

A. Internal fragmentation; use disk compression utilities
B. Virtual memory exhaustion; increase swap file size
C. Cache miss; reset system clipboard
D. External fragmentation; apply compaction or paging
E. Process overload; terminate using task manager

(11) A developer is working with a real-time database system that requires extremely fast data retrieval and direct access to any block in a file using a simple mathematical computation. The system allocates all file blocks sequentially in memory. Which file allocation method is being described and what key drawback should be anticipated during file growth?

A. Linked Allocation; suffers from pointer corruption
B. Indexed Allocation; suffers from inefficient direct access
C. Contiguous Allocation; suffers from external fragmentation and limited scalability
D. Hybrid Allocation; suffers from inode overhead
E. Clustered Allocation; suffers from redundant pointer chains

(12) In a scenario where a new file's size is dynamic and might expand frequently after creation, which file allocation method provides the most flexibility, and what compromise does it bring in terms of file block access?

A. Contiguous Allocation – allows fast random access but has high fragmentation
B. Indexed Allocation – supports expansion with no memory waste
C. Linked Allocation – supports flexible growth but slows down block access due to sequential traversal
D. Contiguous Allocation – supports easy resizing of blocks at runtime
E. Indexed Allocation – eliminates fragmentation and enhances caching

(13) A file system administrator notices that certain small files are consuming a full disk block just to store pointers to the actual file content, leading to poor memory utilization. Which file allocation method is likely being used, and what specific weakness does this highlight?

A. Contiguous Allocation; waste due to block preallocation
B. Linked Allocation; inefficient sequential access
C. Indexed Allocation; pointer overhead for small files
D. Tree Allocation; increased pointer depth
E. Extent-Based Allocation; fragmented extent chains

(14) An operating system stores file metadata that includes both the starting block number and the number of blocks allocated for the file. Which allocation method does this metadata structure most likely support?

A. Linked Allocation
B. Indexed Allocation
C. Contiguous Allocation
D. Clustered Allocation
E. Multi-level Indexing Allocation

(15) A file stored using a particular allocation method shows scattered block addresses throughout the disk but maintains accurate data integrity via internal pointers. However, the system suffers from high

latency when accessing deeper blocks. What allocation method is this and why does it impact performance?

A. Indexed Allocation; caused by large pointer table overhead
B. Contiguous Allocation; caused by large seeks between blocks
C. Linked Allocation; due to sequential traversal between non-contiguous blocks
D. Clustered Allocation; due to sector-to-track latency
E. Hash Allocation; due to non-deterministic address location

(16) In a file allocation system, every file is assigned a separate block that holds addresses of all its associated data blocks. This design eliminates the need for each data block to store pointers. Which file allocation method is being referred to, and what key benefit does it offer?

A. Linked Allocation; improved sequential block reading
B. Indexed Allocation; enables direct block access and no external fragmentation
C. Contiguous Allocation; simplified directory maintenance
D. Chained Allocation; efficient memory reuse
E. Clustered Allocation; optimized caching mechanism

(17) A file storage strategy is implemented in an old MS-DOS system where each file's blocks are connected like a linked list. This allows flexibility in adding data but restricts direct access. What is a direct consequence of using this strategy?

A. Unnecessary storage of full index tables
B. Complex file path resolution
C. High overhead due to repetitive compaction
D. Inability to jump to a specific block without sequential traversal
E. Excessive CPU cycles due to recursive compression

(18) A student creates a file system simulation. When retrieving the 5th block of a file stored using a certain method, the system computes the block address as the starting block plus an offset value. Which allocation strategy is this and what does this reveal about its access nature?

A. Contiguous Allocation; allows direct block access with minimal computation
B. Linked Allocation; uses pointer chains to calculate offset
C. Indexed Allocation; maps index entries to random addresses
D. Segmented Allocation; uses dynamic address mapping
E. Hash-Based Allocation; uses hashed offsets for data indexing

(19) An engineer chooses a storage method that supports fast random access but consumes more memory for storing a large number of pointers. Which allocation method aligns with this choice, and in what case might it become inefficient?

A. Contiguous Allocation; when block size is too small
B. Linked Allocation; when file size exceeds memory limit
C. Indexed Allocation; when storing files with only 1–2 data blocks
D. Clustered Allocation; when clusters are fragmented
E. Logical Allocation; when using cloud-based virtual memory

(20) In a custom file system, a user is restricted from inserting a large file because there isn't a large enough contiguous space, even though plenty of free space exists overall. What is the file system likely suffering from, and what allocation method is in place?

A. External fragmentation; Contiguous Allocation
B. Internal fragmentation; Indexed Allocation
C. Paging delay; Linked Allocation
D. Access violation; Tree-Based Allocation
E. Storage overflow; Compacted Allocation

(21) During process scheduling, the CPU is executing a user process when an I/O request is issued. The process is placed into a new state, while the CPU moves on to execute a different process. What state does the original process enter, and what data structure must be updated accordingly?

A. Ready state; Process Queue
B. Blocked state; Interrupt Vector Table
C. Waiting state; Process Control Block (PCB)
D. Terminated state; Memory Allocation Table
E. New state; CPU Scheduling Table

(22) A process has been swapped out from main memory to secondary storage due to limited RAM availability, even though it was previously ready for execution. According to the Seven State Model, which state best describes this process now?

A. Blocked
B. Waiting
C. Suspend Ready
D. Terminated
E. Ready

(23) An operating system needs to efficiently support context switching between multiple threads in a real-time embedded system. Which of the following correctly identifies what gets saved during a context switch and where this data is stored?

A. Only CPU ID and priority; stored in I/O status table
B. Active memory blocks and system cache; stored in disk buffer
C. Program counter, CPU registers, and memory map; stored in the PCB
D. File descriptors and semaphore counts; stored in kernel stack
E. Instruction queue and process ID; stored in interrupt handler

(24) A newly created process enters the ready queue but must wait in secondary storage until main memory becomes available. What state transition and scheduling mechanism are involved in this scenario?

A. From New to Running; FCFS Scheduling
B. From New to Suspend Ready; Swapping
C. From Blocked to Ready; Round Robin
D. From Running to Waiting; Semaphore Signaling
E. From Ready to Terminated; Preemptive Scheduling

(25) In a multi-programmed OS environment, a running process is preempted due to the arrival of a high-priority process. Which of the following statements best describes this transition and its consequence?

A. The running process is killed immediately and sent to Terminated state
B. The high-priority process is forced to wait until the running process yields
C. The running process is moved to Suspend Blocked to avoid deadlock
D. Context switching occurs, and the current process moves to Ready state
E. The preempted process is converted into a kernel thread

(26) Which of the following best explains why context switching is considered a form of overhead in modern multitasking operating systems?

A. It reduces the need for scheduling and thread management
B. It increases CPU speed by skipping unused instructions
C. It consumes valuable processor time without performing actual process execution
D. It avoids memory fragmentation by isolating process memory
E. It helps I/O-bound processes gain exclusive access to RAM

(27) A process transitions from the "Blocked" state to the "Ready" state after completing its I/O request. Which system component is primarily responsible for handling this transition, and how is the process made eligible again?

A. Process Scheduler; adds it to the I/O wait queue
B. Interrupt Handler; sends a termination signal
C. Memory Manager; loads process into ROM
D. I/O Subsystem; triggers an interrupt to resume the process
E. Task Scheduler; suspends the process permanently

(28) Which of the following transitions would involve the least time for context switching, assuming minimal memory movement and preemptive scheduling is enabled?

A. New → Terminated
B. Waiting → Running
C. Ready → Running
D. Suspend Ready → Ready
E. Running → Blocked

(29) A process remains in memory but does not currently need the CPU due to awaiting user input. What is this state classified as, and how does the OS ensure CPU efficiency during this time?

A. Ready; it is kept in CPU cache to speed up processing
B. Blocked; it relinquishes the CPU for other processes
C. Terminated; it frees all allocated memory
D. Running; it is still using CPU time for polling
E. Suspended Ready; it is moved to secondary storage

(30) In the event of frequent switching between CPU-bound and I/O-bound processes, which of the following strategies should the OS implement to maintain optimal system performance?

A. Delay context switching to reduce RAM usage
B. Increase block size for file management efficiency
C. Use priority inversion to prevent high-priority starvation
D. Apply adaptive scheduling algorithms that favor I/O-bound processes
E. Terminate background threads that are idle for more than 1 second

(31) A file system engineer is analyzing disk performance and notices a slight delay while accessing files stored in non-contiguous clusters. However, the system still supports random access efficiently using a table structure stored in memory. Which file system is being used and what enables this efficiency?

A. Linked Allocation; file access via pointer chaining
B. Indexed Allocation; direct block addressing using index block
C. FAT File System; cached FAT structure allowing quick cluster lookups

D. NTFS; journaling file system with MFT
E. Contiguous Allocation; block preloading into RAM

(32) A 980-byte file is stored using a FAT file system with a cluster size of 512 bytes. The file occupies two clusters. What amount of storage is wasted due to slack space and what term best describes this type of waste?

A. 20 bytes; RAM slack
B. 32 bytes; Drive slack
C. 44 bytes; Slack space
D. 64 bytes; Internal fragmentation
E. 100 bytes; External fragmentation

(33) A FAT32 volume stores two identical copies of the FAT structure for protection. Which component within the system specifies the active FAT copy, and how does it differ from FAT12 and FAT16?

A. Boot Sector; FAT32 allows dynamic FAT selection via Flags field
B. Root Directory; references active FAT via directory entries
C. File Control Block; tracks active FAT through inode referencing
D. BIOS Parameter Block; selects FAT through boot-time interrupts
E. Disk Partition Table; stores FAT in cylinder 0

(34) Which of the following best explains how FAT addresses the primary limitation of linked list allocation, and what impact does this have on file access?

A. FAT stores the file data in a bitmap, eliminating external fragmentation
B. FAT maintains a linked list within data blocks to preserve order
C. FAT maintains a separate allocation table in memory for fast random access
D. FAT compresses the block chain into directory entries to reduce size
E. FAT assigns fixed-size pages instead of clusters for file metadata

(35) In the FAT system, each file or directory entry contains a reference to the file's starting cluster and the file length. When retrieving file data, how does the operating system know which subsequent clusters to read?

A. The directory contains the complete block sequence
B. Each file has an index table that is scanned sequentially
C. The FAT table is used to follow the chain of clusters using pointers
D. Cluster addresses are embedded in the file header
E. Sectors are read continuously without referencing tables

(36) Assume a file logically requires 1,400 bytes and is stored on a FAT system where each cluster is 512 bytes. How many clusters will the file occupy and how much slack space will it contain?

A. 2 clusters; 124 bytes slack
B. 3 clusters; 236 bytes slack
C. 3 clusters; 136 bytes slack
D. 4 clusters; 48 bytes slack
E. 2 clusters; 100 bytes slack

(37) A directory entry in FAT16 stores key metadata about a file or subdirectory. Which of the following is **not** typically found in a FAT directory entry?

A. Starting cluster number
B. File size in bytes
C. File’s owner or permission level
D. File name and extension
E. File creation/modification timestamps

(38) Which of the following is **true** regarding cluster numbering and addressing in the FAT file system?

A. Cluster addresses start from 0 and increase sequentially
B. FAT12 and FAT16 skip cluster 1; cluster 1 is reserved for MBR
C. FAT file systems begin cluster addressing at cluster 2
D. Each file's cluster number is stored in the boot sector
E. Cluster addressing is dynamic and changes with each file

(39) Why does increasing the block or cluster size in a FAT file system reduce the number of FAT entries but simultaneously introduce inefficiency?

A. It increases the need for file compression algorithms
B. It creates external fragmentation between files
C. It leads to excessive memory swapping for smaller files
D. It increases internal fragmentation due to larger slack space
E. It forces the file system into journaled mode

(40) In a FAT-based system, a file occupies clusters 5 and 6. The FAT entry for cluster 5 contains the value 6, and the entry for cluster 6 contains a special end-of-chain marker. What does this signify?

A. The file was deleted and its clusters are marked for reuse
B. Clusters 5 and 6 are linked as part of the file’s storage chain
C. Clusters 5 and 6 are bad sectors and skipped during access
D. The file occupies directory clusters and will be reformatted
E. Cluster 6 is reserved for the boot sector