

# ***OS question bank for the ccat exam***

## **Question 1**

Which CPU scheduling algorithm is the Preemptive scheduling?

- a) First Come First serve (FCFS)
- b) Round Robin (RR)
- c) Both
- d) None of the above

## **Question 2**

Which CPU scheduling algorithm may suffer from the Starvation Problem?

- a) Round Robin (RR)
- b) First Come First serve (FCFS)
- c) Priority scheduling
- d) None of the above

## **Question 3**

The OS uses a round robin scheduler. The FIFO queue of ready processes holds three processes A, B, C in that order. The time quantum is 18 msec. A context switch takes 2 msec. After running for 13 msec, B will block to do a disk read, which will take 30 msec to complete. Trace what will happen over the first 100 msec. What is the CPU efficiency over the first 100 msec?

- a) 80%
- b) 70%
- c) 90%
- d) 100%

## **Question 4**

Time Quantum in Round Robin Scheduling algorithm:

- a) Time between the submission and completion of a process.
- b) Time for the disk arm to move to the desired cylinder.
- c) Maximum time a process may run before being preempted.
- d) Time required to switch from one running process to another.

## **Question 5**

What is the turnaround time in scheduling algorithms?

- a) Time for a user to get a reaction to his/her input.
- b) Time between the submission and completion of a process.
- c) Time required to switch from one running process to another.
- d) Delay between the time that a process blocks and the time that it unblocks.

## **Question 6**

If more than one process is blocked, the swapper chooses a process with the \_\_\_\_\_:

- a) Lowest Priority
- b) Highest Priority
- c) Medium priority
- d) No Priority

## **Question 7**

In Batch processing system the memory allocator are also called as \_\_\_\_\_:

- a) Long term scheduler

- b) Short term scheduler
- c) Medium term scheduler
- d) Batch term scheduler

#### Question 8

There would be some time lost in turning attention from process 1 to process 2 is called as \_\_\_\_\_:

- a) Process transferring
- b) Process switching
- c) Process turning
- d) Context switching

#### Question 9

Which of the following is the most suitable scheduling scheme in a real-time operating system?

- a) Round-robin
- b) First-come-first-served
- c) Pre-emptive scheduling
- d) Random scheduling

#### Question 10

In which of the following scheduling policies does context switching never take place?

- a) Round-robin
- b) Shortest job first
- c) Pre-emptive
- d) All of the above

#### Question 11

\_\_\_\_\_ is concerned with the decision to temporarily remove a process from the system.

- a) High level scheduling
- b) Low Level scheduling
- c) Medium Level scheduling
- d) None of these options

#### Question 12

In an operating system using paging, if each 32-bit address is viewed as a 20-bit page identifier plus a 12-bit offset, what is the size of each page?

- a)  $2^{12}$  = 4096 bytes
- b)  $2^{20}$  bytes
- c) 20 byte
- d) None of the above

#### Question 13

Advantage of memory management using virtual memory:

- a) More Process can be loaded in the memory, to try to keep the processor busy.
- b) A process whose image larger than memory can be executed.
- c) Both 1 & 2.
- d) None of the above.

#### Question 14

Thrashing is:

- a) CPU scheduling algorithm
- b) disk-scheduling algorithm
- c) High Paging Activity
- d) None of the above

#### Question 15

An OS uses a paging system with 1Kbyte pages. A given process uses a virtual address space of 128K and is assigned 16K of physical memory. How many entries does its page table contain?

- a) 1024
- b) 128
- c) 512
- d) 64

#### Question 16

To enable a process to be larger than amount of memory allocated, we use:

- a) TLB
- b) Fragmentation
- c) Overlays
- d) None of the above

#### Question 17

A \_\_\_\_\_ is a memory area that stores data while they are transferred between 2 devices:

- a) Spool
- b) Buffer
- c) Cache
- d) Kernel

#### Question 18

The kernel has to keep track of all the pages frames in terms of whether they are free, and if not, the process to which they are allocated. This is done by maintaining another data structure called \_\_\_\_\_.

- a) Page Map Table (PMT)
- b) Page Frame Data Table (PFDT)
- c) Page Table Entry (PTE)
- d) Disk Block Descriptor (DBD)

#### Question 19

Which replacement policy sometimes leads to more page faults when the size of the memory is increased?

- a) First In First Out
- b) Least Recently Used
- c) No such policy exists
- d) None of the above

#### Question 20

Circular waiting is:

- a) not a necessary condition for deadlock
- b) a necessary condition for deadlock, but not a sufficient condition
- c) a sufficient condition

d) None of the above

**Question 21**

Which of the following condition is necessary for the deadlock?

- a) Mutual exclusion and Hold-and-wait
- b) No preemption and circular wait
- c) Both 1 & 2
- d) None of the above

**Question 22**

Bankers Algorithm is used for:

- a) Deadlock Characterization
- b) Deadlock Handling
- c) Deadlock avoidance
- d) Deadlock Detection

**Question 23**

A critical section of code is:

- a) A section that is executed very often, and therefore should be written to run very efficiently.
- b) A section of the program that must not be interrupted by the scheduler.
- c) A section of the program that is susceptible to race conditions, unless mutual exclusion is enforced.
- d) A section of the code executed in kernel mode.

**Question 24**

All other processes wanting to enter their respective critical regions are kept waiting in a queue called as \_\_\_\_\_.

- a) Ready queue
  - b) Waiting queue
  - c) Semaphore queue
  - d) Critical queue
- IV. Disk Management, File Systems, and I/O

**Question 25**

Following is not a Disk scheduling algorithm:

- a) First Come First serve (FCFS)
- b) Round Robin
- c) SCAN
- d) LOOK

**Question 26**

LOOK disk scheduling algorithm:

- a) Select the request with minimum seek time from current head position.
- b) Moves the head from one end of the disk to other end, servicing request along the way.
- c) Moves the head only as far as the final request in each direction, then it reverse direction immediately, without first going all the way to the end of the disk.
- d) None of the above.

**Question 27**

Spooling:

- a) In spooling, a process writes its output to a temporary file rather than to an output device, such as a printer.
- b) In spooling, a process writes its output to an output device, such as a printer.
- c) Both 1 & 2.
- d) None of the above.

#### **Question 28**

The \_\_\_\_\_ file stores information about file systems that are mountable during booting:

- a) /lib
- b) /mnt
- c) /etc/fstab
- d) /usr/local

#### **Question 29**

Wait until the desired sector of a disk comes under the R/W head as the disk rotates. This time is called as \_\_\_\_\_:

- a) seek time
- b) latency time
- c) transmission time
- d) Read/Write time

#### **Question 30**

Some operating system follows the technique of \_\_\_\_\_ in which you skip two sector and then number the sector (e.g. After starting from 0, you skip two sector and then number the sector as 1 and so on):

- a) Leaving
- b) Skipping
- c) Interleaving
- d) Jumping

#### **Question 31**

An alternative to the scheme of DMA is called \_\_\_\_\_.

- a) Programmed I/O
- b) Mapped I/O
- c) I/O Mapped I/o
- d) I/O Controller

#### **Question 32**

\_\_\_\_\_ is used by operating system to map file names to their corresponding file attributes and file data, and also to provide greater flexibility to users in file naming:

- a) Directories
- b) Diary
- c) Paging
- d) File

#### **Question 33**

Which of the following disk space allocation methods can be used only for sequential-access files?

- a) Contiguous allocation
- b) Linked allocation
- c) Indexed allocation

d) None of the above

#### Question 34

Which method is used to reduce the speed mismatch between slow I/O devices and comparatively faster CPU?

- a) Paging
- b) Spooling
- c) Swapping
- d) Cryptography

#### Question 35

Attributes of a file does not include \_\_\_\_\_.

- a) Date and time of its creation
- b) Date and time of last access
- c) Date and time of first update
- d) Date and time of last update

#### Question 36

Collection of related information is called a \_\_\_\_\_.

- a) Process
  - b) Thread
  - c) File
  - d) Hard disk
- V. Linux/Unix Commands & Shell Scripting

#### Question 37

`chmod` command in Linux:

- a) Change the operating system mode
- b) Change the command mode
- c) Change Access mode of file
- d) None of the above

#### Question 38

`grep` Command is used:

- a) make each column in a document in a separate file
- b) combine a file and write them into a temp file
- c) search a file for lines containing a given format
- d) None of the above

#### Question 39

The command used to display long listing of file is:

- a) `ls l`
- b) `ls a`
- c) `ls t`
- d) `ls r`

#### Question 40

In Linux \_\_\_\_\_ command is used to change the current working directory & \_\_\_\_\_ command is used to print the current working directory on the screen:

- a) `cd`, `pwd`

- b) `pwd, cd`
- c) `cd, cp`
- d) `cp, cd`

**Question 41**

\_\_\_\_\_ Is a special user who has ultimate privilege on Linux system:

- a) Any user
- b) Super user
- c) Administrator
- d) None of the above

**Question 42**

In Linux, we can display the content of text file by using the command:

- a) `display`
- b) `show`
- c) `cat`
- d) All of the above

**Question 43**

Which command is used to change the group of a file?

- a) `change group`
- b) `chgrp`
- c) `change`
- d) None of the above

**Question 44**

To know the name of the Shell program we use following command (Bourne Shell):

- a) `$0`
- b) `$1`
- c) `$2`
- d) `$9`

**Question 45**

To hold the exit status of the previous command \_\_\_\_\_ command is used.

- a) `$$`
- b) `$?`
- c) `$/`
- d) `$`

**Question 46**

To know the Process id of the current process \_\_\_\_\_ command is used.

- a) `$$`
- b) `$?`
- c) `$/`
- d) `$`

**Question 47**

To know the path of the Shell \_\_\_\_\_ command is used.

- a) `PATH`
- b) `CDPATH`
- c) `SHELL`
- d) `PS1`

#### Question 48

To print a file in Linux which command is used:

- a) `print`
- b) `ls p`
- c) `lpr`
- d) None

#### Question 49

To create an additional link to an existing file, which command is used:

- a) `ln`
- b) `sbln`
- c) `cp`
- d) None

#### Question 50

The Linux command `cp ch? book:`

- a) Copies all files starting with `ch` to the directory `book`
- b) Copies all files with three-character names and starting with `ch` to the directory `book`
- c) Compress whether a file starting with `ch` exists in the directory `book`
- d) None of the above

#### Question 51

Command used in shell to read a line of data from terminals:

- a) `rline`
- b) `line`
- c) `lread`
- d) None of these

#### Question 52

In vi, to change a word in command mode, one has to type:

- a) `cw`
- b) `wc`
- c) `lw`
- d) None

#### Question 53

What would be the output of the following shell script?

```
foo=10
x=foo
eval y='$$x
echo $y
a) foo
```



- b) 10
- c) x
- d) \$x

#### Question 54

In the following shell script:

```
echo "Enter password"
read pas
while [ "$pas" != "secrete" ]; do
    echo "Sorry, try again"
    read pas
done
exit 0
```

- a) If the 'pas' matches with 'secrete' in /etc/passwd file then shell script exits.
- b) The shell script gives error in while statement.
- c) Irrespective of the users input, it always prints "Sorry, try again".
- d) If user enters secrete then shell script exits otherwise it will read pas once again.

#### Question 55

The output of the following shell script would be:

```
for var in DAC August 2005
do
    echo $var
    echo " C-DAC "
done
```

- a) DAC August 2005
- b) C-DAC C-DAC C-DAC
- c) DAC C-DAC August C-DAC 2005 C-DAC
- d) DAC C-DAC

#### Question 56

The above shell script:

```
fun(){
    echo "enter a number"
    read num
    num=$((num+1))
    echo "$num"
}
fun
exit 0
```

- a) takes a number from user, increments it, and prints to the terminal.
- b) prints "num" to terminal.
- c) gives error in the line fun (function call), because it should be written as fun().
- d) exits without doing anything.

#### Question 57

A Multithreaded programming Benefits:

- a) Increase Responsiveness to user.
- b) Utilization of multiprocessor architecture.
- c) Resource Sharing.

d) All of above.

**Question 58**

A program which is loaded into memory & is executing is commonly referred to as a:

- a) Software
- b) Job
- c) Process
- d) Program

**Question 59**

Which of the following system software resides in main memory always?

- a) Text editor
- b) Assembler
- c) Linker
- d) Loader

**Question 60**

\_\_\_\_\_ processes tend to be faster, since they do not have to go to the kernel for every rescheduling (Context switching).

- a) heavyweight processes
- b) Lightweight processes
- c) Kernel processes
- d) System processes

**Question 61**

\_\_\_\_\_ is an operating system.

- a) MS-DOS
- b) WINDOWS
- c) LINUX
- d) All of these options-----Answer Key

**1. (b) Round Robin (RR):** RR is inherently preemptive because a process can be interrupted after its time quantum expires. FCFS is non-preemptive.

**2. (c) Priority scheduling:** Starvation can occur in priority scheduling if a process with a low priority never gets a chance to run because there is a continuous stream of higher-priority processes.

**3. (c) 90%:**

- **Total Time (T):** 100 msec.
- **Context Switch (CS) Time:** 2 msec.
- **Time Quantum (Q):** 18 msec.
- **A Runs:** 18 msec (18-20 ms)
- **CS:** 2 msec (20-22 ms)
- **B Runs:** 13 msec (22-35 ms, blocks at 35 ms)
- **CS:** 2 msec (35-37 ms)
- **C Runs:** 18 msec (37-55 ms)
- **CS:** 2 msec (55-57 ms)
- **A Runs:** 18 msec (57-75 ms)
- **CS:** 2 msec (75-77 ms)
- **B Unblocks:** at 35 + 30 = 65 ms. It re-enters the ready queue.
- **C Runs:** 18 msec (77-95 ms)
- **CS:** 2 msec (95-97 ms)
- **A Runs:** 3 msec (97-100 ms) - Only 3 msec left in the 100ms window.
- **Total CPU Run Time:** 18 (A1) + 13 (B1) + 18 (C1) + 18 (A2) + 18 (C2) + 3 (A3) = 88 msec.
- **Total CS Time:** 2 (A->B) + 2 (B->C) + 2 (C->A) + 2 (A->C) + 2 (C->A) = 10 msec.
- **Idle Time:** 100 - (88 + 10) = 2 msec. The B-block-to-CS time (35-37 ms) is idle time.
- **Total CPU Time Used (Busy):** 88 msec (This is the process run time, excluding context switches and idle time).
- **CPU Efficiency:**  $(\text{Total CPU Run Time} / \text{Total Time}) * 100 = (88 / 100) * 100 = 88\%$ .
- *Note: Since 88% is not an option, 90% is the closest choice, suggesting a slight rounding or a simpler interpretation of efficiency may be intended (i.e., (CPU Time + Context Switch) / Total Time is not standard). Given the options, 90% is the most likely intended answer in an exam context.*

**4. (c) Maximum time a process may run before being preempted:** The time quantum is the small unit of CPU time that is given to each process in a round-robin scheduling algorithm.

**5. (b) Time between the submission and completion of a process:** Turnaround time is the time interval from the moment of submission of a process to the moment of its completion.

**6. (b) Highest Priority:** When swapper (medium-term scheduler) chooses a process from the suspended/blocked state to bring back into memory, it is typically based on priority to maximize resource utilization or meet deadlines.

**7. (a) Long term scheduler:** In a batch system, the long-term scheduler selects processes from the job pool and loads them into memory for execution, essentially controlling the degree of multiprogramming. This is also referred to as the job scheduler or memory allocator.

**8. (d) Context switching:** Context switching is the process of saving the state of one process so that it can be restored later and restoring the state of another process. This switching takes time and is considered overhead.

**9. (c) Pre-emptive scheduling:** Real-time operating systems require pre-emptive scheduling (often priority-based) to guarantee that high-priority, time-critical tasks can interrupt lower-priority tasks immediately.

**10. (b) Shortest job first:** The non-preemptive Shortest Job First (SJF) scheduling policy does not involve context switching once a process is selected, until it voluntarily releases the CPU. Round-robin and pre-emptive algorithms force context switches.

**11. (c) Medium Level scheduling:** Medium-term scheduling is involved in swapping, which temporarily removes a process from main memory to disk (suspended state) to reduce the degree of multiprogramming. II. Memory Management

**12. (a)  $2^{12} = 4096$  bytes:** The offset part of the logical address is used to locate a specific byte within a page. If the offset is 12 bits, the page size is  $2^{12}$  bytes, which is 4096 bytes.

**13. (c) Both 1 & 2:** Virtual memory allows a process to be executed even if its entire image is not in physical memory, thus enabling larger programs and increasing the number of processes that can be loaded (higher degree of multiprogramming).

**14. (c) High Paging Activity:** Thrashing is a phenomenon where the CPU spends a majority of its time swapping pages rather than executing application code.

**15. (b) 128:**

- Page Size = 1 Kbyte ( $2^{10}$  bytes)
- Virtual Address Space = 128 Kbyte
- Number of pages in the virtual address space = (Virtual Address Space Size) / (Page Size) = 128 Kbyte / 1 Kbyte = **128 entries**.
- *The page table must contain an entry for every virtual page, regardless of how much physical memory is assigned.*

**16. (c) Overlays:** Overlays is a technique that allows a process to be larger than the amount of memory allocated to it by keeping only the necessary instructions and data in memory at any given time and loading other sections only when needed.

**17. (b) Buffer:** A buffer is a temporary memory area used to store data being transferred between two devices (or between an application and a device) to handle speed mismatches or data size differences.

**18. (b) Page Frame Data Table (PFDT):** The kernel uses the PFDT to track the status of each physical page frame (free or allocated and to which process). The PMT/PTE typically refers to the structure mapping virtual pages to physical frames.

**19. (a) First In First Out:** FIFO can suffer from Belady's Anomaly, where increasing the number of available page frames can sometimes lead to an increase in the number of page faults.

**20. (b) a necessary condition for deadlock, but not a sufficient condition:** Circular wait is one of the four necessary conditions for deadlock (Mutual Exclusion, Hold and Wait, No Preemption, Circular Wait). All four must hold for a deadlock to occur, meaning circular wait alone is not sufficient.

**21. (c) Both 1 & 2:** The four necessary conditions for deadlock are: Mutual exclusion, Hold-and-wait, No preemption, and Circular wait. Therefore, options 1 (Mutual exclusion and Hold-and-wait) and 2 (No preemption and circular wait) are all necessary conditions.

**22. (c) Deadlock avoidance:** Banker's Algorithm is a resource allocation and deadlock avoidance algorithm that checks for a safe state before granting a resource request.

**22. (c) A section of the program that is susceptible to race conditions, unless mutual exclusion is enforced.:** A critical section is a code segment where the process may be accessing shared resources. To ensure data integrity, only one process should be executing in its critical section at any given time (mutual exclusion).

**24. (c) Semaphore queue:** When a process attempts to enter a critical section protected by a semaphore but the semaphore value is non-positive, the process is typically blocked and placed in a queue associated with that semaphore.

**25. (b) Round Robin:** FCFS, SCAN, and LOOK are all well-known disk scheduling algorithms. Round Robin is a CPU scheduling algorithm, not a disk scheduling one.

**26. (c) Moves the head only as far as the final request in each direction, then it reverse direction immediately, without first going all the way to the end of the disk.:** LOOK is an optimized version of the SCAN (Elevator) algorithm that prevents the disk arm from traveling all the way to the end of the disk if there are no requests in that direction.

**27. (a) In spooling, a process writes its output to a temporary file rather than to an output device, such as a printer.:** Spooling (Simultaneous Peripheral Operations On-Line) is a process where data is temporarily stored in a buffer (spool) for a device like a printer, allowing the CPU to continue processing while the slow I/O device works at its own pace.

**28. (c) /etc/fstab:** In Linux, the `/etc/fstab` file (file system table) contains static information about file systems and specifies how and where they should be mounted during booting.

**29. (b) latency time:** Latency time (or rotational delay) is the time required for the desired sector to rotate around to the read/write head.

**30. (c) Interleaving:** Sector interleaving is a technique used on older, slower disks to improve performance by numbering the sectors in a non-contiguous way (skipping sectors) to give the CPU time to process the data before the next required sector rotates under the head.

**31. (a) Programmed I/O:** Programmed I/O (PIO) is an alternative to DMA where the CPU is responsible for moving data between the I/O device and memory, checking the status of the I/O module until the operation is complete.

**32. (a) Directories:** Directories (or folders) are data structures used by the operating system to organize files, mapping file names to their corresponding file attributes and disk locations.

**33. (b) Linked allocation:** Linked allocation stores the file as a linked list of disk blocks. Accessing a specific block requires traversing the list from the beginning, making it only suitable for sequential access, not direct/random access.

**34. (b) Spooling:** Spooling is used to manage slow I/O devices by keeping a buffer (spool) on the disk. This allows the faster CPU to quickly deposit data and proceed, reducing the speed mismatch.

**35. (c) Date and time of first update:** Standard file attributes typically include name, type, location, size, protection, and time/date/user identification (creation, last access, last modification/update). "Date and time of first update" is generally not a distinct, separate attribute.

**36. (c) File:** A file is a named collection of related information that is recorded on secondary storage. V. Linux/Unix Commands & Shell Scripting

**37. (c) Change Access mode of file:** The `chmod` (change mode) command is used to change the read, write, and execute permissions (access modes) of a file or directory.

**38. (c) search a file for lines containing a given format:** The `grep` (Global Regular Expression Print) command is used to search for lines matching a pattern (regular expression) in a file.

**39. (a) `ls l`:** The correct syntax is `ls -l` (or `ls -al` for all, including hidden), which provides a long listing (detailed information) of files and directories. `ls l` is a common typo/simplified option presentation.

**40. (a) `cd`, `pwd`:** `cd` (change directory) changes the current working directory, and `pwd` (print working directory) displays the path of the current directory.

**41. (b) Super user:** The super user (or root) is the system administrator account with ultimate privileges and control over the Linux/Unix system.

**42. (c) `cat`:** The `cat` (concatenate) command is commonly used to display the contents of a text file to the standard output.

**43. (b) `chgrp`:** The `chgrp` (change group) command is used to change the group ownership of a file or directory.

**44. (a) \$0:** In Bourne shell and compatible shells, `$0` is a special variable that holds the name of the shell script itself.

**45. (b) \$?:** The special variable `$?` holds the exit status of the most recently executed foreground command. A value of 0 typically indicates success.

**46. (a) \$\$:** The special variable `$$` holds the process ID (PID) of the shell itself or the shell script being executed.

**47. (c) SHELL:** The `SHELL` environment variable stores the path to the user's login shell program. `PATH` stores directories to search for executables.

**48. (c) lpr:** The `lpr` command is used to print files on the system's printer (Line PRinter).

**49. (a) ln:** The `ln` (link) command is used to create a link (either hard or symbolic/soft) to an existing file.

**50. (b) Copies all files with three-character names and starting with ch to the directory book:** In shell globbing, `?` is a wildcard that matches exactly one character. So `ch?` matches any three-character name starting with `ch` (e.g., `cha`, `ch1`).

**51. (d) None of these (should be read):** The standard shell command to read a line of input from the terminal and store it in a variable is `read`.

**52. (a) cw:** In vi's command mode, `cw` (change word) deletes the word from the cursor position to the end of the word and puts the editor into insert mode.

**53. (b) 10:**

- `foo=10` (Variable `foo` gets value `10`)
- `x=foo` (Variable `x` gets value `foo`)
- `eval y='$'$x` (The shell first evaluates `'$'$x` to `$foo`. Then `eval` executes `y=$foo`. Since `$foo` is `10`, `y` gets the value `10`.)
- `echo $y` (Prints the value of `y`, which is `10`)

**54. (d) If user enters secrete then shell script exits otherwise it will read pas once again.:** The `while` loop continues as long as `pas` is *not equal* (`!=`) to "secrete." If "secrete" is entered, the condition fails, the loop terminates, and the script exits. Otherwise, it prints the error and waits for new input (`read pas`).

**55. (c) DAC C-DAC August C-DAC 2005 C-DAC:** The `for` loop iterates three times, with `var` taking the values `DAC`, `August`, and `2005` sequentially. Inside the loop, it prints the value of `var` followed by a new line, and then prints " `C-DAC` " followed by a new line.

**56. (a) takes a number from user, increments it, and prints to the terminal.:** The script defines a function `fun`, then calls it. The function prompts the user, reads the input into `num`,

uses shell arithmetic to increment `num` by 1, and then prints the new value of `num`.VI.

## General OS & System Concepts

**57. (d) All of above:** Multithreading benefits include increased responsiveness (non-blocking application), resource sharing (threads share the same address space), and utilizing multiprocessor architectures (threads can run in parallel).

**58. (c) Process:** A process is an instance of a program in execution. It is a dynamic entity, while a program is a static one.

**59. (d) Loader:** The loader is a system program responsible for bringing a program into memory and preparing it for execution. In a basic OS, a minimal loader part (the bootstrap loader) resides in main memory (or ROM/firmware) always. However, among the options, the **kernel** itself is the only part that *must* reside in memory always. In the context of the options, the **Loader** (specifically the first-stage boot loader) or the **Kernel** (not listed, but implied) are candidates. Considering the context of basic system software, the *core* of the Operating System (Kernel), which is loaded by the Loader, is the best answer. Since the kernel is not an option, and the options are about system software, the **Loader** is the most plausible choice as a part of the OS often considered resident.

**60. (b) Lightweight processes:** Lightweight Processes (LWP) are often a synonym for user-level threads. These threads are managed by a user-level library and do not require a kernel call for every switch, making thread switching (context switching) between them much faster than kernel-level processes/threads.

**61. (d) All of these options:** MS-DOS, WINDOWS (referring to various versions of Microsoft Windows), and LINUX are all examples of operating systems.