ALLAMA IQBAL OPEN UNIVERSITY

USMAN NOTES
www.usmannotes.com

Name:

Registration No.:

Course Code: 3422

Program: PGD (CS)

Semester: Autumn 2026

ALLAMA IQBAL OPEN UNIVERSITY, ISLAMABAD

Department of Computer Science

Course: Introduction to Computer Concepts (3576/3422) • Level: PGD (CS)

Semester: Autumn 2026

Assignment No. 2

Total Marks: 100 • Pass Marks: 50

SOLVED ASSIGNMENT GUIDE — For Study & Reference

Note: This guide is prepared for study and reference purposes only. Students are advised to read the relevant units of the course book carefully, understand the concepts in their own way, and then write the assignment in their own words. Do not copy this guide word for word; use it to understand the subject, check the expected depth of each answer, and prepare your own original submission.

USMAN NOTES www.usmannotes.com

Question 1: The Operating System and Its Main Functions (20 Marks)

What is an operating system? Explain the main functions of an operating system in a computer

Introduction

Every computer, from a desktop in an office to the phone in a pocket, contains one program without which nothing else can run: the operating system. It is the first software loaded when the machine is switched on and the last to close when it is switched off, and all other work — writing documents, browsing the internet, playing media — happens under its control. Because it works silently in the background, beginners often overlook it, yet the operating system is

what turns a collection of electronic circuits into a usable machine. This answer defines the operating system, gives examples, and then explains its main functions in a computer in detail.

What Is an Operating System?

An operating system (OS) is system software that manages the computer's hardware and software resources and provides a working environment and common services for all other programs. It acts as an intermediary between the user and the hardware: the user and application programs never deal directly with the processor, memory or disk; they request services from the operating system, which performs the actual hardware operations on their behalf. When a computer is switched on, a small permanent program checks the hardware and loads the operating system from the disk into memory, a process called booting; from that moment the OS takes charge. Examples of operating systems include Microsoft Windows and macOS for personal computers, Linux in its many distributions for desktops and servers, and Android and iOS for mobile phones and tablets. Some operating systems serve a single user at a time, while server and mainframe operating systems serve hundreds of users simultaneously, but their basic functions are the same.

Main Functions of an Operating System

The main functions of an operating system can be grouped under process management, memory management, file management, device management, user interface, and security and control. Each is explained below and summarised in the table that follows.

1. Process Management

A running program is called a process. The operating system decides which process will use the CPU, for how long, and in what order, switching between processes thousands of times per second. This scheduling is what makes multitasking possible. music can play while a document is typed and a file downloads, because the CPU's time is shared so quickly that all the tasks appear to run at once. The OS also starts processes when programs are opened, terminates them when they are closed, and handles processes that stop responding.

2. Memory Management

Main memory (RAM) is limited and is shared by the operating system itself and all running programs. The OS allocates memory to each program when it starts, keeps a record of which parts of memory are free and which are in use, protects each program's area so that one faulty program cannot corrupt another's data, and reclaims the memory when the program closes. When RAM is insufficient, the OS uses a reserved part of the disk as virtual memory, moving less-used data out of RAM temporarily so that work can continue.

3. File Management

The operating system organises all stored information into files and folders (directories) on disks and other media. It provides the operations to create, save, open, copy, move, rename and delete files; maintains details such as each file's name, size, location and dates; and ensures that data written by a program can be found and read back correctly later. File systems managed by the OS, such as NTFS in Windows, also record permissions that decide who may read or change each file.

4. Device Management (Input/Output Management)

Every attached device — keyboard, mouse, printer, monitor, disk drive, network card — works differently, yet application programs cannot be written separately for each model. The operating system manages devices through small specialised programs called device drivers, so applications simply ask the OS to print, display or save, and the OS translates the request for the particular hardware. It also buffers and queues work, for example holding several print jobs in order (spooling) so that programs are not kept waiting for a slow printer.

5. Providing the User Interface

The operating system provides the interface through which the user interacts with the computer. Modern systems use a Graphical User Interface (GUI) of windows, icons, menus and a pointer, as in Windows and macOS, while some systems also offer a Command Line Interface (CLI) in which instructions are typed, as in the Linux terminal. A friendly, consistent interface is a major reason modern computers can be used by non-specialists.

6. Security, Error Handling and Accounting

The operating system protects the system and its data: it verifies users through accounts and passwords, controls access to files and devices, and separates users from one another on shared machines. It detects and responds to errors — a failing disk, a full memory, an unresponsive program — and informs the user instead of letting the whole system crash. On larger systems it also keeps accounts of how much CPU time, memory and storage each user or program consumes, which supports fair sharing and billing.

Table 1: Main functions of an operating system at a glance

Function	What the Operating System Does	Everyday Example
Process management	Schedules and controls running programs; shares CPU time	Typing a document while music plays and a file downloads
Memory management	Allocates and protects RAM; provides virtual memory	Several programs stay open without interfering with each other
File management	Organises, stores and retrieves files and folders	Saving an assignment in a folder and opening it next day
Device management	Controls hardware through drivers; queues device work	Printing through any printer without special program code
User interface	Provides GUI or command-line interaction	Desktop, icons, menus and windows of Windows
Security and control	User accounts, permissions, error handling	Password login; warning when a program stops responding

USMAN NOTES www.usmannotes.com

Why No Computer Can Work Without an Operating System

The necessity of the operating system becomes obvious if we imagine its absence. A program that wanted to print a document would itself need to contain complete instructions for every model of printer; a program that read a file would need to know the exact physical layout of every kind of disk; and two programs run together would overwrite each other's memory, because nothing would reserve separate areas for them. Programmers would spend nearly all their effort on hardware details instead of the problem the program is meant to solve, and users would need technical knowledge of the machine for the simplest task. The operating system removes this burden once and for all: it is written once, by specialists, to manage the hardware and enforce order, and every other program and user benefits from that work. This is also why choosing and maintaining the operating system — keeping it updated, protecting it with security software and configuring user accounts properly — is treated as the foundation of good computer practice in homes, offices and institutions alike.

Conclusion

In conclusion, an operating system is the essential system software that manages hardware resources and provides the environment in which all applications run. Its main functions — process management, memory management, file management, device management, providing the user interface, and ensuring security and error control — together convert raw hardware into a safe, orderly and easy-to-use computer. Without an operating system, every program would have to control the hardware itself, and practical computing as we know it would not exist.

Question 2: System Software versus Application Software (20 Marks)

Explain the difference between system software and application software. Give examples of each

Introduction

All the instructions that make a computer work are collectively called software, but software is not of one kind. Some programs exist to run and manage the computer itself, while others exist to perform the particular jobs that users actually want done — writing, calculating, designing, communicating or playing. The first kind is called system software and the second application software. Confusing the two is common among beginners, because both are 'programs', yet their purposes, users and behaviour are quite different. This answer explains both types and then sets out the differences between them systematically, with examples of each.

System Software

System software is the software designed to control, coordinate and manage the hardware of the computer and to provide a platform on which other software can run. It is generally supplied by the computer or operating-system manufacturer, is installed once, and starts working the moment the machine is switched on. The most important system software is the operating system (Windows, Linux, macOS, Android), which manages the processor, memory, files and devices. Other system software includes device drivers, which let the operating system communicate with particular hardware; utility programs, which maintain and protect the system — antivirus software, disk cleanup, backup tools, file compression and disk formatting tools; and language translators — assemblers, compilers and interpreters — which convert

programs written by programmers into the machine language the processor understands. The ordinary user rarely interacts with system software directly; its work is visible mainly in the smooth running of everything else.

Application Software

Application software is software designed to perform specific tasks for the user — the actual work for which the computer is bought. It runs on top of the operating system and cannot function without it. Applications fall into two broad groups. General-purpose application packages serve needs shared by almost everyone: word processors (Microsoft Word, LibreOffice Writer) for documents, spreadsheets (Microsoft Excel) for calculations and accounts, presentation software (Microsoft PowerPoint) for slides, database software (Microsoft Access) for records, web browsers (Google Chrome, Mozilla Firefox) for the internet, media players for audio and video, and email and messaging programs for communication. Special-purpose or customised applications are written for one organisation or job: a bank's core-banking software, a hospital's patient system, an airline reservation system, a school's result-preparation program, accounting packages such as QuickBooks, design software such as AutoCAD and Photoshop, and games and educational apps on mobile phones. Users choose, install and remove application software according to their own needs, and they interact with it directly throughout their work.

Differences Between System Software and Application Software

Table 1: Difference between system software and application software

Basis of Difference	System Software	Application Software
Purpose	Manages and controls the computer hardware and provides a platform for other software	Performs specific tasks required by the user
When it runs	Starts when the computer is switched on and runs continuously in the background	Runs only when the user starts it, and stops when the task is finished
Dependence	The computer cannot work without it; it is independent of applications	Cannot run without system software (especially the operating system)
User interaction	Little or no direct interaction; works behind the scenes	The user interacts with it directly all the time
Who provides it	Usually supplied by the computer / OS manufacturer and pre-installed	Bought, downloaded or developed according to the user's needs
Number and variety	Few in number; one operating system serves the whole machine	Very large variety; many applications can be installed on one machine

USMAN NOTES www.usmannotes.com

Examples	Windows, Linux, Android, device drivers, antivirus, disk utilities, compilers	MS Word, Excel, PowerPoint, Chrome, Photoshop, accounting packages, games
Effect of removal	Removing it stops the computer from functioning	Removing it only ends that particular task facility; the computer keeps working

A simple example makes the relationship clear. Consider preparing an assignment on a laptop. Windows (system software) loads the computer, manages its memory and disk, and displays the desktop. The student then opens Microsoft Word (application software) to type the assignment, perhaps checks facts in Chrome (application software) and calculates figures in Excel (application software). Throughout, the antivirus utility (system software) quietly scans files in the background. If Word is uninstalled, the laptop still works perfectly for every other purpose; if Windows is removed, nothing — neither Word nor any other program — can run at all. In the same way, on a mobile phone Android or iOS is the system software, while WhatsApp, the camera app and games are applications. The two types are therefore not rivals but layers: system software creates and maintains the working environment, and application software does the user's work inside that environment.

How the Two Types Work Together: A Layered View

It also helps to picture the software of a computer as layers resting on the hardware. At the bottom, firmware and the operating system — the core of system software — speak directly to

the processor, memory and devices. Utility programs and drivers, also system software, maintain this foundation: a driver update may be needed before a new printer works, and the antivirus must itself be running before any application can be considered safe. Above this foundation sits the application layer, where all user work happens; applications make requests — open a file, print a page, play a sound — and the operating system fulfils them using the hardware. Neither layer can replace the other: even the finest word processor is useless without an operating system to load it, manage its memory and connect it to the printer, while an operating system with no applications is a perfectly managed machine that does no useful work. When a computer feels slow or unreliable, experienced users therefore diagnose by layer: an outdated driver or damaged system file is a system-software problem, while a fault inside one program's document or settings is an application problem. This layered understanding is the practical reason the distinction in this question matters beyond examinations.

A Note on Utility Programs and Translators

Two families of system software deserve a final word because students often misclassify them. Utility programs — antivirus scanners, disk cleanup and defragmentation tools, backup programs, file compressors such as those producing ZIP archives, and disk formatters — are system software even though the user starts them deliberately, because their purpose is to maintain, protect and improve the computer system itself rather than to produce the user's documents or calculations. Language translators are also system software: a compiler translates a complete program written in a language such as C++ into machine code before it runs, an interpreter translates and executes a program line by line, and an assembler translates

assembly language; without them, applications written in human-readable languages could never execute. The test in every doubtful case is purpose: if the program's job is the health, control or preparation of the machine and its software environment, it is system software; if its job is the user's own task — a letter, an account, a design, a game — it is application software.

Conclusion

To conclude, system software and application software differ in purpose, timing, dependence and examples. System software — operating systems, drivers, utilities and translators — runs and manages the computer itself, starts at switch-on, and is essential for the machine to function at all. Application software — word processors, spreadsheets, browsers, business packages and games — performs the user's specific tasks, runs only when called, and depends entirely on the system software beneath it. A useful computer needs both: system software to be a working machine, and application software to be a useful one.

Question 3: Input Devices and Output Devices (20 Marks)

What is an input device and an output device? Explain with at least three examples of each

Introduction

A computer's processor and memory work entirely with electronic signals, while human beings work with key presses, handwriting, speech, pictures and printed pages. The bridge between the human world and the machine is built by input and output devices, collectively called peripheral devices. Input devices translate human actions and real-world information into the digital form the computer understands; output devices translate the computer's digital results

back into forms human beings can see, hear or hold. No computer system can be used without both. This answer defines each type and explains several important examples of each — more than the minimum of three required — noting how each device works and where it is used.

Input Devices

An input device is any hardware component that sends data or instructions into the computer for processing. The data may be typed, pointed, spoken, scanned or sensed, but in every case the device converts it into binary signals that the system can accept. Four important input devices are explained below.

Keyboard

The keyboard is the most common input device, used to enter text, numbers and commands. It consists of alphabetic, numeric, function and special keys arranged on a board; pressing a key sends a specific code for that character to the computer. Keyboards are used for typing documents, entering data, giving commands and filling forms, and they remain the fastest and most accurate way of entering large amounts of text. Variants include laptop keyboards, wireless keyboards and on-screen touch keyboards on phones and tablets.

Mouse

The mouse is a pointing device that controls a pointer (cursor) on the screen. Moving the mouse moves the pointer; clicking, double-clicking and dragging select icons, menus and text, open programs and move objects. Modern mice detect movement optically with a light sensor, and most connect by cable or wirelessly. The mouse made graphical interfaces practical, because selecting an item by pointing is far easier for most users than memorising commands.

Related pointing devices include the touchpad on laptops, the trackball and the graphics tablet used by designers.

Scanner

A scanner converts printed documents, photographs and drawings into digital images that the computer can store, display and edit. The page is placed on the scanner's glass, light and sensors read it line by line, and software saves the result as an image file; with Optical Character Recognition (OCR) software, scanned text can even be converted into editable text. Scanners are used to digitise old records and photographs, to submit documents electronically, and in offices to archive paperwork. Handheld barcode scanners in shops are a specialised form, reading product codes instantly for billing and stock control.

Microphone

A microphone captures sound and voice as input. It is used for recording narration and music, for voice calls and online classes, for dictating text through speech-recognition software, and for giving voice commands to digital assistants. Webcams — small cameras that input live video — are usually grouped with microphones as the input pair for video communication. Other input devices worth knowing include joysticks and game controllers for games, digital cameras, touch screens (which combine input and output), biometric devices such as fingerprint readers used for attendance and security, and sensors used in scientific and industrial systems.

Output Devices

An output device is any hardware component that receives processed data from the computer and presents it to the user in a usable form — as visible display, printed paper or sound. Output

shown on a screen is called soft copy because it disappears when the device is switched off, while printed output is called hard copy because it is permanent. Four important output devices are explained below.

Monitor

The monitor (display screen or VDU) is the principal output device, displaying text, pictures, video and the whole working environment in real time. Modern monitors use LCD or LED technology, and their clarity is described by resolution — the number of pixels on the screen. The monitor gives immediate feedback: everything the user types, selects and creates is first judged on the screen before being saved or printed, which makes it indispensable for interactive work.

Printer

A printer produces hard-copy output on paper (and sometimes on film or card). Inkjet printers spray tiny droplets of ink and are common in homes and small offices for documents and photographs. Laser printers use toner and a laser beam, print faster and more cheaply per page in volume, and are standard in offices. Printers are used for letters, assignments, reports, photographs, tickets and certificates — any output that must be kept, signed, submitted or read away from a computer.

Speakers and Headphones

Speakers and headphones output sound: music, film audio, the spoken words of a recorded lecture, system alerts and the voices in a call. Speakers serve a room or group, while headphones serve one listener privately — important in libraries, offices and online classes. For

users with visual impairments, sound output combined with screen-reading software is a primary channel of access to computing.

Plotter and Projector

Two further output devices deserve mention. A plotter draws large, precise line drawings with pens or inkjets and is used for engineering drawings, architectural plans and maps that ordinary printers cannot handle at that size and accuracy. A multimedia projector displays the computer's screen enlarged on a wall or screen for a class or meeting, and is now a standard output device in classrooms and conference rooms.

Table 1: Input and output devices compared

Aspect	Input Devices	Output Devices
Direction of data	From the user / environment into the computer	From the computer to the user / environment
Purpose	Enter data and instructions for processing	Present the results of processing
Examples	Keyboard, mouse, scanner, microphone, webcam, barcode reader	Monitor, printer, speakers, headphones, plotter, projector
Form of result	Digital signals the system can process	Display (soft copy), print (hard copy) or sound

Conclusion

In conclusion, input devices — such as the keyboard for text, the mouse for pointing, the scanner for printed material and the microphone for sound — convert human actions and real-world information into digital data for the computer. Output devices — such as the monitor for display, the printer for hard copy, and speakers and headphones for sound — convert processed digital results back into forms people can use. Together they make the computer an interactive tool rather than a sealed box, and choosing the right devices for a task is an essential part of setting up any computer system.

Question 4: Cloud Computing and its Benefits for Storage and Access (20 Marks)

Explain the concept of cloud computing. What are its main benefits for data storage and access

Introduction

For most of the history of personal computing, programs and files lived on the user's own machine: documents on the hard disk, software installed from discs, and data lost if the machine was lost. Cloud computing reverses this arrangement. It delivers computing services — storage, software, processing power and databases — over the internet from large, professionally maintained data centres, so that the user needs only a device and an internet connection. Webmail, online document editors and photo backups that people use every day without thinking are all cloud services. This answer explains the concept of cloud computing, its

service and deployment models, and then details its main benefits, especially for data storage and access, as the question asks.

The Concept of Cloud Computing

Cloud computing is the on-demand delivery of IT resources over the internet on a pay-as-you-go or free-tier basis. Instead of buying and maintaining their own servers and software, users rent what they need from a cloud provider such as Google, Microsoft or Amazon, whose data centres contain thousands of connected servers. The word 'cloud' comes from the cloud symbol traditionally used in diagrams to represent the internet: the user neither knows nor needs to know exactly where the data physically sits. A cloud service has several essential characteristics: it is available on demand and largely self-service (the user signs up and starts without negotiating with the provider); it is accessible over the network from many kinds of devices; the provider pools resources to serve many customers together; capacity can be increased or decreased rapidly (elasticity); and usage is measured, so customers pay for what they actually consume. Common everyday examples make the idea concrete: Gmail stores mail in the cloud, Google Drive and Microsoft OneDrive store files, Google Docs provides word processing in a browser, and YouTube stores and streams video — in each case the heavy storage and processing happen on the provider's machines, not the user's.

Service Models and Deployment Models

Cloud services are usually described in three service models, shown in the figure and table below. In Infrastructure as a Service (IaaS), the provider supplies raw computing infrastructure — virtual servers, storage and networks — and the customer manages everything above it. In

Platform as a Service (PaaS), the provider also supplies the operating system and development platform, so developers can build and run applications without managing servers. In Software as a Service (SaaS), the provider supplies complete working applications through the browser, and the customer simply uses them, as with webmail and online office suites. Clouds are also described by deployment: a public cloud serves the general public (Google Drive), a private cloud serves one organisation, a community cloud serves organisations with shared concerns, and a hybrid cloud combines private and public parts.

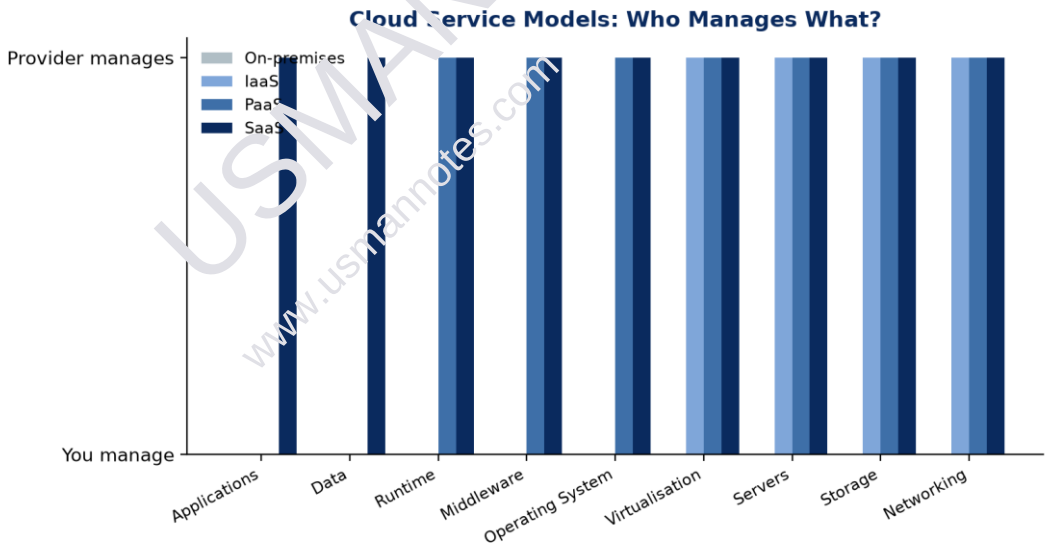


Figure 1: Cloud service models compared — as the model moves from on-premises to SaaS, more of the stack is managed by the provider and less by the user.

Table 1: Cloud service models with examples

Model	What the Provider Supplies	What the User Manages	Examples
IaaS	Virtual servers,	Operating system,	Amazon EC2,

	storage, networking	applications, data	Microsoft Azure Virtual Machines
PaaS	Infrastructure plus operating system and development platform	Own applications and data	Google App Engine, Microsoft Azure App Service
SaaS	Complete application software over the internet	Only own data and settings	Gmail, Google Docs, Microsoft 365 online, Dropbox

USMAN NOTES www.usmannotes.com

Main Benefits for Data Storage and Access

The question's second part concerns storage and access, and here the benefits are substantial. First, access from anywhere and from any device: files stored in the cloud are not tied to one machine, so a student can begin an assignment on a university computer, continue it on a home laptop and read it on a phone, signing in with the same account; this is invaluable for distance learners and office staff who move between locations. Second, automatic backup and protection against loss: cloud providers keep multiple copies of data in different locations, so a stolen laptop, a failed hard disk or a house fire no longer destroys documents and photographs — the data survives because it was never only on the device. Third, easy sharing and collaboration: instead of emailing copies back and forth, users share a link to one cloud document, and several people can view or edit it simultaneously, with changes visible to all in

real time and earlier versions recoverable. Fourth, scalability: storage can grow from a few free gigabytes to terabytes as needs increase, without buying or installing new hardware. Fifth, cost saving: individuals avoid buying large disks and backup devices, and organisations avoid buying servers, employing large maintenance staff and paying for capacity they use only occasionally. Sixth, automatic updating and maintenance: the provider patches, upgrades and secures the systems, while the user's software is always the latest version without any installation effort. Seventh, security and recovery services: reputable providers employ strong encryption, access controls and disaster-recovery systems that a small user could never afford alone, although users must still protect their passwords and choose trustworthy providers. Honest limitations should also be noted: cloud services depend on an internet connection, raise privacy questions because data sits on another company's computers, and can create ongoing subscription costs — so sensitive data and offline backup copies still deserve thought.

Cloud Computing in Education and Everyday Practice

The benefits described above are easy to observe in education. A distance-learning student of Allama Iqbal Open University can keep assignments, notes and past papers in cloud storage and reach them from a mobile phone anywhere, instead of depending on one computer at home; if the phone is replaced, the files are restored simply by signing in again. Teachers share reading material through shared folders rather than photocopying for every student, and a group of students preparing a joint presentation can edit one cloud document from different homes at the same time, seeing each other's contributions immediately. Small businesses similarly keep accounts and customer records in cloud applications, gaining professional backup and security without purchasing a server. These ordinary examples show that the cloud's real contribution is

not a new kind of computer but a new location for computing: storage and software move from the individual's desk to professionally run centres, while access becomes personal, portable and continuous.

Conclusion

In conclusion, cloud computing is the delivery of storage, software and processing over the internet from providers' data centres, offered as IaaS, PaaS and SaaS and deployed in public, private, community or hybrid forms. Its main benefits for data storage and access — anywhere, any-device access; automatic backup and survival of local failures; effortless sharing and real-time collaboration; scalable capacity; lower cost; and provider-managed maintenance and security — explain why cloud storage has become the default way individuals and organisations keep and reach their data.

Question 5: Computer Memory — RAM and ROM Compared (20 Marks)

Write a detailed note on computer memory. Differentiate between RAM and ROM.

Introduction

Memory is the part of the computer that holds data and instructions — both the material being worked on right now and the material kept for the future. A computer without memory could not even start, let alone run programs, because the processor itself can hold only a tiny amount of information at a time. Computer memory is arranged as a hierarchy of types that trade speed against capacity and cost, and beginners most often meet its two famous members: RAM

and ROM. This answer gives a detailed note on computer memory — its units, its hierarchy and its main types — and then differentiates carefully between RAM and ROM, as the question requires.

What Is Computer Memory? Units and Hierarchy

Computer memory stores information in binary form: every item of data, whether a character, a number, a photograph or a program instruction, is held as a pattern of bits (0s and 1s). Eight bits make one byte; storage is then measured in kilobytes (KB), megabytes (MB), gigabytes (GB) and terabytes (TB), each step roughly a thousand times larger than the previous one. Memory types form a hierarchy. At the top are the processor's registers, the fastest and smallest storage, holding the data of the instruction being executed at this instant. Next comes cache memory, a small, very fast memory that keeps frequently needed data close to the processor. Below it sits main memory (RAM), larger and slower than cache but still fast, holding all programs and data currently in use. ROM and its modern descendants hold permanent start-up information. At the bottom, secondary storage (hard disks, SSDs, USB drives) holds by far the most data at the lowest cost per gigabyte, but is the slowest to access; strictly, secondary storage is 'storage' rather than 'memory', though the two are often discussed together. This hierarchy exists because the fastest memories are expensive: the system keeps a little data in fast places and a lot in slow places, and arranges matters so that the processor usually finds what it needs near the top.

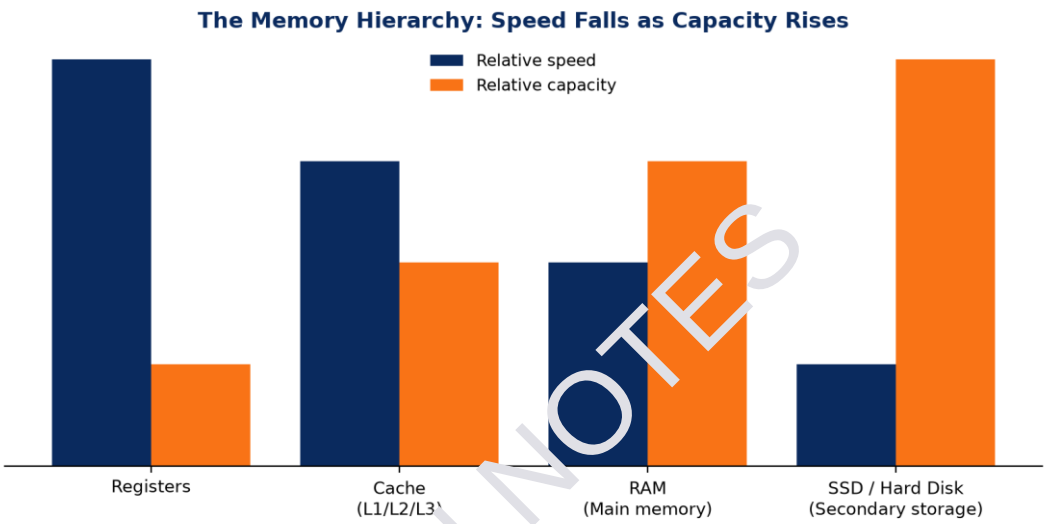


Figure 1: The memory hierarchy — registers and cache are fastest but smallest; secondary storage is slowest but largest.

USMAN NOTES
www.usmannotes.com

Primary Memory: RAM

Random Access Memory (RAM) is the computer's main working memory, also called read/write memory because the processor can both read from it and write to it freely. When a program is started, it is loaded from the disk into RAM; while it runs, its data lives in RAM; and the operating system itself occupies part of RAM from the moment of booting. RAM is volatile: everything in it is lost the instant the power is switched off, which is why unsaved work disappears in a power failure. The word 'random access' means any location in RAM can be reached directly and equally quickly, unlike a tape that must be wound to the right place. The capacity of RAM strongly affects performance: with more RAM (commonly 4 GB to 16 GB or more in modern personal computers), more programs and larger files can be handled together without the machine slowing down by swapping data to disk. The two main RAM technologies

are SRAM (static), fast and used for cache, and DRAM (dynamic), used for main memory in generations such as DDR4 and DDR5.

Primary Memory: ROM and Its Variants

Read Only Memory (ROM) is non-volatile memory whose contents are written permanently (or semi-permanently) and are normally only read, not changed, during ordinary use. Its classic role is to hold the firmware — the start-up instructions (BIOS/UEFI) that test the hardware and load the operating system when the computer is switched on, before any disk is accessible. Because its contents survive without power, ROM guarantees that the machine always knows how to begin. ROM has several variants: PROM can be programmed once by the user; EPROM can be erased with ultraviolet light and reprogrammed; and EEPROM and flash memory can be erased and rewritten electrically — flash technology, developed from EEPROM, is what USB drives, memory cards and SSDs are built from, and modern firmware is stored in reprogrammable flash so it can be updated when necessary.

Difference Between RAM and ROM

Table 1: Difference between RAM and ROM

Basis	RAM (Random Access Memory)	ROM (Read Only Memory)
Full purpose	Main working memory holding programs and data currently in use	Permanent memory holding start-up (firmware) instructions

Volatility	Volatile — contents lost when power is switched off	Non-volatile — contents retained without power
Read / write	Read and write both allowed, freely and repeatedly	Normally read only; writing is impossible or rare and special
Contents	Operating system in use, running programs, data being processed	BIOS/UEFI firmware and fixed routines
When used	Continuously, while the computer is running	Chiefly at start-up (booting) and for fixed control tasks
Capacity	Large — gigabytes (e.g. 8 GB, 16 GB)	Small in comparison — megabytes for firmware
Speed	Very fast access	Slower than RAM in typical use
Cost per unit	More costly than secondary storage; moderate per GB	Cheap per chip, but not sold as expandable memory
Can the user expand it?	Yes — RAM modules can be added or replaced	No — it is a fixed chip on the motherboard
Effect of failure / erasure	Work in progress is lost; machine still starts next time	If firmware is corrupted, the machine may fail to start at

USMAN NOTES www.usmannotes.com

		all

A simple analogy fixes the difference: RAM is like the top of a desk — spacious, cleared every night, holding only the papers being worked on today — while ROM is like the instruction card permanently fixed to the desk that tells the worker how to begin each morning. Both are needed: without ROM the computer would not know how to start and load its operating system; without RAM it would have nowhere to run the programs that the operating system loads.

Secondary Storage in Relation to Memory

Although secondary storage is not, strictly speaking, part of primary memory, a note on memory is incomplete without it, because the two work as partners. Programs and files wait permanently on the hard disk or SSD; when the user opens one, it is copied into RAM, where the processor can work with it quickly; when the user saves, the updated version is written back to the disk. If RAM fills up, the operating system temporarily uses disk space as virtual memory, which keeps the machine working but noticeably more slowly — the familiar slowdown of an overloaded computer, and the practical reason adding RAM is a popular upgrade. Capacity and price explain the division of labour: secondary storage offers hundreds or thousands of gigabytes cheaply but slowly, while RAM offers a much smaller space at a higher price per gigabyte but at the speed the processor demands. Understanding this partnership also explains safe practice: save work frequently, because until it is saved it exists only in volatile RAM.

Conclusion

To conclude, computer memory is the hierarchy of storage — registers, cache, RAM, ROM and secondary storage — that holds the computer's data and instructions in binary form, balancing speed, capacity and cost. Within it, RAM and ROM are complementary opposites: RAM is large, fast, read-and-write but volatile working memory for programs in use, while ROM is small, permanent, non-volatile memory for the firmware that starts the machine. Every computer needs both, and understanding the difference explains everyday experiences such as losing unsaved work in a power cut while the computer itself still starts perfectly the next morning.

USMAN NOTES
www.usmannotes.com