

FBISE - FIRST IN LABELLING

1st Year Computer Science Notes

Complete Chapter-wise Study and Revision Notes

8 Chapters | Explanations | Important Points

Short + Long Question-Answers | MCQs with Answer Keys

Revision Checklist at the End

Original study / revision notes only.

NOT official, NOT leaked, NOT guaranteed. Written in fresh, original wording. Not a textbook or guidebook copy, and not a prediction of any examination paper.

For understanding, practice and revision - in your own words.

Original study / revision notes only - NOT official, NOT leaked, NOT guaranteed.

These notes are written in fresh, original wording for understanding and revision. They are not a textbook, not a guidebook copy, and not an official board publication. They do not reproduce any textbook, guide or website text. No question, topic or mark scheme in this file is claimed to be from any real or future board paper. Use them to learn concepts, practise answering in your own words, and revise before tests - not as a prediction of what will appear in an examination.

Level: 1st Year (Class 11) - Computer Science

Board labelling - first: FBISE

Coverage: 8 chapters, chapter-wise, in one set

What is inside each chapter: simple explanation, important points, short question-answers, long question-answers, and MCQs with an answer key

At the end: a revision checklist for self-checking

How to use these notes

1. Read the explanation first without trying to memorise it. Ask yourself: "Can I explain this to a friend in one minute?"
2. Cover the answers and try the short questions aloud or on paper, then check.
3. Write long answers in your own sentences, using the headings given - introduction, main points, example, conclusion.
4. Attempt the MCQs before looking at the key. A wrong MCQ is useful: note why the other options were wrong.
5. Use the final checklist the night before a test. Tick only what you can actually explain, not what you have merely read.

A good revision rule: understanding first, then wording, then speed. Copying a paragraph from memory is not the same as understanding it.

Contents

- Chapter 1 - Overview of Computer Systems
 - Chapter 2 - Computer Memory
 - Chapter 3 - CPU
 - Chapter 4 - Inside System Unit
 - Chapter 5 - Network Communication and Protocols
 - Chapter 6 - Wireless Communications
 - Chapter 7 - Database Fundamentals
 - Chapter 8 - Database Development
 - Revision Checklist
-

Chapter 1 - Overview of Computer Systems

Explanation

A computer system is not a single box. It is a working combination of parts that take in data, process it in a controlled way, and give back a useful result. When any one part is missing or weak, the whole system works less well, even if the other parts are strong.

Think of it in three simple stages. Input brings data in - for example, typing, tapping, scanning or speaking. Processing works on that data according to instructions. Output presents the result in a form a person can use - on a screen, on paper, as sound, or as a signal sent somewhere else. Storage sits alongside these stages so data and instructions can be kept and used again later.

Hardware is the physical side: the parts you can touch. Software is the instruction side: the programs and rules that tell the hardware what to do. Neither is useful alone. A fast machine with no suitable software does nothing helpful, and good software cannot run without hardware to run it on.

A computer is valued for speed, accuracy, consistency and the ability to store large amounts of data and repeat the same task without getting tired. Its limits matter too. It does not think for itself, it does not judge whether an instruction is sensible, and it only follows what it has been given. Wrong input or a flawed instruction leads to a wrong result, however fast the machine is.

Systems come in different sizes for different jobs - from a phone in a pocket to a large machine serving many users at once - but the same basic idea holds: input, processing, output and storage, controlled by software, working together as one system.

Important points

- A system is hardware + software + data + people + procedures working together.
- The four basic operations are input, processing, output and storage.
- Hardware is physical; software is a set of instructions.
- System software manages the machine; application software does a specific job for the user.
- A computer follows instructions exactly - it does not add common sense of its own.
- Data is raw, unprocessed fact; information is data that has been processed into a useful form.
- Speed and accuracy are strengths; lack of judgement and dependence on correct instructions are limits.

Short question-answers

Q1. What is a computer system?

A computer system is a combination of hardware, software, data and users that works together to accept data, process it and produce useful information.

Q2. What is the difference between data and information?

Data is raw and unorganised, such as a list of numbers with no meaning on its own. Information is data after processing, when it has been arranged or calculated so that it answers a question or

supports a decision.

Q3. What is the difference between hardware and software?

Hardware is the physical equipment of a computer. Software is the set of programs and instructions that tells the hardware what to do and in what order.

Q4. Why is storage needed in a computer system?

Storage is needed so data, programs and results can be kept for later use, even after the machine is switched off or a program is closed, instead of being entered again every time.

Q5. Give one strength and one limitation of a computer.

A strength is that it can repeat a task accurately and at high speed without tiredness. A limitation is that it cannot judge whether an instruction is wise or correct; it simply follows it.

Long question-answers

Q1. Explain the basic operations of a computer system with an example.

Every computer system works through input, processing, output and storage. In input, data enters the system, for example a student types marks into a program. In processing, the computer works on that data following instructions - here it adds the marks and works out an average. In output, the result is shown, for example the average and grade appear on the screen. In storage, the marks and the result are saved so they can be checked later without retyping. If any stage is weak - unclear input, a wrong formula, an unreadable display or lost storage - the final result suffers, which is why all four stages must work as one system, not as separate parts.

Q2. Why is a computer called a system rather than a single machine?

It is called a system because its value comes from parts working together, not from one part alone. The keyboard, processor, memory, screen and storage each do a different job, and software coordinates them. A processor without memory cannot hold the data it is working on; memory without software has no instructions to follow; output without processing has nothing meaningful to show. People also form part of the system, because they give input, check output and decide what to do with the result. Remove or break one link and the chain weakens. That interdependence - many parts, one purpose - is exactly what the word system means.

MCQs

1. Which of the following is the correct order of basic computer operations?

- (a) Output, input, storage, processing
- (b) Input, processing, output, storage
- (c) Processing, storage, input, output
- (d) Storage, output, input, processing

2. Processed data that is useful for a decision is called:

- (a) Hardware (b) Software (c) Information (d) Input

3. Which of the following is software?

- (a) Keyboard (b) Monitor (c) Operating system (d) Printer

4. A computer's main limitation is that it:

- (a) Works slowly (b) Cannot follow instructions (c) Has no judgement of its own (d) Cannot store

data

5. Which part is responsible for keeping data for future use?

(a) Input device (b) Storage (c) Output device (d) Speaker

Answer key - Chapter 1: 1 (b), 2 (c), 3 (c), 4 (c), 5 (b)

Chapter 2 - Computer Memory

Explanation

Memory is where a computer keeps data and instructions while it works, and in some cases after it stops. Without memory, the processor would have nothing to work on and nowhere to put a half-finished result.

It helps to divide memory by two questions: how fast can it be reached, and does it survive switching the machine off? The fastest memory sits closest to the processor and holds only a small amount, because speed costs more. Larger, slower memory holds more but takes longer to reach. This trade-off - speed against size and cost - shapes the whole memory system.

Primary memory works directly with the processor. Part of it is temporary: it holds the programs and data currently in use, and its contents are lost when power goes off. Another part is permanent in normal use: it holds the basic start-up instructions the machine needs every time it is switched on, and ordinary users do not change it.

Secondary storage is different. It is not used directly by the processor in the same way, but it keeps large amounts of data for the long term - documents, programs, photos and videos - even without power. Disks, solid-state drives and other external media belong here. The processor copies what it needs from secondary storage into primary memory, works on it there, and saves results back.

Measuring memory is also part of understanding it. Small units build up into larger ones - bits group into bytes, and bytes into kilobytes, megabytes, gigabytes and beyond. Knowing the scale helps you judge, for example, why a short text file is tiny compared with a long video.

Important points

- Primary memory works directly with the processor; secondary storage keeps data long-term.
- Temporary (volatile) memory loses its contents without power; permanent (non-volatile) memory does not.
- RAM is temporary and holds current programs and data; ROM is permanent and holds start-up instructions.
- Cache is small, very fast memory close to the processor that holds frequently used data.
- Secondary storage is larger and cheaper per unit, but slower to access than primary memory.
- Data is moved from secondary storage to primary memory before processing, and saved back afterwards.
- Memory size is measured in bytes and larger units: KB, MB, GB, TB.

Short question-answers

Q1. What is the difference between RAM and ROM?

RAM is temporary memory that holds the programs and data currently being used, and it is cleared when power is lost. ROM is permanent memory that holds essential start-up instructions

and is not cleared when power is lost.

Q2. What is cache memory and why is it used?

Cache is a small, very fast memory placed close to the processor. It stores copies of data and instructions the processor is likely to need again soon, so the processor spends less time waiting for slower main memory.

Q3. Why is secondary storage needed if a computer already has RAM?

RAM is limited in size and loses everything without power. Secondary storage provides large, permanent space for files and programs, so work survives switching off and can be opened again later.

Q4. What does volatile mean in the context of memory?

Volatile means the stored contents depend on a continuous power supply. Switch the power off, and volatile memory is emptied. Non-volatile memory keeps its contents without power.

Q5. Arrange KB, MB, GB and TB in increasing order of size.

KB (kilobyte), MB (megabyte), GB (gigabyte), TB (terabyte) - each step is much larger than the one before it.

Long question-answers

Q1. Explain the memory hierarchy and why it exists.

The memory hierarchy is the arrangement of memory from fastest and smallest to slowest and largest. At the top, closest to the processor, are registers and cache: tiny, very fast, and expensive per unit, so only a little can be provided. Next is main memory (RAM): larger and slower, holding current work. Below that is secondary storage: very large and much cheaper per unit, but slowest to access. The hierarchy exists because no single type of memory can be fast, large and cheap all at once. By combining them, a system gets speed where it matters most - near the processor - and capacity where it matters most - in long-term storage - at a cost that is practical.

Q2. Describe what happens to data when a document is opened, edited and saved.

When a document is opened, a copy is loaded from secondary storage into RAM, because the processor works from RAM, not directly from the disk. While you edit, changes are held in RAM and in cache for speed, so the file on disk is not yet changed. If power fails at this point, unsaved changes in RAM are lost. When you save, the updated version in RAM is written back to secondary storage, replacing or updating the stored file. That is why saving matters: it moves work from temporary memory, where it is at risk, to permanent storage, where it survives.

MCQs

1. Which memory is lost when the computer is switched off?

(a) ROM (b) RAM (c) Hard disk (d) Flash drive

2. Cache memory is used mainly to:

(a) Store videos permanently (b) Speed up the processor by holding frequently used data (c) Replace the hard disk (d) Display images

3. Which is the largest unit in this list?

(a) KB (b) MB (c) GB (d) TB

4. Start-up instructions are normally held in:

(a) RAM (b) Cache only (c) ROM (d) A printer

5. Secondary storage is best described as:

(a) Fast, small and temporary (b) Large, permanent and slower than RAM (c) Part of the processor (d) Used only for sound

Answer key - Chapter 2: 1 (b), 2 (b), 3 (d), 4 (c), 5 (b)

Chapter 3 - CPU

Explanation

The CPU, or central processing unit, is the part of the computer that carries out instructions and controls the rest of the system. It does not store large files and it does not show output by itself - its job is to decide, calculate and coordinate.

Inside the CPU, three broad roles work together. One part performs arithmetic and logical work: adding, comparing and making simple yes-or-no decisions. Another part directs traffic: it reads instructions, works out what each one means, and tells other parts what to do next. A third part is a set of tiny, very fast holding places that keep the data and addresses the CPU is using right now, so it does not have to wait for slower memory at every step.

The CPU works in a repeating cycle. It fetches an instruction from memory, decodes it to understand what is required, and then executes it - performing the calculation, moving data, or deciding the next step. Then it moves to the next instruction and repeats. Millions or billions of these small cycles happen each second, and from them, all larger tasks are built: opening a file, playing a video, or calculating a result.

Speed is not one simple number. Clock speed tells how many cycles happen per second, but the design of the CPU, how many tasks it can handle at once, and how quickly memory can feed it also matter. A well-balanced system matters more than any single figure on its own.

Important points

- CPU stands for central processing unit; it processes instructions and controls the system.
- The ALU performs arithmetic and logical operations, including comparisons.
- The control unit fetches, decodes and coordinates instructions; it does not do the calculations itself.
- Registers are tiny, very fast storage locations inside the CPU for current data and addresses.
- The CPU cycle is fetch, decode, execute, repeated continuously.
- Clock speed is measured in hertz (for example, gigahertz); higher is not the only measure of performance.
- The CPU works closely with memory: instructions and data come from memory and results return there.

Short question-answers

Q1. What is the function of the CPU?

The CPU carries out program instructions, performs calculations and logical decisions, and controls the other parts of the computer so they work in the right order.

Q2. What does the ALU do?

The ALU, the arithmetic and logic unit, performs arithmetic such as addition and subtraction, and logical operations such as comparing two values to decide whether one is greater, smaller or

equal.

Q3. What does the control unit do?

The control unit reads instructions from memory, decodes them, and sends control signals to the other parts, coordinating the sequence of work. It manages the process rather than doing the arithmetic itself.

Q4. What are registers?

Registers are very small, very fast storage locations inside the CPU. They temporarily hold the data, instructions and addresses the CPU is working with at that moment.

Q5. What are the three steps of the CPU cycle?

Fetch (get the next instruction from memory), decode (work out what it means), and execute (carry it out).

Long question-answers

Q1. Explain the fetch-decode-execute cycle with a simple example.

Suppose a program must add two numbers. In the fetch step, the control unit brings the next instruction from memory into the CPU. In the decode step, it interprets that instruction: it learns that an addition is required and which values are involved. In the execute step, the ALU performs the addition, and the result is placed in a register or sent back to memory. The cycle then repeats for the next instruction - perhaps to store the result or display it. A whole program is simply a long, ordered sequence of such cycles, each one small, but together producing a complete task.

Q2. Why can two CPUs with the same clock speed perform differently?

Clock speed only counts cycles per second; it does not say how much useful work is done in each cycle. Design differences matter: how many instructions can be handled at once, how large and effective the cache is, how quickly memory supplies data, and how efficiently the CPU is organised. A CPU that wastes cycles waiting for data will finish less work than a better-balanced CPU at the same clock speed. That is why clock speed should be read as one clue about performance, not as a complete ranking.

MCQs

1. The part of the CPU that performs calculations is the:

(a) Control unit (b) ALU (c) Register only (d) Hard disk

2. The correct order of the CPU cycle is:

(a) Execute, fetch, decode (b) Decode, execute, fetch (c) Fetch, decode, execute (d) Fetch, execute, decode

3. The control unit mainly:

(a) Stores files permanently (b) Coordinates and directs operations (c) Displays output (d) Connects to the internet

4. Registers are best described as:

(a) Large permanent storage (b) Tiny, very fast storage inside the CPU (c) External devices (d) Software programs

5. Clock speed is measured in:

(a) Bytes (b) Hertz (c) Pixels (d) Volts

Answer key - Chapter 3: 1 (b), 2 (c), 3 (b), 4 (b), 5 (b)

Chapter 4 - Inside System Unit

Explanation

The system unit is the main container of a computer's internal parts. From the outside it may look like a plain box, but inside it, the parts are organised so that power, data and control signals can move quickly and safely between them.

The central board inside connects almost everything. The processor sits on it, memory slots hold RAM, and expansion slots allow extra cards to be added. Storage devices, cooling fans and the power supply also connect to or sit within this arrangement. The board does not do the computing itself, but without it the parts would have no common pathway to communicate.

Power is a separate job from processing. The power supply converts mains electricity into the lower, steady voltages the internal parts need, and protects them from irregular supply as far as its design allows. Cooling matters for a similar reason: processors and other parts produce heat when they work hard, and if heat builds up, performance drops or parts can be damaged. Fans and heat sinks move that heat away.

The ports on the outside of the system unit are the meeting points with the outside world. Through them, keyboards, screens, printers, storage devices and networks connect. Inside, buses carry data between parts - think of them as shared roads with rules about who can travel, in which direction, and how much can travel at once. The width and speed of these pathways affect how quickly the whole machine can work.

Important points

- The system unit houses the motherboard, CPU, memory, storage, power supply and cooling.
- The motherboard is the main board that connects and allows communication between parts.
- The power supply converts and supplies suitable voltages to internal components.
- Heat sinks and fans remove heat and protect parts from overheating.
- Ports connect external devices; buses carry data inside the system unit.
- Expansion slots allow additional cards, such as graphics or network cards, to be added.
- Good airflow and a stable power supply protect both performance and lifespan.

Short question-answers

Q1. What is a motherboard?

A motherboard is the main circuit board inside the system unit. It holds the CPU and memory, provides slots and connectors for other parts, and provides the pathways through which they communicate.

Q2. Why is cooling needed inside a system unit?

Internal parts, especially the processor, produce heat during work. Without cooling, heat builds up, performance can fall, and parts may be damaged or stop working reliably.

Q3. What is the job of the power supply?

The power supply converts mains electricity into the lower, steady direct-current voltages that internal computer parts require, and distributes power to them.

Q4. What is a bus?

A bus is a set of pathways inside the computer that carries data, addresses or control signals between components such as the CPU, memory and input/output devices.

Q5. What is a port?

A port is a connection point, usually on the outside of the system unit, where an external device such as a keyboard, monitor, printer or storage device can be attached.

Long question-answers

Q1. Describe the main components inside a system unit and how they relate.

The motherboard forms the base: the CPU is fitted to it, RAM sits in its memory slots, and storage devices connect to it. The power supply feeds all of them with suitable power. Buses on the motherboard carry data between the CPU, memory and other parts, while ports at the edge allow external devices to join the system. Cooling parts keep temperatures within safe limits while this work goes on. No part works in isolation: the CPU needs memory to hold its working data, memory needs storage to load programs from, and all of them need power and a pathway to communicate. The system unit is therefore best understood as an organised housing for a connected team, not as a collection of unrelated parts.

Q2. Why does the design of the motherboard and buses affect overall speed?

Even a fast CPU cannot work faster than the data it receives. If the pathways between CPU, memory and storage are narrow or slow, the CPU spends time waiting, and that waiting limits the whole system. A well-designed motherboard with fast, wide buses lets data flow smoothly, so the processor, memory and storage can keep pace with each other. Conversely, a bottleneck in one pathway slows every task that depends on it, however strong the other parts are. Balance between parts, not the strength of one part alone, decides real-world speed.

MCQs

1. The main board that connects the parts of a computer is the:
(a) Hard disk (b) Motherboard (c) Monitor (d) Keyboard
2. Which component converts mains electricity for internal parts?
(a) CPU (b) RAM (c) Power supply (d) Cache
3. Heat sinks and fans are used to:
(a) Increase storage (b) Remove heat from components (c) Speed up the internet (d) Print documents
4. A bus in a computer is:
(a) A vehicle (b) A pathway carrying data/signals between parts (c) A type of monitor (d) A software program
5. Expansion slots allow a user to:
(a) Add extra cards to the system (b) Remove the CPU permanently (c) Change the operating system (d) Print without a printer

Answer key - Chapter 4: 1 (b), 2 (c), 3 (b), 4 (b), 5 (a)

Chapter 5 - Network Communication and Protocols

Explanation

A network is a set of computers and devices connected so they can share data, resources and services. Instead of every machine working alone, machines on a network can exchange messages, use a shared printer, access common files, or reach the internet through a shared connection.

Communication over a network needs agreement. If one machine sends data in a form another cannot understand, nothing useful happens. That is why networks use protocols: agreed rules that cover how data is addressed, broken into pieces, sent, checked and reassembled. A protocol is not a device or a cable; it is the shared understanding that makes the devices useful together.

To send a message across a network, data is typically divided into smaller packets. Each packet carries part of the message plus addressing and checking information, so it can travel independently and be put back together in the right order at the other end. If a packet is damaged or lost, the system can detect the problem and request it again, rather than losing the whole message.

Networks are described by their size and reach. A small network within one room or building serves nearby devices. A larger network across a city or country links many smaller networks. The largest example is the internet itself: a network of networks, linked by common protocols, allowing machines around the world to exchange data.

Sharing brings benefits - shared resources, easier communication and centralised storage - but also responsibilities. Access must be controlled, data must be protected in transit, and users must understand that a shared system is only as safe as its weakest access point.

Important points

- A network connects devices so they can share data, resources and services.
- A protocol is a set of agreed rules for communication between devices.
- Data is divided into packets for sending, then reassembled at the destination.
- Addressing tells the network where each packet should go.
- Error checking helps detect damaged packets so they can be resent.
- A LAN covers a small area such as a room or building; a WAN covers a large area.
- The internet is a worldwide network of interconnected networks using common protocols.

Short question-answers

Q1. What is a computer network?

A computer network is a group of connected computers and devices that can exchange data and share resources such as files, printers and internet connections.

Q2. What is a protocol?

A protocol is an agreed set of rules that devices follow when communicating, covering matters such as addressing, data format, error checking and the order of sending and receiving.

Q3. What is a packet?

A packet is a small unit into which a larger message is divided for sending over a network. Each packet carries part of the data plus addressing and checking information, and packets are reassembled at the destination.

Q4. What is the difference between a LAN and a WAN?

A LAN (local area network) covers a small area such as a home, classroom or office building and is usually fast and privately managed. A WAN (wide area network) covers a large geographical area, linking many smaller networks, and is usually slower and managed across different organisations.

Q5. Why is error checking needed in network communication?

Data can be damaged or lost during transmission. Error checking allows the receiving side to detect a faulty packet so that a correct copy can be requested, instead of accepting wrong data silently.

Long question-answers

Q1. Explain how a message travels across a network using packets.

First, the message is divided into packets of manageable size. Each packet is given the destination address and a sequence number, so the receiver knows where it belongs and in what order to rebuild the message. Packets then travel across the network, possibly by different routes, through connecting devices. At the destination, packets are collected, checked for damage, and reassembled in sequence. If a packet is missing or damaged, it can be requested again. This method is more reliable and efficient than sending a whole message as one block, because a failure affects only a small packet, not the entire message, and different packets can share network capacity.

Q2. Why are protocols essential for networks to work?

Devices on a network may differ in maker, size and purpose, yet they must still understand each other. Protocols provide the common language and rules: how to address a message, how to format data, how to start and end a transfer, and how to handle errors. Without protocols, each device would send data in its own way and communication would break down. With them, a phone, a laptop and a server can exchange data reliably even though their internal designs differ. Protocols therefore turn a collection of connected machines into a working network.

MCQs

1. A set of agreed rules for network communication is called a:
(a) Packet (b) Protocol (c) Cable (d) Server
2. Dividing a message into small units for sending produces:
(a) Files (b) Packets (c) Printers (d) Cables
3. A network covering one building is most likely a:

(a) WAN (b) LAN (c) Internet (d) Satellite link

4. The internet is best described as:

(a) A single computer (b) A network of networks (c) A printer (d) A cable

5. Packet sequence numbers are used to:

(a) Increase file size (b) Reassemble packets in the correct order (c) Print packets (d) Slow the network

Answer key - Chapter 5: 1 (b), 2 (b), 3 (b), 4 (b), 5 (b)

Chapter 6 - Wireless Communications

Explanation

Wireless communication sends data without a physical cable between sender and receiver, using waves travelling through the air instead. The user experience is freedom of movement: a device can stay connected while moving around a room, a building or a city, without being tied to a socket.

Different wireless methods suit different distances and purposes. Short-range methods connect nearby devices - for example, a phone to earphones, or a laptop to a nearby printer. Medium-range methods cover a home or office, allowing many devices to share one internet connection. Longer-range mobile methods cover towns and cities through a system of base stations, handing a moving device from one station to the next so the connection continues.

Wireless convenience comes with trade-offs. Signals weaken with distance and can be blocked or disturbed by walls, other devices and interference. Because signals travel through open air, they can in principle be picked up by others nearby, so security - such as encryption and password protection - matters more than on a private wired link. Speed also varies: a device close to a strong signal usually performs better than one far away or sharing a crowded connection with many others.

Choosing a wireless method is therefore a judgement about range, speed, security and cost. The best choice for a pair of earphones is not the best choice for connecting a whole school, and a method designed for wide coverage will not always give the fastest speed indoors.

Important points

- Wireless communication uses waves through the air instead of cables.
- Short-range wireless suits nearby devices; Wi-Fi suits homes/offices; mobile networks suit wide areas.
- A wireless access point allows devices to join a local wireless network.
- Signals weaken with distance and can suffer interference from walls and other devices.
- Encryption scrambles data so unauthorised listeners cannot easily read it.
- Passwords and access control limit who can join a wireless network.
- Wireless offers mobility and easy setup, but speed and security need active management.

Short question-answers

Q1. What is wireless communication?

Wireless communication is the sending and receiving of data without cables, using electromagnetic waves travelling through the air between a sender and a receiver.

Q2. What is Wi-Fi?

Wi-Fi is a wireless technology that allows devices such as laptops and phones to connect to a local network and, through it, to the internet, within the range of a wireless access point or router.

Q3. Why does wireless speed fall as distance increases?

As distance increases, the signal weakens and becomes more vulnerable to interference and obstacles. A weaker signal carries data less reliably, so the effective speed drops and disconnections become more likely.

Q4. Why is encryption important for wireless networks?

Because wireless signals travel through open air, they can be intercepted nearby. Encryption scrambles the data so that even if it is picked up, it cannot be read without the correct key.

Q5. Give one advantage and one disadvantage of wireless communication.

An advantage is mobility: devices can connect without being tied to a cable. A disadvantage is that signals are more exposed to interference and unauthorised access, so security must be managed carefully.

Long question-answers

Q1. Compare wired and wireless communication.

A wired connection uses a physical cable, which usually gives a stable, fast and relatively private link, because the signal travels inside the cable to a specific device. Its drawback is lack of movement: a device must stay near a connection point, and installing cables takes effort and cost. A wireless connection removes the cable, giving mobility and easy setup, especially where many devices need to connect. Its drawbacks are variable speed, interference from distance and obstacles, and greater exposure to interception. In practice the two complement each other: wired links often form the reliable backbone of a network, while wireless links provide flexible access for everyday devices.

Q2. Explain how a mobile phone stays connected while its user moves across a city.

A mobile network is divided into areas called cells, each served by a base station. When a phone is switched on, it connects to the nearest suitable base station. As the user moves, the signal from the current station weakens and the signal from a neighbouring station grows stronger. The network then hands the connection over to the new station, ideally without dropping the call or data session. This handover repeats as the user travels, so a single continuous service is built from many short, local connections. Behind the scenes, the network also tracks the phone's general location so incoming calls and messages can be routed to the right cell.

MCQs

1. Wireless communication sends data using:

(a) Only cables (b) Waves through the air (c) Paper (d) Sound only in a room

2. Wi-Fi is most commonly used for:

(a) A single room-to-city mobile call only (b) Local wireless network access in homes/offices (c) Printing without a device (d) Storing files permanently

3. Which factor weakens a wireless signal?

(a) Short distance (b) Walls and distance (c) Encryption (d) A password

4. Encryption is used in wireless networks to:

(a) Increase distance (b) Protect data from unauthorised reading (c) Remove passwords (d) Slow

all users equally

5. A main advantage of wireless over wired is:

(a) Always faster (b) Mobility without cables (c) No need for security (d) No interference ever

Answer key - Chapter 6: 1 (b), 2 (b), 3 (b), 4 (b), 5 (b)

Chapter 7 - Database Fundamentals

Explanation

A database is an organised collection of related data, stored in a structured way so it can be found, updated and used reliably. A list scribbled on paper can hold information, but it is hard to search, easy to duplicate, and difficult to keep consistent when many people use it. A database solves those problems by imposing structure.

The basic building blocks are simple. A table holds data about one kind of thing - for example, students. Each row in the table describes one item, such as one student. Each column describes one attribute of that item, such as a name or a class. A single value where a row and a column meet is one piece of data, such as one student's name.

Good design avoids repeating the same fact in many places. If a student's address is stored in five tables and changes in only one, the database contradicts itself. A unique identifier for each row helps here: it distinguishes one student from another, even if two students share a name, and it allows tables to be linked without copying all the details each time.

A database management system (DBMS) is the software that manages the database. It provides the tools to create tables, add and change data, search with queries, control who can access what, and protect data from accidental loss or inconsistent updates. Users rarely work with stored files directly; they work through the DBMS, which enforces the rules of the design.

Important points

- A database is an organised collection of related data.
- A table stores data about one type of entity; a row is one record; a column is one field/attribute.
- A field holds a single piece of data; a record is a complete set of fields about one item.
- A primary key uniquely identifies each record in a table.
- A DBMS is software that creates, manages, searches and protects a database.
- Good design reduces duplication and inconsistency.
- A query is a request to retrieve specific data from a database.

Short question-answers

Q1. What is a database?

A database is an organised collection of related data, stored systematically so that it can be easily searched, updated and managed.

Q2. What is the difference between a field and a record?

A field is a single attribute, such as a student's name. A record is a complete row - all the fields together that describe one item, such as one student.

Q3. What is a primary key?

A primary key is a field, or combination of fields, whose value is unique for every record in a table. It identifies each record unambiguously, even when other fields, such as names, are repeated.

Q4. What is a DBMS?

A DBMS (database management system) is software that allows users and programs to create, organise, search, update and protect a database, and to control access to it.

Q5. Why is duplication a problem in a database?

If the same fact is stored in several places, updating it in one place but not others creates inconsistency: the database then gives different answers depending on where you look, and it becomes hard to know which version is correct.

Long question-answers

Q1. Explain the structure of a table using the example of a student database.

Imagine a table called Students. Each column is a field: Student ID, Name, Class and Section, for example. Each row is a record describing one student: one row might hold ID 101, a name, a class and a section. The Student ID serves as the primary key because no two students share it, so each row can be identified exactly. If another table, such as Results, needs to refer to a student, it can store just the Student ID rather than copying the student's full details. This structure keeps data organised, searchable and consistent: to find a student, you search by ID or name; to change an address, you change it once, in one place.

Q2. Why is a database better than keeping the same information in separate files?

Separate files quickly become inconsistent. The same name may be spelled differently in different files, an update may be made in one file and missed in another, and searching across files is slow and error-prone. A database centralises the data under one set of rules: fields are defined once, duplication is reduced, and queries can search across related tables in a controlled way. Access control and backup are also easier to manage centrally. The benefit is not merely storage - it is that the data remains accurate, consistent and usable as it grows and as more people use it.

MCQs

1. In a database table, a row is called a:

(a) Field (b) Record (c) Query (d) DBMS

2. A column in a table represents a:

(a) Record (b) Field/attribute (c) Database (d) Report

3. A primary key must be:

(a) Repeated in every row (b) Unique for each record (c) Always a name (d) Empty

4. Software that manages a database is called a:

(a) Browser (b) DBMS (c) Printer (d) Compiler

5. A query is:

(a) A request to retrieve specific data (b) A type of hardware (c) A deleted table (d) A password

Answer key - Chapter 7: 1 (b), 2 (b), 3 (b), 4 (b), 5 (a)

Chapter 8 - Database Development

Explanation

Database development is the process of building a database step by step, from understanding what is needed to creating a working system and keeping it in good shape. Skipping the early steps and jumping straight to tables is a common mistake: a database built without a clear picture of its purpose soon needs rebuilding.

The process begins with analysis: what data must be stored, who will use it, and what questions must it answer? A school database, for example, must answer questions about students, classes and results, but it does not need to store unrelated details. From this understanding, a design is drawn up: which tables are needed, what fields each table will hold, which field will identify each record, and how tables will link to each other.

Next comes implementation: tables are created in a DBMS, fields are given suitable types (for example, text for names, numbers for marks, dates for birth dates), and sample data is entered to test the design. Testing matters: entering realistic examples quickly reveals problems such as a missing field, a wrong link between tables, or a field that is too short for real values.

After testing, the database is put into use, but development does not end there. Data must be maintained - added, corrected and backed up - and the design may need small changes as needs change. A well-developed database is therefore not just a set of tables, but a planned, tested and maintained system.

Important points

- Database development follows stages: analysis, design, implementation, testing and maintenance.
- Analysis identifies what data is needed, who will use it, and what questions must be answered.
- Design decides tables, fields, keys and relationships between tables.
- A relationship links tables, for example through a shared key, so details need not be duplicated.
- Field types must suit the data: text, number, date and yes/no are used differently.
- Testing with realistic sample data reveals design errors before full use.
- Maintenance includes adding, correcting and backing up data, and updating the design when needs change.

Short question-answers

Q1. What are the main stages of database development?

Analysis (understanding needs), design (planning tables, fields, keys and relationships), implementation (creating the database in a DBMS), testing (checking it with sample data), and maintenance (updating, correcting and backing up data in use).

Q2. Why is analysis the first stage?

Because a database must serve a purpose. Without first understanding what data is needed, who

will use it and what questions it must answer, the design may store the wrong data, miss essential fields, or become difficult to use.

Q3. What is a relationship between tables?

A relationship is a logical link between two tables, usually through a shared key, that allows related data to be connected without duplicating it. For example, a Results table can link to a Students table through Student ID.

Q4. Why must field types be chosen carefully?

Field types control what kind of data can be stored and how it can be used. Numbers stored as text cannot be added reliably; dates stored as text cannot be sorted correctly; a field that is too short will cut off real values. Correct types protect accuracy.

Q5. What is the purpose of testing a database before full use?

Testing with realistic sample data checks whether the design works: whether required data can be entered, whether links between tables work, whether queries return correct results, and whether errors or missing fields appear before real users depend on the system.

Long question-answers

Q1. Describe how you would develop a simple database for a school library.

Start with analysis: the library needs to record books, members and loans, and answer questions such as which books are borrowed, by whom, and when they are due. In design, plan three tables: Books (Book ID as primary key, title, author), Members (Member ID as primary key, name, class), and Loans (Loan ID, Book ID, Member ID, issue date, return date). The Loans table links the other two through their IDs, so book and member details are not duplicated in every loan record. In implementation, create these tables in a DBMS with suitable field types - text for titles, dates for loan dates. In testing, enter sample books, members and loans, and try queries such as "list overdue books". Fix any missing fields or broken links found. In maintenance, keep records updated as books are borrowed and returned, and back up the data regularly.

Q2. Explain why maintenance is part of development, not an afterthought.

A database is used, not just built. New students join, books change, addresses change, and mistakes are discovered. If no one maintains the data, it slowly becomes out of date and untrustworthy, however good the original design was. Maintenance - adding new records, correcting errors, removing obsolete data carefully, and backing up - keeps the database accurate and recoverable. Needs also change over time, so the design itself may need small, planned updates. Treating maintenance as part of development means planning for it from the start: who is responsible, how backups are made, and how changes are checked before they go live.

MCQs

1. The first stage of database development is:

(a) Testing (b) Analysis (c) Maintenance (d) Deleting tables

2. Design of a database decides:

(a) Only colours (b) Tables, fields, keys and relationships (c) The price of computers (d) Internet

speed

3. A Loans table linking Books and Members through IDs is an example of:

(a) Duplication (b) A relationship between tables (c) Deleting data (d) A virus

4. Testing a database is done mainly to:

(a) Find design errors before full use (b) Increase file size (c) Remove all users (d) Hide mistakes

5. Regular backup is part of:

(a) Analysis only (b) Maintenance (c) Deleting the database (d) Printing

Answer key - Chapter 8: 1 (b), 2 (b), 3 (b), 4 (b), 5 (b)

Revision Checklist

Tick a box only when you can explain the point in your own words, without looking.

Chapter 1 - Overview of Computer Systems

- ☐ I can define a computer system and name its four basic operations.
- ☐ I can explain data vs. information with an example.
- ☐ I can explain hardware vs. software.
- ☐ I can give a strength and a limitation of a computer.

Chapter 2 - Computer Memory

- ☐ I can explain RAM vs. ROM.
- ☐ I can explain volatile vs. non-volatile memory.
- ☐ I can explain what cache does and why it helps.
- ☐ I can order KB, MB, GB, TB by size.
- ☐ I can explain what happens to unsaved work if power fails.

Chapter 3 - CPU

- ☐ I can name the ALU, control unit and registers and their jobs.
- ☐ I can explain fetch, decode, execute in order.
- ☐ I can explain why clock speed alone does not rank CPUs.

Chapter 4 - Inside System Unit

- ☐ I can explain what a motherboard does.
- ☐ I can explain the jobs of the power supply and cooling.
- ☐ I can explain ports vs. buses.

Chapter 5 - Network Communication and Protocols

- ☐ I can define a network and a protocol.
- ☐ I can explain packets and why messages are divided.
- ☐ I can explain LAN vs. WAN.
- ☐ I can explain why error checking is needed.

Chapter 6 - Wireless Communications

- ☐ I can explain wireless communication in one sentence.
- ☐ I can compare wired and wireless links.
- ☐ I can explain why encryption matters for wireless.
- ☐ I can explain cell handover in simple terms.

Chapter 7 - Database Fundamentals

- ☐ I can explain table, field and record with an example.

- ☐ I can explain a primary key and why it must be unique.
- ☐ I can explain what a DBMS does.
- ☐ I can explain why duplication causes inconsistency.

Chapter 8 - Database Development

- ☐ I can list the five stages of database development in order.
- ☐ I can explain analysis and design in my own words.
- ☐ I can sketch a three-table design for a library or school example.
- ☐ I can explain why testing and maintenance matter.

Final self-check

- ☐ I attempted all 40 MCQs before checking the keys.
- ☐ I wrote at least one long answer per chapter in my own words.
- ☐ I can re-explain any chapter I ticked, one week later, without the notes.

*End of notes. Original study / revision notes only - NOT official, NOT leaked, NOT guaranteed.
Prepared for FBISE 1st Year Computer Science revision. Not published to any website as part of this task.*