ANSWER KEY & MARKING SCHEME · CBSE CLASS 11

Computer System

Computer Science · Chapter 1 · Use this with the Board Paper · Companion to Quick Drill

HOW TO USE

Attempt the Board Paper first (closed-book, full time). Then come here. For 2-mark+ questions, compare your answer to the model. For 3-4 mark questions, also consult the **Topper Templates** below — these show the exact step-by-step structure that scores full marks per CBSE marking-scheme conventions.

MODEL ANSWERS · BOARD PAPER

Section A — Very short / MCQ (1 × 6 = 6 marks)

Q1. Name the three parts that make up the CPU. [1 mark]

Ans: Arithmetic Logic Unit (ALU), Control Unit (CU), and Registers.

Q2. Which type of memory is volatile — RAM or ROM? [1 mark]

Ans: RAM is volatile (it loses its contents when power is switched off).

Q3. How many bytes are there in 1 KB? [1 mark]

Ans: 1 KB = 1024 bytes.

Q4. Give one example of system software and one of application software. [1 mark]

Ans: System software: Operating System (e.g., Windows/Linux). Application software: MS Word (or a web browser).

Q5. How many characters can standard 7-bit ASCII represent? [1 mark]

Ans: 128 characters.

Q6. In which memory is the boot firmware (BIOS/UEFI) stored? [1 mark]

Ans: ROM (Read-Only Memory), because it is non-volatile and available at power-on.

Section B — Short answer (2 × 6 = 12 marks)

Q7. Differentiate between RAM and ROM (any two points). [2 marks]

Ans: RAM: volatile, read-write, main working memory (e.g., DRAM). ROM: non-volatile, mainly read-only, stores boot firmware (e.g., EEPROM).

Q8. Differentiate between primary memory and secondary storage with one example each. [2 marks]

Ans: Primary memory (e.g., RAM): fast, directly accessed by CPU, mostly volatile, small. Secondary storage (e.g., HDD/SSD): slower, not directly executed by CPU, non-volatile, large, stores files permanently.

Q9. Differentiate between system software and application software with one example each. [2 marks]

Ans: System software manages the hardware/platform (e.g., Operating System). Application software performs a user task (e.g., MS Word).

Q10. Arrange registers, RAM, cache and hard disk from fastest to slowest, and state how cost and capacity change down the order. [2 marks]

Ans: Fastest to slowest: Registers → Cache → RAM → Hard disk. As we go down, speed decreases, cost-per-byte decreases, and capacity increases.

Q11. Differentiate between ASCII and Unicode. [2 marks]

Ans: ASCII: 7-bit code, 128 characters, English/Latin only. Unicode: assigns a unique code point to characters of every script (Devanagari, Chinese, emoji), commonly encoded as UTF-8.

Q12. What is meant by the term 'volatile memory'? Give one example. [2 marks]

Ans: Volatile memory loses its stored contents when the power supply is switched off. Example: RAM.

Section C — Long answer (3 × 4 = 12 marks)

Q13. Draw the block diagram of the von Neumann (functional) architecture and briefly explain each unit. [3 marks]

> **Ans:** Diagram: Input Unit → CPU (containing ALU, Control Unit, Registers) ↔ Memory Unit; CPU → Output Unit; all connected by a bus. Input unit accepts data; CPU processes it (ALU computes, CU controls, registers hold current data); memory stores both instructions and data; output unit delivers results. Key idea: stored-program — instructions and data share one memory; the CPU runs the fetch-decode-execute cycle.

Q14. Explain the memory hierarchy of a computer with reference to speed, cost and capacity. [3 marks]

> **Ans:** From top (fast/costly/small) to bottom (slow/cheap/large): Registers → Cache → Primary memory (RAM) → Secondary storage (SSD/HDD). Moving down: speed decreases, cost-per-byte decreases, capacity increases. The hierarchy gives the illusion of memory that is both large and fast at affordable cost by keeping frequently-used data in fast upper levels.

Q15. Describe, step by step, what happens during the boot process from power-on until the operating system takes control. [3 marks]

> **Ans:** 1) On power-on the CPU runs firmware (BIOS/UEFI) stored in ROM. 2) POST (Power-On Self-Test) checks essential hardware. 3) The firmware finds and runs the bootloader on the boot device. 4) The bootloader loads the OS kernel from secondary storage into RAM. 5) The OS takes control, starts its services, and presents the desktop/login. The boot program must reside in non-volatile ROM because at power-on RAM is empty.

Q16. Distinguish between hardware and software, and between the two main categories of software, with examples. [3 marks]

> **Ans:** Hardware = physical, touchable parts (CPU, RAM, keyboard, monitor). Software = instructions/ programs (non-physical). Software splits into: System software, which manages the machine (Operating System, device drivers, language translators, utilities), and Application software, which performs user tasks (word processor, browser, a Python program you write).

★ TOPPER ANSWER TEMPLATES

3 TEMPLATES · MEMORISE THE FORMAT

★ TOPPER TEMPLATE — 3 marks: Draw and explain the block diagram of the von Neumann (functional) architecture of a computer.

Annual

Step 1 [1 mark]	**Name the four functional units**	A computer following the von Neumann model has four functional units: (1) INPUT UNIT — takes data/instructions from the user (keyboard, mouse), (2) CENTRAL PROCESSING UNIT (CPU) — processes data; it contains the ALU, the Control Unit, and registers, (3) MEMORY UNIT — main memory that stores BOTH program instructions and data, and (4) OUTPUT UNIT — sends results to the user (monitor, printer). These are connected by a system BUS.
Step 2 [1 mark]	**State the stored-program key idea**	The defining idea of the von Neumann architecture is the STORED-PROGRAM concept: both the program instructions and the data they work on are stored TOGETHER in the same main memory, in binary form. The CPU repeatedly performs the FETCH-DECODE-EXECUTE cycle — it fetches the next instruction from memory, the Control Unit decodes it, and the ALU executes it.
Step 3 [1 mark]	**Draw the labelled diagram**	Show Input Unit → CPU (with ALU + Control Unit + Registers inside) ↔ Memory Unit, and CPU → Output Unit, all linked by arrows representing the bus. A clean labelled block diagram earns the presentation mark; an unlabelled or jumbled diagram does not.

COMMON LOSS OF MARKS:

- Forgetting to put ALU + CU + registers INSIDE the CPU box.
- Omitting the stored-program / fetch-decode-execute idea — that is the whole point of 'von Neumann'.
- Drawing arrows without labelling them as the bus / data flow.

★ TOPPER TEMPLATE — 2 marks: Differentiate between RAM and ROM (any two points).
Annual

Step 1 [1.5 marks]	**Give two clear contrasting points**	RAM (Random Access Memory): (i) it is VOLATILE — contents are lost when power is switched off; (ii) it is READ-WRITE — the CPU can both read and write data; (iii) it is the main WORKING memory holding currently-running programs. ROM (Read-Only Memory): (i) it is NON-VOLATILE — contents are retained without power; (ii) it is mainly READ-ONLY; (iii) it stores the permanent boot/firmware program (BIOS/UEFI). Present as a two-column table for full marks.
Step 2 [0.5 mark]	**Add one example each**	Example of RAM types: SRAM (used in cache), DRAM (used as main memory). Example of ROM types: PROM, EPROM, EEPROM. Naming a type each shows depth and secures the presentation half-mark.

COMMON LOSS OF MARKS:
- Swapping the volatility — calling RAM non-volatile or ROM volatile (the most common slip).
- Writing only one distinguishing point when 'any two' were asked.
- Giving prose instead of a labelled two-column comparison (examiners prefer tables here).

★ TOPPER TEMPLATE — 3 marks: Explain the memory hierarchy of a computer with reference to speed, cost, and capacity.
Most years

Step 1 [1 mark]	**Order the hierarchy**	From FASTEST/most-expensive/smallest at the top to slowest/cheapest/largest at the bottom: REGISTERS (inside CPU) → CACHE memory (SRAM, very close to CPU) → PRIMARY MEMORY / RAM (DRAM) → SECONDARY STORAGE (SSD, then HDD) → tertiary/offline (optical, tape).
Step 2 [1 mark]	**State the trade-off**	As you move DOWN the hierarchy, speed DECREASES, cost-per-byte DECREASES, and capacity INCREASES. Fast memory is expensive, so we use only a little of it (registers, cache) and back it with cheaper, larger, slower memory (RAM, then disk).
Step 3 [1 mark]	**Explain the purpose**	The hierarchy exists to give the illusion of a memory that is BOTH large AND fast at a reasonable cost. Frequently-used data is kept in the fast upper levels (cache/registers); the bulk sits cheaply on disk. This is why cache 'hits' make a computer feel fast.

COMMON LOSS OF MARKS:
- Putting secondary storage above RAM, or cache below RAM in the speed order.
- Stating the speed trend but not the matching cost and capacity trends.
- Not explaining WHY the hierarchy exists (the cost-vs-speed compromise).

MARKING SCHEME — GENERAL NOTES
- For differentiate questions: present a two-column table; award marks per correct contrasting point (do not award for repeating the same point twice).
- For the von Neumann diagram: ALU, CU and registers must be shown INSIDE the CPU box; the stored-program / fetch-decode-execute idea must be stated for full marks.
- For RAM vs ROM: deduct if volatility is swapped (RAM is volatile, ROM is non-volatile).
- For memory hierarchy: the speed order must run from inside the CPU outward; reward stating the matching cost and capacity trends.
- For the boot process: reward the correct ordered sequence (ROM firmware → POST → bootloader → OS into RAM); note that ROM is used because RAM is empty at power-on.