Year 11 Mock Exams – November 2025

Subject	Computer Science

Title/Topic	**Format**	**Length**
Paper 1 Computer Systems	Written	1hr 30mins
Paper 1 Programming and problem solving	Written	1hr 15mins

In this Advent assessment I will be asked to show I can…

Year 11 Revision List

Paper 1 Revision List (Computer Systems)

1. Data Representation

- Convert between **binary, denary, and hexadecimal**.
- Perform **binary addition** and **binary shifts**.
- Calculate:
  - The smallest/largest number representable with given bits.
  - Number of colours with a given colour depth.
  - Bits needed for a given character set.
- Understand how **hexadecimal numbers** convert to denary.

2. Networks

- Know what an **IP address** is (examples of IPv4 and IPv6).
- Know what a **MAC address** is (format and purpose).
- Compare **wired vs wireless networks** (advantages/disadvantages).
- Understand **network topologies** (star vs mesh).
- Role of a **switch** in a star topology.

3. Operating Systems & Utility Software

- Functions of an operating system (e.g., memory management, peripheral management, user interface).
- Examples of **utility software**:
  - **Encryption** (purpose, how it works).
  - **Defragmentation** (how it organises files for faster access).

4. Images

- How images are stored using binary in a bitmap:
    - **Resolution**
    - **File Size – W x H X Colour Depth**
- What is metadata?
- What factors affect image file size?

5. Storage & Sound

- **Sound sampling**:
    - What sampling is.
    - How **bit depth** affects sound quality and file size.
- Storage devices:
    - Compare **magnetic vs solid state**.
    - Identify other types (optical, cloud).
- Calculate storage space required (MB/GB/TB conversions).

6. CPU & Performance

- Stages of the **fetch-execute cycle**.
- Key **CPU registers** and their purposes (PC, MAR, MDR, ACC).
- Factors affecting CPU performance (clock speed, cores, cache).

7. Embedded Systems

- Characteristics of **embedded systems**.
- Examples of data stored in **ROM** vs **RAM**.
- Why **virtual memory** isn't needed in embedded systems.

Paper 2 Revision List (Computational Thinking, Algorithms & Programming)

1. Programming Constructs

- Identify **sequence, selection, and iteration**.
- Correctly use if, for, and while.

2. Algorithms & Flowcharts

- Complete and interpret **flowcharts**.

- Recognise and correct:
    - **Syntax errors** (rules of language).
    - **Logic errors** (wrong output).
- Trace and correct algorithms step-by-step.

3. Searching & Sorting

- Understand and demonstrate **binary search** (pre-requisite: data must be ordered).
- Recognise **merge sort** and how it works.
- Write/complete **linear search algorithms**.

4. Defensive Design

- Apply techniques like:
    - Input validation.
    - Preventing invalid inputs in programs.
- Identify **inputs and outputs** of a program.

5. Boolean Logic

- Construct and complete **truth tables**.
- Draw and interpret **logic circuits** for given expressions.

6. Strings & Data Types

- Use string operations:
    - length, upper, left, right.
- Concatenate strings in algorithms.
- Choose correct **data types** (string, integer, float).

7. Programming Languages

- Benefits of **low-level languages** (control, efficiency).
- Benefits of using a **compiler vs interpreter**.

8. Algorithm Maintainability

- Improve maintainability:
    - Use **meaningful variable names**.

  - Use **subroutines/functions**.

9. Writing Functions

- Write functions with:
  - **Parameters**
  - **Return values**
  - Logical constraints (e.g., boundaries 1–512 in moveCharacter).

10. Problem Solving with Algorithms

- Use **trace tables** to test correctness.
- Write validation algorithms (normal, boundary, erroneous test data).
- Create algorithms with **loops, selection, and termination conditions**.

11. Databases & SQL

- Write **SQL queries** (SELECT with conditions).
- Retrieve specific fields (StudentID, TeamName, etc.).

12. Computational Thinking

- Apply **abstraction** (removing unnecessary details).
- Apply **decomposition** (breaking down problems into smaller parts).

What should I do to revise and prepare for this assessment?

Revision Strategy

For each item on the revision list-

- Practice the relevant flashcard set on Quizlet
- Revisit the section in your booklets / watch the GCSE pod video
- Create a mind map for the topic
- Complete the relevant practice question

What useful websites/resources could I use to help me prepare?

GCSE Pod
Quizlet
Youtube (“Craigndave OCR GCSE”)