Computing | GCSE

TEACHER RESPONSIBLE FOR COURSE : Mrs C. Barnes

UNIT	CONTENT	ASSESSMENT TYPE
UNIT 01 Computer systems	Candidates answer all questions. Question paper that includes a mixture of short and long answer questions.	Written paper 1 hour 30 mins 80 marks 50% of the qualification
UNIT 02 Computational thinking, algorithms and programming	Candidates answer all questions. Question paper that includes a mixture of short and long answer questions, some of which will require candidates to write program code.	Written paper 1 hour 30 mins 80 marks 50% of the qualification
UNIT 03 Programming project	Candidates create solutions to computing tasks from a set of options supplied by OCR	Controlled assessment Approx 20 hours 40 marks

UNIT 01 - Computer systems

This component will introduce learners to the context through the study of the ethical, legal, Central Processing Unit (CPU), computer cultural and environmental concerns memory and storage, wired and wireless associated with Computer Science. It is networks, network topologies, system expected that learners will draw on this security and system software. It is expected that learners will become familiar with the Programming Project component (03). Topics covered are:

- Systems Architecture
- Memory and storage
- Computer networks, connections and protocols
- Network security
- System software
- Ethical, legal, cultural and environmental impacts of digital technology

UNIT 02 - Computational thinking, algorithms and programming

This component incorporates and builds on the knowledge and understanding gained in Component 01, encouraging learners to apply this knowledge and understanding using computational thinking. Learners will be introduced to algorithms and programming, learning about programming techniques, how to produce robust programs, computational logic, translators and facilities of computing languages and data representation. It is expected that learners will draw on this underpinning content when completing the Programming Project component (03).

Topics covered are:

- Algorithms
- Programming fundamentals
- Producing robust programs
- Boolean logic
- Programming languages and Integrated Development Environments

UNIT 03 - Programming project

OCR will issue three assessment tasks, each consisting of up to three sub-tasks. The set of tasks within the non-exam assessment will provide opportunities for the learners to demonstrate their practical ability in the skills outlined in the specification.

Although this is not part of the final assessment grading it links heavily to paper 2 and will be an excellent opportunity for revision of that paper.

Learners will need to create suitable algorithms which will provide a solution to the problems identified in the task. They will then code their solutions in a suitable programming language. The solutions must be tested at each stage to ensure they solve the stated problem and learners must use a suitable test plan with appropriate test data.

The code must be suitably annotated to describe the process. Test results should be annotated to show how these relate to the code, the test plan and the original problem.

Learners will need to provide an evaluation of their solution based on the test evidence.

The tasks are open-ended and can be approached in a number of ways. Learners should be encouraged to be innovative and creative in how they approach solving the tasks.

The non-exam assessment (NEA) requires no formal controls in terms of access to the internet or access to materials.

All work submitted by a learner must have been done under observation by their teacher and the final report must be only their own work. External sources can be used but must be referenced and no marks can be awarded for materials submitted which are not the learner’s own. In Component 03 learners must think computationally to solve a series of sub-tasks and while doing so create a report detailing the creation of their solution, explaining what they did and why they did it.

The project can be carried out in many ways but is best approached using an iterative process for developing solutions to the sub-tasks such as below:

- **Success criteria** (what will a successful solution be)
- **Planning and design** (flow charts and pseudocode)

This process will allow learners to demonstrate the key elements of computational thinking:

- Thinking abstractly– removing unnecessary detail
- Thinking ahead - identifying preconditions and inputs and outputs
- Thinking procedurally - identifying components of problems and solutions
- Thinking logically - predicting and analysing problems
- Thinking concurrently - spotting and using similarities.