

GSA Curriculum Map 2023-2024: ICT, Business and Computing

Curriculum Map for GCSE Computer Science

We aim to develop all students into successful and competent programmers who:

- have the desire to develop and apply their analytical, problem-solving, design, and computational thinking skills within programming and Computing as a whole;
- become digitally literate – able to use, and express themselves and develop their ideas through information and communication technology – at a level suitable for the future workplace and as active participants in a digital world;
- understand the impacts of digital technology to the individual, wider society.

	Autumn 1 (HT1)	Autumn 2 (HT2)	Spring 1 (HT3)	Spring 2 (HT4)	Summer 1 (HT5)	Summer 2 (HT6)
Year 10 OCR - J277 - Paper 1 & Programming in Python						
Topic Covered	1.1 Systems Architecture 2.2 Programming Fundamentals	1.2 Memory and storage 2.2 Programming Fundamentals	1.3 Communications & Networking 2.2 Programming Fundamentals	1.4 Network Security 2.2 Programming Fundamentals	1.5 Systems Software 2.2 Programming Fundamentals	1.6 Ethics 2.2 Programming Fundamentals
Knowledge deepened	Each stage of the fetch-execute cycle Role/purpose of each component and what it manages, stores, or controls during the fetch-execute cycle How the common characteristics of CPUs affect their performance The purpose and characteristics of embedded systems How to identify examples of embedded systems	Primary Memory Explain the need for Virtual Memory Magnetic, solid state and optical storage Explain the need for Secondary Storage Calculate between different number systems Boolean operators and data types The use of arrays (or equivalent) when solving problems, including both one-dimensional (1D) and two-dimensional arrays (2D) How characters are represented in binary The number of characters stored is limited by the bits	Benefits and drawbacks of wired versus wireless connection IP addressing and the format of an IP address (IPv4 and IPv6) Encryption IP addressing and MAC addressing Common protocols How layers are used in protocols, and the benefits of using layers. The characteristics of LANs and WANs Different factors that can affect the performance of a network The different roles of computers in a client-server and a peer-to-peer network The hardware needed to connect stand-alone	Forms of attack on a Network Describe the posed threats to a Network How to identify and prevent threats to a network Programming constructs for sequence, selection and iteration How to use subprograms (functions and procedures) to produce structured code	The purpose and functionality of operating systems The purpose and functionality of utility software Create basic string manipulation The use of basic file handling operations How to use SQL to query data	The Impacts of digital technology on wider society <ul style="list-style-type: none"> • Privacy • Legal Acts • Ethics • Moral • Environmental

		available How an image is represented as a series of pixels, represented in binary The effect of colour depth and resolution on file size and resolution How sound can be sampled and stored in digital form	computers into a Local Area Network The Internet as a worldwide collection of computer networks Star and Mesh network topologies			
Skills developed	Computer Hardware Programming Physical Computing Problem Solving	Computer Hardware Programming Abstraction Decomposition	Networking Programming Problem Solving Literacy	Programming Debugging Algorithms Decomposition Network Threats	Programming Problem Solving Software	Mathematics Programming Problem Solving Computer Law & Ethics
Links to National Curriculum or Specification	1.1 Systems Architecture 2.2 Programming Fundamentals	1.2 Memory and Storage 2.2 Programming Fundamentals	1.3 Networking 2.2 Programming Fundamentals	1.4 Network Security 2.2 Programming Fundamentals	1.5 System Software 2.2 Programming Fundamentals	1.6 Ethics 2.2 Programming Fundamentals
	NC: develop their capability, creativity and knowledge in computer science, digital media and information technology NC: develop and apply their analytic, problem-solving, design, and computational thinking skills and understand how changes in technology affect safety NC: understand how changes in technology affect safety, including new ways to protect their online privacy and identity, and how to report a range of concerns					
Flagship Link				Computer Science & Business - Legislation and Ethics		
Cross Curriculum Connections			Science (Physics) Autumn 1 - Y11 Autumn Topic: Magnetism Space Link: Electromagnetics with focus on WiFi (data) Science(Chemistry) Autumn 1 - Y10 Topic: Waves Electricity and Magnetism Link: conductivity of materials	Technology: Y11 NEA - Iterative design with problem solving and analytical thinking.		Philosophy Autumn 2 Y11 Topic: Medical Ethics Link: Exploring and explaining the ethics of a situation/scenario.
Resources to support learning	1.1 Systems Architecture	1.2 Memory and Storage 1.2 Data Representation	1.3 Networking	1.4 Network Security	1.5 System Software	1.6 Ethics

	Autumn 1 (HT1)	Autumn 2 (HT2)	Spring 1 (HT3)	Spring 2 (HT4)	Summer 1 (HT5)	Summer 2 (HT6)
Year 11 OCR - J277 - Paper 2 & Programming in Python						
Topic Covered	2.1 Algorithms 2.5 Programming Languages & Integrated Development Environments	2.3 Producing Robust Programs 2.2 Programming Project	2.2 Programming Pseudocode & Exam Practice 2.2 Programming Project	2.4 Boolean Logic 2.2 Programming Exam Practice	Paper 1 & 2 Revision	Paper 1 & 2 Revision
Knowledge deepened	Create and use trace tables to follow an algorithm Understand, describe and traverse Searching and sorting algorithms Design, Write, Test and Refine Programming in Python Understanding of these principles and how they are used to define and refine problems The differences between high- and low-level programming Languages The differences, benefits and drawbacks of using a compiler or an interpreter The tools that an IDE provides Each of the tools and facilities listed can be used to help a programmer develop a program	Defensive design Purpose of testing Why testing is relevant Refine algorithms using suitable test data Ability to create/complete a test plan Combine the use of programming fundamentals to create more complex systems in Python Refine the use of Pseudocode for answering exam questions	Design, Write, Test and Refine Programming in Python Understanding of these principles and how they are used to define and refine problems Produce simple diagrams to show: - The structure of a problem - Subsections and their links to other subsections Complete, write or refine an algorithm Identify syntax/logic errors in code and suggest fixes	Truth tables for each logic gate How to create, complete or edit logic diagrams and truth tables for given scenarios Design, Write, Test and Refine Programming in Python	Revision for Paper 1 and 2 Exam.	Programming Projects
Skills developed	Debugging Algorithms Decomposition	Abstraction Decomposition Mathematics	Abstraction Problem Solving Mathematics	Abstraction Problem Solving Mathematics	Abstraction Decomposition Problem Solving	

Links to National Curriculum or Specification	2.1 Algorithms 2.5 Programming Languages & Integrated Development Environments	2.3 Producing Robust Programs 2.2 Programming Project	2.2 Programming	2.2 Programming 2.4 Boolean Logic		
	NC: develop their capability, creativity and knowledge in computer science, digital media and information technology and develop and apply their analytic, problem-solving, design, and computational thinking skills					
Flagship Link						
Cross Curriculum Connections	Technology: project: Iterative design skills Design Ventura					
Resources to support learning	2.1 Algorithms 2.5 Programming Languages and Integrated Development Environments	2.3 Producing Robust Programs	2.2 Programming Fundamentals	2.4 Boolean Logic 2.2 Programming Fundamentals		