



Our Lady's Catholic College

GCSE Computer Science Curriculum Overview

Our Lady's Catholic College, Morecambe Road, Lancaster, La12rx

GCSE Computer Science

Year 10

Term	Overview	Links to prior learning	
1	SLR 1.1 Systems architecture In System Architecture students will look at the Fetch, Decode and Execute cycle and how this is applied within processors using Von Neumann Architecture. They study the individual registers that make up a processor and how their individual roles support the processing of data and instructions.	8.3 Computer Systems - The purpose of the CPU - Inputs and outputs - The use and manipulation of Binary numbers - Storage Devices - Data units	
	SLR 1.2 Memory and storage Students begin by learning some of the most important fundamentals of Computer Science, such as how data is stored and represented in different parts of the computer and for different types of data.		
	Topics - Introduction to the course - SLR 1.1 Systems architecture 6 lessons - SLR 1.2 Memory and storage – Part 1 5 lessons - Plus 5 dedicated programming lessons	Know SLR 1.1 Systems architecture - What is the purpose of the CPU? - What actions occur at each stage of the fetch-decode=execute cycle. - The role of each component of the CPU. - The purpose of each register and what it stores. - How the common characteristics such as clock speed, cache and number of cores affect the performance of a CPU. - Purpose and examples of embedded systems.	Know How SLR 1.1 Systems architecture - Using the simulator Little man computer can help understand the flow of data inside the CPU - To interpret the technical specifications of a CPU and how different changes will affect performance. - To draw a diagram which represents the Von Neuman Architecture which shows how the CPU works and communicates with the memory to execute instructions.
	Topics	Know	Know How
2	- SLR 1.2 Memory and storage – Part 1 2 lessons - SLR 1.2 Memory and storage (Part 2) 12 lessons - Plus 6 dedicated programming lessons	SLR 1.2 Memory and storage - Why there is a need for primary storage - The difference between RAM and ROM - The purpose of ROM in a computer system - The purpose of RAM in a computer system - How virtual memory works - The use of secondary storage - Describe common types of storage	SLR 1.2 Memory and storage - To Identify suitable secondary storage needed for given scenarios - To calculate required storage capacity for a given set of files. - To Calculate the file sizes of sound, image and text files: - <i>Sound file size = sample rate x duration (s) x bit depth</i> - <i>Image file size = colour depth x image height (px) x image width (px)</i> - <i>Text file size = bits per character x number of characters</i> - To convert denary values to binary digits

		- Explain the characteristics of different types of storage - How data such as numbers, characters, sound and images are stored in a computer - How character sets are logically ordered. - The two different types of compression. - Advantages and disadvantages of compressing files.	- To convert denary values to hexadecimal values and vice versa - To convert binary digits to denary - To convert binary digits to hexadecimal and vice versa - To calculate the resolution of an image.
	Overview	Links to prior learning	
3	SLR 1.3 Computer networks, connections and protocols (a) Students will learn the purpose of function of Computer Networks, from small LAN Networks to the largest Network we use - the Internet. This includes an understanding of the different hardware used and the functions of different topologies used today. (b) Students then look at the different protocols (or rules) that are followed at different levels within the Network layers to enable error-free communication.	7.3 Networking and cyber security Introduction to networking Connection methods such as Wi-Fi 8.3 Computer Systems • Storage units • Data representation • Compression	
	Topics	Know	Know How
	• SLR 1.3 Computer networks, connections and protocols ◦ 12 lessons • Plus 3 dedicated programming lessons	SLR 1.3 Computer networks, connections and protocols (a) - The purpose & types of Networks. - The factor that affect network performance. - The difference between Client Server & Peer to Peer networks. SLR 1.3 Computer networks, connections and protocols (b) - The networking hardware required. - How the Internet functions. - The different types of Network Topologies. - Wired v Wireless connectivity.	SLR 1.3 Computer networks, connections and protocols (a) - To comparing the pros and cons of different types of Network - To consider different factors when designing a Network. SLR 1.3 Computer networks, connections and protocols (b) - To determine the difference between network connection methods and suggest advantages and disadvantages of connection types - To recommend the most appropriate methods of connection.
4	• SLR 1.3 Computer networks, connections and protocols ◦ 2 lessons • SLR 1.4 Network security ◦ 10 lessons • Plus 3 dedicated programming lessons		
	Overview	Links to prior learning	
	SLR 1.4 Network Security The Network Security unit teaches about the threats that are posed to computer systems and networks. Students will then investigate ways to prevent attacks and mitigate damages received by systems. SLR 1.5 System software	7.3 Networking and cyber security • Cyber Crime & Security • Protection of personal data 8.3 Computer Systems • Peripheral devices • Data compression and encryption	

	System Software delves into the built in programs found on computer systems so that students may develop an understanding of their purpose and function. As part of this they will explore operating systems and utility programs.		
	Topics	Know	Know How
5	- SLR 1.4 Network Security 2 lessons - SLR 1.5 System software 6 lessons - Plus 5 dedicated programming lessons	SLR 1.4 Network Security - Forms of Attacks. - Social Engineering & SQL Injection. - Common attack prevention methods. - The reasons for Firewalls and anti-malware software. - The need for user access levels. SLR 1.5 System software - The Purpose of systems software. - Memory management. - The Purpose and functionality of utility software. - The functions of operating systems. - File and peripheral management software. - Encryption software. - Types of user interface. - User management & Defragmentation.	SLR 1.4 Network Security - To identify and prevent vulnerabilities - To identify appropriate security methods to use in different situations - To use some basic SQL statements. SLR 1.5 System software - To identify the features of different User Interfaces - To apply the appropriate utility to a given scenario.
	Overview	Links to prior learning	
6	SLR 1.6 Ethical, legal, cultural and environmental concerns Within this unit students will learn how the developing use of technology can cause issues in the wider society. They will discuss Ethical, Legal, Cultural, Environmental and Privacy issues, and the laws that are associated	7.1 Networking and Cybercrime - Cybercrime and security - to hacking and computer misuse	
	Topics	Know	Know How
	- SLR 1.6 Ethical, legal, cultural and environmental concerns 9 lessons 8 lesson text-based adventure game	SLR 1.6 Ethical, legal, cultural and environmental concerns - Impact of technology on wider society - Ethical & Cultural Issues - Environmental & Privacy Issues - Legislation (Data Protection Act, Computer Misuse, Copyright Law & Software Licencing)	SLR 1.6 Ethical, legal, cultural and environmental concerns - To consider the positive and negative impacts of technology on society and its stakeholders. - To recommend a type of license for a given scenario including the benefits and drawbacks.

Year 11

Term	Overview	Links to prior learning	
1	SLR 2.1 Algorithms Reading, understanding and changing algorithms are an essential Computer Science skill. Students will learn in this unit how to abstract key information out of problems, decompose problems into smaller sections and think "like a computer". They will learn to express algorithms in several ways such as, Flowcharts and Pseudocode. SLR 2.2 Programming fundamentals Programming Fundamentals builds on the introductory programming unit, first introduced in year 8. Students will explore Programming constructs, in addition to several key Python functions. This unit has spanned the entire course, alongside the learning of theory topics.	7.2 Computing Concepts - Computational thinking - Algorithms and flowcharts - Programming constructs - Data types 8.1 Introduction to Programming, 9.2 Python Programming - Use of an interpreter - Mathematical Operators - Computational and logical thinking Python taught throughout Y10	
	Topics	Know	Know How
	- SLR 2.2 Programming fundamentals 9 lessons - SLR 2.1 Algorithms 5 lessons - Plus 3 paper 2 exam revision lessons	SLR 2.1 Algorithms - Abstraction - Decomposition - Algorithmic thinking - Algorithm design - Trace Tables - Common searching and sorting algorithms SLR 2.2 Programming fundamentals - Iteration; condition controlled and count controlled - Exponentiation - The use of Arrays - Using sub programs (functions and procedures) - Variables Vs Constants - Global and local variables - String manipulation techniques - Boolean operators - The use of records and SQL	SLR 2.1 Algorithms - To identifying inputs, process and outputs - To create, interpret and correct given algorithms. SLR 2.2 Programming fundamentals - To identify For and While loops - To Interpret and adapt algorithms - To use a wide range of programming techniques to independently write programs for a range of purposes.
2	Topics	Know	Know How
	- SLR 2.1 Algorithms 13 lessons	SLR 2.1 Algorithms - Abstraction	SLR 2.1 Algorithms

	Plus 7 paper 2 exam revision lessons	- Decomposition - Algorithmic thinking - Algorithm design - Trace Tables - Common searching and sorting algorithms	- To identifying inputs, process and outputs - To create, interpret and correct given algorithms.
	Overview	Links to prior learning	
3	SLR 2.3 Producing robust programs This unit takes programming up a notch! Students will learn how to anticipate misuse within their programs, and implement input validation and sanitisation in order to improve their reliability. They will practice different methods of testing to ensure their programs are fully operational and are free of errors.	7.2 Computing Concepts - Computational thinking - Algorithms and flowcharts - Programming constructs - Data types 8.1 Introduction to Programming, 9.2 Python Programming - Use of an interpreter - Mathematical Operators - Computational and logical thinking Python taught throughout Y10	
	Topics	Know	Know How
	- SLR 2.3 Producing robust programs 8 lessons - Plus 7 paper 2 exam revision lessons	SLR 2.3 Producing robust programs - Defensive design considerations; anticipating misuse and authentication - Maintainability such as use of sub programs; naming conventions, indentation and commenting - Types of testing; iterative and terminal - Using suitable test data - Identifying syntax and logic errors - Refining algorithms - Input validation - Purpose of testing.	SLR 2.3 Producing robust programs - To spot issues and predict program outcomes - To suggest test data to enter into programs - To understand which parts of a program will need to be explained by the use of comments.
	Overview	Links to prior learning	
4	SLR 2.4 Boolean logic Students will be introduced to several logic gates found inside computer systems. They will then investigate how logic is used within computers in order to solve problems through the use of logic diagrams and truth tables.	8.1 Introduction to Programming, 9.2 Python Programming - Use of an interpreter - Mathematical Operators - Computational and logical thinking	

	SLR 2.5 Programming languages and IDEs This unit explores the different generations of programming languages and the differences between them. It also teaches the different tools that are integrated within development environments in order to support programmers when creating code.	Python taught throughout Y10	
	Topics	Know	Know How
	• SLR 2.4 Boolean logic ◦ 5 lessons • SLR 2.5 Programming languages and IDEs ◦ 6 lessons • Plus 4 paper 2 exam revision lessons	SLR 2.4 Boolean logic - Logic gates AND, OR and NOT - Truth tables - Understanding logic diagrams - Combining gates to solve problems SLR 2.5 Programming languages and IDEs - Characteristics and purpose of different levels of programming language - Purpose of translators - Compilers and interpreters - Tools and facilities available in an IDE such as: editors, error diagnostics, run-time environment, translators.	SLR 2.4 Boolean logic - To create, complete or edit logic diagrams and truth tables for given scenarios - To work with more than one gate in a logic diagram. SLR 2.5 Programming languages and IDEs - To use a range of IDEs along with its tools and features - To use the error checking facility when debugging programs -The basic use of assembly code enables us to write programs.
	Topics		
5	• Further programming experience with the exam revision unit. • Additional time to complete the programming challenges and super challenges presented throughout the SLR theory units and from the dedicated programming resources.		