



Key Stage 4: Year 10 Sept 2021/2022

Term	Topic	Covered in lessons	Intent	NC Focus 1	NC Focus 2	Assessment
HT 1	2.1 Algorithms Computational Thinking 2.1.3 Searching and sorting algorithms	<ul style="list-style-type: none"> Standard searching algorithms: Binary search Linear search Standard sorting algorithms: Bubble sort Merge sort Insertion sort 	Computational Thinking forms the foundation for the entire course. Embedding these skills will allow students to be able to approach real world problems logically and understand the workings of the computer	Design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems	Understand several key algorithms that reflect computational thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility of alternative algorithms for the same problem	End of Topic test
	1.2 Memory and Storage: 1.1.2 – Primary storage	<ul style="list-style-type: none"> The need for primary storage The difference between RAM /ROM The purpose of ROM in a computer system The purpose of RAM in a computer system Virtual memory 	Learn where different types of data can be stored understand the hardware and software	understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems	understand how instructions are stored and executed within a computer system; understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits	End of Topic Test and Exam questions
	1.2.2 - Secondary memory	<ul style="list-style-type: none"> Need for Secondary storage Common types of Secondary storage Suitable devices and media Characteristics of storage 	Learn about external storage	understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems	understand how instructions are stored and executed within a computer system; understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits	End of Topic Test and Exam questions
	1.2.3 Units 1.2.4 – Data Storage 1.2.5 - Compression	Recap from year 9: <ul style="list-style-type: none"> Units of data Data storage Character sets Images (Sound) Compression 	Learn how computers understand and make use of data	understand how instructions are stored and executed within a computer system; understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits		End of Topic Test and Exam questions



Key Stage 4: Year 10 Sept 2021/2022

Term	Topic	Covered in lessons		Intent	NC Focus 1	NC Focus 2	Assessment
HT2	<p>2.5 <u>Programming languages and integrated development Environments</u></p> <p>Languages(Translators and Facilitators) SQL</p>	<p>High / Low level Low</p> <ul style="list-style-type: none"> •Translators •Compiler / Interpreter 		<p>Learn use of the techniques in a high-level language</p> <p>Learn the data types in a high-level language</p> <p>Learn the additional programming techniques I</p>	<p>Use two or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions</p>	<p>Develop and apply their analytic, problem-solving, design, and computational thinking skills</p>	<p>End of Topic Test: High/Low level languages</p> <p>Exam questions</p>
	<p>2.2 <u>Programming</u></p> <p>2.2.2 Data types</p> <p>Text Based Programming</p> <p>2.1.2 <u>Designing, creating and refining algorithms</u></p>	<ul style="list-style-type: none"> • Data types • Techniques • Variables • Constants • Operators • In/Outputs • Assignment • Sequence • Selection • Iteration • Data types 	<ul style="list-style-type: none"> • Arithmetic Operators • Computer related maths • Strings • File handling • Arrays/sub programs 	<p>Further develop the fundamental techniques and concepts of text based programming.</p>	<p>Use two or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions</p>	<p>Develop and apply their analytic, problem-solving, design, and computational thinking skills</p>	<p>Project Based</p>
	<p>2.4 <u>Boolean Logic</u></p>	<ul style="list-style-type: none"> • AND/OR/NOT Gates • Truth tables 		<p>Understand why data needs to be in binary form and how transistors in computers are used to make decision</p>	<p>computational thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility of alternative algorithms for the same problem]</p>	<p>Understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming; understand how numbers can be represented in binary, and be able to carry out simple operations on binary numbers [for example, binary addition, and conversion between binary and decimal</p>	<p>End of Topic Test: Logic Gates</p> <p>Exam questions</p>



Key Stage 4: Year 10 Sept 2021/2022

Term	Topic	Covered in lessons	Intent	NC Focus 1	NC Focus 2	Assessment
HT3	<p>Robot Races</p> <p>2.2 – Programming fundamentals</p> <p>2.3 – Producing robust programs:</p> <p>2.3.1 Defensive design</p> <p>2.3.2 Testing</p>	<ul style="list-style-type: none"> Programming fundamentals Data types Additional programming techniques <p>Defensive design considerations: Anticipating misuse Authentication</p> <ul style="list-style-type: none"> Input validation <p>Maintainability: Use of sub programs Naming conventions Indentation Commenting</p> <ul style="list-style-type: none"> The purpose of testing Types of testing: Iterative Final/terminal Identify syntax and logic errors Selecting and using suitable test data: Normal Boundary Invalid/Erroneous Refining algorithms 	Learn fundamentals of programming with the use of physical objects	<p>use two or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions</p> <p>design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems</p>	<p>develop and apply analytic, problem-solving, design, and computational thinking skills</p> <p>undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users</p>	Project Based



Key Stage 4: Year 10 Sept 2021/2022

Term	Topic	Covered in lessons	Intent	NC Focus 1	NC Focus 2	Assessment
HT4	1.3 Computer Networks, Connections and Protocols: 1.3.1 Networks and Topologies	<ul style="list-style-type: none"> LAN/WAN Performance Client server/Peer to Peer Hardware Internet Topologies 	Understand how different networks are made up and how they work	understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems	design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems	End of Topic Test and Exam questions
HT5	1.3.2 - Wired and Wireless Networks, protocols and Layers	<ul style="list-style-type: none"> Wired/Wireless Encryption IP/MAC addressing Standards Protocols and Layers 	Understand connectivity of networks and rules by which networks operate	understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems	design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems	End of Topic Test and Exam questions
HT6	Programming Project 2.2.1 Programming fundamentals 2.2.3 Additional programming Techniques 2.3 Producing robust Programs 2.3.1 Defensive Design 2.3.2 Testing	Programming project with Flow and Pseudocode	Learn about fundamentals of text programming and ensuring programs are robust	use two or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions	develop and apply analytic, problem-solving, design, and computational thinking skills	Projects based Practice exam questions