

Year 10	Autumn Term 1	Autumn Term 2	Spring 1	Spring 2	Summer 1	Summer 2
Topics Studied in OCR Computer Science	<p>Systems Architecture: Architecture of the CPU</p> <p>Students learn about the components inside a PC, CPU, Von Neumann architecture and embedded systems</p> <p>Students learn how data needs to be converted into a binary format to be processed by a computer alongside the use of binary to represent data. ASCII</p>	<p>Systems Architecture Memory and Storage</p> <p>Student learn of the reasons and uses for primary and secondary storage – RAM, ROM, virtual memory and the different types of storage. Storage methods, advantages and disadvantages are also taught. Optical, magnetic and solid state</p> <p>Computer Networks, connections and protocols</p> <p>Students learn of the use of different networks and their performance alongside hardware for connection</p>	<p>Network Security</p> <p>Students learn of the threats to computer systems to identify and prevent vulnerabilities</p>	<p>Systems Software</p> <p>The purpose and functionality of operating systems:</p> <ul style="list-style-type: none"> o User interface o Memory management and multitasking o Peripheral management and drivers o User management o File management <p>The purpose and functionality of utility software</p> <ul style="list-style-type: none"> o Utility system software: o Encryption software o Defragmentation o Data compression 	<p>Ethical, legal, cultural and environmental impact</p> <p>Students learn about impacts of digital technology on wider society alongside legislation relevant to Computer Science</p>	<p>Programming in Python</p>
Skills and Key Knowledge Taught	<ul style="list-style-type: none"> - Arithmetic operators -Binary use to represent data types -Data representation within computers 	<ul style="list-style-type: none"> - Secondary storage: -Magnetic, optical and solid state RAM, ROM and virtual memory 	<p>Students learn of:</p> <ul style="list-style-type: none"> -Malware -Social engineering -Brute-force Inception and theft -SQL injection 	<p>Purposes of utility software and operating systems</p>	<p>Students learn of:</p> <ul style="list-style-type: none"> o Ethical issues o Legal issues o Cultural issues o Environmental issues 	

	<p>The units of data storage:</p> <ul style="list-style-type: none"> o Bit o Nibble (4 bits) o Byte (8 bits) o Kilobyte (1,000 bytes or 1 KB) o Megabyte (1,000 KB) o Gigabyte (1,000 MB) o Terabyte (1,000 GB) o Petabyte (1,000 TB) 	<p>Network topologies: Star and Mesh, client and servers. LANs and WANs, DNS, Hosting and cloud. Wired and wireless, encryption, IP and MAC addresses. Protocols. Layers</p>	<ul style="list-style-type: none"> -Anti-malware -Firewalls -User Access Levels -Passwords and encryption 		<ul style="list-style-type: none"> o Privacy issues o Legislation relevant to Computer Science: <ul style="list-style-type: none"> o The Data Protection Act 2018 o Computer Misuse Act 1990 o Copyright Designs and Patents Act 1988 o Software licences (i.e. open source and proprietary) 	
Links for Support/ Help at Home	<p>Use of student resources located within WHS SharePoint for students</p> <p>Complete Digital Safety and Digital Literacy courses for free online to ensure students understand E-Safety</p> <p>Use of additional homework booklets, therapy work packs and/or additional resources from the class teacher via Synergy</p> <p>Participation in enrichment opportunities and/or extra-curricular activities</p> <p>Teacher discussions following assessments and/or reports</p> <p>Facilities at home to use and practice programs on (after school clubs available to enable this)</p> <p>Youtube tutorials and guidance on using programs covered within our schemes of learning</p> <p>Researching key figures in the progression of computers to act as role models</p> <p>Accessing STEM resources (www.stem.org.uk) for free learning at home for secondary computing and progression</p> <p>Careers research: researching careers within Computing or STEM</p> <p>Attending fairs, workshops or IT events</p>					

Year 11	Autumn Term 1	Autumn Term 2	Spring 1	Spring 2	Summer 1	Summer 2
Topics Studied in OCR Computer Science	<p>Programming</p> <p>Students should be competent at designing, reading, writing and debugging programs</p>	<p>Algorithms</p> <p>Programming</p> <p>o The use of variables, constants, operators, inputs, outputs and</p>	<p>Programming</p> <p>The use of basic string manipulation</p> <p>o The use of basic file handling operations:</p>	<p>Programming</p> <p>Defensive design considerations:</p> <ul style="list-style-type: none"> o Anticipating misuse 	<p>Programming</p> <p>The purpose of testing</p> <p>o Types of testing:</p>	<p>Programming</p> <p>Characteristics and purpose of different levels of programming</p>

	<p>Algorithms</p> <ul style="list-style-type: none"> -Principles of computational thinking: <ul style="list-style-type: none"> o Abstraction o Decomposition o Algorithmic thinking Students should be aware that computers are able to store and manipulate large quantities of data Searching and sorting 	<p>assignments</p> <ul style="list-style-type: none"> “ The use of the three basic programming constructs used to control the flow of a program: <ul style="list-style-type: none"> o Sequence o Selection o Iteration (count- and condition-controlled loops) “ The common arithmetic operators “ The common Boolean operators AND, OR and NOT 	<ul style="list-style-type: none"> o Open o Read o Write o Close “ The use of records to store data “ The use of SQL to search for data “ The use of arrays (or equivalent) when solving problems, including both one-dimensional (1D) and two-dimensional arrays (2D) “ How to use sub programs (functions and procedures) to produce structured code “ Random number generation 	<ul style="list-style-type: none"> o Authentication “ Input validation “ Maintainability: <ul style="list-style-type: none"> o Use of sub programs o Naming conventions o Indentation o Commenting 	<ul style="list-style-type: none"> o Iterative <ul style="list-style-type: none"> o Final/terminal “ Identify syntax and logic errors “ Selecting and using suitable test data: <ul style="list-style-type: none"> o Normal o Boundary o Invalid/Erroneous “ Refining algorithms 	<p>language:</p> <ul style="list-style-type: none"> o High-level languages o Low-level languages “ The purpose of translators “ The characteristics of a compiler and an interpreter <p>Common tools and facilities available in an Integrated Development Environment (IDE):</p> <ul style="list-style-type: none"> o Editors o Error diagnostics o Run-time environment o Translators <p>Summer Exams Undertaken</p>
<p>Skills and Key Knowledge Taught</p>	<ul style="list-style-type: none"> -Binary and denary integers -Unsigned/signed integers -Hexadecimal notation -ASCII encoding -Algorithms 	<ul style="list-style-type: none"> “ The use of variables, constants, operators, inputs, outputs and assignments “ The use of the three basic programming constructs used to 	<p>The use of basic string manipulation</p> <ul style="list-style-type: none"> “ The use of basic file handling operations: <ul style="list-style-type: none"> o Open o Read o Write o Close 	<p>Defensive design considerations:</p> <ul style="list-style-type: none"> o Anticipating misuse o Authentication “ Input validation “ Maintainability: <ul style="list-style-type: none"> o Use of sub programs o Naming conventions 	<p>The purpose of testing</p> <ul style="list-style-type: none"> “ Types of testing: <ul style="list-style-type: none"> o Iterative o Final/terminal “ Identify syntax and logic errors 	<p>Characteristics and purpose of different levels of programming language:</p> <ul style="list-style-type: none"> o High-level languages

	<ul style="list-style-type: none"> -Differentiate errors in programs Flowcharts Standard searching algorithms: <ul style="list-style-type: none"> o Binary search o Linear search Standard sorting algorithms: <ul style="list-style-type: none"> o Bubble sort o Merge sort o Insertion sort 	<ul style="list-style-type: none"> control the flow of a program: <ul style="list-style-type: none"> o Sequence o Selection o Iteration (count- and condition-controlled loops) The common arithmetic operators The common Boolean operators AND, OR and NOT 	<ul style="list-style-type: none"> The use of records to store data The use of SQL to search for data The use of arrays (or equivalent) when solving problems, including both one-dimensional (1D) and two-dimensional arrays (2D) How to use sub programs (functions and procedures) to produce structured code Random number generation 	<ul style="list-style-type: none"> Indentation Commenting 	<ul style="list-style-type: none"> Selecting and using suitable test data: <ul style="list-style-type: none"> o Normal o Boundary o Invalid/Erroneous Refining algorithms 	<ul style="list-style-type: none"> Low-level languages <ul style="list-style-type: none"> The purpose of translators The characteristics of a compiler and an interpreter Common tools and facilities available in an Integrated Development Environment (IDE): <ul style="list-style-type: none"> Editors Error diagnostics Run-time environment Translators
Links for Support/ Help at Home	<ul style="list-style-type: none"> Use of student resources located within WHS SharePoint for students Complete Digital Safety and Digital Literacy courses for free online to ensure students understand E-Safety Use of additional homework booklets, therapy work packs and/or additional resources from the class teacher via Synergy Participation in enrichment opportunities and/or extra-curricular activities Teacher discussions following assessments and/or reports Facilities at home to use and practice programs on (after school clubs available to enable this) Youtube tutorials and guidance on using programs covered within our schemes of learning Researching key figures in the progression of computers to act as role models Accessing STEM resources (www.stem.org.uk) for free learning at home for secondary computing and progression Careers research: researching careers within Computing or STEM Attending fairs, workshops or IT events 					