Curriculum Overview – Year 11 Computer Science

Unit Title	Learning	How can parents best support?
TERM ONE HTML & GREENFOOT RECAP	• Use HTML tags to mark-up a document to specific requirements. • Design, write, test and refine Java programs within the Greenfoot environment.	Allow students to use the free software provided by Microsoft 'Notepad' to practice typing HTML code using various tags learnt both in lesson and W3 schools online. Students can download an app called SoloLearn. SoloLearn is the a popular app used to to learn C++, Java, Python, SQL, CSS, HTML, C#, and many other languages for free. At first, you must go through the 1st lesson. Once you complete this chapter, the app will display a series of question to test your knowledge. If you don't score well in this test, go through the chapter once again.
Communication	• Explain the characteristics of networks and the importance of different network types, including LAN and WAN. • The importance of common network topologies, including ring, star, bus and mesh, and their advantages and disadvantages. • The importance of wired and wireless connectivity. • Analysing the advantages and disadvantages of circuit switching and packet switching. • Evaluate the importance and the use of a range of contemporary network protocols, including Ethernet, Wi-Fi, TCP/IP, HTTP, HTTPS, FTP and email protocols. • Describe the typical contents of a TCP/IP packet. • Explain the importance of layers and the TCP/IP 5-layer model. • Describe the methods of routing traffic on a network and calculate routing costs. • Explain how Domain Name System (DNS) servers and Internet Protocol (IP) addresses work.	A range of past papers for theory paper 1 are available to download and use from https://www.wjec.co.uk/qualifications/qualification-resources.html?subject=ComputerScience&level=GCSE&pastpaper=true & https://www.wjec.co.uk/qualifications/qualification-resources.html?subject=ComputerScience&level=gcsefrom2017&pastpaper=true This can be used to support learning for all unit 1 topics.
Organisation and structure of data	• Use one-dimensional and two-dimensional arrays, files and records. • Use a variety of data types, including integer, Boolean, real, character and string.	

Programming skills Software engineering Python programming skills CA release - October	• Assign, identify and explain the use of constants and variables in algorithms and programs. • Describe the characteristics and purpose of high-level and low-level languages. • Identify and describe situations that require the use of a high-level or a low-level language. • Explain the role of Integrated Development Environment (IDE) tools in developed and debugging programs. • Use common methods of defining algorithms, including pseudocode and flowcharts. • Identify, explain and use subroutines in algorithms and programs. • Identify, explain and use sequence, selection and iteration in algorithms and programs. • Identify, explain and use counts and rogue values in algorithms and programs.	Students can use www.codecademy.com to study the Python programming project from home - students can refine their Python programming skills to strengthen knowledge from lessons and further develop skills that are vital for the practical exam paper 2. Examples of courses: https://www.codecademy.com/learn/learn-python

TERM TWO
 	
20 hour controlled assessment
	Python programming task:
	
	Given Task
	
	The task for Component 3 will be a set scenario that will be available for completion in the academic year of certification. A different scenario will be set for each academic year. All work carried out for this task should be under teacher supervision, with no access to the Internet or email. The time permitted for this task is 20 hours, and all time spent on the task should be monitored and logged by the centre as detailed in this specification.
	
	The scenario will provide candidates with a description of a client's need for a new computer based solution to a given problem. In addition to a functional computer program, candidates will need to produce a word processed report with an advisory limit of 2000 words. The areas for inclusion in the report are covered in detail in the content for this component and are summarised below:
	
	Section 1 – Scope of the problem
	
	• Description of the given scenario in terms of input, processing and output
	• Objectives, including measurable success criteria for the proposed system.
	
	Section 2 – Design
	
	Descriptions of:
	
	• Input and output facilities required to produce a user interface
	• Data structures that will be required
	• Documentation of the following routines using a standard convention (pseudo code or flowchart):
	  ○ validation routines
	  ○ data handling and processing
	  ○ authentication
	
	Section 3 – Software development
	
	• Annotated listing(s) of all programming code
	• Evidence of the user interface
	
	Section 4 – Test strategy
	
	• Description of the test strategy
	• Description of the purpose of unit, integration and functional testing
	• Test plan and test data
	
	Section 5 – Testing
	
	• Evidence of test outcomes with commentaries
		Students can download an app called SoloLearn. SoloLearn is the a popular app used to to learn C++, Java, Python, SQL, CSS, HTML, C#, and many other languages for free.
		At first, you must go through the 1st lesson. Once you complete this
		chapter, the app will display a series of question to test your knowledge. If you don't score well in this test, go through the chapter once again.

TERM THREE

Assembly language

HTML & GREENFOOT RECAP

Section 6 – Further development

- Discussion of the outcomes of the testing
- Description of the successful features of the solution and identification of areas for further development
- Suggestions for extensions to the solution

The Refinement Log

The refinement log is an integral part of the project and should be completed during each session. The purpose of the log is for candidates to demonstrate that they are working in a logical and systematic manner.

Candidates are expected to record any issues encountered and how these issues were addressed.

Assembly language	Design, write, test and refine simple assembly programs using the following mnemonics: • Input INP • Output OUT • Store STA • Load LDA • Add ADD • Subtract SUB • Branch BRA • End/Stop/Halt HLT • Data definition DAT	For example, candidates could be required to design, write, test and refine simple assembly programs that: • loads register R with the contents of address X • loads register S with the contents of address Y and adds the two numbers together.

- Use HTML tags to mark-up a document to specific requirements.
- Design, write, test and refine Java programs within the Greenfoot environment.

Students can download an app called SoloLearn. SoloLearn is the a popular app used to to learn C++, Java, Python, SQL, CSS, HTML, C#, and many other languages for free.
At first, you must go through the 1st lesson. Once you complete this chapter, the app will display a series of question to test your knowledge. If you do not score well in this test, go through the chapter once again.

Exam technique and theory revision May exam		A range of past papers for theory paper 1 are available to download and use from https://www.wjec.co.uk/qualifications/qualification-resources.html?subject=ComputerScience&level=GCSE&pastpaper=true & https://www.wjec.co.uk/qualifications/qualification-resources.html?subject=ComputerScience&level=gcsefrom2017&pastpaper=true This can be used to support learning for all unit 1 topics.

HTML & Greenfoot taught in Y9 and recapped in term 1, 2 & 3 of Y10.

Tracking assessment based on theory topics learnt so far & practical HTML & Greenfoot task