

	Autumn Term	Spring Term	Summer Term
Year 12	<p>Curriculum and Skills: Data Representation: Includes number systems, binary, hexadecimal, images, sound, encryption and decryption, data compression, before moving onto Boolean algebra. Programming: Review GCSE programming concepts in Python (functions, reading and writing to files) then move onto graphical user interfaces, visualisations, and modular programming.</p>	<p>Curriculum and Skills: Algorithms: Explores a range of algorithms, code tracing, Big O notation, abstract data types, computational thinking, abstraction, and finite state machines. Programming: Covers networking and SQL server-side scripting and Assembly language, HTML, CSS, and JavaScript client-side processing.</p>	<p>Curriculum and Skills: Hardware and Software: Includes processor architecture, communications, and networking. Computing and Society: Legislation and ethical considerations and learn to write extended prose. Programming: Mini coding projects, in preparation of the pre-release material for the Summer Term mock examination. Writing a prototype for Computing project.</p>
	<p>Assessment: Baseline programming assessment and end of unit assessment at the end of Autumn term. Regular programming and other homework assessments.</p>	<p>Assessment: Mock examinations covering Paper 1 content (on screen during lesson) early during Spring Term. Paper 2 (written) assessment on algorithms at the end of Spring term. Regular programming and other homework assessments.</p>	<p>Assessment: Mock examinations covering Paper 1 content (on screen during lesson) and Paper 2 (written). Regular programming and other homework assessments. Computing project proposal with prototype code.</p>
Year 13	<p>Curriculum and Skills: Models of Computation: Turing machines, regular expressions, and reverse polish notation. Data: Floating point numbers, databases, and big data. Programming: Covers other paradigms including object orientated programming and functional programming. Continue to work on Computing project.</p>	<p>Curriculum and Skills: Internet: Security, IP addresses, internet architecture, TCP/IP, JSON and XML, client server model. Programming: Stack frames, code tracing, complete the computing project including testing. Preparation of preliminary material for Paper 1 examination.</p>	<p>Curriculum and Skills: Theory: Revision and review of skills (e.g., code tracing, answering extended response questions) and knowledge required for both Paper 1 and Paper 2. Exam question practice. Programming: Preparation of preliminary materials for Paper 1, modifying and practicing the prerelease code to gain familiarity.</p>
	<p>Assessment: Summative Autumn term assessment. Submission of draft of analysis, design, and implementation sections of the Computing project. Regular programming and other homework assessments.</p>	<p>Assessment: Mock examinations covering Paper 1 content (on screen during lesson) and Paper 2 (written). Submission of completed Computing project. Regular programming and other homework assessments.</p>	<p>Assessment: Formal examinations, Paper 1 content (on screen during lesson) and Paper 2 (written). Regular programming and other homework assessments</p>

*At CamSF, assessment happens at many levels and is perhaps most important when teachers assess what students have learned and remembered within the classroom. Timely feedback is so important in enabling progress and knowledge retention.