

GCSE Computing Year 10 & 11	Curriculum intent: The GCSE curriculum will consolidate and build on the key themes studied at KS3. Students will study aspects of computational thinking, Problem solving, analysis, implementation and evaluation and technical vocabulary to build on their KS3 prior knowledge. Opportunities to revisit key concepts through retrieval practice have been built into the curriculum as well as spaced learning and consolidation elements built into the end of unit tests in order to maximise retention of key knowledge. Whenever possible knowledge and skills are acquired through the use of carefully planned practical activities with an emphasis on an investigative approach. Key skills will be developed with repeated practice. Students will develop understanding of the key concepts covered below and will be given the opportunity to demonstrate this in a range of different contexts.											
Topic	COMP 1				COMP 2				COMP 3			
Interleaving	Key knowledge from previously studied topics. End of unit tests to incorporate questions from earlier units.				Key knowledge from previously studied topics along with applying knowledge and understanding gained in COMP 2 and applying them in COMP 3							
Knowledge	Systems Architecture • Memory • Storage • Wired and wireless networks • Network topologies, protocols and layers • System software layers • Ethical, legal, cultural and environmental concerns • Algorithms *				Algorithms * • Programming techniques • Producing robust programs • Computational logic • Translators and facilities of languages • Data representation				Programming techniques • Analysis • Design • Development • Testing and evaluation and conclusions • Algorithms *			
Understanding	Apply knowledge in a range of contexts including creating exam questions, research projects, group work, and independent study, identifying key words and technical terminology and how to explain those using accurate technical terms. Learners will be able to demonstrate the following key elements of computational thinking: <ul style="list-style-type: none"> • Thinking abstractly – removing unnecessary detail from the problem, and Control and Data abstraction • Thinking ahead – identifying preconditions and inputs and outputs • Thinking procedurally – identifying components of problems and solutions • Thinking logically – predicting and analysing problems • Thinking concurrently – spotting and using similarities. 				Apply knowledge in a range of contexts including creating exam questions, research projects, group work, and independent study, identifying key words and technical terminology and how to explain those using accurate technical terms. Learners will be able to demonstrate the following key elements of computational thinking: <ul style="list-style-type: none"> • Thinking abstractly – removing unnecessary detail from the problem, and Control and Data abstraction • Thinking ahead – identifying preconditions and inputs and outputs • Thinking procedurally – identifying components of problems and solutions • Thinking logically – predicting and analysing problems • Thinking concurrently – spotting and using similarities. During COMP 3 students should be able to demonstrate the use of using an iterative process such as below: <ul style="list-style-type: none"> • Success criteria – what key things must the solution contain? • Planning and design – the solution is broken down and suitable designs created • Development – the iterative development with code explanations • Testing and remedial actions – a log of successful tests including correcting any errors • Evaluation – a review of the success criteria that have been met. 							
Skills	Computational Thinking	Problem Solving	Analysis, Evaluation and Implementation	Technical Vocabulary	Computational Thinking	Problem Solving	Analysis, Evaluation and Implementation	Technical Vocabulary	Computational Thinking	Problem Solving	Analysis, Evaluation and Implementation	Technical Vocabulary
Assessment	End of Unit tests Exam Paper 1				End of Unit tests Exam Paper 2				Controlled assessment			