

CURRICULUM SUMMARY - Computing

YEAR 7 - Computing

TERM 1	TERM 2	TERM 3
CONTENT <ul style="list-style-type: none"> E-Safety Binary 	CONTENT <ul style="list-style-type: none"> Spreadsheets Storage devices and computers 	CONTENT <ul style="list-style-type: none"> Control Systems Algorithmic thinking Block based programming
ASSESSMENTS <ul style="list-style-type: none"> Presentation Online testing 	ASSESSMENTS <ul style="list-style-type: none"> Online testing 	ASSESSMENTS <ul style="list-style-type: none"> Online testing
HOW PARENTS CAN SUPPORT LEARNING <ul style="list-style-type: none"> Discussion around E-safety issues Check homework is completed Support pupils with written work Rehearse presentations 	HOW PARENTS CAN SUPPORT LEARNING <ul style="list-style-type: none"> Check homework is completed Support pupils with written work 	HOW PARENTS CAN SUPPORT LEARNING <ul style="list-style-type: none"> Check homework is completed Support pupils with written work

YEAR 8 - Computing

TERM 1	TERM 2	TERM 3
CONTENT <ul style="list-style-type: none"> Programming in Python 	CONTENT <ul style="list-style-type: none"> Programming with micro:bit – text based programming 	CONTENT <ul style="list-style-type: none"> Developing web content with HTML and CSS
ASSESSMENTS <ul style="list-style-type: none"> Evaluation of programme Evidence Online testing 	ASSESSMENTS <ul style="list-style-type: none"> Evaluation of programme Evidence Online testing 	ASSESSMENTS <ul style="list-style-type: none"> Evaluation of websites Evidence of new site
HOW PARENTS CAN SUPPORT LEARNING <ul style="list-style-type: none"> Check homework is completed Support pupils with written work 	HOW PARENTS CAN SUPPORT LEARNING <ul style="list-style-type: none"> Check homework is completed Support pupils with written work 	HOW PARENTS CAN SUPPORT LEARNING <ul style="list-style-type: none"> Check homework is completed Support pupils with written work

YEAR 9

TERM 1	TERM 2	TERM 3
CONTENT <ul style="list-style-type: none"> • Computational thinking • Programming literacy • Data types • Variables and Constants • Selection 	CONTENT <ul style="list-style-type: none"> • Sequencing • Iteration • Subroutines • Error trapping 	CONTENT <ul style="list-style-type: none"> • Algorithms • Flowcharts • Pseudocode
ASSESSMENTS <ul style="list-style-type: none"> • Online tests • Functionality checks of programmes (does it work and do what it was supposed to do). • Peer assessment 	ASSESSMENTS <ul style="list-style-type: none"> • Online tests • Functionality checks of programmes (does it work and do what it was supposed to do). • Peer assessment 	ASSESSMENTS <ul style="list-style-type: none"> • Online tests • Functionality checks of programmes (does it work and do what it was supposed to do). • Peer assessment
HOW PARENTS CAN SUPPORT LEARNING <ul style="list-style-type: none"> • Check homework is completed • Support pupils with written work • Ensure pupils have access to a computer with Python installed and web access, so online resources can be used. • Give pupils encouragement to get past the inevitable struggles and failures inherent in programming. 	HOW PARENTS CAN SUPPORT LEARNING <ul style="list-style-type: none"> • Check homework is completed • Support pupils with written work • Ensure pupils have access to a computer with Python installed and web access, so online resources can be used. • Give pupils encouragement to get past the inevitable struggles and failures inherent in programming. 	HOW PARENTS CAN SUPPORT LEARNING <ul style="list-style-type: none"> • Check homework is completed • Support pupils with written work • Ensure pupils have access to a computer with Python installed and web access, so online resources can be used. • Give pupils encouragement to get past the inevitable struggles and failures inherent in programming.

YEAR 10 – Computer Science GCSE AQA 8525

TERM 1	TERM 2	TERM 3
<p>CONTENT</p> <ul style="list-style-type: none"> • Computational thinking • Algorithms • Flowcharts • Pseudocode • Programming literacy • Logic circuits • ROM and RAM • Different programming languages • Binary • Hexadecimal 	<p>CONTENT</p> <ul style="list-style-type: none"> • Data structures • Trees and Huffman coding • Understanding search and sort algorithms • Algorithm efficiency • Testing code • Computer system’s architecture • Von Neumann • Networks • Embedded systems • Memory ROM & RAM • Secondary storage • Flowcharts • Pseudocode • Programming literacy 	<p>CONTENT</p> <ul style="list-style-type: none"> • Fetch-execute cycle • Encryption • System security • Social engineering and cyber security • Ethics, the law and environment • Software and its development • Algorithms • Flowcharts • Pseudocode • Programming literacy • Databases
<p>ASSESSMENTS</p> <ul style="list-style-type: none"> • Online tests • Functionality checks of programmes (does it work and do what it was supposed to do). • Peer assessment 	<p>ASSESSMENTS</p> <ul style="list-style-type: none"> • Online tests • Functionality checks of programmes (does it work and do what it was supposed to do). • Peer assessment 	<p>ASSESSMENTS</p> <ul style="list-style-type: none"> • Online tests • Functionality checks of programmes (does it work and do what it was supposed to do). • Peer assessment
<p>HOW PARENTS CAN SUPPORT LEARNING</p> <ul style="list-style-type: none"> • Check homework is completed • Support pupils with written work • Ensure pupils have access to a computer with Python installed and web access, so online resources can be used. • Give pupils encouragement to get past the inevitable struggles and failures inherent in programming. 	<p>HOW PARENTS CAN SUPPORT LEARNING</p> <ul style="list-style-type: none"> • Check homework is completed • Support pupils with written work • Ensure pupils have access to a computer with Python installed and web access, so online resources can be used. • Give pupils encouragement to get past the inevitable struggles and failures inherent in programming. 	<p>HOW PARENTS CAN SUPPORT LEARNING</p> <ul style="list-style-type: none"> • Check homework is completed • Support pupils with written work • Ensure pupils have access to a computer with Python installed and web access, so online resources can be used. • Give pupils encouragement to get past the inevitable struggles and failures inherent in programming.

YEAR 11 – Computer Science GCSE AQA 8525

TERM 1	TERM 2	TERM 3
CONTENT <ul style="list-style-type: none"> • Algorithms • Flowcharts • Pseudocode • Programming literacy • Data structures • Trees and Huffman coding • Understanding search and sort algorithms • Algorithm efficiency • Testing code • Computer system's architecture • Von Neumann • Networks • Embedded systems • Memory ROM & RAM • Secondary storage 	CONTENT <ul style="list-style-type: none"> • Revision and preparation for final exam. • Consolidation of programming knowledge. • Fetch-execute cycle • Encryption • System security • Social engineering and cyber security • Ethics, the law and environment • Software and its development • Algorithms • Flowcharts • Pseudocode • Programming literacy • Databases 	CONTENT <ul style="list-style-type: none"> • Preparation for final exam
ASSESSMENTS <ul style="list-style-type: none"> • Online tests • Functionality checks of programmes (does it work and do what it was supposed to do). • Peer assessment 	ASSESSMENTS <ul style="list-style-type: none"> • Online tests • Functionality checks of programmes (does it work and do what it was supposed to do). • Peer assessment 	ASSESSMENTS <ul style="list-style-type: none"> • Online tests • Functionality checks of programmes (does it work and do what it was supposed to do). • Peer assessment
HOW PARENTS CAN SUPPORT LEARNING <ul style="list-style-type: none"> • Check homework is completed • Support pupils with written work • Ensure pupils have access to a computer with Python installed and web access, so online resources can be used. • Give pupils encouragement to get past the inevitable struggles and failures inherent in programming. 	HOW PARENTS CAN SUPPORT LEARNING <ul style="list-style-type: none"> • Check homework is completed • Support pupils with written work • Ensure pupils have access to a computer with Python installed and web access, so online resources can be used. • Give pupils encouragement to get past the inevitable struggles and failures inherent in programming. 	HOW PARENTS CAN SUPPORT LEARNING <ul style="list-style-type: none"> • Check homework is completed • Support pupils with written work • Ensure pupils have access to a computer with Python installed and web access, so online resources can be used. • Give pupils encouragement to get past the inevitable struggles and failures inherent in programming.

YEAR 12 – Computer Science A-Level AQA 7517

TERM 1	TERM 2	TERM 3
<p>CONTENT</p> <ul style="list-style-type: none"> • Programming literacy • Algorithms • Flowcharts • Pseudocode • Theory of computation • Testing • Finite state machines • Data representation • Data compression and encryption • Hardware and Software • Language classification • Boolean logic • Boolean algebra 	<p>CONTENT</p> <ul style="list-style-type: none"> • Programming literacy • Computer organisation and architecture • The processor instruction set • Assembly language • Input and output devices • Storage devices • Communication technology and consequences • Network topology • Wireless networking • Communication and privacy • Social, legal and cultural issues • Data structures • Queues • Lists • Stacks • Hash tables 	<p>CONTENT</p> <ul style="list-style-type: none"> • Programming literacy • Graphs • Trees • Vectors • Recursive algorithms • Big-O notation • Searching and sorting • Graph traversal algorithms • Optimisation algorithms • Limits of computation • Preparation for year 13 project
<p>ASSESSMENTS</p> <ul style="list-style-type: none"> • Online tests • Functionality checks of programmes (does it work and do what it was supposed to do). • Peer assessment 	<p>ASSESSMENTS</p> <ul style="list-style-type: none"> • Online tests • Functionality checks of programmes (does it work and do what it was supposed to do). • Peer assessment 	<p>ASSESSMENTS</p> <ul style="list-style-type: none"> • Online tests • Functionality checks of programmes (does it work and do what it was supposed to do). • Peer assessment • Practice papers
<p>HOW PARENTS CAN SUPPORT LEARNING</p> <ul style="list-style-type: none"> • Check homework is completed • Ensure pupils have access to a computer with Python installed and web access, so online resources can be used. • Give pupils encouragement to get past the inevitable struggles and failures inherent in programming. 	<p>HOW PARENTS CAN SUPPORT LEARNING</p> <ul style="list-style-type: none"> • Check homework is completed • Ensure pupils have access to a computer with Python installed and web access, so online resources can be used. • Give pupils encouragement to get past the inevitable struggles and failures inherent in programming. 	<p>HOW PARENTS CAN SUPPORT LEARNING</p> <ul style="list-style-type: none"> • Check homework is completed • Ensure pupils have access to a computer with Python installed and web access, so online resources can be used. • Give pupils encouragement to get past the inevitable struggles and failures inherent in programming.

YEAR 13 – Computer Science A-Level AQA 7517

TERM 1	TERM 2	TERM 3
<p>CONTENT</p> <ul style="list-style-type: none"> • Programming literacy • Programming project and submission of NEA • Regular languages • Mealy machines • Sets • Regular expressions • Turing machine • Backus-Naur form • Reverse Polish notation • Structure of the Internet • Packet switching and routers • Internet security • TCP IP Standard application layer protocols • IP addresses • Client server model 	<p>CONTENT</p> <ul style="list-style-type: none"> • Programming literacy • Databases and software development • Entity relationship modelling • Relational databases and normalisation • Introduction to SQL • Defining and updating tables using SQL • Systematic approach to problem solving • Object oriented and functional programming • OOP Design principles • Big data 	<p>CONTENT</p> <ul style="list-style-type: none"> • Programming literacy • Preparation for exams
<p>ASSESSMENTS</p> <ul style="list-style-type: none"> • Online tests • Functionality checks of programmes (does it work and do what it was supposed to do). • Peer assessment • Practice papers 	<p>ASSESSMENTS</p> <ul style="list-style-type: none"> • Online tests • Functionality checks of programmes (does it work and do what it was supposed to do). • Peer assessment • Practice papers 	<p>ASSESSMENTS</p> <ul style="list-style-type: none"> • Online tests • Functionality checks of programmes (does it work and do what it was supposed to do). • Peer assessment • Practice papers
<p>HOW PARENTS CAN SUPPORT LEARNING</p> <ul style="list-style-type: none"> • Check homework is completed • Ensure pupils have access to a computer with Python installed and web access, so online resources can be used. • Give pupils encouragement to get past the inevitable struggles and failures inherent in programming. 	<p>HOW PARENTS CAN SUPPORT LEARNING</p> <ul style="list-style-type: none"> • Check homework is completed • Ensure pupils have access to a computer with Python installed and web access, so online resources can be used. • Give pupils encouragement to get past the inevitable struggles and failures inherent in programming. 	<p>HOW PARENTS CAN SUPPORT LEARNING</p> <ul style="list-style-type: none"> • Check homework is completed • Ensure pupils have access to a computer with Python installed and web access, so online resources can be used. • Give pupils encouragement to get past the inevitable struggles and failures inherent in programming.