Alcester Academy Curriculum Planning: Key Stage 4 Computer Science – Year 9 2023-24

Pupils in Computer Science in year 9 have two (1 hour) lessons each week.

Department: *ICT & Computing*							
Term	Topic/Subject	Assessment Objectives	Knowledge Acquisition	Skill building & intent	Wider reading opportunities, including numeracy & SMSC.	Assessment Task	SEND & PP
Aut 1	**Course Introduction**	• Pupils are introduced to the OCR computer science GCSE course, and Google Classroom delivery model.	Learn how to use the online Google Classroom resources.	Learn how to access all resources through Classroom.	Resources are introduced, and demonstrated to pupils visually.	n/a	Additional support to be provided where required for both SEND & PP pupils to help access resources.
	Transition Workbook	• Able to identify emerging technologies • Able to use cornell note taking method for theory topics • Able to research key terminology • Able to recap core principles of Python coding • Able to identify key equipment in home networks. • Able to understand how to use logic gates & truth tables. • Able to convert denary/binary numbers, and add binary numbers together. • Able to understand different input/output/storage devices. • Able to use decomposition to break down larger programming problems.	Pupils are introduced to the key elements of the computer science course through a series of short, focussed research & practical activities.	Pupils will learn key elements of the computer science course, through completing an introduction transition workbook during Autumn 1. Throughout this workbook, various elements are introduced, relevant to specification theory requirements, and practical coding tasks using Python.	Resources are introduced, and demonstrated to pupils visually initially, then pupils begin to develop individual skills supported by framework of tasks delivered over Google Classroom. Keywords: networks, binary, input, output, storage, decomposition.	Assessed through continual assessment techniques during lessons	Additional support to be provided where required for both SEND & PP pupils to help access resources.
Aut 2	**1.1 Systems Architecture**	• Understand what the CPU of a computer does.	Pupils learn key concepts of systems architecture, including role of cpu, Von	Pupils learn how to use the Google Classroom teaching resources. Theory topics are introduced during each lesson	A combination of visual & google classroom based resources are used, and all are	Smart Revise assessment evidence is used to track	Additional support to be provided where required for both SEND & PP pupils to

		• Know what the registers in a CPU are. • Know the stages of the fetch, execute cycle. • Know what the registers in a CPU are. • Know the stages of the fetch, execute cycle. • Describe the Von Neumann architecture. • Know the components of the Von Neumann architecture. • Know what factors affect the speed of a CPU. • Know the stages of the fetch, execute cycle. • Know what is meant by the term: 'embedded system'. • Know several examples of embedded systems.	Neumann architecture, and fetch, process, execute cycle.	through video based resources, supported by google slide presentations. Pupils use flip learning & cornell note taking methods to complete a series of online workbook based tasks. All resources are accessible at home, and are supported through 'CraignDave' Smart Revise platform of self quizzing.	delivered to pupils in lesson time. Pupils can access all resources, and SmartRevise revision platform is used throughout. Key questions: • What is the "architecture" of a CPU? • What factors affect the CPU performance? • What are embedded systems, and what are their characteristics?	progress from start to end of topic. End of topic written test is also completed.	help access resources.
	Python programming introduction	• Understand how to output text strings • Understand how to input strings & numbers • Understand how to use different string manipulation functions.	Pupils recap core python coding skills learn to date.	Pupils build on core python programming & coding skills already covered in year 8 IT, & transition workbook activities.	A combination of tutorial video, and google slides based resources are used, together with demonstrations outlining core coding concepts.	Pupils complete a range of different coding challenges, covering core concepts.	Additional support to be provided where required for both SEND & PP pupils to help access resources.
Spr 1	**1.2 Memory & Storage**	• Understand the need for primary storage • Know the difference between RAM and ROM.	Pupils learn key concepts of memory & storage, including purpose of RAM & ROM, virtual memory, secondary memory, and characteristics & uses of	Pupils to continue to use the Google Classroom teaching resources for this topic. Theory topics are introduced during each lesson through video based resources, supported by google slide presentations. Pupils use flip	A combination of visual & google classroom based resources are used, and all are delivered to pupils in lesson time. Pupils can access all resources, and SmartRevise	Smart Revise assessment evidence is used to track progress from start to end of topic. End of topic written test is also completed.	Additional support to be provided where required for both SEND & PP pupils to help access resources.

		• Know the purpose of ROM in a computer system. • Know the purpose of RAM in a computer system. • Understand the need for virtual memory. • Understand the need for secondary storage. • Know the common types of storage. • Know the characteristics of storage devices. • Understand the suitability of storage devices for given applications. • Understand the advantages and disadvantages of devices based on their characteristics.	different storage devices.	learning & cornell note taking methods to complete a series of online workbook based tasks. All resources are accessible at home, and are supported through 'CraignDave' Smart Revise platform of self quizzing.	revision platform is used throughout. Key questions: • Why do computers have primary storage? • How does virtual memory work? • Why do computers have secondary storage? • What are the differences between secondary storage devices?		
	Python programming introduction	• Understand how to use selection statements. • Understand how to use arithmetic operators • Understand how to use counter controlled iterations • Understand how to use condition controlled iterations	Pupils continue to build on core python coding skills.	Pupils build on core python programming & coding skills already covered in year 8 IT, & transition workbook activities.	A combination of tutorial video, and google slides based resources are used, together with demonstrations outlining core coding concepts.	Pupils complete a range of different coding challenges, covering core concepts.	Additional support to be provided where required for both SEND & PP pupils to help access resources.
Spr 2	**1.2 Memory & Storage cont.**	• Understand what is meant by the terms bit, nibble, byte, kilobyte, megabyte, gigabyte, terabyte and petabyte. • Know how to represent the capacity of data storage using	Pupils learn key concepts of memory & storage, including units of data, storage capacity, binary/denary/hexadecimal, addition & conversion, character sets, how data is used to represent images &	Pupils to continue to use the Google Classroom teaching resources for this topic. Theory topics are introduced during each lesson through video based resources, supported by google slide presentations. Pupils use flip learning & cornell note taking methods to complete a series	A combination of visual & google classroom based resources are used, and all are delivered to pupils in lesson time. Pupils can access all resources, and SmartRevise revision platform is used throughout.	Smart Revise assessment evidence is used to track progress from start to end of topic. End of topic written test is also completed.	Additional support to be provided where required for both SEND & PP pupils to help access resources.

		these units, and be able to convert between them. ● Understand that data needs to be converted into a binary format to be processed by a computer. ● Know what data capacity means. ● Understand how to calculate data capacity requirements. ● Know how to convert positive denary whole numbers (0–255) into 8 bit binary numbers and vice versa. ● Know how to add two 8 bit binary integers. ● Know how to perform a left and right binary shift. ● Understand what binary shift achieves. ● Know how to convert positive denary whole numbers (0–255) into 2 digit hexadecimal numbers and vice versa. ● Know how to convert from binary to hexadecimal equivalents and vice versa. ● Understand that all data must be represented in binary numbers, including text. ● Know what is meant by the term “character set”. ● Understand the relationship between the number of bits in the character set and	sound, and compression methods.	of online workbook based tasks. All resources are accessible at home, and are supported through ‘CraignDave’ Smart Revise platform of self quizzing.	Key questions: ● Why is data stored in binary? ● How do you calculate data capacity? ● What can happen to the most significant bit when you add two binary numbers together? ● What actions can an ALU perform? ● What is the relationship between denary, binary and hexadecimal? ● How do computers store and use numbers? ● How does a computer store characters and what are the implications for the number of bits used? ● How does a computer store graphics and what are the implications for image size and resolution? ● How do computers store sound and what are the implications for sample rate, duration and bit depth? ● Where is compression used and why? ● What are the effects on a file for each type of compression?		

		the number of characters that can be represented. ● Know two common character sets: ASCII and Unicode. ● Understand how an image is represented as a series of pixels represented in binary. ● Know what is meant by the term 'metadata' and be able to give examples. ● Understand the effect of colour depth and resolution on the size of an image file. ● Know why data is often compressed for transfer and storage. ● Understand the difference between lossy and lossless compression. ● Know why some types of data are only suitable for one type of compression. ● Know why data is often compressed for transfer and storage. ● Understand the difference between lossy and lossless compression. ● Know why some types of data are only suitable for one type of compression.					
	Python programming introduction	● Understand how ararys & lists are used in programming. ● Understand how subroutines, procedures &	Pupils continue to build on core python coding skills.	Pupils build on core python programming & coding skills already covered in year 8 IT, & transition workbook activities.	A combination of tutorial video, and google slides based resources are used, together with demonstrations outlining core coding concepts.	Pupils complete a range of different coding challenges, covering core concepts.	Additional support to be provided where required for both SEND & PP pupils to help access resources

Alcester Academy Curriculum Planning: Key Stage 4 Computer Science – Year 9 2023-24

		functions are used in programming.					
Sum 1	**(1.3) Networks** **Python programmin**	• Know what is meant by 'stand-alone' computers.Know the different types of networks: LAN and WAN. • Understand the advantages of networking. • Understand the implications of networking. • Understand how to program. • Know what factors affect the performance of networks. • Know what a client-server model is. • Know what a peer-to-peer model is. • Understand the different roles computers have in each model. • Know the hardware needed to connect a LAN. • Understand the purpose of each piece of hardware. • Understand what The Internet actually is. • Understand the term DNS (Domain Name Server). • Understand what is meant by the term, 'hosting'. • Understand what is meant by the term, 'cloud'. • Understand what is meant by the terms 'web server' and 'client'. • Understand how to adopt the Try, Investigate, Make,	Pupils learn key concepts of networks, including LAN/WAN,advantages, performance factors, roles/client server/peer-peer, hardware required, DNS, & hosting. Pupils to continue to build on core	Pupils to continue to use the Google Classroom teaching resources for this topic. Theory topics are introduced during each lesson through video based resources, supported by google slide presentations. Pupils use flip learning & cornell note taking methods to complete a series of online workbook based tasks. All resources are accessible at home, and are supported through 'CraignDave' Smart Revise platform of self quizzing. Pupils use TIME programming activities to continue to build	A combination of visual & google classroom based resources are used, and all are delivered to pupils in lesson time. Pupils can access all resources, and SmartRevise revision platform is used throughout. Key questions: • What are the characteristics of LANs and WANs? • What can affect the performance of a network? • What are the differences between peer-to-peer and client-server networks? • How do you set up a LAN? A combination of tutorial video, and	Smart Revise assessment evidence is used to track progress from start to end of topic. End of topic written test is also completed. Pupils complete a range of different	Additional support to be provided where required for both END & PP pupils to help access resources.

	g TIME approach introduction	Evaluate (TIME) approach to developing Python programming skills. ● Understand how to write structured programs, including functions, sub routines, and parameter passing.	programming skills, and start to develop the TIME approach to coding. Try, Investigate, Make, and Evaluate activities through structured workbook tasks.	on core skills & knowledge already delivered. Pupils have 11 workbooks to progress through, beginning with introduction to TIME, then workbook 1 - writing structured programs.	google slides based resources are used, together with demonstrations outlining core coding concepts.	coding challenges, covering core concepts.Pupils to progress through Make activities in workbooks, achieving point scored solutions based on complexity of each challenge completed.	Additional support to be provided where required for both SEND & PP pupils to help access resources
Sum 2	**(1.3) Networks - cont.**	● Know what a star network is. ● Know what a mesh network is. ● Understand The Internet is an example of a partial mesh network. ● Know the advantages and disadvantages of star and mesh networks. ● Understand that Ethernet is a wired method of connection. ● Understand that Wi-Fi and Bluetooth and wireless method of connection. ● Understand the benefits and drawbacks of wired versus wireless connections. ● Be able to commend a connection type for a given scenario. ● Know the basics of how cryptography can work with a simple key. ● Know how wireless devices authenticate with each other before communicating data. ● Understand the difference between a private key and public keys. ● Understand why private (master) keys are never shared.	Pupils continue to learn key concepts of networks, including network topologies, methods of connection, basics of cryptography, MAC/IP addressing, differences between IPv4 & IPv6, and common protocols & layers used.	Pupils to continue to use the Google Classroom teaching resources for this topic. Theory topics are introduced during each lesson through video based resources, supported by google slide presentations. Pupils use flip learning & cornell note taking methods to complete a series of online workbook based tasks. All resources are accessible at home, and are supported through 'CraignDave' Smart Revise platform of self quizzing.	A combination of visual & google classroom based resources are used, and all are delivered to pupils in lesson time. Pupils can access all resources, and SmartRevise revision platform is used throughout. Key Questions: ● How does The Internet work? ● Why is a mesh network better than a star network? ● Which is better, a wired or wireless network? ● What is the purpose of encryption? ● What are the differences between three types of network device addresses? ● What are standards and protocols? ● What are the benefits of layering protocols?	Smart Revise assessment evidence is used to track progress from start to end of topic. End of topic written test is also completed.	Additional support to be provided where required for both SEND & PP pupils to help access resources.

		• Understand the uses of MAC and IP addressing. • Understand the difference between IPv4 and IPv6. • Understand the need for IPv6. • Understand the need for standards in computing. • Understand the 7 common protocols and what they are used for. • Know why protocols are layered.					
	Python programming TIME approach introduction	• Understand how to adopt the Try, Investigate, Make, Evaluate (TIME) approach to developing Python programming skills. • Understand how to use selection. • Understand how to use number data types.	Pupils to continue to build on core programming skills, and start to develop the TIME approach to coding. Try, Investigate, Make, and Evaluate activities through structured workbook tasks.	Pupils use TIME programming activities to continue to build on core skills & knowledge already delivered. Pupils have 11 workbooks to progress through. Workbook 2 - how to use selection, workbook 3 - how to use number data types.	A combination of tutorial video, and google slides based resources are used, together with demonstrations outlining core coding concepts.	Pupils complete a range of different coding challenges, covering core concepts.Pupils to progress through Make activities in workbooks, achieving point scored solutions based on complexity of each challenge completed.	Additional support to be provided where required for both SEND & PP pupils to help access resources