--

Ian Ramsey CE Academy: Computer Studies Curriculum Progression Model

CURRICULUM OVERVIEW

Curriculum What we study. Why study it. Why study it now.

What we need pupils to have learnt at each point/end of each year and the logical connection and the sequential learning between what is studied in the different terms and between years. This is what is to be covered and when, effectively creating the idea that the intent is the curriculum. The intent is everything up to the point of teaching. The purpose of our curriculum and the knowledge we want our pupils to go away with in their working memory.

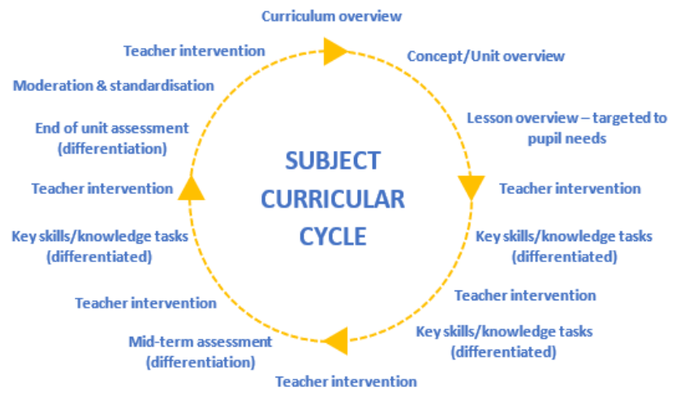
We aim to provide a high-quality computing education which equips pupils to use computational thinking and creativity to understand and change the world. The computing curriculum helps pupils to understand and apply fundamental principles and concepts of computer science. It allows pupils to analyse problems in computational terms and practise writing programs in order to solve such problems. We aim to ensure that pupils are digitally literate, able to develop their ideas and express themselves whilst becoming responsible, competent and creative users of information and communication technology.

How we teach the curriculum

How we make learning memorable and how we support our pupils to remember it. How we use rote, retrieval, interleaving, metacognition etc. in our teaching; why we teach in the way we are and justify decisions around how and why it's being taught this way.

What we expect from the curriculum

How we make it challenging and ambitious for our pupils. How we assess learning, knowledge and understanding; what have they learnt and how well have they learnt it? Consider what assessments we use, when we use them and how and why we assess this way?

	Setting	Designing	Planning	Delivering
YEAR 7	**What:** Office Skills	**What:** How computers work	**What:** Programming with Scratch	The KS3 Computer Studies curriculum has been designed around the three strands of the National Curriculum: computer science, information technology and digital literacy. There is considerable overlap between the three strands and each project will contain aspects of at least 2, if not all 3, of the strands. Digital literacy runs throughout all of the work we do with pupils, giving them the ability to effectively, responsibly, safely and critically navigate, evaluate and create digital artefacts using a range of digital technologies. Pupils are encouraged to develop computational thinking and problem-solving skills with programming concepts being built upon and reinforced through programming projects in each year. These progress from block editors to text editors whilst the actual concepts underlying the programs remain constant.	Assessment for Learning is used in all lessons to provide evidence for use by pupils and teachers to decide where pupils are in their learning, where they need to go and how best to get there. **Formative Assessment** This is used to provide information about what pupils know, understand and can do. This is used by both the teacher and the pupil to determine where pupils are in their learning and how to continue to develop their knowledge and skills within the subject. This will include: • Questioning • Effective teacher feedback (written and verbal) • Peer feedback • Pupil self-assessment **Summative Assessment** This is also used at key points in each year to evaluation pupils' achievement. They allow a holistic view of pupils' progress and support the identification of areas requiring additional focus to improve learning overall.
	What: Pupils develop basic skills for using programs in the Microsoft Office suite. These are skills that are important to support effective teaching and learning across the secondary curriculum and beyond into further study and the world of work. Microsoft Office continues to be widely used by organisations across the world. Pupils are also taught how to use the academy email system for effective communication and to work online via Google Classroom and Google Drive.	**What:** Pupils study the basic theory of computer systems. They learn about the basic hardware components that make up computer systems. They build on this and look at the software that uses this hardware to run. They progress to look at how different computer systems communicate.	**What:** Scratch is an accessible integrated development environment allowing pupils to program with a visual language. Scratch can be used online or downloaded free of charge enabling pupils to continue their learning outside of the classroom. Pupils are encouraged to employ computational thinking strategies to solve problems whilst creating simple games. They are introduced to basic programming concepts.	
	Why now: The skills gained enable pupils to effectively use the IT facilities in the academy across a range of subjects. Many pupils do not have the opportunity to learn this at KS2.	**Why now:** Computing theory is not usually taught at KS2 making it necessary to begin at the most basic level. This can be built on in subsequent years.	**Why now:** This unit ensures that all pupils have experience with blocks-based programming as coverage at KS2 is variable.	
YEAR 8	**What:** Programming with EduBlocks	**What:** Understanding digital communication	**What:** Video and Sound	
	What: EduBlocks is the perfect transition between programming with Scratch and programming with a text-based language, such as Python, which is used for computer science GCSE. It combines blocks with Python text to help pupils learn. EduBlocks is free to use online, enabling pupils to continue their learning outside of the classroom.	**What:** Pupils learn how instructions are stored and executed in a computer system. This includes looking at how text, sound and images are represented and manipulated digitally in the form of binary digits. The content is related to how text, images and sound are sent over apps on smartphones and other devices. This makes the context extremely relevant for the age group.	**What:** Pupils undertake a creative project which allows them to create, reuse, revise and repurpose digital artefacts. They use Serif MoviePlus software to bring together digital assets to form a movie trailer. Pupils learn to work to a brief to create work for a given target audience. They must pay attention to the trustworthiness, design and usability of the digital artefacts that they use.	
	Why now: EduBlocks bridges the gap between programming with a block editor and text-based languages.	**Why now:** This follows on from how computers work in year 7.	**Why now:** Pupils work practically with sound and images (still and moving) which follows on from the theory learnt in the previous unit.	
YEAR 9	**What:** Photo editing	**What:** Programming with GameMaker	**What:** Programming with Small Basic	
	What: Pupils undertake a creative project in which they learn to use Adobe Photoshop to edit photographs and other images. They consider the appropriateness, trustworthiness and usability of assets that they utilise.	**What:** Pupils create a maze-style game using GameMaker software. This provides further programming experience using a different platform. There are virtually endless opportunities for pupils to extend their learning and their games. Pupils are expected to reuse, revise and repurpose digital artefacts to suit the needs of a specific audience.	**What:** Small Basic is the perfect text-based language to start programming with. Pupils will reinforce the knowledge of variables and inputs whilst developing their knowledge of different types of loops and lists by learning to program searching and sorting algorithms.	
	Why now: Ahead of options, Photoshop experience enables pupils to make informed choices regarding photography GCSE.	**Why now:** GameMaker software allows pupils to consolidate their programming experience. It enables pupils to make informed choices regarding computer science GCSE.	**Why now:** The programming skills that will take place in GCSE computer science are very similar to the ones learnt in this unit. It allows a strong foundation of knowledge and skills to be built, preparing pupils for the more complex skills learnt in GCSE computer science.	

CURRICULUM OVERVIEW

Curriculum What we study. Why study it. Why study it now.

What we need pupils to have learnt at each point/end of each year and the logical connection and the sequential learning between what is studied in the different terms and between years. This is what is to be covered and when, effectively creating the idea that the intent is the curriculum. The intent is everything up to the point of teaching. The purpose of our curriculum and the knowledge we want our pupils to go away with in their working memory.

We aim to provide a high-quality computing education which equips pupils to use computational thinking and creativity to understand and change the world. The computing curriculum helps pupils to understand and apply fundamental principles and concepts of computer science. It allows pupils to analyse problems in computational terms and practise writing programs in order to solve such problems. We aim to ensure that pupils are digitally literate, able to develop their ideas and express themselves whilst becoming responsible, competent and creative users of information and communication technology.

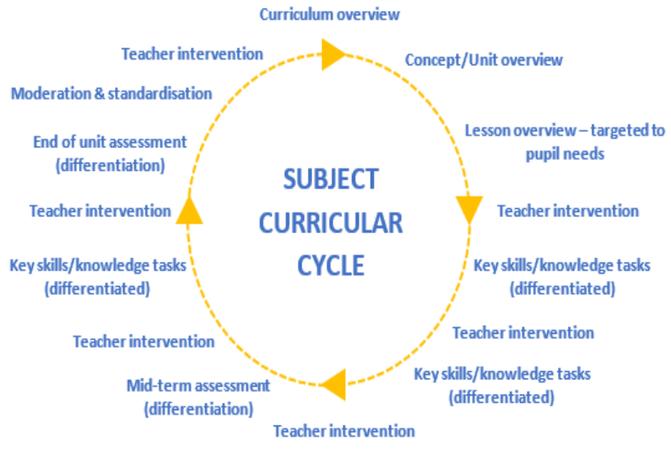
How we teach the curriculum

How we make learning memorable and how we support our pupils to remember it. How we use rote, retrieval, interleaving, metacognition etc. in our teaching; why we teach in the way we are and justify decisions around how and why it's being taught this way.

What we expect from the curriculum

How we make it challenging and ambitious for our pupils. How we assess learning, knowledge and understanding; what have they learnt and how well have they learnt it? Consider what assessments we use, when we use them and how and why we assess this way?

Year 10 and 11 Computer Science

Setting	Designing	Planning			Delivering
What: Programming	**What:** System architecture	**What:** Memory and storage	**What:** Networks	**What:** Network security	**What:** System software	
What: Pupils need to be able to become proficient in programming before their final assessment. This will be taught as a thin unit throughout the course. As they progress, they will link the practical skills they have learnt with the programming theory that they will be assessed on allowing pupils to show a deep level of understanding.	**What:** Pupils need to understand how computers work. This unit of work allows them to understand the link between hardware and software and the user's interaction with it (the input) gives an output.	**What:** Pupils need to learn how data is stored and accessed on different devices. They will learn this in detail looking at binary numbers how they are used and made to work across a variety of systems.	**What:** Pupil must now understand how devices communicate with each other to do the things we take for granted in our daily lives. They will learn about the different ways devices can be connected (and how). They will then look at the rules devices follow to make sure the data arrives to its destination in the correct format.	**What:** Pupils must understand the threats that devices connected to networks face constantly. Learning what the threat is and how to prevent the threat is key to them protecting their data both now and in the future.	**What:** Pupils need to understand how to keep the systems that they are using working efficiently to maximise performance and security. They need to learn which parts the computer will try and do automatically and the bits that the user needs to do.	The learning that take place builds on each of the previous topics by drawing clear and obvious links. This is why recall is a big part of the teaching that takes place. The units will contain opportunities for pupils to practice exam technique and apply the learning that has taken place allowing for feedback on knowledge and skills that is vital to success. Each unit is also assessed in a summative way and the same quality of feedback is given here. Programming is a key part of the subject which is why it is visited almost every week. As these practical skills are built, we begin to apply these to different assessment methods to provide pupils with the skills they need to succeed. The smartrevise.online website is used to promote the recall of facts that pupils need to make progress.	Assessment for Learning is used in all lessons to provide evidence for use by pupils and teachers to decide where pupils are in their learning, where they need to go and how best to get there. **Formative Assessment** This is used to provide information about what pupils know, understand, and can do. This is used by both the teacher and the pupil to determine where pupils are in their learning and how to continue to develop their knowledge and skills within the subject. This will include: • Questioning • Effective teacher feedback (written and verbal) • Peer feedback • Pupil self-assessment **Summative Assessment** This is also used at key points in each year to evaluation pupils' achievement. They allow a holistic view of pupils' progress and support the identification of areas requiring additional focus to improve learning overall.
Why now: It needs to start now so that pupils have as long as possible to retrieve and embed the knowledge.	**Why now:** Pupils must understand the basics of the input, process output model before they can move onto the more complex knowledge this builds upon.	**Why now:** This will allow them to make the link between the inputs and outputs of devices and lead them onto understanding how different devices communicate with each other.	**Why now:** Building on previous learning and the complexities of how data is stored, understanding how to transfer data is the next critical point of learning.	**Why now:** As more and more data is transmitted, it is imperative that this data is kept secure when it is sent and when it is stored at its destination (and its origin).	**Why now:** Pupils have a firm grasp of the fundamentals of computer systems and their communication methods. They need to be able to fine tune this knowledge to different devices and systems.	
What: Wider impacts of technology	**What:** Algorithmic thinking	**What:** Programming theory	**What:** Logic	**What:** Revision		
What: Pupils will need to understand how the systems and devices they have learnt about impact the world that they live in on several different levels. They also need to know the relation these devices, and their uses, have on legal issues within our society.	**What:** Pupils need to be able to break problems down into their subsequent parts so that they know which skills to apply to them to produce a solution. They will need to be able to represent possible solutions in several different ways and link this to different programming concepts that they have studied.	**What:** Pupils will develop their practical and theoretical programming skills by looking at some new skills and being asked to explain existing skills that they know. This will involve the manipulation of data in programs to meet the needs of a problem and make a working solution. Pupils also need to know how to explain how the code they write gets turned into a functional program.	**What:** Logic is a critical part of the decision making within the programs that pupils will write and the decisions that are made at a low-level by the systems that they use. They should be able to explain and represent these decisions in a variety of situations.	**What:** Pupils have covered the theory and skills that they need to be successful within he subject. Now is the opportunity to revisit topics with a focus on both knowledge, but also exam technique to ensure they are fully prepared for the assessment style of the course.		
Why now: Pupils have a firm grasp of the working of devices. They now need to learn about their impact so that they can help to protect their use for the greater good.	**Why now:** Pupils will have a good knowledge of programming by now and so they must begin to dissect the different applications of these skills to solve problems.	**Why now:** Pupils now can devise different solutions, they need to increase their technical understanding of how to achieve these possible solutions.	**Why now:** This is the final piece of the jigsaw in how computer systems work, and it draws on knowledge taught in the previous units of work.	**Why now:** These will be the last lessons before their final external assessment and pupils must be prepared for this.		