

Computer Science Curriculum Overview

Our department's vision

At Reach Academy Hanworth park, our vision for computing is to support children to become technologically literate in an ever-changing, digital world.

What knowledge have we selected in our curriculum and why?

Being technologically literate means pupils will be able to:

1. **Be safe**

Pupils know the opportunities that technology possesses, but also the dangers. They know how to navigate these dangers and who they can go to (both in and out of school) if they encounter a problem.

2. **Think computationally**

Pupils can solve problems, evaluate solutions and design algorithms.

3. **Create, store and retrieve information**

Pupils can find, use and express their ideas through a range of technological platforms.

Our computing curriculum has been designed with our whole-school vision in mind; to give students the skills, attributes and academic qualifications needed to go on to live a life of choice and opportunity. In selecting and sequencing knowledge for our curriculum, we are guided by the national curriculum, Kapow (in Primary) and Teach Computing (in Secondary). Acting as a conduit between our Primary and Secondary curriculum, are our whole-school curriculum design principles:

1. Knowledge-rich: Having adequate knowledge within and around computing is fundamental in preparing pupils for a world which is increasingly technologically advanced. The knowledge we teach in computing is split into three main areas: computer science, digital literacy and information and technology. The core of computing is computer science, in which pupils are taught the principles of information and computation, how digital systems work, and how to put this knowledge to use through programming. Building on this knowledge and understanding, pupils are equipped to use information technology to create programs, systems and a range of digital content. This knowledge ensures that pupils become digitally literate at a level suitable for the future workplace so they can become active participants in a digital world.

2. Backwards planned: At Reach Academy, computing is taught weekly across three half-terms in Primary and weekly for the entire year in Phase 3, with Phase 4 receiving a curriculum integrated with Team Reach and full Computing drop down days to ensure we satisfy and exceed the national curriculum. As the children progress through the school, knowledge in three main areas of computing is deepened, while key concepts are revisited. For example, students study

programming in Scratch Junior in Year 2, begin programming in Scratch in Y3, and continue to use Scratch to deepen their knowledge and understanding of programming before progressing to Python in Y8.

3. Carefully resourced: We utilise a carefully sequenced computing curriculum, guided by the national curriculum, Kapow (in Primary) and Teach Computing (in Secondary). Each lesson begins with retrieval practice of key prior knowledge and then the teaching of key vocabulary. This vocabulary might be recalled from previous learning, which is fundamental for the new learning about to take place, or it could be completely new. Dual-coding is fundamental to the explicit teaching of this vocabulary, so that every child in the classroom is successful. For some units, computing booklets have been created. These booklets ensure the children are being taught vocabulary and knowledge necessary for them to succeed throughout their Secondary computing journey. Each lesson requires children to complete a task, which demonstrates their understanding of a new skill, or a key piece of new knowledge. This might be showcased independently, or collaboratively.

4. Aspirational, inclusive and diverse: As most students will be required to interact with computers daily in their further studies or employment, and some students will pursue a career in computing, our curriculum is designed so that these students have all the skills and knowledge required to confidently seize upon these opportunities. In Key Stage 4 and 5, students are also given extra-curricular opportunities to further their skills and knowledge in computing through physical computing project work, such as robotics and sensor systems, using Arduinos and Raspberry Pis, introducing them to a new programming language, C++, and allowing them to further develop their usage of the Python programming language.

5. Rigorously assessed: Assessment is carried out formatively in each unit through questioning, observation and discussion. At the end of each lesson there is an 'exit ticket', which pupils answer independently. Through the use of rigorous monitoring trackers, teachers are able to record which children have a secure understanding of the content in that lesson, and which are going to need further support. At the end of each unit, pupils sit a knowledge quiz, enabling the teacher to assess what key knowledge individual pupils have consolidated, as well as what might need to be taught again in the next unit to ensure pupils have a secure understanding to build on their prior knowledge in future units. For programming modules in Secondary, students are also assessed with practical programming tasks, to demonstrate their skills and their ability to apply their acquired knowledge.

6. Regularly evaluated and reflected upon: At Reach Academy, we continually adapt and improve our resources for the teaching of computing each year. All plans and accompanying materials are saved on a central drive system so that teachers can edit and improve materials from the previous years and respond to the needs of the cohort they are teaching. The designers of the computing curriculum from Primary and Secondary meet regularly to share reflections on current units being taught and reflect on what more can be done to ensure that children are given the necessary prerequisite knowledge in Primary to access more complex computing knowledge and skills in Secondary.

In their lessons, we want pupils to be inspired and recognise the vast array of opportunities there is within the subject. However, we also want our pupils to be acutely aware of the dangers it can present. This is why in Primary, online safety topics are revisited across all three half-terms and

within Team Reach lessons. In Secondary, online safety is taught as stand alone units, but we are mindful of any new online trends and often have discrete, responsive sessions with children should any issues arise.

5. Rigorously assessed:

We systematically assess pupils in lessons by teaching responsively through a range of AFL strategies; through low-stakes assessments; and in formal assessments at the end of every half-term after which we deliver whole-class feedback.

Our 'Do Nows' are a means of retrieval practice used to recap previously taught topics and are planned meticulously, considering spacing and interleaving of practice. Our low-stakes assessments are a key way of seeking and, via whole class feedback (WCF), closing gaps in knowledge.

Formal assessments provide robust student data that can be used formatively to re-teach content identified in QLAs through WCF and summatively to consider the snapshot of attainment and progress at that time.

6. Regularly evaluated and reflected upon.

We use structures such as subject management, department meeting time and pupil progress meetings after formal assessments to frequently reflect on our curriculum. We consider its design, its resourcing, its implementation and its impact; making changes immediately or logging them for the next academic year. Constant reflection on our implementation takes place through our instructional coaching system, in which every teacher receives a weekly action step to improve their implementation of the curriculum.

Curriculum Maps

Primary Curriculum Map						
Year	Autumn 1	Autumn 2	Spring 1	Spring 2	Summer 1	Summer 2
1	Computing Systems and Networks: Improving mouse skills Programming: Bee-Bots		Programming: Algorithms unplugged		Creating Media: Digital media	
2	Computing Systems and Networks: What is a computer?		Programming 1: Algorithms and Debugging		Programming 2: Scratch Jr	
3	Computing Systems and		Creating Media:		Programming:	

	Networks: Networks and the internet. A journey inside a computer		Video trailers		Scratch	
4	Computing Systems and Networks: Collaborative Learning		Programming 1: Further coding with Scratch Data Handling: Investigating weather		Programming 2: Computational Thinking	
5	Computing Systems and Networks: Search Engines		Data Handling: Mars Rover 1 Creating Media: Stop motion animation		Programming: Music on Scratch	
6	Computing Systems and Networks: Bletchley Park	Programming Essentials: Part 1 (Teach Computing)	Data Handling 1: Big Data 1	Creating Media: History of Computers	Data Handling 2: Big Data 2	Skills Showcase: Inventing a product

Secondary Curriculum

Year 7 - Computer science studied for 1 cycle during enrichment

Topic 1

Networks: from semaphores to the internet

Topic 2

Clear messaging in digital media

Year 8

C1

Computer systems

C2

Programming essentials in scratch

C3

Modelling data using spreadsheets
Using media: gaining support for a cause

Year 9

C1

Programming essentials in

C2

Media: vector graphics

C3

Developing for the web

scratch part 2		
Year 10		
C1	C2	C3
Programming part 1- sequence Computer systems Programming part 2 - selection	Programming part 3 - iteration Programming part 4 - subroutines Algorithms part 1 - the essentials	Programming part 5 - strings and lists Data representations Algorithms part 2 - searching and sorting
Year 11		
C1	C2	C3
Programming part 6- dictionaries and data files Impacts of technologies Computer networks	Cyber security Databases and SQL HTML Object orientated programming	Revision and exams