DOWNHAM MARKET ACADEMY

KS3 Curriculum Overview

Computing

Curriculum Intent

In a rapidly changing digital world we want to prepare students for their future education and employability by deepening their knowledge and understanding of the technology and systems that underpin our digital world including hardware, software, data representation and networking.

We want to create resilient and confident problem solvers who know how to plan and create solutions through a high level programming language. Through this learning journey I want to enthuse students and help them develop a passion for programming.

We want students to know about their data, privacy, threats and how to stay safe online including an understanding of computer related legislation.

The aim is to ensure that we have digitally literate students who are able to use, express and develop their ideas through information and communication technology at a level suitable for the future workplace.

How does the KS3 curriculum build on that from KS2?

The KS3 Computing Curriculum increases the depth and breadth of content within the 5 key foci of: Communication, Networks and Information Technology; Data Representation; Algorithms; Programming and Hardware

What do students *do* with this knowledge or these skills?

Students:

- apply the fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation
- analyse problems in computational terms, and have repeated practical experience of writing computer programs in order to solve problems
- evaluate and apply information technology analytically to solve problems
- become responsible, competent, confident and creative users of information and communication technology

EASTERN LEARNING ALLIANCE

How does the KS3 curriculum align to the National Curriculum?

The KS3 Curriculum aligns to the National Curriculum in the following ways:

- students are taught several key algorithms that reflect computational thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility of alternative algorithms for the same problem
- we use 2 or more programming languages (Python and Edublocks), at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions
- students are taught simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming; understand how numbers can be represented in binary, and be able to carry out simple operations on binary numbers [for example, binary addition, and conversion between binary and decimal]
- students are taught the hardware and software components that make up computer systems, and how they communicate with one another and with other systems
- students are taught how instructions are stored and executed within a computer system; understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits
- students are given the opportunity to create, reuse, revise and repurpose digital artefacts for a given audience, with attention to trustworthiness, design and usability
- students are taught a range of ways to use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy; recognise inappropriate content, contact and conduct, and know how to report concerns

What new knowledge or skills are students taught?

Term	Year 7	Year 8	Year 9
Autumn	**What IT systems the school use and how students should communicate online.** Get a username, login and create a memorable and secure password for an account on the school network and explore the services available at DMA. Know how to respectfully communicate with peers online. **What Computer Systems are and how they work.** Understand what a computer system is and identify input and output device. Identify the key hardware components of a computer system and the role they play in processing data.	**How a computer systems processes information. How one computer systems can perform better than another.** Understand the fetch-decode-execute cycle. Know the key components of the CPU and those within the fetch-decode-execute cycle including CPU Registers. Know the factors that affect the performance of a CPU. **How search engines work.** Know search engines use web crawler programs to index the World Wide Web & how results are ranked.	**What computer networks are and how the web is different from the internet.** Understand why we network computers and explain the role of some network components Understand the difference between the internet and the world wide web. Understand the role of DNS, the need for data packet and the importance of the TCP/IP protocol. Know the factors that affect the performance of a network, specifically bandwidth & latency.

	Compare computer systems against key characteristics.	**What an interactive multimedia product is.** Plan a multimedia promotional presentation to advertise and encourage people to attend a live music event by apply an effective house style and sourcing multimedia. Produce a multimedia product with effective navigation, action buttons and mouse overs.	T**hreats to a computer system/network and how I protect my personal data.** Understand personal data, how organisations might use it and the principles of the DPA. Recognise how human errors pose security risks to data and implement strategies to minimise the risk of data being compromised through human error. Define hacking in the context of cybersecurity and the need for the Computer Misuse Act. Discuss malware threats and prevention strategies
Spring	**How spreadsheets can be used to manipulate data.** Explain the difference between formula and functions including SUM, MIN, MAX and AVERAGE. Apply conditional formatting, validation and countif to model an event. Use an IF statement to highlight data changes and produce a graph to present data. **What flowcharts are and how they are useful.** Identify steps in a process (algorithms) to create flowcharts for control systems. Create, implement and debug a flowchart containing and understanding the need for decision symbols and subroutines. Understand how large/complex control systems are split into manageable chunks.	**Binary, hexadecimal and Boolean logic.** Understand measurements of storage; what the prefixes kilo-, mega-, giga- and tera- mean including converting. Convert binary digits to decimal digits and use ASCii to decode/encode messages. **Algorithms and pseudocode.** Create & sequence instructions in a flowchart using a range of symbols including the decision symbol. Produce flowcharts & pseudocode that incorporate decisions. Produce flowcharts & pseudocode that incorporate decisions. Perform an algorithm trace and use flowgorithm as a problem solving tool and repeat instructions (iteration). To know how computers use variables to store information and be able to distinguish between string, numerical and Boolean data types.	**Representing and storing data.** Understand the need for binary and how to convert binary to denary and vice versa. Describe how an image can be represented as a sequence of bits and define the key terms: 'pixels', 'resolution', and 'colour depth'. Describe how sounds are represented as sequences of bits and define key terms such as 'sample', 'sampling frequency/rate', 'sample size'. Calculate representation size for a given digital sound, given its attributes and explain how attributes such as sampling frequency and sample size affect characteristics such as representation size and perceived quality, and the trade-offs involved. Understand what is meant by Computational Thinking and can use it to solve problems. **How to interpret & understand problems using algorithms and how to plan programs using a flowcharts.** Show an understanding of selection and be able to explain what is meant by iteration and explain the difference between for & while loops. To be able to understand why we use a procedure/functions, effectively use them and understand the difference between them.
Summer	**How to plan, create, debug and modify a program in block and text based languages.** To understand how a sequence of code works and use loops to repeat a sequence. Capture different types of user input into variables and use conditional statements to change output. Understand what a function with argument is, create loops with ranges and select the correct data types. Understand the need for different data types, use integers in programs and apply successful elif when using selection	**Text based programs and programming physical devices.** Understand the use of variables when sequencing and looping images on a Microbit. Use a range of list methods including indexing and understand the efficiency of loops when appending items to a list. Use mathematical operators to produce a calculator program containing procedures that require data to be passed to them. Know the difference between procedures & functions	**Programming structures; sequence, selection, iteration, lists, subroutines.** Produce a variety of programs that demonstrate an understanding of variables, data types, selection, iteration, arrays, functions and procedures.

	Write, call and pass values to procedures		
Rationale for this sequencing	The sequencing of units and lessons allow for a logical progression of understanding and skills. Additionally the units and lessons follow the 5 core foci/strands consistently which enables the students' learning journey to be tracked effectively.		