



## Welcome to KS3 ICT

### Review: Covered. To cover. Changes made

	<b>Year 7</b>	<b>Year 8</b>	<b>Year 9</b>
<b>Autumn 1</b>	<p><b>Introduction</b></p> <ul style="list-style-type: none"> <li>• File management and organisation</li> <li>• Ergonomics</li> <li>• Internet safety</li> </ul> <p>Ergonomics / Physical Stress / Lighting Stress / Social Media / Online Identity / Offline Identity / Acceptable Use Policy / Input / Output/ trust/ Responsible /conduct/ report concerns/ identity/ protecting/secure/ respectful/safe</p>	<p>Introduction</p> <ul style="list-style-type: none"> <li>• Ergonomics</li> <li>• Internet safety</li> <li>• Web browsing</li> <li>• Accessibility tools and settings</li> <li>• Email</li> </ul> <p>Ergonomics / Physical Stress / Lighting Stress / Social Media / Online Identity / Offline Identity / Acceptable Use Policy / Input / Output/ trust/ Responsible /conduct/ report concerns/ identity/ protecting/secure/ respectful/safe</p>	<p>Introduction</p> <ul style="list-style-type: none"> <li>• Ergonomics</li> <li>• Internet safety</li> <li>• Web browsing</li> <li>• Accessibility tools and settings</li> <li>• Email</li> </ul> <p>Ergonomics / Physical Stress / Lighting Stress / Social Media / Online Identity / Offline Identity / Acceptable Use Policy / Input / Output/ trust/ Responsible /conduct/ report concerns/ identity/ protecting/secure/ respectful/safe</p>
<b>Autumn 2</b>	<p>School electrotonic Systems</p> <ul style="list-style-type: none"> <li>• Logging into systems</li> <li>• e-communication-Email</li> <li>• Web browsing</li> <li>• Accessibility tools and settings</li> <li>• Email</li> </ul> <p>Sorting/ Searching/ Privacy/ Password/ Email Address/ Communication/ Understand several key</p>	<p>Game Design &amp; Creation</p> <ul style="list-style-type: none"> <li>• Practise</li> <li>• Design</li> <li>• Create their own game</li> <li>• Exporting executable files and use of Microsoft Publisher</li> </ul> <p>Audience/ Print/ layout /Graph/ types /Sprites/ Objects/ Properties/ Events /Cell reference /Formula Function/</p>	<p>Hardware and App creation</p> <ul style="list-style-type: none"> <li>• Input, Output, Storage devices</li> <li>• Technology for Special Needs</li> <li>• Design Hierarchy &amp; Microsoft PowerPoint</li> <li>• Create own phone app</li> </ul> <p>Syntax/ Debugging/ Sequence/ Iteration /Flowcharts/ Pseudo coding/ Types of Testing: Unit, System/ User Syntax/</p>

	<p>algorithms/ Computational thinking/ Logical reasoning / Hardware/software components/communicate/ systems</p> <p>Using Deliver Computing 360 now to provide this</p>	<p>Feedback/ Evaluation /Filter and Queries creative projects/selecting/using/ combining multiple applications/range of devices/challenging goals/collecting/ analysing data/needs /users</p>	<p>Logic gates – AND, OR, NOT Devices – Input, Output /Storage Instructions/stored/executed/ data/types /text/sounds/pictures/binary digits</p>
<b>Spring 1</b>	<p>Computer Animation Storyboarding /Creating Editing Analysis Sequencing Selection Audience Evidence Queries (Changed to LOGO on Deliver Computing 360)</p>	<p>Spreadsheets &amp; Excel</p> <ul style="list-style-type: none"> <li>• Cells, formatting and printing</li> <li>• Calculations, Formulae and Functions</li> <li>• -Graphs and Filters</li> </ul>	<p>Spreadsheets &amp; Excell</p> <ul style="list-style-type: none"> <li>• Data types</li> <li>• Formatting tables</li> <li>• Sorting tables</li> <li>• Cell referencing</li> <li>• Calculations, Formulae and Functions</li> <li>• Graphs and Filters</li> </ul>
<b>Spring 2</b>	<p>Computer Systems</p> <ul style="list-style-type: none"> <li>• Ecological aspects</li> <li>• History of computing</li> <li>• Computing in the community</li> <li>• Use of Microsoft Word</li> </ul> <p>Ecological/ History of computing/Computing in the community/Microsoft Word</p>	<p>Cells/ formatting/printing/ Calculations/Formulae/ Functions/Graphs/ Filters programming languages/ textual/ computational problems /data structures /lists/tables/arrays/ design/develop/modular programs/procedures /functions</p>	<p>Decomposition/ Flowcharts/ Coding/ Drawing and control create/re-use/revise/ re-purpose digital artefacts/</p>
<b>Summer 1</b>	<p>Scratch Project</p> <ul style="list-style-type: none"> <li>• Drag and drop coding</li> <li>• Repetition and selection</li> </ul> <p>Vocabulary/ real-world problems /physical systems/algorithms/ computational thinking/ sorting/ searching/ logical reasoning</p>	<p>Video Editing</p> <ul style="list-style-type: none"> <li>• creating adverts and short clips</li> <li>• editing sound files</li> </ul> <p>create/re-use/revise/ re-purpose digital artefacts/edit/sound/advert /clip/ create/re-use/revise/ re-purpose digital artefacts/</p>	<p>Binary and Logic</p> <ul style="list-style-type: none"> <li>• Number systems - Denary – Binary – Hex</li> <li>• Logic Gates, Truth Tables and circuits</li> </ul> <p>Number systems/Denary/Binary /Hex/Logic Gates/Truth Tables/circuits Boolean logic: AND, OR and NOT/ circuits/programming/ numbers and binary/binary addition/ conversion/binary/ decimal</p>
<b>Summer 2</b>	<p>Databases &amp; Querying Data</p> <p>Vocabulary/ creative projects/selecting/using/ combining multiple applications/range of</p>	<p>Databases and use of Microsoft Access Database/ Microsoft Access / computer system/accessible /application/ creative projects/selecting/using/ combining multiple</p>	

	devices/challenging goals/collecting/ analysing data/needs /users	applications/range of devices/challenging goals/collecting/ analysing data/needs /users	
--	---	---	--

Made my R.T. To be checked by J.H

**Computing Deliver 360 curriculum KS3:**

Summary:

- 00 Baseline
- 01 Safety and Security
- 02 Word Processing
- 03 LOGO
- 04- Control Systems
- 05- Movie Making
- 06- HTML
- 07- Spreadsheets
- 08-Programming 1
- 09-The Computer
- 10-Animation
- 11- Databases
- 12- Kiosk
- 13- Programming 2
- 14-Binary 1
- 15-Graphics Project

Could be broken down as follows OR Revisit themes each year

Computing Deliver 360 curriculum KS3:

ALL first week back- 00 Baseline

Year 7:

- Autumn 1- 01 Safety and Security, General Navigation
- Autumn 2-02 Word Processing
- Spring 1- -03 LOGO
- Spring 2 -04- Control Systems
- Summer 1 -05- Movie Making
- Summer 2-06- HTML

Year 8:

- Autumn 1- Safety and Security, General Navigation
- Autumn 2- 07- Spreadsheets
- Spring 1- 08-Programming 1

Spring 2- The Computer  
Summer 1- 09- Animation  
Summer 2- 10- Databases

Year 9:

Autumn 1- Internet safety, General Navigation  
Autumn 2- 12- Kiosk  
Spring 1- 13- Programming 2  
Spring 2- 14-Binary 1  
Summer 1- 15-Graphics Project  
Summer 2-11- GCSE Prep information and consolidation of KS3

### **Vocabulary from the National Curriculum:**

computational thinking/creativity/mathematics, science, and design and technology/both natural and artificial systems /computer science information and computation/digital systems/programming/create/programs/systems and a range of content digitally literate/express/information/communication technology/digital/computer science/abstraction, logic/algorithms/data representation/analyse/computational terms/writing computer programs/evaluate and apply /solve problems/responsible/competent/confident/creative

#### KS3 Outcomes:

- real-world problems /physical systems/algorithms/computational thinking/ sorting/ searching/ logical reasoning
- programming languages/ textual/ computational problems /data structures /lists/tables/arrays/design/develop/modular programs/procedures /functions
- Boolean logic: AND, OR and NOT/ circuits/programming/ numbers and binary/binary addition/ conversion/binary/ decimal
- Hardware/software components/communicate/ systems
- Instructions/stored/executed/data/types /text/sounds/pictures/binary digits
- creative projects/selecting/using/ combining multiple applications/range of devices/challenging goals/collecting/ analysing data/needs /users
- create/re-use/revise/ re-purpose digital artefacts/ Audience/trustworthiness/design/usability
- safely/respectfully/responsibly/securely/protecting/online identity/ privacy/inappropriate content/contact/conduct/report concerns

#### KS1/KS2 Recap

Abstraction/Algorithm/Binary/Coding/Communication technology/Compile/ Computational logic and thinking/Data/Debug /Decomposition/Hardware/Information technology/Input /Internet /Logic/Network/Output /Procedure or function/Program/Programming

language/Repetition/Selection/Sequence/Software /System (Operating System)  
/Variable/World Wide Web

### KS1/KS2 Recap

- Abstraction
- Algorithm
- Binary
- Coding
- Communication technology
- Compile
- Computational logic and thinking
- Data
- Debug
- Decomposition
- Hardware
- Information technology
- Input
- Internet
- Logic
- Network
- Output
- Procedure or function
- Program
- Programming language
- Repetition
- Selection
- Sequence
- Software
- System (Operating System)
- Variable
- World Wide Web

### **Vocabulary that can be used in BOLD**

Coverage Check- National Curriculum- KS3 Computing

#### Aims:

Computing programmes of study: key stages 3 and 4

National curriculum in England

#### Purpose of study

A high-quality computing education equips pupils to use computational thinking and creativity to understand and change the world. Computing has deep links with **mathematics, science, and design and technology**, and provides insights into **both natural and artificial systems**. 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 content**. Computing also ensures that pupils become **digitally literate** – able to use, and **express** themselves and develop their ideas through, **information and communication technology** – at a level suitable for the future workplace and as active participants in a **digital** world.

### **Aims**

The national curriculum for computing aims to ensure that all pupils:

- can understand and apply the fundamental principles and concepts of **computer science, including abstraction, logic, algorithms and data representation**
- can **analyse** problems in **computational terms**, and have repeated practical experience of **writing computer programs** in order to solve such problems
- can **evaluate and apply** information technology, including new or unfamiliar technologies, analytically to **solve problems**
- are **responsible, competent, confident and creative** users of information and communication technology.

### **Attainment targets**

By the end of each key stage, pupils are expected to know, apply and understand the matters, skills and processes specified in the relevant programme of study.

Schools are not required by law to teach the example content in [square brackets].

### **Outcomes:**

- design, use and evaluate computational abstractions that model the state and behaviour of **real-world problems** and **physical systems**
- understand 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
- use two or more **programming languages**, 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

- understand 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**]
- understand the **hardware** and **software components** that make up computer systems, and how they **communicate** with one another and with other **systems**
- understand 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**
- undertake **creative projects** that involve **selecting, using, and combining multiple applications**, preferably across a **range of devices**, to achieve **challenging goals**, including **collecting** and **analysing data** and meeting the **needs** of known **users**
- **create, re-use, revise and re-purpose digital artefacts** for a given **audience**, with attention to **trustworthiness**, design and **usability**
- understand 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.**

## Vocabulary for computing KS1 and 2

### Subject Knowledge and Domain Specific Vocabulary for each Year Group

Do you know what your child means when they're talking about compiling algorithms and debugging programs? Computing teacher Billy Rebecchi explains the primary school computing and ICT terms that you might hear from your KS1 or KS2 child.

### **Abstraction**

In computer science the term **abstraction** refers to hiding the complexity of tasks to suit the understanding of the user. For example, for you to use a calculator you only have to press buttons in order to receive the correct answer, however the person that built the calculator understands how it works underneath.

### **Algorithm**

An **algorithm** is a set of instructions that we complete in order to achieve a task. You could write an algorithm to complete mundane tasks such as

making a cup of tea or to complete complex tasks such as calculating the odds that a team will win a football match. In computing an algorithm refers to the set of instructions that a computer follows in the order in which they are given.

### **Binary**

Binary is the language computers use. It is a series of 1s and 0s and is also used in mathematics.

### **Coding**

**Coding** is putting information and commands into a **program**, making it possible for u to create software, apps and websites.

### **Communication technology**

Equipment that we use to communicate with, such as a mobile phone or tablet.

### **Compile**

When we program, we use human words in our codes and programs. However the computer doesn't understand human words, so we have to compile the program – using a compiler – which converts the human words into binary.

### **Computational logic / thinking**

**Computational logic** is a term that describes the decision-making progress used in programming and writing algorithms.

### **Data**

**Data** is Information.

### **Debug**

**Debugging** is checking the code in a computer program to ensure it works, and changing it if it doesn't. When writing a computer program things will often go wrong. When writing a program you have to test and debug your program to ensure that it produces correct results.

### **Decomposition**

**Decomposition** is the process by which a large, difficult problem can be broken down into a series of smaller, simpler problems, thus making the overall problem easier to solve.

### **Hardware**

**Hardware** is the physical part of a computer, which uses electrical signals to complete the calculations needed to make software run. Examples of



hardware are the computer circuit board, memory, processor and/or other equipment related to a computer, such as printers, monitors and keyboards.

### **Information technology**

A term used for all computer-related technology.

### **Input**

Information that goes into the computer.

### **Internet**

A network of computers linked all over the world.

### **Logic**

When making any decision a certain amount of logic is involved; for example, when deciding what to wear in the morning, you make a logical decision based on the season, weather and any number of other factors. **Computational logic** is used to allow a program to decide what to do and when. For example you may write code that says: "When the user clicks this button, perform this calculation."

### **Network**

Computers linked within a building or area.

### **Output**

Information that comes out of the computer.

### **Procedure or function**

A **procedure/function** is used in programming to break a complex task down into simple steps or sections.

### **Program**

A **computer program** is a collection of instructions or algorithms designed to simplify processes, whether that be writing a Word document or connecting to a website. A computer program is written using a **programming language**, which allows a computer scientist to teach a computer how to achieve a result. Examples of programming languages are Scratch, Java, Python, C++ and Ruby.

### **Programming language**

Computers are very good at completing lots of mathematical functions in a short space of time, however they don't have the ability to think for themselves. **Programming languages** bridge this gap and allow us to teach a computer how to do things.

## **Repetition**

Sometimes called iteration, when part of a program repeats itself. For example, in animation you may repeat the movements of a character to make it look like it's moving along.

## **Selection**

When you choose part of something. For example, when you copy and paste text, the passage that you highlight to copy is called the selection.

## **Sequence**

When doing anything in life it is important to complete things in the correct order; you wouldn't pour water into a teacup before you had boiled the kettle, for example! In a program we have to control what happens and when in order to produce correct results. A **sequence** helps us to ensure that things happen in the correct order.

## **Software**

**Software** is created using a **programming language** and is the non-physical part of a computer. Software can be written once and sold multiple times, for instance Microsoft doesn't have to rebuild Microsoft Word every time they have a new customer, they just make a copy of the files they already have.

## **System (Operating System)**

The **Operating System** sits between the software and hardware and acts as a translator. It tells the hardware when to run calculations and passes the answers back to the software so that it can decide what calculations to run next.

## **Variable**

A **variable** is a piece of information in a program that we want to store, but is able to change. We can compare it to a box in which we put information. This information could be a number, and during the program we might change the initial number (for example as part of the scoring system in a game).

## **World Wide Web**

This is like the Operating System for the internet. We use the web to help us communicate with and over the internet.

New planning for new software system Deliver Computing 360 in order og units						
Computing/ ICT	1	2	3	4	5	6
<b>Autumn 1</b> <b>00- Baseline</b> <b>01- Safety and Security</b>	12.09.22  Basics of computing navigation File managemen t and organisation Ergonomics	19.09.22  Internet Safety- Missed due to teacher absence	26.09.22  Accessing computing delivery and complete a year 7 initial assessment	03.10.22  Internet Safety Posters	10.10.22  Just got new Deliver Computing 360 curriculum (KS3) 360 - 00- Baseline	17.10.22  Deliver Computing 360 - Safety and Security
<b>Autumn 2</b> <b>02-Word processin g</b> <b>Emailing</b>	31.10.22 Deliver Computing 360 Stage 1 - Word Processors  Web browsing	07.11.22  Stage 2 - Keyboard Shortcuts & Document Layout	14.11.22 Completing Antibullying survey and sending emails	21.11.22 Word processor- Shortcuts Emailing and completing antibullying survey	28.11.22 Complete as much as possible of 'Word processor' Stage 3 - Adding Objects Stage 4 - Margins & Grammar Stage 5 - Tables & Content Pages Stage 6 - Mail Merge	05.12.22 Using word to make a gonk instruction poster
<b>Spring 1</b> <b>03- LOGO</b>	04.01.23  LESSON Stage 1- Logo Commands	09.01.23 Stage 2 - Loops and Variables  Tasks- house	16.01.23 Stage 3 - Colouring Shapes  Task- Logo poster	23.01.23 Stage 4 - Sub Programs	30.01.23 Stage 5 - Creating Sub Programs	06.02.23  Stage 6 - Produce a Main Program  End of term assessment Unit test
<b>Spring 2</b> <b>04-Control Systems</b>	20.02.23 LESSON Stage 1 - Flowcharts  Tasks- Zebra	27.02.23  Stage 2 - Crossing Patrol Task - Patrol	06.03.23 Stage 3 - Flowcharts and Safety Task- Bridge	13.03.23 Stage 4 - Variables and Conditions (7.5%) Task- Kids Mobile	20.03.23 Task- Pedestrian  Task - Flowchart Symbols	27.03.23  End of term Assessment - Unit test

<b>Summer 1 05-Movie Making</b>	24.04.23 Stage 1 - Movie Making Basics	01.05.23 Stage 2 - Football Valley	08.05.23 Stage 3 - Pan and Zoom	15.05.23 Stage 4 - Action Movie Part 1  Stage 5 - Action Movie Part 2	22.05.23 Stage 6 - Action Movie Part 3 Task: Movie poster End of term assessment Unit test	
<b>Summer 2 06-HTML</b>	05.06.23 Stage 1 - Introduction to HTML Stage 2 - Templates	12.06.23 Stage 3 - Webpage with Graphic  Stage 4 - Colours and Tables	19.06.23 Stage 5 - Bullets and Numbering  Stage 6 - Layouts and Emails	26.06.23 Stage 7 - Advanced Code  End of term assessment Unit test	03.07.23 LAST WEEK OF TERM  Feedback and targets for next year	

### **Touch typing lessons done on TTRS during carousel**

#### **Year 7 Coverage:**

### **National Curriculum Coverage -Computing– Updated Autumn 2022**

#### **Aims**

#### **The national curriculum for computing aims to ensure that all pupils:**

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

#### **Subject content Key stage 3**

#### **Pupils should be taught to:**

- design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems
- understand 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
- use two or more programming languages, 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

- understand 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]
- understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems
- understand 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
- undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users
- create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability
- understand 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.

**Vocabulary definitions:**

- Algorithms- a process or set of rules to be followed in calculations or other problem-solving operations, especially by a computer.