DIGITAL SCHOOLHOUSE together with NINTENDO SWITCH

digitalschoolhouse.org.uk
dsh@ukie.org.uk

Teaching Guide.

F1 Recycled Racers

While the curriculum material within this workshop is suitable for all ages, please be aware that the video game used in this context is rated PEGI 3 - suitable for ages 3+ only. For more information about PEGI ratings please visit AskAboutGames https://www.askaboutgames.com/ or the Video Standards Council Rating Board https://videostandards.org.uk/RatingBoard/.

Introduction

In this workshop, students will learn how to create a balloon powered recycled racer and race it in their own class F1 race. To help them keep track of the winner of their races, they will also learn about different types of data and how to use a table to store it. Students will also learn about how to use a spreadsheet to store and manipulate data. This workshop uses recycled materials. You might like to arrange for learners to bring the recycled materials that they will need for this workshop from home. For more details, see the resources section.

Contents

Learning Outcomes

1. Follow an algorithm written in sequence
2. Identify how energy is transferred to a car
3. Draw a picture to represent inputs and outputs
4. Use a table to present information
5. Choose the correct order of instructions to solve a problem
6. Name the parts of spreadsheet
7. Use a cell reference to identify a cell
8. Identify examples of text and number data types
9. Define the meaning of text and number data types
10. Label the parts of a table
11. Identify inputs as a result of outputs
12. Use logic to predict what will happen when air is blown into a balloon powered car
13. Draw lines to show how an electric circuit is wired together
14. Identify examples of text and number data types from a range of examples
15. Draw a sequence of graphical instructions to solve a problem
16. Collect data and display it in a table
17. Evaluate what will happen when air is blown into a balloon powered car
18. Write their own examples of text and number data types

Files/Resources

U = Unplugged activity, B = Beginner activity, I = Intermediate activity, A = Advanced activity

Please note: Unless otherwise stated, resources are required per student / per pair.

Filename / Resource	Resource Type	Purpose/Description	Activity No
Pen or pencil	Classroom resource	Completing activities	2, 3, 4, 5, 6, 7
Coloured pens, pencils or crayons	Classroom resource	Completing activities	2, 3, 4
DSH-Teaching-Presentation-F1-Recyled-Racers	Teaching presentation	Introducing activities	All activities
DSH-worksheets-F1-Recycled-Racers	Worksheets	Completing unplugged activities	2, 5, 6, 7, 10
DSH-reference sheet-balloon powered car instructions	Instructions	Instructions to create recycled racer	1
4 plastic bottle tops	Classroom resource	Creating recycled racer	1
1 750ml plastic bottle	Classroom resource	Creating recycled racer	1
2 straws	Classroom resource	Creating recycled racer	1
1 balloon	Classroom resource	Creating recycled racer	1
2 bamboo skewers	Classroom resource	Creating recycled racer	1

Ruler	Classroom resource	Creating recycled racer	1
Sticky tape	Classroom resource	Creating recycled racer	1
Pair of scissors	Classroom resource	Creating recycled racer	1
Get started - Recycled racers	Spreadsheet file	Spreadsheet for recording race results	10, 12
Mini whiteboard and pen or scrap paper	Classroom resource	Required for recording race results	11
PC or laptop	Hardware	Required for creating spreadsheet	7, 8, 9, 10, 12
Spreadsheet software e.g., Microsoft Excel or Google Sheets	Software	Required for creating spreadsheet	7, 8, 9, 10, 12

PLEASE NOTE: The activities outlined in this workshop pack are a suggested outline of how the workshop can be delivered. It is envisaged that teachers will adapt the resources and the organisation of them according to the needs of their class.

Session Overview

Session1: If students are not going to punch the holes in the bottle tops to make wheels for themselves, complete this task prior to the session.

Session 4: Save the Get started - Recycled racers file in a location that your students can access (or if your students are more able, they can start with a blank spreadsheet).

Use your masking tape to mark out start and end lines for each student group to use for their race. It is recommended that you have a race area for each group as this will reduce the time it will take to run the race segment of the lesson.

U = Unplugged activity

SESSION 1

Activity No	Session Content / Activity	Resources Used
	Introduce the LOs for the session to your students and then read the class the About recycled racers section. Explain to your students that they will be building cars from recycled materials to take part in a race and that they will learn how to represent the information about who wins the race on their computer. Read your students the fun fact about Bertone's car that runs on recycled plastic.	DSH-Teaching-Presentation-F1-Recyled-Racers slides 1 – 7

U1	Invite your students to follow the instructions in DSH-reference sheet-balloon powered car instructions to build their balloon powered car. Students can build one car each or one between two. Remind your students to take care while using scissors and to use their ruler to measure their parts accurately. Move around the room and help where required. Give your students some time to test their cars and make sure that they understand how to make them move.	DSH-Teaching-Presentation-F1-Recyled-Racers slide 8 DSH-reference sheet-balloon powered car instructions Per student (or pair of students): 4 plastic bottle tops 1 750ml plastic bottle 2 straws 1 balloon 2 bamboo skewers Sticky tape Pair of scissors Ruler
U2	Students should decorate their cars to make them look different to the rest of their classes. Cars can be decorated using coloured pens, pencils or crayons and paper. Adding anything heavy to the cars should be avoided as this will make the cars harder to move. Remind students to make sure that they can still blow into the straw to inflate the balloon.	DSH-Teaching-Presentation-F1-Recyled-Racers slide 9
Check-point	Invite your students to use the letter at the beginning of each instruction to write the correct order that the	DSH-Teaching-Presentation-F1-Recyled-Racers slide 10

	instructions should be in to make the balloon powered car move. **Answer:** C, B, A	DSH-worksheets-F1-Recycled-Racers slide 2 (printed 1 per car)

SESSION 2

Activity No	Session Content / Activity	Resources Used
	Introduce the LOs for the session to your students.	DSH-Teaching-Presentation-F1-Recyled-Racers slides 17 – 18
	Tell your students that a table is used to show information in a way that makes it easier for people to read and understand. Show your students slide showing the parts of a table. Explain that a table is a grid of columns and rows while pointing to the example. State that the little rectangles make up the grid are called cells. Spend some time pointing to the different parts of the table and asking your students to call out the name of that part.	DSH-Teaching-Presentation-F1-Recyled-Racers slides 19
U5	Recap that last session your students created a recycled race car that used a balloon to make the car move. Alternatively,	DSH-Teaching-Presentation-F1-Recyled-Racers slide 20

	you can ask your students to recap the session for you. Explain that Mia would like some help to find out what parts she will need so she can build Beau's race car too. Point to the table in the workbook and show your students how Mia has written the names of the parts that she needs to make a balloon powered racer in the table. Ask your students if they can see anything missing from the table. They should be able to tell you that Mia has forgotten to write down how many she needs of each part. Ask your students to look at the name of the part and its picture in each row (point these out to your students) and write down the number of each part needed in the column with the header 'How many do you need?'. Tell your students that a header is the word we used to talk about the title of a column, explain that the first answer has been done for them. **Answer:** Plastic bottle tops – 4 750 ml plastic bottle – 1 Straws – 2 Balloon – 1 Bamboo skewers - 2	DSH-worksheets-F1-Recycled-Racers slide 3 (printed 1 per student)

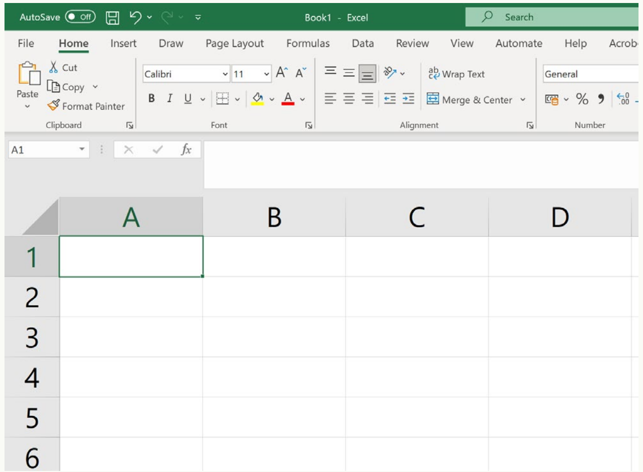
<table>
<tr>
<td><strong>U6</strong></td>
<td>Invite your students to draw lines to form a table around the information displayed on their worksheet.
<strong>Answer:</strong>
<table>
<tr><th>Car colour</th><th>Number of cars</th></tr>
<tr><td>🟥</td><td>3</td></tr>
<tr><td>🟩</td><td>1</td></tr>
<tr><td>⬛</td><td>2</td></tr>
</table>
</td>
<td>DSH-Teaching-Presentation-F1-Recyled-Racers slide 21
DSH-worksheets-F1-Recycled-Racers slide 4 (printed 1 per student)</td>
</tr>
<tr>
<td><strong>Checkpoint</strong></td>
<td>Direct your students to label the parts of the table.</td>
<td>DSH-Teaching-Presentation-F1-Recyled-Racers slide 22</td>
</tr>
</table>

SESSION 3

Activity No	Session Content / Activity	Resources Used
	Introduce the LOs for the session to your students.	DSH-Teaching-Presentation-F1-Recyled-Racers slides 23 – 24
	Begin by reminding your students that they learnt how to use a table to set out information in the previous session. Explain that in this session, your students will learn about software called a spreadsheet which can be used to help them set out information in a table. Tell your students that there are a few different pieces of spreadsheet software but	DSH-Teaching-Presentation-F1-Recyled-Racers slides 25

	two of the most popular are Microsoft Excel and Google Sheets. Introduce the software title that you will be using.	
U7	Direct your students to look at the picture of the spreadsheet on their worksheet and ask them if they can see anything about it that is similar to what they already know about tables. They should be able to tell you that the spreadsheet is also made of rows, columns, and cells. Use printable 1c. Blank spreadsheet to point out the parts of the spreadsheet – table, row, column, cell. Ask your students to use the words in the box to label the parts of the spreadsheet. **Answer:** A: Column B: Table C: Row D: Cell	DSH-Teaching-Presentation-F1-Recyled-Racers slide 26 DSH-worksheets-F1-Recycled-Racers slide 5 (printed 1 per student)
	Explain that spreadsheet software has a lot more rows and columns that the table they used in the previous session. Excel has a maximum of 1,048,576 rows by 16,384 columns and Google Sheets has a maximum of 40,000 rows by 18,278 columns. If time allows, give your students the challenge of scrolling to the bottom of the spreadsheet software you are using.	DSH-Teaching-Presentation-F1-Recyled-Racers slide 27 Spreadsheet software Mini-whiteboard and pen (one per student)

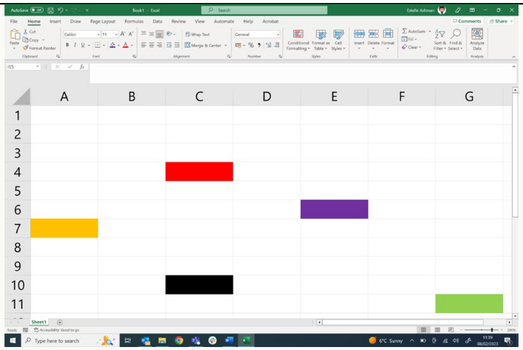
Select cell A1 by left clicking on the cell that intersects row 1 and column A:



Explain that the cell that you have selected has a cell reference which is like an address for the cell. You make up the cell reference by giving the column letter and then the row number. Tell your students that the cell that you have selected has the cell reference A1. Point out that cell references are always written with the letter first and then the number.

Select a different cell and ask your students to write down on their whiteboards what they think the cell reference is. Run through this activity a few times until you are happy that your students understand how to use cell references.

8	Once your students understand how to select cells using a cell reference, follow the steps in the teacher presentation to demonstrate how to colour a cell in using the fill tool. Students will have come across the fill tool while using Scratch, begin the demonstration by asking your students to tell you what the fill tool does. Read the top tip box and reinforce that you can see what cell is selected as it has a darker line around it and that they can check the cell reference of a selected cell by looking at the name box in the top left corner. Give your students some time to experiment with colouring in some cells and then go over how to remove colour from a cell using the information in the top tip box. There is no need to save students work for this session.	DSH-Teaching-Presentation-F1-Recyled-Racers slide 28 Spreadsheet software
9	Direct your students to colour in the cells as referenced in the teacher presentation. **Answer:**	DSH-Teaching-Presentation-F1-Recyled-Racers slide 29 Spreadsheet software

	 A B C D E F G 1 2 3 4 5 6 7 8 9 10 11	
10	Put your students into pairs. Ask your students to play a game of treasure hunt. Each student should secretly use the fill tool to colour one cell yellow on their spreadsheet. Tell your students to keep the coloured cell within columns A – G and rows 1 – 11. Students should then take it in turns to give a cell reference to their partner. If the cell reference is close to the yellow cell, they should say warm. If the reference is far away from the yellow cell they should respond with cold. Students should take it in turns to give cell references and respond accordingly until one person finds the yellow 'treasure' cell.	DSH-Teaching-Presentation-F1-Recyled-Racers slide 30 Spreadsheet software
Check-point	Ask your students to write down the cell reference for the cell that has been filled red as shown in the teacher presentation. **Answer:**	DSH-Teaching-Presentation-F1-Recyled-Racers slide 31

	D3

SESSION 4

Activity No	Session Content / Activity	Resources Used
	Introduce the LOs for the session to your students.	DSH-Teaching-Presentation-F1-Recyled-Racers slides 32 – 33
	Explain to your students that today is race day and that they will be racing the cars that they have made earlier in the workshop. Remind your students that they have learnt how to select cells and use cell references to point out cells in a spreadsheet. Ask one of your students to demonstrate how to select a cell on your teacher computer and then call out a couple of cell references and choose students at random to come and select the cell you referenced on the teacher computer. Tell your students that in this session they will learn how to type information into a spreadsheet. Explain that they will be using a spreadsheet to write down the order that people finish their recycled racers race. Explain that information that is typed into a computer is called data and that they will be learning to type two types of data – text and number.	DSH-Teaching-Presentation-F1-Recyled-Racers slides 34

11	Use the instructions in the teacher presentation to demonstrate to your students how to open Get started - Recycled racers and type their name into cell A4. Demonstrate how to use the shift key to make capital letters and explain that when you type words into a spreadsheet you are using text data. More able students can recreate the whole table in a blank spreadsheet rather than opening Get started – Recycled racers. Put your students into groups of 5 and ask them to type the names of the rest of their group into the cells below where they have typed their own name. Explain that they will need to press the enter key to move to the next cell when they have finished typing. If you have students who have the same name you may like to demonstrate how to use the space key too. Demonstrate how to use the backspace and delete keys to correct any mistakes in their typing.	DSH-Teaching-Presentation-F1-Recyled-Racers slide 35 Spreadsheet software Get started – Recycled racers spreadsheet
U12	Tell your students that it is time to run their race. Explain that they should have their mini whiteboard and pen ready to write down the order that everyone in the group finishes the race. Demonstrate how to line up the cars on the start line in one groups race area and explain that the cars need to cross the finish line – point out where their finish line is. Ask each group to show you where their start and finish line is.	DSH-Teaching-Presentation-F1-Recyled-Racers slide 36 Mini-white boards Balloon powered cars from session 1

	Tell all student groups to get their cars lined up on the start line in their groups race area and tell your students to get their cars ready to move by blowing in the straw, winding up the elastic band or turning on the switch. Give your students a countdown and ask them to let their cars go. Remind your students to write down the order that their cars cross their finish line on their whiteboard to use later.	
13	Once the races are finished, use the instructions in the teacher presentation to demonstrate how to use numbers to add the order that each person finished to their spreadsheet. Explain that we call the order that each person finishes a race their place. State that when you type numbers into a computer you are using the number data type. Demonstrate where your students will find numbers on their keyboard. Allow some time for your students to add the placings for their race to their spreadsheet. Remind them to use the information they have written on their whiteboards.	DSH-Teaching-Presentation-F1-Recyled-Racers slide 37 Spreadsheet software Get started – Recycled racers spreadsheet
Check-point	Ask students to look at the picture on the slide and then match the pieces of data to the data type they are. **Answers:** Chocolate, tree, fish, yellow – text 100, 3, 45, 7 - number	DSH-Teaching-Presentation-F1-Recyled-Racers slide 38

Digital Schoolhouse Progression Matrix

The Digital Schoolhouse progression matrix is a simplified mechanism for measuring pupil progress. It stretches from base level understanding at the beginner level to introducing GCSE content at the advanced level. The shaded statements reflect skills and concepts covered in the workshop, these have been cross referenced to specific activities in order to reflect both their level of complexity and provide a guideline on which to measure progress. For more details about this framework see ‘Enter the Matrix’ included in this workshop pack.

U = Unplugged activity, B = Beginner activity, I = Intermediate activity, A = Advanced activity

Algorithms

	Beginner	Activity No	Intermediate	Activity No	Advanced	Activity No
Understanding	Understands what an algorithm is	1	Understands that algorithms are not the same as programming		Understands that different algorithms exist for the same problem	
Writing	Represents algorithms using graphical notation such as pictures	2, 3	Represents algorithms using structured notation such as flowcharts		Represents algorithms using pseudocode	
Limitations	Understands that computers need precise instructions		Can identify tasks best completed by humans or computers		Understands that some problems cannot be solved computationally	
Planning	Can identify the steps needed to solve a problem	2, 3	Can identify the programming constructs needed to solve a problem (pattern recognition)		Can identify the modules needed to solve a problem e.g. top down design	
Tracing	Uses logical reasoning to predict outputs and show an awareness of inputs	3	Uses logical reasoning to explain how an algorithm works		Evaluates the effectiveness of algorithms and models for similar problems	
Designing	Designs solutions (algorithms) that use sequence, selection i.e. if, then and else and iteration		Designs solutions by decomposing a problem and creating a sub-solution for each of these parts		Designs a solution to a problem that uses generalization to create objects and classes (OOP)	

Programming

	Beginner	Activity No	Intermediate	Activity No	Advanced	Activity No
Writing	Can create a simple program in an environment that does not rely on text e.g. programmable robots etc		Has practical experience of a high-level textual language, including use of standard libraries		Has experience of designing programs that include a graphical user interface	
Program flow	Understands that programs execute by following precise instructions		Understands how modular programs work using sub-routines		Appreciates the effect of the scope of a variable e.g. a local variable can't be accessed from outside its function unless passed as a parameter	
Selection	Uses selection statements in programs, including an if, then and else statement		Understands the difference between, and appropriately uses if and if, then and else Statements		Uses nested selection statements	
Iteration	Uses loops, within programs		Understands the difference between, and uses 'while', 'until' and 'for' loops		Uses nested iteration and recursion	
Debugging	Detects and corrects simple semantic errors i.e. debugging, in programs		Detects and corrects syntactical errors		Applies a modular approach to error detection and correction	
Program design	Creates programs that implement algorithms to achieve given goals		Can design a program based on an algorithm		Designs modular programs using a range of methodologies e.g. RAD, waterfall	
Data types and structures	Declares and assigns variables		Selects appropriate data types		Understands and uses one and two dimensional data structures	
Operators	Uses arithmetic operators		Uses a range of operators and expressions e.g. Boolean		Understands and uses negation with operators e.g. not equal to	

Data

	Beginner	Activity No	Intermediate	Activity No	Advanced	Activity No
Representation	Recognises that digital content can be represented in many forms	5, 11	Understands how bit patterns represent different forms of data e.g. character sets, sound, numbers and images		Understands how the same bit patterns can be used for different forms of data e.g. metadata	
Transfer	Knows that data can be transferred from one computer to another		Knows that computers transfer data in binary		Understands and can explain the need for data compression, and performs simple compression methods	
Types	Recognises different types of data: text, number	6, 11, 12, 13	Defines data types: string, integer, real and Boolean		Understands how different data types can be used within data structures e.g. arrays must be made up of the same data type whereas lists can use several	
Binary	Can carry out simple binary to decimal conversions		Performs operations using bit patterns e.g. binary addition, conversion between binary and hexadecimal, binary subtraction etc		Understands the relationship between binary and electrical circuits, including Boolean logic	
File Size	Understands that data takes up space on a computer		Understands the relationship between binary and file size (uncompressed)		Knows the relationship between data representation and data quality e.g. resolution and colour depth etc, including the effect on file size	
Data and Information	Understands the difference between data and information		Recognises that poor-quality data leads to unreliable results, and inaccurate conclusions		Understand the mechanisms used to cleanse data e.g. validation, range checks etc	
Searching	Can sort data, use filters and perform single criteria searches for information	8, 9, 10	Queries data on one table using a typical query language, including more complex searches for information e.g. using Boolean and relational operators		Queries data on multiple tables using a typical query language	
Structure	Recognises that data can be structured in tables to make it useful	5, 6, 7, 8, 9, 10, 11, 12, 13	Understands that all the data about a person or thing can be stored as a record		Knows what a relational database is, and understands the benefits of storing data in multiple tables	

Hardware and Software

	Beginner	Activity No	Intermediate	Activity No	Advanced	Activity No
Processing	Understands that computers have no intelligence and that computers can do nothing unless a program is executed		Knows that programs are executed by the processor i.e. the CPU		Understand that processors can work in different ways e.g. multitasking, scheduling	
Software	Recognises that all software executed on digital devices is programmed		Knows that there is a range of operating systems and application software for the same hardware		Understands the concept of proprietary and open-source software including how this relates to licencing	
Devices	Recognises that a range of digital devices can be considered a computer		Understands why and when computers are used		Understands how technology has developed e.g. Moore's Law	
Components	Recognises and can use a range of input and output devices	7, 8, 9, 10, 11, 12, 13	Recognises and understands the function of the main internal parts of basic computer architecture		Knows that processors have instruction sets and that these relate to low-level instructions carried out in the main internal parts of a computer	
Operating systems	Understands that the operating system is software that specifies the function of a computing device		Understands the main functions of the operating system		Understands that there are different types of operating system and some of there common uses e.g. real time on auto pilot systems on a plane	
Data transfer	Knows that data is transferred around a computer system using input devices, sensors and application software		Knows that data can be transferred between computer systems using physical, wireless and mobile networks		Knows how data can be transferred between computer systems e.g. packet and circuit switching	
Architecture	Understands the difference between hardware and software		Understands how hardware uses software to execute instructions e.g. the fetch-execute cycle		Understands computer architecture in relation to the fetch execute cycle, including how data is stored in memory	

Communication

	Beginner	Activity No	Intermediate	Activity No	Advanced	Activity No
WWW	Accesses content using a web browser		Understands that web pages are created using HTML and CSS		Understands how dynamic web pages use the client-server model and that web servers process and store data entered by users	
Online safety	Understands why and how to keep personal information private and knows what to do when concerned about something online		Has an awareness of a range of online harms and demonstrates responsible use of technologies and online services in order to protect themselves		Understands how and why online threats are carried out and how to protect against them	
Search engines	Navigates the web and can carry out simple web searches to collect digital content		Understands how to effectively use search engines e.g. Boolean, advanced search functions etc		Knows how search results are selected and ranked, including that search engines use 'web crawler programs'	
Networks	Understands the difference between the internet and internet service e.g. world wide web		Understands data is transmitted between digital computers over networks, including different topologies e.g. ring, star, mesh		Knows the names and purposes of network components and protocols	
Internet services	Shows an awareness of, and can use a range of internet services e.g. email		Selects, combines and uses internet services		Uses internet services to work collaboratively	

Digital skills

	Beginner	Activity No	Intermediate	Activity No	Advanced	Activity No
Invention	Uses software under the control of the teacher to create, store and edit digital content	7, 8, 9, 10, 11, 12, 13	Uses and selects internet services, digital devices and application software to create, store and edit digital content		Evaluates the appropriateness of digital devices, internet services and application software to achieve given goals	
Audience	Understands what an audience is		Recognises the audience when designing and creating digital content		Undertakes creative projects that are tailored to meet the needs of an audience	
Purpose	Can talk about how they use computers		Can talk about how other people use computers		Can discuss the issues around how other people might use computers e.g. Data Protection Act, Computer Misuse Act, Copyright etc	
Evaluation	Can comment on the success of their solution		Designs and uses criteria to critically evaluate the quality of solutions		Documents user feedback, the improvements identified, and the refinements made to the solution	
Content	Can gather content	12	Makes judgements about content when evaluating and repurposing it for a given audience		Evaluates the trustworthiness of content, considers the usability of visual design features and properties of media when designing and creating digital artefacts	

Computing Programmes of Study Links

1.1 understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions

~~1.2 create and debug simple programs~~

1.3 use logical reasoning to predict the behaviour of simple programs

1.4 use technology purposefully to create, organise, store, manipulate and retrieve digital content

1.5 recognise common uses of information technology beyond school

1.6 use technology safely and respectfully, keeping personal information private; identify where to go for help and support when they have concerns about content or contact on the internet or other online technologies

2.1 design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts

2.2 use sequence, selection, and repetition in programs; work with variables and various forms of input and output

2.3 use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs

~~2.4 understand computer networks including the internet; how they can provide multiple services, such as the world wide web; and the opportunities they offer for communication and collaboration~~

~~2.5 use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content~~

2.6 select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information

2.7 use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact

~~3.1 design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems~~

~~3.2 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~~

~~3.3 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~~

~~3.4 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]~~

~~3.5 understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems~~

~~3.6 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~~

3.7 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

3.8 create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability

3.9 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

4.1 develop their capability, creativity and knowledge in computer science, digital media and information technology

4.2 develop and apply their analytic, problem-solving, design, and computational thinking skills

~~4.3 understand how changes in technology affect safety, including new ways to protect their online privacy and identity, and how to identify and report a range of concerns.~~

Computational Thinking Strands

AL – Algorithmic Thinking

Ref.	Activity
A1	Formulating instructions to achieve a desired effect
A2	Formulating instructions to be followed in a given order (sequence)
A3	Formulating instructions that use arithmetic and logical operations
~~A4~~	~~Writing sequences of instructions that store, move and manipulate data (variables and assignment)~~
~~A5~~	~~Writing instructions that choose between different constituent instructions (selection)~~
~~A6~~	~~Writing instructions that repeat groups of constituent instructions (loops/iteration)~~
~~A7~~	~~Grouping and naming a collection of instructions that do a well-defined task to make a new instruction (subroutines, procedures, functions, methods)~~
~~A8~~	~~Writing instructions that involve subroutines that use copies of themselves (recursion).~~
~~A9~~	~~Writing sets of instructions that can be followed at the same time by different agents (computers/people, parallel thinking and processing, concurrency)~~
~~A10~~	~~Writing a set of declarative rules (coding in Prolog or a database query language)~~
~~A11~~	~~Using an appropriate notation to write code to represent any of the above~~
~~A12~~	~~Creating algorithms to test a hypothesis~~

~~A13~~	~~Creating algorithms that give experience-based solutions (heuristics)~~
~~A14~~	~~Creating algorithmic descriptions of real world processes so as to better understand them (computational modelling)~~
~~A15~~	~~Designing algorithmic solutions that take into account the abilities, limitations and desires of the people who will use them~~

DE – Decomposition

Ref.	Activity
D1	Breaking down artefacts into constituent parts to make them easier to work with
D2	Breaking down a problem into simpler versions of the same problem that can be solved in the same way (recursive and divide and conquer strategies)

GE – Generalisation

Ref.	Activity
G1	Identifying patterns and commonalities in artefacts
G2	Adapting solutions, or parts of solutions, so they apply to a whole class of similar problems
G3	Transferring ideas and solutions from one problem area to another

AB – Abstraction

Ref.	Activity
Ab1	Reducing complexity by removing unnecessary detail
Ab2	Choosing a way to represent an artefact, to allow it to be manipulated in useful ways
~~Ab3~~	~~Hiding the full complexity of an artefact (hiding functional complexity)~~
~~Ab4~~	~~Hiding complexity in data, for example by using data structures~~
~~Ab5~~	~~Identifying relationships between abstractions~~
~~Ab6~~	~~Filtering information when developing solutions~~

EV – Evaluation

Ref.	Activity
E1	Assessing that an artefact is fit for purpose
E2	Assessing whether an artefact does the right thing (functional correctness)
E3	Designing and running test plans and interpreting the results (testing)
E4	Assessing whether the performance of an artefact is good enough (utility: effectiveness and efficiency)
E5	Comparing the performance of artefacts that do the same thing
~~E6~~	~~Making trade-offs between conflicting demands~~
~~E7~~	~~Assessing whether an artefact is easy for people to use (usability)~~

~~E8~~	~~Assessing whether an artefact gives an appropriately positive experience when used (user experience)~~
~~E9~~	~~Assessment of any of the above against the specification and set criteria~~
~~E10~~	~~Stepping through processes or algorithms/code step-by-step to work out what they do (dry run/tracing).~~
~~E11~~	~~Using rigorous argument to justify that an algorithm works (proof)~~
~~E12~~	~~Using rigorous argument to check the usability or performance of an artefact (analytical evaluation)~~
~~E13~~	~~Using methods involving observing an artefact in use to assess its usability (empirical evaluation)~~
~~E14~~	~~Assessing whether a product meets general performance criteria (heuristics)~~