DIGITAL SCHOOLHOUSE together with NINTENDO SWITCH

+44 (0) 207 534 0586
dsh@ukie.org.uk

Teaching Guide.

Machine Code Mario

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

This workshop introduces students to binary in an innovative way. Starting with investigating why computers use binary, students explore how to represent decimal numbers in binary and then how to use this knowledge to create Super Mario courses using Super Mario Maker 2 that test the players understanding of binary representation. The design, exploration and development stages of the beginner workshop fit nicely into KS2. However, as data representation is not in the KS2 curriculum this element would be a stretch and challenge exercise introducing binary, and therefore we recommend not attempting anything beyond the beginner's level with primary aged students. The intermediate workshop fits squarely into KS3, teaching students how binary shifts work, and the advanced level into GCSE, explaining how to perform binary addition and subtraction. All versions of the workshop then explore why testing is important in game design by completing alpha testing using black box and destructive testing on their courses. The intermediate version then explores how to make improvements to the courses, in addition, the advanced also explains beta testing.

ukie PlayStation. SEGA Warwickshire County Council UBISOFT

+44 (0) 207 534 0586
dsh@ukie.org.uk

Contents

Digital Schoolhouse, 18a Blackbull Yard, 24-28 Hatton Wall, London EC1N 8JH.

+44 (0) 207 534 0586
dsh@ukie.org.uk

Learning Outcomes

1. To be able to understand that computers transfer data in binary
2. To be able to understand why computers transfer data in binary
3. To be able to use and recognise a range of input and output devices relating to using a games console
4. To be able to convert denary numbers into their binary equivalents
5. To be able to convert binary numbers into their denary equivalents
6. To be able to perform a binary shift
7. To be able to explain the affect of a binary shift on a number
8. To be able to perform binary multiplication
9. To be able to perform binary division
10. To be able to perform binary addition
11. To be able to perform binary subtraction
12. To be able to understand why computer games need to be tested before going to production
13. To be able to explain and complete alpha testing of a game
14. To be able to complete black box testing of a game
15. To be able to complete destructive testing of a game
16. To be able to explain the need for user requirements
17. To be able to explain and complete beta/ acceptance testing of a game
18. To be able to identify and fix errors in their game

+44 (0) 207 534 0586
dsh@ukie.org.uk

Files/Resources

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

Please note activities are numbered alongside their difficulty level..

Filename	Resource Type	Purpose/Description	Activity No
DSH-Teaching-Presentation-Machine-Code-Mario-Beginner	Teaching presentation	Beginner version of workshop (no end of topic assessment)	All beginner activities
DSH-Teaching-Presentation-Machine-Code-Mario-Intermediate	Teaching presentation	Intermediate version of workshop (no end of topic assessment)	All intermediate activities
DSH-Teaching-Presentation-Machine-Code-Mario-Advanced	Teaching presentation	Advanced version of workshop (no end of topic assessment)	All advanced activities
DSH-Worksheets-Machine-Code-Mario	Worksheets	Presentation containing all worksheet activities	All activities
DSH-Worksheets-Machine-Code-Mario Slides 2 - 5	Worksheets	Place value cards – these can either be printed A3 to use at the front of the class or as handouts 4 to a page to use individually	BU1
Nintendo Switch	Games console	Needed to access the Super Mario Maker 2 software	B2, I3, A2, A4
Super Mario Maker 2 game disk	Software	Needed to create Super Mario Maker 2 courses	B2, I3, A2, A4
DSH SMM2 V1	Video	Super Mario Maker 2 tutorial, Episode 1: Introduction to MM2	B2,
DSH-Worksheets-Machine-Code-Mario Slides 6	Worksheet	Blank grid for planning Super Mario Maker course	B2, I3, A2, A4
DSH-Worksheets-Machine-Code-Mario Slides 7 - 8	Worksheets	Place value mats to aid with working out binary shift calculations	BU1, IU1, IU2
Counters or playing cards	Activity resource	Needed to represent binary on the place value mats	BU1, IU1, IU2
DSH SMM2 V2	Video	Super Mario Maker 2 tutorial, Episode 2: Using Warp Doors	I3
Mini whiteboards	Activity resource		AU1, AU3

+44 (0) 207 534 0586
dsh@ukie.org.uk

		Needed for binary addition and subtraction calculations	
Whiteboard pens	Activity resource	Needed for binary addition and subtraction calculations	AU1, AU3
Whiteboard rubbers	Activity resource	Needed for binary addition and subtraction calculations	AU1, AU3
DSH SMM2 V3	Video	Super Mario Maker 2 tutorial, Episode 3: Using pipes and sub-areas	A2
DSH SMM2 V4	Video	Super Mario Maker 2 tutorial, Episode 3: Using tracks and lifts	A4
Coloured pens or pencils	Activity resource	Needed to fill in test plans	B2, I3, A2, A4, B3, B4, I4, I5, A5, A6, A7
DSH-Worksheets-Machine-Code-Mario Slides 9 - 11	Worksheets	Needed to complete alpha testing of Mario courses	B3, B4
DSH-Worksheets-Machine-Code-Mario Slide 12	Worksheets	User requirements, needed to design test plans	I4, I5, A5, A6, A7
DSH-Worksheets-Machine-Code-Mario Slides 13 - 14	Worksheets	Needed to complete alpha testing of Mario courses	I4, I5
DSH-Worksheets-Machine-Code-Mario Slides 15 - 18	Worksheets	Needed to complete alpha testing of Mario courses	A5, A6
DSH-Worksheets-Machine-Code-Mario Slides 19 - 20	Worksheets	Needed to complete beta testing of Mario courses	A7

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.

+44 (0) 207 534 0586
dsh@ukie.org.uk

Session Overview

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

BEGINNER SESSION 1

Activity No	Session Content / Activity	Resources Used
	Welcome and introduction	DSH-Teaching-Presentation-Machine-Code-Mario-Beginner – slides 1 - 2
BU1	Start by reading students the definition of machine code and explain that means that Super Mario is also run in binary! Move on and read the slides explaining how binary is used in a computer and explain the concept of place values using the slides provided When you get to the Let's try it as a class slide ask for 4 volunteers and give each a place value card, spend some time demonstrating each of the binary numbers in the presentation using the binary place value cards	DSH-Teaching-Presentation-Machine-Code-Mario-Beginner - Slides 3 – 25 DSH-Worksheets-Machine-Code-Mario Slides 2 – 5 – place value cards printed A3 or printed 4 to a page as handouts for 1 set each

BEGINNER SESSION 2

Activity No	Session Content / Activity	Resources Used
B2	Read the slide that asks students to imagine a world where you need to know machine code to complete Mario courses Introduce and hand out the Switches and Super Mario Maker 2 software Watch the video explaining how to create the first Mario course using machine code – if demoing from the front you	DSH-Teaching-Presentation-Machine-Code-Mario-Beginner – Slides 26 - 29 Nintendo Switch Super Mario Maker 2 game disk DSH SMM2 V1 video

+44 (0) 207 534 0586
dsh@ukie.org.uk

	may like to pause the video to give time for the students to complete the steps Move to the slide telling students to plan and create their courses Hand out the blank grids and give time for the students to plan their courses – you may like to get the students to swap and complete some peer reviews before they begin making their actual courses Give students time to create their courses – you may like to allow some time for testing here (although there is a dedicated testing section in the last session of this workshop) Students will look at the other videos for creating different types of courses next	DSH-Worksheets-Machine-Code-Mario Slide 6 – blank Super Mario Make course grids for planning courses Coloured pens or pencils to fill in grids

BEGINNER SESSION 3

Activity No	Session Content / Activity	Resources Used
	Testing Read students the slides explaining the need for thorough testing, including the definitions of test plans and alpha testing Read students the user requirements for the Super Mario course	DSH-Teaching-Presentation-Machine-Code-Mario-Beginner – Slides 30 - 35
B3	If using the SCORM or HTML5 version of the workshop students will select their level from a slide Hand out the black box testing sheets Explain that black box testing is where you test the inputs and outputs of your program without worrying about how the program actually works	DSH-Teaching-Presentation-Machine-Code-Mario-Beginner – Slides 36 Enough of the following for 1 per student (if you intend for students to test all of their courses, each student will need 4 copies): Pens or pencils DSH-Worksheets-Machine-Code-Mario Slides 10 – beginner back box testing

Digital Schoolhouse, 18a Blackbull Yard, 24-28 Hatton Wall, London EC1N 8JH.

+44 (0) 207 534 0586
dsh@ukie.org.uk

	Show the example on the slide and then give time for students to complete the tests on at least one of their courses	
B4	Hand out the destructive testing sheets Explain that destructive testing is where you test the program to see if you can break it Show the example on the slide and then give time for students to complete the tests on at least one of their courses	DSH-Teaching-Presentation-Machine-Code-Mario-Beginner – Slides 37 Enough of the following for 1 per student (if you intend for students to test all of their courses, each student will need 4 copies): Pens or pencils DSH-Worksheets-Machine-Code-Mario Slides 11 – beginner destructive testing
	Move to the next slide and give students time to fix any errors they may have found in their games You may like to give time for students to annotate their original plans with any changes they have made	DSH-Teaching-Presentation-Machine-Code-Mario-Beginner – Slides 38 Nintendo Switch Super Mario Maker 2 game disk
	Move to the slide that tells students they have completed their testing and give some time for students to try out each other's courses	DSH-Teaching-Presentation-Machine-Code-Mario-Beginner – Slides 39 Nintendo Switch Super Mario Maker 2 game disk
	Go back over the learning objectives and ask students to point out where they have met each one	DSH-Teaching-Presentation-Machine-Code-Mario-Beginner – Slides 40

Digital Schoolhouse, 18a Blackbull Yard, 24-28 Hatton Wall, London EC1N 8JH.

Activity No	Session Content / Activity	Resources Used
	Welcome and introduction	DSH-Teaching-Presentation-Machine-Code-Mario-Intermediate – slides 1 - 5
IU1	Binary Shifts - multiplication Work through the slides explaining how binary shifts are used to multiply Hand out the place value mats and counters / playing cards Once you have completed the first worked example of a binary shift to the left, show the next slide and then give time for students to use their place values and counters to work out the number created by the shift. Show the next slide with the answer on it and ask students who got it right Complete each of the next conversions, allowing time for students to give their answers before showing the class the answer	DSH-Teaching-Presentation-Machine-Code-Mario-Intermediate – Slides 6 - 23 DSH-Worksheets-Machine-Code-Mario Slides 7 - 8 – place value mats, printed, enough for 1 each Counters or playing cards (1 per student)
IU2	Binary Shifts - division Work through the slides explaining how binary shifts are used to divide Hand out the place value mats and counters / playing cards Once you have completed the first worked example of a binary shift to the right, show the next slide and then give time for students to use their place values and counters to work out the number created by the shift. Show the next slide with the answer on it and ask students who got it right Complete each of the next conversions, allowing time for students to give their answers before showing the class the answer	DSH-Teaching-Presentation-Machine-Code-Mario-Intermediate – Slides 24 - 49 DSH-Worksheets-Machine-Code-Mario Slides 7 - 8 – place value mats, printed, enough for 1 each Counters or playing cards (1 per student)

+44 (0) 207 534 0586
dsh@ukie.org.uk

INTERMEDIATE SESSION 2

Activity No	Session Content / Activity	Resources Used
I3	Watch the video explaining how to create the next Mario course using machine code – if demoing from the front you may like to pause the video to give time for the students to complete the steps Move to the slide telling students to plan and create their courses Hand out the blank grids and give time for the students to plan their courses – you may like to get the students to swap and complete some peer reviews before they begin making their actual courses Give students time to create their courses – you may like to allow some time for testing here (although there is a dedicated testing section in the last session of this workshop)	DSH-Teaching-Presentation-Machine-Code-Mario-Intermediate – Slides 50 - 53 Nintendo Switch Super Mario Maker 2 game disk DSH SMM2 V2 video DSH-Worksheets-Machine-Code-Mario Slide 6 – blank Super Mario Make course grids for planning courses Coloured pens or pencils to fill in grids

INTERMEDIATE SESSION 3

Activity No	Session Content / Activity	Resources Used
	Testing Read students the slides explaining the need for thorough testing, including the definitions of test plans and alpha testing Read students the user requirements for the Super Mario course	DSH-Teaching-Presentation-Machine-Code-Mario-Intermediate – Slides 54 - 69
I4	Hand out the black box testing sheets	DSH-Teaching-Presentation-Machine-Code-Mario-Intermediate – Slides 60

+44 (0) 207 534 0586
dsh@ukie.org.uk

	Explain that black box testing is where you test the inputs and outputs of your program without worrying about how the program actually works. Explain that students should use the user requirements to help them write their tests Show the example on the slide and then give time for students to write and then complete the tests on at least one of their courses	Enough of the following for 1 per student (if you intend for students to test all of their courses, each student will need 4 copies): Pens or pencils DSH-Worksheets-Machine-Code-Mario Slide 12 – user requirements DSH-Worksheets-Machine-Code-Mario Slides 13 – intermediate black box testing
I5	Hand out the destructive testing sheets Explain that destructive testing is where you test the program to see if you can break it Explain that students should use the user requirements to help them write their tests Show the example on the slide and then give time for students to write and then complete the tests on at least one of their courses	DSH-Teaching-Presentation-Machine-Code-Mario-Intermediate – Slides 61 Enough of the following for 1 per student (if you intend for students to test all of their courses, each student will need 4 copies): Pens or pencils DSH-Worksheets-Machine-Code-Mario Slide 12 – user requirements DSH-Worksheets-Machine-Code-Mario Slides 14 – intermediate destructive testing
	Move to the next slide and give students time to fix any errors they may have found in their games You may like to give time for students to annotate their original plans with any changes they have made	DSH-Teaching-Presentation-Machine-Code-Mario-Intermediate – Slides 62 Nintendo Switch Super Mario Maker 2 game disk
	Move to the slide that tells students they have completed their testing and give some time for students to try out each other's courses	DSH-Teaching-Presentation-Machine-Code-Mario-Intermediate – Slides 63

+44 (0) 207 534 0586
dsh@ukie.org.uk

		Nintendo Switch Super Mario Maker 2 game disk
	Go back over the learning objectives and ask students to point out where they have met each one	DSH-Teaching-Presentation-Machine-Code-Mario-Intermediate – Slides 64

ADVANCED SESSION 1

Activity No	Session Content / Activity	Resources Used
	Welcome and introduction	DSH-Teaching-Presentation-Machine-Code-Mario-Advanced – slides 1 - 5
AU1	Binary addition Beginner students should skip straight to B7 and create their course based on binary conversions only Intermediate and advanced only: work through the slides explaining how to add in binary Hand out the mini whiteboards, pens and rubbers. Once you have completed the first worked example of a binary addition, show the next slide and then give time for students to use their white boards to work out the number created by the addition. Show the next slide with the answer on it and ask students who got it right Complete each of the next additions, allowing time for students to give their answers before showing the class the answer	DSH-Teaching-Presentation-Machine-Code-Mario-Advanced – Slides 6 - 48 Mini whiteboards (enough for 1 per student) Whiteboard pens (enough for 1 per student) Whiteboard rubbers (enough for 1 per student)

+44 (0) 207 534 0586
dsh@ukie.org.uk

ADVANCED SESSION 2

Activity No	Session Content / Activity	Resources Used
A2	Watch the video explaining how to create the next Mario course using machine code – if demoing from the front you may like to pause the video to give time for the students to complete the steps Move to the slide telling students to plan and create their courses Hand out the blank grids and give time for the students to plan their courses – you may like to get the students to swap and complete some peer reviews before they begin making their actual courses Give students time to create their courses – you may like to allow some time for testing here (although there is a dedicated testing section in the last session of this workshop)	DSH-Teaching-Presentation-Machine-Code-Mario-Advanced – Slides 49 - 52 Nintendo Switch Super Mario Maker 2 game disk DSH SMM2 V3 video DSH-Worksheets-Machine-Code-Mario Slide 6 – blank Super Mario Make course grids for planning courses Coloured pens or pencils to fill in grids

ADVANCED SESSION 3

Activity No	Session Content / Activity	Resources Used
AU3	Binary subtraction Beginner students should skip straight to B9 and create their course based on binary conversions only Intermediate and advanced only: work through the slides explaining how to subtract in binary Hand out the mini whiteboards, pens and rubbers. Once you have completed the first worked example of a binary subtraction, show the next slide and then give time for students to use their white boards to work out the number	DSH-Teaching-Presentation-Machine-Code-Mario-Advanced – Slides 53 - 87 Mini whiteboards (enough for 1 per student) Whiteboard pens (enough for 1 per student) Whiteboard rubbers (enough for 1 per student)

+44 (0) 207 534 0586
dsh@ukie.org.uk

	created by the subtraction. Show the next slide with the answer on it and ask students who got it right Complete each of the next subtractions, allowing time for students to give their answers before showing the class the answer	

ADVANCED SESSION 4

Activity No	Session Content / Activity	Resources Used
A4	Watch the video explaining how to create the next Mario course using machine code – if demoing from the front you may like to pause the video to give time for the students to complete the steps Move to the slide telling students to plan and create their courses Hand out the blank grids and give time for the students to plan their courses – you may like to get the students to swap and complete some peer reviews before they begin making their actual courses Give students time to create their courses – you may like to allow some time for testing here (although there is a dedicated testing section in the last session of this workshop)	DSH-Teaching-Presentation-Machine-Code-Mario-Advanced – Slides 88 - 89 Nintendo Switch Super Mario Maker 2 game disk DSH SMM2 V4 video DSH-Worksheets-Machine-Code-Mario Slide 6 – blank Super Mario Make course grids for planning courses Coloured pens or pencils to fill in grids

ADVANCED SESSION 5

Activity No	Session Content / Activity	Resources Used
	Testing Read students the slides explaining the need for thorough testing, including the definitions of test plans and alpha testing	DSH-Teaching-Presentation-Machine-Code-Mario-Advanced – Slides 90 - 95

+44 (0) 207 534 0586
dsh@ukie.org.uk

	Read students the user requirements for the Super Mario course	
A5	Hand out the black box testing sheets Explain that black box testing is where you test the inputs and outputs of your program without worrying about how the program actually works. Explain that students should use the user requirements to help them write their tests Show the example on the slide and then give time for students to write and then complete the tests on at least one of their courses Move to the next slide and explain that they need to think about how they would fix any errors and make a note about it on the second sheet	DSH-Teaching-Presentation-Machine-Code-Mario-Advanced – Slides 96 - 97 Enough of the following for 1 per student(if you intend for students to test all of their courses, each student will need 4 copies): Pens or pencils DSH-Worksheets-Machine-Code-Mario Slide 12 – user requirements DSH-Worksheets-Machine-Code-Mario Slides 15 - 16 – advanced black box testing
A6	Hand out the destructive testing sheets Explain that destructive testing is where you test the program to see if you can break it Explain that students should use the user requirements to help them write their tests Show the example on the slide and then give time for students to write and then complete the tests on at least one of their courses Move to the next slide and explain that they need to think about how they would fix any errors and make a note about it on the second sheet	DSH-Teaching-Presentation-Machine-Code-Mario-Advanced – Slides 98 - 99 Enough of the following for 1 per student(if you intend for students to test all of their courses, each student will need 4 copies): Pens or pencils DSH-Worksheets-Machine-Code-Mario Slide 12 – user requirements DSH-Worksheets-Machine-Code-Mario Slides 17 - 18 – advanced black box testing
	Move to the next slide and give students time to fix any errors they may have found in their games You may like to give time for students to annotate their original plans with any changes they have made	DSH-Teaching-Presentation-Machine-Code-Mario-Advanced – Slide 100 Nintendo Switch Super Mario Maker 2 game disk

Digital Schoolhouse, 18a Blackbull Yard, 24-28 Hatton Wall, London EC1N 8JH.

+44 (0) 207 534 0586
dsh@ukie.org.uk

A7	Hand out the beta testing sheets Read the beta / acceptance testing definition to students Explain that students should use the user requirements to help them write their tests Show the example on the slide and then give time for students to write and then complete the tests on at least one of their courses Move to the next slide and explain that they need to think about how they would fix any errors and make a note about it on the second sheet	DSH-Teaching-Presentation-Machine-Code-Mario-Advanced – Slides 101 - 104 Enough of the following for 1 per student(if you intend for students to test all of their courses, each student will need 4 copies): Pens or pencils DSH-Worksheets-Machine-Code-Mario Slide 12 – user requirements DSH-Worksheets-Machine-Code-Mario Slides 19 - 20 – advanced black box testing
	Move to the slide that tells students they have completed their testing and give some time for students to try out each other's courses	DSH-Teaching-Presentation-Machine-Code-Mario-Advanced – Slides 105 Nintendo Switch Super Mario Maker 2 game disk
	Go back over the learning objectives and ask students to point out where they have met each one	DSH-Teaching-Presentation-Machine-Code-Mario-Advanced – Slides 106

Digital Schoolhouse, 18a Blackbull Yard, 24-28 Hatton Wall, London EC1N 8JH.

+44 (0) 207 534 0586
dsh@ukie.org.uk

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		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	B2	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	B2	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	B2	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)	

+44 (0) 207 534 0586
dsh@ukie.org.uk

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	B2	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	B3, B4	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	B2	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	

+44 (0) 207 534 0586
dsh@ukie.org.uk

Data

	Beginner	Activity No	Intermediate	Activity No	Advanced	Activity No
Representation	Recognises that digital content can be represented in many forms	BU1, B2	Understands how bit patterns represent different forms of data e.g. character sets, sound, numbers and images	BU1, B2,	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	BU1	Understands and can explain the need for data compression, and performs simple compression methods	
Types	Recognises different types of data: text, number	BU1, B2, I3, A2, A4	Defines data types: string, integer, real and Boolean	IU1, IU2	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	BU1, B2, I3, A2, A4	Performs operations using bit patterns e.g. binary addition, conversion between binary and hexadecimal, binary subtraction etc	IU1, IU2, AU1, AU3	Understands the relationship between binary and electrical circuits, including Boolean logic	BU1
File Size	Understands that data takes up space on a computer		Understands the relationship between binary and file size (uncompressed)	IU1, IU2, AU1, AU3	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	B2, I3, A2, A4	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		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		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	

Digital Schoolhouse, 18a Blackbull Yard, 24-28 Hatton Wall, London EC1N 8JH.

+44 (0) 207 534 0586
dsh@ukie.org.uk

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	BU1	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	BU1	Understands why and when computers are used	B2, I3, A2, A4	Understands how technology has developed e.g. Moore's Law	
Components	Recognises and can use a range of input and output devices	B2, I3, A2, A4	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	

+44 (0) 207 534 0586
dsh@ukie.org.uk

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 Schoolhouse, 18a Blackbull Yard, 24-28 Hatton Wall, London EC1N 8JH.

+44 (0) 207 534 0586
dsh@ukie.org.uk

IT

	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	B2, I3, A2, A4	Uses and selects internet services, digital devices and application software to create, store and edit digital content	B2, I3, A2, A4	Evaluates the appropriateness of digital devices, internet services and application software to achieve given goals	
Audience	Understands what an audience is	B3, B4	Recognises the audience when designing and creating digital content	I4, I5	Undertakes creative projects that are tailored to meet the needs of an audience	B2, I3, A2, A4, A5
Purpose	Can talk about how they use computers	B2, I3, A2, A4	Can talk about how other people use computers	B2, I3, A2, A4	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	B3, B4	Designs and uses criteria to critically evaluate the quality of solutions	I4, I5	Documents user feedback, the improvements identified, and the refinements made to the solution	A5, A6, A7
Content	Can gather content		Makes judgements about content when evaluating and repurposing it for a given audience	B2, I3, A2, A4	Evaluates the trustworthiness of content, considers the usability of visual design features and properties of media when designing and creating digital artefacts	B2, I3, A2, A4

Digital Schoolhouse, 18a Blackbull Yard, 24-28 Hatton Wall, London EC1N 8JH.

+44 (0) 207 534 0586
dsh@ukie.org.uk

Computing Programmes of Study Links

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
2.5 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
3.4 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

+44 (0) 207 534 0586
dsh@ukie.org.uk

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

+44 (0) 207 534 0586
dsh@ukie.org.uk

Computational Thinking Strands

AL – Algorithmic Thinking

Ref.	Activity
A1	Formulating instructions to achieve a desired effect
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

+44 (0) 207 534 0586
dsh@ukie.org.uk

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)
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
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)