DIGITAL SCHOOLHOUSE together with NINTENDO SWITCH

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

Teaching Guide.

Crazy Mazes

Introduction

Developed at the Townley Grammar Digital Schoolhouse, this workshop provides pupils with an excellent foundation for programming and development. Pupils begin the day by working through the facts related to computers and using these as a starting point for discussion. This then moves pupils on to a brief discussion related to computing careers, particularly computer programmers, which is used as an entry point to discuss algorithms. Pupils develop their own algorithms for a range of scenarios, representing them in different way. Unplugged activities are used to introduce the concepts of iteration and selection as well as encouraging pupils to predict the outcome of their program.

This workshop makes use of Scratch as the programming environment; however, the skills and concepts delivered here can be implemented within any number of programming languages. While the teacher provides the pupils with ideas for a maze game, they are given the flexibility to choose their own sprites and even their own problems to make the game their own. Although, the pupils will be expected to create costumes, use selection, loops, variables and broadcasting in their game.

Pupils will undertake a wide range of evaluation activities, including using peer assessment to provide critical feedback to aid game development. The second half of the day sees pupils using the iterative development model to modify and finalise their own Crazy Maze game.

Computing Programmes of Study Links

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

Progression Pathway bands covered

P&D = Programming & Development: Pink, Yellow, Orange

	Reference
PP2	Executes, checks and changes programs
PP3	Understands that programs execute by following precise instructions
YP1	Uses arithmetic operators, if statements, and loops, within programs.
YP2	Uses logical reasoning to predict the behaviour of programs
YP3	Detects and corrects simple semantic errors i.e. debugging, in programs.
OP1	Creates programs that implement algorithms to achieve given goals.
OP2	Declares and assigns variables.
OP3	Uses post-tested loop e.g. 'until', and a sequence of selection statements in programs, including an if, then and else statement.

ALG = Algorithms: Pink, Yellow, Orange, Blue

	Reference
PA1	Understands what an algorithm is and is able to express simple linear (non-branching) algorithms symbolically.
PA2	Understands that computers need precise instructions.
PA3	Demonstrates care and precision to avoid errors

YA1	Understands that algorithms are implemented on digital devices as programs
YA2	Designs simple algorithms using loops, and selection i.e. if statements.
YA3	Uses logical reasoning to predict outcomes.
YA4	Detects and corrects errors i.e. debugging in algorithms
OA1	Designs solutions (algorithms) that use repetition and two-way selection i.e. if, then and else.
OA2	Uses diagrams to express solutions.
OA3	Uses logical reasoning to predict outputs, showing an awareness of inputs.
BA3	Recognises that different solutions exist for the same problem.

IT = Information Technology: Pink, Yellow, Orange, Blue

Reference

PP2	Executes, checks and changes programs
PP3	Understands that programs execute by following precise instructions
YP1	Uses arithmetic operators, if statements, and loops, within programs.

Computational Thinking Strands

AL – Algorithmic Thinking

Ref.	Activity
AL1	Writing instructions that if followed in a given order (sequences) achieve a desired effect
AL3	Writing instructions that store, move and manipulate data to achieve a desired effect; (variables and assignment)
AL4	Writing instructions that choose between different constituent instructions (selection) to achieve a desired effect;
AL5	Writing instructions that repeat groups of constituent instructions (loops/iteration) to achieve a desired effect;

AB – Abstraction

Ref.	Activity
Ab1	Reducing complexity by removing unnecessary detail;
Ab2	Choosing a way to represent artefacts (whether objects, problems, processes or systems) to allow it to be manipulated in useful ways;

EV – Evaluation

Ref.	Activity
EV1	Assessing that an algorithm is fit for purpose;
EV2	Assessing whether an algorithm does the right thing (functional correctness);
EV3	Designing and running test plans and interpreting the results (testing);
EV8	Assessment of whether a system gives an appropriately positive experience when used (user experience);
EV10	Stepping through algorithms/code step by step to work out what they do (dry run / tracing);
EV13	Using methods involving observing a system in use to assess its usability or performance (empirical evaluation)

EV16	Assessing whether a product meets general performance criteria (heuristics)

GE – Generalisation

Ref.	Activity
GE1	Identifying patterns and commonalities in problems, processes, solutions, or data.
GE2	Transferring ideas and solutions from one problem area to another

Learning Outcomes

1. To be able to understand some facts about computers
2. To be able to understand the importance of correct instructions
3. To understand what an algorithm is
4. To be able to represent an algorithm in pseudocode
5. To be able to evaluate the effectiveness of an algorithm
6. To be able to implement a pre-written algorithm within a programming language such as Scratch
7. To be able to edit and import costumes into Scratch
8. To be able to edit and import backgrounds into Scratch
9. To understand what is meant by the term variable
10. To be able to create and use variables within their program
11. To understand what is meant by the term 'selection' and 'loop'.
12. To be able to use selection and loop statements within their algorithm
13. To be able to use selection and loop statements within their program
14. To be able to dry run/test their algorithm to see if it works and achieves desired results
15. To be able to implement their algorithm as a program
16. To be able to evaluate the effectiveness of their solution
17. To be able to use peer feedback to evaluate the effectiveness of their solution
18. To be able to identify areas for improvement and modification within their program
19. To be able to implement changes to their program, based on feedback and evaluation.

Session Overview

SESSION 1

Session Content / Activity	Resources Used	Prog. Pathway	Comp. Thinking	Computing POS Link
Welcome and Introduction	DSH_WelcomeIntroduction.ppt			
Starter Activity: 'What do I know about computers', collect a range of answers from the pupils for this question. Once they have provided their thoughts about computers, hand out the laminated facts for the pupils to read out to the rest of the class. Emphasise that a computer is a machine that carries out instructions. Ask the question 'What is a programmer?' and tell the pupils that there will be a short video and to watch out for the snippets of information on slide 3. Emphasise that a computer programmer is a person who writes or fixes computer programs. Instruct the pupils to draw a picture from the instructions on slide 7, a	Computing Facts.docx Crazymazes.pptx	ALG PA1, PA2	AL1	2.3

discussion on why it wasn't easy and an explanation of why.				
Explain to the pupils about what an algorithm is and show the flow-chart activity, where they would need to volunteer answers to position instructions onto the flow-chart to draw a square. Discuss the sequential instructions and they could be made more efficient. Show slide 14 to emphasise how programmers like to have less steps using 'repeat'.	Crazymazes.pptx	ALG PA1 – PA3, YA1- YA4, OA1 – OA4	AL1, AL5 and EV13	2.1, 2.2 and 2.3

Pupils try again in pairs, one pupil has a picture and writes instructions to explain to their partner how to draw the picture on slide, after pupils show the picture to the rest of the class, if time swap and produce another image to each pair for the other pupil to explain how it's drawn. Pupils then are asked to feedback to the rest of the class about their peer's instructions.	Mini-Whiteboards Laminated slides 12 and 13 for each pair	ALG PA1 – PA3, YA1- YA4, OA1 – OA4	AL1, AL5 and EV13	2.1, 2.2 and 2.3
Pupils are instructed to log onto Blockly to practise dropping blocks to instruct a little figure to move around a maze. Pupils are encouraged to feedback on 'How easy was this?' and 'What problems did you have?'	Crazymazes.pptx https://blockly-games.appspot.com/maze?lang=en	ALG PA2-3 P&D PP1-3	EV1, EV2	2.1, 2.2 and 2.3
Instruct the students to play the maze game for a couple of minutes. Ask them about the points on slide 19.	http://www.knowledgeadventure.com/games/a-maze-race/	**ALG** YA3, OA3	AL1	2.3

SESSION 2

Session Content / Activity	Resources Used	Prog. Pathway	Comp. Thinking	Computing POS Link
Recap key aspects from session 1, if time each pupil could write a word they have learnt on the whiteboard. Show/explain the purpose of Scratch and the interface e.g. stage etc. Explain the task to the pupils, that they will be making a game with a maze and a sprite, show the final version of the spider maze game. Review the maze game the pupils played in session 1 and ask them to volunteer answers about how the game should work. Write some instructions in pairs, which can be fed back to the rest of the class. (slide 24)	Crazymazes.pptx Spider Maze Broadcast.sb Whiteboards Scratch	ALG PA1, PA2 YA1, YA2	AL1, AL2	2.3
The teacher tells the class that they will need to draw a sprite and they will need to think of a sprite they would like to have in their game. It does not have to be a spider, but would be best if it was a sprite that looks like it is being viewed from above. (bird's eye view). Teacher demonstrates how to draw a spider and adds to the stage. Teacher circulates to support students who are drawing their sprite. (slide 25)	Crazymazes.pptx Scratch	P&D PP1, PP2	EV16	2.3

The teacher explains that they would need to create costumes for their sprite and what costumes are when used in Scratch. Teacher demonstrates how to make additional costumes for the spider pointing up, down, left and right. These must also be named appropriately, together with the sprite. The teacher then circulates to support students who are drawing their sprite. (slide 26)	Crazymazes.pptx	P&D PP1, PP2	EV16	2.1
The teacher explains that they will shortly be interacting with the sprite to move it using the arrow keys, but before this he/she explains that stage and positioning in scratch using the x/y co-ordinates. The x/y grid background is imported to help this explanation. (slides 27-30)	Crazymazes.pptx			
The teacher explains that they will be learning how to move the sprite using the arrow keys and asks the group about the instruction blocks that would be needed (slide 31). The teacher demonstrates and circulates to check understanding.	Crazymazes.pptx	P&D PP2, PP3, YP1, YP2, YP3, OP1	AL1, EV1, EV2	2.1
The teacher asks the group 'What could be improved?' pupils should realise that the sprite should look like it's actually switching position when the arrows are selected. (slide 32) The teacher	Crazymazes.pptx Spider Maze Moving.sb	P&D PP2, PP3, YP1, YP2, YP3, OP1	AL1, EV1, EV2	2.1

demonstrates and circulates to check understanding, pupils will be testing their program.				
The teacher asks the group 'What would improve this program?' Students will volunteer some answers that can be written on the board. One of these may be to have a maze on the background, the teacher demonstrates and circulates to check understanding (Slide 33).	Crazymazes.pptx	P&D PP2, PP3, YP1, YP2, YP3, OP1	EV16	2.1
Teacher explains the concept 'Selection' and that it's helpful when there is decision somewhere in the program. Teacher provides some examples for this concept e.g. If it rains take umbrella Else take sunhat to encourage ideas from the pupils. Teacher provides an example of the sprite touching the side of the maze and returning to the start of the maze. The teacher demonstrates this example and circulates to support pupils. (slides 35-37) Pupils must test and run program but realise that it won't work. The teacher asks the pupils why, which moves the lesson onto repeat and loops.	Crazymazes.pptx	P&D PP2, PP3, YP1, YP2, YP3, OP1, OP2, OP3	EV1, EV2, EV3, AL1, AL2, AL3, AL4, AL5	2.1-3
The teacher then teaches the concept of loops/repeat (slides 39-43). The teacher then shows how a simple loop can help to make the selection	Crazymazes.pptx	P&D PP2, PP3, YP1, YP2,	EV1, EV2, EV3, AL1, AL2, AL3, AL4, AL5	2.1, 2.2, 2.3

blocks work for the spider to return to the start of the maze when it touches the sides of the maze.		YP3, OP1, OP2, OP3		

SESSION 3

Session Content / Activity	Resources Used	Prog. Pathway	Comp. Thinking	Computing POS Link
Recap key aspects from session 2, the teacher hands out the peer assessment sheet. After the pupils write their names on it they should swap seats with their neighbour, play their game and rate parts of the game listed from 1-4. The pupils must read their feedback and feed back to the rest of the class about the improvements they will need to make. (slide 46).	Crazymazes.pptx Game Feedback Worksheet	P&D PP2, PP3, YP1, YP2, YP3	EV1, EV2, EV3	2.1, 2.2, 2.3
The teacher asks the group about the types of sprites that could interact with the main sprite and she/he provides the example of gold coins, although pupils might want to make up their own idea of additional sprites. The teacher shows how to draw this in the paint editor and adds to the stage.	Crazymazes.pptx	P&D PP1, PP2	EV16	2.1
Again it is up to the pupil how they might want the main sprite to interact with the additional ones and	Crazymazes.pptx	P&D PP2, PP3, YP1, YP2,	EV1, EV2, EV3, AL1, AL2, AL3, AL4, AL5	2.1, 2.2. 2.3

the teacher asks the group about their ideas, which can be written on the main white-board. The teacher provides the example shown on slide 49, where there is a score which increases when the spider collects the coin. First the teacher encourages the pupils to think back on the code used before lunch where the spider needed to return to the beginning of the maze if he touched the sides of the maze – the teacher asks what do we want to happen? Use the mini whiteboards to write the instructions in pairs. Teacher then demonstrates how to use hide/show initially (slide 49).		YP3, OP1, OP2, OP3 ALG PA1, PA2, PA3 YA1, YA2, YA3, YA4, OA1, OA2, OA3		
The teacher then asks the question 'How would we then incorporate a scoring system into the game'? Pupils must then edit their instructions in pairs to include the scoring system and feedback to the rest of the class. (slide 50).	Crazymazes.pptx Mini-Whiteboards	ALG PA2, PA3 YA2	AL1, AL2, AL3	2.3
The teacher then explains the concept of variables (slide 51) and demonstrates how to add a variable to their game, circulates and checks progress when pupils are adding/testing their code.	Spider Maze Score.sb Crazymazes.pptx	P&D PP2, PP3, YP1, YP2, YP3,	EV1, EV2, EV3, AL1, AL2, AL3, AL4, AL5	2.1, 2.2. 2.3
The teacher instructs students to swap seats with their neighbour, play their game and additional parts	Game Feedback	P&D PP1, PP2, PP3	EV1, EV2, EV3	2.1, 2.2. 2.3

of the game listed from 1-4. The pupils must read their feedback and feed back to the rest of the class about the improvements they will need to make. (slide 54).	Worksheet	**IT** PI5, YI5, OI3, BI4		
The teacher tells the pupils that will only be ready to duplicate the sprites to make more challenges throughout the maze if their code for the first additional sprite is working perfectly. The teacher demonstrates how to do this and circulates to check progress (slide 55).	Crazymazes.pptx	P&D PP1-3 YP1-3	EV6	2.1, 2.2. 2.3
The teacher asks the group about making the game even more challenging and asks the pupils to discuss in pairs and feedback to the rest of the group. Some can then write their ideas on the board. The teacher shows the example of the spider collecting a golden key, which will mean that the game has reached the next level. The teacher asks students to think about the instructions for this step and pupils work this out in pairs and feedback to the group. The teacher then demonstrates how to add and edit backgrounds (slides 57/8) and circulates to check progress. Once the pupils have made at least one extra background, the teacher can demonstrate how to use broadcast to create a new level using the	Spider Maze Broadcasting.sb Mini-whiteboards	ALG PA1-3 YA1-4 IT PI5, YI5, OI3, BI4 P&D PP2, PP3, YP1, YP2, YP3, OP1, OP2, OP3	EV1, EV2, EV3, AL1, AL2, AL3, AL4, AL5	2.1, 2.2. 2.3

golden key example. He/she then checks progress around the group.				
The teacher introduces the concept of procedures to the group now that they have code being triggered in several parts of their program (slide 62). The teacher instructs students to swap seats with their neighbour, play their game and additional parts of the game listed from 1-4. The pupils must read their feedback and feed back to the rest of the class about the improvements they will need to make. (slide 62).	Game Feedback Worksheet	P&D PP1, PP2, PP3 **IT** PI5, YI5, OI3, BI4	EV1, EV2, EV3	2.1, 2.2. 2.3
The teacher asks the question 'How your game can be improved?' and pupils volunteer answers, which can be written on the main whiteboard. Pupils use their feedback and ideas for improvement to work on their game for the rest of the session, but the teacher should leave time for the plenary quiz on slide 65, which will be the end of the session.	Crazymazes.pptx	**P&D** PP1, PP2, PP3, YP1, YP2, YP3 **IT** PI5, YI5, OI3, BI4	EV1, EV2, EV6, EV7	2.1, 2.2. 2.3

Files/Resources

Filename	Resource Type	Purpose/Description
Crazy Mazes	Power-point	Main teaching PowerPoint
Game Feedback Worksheet	Word document	A worksheet to be used by the pupils
code-4-students	Video	Introductory video to help pupils to understand what a computer programmer does
Computing Facts	Word document	A worksheet that is to be divided up into cards and laminated to use as a starter
Spider Maze Moving	Scratch program	Program to display on the board and/or help the teacher to understand the program so far
Spider Maze Selection	Scratch program	Program to display on the board and/or help the teacher to understand the program so far
Spider Maze Score	Scratch program	Program to display on the board and/or help the teacher to understand the program so far
Spider Maze Broadcasting	Scratch program	Program to display on the board and/or help the teacher to understand the program so far
MiniWhiteboard.pdf (.docx)	Photocopiable	To be printed and laminated for use as a mini whiteboard

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.