

# Teacher Research Projects in Computing

## Session 3

CAS Conference

13th July 2024

Sue Sentance, Rachael Coultart, Pete Dring, Justin  
Heath and Jo Hodge



Teacher Research Projects booklet available at <http://bit.ly/tice2-booklet>

# Teacher research in computing

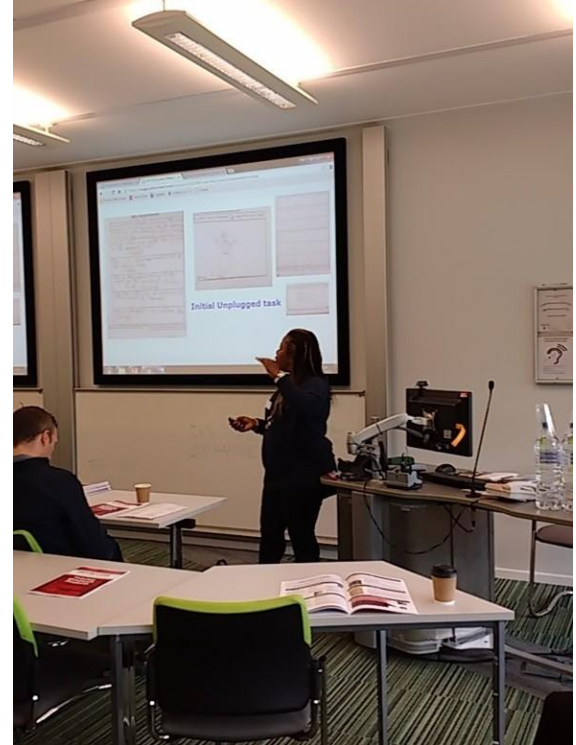
As teachers, it can be transformative to your practice to investigate what might improve your practice.

In computing in school there's even more reason to do this as we've all been learning how to teach it and there isn't a huge amount of research.

However there are barriers: time, confidence, expertise ....

In this project, we wanted to facilitate small-scale computing education inquiry projects, through real partnerships between teachers and academic helpers over a period of months.

**The project is called Teacher Inquiry in Computing Education (TICE).**



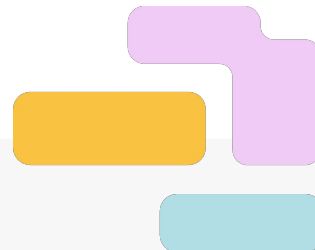
## How TICE works

- Teachers volunteer/apply and construct a research question that is relevant to their own context and practice
- Introduced to a 'helper' who has research experience to be a buddy and offer support
- Webinars/workshops introduce the various stages of a research project
- Teachers write up project using a template



## In this session

- **Rachael Coultart** - Impact of using LitterBox to support debugging skills
- **Pete Dring** - Impact of feedback of students' engagement and progress in KS4 Computer Science. Can AI replace a good teacher?
- **Justin Heath** - FLARE - Deepening understanding of abstraction in block-based languages: A framework for learning about relational elements
- **Jo Hodge** - Which PRIMM activities lend themselves to assessment in two Year 4 classes over the 6-week Spring Term in a primary programming unit



# Rachael Coultart





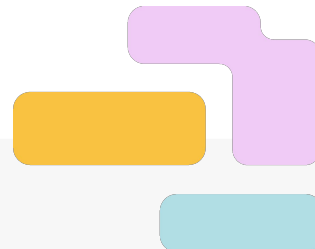
UNIVERSITY OF  
CAMBRIDGE



Raspberry Pi  
Foundation



RASPBERRY PI  
COMPUTING EDUCATION  
RESEARCH CENTRE



# The Impact of using Litterbox to improve debugging skills

TICE2 Research Project Spring 2024

Rachael Coultart supported by Dr Bobby Whyte



# WiPSCE 2023



Codemuster

Bugs	Smells	Perfumes
Ambiguous Custom Block Signature		Missing Clone Call
Ambiguous Parameter Name Used		Missing Clone Initialization
Blocking If-Else		Missing Erase All
Call Without Definition		Missing Initialization
Comparing Literals		Missing Loop
Custom Block With Error		Missing For Loop

In what ways does the use of Litterbox support Year 5 and 6 Code Clubbers to improve their

- perseverance,
- code comprehension and
- debugging skills

in their Scratch projects over 8 weeks in the Spring term?

# I decided to run an intervention...

Model using Litterbox and support learners to interpret the results it produces when they run their projects through it. Encourage them to use the results to improve their projects and talk to them about what they did, how Litterbox helped or hindered.

1. Look at their Scratch accounts
2. [Code Comprehension quiz](#) based on the [‘More Scratch pathway’](#) as baseline
3. Complete projects and use Litterbox to debug and analyse
4. Code Comprehension quiz repeated

Code Comprehension Quiz for TICE2

These questions are taken from the Code Club 'More Scratch' pathway to assess learners' code comprehension skills before and after an intervention using Litterbox.

1. If a button sprite had this code on it, which of the following scripts would hide a sprite when the button is clicked?

when this sprite clicked  
hide sprite

100% Invisible

Option 1  
 Option 2  
 Option 3  
 Option 4

2. What would need to happen for the 'say' script to hide if it had this code?

when the green flag is clicked  
say Hello for 2 secs  
hide sprite

The 'say' script would need to touch the yellow flag.  
 The 'say' script would need to be touching the yellow flag.  
 The 'say' script would need to be visible.  
 The 'say' script would need to be hidden.  
 The 'say' script would need to be visible when the green flag is clicked.  
 The 'say' script would need to be visible when the yellow flag is clicked.

3. If you used a 'say random' block to make an object move in a natural, unpredictable pattern, which 'say random' block could give the number 10?

when the green flag is clicked  
say random 10 100

Option 1  
 Option 2  
 Option 3

4. You write some code to make the doggy move only if it is not touching the mouse pointer. Where would you put a 'start sound' block to make the doggy start a sound each time it moves?

when the green flag is clicked  
play sound

Option 1  
 Option 2  
 Option 3

Code Comprehension Quiz for TICE2

These questions are taken from the Code Club 'More Scratch' pathway to assess learners' code comprehension skills before and after an intervention using Litterbox.

1. If a button sprite had this code on it, which of the following scripts would hide a sprite when the button is clicked?

when this sprite clicked  
hide sprite

100% Invisible

2. What would need to happen for the 'say' script to hide if it had this code?

when the green flag is clicked  
say Hello for 2 secs  
hide sprite

The 'say' script would need to touch the yellow flag.  
 The 'say' script would need to be touching the yellow flag.  
 The 'say' script would need to be visible.  
 The 'say' script would need to be hidden.  
 The 'say' script would need to be visible when the green flag is clicked.  
 The 'say' script would need to be visible when the yellow flag is clicked.

3. If you used a 'say random' block to make an object move in a natural, unpredictable pattern, which 'say random' block could give the number 10?

when the green flag is clicked  
say random 10 100

Option 1  
 Option 2  
 Option 3

4. You write some code to make the doggy move only if it is not touching the mouse pointer. Where would you put a 'start sound' block to make the doggy start a sound each time it moves?

when the green flag is clicked  
play sound

Option 1  
 Option 2  
 Option 3

5. Which of these values can be stored in a variable?

when the green flag is clicked  
say Hello for 2 secs  
hide sprite

All the numbers in the list.  
 Only the first number. You can't store the other numbers in a 'text' variable.  
 Only the number in the first position.  
 None of them.

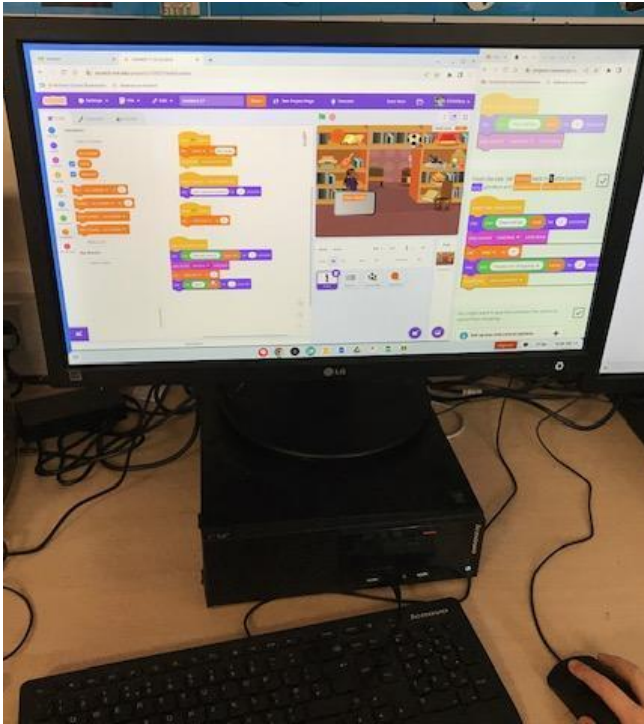
6. What will be the value of the 'name' variable after the player clicks on the rock and the script finishes?

when the green flag is clicked  
say Hello for 2 secs  
hide sprite

name  
 Hello  
 2  
 100

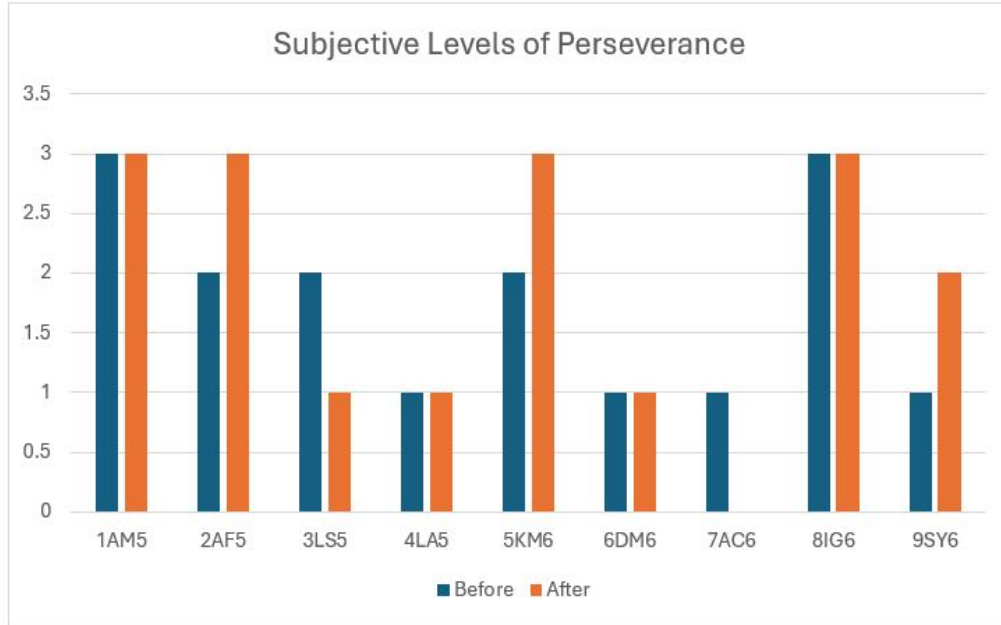


# Hiccups



- Delayed start
- Clash with football match and then SATs Booster classes
- Using the Code Club More Scratch pathway may not have been the best idea
- Interpreting the hints was an issue for some
- It's very difficult to measure perseverance

# Perseverance Results



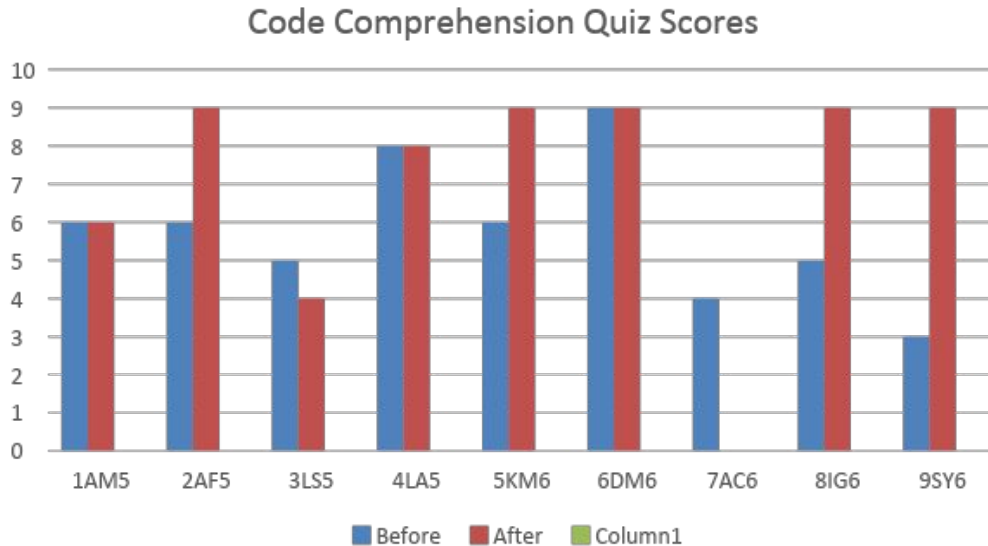
How do you feel when your program doesn't work the way you want it to?

1 = low ie frustrated with no suggestion of strategies for debugging

2 = medium ie some suggestion that they might be able to debug and get project working

3 = high ie expressed confidence at being able to debug and had range of strategies

# Code Comprehension Results



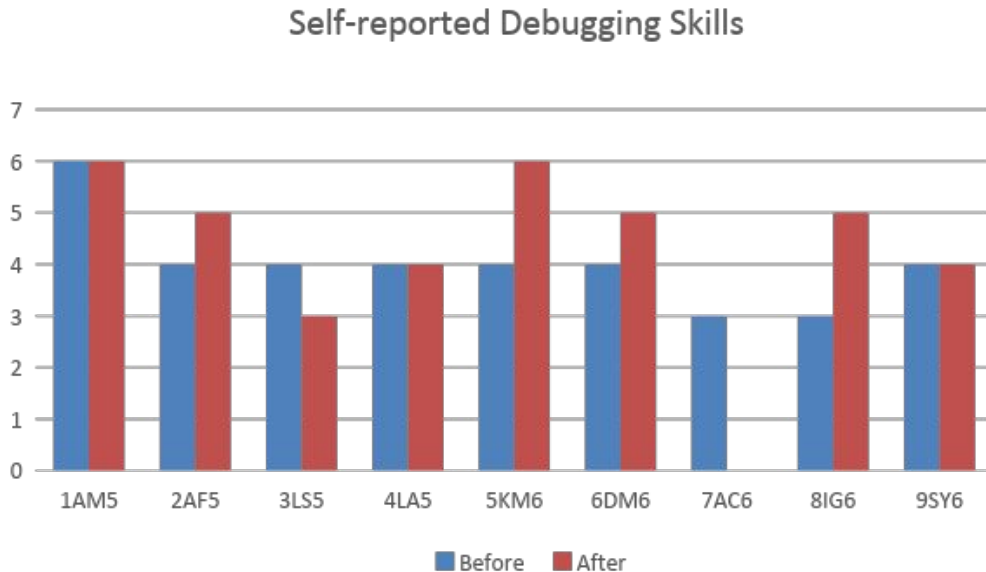
Key concepts made clearer through discussions

Conversations were meaningful and productive

“Litterbox helped me with a useless block.”

Significant improvement in students 8 and 9

# Self-Reported Debugging Skills Results



How do you rate your debugging skills on a scale of 1 to 6?

Litterbox was initially very motivating, but this wasn't sustained for all

Hints needed unpicking

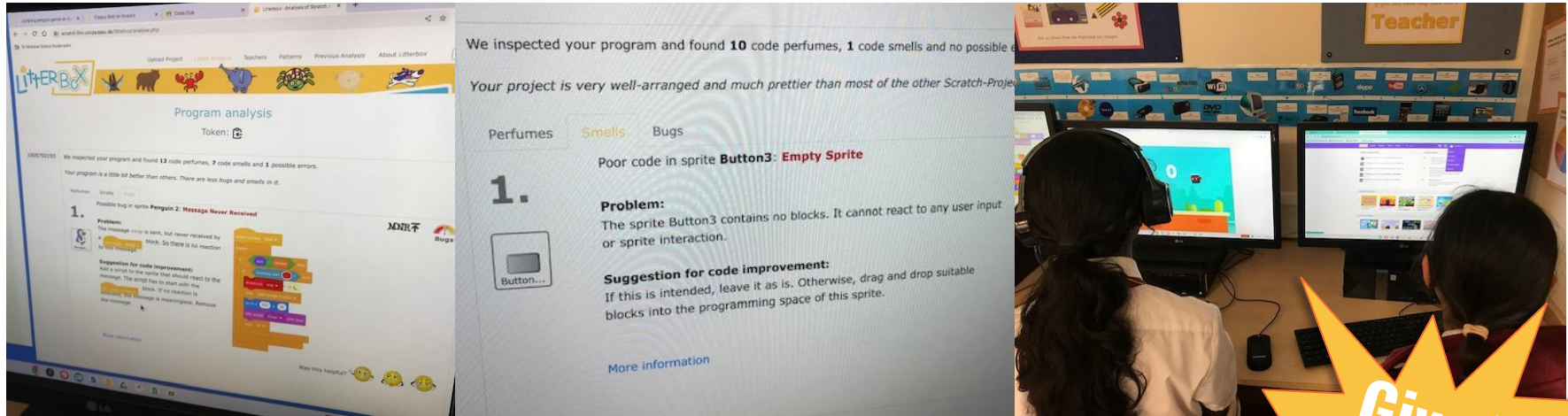
Second analysis was the same!

# Highlights



- Sprites were named appropriately and initialisation improved dramatically
- Conversations about projects were more meaningful and demonstrated improved understanding
- Empowered students to improve own projects
- My improved confidence in supporting debugging
- Didn't want to stop and go home!

# Conclusions



- Fantastic tool for teachers to help analyse projects and provide constructive feedback
- Needs careful introduction to students to ensure remains motivational
- Could potentially be improved with the addition of video tutorials

Give it  
a go!



RASPBERRY PI

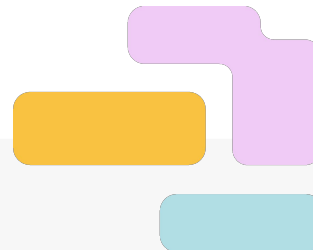
COMPUTING EDUCATION

RESEARCH CENTRE



*Thank you*


To talk more about this, feel free to email me: [rachael@coultart.com](mailto:rachael@coultart.com)



# Pete Dring







# Impact of feedback on students' engagement and progress in KS4 Computer Science

Can AI replace a good programming teacher?

Pete Dring, Fulford School, York



# Context and starting point

- Edexcel GCSE Computer Science
- Practical programming paper 2



The screenshot shows the Education Endowment Foundation website interface. At the top left is the logo for the Education Endowment Foundation. At the top right is a dropdown menu labeled 'Evidence and resources'. Below the logo, the breadcrumb navigation reads 'Evidence and resources > Teaching and Learning Toolkit > Feedback'. A toggle switch for 'View in Cymraeg' is visible. The main heading is 'Feedback', followed by the text 'Very high impact for very low cost based on extensive evidence'. Below this, three metrics are displayed: 'Implementation cost' with five pound symbols (£) in circles, 'Evidence strength' with five padlock icons in circles, and 'Impact (months)' with a red circle containing '+6' and the word 'months' next to it.

Source: [educationendowmentfoundation.org.uk](https://educationendowmentfoundation.org.uk)

# Research questions

What type of feedback has the most positive impact on students'...

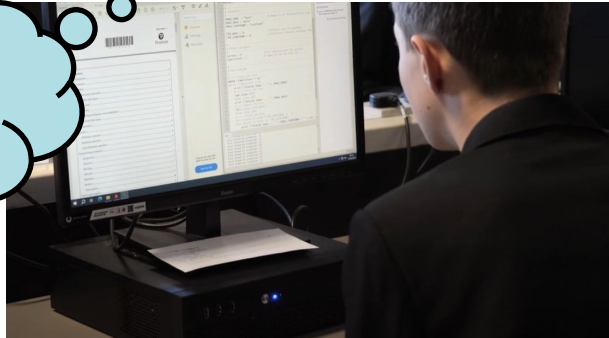
2

... ability to  
improve ...

1

... engagement  
...

I know  
what to  
do



I'm going  
to do it

...in the programming aspect of GCSE Computer Science?

# The process

## Initial survey

- Jan 24
- n = 43



## Intervention

- Feb 24
- n = 36



## Final survey

- Mar 24
- n = 15

2. How useful are each of these types of feedback for improving your motivation (making you want to work hard and succeed in Computer Science)?

	Not useful at all	Not very useful	Quite useful	Very useful
1: a target from the learning objectives you said you were least confident at	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
2: a summary showing you which learning objective you struggled with in a quiz	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Pearson Edexcel Level 1/Level 2 GCSE (9-1)  
 Specimen Assessment Material for first teaching  
 September 2020

Time: 1 hour 30 minutes  
 Total Marks: 100 (2:52)

**Computer Science**  
 Paper 2: Application of Computational Thinking

**The most basic:**  
 1. A simple introduction with appropriate programming language code  
 which defines the problem, but nothing else that you will be able to  
 do. It is a simple introduction to the problem and does not  
 contain any code that you will be able to do.

**Instructions**

- Answer all the questions on these questions.
- Start on a new sheet of paper for each question and place it in the correct place in the booklet.
- You must not use the Internet at any time during the examination.

**Information**

- The 100 MARKS (2:52) taken in your own area includes all the code and data that you need.
- The total mark for this paper is 100.
- The marks for each question are shown in brackets.

**Advice**

- Read each question carefully before you start to answer it.
- Start your work regularly.
- Check your answers are correct if you have time at the end.

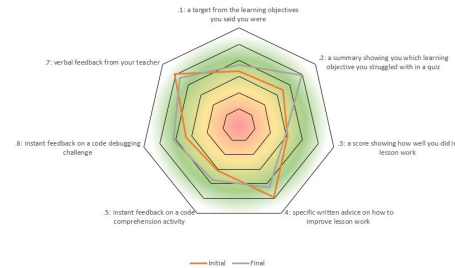
568623A



Non-exam



How useful are each of these types of feedback for improving your motivation (making you want to work hard and succeed in Computer Science)



# Types of feedback

1: a target from the learning objectives you said you were least confident at

6.3.1	be able to write programs that make appropriate use of primitive data types (integer, real, Boolean, char) and one and two-dimensional structured data types (string, array, record)	5	
6.3.2	be able to write programs that make appropriate use of variables and constants	3	
6.3.3	be able to write programs that manipulate strings (length, position, substrings, case conversion)	3	
6.4.2	be able to write programs that read from and write to comma separated value text files	4	
6.6.2	be able to write functions that may or may not take parameters but must return values, and procedures that may or may not take parameters but do not return values	3	

Suggested target:



6.4.2 be able to write programs that read from and write to comma separated value text files



2

... ability to improve ...

1

... engagement

...

# Types of feedback

2: a summary showing you which learning objective you struggled with in a quiz

Quiz navigation

1	2	3	4	5
✓	✓	○	✓	✓

```
def guess_my_password():  
    print("Well done!")  
    print("Here's a secret for you")  
    print(SECRET)
```

### 6: Programming

#### Learning Objective 6.1.2

be able to read, write, analyse and refine programs written in a high-level programming language

low level	✗	:type of programming language where a small amount of code can solve large problems by hiding unnecessary details about how the CPU and memory are managed
high level	✗	:type of programming language which can require a large amount of code to solve even small problems because you have complete control over the CPU and memory management
programming language	✓	:a way of describing data structures and algorithms to tell a computer how to solve a problem
assembly	✗	:example of a high level programming language
python	✗	:example of a low level programming language

2 ... ability to improve ...

1 ... engagement ...

# Types of feedback

3: a score showing how well you did in lesson work

- Y10-01-CT1\_Activities.docx
- Y11\_01\_CT2\_Activities.docx
- Y11\_01\_CT3\_Activities.docx



	Attempted	Good	Awsome		
	1	2	3	Excellent. Please be careful with filenames - you've uploaded the right file but it's called P2 instead of CT1 - could you rename it on the VLE please?	
point	1	2	3		
s	s	s	s		
CT2	not submitted	Attempted	Good	Awsome	Well done
0	1	2	3		
point	point	point	point		
s	s	s	s		
CT3	not submitted	Attempted	Good	Awsome	Please catch up when you can Update: good start. To set a breakpoint in Thonny, double click on the line number. You'll see a red blob - that means that the program will pause when it gets to that line (if you press debug rather than run)
0	1	2	3		
point	point	point	point		
s	s	s	s		

2

... ability to improve ...

1

... engagement ...

# Types of feedback

4: specific written advice on how to improve lesson work

- Y10-01-CT1\_Activities.docx
- Y11\_01\_CT2\_Activities.docx
- Y11\_01\_CT3\_Activities.docx



	Not submitted	Attempted	Good	Awesome	
CT1	0 point s	1 point s	2 point s	3 point s	Excellent. Please be careful with filenames - you've uploaded the right file but it's called P0 instead of CT1 - could you rename it on the VLE please?
CT2	0 point s	1 point s	2 point s	3 point s	Well done
CT3	0 point s	1 point s	2 point s	3 point s	Please catch up when you can update good start. To set a breakpoint in Theory, double click on the line number. You'll see a red box - that means that the program will pause when it gets to that line if you press debug rather than run.

2

... ability to improve ...

1

... engagement ...




# Types of feedback

5: instant feedback on a code comprehension activity

Keywords

The code snippet below contains each of these key python programming concepts. Can you identify which line number they're on?  
Being able to understand and use the right words to describe the code you're looking at helps you know what to search for and how to ask for help when you're writing your code

Progress:  Score: 17%

2/12: Assignment

👏 setting the value of a variable  
Well done!

Line number:

```
1 # Lists Try it Python Challenges
2
3 shopping = ["bananas", "oranges", "chocolate", "toilet roll"]
4
```

2

... ability to improve ...

1

... engagement ...

# Types of feedback

6: instant feedback on a code debugging challenge

The screenshot shows a code editor with Python code for a shopping list challenge. The code includes a list of items, print statements, and a list of challenges. A 'Tests' window is overlaid on the code, showing a table of test results for 'Testing mycode.py'.

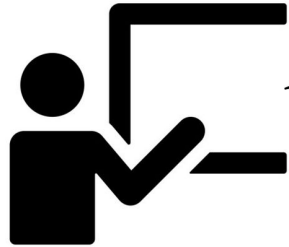
Test	Description	Result
1	Add 'socks' to the shopping list	Pass
2	Remove 'oranges' from the list	Fail
3	Add items to the list until there are exactly 10 items	Fail

2 ... ability to improve ...

1 ... engagement ...

# Types of feedback

7: verbal feedback from your teacher



2

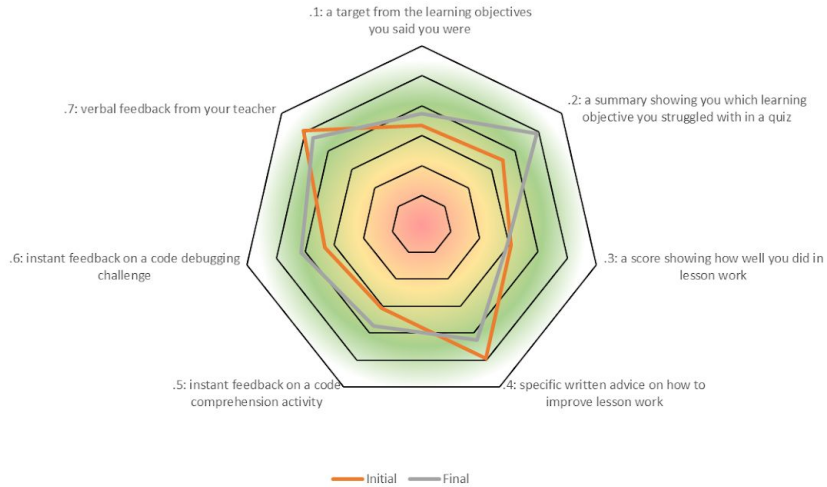
... ability to improve ...

1

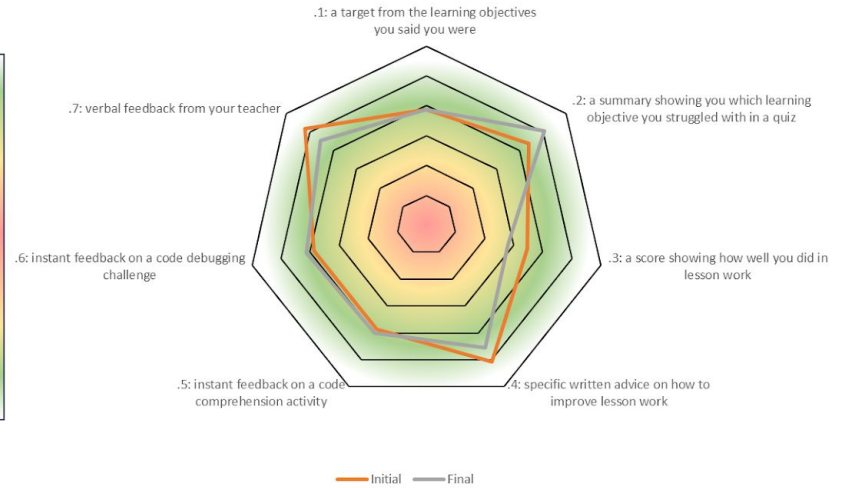
... engagement ...

# Findings

How useful are each of these types of feedback for improving your **motivation** (making you want to work hard and succeed in Computer Science)

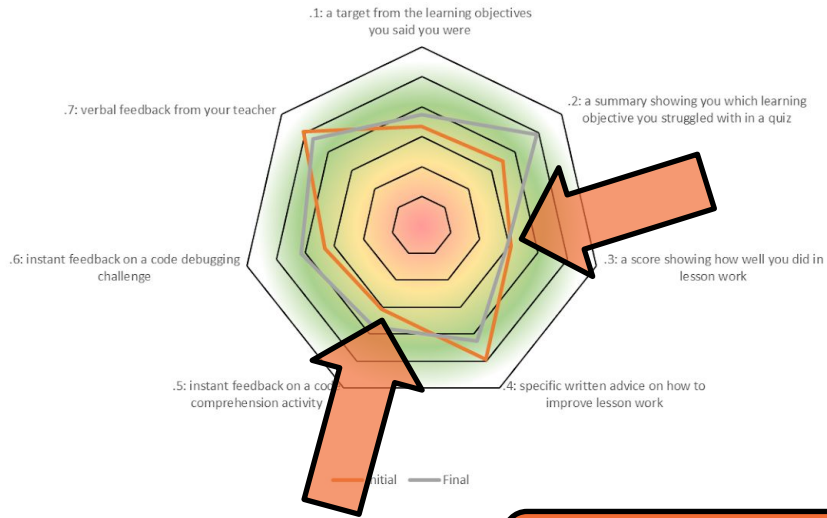


How useful are each of these types of feedback for helping you **improve your understanding** and grades in Computer Science?

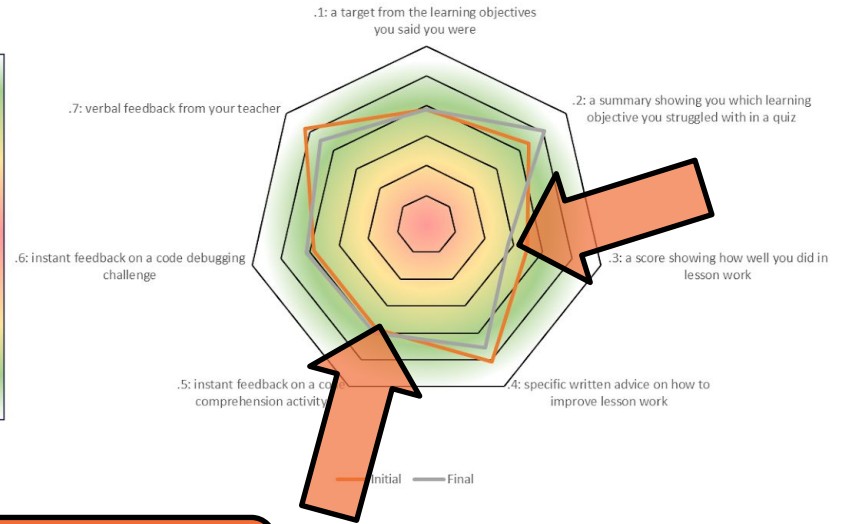


# Findings: least useful

How useful are each of these types of feedback for improving your **motivation** (making you want to work hard and succeed in Computer Science)



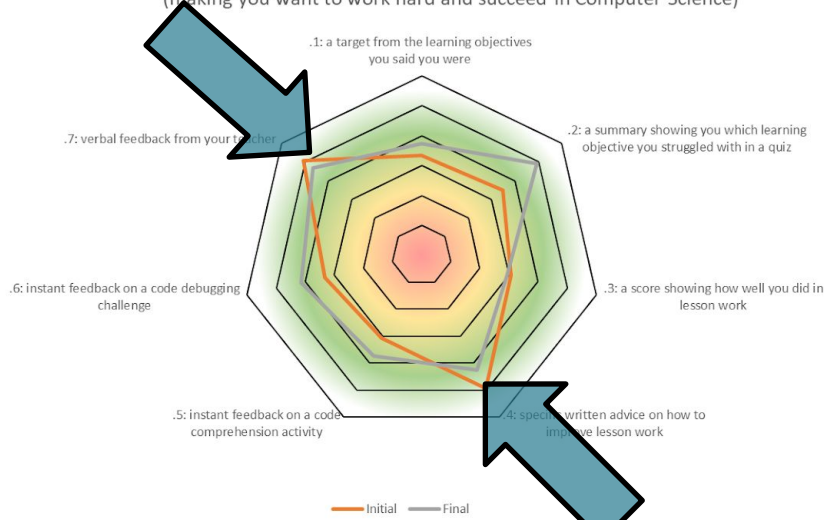
How useful are each of these types of feedback for helping you **improve your understanding** and grades in Computer Science?



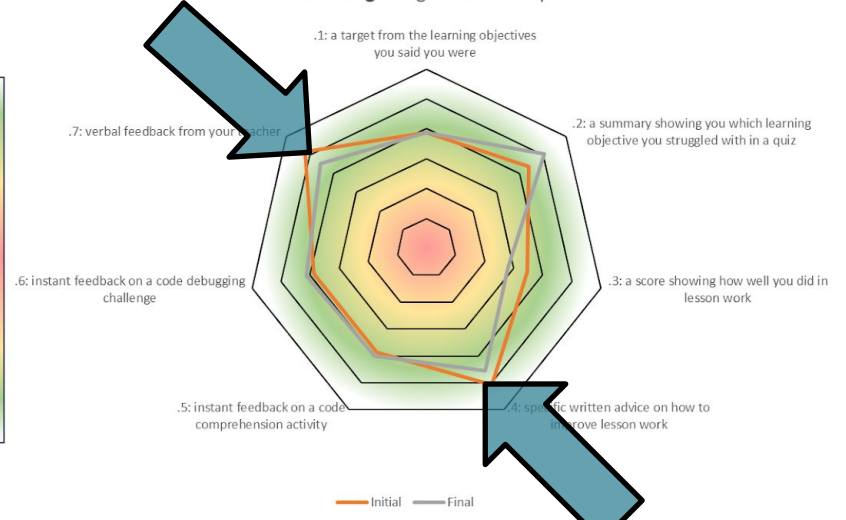
Instant feedback with score

# Findings: most useful

How useful are each of these types of feedback for improving your **motivation** (making you want to work hard and succeed in Computer Science)



How useful are each of these types of feedback for helping you **improve your understanding** and grades in Computer Science?



Specific written or verbal feedback

# Conclusion and next steps



Personal



Automated



2

... ability to improve ...

1

... engagement

...

## Y11 Computer Science Paper 2 Mock Feedback and DIRT

[STUDENT NAME] 10A/Co1

Mr Dring

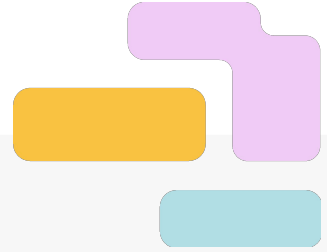
FFT: 5=

Y10 Paper 2 mock grade:

7-

11 marks above FFT

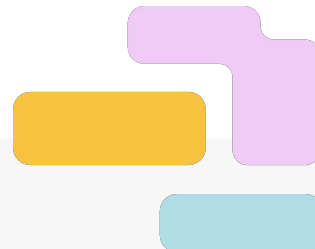
Q	Marks	Notes & Feedback		
Q1	9 / 10	1.1	• Complete creation using '[,]' and any 9 integers between 0 and 100 (1)	0
		1.2	• Provide text and missing double quotes (1)	1
		1.3	• Indent original line to be inside subprogram (1)	1
		1.4	• Add function call <code>len()</code> around 'mySentence'	1
		1.5	• Add colon on end of while loop (1)	1
		1.6	• Add keyword 'if' to front of line (1)	1
		1.7	• Add missing brackets around variable 'location' (1)	1
		1.8	• Amend call to subprogram to fix <code>NameError</code> (1)	1
		1.9	• Add type conversion to fix <code>TypeError</code> (1)	1
		1.10	• Change <code>-</code> operator to a <code>+</code> operator to fix infinite loop (1)	1
Q2	8 / 10	2.1	• import time (1)	1
		2.2	• while ( <code>timesToTest != 0</code> ): (1)	1
		2.3	• <code>time.sleep</code> (15) for 15 seconds (1)	1
		2.4	• <code>time.sleep</code> (TIME_WALK) use of constant over hard-coded, which will not meet requirement of 10 seconds (1)	1
		2.5	• for seconds in range (TIME_COUNTDOWN, 0, -1): to count backwards (1)	1
		2.6	• <code>time.sleep</code> (1) for counting down 1 second (1)	1
		2.7	• <code>timesToTest = timesToTest - 1</code> to count down tests for three times (1)	0
		2.8	• Functionality (3)	2
Q3	6 / 10	3.1	• Variable 'name' set to "" (1)	1
		3.2	• Get value of 'name' from user using 'input' (1)	1
		3.3	• At least one 'if' statement used (1)	1
		3.4	• Presence check using <code>len(name) == 0</code> (1)	1
		3.5	• Length check using <code>'&lt; 3'</code> (1)	0
		3.6	• Length check using <code>'&gt; 20'</code> (1)	0
		3.7	• Use of <code>if...elif...else</code> rather than separate 'if' (1)	1
		3.8	• Presence check identified in comments and at least one length check identified in comments (1)	0
		3.9	• Both value of 'name' and 'All checks passed' printed on same line (1)	1
		3.10	• Functions correctly for normal and erroneous test data, i.e. empty string, name of 2 characters, name of 21 characters, name of 10 characters. (1)	0



# Justin Heath







# FLARE: A Framework for Learning About Relational Elements

**Justin Heath, Arunside School**





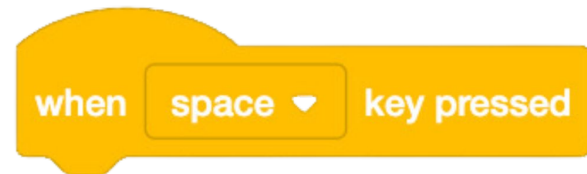
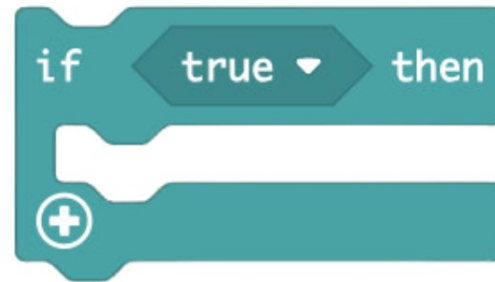
## Research Questions

How do  
teachers  
respond?

How do  
children  
respond?

Can it fit in  
with existing  
curriculum  
plans?

# BLOCKS - Self Contained Actions

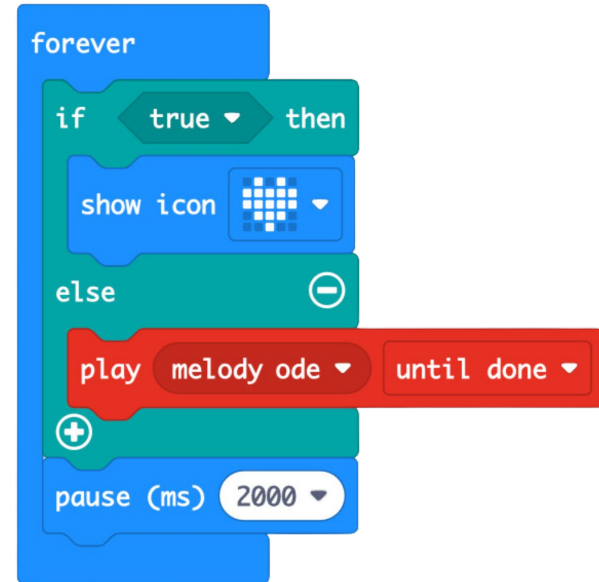


# SEGMENTS - Islands of Blocks



A Scratch script starting with a 'when green flag clicked' event block. The main script contains four blocks: a 'when football collides with net' block, followed by three 'set to' blocks for 'football speed' (0), 'football y' (20), and 'football x' (9), and finally an 'add 1' block for 'scoreNumber'.

```
when green flag clicked  
when football collides with net  
football speed = set to 0  
football y = set to 20  
football x = set to 9  
scoreNumber + add 1
```



A Scratch script starting with a 'forever' loop block. Inside the loop, there is an 'if true then' block. The 'then' branch contains a 'show icon' block. The 'else' branch contains a 'play melody ode until done' block. After the if-then-else block, there is a 'pause (ms) 2000' block.

```
forever  
if true then  
  show icon  
else  
  play melody ode until done  
+  
pause (ms) 2000
```

# RELATIONS - Causal Relationships

The image shows a Scratch script on a light gray background with a grid of small plus signs. On the left, a blue 'forever' loop block contains two sub-blocks: a blue 'call decide\_when\_to\_water' block and a red 'pause (ms) PAUSE\_DURATION' block. To the right, a blue 'function' block is defined with the name 'decide\_when\_to\_water'. Inside the function, there is a red 'set' block that sets the variable 'calibrating' to the value 'true'.

```
forever loop:  
  call decide_when_to_water  
  pause (ms) PAUSE_DURATION  
  
function decide_when_to_water:  
  set calibrating to true
```

The image shows two Scratch code blocks. The top block is a yellow 'when space key pressed' block. The bottom block is a purple 'switch costume to diver2' block.

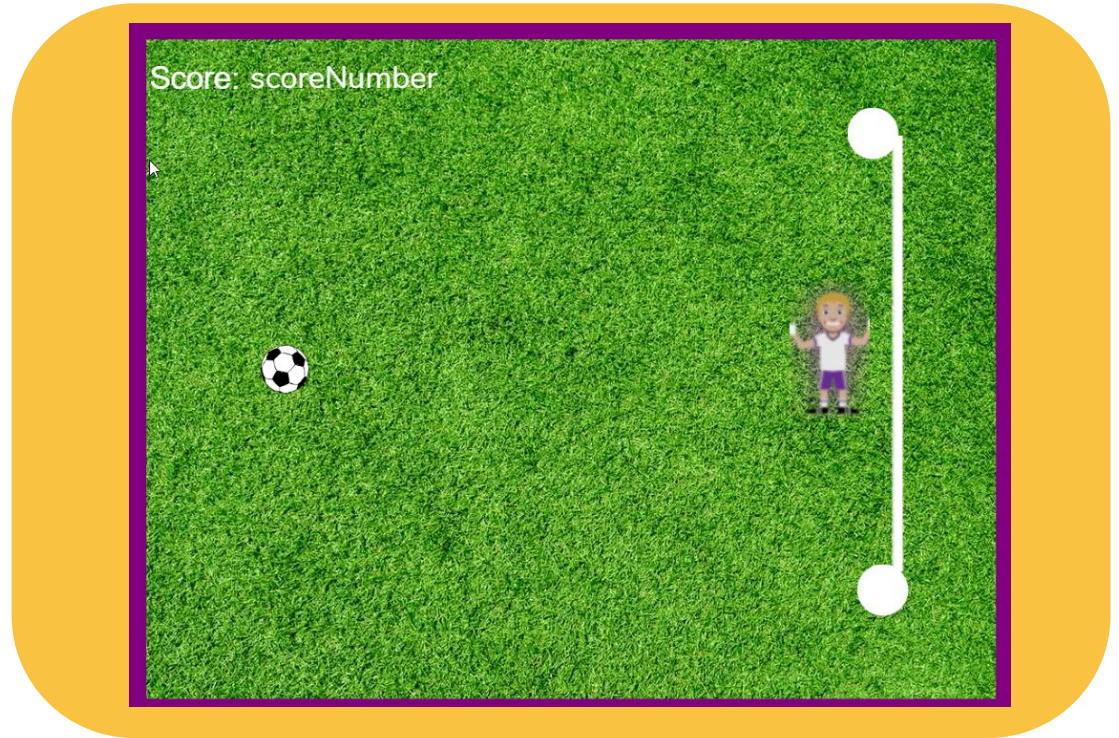
```
when space key pressed  
  switch costume to diver2
```

# Macro - Purpose and Overall Organisation

Predict what this program does.

So what jobs would the code need to do?

Could any jobs be functions?



# Possible Questions for Code Comprehension

## Example Questions

### 1 – Introducing

- Can you find the block that makes the character speak?
- Show the block that controls the background colour.
- Find the block that has a number inside it. What do you think this number might change?

### 2 - Developing

- The program uses this block to make the character move. Why did they choose this specific block?
- We see [a block] a few times in the code. What do you think its purpose is?
- We've used blocks like this one [shape/colour] before. What do these types of blocks often do?
- Can you think of another situation in a different program where you might use this type of block?

### 3 - Challenging

- If we try making this number smaller, what might happen?
- Is there a better way to achieve this same effect using a different type of block?



## Teachers Trialling FLARE

Teacher A

Had taught  
programming  
before.

Teacher B

Never taught  
programming  
before

Teacher C

Computing  
Leader at  
another school





## **Trial the Model not the CPD**

Deck of Slides

Detailed Descriptions  
of Elements

Differentiated Example  
Questions

A snippet suitable for  
the first lesson

## Platforms Used

SCRATCH

purple  
mash

 micro:bit



## Data Collection

- Detailed interviews
- Encouraged to put the boot in
- Otter AI transcription
- Atlas to Code
- Analysis



## Teachers' Responses

“...having a really clear structure that goes from simple to complex made it **easier for me to understand and teach it more effectively.**”

- Teacher A



## Teachers' Responses

"I didn't understand half of them. There were elements that I was **just totally baffled by.**"

(Relations and Macro)

- Teacher B



## Teachers' Responses

"I didn't understand half of them. There were elements that I was **just totally baffled by.**"

(Relations and Macro)

- Teacher B



## Teachers' Response Summary

- Improved planning and questioning
- Increased confidence
- One teacher described being able to adapt the lesson for different needs within a lesson using the structure questions
- Challenges understanding and implementing some FLARE elements
- Materials successfully used to deliver CPD by one participant.



## Children's Responses

**"They were actually way more engaged."**

- Teacher A

**"...they then also began to question me more as well....'Well, could we try that? Or how would...?'"**

- Teacher C





## Children's Response Summary

(Reported by teachers)

- Increased child engagement.
- More interactive and conversational classroom environments.
- Children's confidence in programming abilities raised



## Integrating into the Curriculum

**"I found the macro element hard."**

- Teacher C



## Integrating into the Curriculum

**"It's hard to ask questions about a language  
you're still learning."**

- Teacher B



## Integrating into the Curriculum

"I used it on three separate lessons, primarily as a starter. So I would have the bit of code on the screen and I knew what my questions were because I'd used the template. ...

But equally, when I was moving around when they were working, it structured my questioning."

- Teacher A



## Integrating into the Curriculum

- Successful use within the existing curriculum
- Difficulties applying the Relations and Macro Elements
- Needs understanding of the coding concepts being taught as well as FLARE.



## Teachers' Responses

**"I would have no hesitation in doing it again  
next year."**

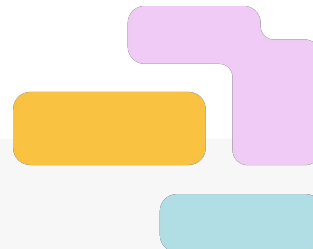
-Teacher B



## Conclusion and What Next

Encouraging, but still some way to go...

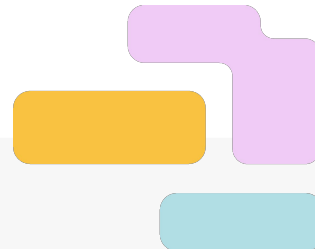
- Improved Clarity for relations and macro
- Visual Aids
- Separate versions for different programming platforms
- Combining with other models from research
- Integrating into the curriculum as a spiral model
- Introducing to more teachers



# Jo Hodge



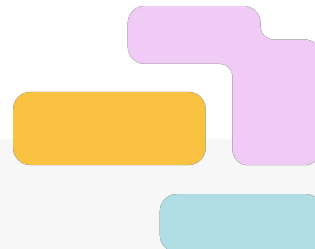




# Which PRIMM activities lend themselves to assessment?

**Jo Hodge - Our Lady of Lourdes Primary School**





# Motivation - Why?

- Looking at assessment within all curriculum subjects.
- PRIMM - assessment tool
- Research-based practice



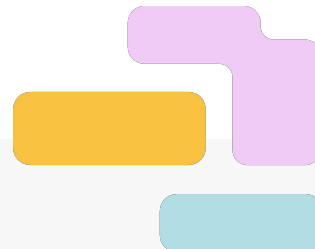
 Predict

 Run

 Investigate

 Modify

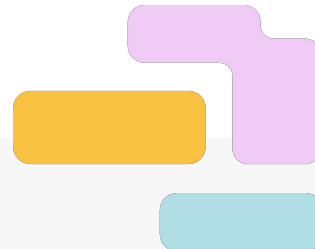
 Make



## Intervention - How?

- Two year 4 classes - adapted Teach Computing Unit to include PRIMM
- Explicit use of PRIMM vocabulary and actions
- Fellow colleague delivered the units independently

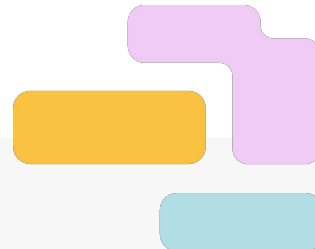




# PRIMM Activities

- Code snippets around code reading, tracing and matching.
- Parson's problems
- Identify gaps in their learning - predict
- Like a maths starter



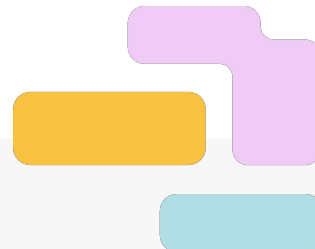


# Data Collection:

## Interview pupils

- How did PRIMM help you learn programming?
- What PRIMM activity did you like best and why?
- What have you enjoyed about the lessons?
- What have you found challenging?





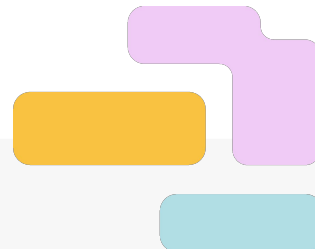
# Data Collection:

## Interview teacher

- Informal dialogue
- Recorded



‘Sometimes it is not about the final outcome, as you learn more about seeing what they’re doing at the time. The actual final piece doesn’t always tell you everything that’s gone on before. The PRIMM journey helps, you know, even without that final piece, who has achieved in your class.’ —Year 4 teacher



# Results: Impact

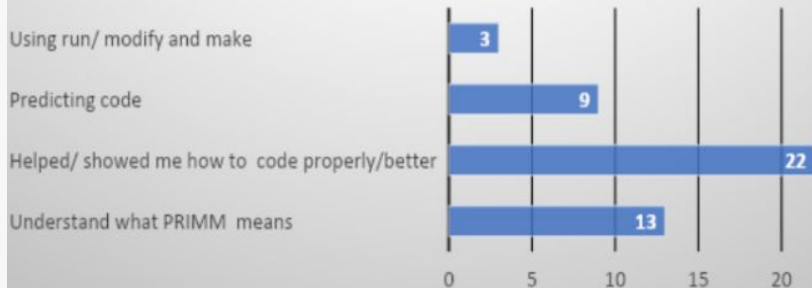
## 47 Pupils interviewed

- Supported the teacher in lesson delivery - independent/resilient
- Supported the SEND/EAL and less able coders - level of challenge/ clear structure
- Provided ongoing assessment opportunities
- PRIMM - predict like a starter very effective

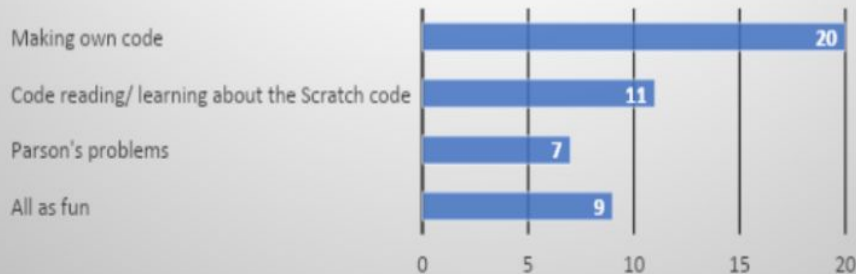




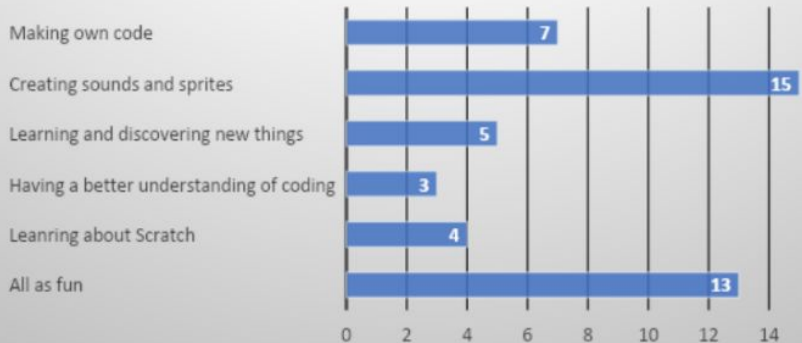
## How did PRIMM help you to learn programming?



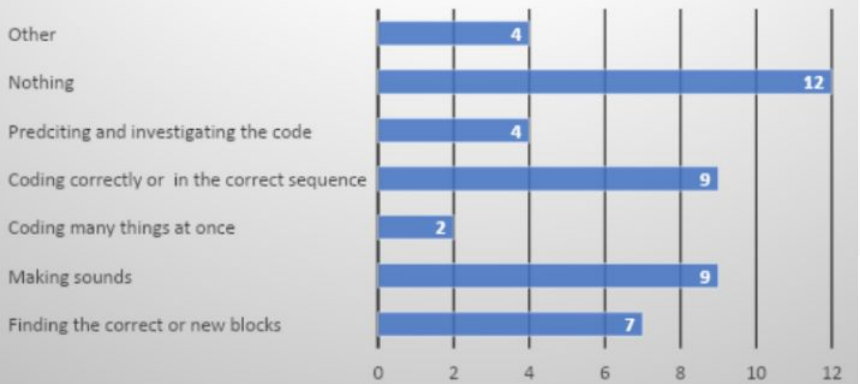
## What PRIMM activity did you like best and why?



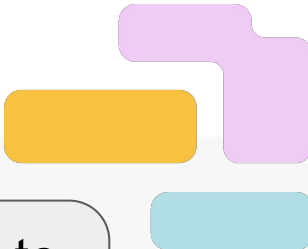
## What have you enjoyed about the lessons?



## What have you found challenging?








It helps me know  
what to do

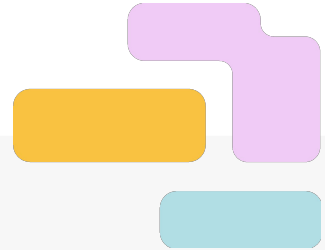
It helped me to  
do it in the right  
order

I like PRIMM because...



It helped me to  
make cooler  
codes

Doing the  
actions  
helped me  
remember

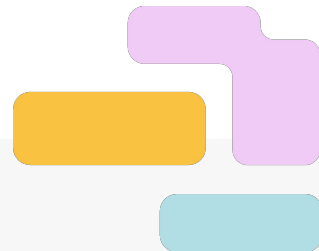


‘This doing it through PRIMM and giving them a chance to just modify it themselves and have a go for themselves. I had nobody coming up to me and saying right, what’s next? Or I’ve finished that, or I don’t know what I’m doing now. Can you come and help me? There was nobody that needed any help.’—Year 4 Teacher

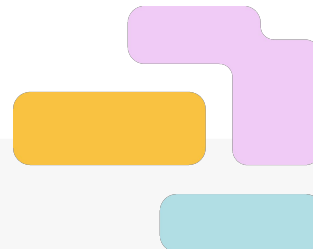


## Next steps...

- Introduced to whole staff
- Integrate into TC curriculum



thank  
you!



# TICE 3 - 2024/2025

Introducing a new partnership between CAS and RPCERC



# We will shortly be recruiting for next year's TICE!

## Face-to-face option (restricted to 10 teachers)

This will include bursary support from CAS to attend 2 full-day research meetings at the BCS offices in London

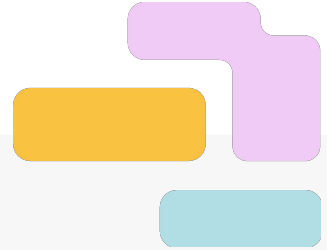
The rest of the programme will be similar to the online option

## Online option

No need to travel to London, and everything is provided online, but no funding is available.

As this year, webinars, workshops, and your helper will guide you through on both options

Sign up forms will be available before 26th July at <http://computingeducationresearch.org>



# Any questions?

