**Collaboration between mathematics and computing departments**

Computing is a relatively new subject in schools. But computing in mathematics education has a long history.

In the 1980s and 90s students could be found drawing ever more complicated patterns and spirals using LOGO on a BBC B microcomputer. And in doing so, they were using their mathematical knowledge and problem solving skills to produce some impressive designs. Fast forward to today, and freely available software such as Geogebra, Desmos and Autograph now provide great opportunities for students to use technology as part of their everyday learning experience.

But what about Scratch or Python?

Recently, I have been experimenting with flowcharts, inspired by Colin Foster’s ‘Flowchart Investigations: Explorations in Mathematics’ produced by The Mathematical Association.

For example:

Think of a number

Add 3

Add 7

Double it

Double it

Subtract

Write down the answer

Students are given the flowchart and asked to explore what happens with different inputs.

* What do they notice?
* Does this always happen?
* What happens if we start with a decimal number?
* What happens if the start number is negative?

No matter what number you enter, the result is always 8. Can students explain their findings and to convince a friend that no matter what the input, 8 will come out? How could they prove it algebraically?

When working on this problem, I wondered whether, rather than plugging numbers into the flowchart, I can write a program to do the calculations for me.

This is my Scratch program:

A screenshot of a cell phone

Description automatically generated with medium confidence

In the classroom, students can be asked to write their own program, or can be given this program and asked to say whether they think it will work. Can they modify the program to make it work more efficiently? Or create a version their own?

Alternatively, the same flowchart can be created using the programming language Python:

# Add and double

a = input(" Type in your starting number ")

a = float(a)

b = a + 7

b = b \* 2

c = a + 3

c = c \* 2

d = b - c

e = str(d)

print("The final answer is "+e)

Last term I worked with a trainee mathematics teacher who had recently completed their mathematics degree. They were bemoaning the fact that they had not done any programming before university and had to take a ‘crash course’ in their first year to catch up.

Could activities like this begin to bridge the gap between subjects in school?

If you know of any activities which require students to code as part of their mathematics studies- I’d be interested to hear from you. Do any computing teachers use mathematics as a context for coding activities? What activities have been successful and how have students reacted to them?

If you are interested in exploring the link between maths and computing further, the Yorkshire Ridings Maths Hub are collaborating with their local Computing Hub are working with teachers to explore further the synergy between the two subjects.

More details of the maths hub/computing hub collaboration can be found [here](https://www.yorkshireridingsmathshub.co.uk/projects/making-connections-between-maths-and-computer-science/)