

My Scratch Work – Amanda Wilson

8 lessons for teaching basic programming concepts.

Notes for Teachers to go along with the lesson workbook

Jan 2011

The following notes accompany the Scratch workbook(teacher version with answers). These lessons were used with children aged 8/9 years old, who were working in pairs. It gives brief details of how I chose the lesson plan and what is involved in each lesson. Each lesson can be completed within one hour and usually the children would have 5 or 10 minutes at the end to modify their work and experiment with Scratch themselves.

Contents

Lesson order justification.....	3
Lesson Plan.....	5
Lesson 1 – Introduction to Scratch	5
Lesson 2 – Sequences	5
Lesson 3 – Sequence 2	5
Lesson 4 – Iteration.....	5
Lesson 5 – Conditions	6
Lesson 6 – Broadcasting.....	6
Lesson 7 – Scratch Cards.....	6
Lesson 8 – Revision lesson/Test.....	6
Appendix 1 - Cambridge starters ICT initial steps – control page 17-18.....	7

Lesson order justification

On deciding what the children should be taught factors such as the children's age had to be taken into account. This was due to the fact that being only 8 the language used within the lessons had to be appropriately aimed at them. Another factor was that while some games had been created by the researcher for a previous project with older children the concepts covered in some of them were not going to be covered in the intended lessons. At this stage in school children are only just being introduced to this.

From looking as well at the scratch site they have a list of the concepts covered with scratch and the following basic concepts that were used were:

- Sequence
- Iteration
- Conditional
- Coordination & Synchronisation/Broadcasting

These concepts were used to build the lessons along with the CfE (LTS, 2009)¹ guidelines and the Cambridge test also– see Appendix 1(University of Cambridge, 2010a)².

This table shows the outcomes that children are supposed to attain and at what stage.

Subject	Outcomes & Stages of CfE		
	Early(Pre-school to end P1)	First(To end of Primary 4)	Second(To end of Primary 7)
Maths	In movement, games, and using technology I can use simple directions and describe positions.	I can compare, describe and show number relationships, using appropriate vocabulary and the symbols for equals, not equal to, less than and greater than.	
		I can describe, follow and record routes and journeys using signs, words and angles associated with direction and turning	
Computing	I explore software and use what I learn to solve problems and present my ideas, thoughts, or information		
	Through discovery, natural curiosity and imagination, I explore ways to construct models or solve problems		
	I am developing problem-solving strategies, navigation and co-ordination skills, as I play and learn with electronic games, remote control or programmable toys.		
		I can create, capture and manipulate sounds, text and images to communicate experiences, ideas and information in creative and engaging ways	
		Having evaluated my work, I can adapt and improve, where appropriate, through trial and error or by using feedback.	

¹ LTS 2009, *Curriculum for Excellence: Technologies Experiences and Outcomes*.

² University of Cambridge 2010a, *Cambridge ICT Starters, Syllabus for Assessment 2010*.

This table shows how the CfE outcomes and programming concepts tie in.

Lesson	Concept	Outcome
1	Sequence	
2	Sequence	
3	Sequence	In movement, games, and using technology I can use simple directions and describe positions. I can describe, follow and record routes and journeys using signs, words and angles associated with direction and turning
4	Sequence, Iteration	
5	Sequence, Conditional	
6	Sequence, conditional, Coordination & synchronisation	I can compare, describe and show number relationships, using appropriate vocabulary and the symbols for equals, not equal to, less than and greater than.
7	Sequence, iteration, Coordination & synchronisation	
8	Sequence, iteration	In movement, games, and using technology I can use simple directions and describe positions. I can describe, follow and record routes and journeys using signs, words and angles associated with direction and turning
All lessons		I can create, capture and manipulate sounds, text and images to communicate experiences, ideas and information in creative and engaging ways
		Having evaluated my work, I can adapt and improve, where appropriate, through trial and error or by using feedback.
		Through discovery, natural curiosity and imagination, I explore ways to construct models or solve problems
		I am developing problem-solving strategies, navigation and co-ordination skills, as I play and learn with electronic games, remote control or programmable toys.
		I explore software and use what I learn to solve problems and present my ideas, thoughts, or information

Lesson Plan

Lesson 1 – Introduction to Scratch

Talk to the children about sequences and how things need to follow a certain order for them to work e.g. I use making your cereal in the morning to illustrate the point. I then tell them the three S's that you need to make Scratch work (scripts sprites and stage).

I then let them go to computers and try out scratch for themselves.

Lesson 2 – Sequences

This lesson introduces the basics of drawing shapes and creating a simple program to do so. We talk at the beginning. I have a remote control toy which I use to demonstrate the idea of movement (e.g. move turn move turn) I also have printed out “blocks” which we use to make a little program for the toy to follow before we then go onto the computers. (Concrete programming to an extent Demo 2008)³ This could be done differently but I only had one toy would be nice to have more to let the kids do it as well. Or time permitting the children could work in teams and take turns at being the robot and following instructions from the other children.

Lesson 3 – Sequence 2

Continuing on from last week's lesson again similar theme. I talk first and do a demo with the remote control toy. This time though I get the children to do some writing first and work out the instructions as to how to get the cat from A to B. I used this as a test to gauge children's knowledge of the little work that we had done (bear in mind when I did this it was 8 lessons and I was trying to see if any improvement had occurred over those 8 lessons – given that there was no set tests/assessments in computing for primary school kids I had to look elsewhere – the Cambridge ICT starters provided me with some inspiration for creating this test. I knew from talking to the class, none of them knew what a computer program was (a lot thought it was to do with playing games or the internet)

Lesson 4 – Iteration

Again using shapes but this time illustrating with repeat to show how the programs can be done quicker. By this lesson I usually just explain to the children using my cut out blocks on the board. I feel that the toy has served its purpose now in getting them to think about things.

³ Demo, G.B., Marciano, G. & Siega, S. 2008, "Concrete programming: Using small robots in primary schools", *8TH IEEE INTERNATIONAL CONFERENCE ON ADVANCED LEARNING TECHNOLOGIES, PROCEEDINGS* pp. 301.)

Lesson 5 – Conditions

I try to get the children to fill out a little piece just to try and remember what they are working on. I'm also trying to get them into the habit of using some terms in scratch e.g. sprites scripts etc. this lesson focuses on making things happen because something else has happened.

Lesson 6 – Broadcasting

This lesson shows the children how to make a simple animation between two sprites and get them to talk to each other. Again I start the lesson off by going over the sheet then let them go off to work on it.

Lesson 7 – Scratch Cards

This lesson simply involves giving the children the set of 12 cards from the scratch site and let them work their way through (I give each pair a card to start with and then we swap so I know when they have completed the card)

Lesson 8 – Revision lesson/Test

There are two ways of doing this, one follow the instructions. Two – maybe get the children to simply draw the path first then add in the extras.

I have added in solutions for both ways.

Appendix 1 - Cambridge starters ICT initial steps – control page 17-18

INTRODUCTION

In this module, students learn to control the actions of a screen turtle by giving a series of instructions and should consider examples of how we can control machines and devices: for example, students and their families control devices by giving them instructions when selecting sweets or a drink from a vending machine, setting a microwave oven to cook some food, or recording a television program using a DVD/video recorder.

Students should be given opportunities to consider instructions for a range of activities, not just those involving a screen turtle: for example they might give one another directions for moving from one place to another in the classroom, or using simple maps. Where resources allow, they might also give instructions to control the actions of a floor turtle: in all cases, students should be encouraged to use instructions involving distance and direction, and to predict the consequences of a set of instructions.

Learning Objectives		Pass/ Merit	Assessment Evidence	Amplification of Assessment Criteria
1	Give a screen turtle a set of instructions to achieve a specified objective	P	Printed copy of task set and screen grab of turtle path	This should be a very simple task, e.g. to move onscreen to hit down a skittle or visit an area requiring only the use of forward and backward moves (or 180 degree turns) and quarter-turns for the first three moves.
2	Record the instructions to the turtle	P	Students' written instructions to the turtle, which can be written on the task sheet or screen grab	Students may need to amend their instructions before writing them down as a set of instructions.
3	Use angles other than 90 or 180 degrees	M	See above	If this is being set as an extension of the Pass task, the turtle path after the first three moves will involve angles such as 60 or 120 degrees.
4	Create a set of instructions involving at least 5 moves to achieve a specific target	M	See above	For example, the students could program the turtle to move along a route visiting certain locations and/or avoiding an obstacle.

GETTING STARTED

You will need a simple version of control software, such as LOGO. Several freeware versions of LOGO can be obtained via the internet (e.g. MSW LOGO). You may choose to change the colour and width of the pen, the background and the font size to make it simpler to use for younger students. An example of control software made specifically for younger students is Softease Turtle.

Before you introduce the software:

- spend time on activities involving instructions to get a friend from one place to another in the room, perhaps going around obstacles;
- develop a set of instructions, perhaps on cards, which students can use to develop sets of instructions for each other;
- use board games and floor tiles to convey the notion of spaces and units;
- if one is available, let students drive a remote-controlled toy car around a course;
- if you have a floor turtle, show students how to give it instructions then let them try out their own combinations of distance and direction instructions to see the effect.

Prior Knowledge	New Words
<ul style="list-style-type: none">• Demonstrate awareness of simple switches• Understand the terms on and off• Understand forward, back, left, right, turn• Count• Estimate distances• Understand quarter, half and full turns.	<ul style="list-style-type: none">• Sequence• Order• Instruction• Screen/floor turtle• Control devices• Commands• Programmable• Start/stop

ASSESSMENT GUIDELINES

Students might be asked to draw a route from one named point to another on a simple plan or map. They could then write a set of instructions to make a screen turtle follow the same path. This should include at least one turn of 90 or 180 degrees.

Students will achieve Pass level if they are able to complete up to this point. In order to achieve Merit level, they must successfully complete the second part of the task, which involves drawing a path from the previous end point to another destination on the plan: this should involve at least two more moves, using angles other than 90 or 180 degrees.

If there are two separate tasks for Pass and Merit, then the first three moves of the Merit task would still involve turns of only 90 or 180 degrees.