Plan and program a monologue

Computing Science Concepts

-Sequence
-Order can be important in sequences
-Algorithm
-Four levels of abstraction
-Inputs

National Curriculum Programs of Study

Pupils should be taught to:

design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts

use sequence, selection, and repetition in programs; work with variables and **various forms of input and output**

use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs

Research Focus

Computer scientists have **four levels of abstraction**[2]. The ideas level, planning level (which includes the algorithm), code level, and execution level (testing the code).

Monologue Planning Version D Modelled Offline

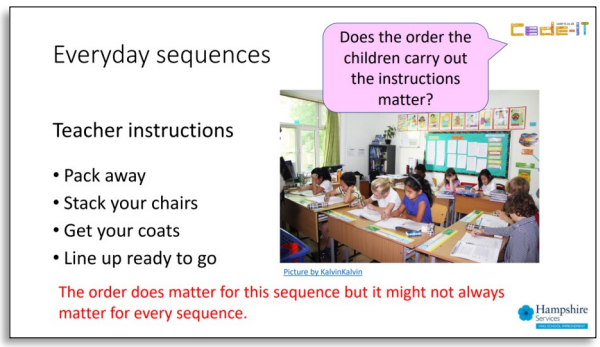
Overview

1. Introduce the concept of sequence through everyday examples
2. Extend the concept of sequence through role play
3. Introduce the challenge of creating a monologue offline
4. Pupils plan their own short monologue
5. Model turning a plan into Scratch code using both methods
6. Pupils turn their algorithmic planning into their own programmed monologue
7. Pupils fill in the summative assessment form

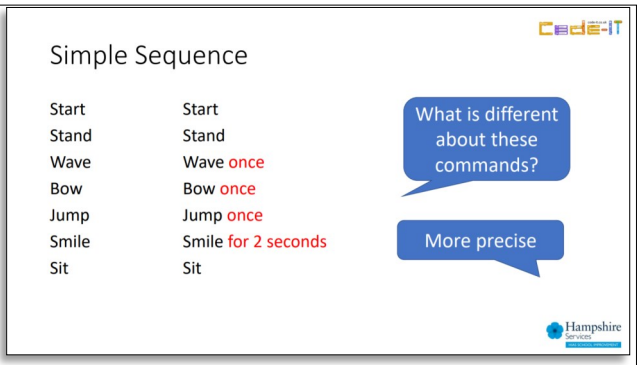
1. Introduce the concept of sequence through everyday examples

Download Everyday computing concepts PDF from http://code-it.co.uk/knowledge or directly from http://code-it.co.uk/wp-content/uploads/2019/04/everydaycomputingconcepts.pdf

Use the first eight slides to introduce the idea of sequence in our everyday lives. The answer mostly appear in red text. There are some sequences where the order is less important and some where the order is paramount. The same will be true for programming sequences. By linking the concept to its everyday use you are linking to known knowledge which means pupils are more likely to assimilate the idea.



2. Extend the concept of sequence through role play



Download Concepts before coding PDF from http://code-it.co.uk/knowledge or directly from http://code-it.co.uk/wp-content/uploads/2019/04/conceptbeforecoding.pdf
Follow the links in the menu to simple sequence. Use those six slides to roleplay and write simple fun sequences. These slides introduce the idea that the more precise a sequence is the more useful it is. Stop when you get to the dance slide.

Formative assessment opportunity

While pupils are writing their own sequences go round and check them all. Is anyone struggling? Have they copied the one on the board exactly? This is often an indication that they are not sure how to create their own or that spelling is an issue. A good supportive activity is to get them to tell you about their own sequence that you scribe for them.

3. Introduce the challenge of creating a monologue offline

Share the Hook

Take a few moments to excite and enthuse your pupils about the monologue hook you or they have chosen. You might want to model one on the board along with some indication of how long it will take to read it in seconds.

4. Pupils plan their own short monologue

Give out the idea and algorithm blank planner. Give pupils time to complete it.

Formative assessment opportunity

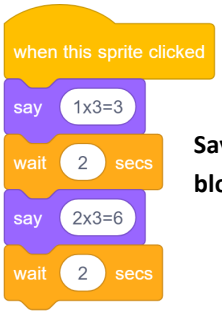
Check pupils algorithms as they are creating. Do they make sense? Have they kept to their idea? Have they included punctuation? Have they included timings?

5. Model turning an algorithmic planner into Scratch code using both methods

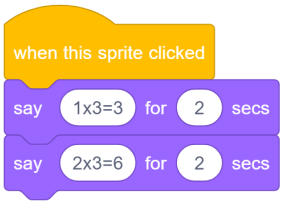
Import a sprites using the choose a sprite button or choose a sprite from library in Scratch 2.0.

Add a when this sprite clicked starting block. Drag out a say hello for two seconds block and a wait block and start to turn your monologue into code. Frequently refer back to your monologue and make a great play over matching the times and order of the code to the algorithm.

Make sure you show pupils both methods and get them to try both but build



Say and wait blocks

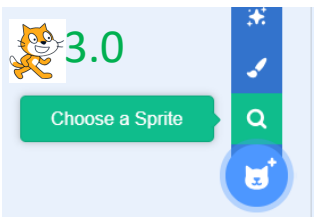


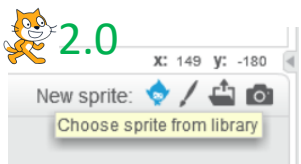
Timed say blocks

Idea and algorithm planners

If pupils are only writing a monologue use the **A4 monologue planner.**

If pupils are going to complete the stage and sound module then use the left half of the **A3 monologue planner.**





Research References

Four levels of abstraction

This article includes an example of the four levels of abstraction and sign posts the work of Waite and Armoni in using them with school level pupils.

http://code-it.co.uk/algprogdiff/

their monologue in only one type.

6, Pupils turn their algorithmic planning into their own programmed monologue

Give pupils time to do this. Those that finish earlier than others can create a different type of monologue.

7, Pupils fill in the summative assessment form ✓

You can find a summative assessment quick Kahoot Quiz linked at http://code-it.co.uk/gold/

Whilst Kahoot is a limited assessment tool it is free and it is easy for teachers to pass the results back to code-it via phil.bagge@code-it.me so I can look at which method provides best short test results. Not conclusive but useful.

If you pass the results back please

1, Anonymise the results by removing the names

2, Ask the head teacher for permission

3, In the email title say which module you are doing (ie Animation D)

Inspired by Behaviour Rubric created with @MarkDorling and linked at http://code-it.co.uk/attitudes/