SJB PRIMARY

foundations and faith
FOR THE FUTURE

Digital Technology Curriculum Hangarau Matihiko

MOE Primary Schools Session

Cushla Young, 2020
cushla@sjb.school.nz
http://cushlyoung.com/

Difference between digital education & digital technology.

1

Our Why

Why are we heading this way?

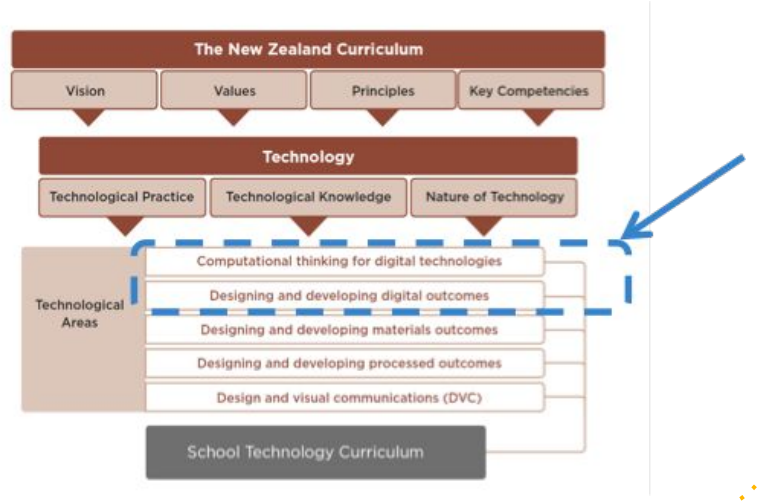
Guiding Principles

Digital Technology Curriculum

Give an intro of the digital education curriculum and background.

Future Focused Students

Link to the front of the curriculum



Want to know more? Link here to docs:
SJB Curriculum: Digital Technology

The Computational Thinkers

concepts

Logic
Predicting & analysing

Evaluation
Making judgements

Algorithms
Making steps & rules

Patterns
Spotting & using similarities

Decomposition
Breaking down into parts

Abstraction
Removing unnecessary detail

approaches

Tinkering
Changing things to see what happens

Creating
Designing & making

Debugging
Finding & fixing errors

Persevering
Keeping going

Collaborating
Working together

We're all computational thinkers here!

When you think about it, whether we're parents, pupils or teachers – we're all natural computer scientists, capable of computational thinking.

Our brains, like computers, process, debug and make

CAS Barefoot

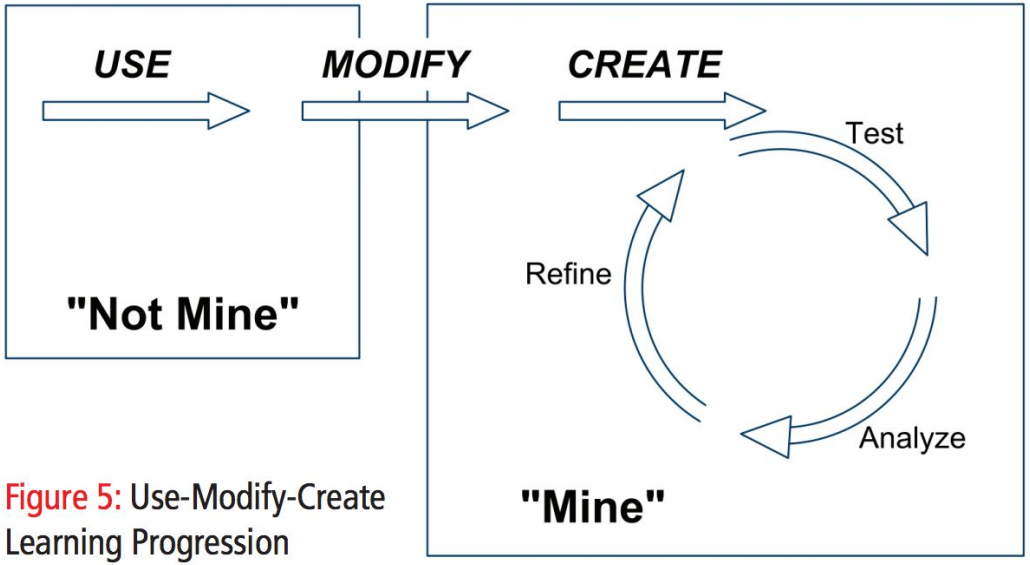


Figure 5: Use-Modify-Create Learning Progression

The big goal of the new curriculum is to take students from be JUST users to having some power over the technology, especially being able to write their own programs

Digging into the digital tech curriculum

- Here you have a list of items
- And some text
- But remember not to overload your slides with content

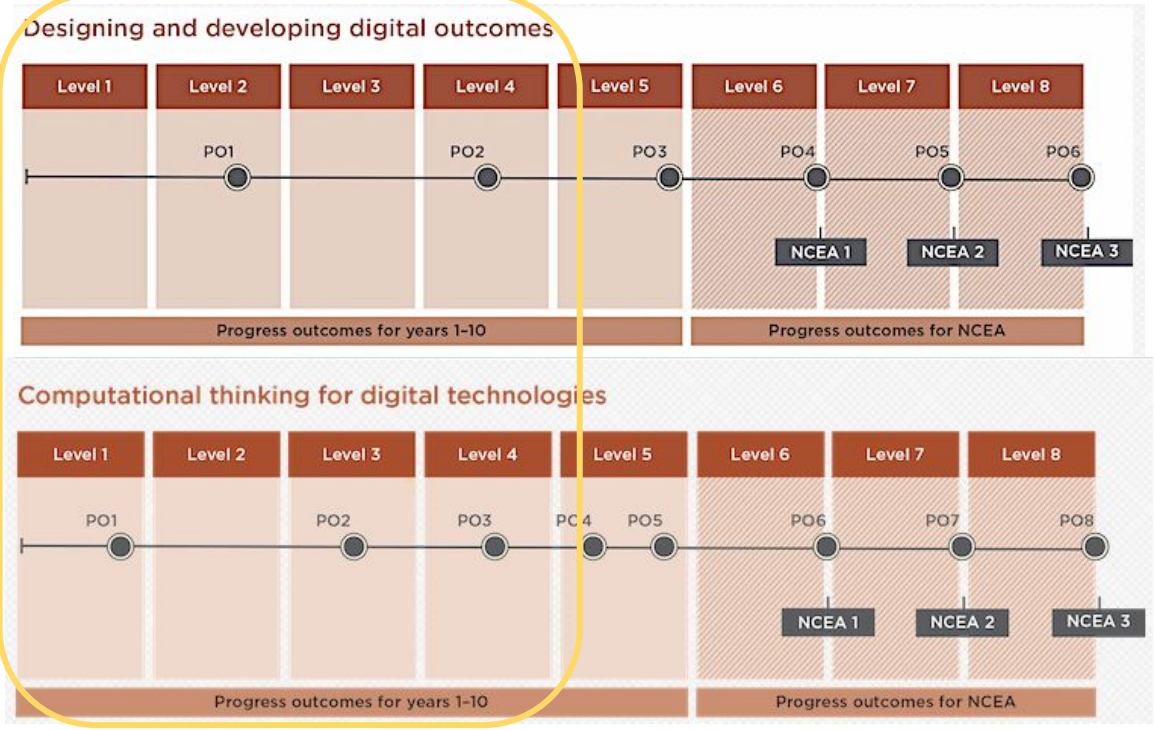
Your audience will listen to you or read the content, but won't do both.

2

The What...

What do we need to consider? What does best practice look like?

The progress outcomes… Where they sit at Primary levels



Let's look at the exemplars for digital tech levels

L1: Designing and developing digital objects

Background

Lindy has been learning that digital devices receive inputs and produce outputs.

She has previously learnt that if she wants to use an image from the Internet, she needs t
make sure the person who created the image allows other people to use it without char

Task

The students are given a brief to create a flowchart that shows the sequence:

- giving input to a digital device
- the device processing the input
- the output of the device.

They work with Mr Hughes to each select an image of a digital device to work with, and find suitable images of an input, process and output for the device.

They then import the images into the Book Creator application on an iPad. They use the app to make their flowchart and briefly describe what is happening at each stage.

The task gives the students the opportunity to practise creating meaningful texts, using a particular layout and visual language features to communicate their understanding.

Some key Concepts to Unpack at DDO PO1:

Input
Device
Output
Sequence

DDO PO1: Exemplar

In authentic contexts and taking account of end-users, students participate in teacher-led activities to develop, manipulate, store, retrieve, and share digital content in order to meet technological challenges. In doing so, they identify digital devices and their purposes and understand that humans make them. They know how to use some applications, they can identify the inputs and outputs of a system, and they understand that digital devices store content, which can be retrieved later.

Student response

TV flowchart

We use the remote. The TV turns on. We can watch the TV.

Mr Hughes:	*Can you tell me which bit of the television system is the input?*
Lindy:	*The remote is the input – it's how I switch it on.*
Mr Hughes:	*Why is the remote the input?*
Lindy:	*Because it's how I tell the television what to do.*
Mr Hughes:	*Can you think of another digital device that has a different kind of input?*
Lindy:	*Our microwave at home has a keypad.*
Mr Hughes:	*So now you've used the remote to switch the television on, could you use the remote in another way?*
Lindy:	*I could turn the volume up or change the channel.*
Mr Hughes:	*So which bit of your sequence is the output?*
Lindy:	*It's what comes out of the television.*

DDO Level 4 Exemplar

Student response

Samuel and Cody create four QR codes, save them to the iPad camera roll and email them to Ms Ryan to be printed out.

Their first QR code gives a set of navigational directions to follow:

Ms Ryan: *What are the inputs, processes and outputs this system has used?*

Samuel: *When we wrote our instructions into the Qrafter software, that was the input. The app processed the instructions and turned them into a QR code, which is the output.*

Cody doesn't initially agree with Samuel, but he understands after Samuel explains it again.

Cody: *No, the QR code is the input. We scan it, so it's the input, and then the app processes it.*

Samuel: *That's when we're scanning them though. First we're making them, so the code is the output at the moment. We scan them later.*

Cody: *Oh, OK.*

L1: Computational Thinking

Background

As part of health and physical education, students have been learning names of body parts, practising movement skills, and developing their awareness of space around their bodies. This has included learning the song "Heads, shoulders, knees, and toes" and its associated actions.

Task

The teacher, Ms Sinclair, asks the students to teach "Heads, shoulders, knees, and toes" to a robot. She explains that computers can only understand specific, simple instructions. So to teach a robot to do something, the task has to be broken down. The students will have to think carefully about the song and its actions and break them down into clear steps.

To help the students with their instructions, Ms Sinclair offers to pretend to be the robot.

Health and physical education provides a meaningful context for this computational thinking task. The task also provides an opportunity for students to work together to achieve a particular purpose.

Some key Concepts to Unpack at DDO PO1:

Decomposition (breaking into parts)

Algorithmic Thinking
(Precise, Unambiguous Instructions)

Simple Debugging
(Find & Fix errors)

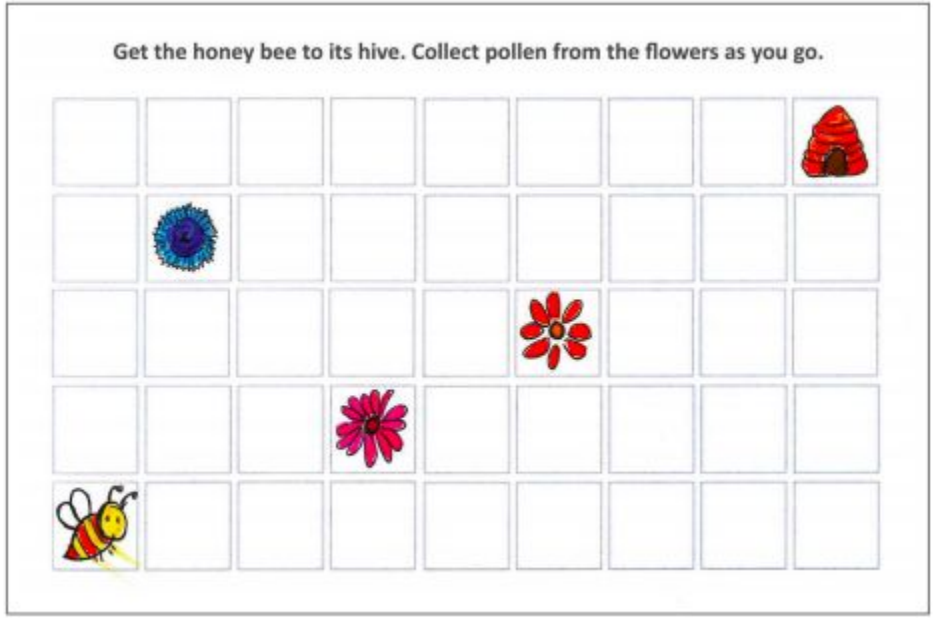
CT PO1: Exemplar

In authentic contexts and taking account of end-users, students use their decomposition skills to break down simple non-computerised tasks into precise, unambiguous, step-by-step instructions (algorithmic thinking). They give these instructions, identify any errors in them as they are followed, and correct them (simple debugging).

Student response

Ms Sinclair:	*What's the first thing you need to tell the robot to do?*
Ani:	*You go "Heads, shoulders ..."* (starts singing the song and doing the actions)
Ms Sinclair:	(in a pretend robot voice) *I don't understand.* (in her normal voice) *How would you explain the first thing you're doing?*
Ani:	*First you do this.* (puts her hands on her head)
Ms Sinclair:	*Can you describe exactly how you do that?*
Ani:	*You put your hands on your head.*
Ms Sinclair:	(puts her hands on her head)
Oliver:	*And then you sing, "Heads".*
Ms Sinclair:	(puts her hands on her head and sings "Heads") *That's right! What should I do next?*
Petra:	*Now you do this.* (sings "shoulders" and puts her hands on her shoulders)
Ms Sinclair:	(in pretend robot voice) *I don't understand.* (in her normal voice) *Could you explain exactly how I do that?*
Petra:	*You put your hands on your shoulders, and then you sing "shoulders".*

CT: Level 1 Exemplar



Student response

right up upup
right right down down
right right up
up right right right up

CT Level 2 Exemplar

Task

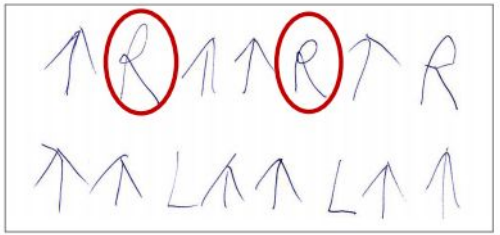
The students' task is to direct Māui (a "robot" student) around a grid to collect flax and take it to Tama-nui-te-rā, the sun. The grid has been marked out on the ground.

The students work in groups to write an algorithm to guide Māui around the grid, collecting all the flax before taking it to Tama-nui-te-rā. They are only allowed to give three different instructions: "take a step forward", "turn to your left", and "turn to your right".

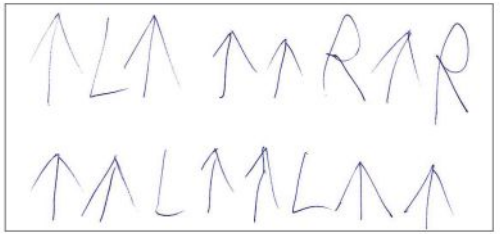
The instructions must be written using these symbols:

- "↑" (take a step forward)
- "L" (turn to your left)
- "R" (turn to your right).

Student response



There are two mistakes in one group's first algorithm.
They work together to fix these and write an algorithm that works.



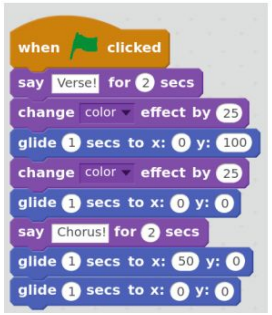
CT: Level 3 Exemplar

Task

Using their new knowledge of iteration, the students are asked to use a block-based programming environment (Scratch) to create an animated character that moves to music. The animation needs to repeat the same sequence of movements in the verse of the song and a different sequence in the chorus.

Mrs Anderson takes the opportunity to link iteration in digital technology to the way her students have been writing repeated patterns in their music activities. The task also enables her to reinforce the students' emerging mathematical understanding of the x and y axes.

Student response

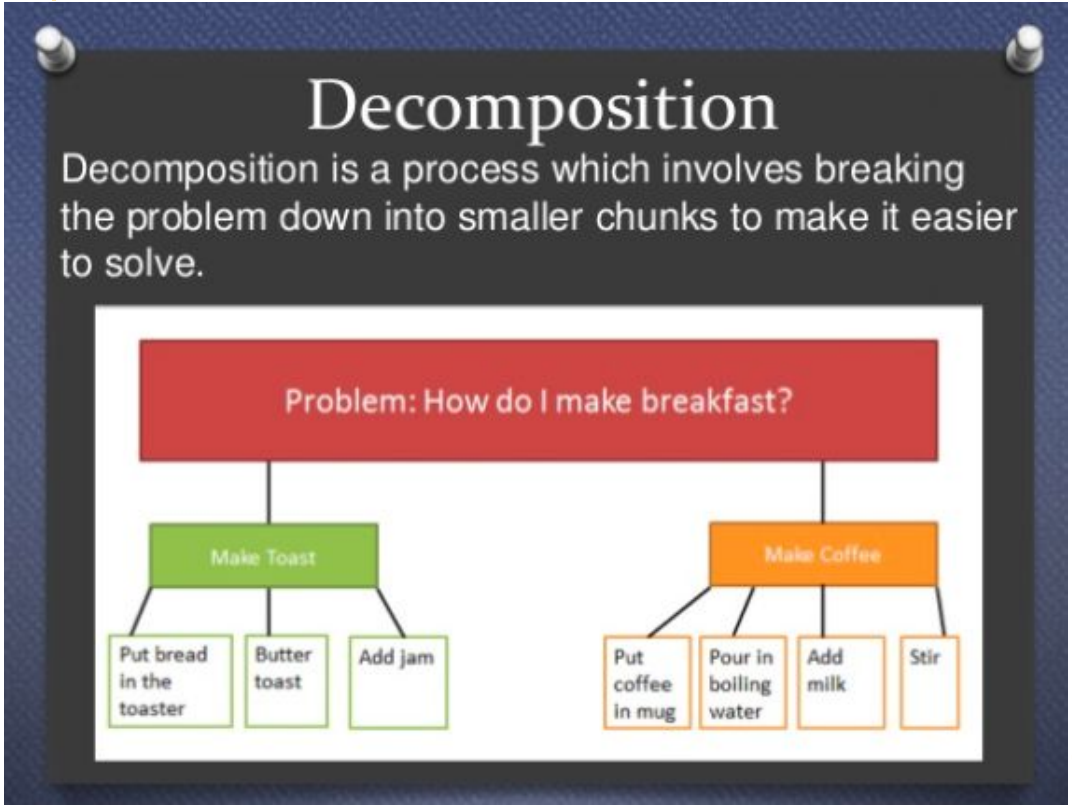


After creating her program, Maia finds that every time she wants her character to dance for a verse and then a chorus, she has to press the green flag again.

Using the knowledge she has gained about iteration, she modifies her program using the repeat block so that her character dances to the verse and chorus five times.

Some Key Vocabulary and Concepts at Levels 1-4

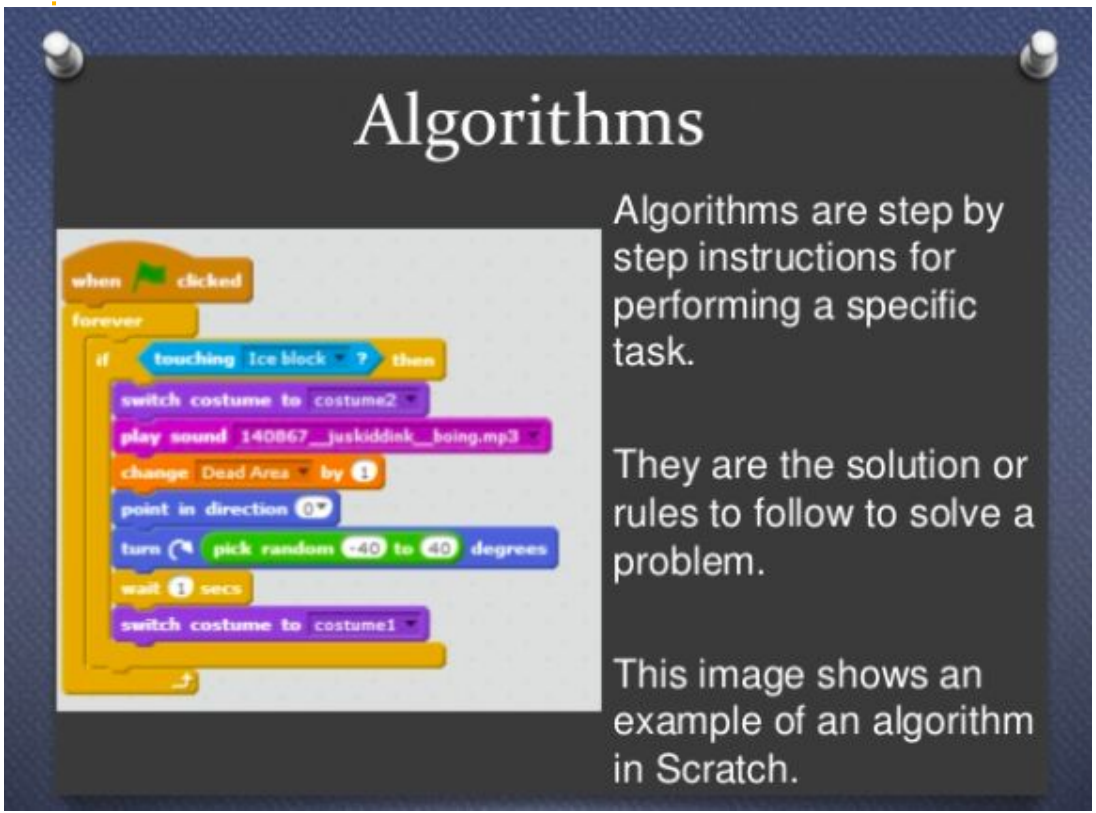
What's Decomposition?



Integration with procedural writing and probability.

In **authentic** contexts and taking account of end-users, students use their **decomposition skills** to break down simple **non-computerised tasks** into precise, unambiguous, step-by-step instructions (algorithmic thinking).

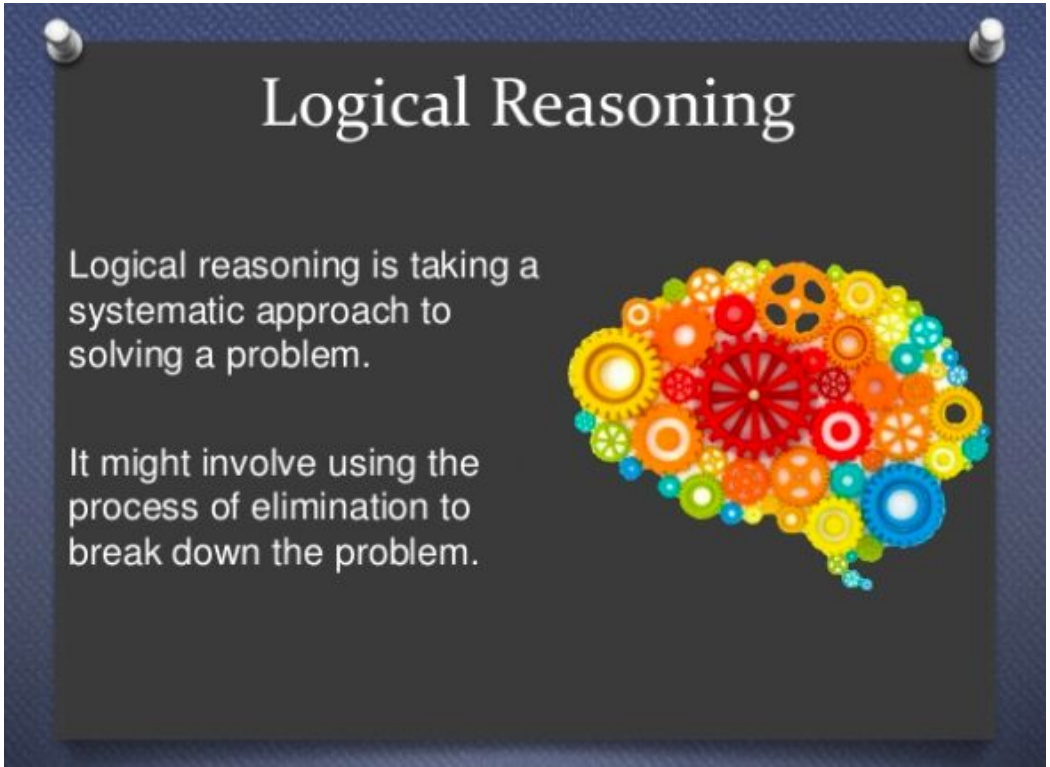
What are Algorithms?



Integration with maths, technology, physical education

In **authentic** contexts and taking account of end-users, students use their decomposition skills to break down simple **non-computerised tasks** into **precise, unambiguous, step-by-step instructions (algorithmic thinking).**

Where does logical reasoning come in?



Integration with maths, decoding in reading.

They give these instructions, identify any errors in them as they are followed, and correct them (simple debugging).

What is Debugging?

Integration with editing in writing, problem solving.

They give these instructions, **identify any errors** in them as they are followed, and **correct them** (simple debugging).

3

The How...

How do we deliver?

Computerised and Non computerised...

In Computational Thinking, the difference between level 1 and level 3 is that level 1 recommends:

Using authentic contexts, considering the end users, to do simple **non-computerised** tasks.

Where to begin? Unplug!

Begin with Unplugged Activities.

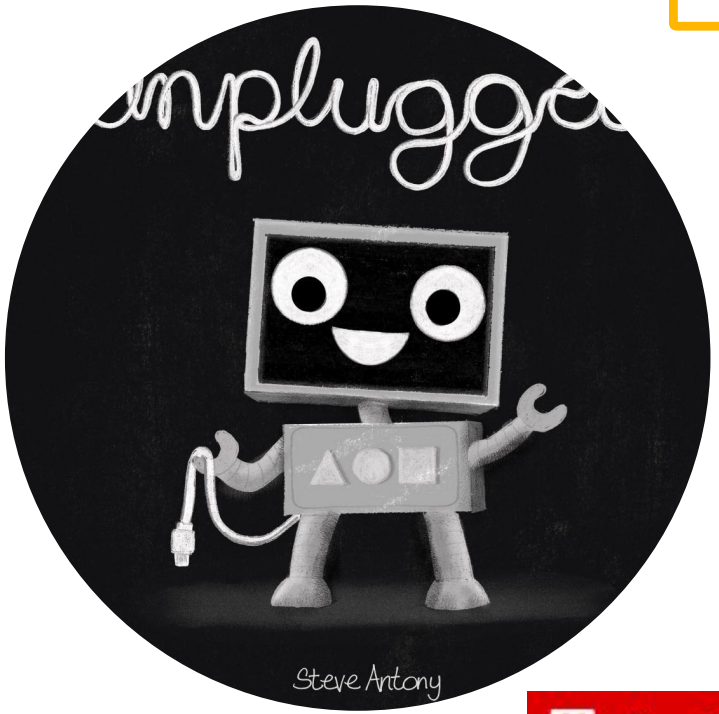
Mix throughout device-using lessons.

Kidbots

Ages 5 to 10
4 lessons
4 curriculum integrations
50 programming challenges

Sorting networks

Ages 5 to 14
4 lessons
2 curriculum integrations



CS UNPLUGGED

Story telling: Kidsbot

Storytelling

Literacy: Speaking

Using the same structure and resources as the Rescue Mission lesson, we can retell a story using Kidbots.

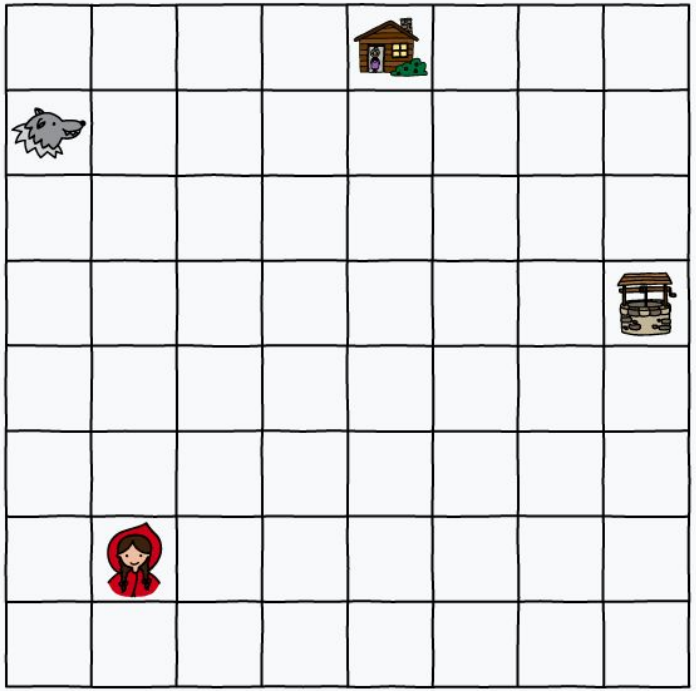
Key questions

- How could thinking about programming help us to retell stories?
- Why is it important to get the sequence right when retelling a story?

Retelling the story

We're going to write our own program that retells the story Little Red Riding Hood we just read. But I'm going to mix up the story!

I've placed on my grid the main ideas from the story.



The Orange Game

Curriculum Links

Great Principles of Computer Science [info]

→ Coordination

ACM K12 Curriculum [info]

→ Level I (Grades 3--5) Topic 11: develop a simple understanding of an algorithm

New Zealand Curriculum [info]

→
- → Technology Level 1: Planning for Practice
  - → Outline a general plan to support the development of an outcome, identifying appropriate steps and resources
- → Health and Physical Education Level 1: Identity, sensitivity, and respect
  - → Demonstrate respect through sharing and cooperation in groups

CS UNPLUGGED

Learning about binary

Background

Computers today use digits to represent information - that's why they're called digital systems. The simplest and most common way to represent digits is the binary number system, with just two digits (usually written as 0 and 1). It is called binary because there are only two different digits used, or two states. This unit and lessons explore how the binary system works and why it's important to understand how data is represented.

Unit Plans

Binary numbers

Ages 5 to 10

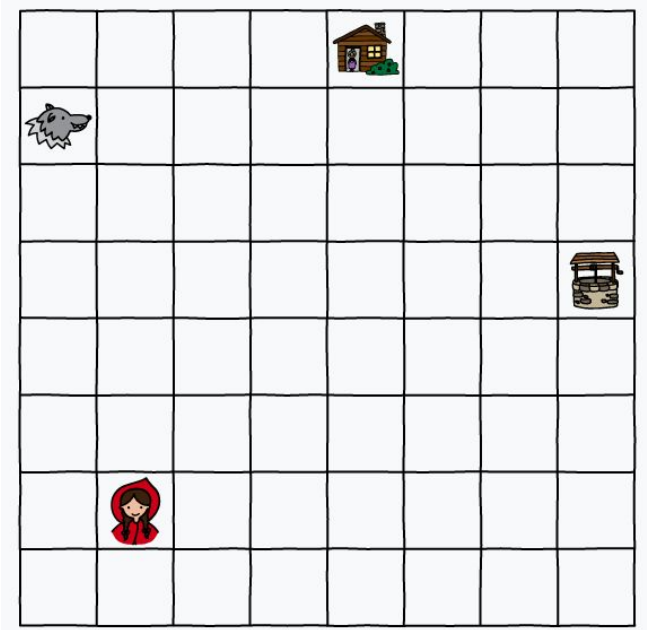
6 lessons

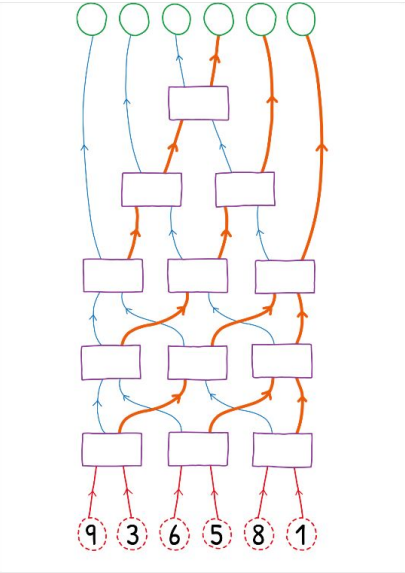
Curriculum Integrations

Activity	Curriculum Areas	Prerequisite Lessons?
Binary Candles or Normal Candles on your Cake	Literacy: Writing	Yes
Whose cake is it?	Literacy: Writing	Yes
Binary Name Necklaces	Art	Yes
Binary Patterns	Art	Yes
Binary Tunes	Performing Arts: Music	Yes
Biographies and binary number system history	Literacy: Reading, Literacy: Writing	No
Binary Art	Art	Yes

Ideas for schools...

Playground markings - links with physical education.





Ideas for schools...

Stop Motion to retell a story or sequence.

Ideas for schools...

Scavenger Hunts for a Native Bush Walk

Or

QR Codes to share their digital work alongside their piece of art.

Digital programming: pepeha, welcome to our community

Digital object

Identify what they're already doing for DDO.

Using Mouse Robots

The mouse robots are very similar to the BeeBots. They are easy to use and a great place to begin.

Mouse Bots

Using mBots

The mBots are the next level up from the mouse bots.

mBots

Using the Spheros

Spheros are amazing devices at teaching the basics of coding. You till need: ipads, Sphero Bolt kit (charged ahead of time)

Spheros

Using ScratchJR

ScratchJR is a great place to begin. It is a free app. There are loads of activities and lessons. They link with unplugged activities too.

Scratch JR

Using Hour of Code

Hour of Code is a very easy, independent and engaging place for students to learn the basics of coding.

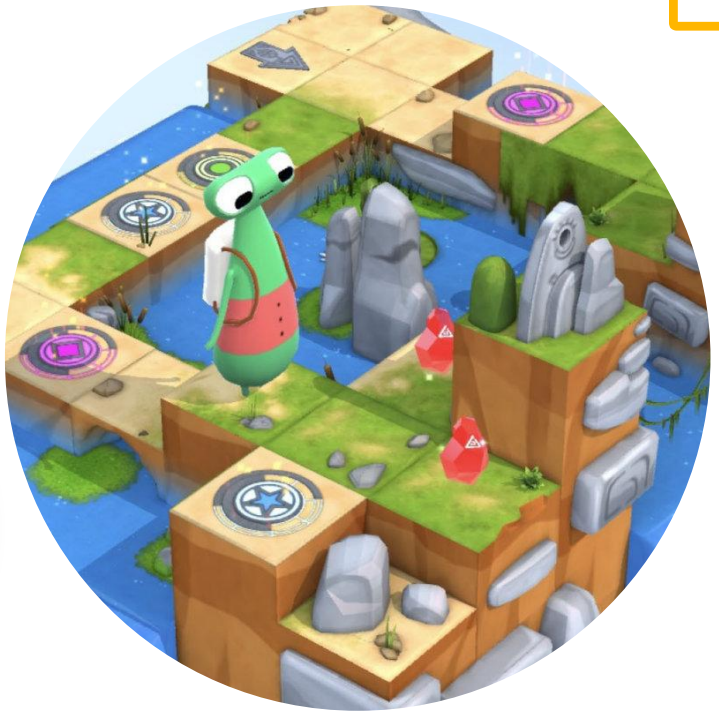
Hour of Code

Using Swift Playgrounds

Swift Playgrounds is a free app. It actively guides students through activities.

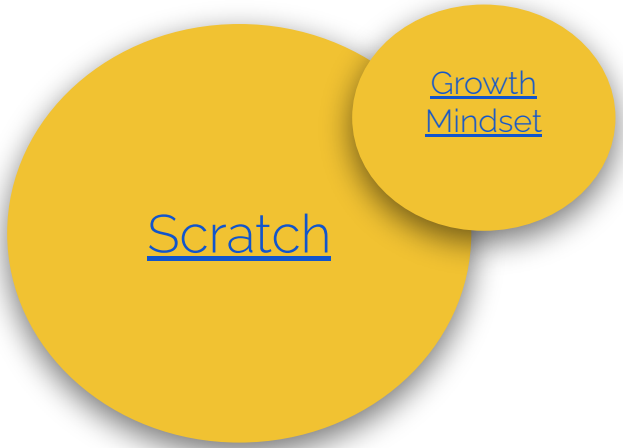
Apple: Everyone Can Code

Swift Playgrounds



What does it mean to programme in Scratch?

It's best to see something in action before we begin learning how.



Using Hopscotch

Hopscotch is another coding app that has a bunch of educator resources and students can work their way through challenges.

Hopscotch

Using Kodable

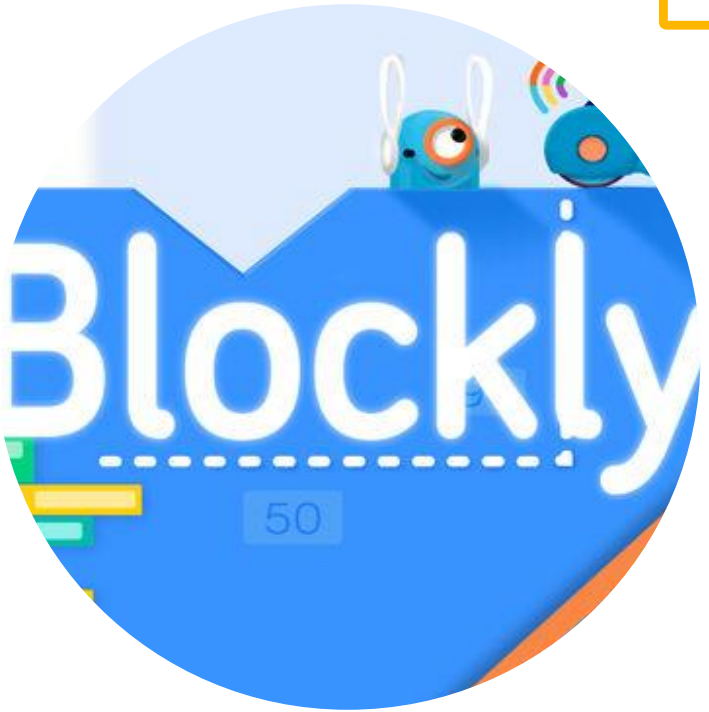
Kodable is a coding app that has some great free lessons. It is limited in order to get teachers to sign up to the Pro version which is expensive.

Kodable

Using Blockly

Kodable is a coding app that has some great free lessons. It is limited in order to get teachers to sign up to the Pro version which is expensive.

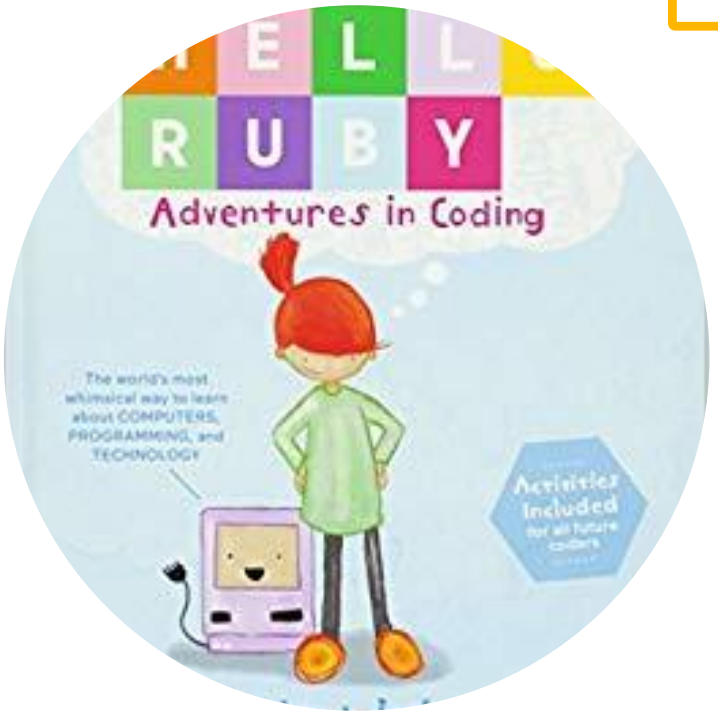
Blockly



Using Hello Ruby

Hello Ruby began as a book and then ventured into activities in coding.

Hello Ruby



Using Tynker

There is a free version of Tynker. It teaches coding

Tynker

4

Further Help...

There is a load of places for further teacher resources...

Digital Passport

Digital Technologies Curriculum

Fact Sheet for Teachers

PLD Roadmap

A Taranaki Cluster of teachers is creating a PLD Roadmap for schools and teachers.

Digtial Technology PLD Roadmap