

# DTiF

Digital Technologies in focus

Initiative of and funded by the Australian Government Department of Education, Skills and Employment

acara

AUSTRALIAN CURRICULUM,  
ASSESSMENT AND  
REPORTING AUTHORITY

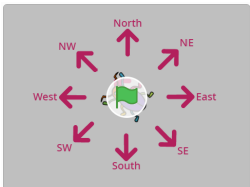
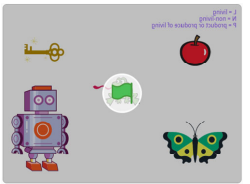
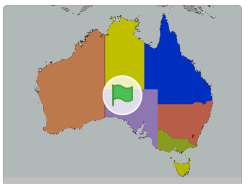
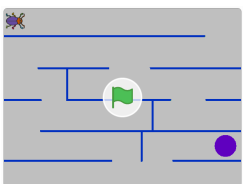


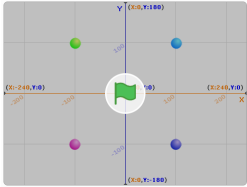
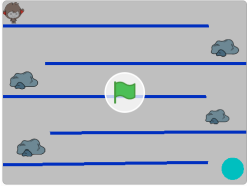

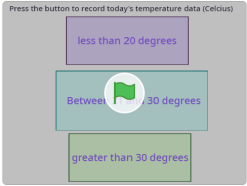


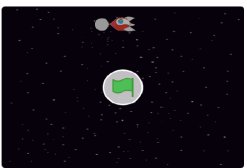
## CLASSROOM IDEAS: YEARS 3–6

### Visual programming with Scratch

The following sample activities (source: [scratch.mit.edu/users/saraheLearn/projects/](https://scratch.mit.edu/users/saraheLearn/projects/)) show a range of different ideas for incorporating visual programming into teaching and learning programs. The projects *are all incomplete* and designed to be used as samples for inspiration or modification by teachers. They show the possibilities Scratch offers for integration.

💡 Did you know Scratch is free, available in 40+ languages and has an F–2 version (ScratchJnr)?

Title of resource	Year levels	Description	Australian Curriculum learning area connections
<a href="#">Compass rose</a> 	3–4	Demonstrate understanding of the compass rose in an integrated Digital Technologies, Mathematics and Geography activity using broadcast messaging (where messages are sent between sprites to trigger animation effects).	Mathematics
<a href="#">Simple classifier</a> 	3–4	A simple quiz game example.	Science
<a href="#">Australian minerals</a> 	3–6	Designed as a way for students to demonstrate their knowledge of minerals in Australia.	Science HASS (Geography)
<a href="#">Buggy maze</a> 	3–6	An example of a maze game using responses to user input, repetition and variables. Test by using the up, down, left and right arrows on the keyboard or by using with a Makey Makey.	Science

<a href="#">Cartesian plane</a> 	3–6	Designed as a simple way to teach students about quadrants and Cartesian plane.	Mathematics
<a href="#">Rock cycle game</a> 	3–6	An example of a maze game designed for a Year 4 Science teacher. Test by using the up, down, left and right arrows on the keyboard or by using with a Makey Makey.	Science
<a href="#">Rocket experiment simulator</a> 	3–6	Designed for a multi-age class (Years 3–6) for students to explain the findings of their rocket experiment.	Science
<a href="#">Temperature data</a> 	5–6	This example was designed to simply explain how to validate data. For example, if there are 30 students in a class and more than 30 responses are received then the data are invalid because a button must have been pressed more than once. (For use with an IWB or Makey Makey to gather daily temperature data.)	Mathematics Science Design and Technologies (Food and fibre production)
<a href="#">Temperature-water</a> 	5–6	When a user adds temperature data (input), instructions are given as output on how much to water a 'plant in a cup'. This was designed as part of a food and fibre unit on sustainable farming practices.	Mathematics Science Design and Technologies (Food and fibre production)
<a href="#">Temperature-humidity-water</a> 	5–6	A second, more advanced temperature data example that requires modification to the code for it to function correctly.	Mathematics Science Design and Technologies (Food and fibre production)
<a href="#">Solar system quiz starter</a> 	5–6	An example of a quiz requiring user input to test knowledge of the solar system. This quiz can be adapted to test knowledge on any topic.	Science

## Links to the Australian Curriculum

Tables 1 and 2 give teachers an opportunity to see related aspects of the Australian Curriculum: Digital Technologies which may be addressed depending upon the task.

Table 1: Links to the Australian Curriculum: Digital Technologies Years 3–4

<b>Digital Technologies</b>  <b>Achievement standard</b>	By the end of Year 4 students create simple digital solutions and use provided design criteria to check if solutions meet user needs. Students process and represent data for different purposes. They follow and describe simple algorithms involving branching and iteration and implement them as visual programs. Students securely access and use digital systems and their peripherals for a range of purposes, including transmitting data. They use the core features of common digital tools to plan, create, locate and share content, and to collaborate, following agreed behaviours. Students identify their personal data stored online and recognise the risks.		
<b>Strand</b> <b>Sub-strand</b>	Digital Technologies knowledge and understanding <ul style="list-style-type: none"><li>• Digital systems</li><li>• Data representation</li></ul> Processes and production skills <ul style="list-style-type: none"><li>• Investigating and defining</li><li>• Generating and designing</li><li>• Producing and implementing</li><li>• Evaluating</li><li>• Collaborating and managing</li></ul>		
<b>Content descriptions</b>	<ul style="list-style-type: none"><li>• explore and describe a range of digital systems and their peripherals for a variety of purposes AC9TDI4K01</li><li>• recognise different types of data and explore how the same data can be represented differently depending on the purpose AC9TDI4K03</li><li>• define problems with given design criteria and by co-creating user stories AC9TDI4P01</li><li>• follow and describe algorithms involving sequencing, comparison operators (branching) and iteration AC9TDI4P02</li><li>• generate, communicate and compare designs AC9TDI4P03</li><li>• implement simple algorithms as visual programs involving control structures and input AC9TDI4P04</li><li>• discuss how existing and student solutions satisfy the design criteria and user stories AC9TDI4P05</li><li>• use the core features of common digital tools to create, locate and communicate content, following agreed conventions AC9TDI4P06</li></ul>		
<b>Technologies Core concepts</b>	<ul style="list-style-type: none"><li>• Systems</li><li>• Data</li><li>• Interactions and impact</li><li>• Systems thinking</li><li>• Computational thinking</li><li>• Design thinking</li><li>• Technologies process and production skills</li></ul>	<b>Digital Technologies Core concepts</b>	<ul style="list-style-type: none"><li>• Digital systems</li><li>• Data representation</li><li>• Abstraction</li><li>• Specification</li><li>• Algorithms</li><li>• Implementation</li></ul>
		<b>General capabilities</b>	<ul style="list-style-type: none"><li>• Digital Literacy</li><li>• Literacy</li><li>• Numeracy</li></ul>

	<ul style="list-style-type: none"> <li>• Project management skills</li> <li>• Enterprise skills and innovation</li> </ul>		
<b>Cross-curriculum priorities</b>		<b>Learning area or subject connections</b>	<ul style="list-style-type: none"> <li>• Mathematics*</li> <li>• Science*</li> <li>• Design and Technologies*</li> </ul>

\*Depending on the task

Table 2: Links to the Australian Curriculum: Digital Technologies Years 5–6

<b>Digital Technologies Achievement standard</b>	By the end of Year 6 students develop and modify digital solutions, and define problems and evaluate solutions using user stories and design criteria. They process data and show how digital systems represent data. Students design algorithms involving complex branching and iteration and implement them as visual programs including variables. They securely access and use multiple digital systems and describe their components and how they interact to process and transmit data. Students select and use appropriate digital tools effectively to plan, create, locate and share content, and to collaborate, applying agreed conventions and behaviours. They identify their digital footprint and recognise its permanence.		
<b>Strand Sub-strand</b>	Digital Technologies knowledge and understanding <ul style="list-style-type: none"> <li>• Digital systems</li> <li>• Data representation</li> </ul> Digital Technologies processes and production skills <ul style="list-style-type: none"> <li>• Investigating and defining</li> <li>• Generating and designing</li> <li>• Producing and implementing</li> <li>• Evaluating</li> <li>• Collaborating and managing</li> </ul>		
<b>Content descriptions</b>	<ul style="list-style-type: none"> <li>• investigate the main internal components of common digital systems and their function AC9TDI6K01</li> <li>• explain how digital systems represent all data using numbers AC9TDI6K03</li> <li>• define problems with given or co-developed design criteria and by creating user stories AC9TDI6P01</li> <li>• design algorithms involving multiple alternatives (branching) and iteration AC9TDI6P02</li> <li>• design a user interface for a digital system AC9TDI6P03</li> <li>• generate, modify, communicate and evaluate designs AC9TDI6P04</li> <li>• implement algorithms as visual programs involving control structures, variables and input AC9TDI6P05</li> <li>• evaluate existing and student solutions against the design criteria and user stories and their broader community impact AC9TDI6P06</li> <li>• select and use appropriate digital tools effectively to create, locate and communicate content, applying common conventions AC9TDI6P07</li> </ul>		
<b>Technologies Core concepts</b>	<ul style="list-style-type: none"> <li>• Systems</li> <li>• Systems thinking</li> <li>• Computational thinking</li> </ul>	<b>Digital Technologies Core concepts</b>	<ul style="list-style-type: none"> <li>• Digital systems</li> <li>• Data representation</li> <li>• Abstraction</li> <li>• Specification</li> </ul>

	<ul style="list-style-type: none"> <li>• Data</li> <li>• Technologies processes and production skills</li> <li>• Interactions and impact</li> <li>• Project management skills</li> <li>• Enterprise skills and innovation</li> </ul>		<ul style="list-style-type: none"> <li>• Algorithms</li> <li>• Implementation</li> </ul>
		<b>General capabilities</b>	<ul style="list-style-type: none"> <li>• Digital Literacy</li> <li>• Literacy</li> <li>• Numeracy</li> </ul>
<b>Cross-curriculum priorities</b>		<b>Learning area or subject connections</b>	<ul style="list-style-type: none"> <li>• Mathematics*</li> <li>• Science*</li> <li>• Design and Technologies*</li> </ul>

\*Depending on the task

## Acknowledgement

Images used in this resource are from the [Scratch 3.0 website](#) and the projects depicted were created by [@sarahelearn](#)..

## Useful links

Australian Curriculum: Digital Technologies F–10

[https://v9.australiancurriculum.edu.au/f-10-curriculum.html/learning-areas/digital-technologies/foundation-year\\_year-1\\_year-2\\_year-3\\_year-4\\_year-5\\_year-6\\_year-7\\_year-8\\_year-9\\_year-10](https://v9.australiancurriculum.edu.au/f-10-curriculum.html/learning-areas/digital-technologies/foundation-year_year-1_year-2_year-3_year-4_year-5_year-6_year-7_year-8_year-9_year-10)

Scratch website [www.scratch.mit.edu/](http://www.scratch.mit.edu/)

- A guide for teachers on ScratchJr blocks [www.scratchjr.org/learn/blocks](http://www.scratchjr.org/learn/blocks)
- Scratch tutorials <https://scratch.mit.edu/projects/editor/?tutorial=getStarted>
- Resources for teachers [www.scratch.mit.edu/educators/#resources](http://www.scratch.mit.edu/educators/#resources)

*All images in this resource used with permission or under Creative Commons (CC) licensing.*