Digital Technologies – 5 and 6_ Data representation

	STRAND		Knowledge and understanding				Processes and production skills													
			Digital Systems		Representation of data		Collecting, managing and analysing data		*Creating Digital Solutions by:*											
									Investigating and defining		Generating and designing				Producing and implementing		Evaluating		Collaborating and managing	
	Content Description		Examine the main components of common digital systems and how they may connect together to form networks to transmit data (ACTDIK014)		Examine how whole numbers are used to represent all data in digital systems (ACTDIK015)		Acquire, store and validate different types of data, and use a range of software to interpret and visualise data to create information (ACTDIP016)		Define problems in terms of data and functional requirements drawing on previously solved problems (ACTDIP017)		Design a user interface for a digital system (ACTDIP018)		Design, modify and follow simple algorithms involving sequences of steps, branching, and iteration (repetition) (ACTDIP019)		Implement digital solutions as simple visual programs involving branching, iteration (repetition), and user input (ACTDIP020)		Explain how student solutions and existing information systems are sustainable and meet current and future local community needs (ACTDIP021)		Plan, create and communicate ideas and information, including collaboratively online, applying agreed ethical, social and technical protocols (ACTDIP022)	
Sequence of Lessons / Unit	**Approx. time rq'd (hrs)**	**Year 5or6**	CD	Achievement standard #	CD	Achievement standard #	CD	Achievement standard #	CD	Achievement standard #	CD	Achievement standard #	CD	Achievement standard #	CD	Achievement standard #	CD	Achievement standard #	CD	Achievement standard #
Representing images using binary	6hrs	6	☐		☑	2	☑	6	☐		☐		☐		☐		☐	7	☐	

Levels 3 and 4 Achievement Standard	**Levels 5 and 6 Achievement Standard** The numbering of the Achievement Standards below is reflected in the grid above to show coverage across the 8 units.	**Levels 7 and 8 Achievement Standard**
By the end of Year 4 • Students describe how a range of digital systems (hardware and software) and their peripheral devices can be used for different purposes • They explain how the same data sets can be represented in different ways. • Students define simple problems, design and implement digital solutions using algorithms that involve decision-making and user input. • They explain how the solutions meet their purposes. • They collect and manipulate different data when creating information and digital solutions. • They safely use and manage information systems for identified needs using agreed protocols and describe how information systems are used.	By the end of Year 6: • Students explain the fundamentals of digital system components (hardware, software and networks) and how digital systems are connected to form networks. (1) • They explain how digital systems use whole numbers as a basis for representing a variety of data types. (2) • Students define problems in terms of data and functional requirements and design solutions by developing algorithms to address the problems. (3) • They incorporate decision-making, repetition and user interface design into their designs and implement their digital solutions, including a visual program. (4) • They explain how information systems and their solutions meet needs and consider sustainability. (5) • Students manage the creation and communication of ideas and information in collaborative digital projects using validated data and agreed protocols. (6)	By the end of Year 8 • Students distinguish between different types of networks and defined purposes. • They explain how text, image and audio data can be represented, secured and presented in digital systems. • Students plan and manage digital projects to create interactive information. • They define and decompose problems in terms of functional requirements and constraints. • Students design user experiences and algorithms incorporating branching and iterations, and test, modify and implement digital solutions. • They evaluate information systems and their solutions in terms of meeting needs, innovation and sustainability. • They analyse and evaluate data from a range of sources to model and create solutions. • They use appropriate protocols when communicating and collaborating online.

Representing images using binary

Year Level 6 TOPIC Data representations Time: 6 HOURS

Digital images are encoded in binary. Students learn about pixels and the way computers store an image as an array of individual pixels, each of which has a particular colour. Students make connections between the amount of data used by the computer to store, transmit and create an image and the file size of different images (generally, the more pixels and more bits to encode the image, the larger the file size).

Flow of activities				
Short text	Coding instructions Create instructions to shade a grid to explore encoding.	Coding an image using binary Create instructions to shade a grid made up of pixels encoding using 0 or 1.	Coding a colour image Explore the use of RGB represented as three binary digits to create colours.	Create an image using binary Students apply their understanding about how binary is used to create an image.
Question to guide exploration	*How can you use code to represent instructions?*	*How can I encode a black and white image using binary?*	*How can I encode a colour image using binary?*	*How can I design my own pixel image?*
AC Alignment	*Representation of data (ACTDIK015)*	*Representation of data (ACTDIK015)* *Collecting, managing and analysing data (ACTDIP016)*	*Representation of data (ACTDIK015)* *Collecting, managing and analysing data (ACTDIP016)*	*Representation of data (ACTDIK015)* *Collecting, managing and analysing data (ACTDIP016)*
What's this about?	A grid is a simple way of representing picture elements (pixels). Each square can represent a pixel. Creating instructions to shade a grid gives students an understanding of the ways to represent the square: shaded or not shaded (on/off). On/off state is the basis of the binary system.	Students create an image made up of only black and white squares. This task introduces them to how binary numbers are used to represent images. 0 represents black and 1 represents white. One binary digit is a bit. This is an example of colouring 1 bit per pixel; it is either black or white. There is a direct relationship between the width and height of the image and the number of pixels and image detail.	On a computer screen a colour image is made by mixing red, green and blue (RGB) light. This mix of light can be represented in binary in three digits. For example, (0,0,0) is no red, no green and no blue (making black). White is represented as (1,1,1). This is an example of RGB colouring 3 bits per pixel and results in 8 colours.	Students apply their understanding about how binary is used to create an image. They create their own image for a particular purpose by encoding a grid of pixels.
Learning tasks	Introduce the task of shading a grid made up of squares, for example 5 x 5. One student shades the square then comes up with their own code to explain to a partner how to replicate the pattern of shaded squares. Students choose from a variety of ways to encode their pattern. There are many ways to represent the command of shading a square: a letter B for black, a scribble, a pen icon, a plus sign, a number.	Provide students with the task of creating an image based on either a black or white pixel. If doing this as an unplugged activity students work in pairs to encode their own image using 0s and 1s and swap with their partner to decode and then compare to the original image. Alternatively, use relevant online tools such as Code.org's pixilation tutorial.	Students interested in finding out how to create colours using binary numbers can progress through the Code.org pixilation tutorial, and the colour tutorial. Challenge students to create a colour image using up to 8 colours. Explore changing the width and length of the image and the relationship between image detail (quality) and the number of pixels.	Set a task of creating a pixel image for a particular purpose; for example, create a character to import into a visual programming software such as Scratch; an avatar to use in safe online environments; a card for a celebration; or a symbol to convey a message to be displayed around the school. Collectively establish the criteria and evaluate the solutions.
Supporting resources and tools and how to use them	Introducing algorithms Use the 'Dare to square' task of the lesson idea. Students encode and decode the shading of a grid. Adapt this activity to focus on ways to represent the command to code the square being shaded.	Using binary to create on/off pictures This lesson idea introduces students to representing images using 0 and 1. Black and white pixilation tutorial Explore creating black and white images using Code.org.	Using binary to create on/off pictures Use the learning demo part of the lesson to explore how to create a colour image using RGB, represented in binary as 3 bits. Colour pixilation tutorial Explore creating pixel images using this Code.org tutorial.	Colour pixilation tutorial Design and create a pixel image using this Code.org tutorial. How to do pixel art with MS Windows 7 Paint View a YouTube tutorial showing how to use a paint program to create an image.
Assessment	**Checklist** **NOTE: By year 4, Explain** how the same data can be represented in different ways.	• Artefact analysis: Represent an image using 1 bit/pixel (0 or 1) • Explain using Think Aloud **Achievement standard** **Explain** how digital systems use whole numbers as a basis for representing a variety of data types.	• Artefact analysis: Represent an image using 3 bits/pixel (RGB) • Explain using Think Aloud **Achievement standard** **Explain** how digital systems use whole numbers as a basis for representing a variety of data types.	• Rubric: Scope for a range of abilities, skills and understanding (creativity), based on solo taxonomy (below) **Achievement standard** **Explain** how digital systems use whole numbers as a basis for representing a variety of data types.