Scope and sequence

Digital Technologies

Year 7 to 10

v1.0

September 2020

Government of South Australia
Department for Education

Digital technologies: Scope and sequence year 7 to 10

Contents

Context statement

Digital Technologies is about exciting interest in computer science. It is about empowering students to understand how the devices they use, actually work. Students will develop the confidence to create their digital solutions.

Computational, systems and design thinking are fundamental to Digital Technologies learning.

Computational thinking is a problem-solving process involving:

- Pattern Recognition: understanding trends, similarities and patterns in data to define problems
- Decomposition: breaking down complex problems into simpler parts
- Abstraction: identifying and removing unnecessary details to simplify a problem.
- Algorithmic design: creating a step by step solution to a problem

Systems thinking is the ability to see the big picture. Students need to understand the impact of digital solutions. They need to see the solution through legal, ethical and sustainability lenses.

Design thinking is the process of imagining, creating and realising solutions.

When using these thinking strategies together, students can develop powerful digital solutions.

This document:

- provides explicit plain English interpretation of the Australian Curriculum content descriptors
- identifies the specific knowledge, skills and understanding learners need at each year level
- guides educators to teach and model computational, systems and design thinking
- supports educators with the processes to design, create and produce digital solutions

Achievement standards

Years 7 to 8	Years 9 to 10
By the end of year 8, students: • **distinguish** between different types of networks and defined purposes. • **explain** how text, image and audio data can be represented, secured and presented in digital systems. • **plan** and **manage** digital projects to create interactive information. • **define** and **decompose** problems in terms of functional requirements and constraints. • **design** user experiences and algorithms incorporating branching and iterations, and test, modify and implement digital solutions. • **evaluate** information systems and their solutions in terms of meeting needs, innovation and sustainability. • **analyse** and **evaluate** data from a range of sources to model and create solutions. • **use** appropriate protocols when communicating and collaborating online.	By the end of year 10, students: • **explain** the control and management of networked digital systems and the security implications of the interaction between hardware, software and users. • **explain** simple data compression, and why content data are separated from presentation. • **plan** and **manage** digital projects using an iterative approach. • **define** and **decompose** complex problems in terms of functional and non-functional requirements. • **design** and **evaluate** user experiences and algorithms. • **design** and **implement** modular programs, including an object-oriented program, using algorithms and data structures involving modular functions that reflect the relationships of real-world data and data entities. • **take** account of privacy and security requirements when **selecting** and **validating** data. • **test** and **predict** results and implement digital solutions. • **evaluate** information systems and their solutions in terms of risk, sustainability and potential for innovation and enterprise. • **share** and **collaborate** online, establishing protocols for the use, transmission and maintenance of data and projects.

Strand: Knowledge and Understanding

Thread	Year 7	Year 8	Year 9	Year 10
Digital systems Focuses on the components of digital systems: hardware, software and networks. In the early years, students learn about a range of hardware and software and progress to an understanding of how data are transmitted between components within a system, and how the hardware and software interact to form networks.	Data is transmitted between connected digital systems using different wired, wireless and mobile medium. Data can be stored and transmitted between networked systems. • **Recognise** hardware components in their home/school that enables them to connect to: ○ Printer ○ Internet. • **Understand** the materials used to transmit data within a digital system and between them, as well comparing reliability and speed. For example: ○ circuit boards ○ copper cables ○ fibre cables ○ radio waves. • **Understand** that data travelling on a network needs to know where it is going. This could include knowing about: ○ IP Addresses – Internet Protocol Addresses ○ DNS – Domain Name System ○ URLs – Uniform Resource locator. • **Investigate** and **Compare** digital systems in particular: ○ RAM Storage ○ Data Storage ○ Processor speed. • **Understand** how data is stored on a range of mediums: ○ USB, SSD and Digital circuits (including cloud) ○ DVD ○ Hard Disk Drives (including cloud).	Data is transmitted and secured in wired, wireless and mobile networks, and different hardware specifications can affect the performance of particular tasks. • **Investigate** how bandwidth affects the speed of downloads. • **Realise** that data is transmitted via different paths depending on the network configuration and required purposes: ○ LAN ○ PAN ○ MAN ○ WAN or Mesh. • **Discuss** the different communications protocols for transmitting data in networks and how they operate, for example: ○ hypertext transfer protocol (HTTP) ○ file transfer protocol (FTP) ○ transmission control protocol/internet protocol (TCP/IP) ○ simple mail transfer protocol (SMTP). • **Investigate** the purpose of key computer components and their impact on performance. ie: ○ 8Mb vs 16Gb of RAM ○ Hard Disk Drive (HDD) Vs Solid State Drive (SSD) ○ 1Mbps vs 1Gbps ○ Other.	The hardware and software components in networked systems are used to manage, control and secure the movement of data. • **Investigate** how an operating system manages the relationship between hardware, applications and system software. • **Realise** the purpose of the hardware in a known network: ○ home ○ school ○ other. • **Discuss** how the network components that control the movement of data operate in regards to security and speed, for example: ○ routers ○ hub ○ switches and bridges.	Hardware and software are developed to manage, control and secure the movement of and access to data in networked digital systems. • **Investigate** the role of a network operating system and its ability to manage ○ users (passwords) ○ groups ○ permissions. • **Investigate** how routers learn to move packets of data securely around a network. • **Discuss** a range of systems (process and procedures) that organisations using a network should implement to ensure systems are kept safe and secure. For example: ○ password and group permissions ○ location of digital systems that store private/confidential data ○ confidential agreements for staff and clients. • **Investigate** why and how different types of encryptions work. For example: ○ secret keys and 'exclusive or' (XOR) ○ hashing algorithms. • **Understand** the seven layers of the OSI model and what role each layer performs in regards to network communication.
Representation of data Focuses on how data are represented and structured symbolically for use by digital systems. Different types of data are studied in the bands including text, numeric, images (still and moving) and sound. Categorical and relational data should be understood in Year 9 and 10.	Text, images and sound are encoded and can be decoded into binary digits. • **Investigate** how text is represented, secured and presented in digital systems. For example that a character is represented by a byte and text by a series of bytes. • **Investigate** file size and its relationship to text. For example: ○ 8 bits = 1 byte ○ 1 byte = 1 character (ASCII) ○ 1024 Bytes = 1 Kilobyte ○ 1024 Kb = 1 Megabyte. • **Discuss** the difference between a Bitmap and vector image.	Digital systems represent text, image and audio data in binary. • **Investigate** how text is represented, secured and presented in digital systems. For example, decode a binary message. • **Investigate** how an image is represented, secured and presented in digital systems, can include: ○ Black and white images as binary ○ Colour images as binary. • **Investigate** how audio is converted from analogue to digital, students should know: ○ Amplitude and Wavelength ○ Sample rate, Bitrate and their implications on quality and file size.	A variety of data types can be compressed by removing redundancy and repetition but maintain quality. • **Examine** and **explain** why the size of files used every day differs depending on the software or saved format used, For example: ○ txt vs Docx ○ CSV vs Xls ○ BMP vs jpeg. • **Analyse** how an image is represented when the number of colours increases and the use of Hexadecimal to solve file size issues. • **Examine** how folders can be made smaller through a process called Zipping. • **Understand** run-length Encoding.	Data can be compressed to conserve digital memory without loss of information. Data can be described by its content and how it is presented. • **Analyse** how Lossy (JPG/MP3) and lossless file (PNG/TIFF/RAW/Codec) compression work. • **Consider** data as content and alter ways it can be presented, for example: ○ **Apply** the use of Cascading Style sheets (CSS) in web programming. ○ **Organise** data as information (table, analysis, chart) using a database (links to Collecting, Managing and Analysing Data Strand).

Strand: Process and Production Skills - Creating solutions using design and computational thinking

Thread	Year 7	Year 8	Year 9	Year 10
Collecting, managing and analysing data Understand the nature and properties of data, how it is collected and interpreted using a range of digital systems and peripheral devices and interpreting data to create information. **Pattern recognition:** look for similarities between and within data. **Decomposition**: break information down into smaller components of data to make management easier. **Abstraction:** identify and remove irrelevant data that can be created another way (ie Age = Date Now – DOB). **Algorithmic design:** identify and organise the data so it can solve the problem or create information.	Acquire data from a range of sources and use software to record data in ways that allow it to be easily accessed and manipulated. • **Acquire** and **organise** data from a range of sources, primary and secondary, for example: ○ people ○ websites ○ probes ○ data repositories. • **Generate** search engine queries to gather information, evaluating results: ○ on the URL ○ author ○ age of the content. • **Design** user interface for data collection form to make collection easier to manage for example: • Radio Buttons • pull down options. Organise and visualise data using a range of software and model objects and events according to their attributes. • **Understand** that data is stored in a table and a cell is data but a row is information. • **Identify** statistical information from tables of data by using: ○ Sum ○ Count ○ Min ○ Max ○ Average. • **Explore** software to create visual representations of data.	Acquire data from a range of sources and evaluate authenticity, accuracy and timeliness. • **Acquire** and **organise** data from a range of primary and secondary sources • **Generate** advance search engine queries, using Boolean language, to gather information and test for: ○ relevance ○ accuracy ○ bias/perspective ○ reliability. Analyse and visualise data using a range of software to create information, and use structured data to model objects or events. • **Realise** that to ensure integrity in a table of data, it needs to include: ○ field names ○ types of data ○ sizes ○ keys – unique identifiers ○ required values and the appropriate validation rules ○ default values ○ formats ○ input masks. • **Manipulate** data tables to find and sort information • **Understand** how data can be filtered using Boolean searches: ○ And ○ Or ○ Not. • **Display** data in a range of formats.	Investigate quantitative and qualitative data that can be collected in ethical ways and how this data is used to determine patterns and trends. • **Investigate** ways to capture accurate and usable data of different formats • **Discuss** the difference between quantitative and qualitative data • **Create** a system to store quantitative and qualitative data • **Investigate** different organisations and the data they hold and the methods they use to collect this data, ie Social Media, Shopping reward cards, etc. Analyse and visualise data to create information and address problems, using structured data to model processes and their relationships. • **Create** queries to gather required data. • **Organise** sets of data visually to create information.	Develop techniques for acquiring, storing and validating quantitative and qualitative data from a range of sources, considering privacy and security requirements. • **Identify** strengths and weaknesses of collecting data using different methods • **Create** a system to collect and store quantitative and qualitative data • **Construct** an electronic form that gathers quantitative and qualitative data that validates the data being entered • **Discuss** an organisation's data collection and storage processes considering how secure it is and how they ensure the data is kept private. • **Show** that two or more tables of data may be used to find information, through unique identifiers (primary key). Analyse and visualise data to create information and address complex problems, and model processes, entities and their relationships using structured data. • **Create** complex queries to gather the required information from joined tables of data. • **Organise** sets of data visually to create information.
Investigating and defining **Pattern recognition**: Organise information so that similarities can be easily seen and problems identified. **Decomposition**: break the current problems down into simpler, less complex steps. **Abstraction:** identify and remove irrelevant details. **Algorithmic design**: identify and organise the current system into steps.	Break down and define authentic problems in terms of requirements and constraints including consideration of stakeholder needs identified through data collection and analysis. • **Define** and **document** a given need or problem that considers: ○ What do we want the solution to achieve? ○ Who is it for? ○ Why is it required? • **Identify** the key steps in the current system and where problems might be occurring ○ How do you want it to work or how does it currently work?	Define and decompose real-world problems taking into account functional requirements and economic, environmental, social, technical and usability constraints. • **Define** and **document** a given need or opportunity for a specific problem that considers: ○ the situation – What is the problem with the current system? ○ what do we want the solution to achieve? ○ is the new solution likely to save time and money? • **Identify** the key steps in the current system, and where problems might be occurring.	Define and decompose real-world problems, taking into account requirements and constraints including economic, environmental, social, and technical and usability concerning identified stakeholder needs. • **Define** and **document** a current or proposed system to identify a user's need to enable a product to be developed that satisfies their requirements. Needs to include: ○ problem identification ○ expected outcomes ○ available data ○ user interfaces ○ constraints. • **Document** and **define** the processes/steps involved in moving from inputs to achieve desired outputs.	Define and decompose real-world problems precisely, taking into account functional and non-functional requirements and including interviewing stakeholders to identify needs. • **Define** and **document** a current or proposed system to identify a user's needs to enable a product to be developed that satisfies their requirements. Needs to include: ○ problem identification ○ expected outcomes/outputs ○ available data ○ user interfaces ○ constraints. • **Document** and **define** the processes/steps involved in moving from inputs to achieve desired outputs.

Strand: Process and Production Skills - Creating solutions using design and computational thinking

Thread	Year 7	Year 8	Year 9	Year 10
Generating and designing **Pattern recognition**: look for repetitive steps. **Decomposition**: break the current problems down into simpler, less complex parts. **Abstraction**: remove unnecessary details to identify a more efficient set of steps. **Algorithmic design**: organise into a specific sequence of steps.	Design and evaluate a user interface for a digital system taking into account the user experience and feedback. • **Evaluate** a range of user interfaces identifying specific features and purposes. Identify the necessary features: ○ Input – buttons, text ○ Output – scores ○ Calculated – age. • **Create** a user's experience for a digital solution seeking feedback ideas. Design diagrammatic representations of algorithms or modify existing ones and test to fix errors or change functionality. • **Develop** a step by step sequence based on initial investigation, a structure or flow chart that includes branching and iteration. • **Document** and **test** a range of inputs against an algorithm showing predicted outcome and the actual outcome.	Design the user experience of a digital system, generating, evaluating and communicating alternative designs. • **Design** a user interface for a digital system including seeking feedback. Consider features like: ○ input ○ output ○ calculations. Design algorithms represented diagrammatically and in English, and trace algorithms to predict output for a given input and to identify errors. • **Design** a documented process showing steps broken down as a series of steps written in sequence, which includes branching and iteration, to define the solution to a problem - an algorithm. Expressed as: ○ pseudocode ○ structure charts or flowcharts. • **Document** and **test** a range of inputs in a specified algorithm showing predicted outcome and the actual outcome.	Design the user experience of a digital system by evaluating alternative designs against criteria including functionality, and usability. • **Create**, **evaluate** and **modify** the user interface and experience to meet the stakeholders' needs in regards to initial investigation, focussing on: ○ performance ○ ease of use ○ appeal. Design algorithms represented diagrammatically and in structured English and predict outputs for a given input and to identify errors. • **Design** a documented process showing steps broken down as a series of steps written in sequence, which includes branching and iteration, to define the solution to a problem - an algorithm. It can be expressed in: ○ pseudocode ○ structure charts or flowcharts ○ Nasi-Schnider Diagrams.	Design the user experience of a digital system by evaluating alternative designs against criteria including functionality, accessibility, usability, and aesthetics. • **Create**, **evaluate** and **modify** the user interface and experience to meet the stakeholders' needs in regards to initial investigation, focussing on: ○ performance ○ ease of use ○ appeal. Design algorithms represented diagrammatically and in structured English and validate algorithms and programs through tracing and test cases. • **Design** a documented process showing steps broken down as a series of modules and written in sequence, which includes branching and iteration, to define the solution to a problem - an algorithm. It can be expressed in: ○ pseudocode ○ structure charts or flowcharts ○ Nasi-Schnider diagrams. • **Design** procedures/modules that break the problem into manageable parts.
Producing and implementing **Pattern recognition**: looking for similarities between code. **Decomposition:** break the coding into simpler, less complex parts – use functions. **Abstraction**: identify more efficient ways of writing code. **Algorithmic design:** organise code into functions.	Implement and modify digital solutions with user interfaces involving branching, iteration and functions using a programming language. • **Construct** programs from algorithms that include branching, iteration and functions. • **Document** this process with annotated images while **testing** and **modifying** their solution.	Implement and modify programs with user interfaces involving branching, iteration and functions in a general-purpose programming language. • **Construct** programs from specified algorithms that include branching, iteration and functions. • **Document** this process with annotated images while **testing** and **modifying** their solution.	Implement and modify programs using an object-oriented programming language, applying selected algorithms and data structures. • **Construct** programs from algorithms using procedures and/or user-defined functions. • **Recognise** types of errors in terms of their effect on the running of the program and investigate methods to reduce the occurrence of errors (eg tracing and debugging). • **Produce** the output of a program in a user-friendly way, including documentation on how it operates.	Implement modular programs, applying selected algorithms and data structures including using an object-oriented programming language. • **Construct** programs from algorithms using procedures and/or user-defined functions. • **Consider** solution is correct by desk-checking for errors. • **Recognise** types of error in terms of their effect on the running of the program and investigate methods to reduce the occurrence of errors (eg tracing and debugging). • **Produce** the output of a program in a user-friendly way.

Strand: Process and Production Skills - Creating solutions using design and computational thinking

Thread	Year 7	Year 8	Year 9	Year 10
Evaluating **Pattern recognition:** looking for similarities between and within problems **Decomposition**: Did we properly identify all the problems? Is it easily understood? **Abstraction**: Did it solve every aspect of the problem? Or is it incomplete? (ie some parts of the problem may have been left out accidentally) **Algorithmic design**: Is it fit for purpose? Did it solve the problem, making the best use of the available resources and meet the design criteria?	Evaluate how student solutions and existing information systems are sustainable, innovative and meet current and future community needs. • **Evaluate** and **compare** the initial investigation and the final solution against desired outcomes. • **Consider** recommendations for further improvement. • **Compare** solution with existing products (including those developed by other students) identifying innovations.	Evaluate how student solutions and existing information systems meet needs, are innovative, and take account of future risks and sustainability. • **Evaluate** and **compare** the initial investigation and the designed solution against desired outcomes • **Consider** recommendations for further improvement. • **Compare** solution with existing products (including those developed by other students) identifying innovations.	Evaluate how student solutions and existing information systems, take account of future risks and sustainability and provide opportunities for innovation and enterprise. • **Evaluate** and **compare** the initial investigation against the designed solution focussing on desired outcomes • **Determine** a set of recommendations for further improvement. • **Develop** a marketing campaign for a digital solution to increase more widespread adoption.	Evaluate critically how student solutions and existing information systems and policies, take account of future risks and sustainability and provide opportunities for innovation and enterprise. • **Produce** documentation for a digital solution including instructions for operation and guidelines for its implementation that reduce risks. For example: ○ password protection ○ storage and location of devices. • **Evaluate** and **compare** the initial investigation against the final solution focusing on desired outcomes. • **Determine** a set of recommendations for further improvement. • **Develop** a marketing campaign for a digital solution to increase more widespread adoption.
Collaborating and managing Develop skills to manage projects to successful completion through planning, organising and monitoring timelines, activities and the use of resources. This includes considering resources and constraints to develop the resource, finance, work and time plans; assessing and managing risks; making decisions; controlling quality; evaluating processes and collaborating and communicating with others at different stages of the process.	Identify the key steps for managing a project that communicates ideas and information. Identify, develop and use protocols for working collaboratively online. • **Create** a solution that communicated ideas and information. • **Use** a system development life cycle process to complete a project. • **Demonstrate** the skills needed to work collaboratively online considering: ○ safety ○ social and ethical protocols. • **Use** techniques to communicate and present information or ideas online using a collaborative tool. • **Explain** how roles were assigned, timelines managed and outcomes achieved when working on a project. • **Evaluate** the contribution students make as an individual to teamwork.	Plan and manage projects that create and communicate ideas and information collaboratively online, taking safety and social contexts into account. • **Create** a solution that communicates ideas and information. • **Use** a system development life cycle process to complete a project. • **Demonstrate** the skills needed to work collaboratively online considering: ○ safety ○ social. • **Use** techniques to communicate and present information or ideas online using a collaborative tool. • **Explain** how roles were assigned, timelines managed and outcomes achieved when working on a project. • **Evaluate** the contribution they make as an individual to teamwork.	Investigate and develop a range of interactive solutions for sharing ideas and information online, considering safety, security, social and legal contexts. • **Create** an interactive solution for sharing ideas and information that considers: ○ safety ○ security ○ social ○ legal responsibilities. • **Practice** techniques for sharing ideas and information online. • **Determine** how to assign team roles. • **Evaluate** the contribution they make as an individual to teamwork. • **Use** a system development life cycle process to manage a project collaboratively.	Create interactive solutions for sharing ideas and information online, taking into account safety, social contexts and legal responsibilities. • **Create** an interactive solution for sharing ideas and information that considers: ○ safety ○ privacy and security ○ social contexts ○ legal responsibilities. Plan and manage projects using an iterative and collaborative approach, identifying risks and considering safety and sustainability. • **Use** a system development life cycle process to complete a project. • **Practice** techniques for sharing ideas and information online. • **Establish** and build positive relationships with others. • **Determine** how to assign team roles. • **Evaluate** the contribution they make as an individual to teamwork. • **Use** a system to track progress and deadlines of a project. • **Document** changes in design during the producing and implementation stage should include tester's feedback.