Digital Technologies Glossary

Abstraction

The process of reducing complexity to formulate generalised ideas or concepts, for example reducing a computing problem to its fundamental concepts.

Algorithm

A description of the steps and decisions required to solve a problem. For example, to find the largest number in a list of positive numbers:

1. Note the first number as the largest
2. Look through the remaining numbers, in turn, and if a number is larger than the number found in 1, note it as the largest.
3. Repeat this process until complete. The last noted number is the largest in the list.

Flowcharts are often useful in visualising an algorithm.

Binary

The use of two states or permissible values to represent data*,* such as the on and off position of a light switch or the transistors in a computer silicon chip that can be in either the electrical state of ON or OFF.

Binary data are typically represented as a series of single digits referred to as binary digits (or bits) due to each taking on the value of either 0 or 1. For example, if we wanted to represent a four-colour system (e.g. CMYK - cyan, magenta, yellow, and key [black]) in binary the two-digit codes 00, 01, 10 and 11 could be used.

Branching

Branching occurs when an algorithm makes a choice to do one of two or more actions depending on sets of conditions and the data provided.



Components

Parts or elements that make up a whole and that perform a function. The components of a digital system may be a central processing unit (chips that follow instructions to control other components and move data); memory chips and a hard disk (for storing data and instructions); keyboard, mouse, camera and microphone (to input instructions and data for the central processing unit); screen, printer, and speakers (to output data); USB and ethernet cards (to communicate with other systems or components).

Compression

The process of encoding information using fewer bits (ie 0 or 1) than the original representation to reduce file size – typically using mathematical formulae to remove repeated data, combine related data or simplify data (e.g. a line segment can be represented by the position of the endpoints instead of every dot on it). Common examples include:

* .zip files which can contain one or more files or folders that have been compressed
* .jpg files in digital photography are produced by processing the complete (lossless) data from the camera’s sensor through compressing (looking for redundant/unnecessary data) into a smaller file size
* .mp3 files for audio which compress the original audio source to reduce the file size significantly but still sound like an exact copy of the original.

Computational thinking

A problem-solving method that involves various techniques and strategies in order to solve problems that can be implemented by digital systems, such as organising data logically, breaking down problems into components, and the design and use of algorithms, patterns and models.

Data

Data refers to the discrete representation of information using number codes. Data may include characters (e.g. alphabetic letters, numbers and symbols), numbers, images, sounds and/or instructions that when represented by number codes can be manipulated, stored and communicated by digital systems.For example, characters may be represented using ASCII code or images may be represented by a bitmap of numbers representing each ‘dot’ or pixel***.***

Data states

Classifications of data based on its level of activity. There are three basic data states: data at rest (stored, inactive data as it is not being transmitted, read or processed), data in motion (data being transmitted across a network) and data in use (data that is currently being processed or changed).

Database

A collection of data organised by records and fields that can be easily stored, accessed, managed and updated. Each discrete piece of data to be stored is represented by a field (e.g. song title, song artist or bank account number, date transaction,) and the values in the fields that are associated with an entity (e.g. a song, a bank transaction) are a record. Interaction with the database usually takes place through a user interface designed specifically for the structure and use of the data stored in it.

Decompose

Separate a complex problem into parts to allow a problem to be more easily understood. Decomposition involves breaking down a problem into simpler, less complex components that allow it to be better understood. For example, to create an interactive story the problem could be decomposed to a list of characters and their characteristics (e.g. clothing), the actions the characters make, the backdrops and the sequence of scenes with reference to which characters, actions and backdrops are involved in each scene.

Design thinking

Purposeful use of strategies for understanding design problems and opportunities, visualising and generating creative and innovative ideas, and analysing and evaluating those ideas that best meet the criteria for success and planning. Designing stems from the notion that current products, processes, systems or services are either unsuitable for our needs or can be improved.

Digital solution

The result (or output) of transforming data into information using digital systems, skills, techniques and processes to meet a need or opportunity.

Digital systems

Digital hardware and software components (internal and external) used to transform datainto digital solutions. When digital systems are connected they form a network. For example:

* a smartphone is a digital system that has software (apps, an operating system), input components (e.g. touch screen, keyboard, camera and microphone), output components (e.g. screen and speakers), memory components (e.g. silicon chips, microSD card), communication components (e.g. SIM card, Wifi card, antennae), and a processor made up of one or more silicon chips.
* a desktop computer with appropriate hardware and software used for dairy farming. The computer is connected via cables to milking equipment and via wifi to sensors that read tags on the cows. This allows the recording of how much milk each cow provides and has the potential to control the provision of feed and opening of gates.

Ethical protocols

Generally accepted ‘rules’ or behaviours when undertaking research and collecting and using information from primary and secondary sources, for example, confidentiality, informed consent, citation and integrity of *data*.

Functional requirements

A set of specifications or functions that a digital solution must provide, for example a nominated processing speed, a storage capacity or a security function.

General-purpose programming language

A programming language designed to solve a wide range of programming problems (rather than a language designed for solving domain-specific problems or designed for pedagogical reasons). It includes procedural, functional and object-oriented programming languages, but does not include declarative programming languages such as Prolog or SQL. It includes scripting and/or dynamically typed languages such as Python and Ruby. Examples include C#, C++, Java, JavaScript, Python, Ruby and Visual Basic.

Information systems

The combination of digital hardware and software components (digital systems), data, processes and people that interact to create, control and communicate information.

Iteration

Repetition of a process or set of instructions in computer programming.

Multimedia

The use of digital systems and peripheral devices to present text, graphics, video, animation and sound in an integrated way. Examples include interactive games, media-rich websites, electronic books (eBooks) and/or animated short films.

Non-functional requirements

A set of attributes or characteristics that a solution should possess such as reliability, robustness and useability.

Object-oriented programming (OOP) language

A programming language that supports the object-oriented programming paradigm. In object-oriented programming, objects represent a combination of data (the attributes of an object) and the actions that can be performed on or with that data (the methods of the object). An example might be the declaration of a ‘car’ which has attributes that describe its physical nature (such as the number of doors, its colour, the size of the engine) and the actions it can perform (such as accelerating, braking and turning).

The valid attributes and methods of an object are defined by its class, and these attributes and methods can be inherited from the definition of another class. Examples of OOP languages include C++, Eiffel, Java, Python and Scala.

Peripheral devices

Digital components that can be connected to a digital system but are not essential to the system, for example printer, scanner, digital camera.

Project

The set of activities undertaken by students to address specified content, involving understanding the nature of a problem, situation or need; creating, designing and producing a solution to the project task and documenting the process. Project work has a benefit, purpose and use; a user or audience who can provide feedback on the success of the solution; limitations to work within; and a real-world technologies context influenced by social, ethical and environmental issues. Project management criteria are used to judge a project’s success.

Protocols

Generally accepted standards or 'rules' that govern relationships between and within information systems.

Social protocols

Generally accepted 'rules' or behaviours when people interact in online environments, for example, using language that is not rude or offensive to particular cultures, and not divulging personal details about people without their permission.

Structured English

The use of the English language to describe the steps of an algorithmin clear, unambiguous statements that can be read from start to finish. The use of keywords, such as START, END, IF, UNTIL, provides a syntax similar to that of a programming language to assist with identifying logical steps necessary to properly describe the algorithm.

An example of the use of structured language can be demonstrated using the following problem:

Description of the problem:

Describing the decision a person makes about how to get to a destination based on the weather and the distance from their current location to their destination.

Structured English example:

START

IF it is raining outside THEN

 Catch the bus

ELSE

 IF it is less than 2km to the destination THEN

 Walk

 ELSE IF it is less than 10km to the destination THEN

 Ride a bicycle

 ELSE

 Catch the bus

 ENDIF

ENDIF

END

The Structured English description can easily be translated into code using a programming language and accurately captures the logical elements that must be followed to answer the question posed.

Sustainability factors

Economic, environmental and social decisions that impact on current design decisions without compromising future needs.

Systems thinking

A holistic approach to the identification and solving of problems where parts and components of a system, their interactions and interrelationships are analysed individually to see how they influence the functioning of the whole system. This approach enables students to understand systems and work with complexity, uncertainty and risk.

Technical protocols

A set of ‘rules’ or behaviours that involve using technical facilities to support personal safety and security of data***,*** such as activating privacy settings and online filtering features.

User experience

Aspects that affect how an end-user interacts with digital systems, such as navigation design, expertise of user and ergonomics.

User interface

The characteristics of the boundary between users and a digital system**,** or the manner in which users interact with digital system components (hardware and software). In software, this usually comprises of fields for text and number entry, mouse pointers, buttons and other graphical elements. In hardware, switches, dials and LEDs provide information about the interactions between the user and machine.

Validate

The process of checking that input data is reasonable such as its accuracy, consistency, completeness and bias.

Visual programming

A programming language or environment where the program is represented and created visually rather than as text. A common visual metaphor represents statements and control structures as blocks that can be composed to form programs, allowing programming without having to deal with syntax errors. Examples of visual programming languages include: Alice, GameMaker, Kodu, Lego Mindstorms, MIT App Inventor, Scratch (Build Your Own Blocks and Snap).

*Note:* A visual programming language should not be confused with programming languages for creating visualisations or programs with user interfaces, for example, Processing or Visual Basic.

Wireless devices

Devices that transmit and receive data from other sources over a communications medium that is not connected by electrical conductors. A common example of a wireless device is a mobile phone, which uses radio waves to connect to telecommunications towers for the purpose of communication.