ICT & Computing Progress Grid

Pupil Progress in...	Algorithms	Programming & Development
9	Select, justify and apply appropriate techniques and principles to develop data structures and algorithms for the solution of problems	Describe,explain and implement the systematic development of high-quality solutions to problems and the techniques for implementing such solutions, including the use of programming languages
8	• Designs a solution to a problem that depends on solutions to smaller instances of the same problem (recursion). • Understands that some problems cannot be solved computationally.	• Designs and writes nested modular programs that enforce reusability utilising sub-routines wherever possible. • Understands the difference between 'While' loop and 'For' loop, which uses a loop counter. • Understands and uses two dimensional data structures. Understands, explains and implements robust programs (defensive design considerations)
7	• Recognises that the design of an algorithm is distinct from its expression in a programming language (which will depend on the programming constructs available). • Evaluates the effectiveness of algorithms and models for similar problems. • Recognises where information can be filtered out in generalizing problem solutions. • Uses logical reasoning to explain how an algorithm works.	• Appreciates the effect of the scope of a variable e.g. a local variable can't be accessed from outside its function. • Understands and applies parameter passing. • Understands the difference between, and uses, both pre-tested e.g. 'while', and post-tested e.g. 'until' loops. • Applies a modular approach to error detection and correction.

	• Represents algorithms using structured language.	

6	• Understands a recursive solution to a problem repeatedly applies the same solution to smaller instances of the problem.	• Uses nested selection statements.
	• Recognises that some problems share the same characteristics and use the same algorithm to solve both.	• Appreciates the need for, and writes, custom functions including use of parameters.
	• Understands the notion of performance for algorithms and appreciates that some algorithms have different performance characteristics for the same task.	• Knows the difference between, and uses appropriately, procedures and functions.
		• Understands and uses negation with operators.
		• Uses and manipulates one and two dimensional data structures.
		• Detects and corrects syntactical errors.
		<u>Can write basic file handling routines</u>

5	• Understands that iteration is the repetition of a process such as a loop.	• Understands that programming bridges the gap between algorithmic solutions and computers.
	• Recognises that different algorithms exist for the same problem.	• Has practical experience of a high-level textual language, including using standard libraries when programming.
	• Represents solutions using a structured notation.	• Uses a range of operators and expressions e.g. Boolean, and applies them in the context of program control.

	• Can identify similarities and differences in situations and can use these to solve problems (pattern recognition).	• Selects the appropriate data types.
	Be able to create a basic search and bubble sort algorithm	Recognises the purpose of translators and facilities of languages
	• Shows an awareness of tasks best completed by humans or computers.	• Understands the difference between, and appropriately uses if and if, then and else statements.
4	• Designs solutions by decomposing a problem and creates a sub-solution for each of these parts.	• Uses a variable and relational operators within a loop to govern termination.
	• Recognises that different solutions exist for the same problem.	• Designs, writes and debugs modular programs using procedures.
		• Knows that a procedure can be used to hide the detail with sub-solution.
3	• Designs solutions (algorithms) that use repetition and two-way selection i.e. if, then and else.	• Creates programs that implement algorithms to achieve given goals.
	• Uses diagrams to express solutions.	• Declares and assigns variables.
	• Uses logical reasoning to predict outputs, showing an awareness of inputs.	• Uses post-tested loop e.g. 'until', and a sequence of selection statements in programs, including an if, then and else statement.
	• Understands that algorithms are implemented on digital devices as programs.	• Uses arithmetic operators, if statements, and loops, within programs.
	• Designs simple algorithms using loops, and selection i.e. if statements.	• Uses logical reasoning to predict the behaviour of programs.

2

- Uses logical reasoning to predict outcomes.

- Detects and corrects errors i.e. debugging, in algorithms.

- Detects and corrects simple semantic errors i.e. debugging, in programs.

1

- Understands what an algorithm is and is able to express simple linear (non-branching) algorithms symbolically.

- Knows that users can develop their own programs, and can demonstrate this by creating a simple program in an environment that does not rely on text e.g. programmable robots etc.

- Understands that computers need precise instructions.

- Executes, checks and changes programs.

- Demonstrates care and precision to avoid errors.

- Understands that programs execute by following precise instructions.

Data & Data Representation | Hardware & Processing

Data & Data Representation	Hardware & Processing
Can perform a range of calculations and conversions of binary using a wde range of operands and number systems. Is able to design and implement an effective relational database independently for a range of purposes.	Design, implement and document an effective solution using appropriate hardware and software
• Performs operations using bit patterns e.g. conversion between binary and hexadecimal, binary subtraction etc.	• Has practical experience of a small (hypothetical) low level programming language.
• Understands and can explain the need for data compression, and performs simple compression methods.	• Understands and can explain Moore's Law.
• Knows what a relational database is, and understands the benefits of storing data in multiple tables.	• Understands and can explain multitasking by computers.
• Knows the relationship between data representation and data quality.	• Knows that processors have instruction sets and that these relate to low-level instructions carried out by a computer.
• Understands the relationship between binary and electrical circuits, including Boolean logic.	
• Understands how and why values are data typed in many different languages when manipulated within programs.	

• Understands how numbers, images, sounds and character sets use the same bit patterns.	• Understands the **von Neumann** architecture in relation to the fetch- execute cycle, including how data is stored in memory.
• Performs simple operations using bit patterns e.g. binary addition.	Understand the purpose of embedded systems
• Understands the relationship between resolution and colour depth, including the effect on file size.	• Understands the basic function and operation of location addressable memory.
• Distinguishes between data used in a simple program (a variable) and the storage structure for that data.	
Knows a wide range of system security vulnerabilities and how to avoid them	

• Knows that digital computers use binary to represent all data.	• Recognises and understands the function of the main internal parts of basic computer architecture. Understands CPU components and their functions and how they relate to memory
• Understands how bit patterns represent numbers and images.	• Understands the concepts behind the fetch-execute cycle.
• Knows that computers transfer data in binary.	• Knows that there is a range of operating systems and application software for the same hardware. Knows the utilities available to maintain them.

• Understands the relationship between binary and file size (uncompressed).

• Defines data types: real numbers and Boolean.

• Queries data on one table using a typical query language.

Identify forms of attack and how to avoid them eg DDOS, SQL injections

• Performs more complex searches for information e.g. using Boolean and relational operators.	• Understands why and when computers are used.
• Analyses and evaluates data and information, and recognises that poor quality data leads to unreliable results, and inaccurate conclusions.	• Understands the main functions of the operating system.
List a wide range of security measures	• Knows the difference between physical, wireless and mobile networks.
• Understands the difference between data and information.	• Knows that computers collect data from various input devices, including sensors and application software.
• Knows why sorting data in a flat file can improve searching for information.	• Understands the difference between hardware and application software, and their roles within a computer system.
• Uses filters or can perform single criteria searches for information.	
Can explain basic data security rules	
• Recognises different types of data: text, number.	• Recognises that a range of digital devices can be considered a computer.
• Appreciates that programs can work with different types of data.	• Recognises and can use a range of input and output devices.

• Recognises that data can be structured in tables to make it useful.

Knows some ways of keeping data safe

• Understands how programs specify the function of a general purpose computer.

• Recognises that digital content can be represented in many forms.

• Understands that computers have no intelligence and that computers can do nothing unless a program is executed.

• Distinguishes between some of these forms and can explain the different ways that they communicate information.

• Recognises that all software executed on digital devices is programmed.

Recognises the need to keep data safe

Communication & Networks	Information Technology
Understands in detail how to configure the hardware and associated properties and protocols for both LAN and WAN networks. Can select suitable topologies	Comment critically on the consequences of current uses of computing, including economic, social, legal and ethical issues explains emerging technologies and their implications for future use of ICT.
• Understands the hardware associated with networking computer systems, including WANs and LANs, understands their purpose and how they work, including MAC addresses.	• Understands the ethical issues surrounding the application of information technology, and the existence of legal frameworks governing its use e.g. Data Protection Act, Computer Misuse Act, Copyright etc.
• Knows the purpose of the hardware and protocols associated with networking computer systems. .	• Undertakes creative projects that collect, analyse, and evaluate data to meet the needs of a known user group.
• Understands the client-server model including how dynamic web pages use server-side scripting and that web servers process and store data entered by users.	• Effectively designs and creates digital artefacts for a wider or remote audience.
• Recognises that persistence of data on the internet requires careful protection of online identity and privacy.	• Considers the properties of media when importing them into digital artefacts.
	• Documents user feedback, the improvements identified and the refinements made to the solution.

	• Explains and justifies how the use of technology impacts on society, from the perspective of social, economical, political, legal, ethical and moral issues.
• Knows the names of hardware e.g. hubs, routers, switches, and the names of protocols ; SMTP, iMAP, POP, FTP, HTTP/S,TCP/ IP, associated with networking computer systems.	• Justifies the choice of and independently combines and uses multiple digital devices, internet services and application software to achieve given goals.
• Uses technologies and online services securely, and knows how to identify and report inappropriate conduct.	• Evaluates the trustworthiness of digital content and considers the usability of visual design features when designing and creating digital artifacts for a known audience.
Understands packet switching	• Identifies and explains how the use of technology can impact on society.
	• Designs criteria for users to evaluate the quality of solutions, uses the feedback from the users to identify improvements and can make appropriate refinements to the solution.
• Understands how search engines rank search results.	• Evaluates the appropriateness of digital devices, internet services and application software to achieve given goals.
• Understands how to construct static web pages using HTML and CSS.	• Recognises ethical issues surrounding the application of information technology beyond school.
• Understands data transmission between digital computers over networks; Including the cloud and the concept of virtual networks including the internet i.e. IP addresses and packet switching.	• Designs criteria to critically evaluate the quality of solutions, uses the criteria to identify improvements and can make appropriate refinements to the solution.

Understands the difference between a LAN and WAN and can explain the function of the main components.

Recognises star and mesh network topologies

• Understands how to effectively use search engines, and knows how search results are selected, including that search engines use 'web crawler programs'.	• Makes judgements about digital content when evaluating and repurposing it for a given audience.
• Selects, combines and uses internet services.	• Recognises the audience when designing and creating digital content.
• Demonstrates responsible use of technologies and online services, and knows a range of ways to report concerns.	• Understands the potential of information technology for collaboration when computers are networked. • Uses criteria to evaluate the quality of solutions, can identify improvements making some refinements to the solution, and future solutions.
• Understands the difference between the internet and internet service e.g. world wide web.	• Collects, organises and presents data and information in digital content.
• Shows an awareness of, and can use a range of internet services e.g. VOIP.	• Creates digital content to achieve a given goal through combining software packages and internet services to communicate with a wider audience e.g. blogging.
• Recognises what is acceptable and unacceptable behaviour when using technologies and online services.	• Makes appropriate improvements to solutions based on feedback received, and can comment on the success of the solution.
• Navigates the web and can carry out simple web searches to collect digital content.	• Uses technology with increasing independence to purposefully organise digital content.
• Demonstrates use of computers safely and responsibly, knowing a range of ways to report unacceptable content and contact when online.	• Shows an awareness for the quality of digital content collected.

	• Uses a variety of software to manipulate and present digital content: data and information.
	• Shares their experiences of technology in school and beyond the classroom.
	• Talks about their work and makes improvements to solutions based on feedback received.
• Obtains content from the world wide web using a web browser.	• Uses software under the control of the teacher to create, store and edit digital content using appropriate file and folder names.
• Understands the importance of communicating safely and respectfully online, and the need for keeping personal information private.	• Understands that people interact with computers.
• Knows what to do when concerned about content or being contacted.	• Shares their use of technology in school.
	• Knows common uses of information technology beyond the classroom.
	• Talks about their work and makes changes to improve it.