DIXONS TRINITY CHAPELTOWN

Computing

Curriculum principles

By the end of their all-through education, a student of computing at Dixons Trinity Chapeltown will:

- know the importance of computer science and the contribution emerging technologies can make in society. Students will know how to program using industry standard programming languages and how computers communicate internally and globally. Students will be able to communicate, with confidence, the effects of the technological and cultural divide facing society. Students will be taught the GCSE Computer Science curriculum to prepare them for A-Level studies in the subject and beyond. To enable them to thrive in a top job and have a great life.
- understand that we are now in a digital age. Computer scientists can theorise, design, develop and implement new programmes that have stance globally. Students at Dixons Trinity Chapeltown, will be exposed to a diverse curriculum that will effectively prepare them for further study at university and develop their cultural capital.

Our uniting 'sentence' is: "The Computing Department worked tirelessly to ensure all students were exposed to an exciting computing curriculum in order to contribute effectively towards the development of a technological society."

In order to achieve a true understanding of computing, topics have been intelligently sequenced based on the following rationale:

- the computing curriculum is built upon abstract skills, like algorithmic thinking, problem solving, and decomposition from Y1. Computer Science will have cross-curricular links to subjects such as: PE, science, mathematics and art and design.
- the curriculum has clear literacy links to embed key terminology, so children are speaking as computer scientists from a young age.
- the curriculum is designed to be spiralised, in line with Bruner's spiral curriculum theory. Topics are covered, then revisited through low stakes quizzing throughout the cycle and then in future years. This ensures the forgetting is interrupted and revision occurs throughout school. Delivering the curriculum in this way allows for further, extensive teaching once the foundations are established.
- students of computing are introduced to GCSE computer science topics and vocational based skills throughout their studies from Y1. Students will use terminology correctly, such as 'algorithm' in primary computing lessons and understand the purpose of following instructions clearly which can then be applied to understanding how a computer processes these instructions as commands.

The computing curriculum will address social disadvantage by addressing gaps in students' knowledge and skills:

- there is no assumption from the department that students will have access to specialist hardware and software outside of school. As such, extra-curricular clubs will have access to technology should any student wish to continue learning about a specific topic. The laptops will have all required software. Interleaved revision at home will take place on the knowledge organisers and there is no requirement for students to have access to computer equipment at home to complete.
- students requiring additional support benefit from interleaving and frequent low stakes quizzing. This ensures intervention is proactive and data driven. On a regular basis, members of the computing department address the gaps identified from in-class or cycle assessment data to offer provision to eradicate these differences.
- students will also complete a computing project where they are to plan, design, create, test and evaluate a solution to a real-life problem. These projects will be filmed and shown to family members.
- students in upper peak are provided with a wealth of resources to reduce the 'digital divide' including revision guides, exam workbooks, flash cards, paper-based homework and mini-tests
- students in middle and upper peak have access to the computer suite before school, during their social time and after school

We fully believe computing can contribute to the personal development of students at DTC:

- valuable team-working skills will be developed by working together to debug and resolve issues in code. Students will learn programming through a driver-navigator method where each person has a responsibility for reading and writing code accurately. They will learn how to write code through an understanding that making mistakes is vital for our development and mistakes in code teach us different ways to solve the same problem. The computing curriculum requires resilience, especially in topics such as Python programming and this skill will help students throughout their time at school and beyond.
- students will learn how to decompose problems and think abstractly in order to develop problem solving skills, from Y1 throughout school, which will benefit them in all other lessons and throughout their career. The ability to decompose a problem, think abstractly and use logical reasoning to create a solution to a problem, will be developed in each computing lesson to ensure this skill is fully embedded.
- the computing curriculum offers the opportunity to investigate the legal implications of computing. Students will be able to give their opinions on matters affecting computing, like ethical hacking, data storage and targeted advertising. They will also be able to argue

on topics such as ethical, cultural and moral issues within computer science - investigating the environmental impact of technology development and recycling of materials.

At Middle Peak and Upper Peak, our belief is that homework should be interleaved revision of powerful knowledge that has been modelled and taught in lessons. This knowledge is recalled and applied through a range of low stakes quizzing and practice.

Opportunities are built in to make links to the world of work to enhance the careers, advice, and guidance that students are exposed to:

- the computing curriculum provides students with opportunities to consider the world of work and how the development of IT and computational skills lead to successful careers. The SoW refers to how the skill in question relates to specific careers in a 'careers spotlight'.
- students will be able to put their knowledge of computer science into practice during expeditions to universities in Y9 and Y10 where they will meet undergraduates studying in the field and preparing to begin a career in computing. Further expeditions to workplaces within Leeds will show students how the city is contributing to the development of new technology in Leeds.
- the curriculum provides multiple career opportunities for students. They will investigate the wealth of careers available in the field of computer Science from Y2 and as they progress through the academy, they will be shown real-life examples of careers and see first-hand the requirements of roles and the types of responsibilities computer scientists have.
- evident links to careers are made cross-phase from base camp fluidly up until Year 11. These links are signposted in the LTP, SOW and directly into the E-Work booklets completed by students

A true love of computing involves learning about various cultural domains. We teach beyond the specification requirements, but do ensure students are well prepared to be successful in GCSE examinations:

- the Computing Department also run co-curricular electives in robotics, and the Duke of York iDEA award, so students are able to put their programming skills into contextualised practice, to instil a love of computing-based subjects.
- students will cover large topics like the three constructs (abstraction, decomposition and computational thinking) through play-based learning in Y1 and more academic lessons from KS2. Students will be taught algorithmic practice from Y1, in the form of following specific instructions and understanding how a computer processes these instructions. The GCSE Computer Science specification, alongside the National Curriculum for Computing, underpins the entire scheme from early years through to GCSE and topics are intelligently sequenced and revisited.
- enrichment opportunities are diverse and include: robotics, animation, the iDEA award and digital journalism
- the computing digital ambassadorship responsibility ensures students are actively supporting younger students to be successful. The program includes a peer mentorship element where students studying computing in the upper peak will develop provision for students in lower peak

Further information

- All through curriculum overview
- Long term plans
- Knowledge organisers

Curriculum Overview

All children are entitled to a curriculum and to the powerful knowledge that will open doors and maximise their life chances. Below is a high-level overview of the critical knowledge children will learn in this particular subject, at each key stage from Reception through to Year 11, in order to equip students with the cultural capital they need to succeed in life. The curriculum is planned vertically and horizontally giving thought to the optimum knowledge sequence for building secure schema.

		Knowledge; skills and understanding to be gained at each stage*		
		Cycle 1	**Cycle 2**	**Cycle 3**
EYFS	**Know and remember**	Play interactive games using technology**;** cause and effect; toys that wind / buttons / movement / sound books	Programming – use Bee-Bots to start to understand sequencing	Know that a range of technology is used in places such as homes and schools. Select and use technology for different purposes
	Do	Enabling Environments: playing with a range of materials and objects that work in different ways for different purposes; playing with a range of programmable toys and equipment including computers; using search engines to research; using telephones for communication; use the Paint programme within the creative area		
	e- safety	Feeling safe online; adult supervision and how to ask for support and permission		
YEAR 1	**Know and remember**	**Safe use of technology** Logging on, use of a trackpad/mouse. Opening and closing programs, keeping password and username safe. Dragging, copying, pasting, left-click, ricght-click **Use of technology** Technology outside of school (self-service checkouts / ticket machines); impact to society. How technology is used within education, hospitals, retail	**Personal Information** How to keep safe online. Investigating which information can be shared and which needs to be kept private. Researching personal information and offering advice on how to protect data **Careers in ICT / Computing** Common careers in the field and routes to get there. Personal project researching careers and using software to present findings	**Algorithms** Importance of clear instructions within a computer system **Scratch** Directional programming in a block-based environment. Decomposition, abstraction and iteration **Micro:Bit** Block-based coding using the online platform that is downloaded onto the physical devices
	ICT skills	Typing, saving, editing, undo and redo, select and format text		
	e- safety	Keeping safe online; searching for images; personal information; owning your creative work		
	NCC aims	1D	1A, 1B, 1C	1A, 1B, 1E
	Vertical and horizontal interleaving	Links to e-safety throughout the curriculum each year	E-safety – how to ask for support, built upon to understand how to report concerning content	Foundation skills to be built on in subsequent years / cycles
YEAR 2	**Know and remember**	**Safe use of technology** Recapping how to use PC equipment safely and responsibly **Internet explorer** Developing independent internet research skills – on the topic of e-safety	**Algorithm prediction** Tracing pre-designed algorithms in order to predict their outcomes Using patterns and logic to solve problems	**Creating digital content** Creative freedom to produce a web blog, digital graphic or video surrounding algorithms. Enhancing the aesthetics and creating folders for organisation
	ICT skills	Create a folder; new slide; new layout; add and format images; reorder slides; search and print		
	e- safety	Digital footprints; using technology safely and respectfully; keeping personal information private; being kind online		
	NCC aims	1A, 1B, 1C	1A	1D, 1F, 1E
	Vertical and horizontal interleaving	Word processing and computer skills (Y1) Geography – Interactive maps (C1 Y2)	Algorithms (Y1), directional programming in a block based environment	Safe use of a computer (Y1) Computer skills (typing, formatting, editing images) (Y1)
YEAR 3	**Know and remember**	**Physical systems** Creating working systems with hardware or through online software **Designing, writing and debugging algorithms** What are algorithms and how do we write them? Use of flowchart shapes to create solution; coding in scratch	**Decomposition** Investigating how to break down a large problem into smaller, independent programs **Abstraction** Removal of the non-essential data so as to focus on important elements	**Sequence, selection, iteration** Using options within their programs and investigate the best way to structure code Use of loops and evaluate the effectiveness on the efficiency

		Knowledge; skills and understanding to be gained at each stage*		
		Cycle 1	**Cycle 2**	**Cycle 3**
YEAR 3	**ICT skills**	Case change; align text; bullets and numbering; keyboard shortcuts; text boxes and text wrapping		
	e- safety	e-mails; online communications; cyber bullying		
	NCC aims	2A, 2B	1A, 2A	2B
	Vertical and horizontal interleaving	Bullying in PSHCE (Y3 C1) Computer skills and word processing (Y1 / Y2) Creating digital content (Y2 C3)	Algorithms (Y1-2), directional programming in a block based environment	Algorithms (Y1-2), directional programming in a block based environment
YEAR 4	**Know and remember**	**Networks and the internet** Explore how data is transferred across a network and investigate the differences between the internet and the World Wide Web	**Input and output systems** Research different devices and their purpose within a computer **Storage devices** Students will investigate how data is stored on a computer system and research both primary and secondary storage methods	**Memory** Explore how data is stored in binary and how devices communicate within a computerized system **Coding** Block based coding (scratch / Kodu) using IF / THEN statements
	ICT skills	Format images; layout; spell check; tables; hyperlinks		
	e- safety	Plagiarism; using search engines; being a good online citizen and how to deal with cyberbullying		
	NCC aims	1E, 2A, 2B	2E, 2F, 2G	2A, 2B, 2C
	Vertical and horizontal interleaving	Cyber-bullying (Y3) Word processing (Y1-3) Networking (Y3)	Computer skills and word processing (Y1 / Y2)	Algorithms (Y1-3), directional programming in a block based environment.
YEAR 5	**Know and remember**	**Debugging** Investigating errors in code and writing solutions **Networking** LAN, MAN, WAN, peer to peer networks How devices communicate in a network	**Internet** Explore the WWW and internet as a collection of web pages **Communication and collaboration** Investigate how devices can be used as a communication and collaboration tool	**Analysing digital content** Assessing against a brief and identifying improvements
	e- safety	Spam e-mails; citing sources; safe passwords; fake images; investigating methods of keeping information safe; using websites correctly and reporting concerns		
	NCC aims	1E, 2D	2D, 2E, 1E	2E, 2F, 2G
	Vertical and horizontal interleaving	Networking (Y3 C2) Algorithms (Y1, Y2)	Emails (Y3 C1)	Use of software to create content (Y3 C1)
YEAR 6	**Know and remember**	**Sensible use of technology** Exploring ways to use websites safely **Software for presenting** Use of PowerPoint and Publisher to produce presentational pieces	**Producing digital content** Reviewing online content and creating own web blog / website **Games design** Producing own games in Scratch / alternative block-based coding	**Games development** Producing games in scratch using variables, iteration, sequence and selection **Testing and evaluation** Writing accurate tests to assess the functionality of developed game Recommendations for future development
	e- safety	Internet is media which forms stereotypes		
	NCC aims	1E, 2E, 2F, 2G	2C, 2B	2A, 2B, 2G, 2F,1C, 1G
	Vertical and horizontal interleaving	Presenting (Y2 C1) PSHCE – In the media and stereotypes (Y5 C1)	PSHCE – Playing online games safely (Y4 C2) Safe use of the PC (Y1 C2)	Algorithms (Y1-5), directional programming in a block based environment.

		Knowledge; skills and understanding to be gained at each stage*		
		Cycle 1	**Cycle 2**	**Cycle 3**
YEAR 7	**Know and remember**	**Digital literacy and e-safety** Recognise reliable information sources and list way to keep safe online **Binary** Converting from binary to denary and visa versa Performing binary addition and explaining the overflow error	**Algorithms** Importance of writing clear instructions **Computational thinking** Thinking like a computer in order to solve a problem	**Python programming** Writing programs in Python using sequence, selection and iteration **Hardware and software** Investigate the hardware needed to build a PC system and the software that runs on top
	NCC aims	3I, 3H, 3G, 3E	3A	3C, 3F
	Skills revisited	Builds upon knowledge of how to keep oneself safe on computer devices and place value in binary	Builds upon algorithmic thinking practice skills taught in lower and middle peak	Builds upon computational thinking and algorithmic practice. Scratch programming skills revisited
	CEIAG	Careers in mathematics and data analysis (Topic 2: binary)	Careers in software development (Topic 3: Algorithms / Topic 4: Computational Thinking)	Careers in programming (Topic 5: Python Programming)
YEAR 8	**Know and remember**	**Animation** creating moving imagery through shape, class and motion tweens. Investigate the history of animation	**Hardware and Software** Investigating the physical components of a computer system and the applications run on top of this.	**Python programming** Introduction to programming using block and object-orientated language, using the PRIMM methodology
	NCC aims	3E, 3I	3F, 3D	3G, 3H, 3E
	Skills revisited	New topic, will use key close reading skills and Teams to support – but knowledge is new	Builds upon the knowledge of hardware and software required to build and secure a computer	Builds upon python programming skills developed in middle peak and links to algorithmic practice and computational theory
	CEIAG	Careers in Cyber Security (Topic 1: E-Safety)	Careers in Game/Software Design (Topic 2: Hardware and Software)	Careers in Programming (Topic 3:Programming)
YEAR 9	**Core**	**Duke of York IDEA award** Students complete Information Technology challenges to earn points towards badges, iDEA is the digtial and enterprise equivalent of the Duke of Edinburgh award. Bronze, Silver and Gold award. Challenge elements include: citizen, worker, maker and entrepreneur for Bronze award. Challenges relate to E-Safety, Digital Literacy, Cloud-based systems and storage, cyber security, networking, social media ethics,big data, internet of things, user interfaces and experiences, design psychology, automation, virtual reality, video editing, coding, research and problem solving and more. The iDEA award links to the national curriculum for computing and covers concepts such as programming through the creation and evaluation of computational abstractions in the programming modules and use of more than one programming language. In the iDEA award, students use javascript, python and SQL to interpret, correct and create code. Students will learn to work with Boolean logic during the binary module and during the user interface and sys admin, students will learn how hardware and software communicate.		
	NCC aims	3B, 3D, 3I, 3C, 1E, 3G, 3F, 3A		
	CEIAG	Careers in mechanical engineering, programming, software development and technical support.		
	Examination	**1.1 Systems Architecture** Components of the PC and their functionality. FDE Cycle and Von Nuemann architecture **1.2 Memory and Storage** Primary and secondary storage, data capacity calculations, binary representation, compression **1.3 Networks, connections and protocols** How devices communicate across a nework, the internet, the world wide web, rules for transmission **1.4 Network security** Protecting the network against internal and external threats	**1.5 Systems software** Investigate the hardware needed to build a PC system and the software that runs on top **1.6 Ethical, legal, cultural and Environmental impacts** Analysing the impacts of computing technology on the wider community **2.1 Algorithms** Pseudocode and flowcharts used to create algorithms **2.2 Programming fundamentals** Representation of algorithms, defensive design considerations	**2.3 Producing robust programs** Writing in Python using variables, inputs processes and outputs **2.4 Boolean Logic** Logic gates, truth tables and combining gates to produce logic diagrams **2.5 Programming languages** Using a compiler and an interpreter to produce programs

		Knowledge; skills and understanding to be gained at each stage*		
		Cycle 1	**Cycle 2**	**Cycle 3**
YEAR 9	**NCC aims**	3E, 3I	3E, 3I	3A, 3C
	Skills revisited	Builds upon the knowledge of hardware and software and how to be safe, and protect a computer device	Builds upon the Python and Scratch programming languages developed in middle and lower peak, alongside hardware within a computing device	Builds upon knowledge of numbering systems and place value. Links to Scratch block-based programming and pythonic language
	CEIAG	Careers in animation design (Topic 1-3: animation)	Careers in Systems Analysis (Topic 2 memory and topic 3 Operating Systems)	Careers in Multimedia programming (Topic 4/5/6: Programming)
YEAR 10	**Creative Media Production**	**Unit 1 – Exploring media products** Investigating the impact of media and how media has progressed through the years. Explore how media can be used for a variety of purpose and the varieties of audience. (This unit will be completed in Cycle 2)		
	OCR GCSE Computer Science	**Component 1 – systems architecture** Memory; storage; wired and wireless networks; network topologies; protocols and layers; system security; system software; ethical, legal, cultural and environmental concerns	**Component 2 – algorithms and programming** Algorithms; programming techniques; producing robust programs; computational logic; translators and facilities of languages; data representation	**Component 3 – project / exam prep** Programming techniques; analysis; design; development; testing, evaluation and conclusions
	Skills revisited	Builds upon knowledge of primary and secondary storage and compatibility of technical components within a computer	Builds upon knowledge of programming in text-based and high-level programming languages. Links to machine code taught in middle peak	Builds upon knowledge of sequencing, selection and iteration alongside the software design lifecycle
	CEIAG	Careers in Technical Writing (Topic 1: Systems Architecture)	Careers in Multimedia (LO1- LO4) Careers in software development (Component 2)	Careers in Website Design (LO1-LO4)
YEAR 11	**OCR GCSE Computer Science**	**Component 1** Systems architecure; recap of systems architecure; memory; storage; wired and wireless networks; network topologies; protocols and layers; system security; system software; ethical, legal, cultural and environmental concerns	**Component 2** Algorithms and programming Producing robust programs; computational logic; translators and facilities of languages; data representation	
	CEIAG	Careers in Network Security (Component 1)	Careers in Programming (Component 2)	

*A powerful, knowledge-rich curriculum teaches both **substantive knowledge** (facts; knowing that something is the case; what we think about) and non-declarative or **procedural knowledge** (skills and processes; knowing how to do something; what we think with). There are no skills without bodies of knowledge to underpin them. In some subjects, a further distinction can be made between substantive knowledge (the domain specific knowledge accrued e.g. knowledge of the past) and disciplinary knowledge (how the knowledge is accrued e.g. historical reasoning). Please refer to the DAT Curriculum Principles, published on our website, for further information about how we have designed our all-through curriculum.

DIXONS TRINITY CHAPELTOWN

Year 1 Long Term Plan

	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13
Cycle 1	**Induction**									**Planning days**	**Y8 expedition**		
	Logging on How to log on to a computer. Typing in our usernames and passwords	**Use of trackpad/mouse** Clicking, dragging, selecting, left and right options	**Opening. Closing and minimizing programs** How to use the mouse to open and close documents	**Copying/Pasting** How to add images from the internet using keyboard shortcuts	**Left and Right Click** Using the mouse correctly to locate information and use the menus	**Word skills** typing and font alterations in Microsoft Word	**Technology** Investigating technology outside of school	**Self Service machines** How technology helps us complete tasks	**Technology in schools** How do our teachers use technology to help us	**Technology in hospitals** How are computers used in the health industry	**Microsoft Word** Using Word to alter incorrect spellings	**Microsoft Excel** Using Spreadsheets to calculate	**Review** Recap of taught content - typing game
Cycle 2							**Cycle assessment weeks**		**Data input**		**Y7 expedition**		
	Algorithms What an algorithm is	**Algorithms** Creating algorithms in flowcharts	**Algorithms** Sequencing in flowchart	**Algorithms** Writing algorithms in pseudocode	**Algorithms** Recap – flowchart shapes. Create own flowcharts using flowgorithm	**Algorithms** Loops in pseudocode – while loop	**Careers** What is a career?	**Careers in computing** What careers are available and what do they do?	**Careers** What do Computer Scientists do?	**Careers** What skills do you need to work in the field of computing?	**Careers** Software to present – careers PowerPoint	**Careers** Software to present – careers PowerPoint (Animations and transitions)	**Careers** Software to present – careers PowerPoint (Animations and transitions)
Cycle 3									**Cycle assessment weeks**		**Y9 expedition**		**Recognition**
	Algorithms What is an algorithm?	**Algorithms** The importance of giving clear instructions	**Algorithms** Iteration in algorithms	**Algorithms** Sequence in an algorithm	**Algorithms** Selection in an algorithm	**Algorithms** Using scratch to create algorithms		**Micro:bIt** Use of the website, block based coding challenge	**Micro:bIt** Connecting the Micro:bit to the computer using the USB cable Creating and downloading programs onto the device Strings, inputs and outputs Review				

DIXONS TRINITY CHAPELTOWN

Year 2 Long Term Plan

	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13
Cycle 1	**Induction**									**Planning days**	**Y8 expedition**		
	Logging on How to log on to a computer. Typing in our usernames and passwords	**Use of trackpad/mouse** Clicking, dragging, selecting, left and right options	**Opening. Closing and minimizing programs** How to use the mouse to open and close documents	**Internet** What is the internet?	**Search engines** What is a search engine and how do they work?	**Websites** Use a search engine to find relevant websites	**Useful content** Use search engines to find appropriate images for report	**Use of software** Organize research into a report	**Include content** Update PowerPoint to include E-Safety information		**Include content** Save and insert relevant images	**Top tips** Write recommendations for others	
Cycle 2							**Cycle assessment weeks**		**Data input**		**Y7 expedition**		
	Iteration in Scratch Count controlled loops in Scratch	**Creating shapes** Using iteration in scratch to produce shapes	**Pattern recognition** Identifying repeated code	**Logic errors** Finding and resolving logical errors in code	**Sequencing** Trace through a flowchart and rearrange in the correct order	**Loops** Count controlled loops in Scratch	**Loops** Condition controlled loops in Scratch	**Tracing algorithms** What is happening in the algorithm?	**Algorithm prediction** What happens when we alter sections of the same algorithm	**Testing and debugging** Knowing how to find and resolve errors in own code	**Autonomous Programming** Either using tutorials, success criteria or game ideas produce own code using sequence, selection and iteration.		
Cycle 3									**Cycle assessment weeks**		**Y9 expedition**		**Recognition**
	Online software An introduction to the software types online and their capabilities	**PowerPoint online** Creating, editing, renaming and adding content	**Word online** Creating, editing, renaming and adding content	**Forms online** Creating, editing, renaming and adding content	**Storyboarding online** Creating, editing, renaming and adding content	**Web Blog** Creating, editing, renaming and adding content	**Planning digital content** Deciding on software to use, choosing topic, finding information and content.	**Creating own digital content** Using the software to produce own online portfolio with multimedia objects.		**Editing and presenting** Altering the aesthetics, fonts, images, transitions, animations and effects			

DIXONS TRINITY CHAPELTOWN

Year 3 Long Term Plan

	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13
Cycle 1	**Induction**									**Planning days**	**Y8 expedition**		
	Networks What is a network?	**Communication** How do computers communicate on a network?	**LAN/WAN** What are local and wide area networks	**Topologies – Bus and ring** How networks are designed and how this effects resources	**Topologies – Star and Mesh** How networks are designed and how this effects resources	**Sending data** Moving a data packet across a network to an intended recipient	**What are algorithms** Investigating the differences between pseudocode and flowcharts	**How do we write algorithms?** Listing instructions sequentially or as a diagram	**Pseudocode** Producing a step by step algorithm in a language-independent manner		**Flowchart Shapes (Terminator and parallelogram)** Producing algorithms in a graphical manner **Flowchart Shapes (Decision and rectangle)** Producing algorithms in a graphical manner		**Coding from flowcharts** Producing code within scratch from designed flowcharts
Cycle 2							**Cycle assessment weeks**		**Data input**		**Y7 expedition**		
	Decomposition How to break down problems into more manageable sections	**Decomposition** Producing decomposed algorithms – brushing teeth	**Decomposition** decomposition diagram for brushing teeth algorithm	**Decomposition** Designing a flowchart for each task within the brushing teeth algorithm	**Decomposition** Producing decomposed algorithms – family dining	**Decomposition** decomposition diagram for family dining algorithm	**Abstraction** Removing data that is not needed	**Pattern recognition** Identifying similarities	**Abstracting data** Considering the elements that can be safely removed	**Writing abstracted algorithms** Producing the code required, ignoring the nor-essential elements	**Pirate map creation (using abstraction** Creating pirate treasure maps after removing the unnecessary information.		
Cycle 3									**Cycle assessment weeks**		**Y9 expedition**		**Recognition**
	Sequence Why is order important? How does it influence our code?	**Selection** How to use user input to creative interactive programs	**Iteration** How loops can be created in code to repeat elements	**Sequence** Re-arrange the instructions in a flowchart so the program is no longer nonsensical	**Selection** Store and reply to user input in code	**Iteration** Produce times tables in code using count controlled loops	**Programming project** Creating autonomous projects using Scratch/Micro: bit incorporating sequence, selection and iteration						

DIXONS TRINITY CHAPELTOWN

Year 4 Long Term Plan

	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13
Cycle 1	**Induction**									**Planning days**	**Y8 expedition**		
	Connecting networks reasons for connecting	**protecting networks** How to prevent unwanted access	**Internet** Purpose of a router / packet routes	**World wide web** visit and analyse websites	**Sharing information** How does a video get from the US to the UK?	**How to access the WWW** tools / software required for access	**What is a website?** List features of webpages	**Creating content** Create music using the WWW	**The internet** Who owns the content on the web?	**Reliability of content** Can we trust everything we find on the internet?	**How is content shared?** Can we trace where the information came from?	**Summative assessment** questions relating to the WWW and content	**Summative review** Designing a new GUI for a user
Cycle 2							**Cycle assessment weeks**		**Data input**		**Y7 expedition**		
	Sourcing and storing Finding suitable (copyright free) images on the internet and storing for future use	**Editing images** Adding effects to existing images		**Composition** how the layout of digital imagery affects the meaning	**Altering purpose** changing images to make them suitable for different purposes		**Creating own images** Creating and editing own images to use in publication			**Making and evaluating a publication** Using created / edited images to produce an informative poster (cross curricular links) peer assessment, review and re-draft			
Cycle 3									**Cycle assessment weeks**		**Y9 expedition**		**Recognition**
	Data What is data and what are the different types of data?	**Collecting data** deciding on meaningful questions for a survey	**Collecting data** carry out survey	**Manipulating data** using excel to create graphs / charts	**Data analysis** What does the data show us?	**Hypothesise** Plan a question to be answered using data	**Collect data** using primary and secondary sources to collect data in support of the question		**Manipulating data** using excel to create graphs / charts in support of the question	**Data analysis** What does the data show us? – does this data support / reject the question	**Data report** Present question, how the data was obtained, what the data shows and whether the data has answered the question. Team challenge		

DIXONS TRINITY CHAPELTOWN

Year 7 Long Term Plan

	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13
Cycle 1	**Induction**									**Planning days**	**Y8 expedition**		
	Induction GL assessments Office 365 Login and familiarisation	E-safety **E-safety** L1 – **Content** Personal Information	**E-safety** L2 – **Content** misuse of information **E-safety** L3 - **Content** Identifying reliable sources of information	**E-safety** L4 - **Content** analysing 'fake news' **E-safety** L5 – **Conduct** Email etiquette	**E-safety** L6 – **Conduct** cyberbullying case studies **Messy marking** **E-safety** L7 – **Conduct** Digital footprint Case studies	**E-safety** L8 – **Conduct** Positive platform promotion **E-safety** L9 – **Contact** Reporting concerns. Privacy settings and blocking	**E-safety** L10 – **Contact** Legal implications: creating and sharing **E-safety** L11 – **Contact** Copyright law	**E-safety** L12– **Contact** Intellectual property and defamation of character **Review** **E-safety**	**Binary** 1s and 0s - purpose **Binary** Numbering systems – binary and denary	**Binary** 2-bit denary conversions **Binary** 4 & 8 bit binary conversions	**Binary** Extended writing 1 **Messy marking** **Binary** Image representation	**Binary** sound representation **Binary** text representation	**Binary** addition **Binary** Binary - hex Extended writing two
Cycle 2							**Cycle assessment weeks**		**Data input**		**Y7 expedition**		
	Algorithms What an algorithm is **Algorithms** Flowchart shapes	**Algorithms** Creating algorithms in flowcharts **Algorithms** Correcting algorithms in flowchart	**Algorithms** Sequencing in flowchart **Algorithms** Introduction to pseudocode	**Algorithms** Writing algorithms in pseudocode **Algorithms** Correcting pseudocode **Messy marking**	**Algorithms** Loops in Scratch– repeat loop **Algorithms** Gap closing	**Algorithms** Loops in pseudocode – while loop **Algorithms** Building algorithms from code - flowchart	**Algorithms** Building algorithms from code - pseudocode	**Algorithms** **Computational thinking** Abstraction	**Computational thinking** abstraction **Computational thinking** abstraction	**Computational thinking** Decomposition **Review** **Cycle 1 and 2**	**Computational thinking** Decomposition **Computational thinking** Decomposition diagram	**Computational thinking** Technology in the wider world - education **Computational thinking** Technology in the wider world	**Computational thinking** Technology in the wider world - automotive **Computational thinking** Solutions to real world problems
Cycle 3									**Cycle assessment weeks**		**Y9 expedition**		**Recognition**
	Programming Syntax **Programming** Errors in code	**Programming** PRIMM introduction **Programming** Chatbot - extended	**Programming** Driver / navigator For loops **Programming** Driver / navigator	**Programming** selection **Programming** sequence **Messy marking**	**Programming** iteration **Programming** Scratch iteration	**Programming** Python iteration **Programming** Independent programming.	**Programming** Iteration in turtle **Programming** Independent programming.	**Hardware and software** Input Devices **Hardware and software** Purpose of hardware	**Hardware and software** List devices needed to create a PC **Hardware and software** Output devices	**Hardware and software** Storage devices **Hardware and software** Recommend hardware for given scenarios	**Cycle 1,2,3** **Assessment** **Cycle 1,2,3**	**Hardware and software** Purpose of Hard drive, SSD, Flash **Hardware and software** Role of the CPU in a computer system	**Hardware and software** Computer operating systems **Hardware and software** Operating systems for mobile

DIXONS TRINITY CHAPELTOWN

Year 8 Long Term Plan

	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13
Cycle 1	**Induction**									**Planning days**	**Y8 expedition**		
	Introduction – teams / do now	Animation basics (stage, FPS, tools, properties, layers	Frame by Frame	Use of tweens (classic)	Use of tweens (shape)	Animating text	Use of layers	assessment	storyboarding	Creating scenes – scene 1,2,3		Review and evaluate **messy marking**	Suggest improvements
Cycle 2							**Cycle assessment weeks**		**Data input**		**Y7 expedition**		
	Induction baseline assessment	media consumption	A creator's responsibilities	Safe online talk	Which me should I be? **messy marking**	Gender stereotypes online	**Revision**	**Assessment**	Digital footprint	Cyberbullying: crossing the line	The reality of digital drama	Identifying high-quality sites	Advice report
Cycle 3									**Cycle assessment weeks**		**Y9 expedition**		**Recognition**
	Introduction to programming and PRIMM recap	Scratch programming	Algorithm analysis **Messy marking**	Python programming	Sorting algorithms: Insertion	Sorting algorithms: bubble	Sorting algorithms: merge	Searching algorithms: binary	Searching algorithms: linear **Messy marking**	Logic gates	Boolean logic	**Review Hardware and software**	**Review Hardware and software**

DIXONS TRINITY CHAPELTOWN

Year 9 Long Term Plan (BTEC)

	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13
Cycle 1	**Induction**									**Planning days**	**Y8 expedition**		
	Induction Course introduction	Audience (age, gender and ethnicity)	Purpose and audience	Target audience comparison	How media has transformed over the years **Dr Who Ghostbusters** **Messy marking**	Genre characteristics	Assignment practice – publishing (present)	Assignment practice – audio/moving image (present)	Assignment practice – interactive (present)	Assignment brief analysis	Assignment – Paragraphs 3 & 4 **Messy marking**	Assignment – Paragraphs 5 and 6	Assignment – Paragraph 7 **Messy marking**
	Audience profiling	Media consumption purposes	Lifestyle profiles	Research task – radio advertising		Assignment practice – publishing (past)	Assignment practice – audio/moving image (past)	Assignment practice – interactive (past)	**Feedback and review**	Assignment – Paragraphs 1 & 2	Assignment – feedback 1-4		Assignment – feedback 5,6,7
Cycle 2							**Cycle assessment weeks**		**Data input**		**Y7 expedition**		
	Assignment – Paragraphs 8 and 9	Assignment – Paragraph 10 **Messy marking**	Assignment – Paragraphs 11,12 and 13 **Messy marking**		Assignment – feedback 11,12,13	Assignment – Paragraphs 15 and 16 **Messy marking**	Assignment – feedback 14,15,16	Assignment – Paragraphs 18 and 19 **Messy marking**	**Review**	Assignment – feedback 17,18,19	Assignment – Paragraph 21 **Messy marking**	Conclusion	Assignment – feedback
		Assignment – feedback 8,9,10			Assignment – Paragraph 14		Assignment – Paragraph 17		**Assessment**	Assignment – Paragraph 20	Assignment – feedback 21 and 21		Final hand-in
Cycle 3									**Cycle assessment weeks**		**Y9 expedition**		**Recognition**
	Genre (page76)	Representation (75/76)	Connotations and denotation	Media production techniques – mise en scene	Media production techniques – framing shot	Media production techniques – editing list (102) Media production assignment	Media production techniques – interactivity (workshop)	Learning Aim B practice assignment	**Learning aim B Assignment**				
	Narrative (page 80)	audience interpretation (81)	Genre, representation and audience response assignment (page 87)	Media production techniques – camera angles	Media production techniques – lighting (page97)	Media production techniques – photo editing (Workshop)	Media production techniques – game features (workshop)						

DIXONS TRINITY CHAPELTOWN

Year 9 Long Term Plan (GCSE)

	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13
Cycle 1	**Induction**									**Planning days**	**Y8 expedition**		
	Student Induction Creation of folders for all units **Student Induction** Demonstration of Cornell notes taking method and online resources	**1.1 Systems Architecture** Fetch-execute cycle**1.1 Systems Architecture** CPU Components ALU, cache, registers, CU	**1.1 Systems Architecture** Von Neumann architecture (MAR, MDR, PC and ACC) **1.1 Systems Architecture** Common characteristics (cache, clock, cores)	**1.1 Systems Architecture** Embedded systems**1.2 Memory and Storage** Primary and secondary storage	**1.2 Memory and Storage** data capacity calculations and storage units**1.2 Memory and Storage** Binary conversions, additions and shifts	**1.2 Memory and Storage** Binary: characters, images and sound **Review 1.1 and 1.2**	**1.2 Memory and Storage** Compression: lossy and lossless **1.3 Computer networks** Types of network (LAN and WAN)	**Revision** 1.1, 1.2 and Year8 content **Assessment** Cumulative of yr7,yr8 and cycle 1 yr9	**1.3 Computer networks, connections and protocols** Network hardware and the role of client/server/peer networks	**1.3 Computer networks, connections and protocols** Encryption**3 Computer networks, connections and protocols** Protocols and layers	**1.4 Network Security** Forms of attack (first 3 in the spec) **1.4 Network Security** Forms of attack (next 3 in the spec)	**1.4 Network Security** Prevention methods (first 4 in the spec) **1.4 Network Security** Prevention methods (last 3 in the spec)	**1.4 Network Security** Offer guidance to users of the network from the perspective of network administrator **Review 1.1, 1.2, 1.3 and 1.4**
Cycle 2							**Cycle assessment weeks**		**Data input**		**Y7 expedition**		
	1.5 Systems Software Operating Systems**1.5 Systems Software** Functions of an OS	**1.5 Systems Software** Utility software - purpose**1.5 Systems Software** Encryption software	**1.5 Systems Software** Defragmentation **Review** 1.4 – 1.5	**1.6 Ethical, legal, cultural and environmental impacts of digital technology** Impacts of digital technology	**1.6 Ethical, legal, cultural and environmental impacts of digital technology** Legal impact and legislation	**1.6 Ethical, legal, cultural and environmental impacts of digital technology** Environmental **Revision** 1.1 – 1.4	**Revision** 1.5 and 1.6 **Assessment** Cycle 1 & Cycle 2 content	**2.1 Algorithms** Abstraction, decomposition and algorithmic thinking**2.1 Algorithms** input, processes and outputs for a problem	**2.1 Algorithms** Pseudocode and flowcharts **2.1 Algorithms** Searching algorithms	**2.1 Algorithms** Sorting algorithms**2.2 Programming fundamentals** String manipulation	**2.2 Programming fundamentals** File handling operations**2.2 Programming fundamentals** Data types	**2.2 Programming fundamentals** sequence, selection and iteration**2.2 Programming fundamentals** functions and procedures	**2.3 Producing robust programs** Defensive design considerations**2.3 Producing robust programs** input validation
Cycle 3									**Cycle assessment weeks**		**Y9 expedition**		**Recognition**
	2.3 Producing robust programs Maintainability**2.3 Producing robust programs** Testing:	**2.3 Producing robust programs** Selecting and using suitable test data **Review 2.1 – 2.3**	**2.4 Boolean Logic** Logic gates AND/ OR/NOT**2.4 Boolean Logic** Logic gates AND/ OR/NOT	**2.4 Boolean Logic** Truth tables**2.4 Boolean Logic** Combining gates	**2.4 Boolean Logic** Editing logic diagrams **Review 1.1**	**2.5 Programming languages and IDEs** High level languages**2.5** Low level languages	**2.5 Programming languages and IDEs** Translators**2.5** Compiler and interpreter	**2.5 Programming languages and IDEs** IDE tools and techniques **Assessment** Cycle 1, Cycle 2 & Cycle 3 content	**Exam practice** Exploding 2 mark exam questions**.** Exploding 4 mark exam question	**Exam practice** Exploding 5 mark exam questions Exploding 6 mark exam questions	**Exam practice** Exploding 8/9 mark exam questions Exploding 8/9 mark exam questions	**Craig n Dave** Gap closing using online resources and tailored questioning Gap closing	**Exam paper practice** Full exam paper in test conditions Marking assessment

DIXONS TRINITY CHAPELTOWN

Year 10 Long Term Plan (BTEC)

	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13
Cycle 1	**Induction**									**Planning days**	**Y8 expedition**		
	Course refresher and baseline test	Recap of programming fundamentals	data types in Python	file handling operations – read/write	Project analysis and selection	Project design (flowchart)	Produce IPOD and testing table	Produce own fully commented coded solution	test program using suitable test data	Analyse project – what improvements would you make?	LAN & WAN	Network topologies	Disadvantages of network topologies
	Student Induction Demonstration of Cornell notes taking method and online resources	**Comp3** string manipulation	**Comp3** file handling operation – open/ close/ append	**Comp3** arrays and list	**Comp3** Analysis of the project brief	**Comp3** project pseudocode	**Comp3** analyse given coded solution	**Assessment** Cumulative of yr7,yr8 and cycle 1 yr9	**Comp3** Analyse project – what went well	**Component 2** Introduction to networking	**Component 2** Network hardware	**Component 2** advantages of topologies	**Component 2** exam question practice
Cycle 2							**Cycle assessment weeks**		**Data input**		**Y7 expedition**		
	Hardware in a standalone network	The internet	Concept of servers providing services	Modes of connection (wired and wireless)	IP addressing and Mac addressing	Common protocols (TCP/IP, HTTP, HTTPS)	the concept of layers	Forms of attack (Malware, Social engineering and brute-force)	Common prevention methods including: penetration testing, anti-malware software, firewalls, user access levels, passwords, encryption, physical security		Abstraction	Algorithmic thinking	How are they used to define and refine principles
	Role of the NIC	DNS	The cloud	encryption	Standards	Common protocols (FTP, POP/IMAP/SMTP)	The 4-layer TCP/IP model	Forms of attack (DOS, data interception, SQL injection			Decomposition		Map creation
Cycle 3									**Cycle assessment weeks**		**Y9 expedition**		**Recognition**
	Reflect on development	Review production practices	Annotate examples of practical work	Suggest areas for improvements	Explain the skills developed during the pre-production stage	Explain the skills developed during the production stage	Explain the skills developed during the post-production stage	Assessment feedback and improvements					
	Reflect on application of skills	Document all progress from workshops	Assess strengths	Produce skills audit									

DIXONS TRINITY CHAPELTOWN

Year 10 Long Term Plan (GCSE)

	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13
Cycle 1	**Induction**									**Planning days**	**Y8 expedition**		
	Course refresher and baseline test	Recap of programming fundamentals	data types in Python	file handling operations – read/write	Project analysis and selection	Project design (flowchart)	Produce IPOD and testing table	Produce own fully commented coded solution	test program using suitable test data	Analyse project – what improvements would you make?	LAN & WAN	Network topologies	Disadvantages of network topologies
	Student Induction Demonstration of Cornell notes taking method and online resources	**Comp3** string manipulation	**Comp3** file handling operation – open/ close/ append	**Comp3** arrays and list	**Comp3** Analysis of the project brief	**Comp3** project pseudocode	**Comp3** analyse given coded solution	Assessment Cumulative of yr7,yr8 and cycle 1 yr9	**Comp3** Analyse project – what went well	**Component 2** Introduction to networking	**Component 2** Network hardware	**Component 2** advantages of topologies	**Component 2** exam question practice
Cycle 2							**Cycle assessment weeks**		**Data input**		**Y7 expedition**		
	Hardware in a standalone network	The internet	Concept of servers providing services	Modes of connection (wired and wireless)	IP addressing and Mac addressing	Common protocols (TCP/IP, HTTP, HTTPS)	the concept of layers	Forms of attack (Malware, Social engineering and brute-force)	Hardware in a standalone network	The internet	Concept of servers providing services	Modes of connection (wired and wireless)	IP addressing and Mac addressing
	Role of the NIC	DNS	The cloud	encryption	Standards	Common protocols (FTP, POP/IMAP/SMTP)	The 4-layer TCP/IP model	Forms of attack (DOS, data interception, SQL injection	Role of the NIC	DNS	The cloud	encryption	Standards
Cycle 3									**Cycle assessment weeks**		**Y9 expedition**		**Recognition**
	Inputs, process and outputs for a problem	structure diagrams - decomposition	pseudocode – OCR reference language	creating algorithms using flowcharts		creating algorithms using pseudocode		identify syntax / logic errors in flowcharts	binary search	merge sort	Sorting algorithm - flowchart	sorting algorithm - pseudocode	exam question practice
			algorithm for bubble sort - pseudocode					identify syntax / logic errors in flowcharts	linear search	bubble sort		exam question practice	exam question practice