The Holy Trinity Church of England Secondary School

Computing – Curriculum Guide

WHOLE SCHOOL CURRICULUM INTENT STATEMENT

Curriculum Intent:

- As a Church of England School we are committed to the belief that every child is a child of God – uniquely blessed with individual gifts and talents. The intention of our curriculum is to provide opportunities for all students to flourish, drawing out their individual potential and developing them as fully integrated human beings. Our curriculum reflects our Christian ethos and is designed to be challenging, fulfilling, enjoyable and successful for all, regardless of academic or social barriers.

- The curriculum is designed with the intention of being rich and balanced, giving a breadth of opportunity at Key Stage 3 supporting the development of the skills needed to succeed at Key Stage 4 and Key Stage 5. Key Stage 4 has a core which includes RS GCSE for all, alongside both academic and vocational courses. Key Stage 5 provides appropriate pathways for the majority of students to progress to, and allows students to prepare for further study and the world of work. Alongside this, students are provided with a broad enrichment and work-related offer that stimulates their aspirations as global citizens - resilient in the face of difficulty, skillful when faced with new challenges, confident to step out into the unknown and assured of their individual self-worth and of the unique gifts that they each have to bring to the world.

We review our curriculum on a yearly basis and consider the following principles. We believe our curriculum should:

- Embody the school vision which is to be a centre of excellence for learning, inspired by Christian values, where every person in our school community fulfils their potential.

- Provide a broad and balanced range of subjects, including those that are creative and practical, at all Key Stages.

- Include opportunities in a range of both academic and vocational courses at Key Stages 4&5.

- Ensure Religious Studies sits alongside Maths, English and Science as a Core subject which every student will take up to GCSE level.

- Ensure that every student has the opportunity to develop their character through a variety of experiences within REACH, PE, and the Creative Arts.

- Be coherent and sequential so that new knowledge and skill builds on what has been taught, both within an individual subject and across the wider curriculum.

- Be informed by educational research e.g. Growth Mindset (Carol Dweck) and Feedback Loop (John Hattie).

- Acknowledge our contextual needs by giving a high priority to increasing students' mastery of functional literacy, vocabulary and numeracy to maximise students' opportunities for future learning and employment.

- Provide opportunities for learning outside of the classroom. At KS3 this is being embedded through the concept of a 'Passport'; a collection of co-curricular experiences that every student will have experienced.

- Make explicit links to future career pathways alongside a programme of advice and guidance based on the Gatsby benchmarks to prepare students for life and work. Ensure effective transitions at all key stages; from Year 6 to Year 7, Year 9 to Year 10 and Year 11 to Year 12.

- Meet statutory National Curriculum expectations.

Whole School Curriculum Principles:

- Our curriculum is structured in a logical and chronological manner.

- Leaders carefully plan to ensure that appropriate knowledge and skills build upon what has been taught previously and are carefully developed before application into more challenging and complex situations, appropriate to the different curricular end points. These skills are recognised as the 'big ideas' within each subject area.

- Knowledge is considered as a 'big idea' within all subject areas. We are developing our understanding of cognitive psychology to enable students to maximise their learning potential. This is best seen through the regular opportunities provided within subject areas to revisit knowledge acquired to enable improved retention and recall.

- Our curriculum runs within a two-week timetable made up of 50 periods a fortnight, with 5 x 1 hour lessons plus assembly and Tutor time every day.

- Each day begins at 8.45am with registrations, assemblies and a period of reflection and students are dismissed from this regular timetable at 3.10pm.

- There are a range of additional activities such as study studio and numerous co-curricular opportunities which take place after this time on a daily basis.

Whole School Learning Principles:

- Progress at Holy Trinity is informed by 'Accelerated Learning' which emphasises efficiency by engaging learners in order to speed up the process of learning. It enables students to commit knowledge to their long-term memory and develop and refine their skills and performance over time.

- This in turn supports the OFSTED definition of learning; "Learning is defined as an alteration in long-term memory. If nothing has altered in long-term memory nothing has been learned".

Subject Intent Statement:

Computing

The study of Computing aims to ensure that all students develop confidence in their problem solving & computational thinking skills so they can use apps and high-level programming languages with ease in an increasingly digital world.

Students and parents know what will be studied at the start of each year by reference to the curriculum guides that are released. Lessons also start with a reference to the outcomes and success criteria to ensure that students know what they are studying and the intended end point.

The curriculum is designed in a spiral manner so that basic principles taught in year 7 can be revisited in subsequent years in order to help students secure the appropriate knowledge and skills. For example, in year 7 students will be taught about flowcharts and the symbols used and will use these symbols to construct flowcharts using the software Flowol. Simple flowcharts will be built. In year 8, the principle is revisited with the basics being revised at the beginning of the topic and then more complex ideas being introduced. This method of revisiting topics over and over again right through to GCSE helps to cement the basic knowledge for a student and aids their ability to retain and recall key information.

The year 9 curriculum is designed to direct students towards specific pathways according to their skills and talents, thus enabling them to select appropriate KS4 and 5 qualifications.

We have exciting enrichments in the form of the Inspiring Digital Enterprise Award (iDEA) and competitions such as the UK schools' Computer Animation competition to promote cultural capital, support knowledge and skill development in the taught curriculum and support students in developing skills and making informed choices in relation to further careers in this area. Specific job skills, communication, initiative, organisation, problem solving, resilience, teamwork, basic IT, encouraging creativity, self-reliance, innovation & invention are all encouraged through the design of the lessons.

As a department we aim to ensure that all students are successful, regardless of their starting point or socio-economic background we actively operate in lessons to target additional support to those that need it whilst fostering an atmosphere of trust and encouragement between the student and the teacher. After school intervention is also regularly available for those that need additional support.

As a department we review the progress of our students regularly and from this we can identify those areas of the curriculum where there are underlying weaknesses in relation to its delivery. These are then identified and turned into action points on our whole school development plan. They may relate to changes in the design of the

curriculum and how this needs to be re-developed or may identify pedagogical deficits that need to be developed via CPD.

Curriculum Aim:

- The overall curriculum aim of the Computing team is to deliver a broad curriculum that allows students to develop problem solving & computational thinking skills so they can use apps and high-level programming languages.

- In addition, we want learners to enjoy their learning, which should allow them to progress and reach their potential.

Curriculum Principles:

- The curriculum is designed to meet the requirements of the national curriculum for Computer Science

- Each of the aspects of algorithms, program development, data, data representation, computer hardware, networks and IT will be covered.

Key Stage 3:

- Years 7 & 8 are designated to be the foundation years where there is full coverage of the national curriculum.
- Topics are delivered over an extended period of time (usually between 4 – 6 weeks), to enable students to build up skills and apply these to an appropriate context developing their independent learning skills along the way.
- At the end of year 7, students should be confident in accessing all the programs and languages and be able to use the school network and systems independently.
- At the end of year 8, students will have developed their knowledge of the programs and languages that they have been using, now having the confidence to experiment with the features, functions and commands that they have learned to develop solutions to defined problems and challenges.
- At the end of year 9 students will have developed their ability to build programs to resolve a challenge set and to build IT solutions to meet a client brief.

Key Stage 4:

- At key stage 4 there are 3 possible routes, GCSE Computer Science for those with good computational thinking skills, BTEC DIT for those good with

building business systems requiring the use of user interfaces and spreadsheets and Cambridge National iMedia for those good at devising creative solutions to business problems e.g. creating a visual identity.

- The routes available at KS4 are designed to meet the diverse range of skills and talents of all learners whilst providing identifiable career paths and routes into further education.
- All courses build on the foundations laid in KS3.
- In the GCSE Computer Science course, the curriculum time is split over the fortnight so that two fifths of the time is used for delivering paper 1 topics and 3 fifths is used for delivering paper 2 topics including the skills of programming in Python.
- In the BTEC DIT course, the curriculum time is split initially so that students spend one third of the time learning important concepts and theory and two thirds is spent developing skills for the controlled assessments. During year 10 students complete the first controlled assessment brief set by the exam board. During year 11 students complete the second controlled assessment brief set by the exam board. In the spring and summer term of year 11 there is a focus on the theory work in preparation for the exam.
- In the Cambridge National iMedia course students study the exam module alongside the first controlled assessment (graphics) before submitting by the end of year 10. In year 11 students complete the second controlled assessment (gaming) before preparing for the final exam.

Key Stage 5:

- The A level Computer Science course is usually delivered by two staff with one spending time to deliver the theory aspects of the course and the other spending time on the development of programming skills and then on the development of a coursework solution. Time is split evenly between these two objectives enabling both aspects of the course to be taught concurrently.

Curriculum Overview for Year 7:

Year 7 – Computer Science		
Key topics		**Assessment**
Autumn 1: An introduction to the network and e-safety. Animation (Drawplus)	Students are taught about: • How to use the network • Internet safety and cyberbullying • How to plan an animation using storyboard techniques • How to create an animation including the skills of cloning and onion skinning • How to evaluate your work	Storyboard techniques The animation that has been created. Homework set electronically via Class Charts.
Autumn 2: Spreadsheets (Excel)	Students are taught about: • How to enter data, format a spreadsheet and perform simple calculations. • Simple modelling techniques including goal seek • Validation & verification techniques • Sorting & filtering of data.	Assessment of a variety of spreadsheets created. Homework set electronically via Class Charts.
Spring 1: Computer hardware Control (Flowol)	Students are taught: • How to identify & explain the role of the hard disk, CPU, motherboard, RAM, graphics card and power supply. • How to write basic commands in Logo to control a turtle including the use of the repeat function. • How to identify inputs, outputs, delays and decisions on a flow chart. • How to set up a flow chart to operate a lighthouse, traffic lights and automated greenhouse. • How to use open and closed loop flow charts. • How to use sub-routines.	Flow chart plans for the automated greenhouse. The actual greenhouse flow chart created on flowol. Homework set electronically via Class Charts.
Spring 2: Python	Students are taught: • How to write programs that use print and input commands. • How to use a variable • How to use an IF statement	The python programs created are assessed according to the skills demonstrated i.e., complexity of the script used. Homework set electronically via Class Charts.

Summer 1: Web pages (HTML)	Students are taught: • Key terms such as search engine, URL & keywords • How to use HTML to create a webpage using: • Open and close tags, head, body and title and table tags. • How to insert an image • How to create hyperlinks. • Use of CSS to create style for the pages for headings, the body and paragraphs	The web pages created are assessed according to the skills demonstrated i.e., insertion of graphics, use of tables. Homework set electronically via Class Charts.
Summer 2: Scratch	Students are taught: • How to set up a background & create a sprite, including editing sprites in the gallery. • How to move a sprite and add sound effects. • How to set the xy coordinates for a sprite. • How to use broadcast and receive.	The game created is assessed according to the skills demonstrated i.e., complexity of script employed. Homework set electronically via Class Charts.

Suggestions for independent study and home support:

- www.teach-ict.com – To access information and help sheets
- www.codecademy.com – To enable students to practise their programming skills
- http://scratch.mit.edu/ - To enable student to practise their programming skills using a drop & drag interface.

KEY SKILLS

Literacy:	**Numeracy:**	**Other:**
Animation, Binary, Cell replication, Chatroom, Cloning, CPU, Cyberbullying, Decision, Filter, Flow chart, Formula, Input, Keyword search, Motherboard, Onion skinning, Output, Power supply, Print, RAM, Repeat, Search engine, Sort, Storyboard, Tag, URL, Validation, Variable, Verification, Web browsers	Intrinsic to computing and including: Calculating averages Finding the minimum and maximum number in a range. Binary to denary conversion Denary to Binary conversion Creation of graphs	Computational thinking skills • Decomposition • Pattern recognition • Abstraction • Algorithms Independent learning skills including critical thinking and problem solving.

Curriculum Overview for Year 8:

Year 8 – Computer Science		
Key topics		**Assessment**
Autumn 1: Graphics (Draw plus) Scratch	Students are taught: • How graphics are stored on a computer – pixels, bits, compression. • How to create sprites for a pong game • How to create a game using scratch using xy coordinates, if statements, movement, sensing, variables, broadcast and receive.	The game created is assessed according to the skills demonstrated i.e. complexity of script employed. Homework set electronically via Class Charts.
Autumn 2: Spreadsheets (Excel)	Students are taught: • How to plan a spreadsheet that will meet the audience and purpose. • How to create a spreadsheet using currency formatting, formula (+ - * /, absolute cell referencing), data validation. • How to use a spreadsheet for 'what if' analysis including goal seek. • How to present information in a spreadsheet using graphs and reports.	The spreadsheet created is assessed according to the skills demonstrated i.e. formula used. Homework set electronically via Class Charts.
Spring 1: Databases (Access)	Students are taught: • How to plan to create a database including choosing sensible field names and data types. • How to create a database using validation • How to use a database to search for answers to questions using OR, NOT and AND. • How to create relational databases and query it.	The database created is assessed according to the skills demonstrated i.e. formula used. Homework set electronically via Class Charts.
Spring 2: Control (Flowol)	Students are taught: • How to set up a complex flow chart to operate the parts of the autohome. • How to use loops in their flowcharts and add sound effects as outputs. • How to use sub-routines to improve the efficiency of their solution.	Flow chart plans for the autohome. The actual autohome flow chart created on flowol. Homework set electronically via Class Charts.
Summer 1: Python	Students will be taught how to: • How to write programs that use print and input commands. • How to use a variable • How to use an IF statement	The python programs created are assessed according to the skills

Networks	• How to use iteration and lists within their programs How to identify & explain star, bus and ring networks and the advantages and disadvantages of these network structures.	demonstrated i.e. complexity of the script used. Homework set electronically via Class Charts.
Summer 2: Websites (openElement)	Students will be taught how to: • Create a storyboard to set up the design of their website • Create the website using tables, adding images, hyperlinks, marquees, photo galleries, sound and movies. • Critically evaluate their work.	The actual website created and the skills demonstrated in this work Homework set electronically via Class Charts.

Suggestions for independent study and home support:

- www.teach-ict.com – To access information and help sheets
- www.codecademy.com – To enable students to practise their programming skills
- http://scratch.mit.edu/ - To enable student to practise their programming skills using a drop & drag interface.

KEY SKILLS

Literacy:	**Numeracy:**	**Other:**
Absolute cell replication, Compression, Decision, Field, Fill, Filter, Flow chart, Formula, Input, Iteration, List, Loops, Marquee, Network, Node, Output, Photo gallery, Pixel, Print, Query, Record, Relative cell references, Repeat, Search, Sort, Storyboard, Sub-routine, Topology, Validation, Variable.	Intrinsic to computing and including: Calculating averages Finding the minimum and maximum number in a range. Creation of graphs Using xy coordinates	Computational thinking skills • Decomposition • Pattern recognition • Abstraction • Algorithms Independent learning skills including critical thinking and problem solving.

Curriculum Overview for Year 9:

Year 9 – Computer Science		
Key topics		**Assessment**
Autumn 1: Scratch	Students work independently to: • Design a pacman game with reference to the audience and purpose. • Create a pacman game with more than one level using scratch. They use xy coordinates, if statements, movement, sensing, variables, broadcast and receive, sound. • Create backgrounds & sprites, including editing sprites in the gallery.	The game created is assessed according to the skills demonstrated i.e. complexity of script employed. Homework set electronically via Class Charts.
Autumn 2: Animation Draw plus) Spreadsheet (Excel)	Students work independently to: • Plan an animation using storyboard techniques • Create an animation including the skills of cloning and onion skinning, use of backgrounds, layers, grouping, frame rate timings. • Evaluate their work • Plan a spreadsheet that will meet the audience and purpose. • Create a spreadsheet using currency formatting, formula (+ - * /, absolute cell referencing, average, min, max), data validation, pivot tables, macros • Use a spreadsheet for 'what if' analysis including goal seek. • Present information in a spreadsheet using graphs and reports.	Storyboard techniques The animation that has been created. Homework set electronically via Class Charts. The spreadsheet created is assessed according to the skills demonstrated i.e. formula used. .
Spring 1: Database (Access) HTML	Students work independently to: • Plan a database including choosing sensible field names and data types. • Create a database using validation • Use a database to search for answers to questions using OR, NOT and AND. • Create a relational database and query it. • How to use HTML to create a webpage using: Open and close tags, head, body and title and table tags. How to insert an image. How to create hyperlinks. • Use of CSS to create style for the pages for headings, the body and paragraphs	The database created is assessed according to the skills demonstrated i.e. formula used. Homeworks from the year 9 homework booklet.

Spring 2: Python	Students work independently to: • Write programs that use print and input commands. • Use variables and constants • Use selection and iteration • Create and use arrays	The Python programs created are assessment against key skills. Homework set electronically via Class Charts.
Summer 1: Supersports project	Students work independently to: • Create a solution for Supersports. This will include: • Development of leaflet materials to promote the business • Development of a spreadsheet to calculate profit • Development of a database to record details of customers • Mail merged literature.	The supersports project is assessed against the client brief using criteria similar to the BTEC DIT course.
Summer 2:	Start of GCSE Course	

Suggestions for independent study and home support:

- www.teach-ict.com – To access information and help sheets
- http://scratch.mit.edu/ - To enable student to practise their programming skills using a drop & drag interface.

KEY SKILLS

Literacy:	**Numeracy:**	**Other:**
Absolute cell replication, Background, Compression, Field, Fill, Filter, Formula, Frame rate, Grouping, Layers, Output, Pivot table, Print, Query, Record, Relative cell references, Repeat, Search, Sort, Storyboard, Timing, Validation, Variable.	Intrinsic to computing and including: Calculating averages Finding the minimum and maximum number in a range. Creation of graphs Using xy coordinates	Independent learning skills including critical thinking and problem solving.

Curriculum Overview for Year 10:

Year 10 – Computer Science GCSE			
Exam Board & course title/code	Unit	Date of Exam	% of Total Exam
OCR Computer Science J277	**Paper 1:** Computer Systems	Summer year 11	50%
	Paper 2: Computational thinking, algorithms and programming	Summer year 11	50%

Key topics	Course content	Assessment
Year 9 Summer 2 / Year 10 Autumn 1: System architecture Memory & Storage Programming: sequencing, selection & iteration	Students will be taught: • Architecture of the CPU • CPU performance • Embedded systems • Primary storage • Secondary storage • Units of data Python skills: print, input, constants, variables, data types, selection (if/elif), iteration (for/while).	Exam questions at the end of each section of theory. Questions from the year 10 homework booklet.

Autumn 2:
 	
Memory & Storage continued
 	
Programming: flow charts, pseudocode, testing
	Students will be taught:
	• Data storage
	  ○ Numbers
	  ○ Images
	  ○ Sound
	  ○ Characters
	• Compression
	
	Python skills: flowcharts, testing, trace tables, pseudocode.
		Exam questions at the end of each section of theory.
		
		Questions from the year 10 homework booklet.

Spring 1: Computer Networks Programming skills: string manipulation, files, arrays, SQL	Students will be taught: • Network types & performance • Client servers • Hardware for networking • The internet • Network topologies • Wired & wireless networks Python skills: string manipulation, open, read, write & close files, logical operators, write simple SQL, use lists (including 2 columns)	Exam questions at the end of each section of theory. Questions from the year 10 homework booklet.
Spring 2: Computer Networks continued	Students will be taught: • Encryption • Network protocols • Network dangers	Exam questions at the end of each section of theory.

Network security Programming skills: Sub programs & programming challenges	Python skills: SQL in Access, functions & procedures, validation, short programming challenges	Questions from yr. 10 homework booklet.
Summer 1: System software Programming skills: system life cycle	Students will be taught: • Operating systems • Utility programs Python skills: Creating 2 programs following life cycle: analysis, flow diagram, pseudocode, coding, testing, validation, improving, evaluating.	Exam questions at the end of each section of theory. Questions from the year 10 homework booklet.
Summer 2: Ethical, legal, cultural and environmental concerns Algorithms Searching & sorting algorithms	Students will be taught: • Environmental issues • Ethical & cultural issues • Computer legislation • Open source v proprietary software • Algorithmic thinking • Linear & binary searching • Bubble, merge & insertion sort	Exam questions at the end of each section of theory. Questions from the year 10 homework booklet.

Suggestions for independent study and home support:

- www.teach-ict.com – To access information and help sheets
- http://www.codecademy.com/– To enable students to practise their programming skills
- OCR GCSE Computer Science My Revision Notes by George Rouse

KEY SKILLS

Literacy:	**Numeracy:**	**Other:**
Accumulator, Adware, ALU, anti-malware, Application software, BIOS, Blagging, brute force attack, Bus, cache memory, censorship, client, client-server network, clock speed, cloud computing, command-line, Compression software, control unit, copyright, core (CPU), cyberbullying, Defragmentation, denial-of-service attack, device driver, DNS, embedded system, encryption, Ethernet, E-waste, Firewall, GUI ……	Converting between different data capacity units e.g. MB to GB. Calculating the size of a file Calculating whether the processor speed has halved, doubled or quadrupled. Logic gates & basic Boolean algebra Organising data into sequence.	Computational thinking skills • Decomposition • Pattern recognition • Abstraction • Algorithms Independent learning skills including critical thinking and problem solving.

Year 10 – BTEC DIT			
Exam Board & course title/code	Unit	Date of Exam	% of Total Exam
BTEC Tech Award in Digital Information Technology 603/7050/6	Component 1 – Exploring user interface design principles and project planning techniques	May year 10	30%
	Component 2 – Collecting, presenting and interpreting data	December year 11	30%
	Component 3 – Effective digital working practices	Summer year 11	40%
Key topics	Course content		Assessment
Summer 2 / Autumn 1: Component 1 Learning Aim A Component 1 Learning Aim B & C Component 3 Learning Aim A	Students will be taught: • User interfaces • Factors and influences • Audience needs • Design principles • Design psychology • Designing efficient user interfaces • Project planning techniques • Creating project plans • Initial designs • Developing a user interface • Review • Investigating communication techniques • Examining cloud storage • Cloud technologies • Modern team working • Inclusivity & accessibility • Impact of modern technologies		3 end of unit tests, PIA marked

Autumn 2: Component 1 Practice CA Component 3 Learning Aim B	Students will be taught: • Cyber security • Internal threats and breaches • User restrictions and weaknesses Students will complete a practice controlled assessment	
Spring 1: Controlled assessment 1 Component 3 Learning Aim B	Students will be taught: • Data level protection • Policy, backups and data recovery Students start their first controlled assessment	Controlled Assessment Component 3 learning aim B end of unit test, PIA marked
Spring 2: Controlled assessment 1	Students complete their first controlled assessment	Controlled Assessment
Summer 1: Component 2 Learning Aim A Component 3 Learning Aim C	Students will be taught: • Characteristics of data and information • Representing information • Ensuring data is suitable • Data collection • Quality of information • Sectors that use data modelling • Threats to individuals • Shared data • Impact on the environment • Equal access & net neutrality	Component 2 learning aim A end of unit test, PIA marked
Summer 2: Component 2 Learning Aim B Component 3 Learning Aim C	Students will be taught: • Spreadsheet basics • Data manipulation methods • Acceptable use policies • Data protection • Criminal use of computer systems	Component 3 learning aim C end of unit test, PIA marked Year 10 PPE. PIA marked.

Suggestions for independent study and home support:
This is a new course. We expect revision guides and workbooks to be published during this year.

KEY SKILLS

Numeracy: Spreadsheet formulas	**Other:** Independent learning skills including critical thinking and problem solving.	**Literacy:** Key terms: GUI, WIMP, icon, sensor, demographics, breadcrumbs, auto-fill, shortcut, link, GANTT, mood board, waterfall, agile, scrum, audience, milestone, timescale, storyboard, accessibility, blackspot,

	Time management	encryption, tethering, scalability, redundancy, multicultural, inclusivity, malware, virus

Year 10 – Cambridge National Certificate iMedia J834			
Exam Board & course title/code	Unit	Date of Exam	% of Total Exam
OCR Cambridge National in Creative iMedia J834	R093 – Creative iMedia in the media industry	Summer Year 11	40%
	R094 – Visual Identity & digital graphics - CA	Summer Year 10	30%
	R099 – Digital games - CA	Summer Year 11	30%

Key topics	Course content	Assessment
Autumn 1: R093 – Creative iMedia in the media industry	Students will be taught: • Sectors and products of the media industry • Job roles in the media industry • Purpose, style, content & layout • Client requirements and audience • Research • Media codes • Cameras and lighting Pre-production Planning to include: • Mindmaps • Scripts • Storyboards • Visualisation diagrams • Wireframes & flowcharts	All through lesson, handwritten exam questions where answers will be given – self/peer assessed Homework on Class Charts with exam questions – teacher assessed Test 1 – (Media industry and product design) PIA Test 2 -(Pre-production planning) PIA
Autumn 2: R094 – Visual Identity & digital graphics	Students will be taught: • Purpose of visual identity • Visual identity components and elements • Design & layout • Fire types & formats • Asset sourcing, licenses & permissions • Planning graphics	Test 3 (Visual identity & assets) PIA Homework Booklet questions to help prepare

Spring 1: R094 – Visual Identity & digital graphics	Students will be taught: • Sourcing & creating assets • Isolating & retouching assets • Compiling an image • Other basic tools • Text & advanced tools • Typography & gradients • Saving & exporting Students will practice their Graphics skills. • Create a poster • Create a CD cover • Create a DVD cover • Create a book cover	Test 4 (Creating visual identity & digital graphics) This is preparation theory for CA1. Homework Booklet questions to help prepare
Spring 2: R094 – Visual Identity & digital graphics	Students will prepare real Graphics CA brief	CA1
Summer 1: R094 – Visual Identity & digital graphics R099 – Digital games	Students will prepare real Graphics CA brief Students will be taught: • Types, characteristics, and conventions of digital games	CA1
Summer 2: R099 – Digital games	Students will be taught: • Types, characteristics, and conventions of digital games Revision for PPEs PPEs	This is preparation theory for CA2. End of year PPE. PIA marked.

Suggestions for independent study and home support:

- Hodder Creative iMedia Textbook
- My Revision Notes: OCR Cambridge Nationals in Creative iMedia

KEY SKILLS

Numeracy: Creating a work plan / Gantt chart and converting file size and image sizes.	Literacy: Moodboard, mindmap, storyboard, script, visualisation diagram, cameras and lighting, file formats, planning, pre-production, research, work plans, schedules, GANTT, hardware and software, legislation, assets, digital, graphics, purpose, design, layout, client brief, requirements, resources, physical, idea, images, text, typography, colour, swatches, video, sound, feelings, logos, assets, properties, embedded content, navigation, CSS, Hyperlink, sitemap	Other: Independent learning skills including critical thinking and problem solving Time management Communication skills Evaluation skills

Curriculum Overview for Year 11:

Year 11 – Computer Science GCSE			
Exam Board & course title/code	Unit	Date of Exam	% of Total Exam
OCR Computer Science J277	**Paper 1:** Computer Systems	Summer year 11	50%
	Paper 2: Computational thinking, algorithms and programming	Summer year 11	50%

Key topics	Course content	Assessment
Autumn 1: Ethical, legal, cultural and environmental concerns	Students will be taught: • Environmental issues • Ethical & cultural issues • Computer legislation • Open source v proprietary software Students will be taught about logic gates and basic Boolean algebra	Exam questions at the end of each section of theory. Questions from the year 11 homework booklet.
Autumn 2: Ethical, legal, cultural and environmental concerns	Students will be taught: • Environmental issues • Ethical & cultural issues • Computer legislation • Open source v proprietary software	Exam questions at the end of each section of theory. Pre-public exams. Questions from the year 11 homework booklet.
Spring 1: Translators and facilities of language Revision of paper 1 topics	Students will be taught: • Characteristics and purpose of different levels of programming language, including low level languages • the purpose of translators • the characteristics of an assembler, a compiler and an interpreter • common tools and facilities available in an integrated development environment (IDE): editors, error diagnostics, run-time environment, translators.	Exam questions at the end of each section of theory. Questions from the year 11 homework booklet.
Spring 2: Programming exam questions / Robust programs Revision of paper 1 topics	Students will be taught: • The elements of defensive program design • Maintainability of programs • The role of testing • Types of program errors • Difference between iterative & terminal testing • Selecting suitable test data.	Exam questions at the end of each section of theory. Questions from the year 11 homework booklet.

		Pre public exam for paper 2
Summer 1: Revision	Students will review their learning of paper 1 and 2 topics including further in class pre-public exams.	Further pre-public exams.
Summer 2:		

Suggestions for independent study and home support:

- www.teach-ict.com – To access information and help sheets
- http://www.codecademy.com/– To enable students to practise their programming skills
- OCR GCSE Computer Science My Revision Notes by George Rouse

KEY SKILLS

Literacy:	**Numeracy:**	**Other:**
Amplitude, array, ASCII, assembler, binary, binary search algorithm, bit, bit depth, bit rate, Boolean, bubble sort, byte, casting, character set, check digit, colour depth, compiler, compression, constant, data type, denary, erroneous data, extreme data, flow diagram, function, gigabyte, hexadecimal, high-level language, IDE, insertion sort, integer, interpreter, iterative testing, kilobyte, linear search algorithm, logic gate, lossless & lossy compression, low-level language……	Organising data into sequence. Logic gates & basic Boolean algebra Converting MB to GB etc. Converting numbers from denary to binary and back again. Adding binary numbers together Multiplying and dividing in binary. Hexadecimal calculations Calculations to find out the size of an image. Plot graphs to show frequency. Calculations to find the size of a sound file.	Computational thinking skills • Decomposition • Pattern recognition • Abstraction • Algorithms Independent learning skills including critical thinking and problem solving.

Year 11 - BTEC DIT			
Exam Board & course title/code	Unit	Date of Exam	% of Total Exam
BTEC Tech Award in Digital Information Technology 603/7050/6	Component 1 – Exploring user interface design principles and project planning techniques	May year 10	30%
	Component 2 – Collecting, presenting and interpreting data	December year 11	30%
	Component 3 – Effective digital working practices	Summer year 11	40%

Key topics	Course content	Assessment
Autumn 1: Component 2 PSA	Students will undertake the controlled assessment in this period of time.	Assessment of controlled assessment work against criteria
Autumn 2: Component 2 PSA	Students will undertake the controlled assessment in this period of time.	Assessment of controlled assessment work against criteria
Spring 1: Component 3 Learning Aim D	Students will study: • Investigating information • Creating flowcharts • Creating & interpreting systems diagrams • Use of tables and written information	Test PIA marked
Spring 2: Component 3 Revision	Students will revise Learning Aim A/B	Past papers during lessons
Summer 1: Component 3 Revision	Students will revise Learning Aim C	Past papers during lessons
Summer 2:	Summer exam for Component 3	

Suggestions for independent study and home support:
This is a new course. We expect revision guides and workbooks to be published during this year.

KEY SKILLS		
Literacy:	**Numeracy:**	**Other:**

Data flow diagram Information flow diagram System Diagram Flowchart Variable Chart Range Maximum Minimum	The creation of tables and graphs to represent data and flow charts for the planning of projects.	Independent learning skills including critical thinking and problem solving. Time management

Year 11 – Cambridge National Certificate iMedia J834			
Exam Board & course title/code	Unit	Date of Exam	% of Total Exam
OCR Cambridge National in Creative iMedia J834	RO93 – Creative iMedia in the media industry	Summer Year 11	30%
	R094 – Visual Identity & digital graphics - CA	Summer Year 10	30%
	R099 – Digital games - CA	Summer Year 11	40%

Key topics	Course content	Assessment
Autumn 1: Controlled Assessment R099 Digital Games	Students will be taught: Students will prepare the evidence for the Website CA	CA
Autumn 2: Controlled Assessment R099 Digital Games	Students will be taught: Students will prepare for the Gaming project Jan exam for R081	CA
Spring 1: Controlled Assessment R099 Digital Games	Students will Students will prepare the evidence for the Gaming CA	CA
Spring 2: R093 – Creative iMedia in the media industry	Students will study: • Legal Issues • Regulation, certification & classification • Health & Safety • Distribution platforms and media • File types and formats	Test (R093 Pack C) PIA Homework questions to help prepare for exam
Summer 1: R093 Revision	Students will Students will revise all topics for R093	Sample / past exam paper questions during lessons
Summer 2:	Summer exam for R093	

Suggestions for independent study and home support:

- Hodder Creative iMedia Textbook
- My Revision Notes: OCR Cambridge Nationals in Creative iMedia

KEY SKILLS		
Numeracy: Creating a work plan / Gantt chart and converting file size and image sizes. Gaming Construct 3 skills / possible programming	Literacy: Moodboard, mindmap, storyboard, script, visualisation diagram, cameras and lighting, file formats, planning, pre-production, research, work plans, schedules, GANTT, hardware and software, legislation, assets, digital, graphics, purpose, design, layout, client brief, requirements, resources, physical, idea, images, text, typography, colour, swatches, video, sound, feelings, logos, assets, properties, embedded content, navigation, CSS, Hyperlink, sitemap, Construct3 skills	Other: Independent learning skills including critical thinking and problem solving Logic skills Time management Communication skills Evaluation skills

Curriculum Overview for Year 12:

Year 12 – Computer Science GCE			
Exam Board & course title/code	Unit	Date of Exam	% of Total Exam
OCR H446/01	Unit 1: Computer Systems	Summer – Year 13	40%
OCR H446/02	Unit 2: Algorithms and Programming	Summer – Year 13	40%
OCR H446/03	Unit 3: Programming Project	N/A Created during Year 13 lessons	20%

Key topics	Course content	Assessment
Autumn 1: Components of a computer Computational Thinking	**Structure & function of the processor:** ALU, Control Unit, Registers, Buses, Fetch-Decode-Execute, Performance, Pipelining, Architecture, RISC & CISC, GPUs. **Input, Output, Storage**: Choices, Types, RAM, ROM, Virtual **Programming Techniques:** Sequence, Iteration, Branching, Recursion, Variables, Modularity, Functions, Procedures, Parameters, IDEs, Object-oriented	Exam questions Homework End of unit tests
Autumn 2: Systems Software Programming Techniques	**Systems Software:** Functions , Types of OS, Memory management, Scheduling, BIOS, Drivers, Virtual machines **Applications:** Selecting applications, Utilities, Open/Closed source, Translators, Compilation, Linkers, Libraries **Computational Thinking**: Abstraction, Planning ahead, Thinking procedurally, Logic, Concurrence	Exam questions Homework End of unit tests
Spring 1: Software Techniques Computational Methods	**Software Development:** Lifecycle, Methodologies, Algorithms **Types of programming language:** Paradigms, procedural, assembly language, memory addressing, object-oriented **Computational Methods:** Features, Problem recognition & decomposition, Divide & Conquer, Abstraction, Backtracking, Data Mining, Heuristics, Performance modelling, Visualisation	Exam questions Homework End of unit tests

Spring 2: Exchanging Data Practice NEA	**Compression:** Lossy / Lossless, Encryption, Hashing **Databases:** Relational, Normalisation, SQL, referential integrity, processing methods, ACID **Develop programming including practice NEA (Non Examination Assessment) task**	Exam questions Homework End of unit tests
Summer 1: Networks & Web Technologies Practice NEA	**Networks:** Protocols, Standards, TCP/IP, DNS, Layers, LAN / WAN, Packets, Security, Hardware, Client-server, Peer-to-peer **Develop programming including practice NEA (Non Examination Assessment) task**	Exam questions Homework End of unit tests
Summer 2: Networks & Web Technologies Practice NEA	**Internet:** HTML, CSS, Web forms, JavaScript, Search engines, PageRank **Develop programming including practice NEA (Non Examination Assessment) task**	Exam questions Homework End of unit tests

Suggestions for independent study and home support:

- Access past exam questions for the previous OCR Computer Science specification and use these to help you practice answering past exam question
- Watch craig'n'dave You Tube clips and make notes and flash cards independently.
- Get and use "PG online" textbook by PM Heathcote

KEY SKILLS

Literacy:	Numeracy:	Other:
Spelling key terms Learning definitions Analytical skills	Lots of Maths throughout including: Data Types, Data Structures, Boolean Algebra, Algorithms, Programming Techniques	Problem Solving Communication Independent skills Time management

Curriculum Overview for Year 13:

Year 13 – Computer Science GCE			
Exam Board & course title/code	Unit	Date of Exam	% of Total Exam
OCR H446/01	Unit 1: Computer Systems	Summer – Year 13	40%
OCR H446/02	Unit 2: Algorithms and Programming	Summer – Year 13	40%
OCR H446/03	Unit 3: Programming Project	N/A Created during Year 13 lessons	20%

Key topics	Course content	Assessment
Autumn 1: Data Types NEA	**Data Types:** Standard types, Sign & Magnitude, Two's complement, Binary addition & subtraction, Hex, Converting Binary to Hex, Floating point numbers, Floating Point Arithmetic, Bitwise manipulation & masks, ASCII, UNICODE **NEA (Non Examination Assessment) task** **Algorithms:** Design of algorithms, Big O notation, Searching and Sorting algorithms	Exam questions Homework End of unit tests NEA
Autumn 2: Data Structures NEA	**Data Structures:** Arrays, Lists, Records, Tuples, Linked-list, Graph, Stacks, Queues, Trees, Binary search tree, Hash table, How to create, traverse, add & remove data from the above. **NEA (Non Examination Assessment) task** **Algorithms:** Graph Traversal algorithms, efficiency, optimisation and complexity of algorithms.	Exam questions Homework End of unit tests NEA
Spring 1: Boolean Algebra NEA	**Boolean Algebra:** Defining problems, Manipulating expressions, Karnaugh maps, De Morgan's Laws, Commutation, Logic gates, Truth tables, D type flip flops, half & full adders. **NEA (Non Examination Assessment) task**	Exam questions Homework End of unit tests NEA

Spring 2: Legal, Moral, Ethical & Cultural Issues Algorithms NEA	**Legislation:** Data Protection Act, Computer Misuse Act, Copyright Design & Patents Act, Regulation of Investigatory Powers Act **Moral, Social & Ethical issues:** Workforce, AI, Environment, Censorship, Monitoring, Analysing, Piracy. **NEA (Non Examination Assessment) task**	Exam questions Homework End of unit tests NEA
Summer 1: Exam Preparation	**Exam preparation**	Exam questions Homework NEA

Suggestions for independent study and home support:

- Access past exam questions for the previous OCR Computer Science specification and use these to help you practice answering past exam question
- Watch craig'n'dave You Tube clips and make notes and flash cards independently.
- Get and use "PG online" textbook by PM Heathcote

KEY SKILLS

Literacy:	Numeracy:	Other:
Spelling key terms Learning definitions Analytical skills	Lots of Maths especially in year 13. Including: Data Types, Data Structures, Boolean Algebra, Algorithms, Programming Techniques	Problem Solving Communication Independent skills Time management

Subject Assessment Reporting and Recording:

ASSESSMENT, MARKING AND REPORTING POLICY DOCUMENT FOR THE COMPUTING & IT DEPARTMENT

The aim of marking students work is to:

- Let students know what they need to do to improve in order to set realistic goals and targets.
- Monitor students' progress and give them an indication of their level of achievement.
- Provide information for parents about students' progress.
- Acknowledge achievement.

The Computing & ICT department will achieve these aims by planning the following types of assessment into their schemes of work:

- Formative/diagnostic - what do students need to do to improve?
- Summative - recording of overall achievement/attainment at critical points, e.g., end of year exams, Teacher Assessments at end of KS3, KS4 & KS5 which are moderated.

As a result of these assessments, teachers will adjust their lesson plans and schemes of work.

Teachers will record marks in order to:

- monitor progress over time.
- provide accurate historic information for students/parents/other teachers.
- provide evidence to support reporting and target setting.

Marks should be accessible to Line Managers and will be recorded in an electronic mark book.

In order to support these aims:

- Teachers will access baseline data, attendance and other information such as SEND, EAL and G&T in order to inform their planning and teaching.

Key Responsibilities

The responsibilities of Students

- There should be a student response to the teacher comments; something should be done; it could be to do a task set by the teacher, redo a piece of work, to complete or improve the work they have already handed in, to check and learn some spellings, to do some more research, to complete a new task, to set a new target to ensure future goals are understood.
- Student feedback may be completed in green pen if appropriate but may well be produced on the computer.

The responsibilities of Teachers

- Work will be regularly marked by the teacher in red or purple.
- All homework checked & logged in mark book.
- **Key stage 3** projects to be assessed using PIA comments 2/3rds of the way through the project, thereby allowing time for students to improve their work (DIRT lesson).
- **Key stage 3** projects to be assessed at the end of each project to establish if students are:

    - ‘Exceeding National Standards’ is that of a child who is building up skills towards a 9/8 at GCSE Level
    - ‘Working Above National Standards’ is that of a child who is building skills towards a 7/6 at GCSE Level
    - ‘At Expected National Standard’ is that of a student who is building skills towards a 5 at GCSE Level

- 'Working Towards National Standard' is a student who is building skills towards a 4/3 at GCSE Level.
- 'Working Below National Standards' is a student who is building skills towards an outcome of an 2/1 or lower at GCSE Level.

Teachers should keep a record of the performance of each key stage 3 student (using the reporting numbers 1 – 9) for each project so that trends can be quickly observed and used for accurate data sweeps.

- **Key stage 4 & 5** controlled assessment should be assessed weekly and progress recorded using a tracking grid. Feedback should be given in line with guidance given by the exam boards.

- **Key stage 4 & 5** theory work should be assessed approximately each fortnight (or after every 5th lesson). The form of this assessment will depend on the topic and the stage of the course. This could be via the use of past exam questions or via the assessment of a project using PIA principles. After a series of theory lessons has been taught (perhaps over several weeks), the teacher should allow time for DIRT activities to take place.

- Teachers will use the PIA criteria (Appendix 1. PIA statements will be linked where possible to Big Ideas (see appendix 2) or GCSE/GCE grades.

- The school marking codes should be used for Literacy Feedback within written work.

- At the beginning of each Key Stage a Target Level or Grade will be decided for each student. Marking will show the student the progress they are making against their target.

- Each term all teachers will give a progress report against National Standards and Individual Progress. The following pattern will be adhered to:

Year group	**Pattern of reporting**
7	Twice a year, reporting classwork & homework effort and progress (only nce a year). The judgement is to be based on the outcomes of the projects that have been assessed.
8	
9	
10	Predicted grades, classwork & homework effort grades twice a year.
11-13	Predicted grade, classwork & homework effort grades three times a year

- All assessments will be supported by evidence and moderated (appendix 3) by the department. These results will be analysed in order to identify any underachievement where intervention may be necessary.

Homework

Years 7 to 11 are set homework electronically via Class Charts. for year 7-9 students are set videos to watch or information to read and take notes, followed by a quiz on Class Charts. They also have occasional spelling tests via Class Charts as part of our literacy strategy. Year 10-11 students have a mixture of videos, information to take notes on, Class Charts quizzes, Seneca Learning, and written exam style questions.

Appendix 1:

PIA

P – Positive

What is really good about this work? Which skills have been demonstrated really well here?

I – Improvement

What mistakes have been made? Which areas have not been done correctly?

A – Action

The task that should now be completed so that the student can make the required improvement
(Please note that sometimes that the Improvement and Actions could be very similar, if not one and the same)

S – Student Response

This is the next part of the feedback and represents a part of the ongoing dialogue. The student should be **acting** upon the advice given.

Appendix 2:

Big ideas

- KNOWLEDGE
- DESIGN /PLANNING
- CREATING A SOLUTION
- TESTING / DEBUGGING
- EVALUATION

Appendix 3:

Moderation procedures:

1. A marking exercise consisting of several pieces of work will be distributed to staff.
2. Staff will assess the work against the AWL sheet and submit the sheet to the HOD or 2nd in dept (position vacant).
3. The HOD or 2nd in dept will check that staff have assessed the work as expected against the AWL.
4. Follow up conversations will take place with the dept or on a one to one basis (using department meeting time).

Appendix 4:

The big ideas are applied to each of the topic areas. See the AWL sheets in appendix 6:

Topic	**Year 7**	**Year 8**	**Year 9**
Algorithms & Programming	Control/ Python/ Scratch	Control/ Python/ Scratch	Python/ Scratch
Data & Data representation	Databases/ Binary	Databases	Databases/ Logic
Hardware & processing	Leaflet on computer hardware / Storage / Sensors (covered in control)	Networks Sensors (covered in control)	Threats to computer systems
Communication & networks	HTML / E-safety	Web design	HTML / Websites
IT	Animation / SS	Graphics / SS	Animation / Spreadsheet/

Medium Term Plans for Year 7

			Core knowledge	Core Skills			
Year	**Term**	**Units of Work**	**Knowledge**	**Design / Planning**	**Creating a solution**	**Testing/ debugging**	**Evaluation**
7	1	E-Safety animation	• Explain the meaning of the term e-safety • Explain how to use the internet safely • Explain what a chatroom is and how you can stay safe in one • Define the term cyber bullying • Give some examples of cyber bullying • Explain how to cope if you are being cyber-bullied. • Explain how to reduce the risk of being cyber-bullied. • Explain what a storyboard is • Explain what should be included in a storyboard • Explain why it is important to develop a storyboard • Be able to talk about their animation using key terminology such as clone and onion skin.	Create a storyboard for the e-safety animation showing audience, purpose, frame rate, colour scheme, annotation of the story.	• Use cloning and onion skinning • Insert a background frame • Use standard shapes and fill to create the images • Use the gallery of objects to create an animation	As a result of peer and self review, further develop the animation to make it look more professional.	• Describe what is good about their animation • Explain why their animation is good • Analyse what needs to be improved in their animation
		Harry Plotter spreadsheet	State some spreadsheet key terms & use them when talking about spreadsheets. Explain the purpose of a spreadsheet. State what verification is. State what validation is.		Enter data into a spreadsheet. Format a spreadsheet (including currency, borders, fill & cell merging) Perform calculations using simple formulae (sum, average, min, max) Enter data into a spreadsheet	Change variables. Use verification techniques to check data for errors. Use validation techniques to ensure data entered is sensible.	

					Add new variables and replicate formulae. Create a bar graph and a pie chart & use titles and axis labels. Sort & filter data	Use goal seek to answer a question	
	2	School meals database	List key terms used when describing a database e.g. field, record, file.	Plan fields /searching questions to be used in the planet database. Research information about planets and complete the data entry form accurately. Design a database table selecting suitable field names & data types.	Set up a database and add field names & data. Select the most suitable data types. Sort the data Create a form to add data to the database. Add a data validation rule	Test the data validation rule works. Set up a database search and enter sensible search criteria. (AND, > <)	
		Python Programming	Explain the purpose of the PRINT & INPUT commands.		Write a simple program using the PRINT & INPUT command. Use a number variable Perform a calculation using variables. Write a simple program using text strings	Run programs to see if they work and adjust their code to make their program work.	

					Use If statements in a program. Use a score variable.		
	3	Control…. Greenhouse	Identify the symbols for inputs and outputs on a flowol chart and state what each of the symbols does. Identify an output and an input and explain the difference between them. Name some sensors and explain how they are used / operate.	Plan the outcomes for the greenhouse mimic using values for some of the outputs.	Set up and run a flow chart to operate a zebra crossing/ lighthouse /bridge lights / kitchen / greenhouse. Re-create the window aspect of the greenhouse using sub-routines and test this aspect of the flowchart to see if the window opens and shuts more smoothly.	Test to see if the control system set up meets its purpose	Evaluate the work explaining what went well and what could be improved
		HTML project	Define the key terms web browser, search engine, URL and keywords. Explain how search engines work page. Identify the purpose of various tags which can be used when creating a web page.		Use Boolean operators such as OR & NOT, as well as “” and * when using a search engine. Use tags to insert instructions to make a web page (opening, closing, head, body, title, headings). Insert image tags to insert an image onto their web page. Insert image tags into their web page. Use CSS to create style for the web pages (headings, body, paragraphs, backgrounds,	Peer & self assess web pages created to check they meet the intended audience.	

					font sizes, borders, navigation)		

Medium Term Plans Year 8

			Core knowledge	Core Skills			
Year	**Term**	**Units of Work**	**Knowledge**	**Design / Planning**	**Creating a solution**	**Testing/ debugging**	**Evaluation**
8	1	Graphics & Scratch – Ping Pong	Demonstrate and discuss how computers store and represent black and white and colour images. Understand why computers need to store images, giving examples, and consider what they need to do with them. Explain that colours can be selected on a graphics package using RGB colour codes or hex codes. Explain why graphics files need to be compressed. Explain the difference between lossy and lossless compression. Understanding the difference between exporting images and saving images in Drawplus.		Draw a simple line using the Pen tool. Add Quick Shapes to a page. Use the Node tool to edit Quick Shapes. Use the fill tool to colour a shape. Change page setup. Using transparency and selection when exporting images. Using gallery objects, grouping and ungrouping and other editing skills. Importing graphics into Scratch Shrink, enlarge or rotate existing sprites. Using script to program sprites including: Positioning of sprites, Using IF statements, Using Movement, Using Sensing, Adding Variables, Using Broadcast and Receive, Adding Costumes for Backgrounds,	Play their game to ensure it works correctly. Adjust their code to fix any bugs.	

					Adding Operators, adding sound.		
		Spreadsheet - Disco	Be able to explain how to create formulas. Use subject specific terms such as font, cell format, sum.	Work out which formula you are likely to need so that you can calculate the profit from running the school disco. Work out the what formatting to use on your plan – colour scheme, cell formats e.g. currency, font, borders Identify the audience and purpose for the spreadsheet. Olan the formulas required.	Use the plan to layout the spreadsheet and enter data. Be able to set up a simple formula e.g. + - * / Be able to set up a formula using the SUM function. Format the spreadsheet as planned (borders, fonts, colours, titles, currency, text wrap, merged cells) Use absolute cell referencing. Create graphs of income, costs & profit. Label graphs accurately – title/ axis labels. Create a pie chart with labels. Use data validation on the ticket sales data. Use data validation (independently) on another variable e.g. the ticket price. Use Conditional formatting & Macros.	Change the number of tickets sold and view how profit levels change. Record the changes in income, costs and profits change in a report. Test their validation rule. Change data using goal seek.	Recommend a day for the Disco. Explain what happened when they changed data using goal seek. Comment on the success of the spreadsheet. Comment on the quality of the spreadsheet. Identify improvements that could be made to the work. Work with a peer to identify and comment on improvements to the work.
	2	Database – Travel Agents	Use key terms in the write up such as field, data type, record, query.	Choose appropriate field names and data types. Be able to design validation rules.	Create a database using the field names and data types planned. Add at least 10 holidays ensuring they have chosen suitable holidays for the questions. Added a data validation rule . Added a lookup wizard.	Test their Validation rule. Run their queries to test whether they work.	Compare the database with the features list and identify what they have/ have not included. Explain what was good about the database.

				Pick suitable questions that could be answered by a Holiday Database.	Create a simple query. Create a query using two fields. Create an OR query. Create a NOT query. Create a query using > <.		Explain what could be improved. Justify why improvements are needed.
		Control - Autohome	Identify which sensors are important in the operation of the autohome. Identify the inputs and outputs on a mimic. Explain what the different parts of the mimic do.	Write some statements explaining what the autohome will do when it is built using values. Include Inputs, Outputs and Motors.	Create some flowcharts using Inputs, Outputs and Motors. Create a flowchart using > <. Accurately use loops. Create sub routines.	Run their Flowchart to see if it works. Adjust their flowchart to fix bugs.	Write some comments stating the good/ best parts of the autohome diagram. Explain the flow chart work is a success. Identify improvements.
	3	Programming - Python	Explain what the Print statement does Explain what the Input statement does Explain the term iteration.		Use the print command Use the input command Use a variable in your program Use If statements and indents Write a simple program that uses iteration Use a simple list Use a list that holds two columns of data Create a simple game using the list.	Run their program to see if it works. Adjust their code to fix bugs.	
		Website		Plan their websites using a storyboard planning sheet Include audience and purpose	Create their websites using openElement. Include a table. Add pictures and hyperlinks to all pages. Add a scrolling marquee and photo gallery.		Improve their storyboards. State what is good & bad about the website and explain why in relation to the

				Plan includes design elements such as fonts, hyperlinks. Research at text and images for the website.	Created a website that is fit for purpose using text and images. Include an animated gif or simple animated text, and a mouseover effect . Include special effects e.g. a JavaScript code or an applet, own recorded video etc.		audience and purpose. Explain how they could improve the website. Improve it following the peer review received.

Medium Term Plans Year 9

			Core knowledge	Core Skills			
Year	**Term**	**Units of Work**	**Knowledge**	**Design / Planning**	**Creating a solution**	**Testing/ debugging**	**Evaluation**
9	1	Scratch Pacman	Identify the parts of the interface: Sprites, costumes, script, stage Identify key terms (variables, if/else, looping, random,	Create a plan for the Pacman game showing: audience who will play their game, purpose of the game, identified the variables high score / time limit, what the game will look like (design / colour) How the game will be played and how they will win.	Paint a maze sprite – the path Paint a wall sprite Create a pacman sprite with 2 costumes Add commands to move the pacman around the maze. Create the food for pacman Create a variable to add up the score Create a baddie Create a second variable e.g. speed or timer	As a result of peer and self review, further develop the game to make it look more professional and improve functionality	Describe what is good about their game Explain why their game is good Analyse what needs to be improved in their game
	2	Python	Explain how print, input, variables, if/else and while loops work in their programs Explain iteration	Design an algorithm (flow chart) to solve a python program	Create and edit programs using Python that uses the print, input, variables. if/else and while loops	Test existing programs by running the code and to improve the code	Evaluate progress of development of programs using review sheet.

			Explain the difference between a for loop and a while loop To develop knowledge of how lists can be used.		Create and edit simple lists in Python		
	3	Animation of advert	Be able to talk about their animation using key terminology such as clone and onion skin, layering, grouping, timing.	Create a storyboard for the advert animation showing audience, purpose, frame rate, colour scheme, description, techniques to be used	Create animation that includes use of cloning and onion skinning, background frame, layers, grouping techniques Use standard shapes and fill to create the images Use the gallery of objects and internet images.	As a result of peer and self review, further develop the animation to make it look more professional.	Describe what is good about their animation Explain why their animation is good Analyse what needs to be improved in their animation
	4	Pacman Sales spreadsheet	Use subject specific terms such as font, cell format, sum. Work out which formula you are likely to need so that you can calculate the profit from selling your game. Explain the impact this has had on your profits Explain formulae and features used using key terms (sum,max,min,average,cell reference, data validation, graphs, titles, conditional formatting, formulae)	Plan formatting of spreadsheet to use on your plan – colour scheme, cell formats e.g. currency, font, borders Plan the formulae and features to be included Identify the audience for the spreadsheet and Identify the purpose of the spreadsheet	Enter data into a spreadsheet and format the spreadsheet as planned (borders, fonts, colours, titles, currency) Use simple formulae (+/-) and functions (SUM) Use the spreadsheet to answer questions (goal seek) Use data validation on the sales data and another variable Created a chart with a title Create conditional formatting techniques	Test spreadsheet by changing some data in a spreadsheet and Test goal seek by screenshots in your explanation Predict a problem & discuss using screenshots the results	Describe what is good about their spreadsheet Explain why their spreadsheet is good Analyse what needs to be improved in their spreadsheet

					Create macros to click buttons to perform functions.		
	5	Database	Use subject specific terms such as field, record, table, query, sort, ascending, descending, data validation, range check. Identify a field, different datatypes. Understand the different operators used in queries e.g. < > <= >= not OR AND	Create a database design structure for a database using appropriate fields, datatypes and data validation	Create a populated database with appropriate field names and data types. Create data validation techniques Sort the data. Identify a field and a record (most) Search the database to find a result (most)	Create queries to test criteria used to search for specific data. Screenshot evidence	Describe what is good about their database Explain why their database is good Analyse what needs to be improved in their database
	6	HTML	Explain and understand how the opening and closing tags work and use of different HTML and CSS tags in web pages. .	N/A	Create HTML/CSS web pages using tags to insert instructions with different headings styles (H1 to H6), changing the font and colour of the font using CSS and/or HTML. Create background colours using CSS/HTML Create font styles using CSS. Create hyperlinks, lists, images, tables, navigation bar and borders/heading bars in CSS.	Testing their web pages in Internet Explorer to view their pages working for the functions created in code.	Evaluate progress of development of programs using review sheet.

Assessment without levels – Modelling

Success Criteria	Pupil	Teacher
Expected level		
Entered sensible data into a spreadsheet		
Used currency or decimal places as a number format		
Used a simple formula e.g. + - / *		
Used a function such as SUM accurately		
Used relative cell referencing to replicate a formula		
Created a useful chart		
Used a simple border		
Annotated the work to indicate features used		
Used a what if investigation, changing data		
Have described one positive aspect of the work and one area that stills needs to be developed		
Working above expected level		
Has entered sensible data onto more than one worksheet		
Formatted spreadsheet using cell merging OR text wrap		
Used other functions such as Average, Max, Min		
Used goal seek to answer a question		
Added a title, axis label and legend to the chart		
Used a validation technique		
Annotated the work to explain the formula used		
Have explained one positive aspect of the spreadsheet model and one area for development		
Exceeding expected level		
Has created a model which is accurate and fit for the purpose		
Used an advanced function such as IF, lookup, Count		
Used absolute cell referencing or 3D cell referencing		
Used a what if calculation to answer a hypothesis OR a what if calculation changing formula		
Used conditional formatting		
Has created macros		
Has explained why the spreadsheet is good and analysed what needs to be improved.		

Assessment without levels – Programming

Success Criteria	Pupil	Teacher
Expected level		
Has used PRINT command to display sentences with text data type		
Has collected INPUT from the user as a stored variable		
Has used a stored variable in a printed sentence		
Has used a stored variable to return result of a calculation		
Has created a simple IF statement		
Has created a WHILE or FOR loop		
Has created a simple LIST with 1 column		
Has tested part of a program and given 1 improvement to fix		
Working above expected level		
Has used a variety of data types (TEXT, INTEGER & FLOAT)		
Has used a stored variable to keep a running count e.g. score = score +1		
Has used an imported FUNCTION e.g. random or sleep		
Has created a LIST with 2 columns		
Has created multiple IF statements using ELIF or ELSE		
Has displayed a text file using an existing notepad file		
Has tested all of the program to check it works correctly		
Exceeding expected level		
Has used 'AND' within an IF statement or another operator		
Has created their own txt file within code		
Has displayed their own txt file (created within code)		
Has added to an existing text file within the code		
Has created their own function using DEF		
Has created a program using multiple techniques within the same program e.g. A play again function that has a while loop with a running score		
Has tested a variety of programs as all/most programs work correctly		

Assessment without levels: HTML

Success Criteria	Pupil	Teacher
Expected level		
Has created a basic web page using HTML		
Has used different headings using H1 – H6		
Used style tags to format a paragraph of text or heading		
Inserted an image into the web page.		
Used a basic table to position some text or graphics		
Created a hyperlink		
Created a list		
Has described one positive aspect of the work and one area that stills needs to be developed		
Working above expected level		
Has created a web page of a good standard, using text fit for purpose		
Has used style tags to format several features e.g. text, background, headings		
Has inserted multiple images into the web page		
Has created a table using some formatting		
Has explained one positive aspect of the webpage and one area for development		
Exceeding expected level		
Used CSS to create a bar with text links		
Used CSS for more complex formatting e.g. borders		
Used html for layout i.e. div tags		
Has created multiple pages that link together		
Has explained why the webpage is good and analysed what needs to be improved.		

Assessment without levels: Databases

Success Criteria	Pupil	Teacher
Expected level		
Planned mostly suitable fields		
Planned mostly suitable data types		
Imported data from a CSV file		
Created a database using key field		
Added 2 or more suitable fields and data types		
Used a simple search		
Has described one positive aspect of the work and one area that stills needs to be developed		
Working above expected level		
Planned suitable fields and a variety of data types		
Added a validation technique		
Added suitable data for 5 records (for additional fields only)		
Used several double searches		
Created an on-screen form		
Has explained one positive aspect of the database model and one area for development		
Exceeding expected level		
Planned suitable fields and data types for a 2nd table		
Searched using logical operators (< >)		
Has used command buttons on the form		
Has created a report to represent the results of a query.		
Has tested the data validation works		
Created a 2nd table which is related to the 1st table		
Has explained why the database is good and analysed what needs to be improved.		

Assessment without levels: Animation

Success Criteria	Pupil	Teacher
Expected level		
Created a storyboard for the animation (at least 3 frames)		
Created an animation of at least 3 frames i.e. there must be movement of an object across 3 frames		
Has used cloning		
Has used onion skinning		
Has used text		
Has used a background		
Has described one positive aspect of the work and one area that stills needs to be developed		
Working above expected level		
Created a detailed storyboard for the animation (6 frames)		
Created a complex animation of at least 6 frames i.e. there must be movement of at least 2 objects across 6 frames.		
Has used grouping to aid the positioning of objects / ungrouping		
Has used two backgrounds		
Has explained one positive aspect of the animation and one area for development		
Exceeding expected level		
Created a storyboard for the animation and annotated using key terms		
Has explained the frame rate used.		
Has used looping (a number of times rather than the default of forever)		
Has explained why the animation is good and analysed what needs to be improved.		

Assessment without levels: Control

Success Criteria	Pupil	Teacher
Expected level		
Planned the greenhouse tasks using inputs and outputs		
Created a flowchart that controls outputs		
Created a flow chart that controls inputs		
Created a flow chart with delays		
Created a flow chart with loops		
Has described one positive aspect of the work and one area that stills needs to be developed		
Working above expected level		
Planned the greenhouse task using motors (forward/reverse)		
Created several flow charts for the greenhouse that accurately use inputs, outputs and loops		
Created a flow chart that uses variables and motors (window)		
Created sub-routines with help.		
Has explained one positive aspect of the greenhouse model and one area for development		
Exceeding expected level		
Planned the greenhouse task using values for the inputs.		
Created the greenhouse flow chart to cover all scenarios, including the use of subroutines		
Has explained why the greenhouse is good and analysed what needs to be improved.		

Assessment without levels: Scratch

Success Criteria	Pupil	Teacher
Expected level		
Has created 3 graphics for a game to be used as sprites or imported sprites		
Has created variables that reset and change e.g. score /timer		
Has used suitable positioning of sprites i.e. XY co-ordinates		
Has used multiple costumes		
Has used forever/ nested ifs accurately e.g. to move a sprite on the screen using key pressed		
Has described one positive aspect of the work and one area that still needs to be developed.		
Working above expected level		
Has created more than 3 graphics for a game to be used as sprites that work well		
Has created additional variables e.g. speed		
Has created broadcast and receive scripts		
Has created multiple costumes for different stages e.g. gameover, levels		
Has created programs that use > / = / <		
Has created programs that use random		
Has explained one positive aspect of the game and one area for development.		
Exceeding expected level		
Has created a game that has additional levels that work well		
Has created a game that uses several complex nested ifs/else		
Has extended game to include more than 1 player or additional features		
Has created a game that includes more complex numbering techniques (AND,NOT,OR)		
Has explained one positive aspect of the game and analysed what needs to be improved.		