

Long Term Departmental Planning Overview

Subject: Computer Science & ICT

Rationale behind curriculum decisions in light of Covid-19;

Curriculum (KS3/4/5)

To ensure skills and knowledge missed that have been missed in light of Covid-19 the following steps have been taken:

Yr 13 - Component 1, this will be consolidated in the first half of the Autumn term (1) and then revisited/revised in subsequent terms as starter activities and through mock exam assessments. These topics are the Processor, Types of Processor, Input, Output and storage, Systems Software incl. Operating Systems, Data types, Exchanging data, Boolean Algebra, Data Structures, Algorithms, Applications Generation, Databases, Networks, Web Technologies and Computing related legislation including Moral and Ethical Issues. A blended learning approach will be used to encourage independent study. These include the Isaac Computer Science platform, Diagnostic Questions and Seneca Learning, these provide a virtual textbook and allow students to complete assessments that pinpoint gaps in knowledge independently. These systems will enable staff to see what parts of the curriculum have been revised.

Yr 11 – It is no longer compulsory for the NEA to be completed under supervised conditions, in the time that would have been used for this Component 1 will now be revisited. The delivery and assessment of this will be completed through a blended/flipped learning approach using an on-line platform, the Exa Foundation. This approach will encourage students to independently revise Component 1 and then revisited in class, this will be able to be monitored quite easily as students will be able to submit a visual record of their notes made in their sketch books, these will be photographed and uploaded to MS Teams. To further encourage independent study other platforms will be used including Diagnostic Questions, The Oak Academy and Seneca Learning. These provide a virtual textbook and allow students to complete automated assessments that pinpoint gaps in knowledge. These systems will enable staff to see what parts of the curriculum have been revised. Midway through this half term the students will sit a mini mock and then sit a mock exam for the whole of Component 1 at the end of the term. After each of these assessments students' will complete tests using additional exam style questions on those areas where a gap in knowledge has been identified.

Yr 10 – Both the Computer Science and Creative iMedia curriculums will be delivered as originally planned without any alterations in light of Covid-19.

KS3 - A spiralling curriculum is used, this ensures all topics from the previous year that have been partially completed, or not at all, will be revisited.

Assessment (KS3/4/5)

At KS3 automated assessment will take place using the online platform, Diagnostic Questions, this will allow all students to be assessed and will facilitate the provision of feedback. The assessment will take place at the end of the first term once the first unit, Data Representation, has been completed. By the end of the second half term the Systems Architecture unit will have been assessed in the way.

Yr 13 - An initial baseline assessment has not been used as this has the potential to affect an individual students' motivation and confidence. Midway through the first half term the students will sit a mini mock and then sit a mock exam for the whole of Component 1 at the end of the term. After each of these assessments students' will complete tests using additional exam style questions on those areas where a gap in knowledge has been pinpointed. Students will also use the Learning Platforms to complete their own revision and automated assessment of this. Assessment of these topics will then be repeated when these topics are revisited in the second half of the spring term.

Yr 11 – An initial baseline assessment has not been used as this has the potential to affect an individual students' motivation and confidence. Students will sit a series of mini mocks throughout the first term, at the end of this term they will sit a mock exam for component 1. After each of these assessments students' will complete tests using additional exam style questions on those areas where a gap in knowledge has been identified. Students will also use the Learning Platforms to complete their own revision and automated assessment of this.

Intended Impact at KS3

To develop a basic understanding of the fundamental principles and concepts of Computer Science that will then be further developed at KS4 in preparation for the Computer Science GCSE and the Creative I-Media courses.

To embed a deep understanding of the basic skills across the five areas of Maths in preparation for applying them to GCSE questions and across the wider whole school curriculum and in real life.

Intended Impact at KS4

To develop a secure understanding of the skills and knowledge of either the Computing GCSE or Creative iMedia curriculum in order to ensure an individual's maximum potential is reached at GCSE facilitating the transition into KS5 or a career in the ICT industry.

Intended Impact at KS5

To develop the Computing and ICT knowledge and skill sets needed for their further education and future in the workplace.

Year	Curriculum Title	HT1 topics	HT2 topics	HT3 topics	HT4 topics	HT5 topics	HT6 topics
7	Yr 7 Computer Science	Introduction to the school network Cyber safety – 2 Lessons followed by a unit on Data Representation.	Components of a Computer System unit.	Cyber safety (1 lesson) followed by a unit on Computational Abstractions and Models using Flowol.	Computation al Thinking (Algorithms and Programming) using textual programming techniques (Python)	Cyber safety (1 lesson) followed by a unit on Networking.	Creative project using the MS Office Suite.
8	Yr 8 Computer Science	Introduction to the school network Cyber safety – 2 Lessons followed by a unit on Data Representation.	Components of a Computer System unit.	Cyber safety (1 lesson) followed by a unit on Computational Abstractions and Models using Flowol using Scratch.	Computation Thinking (Algorithms and Programming) using textual programming techniques (Python)	Cyber safety (1 lesson) followed by a unit on Networking.	Creative project using the Adobe Photoshop, graphics software.

9	Yr 8 Computer Science	Introduction to the school network Cyber safety – 2 Lessons followed by a unit on Data Representation.	Components of a Computer System unit.	Cyber safety (1 lesson) followed by a unit on Computational Abstractions and Models using Kodu.	Computation Thinking (Algorithms and Programming) using textual programming techniques (Python)	Cyber safety (1 lesson) followed by a unit on Networking.	Creative project using the Adobe Flash, animation software.
10	Yr 10 Computer Science	<p>2.2 (part a) Computational Thinking (Algorithms and Programming) using textual programming techniques (Python): variables, constants, operators, data types, Boolean Operators, Constructs for program flow, file handling.</p> <p>1.1 System architecture including Computer Systems, the CPU, CPU and System Performance.</p> <p>1.2 Memory and storage.</p> <p>Practise skills required for Component 2, Section B, students' ability to</p>	<p>2.2 (part b) Computation Thinking (Algorithms and Programming) using textual programming techniques (Python): program flow techniques such as iteration and file handling.</p> <p>1.3 Computer networks, connections and protocols</p> <p>1.4 Network security.</p> <p>1.5 Systems software.</p> <p>1.6 Legal concerns.</p> <p>Practise skills required for Component 2, Section B,</p>	<p>2.1 Algorithms</p> <p>2.3 Producing robust programs.</p> <p>2.4 Boolean logic.</p> <p>2.5 Programming languages and Integrated Development Environments.</p> <p>1.6 Cultural concerns.</p> <p>Practise skills required for Component 2, Section B (ability to write/refine algorithms using a high-level programming language)</p>	<p>2.2 (part a) Computational Thinking (Algorithms and Programming) using textual programming techniques (Python): variables, constants, operators, data types, Boolean Operators, Constructs for program flow, file handling. Including Skills required for Component 2, Section B (ability to write/refine algorithms using a high-level programming language).</p> <p>1.1 Revise System architecture including Computer Systems, the CPU, CPU and System Performance.</p>	<p>2.2 (part b) Computation Thinking (Algorithms and Programming) using textual programming techniques (Python): program flow techniques such as iteration and file handling. Including Skills required for Component 2, Section B (ability to write/refine algorithms using a high-level programming language).</p> <p>1.3 Revise Computer networks, connections and protocols</p>	<p>2.1 Revise Algorithms.</p> <p>2.3 Revise Producing robust programs.</p> <p>2.4 Revise Boolean logic.</p> <p>2.5 Revise Programming languages and Integrated Development Environments.</p> <p>1.6 Revise cultural and environmental concerns.</p> <p>Practise skills required for Component 2, Section B (ability to write/refine algorithms using a high-level programming language)</p>

		<p>write/refine algorithms using a high-level programming language.</p> <p>1.6 Ethical concerns.</p>	<p>students' ability to write/refine algorithms using a high-level programming language.</p>		<p>1.2 Revise Memory and storage.</p> <p>1.6 Environmental concerns.</p>	<p>1.4 Revise Network security.</p> <p>1.5 Revise Systems software.</p> <p>1.6 Revise Ethical and legal concerns.</p>	
10	OCR Nationals Creative iMedia	<p>R081 - Pre-Production Documents</p> <p>LO1 – understanding the purpose of pre-production documents</p> <p>(Mood boards, Storyboards, Visualisation documents, scripts)</p> <p>LO3 – creating examples of the above pre-</p>	<p>R081 -Pre-Production Documents</p> <p>LO2 – Theory on planning pre-production including legislation and hardware/software requirements.</p> <p>LO4 – reviewing pre-production documents.</p> <p>R082- Creating a Digital Graphic</p>	<p>R082- Creating a Digital Graphic</p> <p>LO1 – Identifying the purpose of digital graphics</p> <p>2 hour assessed lessons on LO1.</p> <p>R082- Creating a Digital Graphic</p> <p>LO2 – Planning to create a digital graphic. 3 hour assessed lessons on LO2.</p> <p>R082- Creating a Digital Graphic</p>	<p>R082- Creating a Digital Graphic</p> <p>LO4 – Identifying the purpose of digital graphics</p> <p>2 hour assessed lessons on LO4</p> <p>R086 – Creating Digital Animations – software skills</p> <p>Introduction to Flash and other animation software</p>	<p>R086 – Creating Digital Animations</p> <p>LO1 – identifying uses and purposes of digital animations</p> <p>2 hour assessed lessons on LO1</p> <p>LO2 – Planning to create a digital animation.</p> <p>3 hour assessed lessons on LO2</p>	<p>R087- Creating a Digital Animation</p> <p>LO3 – Creating a digital animation</p> <p>3hour assessed lessons on LO3.</p> <p>LO4 – Identifying the purpose of digital animation</p> <p>2 hour assessed lessons on LO4</p> <p>Submission of Unit 2 – 75% of the course.</p>

		<p>production documents.</p> <p>R082- Creating a Digital Graphic – Photopea/ Photoshop software skills.</p>	<p>LO1 – Identifying the purpose of digital graphics.</p>	<p>LO3 – Creating a digital Graphic 3hour assessed lessons on LO3.</p>		<p>Path 2</p> <p>Pre-Production Documents revision (L01 – LO4).</p>	<p>Path 2</p> <p>Pre-Production exam revision and question practice</p>
11	Yr 11 Computer Science	<p>2.2 (part a) Computational Thinking (Algorithms and Programming) using textual programming techniques (Python): variables, constants, operators, data types, Boolean Operators, Constructs for program flow, file handling.</p> <p>1.1 System architecture including Computer Systems, the CPU, CPU and System Performance.</p> <p>1.2 Memory.</p>	<p>1.4 Wired and wireless networks.</p> <p>1.5 Topologies, protocols and layers.</p> <p>1.6 System security.</p> <p>2.2 (part b) Computation Thinking (Algorithms and Programming) using textual programming techniques (Python): sub routines, lists, arrays, SQL and string handling.</p> <p>2.5 Translators and facilities of</p>	<p>2.2 Revise (part a) Computational Thinking (Algorithms and Programming) using textual programming techniques (Python): variables, constants, operators, data types, Boolean Operators, Constructs for program flow, file handling.</p> <p>1.1 Revise System architecture including Computer Systems, the</p>	<p>Revise 1.4 Wired and wireless networks.</p> <p>Revise 1.5 Topologies, protocols and layers.</p> <p>Revise 1.6 System security.</p> <p>2.2 (part b) Computation Thinking (Algorithms and Programming) using textual programming techniques (Python): sub routines, lists, arrays, SQL and string handling.</p>	<p>Component 1 revision.</p> <p>Component 2 revision.</p>	

		<p>1.3 Secondary storage.</p> <p>1.7 Systems software.</p> <p>1.8 Ethical, legal, cultural and environmental concerns.</p>	<p>Languages.</p> <p>2.1 Algorithms.</p> <p>Exam techniques: Pseudo Code.</p> <p>2.6 Data representation.</p> <p>2.4 Computational logic.</p> <p>Assessment using Pseudo Code.</p> <p>2.3 Producing robust programs.</p>	<p>CPU, CPU and System Performance.</p> <p>1.2 Revise Memory.</p> <p>1.3 Revise Secondary storage.</p> <p>1.7 Revise Systems software.</p> <p>1.8 Revise Ethical, legal, cultural and environmental concerns.</p>	<p>2.5 Translators and facilities of Languages.</p> <p>2.1 Algorithms.</p> <p>Exam techniques: Pseudo Code.</p> <p>2.6 Data representation.</p> <p>2.4 Computational logic.</p> <p>Assessment using Pseudo Code.</p> <p>2.3 Producing robust programs.</p>		
13	A-Level Computer Science	<p>1.1.1 Structure and function of the processor</p> <p>1.1.2 Types of processor</p> <p>1.1.3 Input, output, storage devices, Memory</p> <p>1.2.1 System Software</p> <p>1.4.1 Data Types</p> <p>1.4.2 Data Structures – partial – need to do</p>	<p>1.3.4 Web Technologies</p> <p>1.3.2 Databases</p> <p>1.3.1 Compression, Encryption and Hashing</p> <p>1.5.2 Moral and ethical Issues</p> <p>2.2.1 Programming Techniques continued</p>	<p>3.2 Design of the solution continued (15 marks) (HW) NEA project</p> <p>3.3 Developing the solution (25 marks) (HW) NEA project</p> <p>1.1.1 Structure and function of the processor</p> <p>1.1.2 Types of processor</p>	<p>3.4 Evaluation (20 marks) (HW) NEA project</p> <p>1.3.4 Web Technologies</p> <p>1.3.2 Databases</p> <p>1.3.1 Compression, Encryption and Hashing</p> <p>2.2.1 Programming Techniques</p> <p>2.3.1 Algorithms</p>	<p>Revision COM1</p> <p>Revision COM2</p> <p>External Exams:</p> <p>18/05/21 – Paper 1</p> <p>27/05/21 – Paper 2</p>	

	<p>stacks, queues and hash tables</p> <p>1.4.3 Boolean Algebra – still to do simplifying expressions</p> <p>1.2.2 Applications Generation (a,b,c)</p> <p>September Mini Mock</p> <p>1.2.2 Applications Generation (d,e,f)</p> <p>1.3.3 Networks</p> <p>2.2.1 Programming Techniques</p> <p>Half Term Mock inc Extended write on 1.5.1</p> <p>1.5.2 Moral and ethical Issues</p> <p>October Mini Mock</p>	<p>2.3.1 Algorithms</p> <p>2.1.1 Thinking abstractly</p> <p>2.1.2 Thinking ahead</p> <p>2.1.3 Thinking procedurally</p> <p>2.1.4 Thinking logically</p> <p>2.1.5 Thinking concurrently</p> <p>2.2.2 Computational methods</p> <p>3.1. Analysis of the problem (10 marks) (HW) NEA project</p> <p>3.2 Design of the solution (15 marks) (HW) NEA project</p> <p>1.3.4 Web Technologies</p> <p>1.3.2 Databases</p> <p>1.3.1 Compression, Encryption and Hashing</p>	<p>1.4.1 Data Types</p> <p>1.4.2 Data Structures</p> <p>1.4.3 Boolean Algebra</p> <p>1.2.2 Applications Generation</p> <p>1.3.3 Networks</p> <p>1.5.2 Moral and ethical Issues</p>	<p>Pseudo code exam question practise</p> <p>Extended writing question practise</p> <p>1.5.2 Moral and ethical Issues</p>		
--	--	---	--	---	--	--

			1.5.2 Moral and ethical Issues December Mini Mock				
--	--	--	--	--	--	--	--