CURRICULUM AND ASSESSMENT PLAN

YEAR 12 | COMPUTER SCIENCE

'The curriculum and assessment of students at this stage of education has been carefully designed to promote deep learning of computer science and develop students into knowledgeable practitioners:

At KS5, students are invited to study computer science. The A'Level computing course allows students to build on the knowledge obtained from KS3 and KS4 computing. Throughout the course, students will deepen their understanding of programming using a high-level language known as Python. They will build on their understanding of the text-based language taught at key stage 3 and 4. Students will enhance their understanding with knowledge and skills in object orientated programming, recursion and 3d arrays. Much of the theory extends the knowledge gained in key stage 4 computer science with additional theory including HTML, CSS, JavaScript and databases amongst others. The computer science course is closely linked to mathematics and students will develop skills in simplifying Boolean expressions, representing decimal and negative values in binary, and widening their understanding of algorithms. The course title is OCR AS and A Level Computer Science. The course codes are: - (AS)H046, (A2) H446.

HALF TERM 1 Data Representation / Components of a Computer

All students will know:

Component 1:
1.4.1 - Data Types
- a - Primitive Data Types
- b - Represent positive integers
- c - Sign and Magnitude & Two's Complement
- d - Addition and Subtraction
- e - Normlaisation of floating-point binary
- f - Hexadecimal
- g - Floating point arithmetic
- i - Bitwise manipulation
- j - Character Sets

1.1.1 - Structure and function of the processor
- a - CPU Components
- b - FDE Cycle
- c - CPU Performance
- e - Von Neumann, Harvard and Contemporary Processor Architecture

1.1.2 - Types of processors
- a - CISC and RISC
- c - Multicore and Parallel Systems

1.1.3 - Input, output and storage
- a- Applying I/O/S to problems
- b - Magnetic, flash & optical storage
- c - RAM and ROM
- d -Virtual Storage

Component 2:
2.2.1 - Programming Techniques (Iteration, File Handling, 1D/2D Arrays, Sub-routines, Recursion, OOP)

All students will be assessed:

Students will complete two key assessments during this half term which will assess their understanding of each of the topics they have covered this half term. Students will also be assessed on their programming skills.

Key Assessment Point 1 – Data Representation

Key Assessment Point 1 – Computer Systems

Reading skills needed for this unit:

Students will need to be able to read a range of real-life scenarios and use decomposition.

Key vocabulary:

CPU, ALU, Processor, Storage, Virtual Storage, Architecture, Data Representation, Fixed-Point, Floating-Point, Normalisation

HALF TERM 2 Databases / Boolean Logic / Computational Thinking

All students will know:

Component 1:
1.3.1 - Compression
- a - Lossy vs Lossless Compression

1.3.2 - Databases
- a - Relational databases, flat file, primary key, foreign key, secondary key, entity relationship, modelling, normalisation and indexing
- b - Methods of capturing, selecting, managing and exchanging data.
- c - Normalisation to 3NF
- d - SQL
- e - Referential Integrity

Component 1:
1.4.3 - Boolean Algebra
- a - Define problems using Boolean logic
- b - Manipulate Boolean expressions, including the use of Karnaugh maps
- d - Using logic gate diagrams and truth tables

Component 2:
2.1 - Computational Thinking
- Thinking abstractly
- Thinking ahead
- Thinking procedurally
- Thinking logically
- Thinking concurrently

All students will be assessed:

Students will complete two key assessments this half term. Summative assessments will be used for each of the topics studied and will cover previous topics studied. A range of exam-style questions will be used to give students the experience needed for these topics.

Key Assessment Point 2 – Data Representation / Boolean Algebra

Key Assessment Point 2 - Databases

Exam-style questions will be used throughout lessons to prepare students to answer questions.

Students will also be assessed during lessons through the following methods:

- Multiple choice questions
- Exam questions – past exam questions
- Live marking
- Whole class feedback
- Verbal feedback
- Programming challenges
- Self-assessment

Reading skills needed for this unit:

Students will need to be able to read a range of real-life scenarios and use decomposition. Students need to read expressions in order for them to simplify.

Key vocabulary:

Compression, Abstraction, Decomposition, Computational Thinking, Boolean Expressions, Referential Integrity, Database, SQL.

HALF TERM 3 Programming

All students will know:

Component 2:

2.2.1 - Programming

- Object-Orientated Programming
- Recursion

Component 1/2:

1.4.2 - Data Structures

- a - Arrays, records, lists, tuples
- b - Linked-list, graph (directed and un-directed), stack, queue, tree, binary search tree, hash table
- c- How to create, traverse, add data to and remove data from the data structures above

2.2 Problem Solving and Programming

- 2.2.1 - Programming Techniques
- 2.2.2 Computational Methods

All students will be assessed:

Students will complete summative assessments for the topics they have studied this half term. These will include an assessment on programming where students will be required to use all of the skills they have developed. They will also be required to show their problem-solving skills.

Students will also be assessed during lessons through the following methods:

- Multiple choice questions
- Exam questions – past exam questions
- Live marking
- Whole class feedback
- Verbal feedback
- Programming challenges
- Self-assessment

Reading skills needed for this unit:

Students will need to read a high-level language and identify what each of the programs they read are doing. Students will need to apply inference when reading code to make predictions.

Key vocabulary:

Lists, Arrays, Object-Orientated Programming, Data Structures, Data Structures

HALF TERM 4 Operating Systems / Software / Programming / Web Development

All students will know:

Component 1:

1.2.1 - Systems Software

- a - Operating Systems
- b - Memory Management
- c - Interrupts
- d - Scheduling
- e - Distributed, embedded, multi-tasking, multi-user and Real Time operating systems
- f - BIOS
- g - Device drivers
- h - Virtual machines

1.2.2 - Applications Generation

- a - The nature of applications, justifying suitable applications for a specific purpose
- b - Utilities
- c - Open source vs Closed source
- d - Translators (Interpreters, compilers and assemblers)

1.2.4 - Types of programming language

- a - Characteristics of programming paradigms
- b - Procedural languages
- c - Assembly language (LMC)
- d - Modes of addressing memory
- e - OOP

Component 1:

1. 3. 4 - Web Technologies

- a - HTML, CSS and JavaScript
- b - Search Engine Indexing

All students will be assessed

Students will complete two key assessments during this half term which will consist of everything that has been covered in previous units and the current topics being studied.
Key Assessment 3 – System Software
Key Assessment 3 – Web Technologies

Students will also be assessed during lessons through the following methods:

- Multiple choice questions
- Exam questions – past exam questions
- Live marking
- Whole class feedback
- Verbal feedback
- Programming challenges
- **Self-assessment**

Reading skills needed for this unit:

Students need to read complex text and scenarios with fluency. Students need to have a wide understanding of technical vocabulary.

Key vocabulary:

HTML, Translators, Interpreter, Compiler, Utilities, Operating Systems

CURRICULUM AND ASSESSMENT PLAN

YEAR 12 | A LEVEL COMPUTER SCIENCE

HALF TERM 5

All students will know:

Component 1:
1.5.1 - Computing Related Legislation
- a - The Data Protection Act 1998
- b - The Computer Misuses Act 1990
- c - The Copyright Design and Patents Act 1998
- d - The Regulation of Investigatory Powers Act 2000
1.5.2 - Moral and Ethical Issues
1.3.3 - Networks
- a - Characteristics of networks and the importance of protocols and standards
- b - The Internet structure
- c - Network security and threats
- d - Network hardware
- e - Client-server and peer to peer
Component 2:
2.3.1 - Algorithms
- b - Suitability of different algorithms
- c -Measures and methods to determine the efficiency of different algorithms, Big O Notation.
- f - Standard sorting and searching algorithms.
Component 2/3
- Mock Project

All students will be assessed:

Students will complete summative assessments for the topics they have studied this half term. These will include assessments on legislation, algorithms and networking.

Students will also be assessed during lessons through the following methods:

- Multiple choice questions
- Exam questions – past exam questions
- Live marking
- Whole class feedback
- Verbal feedback
- Programming challenges
- Self-assessment

Reading skills needed for this unit:

Students need to read complex text and scenarios with fluency. Students need to have a wide understanding of technical vocabulary.

Key vocabulary:

Networks, Hardware, Legislation, Ethical, Client-Server, Peer-to-Peer, Algorithms

HALF TERM 6

All students will know:

Component 1:
1.2.3 - Software Development
- a - Understand the waterfall lifecycle, agile methodologies, extreme programming, the spiral model and rapid application development.
- b - The relative merits and drawbacks of different methodologies and when they might be used.
- c - Writing and following algorithms.
Component 1/2:
- AS Mock Papers
Component 3 (NEA):
- Feasibility, analysis, design.

All students will be assessed:

Students will complete two key assessments this half term which will consist of everything that has been taught throughout the year. This will consist of two mock exams based on the AS content.
Key Assessment Point 4 – H046/01 Computer Systems
Key Assessment Point 4 – H046/02 Algorithms and Problem Solving
Students will also start their NEA which will last the duration of the course

Students will also be assessed during lessons through the following methods:

- Multiple choice questions
- Exam questions – past exam questions
- Live marking
- Whole class feedback
- Verbal feedback
- Programming challenges
- Self-assessment

Reading skills needed for this unit:

Students need to read complex text and scenarios with fluency. Students need to have a wide understanding of technical vocabulary.

Students will need to read a high-level language and identify what each of the programs they read are doing. Students will need to apply inference when reading code to make predictions.

Key vocabulary:

Software, Waterfall, Methodologies, Agile, Algorithms, Analysis, Design, Testing.

HOW STUDENTS CAN BESUPPORTED AT HOME

Students can access a range of websites including:

Python Tutorial (w3schools.com)

HTML Tutorial (w3schools.com)

PyGame: A Primer on Game Programming in Python – Real Python

Students can also purchase OCR endorsed text books and revision guides which will allow them to develop their knowledge of the content that has been devlivered within lessons. They also have the opportunity to purchase books which consist of a range of exam-style questions.

HOW THIS LEARNING WILL BE EMBEDDED ELSEWHEREIN THE CURRICULUM

This learning can be implemented in mathematics for those that are studying this course. Students will also be able to use their knowledge of HTML and CSS to develop websites which could be used elsewhere such as in enrichment courses.