Biddulph High School Curriculum Intent

To deliver a broad and enriching curriculum through engaging and challenging lessons that provide a wide range of opportunities for all students to achieve their potential.
Students will all be prepared to take their next steps in a diverse and ever changing future ready to make a positive contribution to society.
Through a broad programme of extracurricular activities students will have the opportunities to showcase their talents and experience new challenges.
We value individuals and all that they can offer as well as supporting each other with kindness and empathy.

Curriculum Intent for Computing

"Those who can imagine anything, can create the impossible." Alan Turing

Our aim in the Computing department is centred around equipping students for their future, regardless of the individual pathway they may decide to choose, ensuring that students are prepared for the challenge of a rapidly developing and changing technological world. We will equip learners with the key technical skills to support their learning across the curricula, for future studies and ultimately for their chosen career pathway. We believe in delivering a mixture of both ICT and Computer Science in our curriculum to develop core employability skills, such as problem solving and critical thinking. We also develop "Internet Citizens" who understand the importance of being responsible in the digital world. Our curriculum is mapped from KS3 to KS5 ensuring that students have the opportunity to grow both their knowledge and technical skills. We will provide a variety of extra curricula activities including entering national competitions, providing opportunities for students to acquire further technical qualifications and conferences/ visits to inspire students to follow a future in technology.

All teachers will follow the schemes of work provided by the department. This will ensure that all students receive the same high-quality provision. All units of work will provide a clear outline of the knowledge and skills required and assessments will ensure that this knowledge has been retained and that skills can be evidenced.

Teachers will ensure that gaps are closed through regular monitoring within the classroom. DINT activities will allow for interleaving and recap of previous learning. Misconceptions will be identified through effective questioning and the regular inspection of student work.

Computing Long Term Overview						
Year Group	**Autumn 1**	**Autumn 2**	**Spring 1**	**Spring 2**	**Summer 1**	**Summer 2**
OCR GCSE Computer Science						
10	Computational thinking, algorithms, programming fundamentals, data types and ethical, legal and environmental considerations **Booklet 1**	Computational thinking, algorithms, programming fundamentals, data types and ethical, legal and environmental considerations **Booklet 1**	An understanding of different number systems used within Computer Science. They will also consider types of defence and testing to maintain computer security. **Booklet 2**	An understanding of searching and sorting algorithms. They will also be introduced to Boolean logic and how to use logic diagrams. **Booklet 3**	An understanding how data is stored within computer systems, and how compression be used to minimise the file sizes. **Booklet 4**	Practical programming - Learners will develop their programming skills focusing on sequence, selection and iterations to solve a number of problems in given scenarios.
11	An understanding of what the CPU is responsible for, the difference between RAM and ROM and other storage types. Different types of networks are also explored in this topic and basic fundamentals of programming. **Booklet 5**	Learners will explore different type of network topologies and potential network threats. **Booklet 6**	Learners will discover different operating systems, utility software and explore legal, environmental, cultural and ethical impacts of computing. **Booklet 7**	Learners will develop further their programming skills to a more advanced level through the creation of working computer programs. **Booklet 8**	Learners will be revising theory and practical skills during this half term in preparation for their examination.	

BIDDULPH HIGH SCHOOL CURRICULUM DOCUMENTATION

OCR A Level Computer science						
12	1.1 The characteristics of contemporary processors, input, output and storage devices 1.4 Data types, data structures and algorithms	1.2.1 Software and Software Development 2.1 Elements of computational thinking	1.2 Application Generation 2.2 Problem solving and programming	1.3.2 Databases 2.2 Problem solving and programming	NEA Programming Project	NEA Programming Project
13	1.3 Exchanging Data NEA Programming Project	1.3.3 Networks NEA Programming Project	1.3.4 Web Technologies	1.5 Legal, moral, cultural and ethical issues	Revision	

Computing: Computer Science Medium Term Overview

Autumn Term 1 ***Miss Deavall***	**Unit Title:** 1.1 The characteristics of contemporary processors, input, output and storage devices	**No of Lessons: 15 lessons**

Overview This unit will develop learners understanding of the components of the computer and their uses

Assessment There will be an assessment of examination style questions before October half term**.**

<u>Essential Knowledge (what must students know):</u>

1.1.1

(a)The Arithmetic and Logic Unit; ALU, Control Unit and Registers (Program Counter; PC, Accumulator; ACC, Memory Address Register; MAR, Memory Data Register; MDR, Current Instruction Register; CIR). Buses: data, address and control: how this relates to assembly language programs.

(b) The Fetch-Decode-Execute Cycle; including its effects on registers.

(c) The factors affecting the performance of the CPU: clock speed, number of cores, cache.

(d) The use of pipelining in a processor to improve efficiency.

(e) Von Neumann, Harvard and contemporary processor architecture.

1.1.2

(a) The differences between and uses of CISC and RISC processors.

(b) GPUs and their uses (including those not related to graphics)

c) Multicore and Parallel systems.

1.1.3

a) How different input, output and storage devices can be applied to the solution of different problems.

<u>Essential Skills (what must students be able to demonstrate):</u>

Students will be able to:

- an understanding of and ability to apply the fundamental principles and concepts of computer science including; abstraction, decomposition, logic, algorithms and data representation
- the ability to analyse problems in computational terms through practical experience of solving such problems including writing programs to do
- the capacity for thinking creatively, innovatively, analytically, logically and critically
- the capacity to see relationships between different aspects of computer science
- the ability to articulate the individual (moral), social (ethical), legal and cultural opportunities and risks of digital technology

Lesson topics:

- The Arithmetic and Logic Unit
- Registers
- The FDE Cycle
- Factors affecting performance of CPU
- Pipelining
- Von Neumann
- Harvard and Contemporary processor architecture
- CISC
- RISC
- Graphical
- Multicore and Parallel Systems

(b) The uses of magnetic, flash and optical storage devices. (c) RAM and ROM (d) Virtual storage. **Terminology:** See key terms glossary A level computer science glossary of terms.pdf		
Careers Links:	**Enrichment:** NA	

Computing: Computer Science Medium Term Overview		
Autumn Term 1 ***Mrs Sharrock***	**Unit Title:** 1.4 Data types, data structures and algorithms	**No of Lessons: 15**
Overview: This unit will develop learners understanding of how data is represented and stored within different structures. Different algorithms that can be applied to these structures **Assessment**: There will be an assessment of examination style questions before October half term		
Essential Knowledge (what must students know): 1.4.1 Data Types (a Primitive data types, integer, real/floating point, character, string and Boolean. (b) Represent positive integers in binary. (c) Use of sign and magnitude and two's complement to represent negative numbers in binary. (d) Addition and subtraction of binary integers. (e) Represent positive integers in hexadecimal. (f) Convert positive integers between binary hexadecimal and denary. (g) Representation and normalisation of floating point numbers in binary. (h) Floating point arithmetic, positive and negative numbers, addition and subtraction.	**Essential Skills (what must students be able to demonstrate):** **Students will be able to:** • an understanding of and ability to apply the fundamental principles and concepts of computer science including; abstraction, decomposition, logic, algorithms and data representation • Mathematical skills • the ability to analyse problems in computational terms through practical experience of solving such problems including writing programs to do • the capacity for thinking creatively, innovatively, analytically, logically and critically	**Lesson topics:** • Binary addition and subtraction • Negative numbers in binary • Subtracting Twos Complement • Fractional numbers in binary • Normalisation of floating point • Floating point arithmetic • Logic gates • Logic notation and simplification • Karnaugh maps • Bitwise manipulation • Flip flops half and full adders

(i) Bitwise manipulation and masks: shifts, combining with AND, OR, and XOR. # (j) How character sets (ASCII and UNICODE) are used to represent text. 1.4.3 Boolean Algebra (a)Define problems using Boolean logic. See appendix 5d. (b) Manipulate Boolean expressions, including the use of Karnaugh maps to simplify Boolean expressions. (c) Use the following rules to derive or simplify statements in Boolean algebra: De Morgan's Laws, distribution, association, commutation, double negation. (d) Using logic gate diagrams and truth tables. See appendix 5d. (e) The logic associated with D type flip flops, half and full adders. **Terminology:** A level computer science glossary of terms.pdf	• the capacity to see relationships between different aspects of computer science • the ability to articulate the individual (moral), social (ethical), legal and cultural opportunities and risks of digital technology	
Careers Links:	**Enrichment:** NA	

Computing: Computer Science Medium Term Overview

Autumn Term 2 ***Miss Deavall***	**Unit Title: 1.2.1 Software and Software Development**	**No of Lessons: 14 lessons**

Overview: This unit will develop learners understanding of different types of software and the software methodologies to develop software.

Assessment: Homework tasks and assessment next half term

Essential Knowledge (what must students know):

1.2.1

(a) The need for, function and purpose of operating systems

(b) Memory Management (paging, segmentation and virtual memory)

(c) Interrupts, the role of interrupts and Interrupt Service Routines (ISR), role within the Fetch- Decode-Execute Cycle

(d)Scheduling: round robin, first come first served, multi-level feedback queues, shortest job first and shortest remaining time

(e) Distributed, embedded, multi-tasking, multi-user and Real Time operating systems.

(f) BIOS

(g) Device drivers

(h)Virtual machines, any instance where software is used to take on the function of a machine, including executing intermediate code or running an operating system within another.

1.2.3

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.

Terminology:

Essential Skills (what must students be able to demonstrate):

Students will be able to:

- an understanding of and ability to apply the fundamental principles and concepts of computer science including; abstraction, decomposition, logic, algorithms and data representation
- the ability to analyse problems in computational terms through practical experience of solving such problems including writing programs to do
- the capacity for thinking creatively, innovatively, analytically, logically and critically
- the capacity to see relationships between different aspects of computer science
- the ability to articulate the individual (moral), social (ethical), legal and cultural opportunities and risks of digital technology

Lesson topics:

- Operating Systems
- Memory Management
- Virtual Memory
- Interrupts
- Scheduling
- Types of Operating System
- BIOS
- Device Drivers
- Virtual Machines
- Waterfall Lifecycle
- Agile Methodology
- Extreme Programming and RA
- Spiral Model
- RAD

See key terms glossary A level computer science glossary of terms.pdf		
Careers Links:	**Enrichment:** CSUk Christmas Coding Competition: a challenge a day	

Computing: Computer Science Medium Term Overview

Autumn Term 2 ***Mrs Sharrock***	**Unit Title:** 2.1 Elements of computational thinking	**No of Lessons: 15**	

Overview: This unit will develop an understanding of key computational thinking skills

Assessment: Homework tasks and assessment task

Essential Knowledge (what must students know):

2.1.1 Thinking abstractly
(a) The nature of abstraction.
(b) The need for abstraction.
(c) The differences between an abstraction and reality.
(d) Devise an abstract model for a variety of situations.
2.1.2 Thinking ahead
(a) Identify the inputs and outputs for a given situation.
(b) Determine the preconditions for devising a solution to a problem.
(c) The nature, benefits and drawbacks of caching. (d) The need for reusable program components. 2.1.3 Thinking procedurally
(a) Identify the components of a problem.
(b) Identify the components of a solution to a problem.
(c) Determine the order of the steps needed to solve a problem.
(d) Identify sub-procedures necessary to solve a problem.
2.1.4 Thinking logically
(a) Identify the points in a solution where a decision has to be taken.

Essential Skills (what must students be able to demonstrate):

Students will be able to:

- an understanding of and ability to apply the fundamental principles and concepts of computer science including; abstraction, decomposition, logic, algorithms and data representation
- Mathematical skills
- the ability to analyse problems in computational terms through practical experience of solving such problems including writing programs to do
- the capacity for thinking creatively, innovatively, analytically, logically and critically
- the capacity to see relationships between different aspects of computer science
- the ability to articulate the individual (moral), social (ethical), legal and cultural opportunities and risks of digital technology

Lesson topics:

- Abstraction
- Thinking Ahead
- Thinking procedurally
- Thinking logically and concurrently
- Problem recognition

(b) Determine the logical conditions that affect the outcome of a decision. (c) Determine how decisions affect flow through a program. 2.1.5 Thinking concurrently (a) Determine the parts of a problem that can be tackled at the same time. (b) Outline the benefits and trade offs that might result from concurrent processing in a particular situation. **Terminology:** See key terms glossary A level computer science glossary of terms.pdf		
Careers Links:	**Enrichment:** Bebras Computational Thinking Competition	

Computing: Computer Science Medium Term Overview

Spring Term 1 ***Miss Deavall***	**Unit Title: 1.2 Application Generation**	**No of Lessons: 10 Lessons**

Overview: This unit will develop learners understanding of different types of software and the software methodologies to develop software.

Assessment: Assessment window in this half term

Essential Knowledge (what must students know):	**Essential Skills (what must students be able to demonstrate):**	**Lesson topics:**
1.2.2 (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. (e) Stages of compilation (lexical analysis, syntax analysis, code generation and optimisation). (f) Linkers and loaders and use of libraries **Terminology:** See key terms glossary A level computer science glossary of terms.pdf	**Students will be able to:** • an understanding of and ability to apply the fundamental principles and concepts of computer science including; abstraction, decomposition, logic, algorithms and data representation • the ability to analyse problems in computational terms through practical experience of solving such problems including writing programs to do • the capacity for thinking creatively, innovatively, analytically, logically and critically • the capacity to see relationships between different aspects of computer science • the ability to articulate the individual (moral), social (ethical), legal and cultural opportunities and risks of digital technology	• Types of application • Utility Software • Open and closed software • Translators • Stages of Compilation • Linkers, Loaders and Libraries
Careers Links:	**Enrichment:** NA	

Computing: Computer Science Medium Term Overview

Spring Term 1 ***Mrs Sharrock***	**Unit Title:** 2.2 Problem solving and programming	**No of Lessons: 15**

Essential Knowledge (what must students know):

2.2.1 Programming techniques (a)
Programming constructs: sequence, iteration, branching.
(b) Recursion, how it can be used and compares to an iterative approach.
(c) Global and local variables.
(d) Modularity, functions and procedures, parameter passing by value and by reference.
(e) Use of an IDE to develop/debug a program.
(f) Use of object oriented techniques.
2.2.2 Computational methods
(a) Features that make a problem solvable by computational methods.
(b) Problem recognition.
(c) Problem decomposition.
(d) Use of divide and conquer.
(e) Use of abstraction.
(f) Learners should apply their knowledge of:
• backtracking
• data mining
• heuristics
• performance modelling
• pipelining
• visualisation to solve problems.

Terminology:
See key terms glossary
A level computer science glossary of terms.pdf

Essential Skills (what must students be able to demonstrate):

Students will be able to:

- an understanding of and ability to apply the fundamental principles and concepts of computer science including; abstraction, decomposition, logic, algorithms and data representation
- Mathematical skills
- the ability to analyse problems in computational terms through practical experience of solving such problems including writing programs to do
- the capacity for thinking creatively, innovatively, analytically, logically and critically
- the capacity to see relationships between different aspects of computer science
- the ability to articulate the individual (moral), social (ethical), legal and cultural opportunities and risks of digital technology

Lesson topics:

- Programming techniques
- Sorting algorithms
- Searching algorithms

Careers Links:	**Enrichment:** Turing Challenge	

Computing: Computer Science Medium Term Overview

Spring Term 2 ***Miss Deavall***	**Unit Title: 1.3.2 Databases**	**No of Lessons: 14 Lessons**	

Overview: This unit will develop learners understanding of database software including key skills for creation. Learners will also look at how data is captured and ways to prevent errors in transactions

Assessment: Homework activities

Essential Knowledge (what must students know): **1.3.2** (a) Relational database, flat file, primary key, foreign key, secondary key, entity relationship modelling, normalisation and indexing. See appendix 5f. (b) Methods of capturing, selecting, managing and exchanging data. (c) Normalisation to 3NF. (d) SQL – Interpret and modify. See appendix 5d. (e)Referential integrity. (f) Transaction processing, ACID (Atomicity, Consistency, Isolation, Durability), record locking and redundancy **Terminology:** See key terms glossary A level computer science glossary of terms.pdf	**Essential Skills (what must students be able to demonstrate):** **Students will be able to:** • Create and set up databases for a given scenario, including setting data types, tables and relationships • The database must meet the needs of the user • Use SQL to modify and add entries to a database • Mathematical skills	**Lesson topics:** • Intro to database basics • Database key terminology • Relational databases • Setting up databases to meet a user need • Methods of data capture • Normalisation • SQL • Referential Integrity • Transaction Processing
Careers Links: Databases in the workplace	**Enrichment:** NA	

Computing: Computer Science Medium Term Overview		
Spring Term 2 ***Mrs Sharrock***	**Unit Title:** 2.2 Problem solving and programming	**No of Lessons: 15**
Overview: This unit develops learners understanding of key programming techniques Assessment: Homework tasks		
Essential Knowledge (what must students know): 2.2.1 Programming techniques **Terminology:** See key terms glossary A level computer science glossary of terms.pdf	**Essential Skills (what must students be able to demonstrate):** **Students will be able to:** • the ability to analyse problems in computational terms through practical experience of solving such problems including writing programs to do • the capacity for thinking creatively, innovatively, analytically, logically and critically	**Lesson topics:** • Databases SQL • Graphical User Interfaces
Careers Links:	**Enrichment:** NA	

Computing: Computer Science Medium Term Overview

Summer Term 1 Mrs Sharrock & Miss Deavall	**Unit Title: 3. NEA Programming Project**	**No of Lessons: 9**	

Overview: This unit is to introduce the 3rd Unit: The Programming Project

Assessment: This is also the window for Mock examinations for Unit 1 and Unit 2 content that has been covered

Essential Knowledge (what must students know):	**Essential Skills (what must students be able to demonstrate):**	**Lesson topics:**
3.1. Analysis of the problem (10 marks) 3.1.1 Problem identification (a) Describe and justify the features that make the problem solvable by computational methods. (b) Explain why the problem is amenable to a computational approach. 3.1.2 Stakeholders (a) Identify and describe those who will have an interest in the solution explaining how the solution is appropriate to their needs (this may be named individuals, groups or persona that describes the target end user). 3.1.3 Research the problem (a) Research the problem and solutions to similar problems to identify and justify suitable approaches to a solution. (b) Describe the essential features of a computational solution explaining these choices. (c) Explain the limitations of the proposed solution. 3.1.4 Specify the proposed solution (a)Specify and justify the solution requirements including hardware and software configuration (if appropriate).	**Students will be able to:** • Learners will be expected to analyse, design, develop, test, evaluate and document a program written in a suitable programming language. The underlying approach to the project is to apply the principles of computational thinking to a practical coding problem. Learners are expected to apply appropriate principles from an agile development approach to the project development.	• Programming project launch

(b) Identify and justify measurable success criteria for the proposed solution. **Terminology:** See key terms glossary A level computer science glossary of terms.pdf .		
Careers Links:	**Enrichment:** NA	

Computing: Computer Science Medium Term Overview

Summer Term 2 ***Miss Deavall***	**Unit Title: Consolidation**	**No of Lessons: 8 lessons**

Overview: This unit is aimed to consolidate the key components covered this year. Including the key skills to be able to answer the longer mark questions

Assessment: Mock examination this half term

Essential Knowledge (what must students know):	**Essential Skills (what must students be able to demonstrate):**	**Lesson topics:**
Summary of the content covered this year **Terminology:** See key terms glossary A level computer science glossary of terms.pdf	**Students will be able to:** • an understanding of and ability to apply the fundamental principles and concepts of computer science including; abstraction, decomposition, logic, algorithms and data representation • Mathematical skills • the ability to analyse problems in computational terms through practical experience of solving such problems including writing programs to do • the capacity for thinking creatively, innovatively, analytically, logically and critically • the capacity to see relationships between different aspects of computer science • the ability to articulate the individual (moral), social (ethical), legal and cultural opportunities and risks of digital technology	• Review 1.1.1 • Review 1.1.2 • Review 1.1.3 • Review 1.2.1 • Review 1.2.3 • Review 1.3.2
Careers Links:	**Enrichment:** NA	

Computing: Computer Science Medium Term Overview

Summer Term 2 ***Mrs Sharrock***	**Unit Title: 3. NEA Programming Project**	**No of Lessons: 15**	

<u>Essential Knowledge (what must students know):</u> 3.2 Design of the solution (15 marks) 3.2.1 Decompose the problem (a) Break down the problem into smaller parts suitable for computational solutions justifying any decisions made. 3.2.2 Describe the solution (a) Explain and justify the structure of the solution. (b) Describe the parts of the solution using algorithms justifying how these algorithms form a complete solution to the problem. (c) Describe usability features to be included in the solution. (d) Identify key variables / data structures / classes justifying choices and any necessary validation. 3.2.3 Describe the approach to testing (a) Identify the test data to be used during the iterative development and post development phases and justify the choice of this test data **Terminology:** See key terms glossary A level computer science glossary of terms.pdf	**<u>Essential Skills (what must students be able to demonstrate):</u>** **Students will be able to:** • Learners will be expected to analyse, design, develop, test, evaluate and document a program written in a suitable programming language. The underlying approach to the project is to apply the principles of computational thinking to a practical coding problem. Learners are expected to apply appropriate principles from an agile development approach to the project development.	**Lesson topics:** • Programming Project Support
<u>Careers Links:</u>	**<u>Enrichment:</u>** NA	

Year 13

Computing: Computer Science Medium Term Overview			
Autumn Term 1 ***Miss Deavall***	**Unit Title: 1.3 Exchanging Data**	**No of Lessons: 14 lessons**	
Overview: This unit is where leaners will discover how data is exchanged between different systems Assessment: Assessment week of examination style questions before OCtober Half Term			
Essential Knowledge (what must students know): **1.3.1** (a) Lossy vs Lossless compression. (b) Run length encoding and dictionary coding for lossless compression. (c) Symmetric and asymmetric encryption. (d) Different uses of hashing. **Terminology:** See key terms glossary A level computer science glossary of terms.pdf	**Essential Skills (what must students be able to demonstrate):** **Students will be able to:** • an understanding of and ability to apply the fundamental principles and concepts of computer science including; abstraction, decomposition, logic, algorithms and data representation • Mathematical skills • the ability to analyse problems in computational terms through practical experience of solving such problems including writing programs to do • the capacity for thinking creatively, innovatively, analytically, logically and critically • the capacity to see relationships between different aspects of computer science • the ability to articulate the individual (moral), social (ethical), legal and cultural opportunities and risks of digital technology	**Lesson topics:** • Lossy Vs Lossless compression • Run length encoding • Dictonary coding • Symmetric encryption • Asymmetric encryption • Hashing •
Careers Links:	**Enrichment:** NA	

Computing: Computer Science Medium Term Overview

Autumn Term 1 ***Mrs Sharrock***	**Unit Title: NEA Programming Project**	**No of Lessons: 15**

<u>Essential Knowledge (what must students know):</u> 3.3 Developing the solution (25 marks) 3.3.1 Iterative development process (a) Provide annotated evidence of each stage of the iterative development process justifying any decision made. (b) Provide annotated evidence of prototype solutions justifying any decision made. 3.3.2 Testing to inform development (a)Provide annotated evidence for testing at each stage justifying the reason for the test. (b)Provide annotated evidence of any remedial actions taken justifying the decision made. **Terminology:** See key terms glossary A level computer science glossary of terms.pdf .	**<u>Essential Skills (what must students be able to demonstrate):</u>** **Students will be able to:** • Learners will be expected to analyse, design, develop, test, evaluate and document a program written in a suitable programming language. The underlying approach to the project is to apply the principles of computational thinking to a practical coding problem. Learners are expected to apply appropriate principles from an agile development approach to the project development.	**Lesson topics:** • Programming Project
<u>Careers Links:</u>	**<u>Enrichment:</u>** NA	

Computing: Computer Science Medium Term Overview

Autumn Term 2	**Unit Title: 1.3.3 Networks**	**No of Lessons: 12 Lessons**	

Overview: This unit will cover how computers communicate.

Assessment: Fortnightly homework and January mock

Essential Knowledge (what must students know):	**Essential Skills (what must students be able to demonstrate):**	**Lesson topics:**
1.3.3 (a) Characteristics of networks and the importance of protocols and standards. (b) The internet structure: • The TCP/IP Stack. • DNS • Protocol layering. • LANs and WANs. • Packet and circuit switching. (c) Network security and threats, use of firewalls, proxies and encryption. (d) Network hardware. (e) Client-server and peer to peer **Terminology:** See key terms glossary A level computer science glossary of terms.pdf	**Students will be able to:** • an understanding of and ability to apply the fundamental principles and concepts of computer science including; abstraction, decomposition, logic, algorithms and data representation • Mathematical skills • the ability to analyse problems in computational terms through practical experience of solving such problems including writing programs to do • the capacity for thinking creatively, innovatively, analytically, logically and critically • the capacity to see relationships between different aspects of computer science • the ability to articulate the individual (moral), social (ethical), legal and cultural opportunities and risks of digital technology	• Network introduction • Protocols & Standards • The structure of hte internet • Network security • Network hardware • Client server and peer to peer network
Careers Links:	**Enrichment:** NA	

Computing: Computer Science Medium Term Overview

Autumn Term 2 ***Mrs Sharrock***	**Unit Title: NEA Programming Project**	**No of Lessons: 15**

Overview: This unit is Component 3: the computing project

Assessment: The project is assessed as per the NEA assessment in the specification

<u>Essential Knowledge (what must students know):</u>

3.4 Evaluation (20 marks)
3.4.1 Testing to inform evaluation
(a)Provide annotated evidence of testing the solution of robustness at the end of the development process.
(b)Provide annotated evidence of usability testing (user feedback).
3.4.2 Success of the solution
(a) Use the test evidence from the development and post development process to evaluate the solution against the success criteria from the analysis.
3.4.3 Describe the final product
(a) Provide annotated evidence of the usability features from the design, commenting on their effectiveness.
3.4.4 Maintenance and development
(a) Discuss the maintainability of the solution.
(b) Discuss potential further development of the solution.

Post Project:
1.2.4 Types of Programming Language
(a)Need for and characteristics of a variety of programming paradigms.

<u>Essential Skills (what must students be able to demonstrate):</u>

Students will be able to:

- an understanding of and ability to apply the fundamental principles and concepts of computer science including; abstraction, decomposition, logic, algorithms and data representation
- Mathematical skills
- the ability to analyse problems in computational terms through practical experience of solving such problems including writing programs to do
- the capacity for thinking creatively, innovatively, analytically, logically and critically
- the capacity to see relationships between different aspects of computer science
- the ability to articulate the individual (moral), social (ethical), legal and cultural opportunities and risks of digital technology

Lesson topics:

- Programming Project support

Post Project:

- Programming Paradigms
- LMC
- OOP

(b) Procedural languages. (c) Assembly language (including following and writing simple programs with the Little Man Computer instruction set). See appendix 5d. (d) Modes of addressing memory (immediate, direct, indirect and indexed). (e) Object-oriented languages (see appendix 5d for pseudocode style) with an understanding of classes, objects, methods, attributes **Terminology:** See key terms glossary A level computer science glossary of terms.pdf		
Careers Links:	**Enrichment:** NA	

Computing: Computer Science Medium Term Overview

Spring Term 1 ***Miss Deavall***	**Unit Title: 1.3.4 Web Technologies**	**No of Lessons: 15**

Overview: This unit is where learners understand how the internet works

Assessment: January Mock examination

Essential Knowledge (what must students know):	**Essential Skills (what must students be able to demonstrate):**	**Lesson topics:**
a) HTML, CSS and JavaScript. See appendix 5d. (b) Search engine indexing. (c) PageRank algorithm. (d) Server and client side processing **Terminology:** See key terms glossary A level computer science glossary of terms.pdf	**Students will be able to:** • Create a website using HTML, CSS and Javascript • Mathematical skills • the ability to analyse problems in computational terms through practical experience of solving such problems including writing programs to do • the capacity for thinking creatively, innovatively, analytically, logically and critically • the capacity to see relationships between different aspects of computer science • the ability to articulate the individual (moral), social (ethical), legal and cultural opportunities and risks of digital technology	• HTML key skills • CSS key skills • Java script keyskills • Search engine indexing • PageRank Algorighm • Server side processing • Client side processing
Careers Links: Links to web designers	**Enrichment:** NA	

Computing: Computer Science Medium Term Overview

Spring Term 1 ***Mrs Sharrock***	**Unit Title: Programming Languages**	**No of Lessons: 15**	

Overview: 1.2.4 Types of Programming Language

Assessment: Mock examination window in January

Essential Knowledge (what must students know): 1.2.4 Types of Programming Language (d) Modes of addressing memory (immediate, direct, indirect and indexed). 1.4.2 Data Structures (a) Arrays (of up to 3 dimensions), records, lists, tuples. (b) The following structures to store data: linked-list, graph (directed and undirected), stack, queue, tree, binary search tree, hash table. (c) How to create, traverse, add data to and remove data from the data structures mentioned above. (NB this can be either using arrays and procedural programming or an object-oriented approach). **Terminology:** See key terms glossary A level computer science glossary of terms.pdf	**Essential Skills (what must students be able to demonstrate):** **Students will be able to:** • an understanding of and ability to apply the fundamental principles and concepts of computer science including; abstraction, decomposition, logic, algorithms and data representation • Mathematical skills • the ability to analyse problems in computational terms through practical experience of solving such problems including writing programs to do • the capacity for thinking creatively, innovatively, analytically, logically and critically • the capacity to see relationships between different aspects of computer science • the ability to articulate the individual (moral), social (ethical), legal and cultural opportunities and risks of digital technology	**Lesson topics:** • Modes of memory addressing • Data Structures • Linked Lists • Stacks and Queues • Hash Tables • Graphs • Trees
Careers Links:	**Enrichment:** NA	

Computing: Computer Science Medium Term Overview

Spring Term 2 ***Miss Deavall***	**Unit Title:** 1.5 Legal, moral, cultural and ethical issues	**No of Lessons: 15**

Overview: Learners will investigate the individual moral, social, ethical and cultural opportunities and risks of digital technology. And the legislation surrounding the use of computers and ethical issues that can or may in the future arise from the use of computers

Essential Knowledge (what must students know):	**Essential Skills (what must students be able to demonstrate):**	**Lesson topics:**
1.5.1 (a) The Data Protection Act 1998. (b) The Computer Misuse Act 1990. (c) The Copyright Design and Patents Act 1988. (d)The Regulation of Investigatory Powers Act 2000 **1.5.2** The individual moral, social, ethical and cultural opportunities and risks of digital technology: • Computers in the workforce. • Automated decision making. • Artificial intelligence. • Environmental effects. • Censorship and the Internet. • Monitor behaviour. • Analyse personal information. • Piracy and offensive communications. • Layout, colour paradigms and character sets. **Terminology:** See key terms glossary A level computer science glossary of terms.pdf	**Students will be able to:** • the capacity to see relationships between different aspects of computer science • the ability to articulate the individual (moral), social (ethical), legal and cultural opportunities and risks of digital technology	• Data Protection Act • Computer Misuse Act • Copyright Act • Regulation of Investigatory Powers • Computers in the workforce. • Automated decision making. • Artificial intelligence. • Environmental effects. • Censorship and the Internet. • Monitor behaviour. • Analyse personal information. • Piracy and offensive communications. • Layout, colour paradigms and character sets.
Careers Links: Legislation in the workplace	**Enrichment:** NA	

Computing: Computer Science Medium Term Overview

Summer Term 1 ***Mrs Sharrock***	**Unit Title: Algorithms**	**No of Lessons: 15**

Overview:

Assessment: Examinations

Essential Knowledge (what must students know):	**Essential Skills (what must students be able to demonstrate):**	**Lesson topics:**
2.3.1 Algorithms (a) Analysis and design of algorithms for a given situation. (b) The suitability of different algorithms for a given task and data set, in terms of execution time and space. (c) Measures and methods to determine the efficiency of different algorithms, Big O notation (constant, linear, polynomial, exponential and logarithmic complexity). (d) Comparison of the complexity of algorithms. (e) Algorithms for the main data structures, (stacks, queues, trees, linked lists, depth-first (post-order) and breadth-first traversal of trees). (f) Standard algorithms (bubble sort, insertion sort, merge sort, quick sort, Dijkstra's shortest path algorithm, A* algorithm, binary search and linear search). **Terminology:** See key terms glossary A level computer science glossary of terms.pdf	**Students will be able to:** • an understanding of and ability to apply the fundamental principles and concepts of computer science including; abstraction, decomposition, logic, algorithms and data representation • Mathematical skills • the ability to analyse problems in computational terms through practical experience of solving such problems including writing programs to do • the capacity for thinking creatively, innovatively, analytically, logically and critically • the capacity to see relationships between different aspects of computer science • the ability to articulate the individual (moral), social (ethical), legal and cultural opportunities and risks of digital technology	• Dijkstra • A Star • Big O Notation
Careers Links:	**Enrichment:** NA	**Examination Dates:** **Paper 1:** **11th June 2025 AM (2.30)**

		Paper 2: **18th June 2025 AM (2.30)**