*Curriculum Map: Computing*
*2019/2020*

# Northgate High School

***Inspiring excellence, fulfilling potential***

***Respect – Readiness – Responsibility – Resilience - Reflectiveness***

| | Autumn Term | | Spring Term | | Summer Term | |
|---|---|---|---|---|---|---|
| | 1 | 2 | 1 | 2 | 1 | 2 |
| **Year 7** | **Content:**<br>Safe use of computers<br>Cyberbullying and being safe online<br>Using PowerPoint to design posters<br>Animations<br>**Skills:**<br>Awareness of being online. Using PowerPoint to make presentations and animations | **Content:**<br>Google SketchUp<br>Design a dream house using Computer Aided Design<br>Research skills and costing<br>**Skills:**<br>Students will develop an understanding of software to research, plan and design. | **Content:**<br>Databases using ACCESS<br>Writing reports in Word<br>Photoshop skills<br>**Skills:**<br>Using software to calculate cost and writing reports. Using computers to create visually interesting reports and web design. | **Content:**<br>Designing a festival using various software.<br>Using excel to make surveys and analyse data<br>**Skills:**<br>Using software to show how to present findings visually and how to present the information. | **Content:**<br>Programming using Scratch<br>**Skills:**<br>Understanding coding and how to use it when designing. | **Content:**<br>Programming using Micro Bits<br>**Skills:**<br>Understand how to use programming to show inputs, processes and outputs. |
| **Year 8** | Science:<br>**Content:**<br>Denary, Binary and Hexadecimal<br>Representing Text-ASCII codes<br>**Knowledge:**<br>Students will gain an awareness of binary numbers and how to convert a denary number 0-255 to a binary byte, and a binary byte to denary. They will be able to explain how this links to ASCII codes and the storage events that happen when a key is pressed. They will gain knowledge in hexadecimal codes including the terms "byte" , kB, and "nibble"<br>**Skills:**<br>Students will develop skills so they can convert binary to denary and denary to binary for various examples. Students will use Python's IDE to convert characters to binary values and binary values to characters (Only use of command line driven single event instructions). | Science:<br>**Content:**<br>Binary representing Sounds<br>**Knowledge:**<br>Students will learn about the process of sampling sounds using different quantisation levels ranging from 2-bit depth through to 4-bit depth and the effect this has on the quality of the sound. They will know that sounds are represented in a file by a sequence of binary codes that represent each sample.<br>**Skills:**<br>Students understand and develop the skills needed to carry-out simple quantisation on an analogue signal, sampling at different bit-depths (resolutions) and various sampling frequencies. They will develop the skills needed to estimate file sizes needed for storage of sounds. | Science:<br>**Content:**<br>Binary representing Images<br>**Knowledge:**<br>Students will gain knowledge in understanding bitmapped images using various bit-depths and the effect this and the resolution has on image quality. They will know the similarities between sound and image files.<br>Students will recognise the terms "bit-depth, resolution".<br>**Skills:**<br>Students will be able to recognise the possible number of colours used in an image from the bit depth and they will be able to explain the effect that resolution has on the quality of the bit mapped graphics. They will generate their own small image using binary representations of colours and relate this to the 24-bit RBG hexadecimal colour codes used in computers. | Science:<br>**Content:**<br>Using computer models to generate data:<br>Chemistry Anim. "Basic pH scale"<br>**Knowledge:**<br>Students will investigate the pH scales of common everyday substances, acids and alkali using an online animation<br>They will use Excel to provide an appropriate summary of this data in a graphical manner for analysis.<br>They could then compare this to a real experiment using data loggers to compare the pH's and some examples from the animation.<br>**Skills:**<br>Use animations and models to generate data. Develop skilled use of Excel to create a Bar chart or other appropriate data to present discrete data. Student will gain skills in using pH probes and data loggers. | Science:<br>**Content:**<br>Using Computer models to generate data:<br>Physics Anim. "Black-Body Radiation"<br>**Knowledge :**<br>Students will investigate the relationship between the maximum emission of wavelength and the temperature of an object. Students collect data using the animation. They use excel to plot a line chart or scatter graph to show the relationship between the two variables (continuous data). Research should be carried out to explain how this is used to determine the size of distant stars.<br>**Skills:**<br>Use of animations and models to generate data. Developing skilled use of Excel and research the Wien's displacement law to relate the equations presented against the actual data collected from the animation. | Science:<br>**Content:**<br>**Using Computer models to generate data – Biology**<br>Animation on "Natural Selection "<br>**Knowledge:**<br>Using an animation investigate different factors. Students use the model to investigate the effect of different factors on population growth, this includes mutations and competition / natural selection. They will then be able to write a report explaining the data they have collected and draw on the data provided by the animations.<br>**Skills:**<br>Use of animations and models to generate explanations of observed data. Developing skilled use of Excel, research and power-point. |
| | English:<br>"Interactive Adventure Game"<br>Stage: Analysis<br>**Skills:**<br>Recognise some of features of interactive adventure games. Use the common features of interactive games to devise success criteria. Devise success criteria which are SMART, and which can be used to test effectiveness. | English:<br>"Interactive Adventure Game"<br>Stage: Design<br>Develop a basic theme for the game and produce a basic story outline. Produce a design for a branching story. Deviate from the template provided to produce a design with additional complexity. | English:<br>"Interactive Adventure Game"<br>Stage: Prototyping<br>Produce a simple prototype with limited content before final development. Produce a detailed prototype which could be implemented by a third-party developer. As a result of the prototype, make changes to the design of the interactive story | English:<br>"Interactive Adventure Game"<br>Stage: Development part 1<br>Convert the paper- based story into an electronic story, with slides to represent the pages. Use hyperlinks between slides to make the story robust and immersive. Use a consistent style which is suitable for the story's audience. | English:<br>"Interactive Adventure Game"<br>Stage : Development part 2<br>Convert the paper- based story into an electronic story, with slides to represent the pages. Use hyperlinks between slides to make the story robust and immersive. Use a consistent style which is suitable for the story's audience. | English:<br>"Interactive Adventure Game"<br>Stage : Testing.<br>Use the results of a survey to write an evaluation of the project. Use carefully selected survey questions which seek to address specific success criteria. As a result of testing, make changes to the product to make it more fit for purpose. |

*Curriculum Map: Computing*
*2019/2020*

# Northgate High School

***Inspiring excellence, fulfilling potential***

***Respect – Readiness – Responsibility – Resilience - Reflectiveness***

| | | | | | | |
|---|---|---|---|---|---|---|
| **Year 9** | Science:<br><br>**Content:**<br>Introducing Basic Logic Gates and Boolean expressions.<br><br>**Knowledge:**<br>Students will gain knowledge in explaining the logic truth tables and how the Boolean expressions for the AND, OR and NOT gates relate to decisions made when solving logical problems. They will learn about the symbols used to represent each logic gate and how these logical expressions are of a Boolean type output. This will be reinforced with construction of basic logic circuits AND / OR gates using circuit boards and suitable components.<br><br>**Skills:**<br>Students will gain the skills to be able to relate simple circuits using logic gates with solutions to real problems, and how these are reflected in the truth tables. Students will learn how to build a basic logic circuit using just an LED, and 2 input switches.<br><br>English:<br><br>Creating Wiki pages for specified texts. Ongoing through-out year | Science:<br><br>**Content:**<br>Basic programming constructs<br><br>**Knowledge:**<br>Students will gain knowledge in the use of the three basic programming constructs that they will revisit in key stage 4. This is achieved through a guided task that involves the use of sequence; selection and iteration for a simulator that tests logic inputs for AND, NOT and OR gates and outputs the correct output according to the truth tables.<br><br>**Skills:**<br>Students will develop their skills in learning to use Python's IDE called "IDLE". They will write a simple program, compile, execute and debug it, from an algorithm provided in flowchart algorithmic form<br><br>English:<br><br>Creating Wiki pages for specified texts. Ongoing through-out year | Science:<br><br>**Content:**<br>"Mini-project" Logic simulator<br><br>**Knowledge:**<br>Students will gain knowledge in the use the micro-bits to create a logic circuit simulator that use the buttons on the micro-bits as the inputs A and B and the LED's as the outputs. The simulator will cover the logic gates AND, OR and NOT gates<br><br>**Skills:**<br>Students will develop their understanding on how microprocessors can be used to collect external stimulus, manipulate the data and create an output appropriately.<br><br>English:<br><br>Creating Wiki pages for specified texts. Ongoing through-out year | Science:<br><br>**Content:**<br>**Reaction timing:**<br>Analysing data using Histograms.<br><br>**Knowledge:**<br>Students will gather data using a reaction tester. Students will analyse class data, using excel and fixed size bins to generate a histogram to show the distribution of reaction times across the data set. They will be able to calculate the range of readings and averages. This investigation could be extended to compare data from "caffeine free" coke and "normal" coke drinkers<br><br>**Skills :**<br>Students will use excel to create a graph of the reaction time data. This data will be used in a python program in the next half term. They will make links between scientific thinking, data sets, shortfalls and the use of computers in manipulating data to make patterns, or lack of, visible.<br><br>English:<br><br>Creating Wiki pages for specified texts. Ongoing through-out year | Science:<br><br>**Content:**<br>Introducing some Python constructs sequence; iterations and selection to put the data created in previous reactions into 0.1 second bins. Data then plotted in Excel to show distribution of data on a histogram.<br><br>**Knowledge :**<br>Students will write and implement an algorithm using flowcharts to collate the data gained in the previous term to create fixed size bins and a histogram of reaction times. This data will be a large data set and so the use of a computer and algorithm will be required to process.<br><br>**Skills:**<br>Students will write some basic Python code including for, if and accumulative sums for reaction times within boundaries . This will require guidance.<br><br>English:<br><br>Creating Wiki pages for specified texts. Ongoing through-out year | Science:<br><br>**Content:**<br>"Mini-project" Moments<br><br>**Knowledge:**<br>Students will use the micro-bits to code a balance created from a ruler, pivot and small masses. They will use the micro-bits they have coded to carry-out experiments to take physical measurements.<br><br>**Skills:**<br>Students will develop their understanding on how microprocessors can be used to collect external stimulus, manipulate the data and create an output appropriately.<br><br>English:<br>Creating Wiki pages for specified texts. Ongoing through-out year |

*Curriculum Map: Computing*
*2019/2020*

# Northgate High School

***Inspiring excellence, fulfilling potential***

***Respect – Readiness – Responsibility – Resilience - Reflectiveness***

| | | | | | | |
|---|---|---|---|---|---|---|
| **Year 10** | **Content:**<br>Computational logic;<br>logic gates; Boolean expressions.<br>**Knowledge:**<br>Students will gain knowledge in why computers use binary, in terms of "transistors". They will know the equivalent transistor circuits of the logic gates. They will learn how to recognise solutions to problems using combinations of logical operators, to at least 2 levels. They will know the annotation used for logic gates "v" , "^" and "¬". They will be able to write expressions given the circuit and symbols and draw the circuits given the expressions .<br>**Skills:**<br>Applying logical thinking to solve various problems using combinational logic circuits and Boolean expressions. Designing, building and simulating logic circuits in online simulators such as logic.ly . They will develop skills in writing pseudo code to solve problems and designing flowcharts to match the requirements of a problem. They will also develop skills to use the three constructs to simulate the logic gates and circuits. This builds on lessons in year 8 at a more independent level. | **Content:**<br>Data Representation<br>**Knowledge:**<br>Students learn to apply computing-related mathematics though practical examples, this includes the operators +, –, /, *, Exponentiation (^), MOD and DIV. Students develop the skills needed to carry-out base conversions. Binary (bytes) to Denary and Denary to Binary (bytes) and hexadecimal. They will learn about binary shifts and bitwise logical operations. The need for, and common examples of, the check digit will also be covered.<br><br>**Skills:**<br>Students carry-out calculations using the mathematical operators to generate check-bits. They use their developing coding skills in Python to consolidate the learning in base conversions Bin2Den, Den2Bin, Bin2Hex and Hex2Bin. They also simulate parity check-bits.<br>Students will continually develop their Python coding skills through-out the course, here the use of lists, their functions and arrays are introduced. | **Content:**<br>Images, sound and compression<br>**Knowledge:**<br>Students will learn how an image is represented as a series of pixels represented in binary. They will know typical examples of metadata and the effect of colour depth and resolution on the quality and size of an image file. Students will also learn how sounds can be sampled and stored in digital form and how sampling frequency, sample size and the bit rate relate to a sound file. Students will also gain knowledge in compression techniques such as Lossless and Lossy and how they can be applied to images and sound.<br>**Skills:**<br>This builds on the skills developed of denary to analogue conversions i.e. methods of representing sounds as binary by sampling at different bit depths. Students will quantise sounds and images with different sampling frequencies and bit-depths. They will carry out calculations to determine the colour depth from number of bits and compress examples using Run-length encoding. | **Content:**<br>Algorithms and programming (searching)<br>**Knowledge:**<br>Students will start to solve problems more independently using computational thinking skills such as abstraction, decomposition and algorithmic thinking. They will gain knowledge in the functioning of standard searching algorithms such as the binary and linear search. Programming techniques will be delivered through every topic.<br>**Skills:**<br>Students will continue to develop their skills is the use of arrays and their associated indexes to access certain elements. They will continue to gain skills in using the three constructs: Iteration, Selection and Sequence to solve simple problems based around the ordering of lists. They will further develop their skills to plan algorithms using flowcharts following the computational thinking strands to simplify their solutions. They will use files to read and write data for storage and gain the skills needed for file manipulation . | **Content:**<br>Algorithms and programming (sorting)<br><br>**Knowledge:**<br>Students will continue to develop computational thinking skills such as abstraction, decomposition and algorithmic thinking. They will gain knowledge in the functioning of standard sorting algorithms such as bubble sort; merge sort; insertion sort. They will revisit the three Programming Constructs: sequence; selection; iteration. They will be led through the strands of computational thinking using two examples: "Chess Knight" moves challenge and a "maze" solving problem. Only the concepts covered here not the implementation (A level only)<br>**Skills:**<br>Students will recognise and apply the three programming constructs to basic algorithms with greater autonomy. They will be able to identify and recognise the strands of computational thinking when presented with developed solutions to real challenges (maze and knight moves challenges). They will recognise and identify some of these techniques used in everyday life e.g. underground maps. | **Content:**<br>NEA Preparation Project: Project Battleship<br><br>**Knowledge:**<br>Student will gain knowledge in techniques used to produce robust programs:<br>They will be introduced to appropriate techniques and apply them to examples such as using anticipation and protection. This will also include validation, sanitisation, authentication, maintenance and testing. Students will be fully aware of the NEA mark scheme. The project covers all the following sections : Analysis; design; development, testing and evaluating stages of a project design.<br><br>Skills:<br>Apply appropriate techniques to the development of a class led battleship game. Completion of a fully functional battleship project using the OCR Process for Success as a template for completion. |
| | **Content:**<br>Systems Architecture:<br>**Knowledge:**<br>Students will know the purpose of the CPU and what is meant by a Von Neumann architecture including the roles of the various registers in the function of the CPU as it fetches and executes instructions stored in memory. They will know how common characteristics of CPUs affect their performance. They will also know the purpose and examples of embedded systems.<br>**Skills:**<br>Students will start to make links between different parts of a computer and the computational logic module, e.g. the ALU for adding bytes and bitwise logical operators. They will be able to explain the fetch-decode execute cycles and how this relates to the registers in the CPU. | **Content:**<br>Memory; Storage<br>**Knowledge:**<br>Students will know the differences and purposes of RAM and ROM. They will also understand the need for virtual memory and how flash memory differs from other forms.<br><br>**Skills:**<br>Students will be able to determine suitable storage devices and storage media for a given application, and the advantages and disadvantages of these, using characteristics such as capacity; speed; portability; durability; reliability and cost. They will also gain the skills required to complete memory capacity calculations e.g. KB, MB etc | **Content:**<br>Wired and Wireless Networks<br>**Knowledge:**<br>Students will know the major types of networks LAN and WAN and factors that affect their performance. They will know the different roles of computers in a client-server and a peer-to-peer network, and the hardware needed to connect networks including their roles. They will also understand the internet is a worldwide collection of computer networks and explain terms such as DNS (Domain Name Server); hosting, the cloud and the concept of virtual networks.<br>**Skills:**<br>Researching and presenting skills "how networks interact" WAN versus LAN project) . | **Content:**<br>Networks Topologies and Protocols<br>**Knowledge:**<br>Students will learn about the Star and mesh networks they will make comparisons of the pros and cons. They will be able to explain the key ideas behind networks such as IP addresses; protocol's in general and provide the details of the TCP/IP standard. They will also know and explain the terms Virtual networks and the Cloud.<br>**Skills:**<br>Students will develop their coding skills through the development of some methods of encryption such as the Caesar cypher. | **Content:**<br>Systems software: Operating Systems and Utility software<br>Knowledge:<br>Students will know the purpose and functionality of systems software and operating systems including user interface; memory management; multitasking; peripheral management and drivers; user management and file management. They will also know of the term "utility" system software and a functional knowledge of encryption software; defragmentation; data compression and the role and methods of backup.<br>**Skills:**<br>Students will start to make links between software and hardware due to the interaction of the OS and hardware | |

*Curriculum Map: Computing*
*2019/2020*

# Northgate High School

***Inspiring excellence, fulfilling potential***

***Respect – Readiness – Responsibility – Resilience - Reflectiveness***

DEREHAM SIXTH FORM

## Year 11

**Content:**
Translators & Facilities
**Knowledge:**
Students will gain knowledge in the comparisons of High level and low-level languages, including common assembly code syntax. They will Investigate the relationship between High level languages and assembly code through examples providing 1-to-1 translation. They will know the roles of compilers, interpreters and assemblers and how each compare. They will gain knowledge in the use of common tools and facilities available in an integrated development environment (IDE): such as: editors; error diagnostics ; run-time environment and translators.
**Skills:**
Students will develop skills in simple assembly code instructions using LMC and related to the systems architecture. They will develop skills in using IDE to debug and apply set points within the program.

---

**Content:**
Ethical, legal, cultural & Environmental issues.
**Knowledge:**
Students will gain an understanding of the Ethical and environmental issues related to the life cycle and production of computers. They will gain knowledge in the laws governing data and computer usage including the Data Protection Act 1998; the Computer Misuse Act 1990; the Copyright Designs and Patents Act 1988; the Creative Commons Licensing and the freedom of information act.
They will also learn how to investigate and discuss the impact of Computer Science technologies while considering: ethical issues; legal issues; cultural issues; environmental issues and privacy issues.

**Skills:**
Developing researching skills. Students investigate the computer related laws through some case study examples. Skills developed involve research, analysis and evaluation of sources.

**Content:**
Manipulating Data
**Knowledge**
Students will be introduced to the concepts of databases, records and fields. They will learn the SQL key instructions and understand the concept and usefulness of databases and SQL queries.
**Skills:**
Students will be able to identify and write basic SQL code to query a database, including appropriate use of wildcards "*" to return field and the "%" wild character used in the LIKE command. They will also be able to use the SQL instructions: SELECT * FROM, WHERE , IN ORDER. This will include effective use of Boolean operators AND, OR and NOT to narrow down Queries.

---

**Content:**
System Security
**Knowledge:**
Students will gain knowledge in the details of the many threats to networks and the preventative measures. Students will develop an understanding in Threats such as malware; phishing; people as the 'weak point' in secure systems (social engineering); brute force attacks; denial of service attacks; data interception and theft; the concept of SQL injection; poor network policy. They will develop their understanding in methods of preventing vulnerabilities such as penetration testing; network forensics; network policies ; anti-malware software; firewalls; user access levels; passwords and encryption.
**Skills:**
Students will develop skills in comparing different attacks, how they are carried out and the implications of each. They will research and present case studies including malware including viruses, worms and phishing.

**Content:**

NEA-Programming Skills Audit
This is 20 hours of curriculum time (Course requisite). The sections cover:
• Success criteria – what key things must the solution contain?
• Planning and design – the solution is broken down and suitable designs created
• Development – iterative development with code explanations
• Testing and remedial actions – a log of successful tests including correcting any errors
• Evaluation – a review of the success criteria that have been met.

**Knowledge:**

Students will know how to identify and use variables, operators, inputs, outputs and assignments. They will understand and use the three basic programming constructs used to control the flow of a program including suitable loops including count and condition-controlled loops. They will know how to use different types of data, including Boolean, string, integer and real, appropriately in solutions to problem. They will know how to understand and use basic string manipulation. In addition, students will know how to use basic file handling operations: open; read; write and close. They will know how to define and use arrays as appropriate when solving problems and write functions and explain why these are needed in structured code.

**Skills:**

Students develop and enhance the programming skills delivered through-out the course in their development of a solution to a Dice game scenario. The solution will implement the following coding skills, which will be taught/revisited in preparation lessons throughout this stage of the course

Students will develop their skills in analysis and identification of the requirements for a solution to a problem. They develop skills to use abstraction and decomposition to design the solution to a problem. They will develop skills in identifying test procedures to be used during and after development to check their system against the success criteria. They will also develop skills in validation to ensure a robust solution to a problem.

**Content**

Revision topics for examinations:

Component 1:
- Systems Architecture
- Memory
- Storage
- Wired and wireless networks
- Network topologies, protocols and layers
- System security
- System software
- Ethical, legal, cultural and
- environmental concerns

Component 2:
- Algorithms *
- Programming techniques
- Producing robust programs
- Computational logic
- Translators and facilities of
- languages
- Data representation

Summer Examinations

*Curriculum Map: Computing*
*2019/2020*

# Northgate High School

***Inspiring excellence, fulfilling potential***

***Respect – Readiness – Responsibility – Resilience - Reflectiveness***

| | | | | | | |
|---|---|---|---|---|---|---|
| **Year 12** | **Contents:**<br>1.4.3 Boolean Algebra<br>Digital logic gates; k-maps; Boolean; adders; masks<br><br>**Knowledge:**<br>Students will define problems using Boolean logic. They will have the knowledge to manipulate Boolean expressions, including the use of Karnaugh maps to simplify Boolean expressions. They will know how to apply the following rules to derive or simplify statements in Boolean algebra: De Morgan's Laws, distributive, associative, commutative and double negation rules. They will use logic gate diagrams and truth tables competently for up to 4 variables.<br>They will understand the logic associated with D type flip flops, half and full adders.<br><br>**Skills**<br>Students will be able to apply all the tools delivered in this topic to create simplified solutions to problems involving up to four variables. They will be able to design circuits to meet solutions for various presented logic problems and be able to draw circuits from K-maps and truth tables and vice versa. | **Contents:**<br>1.1 The characteristics of contemporary processors, input, output and storage devices<br>Hardware; CISC and RISC machines; assembly code.<br><br>**Knowledge**<br>Students will gain knowledge about the parts of the CPU: ALU, Control Unit and Registers ( PC, ACC, MAR, MDR, CIR). They will also learn about the system buses: data, address and control and how this relates to assembly language programs. They will gain knowledge in how to follow and use Assembly language to create simple programs with the Little Man Computer instruction set. Including the modes of addressing memory such as immediate, direct, indirect and indexed addressing.<br>They will learn the stages of the Fetch-Decode-Execute Cycle and its effects on registers. In addition, they will learn about the factors affecting the performance of the CPU: clock speed, number of cores, cache and the use of pipelining in a processor to improve efficiency. The Von Neumann, Harvard and contemporary processor architecture and differences between and uses of CISC and RISC processors are covered. The following subjects are also covered: GPUs and their uses; Multicore and Parallel systems. The uses of magnetic, flash and optical storage devices; RAM, ROM and Virtual storage.<br><br>**Skills:**<br>Students will develop skills in writing assembly code including implementing branches and selections and explain how this relates to high level language code. They will be able to write assembly code to multiply two numbers together using the LMC assembler. They will be able to relate the LMC to the hardware in the CPU, including message flow, registers and buses | **Contents:**<br>1.4.2 Data Structures<br>Data structures: tuples, arrays, records, stacks, queues, linked-lists, hash tables, binary trees and graphs.<br><br>**Knowledge**<br>Students will gain knowledge to explain what abstract data types are, their applications and how they are implemented. Students will also gain knowledge in a variety of data structures including arrays of up to three dimensions, records, lists, tuples. They will know the characteristics of each type of data structure. Students will gain the knowledge of the following structures to store data: linked-list, graph (directed and undirected), stack, queue, tree, binary search tree, hash table. In addition, they will know how to create, traverse, add data to and remove data from the data structures.<br><br>**Skills:**<br>This is a coding opportunity to implement Abstract data types and develop pseudo coding skills.<br>Linked lists provide the opportunity to introduce classes at a basic level leading to Object Oriented Programming later in the year. The key focus is development of programming skills through practical implementation of the data structures. | **Contents:**<br>1.4.1 Data Types<br>Data types. Fixed point binary, two's compliment; floating point arithmetic<br><br>**Knowledge:**<br>Students will gain knowledge about Primitive data types such as integers, real/floating point, characters, strings and Boolean. They will learn how positive integers are represented in binary, but also how the use of sign and magnitude and two's complement can be used to represent negative numbers in binary. Students will also learn how to add and subtract binary integers. They will learn how to represent positive integers in hexadecimal and convert positive integers to hexadecimal.<br>In addition, students will gain the knowledge to represent fractional numbers as floating point numbers in exponent, mantissa binary normalised form. This will include using floating point arithmetic with positive and negative numbers i.e. addition and subtraction. Bitwise manipulation and masks: shifts, combining with AND, OR, and XOR and how character sets (ASCII and UNICODE) are used to represent text.<br><br>**Skills:**<br>Students will be able to generate two's complement binary format for negative numbers and convert fractional decimal values to fixed point binary. They will then be able to convert floating-point values in binary in mantissa and exponent format to denary values and via versa. They will develop the skills to be able to carry-out floating-point arithmetic. | **Contents:**<br>1.2 Software and software development<br>1.2.2 Applications Generation<br>1.2.3 Software Development<br><br>**Knowledge:**<br>Students will investigate the need for, function and purpose of operating systems. They will learn the many roles of the operating system such as: Memory Management including paging, segmentation and virtual memory; Interrupt handling and the role within the Fetch-Decode-Execute Cycle ; processor scheduling including round robin, first come first served, multi-level feedback queues, shortest job first and shortest remaining time. They will learn the about the different types of operating systems such as the Distributed, embedded, multi-tasking, multi-user and Real Time.<br>In addition, they students will learn about the role of the BIOS, Device drivers and Virtual machines. Students will investigate the nature of applications such as Utilities; translators: Interpreters, compilers and assemblers. They will learn all the Stages of compilation and the roles of Linkers and loaders and use of libraries. students will be able to discuss the relative merits and drawbacks of different software development methodologies such as the waterfall lifecycle, agile, extreme programming, the spiral model and rapid application development.<br><br>**Skills:**<br>The students must then apply these design techniques to a "mini-project" which is run through the remainder of the academic year. Programming skills; analysis and evaluative skills are developed through the project report. | **Contents:**<br>Programming techniques including:<br>1.2.4 Types of Programming Language.<br>Object Oriented Programming<br>Recursion<br><br>**Knowledge:**<br>Students will gain knowledge in Object-Oriented languages and associated pseudo-code style with an understanding of classes, objects, methods, attributes, inheritance, encapsulation and polymorphism.<br>Students will develop their programming skills through recursive algorithms and the use backtracking to solve maze problems.<br>They will gain knowledge in measures and methods to determine the efficiency of different algorithms using Big O notation for describing algorithm complexity.<br>They will develop and implement algorithms for the main data structures such as stacks, queues, trees, linked lists. They will also gain knowledge in search algorithms such as depth-first and breadth-first traversal. They will also learn Standard algorithms such as bubble sort, insertion sort, merge sort, quick sort as well as path optimisation algorithms such as Dijkstra's, A*, binary search and linear search.<br><br>Students also start the analysis section of the NEA coursework section of the course.<br><br>**Skills:**<br>Students will develop skills that will allow them to write backtracking algorithms in a maze finding game which uses the graph data structure, depth first, breadth first; A* and Dijkstra's algorithm. |

*Curriculum Map: Computing*
*2019/2020*

# Northgate High School

***Inspiring excellence, fulfilling potential***

***Respect – Readiness – Responsibility – Resilience - Reflectiveness***

| | | | | | |
|---|---|---|---|---|---|
| **Year 13** | **Contents:**<br>1.3 Exchanging data<br>1.3.1 Compression, Encryption and Hashing<br>1.3.2 Databases<br><br>**Knowledge:**<br>Students will gain knowledge in Lossy versus Lossless compression; they will know about examples including Run length encoding and dictionary coding. They will also know the differences and explain examples of Symmetric and Asymmetric encryption and the different uses of hashing. Students will also be able to explain the terms Relational databases, flat files, primary key, foreign key, secondary key, entity relationship modelling, normalisation and indexing. They will be able to normalise databases to third normal form and write SQL statements to generate and modify tables and generate queries. They will also know how to ensure referential integrity and what the terms transaction processing, atomicity, consistency, isolation, durability, record locking and redundancy mean. They will understand methods of capturing, selecting, managing and exchanging data.<br><br>**Skills:**<br>Students will be able to takes a flat file system and convert it to the 3rd normalised form relational database. They will be able to use SQL queries and generate tables form this relational database. | **Contents:**<br>1.3.3 Networks<br>1.3.4 Web Technologies<br><br>**Knowledge:**<br>Students will gain knowledge in the characteristics of networks and the importance of protocols and standards They will know the names of the layers and roles of each in the TCP/IP Stack protocol. They will also know the function of the DNS; Protocol layering; LANs and WANs including data transmission using Packet and circuit switching.<br>They will know the network specific hardware and how this is arranged in peer-to-peer and client server applications. They will know the implications of Network security and threats, use of firewalls, proxies and encryption. They will know the basic syntax for HTML, CSS and JavaScript. They will gain knowledge and be able to explain search engine indexing i.e. the PageRank algorithm. In addition; they will have some familiarity with the terms “Server” and “client-side” processing.<br><br>**Skills:**<br>Students should be able to make the links between the network protocols, hardware and devices, to explain how networks provide services to users through the internet and local networks. They will develop an html page with some PHP code to search a database using SQL and the XAMP server. | **Content:**<br>**1.5 Legal, moral, cultural and ethical issues**<br>1.5.1 Computing related legislation<br>1.5.2 Moral and ethical Issues<br><br>**Knowledge:**<br>Students will study and gain knowledge in “The Data Protection Act 1998” and “The Computer Misuse Act 1990”. They will also study “The Copyright Design and Patents Act 1988” and “The Regulation of Investigatory Powers Act 2000”. They will also study the individual moral, social, ethical and cultural opportunities and risks of digital technology. This section includes the general topics of moral and ethical issues such as computers in the workforce; automated decision making; artificial intelligence; environmental effects; censorship and the internet; monitoring behaviour ; analysis of personal information; piracy and offensive communications; layout, colour paradigms and character sets.<br><br>**Skills :**<br>Students will be able to apply their understanding of the laws and ethical issues to a scenario and respond with well-structured and balanced arguments. Analysis and Evaluation are the key skills tested here. | **Content:**<br>NEA Coursework<br>And<br>Revision<br><br>Time will be allocated through-out the course towards completion of the NEA aspect. This will account for a greater percentage of time towards the end of the course and this will be combined with revision lessons.<br><br>• Topic assessments are modular and half-termly.<br>• There are summative examinations in year 12 and 13 in December.<br>• Homework in year 12 are based on developing programming tasks through “Snakeify” or alternative class set specific tasks. This leads onto year 13 tasks based on project development. | **Content:**<br>Revision:<br><br>Revisit aspects of the course:<br><br>• Systems software<br>• Operating systems<br>• The characteristics of contemporary processors, input, output and storage devices<br>• Software and software Development<br>• Exchanging data<br>• Data types, data structures and algorithms<br>• Legal, moral, cultural and ethical issues<br>• Elements of computational thinking<br>• Problem solving and programming<br>• Algorithms to solve problems and standard algorithms. | |