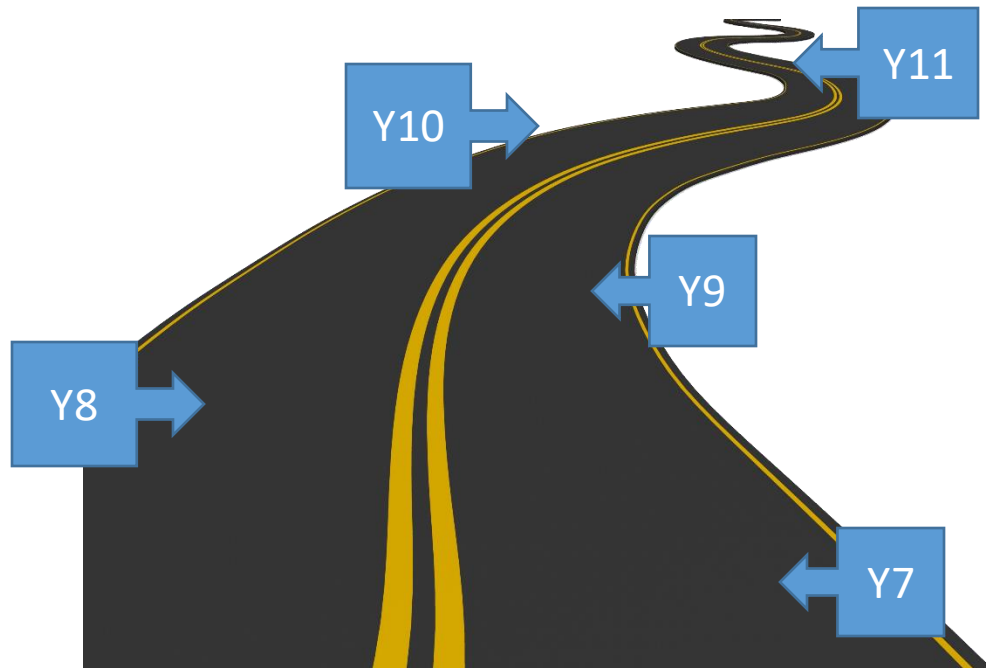




HALEWOOD
ACADEMY



Curriculum Milestones Year 10 - 11

Computer Science



KEY STAGE 4

At the end of Year 10, Students will know about and be able to do the following:	At the end of Year 11, Students will know about and be able to do the following:
Knowledge System Architectures: Characteristics of CPU architecture, including the Von Neumann model, and the role of CPU components in the fetch-execute cycle. Storage: Main and contemporary secondary storage methods, including magnetic, optical, and solid-state devices. Data Storage: Methods of storing data on devices and calculation of data capacity requirements. Hardware and Embedded Systems: Various hardware components and their roles, along with embedded systems. Algorithms: Understanding of binary search and merge sort algorithms. Algorithmic Problem Solving: Following and writing algorithms to solve problems, including the use of sequence, selection, and iteration. Data Types and Structures: The concept of data types (integer, Boolean, real, character, string) and data structures (records, one- and two-dimensional arrays). Number Representation: Representation of numbers in binary and hexadecimal; conversion between these and decimal; binary addition and shifts. Multimedia Representation: Representation of text, sound, and graphics inside computers. Boolean Logic: Use of Boolean logic operators (AND, OR, NOT) and combinations of these. Skills Problem Solving: Take a systematic approach to problem solving, including the use of decomposition and abstraction. Conventions: Utilize conventions such as pseudocode and flowcharts. Programming: Design, write, test, and refine programs using one or more high-level programming languages with a textual program definition, either to a specification or to solve a problem.	Knowledge Networks: Understanding the importance of connectivity (both wired and wireless), types of networks, common network topologies, and network security. Familiarity with networking protocols such as Ethernet, Wi-Fi, TCP/IP, HTTP, HTTPS, FTP, and email protocols. Cyber Security: Knowledge of various forms of cyber attacks (based on technical weaknesses and behaviour), methods of identifying vulnerabilities, and ways to protect software systems during design, creation, testing, and use. Ethical, Legal, and Environmental Impacts: Awareness of the ethical, legal, and environmental impacts of digital technology on wider society, including privacy and cyber security issues. Programming Languages: Characteristics and purpose of different levels of programming languages, including low-level languages. Recap: Revision and review of all topics covered. Skills Security Techniques: Implement appropriate security techniques, including validation and authentication. Algorithm Evaluation: Evaluate the fitness for purpose of algorithms in meeting requirements efficiently using logical reasoning and test data. Abstraction: Use abstraction effectively to model selected aspects of the external world in a program. Modular Programming: Structure programs into modular parts with clear, well-documented interfaces. Computing-Related Mathematics: Apply computing-related mathematics in problem-solving.