Bedford Greenacre Independent School

Curriculum Overview, Computer Science, Years 1 to 13

Ref: NCCE (National Centre for Computing Education): https://teachcomputing.org/curriculum

Juniors						
	Unit 1	**Unit 2**	**Unit 3**	**Unit 4**	**Unit 5**	**Unit 6**
Year 1	**Technology around us** Recognising technology in school and using it responsibly.	**Digital painting** Choosing appropriate tools in a program to create art, and making comparisons with working non-digitally.	**Moving a robot** Writing short algorithms and programs for floor robots, and predicting program outcomes.	**Grouping data** Exploring object labels, then using them to sort and group objects by properties	**Digital writing** Using a computer to create and format text, before comparing to writing non-digitally	**Programming animations** Designing and programming the movement of a character on screen to tell stories.
Year 2	**Information technology around us** Identifying IT and how its responsible use improves our world in school and beyond	**Digital photography** Capturing and changing digital photographs for different purposes.	**Robot algorithms** Creating and debugging programs, and using logical reasoning to make predictions.	**Pictograms** Collecting data in tally charts and using attributes to organise and present data on a computer	**Making music** Using a computer as a tool to explore rhythms and melodies, before creating a musical composition.	**Programming quizzes** Designing algorithms and programs that use events to trigger sequences of code to make an interactive quiz
Year 3	**Connecting computers** Identifying that digital devices have inputs, processes, and outputs, and how devices can be connected to make networks.	**Stop-frame animation** Capturing and editing digital still images to produce a stop-frame animation that tells a story.	**Sequencing sounds** Creating sequences in a block-based programming language to make music.	**Branching databases** Building and using branching databases to group objects using yes/no questions.	**Desktop publishing** Creating documents by modifying text, images, and page layouts for a specified purpose.	**Events and actions in programs** Writing algorithms and programs that use a range of events to trigger sequences of actions.
Year 4	**The internet** Recognising the internet as a network of networks including the WWW, and why we should evaluate online content.	**Audio editing** Capturing and editing audio to produce a podcast, ensuring that copyright is considered.	**Repetition in shapes** Using a text-based programming language to explore count-controlled loops when drawing shapes.	**Data logging** Recognising how and why data is collected over time, before using data loggers to carry out an investigation.	**Photo editing** Manipulating digital images, and reflecting on the impact of changes and whether the required purpose is fulfilled.	**Repetition in games** Using a block-based programming language to explore count-controlled and infinite loops when creating a game.
Year 5	**Sharing information** Identifying and exploring how information is shared between digital systems.	**Video editing** Planning, capturing, and editing video to produce a short film.	**Selection in physical computing** Exploring conditions and selection using a programmable microcontroller.	**Flat-file databases** Using a database to order data and create charts to answer questions.	**Vector drawing** Creating images in a drawing program by using layers and groups of objects.	**Selection in quizzes** Exploring selection in programming to design and code an interactive quiz.
Year 6	**Internet communication** Recognising how the WWW can be used to communicate and be searched to find information.	**Webpage creation** Designing and creating webpages, considering copyright, aesthetics, and navigation.	**Variables in games** Exploring variables when designing and coding a game.	**Introduction to spreadsheets** Answering questions by using spreadsheets to organise and calculate data.	**3D modelling** Planning, developing, and evaluating 3D computer models of physical objects.	**Sensing** Designing and coding a project that captures inputs from a physical device.

Bedford Greenacre Independent School

Curriculum Overview, Computer Science, Years 1 to 13

SENIORS					
	Unit 1	**Unit 2**	**Unit 3**	**Unit 4**	**Unit 5**
Year 7	**Impact of Technology** Identifying how to use online collaboration tools respectfully. An introduction to the computing lab.	**Interactive Animations and Games** Using block-based coding to create animations and games. Computing concepts and problem solving.	**Control Technology** Exploring real-world examples of control systems such as traffic lights and home automation.	**Networks** What is a network? Investigating the hardware to create a network, topologies, types and network performance. Understanding packet switching.	
Year 8	**Computer Systems** What is a computer? Understanding the key components and how they work together.	**Web Development** An introduction to HTML and CSS to create and edit web pages.	**JavaScript** Exploring commands and syntax of text-based coding in JavaScript to solve visual puzzles.	**Mobile App Development** Extending JavaScript to create a mobile game app.	**Vector Graphics** Creating SVG vectors, exploring Inkscape and fundamental graphics and design concepts.
Year 9	**Intro to Python** Introduction to text-based coding in Python. Understanding programming fundamentals in the language.	**Data Representation** What is binary and why do computers use it? How are numbers and other types of data stored as binary?	**Physical Computing** Coding the MicroBit computer and exploring analogue to digital conversion.	**Cybersecurity** What can we do with the data we share? Looking at data privacy from the attacker's perspective, malware, cyber-attacks and impacts.	**Minecraft** Coding in Minecraft, from raining chickens to automating building construction.

Bedford Greenacre Independent School

Curriculum Overview, Computer Science, Years 1 to 13

GCSE Computer Science						
	Unit 1	**Unit 2**	**Unit 3**	**Unit 4**	**Unit 5**	**Unit 6**
Year 10	**Introduction to Programming** Decomposition, data types and variables, debugging Inputs, Flowcharts **Binary data** Introduction to Base 2, Binary arithmetic,2s complement	**Program flow** String manipulation, If, If-else, Repetition **Data Representation** Hexadecimal, ASCII, Logical binary shifts	**Advanced programming** One-dimensional lists, repetition, sub programs **Stored Program concept** Fetch-decode execute, Secondary Storage	**Advanced programming** Two-dimensional lists, validation, linear search **Operating Systems** File management, process management, utility software	**Searching and sorting** Merge sort reading and writing files **Malware** Hackers, malware & anti-malware	**Turtle graphics** pens and lines, movement, colours, sub programs **Networks** LANS & WANS, connectivity wired v wireless, Network topologies
Year 11	**Advanced programming** Local & global variables, maths & times, problem solving **Networks** IoT, Packet Switching, TCP/IP	**Searching and Sorting** Binary search, Bubble sort, **Impacts** Environmental issues, ethical issues	**Trace Tables** Dealing with errors, problem solving **Data Representation** Bitmaps & sound	**Data Structures** Two-dimensional lists, test data, files **AI** AI & machine learning, privacy & ownership	**Revision**	**Revision**

A Level Computer Science						
	Unit 1	**Unit 2**	**Unit 3**	**Unit 4**	**Unit 5**	**Unit 6**
Year 12	**Information representation** Number bases, text, sound, graphics, compression **Algorithm design and problem solving**	**Hardware & Processor Fundamentals** Storage, Low Level programming, CPU **Data Types & structures** Arrays, ADT's	**Communication** LAN, WAN, P2P, client-server, wired & wireless networking **Searching & sorting** Standard sorting algorithms	**Ethics & Ownership** Law's, Copyright **ADTs** Lists, records, dictionary	**Security & Privacy** Networks, Malware **Databases** SQL	**System Software** Operating Systems **Problem solving**
Year 13	**Software development** Water fall models, testing **Programming** Python, Pseudocode, Object Orientated	**Data Representation** Two complement **Computational Thinking & problem solving** Big O notation, linked lists, binary trees	**Hardware & Virtual Machines** RISC, CISC, SISD, SIMD **Computational Thinking & problem solving** Recursion	**Communication & Internet Technologies** Protocols, switching, **Computational Thinking & problem solving** Reading & writing to files	**Revision**	**Revision**