SJBC Curriculum Termly Plan: Y10 Computer Science

Term	Topic(s) and links to other subjects	Core Knowledge	Core Vocabulary	Assessment	Resources
Autumn 1	Block 1	Students will cover the following objectives: • Define the term 'program' • Identify types of programs used every day • Identify Python as a programming language • Access an integrated development environment • Load and run a Python program • Change a Python program • Save a Python program • Use arithmetic operators and BIDMAS • Layout code to be readable and maintainable • Correct errors in programs • Use variables in algorithms and programs • Define the term 'decomposition' • Define the term 'algorithm' • Decompose a problem • Order the pieces of an algorithm (unplugged) • Order the pieces of an algorithm (IDE) • Define the term 'sequence' and use sequence in algorithms and program code • Interpret error messages • Correct errors in ordering • Recognise primitive data types (int, real, char, string) • Define the term 'variable' • Create variables of all types • Create meaningful identifier names • Assign values to variables, with the correct data types • View contents of memory (variable) in IDE • Take input and create output • Define the term 'runtime error' • Find and fix runtime errors • Use primitive data types (integer, real, char, string) • Translate code into flowchart symbols	Program Python IDE (Integrated Development Environment) Arithmetic Operator BIDMAS Variable Decomposition Algorithm Sequence Error Message Runtime Error Primitive Data Type Integer (int) Real Character (char) String Identifier Input Output Flowchart Flowchart Symbol Digital Computer Binary Bit Nibble Byte Overflow Error Signed Integer Unsigned Integer Two's Complement 8-bit Binary 2^n Rule	Rubric/ End of topic test (40 mins)	Core resources: Pearson-Edexcel-CS-SoW-Block-1 Enrichment and extension resources:

		• Represent an algorithm in a flowchart • Translate a flowchart into code • Represent algorithms in flowcharts • Create code from algorithms represented in flowcharts • Translate code into flowchart symbols • Represent an algorithm in a flowchart • Translate a flowchart into code • Define what is meant by the term 'digital computer' • Give examples of different types of computer • Define what is meant by the terms 'binary' and 'bit' • Explain why binary is used to represent data and program instructions in a computer • Describe the relationship between the number of available bits and the range of unique values that can be represented • Determine the number of unique values that can be represented by a binary pattern of a given length (2^n) • Define what is meant by the terms 'nibble' and 'byte' • Convert between denary and 8-bit binary numbers • Add together two positive 8-bit binary integers • Define what is meant by the term 'overflow error' • Describe the effects of an overflow error • Differentiate between signed and unsigned integers • Describe how positive and negative numbers are represented in two's complement • Find the two's complement of a positive binary number			
Autumn 2	Block 2	Students will cover the following objectives: • Use string manipulation functions (index, left, right, upper, lower, isalpha, …, etc.) • Use relational operators in flowchart and code • Use 'if' and 'if else' in code • Use flowchart decision symbol • Use relational operators in flowchart and code • Use 'if' and 'if else' in code	string index left right upper lower isalpha relational operators	End of topic test (40 mins)	Core resources: Pearson-Edexcel-CS-SoW-Block-2

		• Use 'if elif else' in code • Use flowchart decision symbol • Use comments, white space, meaningful identifiers, and indentation in code • Identify parts of code (variables, constants, selection, repetition) • Define 'AND', 'NOT' and 'OR' • Construct truth tables for Boolean operators and combinations • Use relational operators in flowchart and code • Use repetition (condition-controlled loops) in algorithms • Use repetition (condition-controlled loops) in code • Use repetition (condition-controlled loops) in flowcharts • Use flowcharts to represent selection and repetition • Identify parts of a program • Solve problems using code • Use repetition in code • Use selection in code • Convert between signed denary numbers and two's complement binary numbers • Determine the range of values that can be represented in two's complement by a binary number of a given length • Apply logical left and right shifts to binary integers • Use logical binary shifts to multiply and divide unsigned binary integers by powers of 2 • Explain why a number may be less precise after a binary shift right has been applied • Apply arithmetic left and right shifts to signed binary numbers • Describe how an arithmetic right shift differs from a logical right shift • Define what is meant by the term 'hexadecimal' • Explain why hexadecimal notation is used • Convert between hexadecimal and binary	if if else elif decision symbol comment white space identifier indentation variable constant selection repetition AND NOT OR truth table loop flowchart two's complement signed number logical shift arithmetic shift hexadecimal character set ASCII Unicode		Enrichment and extension resources:

		• Define what is meant by the term 'character set' • Describe how characters are represented in 7-bit ASCII • Given the ASCII code for one character derive the code for another • Outline the shortcomings of ASCII and how encoding systems that use more bits overcome them			
Spring 1	Block 3	Students will cover the following objectives: • Define the terms 'array' and 'list' • Access each item in a list using indexing • Create, append, delete items from a list • Explain that the range() function generates a sequence of numbers • Use iteration 'for' to process every item in a one-dimensional data structure • Define the term 'procedure' • Define the term 'parameter' • Create procedures • Define the term 'function' • Define the term 'return value' • Create functions • Use 'separation of concerns' • Use 'lists' • Use 'range()' • Use 'for' • Define what is meant by the 'stored program concept' • Describe the hardware components used in the von Neumann architecture and explain their role in the fetch-decode-execute cycle • Draw and label a diagram of the inside of a computer; label each hardware component and briefly describe its role • Explain how the speed of the clock impacts on performance • Explain how pipelining improves the performance of the CPU	Array List Indexing Append Delete range() function Iteration For Loop One-dimensional data structure Procedure Parameter Function Return Value Separation of Concerns Stored Program Concept Von Neumann Architecture Fetch-Decode-Execute Cycle Clock Speed Pipelining Address Bus Memory Locations Secondary Storage Magnetic Storage Optical Storage Solid-State Storage Capacity Speed Portability Data Storage Requirements	End of topic test (40 mins)	Pearson-Edexcel-CS-SoW-Block-3

		• Explain the relationship between the width of the address bus and the number of memory locations that can be addressed • Calculate the number of addressable memory locations provided by an address bus of a specified width • Explain why secondary storage is needed • Describe how data are stored on magnetic, optical and solid-state media • Compare the capacity, speed and portability of magnetic, optical and solid-state storage devices • Select an appropriate type of storage for a particular purpose • Construct an expression to calculate data storage requirements			
Spring 2	Block 4	Students will cover the following objectives: • Format output to meet requirements • Format output suitable for the end user • Give characteristics of one-dimensional and • two-dimensional data structures • Use indexing to access any item in a two-dimensional structure • Use 'for' to iterate over every item in a • two-dimensional structure • Use 'while' to find a row in a two-dimensional structure • Validate input using presence check, length check, range check, pattern check • Apply a linear search to a one-dimensional list (paper) • Complete a linear search algorithm in a flowchart • Write a linear search for a single item in a one-dimensional list (code) • Apply a linear search to a two-dimensional list (paper) • Complete a linear search algorithm in a flowchart • Write a linear search for a single record in a two-dimensional list (code) • Use one-dimensional and two-dimensional lists	Format output One-dimensional list Two-dimensional list Indexing For loop While loop Presence check Length check Range check Pattern check Linear search Flowchart Record File Operating system (OS) File system File permissions CPU scheduling Round-robin Paging Peripheral Driver	End of topic test (40 mins)	Pearson-Edexcel-CS-SoW-Block-4

		• Find a single item in a one-dimensional list • Find a single record and file in a two-dimensional list • Describe the role of the operating system in a computer system • Identify tasks carried out by an OS • Describe how the OS organises files and allocates space on a hard drive • Construct an expression to calculate the number of blocks of space on a hard drive needed to store a file of a given size • Describe how file permissions are used to control access to files • Select an appropriate level of file access (read, write, delete, none) for a user • Describe how an OS uses scheduling to give each active process a share of CPU time • Describe the features of the round-robin scheduling algorithm • Describe how the OS uses a paging algorithm to swap programs in and out of main memory. • Define what is meant by the term 'peripheral' • Describe how the OS uses drivers to communicate with and manage peripherals • Explain the purpose of a user interface and describe features of a user interface • Define what is meant by the term 'access control' • Describe commonly used methods of authentication • Select suitable access right for specified individuals • Define what is meant by the term 'utility software' • Identify different types of utility software • Describe the purpose of: – file repair/recovery software – backup/recovery software – file compression software – disk defragmentation software • Select which utility software tool to use for a particular task	User interface (UI) Access control Authentication Access rights Utility software File repair Backup software File compression Disk defragmentation		

Summer 1	Block 5	Students will cover the following objectives: • Describe the merge sort algorithm • Merge two sorted lists (paper, code) • Open files for reading • Read lines from text files • Close a file • Split lines on commas • Store items in lines as records in two-dimensional structure • Open files for writing • Construct comma-separated value line from record in two-dimensional structure • Write comma separated text (records) to a file • Define the term 'authentication' • Create a flowchart for algorithm • Implement authentication using a two-dimensional structure with at least two • columns • Validate input • Read and write files • Iterate over all records in a two-dimensional structure • Define what is meant by the term 'cyberattack' • Describe the financial, reputational and legal damage that a cyberattack can cause • Describe the characteristics of and threat posed by different types of malware • Describe how anti-malware works • Explain why it is important to keep anti-malware up-to-date • Define what is meant by the term 'hacker' • Explain why unpatched software is a target for hackers • Explain the function of a firewall • Explain how ethical hacking and penetration testing help identify vulnerabilities • Define what is meant by the term 'social engineering'	Merge Sort Sorted List File Handling Open File (Read/Write) Read Line Close File Split Record Two-Dimensional Structure CSV (Comma-Separated Values) Authentication Flowchart Input Validation Iterate Cyberattack Malware Anti-Malware Hacker Unpatched Software Firewall Ethical Hacking Penetration Testing Social Engineering Phishing Pretexting Baiting Quid Pro Quo Acceptable Use Policy Encryption Backup & Recovery Access Control Robust Software Code Vulnerability Secure Coding Bad Coding Practices Code Review Audit Trail	End of topic test (40 mins)	Pearson Edexcel CS SoW Block 5

		• Describe commonly used social engineering tactics (phishing, pretexting, baiting, quid pro quo) used by hackers • Explain the purpose of an acceptable use policy and what it typically includes • Explain how data are protected by encryption • Describe how backup and recovery procedures protect against data loss • Explain how access control helps to protect systems and data • Define what is meant by the term 'robust software' • Explain how a hacker can exploit a code vulnerability • Describe examples of bad coding practices and secure coding practices • Explain how code reviews and audit trails help to identify vulnerabilities			
Summer 2	Block 6	Students will cover the following objectives: • Decompose a problem • Use turtle graphics to draw lines • Use Cartesian coordinates • Incorporate selection, repetition, and iteration into turtle graphics • Use subprograms • Use turtle pens of different colours • Use turtle pens of different sizes • Use turtle fill in closed shapes • Combine subprograms to produce a turtle graphics image • Combine subprograms to create a solution • Use the turtle module, programming constructs, and subprograms to create images • Give reasons why computers are connected on a network • Differentiate between a LAN and a WAN • Categorise tasks according to the type of network used to carry them out	Decomposition Turtle Graphics Cartesian Coordinates Selection Iteration Subprogram Pen Colour Pen Size Fill Module Network LAN WAN Protocol IP Address Bandwidth Latency bps Transmission Rate Wired Connection Wireless Connection	End of topic test (40 mins)	Pearson Edexcel CS SoW Block 6

		• Explain the benefits to organisations of a WAN • Explain why protocols are needed on a network • Describe the purpose of an IP address • Define the meanings of the terms 'bandwidth' and 'latency' • Explain how bandwidth and latency affect the performance of a network • Use bits per second (bps) to describe network speed • Construct expressions involving file size, transmission rate and time • Differentiate between wired and wireless connectivity • Explain how data are transmitted along copper and fibre-optic cables • Compare the performance of copper and fibre-optic cables and give examples of their use • Describe how high-speed broadband is delivered • Describe how devices are connected on a wireless network • Compare the performance of wired and wireless LANs and give examples of situations where one is preferable to the other • Summarise the characteristic of Wi-Fi, Bluetooth, RFiD, Zigbee and NFC and give examples of their use • Define the term 'topology' • Describes the characteristics of bus, star and mesh network topologies • Draw and label a diagram of each topology • Match descriptions to network topologies • Match descriptions of what they do to internet components (backbone, POP, NAP, router)	Copper Cable Fibre-Optic Cable Broadband Wi-Fi Bluetooth RFiD Zigbee NFC Topology Bus Topology Star Topology Mesh Topology Backbone POP NAP Router		

SJBC Curriculum Termly Plan: Y11 Computer Science

Term	Topic(s) and links to other subjects	Core Knowledge	Core Vocabulary	Assessment	Resources
Autumn 1	Block 1	Students will cover the following objectives: Use: – sequence in programs – if, elif, else in programs – repetition (while) in programs – iteration (one-dimensional) in programs Define the terms: – 'procedure' – 'function' – 'parameter' – 'return value' Create: – procedures – functions Use: – 'separation of concerns' Define the terms: – 'local variable' – 'global variable' • justify using either or both • Be able to: – use math library methods – round real numbers to given decimal places • Write programs that take input and give output • Use arithmetic operators • Use maths library methods • Use time library methods • Format numeric output using round() • Define what is meant by the term 'embedded system' • Explain how an embedded system differs from a general-purpose computer	Sequence Repetition Iteration Condition Procedure Function Parameter Argument Return value Modularity Decomposition Local variable Global variable Encapsulation Embedded system General-purpose computer Microcontroller Dedicated function Firmware Internet of Things (IoT) Connected devices Sensors Network Sensors Device control	Rubric/ End of topic test (40 mins)	Core resources: Pearson-Edexcel-CS-SoW-Block-7 Enrichment and extension resources: *Students introduced to https://smartrevise.online/ and encouraged with weekly competitions. This term should focus on quiz and terminology sections.*

		• Identify hardware and software components of embedded systems • Describe applications of embedded systems • Define what is meant by the term 'Internet of Things' (IoT) • Explain the role of embedded systems in the • IoT • Outline security and privacy issues associated • with the IoT • Explain why power is an important consideration for many IoT devices • Describe how packet switching is used to • transmit data between devices on the • internet • Explain the purpose of an IP address. • Describe the role of routers • Explain how the TCP/IP stack enables different types of devices attached to different networks to communicate with each other across the internet • Put the layers of the stack in the correct order • Describe what each layer of the stack does • List the protocols that operate in each layer • Describe what each protocol does	Data privacy Authentication Encryption Hacking Packet switching Data packets Network Routing Packet headers Reassembly IP address Unique identifier Network address Device identification Router Packet forwarding Path selection Network traffic TCP/IP stack Communication protocols Network layers Encapsulation Interoperability		
Autumn 2	Block 2	Students will cover the following objectives: • Determine the value of variables using trace • tables • Predict output, give input • Identify errors (runtime, syntax, logical) • Fix errors (syntax, runtime, logical) • Describe the characteristics of a bubble sort • Apply a bubble sort algorithm to a list of items (numbers and strings)	Trace Table Variable Input Output Syntax Error Runtime Error Logic Error Bubble Sort Pass (in sorting)	End of topic test (40 mins)	Core resources: Pearson-Edexcel-CS-SoW-Block-8

		• Recognise, amend, and trace the code for a bubble sort • Describe the characteristics of a binary search • Apply a binary search algorithm to a list of items (numbers and strings) • Recognise, amend, and trace the code for a binary search • Decompose a problem • Read and write text files • Use subprograms that take parameters and return results • Write code in a high-level programming language • Compare algorithm efficiency (number of passes, number of compares) • Apply a bubble sort to a list • Apply a binary search to a sorted list • Trace an algorithm to determine state of variables • Locate and fix logic errors in algorithm (paper) • Describe the environmental impact of the manufacture of digital technology • Describe ways in which the environmental impact can be reduced • Describe how the energy consumed by digital devices harms the environment • Describe how energy consumption can be reduced • Define what is meant by the term 'e-waste' • Describe environmental issues associated with the disposal of digital technology • Explain how responsible recycling can reduce the environmental impact of digital technology • Explain how the short replacement cycle of mobile phones and other digital devices impacts on the environment • Define what is meant by the terms 'low-level language' and 'high-level language' • Explain why each processor has its own unique instruction set • Describe how writing a program in a low-level language differs from writing one in a high-level language	Binary Search Decomposition Subprogram Parameter Return Value High-Level Language Low-Level Language Translator Compiler Interpreter Machine Code Instruction Set Efficiency (Algorithm) Text File Environmental Impact E-waste Energy Consumption Responsible Recycling Intellectual Property Copyright Patent Trademark Open Source Software Proprietary Software		Enrichment and extension resources: *https://smartrevise.online/*

		• Compare features of low-level and high-level languages and identify tasks for which each is best suited • Explain the need for program translators • Define what is meant by the terms 'compiler' and 'interpreter' • Compare the way in which interpreters and compilers translate high-level code into machine code • Describe the advantages/disadvantages of each approach • Select and justify which method of translation to use for a given purpose • Define what is meant by the term 'intellectual property' • Describe possible consequences of IP theft • Explain how copyright, patents and trademarks help to protect IP • Compare features of open source and proprietary software			
Spring 1	Block 3	Students will cover the following objectives: • Use primitive data types • Define the terms valid, erroneous, boundary (extreme) data • Design data to test all three conditions • Test code using test data • Understand the characteristics of one-dimensional data structures • Choose appropriate use of one-dimensional data structures • Reverse traverse a one-dimensional data structure • Discuss efficiency considerations for one-dimensional structures • Determine the value of variables using trace tables • Predict output, give input • Locate errors (runtime, syntax, logical) • Fix errors (syntax, runtime, logical) • Decompose problems • Design subprogram interfaces • Read and write files • Use string manipulation methods	Primitive Data Types Valid Data Erroneous Data Boundary Data Test Data Trace Table One-Dimensional Data Structure Traversal Reverse Traversal Efficiency Runtime Error Syntax Error Logic Error Decomposition Subprogram Interface File Handling String Manipulation Bitmap Pixel	End of topic test (40 mins)	Pearson-Edexcel-CS-SoW-Block-9 Focus on Exam questions now: *https://smartrevise.online/*

		• Use one-dimensional data structures • Describe how bitmap images are represented in binary • Define what is meant by the terms 'bitmap', 'pixel', 'resolution' and 'colour depth' • Construct an expression to calculate the size of an image in pixels • Differentiate between image size and image resolution • Convert binary data into bitmap images and generate the binary data for bitmap images • Construct an expression to calculate the file size of an image (width x height x colour depth) and – given the file size and the values of any two of the variables – to calculate the value of the remaining one • Explain how the number of available bits impacts on the accuracy of the representation and why there is always a trade-off between resolution and storage space/bandwidth. • Differentiate between analogue and digital data • Define what is meant by the terms 'amplitude', 'sample rate', 'bit depth' and 'sample interval' • Describe the process of converting analogue sound into binary data. • Identify factors that affect the accuracy of the digital representation. • Draw and label a diagram illustrating ADC • Explain why an analogue sound is never fully reproducible in binary • Explain factors that affect the fidelity of the digital representation • Construct an expression to calculate the file size of a sound (sample rate x bit depth x time) • Give reasons for wanting to reduce file sizes (storage, streaming) • Describe how compression affects file sizes • Identify potential drawback of compressing files • Explain the difference between lossless and lossy compression	Resolution Colour Depth Analogue Data Digital Data Amplitude Sample Rate Bit Depth Sample Interval ADC (Analogue to Digital Conversion) Fidelity Compression Lossless Compression Lossy Compression		

		• Describe the advantages/disadvantages of each			
Spring 2	Block 4	Students will cover the following objectives: • Recall the characteristics of two-dimensional structures (record/entity/row, column/field, • mixed types) • Use indexing to locate records and fields in two-dimensional structure • Traverse a two-dimensional structure • Display a record/entity/row in columnar format • Define the terms local and global in terms of variables • Define the terms function, procedure, parameters, return value • Decompose problems • Write functions and procedures with/without parameters • Decompose a problem • Read a text file • Build a two-dimensional data structure • Search a two-dimensional data structure • Use string.format() to make output fit for • purpose • Create a flowchart given working code • Use trace tables to determine state of variables • Recognise the code for a bubble sort • Decompose a problem • Create subprograms (procedures, functions) using parameters and return values • Design and use test data (valid, erroneous, boundary) • Use a linear search • Use a trace table to find and fix errors • Create code for a bubble sort • Linear search two-dimensional structure • Design and create test data • Translate a flowchart to code • Define the meaning of the terms 'AI', 'machine learning' and 'robotics' • Describe applications of these technologies	Record Field Row Column Indexing Traverse Local Variable Global Variable Function Procedure Parameter Return Value Decomposition Subprogram Text File Two-Dimensional Structure Linear Search Bubble Sort Trace Table Flowchart string.format() Valid Data Erroneous Data Boundary Data AI (Artificial Intelligence) Machine Learning Robotics Algorithmic Bias Digital Footprint Personal Data Identity Theft Data Misuse Data Subject Rights UK Data Protection Act Computer Misuse Act	End of topic test (40 mins)	Pearson-Edexcel-CS-SoW-Block-10 Past Papers *https://smartrevise.online/*

		• Describe ethical issues associated with the use of these technologies • Describe safety and accountability issues associated with the use of these technologies • Define the meaning of the term 'algorithmic bias • Give examples of algorithmic bias • Weigh-up the benefits and drawbacks of these technologies and recommend how they should be regulated • Define what is meant by the term 'digital footprint' and give examples of activities in which digital footprints (active or passive) are generated • Explain how and why organisations collect personal data • Describe benefits and drawbacks of sharing personal data with other people and organisations. • Describe privacy concerns associated with the collection and use of personal data • Explain why it is difficult to attribute ownership of personal data to a specific individual • Define the meaning of the terms 'identity theft' and 'data misuse' • Explain the rights of data subjects and the obligations of organisations laid down in the UK Data Protection Act • Outline how the Computer Misuse Act deters criminals from stealing personal data • Give examples of misuse of personal data			
Summer 1		If any lessons left or Intervention students will go through Paul Longs Walking/Talking Mocks and also given the Exam technique and revision pack to work through as well as			Past Papers
Summer 2					