|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Features** | | | | | | | | | | | |
| * At KS1, the key knowledge is aligned with the National Curriculum and at Carlinghow Academy the following strands feature within our curriculum: * At KS2, the key knowledge is aligned with the National Curriculum’s strands of: * Assessment * Developing Experts Knowledge Organisers * Retrieval Challenge Grids | | | | | | | C:\Users\hlawless\AppData\Local\Microsoft\Windows\INetCache\Content.Word\knowledge logo.jfifSkills are reliant upon specific knowledge. A skill the capacity to perform from drawing upon retained knowledge.  C:\Users\hlawless\AppData\Local\Microsoft\Windows\INetCache\Content.Word\vocab logo.png  Children are taught specific vocabulary in line with their topic and the Statutory Spellings of their year group. | | | | |
| **National Curriculum** | | | | | | | | | | | |
| *KS1* | **Pupils should be taught to:**   * **understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions** * **create and debug simple programs** * **use logical reasoning to predict the behaviour of simple programs** * **use technology purposefully to create, organise, store, manipulate and retrieve digital content** * **recognise common uses of information technology beyond school** * **use technology safely and respectfully, keeping personal information private; identify where to go for help and support when they have concerns about content or contact on the internet or other online technologies** | | | | | | | | | | |
| *KS2* | **Pupils should be taught to:**   * **design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts** * **use sequence, selection, and repetition in programs; work with variables and various forms of input and output** * **use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs** * **understand computer networks including the internet; how they can provide multiple services, such as the world wide web; and the opportunities they offer for communication and collaboration** * **use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content** * **select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information** * **use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact.** | | | | | | | | | | |
| *Our Computing units* | **Computing systems and networks** | | | **Creating Media 1** | | **Creating Media 2** | | **Data & Information** | **Programming A** | | **Programming B** |
| **Sticky facts threading through our Computing curriculum units** | | | | | | | | | | | |
| *Strand* | | **Nursery** | **Reception** | | **Year 1**  **Year 2** | | **Year 3**  **Year 4** | | | **Year 5**  **Year 6** | |
| **Computing systems and networks** | |  | | | Cycle A –   * Identify examples of technology and explain how they can help us * Recognise that a computer is an example of technology * Describe what a keyboard is for * Know a computer stores work in files * Give examples of rules to keep them safe and healthy when they are using technology in and beyond the home   Cycle B –   * Recognise different types of computers used in schools * Identify that a computer is a part of information technology and describe some used of computers. * Recognise the features of information technology. * Identify information technology in school and beyond school. * To talk about the uses of information technology and how rules for using information technology can help us. * Explain how information technology benefits us, recognise that choice are made when using information technology and how to use it safely. | | **Cycle A –**   * **Input, Process, Output Concepts**: * Describe what an input is. * Explain that a process acts on the inputs. * Explain that an output is produced by the process. * Identify how changing the process can affect the output. * **Devices and Components**: * Identify input and output devices. * Recognize how a digital device consists of internal parts. * Explain that computer systems consist of hardware and software performing different tasks. * **Networking Concepts**: * Recognize that computers can be connected to each other. * Identify that few devices in a network are connected with one another. * Recognize that a network consists of a number of components. * Explain how information is passed through multiple connections. * Explain the role of switches, hubs, and routers within networks. * Identify that network devices transmit data. * Explain how information can be transmitted across networks.   **Cycle B –**   * **Understanding Networks and the Internet**: * Describe how networks connect to other networks. * Recognize that the global interconnection of networks is the internet. * Recognize the need for security on the internet. * **World Wide Web (WWW) Concepts**: * Recognize that the World Wide Web is part of the internet. * Explain that the internet enables us to view the World Wide Web. * Explain that the World Wide Web comprises websites and web pages. * **Access and Use of the WWW**: * Describe how to access the World Wide Web. * Outline how information can be shared via the World Wide Web. * Describe the types of content/media that can be added, created, and shared on the WWW. * Explain how the content of the WWW is created, owned, and shared by people. * **Critical Thinking and Evaluation**: * Describe current limitations of World Wide Web media. * Evaluate the reliability of content and the consequences of unreliable content. * Explain the benefits of the World Wide Web. | | | **Cycle A –**   * **Understanding Systems**: * Recognize that a system is a set of interrelated components with a common purpose. * Explain how components can be combined to perform tasks. * Identify what data can be gathered from systems. * Recognize inputs, processes, and outputs in larger IT systems. * **Search Engines**: * Describe the role of a search engine in a system. * Describe the input and output of a search engine. * Demonstrate that different search engines produce different results. * Explain why search engines differ and how they work together to produce results. * Explain the role of web crawlers in creating an index. * Explain how search results are ranked. * Explain that ranking orders results by relevance to the user's query or needs.   **Cycle B –**   * **Data Transfer and Networks**: * Recognize that data is transferred across networks using agreed protocols. * Explain that data is transferred in packets. * Recognize that connections between computers allow access to shared files. * **Internet Collaboration and Communication**: * Recognize that internet-connected computers enable collaboration across locations. * Discuss opportunities technology offers for communication and collaboration. * Outline methods of communicating and collaborating using the internet. * Choose appropriate methods of internet communication and collaboration for specific purposes. * Evaluate different methods of online communication and collaboration. * **Media Sharing and Online Safety**: * Explain which types of media can be shared through the internet. * Decide what should and should not be shared online. * Explain how internet communication and collaboration can be public or private. | |
| **Creating Media 1** | |  | | | **Cycle A –**   * Explain what different freehand tools do * Recognise that computers can be used to create a range of art * Recognise a tool can be adjusted   **Cycle B –**   * Explain some aspects of taking a good photograph * Know that a photo can be portrait or landscape | | **Cycle A –**  ● Explain the key requirements of the task  ● Storyboard has a clear beginning, middle, and end  ● Movement is smooth  ● The animation follows the storyboard  ● Make some improvements  ● Add some additional media  ● Evaluate how successful they were in meeting the task requirements  **Cycle B –**  ● Explain the key requirements of the task  ● Explain the key information that the podcast will include  ● Identify the types of sound that will be included  ● Include intro, main content, and outro sections in the plan  ● Voice recordings are clear and relevant  ● Appropriate audio is imported  ● Use editing tools to remove some unneeded sounds or pauses  ● Additional audio is appropriately placed to play alongside the voice recording  ● Appropriate volume is set on all tracks  ● Evaluate how successful they were in meeting the task requirements | | | **Cycle A –**  **●** Explain the key requirements of the task  ● Show sequence and progression with a clear beginning, middle, and end  ● Use appropriate filming techniques and capture the scenes from the storyboard  ● Captured audio is clear  ● Edit the video to join scenes, matching the storyboard  ● Evaluate how successful they were in meeting the task requirements  **Cycle B –**  ● Explain the key requirements of the task  ● Layout contains multiple sections  ● Layout relates to a relevant purpose/audience  ● Use copyright-free images  ● Design is clear and organised  ● Add subpages  ● Add internal and external hyperlinks  ● Suggest some improvements  ● Design considers how the page will look on different devices  ● Evaluate how successful they were in meeting the task requirements | |
| **Creating Media 2** | |  | | | **Cycle A –**   * Know that a keyboard is used to enter text into a computer * Know that the appearance of text can be changed   **Cycle B –**   * Reflect on a piece of music * Follow a rhythm pattern * Understand that a computer can generate different sounds * Understand that a computer can be used to make a sequence of notes * Understand how pattern and rhythm can be used to depict an animal | | **Cycle A –**  **●** Describe how different challenges require different solutions  ● Give an example of when using text, images or emojis online could be misinterpreted.  ● Choose an appropriate layout for a given scenario  ● Use placeholders appropriately to divide the page (magazine)  ● Add text and images  ● Format some of the text  ● Evaluate how successful they were in meeting the task requirements  **Cycle B -** | | | **Cycle A –**  **●** Explain the key requirements of the task  ● Choose an item in the classroom and consider how it’s relevant to the task  ● Add and remove objects to create a drawing of a chosen artefact  ● Use copy and paste to maintain consistency within the drawing  ● Manipulate an object’s size, colour, and proportion to represent a chosen artefact  ● Purposefully position and rotate objects  ● Move objects to different layers to create a specific aspect of a drawing  ● Manipulate multiple objects concurrently  ● Group objects to make them easier to work with  ● Evaluate how successful they were in meeting the task requirements  **Cycle B –**  ● Describe the purpose of their project: to create a 3D model of a building  ● Explain shapes that are representative of a real-world object to make a model  ● Recognise that changing perspective does not change the position of objects  ● Position 3D objects to create a chosen artefact  ● Accurately resize objects  ● Create holes in objects  ● Use and combine variations of one 3D shape  ● Evaluate how successful they were in meeting the task requirements | |
| **Data & information** | |  | | | **Cycle A –**   * Recognise that objects can be grouped * Know that labels are used to identify a group * Explain how objects have been grouped * Know that labels are used to identify a group with similar characteristics   **Cycle B –**   * Collect data using tally charts. * Suggest appropriate headings for tally charts. * Enter data onto a computer. * Present information using a computer. * Use a computer program to present information in different ways (e.g., graphs, tables, pictograms). * Recognize that people, animals, and objects can be described by attributes. * Compare objects grouped by attributes.: * Construct comparison questions (e.g., “Are there more \_\_\_ balls than \_\_\_ balls?”). * Use computers to answer comparison questions * Use pictograms to answer single-attribute questions. * Explain why some information should not be shared. | | **Cycle A –**   * **Yes/No Questioning and Data Structuring**: * Investigate questions with yes/no answers. * Identify attributes suitable for yes/no questions. * Create yes/no questions to divide objects. * Select attributes to split objects into two similarly sized groups. * Choose questions that divide objects into evenly sized subgroups. * Repeatedly create subgroups of objects. * Recognize that data sets can be structured using yes/no questions. * **Branching Databases**: * Explain that a branching database is an identification tool. * Identify objects using a branching database. * Retrieve information from different levels of a branching database. * Relate two levels of a branching database using "AND". * Explain that a well-structured branching database helps identify objects with fewer questions. * Suggest real-world applications for branching databases   **Cycle B –**   * **Data Collection and Logging**: * Suggest questions that can be answered using a table of data. * Identify data that can be logged over time. * Recognize that sensors are input devices. * Understand that sensors can be used for data collection. * Use a digital device to collect data automatically. * Choose an appropriate timeframe for automatic data collection. * Explain that a data logger saves sensor data points over time. * **Data Analysis and Output**: * Use logged data to find information. * Use a computer program to sort data by one attribute. * Export information in different formats. | | | **Cycle A –**   * **Organizing and Analyzing Data**: * Explain that computer programs can be used to organize data. * Use tools to select data to answer questions. * Outline different ways to view data. * Explain how ordering data helps answer certain questions. * **Filtering and Searching**: * Choose attributes and values (operands) to search by. * Use operands to filter data. * Ask questions that require more than one attribute to answer. * Choose which attribute to sort data by to answer a question. * **Visual Comparison and Presentation**: * Explain that computer programs can visually compare data. * Select appropriate graphs for visual comparison. * Choose suitable ways to present information to others.   **Cycle B –**   * **Understanding Spreadsheet Data**: * Identify questions that can be answered using spreadsheet data. * Explain what an item of data is in a spreadsheet. * Explain how data types affect how spreadsheets process data. * Outline different software tools available for working with data. * Explain the importance of organizing data in a spreadsheet. * **Using Formulas and Functions**: * Use functions to create new data. * Calculate data using formulas for various operations. * Recognize that cells can be linked. * Use existing cells within a formula. * Understand that a cell’s value updates automatically when a linked cell changes. * **Evaluating and Presenting Data**: * Evaluate results in relation to the original question. * Choose appropriate ways to present spreadsheet data. | |
| **Programming A** | |  | | | **Cycle A –**   * Explain what a given command does * Predict the outcome of a sequence involving up to four commands * Match a command to an outcome * Understand that a program is a set of commands that a computer can run * Know that a series of instructions can be issued before they are enacted   **Cycle B –**   * Understand a series of instructions * Understand different algorithms by changing the sequence of commands * Predict what a sequence of commands will do | | **Cycle A –**  ● Describe the purpose of the project, for example, to create sounds when keys are pressed  ● Choose a name that describes the action of the sprite  ● Choose relevant backdrops and costumes  ● Create an algorithm for each sprite  ● Explain what sequence means and demonstrate it in an algorithm  ● Adapt their code for additional named sprites  ● Explain why the code is in thar particular sequence  ● Run their code and identify if it meets the requirements of the task  ● Evaluate how successful they were in meeting the task requirements  **Cycle B –**   * **Understanding Impact**: * Explain what "impact" means. * Identify emerging risks from trends, regulations, market developments, and lessons learned. * Use tools like AI to predict potential impacts on the organization. * **Framework for Assessing Impact**: * Use a framework based on likelihood and consequence. * Identify preferred risk and impact. * **Organizational Alignment**: * Get buy-in from within the program. * Justify choices and keep stakeholders informed. * Link decisions back to risk, likelihood, and consequence to maintain consistency. * **Individual and Program-Level Actions**: * Explain that individuals can take actions to generate impact. * Assess individual actions using the same framework. * **Communication and Comfort**: * Ensure people feel comfortable with the process. * Acknowledge that this is a long-term, ongoing process. * Consider the distance to the desired future state. * **Continuous Improvement**: * Recognize the need for continuous improvement cycles for updates and personalization. | | | **Cycle A –**  **●** Describe the requirements of the task, including the use of selection  ● Construct a wiring diagram to show how components will be connected  ● Build a model that supports the hardware that will be used in the task  ● Write an algorithm that uses selection to control a sequence using output devices  ● Combine appropriate blocks to implement their algorithm  ● Suggest a strategy to fix the code when it is not working  ● Test their code with their model  ● Evaluate how successful they were in meeting the task requirements  **Cycle B –**   * **Comparing Hypotheses**: * It may not be possible to distinguish between two hypotheses. * One hypothesis may have a higher: * Prior probability. * Prior probability and/or likelihood. * Posterior probability. * Preference for one hypothesis may be based on other reasons (e.g., simplicity). * **Learning from Data**: * Data help us learn about parameters of interest (e.g., effect size). * Data also inform us about the uncertainty associated with these parameters. * **Bayesian Methods**: * Bayesian methods can be used for parameter estimation and uncertainty quantification. * The posterior distribution reflects updated beliefs after observing data. * The credible interval gives an interval estimate with a specified coverage probability. * The Bayes factor compares evidence for different models or hypotheses. * The posterior predictive distribution enables predictions about future observations. | |
| **Programming B** | |  | | | **Cycle A –**   * Explain what a sprite is * Compare different programming blocks * Know a series of commands can be joined together to form a program * Understand that a program is a set of commands a computer can run   **Cycle B –**   * Know that a sequence can be started using a variety of event blocks * Know that a sequence has an outcome, and identify different programs that have the same outcome * Know the backgrounds can be changed through the programming blocks * Understand the role of the numbers on ScratchJr blocks | | **Cycle A –**   * **Starting Point:** * Programs begin due to an **input**. * **Understanding Sequences:** * Explain what a **sequence** is. * Identify that a program includes **sequences of commands**. * **Developing Sequences:** * Build a **sequence of commands**. * Combine **commands** in a program. * Recognize that the **sequence** of a program is a **process**. * **Ordering Commands:** * Order **commands** in a program. * Understand that the **order** of commands affects the **output**. * **Outcome-Based Sequencing:** * Create a **sequence** to produce a **given outcome**. * Identify that: * Different sequences can achieve the **same output**. * Different sequences can achieve **different outputs**.   **Cycle B –**   * **Understanding Logic:** * Explain what **"logical"** means. * Explain that **logic can produce a given outcome**. * Justify the need to **follow certain rules**. * **Error Identification:** * Identify **emerging faults** in a program: * **Syntax errors** * **Semantic errors** * **Algorithmic Thinking:** * View problems from an **algorithmic perspective**. * Use **correct symbols** when writing algorithms. * **Pattern Recognition:** * Identify **patterns in a sequence**. * Identify **patterns in everyday life** (e.g., predicting what comes next). * **Problem Solving with Logic:** * Use an **operational approach** to represent logical problems. * Recognize that logic helps **break down problems** (decomposition). * Explain and justify **logical steps** taken to reach an outcome. * Explain whether a **solution is verified or disproved**. | | | **Cycle A –**   * **Understanding Conditions and Loops**: * A condition can only be true or false. * A loop contains a condition. * A condition-controlled loop repeats until the exit condition is met. * A count-controlled loop differs from a condition-controlled loop. * When an exit condition is met, all complete loops stop. * **Programming with Conditions**: * Choose appropriate conditions for use in a program. * Create condition-controlled loops. * **Using Selection in Programming**: * Use selection to switch program flow. * Use if, if...then..., and else statements to control flow in two ways. * Use selection to: * Break the flow of programs. * Check whether conditions have been met. * Use conditional statements in the correct order.   **Cycle B –**  ● Describe what will be shown if someone has walked more than a set number of steps  ● Identify what will be displayed and how the user will see it  ● Choose an appropriate name for a variable  ● Choose when and where to set a variable  ● Create an algorithm to describe how the program will process each input  ● Combine appropriate blocks to implement their algorithm  ● Run their code on the emulator to test their program  ● Propose a strategy to fix the code if it is not working  ● Evaluate how successful they were in meeting the task requirements | |
| **Vocabulary threading through our Computing curriculum strands** | | | | | | | | | | | |
| *Strand* | | **Nursery** | **Reception** | | **Year 1**  **Year 2** | | **Year 3**  **Year 4** | | | **Year 5**  **Year 6** | |
| **Computing systems and networks** | |  | | | **Cycle A -** technology, computer, mouse, trackpad, keyboard, screen, double-click, typing.  **Cycle B -** Information technology (IT), computer, barcode, scanner/scan | | **Cycle A –** digital device, input, process, output, program, digital, non-digital, connection, network, switch, server, wireless access point, cables, sockets  **Cycle B -** internet, network, router, security, switch, server, wireless access point (WAP), website, web page, web address, routing, web browser, World Wide Web, content, links, files, use, download, sharing, ownership, permission, information, accurate, honest, content, adverts | | | **Cycle A –** system, connection, digital, input, process, storage, output, search, search engine, refine, index, bot, ordering, links, algorithm, search engine optimisation (SEO), web crawler, content creator, selection, ranking.  **Cycle B -** communication, protocol, data, address, Internet Protocol (IP), Domain Name Server (DNS), packet, header, data payload, chat, explore, slide deck, reuse, remix, collaboration, internet, public, private, oneway, two-way, one-to-one, one-to-many. | |
| **Creating Media 1** | |  | | | **Cycle A -** paint program, tool, paintbrush, erase, fill, undo, shape tools, line tool, fill tool, undo tool, colour, brush style, brush size, pictures, painting, computers  **Cycle B -** music, quiet, loud, feelings, emotions, pattern, rhythm, pulse, pitch, tempo, rhythm, notes, create, emotion, beat, instrument, open, edit. | | **Cycle A –** text, images, advantages, disadvantages, communicate, font, style, landscape, portrait, orientation, placeholder, template, layout, content, desktop publishing, copy, paste, purpose, benefits.  **Cycle B -** audio, microphone, speaker, headphones, input device, output device, sound, podcast, edit, trim, align, layer, import, record, playback, selection, load, save, export, MP3, evaluate, feedback. | | | **Cycle A –** vector, drawing tools, object, toolbar, vector drawing, move, resize, colour, rotate, duplicate/copy, zoom, select, align, modify, layers, order, copy, paste, group, ungroup, reuse, reflection  **Cycle B -** website, web page, browser, media, Hypertext Markup Language (HTML), logo, layout, header, media, purpose, copyright, fair use, home page, preview, evaluate, device, Google Sites, breadcrumb trail, navigation, hyperlink, subpage, evaluate, implication, external link, embed. | |
| **Creating Media 2** | |  | | | **Cycle A -**  word processor, keyboard, keys, letters, type, numbers, space, backspace, text cursor, capital letters, toolbar, bold, italic, underline, mouse, select, font, undo, redo, format, compare, typing, writing.  **Cycle B -**  device, camera, photograph, capture, image, digital, landscape, portrait, framing, subject, compose, light sources, flash, focus, background, editing, filter, format, framing, lighting, | | **Cycle A –** animation, flip book, stop frame, frame, sequence, image, photograph, setting, character, events, onion skinning, consistency, evaluation, delete, media, import, transition.  **Cycle B -** image, edit, digital, crop, rotate, undo, save, adjustments, effects, colours, hue, saturation, sepia, vignette, image, retouch, clone, select, combine, made up, real, composite, cut, copy, paste, alter, background, foreground, zoom, undo, font | | | **Cycle A –** video, audio, camera, talking head, panning, close up, video camera, microphone, lens, mid-range, long shot, moving subject, side by side, angle (high, low, normal), static, zoom, pan, tilt, storyboard, filming, review, import, split, trim, clip, edit, reshoot, delete, reorder, export, evaluate, share.  **Cycle B -** TinkerCAD, 2D, 3D, shapes, select, move, perspective, view, handles, resize, lift, lower, recolour, rotate, duplicate, group, cylinder, cube, cuboid, sphere, cone, prism, pyramid, placeholder, hollow, choose, combine, construct, evaluate, modify. | |
| **Data & information** | |  | | | **Cycle A** - object, label, group, search, image, property, colour, size, shape, value, data set, more, less, most, fewest, least, the same  **Cycle B** - more than, less than, most, least, common, popular, organise, data, object, tally chart, votes, total, pictogram, enter, data, compare, objects, count, explain, attribute, group, same, different, conclusion, block diagram, sharing | | **Cycle A –** attribute, value, questions, table, objects, branching, database, objects, equal, even, separate, structure, compare, order, organise, selecting, information, decision tree.  **Cycle B -** data, table, layout, input device, sensor, logger, logging, data point, interval, analyse, dataset, import, export, logged, collection, review, conclusion. | | | **Cycle A –** database, data, information, record, field, sort, order, group, search, value, criteria, graph, chart, axis, compare, filter, presentation.  **Cycle B -** data, collecting, table, structure, spreadsheet, cell, cell reference, data item, format, formula, calculation, spreadsheet, input, output, operation, range, duplicate, sigma, propose, question, data set, organised, chart, evaluate, results, sum, comparison, software, tools. | |
| **Programming A** | |  | | | **Cycle A -**  Bee-Bot, forwards, backwards, turn, clear, go, commands, instructions, directions, left, right, route, plan, algorithm, program.  **Cycle B -**  instruction, sequence, clear, unambiguous, algorithm, program, order, prediction, artwork, design, route, mat, debugging, decomposition | | **Cycle A –** Scratch, programming, blocks, commands, code, sprite, costume, stage, backdrop, motion, turn, point in direction, go to, glide, sequence, event, task, design, run the code, order, note, chord, algorithm, bug, debug, code.  **Cycle B -** Logo (programming environment), program, turtle, commands, code snippet, algorithm, design, debug, pattern, repeat, repetition, count-controlled loop, value, trace, decompose, procedure. | | | **Cycle A –** microcontroller, USB, components, connection, infinite loop, output component, motor, repetition, count-controlled loop, Crumble controller, switch, LED, Sparkle, crocodile clips, connect, battery box, program, condition, Input, output, selection, action, debug, circuit, power, cell, buzzer  **Cycle B -** variable, change, name, value, set, design, event, algorithm, code, task, artwork, program, project, code, test, debug, improve, evaluate, share, assign, declare | |
| **Programming B** | |  | | | **Cycle A -**  ScratchJr, command, sprite, compare, programming, area, block, joining, start, run, program, background, delete, reset, algorithm, predict, effect, change, value, instructions, design.  **Cycle B -**  sequence, command, program, run, start, outcome, predict, blocks, design, actions, sprite, project, modify, change, algorithm, build, match, compare, debug, features, evaluate, decomposition, code. | | **Cycle A – m**otion, event, sprite, algorithm, logic, move, resize, extension block, pen up, set up, pen, design, action, debugging, errors, setup, code, test, debug, actions.  **Cycle B -** Scratch, programming, sprite, blocks, code, loop, repeat, value, infinite loop, count-controlled loop, costume, repetition, forever, animate, event block, duplicate, modify, design, algorithm, debug, refine, evaluate. | | | **Cycle A –** Selection, condition, true, false, count-controlled loop, outcomes, conditional statement, algorithm, program, debug, question, answer, task, design, input, implement, test, run, setup, operator  **Cycle B -** Micro:bit, MakeCode, input, process, output, flashing, USB, trace, selection, condition, if then else, variable, random, sensing, accelerometer, value, compass, direction, navigation, design, task, algorithm, step counter, plan, create, code, test, debug. | |