**Y2 Computing Whole School Progression of Knowledge and Skills**

**Digital Literacy, Online Safety and ICT**

**Computational Thinking**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **YEAR TWO** | | | | | |
| **Vocabulary/Significant Knowledge** | **Communicating**  **Text and Images**  How do I use a  computer as a writer? | **Communicating**  **Multimedia**  How do I create a multimedia story | **Understanding and**  **Sharing Data**  What is a branching database? | **Programming A**  How do I improve my algorithms? | **Programming B**  How do I improve my programs? |
| Computer Technology Hardware Software Password Input / Output Save / Open Document File Folder Font Edit Apps Personal Information Acceptable use Screen / mouse / microphone / keyboard / printer / speakers | Sound Text Image Video File Record Play Stop Pause Media Frame Animation Effect Soundtrack | Data Information Branching database Identify Chart Personal information Debug Private / public | program algorithm computer sequence instructions commands sprite to debug Plus directional language: forwards backwards left turn right turn | program to program algorithm computer sequence instructions commands to debug sprite Decompose/decomposition Plus directional language if using Bee-Bot: forwards backwards left turn right turn |
| **Enquiry Questions** | Can you open and edit a document?  How do these changes make a document more or less effective?  The computer gives us information via the screen, printer and speakers (output devices). How this is similar or different to devices pupils use at home?  How can we be safe and responsibly use technology?  Can we create an acceptable use policy? | What does stop-motion animation mean?  Can you create a short animation to make an object move?  Can you export the animation as a video and play back?  How could the animation be improved? | How can you find out information?  Where can information be stored?  Can you create a human branching database?  Can you search a given branching database to identify different objects?  Can you create a paper-based branching database?  Can you plan, create and test a branching database using a sequence of yes/no questions?  Can you peer test and review other pupils’ databases. | What is de bugging?  Can you complete unplugged activities to explore the importance of clear and instructions in algorithms? | What is important in an algorithm?  Why is debugging important in programming?  How can we debug? |
| **Skills** | - Create simple digital content for a purpose, e.g. digital art.  - Recognise that we can use technology to record and  playback audio or take and view photographs.  - Apply edits to digital content to achieve a particular effect, e.g. emphasise part of a text.  - Present ideas and information by combining media, e.g. text and images.  - Explain that you can search for information on the internet.  - Plan out digital content, e.g. a simple sketch or storyboard.  - Identify the common features of digital content, e.g. title, images.  - Recognise that we can use different types of media to convey information, e.g text, image, audio, video. | | - Identify different forms of digital content, i.e. text, image, video and audio.  - Recognise charts, pictograms and branching databases, and why we use them.  - Identify an object using a branching database  - Recognise an error in a branching database.  - Create a branching database using pre-prepared images and questions  - Identify the features of a good question in a branching database.  - Independently plan out and create a branching database.  - Evaluate a given branching database and suggest improvements. | - Explain that computers have no intelligence and we have to program them to do things.  - Create a program with multiple steps e.g. to control a floor robot.  - Predict the outcome of an algorithm or program with multiple steps.  - Recognise that the instructions in an algorithm need to be clear and unambiguous.  - Identify and correct errors in a given algorithm or program, and recognise the term debugging.  - Explain what an algorithm is, and that when inputted on a computer it is called a program.  - Plan out a program by creating an algorithm, and evaluate its success | |