



## Computing Knowledge and Skills Progression

### Computing Curriculum Statement of Intent

At TPA, we want pupils to be masters of technology and not slaves to it. Technology is everywhere and will play a pivotal part in students' lives. Therefore, we want to model and educate our pupils on how to use technology positively, responsibly and safely. We want our pupils to be creators not consumers and our broad curriculum encompassing computer science, information technology and digital literacy reflects this. We want our pupils to understand that there is always a choice with using technology and as a school we utilise technology (especially social media) to model positive use. We recognise that the best prevention for a lot of issues we currently see with technology/social media is through education. Building our knowledge in this subject will allow pupils to effectively demonstrate their learning through creative use of technology.

We recognise that technology can allow pupils to share their learning in creative ways. We also understand the accessibility opportunities technology can provide for our pupils. Our knowledge rich curriculum has to be balanced with the opportunity for pupils to apply their knowledge creatively which will in turn help our pupils become skilful computer scientists.

We encourage staff to try and embed computing across the whole curriculum to make learning creative and accessible. We want our pupils to be fluent with a range of tools to best express their understanding and hope by Upper Key Stage 2, children have the independence and confidence to choose the best tool to fulfil the task and challenge set by teachers.

### **Implementation**

Our whole curriculum is shaped by our school vision which aims to enable all children, regardless of background, ability, additional needs, to flourish to become the very best version of themselves they can possibly be – we are ambitious for all. We teach the National Curriculum, supported by a clear skills and knowledge progression. This ensures that skills and knowledge are built on year by year and sequenced appropriately to maximise learning for all children.

To ensure a broad range of skills and understanding, Computing is taught across three main strands: digital literacy, computer science and information technology. As part of information technology, children learn to use and express themselves and develop their ideas through ICT for example writing and presenting as well as exploring art and design using multimedia. Within digital literacy, children develop practical skills in the safe use of ICT and the ability to apply these skills to solving relevant, worthwhile problems for example understanding safe use of internet, networks and email. In computer science we teach children to understand and apply the fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation. Also to analyse problems to computational terms, and have repeated



practical experience of writing computer programs in order to solve such problems. We also teach a progression of Computing vocabulary to support children in their understanding.

At TPA, we give children access to a wide range of good quality resources and provide cross curricular opportunities for children to apply their Computing knowledge and skills. Online safety is taught within each Computing lesson as a short starter activity as well as being taught as a unit each year. Online safety procedures are communicated with all staff and parents.

**In EYFS**, computing learning begins in the Early Years at TPA. Though not explicitly mentioned in the EYFS curriculum, we ensure that all our learners develop a solid foundation by building on their curiosity. For example, classrooms could contain a role play area with a range of technology, both functioning and model / broken devices, or a variety of electronic toys, such as remote controlled cars, walkie-talkies and interactive pets, as part of continuous provision. Further technology could be included in conjunction with other activities, such as digital cameras for pupils to photograph their own learning, although children should ideally be given the opportunity to select and use technology for a certain purpose, rather than simply being given a device. The pedagogical approaches used in this age group should also be carefully considered, which includes the need to tinker, or play, with a device, in order to discover how it functions. Across the curriculum, in both the inside and outside areas, children are also encouraged to utilise simple programming tools such as Bee Bots. This allows them to build on their communication and language. In Expressive Arts, children could use programmes such as 2Paint a Picture.

### Computing National Curriculum

| Key Stage One   | Key Stage Two   |
|---|---|
| <p>Pupils should be taught to:</p> <ul style="list-style-type: none"> <li>● understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions</li> <li>● create and debug simple programs</li> <li>● use logical reasoning to predict the behaviour of simple programs</li> <li>● use technology purposefully to create, organise, store, manipulate and retrieve digital content</li> </ul> | <p>Pupils should be taught to:</p> <ul style="list-style-type: none"> <li>● design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts</li> <li>● use sequence, selection, and repetition in programs; work with variables and various forms of input and output</li> <li>● use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs</li> </ul> |



- 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.

- 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.

## Computing Terms

### 1. Substantive knowledge

Substantive knowledge in computing is understanding how to use technology, how to be safe and knowing how to program. This is developed through deliberate practice and by children applying their knowledge of how to be computational thinkers.

“Computational thinking is an important life skill, which all pupils now need to develop. It is central to both living in and understanding our digitally enriched world. It is a cognitive process involving logical reasoning by which problems are solved across the whole curriculum and through life in general.” (Computing at School, 2015)

In order to develop as computational thinkers, children engage with computational concepts and approaches:



| Concepts  | Approaches  |
|---|---|
| <ul style="list-style-type: none"><li>• Logic: predicting and analysing</li><li>• Algorithms: making steps and rules</li><li>• Decomposition: breaking down into parts</li><li>• Patterns: spotting and using similarities</li><li>• Abstraction: removing unnecessary detail</li><li>• Evaluation: making judgements</li></ul> | <ul style="list-style-type: none"><li>• Tinkering: experimenting and playing</li><li>• Creating: designing and making</li><li>• Debugging: fixing and finding errors</li><li>• Persevering: keeping going</li><li>• Collaborating: working together</li></ul> |

### **Disciplinary knowledge**

Disciplinary knowledge in computing is the use and interpretation of substantive knowledge in order to develop original digital content and programs.

### **Impact**

The implementation of this curriculum ensures that when children leave TPA, they are competent and safe users of ICT with an understanding of how technology works. They will have developed skills to express themselves and be creative in using digital media and be equipped to apply their skills in Computing to different challenges going forward.



|   | <u>Autumn 1</u>                                 | <u>Autumn 2</u>                                 | <u>Spring 1</u>   | <u>Spring 2</u>   | <u>Summer 1</u>  | <u>Summer 2</u>  |
|---|---|---|---|---|--|--|
| <p><b>Reception</b></p> <p><b>Computing Systems and Networks 1 - Using a computer /1 Hour of Code.org</b></p> <p><b>Programming 1 - All about instructions</b></p> <p><b>Computing Systems and network 1 - Exploring hardware</b></p> <p><b>Data Handling- Introduction to data</b></p> |   |   | <p>Lesson 1: Using a computer.</p> <p>Lesson 2: Logging in and out.</p> <p>Lesson 3: Mouse Control.</p> <p>Lesson 4: Mouse control (clicking)</p> <p>Lesson 5: Mouse control clicking and dragging.</p> | <p>Lesson 1: Following instructions.</p> <p>Lesson 2: Giving simple instructions.</p> <p>Lesson 3: Dressing up instructions.</p> <p>Lesson 4: Debugging instructions.</p> <p>Lesson 5: Predictions.</p> | <p>Lesson 1: Tinker Tray.</p> <p>Lesson 2: Real world Tinker Tray.</p> <p>Lesson 3: Pictures of play.</p> <p>Lesson 4: Pictures of walk.</p> <p>Lesson 5: Class photo album.</p> | <p>Lesson 1: Loose parts play.</p> <p>Lesson 2: Sorting ourselves.</p> <p>Lesson 3: Yes or no.</p> <p>Lesson 4: Creating a branching database.</p> <p>Lesson 5: Exploring pictograms</p> |
| <p><b>Year 1</b></p> <p><b>Programming 1 - All about instructions</b></p>   | <p>Curriculum Links: Early Years and year 1</p> | <p>Curriculum Links: Early Years and year 1</p> | <p>Lesson 1 : What is an algorithm?</p>   | <p>Lesson 1: Rocket Material</p> <p>Lesson 2: Rocket Design.</p>  | <p>Lesson 1: Planning A Photo Story.</p> <p>Lesson 2: Taking Photos.</p>   | <p>Lesson 1: Zoo Data.</p> <p>Lesson 2: Picture Data.</p>  |



|  |  |  |  |   |   |  |
|--|--|--|--|---|---|--|
| <p><b>Computing Systems and network 1 - Exploring hardware</b></p> <p><b>Data Handling- Introduction to data</b></p> | <p>Concepts &amp; Approaches:</p> <p>Creating, Pattern, Logic, Algorithms, Decomposition, Collaborating</p> <p>Three Autumn themed activities which see the children explore patterns in Garlands Galore, create a leaf labyrinth and make Pumpkin Soup using computational thinking skills.</p> | <p>Concepts &amp; Approaches:<br/>Algorithms, Creating, Collaboration, Decomposition, Tinkering, Persevering</p> <p>Snowmen scarves and patterns, creating igloos and bird feeders- all take centre stage in our three winter themed activities.</p> <p>Boats Ahoy! Takes children on a journey of discovery as they investigate boats. Four activities make up this set of resources. Includes different uses of boats, floating and sinking predictions, creating a good boat through exploring designs and role play.</p> | <p>Lesson 2: Algorithm Pictures.</p> <p>Lesson 3: Virtual Assistant.</p> <p>Lesson 4: Step by step.</p> <p>Lesson 5: Debugging directions.</p> | <p>Lesson 3: Rocket Building Instructions.</p> <p>Lesson 4: Making A Rocket.</p> <p>Lesson 5: Rocket Launching.</p> | <p>Lesson 3: Editing Photos.</p> <p>Lesson 4: Searching Images.</p> <p>Lesson 5: Photo Collage.</p> | <p>Lesson 3: MiniBeast Hunt.</p> <p>Lesson 4: Animal Branching Database</p> <p>Lesson 5: Invention</p> |
| <p><b>Year 2</b></p>   |  | <p>Curriculum Link: Year 2</p>   | <p>Lesson 1: Computer Parts.</p>   | <p>Lesson 1: Dinosaur Algorithm</p>   | <p>Lesson 1: Getting To Know The Keyboard.</p>  | <p>Lesson 1: Using ScratchJr</p>   |



# Twickenham PRIMARY ACADEMY

|   |  |  |   |  |   |   |
|---|--|--|---|--|---|---|
| <p><b>Code.org (Hour of Code)</b></p> <p><b>Code.org (Hour of Code)</b></p> <p><b>Code.org (Hour of Code)</b></p> <p><b>Code.org (Hour of Code)</b></p> <p><b>Code.org (Hour of Code)</b></p> <p><b>Code.org (Hour of Code)</b></p> | <p>Curriculum Links:<br/>Year 2</p> <p>Concepts &amp; Approaches:</p> <p>Creating, Pattern, Logic, Algorithms, Decomposition, Collaborating</p> <p>Three Autumn themed activities which see the children explore patterns in Garlands Galore, create a leaf labyrinth and make Pumpkin Soup using computational thinking skills.</p> | <p>Concepts &amp; Approaches:<br/>Algorithms, Creating, Collaboration, Decomposition, Tinkering, Persevering</p> <p>Snowmen scarves and patterns, creating igloos and bird feeders- all take centre stage in our three winter themed activities.</p> | <p><b>Lesson 2: Inputs</b></p> <p><b>Lesson 3: Technology Safari</b></p> <p><b>Lesson 4: Invention</b></p> <p><b>Lesson 5: Real World Roleplay.</b></p> | <p><b>Lesson 2 : Machine Learning.</b></p> <p><b>Lesson 3: Through The Maze</b></p> <p><b>Lesson 4: Making Maps.</b></p> <p><b>Lesson 5: Unplugged Debugging</b></p> | <p><b>Lesson 2: Getting Started With Word Processing.</b></p> <p><b>Lesson 3: Newspaper Writer.</b></p> <p><b>Lesson 4: Poetry Book.</b></p> <p><b>Lesson 5: What Happens When I Post Online.</b></p> | <p><b>Lesson 2: Creating An Animation.</b></p> <p><b>Lesson 3: Making A Musical Instrument.</b></p> <p><b>Lesson 4: Programming A Joke</b></p> <p><b>Lesson 5: 'The Three Little Pigs' Algorithms</b></p> |
| <p><b>Year 3</b></p> <p><b>Code.org (Hour of Code)</b></p> <p><b>Code.org (Hour of Code)</b></p>  | <p>Lesson 1: Hello world - Animals.</p> <p>Lesson 2: Hello World - Emoji</p>   | <p>Code.org creates an application from scratch using coding. This is done throughout the whole of Autumn 2.</p>   | <p><b>Lesson 1: What Is A Network.</b></p> <p><b>Lesson 2: A Files Journey.</b></p> <p><b>Lesson 3: A Websites Journey</b></p>                          | <p><b>Lesson 1: Tinkering With Scratch.</b></p> <p><b>Lesson 2: Using Loops.</b></p> <p><b>Lesson 3: Making An Animation.</b></p>                                    | <p><b>Lesson 1: Sending An Email.</b></p> <p><b>Lesson 2: Adding Attachments.</b></p> <p><b>Lesson 3: Be Kind Online.</b></p>   | <p><b>Lesson 1: Records, Field and Data.</b></p> <p><b>Lesson 2: Race Against The Computer.</b></p> <p><b>Lesson 3: Sorting and Filtering.</b></p>  |



|                                |                                  |   |                                 |                                       |                                |                                     |
|--------------------------------|----------------------------------|---|---------------------------------|---------------------------------------|--------------------------------|-------------------------------------|
| <b>Code.org (Hour of Code)</b> | Lesson 3: Hello World - Food     |   | Lesson 4: Routers               | Lesson 4: Storytelling.               | Lesson 4: Cyberbullying.       | Lesson 4: Representing Data.        |
| <b>Code.org (Hour of Code)</b> | Lesson 4: Hello World - Retro    |   | Lesson 5: Understanding Packets | Lesson 5: Programming A Game.         | Lesson 5: Fake Emails.         | Lesson 5: Planning A Holiday.       |
| <b>Code.org (Hour of Code)</b> | Lesson 5- Hello World - Soccer   |   |                                 |                                       |                                |                                     |
| <b>Code.org (Hour of Code)</b> | Lesson 6 - Hello World - Space   |   |                                 |                                       |                                |                                     |
| <b>Year 4</b>                  |                                  | Code.org creates an application from scratch using coding. This is done throughout the whole of Autumn 2. | Lesson 1: Teamwork              | Lesson 1: Scratch Reminder            | Lesson 1: Google Site Skills.  | Lesson 1: What Is Weather?          |
| <b>Code.org (Hour of Code)</b> | Lesson 1: Hello world - Animals. |   | Lesson 2: Sharing A Document.   | Lesson 2: Identifying What Code Does. | Lesson 2: Book Review Webpage. | Lesson 2: Weather Stations.         |
| <b>Code.org (Hour of Code)</b> | Lesson 2: Hello World - Emoji    |   | Lesson 3: Slide Presentation.   | Lesson 3: Introduction To Variables.  | Lesson 3: Creating A Webpage.  | Lesson 3: Extreme Weather.          |
| <b>Code.org (Hour of Code)</b> | Lesson 3: Hello World - Food     |   | Lesson 4: Google Forms.         | Lesson 4: Making A Variable.          | Lesson 4: Planning My Website. | Lesson 4: Satellites and Forecasts. |
| <b>Code.org (Hour of Code)</b> | Lesson 4: Hello World - Retro    |   | Lesson 5: Shared Spreadsheet.   | Lesson 5: Times Table Project.        | Lesson 5: Creating My Website. | Lesson 5: Presenting Forecasts.     |
| <b>Code.org (Hour of Code)</b> | Lesson 5- Hello World - Soccer   |   |                                 |                                       |                                |                                     |
| <b>Code.org (Hour of Code)</b> | Lesson 6 - Hello World - Space   |   |                                 |                                       |                                |                                     |



|  |   |  |   |  |   |  |
|--|---|--|---|--|---|--|
| <p><b>Year 5</b></p> <p><b>Code.org (Hour of Code)</b></p> <p><b>Code.org (Hour of Code)</b></p> <p><b>Code.org (Hour of Code)</b></p> <p><b>Code.org (Hour of Code)</b></p> <p><b>Code.org (Hour of Code)</b></p> <p><b>Code.org (Hour of Code)</b></p> | <p>Lesson 1: Hello world - Animals.</p> <p>Lesson 2: Hello World - Emoji</p> <p>Lesson 3: Hello World - Food</p> <p>Lesson 4: Hello World - Retro</p> <p>Lesson 5- Hello World - Soccer</p> <p>Lesson 6 - Hello World - Space</p> | <p>Code.org creates an application from scratch using coding. This is done throughout the whole of Autumn 2.</p> | <p>Lesson 1: Searching Basics.</p> <p>Lesson 2: Inaccurate Information.</p> <p>Lesson 3: Web Quest.</p> <p>Lesson 4: Information Poster.</p> <p>Lesson 5: Web Crawlers.</p> | <p>Lesson 1: Tinkering With Scratch Music Elements.</p> <p>Lesson 2: Scratch Soundtracks.</p> <p>Lesson 3: Planning A Soundtrack.</p> <p>Lesson 4: Programming A Soundtrack.</p> <p>Lesson 5: Battle Of The Bands.</p> | <p>Lesson 1: Mars Rover.</p> <p>Lesson 2: Binary Code.</p> <p>Lesson 3: Computer Architecture.</p> <p>Lesson 4: Using Binary - Numbers.</p> <p>Lesson 5: Using Binary - Text.</p> | <p>Lesson 1: Pixels.</p> <p>Lesson 2: Compressing Images.</p> <p>Lesson 3: Fetch, Decode and Execute.</p> <p>Lesson 4: Tinkering With CAD.</p> <p>Lesson 5: Tinkering CAD Tutorials.</p> |
| <p><b>Year 6</b></p> <p><b>Code.org (Hour of Code)</b></p> <p><b>Code.org (Hour of Code)</b></p>   | <p>Lesson 1: Hello world - Animals.</p> <p>Lesson 2: Hello World - Emoji</p>  | <p>Code.org creates an application from scratch using coding. This is done throughout the whole of Autumn 2.</p> | <p>Lesson 1: Secret Codes.</p> <p>Lesson 2: Brute Force Hacking.</p> <p>Lesson 3: Bletchley Park.</p>   | <p>Lesson 1: Tinkering With Logo.</p> <p>Lesson 2: Nested Loops.</p> <p>Lesson 3: Using Python.</p>  | <p>Lesson 1 : Barcodes.</p> <p>Lesson 2: Transmitting Data.</p> <p>Lesson 3: RFID</p>   | <p>Lesson 1: Creating Media.</p> <p>Lesson 2: Radio Plays.</p> <p>Lesson 3: First Computers.</p>   |



|                                |                                |  |                                    |                                  |                           |   |
|--------------------------------|--------------------------------|--|------------------------------------|----------------------------------|---------------------------|---|
| <b>Code.org (Hour of Code)</b> | Lesson 3: Hello World - Food   |  | Lesson 4: Computing Heroes.        | Lesson 4: Using Loops In Python. | Lesson 4: Using RFID.     | Lesson 4: Computers That Changed The World. |
| <b>Code.org (Hour of Code)</b> | Lesson 4: Hello World - Retro  |  | Lesson 5: Computing Heroes Part 2. | Lesson 5: Coding Mondrian.       | Lesson 5: Transport Data. | Lesson 5: Future Computer.                  |
| <b>Code.org (Hour of Code)</b> | Lesson 5- Hello World - Soccer |  |                                    |                                  |                           |   |
| <b>Code.org (Hour of Code)</b> | Lesson 6 - Hello World - Space |  |                                    |                                  |                           |   |