# Tavistock Primary School – Computing Curriculum

**Aims**

The national curriculum for computing aims to ensure that all pupils:

- can understand and apply the fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation
- can analyse problems in computational terms, and have repeated practical experience of writing computer programs in order to solve such problems
- can evaluate and apply information technology, including new or unfamiliar technologies, analytically to solve problems
- responsible, competent, confident and creative

**Attainment targets**

By the end of each key stage, pupils are expected to know, apply and understand the matters, skills and processes specified in the relevant programme of study.

**Key stage 1**

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

**Key stage 2**

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.

Inclusion

Information taken from the Teacher Handbook: SEND – Embedding inclusive practice pg 109

**Planning inclusive lessons**

# Tavistock Primary School – Computing Curriculum

Tasks – Incorporate learning materials that are accessible for learners of all abilities using specific resources or approaches that allow everyone to access the curriculum. Scaffold learning so that learners benefit from support in the initial stages of learning.
Problem Solving – Encourage learners to take ownership of their own learning. If the learner struggles with a multi-step problem, allow for additional support at the beginnings and slowly remove support as their skill set grows.
High Expectations – Challenge learners (and yourself) to keep high expectations and look for opportunities to connect learning to personal experience, meaning learning is relatable and purposeful.
Vocabulary – Find opportunities for learners to encounter tier 2 words. This will empower them to access their learning and communicate and understand ideas across the curriculum.
Vision impairment – Consider the use of braille where necessary; consider colours of resources, using screen readers and magnifier aids.
Space – Arrange the workspace so that children can fully engage with their learning, including children who need support with mobility.

**Reception –** See curriculum – EYFS Computing curriculum 24-25

*We currently have 1 mixed Reception/Year 1 class at Tavistock Primary. We deliver the EYFS and Year 2 curriculum topics simultaneously through practical, playful approach to learning, based on the needs and interests of our children. We teach children individually, in small groups and as a whole class. Through a combination of teacher led activities and continuous provision opportunities, we encourage children to develop their learning independently through exploration and challenge.*

*Year 1 and Year 2 mix - All children will still be taught and will learn the skills relevant to their age group, despite the mixing of year groups. This may be done by careful planning, or adaptations throughout the lesson to best meet the needs of the children which with then ensure they meet the learning outcomes for the sequence.*

|  | Autumn 1 Unit 1 | Autumn 2 Unit 2 | Spring 1 Unit 3 | Spring 2 Unit 4 | Summer 1 Unit 5 | Summer 2 Unit 6 |
|---|---|---|---|---|---|---|
| **Year 1** | Computing systems and networks – Technology around us | Creating media – Digital painting | Programming A – Moving a robot | Data and information – Grouping Data | Creating Media – Digital writing | Programming B – Programming animations |
| **Vocabulary** | technology, computer, mouse, trackpad, keyboard, screen, double-click, typing. | paint program, tool, paintbrush, erase, fill, undo, shape tools, line tool, fill tool, undo tool, colour, brush style, brush size, pictures, painting, computers | Bee-Bot, forwards, backwards, turn, clear, go, commands, instructions, directions, left, right, route, plan, algorithm, program. | object, label, group, search, image, property, colour, size, shape, value, data set, more, less, most, fewest, least, the same | 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. | ScratchJr, command, sprite, compare, programming, area, block, joining, start, run, program, background, delete, reset, algorithm, predict, effect, change, value, instructions, design. |
| **Knowledge and Skills** | Developing learners understanding of technology and how | Explore the world of digital art and its tools. | Introduction to early programming concepts. | Introduction to data and information. Begin | Creating and changing texts. Familiarise children | Introduction to onscreen programming, |

# Tavistock Primary School – Computing Curriculum

| | | | | | | |
|---|---|---|---|---|---|---|
| | it can help them. Become more familiar with different components of a computer by developing keyboard and mouse skills and consider how to use technology responsibly. | Create their own paintings inspired from other artists and give an opinion when painting and creating with digital devices | Explore using individual commands. Identify commands of a floor robot and predict the outcome of the programs. Introduction to algorithms. | to put objects into groups and label these. Sort objects into groups based on their properties and answer questions about data. | with typing and use toolkits look at writing. Comparing creating text on a computer to writing. | exploring investigating sprites and backgrounds. Children will use programming blocks to use, modify and create programs |
| Year 2 | Computing systems and networks – IT around us | Creating media – Digital photography | Programming A – Robot algorithms | Data and information – Pictograms | Creating media – Digital music | Programming B – Programming quizzes |
| Vocabulary | Information technology (IT), computer, barcode, scanner/scan | device, camera, photograph, capture, image, digital, landscape, portrait, framing, subject, compose, light sources, flash, focus, background, editing, filter, format, framing, lighting, | instruction, sequence, clear, unambiguous, algorithm, program, order, prediction, artwork, design, route, mat, debugging, decomposition | 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 | music, quiet, loud, feelings, emotions, pattern, rhythm, pulse, pitch, tempo, rhythm, notes, create, emotion, beat, instrument, open, edit. | sequence, command, program, run, start, outcome, predict, blocks, design, actions, sprite, project, modify, change, algorithm, build, match, compare, debug, features, evaluate, decomposition, code |
| Knowledge and skills | Explore how IT is used for good – benefits to sociaities, hospitals, libraries etc Making smart choices using technology | Recognise different devices used to capture photographs. Children will capture, edit and improve photos and understand the difference between reality and false images | Develop understanding of instructions in sequences and the use of logical reasoning to predict outcomes. Children give commands in different orders and develop | Introduction to 'data' and using this in a chart. Learning the word attribute and use this to organise data. Use data from graphs to answer questions. | Listening to music to make patterns digitally. Children will create different rhythms and tunes and share creations digitally and non digitally | Children understand that sequences of commands have an outcome and make predictions based on their learning. Children to make and modify designs for a quiz and evaluate their work, |

# Tavistock Primary School – Computing Curriculum

| | | | understanding on order affecting outcome. Children will develop artwork and test it for a program and test algorithms as programs and debug them. | | | making improvements when needed. |
|---|---|---|---|---|---|---|
| **Year 3** | Computing systems and networks – Connecting computers | Creating media – Stop-frame animation | Programming A – Sequencing sounds | Data and information – branching databases | Creating media – desktop publishing | Programming B – Events and actions in programs |
| **Vocabulary** | digital device, input, process, output, program, digital, non-digital, connection, network, switch, server, wireless access point, cables, sockets | animation, flip book, stopframe, frame, sequence, image, photograph, setting, character, events, onion skinning, consistency, evaluation, delete, media, import, transition. | 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. | attribute, value, questions, table, objects, branching, database, objects, equal, even, separate, structure, compare, order, organise, selecting, information, decision tree. | text, images, advantages, disadvantages, communicate, font, style, landscape, portrait, orientation, placeholder, template, layout, content, desktop publishing, copy, paste, purpose, benefits. | motion, event, sprite, algorithm, logic, move, resize, extension block, pen up, set up, pen, design, action, debugging, errors, setup, code, test, debug, actions. |
| **Knowledge and skills** | Challenge your learners to develop their understanding of digital devices, with an initial focus on inputs, processes, and outputs. Start by comparing digital and non-digital devices, before introducing them to | Learners will use a range of techniques to create a stop-frame animation using tablets. Next, they will apply those skills to create a story-based animation. This unit will | This unit explores the concept of sequencing in programming through Scratch. It begins with an introduction to the programming environment, which will be new to most learners. They will be introduced to a | Learners will develop their understanding of what a branching database is and how to create one. They will use yes/no questions to gain an understanding of what attributes are and how to | During this unit, learners will become familiar with the terms 'text' and 'images' and understand that they can be used to communicate messages. They will use desktop publishing software and consider | This unit explores the links between events and actions, whilst consolidating prior learning relating to sequencing. Learners will begin by moving a sprite in four directions (up, down, left and right). They will |

| | | | | | | |
|---|---|---|---|---|---|---|
| | computer networks that include network infrastructure devices like routers and switches. | conclude with learners adding other types of media to their animation, such as music and text. | selection of motion, sound, and event blocks which they will use to create their own programs, featuring sequences. The final project is to make a representation of a piano. The unit is paced to focus on all aspects of sequences, and make sure that knowledge is built in a structured manner. Learners also apply stages of program design through this unit. | use them to sort groups of objects. Learners will create physical and on-screen branching databases. To conclude the unit, they will create an identification tool using a branching database, which they will test by using it. They will also consider real-world applications for branching databases. | careful choices of font size, colour and type to edit and improve premade documents. Learners will be introduced to the terms 'templates', 'orientation', and 'placeholders' and begin to understand how these can support them in making their own template for a magazine front cover. They will start to add text and images to create their own pieces of work using desktop publishing software. Learners will look at a range of page layouts thinking carefully about the purpose of these and evaluate how and why desktop publishing is used in the real world. | then explore movement within the context of a maze, using design to choose an appropriately sized sprite. This unit also introduces programming extensions, through the use of pen blocks. Learners are given the opportunity to draw lines with sprites and change the size and colour of lines. The unit concludes with learners designing and coding their own maze tracing program. |
| **Year 4** | <span style="color:red">Computing systems and networks – The internet</span> | <span style="color:blue">Creating media – Audio production</span> | <span style="color:green">Programming A – Repetition in shapes</span> | <span style="color:orange">Data and information – Data logging</span> | <span style="color:pink">Creating media – Photo editing</span> | <span style="color:magenta">Programming B – Repetition in games</span> |
| **Vocabulary** | internet, network, router, security, switch, server, wireless access point (WAP), website, web page, | audio, microphone, speaker, headphones, input device, output device, | Logo (programming environment), program, turtle, commands, code snippet, algorithm, design, debug, | data, table, layout, input device, sensor, logger, logging, data point, interval, analyse, | image, edit, digital, crop, rotate, undo, save, adjustments, effects, colours, hue, saturation, sepia, vignette, | Scratch, programming, sprite, blocks, code, loop, repeat, value, infinite loop, count-controlled loop, |

|  | web address, routing, web browser, World Wide Web, content, links, files, use, download, sharing, ownership, permission, information, accurate, honest, content, adverts | sound, podcast, edit, trim, align, layer, import, record, playback, selection, load, save, export, MP3, evaluate, feedback. | pattern, repeat, repetition, count-controlled loop, value, trace, decompose, procedure. | dataset, import, export, logged, collection, review, conclusion. | image, retouch, clone, select, combine, made up, real, composite, cut, copy, paste, alter, background, foreground, zoom, undo, font. | costume, repetition, forever, animate, event block, duplicate, modify, design, algorithm, debug, refine, evaluate. |
|---|---|---|---|---|---|---|
| **Knowledge and skills** | Learners will apply their knowledge and understanding of networks, to appreciate the internet as a network of networks which need to be kept secure. They will learn that the World Wide Web is part of the internet, and will be given opportunities to explore the World Wide Web for themselves in order to learn about who owns content and what they can access, add, and create. Finally, they will evaluate online content to decide how honest, accurate, or reliable it is, and understand the consequences of false information. | Learners will identify the input device (microphone) and output devices (speaker or headphones) required to work with sound digitally. Learners will discuss the ownership of digital audio and the copyright implications of duplicating the work of others. In order to record audio themselves, learners will use Audacity to produce a podcast, which will include editing their work, adding multiple tracks, and opening and saving the audio files. Finally, | This unit is the first of the two programming units in Year 4, and looks at repetition and loops within programming. Pupils will create programs by planning, modifying, and testing commands to create shapes and patterns. They will use Logo, a text-based programming language. | In this unit, pupils will consider how and why data is collected over time. Pupils will consider the senses that humans use to experience the environment and how computers can use special input devices called sensors to monitor the environment. Pupils will collect data as well as access data captured over long periods of time. They will look at data points, data sets, and logging intervals. Pupils will spend time using a computer to review and | Learners will develop their understanding of how digital images can be changed and edited, and how they can then be resaved and reused. They will consider the impact that editing images can have, and evaluate the effectiveness of their choices. | This unit explores the concept of repetition in programming using the Scratch environment. It begins with a Scratch activity similar to that carried out in Logo in Programming unit A, where learners can discover similarities between two environments. Learners look at the difference between count-controlled and infinite loops, and use their knowledge to modify existing animations and games using repetition. Their final project is to design and create a game which uses |

# Tavistock Primary School – Computing Curriculum

| | | learners will evaluate their work and give feedback to their peers. | | analyse data. Towards the end of the unit, pupils will pose questions and then use data loggers to automatically collect the data needed to answer those questions | | repetition, applying stages of programming design throughout. |
|---|---|---|---|---|---|---|
| **Year 5** | Computing systems and networks – Systems and searching | Creating media – Video production | Programming A – Selection in physical computing | Data and information – Flat-file databases | Creating media – Introduction to vector graphics | Programming B – Selection in quizzes |
| **Vocabulary** | 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. | 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 | 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 | database, data, information, record, field, sort, order, group, search, value, criteria, graph, chart, axis, compare, filter, presentation. | 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 | Selection, condition, true, false, count-controlled loop, outcomes, conditional statement, algorithm, program, debug, question, answer, task, design, input, implement, test, run, setup, operator |
| **Knowledge and skills** | Learners develop their understanding of computer systems and how information is transferred between | This unit gives learners the opportunity to learn how to create short videos in groups. | Learners will use physical computing to explore the concept of selection in programming through the use of | This unit looks at how a flat-file database can be used to organise data in records. Pupils use tools | In this unit, learners start to create vector drawings. They learn how to use different drawing tools to | pupils develop their knowledge of selection by revisiting how conditions can be used in programs |

| | | | | | |
|---|---|---|---|---|---|
| | systems and devices. Learners consider small-scale systems as well as large-scale systems. They explain the input, output, and process aspects of a variety of different real-world systems. Learners discover how information is found on the World Wide Web, through learning how search engines work (including how they select and rank results) and what influences searching, and through comparing different search engines. | As they progress through this unit, they will be exposed to topic-based language and develop the skills of capturing, editing, and manipulating video. Active learning is encouraged through guided questions and by working in small groups to investigate the use of devices and software. Learners are guided with step-by-step support to take their idea from conception to completion. At the teacher's discretion, the use of green screen can be incorporated into this unit. At the conclusion of the unit, learners have the opportunity to reflect on and assess their progress in creating a video. | the Crumble programming environment. Learners will be introduced to a microcontroller (Crumble controller) and learn how to connect and program components (including output devices- LEDs and motors) through the application of their existing programming knowledge. Learners are introduced to conditions as a means of controlling the flow of actions and make use of their knowledge of repetition and conditions when introduced to the concept of selection (through the if, then structure) | within a database to order and answer questions about data. They create graphs and charts from their data to help solve problems. They use a real-life database to answer a question, and present their work to others. | help them create images. Learners recognise that images in vector drawings are created using shapes and lines, and each individual element in the drawing is called an object. Learners layer their objects and begin grouping and duplicating them to support the creation of more complex pieces of work. | and then learning how the If… Then… Else structure can be used to select different outcomes depending on whether a condition is true or false. They represent this understanding in algorithms and then by constructing programs using the Scratch programming environment. They use their knowledge of writing programs and using selection to control outcomes to design a quiz in response to a given task and implement it as a program. |
| **Year 6** | Computing systems | Creating media – | Programming A – | Data and | Creating media – | Programming B – |

# Tavistock Primary School – Computing Curriculum

| | and networks – Communication and collaboration | Web page creation | Variables in games | information – Introduction to spreadsheets | 3D modelling | Sensing movement |
|---|---|---|---|---|---|---|
| **Vocabulary** | 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. | 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. | variable, change, name, value, set, design, event, algorithm, code, task, artwork, program, project, code, test, debug, improve, evaluate, share, assign, declare | 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. | 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. | 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. |
| **Knowledge and skills** | In this unit learners explore how data is transferred over the internet. Learners initially focus on addressing, before they move on to the makeup and structure of data packets. Learners then look at how the internet facilitates online communication and collaboration; they complete shared projects online and | This unit introduces learners to the creation of websites for a chosen purpose. Learners identify what makes a good web page and use this information to design and evaluate their own website using Google Sites. Throughout the process learners | This unit explores the concept of variables in programming through games in Scratch. First, learners find out what variables are and relate them to real-world examples of values that can be set and changed. Then they use variables to create a simulation of a scoreboard. In Lessons 2, 3, and | This unit introduces the learners to spreadsheets. They will be supported in organising data into columns and rows to create their own data set. Learners will be taught the importance of formatting data to support calculations, while also being | Learners will develop their knowledge and understanding of using a computer to produce 3D models. Learners will initially familiarise themselves with working in a 3D space, moving, resizing, and duplicating objects. They will then create hollow objects using | This unit is the final KS2 programming unit and brings together elements of all the four programming constructs: sequence from Year 3, repetition from Year 4, selection from Year 5, and variables (introduced in Year 6 – 'Programming A'). It offers pupils the opportunity to use all of these |

| | | | | | |
|---|---|---|---|---|---|
| evaluate different methods of communication. Children learn how to communicate responsibly by considering what should and should not be shared on the internet. | pay specific attention to copyright and fair use of media, the aesthetics of the site, and navigation paths. | 5, which follow the Use-Modify-Create model, learners experiment with variables in an existing project, then modify them, before they create their own project. Focus on design and apply their knowledge of variables and design to improve their games in Scratch. | introduced to formulas and will begin to understand how they can be used to produce calculated data. Learners will be taught how to apply formulas that include a range of cells, and apply formulas to multiple cells by duplicating them. Learners will use spreadsheets to plan an event and answer questions. Finally, learners will create charts, and evaluate their results in comparison to questions asked. | placeholders and combine multiple objects to create a model of a desk tidy. Finally, learners will examine the benefits of grouping and ungrouping 3D objects, then go on to plan, develop, and evaluate their own 3D model of a building | constructs in a different, but still familiar environment, while also utilising a physical device — the micro:bit. The unit begins with a simple program for pupils to build in and test within the new programming environment, before transferring it to their micro:bit. |
| **Year 7** | <td colspan="5">• design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems<br>• understand several key algorithms that reflect computational thinking [for example, ones for sorting and searching; use logical reasoning to compare the utility of alternative algorithms for the same problem<br>• use 2 or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions<br>• understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming; understand how numbers can be represented in binary, and be able to carry out simple operations on binary numbers [for example, binary addition, and conversion between binary and decimal]<br>• understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems<br>• understand how instructions are stored and executed within a computer system; understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits</td> |

| | |
|---|---|
| | • undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users<br>• create, reuse, revise and repurpose digital artefacts for a given audience, with attention to trustworthiness, design and usability<br>• understand a range of ways to use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy; recognise inappropriate content, contact and conduct, and know how to report concerns |