Computing



Subject Overview

Limitless Dreams Endless Opportunities

Computing curriculum intent

We are educating pupils who will one day work in careers that may as yet not have been created. Our Computing curriculum aims to equip children with the knowledge and skills to use technology confidently, creatively and safely, understanding not just how things occur but why they occur. As many of our children do not have access to a variety of safe and appropriate technology at home, school provides rich opportunities to explore programming, multimedia and data handling, and to see the possibilities technology offers beyond how they use it out of school.

Through our study of Computing, we aim to ensure that all pupils:

- understand and apply the fundamental principles of computer science, including logic, algorithms and data representation,
- analyse problems in computational terms and gain repeated practical experience of writing and debugging programs,
- evaluate and apply information technology, including new or unfamiliar technologies, to solve problems,
- are responsible, competent, confident and creative users of technology who know how to stay safe online.

Implementation

Curriculum structure & sequencing

Computing is taught using the Teach Computing Curriculum, developed by the National Centre for Computing Education. Each year group from Year 1 to Year 6 studies six units, taught in the scheme's suggested order, with every unit made up of six sequenced lessons with clear learning objectives and success criteria. Units revisit and build on prior learning year after year, ensuring National Curriculum aims are met.

Content & concepts

Each year covers four strands: Computing systems and networks; Creating media; Programming (two units per year, progressing from moving a robot in Year 1 to Scratch, text-based programming and physical computing in KS2); and Data and information. Units are mapped to the Teach Computing taxonomy, cross-curricular links and Education for a Connected World. Key vocabulary is discussed in every lesson and revisited as children progress through school.

Enrichment and personal development

Online safety is promoted through Safer Internet Day, police workshops and CEOP ambassador training for staff, with parents kept informed through Wake Up Wednesday and newsletters. An after-school computing club offers further opportunities to code and create.

Assessment and next steps

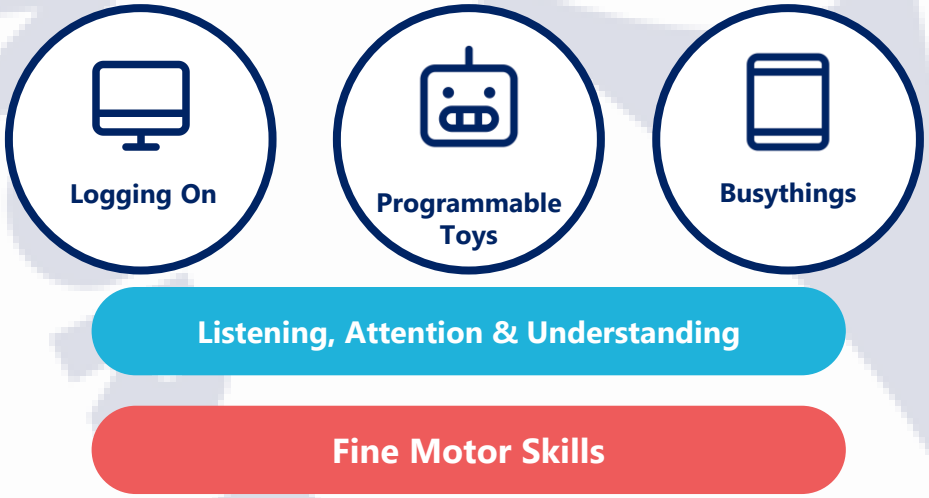
Children produce pieces of work in each unit that independently demonstrate the skills they have been taught. Evidence of endpoints is saved on Seesaw and in pupil drives, and data is put on Insight each term, where it is analysed by the subject leader and next steps identified.

Computing in the Early Years Foundation Stage

Computing is not a specific area of learning in the Early Years Foundation Stage (EYFS); however, children's early computing skills are developed through 'Communication and Language' and 'Physical Development', supported by the non-statutory guidance provided by Development Matters.

At Manor Park, we equip our youngest children to participate in a rapidly changing world which is transformed by technology. Foundation Stage children have access to technology as part of child-initiated activities, as well as a weekly session in the computing suite, where they gain familiarity by logging in using a class username and experimenting with simple software and programming.

Busythings is subscribed to as a school for use both at school and at home to encourage early computing skills, helping children to explore, analyse, exchange and present information in a safe way.



C&L: Listening & Understanding – Development Matters

Children in Reception will be learning to:

- Understand how to listen carefully and why listening is important;
- Ask questions to find out more and to check they understand what has been said to them;
- Learn rhymes, poems and songs.

Physical Development: Fine Motor Skills – Development Matters

Children in Reception will be learning to:

- Develop their small motor skills so that they can use a range of tools competently, safely and confidently;
- Develop the foundations of a handwriting style which is fast, accurate and efficient.

C&L: Listening, Attention & Understanding – ELGs

Children at the end of Reception will:

1. Listen attentively and respond to what they hear with relevant questions, comments and actions;
2. Make comments about what they have heard and ask questions to clarify their understanding;
3. Hold conversation when engaged in back-and-forth exchanges with their teacher and peers.

Physical Development: Fine Motor Skills – Early Learning Goals

Children at the end of Reception will:

1. Hold a pencil effectively in preparation for fluent writing;
2. Use a range of small tools, including scissors, paint brushes and cutlery;
3. Begin to show accuracy and care when drawing.

Inclusion within Computing

We are an inclusive school and as such, do not believe in narrowing the curriculum for any learner. Our curriculum is designed with inclusion of all at heart, and our curriculum intent is therefore the same for all children.

However we are mindful that there are an abundance of factors which need to be considered in order for all learners to be able to access learning according to their individual needs; perhaps none more so than for those learners with Special Educational Needs and Disabilities (SEND).

Therefore, whilst our curriculum intent is the same for all learners; our implementation of the curriculum may well look different for different groups of pupils. Teachers will plan, scaffold, challenge and embed learning through activities which are adapted to meet children's needs – we call this adapted implementation. This is to ensure that our curriculum can be met by all within an inclusive environment, mindful and responsive to children's needs.

We use guidance set out within the NASEN teacher handbook to assist us in amending our implementation within Computing. Examples of this, though not an exhaustive list, can be seen to the right. Note, these are suggestions of what may be implemented but all teachers will amend according to learner need.



Same intent, adapted implementation

- Colour-coded word banks and Widgit symbols to support computing vocabulary.
- Unplugged activities and role play, such as acting out an algorithm, before working on screen.
- Small steps and concrete examples before abstract ideas, e.g. sorting objects before databases.
- Consistent routines, such as using the same device and logging in as soon as pupils sit down.
- Carefully chosen working groups and peer support during project work.

Manor Park Primary School Curriculum Road Map – Computing

Year R

- Logging On
- Programmable Toys
- Busythings

C&L — Listening, Attention & Understanding

PD — Fine Motor Skills

Teach Computing Curriculum – units in teaching order (1–6)

Year 1

1. Technology around us
2. Digital painting
3. Moving a robot
4. Grouping data
5. Digital writing
6. Programming animations

Year 2

1. IT around us
2. Digital photography
3. Robot algorithms
4. Pictograms
5. Digital music
6. Programming quizzes

Year 3

1. Connecting computers
2. Stop-frame animation
3. Sequencing sounds
4. Branching databases
5. Desktop publishing
6. Events & actions

Year 4

1. The Internet
2. Audio production
3. Repetition in shapes
4. Data logging

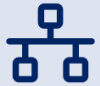
Year 5

1. Systems & searching
2. Video production
3. Selection in physical computing
4. Flat-file databases
5. Vector graphics
6. Selection in quizzes
5. Photo editing
6. Repetition in games

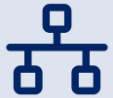
Year 6

1. Communication & collaboration
2. Web page creation
3. Variables in games
4. Spreadsheets
5. 3D modelling
6. Sensing movement

Computing Endpoints

Unit (order)	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
 **1. Computing systems & networks**	**Technology around us** To identify technology, name the main parts of a computer and use a mouse and keyboard, following rules for using technology responsibly.	**IT around us** To recognise the uses and features of information technology in school and beyond, and explain how to use it safely.	**Connecting computers** To identify input and output devices and explain how a computer network shares information between connected devices.	**The Internet** To describe how networks connect to form the internet and how content on the World Wide Web is created, shared and judged for reliability.	**Systems & searching** To explain how computers connect to form systems, and how search engines select and rank results.	**Communication & collaboration** To explain how data is transferred across the internet, and evaluate different ways of communicating and collaborating online.
 **2. Creating media**	**Digital painting** To use freehand, shape and line tools to paint a digital picture, explaining my choices and comparing it with painting on paper.	**Digital photography** To take, improve and edit digital photographs, recognising that photos can be changed.	**Stop-frame animation** To plan, create, review and improve a stop-frame animation, evaluating the impact of adding other media.	**Audio production** To record, edit and combine audio to produce and evaluate a podcast.	**Video production** To plan with a storyboard, then capture and edit an effective video, considering the impact of sharing it.	**Web page creation** To plan and create a web page with a navigation path, considering copyright and linking to content owned by others.
 **3. Programming A**	**Moving a robot** To predict the outcome of commands, and plan and debug a sequence of commands to move a robot.	**Robot algorithms** To design an algorithm, use logical reasoning to predict its outcome, and create and debug my own program.	**Sequencing sounds** To create a Scratch project from a task description, sequencing commands to play sounds and change its appearance.	**Repetition in shapes** To create programs in a text-based language that use count-controlled loops to draw shapes.	**Selection in physical computing** To design and program a physical computing project that uses loops and selection to control a circuit.	**Variables in games** To design, create and evaluate a game that uses variables.
 **4. Data & information**	**Grouping data** To label, count and group objects by their properties, and answer questions about the groups.	**Pictograms** To use tally charts and pictograms to count, compare and present information using a computer.	**Branching databases** To plan and independently create a well-structured branching database to identify objects.	**Data logging** To use a data logger to collect data from sensors over time and use it to answer questions.	**Flat-file databases** To use forms and a flat-file database to group, sort and select data, answering questions with a real-world database.	**Spreadsheets** To build a data set in a spreadsheet, apply formulas and choose suitable ways to present data, including to plan an event.
 **5. Creating media**	**Digital writing** To use a computer to write, adding, removing and changing the look of text, and compare typing with writing on paper.	**Digital music** To experiment with sound and create a musical pattern on a computer for a purpose, reviewing and refining my work.	**Desktop publishing** To combine text and images using desktop publishing, choosing page settings and layouts to suit a purpose.	**Photo editing** To change the composition and colours of digital images and combine images for a purpose, evaluating the changes.	**Vector graphics** To create a vector drawing by combining shapes, using layers and grouping objects to achieve an effect.	**3D modelling** To plan and create my own digital 3D model for a given purpose.
 **6. Programming B**	**Programming animations** To design an animation and use my algorithm to program sprites, joining commands and changing values.	**Programming quizzes** To create a quiz program from a given design and then from my own design, deciding how it can be improved.	**Events & actions in programs** To design and create a maze-based challenge, using events and actions to move a sprite and fixing bugs.	**Repetition in games** To design and create a game that uses count-controlled and infinite loops, including loops that run at the same time.	**Selection in quizzes** To design, create and evaluate a quiz that uses selection to direct the flow of the program.	**Sensing movement** To design and develop a program that uses inputs, outputs, selection and variables on a controllable device.

Computing - Progression of Knowledge & Skills

Strand	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
 **Computing systems & networks**	Identify technology and the main parts of a computer; use a mouse and keyboard; create rules for using technology responsibly.	Recognise the uses and features of IT in school and beyond; explain how IT helps us and how to use it safely.	Explain how digital devices work; identify input and output devices; explain how networks share information and name their physical parts.	Describe how networks connect to form the internet; explain how the World Wide Web shares content; evaluate unreliable content.	Explain that computers connect to form systems; describe how search engines select and rank results, and why order matters.	Explain internet addresses and how data is transferred; evaluate ways of working together and communicating online.
 **Creating media**	Paint digital pictures using freehand, shape and line tools; write, edit and format text on a computer, comparing with paper.	Take and improve digital photographs and recognise that photos can be changed; create musical patterns on a computer.	Create stop-frame animations; combine text and images in desktop publishing, choosing layouts for a purpose.	Record and edit audio to produce a podcast; edit the composition and colour of images and combine them.	Storyboard, film and edit a video; create vector drawings using shapes, layers and grouping.	Plan and create web pages with navigation, considering copyright; create and modify 3D models for a purpose.
 **Programming**	Predict what commands do and combine them into sequences to move a robot; join commands and change values to program animations.	Describe instructions as sequences; use logical reasoning to predict outcomes; design algorithms and debug programs, including quizzes.	Use Scratch to sequence commands and sounds; use events and actions to move sprites; identify and fix bugs.	Use count-controlled loops in a text-based language to draw shapes; use count-controlled and infinite loops in games.	Use conditions and selection to control a physical computing project and to direct the flow of a quiz.	Define and use variables in games; use inputs, outputs, selection and variables on a controllable device.
 **Data & information**	Label, count and group objects by their properties, and answer questions about groups of objects.	Use tally charts and pictograms to count, compare and present information on a computer.	Create yes/no questions and structured branching databases to identify objects.	Use data loggers to collect data points from sensors over time, and analyse data to answer questions.	Use forms and flat-file databases to group, sort, select and compare data to answer questions.	Build data sets in spreadsheets, use formulas to calculate data and choose suitable ways to present it.

Assessment within Computing

We place great emphasis on the importance of assessing children's knowledge, understanding and skills within Computing.

When assessing Computing, it is first essential to clearly articulate two important areas:

1. The specific endpoint for the unit being delivered,
2. The substantive and disciplinary knowledge to be taught to reach this endpoint.

At Manor Park Primary School, we have clearly mapped out the endpoints for all the Computing units to be delivered, before specifying what substantive and disciplinary knowledge is to be taught within each unit to reach each endpoint. It is this knowledge and understanding that we assess children upon, believing accurate assessment can only be a reflection of what is taught. Endpoints are made up primarily of the knowledge and skills of programming, using technology and staying safe online taught across a unit; with the pieces of work children produce independently (saved on Seesaw and in pupil drives) providing evidence of these endpoints.

Making judgements – formative assessment

When delivering lessons, teachers record notes, comments and reflections pertinent to the formative assessment of Computing to inform their future planning. This feedback is shared at the start of the following lesson, so that children recap prior learning, address misconceptions and build upon what has been taught.

Making judgements – summative assessment

With the unit endpoint in mind, teachers form a summative assessment for each child within a particular unit. This will be either working towards / working at / working above the expected standard.

We define the expected standard by listing the essential substantive and disciplinary knowledge children should know, and what would classify a pupil as working below or above this. Teachers record this judgement on a single page at the end of each unit, drawing on their formative assessments and evidence on Seesaw and on pupil drives.