ASPIRE
ACADEMY
BEXLEY

Computing Scheme of Work 2021/22 (Cycle B)

- Computing is taught in line with the National Curriculum every term and is based around the 'Purple Mash' scheme of work for mixed year group classes.
- Units cover coding and computational thinking, spreadsheets, internet & email, art & design, music, databases & graphing, writing and presenting and communication & networks.
- In addition to the lessons outlined below, there will be one lesson on internet safety each half term. This may be covered within the PSHE curriculum.
- Additional Internet Safety lessons will be taught responsively, according to the needs of each class.
- Computing skills are also taught discreetly and in cross curricular lessons. Pupils have access to a laptop, chrome book and/or iPad throughout the day.
- Pupils can choose to attend computing curriculum clubs on a Friday afternoon, which will include film making and coding, to further develop their interests and skill set. This will also provide opportunities to become familiar with other coding software, including scratch and hour of code.
- The Aspire curriculum is a progression of skills and therefore pupils are taught according to their ability and past experiences not necessarily their year group.

Computing Cycle A (Covid 19 Pandemic) 2020/21

Term	Topic	KS1 (Y1/2)	LKS2 (Y3/4)	UKS2 (Y5/6)*
Autumn 1	Bounce Back	*Please see above	*Please see above	*Please see above
Autumn 2	The UK	**Digital Literacy & Information Technology** Internet and email	**Computer Science** Coding and computational thinking	**Computer Science** Coding and computational thinking
Spring 1	Romans	**Computer Science** Coding and computational thinking **Digital Literacy** Communication and networks	**Information Technology** Spreadsheets Writing and presenting	**Information Technology** Spreadsheets
Spring 2	Local History: Eltham Palace	**Information Technology** Art and design	**Digital Literacy** Internet and email	**Information Technology** Databases and Graphing
Summer 1	The Rainforest	**Information Technology** Coding and computational thinking Spreadsheets	**Information Technology** Databases and Graphing Communication and networks	**Computer Science** Art and design
Summer 2	Water	**Information Technology** Coding and computational thinking	**Information Technology** Databases and Graphing	**Information Technology** Writing and presenting

* In UKS2 Digital Literacy objectives are covered discreetly throughout the year, and will not appear as a named topic.

Computing Cycle A (Non-Covid) 2022/23

Term	Topic	KS1 (Y1/2)	LKS2 (Y3/4)	UKS2 (Y5/6)*
Autumn 1	Ancient Greece	**Digital Literacy & Information Technology** Internet and email Purple Mash units 1.1, 2.5	**Computer Science** Coding and computational thinking *See table below for breakdown (Cycle A)*	**Computer Science** Coding and computational thinking *See table below for breakdown (Cycle A)*
Autumn 2	Around the world	**Computer Science & Digital Literacy** Coding and computational thinking Communication and networks Purple Mash units 1.4, 1.9	**Information Technology** Spreadsheets Purple Mash unit 3.3	**Information Technology** Spreadsheets Purple Mash units 5.3
Spring 1	Arctic/Antarctic	**Information Technology** Coding and computational thinking Art and design Purple Mash units 1.2, .2.6	**Information Technology** Writing and presenting Purple Mash units 3.4	**Information Technology** Databases and Graphing Purple Mash units 5.4
Spring 2	Looking after our world	**Information Technology** Art and design Spreadsheets Purple Mash units 2.6 cont. ,1.8	**Digital Literacy** Internet and email Purple Mash units 3.5	**Computer Science** Art and design Purple Mash units 5.5
Summer 1	Vikings	**Information Technology** Coding and computational thinking Purple Mash unit 1.7	**Information Technology** Databases and Graphing Communication and networks Purple Mash units 3.6, 3.7	**Computer Science** Art and design Purple Mash units 5.6
Summer 2	Local History: WW2	**Information Technology** Coding and computational thinking Purple Mash unit 2.1	**Information Technology** Communication and networks Databases and Graphing Purple Mash units 3.7 cont., 3.8	**Information Technology** Writing and presenting Purple Mash units 5.7

* In UKS2 Digital Literacy objectives are covered discreetly throughout the year, and will not appear as a named topic.

Computing Cycle B 2021/22

Term	Topic	KS1 (Y1/2)	LKS2 (Y3/4)	UKS2 (Y5/6)*
Autumn 1	Shang Dynasty	**Digital Literacy & Computer Science** Internet and email Coding and computational thinking Purple Mash units 1.1, 1.5.	**Computer Science** Coding and computational thinking *See table below for breakdown (Cycle B)*	**Computer Science** Coding and computational thinking *See table below for breakdown (Cycle B)*
Autumn 2	North and South America	**Information Technology** Databases and Graphing Purple Mash Unit 2.4	**Information Technology** Spreadsheets Purple Mash Unit 4.3	**Information Technology** Spreadsheets Purple Mash Unit 6.3
Spring 1	Local History: Hall Place Victorians	**Digital Literacy & Information Technology** Internet and email Music Purple Mash Unit 2.2, 2.7.	**Information Technology** Writing and presenting Purple mash Unit 4.4	**Computer Science & Digital Literacy** Writing and presenting Purple Mash Unit 6.4
Spring 2	Mountains	**Information Technology** Art and design Purple mash Unit 1.6	**Computer Science** Coding and computational thinking Purple Mash unit 4.5	**Computer Science & Information Technology** Coding and computational thinking Purple Mash Unit 6.5
Summer 1	Anglo Saxons	**Information Technology** Spreadsheets Purple Mash Unit 2.3	**Information Technology** Art and design Purple Mash unit 4.6	**Computer Science** Communication and networks Purple Mash Unit 6.6
Summer 2	Safari Comparison with life in another country	**Information Technology** Databases and Graphing Writing and presenting Purple Mash Unit 1.3, 2.8	**Computer Science & Information Technology** Internet and email Communication and networks Purple mash Unit 4.7, 4.8	**Information Technology** Writing and presenting Purple Mash Unit 6.7

* In UKS2 Digital Literacy objectives are covered discreetly throughout the year, and will not appear as a named topic.

Computing Curriculum Overview

Please find detailed planning and support materials on Purple Mash.

Theme Key

Coding and computational thinking	Spreadsheets	Internet and email	Art and design	Music	Databases and Graphing	Writing and presenting	Communication and networks

<table>
<tr><th></th><th></th><th colspan="2">Autumn 1</th><th colspan="2">Autumn 2</th><th colspan="2">Spring 1</th><th colspan="2">Spring 2</th><th colspan="2">Summer 1</th><th colspan="2">Summer 2</th></tr>
<tr><td rowspan="2">KS1 Yr 1/2</td><td>Cycle A</td><td>Unit 1.1
Online Safety & Exploring Purple Mash
4 Sessions</td><td>Unit 2.5
Effective Searching
3 sessions</td><td>Unit 1.4
Lego Builders (2DIY)
3 sessions</td><td>Unit 1.9
Technology outside school
2 Sessions</td><td>Unit 1.2
Grouping & Sorting
(2DIY)
2 sessions</td><td colspan="2">Unit 2.6
Creating Pictures
(2PaintAPicture)
5 sessions</td><td>Unit 1.8
Spreadsheets
(2Calculate)
3 sessions</td><td colspan="2">Unit 1.7
Coding
(2Code)
6 sessions</td><td colspan="2">Unit 2.1
Coding
(2Code)
5 sessions</td></tr>
<tr><td>Cycle B</td><td>Unit 1.1
Online Safety & Exploring Purple Mash
4 Sessions</td><td>Unit 1.5 Maze Explorers
(2Go)
4 sessions</td><td colspan="2">Unit 2.4 Questioning
(2Question, 2 Investigate)
5 sessions</td><td>Unit 2.2 Online Safety
3 sessions</td><td>Unit 2.7 Making Music
(2sequence)
3 sessions</td><td colspan="2">Unit 1.6 Animated Story Books
(2 Create A Story)
5 sessions</td><td>Unit 2.3 Spreadsheets
(2 Calculate)
4 sessions</td><td colspan="2">Unit 1.3 Pictograms
(2 Count)
3 sessions</td><td>Unit 2.8 Presenting Ideas
4 sessions</td></tr>
<tr><td>LKS2 Yr 3/4</td><td>Cycle A Covid</td><td colspan="2">*Please see above.</td><td colspan="2">Coding
See table below for breakdown
(2Code)
6 sessions</td><td>Unit 3.3
Spreadsheets
(2Calculate)
3 sessions</td><td>Unit 3.4
Touch Typing
(2 type)
4 Sessions</td><td colspan="2">Unit 3.5
Email
(including email safety)</td><td>Unit 3.6 Branching Databases
(2Question)
4 sessions</td><td colspan="2">Unit 3.7 Simulations
(2simulate, 2 Publish)
3 sessions)</td><td>Unit 3.8 Graphing
(2Graph
3 sessions</td></tr>
</table>

	Cycle A	Coding *See table below for breakdown* (2Code) 6 sessions	Unit 3.3 Spreadsheets (2Calculate) 3 sessions	Unit 3.4 Touch Typing (2 type) 4 Session	Unit 3.5 Email (including email safety)	Unit 3.6 Branching Databases (2Question) 4 sessions	Unit 3.7 Simulations (2simulate, 2 Publish) 3 sessions)	Unit 3.8 Graphing (2Graph 3 sessions
	Cycle B	Coding *See table below for breakdown* (2Code) 6 sessions	Unit 4.3 Spreadsheets (2Calculate) 6 sessions	Unit 4.4 Writing for different audiences (2Email, 2Connect, 2DIY) 5 sessions	Unit 4.5 Logo 4 sessions	Unit 4.6 Animation (2Animate) 3 Sessions	Unit 4.7 Effective Search 3 sessions	Unit 4.8 Hardware Investigators 2 sessions
UKS 2 Yr 5/6	**Cycle A Covid**	*Please see above.	Coding *See table below for breakdown* 6 lessons	Unit 5.3 Spreadsheets (2Calculate) 5 sessions	Unit 5.4 Databases (2Question, 2investigate) 4 sessions	Unit 5.5 Game Creator (2DIY 3D) 5 sessions	Unit 5.6 3D Modelling (2Design and Make) 4 sessions	Unit 5.7 Concept Maps (2Connect) 4 sessions
	Cycle A	Coding *See table below for breakdown* 6 lessons	Unit 5.3 Spreadsheets (2Calculate) 5 sessions	Unit 5.4 Databases (2Question, 2investigate) 4 sessions	Unit 5.5 Game Creator (2DIY 3D) 5 sessions	Unit 5.6 3D Modelling (2Design and Make) 4 sessions	Unit 5.7 Concept Maps (2Connect) 4 sessions	

	Cycle B	Coding *See table below for breakdown* 6 sessions	Unit 6.3 Spreadsheets (2 Calculate) 5 sessions	Unit 6.4 Blogging (2Blog) 5 sessions	Unit 6.5 Text Adventures (2Code, 2Connect) 5 sessions	Unit 6.6 Networks 3 sessions	Unit 6.7 Quizzing (2Quiz, 2DIY, Text Toolkit, 2Investigate) 6 sessions

LKS2 (Yr. ¾): Autumn Term Coding Breakdown						
Cycle A	**Review previous coding – Year 3, Lesson 1**	**Simulating a physical system – Year 3, Lesson 2**	**Making a timer – Year 4, Lesson 4**	**Debugging – Year 3, Lesson 6**	**Making a control simulation – Year 4, Lesson 5**	**Decomposition and Abstraction – Year 4, Lesson 6**
Cycle B	**Review previous coding, Y4, lesson 1**	**Introducing 'if' statements – Year 3, Lesson 4**	**'if/else' statements – Year 4, Lesson 2**	**Repetition – Year 3, Lesson 5**	**Repeat until - Year 4, Lesson 3**	**Variables – Year 3, Lesson 4**

UKS2 (Yr. ⅚): Autumn Term Coding Breakdown						
Cycle A	**Review previous coding – Year 5, Lesson 1**	**Simulating a physical system – Year 5, Lesson 2**	**Creating a game with a score and timer – Year 5 Lessons 4 and 5**		**The Launch Command – Year 5 Lesson 6**	**Using User Input – Year 6, Lesson 4**
Cycle B	**Designing and writing a more complex program – Year 6 Lessons 1 and 2**		**Introducing text variables – Year 5 Lesson 3**	**Introducing Functions – Year 6 Lesson 3**	**Flowcharts and control simulations – Year 6, Lesson 5**	**Text Adventure – Year 6 Lesson 6**
There is an optional unit 6.8 – Understanding Binary that can be used in addition to the above units. It is a four week unit.						

Computing Progression of Skills

		KS1 (Y1/2)	LKS2 (Y3/4)	UKS2 (Y5/6)
Computer Science	**NC Statement**	**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.	**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.	**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.
	Outcome	1. Children understand that an algorithm is a set of instructions used to solve a problem or achieve an objective. They know that an algorithm written for a computer is called a program. 2. Children can explain that an algorithm is a set of instructions to complete a task. When designing simple programs, children show an awareness of the need to be precise with their algorithms so that they can be successfully converted into code.	3. Children can turn a simple real-life situation into an algorithm for a program by deconstructing it into manageable parts. Their design shows that they are thinking of the desired task and how this translates into code. Children can identify an error within their program that prevents it following the desired algorithm and then fix it. 4. When turning a real life situation into an algorithm, the children's design shows that they are thinking of the required task and how to accomplish this in code using coding structures for selection and repetition. Children make more intuitive attempts to debug their own programs.	5. Children may attempt to turn more complex real-life situations into algorithms for a program by deconstructing it into manageable parts. Children are able to test and debug their programs as they go and can use logical methods to identify the approximate cause of any bug but may need some support identifying the specific line of code. 6. Children are able to turn a more complex programming task into an algorithm by identifying the important aspects of the task (abstraction) and then decomposing them in a logical way using their knowledge of possible coding structures and applying skills from previous programs. Children test and debug their program as they go and use logical methods to identify the cause of bugs, demonstrating a systematic approach to try to identify a particular line of code causing a problem.

	NC Statement	**Pupils should be taught to...** Create and debug simple programs.	**Pupils should be taught to…** Use sequence, selection and repetition in programs; work with variables and various forms of input and output.	**Pupils should be taught to…** Use sequence, selection and repetition in programs; work with variables and various forms of input and output.
	Outcome	1. Children can work out what is wrong with a simple algorithm when the steps are out of order, e.g. The Wrong Sandwich in Purple Mash and can write their own simple algorithm, e.g. Colouring in a Bird activity. Children know that an unexpected outcome is due to the code they have created and can make logical attempts to fix the code, e.g. Bubbles activity in 2Code. 2. Children can create a simple program that achieves a specific purpose. They can also identify and correct some errors, e.g. Debug Challenges: Chimp. Children's program designs display a growing awareness of the need for logical, programmable steps.	3. Children demonstrate the ability to design and code a program that follows a simple sequence. They experiment with timers to achieve repetition effects in their programs. Children are beginning to understand the difference in the effect of using a timer command rather than a repeat command when creating repetition effects. Children understand how variables can be used to store information while a program is executing. 4. Children's use of timers to achieve repetition effects are becoming more logical and are integrated into their program designs. They understand 'if statements' for selection and attempt to combine these with other coding structures including variables to achieve the effects that they design in their programs. As well as understanding how variables can be used to store information while a program is executing, they are able to use and manipulate the value of variables. Children can make use of user inputs and outputs such as 'print to screen'. e.g. 2Code.	5. Children can translate algorithms that include sequence, selection and repetition into code with increasing ease and their own designs show that they are thinking of how to accomplish the set task in code utilising such structures. They are combining sequence, selection and repetition with other coding structures to achieve their algorithm design. 6. Children translate algorithms that include sequence, selection and repetition into code and their own designs show that they are thinking of how to accomplish the set task in code utilising such structures, including nesting structures within each other. Coding displays an improving understanding of variables in coding, outputs such as sound and movement, inputs from the user of the program such as button clicks and the value of functions.

	NC Statement	**Pupils should be taught to...** Use logical reasoning to predict the behaviour of simple programs.	**Pupils should be taught to…** Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs.	**Pupils should be taught to…** Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs.
	Outcome	1. 3When looking at a program, children can read code one line at a time and make good attempts to envision the bigger picture of the overall effect of the program. Children can, for example, interpret where the turtle in 2Go challenges will end up at the end of the program. 2. Children can identify the parts of a program that respond to specific events and initiate specific actions. For example, they can write a cause and effect sentence of what will happen in a program.	3. Children's designs for their programs show that they are thinking of the structure of a program in logical, achievable steps and absorbing some new knowledge of coding structures. For example, 'if' statements, repetition and variables. They make good attempts to 'step through' more complex code in order to identify errors in algorithms and can correct this. e.g. traffic light algorithm in 2Code. In programs such as Logo, they can 'read' programs with several steps and predict the outcome accurately. 4. Children's designs for their programs show that they are thinking of the structure of a program in logical, achievable steps and absorbing some new knowledge of coding structures. For example, 'if' statements, repetition and variables. They can trace code and use step-through methods to identify errors in code and make logical attempts to correct this. e.g. traffic light algorithm in 2Code. In programs such as Logo, they can 'read' programs with several steps and predict the outcome accurately.	5. When children code, they are beginning to think about their code structure in terms of the ability to debug and interpret the code later, e.g. the use of tabs to organise code and the naming of variables. 6. Children are able to interpret a program in parts and can make logical attempts to put the separate parts of a complex algorithm together to explain the program as a whole.
	NC Statement		**Pupils should be taught to…** Understand computer networks, including the internet; how they can	**Pupils should be taught to…** 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.	provide multiple services, such as the World Wide Web, and the opportunities they offer for communication and collaboration.
	Outcome		3. Children can list a range of ways that the internet can be used to provide different methods of communication. They can use some of these methods of communication, e.g. being able to open, respond to and attach files to emails using 2Email. They can describe appropriate email conventions when communicating in this way. 4. Children recognise the main component parts of hardware which allow computers to join and form a network. Their ability to understand the online safety implications associated with the ways the internet can be used to provide different methods of communication is improving.	5. Children understand the value of computer networks but are also aware of the main dangers. They recognise what personal information is and can explain how this can be kept safe. Children can select the most appropriate form of online communications contingent on audience and digital content, e.g. 2Blog, 2Email, Display Boards. 6. Children understand and can explain in some depth the difference between the internet and the World Wide Web. Children know what a WAN and LAN are and can describe how they access the internet in school.
Information Technology	**NC Statement**	**Pupils should be taught to…** Use technology purposefully to create, organise, store, manipulate and retrieve digital content.	**Pupils should be taught to…** Use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content.	**Pupils should be taught to…** Use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content.
	Outcome	1. Children are able to sort, collate, edit and store simple digital content e.g. children can name, save and retrieve their work and follow simple instructions to access online resources, use Purple Mash 2Quiz example (sorting shapes), 2Code design mode (manipulating backgrounds) or using pictogram software such	3. Children can carry out simple searches to retrieve digital content. They understand that to do this, they are connecting to the internet and using a search engine such as Purple Mash search or internet-wide search engines. 4. Children understand the function, features and layout of a search engine. They can	5. Children search with greater complexity for digital content when using a search engine. They are able to explain in some detail how credible a webpage is and the information it contains. 6. Children readily apply filters when searching for digital content. They are able to explain in detail how credible a

		as 2Count. 2. Children demonstrate an ability to organise data using, for example, a database such as 2Investigate and can retrieve specific data for conducting simple searches. Children are able to edit more complex digital data such as music compositions within 2Sequence. Children are confident when creating, naming, saving and retrieving content. Children use a range of media in their digital content including photos, text and sound.	appraise selected webpages for credibility and information at a basic level.	webpage is and the information it contains. They compare a range of digital content sources and are able to rate them in terms of content quality and accuracy. Children use critical thinking skills in everyday use of online communication.
	NC Statement		**Pupils should be taught to…** 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.	**Pupils should be taught to…** 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.
	Outcome		3. Children can collect, analyse, evaluate and present data and information using a selection of software, e.g. using a branching database (2Question), using software such as 2Graph. Children can consider what software is most appropriate for a given task. They can create purposeful content to attach to emails, e.g. 2Respond. 4. Children are able to make improvements to digital solutions based on feedback. Children make informed software choices when presenting	5. Children are able to make appropriate improvements to digital solutions based on feedback received and can confidently comment on the success of the solution. e.g. creating their own program to meet a design brief using 2Code. They objectively review solutions from others. Children are able to collaboratively create content and solutions using digital features within software such as collaborative mode. They are able to use several ways of sharing digital

			information and data. They create linked content using a range of software such as 2Connect and 2Publish+. Children share digital content within their community, i.e. using Virtual Display Boards.	content, i.e. 2Blog, Display Boards and 2Email. 6. Children make clear connections to the audience when designing and creating digital content. The children design and create their own blogs to become a content creator on the internet, e.g. 2Blog. They are able to use criteria to evaluate the quality of digital solutions and are able to identify improvements, making some refinements.
Digital Literacy	**NC Statement**	**Pupils should be taught to…** Recognise common uses of information technology beyond school.	**Pupils should be taught to…** Use technology safely, respectfully and responsibly; recognise acceptable/ unacceptable behaviour; identify a range of ways to report concern about content and contact.	**Pupils should be taught to…** Use technology safely, respectfully and responsibly; recognise acceptable/ unacceptable behaviour; identify a range of ways to report concern about content and contact.
	Outcome	1. Children understand what is meant by technology and can identify a variety of examples both in and out of school. They can make a distinction between objects that use modern technology and those that do not e.g. a microwave vs. a chair. 2. Children can effectively retrieve relevant, purposeful digital content using a search engine. They can apply their learning of effective searching beyond the classroom. They can share this knowledge, e.g. 2Publish example template. Children make links between technology they see around them, coding and multimedia work they do in	3. Children demonstrate the importance of having a secure password and not sharing this with anyone else. Furthermore, children can explain the negative implications of failure to keep passwords safe and secure. They understand the importance of staying safe and the importance of their conduct when using familiar communication tools such as 2Email in Purple Mash. They know more than one way to report unacceptable content and contact. 4. Children can explore key concepts relating to online safety using concept mapping such as 2Connect. They can	5. Children have a secure knowledge of common online safety rules and can apply this by demonstrating the safe and respectful use of a few different technologies and online services. Children implicitly relate appropriate online behaviour to their right to personal privacy and mental wellbeing of themselves and others. 6. Children demonstrate the safe and respectful use of a range of different technologies and online services. They identify more discreet inappropriate behaviours through developing critical thinking, e.g. 2Respond activities. They recognise the

		school e.g. animations, interactive code and programs.	help others to understand the importance of online safety. Children know a range of ways of reporting inappropriate content and contact.	value in preserving their privacy when online for their own and other people's safety.
	NC Statement	**Pupils should be taught to…** 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.		
	Outcome	1. Children understand the importance of keeping information, such as their usernames and passwords, private and actively demonstrate this in lessons. Children take ownership of their work and save this in their own private space such as their My Work folder on Purple Mash. 2. Children know the implications of inappropriate online searches. Children begin to understand how things are shared electronically such as posting work to the Purple Mash display board. They develop an understanding of using email safely by using 2Respond activities on Purple Mash and know ways of reporting inappropriate behaviours and content to a trusted adult.		