Y7 Computing

Subject	Computing
Term	Cycle 1
Duration (approx.)	6+ lessons
Module	Kodu Game Programming

Skills and concepts to be developed and assessed (linking to identified AOs)

- Planning a computer game.
- Creating algorithm (step by step instructions) for how the code in the game will work.
- Kodu game programming skills, for example:
- *Creating worlds*
- *Adding behaviours (sequencing)*
- *Using clones and creatables*
- *Using pages or IF tests (selection)*
- *Using scoring, timers, health, power-ups*
- Fixing code bugs to make game work.

Factual knowledge to be taught and assessed (including subject specific vocabulary).

- What algorithms are and why they are important in computer programs.
- What *sequencing* means in computer programming.
- What *selection* means in computer programming.

Formative Assessment/key piece of work prior to end of unit:

- Creation of game plan and pseudo-code to explain how code will work.

Summative Assessment

- The final plan.
- Finished Kodu game.

Building Retention: What prior learning must be built upon/revisited and how will it be assessed?

Pupils will have varying experiences of programming and of computer games. The purpose of this unit is to introduce some key programming concepts and link to an understanding of how games work. The purpose of an algorithm can be linked to everyday activities.

Spelling-Punctuation-Grammar How will you promote high standards within this module?

Encourage use of spell checker in software as well as proof-reading. Encourage this as part of the process of bug fixing.

Link forward: where next for the learning?

Programming terminology will be revisited in the Small Basic and Microbit units; links will be made between the same terms but their use to create a different type of product.

Y7 Computing

Subject	Computing
Term	Cycle 2
Duration (approx.)	6 weeks
Module	Small Basic

Skills and concepts to be developed and assessed (linking to identified AOs)

- How to use a text based programming language.
- Introduction of key programming terms and how to use these:
- *Understanding what a variable is in programming and how to create and use variables.*
- *How to sequence in programming to make instructions run in order.*
- *How to use selection in programming to make code have different possible outcomes using IF and IF...ELSE.*
- *How to use iteration in programming to make code repeat using FOR and WHILE loops.*
- How to accept input from the user in a program and use this.
- Using debugging skills to check for errors in code and correct these.
- Using code commenting to show understanding of what is created.

Factual knowledge to be taught and assessed (including subject specific vocabulary).

The key programming terms: sequence, selection, iteration and variable.

Formative Assessment/key piece of work prior to end of unit:

Check of 'selection' coding to check mid-point progress and ensure syntax used correctly and commenting of good quality.

Summative Assessment.

Final coded program using range of skills learnt.

Building Retention: What prior learning must be built upon/revisited and how will it be assessed?

Link back to Kodu unit just completed—seeing how the key programming terms are the same when applied to all languages.

Spelling-Punctuation-Grammar How will you promote high standards within this module?

Correct syntax needed for coding to work.

Link forward: where next for the learning?

Direct link to Python programming unit in Year 8 where key terms will be revisited in an industry standard language.
Links to Microbit later in Year 7 where block coding will be used to carry out sequence, selection, iteration and creation of variables.

Y7 Computing

Subject	Computing
Term	Cycle 2
Duration (approx.)	6 lessons
Module	Computing Basics

Skills and concepts to be developed and assessed (linking to identified AOs)

- Converting binary to decimal and decimal to binary.
- Using software to present research.

Factual knowledge to be taught and assessed (including subject specific vocabulary).

Technical knowledge of computing architecture to be covered:

- Input and output devices
- Main parts of a computer system
- Understanding binary code
- Connecting to the internet
- What networks are and how they work
- Possible health and safety impacts of using computers

Formative Assessment/key piece of work prior to end of unit:

Lesson research tasks to be recorded.

Summative Assessment.

- Completed research tasks.
- Final Assessment quiz to check understanding of the key theory learnt.

Building Retention: What prior learning must be built upon/revisited and how will it be assessed?

Pupils will all have experiences of using computers, but will have a varying understanding of the terminology of the equipment they use and its purpose. This unit will assume little prior knowledge.

Spelling-Punctuation-Grammar How will you promote high standards within this module?

Encourage use of spell checker software as well as proof reading. Use guides and displays to help with spelling of key terminology.

Link forward: where next for the learning?

Future links can be made back to the essentials of how a computer operates and binary code in the final programming unit.

Subject	Computing
Term	Cycle 3
Duration (approx.)	6 lessons
Module	Spreadsheet Modelling

Skills and concepts to be developed and assessed (linking to identified AOs)

- How to enter data into a spreadsheet.
- How to use formatting tools in a spreadsheet to make it easier to read.
- How to use formulae and functions to carry out calculations.
- How to test formulae are working accurately.
- How to use data in a spreadsheet to create charts.

Factual knowledge to be taught and assessed (including subject specific vocabulary).

- What spreadsheets are used for and why they are useful.
- What spreadsheet modelling is and how this can be used to solve problems.

Formative Assessment/key piece of work prior to end of unit:

Development of spreadsheet to contain data and appropriate formatting.

Summative Assessment.

- The final spreadsheet.
- Final modelling task to check pupils' understanding of how to use spreadsheets to answer questions.

Building Retention: What prior learning must be built upon/revisited and how will it be assessed?

It is unlikely pupils will have used spreadsheets before. Links can be made, however, with the first two units (Kodu Game programming and Small Basic) and the importance of testing formulae.

Spelling-Punctuation-Grammar How will you promote high standards within this module?

Encourage use of spell checker in software as well as proof-reading. Encourage this as part of the process of testing the accuracy of the spreadsheet.

Link forward: where next for the learning?

Links can be made to the final unit in the importance of breaking instructions down into small tasks and testing thoroughly.

Y7 Computing

Subject	Computing
Term	Cycle 3
Duration (approx.)	6+ lessons
Module	BBC Microbit Programming

Skills and concepts to be developed and assessed (linking to identified AOs)

- How to use Microsoft block programming to create code for the Microbit.
- How to write and test code in the editor.
- How to compile, download and run code on the Microbit.
- How to use a range of programming techniques.
- Using algorithms to represent code as 'pseudocode'.

Factual knowledge to be taught and assessed (including subject specific vocabulary).

Understanding key programming terminology—sequence, selection, iteration, variables.

Formative Assessment/key piece of work prior to end of unit:

Code created using lesson tutorials.

Summative Assessment.

- Development of algorithm to give step by step instructions for a piece of code.
- Final range of code created using development of skills.

Building Retention: What prior learning must be built upon/revisited and how will it be assessed?

All pupils have completed Unit 7.1 Kodu Game Programming and Unit 7.2 Small Basic. Links can be made to these units to explain the code created and the algorithms needed to break a coding problem down into steps.

Spelling-Punctuation-Grammar How will you promote high standards within this module?

Encourage use of spell checker in software for written work. Introduce testing code (including spelling errors) as an important part of the programming process.

Link forward: where next for the learning?

Further coding units in Years 8 and 9 can be linked back to the experiences of using the Microbit. When pupils advance their writing of text based code in Year 8, they will see similar patterns to what they have created using the block editor.