Unit 18: Computational Thinking

Level: **1 and 2**

Unit type: **Mandatory**

Guided learning hours: **30**

Assessment type: **Internal**

Unit introduction

Computers influence much of the world around us and are used to solve a range of problems, including complex problems such as generating the visual effects in films and landing the Curiosity Rover (unmanned robot) on Mars. Computational thinking enables us to solve these problems using mathematics and logic.

Therefore, computational thinking skills are required extensively in any IT/computing career. This unit provides a coherent and formalised introduction to computational thinking. Employability skills, such as communication, numeracy and teamwork, are also important in the workplace. Some of these skills occur naturally within other units in the BTEC in Information and Creative Technology.

In this unit you will understand some of the mathematical methods, such as Boolean operations and functions, that underpin computational thinking skills used to solve problems. There are three types of computational thinking skills, specifically:

1. Logical thinking is about using readily available information to solve problems without making assumptions or taking short cuts and to check the validity of the result. The number puzzle Sudoku is an example of a problem which can be solved this way.
2. Algorithmic thinking involves using a predefined set of instructions to solve problems. For example, in strategy games like noughts and crosses or chess, players will use set moves (instructions) to try to gain advantage over their opponent and win.
3. Optimal thinking involves refining one of the many possible solutions to a problem to make it more efficient, i.e. the quickest and/or most effective method to obtain the desired result. For example, completing the Towers of Hanoi or a Rubik's Cube puzzle using the fewest moves.

In this unit you will use computational thinking (logical, algorithmic and optimal), underpinned by mathematical methods, to solve a problem.

This unit supports all of the optional specialist units in the Pearson BTEC Level 1/ Level 2 Firsts in Information and Creative Technology.

Learning aims

In this unit you will:

A understand mathematical methods for calculations and their uses in computational thinking

B understand mathematical methods for functions and Boolean operations and their uses in computational thinking

C apply computational thinking to solve a simple problem.

Learning aims and unit content

What needs to be learnt
Learning aim A: Understand mathematical methods for calculations and their uses in computational thinking
Types of numbers ● Natural numbers (1, 2, 3 …). ● Integers (… –2, –1, 0, 1, 2, …). ● Rational numbers (integers, fractions, decimals). **Number systems** ● Base 10 (decimal). ● Base 2 (binary). ● Base 16 (hexadecimal). **Using basic calculations** ● Brackets and the hierarchy of operations (BIDMAS). ● Rounding, e.g. specified number of decimal places, whole number, degree of accuracy. ● Errors, e.g. generated by rounding and truncation due to size of memory locations, division by zero, underflow, overflow. ● Powers, e.g. use of the laws of indices, writing numbers using powers, expressing numbers using standard form ● Convert between number systems, e.g. binary to hexadecimal, using positive numbers up to a maximum of 6 bits in size (1 to 63 in decimal). ● One and two's complement notation representing signed integers up to 4 bits in size. **Example uses of number systems in computers, e.g.:** ● computer memory representing integers and rational numbers, including: ○ bits, bytes and word length ○ two's complement ○ registers, e.g. 8-bit, 16-bit, 32-bit. ● character encoding schemes, including: ○ ASCII (American Standard Code for Information Interchange) – 7 bits ○ UTF8 (UCS Transformation Format) – 8 bits. ● colour models, including: ○ RGB (Red Green Blue) ○ CMYK (Cyan Magenta Yellow Key-Black) ○ hexadecimal.

What needs to be learnt

Learning aim B: Understand mathematical methods for functions and Boolean operations and their uses in computational thinking

Defining a function

- Inputs (domain).
- Outputs (range).
- Types of functions, including:
  - linear, e.g. $y = mx + c$

    (gradient of a straight line is defined by 'm' and the y-axis intercept is defined by 'c')
  - quadratic, e.g. $y = ax^2 + bx + c$

Graphical representation of functions

Using software, e.g. spreadsheet or graphics package, to represent linear and quadratic functions.

Example uses of functions in computers, e.g.:

- graphics, e.g. positioning objects on screen or obtaining the function from a straight line graph.
- programming, e.g. sorting numbers, calculations, randomising numbers.

Boolean algebra

Know that Boolean algebra is an alternative of elementary algebra of numbers differing in its values, operations and laws. Boolean algebra is the algebra of truth values 0 and 1.

- Operations – AND; OR; NOT; XOR.
- Truth tables.
- Logic gates.

Example uses of Boolean operations in computers, e.g.:

- software programs
- design of circuits and chips in technology systems
- cryptography.

What needs to be learnt

Learning aim C: Apply computational thinking to solve a simple problem

Representation of a problem

Understand that mathematical methods can be used to represent a sub-problem or solution to a problem, e.g.:

- calculations
- truth tables for logical (Boolean) operations: AND, OR, NOT, XOR
- simple functions, e.g. linear, quadratic and algorithms.

Computational thinking skills

Know that computational thinking can be broken down into logical thinking, algorithmic thinking and optimal thinking.

Logical thinking
Know that logical thinking is about using readily available information to solve problems without making assumptions or taking short cuts and checking the validity of the result. These thinking skills can be used to solve problems, e.g. developing a set of rules to create a three by three number square using the numbers 1 to 9.

Algorithmic thinking
Know that algorithmic thinking involves using a predefined set of instructions to solve problems. These thinking skills can be used to solve problems, e.g. converting base 10 numbers to binary representations.

Optimal thinking
Know that optimal thinking involves refining one of the many possible solutions to a problem to make it more efficient, e.g. the quickest or with minimal waste and/or most effective, e.g. to obtain an improved result. These thinking skills are used to solve the problems, e.g. compare algebraic and graphical methods to solve functions.

A computational thinking process

A process using computational thinking to create a solution to a problem follows:

1 Breaking down a problem into a number of smaller, more manageable sub-problems with defined inputs and outputs.
2 Use of visualisation tools (such as diagrams, flow charts, functions, storyboards and models) to represent an existing problem or system.
3 Designing a solution to a problem using mathematical methods, e.g. pseudocode, a flow chart, Boolean operations or an algorithm.
4 Create/develop a solution to a problem (for the purposes of this unit a model of the solution with local and global variables is sufficient).
5 Test the solution using test data (normal, extreme and erroneous values) to demonstrate the validity of the solution.
6 Use optimal thinking (e.g. by recognising the similarities and patterns of output data) to refine the solution (e.g. to reduce waste or to increase the speed of calculating the solution).

Assessment criteria

Level 1	Level 2 Pass	Level 2 Merit	Level 2 Distinction
Learning aim A: Understand mathematical methods for calculations and their uses in computational thinking			
1A.1 Identify the computational uses of calculations and use calculations, with guidance, to: • write numbers using power notation • convert from decimal to binary and hexadecimal number systems.*	2A.P1 Explain the computational uses of calculations and use calculations to: • write numbers using power notation • convert from decimal to binary and hexadecimal number systems.*	2A.M1 Use calculations to: • identify rounding errors • show how natural numbers are represented in a computer • convert between all the three number systems.*	2A.D1 Discuss how integers and rational numbers can be represented in a computer.

Level 1	Level 2 Pass	Level 2 Merit	Level 2 Distinction
Learning aim B: Understand mathematical methods for functions and Boolean operations and their uses in computational thinking			
1B.2 Identify the computational uses of functions, and with guidance: • identify linear functions • obtain the function of a straight line from a graph.*	2B.P2 Explain the computational uses of functions and: • identify linear and quadratic functions • obtain the equation of a straight line from a graph.*	2B.M2 Use software to represent and define the range and domain of different types of linear and quadratic functions.*	2B.D2 Discuss how functions are used to calculate the position of objects and generate the information needed to plot shapes on screen.*
1B.3 Identify the computational uses of Boolean operations, and with guidance, produce truth tables and logic gates corresponding to at least **two** different operations.*	2B.P3 Explain the computational uses of Boolean operations, and produce truth tables and logic gates corresponding to at least three different operations.*	2B.M3 Use more complex Boolean operations to produce truth tables and logic gates corresponding to at least three different combinations of operations in series.*	2B.D3 Use logic gates to solve simple problems.*

Level 1	Level 2 Pass	Level 2 Merit	Level 2 Distinction
Learning aim C: Apply computational thinking to solve a simple problem			
1C.4 Design a solution, with guidance, including: • a description of the problem divided into smaller sub-problems as required • visualisation tools to represent the problem • mathematical methods to represent a possible solution to the problem.*	2C.P4 Design a solution, including: • a description of the problem divided into smaller sub-problems as required • appropriate visualisation tools to represent the problem • mathematical methods to represent a possible solution to the problem.*	2C.M4 Design a detailed solution to a problem using: • alternative solutions • test data.*	2C.D4 Justify the design decision taken, including: • how it will solve the problem • any design constraints.*
1C.5 With guidance, create a model of the solution and test the model for functionality.*	2C.P5 Create a model of the solution and test the model for functionality and repair any faults.*	2C.M5 Create a functional model which solves the problem and use test data to demonstrate the validity of the solution. Repair any faults.*	2C.D5 Use optimal thinking to refine the model.*

*Opportunity to assess mathematical skills

Teacher guidance

Resources

Learners will need access to a computer system with a spreadsheet package.

If not provided by the spreadsheet, graphing facilities capable of satisfying the outcomes described will need to be available. A good range of case study examples and exercises is needed.

Assessment guidance

This unit is assessed internally by the centre and externally verified by Pearson. Please read this guidance in conjunction with *Section 8 Internal assessment.*

Learners will be familiar with the content of the National Curriculum programme of study for mathematics in Key Stage 3.

It is suggested that this unit is assessed using a series of assignments.

It may be appropriate to use prepared exercises to provide evidence for some of the criteria in this unit, but it is recommended that, where possible, these be set to a vocational context. It is not expected that the assignments should be completed under controlled conditions.

If any exercises are used to provide evidence then outcomes should be considered against the assessment criteria rather than the award of a grade or numeric mark.

Where the unit content asks for a particular set of calculations to be completed, then all calculations must be completed successfully by individual learners and workings must be shown where appropriate.

Although some of this unit will be evidenced on paper or using a spreadsheet, centres should consider the use of assessment methods such as presentations, posters, leaflets, videos and podcasts. The use of these methods will allow assignments to be set in vocational contexts.

All three learning outcomes could be assessed in a similar way, with learners producing material for technical audiences specified by learners or the teacher.

Learners should use evidence from this unit as part of the learner's digital portfolio *(Unit 3: A Digital Portfolio).*

Learning aim A

It is important for learners to understand the purpose and uses of calculations as part of the fundamental concepts of computational thinking. Learners are expected to demonstrate how to use calculations by using appropriate mathematical methods to find solutions to a set of given problems. Learners need to be familiar with working with different types of numbers, including natural, integer, rational and binary. Most learners may be less familiar with binary and hexadecimal number systems and may need to spend additional time working with them.

For 2A.P1: learners are required to explain the computational uses of calculations and use calculations to write numbers using power notation. Therefore, they need to be familiar with numbers raised by powers and numbers expressed in standard form. Learners need to be given sufficient opportunities to practise the use of these numbers to ensure that they are comfortable with them. Teachers may wish to concentrate on working with powers of two in order to support the work with binary numbers later in the unit.

Learners are required to use calculations to convert positive decimal numbers to binary and hexadecimal number systems. Decimal numbers should be from 1 to 63.

For level 1, as a minimum, learners should identify the computational uses of calculations and use calculations to write numbers using power notation. Learners will also convert positive decimal numbers (in the range of 1 to 63) to binary and hexadecimal.

They are likely to need some assistance with these activities.

For 2A.M1: Learners need to be aware that using technology to carry out calculations can introduce errors, such as the rounding of numbers during calculations due to the amount of memory available to store the numbers. Learners should be able to use calculations to identify rounding errors.

Learners should concentrate on using an 6-bit representation to demonstrate how numbers from 1 to 63 can be represented in computer memory. Their previous work on powers of two should support these activities.

Learners are required to use calculations to convert between all the three number systems, including binary to hexadecimal and hexadecimal to decimal.

For 2A.D1: learners are required to discuss how integers and rational numbers can be represented in computer memory. Therefore, they should be able to represent real numbers in two's complement format and be able to explain the purpose of two's complement representations in computer memory. Real numbers can be converted to binary and represented in computer memory.

Learning aim B

It is important for learners to understand the purpose and uses of functions and Boolean operations as part of the fundamental concepts of computational thinking. Learners are expected to demonstrate how to use functions and Boolean operations by using appropriate mathematical methods to find solutions to a set of given problems.

For 2B.P2: learners are required to explain the computational uses of functions and identify linear and quadratic functions. Learners should be presented with a series of simple problems such as 'Two brothers bought ten pens. If one brother bought four pens, calculate how many pens were bought by the other brother.' Learners should practise expressing these problems as functions.

Learners are also required to obtain the equation of a straight line from a graph. Therefore, learners need to develop their skills in the use of coordinates to allow them to describe a given straight line on a graph in terms of $y = mx + c$. Learners should gain experience of lines with both positive and negative gradients and different y-axis incepts.

For level 1, as a minimum, learners should identify the computational uses of functions and use functions to identify linear functions, and obtain the function of a straight line from a graph.

They are likely to need some assistance with these activities.

For 2B.M2: learners are required to use software to represent and define the range and domain of different types of linear and quadratic functions. Therefore, learners should be able to use software to represent a range of applications for different types of functions. Learners should consider the practical applications for these functions.

For 2B.D2: learners are required to discuss how functions are used in programming output to screen-based devices. Therefore, they should be able to demonstrate an understanding of how functions are used to produce designs for screen layouts. They should also be able to explain how the mathematical concepts are used to create and position objects, such as circles, on screen.

For 2B.P3: learners are required to explain the computational uses of Boolean operations and produce truth tables and logic gates corresponding to at least **three** different operations. Therefore, learners should become familiar with the range of Boolean operations given in the unit content. In many cases learners may find it easier to relate these operations to real-life situations to understand the output related to the inputs. For example, if searching for boys who are more than 11 years old then an AND operation would be TRUE if the input was 'boy' and '12' and FALSE if the input was 'boy' and '10'. Learners should be able to draw up truth tables for each of the operations.

Learners should be introduced to the practical use of Boolean operations through the construction of logic gates. They need to appreciate that logic gates can have several inputs but only one output.

*For level 1, as a minimum, learners should identify the computational uses of Boolean operations (AND, OR, NOT, XOR) and produce truth tables and logic gates corresponding to at least **two** different operations. They are likely to need some assistance with these activities.*

For 2B.M3: learners are required to use more complex Boolean operations to produce truth tables and logic gates corresponding to at least **three** different combinations of operations in series. Therefore, learners need to be able to produce truth tables that represent the use of a combination of Boolean operations.

For 2B.D3: learners are required to use logic gates to solve simple problems. Therefore, learners should be able to demonstrate that they can solve problems that involve multiple inputs (at least **three**) and represent more complex processes. For example, this combination of NOT, AND and OR gates has three inputs: A, B and C. The outputs from the NOT and AND gates are combined through the OR gate to give the final output Z. A NOT gate has just one input. NOT gates are often used in emergency-stop buttons on machine tools. If an OR gate is used in a simple lighting circuit with two switches in parallel, the lamp will light if either switch is pressed. If an AND gate was used, both switches would have to be pressed at the same time.

Learning aim C

Learners will now relate the theory and application of different mathematical methods of calculations, functions and Boolean operations to solve a given problem by using computational thinking. Learners should have access to a suitable assessment brief, which outlines the problem.

Learners should apply logical thinking skills to solve a problem. Examples of problems could include:

- defining the nth term in a number sequence based on a practical problem
- combining logic gates to analyse data and provide correct output
- allocating network addresses to hosts (computers), routers, servers and switches using number systems
- designing control systems, including the way routers and switches organise traffic using logic gates.

Learners should also apply algorithmic thinking skills to solve a problem. Examples of problems could include:

- solving equations
- plotting graphs of linear and quadratic functions
- applying file permissions on web servers and other networked file systems
- calculating the route taken for data to travel across the network using linear and quadratic functions
- the route taken by the data, represented by linear and quadratic functions.

Learners should also apply optimal thinking skills to solve a problem. Examples of problems could include:

- refine algorithms to carry out procedures in the fewest possible steps
- perfective maintenance of a computer program.

For 2C.P4: learners should design a solution to the given problem. The design documentation should include:

- a description of the problem divided into smaller sub-problems (as required)
- appropriate visualisation tools to represent the problem, e.g. flow chart, logic gate diagrams
- mathematical methods to represent a possible solution.

For level 1, as a minimum, learners should design a solution to the given problem, and will need some assistance with this activity. The proposed solution will contain:

- *a description of the problem divided into smaller sub-problems (as required)*
- *visualisation tools to represent the problem*
- *mathematical methods to r*epresent a possible solution.

For 2C.M4: in addition to the requirements for the pass grade, learners should produce a detailed solution to a problem using:

- alternative solutions, e.g. using different mathematical methods, for example by using different functions
- test data (containing normal, extreme and erroneous values) to demonstrate the validity of the result.

For 2C.D4: learners are expected to be able to justify their design decisions, including a justification of how it will solve the problem, and to explain any design constraints.

For 2C.P5: learners should create a model of the solution. Learners should test their model for functionality and repair any faults.

For a pass grade, the model may not produce a valid result in all cases/situations.

For level 1, as a minimum, learners should create a model of the solution and test the model for functionality. They are likely to need support with this activity.

For 2C.M5: learners should create a functional model which solves the problem and use test data to demonstrate the validity of the solution. Learners are likely to need to adapt their designs to create a fully functioning model.

For 2C.D5: learners should use optimal thinking to refine their model – for example, make the solution more efficient, i.e. the quickest and/or most effective way to obtain the desired result.

Suggested assignment outlines

The table below shows a programme of suggested assignment outlines that covers the assessment criteria. This is guidance and it is recommended that centres either write their own assignments or adapt any assignments we provide to meet local needs and resources.

Criteria covered	Assignment	Scenario	Assessment evidence
2A.P1, 2A.M1, 2A.D1 (1A.1)	Calculations	A company has asked you to produce a learning tool, such as an induction booklet, for work experience learners, which will be applied in the workplace to solve real-world problems. The company would like you to support your brief guide by demonstrating how calculations can be used to solve problems. Calculations should include: • conversions on number systems • rounding of errors • representing natural numbers, integers and rational numbers in a computer.	• Booklet. • Posters. • Set of mathematical tasks. • Podcast. • Video.

Criteria covered	Assignment	Scenario	Assessment evidence
2B.P2, 2B.M2, 2B.D2 (1B.2)	Functions	The company now asks you to put together a brief guide to demonstrate how problems can be written as functions and to identify the type of functions that would be useful in representing data in graphs, and obtain an equation for a straight line from a given graph. The guide should include a section explaining the significance and use of software to represent the range and domain of linear and quadratic functions. The company would like you to support your guide by demonstrating how the functions are used in a software package such as a spreadsheet. In particular, to calculate the position of objects and generate simple shapes onscreen.	• Booklet. • Posters. • Set of mathematical tasks. • Podcast. • Video.
2B.P3, 2B.M3, 2B.D3 (1B.3)	The Logical Choice	The company would now like you to put together a presentation on Boolean operations and how they are used in digital devices. The company would like you to demonstrate the use of logic gates and truth tables to solve simple problems using three or more different operations in series.	• Booklet. • Posters. • Set of mathematical tasks. • Podcast. • Movie (combining still images with text/narration).

Criteria covered	Assignment	Scenario	Assessment evidence
2C.P4, 2C.P5, 2C.M4, 2C.M5, 2C.D4, 2C.D5 (1C.4, 1C.5)	Solving a Problem	The company has presented you with a problem to solve using computational thinking. The problem is to sort a large list of numbers in order. Design a solution to a problem including: • a description of the problem divided into small sub-problems • appropriate visualisation tools to represent the problem • structured mathematical methods to represent a possible solution • alternative solutions • test data. Create a model of the solution and test the model for functionality. Repair any faults as necessary. Use optimal thinking to refine the model – for example, to improve the efficiency of the process to gain the desired result.	• Working model. • Design documentation. • Test plan.