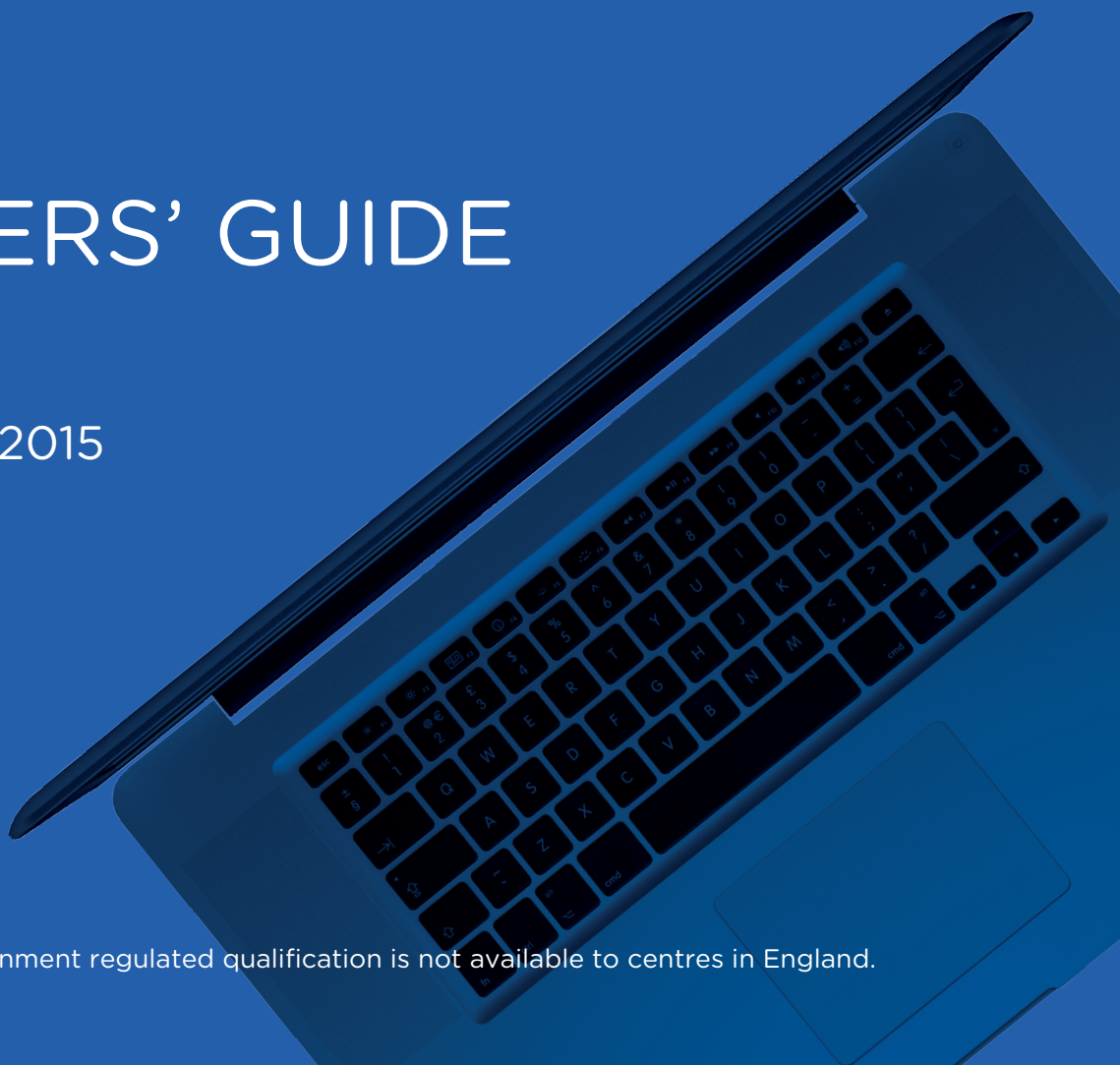
GCE AS/A LEVEL

WJEC GCE AS/A LEVEL in COMPUTER SCIENCE

ACCREDITED BY WELSH GOVERNMENT

TEACHERS' GUIDE

Teaching from 2015



This Welsh Government regulated qualification is not available to centres in England.

INTRODUCTION

The **WJEC AS and A level Computer Science** qualifications, accredited by Welsh Government for first teaching from September 2015, are available to:

- All schools and colleges in Wales
- Schools and colleges in independent regions such as Northern Ireland, Isle of Man and the Channel Islands

The AS will be awarded for the first time in Summer 2016, using grades A–E; the A level will be awarded for the first time in Summer 2017, using grades A*–E.

The qualifications have been developed following detailed consultation with experienced examiners, teaching practitioners and with Higher Education. They offer a contemporary approach to the subject from both theoretical and practical perspectives. They encourage students to apply the fundamental principles and concepts of computer science to a wide range of situations by offering a broad course of study that allows students the opportunity to analyse problems and to solve these problems using computational thinking. Students are encouraged to think creatively and innovatively in producing their solutions to problems and to consider relationships between different aspects of computer science as well as the impact of computer science on today's global society.

An emphasis on programming, mathematics and algorithms underpins each unit, mirroring the ubiquity of programming in computer science and encouraging learners to think like programmers.

Units 1, 2, 3 and 4 of the qualification encourage candidates to study a wide range of concepts and principles of computer science and are assessed by internal examination, with Unit 2 being on-screen examination. Unit 5 encourages candidates to utilise their theoretical knowledge in a practical context by programming a solution to a problem.

The WJEC AS and A level Computer Science qualifications have been designed to give a broad scope of study opportunities that prepare candidates for Higher Education or employment in a computer-related field. Equally, they are relevant to students who continue their studies in an unrelated field, given the widespread nature of computer science all areas of today's society.

These qualifications have been designed to free centres to concentrate on innovative delivery of the course by having a streamlined, uncomplicated, future-proof structure, with realistic technological requirements.

Additional ways that WJEC can offer support:

- Specimen assessment materials
- Face-to-face CPD events
- Examiners' reports on each question paper
- Free access to past question papers and mark schemes via the secure website
- Direct access to the subject officer
- Free online resources
- Exam Results Analysis
- Online Examination Review

AIMS OF THE TEACHERS' GUIDE

The principal aim of the Teachers' Guide is to support teachers in the delivery of the new **WJEC A level Computer Science** specification and to offer guidance on the requirements of the qualification and the assessment process.

The guide is **not intended as a comprehensive reference**, but as support for professional teachers to develop stimulating and exciting courses tailored to the needs and skills of their own students in their particular institutions.

The guide offers assistance to teachers, suggesting possible classroom activities and links to digital resources (both our own, freely available, digital materials and some from external sources) to provide ideas for immersive and engaging lessons.

The guide will concentrate on those areas new to WJEC subject specifications and those subject areas where guidance has been requested most.

ASSESSMENT STRATEGY : KEY EXAMINATION COMMAND WORDS (1 OF 2)

ASSESSMENT OBJECTIVES AND THEIR RELATED COMMAND WORDS

Assessment Objective	Description	Command words	Example questions
AO1	Demonstrate **knowledge** and understanding of the principles and concepts of computer science, including abstraction, logic, algorithms and data representation	State Give Discuss Define Identify Compare Name Suggest Describe Outline Summarise Explain what is meant Explain why	Give one advantage and one disadvantage of using a binary tree to store data compared with a linked list. [2] Explain the difference between a value and a reference parameter, giving an advantage of using a value parameter. [3] Define simplex, half duplex and full duplex data transmission. [3] Access to computer files may be sequential or indexed sequential. Compare these two types of access. [5] Explain what is meant by a data collision on a network. Describe what should happen when this occurs. [3]
AO2	**Apply knowledge and understanding** of the principles and concepts of computer science, including to analyse problems in computational terms	Draw Apply Use Demonstrate Calculate Convert Design Show Adapt (e.g. insert/delete) Simplify Produce Determine Explain by…	Draw a representation of a dynamic binary tree with pointers using the following data: *Newport, Canterbury, Oswestry, Warrington, Rugby, Bath, Derby* [2] Giving examples, compare a balanced and an unbalanced binary tree and evaluate their effectiveness to solve problems by comparing the maximum number of comparisons to locate an item in each of these trees. [4] Calculate the effect of carrying out an arithmetic shift left by two places on the eight-bit positive integer 00001111 and state the effect of this operation on the number. [2]

ASSESSMENT STRATEGY : KEY EXAMINATION COMMAND WORDS (2 OF 2)

ASSESSMENT OBJECTIVES AND THEIR RELATED COMMAND WORDS

Assessment Objective	Description	Command words	Example questions
AO3	**Design, program and evaluate** computer systems that solve problems, making reasoned judgements about these and presenting conclusions	Assess Evaluate Write algorithms Write a command Debug Correct Examine Determine	Evaluate the efficiency of the algorithm and, using Big O notation, determine the growth rate for the time performance. Your answer should refer to the addition and multiplication calculations performed by the algorithm. [5] Write an algorithm fragment which is suitable for parallel processing. To gain both marks, your fragment should demonstrate at least three calculations which can be carried out in parallel. [2] Write an SQL command to output the names of students who have the tutor with number 378. [1]

KEY ASPECTS OF THE SPECIFICATION FROM 2015

UNIT 1: FUNDAMENTALS OF COMPUTER SCIENCE	
AREA OF STUDY	**DESCRIPTION**
Parallelisation	Explain parallel processing.
Logical Operations	Draw truth tables for Boolean expressions consisting of AND, OR, NOT and XOR, logical operations and apply them to combinations of conditions in programs.
Simplifying Boolean Expressions	Show how to simplify Boolean expressions using Boolean identities and rules.
The Von Neumann Architecture and Fetch-Execute Cycle	Describe the fetch-execute cycle in terms of the flow of processes through a computer (specifically Von Neumann architecture).
Sign and Magnitude and Two's Complement	Represent positive and negative numbers using both sign/magnitude and two's complement notation.

UNIT 3: PROGRAMMING AND SYSTEM DEVELOPMENT	
AREA OF STUDY	**DESCRIPTION**
Binary Trees	Describe, interpret and manipulate binary trees and represent their operation by using pointers and arrays.
Linked Lists	Describe, interpret and manipulate linked lists and represent their operation by using pointers and arrays.
Hash Tables	Describe, hash tables and justify their use in given situations.
Logical Operations	Draw truth tables for Boolean expressions consisting of AND, OR, NOT, XOR, NAND and NOR logical operations and apply them to combinations of conditions in programs.
Simplifying Boolean Expressions	Show how to simplify Boolean expressions using Boolean identities,rules and De Morgan's laws
Big O Notation	Use Big O notation to determine the efficiency of different sorting algorithms in terms of their time and space requirements and to compare the efficiency of different sorting and searching algorithms. Use this notation to determine the complexity and efficiency of given algorithms in terms of their execution time, their memory requirements and of different algorithms that perform the same task
Shortest-Path Algorithms	Explain and apply a shortest-path algorithm (e.g. Dijkstra's algorithm).
Backus-Naur Form	Express language syntax using Backus-Naur form.
Waterfall and Agile	Describe the different approaches to the analysis and design of programs that Waterfall and Agile methodologies bring.

KEY ASPECTS OF THE SPECIFICATION FROM 2015

UNIT 4: COMPUTER ARCHITECTURE, DATA, COMMUNICATION AND APPLICATIONS	
AREA OF STUDY	**DESCRIPTION**
Parallelisation	Explain parallel processing and discuss how parallelisation may be limited.
Assembly Language	Write simple programs in assembly language and show how these programs could be executed by a computer.
Sign and Magnitude and Two's Complement	Represent positive and negative numbers using both sign/magnitude and two's complement notation.
Structured Query Language (SQL)	Construct queries in SQL to interrogate a database.
Lowest-Cost Routes	Calculate and determine the lowest-cost routes on a network.
Cryptography Algorithms	Explain how cryptography secures data and the methods by which data is made unreadable.

UNIT: 1 AND 4 EXAM LEVEL: AS/A2

AREA OF STUDY

Parallelisation

Key Points:

- Learners should understand that parallel processing is a form of computation in which many calculations are carried out simultaneously, operating on the principle that large problems can often be divided into smaller ones which are then solved concurrently.
- Questions on parallel processing will relate to AO1 and AO2.

Provides an opportunity to develop the following skills:

Speaking, Listening and understanding; Reading; Writing; Understanding numerical data; Carrying out calculations; Interpreting results; Presenting findings; Using ICT systems; Finding, selecting and exchanging information; Improving own learning and performance; Developing and presenting information using ICT; Working with others - plan do and review.

INSPIRATION FOR TEACHING

Example

Activity Name : Four Heads better Than One?

- Ask the class for five volunteers to help you demonstrate the advantages of parallel processing.
- Explain to the class that the task of sorting a deck of playing cards by suit (Clubs, Diamonds, Hearts and Spades), and in ascending order (from Ace to King) can be completed much quicker by four people (a Quad-core processor) than by one person (a Single-core processor).
- Shuffle two decks of cards, hand one deck to the lone learner and give a quarter of the other deck to each of the learners in the group of four. The task is to sort both decks of cards as quickly as possible.
- Ask the class to investigate situations where parallelisation is not always quicker than a single-core processor.
- You could ask learners to research Amdahl's law:

$$T(n) = T(1)(B + \frac{1}{n}(1 - B))$$

ADDITIONAL RESOURCES

WJEC > A Level Computer Science > Specification from 2015

WJEC > A Level Computer Science > Specimen Assessment Materials

UNITS: 1 AND 3 EXAM LEVEL: AS/A2

AREA OF STUDY

Logical Operations

Key Points:

- Learners should understand that logical operators establish relationships between statements that can be either TRUE or FALSE.
- There are six logical operators that students are required to know; AND, OR, NOT, XOR, NAND and NOR.
- Questions on logical operators will relate to AO1 or AO2.

Provides an opportunity to develop the following skills:

Speaking, Listening and understanding; Reading; Writing; Understanding numerical data; Carrying out calculations; Interpreting results; Presenting findings; Using ICT systems; Finding, selecting and exchanging information; Improving own learning and performance; Developing and presenting information using ICT; Working with others - plan do and review.

INSPIRATION FOR TEACHING

Example

Activity Name : Put Your Hands Up If…

- Tell learners that they are going to help demonstrate the meaning of each of the logical operators AND, OR and NOT.
- Ask all the students wearing blue to raise their hands and count the number of hands raised. Then ask all the students who are wearing blue AND a watch to keep their hands in the air, ewer hands should be raised. Explain that using the AND operator has narrowed the results.
- Ask all the learners wearing blue OR a watch to raise their hands, more hands should be raised. Explain that using the OR operator has expanded the results.
- Explain that logical operators can be used in combination. For example, ask all the students who are NOT wearing a watch AND have brown hair OR are wearing earrings to raise their hands.
- Explain that a raised hand can be represented as being TRUE or '1' and a lowered hand as FALSE or '0' and so be used to represent each operator in a truth table.

ADDITIONAL RESOURCES

WJEC > A Level Computer Science > Specification from 2015

WJEC > A Level Computer Science > Specimen Assessment Materials

UNITS: 1 AND 3 EXAM LEVEL: AS/A2

AREA OF STUDY

Simplifying Boolean Expressions

Key Points:

- Learners should understand that different Boolean Identities can be used to simplify Boolean expressions.
- Questions on Boolean expressions will relate primarily to AO2.

Provides an opportunity to develop the following skills:

Speaking, Listening and understanding; Reading; Writing; Understanding numerical data; Carrying out calculations; Interpreting results; Presenting findings; Using ICT systems; Finding, selecting and exchanging information; Improving own learning and performance; Developing and presenting information using ICT; Working with others plan do and review.

INSPIRATION FOR TEACHING

Example

Activity Name : Using Real-Life Situations to Simplify Boolean Expressions

- Explain to the learners that there are Boolean identities that they will need to use in order to simplify Boolean expressions.
- Tell the class that identities can be explained easily when applied to real-life situations. For example, the identity "*A.A = A*" seems complex, but when assigned to a real-life situation such as *'it is raining*' it becomes clear that the sentence *'it is raining* AND *it is raining'* means the same as '*it is raining'.*
- The same can be said for the Boolean identity $A.\bar{A} = 0$, where the sentence *'it's raining* AND *it's* NOT *raining*' is impossible and so this is always *false* or *0*.
- **Select the image (left)** for a list of example Boolean identities that you can provide to the class and ask them to apply the examples to real-life situations
- Explain to the class that these Boolean identities will allow them to simplify complex Boolean expressions.

ADDITIONAL RESOURCES

WJEC > A Level Computer Science > Specification from 2015

WJEC> A Level Computer Science > Specimen Assessment Materials

EXAMPLE BOOLEAN IDENTITIIES

Apply the following Boolean identities to real-life situations:

i) $1 + A = 1$
ii) $0 + A = A$
iii) $A + A = A$
iv) $0.A = 0$
v) $1.A = A$
vi) $A + B = B + A$

Apply the following Boolean identities where A is "*It's raining*" and B is "*I have an umbrella*".

i) $A.B = B.A$
ii) $A + (A.B) = A$
iii) $A.(A+B) = A$

ADDITIONAL RESOURCES

WJEC > A Level Computer Science > Specification from 2015

WJEC > A Level Computer Science > Specimen Assessment Materials

UNIT: 1 EXAM LEVEL: AS/A2

AREA OF STUDY

The Von Neumann Architecture and Fetch-Execute Cycle

Key Points:

- Learners should be able to identify and describe the main components of Von Neumann computer architectures.
- Learners should be able to describe the fetch-execute cycle, including how data can be read from RAM into registers.
- Questions on the Von Neumann architecture and the fetch-execute cycle will relate primarily to AO1.

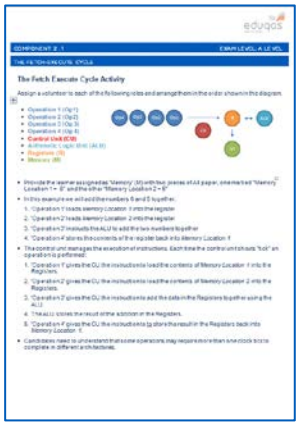
Provides an opportunity to develop the following skills:

Speaking, Listening and understanding; Reading; Writing; Understanding numerical data; Carrying out calculations; Interpreting results; Presenting findings; Using ICT systems; Finding, selecting and exchanging information; Improving own learning and performance; Developing and presenting information using ICT; Working with others - plan do and review.

INSPIRATION FOR TEACHING

Example

Activity Name : The Class as a Fetch-Execute Cycle



- **Select the image (left)** for detailed instructions and a diagram of an activity that explains the fetch-execute cycle using volunteers from the class.
- Ideally you will require eight volunteers to perform this activity. If your class is large enough you may choose to split the learners into groups of eight.
- In this example we are adding the numbers 6 and 5 together, but feel free to add your own instructions and amend the activity in any way you wish.

ADDITIONAL RESOURCES

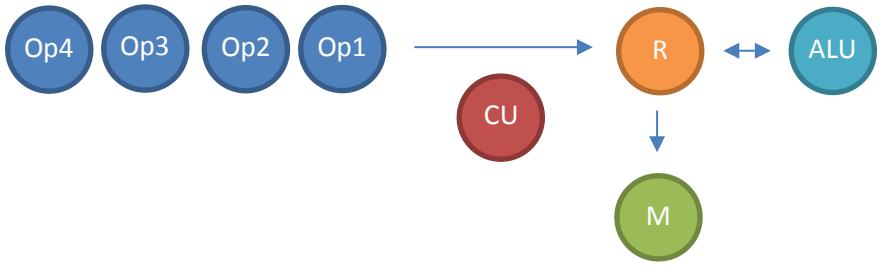
WJEC > A Level Computer Science > Specification from 2015

WJEC > A Level Computer Science > Specimen Assessment Materials

The Fetch-Execute Cycle Activity

Assign a volunteer to each of the following roles and arrange them in the order shown in the diagram.

- **Operation 1 (Op1)**
- **Operation 2 (Op2)**
- **Operation 3 (Op 3)**
- **Operation 4 (Op 4)**
- **Control Unit (CU)**
- **Arithmetic Logic Unit (ALU)**
- **Register (R)**
- **Memory (M)**



- Provide the learner assigned as 'Memory' (M) with two pieces of A4 paper, one marked "Memory Location 1 = 6" and the other "Memory Location 2 = 5"
- In this example we will add the numbers 6 and 5 together:
  1. 'Operation 1' loads *Memory Location 1* into the Register.
  2. 'Operation 2' loads *Memory Location 2* into the Register.
  3. 'Operation 3' instructs the ALU to add the two numbers together.
  4. 'Operation 4' stores the contents of the Register back into *Memory Location 1*.
- The CU manages the execution of instructions. Each time the CU shouts "tick" an operation is performed:
  1. 'Operation 1' gives the CU the instruction to load the contents of *Memory Location 1* into the Register.
  2. 'Operation 2' gives the CU the instruction to load the contents of *Memory Location 2* into the Register.
  3. 'Operation 3' gives the CU the instruction to add the data in the Register together using the ALU.
  4. The ALU stores the result of the addition in the Register.
  5. 'Operation 4' gives the CU the instruction to to store the result in the Register back into *Memory Location 1*.
- Candidates need to understand that some operations may require more than one clock tick to complete in different architectures.

UNITS: 1 AND 4 EXAM LEVEL: AS/A2

AREA OF STUDY

Sign and Magnitude and Two's Complement

Key Points:

- Learners should understand that there are two ways in which negative integers can be represented within computer systems.
- Questions on sign and magnitude and two's complement will relate to AO1 and AO2.

Provides an opportunity to develop the following skills:

Speaking, Listening and understanding; Reading; Writing; Understanding numerical data; Carrying out calculations; Interpreting results; Presenting findings; Using ICT systems; Finding, selecting and exchanging information; Improving own learning and performance; Developing and presenting information using ICT; Working with others; plan do and review.

INSPIRATION FOR TEACHING

Example

Activity Name : Representing Negative Binary Numbers

- Ask for eight volunteers from the class and say that they are going to help demonstrate how negative integers can be represented using binary digits.
- Line up the volunteers at the front of the class and hand each of them a piece of card with a '1' on one side and a '0' on the other. Explain that each volunteer will represent a single bit.
- Ask the learners to represent the number 7_{10} (00000111_2). Explain that when negative numbers are represented in 'sign and magnitude' form, the most significant bit is used to represent the sign of the number- '0' for positive and '1' for negative numbers. Ask the students to represent the number -7_{10} (10000111_2).
- Explain to the learners that there is a second method used to represent negative integers, called 'Two's complement', a two-step process whereby all of the bits in the binary number are 'flipped' (0s to 1 and 1s to 0) and 1 is added, in this example $-7_{10} = 11111001_2$.
- Ask the learners to represent other negative integers using both methods.

ADDITIONAL RESOURCES

WJEC > A Level Computer Science > Specification from 2015

WJEC > A Level Computer Science > Specimen Assessment Materials

UNIT: 3 EXAM LEVEL: A2

AREA OF STUDY

Binary Trees

Key Points:

- Learners should understand that binary trees are useful data structures for rapidly storing and retrieving stored data.
- A binary tree has a root node, and every node has at most two children.
- Questions on binary trees will relate primarily to AO2.

Provides an opportunity to develop the following skills:

Speaking, Listening and understanding; Reading; Writing; Understanding numerical data; Carrying out calculations; Interpreting results; Presenting findings; Using ICT systems; Finding, selecting and exchanging information; Improving own learning and performance; Developing and presenting information using ICT; Working with others - plan do and review.

INSPIRATION FOR TEACHING

Example

Activity Name : Sorting Numbers

- Split the class into pairs and give each pair squares of card/paper numbered 10, 6, 15, 8, 5, 11, 18, stacked in that order with 10 on top.
- Explain that you are going to show a method of sorting the cards in a way that will make it easier to find a specific number, but that you are NOT GOING TO PUT THEM INTO NUMERICAL ORDER.
- Arrange the 10, 6, 15 and 8 cards as in the picture on the left. Discuss the arrangement with the class.
- Ask learners to add the 5, 11 and 18 cards to the arrangement and to call out random numbers. Discuss with the class where these random numbers would be added to the tree.
- Ask learners to find card '18' by traversing the tree. Discuss how many comparisons this took and compare this with the number of cards that would have to be searched if the list were in ascending numerical order.

ADDITIONAL RESOURCES

WJEC > A Level Computer Science > Specification from 2015

WJEC > A Level Computer Science > Specimen Assessment Materials

UNIT: 3 EXAM LEVEL: A2

AREA OF STUDY

Linked Lists

Key Points:

- Learners should understand that linked lists provide a **logical order**, **without providing a physical order** to data items.
- Linked lists must contain **3 elements** to work, a start pointer, pointers and nodes.
- Questions on linked lists will relate primarily to AO2.

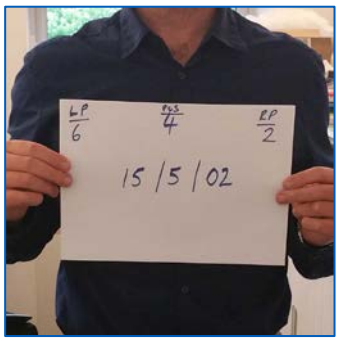
Provides an opportunity to develop the following skills:

Speaking, Listening and understanding; Reading; Writing; Understanding numerical data; Carrying out calculations; Interpreting results; Presenting findings; Using ICT systems; Finding, selecting and exchanging information; Improving own learning and performance; Developing and presenting information using ICT; Working with others - plan do and review.

INSPIRATION FOR TEACHING

Example

Activity Name : Arranging by Date of Birth



- Tell the class that you want them to arrange themselves in order of their date of birth, using the concept of a linked list.
- Give each learner a marker pen and a piece of A4 paper with a number indicating the node, e.g. if there are eight learners in the class number the sheets 1–8.
- Ask the learners to reveal their dates of birth to each other (but not to you) and to mark on their sheets the pointers indicating the order of their dates of birth. The eldest learner should add a start pointer to their sheet and a pointer to the node of the next oldest, the next oldest learner should add a pointer to the node of the next oldest and so on, until the youngest adds a pointer of 0 to their sheet of A4.
- Ask each learner in the class to hold up their sheets of A4 paper. By using the linked list you should be able to determine the order of the learners' dates of birth. Explain to the class that they are now placed in logical order, even though their physical order hasn't changed.

ADDITIONAL RESOURCES

WJEC > A Level Computer Science > Specification from 2015

WJEC > A Level Computer Science > Specimen Assessment Materials

UNIT: 3 EXAM LEVEL: A2

AREA OF STUDY

Hash Tables

Key Points:

- Learners should understand that hash tables use a **hash function** to calculate an **index value** for an array, in which items can be **stored** or **found.**
- Questions on hash tables will relate primarily to AO2.

Provides an opportunity to develop the following skills:

Speaking, Listening and understanding; Reading; Writing; Understanding numerical data; Carrying out calculations; Interpreting results; Presenting findings; Using ICT systems; Finding, selecting and exchanging information; Improving own learning and performance; Developing and presenting information using ICT; Working with others – plan do and review.

INSPIRATION FOR TEACHING

Example

Activity Name : Arranging by First Names

- Tell the class that searching for their first name on an attendance register is not efficient, as their first names are not in any partiular order. So you want them to arrange their first name in order using the concept of a hash table.
- Split the learners into groups and give each group a piece of A4 paper with a hash function written on it, for example:

  Length of first name **MOD** (size of group –1)
- Tell the members of each group that they will form a line (hash table) by performing the hash function on their first name and standing in the calculated position.
- As the hash function given will result in two learners having the same hash number, there will be a clash and both will try to stand in the same place. Ask the class how this problem could be resolved?
- You may wish to suggest different methods of resolving the problem, such as the use of a larger hash table, allowing clashes and storing data in overflow areas or using an improved hashing algorithm.

ADDITIONAL RESOURCES

WJEC > A Level Computer Science > Specification from 2015

WJEC > A Level Computer Science > Specimen Assessment Materials

UNIT: 3 EXAM LEVEL: A2

AREA OF STUDY

Big O Notation

Key Points:

- Learners should understand that Big O notation is used in computer science to describe the performance or complexity of an algorithm.
- Learners should know that Big O describes the worst-case scenario for the execution time required and the space used (e.g. in memory) by an algorithm.
- Questions on Big O notation may relate to AO1, AO2 or AO3.

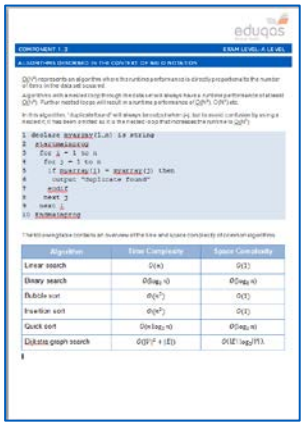
Provides an opportunity to develop the following skills:

Speaking, Listening and understanding; Reading; Writing; Understanding numerical data; Carrying out calculations; Interpreting results; Presenting findings; Using ICT systems; Finding, selecting and exchanging information; Improving own learning and performance; Developing and presenting information using ICT; Working with others - plan do and review.

INSPIRATION FOR TEACHING

Example

Activity Name : Explaining Big O Notation in the Context of Different Algorithms



- Big O notation can be a difficult concept to grasp. However, learners who are confident in their understanding of algorithms may appreciate Big O notation being explained in the context of different algorithms.
- Explain to the class that an algorithm that will always execute in the same time, regardless of the size of the input data set, would be described as O(1).
- **Select the image (left)** for an example O(1) algorithm, examples of other algorithms and a table displaying the time and space complexity of common algorithms that you can use to discuss with the class.
- Ask the learners to carry out further research on other algorithms with different runtime performance characteristics.

ADDITIONAL RESOURCES

WJEC > A Level Computer Science > Specification from 2015

WJEC > A Level Computer Science > Specimen Assessment Materials

ALGORITHMS DESCRIBED IN THE CONTEXT OF BIG O NOTATION

An algorithm that will always execute in the same time, regardless of the size of the input data set, would be described as O(1), i.e. this algorithm will take the same amount of time to execute if n =10, 100 or 1,000:

```
1 declare myarray(1…n) is string
2  startmainprog
3   if myarray(0) = null then
4    output “true”
5   endif
6  Endmainprog
```

In this algorithm, where the runtime performance is O(N), the runtime will grow linearly in direct proportion to the size of the dataset; each element in `myarray` will be checked once. So as n increases so will the runtime – hence this can be described as O(N).

```
1 declare myarray(1…n) is string
2  startmainprog
3   input SearchValue
4    for i = 1 to n
5     if myarray(i) = SearchValue then
6      output “found”
7     endif
8    next i
9  Endmainprog
```

Continued on next page

ADDITIONAL RESOURCES

WJEC > A Level Computer Science > Specification from 2015

WJEC > A Level Computer Science > Specimen Assessment Materials

ALGORITHMS DESCRIBED IN THE CONTEXT OF BIG O NOTATION

$O(N^2)$ represents an algorithm where the runtime performance is directly proportional to the number of items in the data set squared.

Algorithms with a nested loop through the data set will always have a runtime performance of at least $O(N^2)$. Further nested loops will result in a runtime performance of $O(N^3)$, $O(N^4)$ etc.

In this algorithm, "duplicate found" will always be output when i=j, but to avoid confusion by using a nested if, it has been omitted as it is the nested loop that increases the runtime to $O(N^2)$.

```
1 declare myarray(1…n) is string
2  startmainprog
3   for i = 1 to n
4    for j = 1 to n
5     if myarray(i) = myarray(j) then
6      output "duplicate found"
7     endif
8    next j
9   next i
10 Endmainprog
```

The following table contains an overview of the time and space complexity of common algorithms:

Algorithm	Time Complexity	Space Complexity
Linear search	$O(n)$	$O(1)$
Binary search	$O(\log_2 n)$	$O(\log_2 n)$
Bubble sort	$O(n^2)$	$O(1)$
Insertion sort	$O(n^2)$	$O(1)$
Quick sort	$O(n \log_2 n)$	$O(\log_2 n)$
Dijkstra graph search	$O(\|V\|^2 + \|E\|)$	$O(\|E\| \log_2 \|V\|)$.

UNIT: 3 EXAM LEVEL: A2

AREA OF STUDY

Shortest-Path Algorithms

Key Points:

- Learners should understand that computer systems calculate the distance between two locations by determining all possible routes and finding the shortest path.
- Dijkstra's algorithm is an example that finds the shortest path between two vertices on a graph.
- Questions on Shortest-Path Algorithms will relate to AO1, AO2 and AO3.

Provides an opportunity to develop the following skills:

Speaking, Listening and understanding; Reading; Writing; Understanding numerical data; Carrying out calculations; Interpreting results; Presenting findings; Using ICT systems; Finding, selecting and exchanging information; Improving own learning and performance; Developing and presenting information using ICT; Working with others - plan do and review.

INSPIRATION FOR TEACHING

Example

Activity Name : Wandering the School/College

Note: In this example, we will use the terms 'crossroad', 'corridor' and 'building'. However, in formal notation these terms are 'vertex', 'edge' and 'graph' respectively.

- Ask each learner to take three different routes from the classroom to the main reception or another location in the building.
- Learners should take a different corridor each time they reach a crossroad.
- Ask each learner to record carefully the number of steps taken for each route.
- When the learners have returned to the classroom ask them to each determine the shortest path of all three routes travelled.

ADDITIONAL RESOURCES

WJEC > A Level Computer Science > Specification from 2015

WJEC > A Level Computer Science > Specimen Assessment Materials

UNIT: 3 EXAM LEVEL: A2

AREA OF STUDY

Backus-Naur Form (BNF)

Key Points:

- Learners should understand that BNF is used to describe, **unambiguously**, the syntax of a programming computer language, whereas natural languages (such as English and Welsh) are normally ambiguous.
- Questions on BNF will relate primarily to AO1 or AO2.

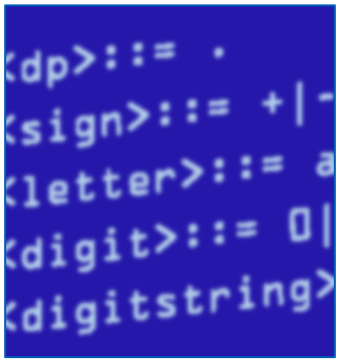
Provides an opportunity to develop the following skills:

Speaking, Listening and understanding; Reading; Writing; Understanding numerical data; Carrying out calculations; Interpreting results; Presenting findings; Using ICT systems; Finding, selecting and exchanging information; Improving own learning and performance; Developing and presenting information using ICT; Working with others - plan do and review.

INSPIRATION FOR TEACHING

Example

Activity Name : Defining Syntax



- Provide learners with the following numbers: **+2.3409**, **-6.3089**, **9.4000** and **+11.9901**.
- Ask the class to discuss the format of the numbers above, they should note that the numbers contain a sign or null, a digit string and a decimal point followed by four digits.
- Ask the class to define the syntax for the leading sign or null (e.g. <sign> : := +|-|null) and for the decimal point (e.g. <dp> : : .)
- Explain the BNF notation used to define digits (e.g. <digit>: := 0|1|2|3|…|8|9).
- The number contains a digit string. Explain the concept of recursion, where a digit string is determined in terms of itself (e.g. <digitstring>: := <digit>|<digit><digitstring>.
- Tell the class that they have now defined all of the components required for the format provided in the example and that the syntax for the numbers shown above is:

  <number>::= <sign><digitstring><dp><digit><digit><digit><digit>

ADDITIONAL RESOURCES

WJEC > A Level Computer Science > Specification from 2015

WJEC > A Level Computer Science > Specimen Assessment Materials

UNIT: 3 EXAM LEVEL: A2

AREA OF STUDY

Waterfall and Agile Approaches

Key Points:

- Learners should understand that the Waterfall approach has different discrete stages arranged linearly, with each stage cascading down to the next. It is impossible to begin one stage of the development until the preceding stage has been completed.
- Learners should understand that the Agile approach is an incremental approach to development, in which developers start off with a simple project design, instead of a huge document, and work on small modules at a time in an iterative way.
- Learners should appreciate that the Agile approach does not fully dismiss the Waterfall approach, but rather aims to improve the process of systems analysis.
- Questions on Waterfall and Agile approaches will relate to AO1.

Provides an opportunity to develop the following skills:

Speaking, Listening and understanding; Reading; Writing; Understanding numerical data; Carrying out calculations; Interpreting results; Presenting findings; Using ICT systems; Finding, selecting and exchanging information; Improving own learning and performance; Developing and presenting information using ICT; Working with others - plan do and review.

INSPIRATION FOR TEACHING

Example

Activity Name : Class Discussion on Waterfall and Agile Approaches

- Tell the class that there are two distinct approaches to systems analysis and development, namely Waterfall and Agile, and that they will need to be able to differentiate between them.
- Inform the learners that the name ‘Agile’ was coined as a result of a meeting between programmers, where lightweight software development methods were discussed. Ask the learners to research the Agile manifesto.
- **Select the image (left)** for an overview of the characteristics of both the Waterfall and Agile approaches.
- Hold a class discussion on the advantages and disadvantages of both approaches.

ADDITIONAL RESOURCES

WJEC > A Level Computer Science > Specification from 2015

WJEC > A Level Computer Science > Specimen Assessment Materials

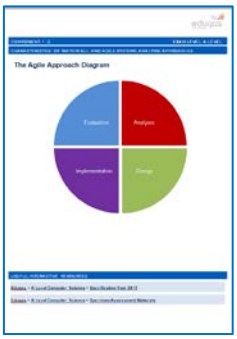
The Waterfall Approach

- The Waterfall approach has distinct stages.
- You complete one stage, e.g. analysis, before beginning the next stage, e.g. design.
- The name ‘Waterfall’ is used to describe how water (representing the project) flows down a set of steps – one stage flows into the next and there is no way for the water to flow back uphill to previous stages.

The Agile Approach

- The name ‘Agile' was coined as a result of a meeting between programmers where lightweight software development methods were discussed.
- The Agile approach advocates a less rigid approach to the structure of systems analysis than the Waterfall approach.
- The process is iterative. Lessons learned from one stage, e.g. implementation, are used in other stages, such as design, with the overall aim of customer satisfaction placed above adherence to strict requirements as set out in the analysis stage.

Select the image (left) for a Waterfall approach diagram



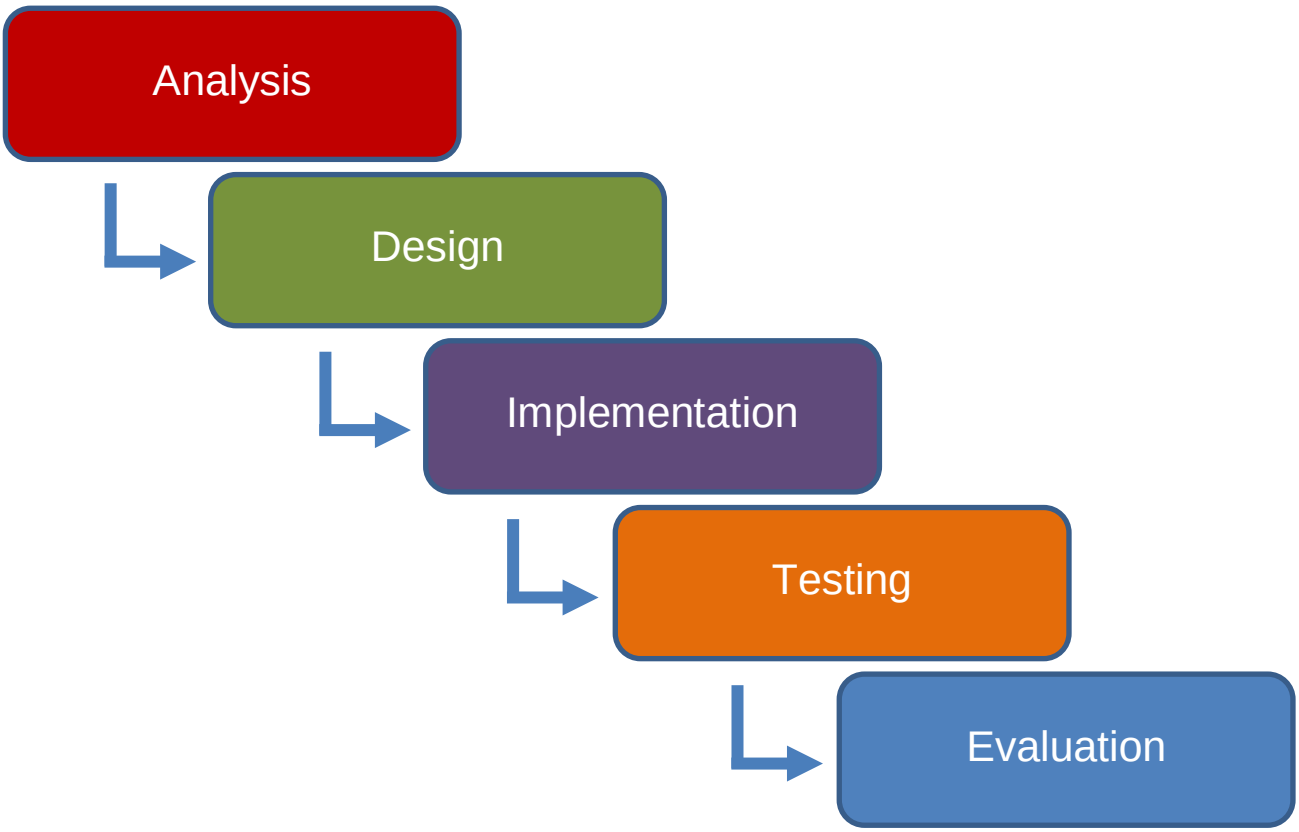
Select the image (left) for an Agile approach diagram

ADDITIONAL RESOURCES

WJEC > A Level Computer Science > Specification from 2015

WJEC > A Level Computer Science > Specimen Assessment Materials

The Waterfall Approach Diagram

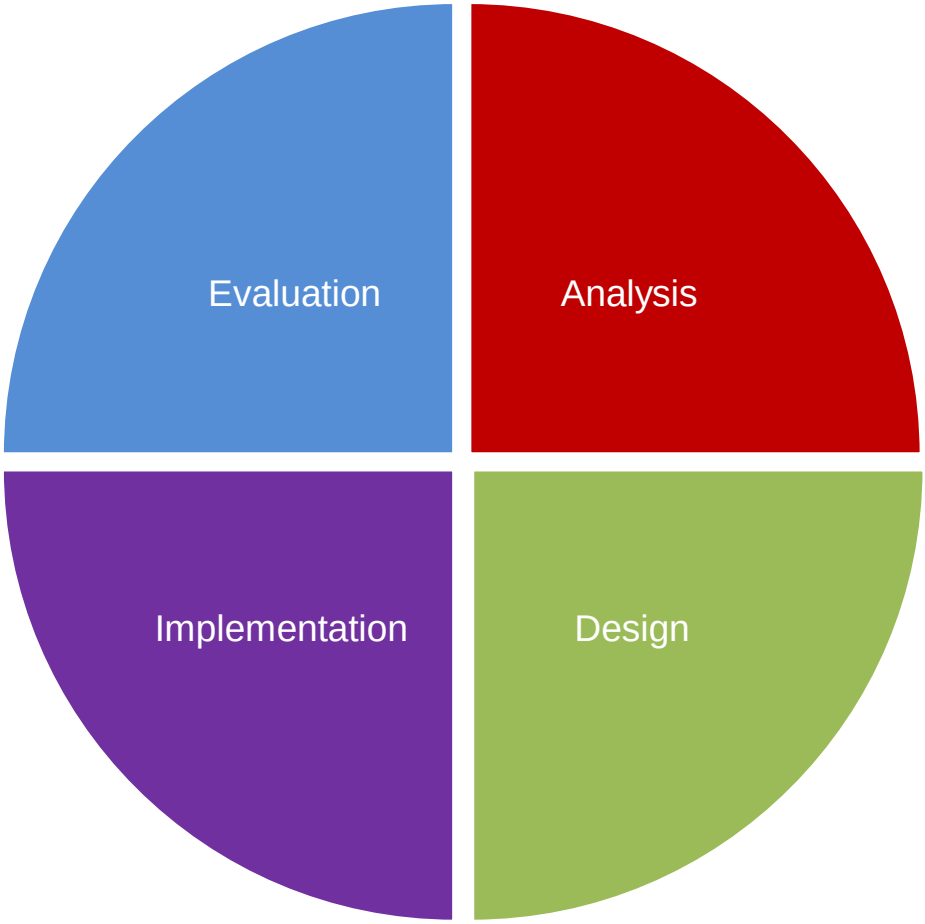


ADDITIONAL RESOURCES

WJEC > A Level Computer Science > Specification from 2015

WJEC > A Level Computer Science > Specimen Assessment Materials

The Agile Approach Diagram



ADDITIONAL RESOURCES

WJEC > **A Level Computer Science** > **Specification from 2015**

WJEC > **A Level Computer Science** > **Specimen Assessment Materials**

UNIT: 4 EXAM LEVEL: A2

AREA OF STUDY

Assembly Language

Key Points:

- Learners should be able to write simple programs in assembly language and demonstrate how these programs could be executed.
- Learners should understand that an assembly language instruction set is specific to the architecture of the CPU in the computer system.
- Learners should know that assembly language is converted into executable machine code by a program called an assembler
- Questions on assembly language will relate primarily to AO2.

Provides an opportunity to develop the following skills:

Speaking, Listening and understanding; Reading; Writing; Understanding numerical data; Carrying out calculations; Interpreting results; Presenting findings; Using ICT systems; Finding, selecting and exchanging information; Improving own learning and performance; Developing and presenting information using ICT; Working with others - plan do and review.

INSPIRATION FOR TEACHING

Example

Activity Name : Writing a Simple LMC Assembly Language Program

- Explain to the class that there are many different assembly language instruction sets available and that the Little Man Computer (LMC) instruction set is one example.
- Tell the learners that invariably an assembly language mnemonic maps to a machine code instruction and that some mnemonics are accompanied by an operand (parameters), which contains the data on which operations are carried out.
- Ask the learners to research the nine available LMC instructions,"Input" (INP); "Output" (OUT); "Store" (STA), etc.
- Ask the students to write a simple program in LMC assembly language for an algorithm that determines the highest value of two input numbers.

ADDITIONAL RESOURCES

WJEC > A Level Computer Science > Specification from 2015

WJEC > A Level Computer Science > Specimen Assessment Materials

UNIT: 4 EXAM LEVEL: A2

AREA OF STUDY

Structured Query Language (SQL)

Key Points:

- Learners should understand the construction of SQL statements and be able to run queries.
- Learners should know that SQL is a special-purpose programming language, used in relational database systems. Learners may be required to write SQL during an exam.
- SQL has the features of a number of different programming paradigms, including 4th generation languages and declarative languages. It also contains some procedural elements.
- Questions on SQL will relate to AO2 and AO3.

Provides an opportunity to develop the following skills:

Reading; Writing;

Interpreting results; Presenting findings;

Finding, selecting and exchanging information; Improving own learning and performance;

Developing and presenting information using ICT;

INSPIRATION FOR TEACHING

Example

Activity Name : Writing, Interrogating and Updating a SQL Database

- Explain to the class that SQL is particularly useful when designing, interrogating and updating data stored in a database.
- **Select the image (left)** for an example database creation, interrogation and updating task. You will also find a list of all the SQL commands that learners are required to know.
- You may provide the learners with the set task or develop your own similar data set and instructions if you prefer.
- Once the task is completed, you could ask the learners to devise their own queries and write the necessary SQL statements.

ADDITIONAL RESOURCES

WJEC > A Level Computer Science > Specification from 2015

WJEC > A Level Computer Science > Specimen Assessment Materials

ALGORITHMS DESCRIBED IN THE CONTEXT OF BIG O NOTATION

1. Write the SQL commands needed to create the following table:

Pets			
PetNum	PetName	Species	OwnerNum
10002	Max	Dog	102
11054	Willow	Dog	121
12767	Denzil	Cat	102
13121	Rex	Cat	213
13318	Princess	Dog	121

Owners	
OwnerNum	OwnerName
102	Thomas
121	Jones
213	Davies

2. Write the SQL commands needed to output:
   - the names of all pets
   - the names of all dogs
   - the names of the owners of all cats
   - the name and number of the owner of Princess

3. The owner of Denzil has sold the cat to Davies. Write the SQL commands needed to update the data.

List of All the SQL Commands Required			
`CREATE TABLE …`	`DateTime`	`OR`	`>=`
`PRIMARY KEY`	`INSERT INTO … VALUES`	`ORDER BY`	`<`
`NOT NULL`	`SELECT … FROM … WHERE …`	`GROUP BY`	`<=`
`Int`	`SELECT * FROM … WHERE …`	`UPDATE … SET …`	`<>`
`Char(n)`	`IN`	`=`	
`Numeric(m,n)`	`AND`	`>`	

UNIT: 4 EXAM LEVEL: A2

AREA OF STUDY

Lowest-Cost Routes

Key Points:

- Learners should understand that computer systems need to calculate the lowest cost for sending data from one network node to another.
- Learners should know that cost can be expressed as the amount of time taken to transfer data.
- Questions on lowest-cost routes will relate primarily to AO2.

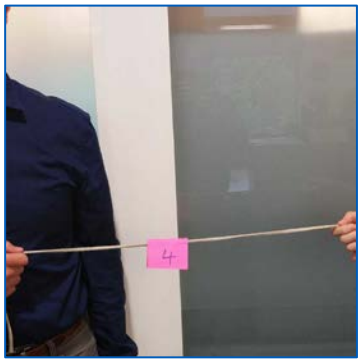
Provides an opportunity to develop the following skills:

Speaking, Listening and understanding; Reading; Writing; Understanding numerical data; Carrying out calculations; Interpreting results; Presenting findings; Using ICT systems; Finding, selecting and exchanging information; Improving own learning and performance; Developing and presenting information using ICT; Working with others - plan do and review.

INSPIRATION FOR TEACHING

Example

Activity Name : Sorting Numbers



- Split the class into groups of six and give five students in each group one of five pieces of paper labelled from 'A' to 'E'. The sixth person in each group represents a data packet.
- Each person (node) in each group is connected to at least one other. Show these connections with pieces of string. Attach a note that contains a number to each length of string.
- Starting at 'node A' the 'data packet' must follow the string to calculate the route costs between the nodes. The 'data packet' must find the lowest-cost route to all the 'nodes'.
- Ask the learners to produce a forwarding table for the nodes to all other nodes on the network.

ADDITIONAL RESOURCES

WJEC > A Level Computer Science > Specification from 2015

WJEC > A Level Computer Science > Specimen Assessment Materials

UNIT: 4 EXAM LEVEL: A2

AREA OF STUDY

Cryptography Algorithms

Key Points:

- Learners should understand that there is a need for cryptography for purposes such as the protection of confidential data being stored or transmitted over communication networks.
- Learners should compare cryptographic methods and their relative strengths and weaknesses.
- Questions on cryptography will relate to AO1, AO2 and AO3.

Provides an opportunity to develop the following skills:

Speaking, Listening and understanding; Reading; Writing; Understanding numerical data; Carrying out calculations; Interpreting results; Presenting findings; Using ICT systems; Finding, selecting and exchanging information; Improving own learning and performance; Developing and presenting information using ICT; Working with others - plan do and review.

INSPIRATION FOR TEACHING

Example

Activity Name : Break the Code

- Explain the concept of the Caesar cypher algorithm to the class,whereby an offset is used to encrypt letters of the alphabet. For example 'A' with an offset of 2 is encrypted as 'C'.
- Arrange the class into groups of three. Each group will contain a 'sender' a 'receiver' and a 'hacker'.
- Ask the 'sender' and the 'receiver' in each group to secretly agree an offset value (0–25) for their cryptography algorithm. The 'sender' should then write an encrypted message for the 'receiver' using this offset.
- The encrypted message should then be passed to the 'hacker' in each group for them to try and decrypt. Ask the groups to discuss their findings.
- Explain the inherent weakness of the Caesar cypher algorithm to the class, namely that because some letters appear more frequently in English sentences (e.g. letter 'E' far more frequent than 'Q') statistical analysis would allow a hacker to determine the offset and decrypt any messages.

ADDITIONAL RESOURCES

WJEC > A Level Computer Science > Specification from 2015

WJEC > A Level Computer Science > Specimen Assessment Materials