



De Lisle preparing for A Level Computer Science 2024-25

Threshold Concept	Description	Task title	Mastery level (RAG)	Date
1	Components of computer systems	• Build a PC Completion Date – 22/05/2020		
2	Networks	• Home network survey Completion Date – 05/06/2020		
3	Software	• My favourite OS Completion Date – 12/06/2020		
4	Algorithms and Problem Solving	• Game Design Completion Date – 19/06/2020		
5	Programming	1. Python Basics – 26/06/2020 2. Game Part 1 – 3/07/2020 3. Game Part 2 – 10/07/2020		

Resources to use:

- **CraigNDave Youtube Videos for GCSE and A-Level**
- **Seneca online learning – GCSE–**
- **GCSE revision guides & class notes alongside GCSE bitesize**

In order to confidently approach your A Level Computer Science studies there are certain fundamental concepts that you need to understand. Ensuring you have a secure foundation of GCSE knowledge to build upon will help to make the transition from GCSE to A level smoother and maximise your chances of success.

1. Components of a computer system

A Level Specification description	9 - 1 GCSE level specification description
1.1 The characteristics of contemporary processors, input, output and storage devices	1.1 Systems architecture
1.1.1 Structure and function of the processor – Continued work on the Von Neuman Architecture but also a look at the Harvard model and other contemporary models.	1.1 Systems architecture – What are the main components of a Von Neumann computer system and a typical CPU. How do they work together during the execution of programs.
1.1.2 Types of processor – How and why different processor designs exist - RISC and CISC, GPUs etc	Not introduced at GCSE
1.1.3 Input, output and storage	1.2 Memory – The difference between ROM, RAM, and Virtual Memory 1.3 Storage – What are the main types of storage and

1.1 What do you need to know?

The A-Level specification builds on your understanding of the Von Neumann architecture and the knowledge that all computer systems share the same core components. For this section, you will be required to research these components and demonstrate that you have a thorough grasp of the key components of computer systems and the role that each component fulfils.

Key questions that need to be answered

- What is a computer system by definition?
- What do all computer systems consist of?
- What is the Von Neumann architecture?
- What are the main components of a computer system and their purpose?
- How do these components work together to produce a functioning system?
- How do these components differ for different types of system?
- How do we select appropriate components for a given system?

1.2 Description of the tasks

There are two main activities to this section.

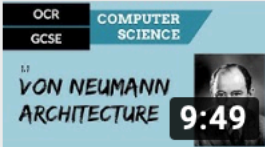
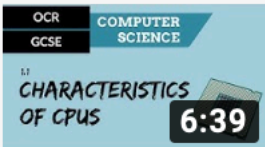
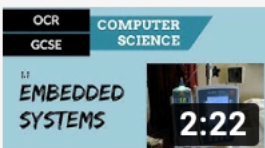

Part 1 - You must carry out your research into the various terms and components, so that you can complete the table of information and short activities below.

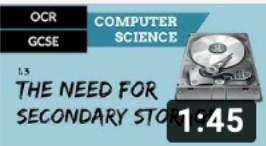
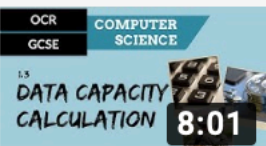

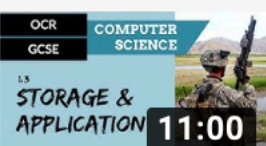
Part 2- You must use a suitable website to help build a computer systems for a specific purpose, selecting and justifying the components chosen.

1.3 Help and guidance for the task

There Help and Guidance-

Videos to watch on YouTube (see the CraigNDave channel) -

	OCR GCSE 1.1 The von Neumann architecture craigndave
	OCR GCSE 1.1 How common characteristics of CPUs affect their... craigndave
	OCR GCSE 1.1 Embedded systems craigndave
	OCR GCSE 1.2 RAM and ROM craigndave

	OCR GCSE 1.3 The need for secondary storage craigndave
	OCR GCSE 1.3 Data capacity and calculation of data capacity... craigndave
	OCR GCSE 1.3 Common types of storage craigndave
	OCR GCSE 1.3 Suitable storage devices and storage media for a give... craigndave

Background Reading-

1.4 Tasks to complete

Part 1 – Research into Computer Systems

Watch the suggested videos and read-up on the topic from the provided websites.

Complete the table below to demonstrate your understanding of Key Terminology for this topic.

Key Term / Component	Definition/meaning
Von Neumann Architecture	
Computer System	
Inputs	
Outputs	
Hardware	
Memory (RAM)	
Processor (CPU)	
Buses	
Motherboard	
Secondary Storage	
HDD	


SSD	
GPU	
Power Supply	

Part 2 – Build a PC

You have a budget of £1000 to build a desktop computer. The computer will be used for completing work, browsing the internet, streaming media, editing photos and videos and casual gaming.

You will also need to factor in additional hardware required such as a monitor and input/output devices (mouse and keyboard).

You can use a website like [PC Part Picker](#) to help build your computer, but then you need to transfer the specific details to the table below so that you can then justify your choice of components.

Part Name, Make and Model	Description, Purpose and Price	Justification of purchase
Eg. Intel i5 9500 Processor 	This processor has 6 cores each capable of 2 threads. The cores operate at a clock speed of 3.0ghz upto a maximum of 4.4ghz. Etc etc..... Price = £205	I have chosen this processor because...
RAM		
Motherboard		
Component 4		
Component 5		
Component 6		
Component 7		
Add any components you want to		

2. Networks

A Level Specification points	9 - 1 GCSE
1.3 Exchanging data 1.3.3 Networks	1.4 Wired and wireless networks 1.5 Network topologies, protocols and layers
1.3.4 Web Technologies	1.6 System security

2.1 What do you need to know?

The A-Level specification builds on your understanding of the different types of networks that exist. You should know about typical network devices and technology, how the roles that they all play in making our networks function.

Key questions that need to be answered

- What is a network by definition?
- What are the main pieces of network hardware and their purposes?
- What protocols and standards are required for network and internet communication?
- How is the client-server model used in a variety of applications – such as web-hosting?
- How is software used in conjunction with hardware, to facilitate and manage networks?
- How are networking capabilities and demands changing?

2.2 Description of the tasks

There are two main activities to this section.

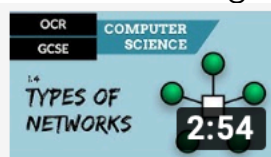
Part 1 – Make sure that your knowledge and understanding of networks is sound. Watch the videos and complete the definition table.

Part 2- Carry out a survey on your home network and internet provider. Create a diagram showing the structure of your home network and how it connects to the internet.

2.3 Help and guidance for the task

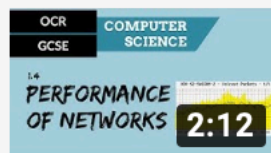
Videos to watch

Watch the CraigNDave videos on Networks



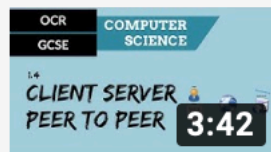
OCR GCSE 1.4 Types of networks

craigndave



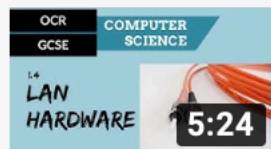
OCR GCSE 1.4 Factors that affect the performance of networks

craigndave



OCR GCSE 1.4 Client server and peer to peer networks

craigndave



OCR GCSE 1.4 Hardware to connect to a LAN

craigndave



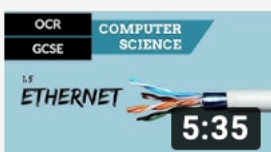
OCR GCSE 1.4 The Internet

craigndave



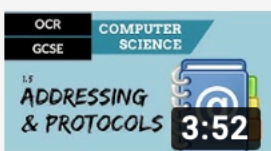
OCR GCSE 1.5 WiFi

craigndave



OCR GCSE 1.5 Ethernet

craigndave

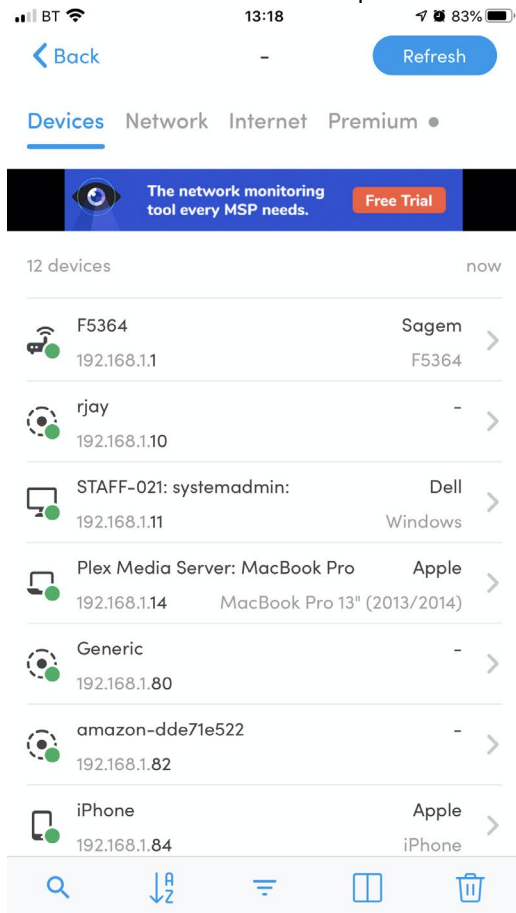


OCR GCSE 1.5 The uses of IP, MAC addressing and protocols

craigndave

Useful Tools

Download and try out a network scanner like **Fing – Network Scanner**. This allows you to see the devices that are on the same network as your phone. It will give a name and IP address of each device. The icons give you a clue as to what it is, if the name is not obvious. This sort of app is very useful when surveying networks and you want to find an IP address of a particular device.



2.4 Tasks to complete

Part 1 – Research into Computer Systems

Watch the suggested videos and read-up on the topic from the provided websites.

Complete the table below to demonstrate your understanding of Key Terminology for this topic.

Key Term / Component	Definition/meaning
Network	
LAN	

WAN	
IP Address	
Server	
Client	
Router	
Switch	
Wireless Access Point	
Ethernet	
Wi-Fi	
Data Packet	
Packet Switching	
TCP / IP	
Internet	
Web Server	

Part 2a – Network Survey and Diagram

Carry out a survey of your home and the hardware that you have that forms the network.

Complete the survey below-

- 1) **Who is your internet service provider (eg. BT, Virgin Media etc)?**
- 2) **What type of connection is it? (Cable / Phoneline / Fibre)**
- 3) **What speed connection do you get to your ISP (Google “speed test”)**
 - a. **Download Speed**
 - b. **Upload Speed**
- 4) **What router do you have to connect to your ISP?**
- 5) **How many ethernet ports does your router have?**
- 6) **What is the external IP address of your home? (google “whats my ip?”)**
- 7) **What is the IP address of your router? (google your router model)**
- 8) **List the devices that are connected to your home network and their IP address (Use an app like “Fing – Network Scanner”)**

Device	IP Address
Eg. BT Homehub Router	192.168.1.1

Part 2b – Create a Diagram

You need to show how the different devices in your house are connected together. Your diagram should give an approximate guide to position of device in the home and how they are connected to the router. You should also include other details that you found out during the survey, and then any else that you think would help explain your home network.

Use the template on the next page OR draw on paper and add photograph of your work. You can use the icons below to represent the different devices

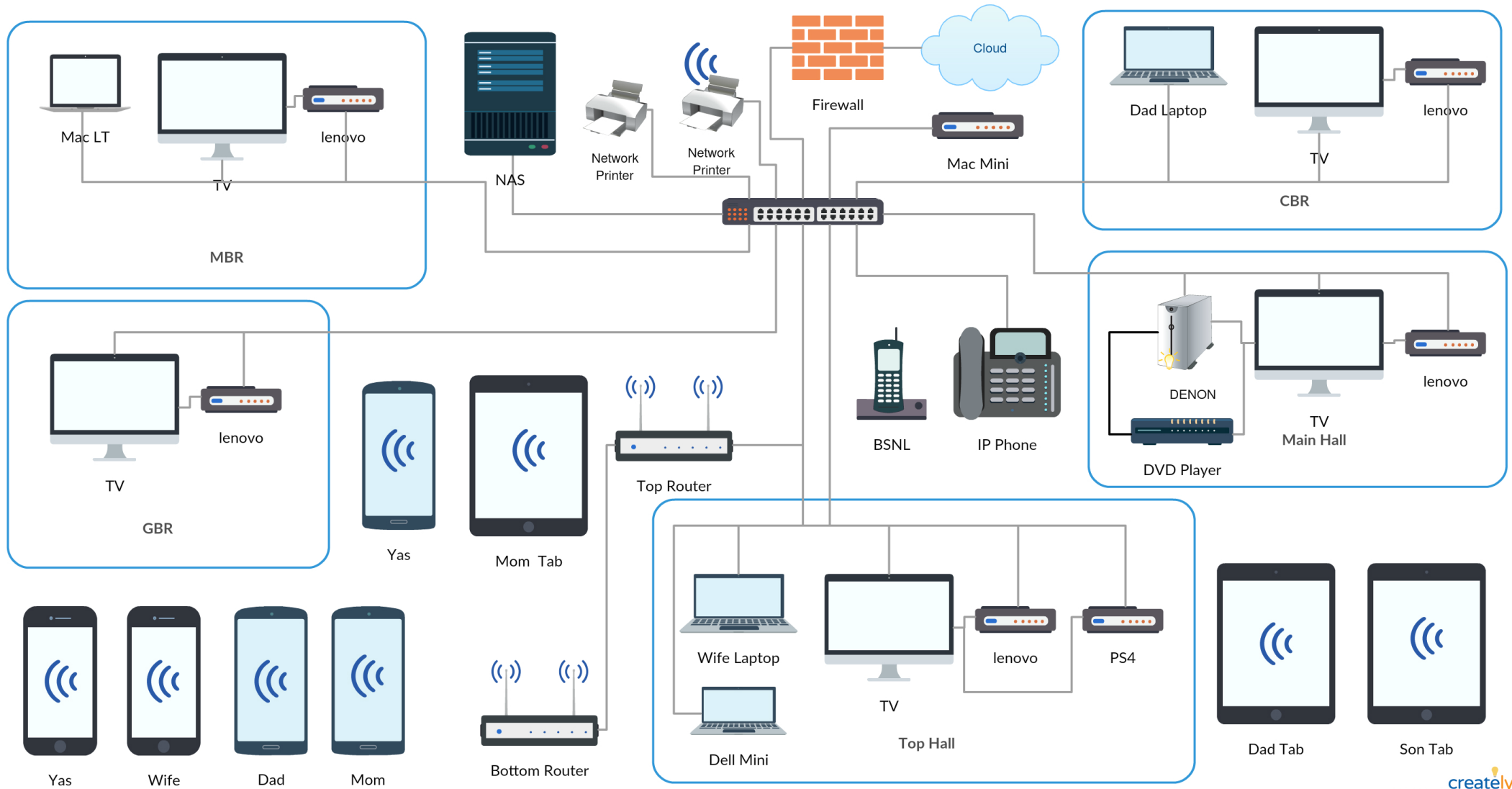
There is an example on the next page, and you can use the same icons that are used below. You can also find many examples if you search for “network icons”.

Diagram Checklist

Your diagram should include-

1. Icons/Images to represent each of the devices you identified in your survey
2. Additional network items like wi-fi range extenders or powerline adapters
3. IP addresses of each of the devices
4. Lines to represent how the device is connected (wired or wireless)
5. Details about your internet service provider (ISP) including your public IP address

Example Network Plan – Use this to help, yours may be simpler



Your Home Network Plan – Edit and complete

Room 1- Name

Room 3 - Name

Room 2- Name

Room 4- Name

ISP Details

2.5 Network Exam style question

People often have many devices connected together in a home network, and therefore they are able to share files between these different devices. Explain how a music file on a phone can be shared to play on a wi-fi enable speaker in the home. You should use your knowledge of network hardware and protocols such as TCP/IP to explain your answer. [8 marks]

Answer Box

3. Software

A Level Specification points	9 - 1 GCSE
1.2.1 Systems Software (a) The need for, function and purpose of operating systems. (b) Memory Management (c) Interrupts, the role of interrupts (d) Scheduling of the CPU (e) Device drivers.	1.7 Systems software <ul style="list-style-type: none">the purpose and functionality of systems softwareoperating systems:<ul style="list-style-type: none">user interfacememory management/multitaskingperipheral management and driversuser managementfile management

3.1 What do you need to know?

In your GCSE studies, you learnt about different software. At the heart of this topic is the most critical piece of software, the operating system. At A Level you will develop your knowledge and you will learn in much more technical detail, how the operating systems performs it's role.

Key questions that need to be answered

- What is software?
- What are the main categories of software ?
- What are the main functions of an operating system?
- How does the Operating System manage resources?

3.2 Description of the tasks

There are two main activities to this section.

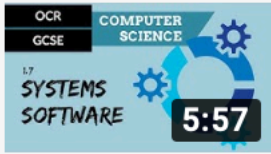
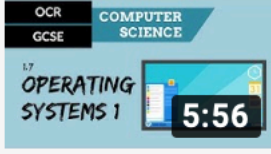
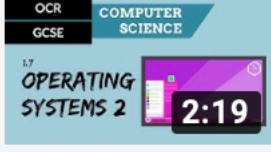
Part 1 – Research the role of the operating system to ensure that your knowledge and understanding is good.

Part 2 – Create a study on an operating system of your choice. Carry out detailed research into the features of your chosen operating system and what it enables the user to do. You can also compare it to alternative operating systems.

3.3 Help and guidance for the task

Videos to watch

Watch the CraigNDave videos on Operating Systems

 The thumbnail shows a blue background with a gear icon and the text 'OCR GCSE COMPUTER SCIENCE 1.7 SYSTEMS SOFTWARE' and a duration of '5:57'.	OCR GCSE 1.7 The purpose and functionality of systems software craigndave
 The thumbnail shows a blue background with a laptop icon and the text 'OCR GCSE COMPUTER SCIENCE 1.7 OPERATING SYSTEMS 1' and a duration of '5:56'.	OCR GCSE 1.7 Operating systems part 1 craigndave
 The thumbnail shows a blue background with a laptop icon and the text 'OCR GCSE COMPUTER SCIENCE 1.7 OPERATING SYSTEMS 2' and a duration of '2:19'.	OCR GCSE 1.7 Operating systems part 2 craigndave

Make whatever notes you need whilst watching the videos. These will help you complete the next tasks.

3.4 Tasks to complete

Part 1 – Research into Computer Systems

Watch the suggested videos and read-up on the topic from the provided websites. Use your notes to answer the following questions.

Write your answer below.

Question 1

What is an operating system and give two examples of operating systems you know of? [2 marks]

Question 2

Identify and then explain 4 different functions that the operating system performs [8 marks]

Function	Explanation
Eg. The operating system provides an Interface	

Question 3

Think about the different operating systems that you have used on different devices. Identify and explain THREE common pieces of software/apps that come as part of the operating system. [8 marks]

Software	Explanation (what is it for?)
Eg. File Manager	

Question 4

Discuss why different operating systems are required for different devices. For example, why is there a need for different operating systems for TVs, games consoles and PCs? [6 marks]

Part 2 – Investigate an Operating System

For this task you need to choose and investigate an operating system. You should choose an operating system that you have access to and can use on a device in your home. This will make it easier to complete the activity.

Use the template on the next page OR draw on paper and add photograph of your work. You can use the icons below to represent the different devices

2a. Outline to Investigation – Complete this form

Name of chosen OS –	
Current Version –	
Year of first version –	
Platform on which it runs Eg. PC, iPhone, TVs etc	
Primary method of control eg. Mouse	
Other input/control methods	
Proprietary or Open- Source	
Main competitors/alternatives	
Unique Features Eg. IOS has TouchID integration for logging into apps	

2b. Main investigation

Produce a short report about your chosen OS. It can be done as a document or presentation. Send your presentation to matthew.jackson@delisle.leics.sch.uk

In your report you should include sections/slides on the following-

- Introduction to the OS – What is it?
- How has it changed (looking at older to current version)
- Main features
- How does it compare to competitors – strengths and weaknesses

4. Algorithms

A Level specification points	9 - 1 GCSE
2.3 Algorithms (a) Analysis and design of algorithms for a given situation. (b) The suitability of different algorithms for a given task (c) algorithm efficiency (d) Comparison of the complexity of algorithms. (e) Algorithms for the main data structures (f) Standard algorithms (bubble sort, insertion sort, merge sort, quick sort, Dijkstra's shortest path algorithm, A* algorithm, binary search and linear search).	computational thinking: <ul style="list-style-type: none">• abstraction• decomposition• algorithmic thinking how to produce algorithms using: <ul style="list-style-type: none">• pseudocode• using flow diagrams

4.1 What do you need to know?

You need to be able to problem solve using a range of techniques. You should be able to decompose problems into smaller parts and then design solutions for these parts. You should be able to plan your solutions in the form of written algorithms and flowcharts.

Read through the information below to refresh your memory on this topic. The terms and concepts should be familiar to you. There are also some videos to help explain.

Summary of information

What is an Algorithm?

An algorithm is simply a set of steps/instructions needed to complete a task. A recipe is an example of an algorithm. For instance you could define an algorithm to make a cup of tea. You start by filling the kettle, then place a tea bag in the cup and so on. When the word algorithm is used in computing, it defines the way in which a software task can be completed or the way a problem can be solved.

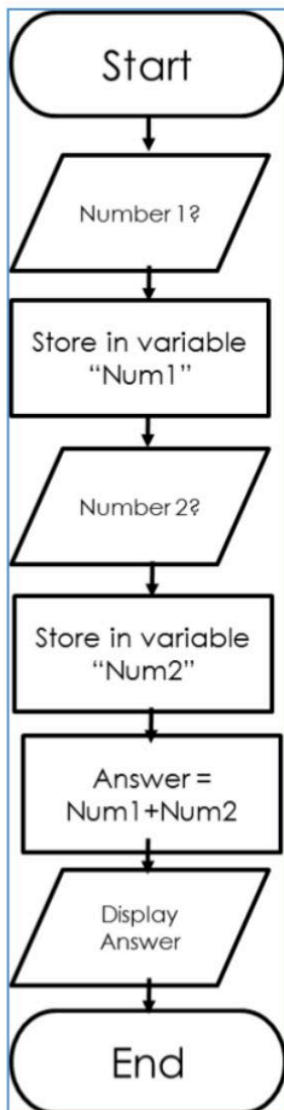
Writing algorithms is all about problem solving. After being given a problem to solve, programmers never begin programming straight away, they will first do the following:

- Think about the important parts of the problem (Abstraction),
- Break the problem down into smaller chunks (Decomposition),
- Think of the steps needed to solve each chunk (Algorithmic Thinking),

- Write down the steps – which in turn becomes the algorithm (Algorithmic Thinking).

Algorithms can be represented as a **Flowchart** or in **Pseudocode**. See an example of each below – both solve the same problem.

Flowchart



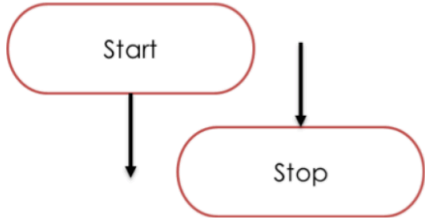


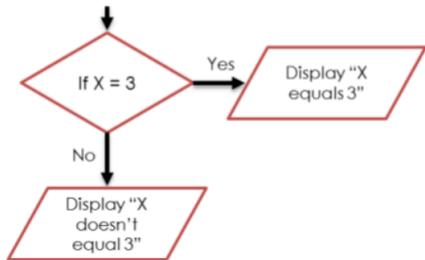
Pseudocode

```
number1 = integer
number2 = integer
answer = integer

INPUT number1
INPUT number2
answer = number1 + number2

PRINT answer
```

Flowcharts and their symbols

Representation	Explanation	Symbol
Start and Stop (Terminal) Symbols	<p>All flow charts begin with a Start Symbol and at the end of the flow chart (or at various end points of the chart) we place a Stop Symbol.</p> <p>There are drawn as a rectangle with curved ends</p>	
Process Symbols	<p>Most of the time a flow chart will demonstrate the sequence of instructions to be carried out.</p> <p>Simple processes (like "Add 1 to x" or "append x to List") are shown using a standard rectangle.</p>	
Input / Output Symbols	<p>At times your program will most certainly ask the user for inputs and output values too.</p> <p>Inputs and Outputs (like "Name?" or "...display age") are shown using a parallelogram.</p>	
Decision Symbols	<p>At times your program will be programmed to make a decision based on certain conditions.</p> <p>Decisions (like "IF X = 3" or "WHILE Y > 3") are shown using a diamond.</p>	



Pseudocode

Algorithms can be written in ‘generic code’ (which doesn’t follow any particular syntax) as well as a flowchart. This ‘generic code’ is known as pseudocode. Pseudocode has keywords such as IF, ELSE and FOR and so mimics a programming language and therefore the logic is easy to follow and easy to turn into code. Although there are no STRICT STANDARDS for pseudocode there are some generally accepted keywords that you should be aware of.

Pseudocode Example:

- A central heating system will try to keep the temperature between 2 values (19 and 21)
- If the temperature falls below 19 it will turn the heating system on
- If the temperature rises above 21 it will switch the heating system off.

START
IF Temp < 19 **THEN**
 Heating ← On
ELSE
IF Temp > 21 **THEN**
 Heating ← Off
END

Pseudocode and commonly used keywords

Representation	Explanation	Symbol
Start and End “Key Words”	<p>Pseudocode begin with a START and ends with END</p> <p>The algorithm goes in between.</p>	START END
Process “Key Words”	<p>Most of the time pseudocode will outline the logical sequence of instructions to be carried out.</p> <p>Simple processes will often use the key words shown below (like “CALCULATE X*2” or “INCREMENT X by 1)”</p> <p>You don’t have to always use these words, for example the logic statements such as “Add 1 to x” or “append x to List”) are fine too.</p>	Compute Calculate Determine Increment; ++ or += Decrement; -- or -=

Input / Output "Key Words"	<p>At times your program will most certainly ask the user for inputs and output values too.</p> <p>Inputs and Outputs (like "Name?" or "...display age") are indicated using the following words. Usually a programmer will choose one and stick with it throughout their algorithm.</p>	<p><u>INPUTS:</u> <u>OUTPUTS:</u></p> <p>READ PRINT OBTAIN DISPLAY GET SHOW INPUT OUTPUT</p>
Variable Assignment "Key Words"	<p>At times your program will assign values to variables. In pseudocode, this is done using the following key words.</p>	<p>SET INIT</p> <p>Sometimes an arrow is used instead e.g.:</p> <p>X ← 10 <i>This means SET X to 10</i></p>
Decision/Selection "Key Words"	<p>At times your program will be programmed to make a decision based on certain conditions.</p> <p>Decisions (like "IF X = 3, THEN ...") are shown using, unsurprisingly, the following key words.</p>	<p>IF THEN ELSE ELSE-IF ENDIF</p>
Loops / Iterations "Key Words"	<p>Programs will often loop in places while certain conditions occur (condition controlled) or for a set number of times (count controlled).</p> <p>Loops use the following key words:</p>	<p>Count Controlled:</p> <p>FOR</p> <p>Condition Controlled:</p> <p>WHILE / ENDWHILE</p> <p>REPEAT / UNTIL</p>

4.2 Description of the tasks

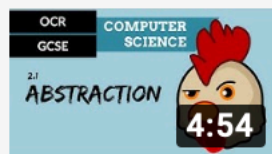
There is one main task for this topic.

Main Task – Write algorithms for the functions outline in the task. You will need to create flowcharts and pseudocode algorithms

4.3 Help and guidance for the task

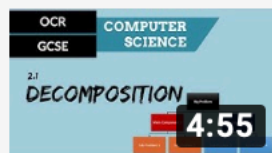
Videos to watch

Watch the CraigNDave videos on YouTube about Problem Solving and Algorithms



OCR GCSE 2.1 Abstraction

craigndave



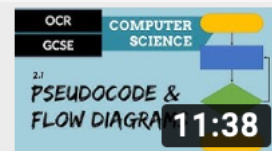
OCR GCSE 2.1 Decomposition

craigndave



OCR GCSE 2.1 Algorithmic thinking

craigndave



OCR GCSE 2.1 How to produce algorithms using pseudocode and flow...

craigndave

Make whatever notes you need whilst watching the videos. These will help you complete the next tasks.

4.4 Tasks to complete

Algorithms are used to design and plan solutions. The next topic is programming and you will be required to create a simple game. Therefore, in this task you are going to prepare for this by planning and designing what the game should do.

The game that you are going to create is a computerised version of the game “Hangman”.

Main Task

If we decompose the game of hangman, and break it down into parts, it would be fair to say that the main stages to a game could be written as follows-

- A simple menu for starting the game (main_menu)
- The computer selects a random word from a list or file (get_word)

- The computer displays the blank spaces for the word (display_word)
- The user makes guesses (make_guess)
- The computer checks guesses and takes appropriate actions (check_guess)

You need to think about each of these functions/stages and plan how a program could perform each of these stages.

You can write your plan in the form of a **flowchart** or **pseudocode** algorithm.

Function – main_menu()

Write your pseudocode or flowchart here

Function – get_word()

Function – display_word()

Function – make_guess()

Function – check_guess()

5. Programming

A Level Specification Points	9 - 1 GCSE
<p>2.2.1 Programming techniques</p> <p>(a) Programming constructs: sequence, iteration, branching.</p> <p>(b) Recursion, how it can be used and compares to an iterative approach.</p> <p>(c) Global and local variables.</p> <p>(d) Modularity, functions and procedures, parameter passing by value and by reference.</p> <p>(e) Use of an IDE to develop/debug a program.</p> <p>(f) Use of object oriented techniques.</p>	<p>2.2 Programming techniques</p> <ul style="list-style-type: none">the use of variables, constants, operators, inputs, outputs and assignmentsthe use of the three basic programming constructs used to control the flow of a program:<ul style="list-style-type: none">sequenceselectioniteration (count and condition controlled loops)the use of basic string manipulationthe use of basic file handling operations

Summary

At GCSE level you are required to learn a suitable programming language to a introductory level. Many students, however, greatly enjoy the programming element of the course, and will have developed their programming skills beyond those needed for the GCSE course.

It is essential that you have a very good understanding of a programming language to a GCSE level, as a minimum. You will be able to judge whether or not you are at that level, by completing this mini project, creating a computerised version of hangman.

This mini-project is similar in complexity to the GCSE coursework projects, and a student considering Computer Science at A Level should be able to approach this work independently. You are NOT expected to be able to do it from memory, and you are encouraged to use the help, websites and videos available.

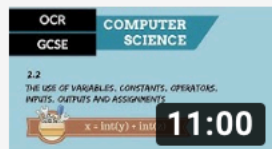
The project covers most of the key techniques identified in the table above.

Before you begin: Choose your programming language

Many of you will have learnt to program in Python and this is the language that I would encourage to use as we will use it throughout the A Level course and for you're a Level coursework project.

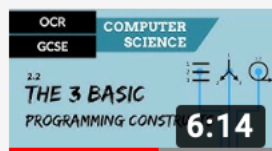
Part 1 - Practice and Develop your programming skills

If you need to, watch these videos on the basic programming techniques required.



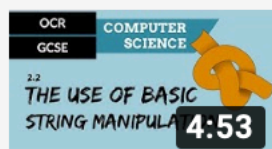
OCR GCSE 2.2 The use of variables, constants, operators, inputs, outputs...

craigndave



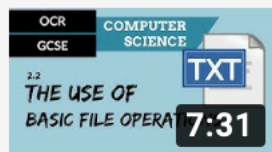
OCR GCSE 2.2 The use of the three basic programming constructs

craigndave



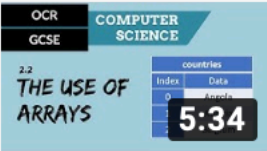
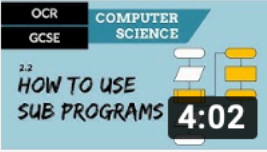
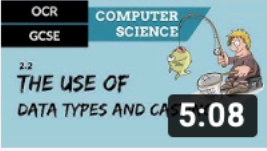
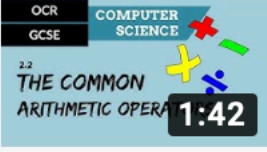
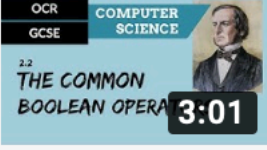
OCR GCSE 2.2 The use of basic string manipulation

craigndave



OCR GCSE 2.2 The use of basic file handling operations

craigndave

 <p>OCR GCSE 2.2 COMPUTER SCIENCE THE USE OF ARRAYS 5:34</p>	OCR GCSE 2.2 The use of arrays craigndave
 <p>OCR GCSE 2.2 COMPUTER SCIENCE HOW TO USE SUB PROGRAMS 4:02</p>	OCR GCSE 2.2 How to use sub programs craigndave
 <p>OCR GCSE 2.2 COMPUTER SCIENCE THE USE OF DATA TYPES AND CASTING 5:08</p>	OCR GCSE 2.2 The use of data types and casting craigndave
 <p>OCR GCSE 2.2 COMPUTER SCIENCE THE COMMON ARITHMETIC OPERATORS 1:42</p>	OCR GCSE 2.2 The common arithmetic operators craigndave
 <p>OCR GCSE 2.2 COMPUTER SCIENCE THE COMMON BOOLEAN OPERATORS 3:01</p>	OCR GCSE 2.2 The common Boolean operators craigndave

You can find brilliant help and examples for all programming languages on the W3Schools website – www.w3schools.com/python

You can also practice the basics by completing the activities on this website <https://www.learnpython.org>

Part 2 – Create the game of Hangman

Once you have practiced the basic commands and have a good understanding of Python, you should then begin to make your version of the game of Hangman.

Get yourself organized and create a folder for the project. Make sure you save backups/version of your game just in case you break something when adding code.

How do you make the game?

You should build the game up in stages. Use the plan from the previous “algorithm” section. Work through each of the functions in order, building up the game gradually. If you get stuck on one section, it can be possible to remove it and carry on eg. If you can't get the game to pull a random word from a file, just set it to a fixed word.

What do you need to hand in and when?

Evidence of creating the game

Practicing the Python skills and then creating the game is going to take you many hours, spread over a few weeks. Some of you may complete it quick than others, but you should try and stick to the guidelines below.

Week 1 – Basic Skills Practice and Preparation

Watch the videos and recap on your Python skills. Go to the website and have a go at some of the activities. Look back at some of your own programs and remind yourself of what is possible in Python.

Create some simple programs that cover the basic skills. Simple practice programs could include-

- A multi-function calculator
- A simple THREE question, text-based quiz. Answers could be multiple choice.

Evidence to hand-in – Screenshots of code or website progress.

Week 2 – Create the game Part 1.

Start making your game. You should be able to create the first few stages of the game.

- Main Menu
- Getting the word
- Displaying the blanks for the word

Evidence to hand-in – Screenshots of code

Week 3 – Create the game Part 2.

Start making your game. You should be able to create the first few stages of the game.

- Make a guess
- Check a guess
- Checking if the game is won or lost

Evidence to hand-in – Screenshots of code

Further Extensions – There are many possible additions that could be made to the game to improve it. Add your own features and evidence them.