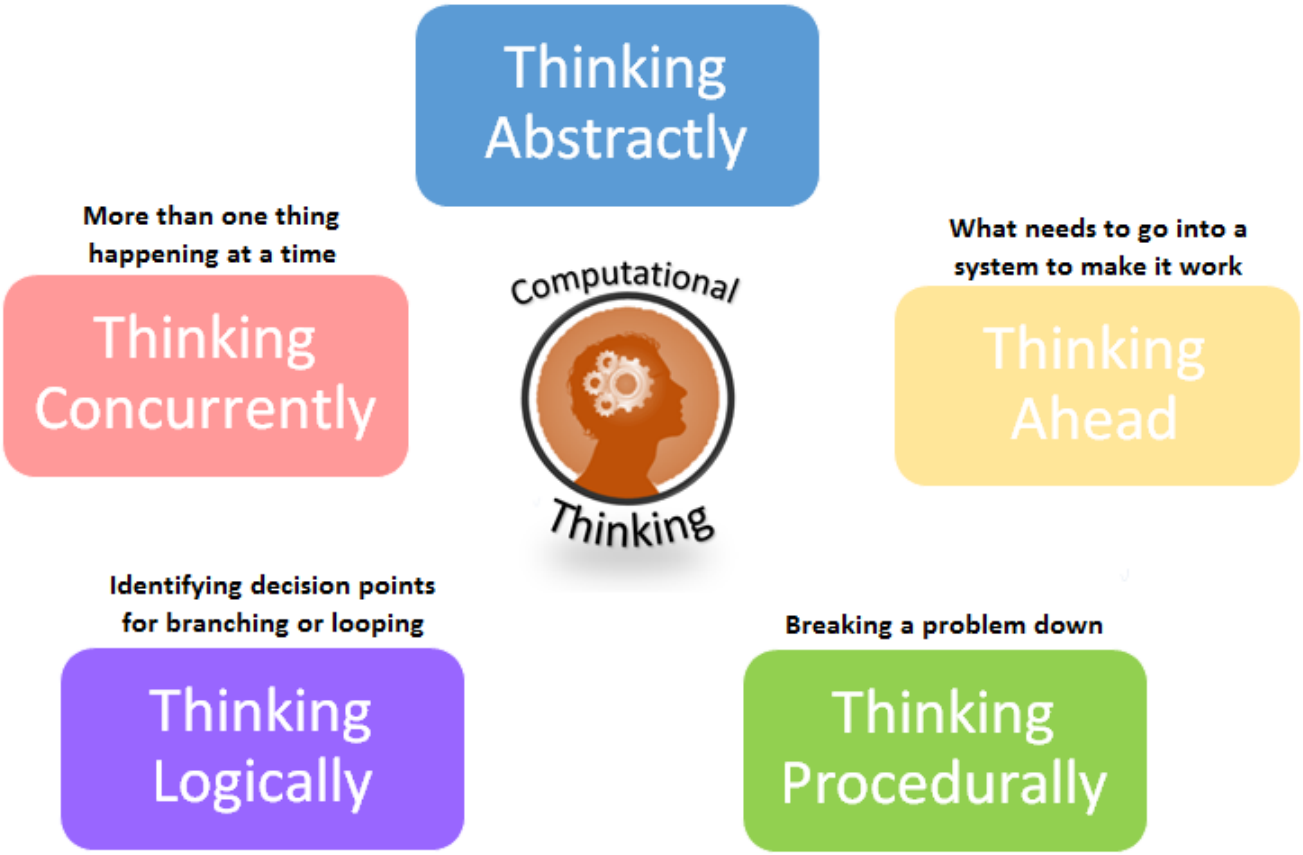
2.1. Elements of computational thinking	
Understand what is meant by computational thinking	
2.1.1 Thinking abstractly	a) The nature of abstraction. b) The need for abstraction. c) The differences between an abstraction and reality. d) Devise an abstract model for a variety of situations.
2.1.2 Thinking ahead	a) Identify the inputs and outputs for a given situation. b) Determine the preconditions for devising a solution to a problem. c) The need for reusable program components.
2.1.3 Thinking procedurally	a) Identify the components of a problem. b) Identify the components of a solution to a problem. c) Determine the order of the steps needed to solve a problem. d) Identify sub-procedures necessary to solve a problem.
2.1.4 Thinking logically	a) Identify the points in a solution where a decision has to be taken. b) Determine the logical conditions that affect the outcome of a decision. c) Determine how decisions affect flow through a program.

Representing a system by removing unnecessary details



[redacted]

2.1.1 Thinking abstractly

Abstraction is when you represent something by removing unnecessary details.

The nature of abstraction

Abstraction is the ability to filter the details that we do not need out of a problem so that we can create a more general idea of what the problem is. We may then be able to go on and apply the same problem solving procedure to other problems. For instance, you could give a list of each London underground station and which other stations they are connected to, or you could produce a map like the London Underground tube map that abstracts the problem, making it easier to understand. Since this was so famous, this is now a common 'template' to use for other underground mapping systems.

Why is abstraction needed?

Abstraction is required to enable complex procedures or events to be modelled. Computer programs cannot account for every single element of an event as this would be extremely complex and time consuming, so therefore they need to remove unnecessary detail.

"Abstraction" guidance from OCR exam board:

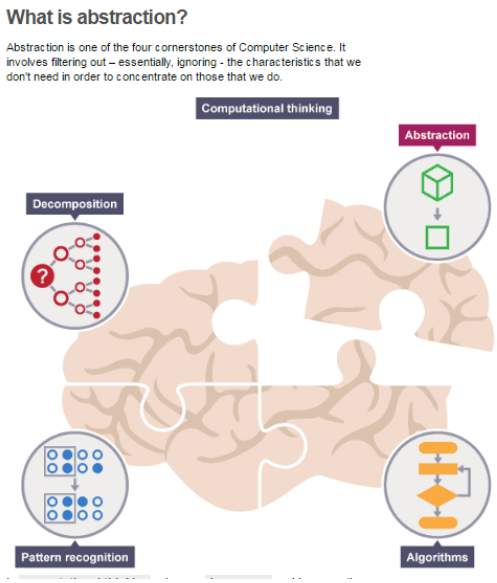
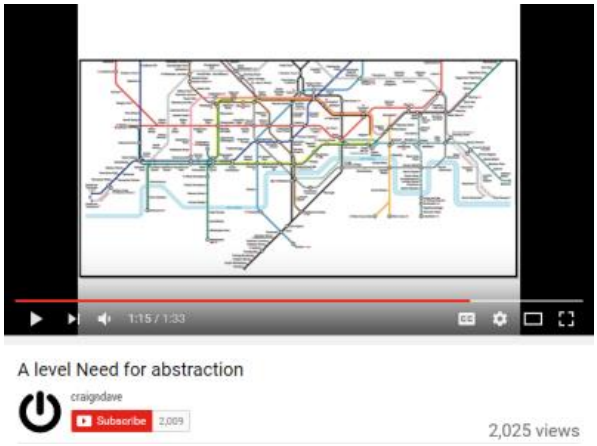
Abstraction is initially a difficult concept to understand as there are various layers of abstraction dependent on how much detail you need a user to have access to.

Most students have some form of smartphone or mobile device. This device can perform a vast range of functions and processes that can be easily controlled by the user via simple hand gestures, finger swipes or pressing a button. To open an app you press the icon, to take a photograph you press the camera button, to listen to music you select the track. However, underneath each button press there is a very complex set of instructions and procedures that take place to enable, say, the camera. Abstraction enables us to take these complex processes and hide them from the user. Abstraction removes or hides these details from the view of the user to make the process appear simple. Consider a simple stop sign near a junction. Abstraction means that you see the sign and stop, the desired outcome. The reality of what is happening is very different; you see the sign, your brain recognises the shape and the colour and compares this with the knowledge you already have that identifies it as a stop sign, and you receive a message that you should stop the car. There are many layers of abstraction depending on the level of detail you require. Consider the stop sign again. The light is reflected off the sign which is received as photons onto your eyes. The cones in your eyes respond by interpreting the light and colours. This information is sent via the nerve to the brain. In this example, abstraction enables this superficial detail to be hidden or removed from the user.

A common misconception is that abstraction removes detail. In a way it does but it removes the unnecessary detail from the user's experience, making it more accessible. In some instances the level of abstraction is reduced, as more detail is added to improve the model. Consider an old computer game where the player can only move up, down, left or right. Current game characters can jump, lean, swing and their individual joints move. In games, physics allows for more realistic modelling of interactions such as water, flight and properties of various materials and natural elements. In these examples more detail is being added to make the experience closer to the reality.

Task 1 – Make your own extra notes on Abstraction.

Use these four resources:

http://www.bbc.co.uk/education/guides/zttrcdm/revision **BBC website**  	https://www.youtube.com/watch?v=4Ydk3IMyn7g **CraignDave "A Level Need for abstraction"** 1.5 mins 
(C) The difference between abstraction and reality. **CraignDave "A Level Abstraction & reality"**  	https://www.youtube.com/watch?v=p7nGcY73epw **Abstraction examples from Computerphile** 5 min video 

Task 2 - Worked Question / Scenario

X-Plane is a flight simulator progam designed to allow trainee pilots to experience flying. The simulator has real models of airports to allow pilots to practise taking off and landing in life-life situations. The developers study real airports/runways/landscapes and then they use abstraction on their findings before designing a virtual environment.

1/ Why will the programmers use abstraction when they developing the flight simulator?

In order to remove any unnecessary details that would make the program too complex to understand or code. The simulator should test the flying abilities in realistic way but complicated unnecessary details that may b prevalent in real life such as natural disasters, birds flying, strikes can be left out of the program.

2/ As a result of abstraction there will be similarities and differences between the virtual and real airport landscape and runways etc.
State what <u>similarities</u> there will be and why.

Other aircraft using the runway because this happens in reality and it is necessary to learn how to fly with other air traffic as in real life.

Length of runway, the piolet must be able to testy their abilities of taking off and landing from a realistic runway of an appropriate length (i.e. not too long or short).

Weather on the runway may be shown as in reality weather will change vastly and different conditions affect flying ability.

3/ State what <u>differences</u> there will be and why.

The manufactures of the planes – these are unnecessary for the training

Any events such as strikes/natural disasters that may disrupt flying since this would be unnecessary and inconvenient in the program.

Loading of the luggage and people onto the plane or any delays– this is unnecessary and would waste the user' time.

Examples of abstraction - NOTES

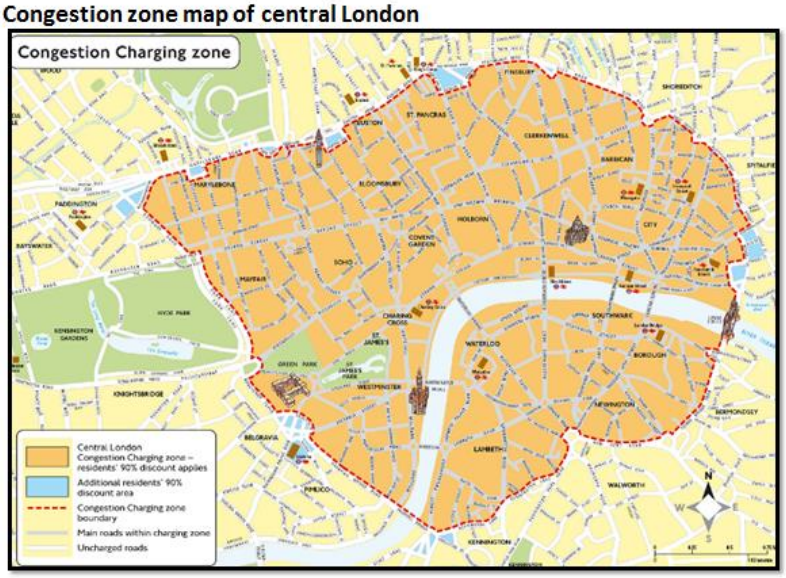
A-Z Travel map of London for people travelling by car

How has abstraction been used in this situation and why was it needed?

- *Different classification of roads coloured (M, A, B etc.)*
- *Road names labelled in boxes*
- *Major areas written in CAPITALS*
- *Minor areas written in lower case*
- *Most other features removed*
- *Grid lines added for easy reference*

This has all been done to making planning and following a route via car as easy as possible.

Congestion zone map of central London

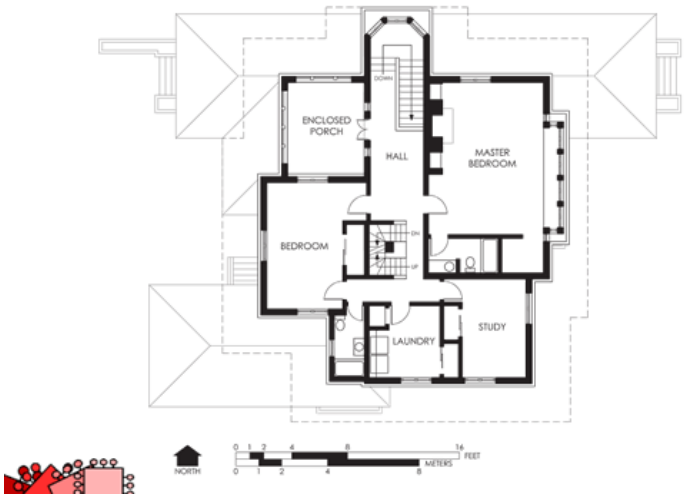


How has abstraction been used in this situation and why was it needed?

- *Key added*
- *Colour shading for various zones*
- *Red dotted border added to show boundary of congestion charging zone*
- *Parks and river highlighted*
- *Major roads shown and labelled*
- *Icons used to show tube and train stations*
- *Major land marks shown with graphics*

This map has been generated with a the sole purpose of clearly delimiting the boundaries of the London congestion charging zone and to make sure this is as clear as possible.

A house plan needs to be drawn up by a structural engineer to provide to a planning officer. How might abstraction be used by the engineer when producing the drawing?



- *Top down view used*
- *Scale added*
- *Most internal features removed*
- *Structural elements, such as main walls the feature, bold thick black lines*
- *Location of windows clearly shown*
- *All doors shown along with direction of opening*
- *Rooms labelled*
- *Stairs clearly shown with direction*