3.3 Fundamentals of data representation

3.3.1 Number bases

Content	Additional information	Chk
Understand the following number bases: ● **decimal** (base 10) ● **binary** (base 2) ● **hexadecimal** (base 16).		
Understand that computers use binary to **represent** all data and instructions.	Students should be familiar with the idea that a **bit pattern** could represent different types of data including text, image, sound and integer.	
Explain why hexadecimal is often used in computer science.		

3.3.2 Converting between number bases

Content	Additional information	Chk
Understand how binary can be used to represent whole numbers.	Students must be able to represent decimal values between 0 and 255 in binary.	
Understand how hexadecimal can be used to represent whole numbers.	Students must be able to represent decimal values between 0 and 255 in hexadecimal.	
Be able to convert in both directions between: ● binary and decimal ● binary and hexadecimal ● decimal and hexadecimal.	The following equivalent maximum values will be used: ● decimal: 255 ● binary: 1111 1111 ● hexadecimal: FF	

3.3.3 Units of information

Content	Additional information	Chk
Know that: ● a **bit** is the fundamental unit of information ● a **byte** is a group of 8 bits.	A bit is either a 0 or a 1. ● b represents bit ● B represents byte	
Know that quantities of bytes can be described using prefixes. Know the names, symbols and corresponding values for the decimal prefixes: ● kilo, 1 kB is 1,000 bytes ● mega, 1 MB is 1,000 kilobytes ● giga, 1 GB is 1,000 Megabytes ● tera, 1 TB is 1,000 Gigabytes.	Students might benefit from knowing that historically the terms kilobyte, megabyte, etc have often been used to represent powers of 2. The SI units of kilo, mega and so forth refer to values based on powers of 10.	

3.3.4 Binary arithmetic

Content	Additional information	Chk
Be able to add together up to three binary numbers.	Students will need to be able to add together up to three binary numbers using a maximum of 8 bits per number. Students will only be expected to add together a maximum of three 1s in a single column. Answers will be a maximum of 8 bits in length and will not involve carrying beyond the eight bits.	
Be able to apply a **binary shift** to a binary number.	Students will be expected to use a maximum of 8 bits. Students will be expected to understand and use only a logical binary shift. Students will not need to understand or use fractional representations.	
Describe situations where binary shifts can be used.	Binary shifts can be used to perform simple multiplication/division by powers of 2.	

3.3.5 Character encoding

Content	Additional information	Chk
Understand what a **character set** is and be able to describe the following character encoding methods: ● **7-bit ASCII** ● **Unicode**	Students should be able to use a given character encoding table to: ● convert characters to character codes ● convert character codes to characters	
Understand that **character codes** are commonly grouped and run in sequence within encoding tables.	Students should know that character codes are grouped and that they run in sequence. For example in ASCII 'A' is coded as 65, 'B' as 66, and so on, meaning that the codes for the other capital letters can be calculated once the code for 'A' is known. This pattern also applies to other groupings such as lower case letters and digits.	
Describe the purpose of Unicode and the advantages of Unicode over ASCII. Know that Unicode uses the same codes as ASCII up to 127.	Students should be able to explain the need for data representation of different alphabets and of special symbols allowing a far greater range of characters. It is not necessary to be familiar with UTF-8, UTF-16 or other different versions of Unicode.	

3.3.6 Representing images

Content	Additional information	Chk
Understand what a **pixel** is and be able to describe how pixels relate to an image and the way images are displayed.	Students should know that the term pixel is short for Picture Element. A pixel is a single point in a graphical image. VDUs display pictures by dividing the display screen into thousands (or millions) of pixels, arranged into rows and columns.	
Describe the following for bitmaps: • **image size** in pixels • **colour depth**. Know that the size of a bitmap image in pixels (width x height)	The size of an image is expressed directly as width of image in pixels by height of image in pixels using the notation width x height. Colour depth is the number of bits used to represent each pixel.	
Describe how a **bitmap** represents an image using pixels and colour depth.	Students should be able to explain how bitmaps are made from pixels.	
Describe using examples how the number of pixels and colour depth can affect the file size of a bitmap image.	Students should be able to describe how higher numbers of pixels and higher colour depths can affect file size and should also be able to use examples.	
Calculate bitmap image file sizes based on the number of pixels and colour depth.	Students only need to use colour depth and number of pixels within their calculations. Size (bits) = W x H x D Size (bytes) = (W x H x D)/8 W = image width H = image height D = colour depth in bits.	
Convert binary data into a bitmap image.	Given a binary pattern that represents a simple bitmap, students should be able to draw the resulting image as a series of pixels.	
Convert a bitmap image into binary data.	Given a simple bitmap, students should be able to write down a bit pattern that represents the image.	

3.3.7 Representing sound

Content	Additional information	Chk
Understand that sound is **analogue** and that it must be converted to a **digital form** for storage and processing.		
Understand that **analogue signals** are **sampled** to create the digital version of sound.	Understand that a sample is a measure of amplitude at a point in time.	
Describe the **digital representation** of sound in terms of: • **sampling rate** • **sample resolution**.	Sampling rate is the number of samples taken in a second and is usually measured in hertz (1 Hertz = 1 sample per second). Sample resolution is the number of bits per sample.	
Calculate sound file sizes based on the sampling rate and the sample resolution.	File size (bits) = rate x res x secs rate = sampling rate res = sample resolution secs = number of seconds	

3.3.8 Data compression

Content	Additional information	Chk
Explain what **data compression** is. Understand why data may be compressed and that there are different ways to compress data.	Students should understand that it is common for data to be compressed and should be able to explain why it may be necessary or desirable to compress data.	
Explain how data can be compressed using **Huffman coding**. Be able to interpret **Huffman trees**.	Students should be familiar with the process of using a tree to represent the Huffman code.	
Be able to calculate the number of bits required to store a piece of data compressed using Huffman coding. Be able to calculate the number of bits required to store a piece of uncompressed data in ASCII.	Students should be familiar with carrying out calculations to determine the number of bits saved by compressing a piece of data using Huffman coding.	
Explain how data can be compressed using **run length encoding** (RLE).	Students should be familiar with the process of using frequency/data pairs to reduce the amount of data stored.	
Represent data in RLE frequency/data pairs.	Students could be given a bitmap representation and they would be expected to show the frequency and value pairs for each row, eg 0000011100000011 would become 5 0 3 1 6 0 2 1.	