Syllabus Content:

1.2 Images, sound and compression

- Understand how and why a computer represents an image, including the effects of the resolution and colour depth
- An image is a series of pixels that are converted to binary, which is processed by a computer
- The resolution is the number of pixels in the image
- The colour depth is the number of bits used to represent each colour
- The file size and quality of the image increases as the resolution and colour depth increase
- Understand how and why a computer represents sound, including the effects of the sample rate and sample resolution
- A sound wave is sampled for sound to be converted to binary, which is processed by a computer
- The sample rate is the number of samples taken in a second
- The sample resolution is the number of bits per sample
- The accuracy of the recording and the file size increases as the sample rate and resolution increase
- Understand the purpose of and need for data compression
- Compression exists to reduce the size of the file
- The impact of this is, e.g.: – less bandwidth required – less storage space required – shorter transmission time
- Lossless compression reduces the file size without permanent loss of data, e.g. run length encoding (RLE)
- Lossy compression reduces the file size by permanently removing data, e.g. reducing resolution or colour depth, reducing sample rate or resolution

Graphic images that have been processed by a computer can usually be divided into two distinct categories.

Pixels vs. Vector

Such images are either bitmap files or vector graphics. This section will cover the two main image types: **vector** and **bitmap(pixels)**, along with some compression techniques.

Pixel (Picture element):

The smallest possible addressable area defined by a solid color, represented as binary, in an image.

Graphics on a screen are made up of tiny blocks of pixels. The more pixels on the screen, the higher the resolution and the better the quality of the picture will be.

The higher the image resolution, the more memory is needed to store the graphic.

Image files can be either **bitmaps** or **vectors**.

Bitmap Graphics:

Bitmaps images are exactly what their name says they are: a collection of bits that form an image.

  🟢 **Bitmap images** (or raster) images are stored as a series of tiny dots called pixels.

- Each pixel is actually a very small square that is assigned a color, and then arranged in a pattern to form the image.
- The image consists of a matrix of individual dots (or pixels (picture element)) that all have their own color (described using bits, the smallest possible units of information for a computer).
- Bitmap images are organised as a grid of coloured squares called pixels (short for 'picture elements').
- When zooming in or enlarging a bitmap image, the pixels are stretched and made into larger blocks. This is why bitmap images appear as poor quality when enlarged too much.
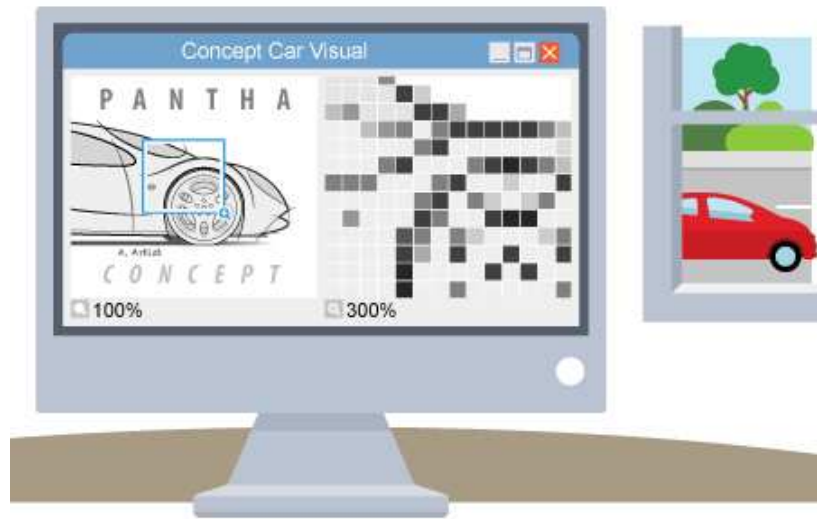
Each colour of an image is stored as a binary number. In the black-and-white image below, each pixel is either black or white. You need a binary value for each different colour.As each pixel is either black or white, this image can be encoded with a value of 0 for white and 1 for black.
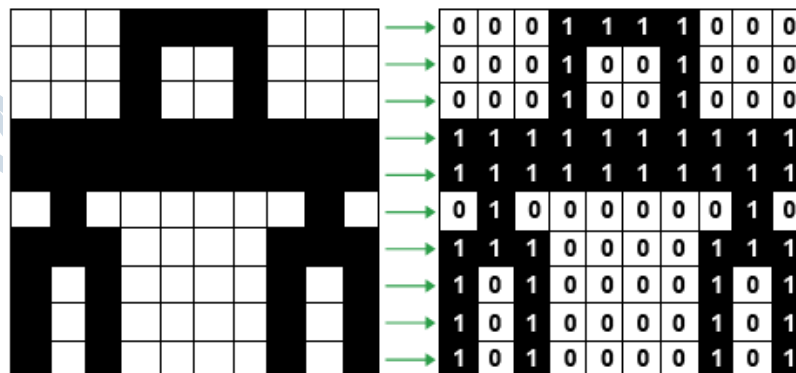
When you zoom in on a bitmap image you can see the individual pixels that make up that image. Bitmap graphics can be edited by erasing or changing the color of individual pixels using a program such as Adobe Photoshop.
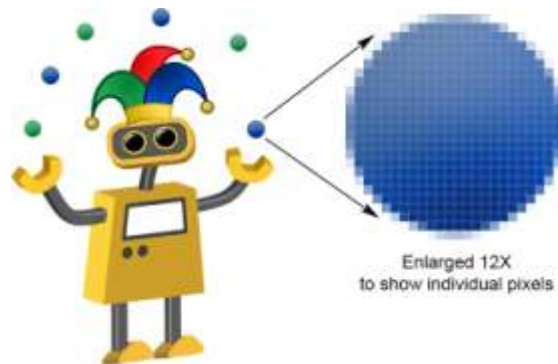
Enlarged 12X
to show individual pixels

Pixel (Picture element):
The smallest possible addressable area defined by a solid color, represented as binary, in an image.

File header:
BMP, PCX, JPEG, FLI/FLC, and AVI files include headers that define the image size, number of colors, and other information needed to display the image. Fastgraph provides functions for reading the image headers and retrieving their more useful items. However, there may be times when you need additional information stored in the file header e.g.

- Confirmation that the file is a BMP
- File size
- Location/offset of image data within the file
- Dimensions of the image (in pixels) // image resolution
- Colour depth (bits per pixel, 1, 4, 8, 16, 24 or 32)
- Type of compression used, if any

Resolution:

Screen Resolution - The screen resolution tells you how many pixels your
screen can display horizontally and vertically. It's written in the form 1024 x 768. In this example, the screen can show 1,024 pixels horizontally, and 768 vertically:

1,024 pixels
768 pixels
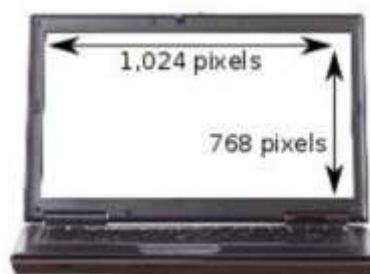
Image Resolution - how many pixels do an image contains per inch/cm?
Measurements of **dots per inch (dpi)** are called image resolution.

- The number of pixels per unit measurement
- or the number of pixels in an image
- or The number of pixels wide by the number of pixels high
- or number of pixels per row by the number of rows

Resolution refers to the number of pixels in an **image**. Resolution is sometimes identified by the width and height of the **image** as well as the total number of pixels in the **image**. For example, an **image** that is 2048 pixels wide and 1536 pixels high (2048 x 1536) contains (multiply) 3,145,728 pixels (or 3.1 Megapixels).

The higher the resolution, the more pixels are available. Therefore the crisper the picture

4x6 = 1200 x 1800 pixels at 300dpi

Same photo printed at a larger size or lower resolution
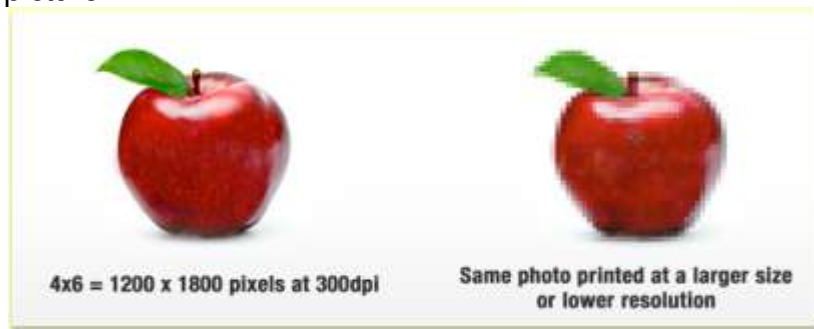
Colour depth

The colour depth of an image is measured in bits. The number of bits indicates how many colours are available for each pixel. In the black and white image, only two colours are needed. This means it has a colour depth of 1 bit.
A 2-bit colour depth would allow four different values: 00, 01, 10, 11. This would allow for a range of colours such as:

Binary code	Colour
00	White
01	Light grey
10	Dark grey
11	Black

Greater the colour depth (bits per pixel), the more colours are available.

Colour depth	Available colours
1-bit	$2^1 = 2$
2-bit	$2^2 = 4$
3-bit	$2^3 = 8$
4-bit	$2^4 = 16$
5-bit	$2^5 = 32$
6-bit	$2^6 = 64$
7-bit	$2^7 = 128$
8-bit	$2^8 = 256$

Example: Calculating screen resolutions & Picture file size:

Using the diagram above we are going to work out how many pixels are required to display a single frame on a **VGA** screen.

Checking the resolution:
Height = 480 Width = 640
Area = Width x Height = Total Pixels   Area = 640 **x** 480 = 307200 pixels
Hence, it takes 307,200 pixels to display on a VGA screen!

File size = Total Pixels X Colour Depth

Coloured Images:
- At least **eight bits per pixel are necessary to code a coloured image.** The number of bits per pixel is sometimes referred to as the **colour depth**.
  (8 bits or 1 byte for 1 pixel in coloured image) (Computer Science Book)

Hence, it takes 307,200 pixels X 8 bits = 2457600 bits or 2457600/8 = 307200 bytes or 300KB.

Monochrome or Black & White image:

The simplest option is to use **one bit** to represent the **colour**, so that the pixel is either black or white. **(1 bit for 1 pixel in Black & White or monochrome Picture)**

Greyscale image:

🔖 Storage of the colour in **four bits** would allow simple **greyscale colouring.**

(4 bits for 1 pixel in GreyScale image)

Greyscale

Coloured Images:

🔖 At least **eight bits per pixel are necessary to code a coloured image.** The number of bits per pixel is sometimes referred to as the **colour depth**.
(8 bits or 1 byte for 1 pixel in coloured image) (Computer Science Book)

🔖 **24 bits** almost always uses **8 bits of each of RGB(Red:8 Green:8 Blue:8) 8X8X8 = 24**

🔖 As of 2018 **24-bit** color depth is used by virtually every computer and phone display and the vast majority of image storage formats. (Wikipedia)

(1.1.3) Sound:

Sound is an oscillation of pressure transmitted through a solid, liquid, or gas (there is no sound in outer space as space is a vacuum and there is no solid, liquid or gas to transmit sound through!). A speaker works by moving its center cone in and out; this causes the air particles to bunch together forming waves. These waves spread out from the speaker travelling at 340 m/s. If your ear is in the way, then the waves of sound particles will collide with your ear drum, vibrating it and sending a message to your brain. This is how you hear:
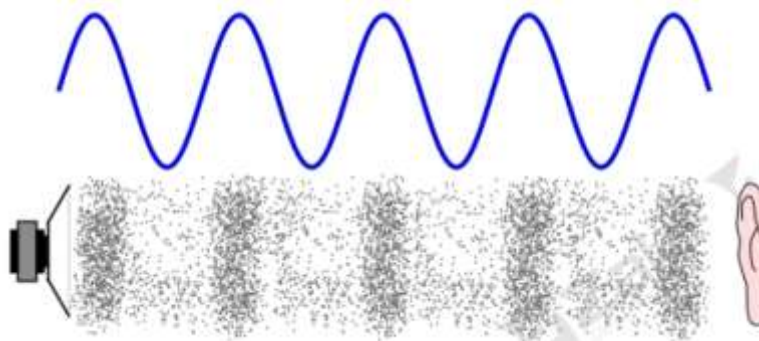
Natural sound consists of variations in pressure which is detected by the human ear. A typical sound contains a large number of individual waves each with a defined frequency. The result is a wave form in which the amplitude of the sound varies in a continuous but irregular pattern.

If there is a need to store sound or transmit it electronically the original analogue sound signal has to be converted to a binary code.

A sound encoder has two components.

1. Band limiting Filter
2. Analogue to digital converter.

Band-limiting filter:

This is needed to remove high-frequency components. The ear would not be able to detect these and they could cause problems for the coding if not removed.

Analogue-to-digital converter (ADC).

The method of operation of the ADC is described with reference to Figure. The amplitude of the wave (the red line) has to be sampled at regular intervals. The blue vertical lines indicate the sampling times. The amplitude cannot be measured exactly; instead the amplitude is approximated by the closest of the defined amplitudes represented by the horizontal lines. In Figure 1.08, sample values 1 and 4 will be an accurate estimate of the actual amplitude because the wave is touching an amplitude line. In contrast, samples 5 and 6 will not be accurate because the actual amplitude is approximately half way between the two closest defined values.

Figure 1.08 ADC sampling
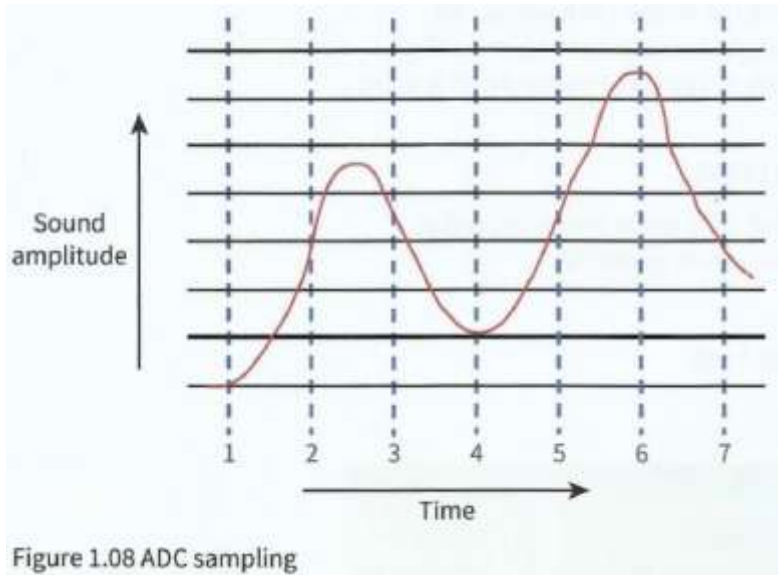
Analogue and digital:

For a computer to store sound files we need to get the continuous analogue sound waves into discrete binary values:
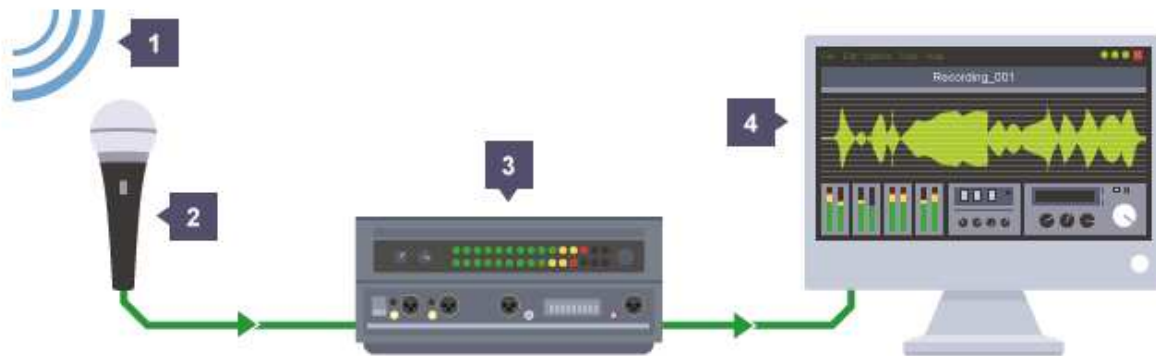
1	Microphone measures change in air pressure
2	Microphone translates air pressure into electrical voltage
3	Analogue to Digital Converter digitises the electrical voltage to bytes of information
4	Computer displays the digitised sound for manipulation

Sound input through a microphone is converted to digital for storage and manipulation

An analogue sound wave is picked up by a microphone and sent to an Analogue to Digital (ADC) converter in the form of analogue electrical signals. The ADC converts the electrical signals into digital values which can be stored on a computer.

Once in a digital format you can edit sounds with programs such as **audacity**.
To play digital audio you convert the sound from digital values into analogue electrical signals using the

DAC, these signals are then passed to a speaker that vibrating the speaker cone, moving the air to create sound waves and analogue noise.

Digital audio: Sounds created on a computer exist as digital information encoded as audio files.
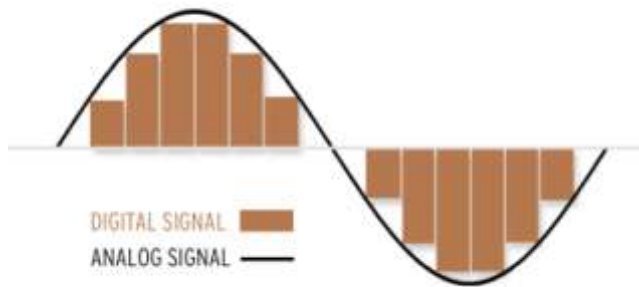

DIGITAL SIGNAL ▬
ANALOG SIGNAL ▬

Analogue to Digital Converter (ADC)
Converts analogue sound into digital signals that can be stored on a computer.

Digital to Analogue Converter (DAC)
Converts digital signals stored on a computer into analogue sound that can be played through devices such as speakers.

This section will cover how we record, store and transmit sound using computers.

Sound waves in nature are **continuous**, this means they have an almost infinite amount of detail that you could store for even the shortest sound. This makes them very difficult to record perfectly, as computers can only store **discrete** data, data that has a limited number of data points

Bandwidth:
Bandwidth is also the amount of data that can be transmitted in a fixed amount of time. For digital devices, the **bandwidth** is usually expressed in bits per second(bps) or bytes per second. For analog devices, the **bandwidth** is expressed in cycles per second, or Hertz (Hz).

Digital audio quality Factors that affect the quality of digital audio include:

- **sample rate** - the number of audio samples captured every second
- **sample resolution/ bit depth:** - the number of bits assigned to each sample of sound
- **bit rate** - the number of bits used to store one second of sound

Sound Sampling:
- amplitude of sound wave taken at different points in time
- measurement of value of analogue signal at regular time intervals/a point in time
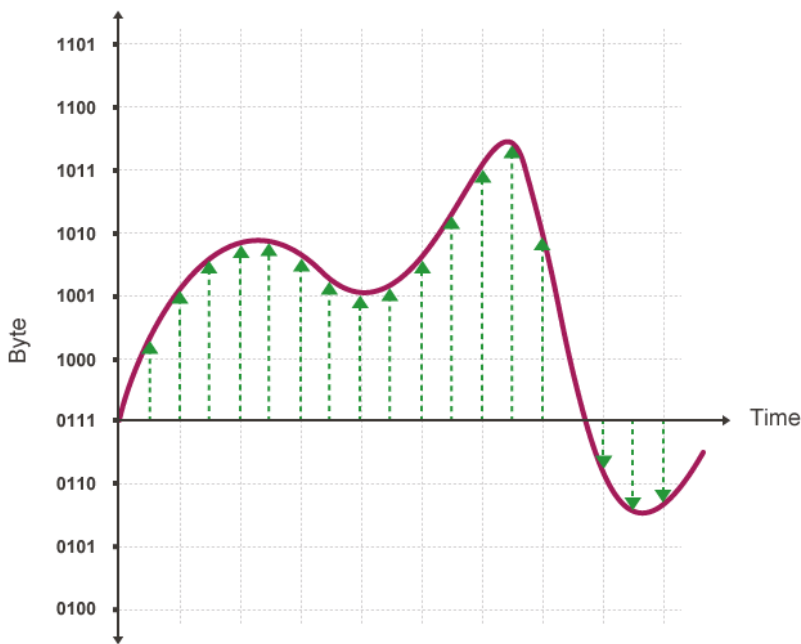
Measurement of amplitude   Sound wave

Sample rate

The **sample rate** is how many samples, or measurements, of the sound are taken each second.

The more samples that are taken, the more detail about where the waves rise and fall is recorded and the higher the quality of the audio. Also, the shape of the sound wave is captured more accurately.

Each sample represents the amplitude of the digital signal at a specific point in time. The amplitude is stored as either an **integer** or a **floating point number** and encoded as a **binary number**.
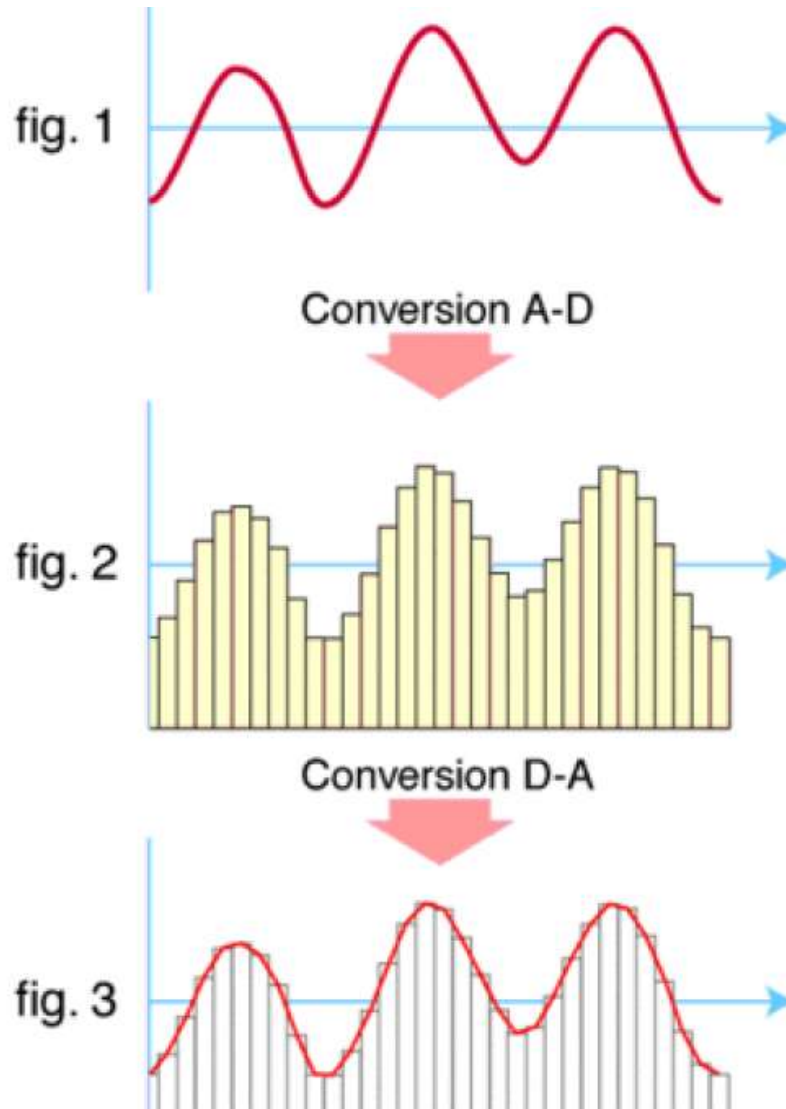
1. The original analogue sound wave is a continuous set of points
2. ADC converts sound into digital data
3. DAC converts digital data into analogue sound, the analogue wave produced may differ significantly from the original sound wave.

Sound Sampling

So we should know by now that sound waves are continuous and computers can only store discrete data. How exactly does an Analogue to Digital Converter convert a continuous sound wave into discrete digital data? To do this we need to look at how computers sample sound.

Original Sound	High sample rate	1/2 high sample rate	1/3 high sample rate	1/4 high sample rate
original continuous sound wave	digital looks like original	digital loses sharpness	loss of peaks	poor resemblance to original!
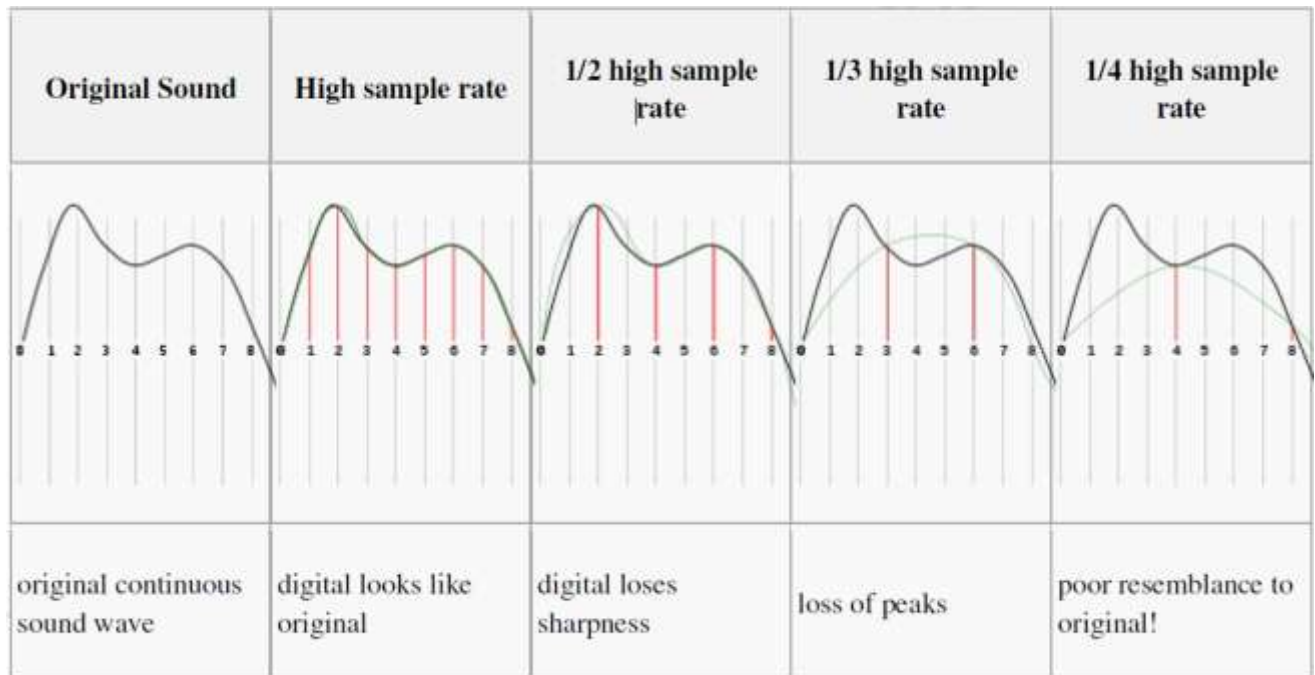
Sampling Rate - The number of samples taken per second

A higher sampling rate means the waveform will be converted from analog to digital form more accurately.

Sampling Resolution (bit depth): number of bits assigned to each sample.
The sampling resolution allows you to set the range of volumes storable for each sample.

The quality of a sound produced using a sound sampler depends on the sampling rate and sampling resolution.

🔰 The higher the sampling resolution (bit depth), the higher the quality of the audio.
🔰 Higher sampling rate/resolution means larger file size

Bit depth is the number of bits available for each sample.
Sampling resolution (Bit depth) is usually 16 bits on a CD and 24 bits on a DVD.

Bit Rate: number of bits required to store 1 second of sound.

$$Bit\ Rate = Sampling\ Rate \times Sampling\ Resolution$$
$$File\ Size = Bit\ Rate \times Length\ of\ Sound$$

File Size = Sample Rate X Sample Resolution X Length of sound

Calculating bit rate

The bit rate is calculated using the formula:

Frequency × bit depth × channels = bit rate

A typical, uncompressed high-quality audio file has a sample rate of 44,100 samples per second, a sample resolution (bit depth) of 16 bits per sample and 2 channels of stereo audio. The bit rate for this file would be:

44,100 samples per second × 16 bits per sample × 2 channels = 1,411,200 bits per second (or 1,411.2 kbps)

A four-minute (240 second) song at this bit rate would create a file size of:

14,411,200 × 240 = 338,688,000 bits (or 40.37 megabytes)
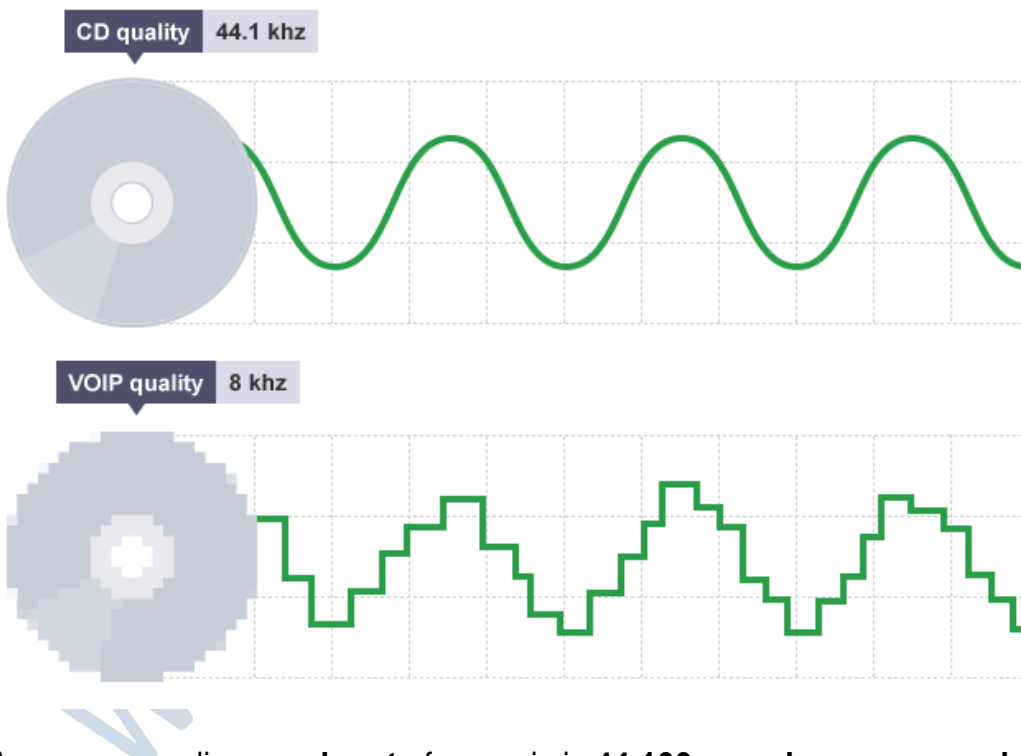
A common audio **sample rate** for music is **44,100 samples per second**. The unit for the sample rate is hertz (Hz). 44,100 samples per second is 44,100 hertz or 44.1 kilohertz (kHz).

Telephone networks and VOIP services can use a sample rate as low as 8 kHz. This uses less data to represent the audio. At 8 kHz, the human voice can still be heard clearly - but music at this sample rate would sound low quality.

1.3 Compression

Compression is a useful tool for reducing file sizes. When images, sounds or videos are compressed, data is removed to reduce the file size. This is very helpful when streaming and downloadingfiles.

Streamed music and downloadable files, such as MP3s, are usually between 128 kbps and 320 kbps - much lower than the 1,411 kbps of an uncompressed file.

Videos are also compressed when they are streamed over a network. Streaming HD video requires a high-speed internet connection. Without it, the user would experience buffering and regular drops in quality. HD video is usually around 3 mbps. SD is around 1,500 kbps.

Compression can be **lossy** or **lossless**.

Lossless compression:

Lossless compression means that as the file size is compressed, the audio **quality remains the same** - it does not get worse. Also, the file can be restored back to its original state. **FLAC** and **ALAC** are open source lossless compression formats. Lossless compression can reduce file sizes by up to 50% without losing quality. **Lossless Compression** allows original data to be perfectly reconstructed from compression.

Run-length encoding: compression in which sequences with same data value in many consecutive values are stored as a single data value and count
E.g. **00001234111111** can be written as **(0-4)1234(1-6)**

Using **RLE on text data** Consider the following text string: **'aaaaabbbbccddddd'.**

Assuming each character requires 1byte then this string needs 16bytes. If we assume ASCII code is being used, then the string can be coded as follows:

a	a	a	a	A	b	b	b	b	c	c	d	d	d	d	d
05 97					04 98				02 99		05 100				

Past paper question:
(iii) The following diagrams show:
- ☑ the denary colour code that represents each colour
- ☑ the first three rows of a bitmap image

Colour symbol	Colour code (denary)
B	153
W	255

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	B	B	B	B	B	B	B	B	B	B	W	W	W	B	B	B
1	B	B	B	B	B	B	B	B	B	W	W	W	W	W	W	B
2	B	B	B	B	B	B	B	W	W	W	W	W	W	W	W	W
...																
95																

Show how RLE will compress the first three rows of this image.

Row 1: ....

Row 2: ....…

Row 3: ....[2]

Answer:

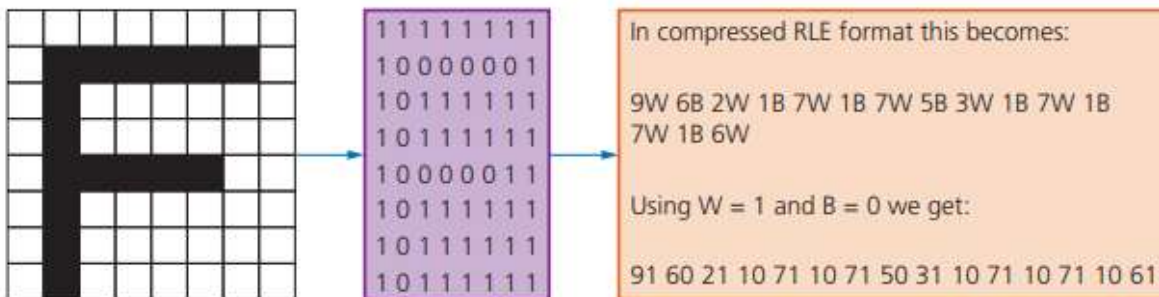**(iii) Two marks for three correct rows, one mark for two correct rows.**   **[2]**

Row 1: 153 10 255 3 153 3
Row 2: 153 9 255 6 153 1
Row 3: 153 7 255 9

Alternative correct answer:

Row 1: 153 9 255 2 153 2
Row 2: 153 8 255 5 153 0
Row 3: 153 6 255 8

Black and White image using RLE.

Figure 1.12 shows the letter 'F' in a grid where each square requires 1 byte of storage.
A white square has a value 1 and a black square a value of 0:

```
11111111
10000001
10111111
10111111
10000011
10111111
10111111
10111111
```

In compressed RLE format this becomes:

9W 6B 2W 1B 7W 1B 7W 5B 3W 1B 7W 1B
7W 1B 6W

Using W = 1 and B = 0 we get:

91 60 21 10 71 10 71 50 31 10 71 10 71 10 61

Lossy compression:

Lossy compression permanently removes data. For example, a WAV file compressed to an **MP3** would be lossy compression. The bit rate could be set at 64 kbps, which would reduce the size and quality of the file. However, it would not be possible to recreate a 1,411 kbps quality file from a 64 kbps MP3.

With lossy compression, the original bit depth is reduced to remove data and reduce the file size. The bit depth becomes **variable**.

MP3 and **AAC** are lossy compressed audio file formats widely supported on different platforms. MP3 and AAC are both patented codecs. **Ogg Vorbis** is an open source alternative for lossy compression.

Lossy Compression is in which file accuracy is low, but file size is smaller than lossless

Perceptual coding: works by reducing certain parts of a sound which are less audible to human hearing.

Lossy Data Compression:
- These programs eliminate unnecessary bits of data
- Reduces data file size more than lossless
- Unable to get back original version

Lossless Data Compression:
- Breaks data into smaller form of transmission
- Allows recreation of original

References:
https://www.bbc.com/bitesize/guides/zqyrq6f/revision/2
https://en.wikipedia.org/wiki/Vector_graphics
https://en.wikipedia.org/wiki/File:Bitmap_VS_SVG.svg
https://www.corporate3design.com/blog/112/
http://etc.usf.edu/techease/win/images/what-is-the-difference-between-bitmap-and-vector-images/
https://www.klipsch.com/blog/digital-vs-analog-audio
https://www.bbc.com/education/guides/z7vc7ty/revision/2
http://www.globalspec.com/reference/37829/203279/5-7-spatial-and-temporal-redundancy-in-mpeg
https://robbfoxx.wordpress.com/2015/07/12/discussion-6-2-1-what-is-redundancy-temporal-redundancy-and-spatial-redundancy/
http://www.pcmag.com/encyclopedia/term/56086/spatial-redundancy
http://christmanwd.weebly.com/uploads/2/5/9/5/25956852/805063_orig.jpg
https://microscope-microscope.org/microscope-info/image-resolution/