AQA Computer Science A-Level
4.5.6 Representing images, sound and other data
Concise Notes

Specification:

4.5.6.1 Bit patterns, images, sound and other data:
Describe how bit patterns may represent other forms of data, including graphics and sound.

4.5.6.2 Analogue and digital:
Understand the difference between analogue and digital:
- data
- signals

4.5.6.3 Analogue/digital conversion:
Describe the principles of operation of:
- an analogue to digital converter (ADC)
- a digital to analogue converter (DAC)

Know that ADCs are used with analogue sensors.

Know that the most common use for a DAC is to convert a digital audio signal to an analogue signal.

4.5.6.4 Bitmapped graphics:
Explain how bitmaps are represented.
Explain the following for bitmaps:
- resolution
- colour depth
- size in pixels

Calculate storage requirements for bitmapped images and be aware that bitmap image files may also contain metadata.

Be familiar with typical metadata.

4.5.6.5 Vector graphics:
Explain how vector graphics represents images using lists of objects.
Give examples of typical properties of objects.
Use vector graphic primitives to create a simple vector graphic.

4.5.6.6 Vector graphics versus bitmapped graphics:

Compare the vector graphics approach with the bitmapped graphics approach and understand the advantages and disadvantages of each.

Be aware of appropriate uses of each approach.

4.5.6.7 Digital representation of sound:

Describe the digital representation of sound in terms of:

- sample resolution
- sampling rate and the Nyquist theorem

Calculate sound sample sizes in bytes.

4.5.6.8 Musical Instrument Digital Interface (MIDI):

Describe the purpose of MIDI and the use of event messages in MIDI.

Describe the advantages of using MIDI files for representing music.

4.5.6.9 Data compression:

Know why images and sound files are often compressed and that other files, such as text files, can also be compressed.

Understand the difference between lossless and lossy compression and explain the advantages and disadvantages of each.

Explain the principles behind the following techniques for lossless compression:

- run length encoding (RLE)
- dictionary-based methods

4.5.6.10 Encryption:

Understand what is meant by encryption and be able to define it.

Be familiar with Caesar cipher and be able to apply it to encrypt a plaintext message and decrypt a ciphertext. Be able to explain why it is easily cracked.

Be familiar with Vernam cipher or one-time pad and be able to apply it to encrypt a plaintext message and decrypt a ciphertext. Explain why Vernam cipher is considered as a cypher with perfect security.

Compare Vernam cipher with ciphers that depend on computational security.

Bit patterns, images, sound and other data

- Computers use bit patterns to represent all forms of data

Analogue and digital

- Analogue data is continuous
- Digital data is discrete
- Analogue signals can take any values and can change as frequently as required
- Digital signals must always take one of a specified range of values and can only change value at specified intervals
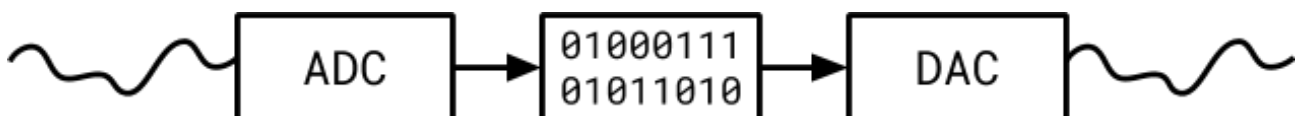

Analogue signal   Digital signal

- Digital signal traces are characterised by sharp edges and vertical drops
- Analogue signals tend to be smooth curves

Analogue/digital conversion

Digital to analogue conversion
- When converting from digital to analogue, a digital to analogue converter is used
- The device reads a bit pattern representing an analogue signal and outputs an alternating, analogue, electrical current
- Their most common use is converting digital audio signals to analogue signals

Analogue to digital conversion
- Sensors like temperature sensors and microphones output an analogue signal
- Computers use analogue to digital converters to convert the analogue signal to a digital bit pattern.
- The device takes a reading of an analogue signal at regular intervals and records the value in a process called sampling
- Samples are taken at a specific frequency, given in Hertz
- This determines the number of samples taken per second
- Greater sampling frequencies result in a better reproduction of an analogue signal

Bitmapped graphics
- Images are broken into pixels
- Each pixel is assigned a binary value
- The value assigned to a pixel determines the colour of the pixel
- An image's resolution can be expressed as a number of dots per square inch, where a dot is a pixel
- Resolution can also refer to the number of pixels in an image
- The number of bits assigned to a pixel is called the image's colour depth
- The number of different colours that can be represented by n pixels is 2^n
- To calculate the storage required for a bitmap image, multiply the number of pixels by the bit depth
- This produces a minimum value as files may also contain metadata such as the image's width, height, date created or colour depth

Vector graphics
- Geometric objects and shapes are used to create images
- Properties for each geometric object or shape in the image are stored in a list

Vector graphics versus bitmapped graphics

Vector Graphics	Bitmapped Graphics
Can be scaled without losing quality	Become blurry or even pixelated when enlarged
Well suited to simple images which use shapes but no good for photographs	Used for storing photographs
Frequently use less storage space than bitmapped graphics	Often use more storage space than vector graphics

Digital representation of sound

- Computers represent sound as a sequence of samples
- Each sample takes a discrete digital value
- The number of samples per second is called the sampling rate
- The sampling rate is expressed in Hertz
- The number of bits allocated to each sample is called the sample resolution
- Higher sample resolutions result in greater audio quality but also increased file size
- The size of a sound sample is calculated by multiplying the duration of the sample in seconds, the sampling rate in Hertz and the sample resolution
- Metadata can be stored in an audio file, increasing the space required to store the file

The Nyquist Theorem
- The sampling rate must be at least twice the frequency of the signal

Musical Instrument Digital Interface (MIDI)

- Used with electronic musical instruments which can be connected to computers
- Stores sound as a series of event messages
- Each event message represents an event in a piece of music

Event messages could contain information such as:
- The duration of a note
- The instrument with which a note is played
- How loud a note is (its volume)
- If a note should be sustained

Benefits of MIDI
- MIDI allows easy manipulation of music without loss of quality
- The instruments on which notes sound can be changed
- Notes can be transposed
- The duration of notes can be altered
- MIDI files are often smaller than sampled audio files

Drawbacks of MIDI
- MIDI can't be used for storing speech
- MIDI sometimes results in a less realistic sound than sampled recordings

Data compression

- FIles are compressed to reduce their size
- Smaller files can be transferred faster
- Images are often compressed
- Sound files and text files can also be compressed
- There are two categories of compression, lossy and lossless

Lossy compression
- Files are reduced in size
- Some information is lost in the process
- The quality of a file is reduced
- There is no limit to the amount of compression that can be applied

Lossless compression
- Files are reduced in size
- No information is lost
- The quality of a file remains the same
- There is a limit to the amount of compression that can be applied
- Two methods are run length encoding and dictionary-based methods
- Both methods are most effective on files that contain lots of repeated data

Run length encoding (RLE)
- Removes repeated information
- Replaces it with one occurance of the information
- This is followed by the number of times it is to be repeated
- Not all data is suitable for compression by run length encoding

Dictionary-based methods
- A dictionary containing repeated data is appended to the file
- Results is a significant reduction in size
- The dictionary used to compress the data must be present in the file
- This will increase the size of the compressed file

Encryption

- The process of scrambling data
- So that it cannot be understood if intercepted
- This keeps it secure during transmission
- Unencrypted information is called plaintext
- Encrypted information is called ciphertext
- A cipher is a type of encryption method
- In order to decrypt ciphertext, you must know the encryption method used
- You must also know the key used to encrypt the plaintext

Caesar ciphers

- Encrypt information by replacing characters
- One character is always replaced by the same character
- There are two types of Caesar cipher - shift ciphers and substitution ciphers
- Can be easily cracked

Shift ciphers

- All letters in the alphabet are shifted by the same amount
- This amount forms the key
- Can be cracked as soon as one letter's replacement is discovered

Substitution ciphers

- Letters are randomly replaced
- A little better than shift ciphers but still relatively easy to crack

Vernam ciphers

- An example of a one-time pad cipher
- Each key should only ever be used once
- The key must be random
- The key must also be at least as long as the plaintext
- Works by:
  1. Aligning the characters of the plaintext and the key
  2. Converting each character to binary
  3. Applying the XOR operation to the bit patterns
  4. Converting the result back to a character
- The only cipher mathematically proven to be completely secure

Computational security

- All ciphers other than the Vernam cipher are crackable
- But not within a reasonable timeframe given current computing power