Logic Gates apply Boolean Operators to Inputs
- Logic gates are special circuits built into computer chips.
- They receive **binary data**, apply a **Boolean operation**, then **output** a binary result.

AND

OR

NOT
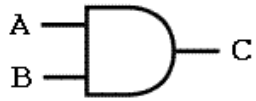
Inputs		Output
A	**B**	**C**
0	0	0
0	1	0
1	0	0
1	1	1

Inputs		Output
A	**B**	**C**
0	0	0
0	1	1
1	0	1
1	1	1

Input	Output
A	**C**
0	1
1	0

You need to learn the equivalent expressions and notations

Gate	Expression	Notation
NOT	NOT A	¬ A
AND	A AND B	A ∧ B
OR	A OR B	A ∨ B

(A ^ B) ∨ C

P = NOT (A AND B)
Only outputs 1 if the output
of A AND B is NOT 1
also called a NAND gate

A	B	R	P
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

P = NOT (A OR B)
Only outputs 1 if the output
of A OR B is NOT 1
also called a NOR gate

A	B	R	P
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

P = (A AND B) OR C
Outputs 1 if the output
of (A AND B) is 1, OR C is 1

A	B	C	R	P
0	0	0	0	0
0	0	1	0	1
0	1	0	0	0
0	1	1	0	1
1	0	0	0	0
1	0	1	0	1
1	1	0	1	1
1	1	1	1	1

P = (A OR B) AND C
Only outputs 1 if the output
of (A OR B) is 1, AND C is 1

A	B	C	R	P
0	0	0	0	0
0	0	1	0	0
0	1	0	1	0
0	1	1	1	1
1	0	0	1	0
1	0	1	1	1
1	1	0	1	0
1	1	1	1	1

A

B

AND

C

OR
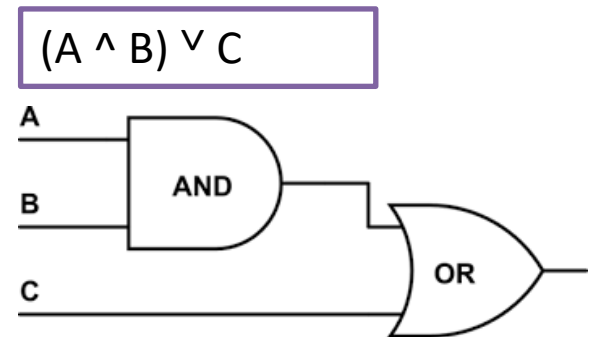
Active revision for Logic gates

Describe out loud the difference between a NOT, AND and OR gate. Use the number of inputs and outputs in your answer

For each gate draw its notation.

For each gate draw their symbol and truth table

Draw the circuit for (A ∨ B) ^ C

Create a logic circuit and truth table for the following logic statement:

Z= *N*OT A OR (B AND C)

Draw the circuit for Q= NOT(C OR D)

Draw the circuit for P = NOT (A AND B)

Draw the logic circuit for the following logic statement: ¬ (A ∧ B).

A

B

AND

OR

C
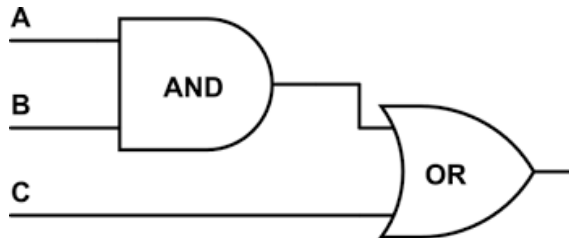
What is the logic statement for this circuit?

UNITS OF DATA MEASUREMENT
The units of measurement used for
DATA STORAGE

BIT Smallest unit of data in computing **0** or **1**

NIBBLE 4 bits **0011**

BYTE 8 bits a single letter on a keyboard **A**

KILOBYTE 1,000 bytes Clipart image

MEGABYTE 1,000,000 bytes MP3 Song

GIGABYTE 1,000,000,000 bytes Movie

TERABYTE 1,000,000,000,000 bytes 200 Movies

Parity Bits are used to check for errors.
- Digits added to the end of numbers e.g. ISBNs on books and are calculated using the other digits in the number.
- An **even parity bit** is added to make a binary string have an even number of 1's.
- An **odd parity bit** is added to make a binary string have an odd number of 1's.

Parity Bits

Even Parity bit ensures an even number of 1s in the binary string

| 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 |

Parity Bit

| 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |

Parity Bit

Parity Bit:

A method of checking binary codes by counting the numbers of 0s and 1s

A bit is added to a block of data for error detection purposes

(Definitions)

Binary Numbers
- The standard number system (0,1,2,3,4,5,6,7,8,9) is known as **denary, decimal or base 10.**
- **Binary** only uses two different digits (0 & 1).
- **Binary** is known as **Base-2.**

Adding Binary Numbers using Column Addition
- Remember the rules-
  - **0 + 0 = 0**
  - **1 + 0 = 1**
  - **0 + 1 = 1**
  - **1 + 1 = 10**
  - **1 + 1 + 1 = 11**

$$0 \quad 1 \quad 0 \quad 1$$
$$+0 \quad +0 \quad +1 \quad +1$$
$$\overline{00} \quad \overline{01} \quad \overline{01} \quad \overline{10}$$

carried bit

Denary and their binary equivalents.

0 = 0	4 = 100	8 = 1000	12 = 1100
1 = 1	5 = 101	9 = 1001	13 = 1101
2 = 10	6 = 110	10 = 1010	14 = 1110
3 = 11	7 = 111	11 = 1011	15 = 111

Binary to Denary using the table

128	64	32	16	8	4	2	1
0	1	0	1	0	1	0	1

64 + 16 + 4 + 1= 85

1. Put the binary numbers into columns

2. Start from the right and add each column

1	0	0	0	1	1	0	1
0	1	0	0	1	0	0	0
1	1	0	1	0	1	0	1
			1				

3. Remember the rules above- 1 + 1 = 10 therefore carry the 1.

Byte = 8 bits

Nibble = 4 bits

x1 bit

128	64	32	16	8	4	2	1
1	1	1	1	1	1	1	1

128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 = 255

Overflow errors occur when a number has too many bits.
- A CPU with a capacity of 8 bits has a capacity of up to 11111111 in binary.
- If one more bit was added there would be an overflow error.
- **Computers deal with these extra bits by storing them elsewhere.**

| GCSE Computer Science | **Data Representation**

Binary Shifts can be used to Multiply or Divide by 2.
- Also known as a logical shift/ Gaps at the beginning or end are filled with 0's.

0	0	1	0	0	0	1	1

=35

0	1	0	0	0	1	1	0

=70

Left shifts Multiply a binary number. For every place shifted **left**, the number is **doubled**.

Right shifts Divide a binary number. For every place shifted **right**, the number is **halved**.

0	0	1	0	0	0	1	1

=35

0	0	0	1	0	0	0	1

=17

Diving using a right binary shift has the same effect as using the DIV operator- Your answer does not include any remainders.

Hexadecimal- numbering system with 16 possible values for an individual digit.

Denary	Hex	Binary	Denary	Hex	Binary
0	0	0000	8	8	1000
1	1	0001	9	9	1001
2	2	0010	10	A	1010
3	3	0011	11	B	1011
4	4	0100	12	C	1100
5	5	0101	13	D	1101
6	6	0110	14	E	1110
7	7	0111	15	F	1111

Hexadecimal   **5F**

Binary   **0101 | 1111**

128 64 32 16 8 4 2 1
01011111
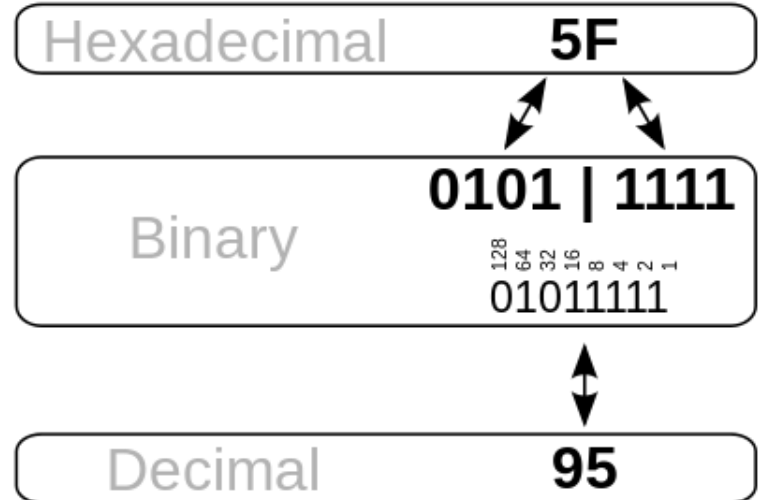
Decimal   **95**

Active revision for Units of data, binary, logical shifts and hexadecimal

Calculate how many 10MB music files Billy can save on a 260 GB SSD.

How many bits are in a byte? Or a nibble?

How many bytes do 12bits make?

How many megabytes are in 3 gigabytes?

Convert your age into an 8 bit binary value

Convert the year you were born into an 8 bit binary value

Show how to add 10001100 to 00110110

What's an overflow error?

Show two 8 bit binary numbers being added when the result of an overflow error occurs.

Show a logical shift **right by two** on 00101100

Show a logical shift **left by two** on 00101100

What is the denary value of **3F** when represented as Hexadecimal?

Draw the hexadecimal table

Describe how to convert 120 into Hexadecimal

Paper 2 | GCSE Computer Science | **Data Representation**

Characters- Each character can be represented in binary
- **ASCII** is the most common **character set**
- Characters are both letters, numbers and symbols.
- **ASCII character set** gives each character a 7 bit binary code with 128 combinations
    - includes alphabet, numbers, symbols and commands
- **ASCII** start with 0 this means characters fit nicely into 1 byte
- **Extended ASCII** uses all 8 bits and allows for 256 characters, the first 128 is the same as ASCII
- **Unicode** differs from ASCII and covers all possible characters and all languages. Often 16-bit and 32-bit

Images are stored as pieces of data. Made of bits.
- Bitmap images are used for photos. They are made of tiny dots called **pixels.**
- **Colour** of each pixel represented by binary code.
- **Number** of colours = number of bits the code has.

B & W images x2 colours
0= white & 1= black

0	1	0	1
1	0	1	0
0	1	0	1
1	0	1	0
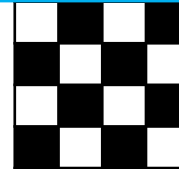
2-bit images have 4 colours. Pixel has one of 4 binary values- **00/ 01/ 10/ 11**

00	01	10	11
00	01	10	11
00	01	10	11
00	01	10	11

Examples of ASCII Characters

| Character	Binary	Hex	Denary
Backspace	00001000	8	8
+	00101011	2B	43
3	00110011	33	51
B	01000010	42	66
b	01100010	62	98

Resolution
- Resolution is the **density** of pixels in an image
- Measured in **DPI** (Dots Per Inch)
- Higher resolution means more pixels in a certain area
- Higher resolution increases file size
- Higher resolution increases image quality

Colour Depth
- Is the number of bits used for each pixel
- Num of bits= Num of colours can be used
- Formula for colour depth is 2n where n is the number of bits per pixel
- **1-bit image = 2^1 = 2 colours**
- **2-bit image = 2^2 = 4 colours**
- **4-bit image = 2^4 = 16 colours**
- **24-bit image = 2^{24} = 16777216 colours**
- Most devices use 24 bits
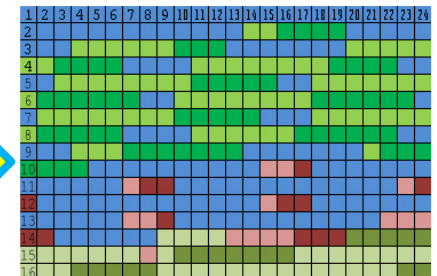- Higher colour depth increases file size

Metadata
- Metadata is information stored in an image/ it is needed to display the image
- Metadata contains the Height, Width, File format, Colour depth, Resolution
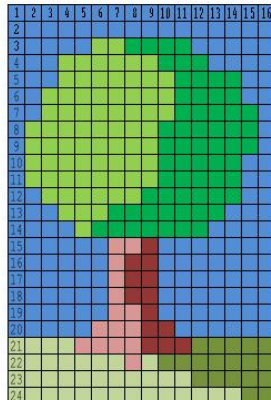
Correct metadata 16 x 24 pixels

**Same data but with WRONG size metadata- 24 x 16 pixels**

**Storing Sound-** Sound is made of bits.
1. Sound is recorded as an **analogue signal**.
Analogue signals are pieces of continually changing data**
2. Analogue signal converted to **digital data** using a **analogue to digital converter. Known as sampling.**

Sampling frequency = how often a sample is taken and is measured in Hertz (Hz). 1 Hz means once per second. 1 MHz (megahertz) means one million times per second. **Higher sampling frequency = higher accuracy but more storage****

Digital Signals- series of pulses consisting of just 2 states ON (1) or OFF (0). E.g. DAB radio.

Analogue Signals- e.g. FM and AM radios.

Low sampling rate
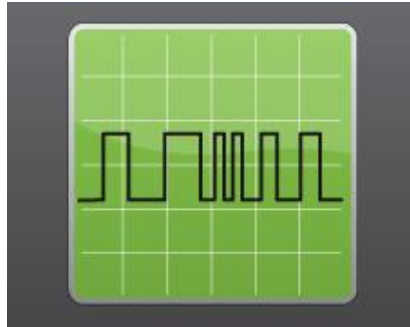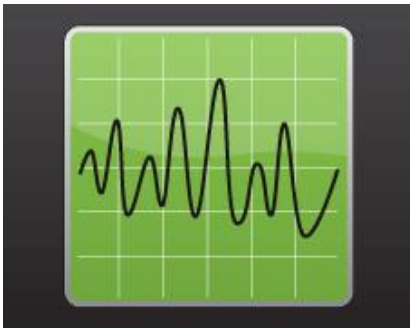
Sampling intervals- are the gaps between each of the points where the analogue recording is sampled.

Higher sampling rate
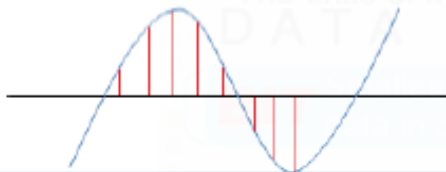
Example- An analogue recording is sampled every 5 milliseconds (ms) the sampling interval would be 5ms
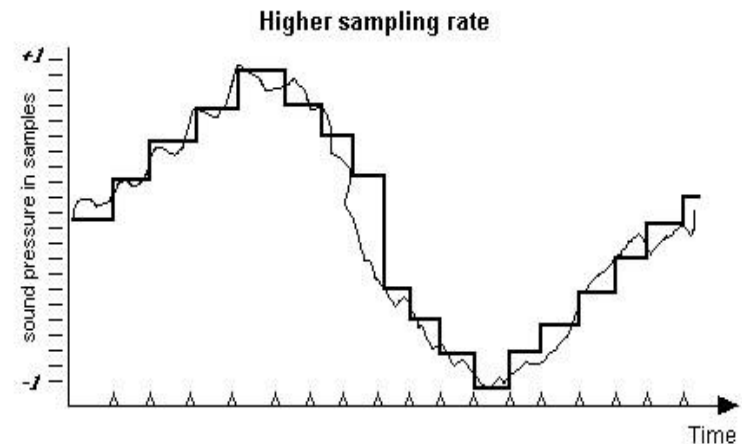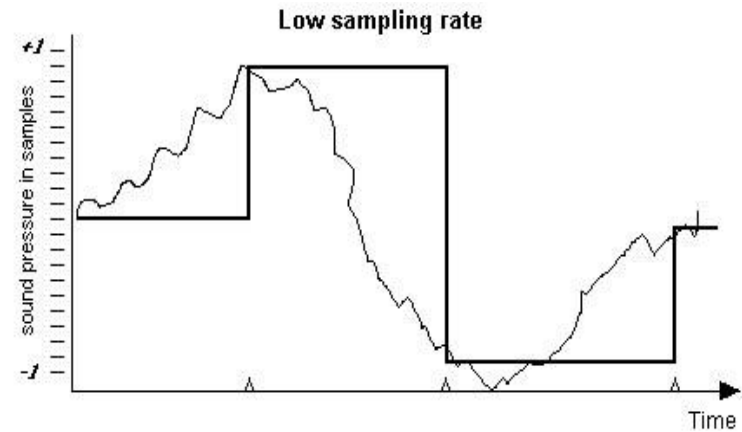
Bit rate = Sampling frequency x sample size.

Compression- data compression is when files are made smaller whilst still keeping the compressed file as true to the original as possible.
Compression changes some attributes of the file-
• File type/ Resolution/ Dimensions/ Bit depth.

Compressed files have many uses

• **Smaller files** = smaller space on devices.
• **Streaming/downloading** quicker as they take up less **bandwidth.**
• Allows web pages to **load quicker** in web browsers.
• Allows files to be **sent** over emails if there is a limit on size.

Two types of compression- Lossy and Lossless
• **Lossy-** works by permanently removing data from the file. This limits the number of bits the file needs and so reduces its size.
• **Lossless-** makes the file smaller by temporarily removing data to store the file and then restores it to its original state when its opened.

Types of compressed image files
There are a number of types of compressed image file. These include the following.
• **PNG** is a **lossless** compression type. It is often used where the graphic might be changed by another person or where the image contains layers of graphics that need to be kept separate from each other. It is **high quality**.
• **JPEG** is often used for digital camera images because it has a fairly small file size for the quality that it displays. JPEG is a **lossy format** that offers a higher compression rate than PNG in the trade-off for quality.

This Character has alot of pixels if we want to keep it the same then we have to use Lossless Compression and it will remain the same.

However if we use lossy compression it will look alot worse then expected as pixels will stretch fix the area and try to retain colour so colours may change slightly.

Lossless Compression.

Lossy Compression.

Active revision for Characters, storing images, storing sound and compression

What is meant by the term character set?

What does ASCII stand for?

Why would extended ASCII be used?

What is meant by Unicode?

How many colours can be represented in an image with 8 bits? Show your workings out.

Why is metadata included in a file?

How does the resolution of an image affect the size of the file?

What is sampling frequency? What happens if you increase it?

What is sampling interval? What happens if you increase it?

What is sampling size? What happens if you increase it?

What is bit rate? What happens if you increase it?

What advantages are there for compressing files?

What are the pros and cons for Lossy compression?

What are the pros and cons for Lossless compression?