IGCSE 01 Data Representation

Hexadecimal | Octonary

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

Binary to Denary

1	0	1	1	1
2^4	2^3	2^2	2^1	2^0

$1\times2^4+0\times2^3+1\times2^2+1\times2^1+1\times2^0 = 23$

Octonary to Denary

3	2	0	7
8^3	8^2	8^1	8^0

$3\times8^3+2\times8^2+0\times8^1+7\times8^0 = 1671$

Hexadecimal to Denary

3	A	F
16^2	16^1	16^0

$3\times16^2+10\times16^1+15\times16^0 = 943$

Denary to Binary

2 | 59 → 1
2 | 29 → 1
2 | 14 → 0
2 | 7 → 1
2 | 3 → 1
2 | 1 → 1
0

59 ≈ 111011

Denary to Octonary

8 | 1671 → 7
8 | 208 → 0
8 | 26 → 2
8 | 3 → 3
0

1671 ≈ 3207

Denary to Hexadecimal

16 | 943 → 15~F
16 | 58 → 10~A
16 | 3 → 3
0

943 ≈ 3AF

Denaty to Octonary

binary: 10111101

1	0	1	1	1	1	0	1

2	7	5

octonary: 275

Denary to Hexadecimal

binary: 10111101

1	0	1	1	1	1	0	1

C	E

octonary: 275

Overflow: a condition when the result of a calculation is too large to fit into the number of bits defined for storage

Binary Add

```
  1 0 1 1 1 0 1 0   186
+ 1 1 1 0 1 1 0 1   237
  1 1 1 1 1 0 0 0
1 1 0 1 0 0 1 1 1   423
```

unit of measurement	abbreviation	conversion
bit	b	1 bit
nibble		4 bits
Byte	B	8 bits
Kilobyte	kB	1024 bytes
Megabyte	MB	1024 kB
Gigabyte	GB	1024 MB
Terabyte	TB	1024 GB

HTML Color

#FFFFFF #333333 #FF00000 #00FF00 #0000FF #00FFFF #9324FC

color depth: **the number of bits** used to represent each pixel
black and white, **1bit** per pixel
4 color, **2 bits** per pixel
8 color, **3 bits** per pixel

Image resolution: **the number of pixels** that make up an image.
400*600 pixels

400 * 600 image, RGB 3 * 16 * 16 color depth, **file size**:
400 * 600 * 3 * ($\log_2256$) bits
= 240000 * 3 * 8 bits = 720000 byte
= 720000 / 1024 kB = 703 kB
= 703 / 1024 MB = 0.68MB

	R	G	B	range
hexadecimal	93	24	FC	00~FF
denary	147	35	252	0~255
binary	10010101	00100100	11111100	

Calculation of file size:
image file size:
image resolution(in pixels) * colour depth (in bits)
sound file size:
sample rate(in HZ) * sample resolution (in bits) * length of sample (in seconds)

binary usages: register, memory, Logic Gates, Boolean Algebra, Machine Language, Data Representation
hexadecimal usages: color in HTML, MAC address, assembly languages, machine code, IPv6.
why use binary store data?
1. a computer can only work with binary data
2. computers use switches/ logic gates
3. only use 2 states, On or Off, 1 or 0.
why use hexadecimal?
1. more **convenient** to use
2. one hex digit represent four binary digits
3. hex number is far easier for humans to remember, copy and work with.
why designer use hexadecimal?
1. Uses fewer characters // shorter
2. Easier to read / write / understand
3. Less likely to make mistakes // less error prone
4.Easier to debug

Logical shift:
left logical shift: multiplying by 2 for each shift
right logical shift: dividing by 2 for each shift
multiple shift

Bits shifted from the end of the register are lost and zeros are shifted in at the opposite end of the register

two's complement: represent positive and negative 8-bit binary integers

positive number:
sign bit 0, positive binary value
negative number:
sign bit 1
1. **write positive binary value**
2. **invert each binary value**
3. **add 1 to the number**

ASCII code
8 bit length
Standard ASCII code character set consists of 7-bits codes
Extended ASCII use **8 bit codes** give another 128 codes to allow for non-English alphabets.
Unicode
1. represent non-Western languages, such as Chinese or Japanese characters.
2. up to 32 bits per character

Sound
sampling resolution: the **number of bits per sample**.
sampling rate: the number of **sound samples** taken **per second**.

The greater the number of bits used to represent the amplitude, the greater the accuracy of the sampled sound.

Lossy and Lossless Compression
why compression?
1. **save storage** on devices
2. **reduce the time** taken to stream a music or video file
3. reduce the time taken **upload**, **download** or file across a network
4. reduce file size also **reduce costs**.
Lossy compression: The original file **cannot be reconstructed** once it has been compressed.
How lossy compression: the algorithm used int the lossy technique have to decide which parts the file need to be retained and which can be discarded
JPEG
1. reducing resolution or color depth
2. reducing sample rate or resolution
Lossless compression:
all the data from the original uncompressed file **can be reconstructed**.

Run-length encoding:
1. reducing the size of a string of adjacent, identical data items
2. the repeating unit is encoded into two values: first value represents number of identical data items I, second value represents code (such as ASCII) of data item.
usages: image, text, code.