Week 1 – Bits & Bytes

Student number: 585732

Assignment 1.1: Bits & Bytes intro

What are Bits & Bytes?

- A bit (short for binary digit) is a single place or symbol in a binary number. It is the smallest unit of data in a computer, and its value can be either 0 or 1.

- A byte is a sequence of bits grouped together in sets of eight. Computers group bits into bytes to make these sequences easier to manage.

What is a nibble?

A nibble is a group of 4 bits, which is exactly half of a byte. Because 4 bits can represent 16 different values, a nibble is often used when working with hexadecimal numbers, where each hex digit corresponds to one 4-bit nibble.

What relationship does a nibble have with a hexadecimal value?

A nibble has a direct relationship with a hexadecimal value because a single hexadecimal symbol can represent four bits (a nibble). A 4-bit number can represent 16 unique combinations (from 0 to 15 in decimal) , and the hexadecimal system uses 16 symbols (0-9 and A-F).

Why is it wise to display binary data as hexadecimal values?

It is wise to display binary data using hexadecimal values because hexadecimal offers a more compact and human-readable representation of the binary data.

- Since one hexadecimal symbol represents four binary bits, a large binary number can be shortened by a factor of four when converted to hex.
- For example, when inspecting computer memory, a format is often used where memory addresses and the contents of the memory bytes are represented using hexadecimal. This makes the memory inspection more compact compared to representing every bit individually.

What kind of relationship does a byte have with a hexadecimal value?

A byte has a direct and simple relationship with a hexadecimal value:

- A byte is made up of 8 bits.
- Since a single hexadecimal symbol represents 4 bits (a nibble), a full byte (8 bits) is perfectly represented by two hexadecimal symbols.

- Example: The binary byte `11111111` is represented as the hexadecimal value `FF`.
- Example: The binary byte `01001000` (which is the ASCII for 'H') is represented as the hexadecimal value `48`.

An IPv4 subnet is 32-bit, show with a calculation why this is the case.

The IPv4 subnet mask, which is used to define the boundaries of a network, is 32 bits because an IPv4 address itself is 32 bits in length.

Calculation and Explanation:

1. An IPv4 address is typically displayed in dotted decimal notation.
2. This notation separates the 32 bits into four groups of 8 bits each. Each 8-bit group is referred to as an octet.
3. The calculation is a simple summation of the bits in each octet:

   4 octets × 8 bits per octet = 32 bits total

   The subnet mask (also a 32-bit number) aligns directly with the 32-bit IP address, using `1`s for the network prefix bits and `0`s for the host ID bits. The example subnet masks you provided are simply the dotted decimal representations of a 32-bit number:

- **255.0.0.0** (An 8-bit network prefix)

  Binary: 11111111 . 00000000 . 00000000 . 00000000 (8 bits of '1's + 24 bits of '0's = 32 bits)

- **255.255.0.0** (A 16-bit network prefix)

  Binary: 11111111 . 11111111 . 00000000 . 00000000 (16 bits of '1's + 16 bits of '0's = 32 bits)

- **255.255.255.0** (A 24-bit network prefix)

  Binary: 11111111 . 11111111 . 11111111 . 00000000 (24 bits of '1's + 8 bits of '0's = 32 bits)

Assignment 1.2: Your favourite color

My favourite color is: Black

Hexadecimal color code: #000000

The hexadecimal color code #000000 defines my favorite color, black, by using the RGB color model. This model uses three pairs of hexadecimal digits to specify the amount of Red, Green, and Blue light used to create the color: #RRGGBB.

Since black is the absence of all light, the code requires the minimum intensity for every component. In hexadecimal, 00 represents the lowest possible intensity (zero).

Therefore, setting all three pairs to `00`:

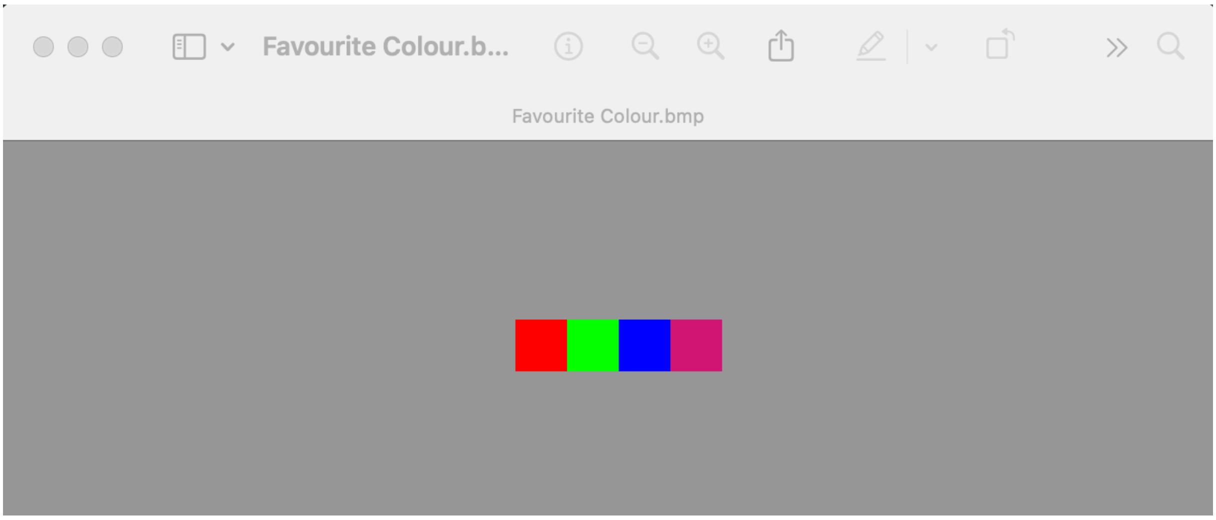
- Red is off (`00`)
- Green is off (`00`)
- Blue is off (`00`)

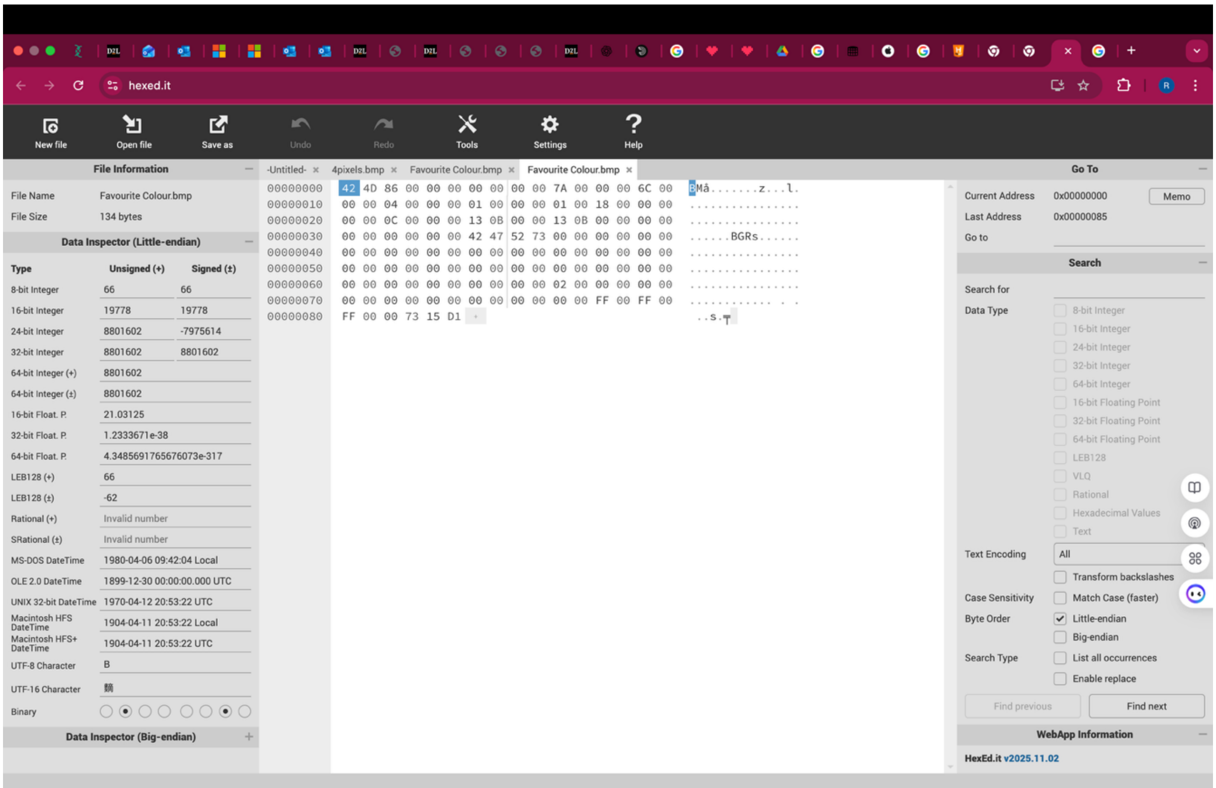
This combination results in **#000000**, which is black.

Assignment 1.3: Manipulating binary data

Color	Color code hexadecimaal (RGB)	Big Endian	Little Endian
RED	FF0000	FF 00 00	00 00 FF
GREEN	00FF00	00 FF 00	00 FF 00
BLUE	0000FF	00 00 FF	FF 00 00
WHITE	FFFFFF	FF FF FF	FF FF FF
Favourite (previous assignment)	D11573	D1 15 73	73 15 D1

Screenshot modified BMP file in hex editor:





Assignment 1.4: Student number to HEX and Binary

Convert your student number to a hexadecimal number and a binary number.

My student number: 585732

Hexadecimal value of my student number: 8F004

Binary value of my student number:

Explain in detail that the calculation is correct. Use the PowerPoint slides of week 1.

Decimal to Hexadecimal Conversion:

As my student number is decimal number and I have to convert it to hexadecimal value. The base of hexadecimal is 16. I divided my student number by the base 16 and note down the remainder and then I divided the result by 16 and each time and note down the remainder till the result of my division is 0. From 10 to 15 the hexadecimal digits are (A- F) so I converted the digit to hexadecimal digit. Then, I listed the numbers from bottom to top and got the hexadecimal value of my student number.

	Result	Remainder	Hexadecimal Digit
585732÷16	36608	4	4
36608÷16	2288	0	0
2288÷16	143	0	0
143÷16	8	15	F
8÷16	0	8	8

Decimal to Binary Conversion: It is the same calculation as the previous one. After the division was complete, I listed the remainders from bottom to top and got the binary value of my student number.

	Result	Remainder
585732÷2	292866	0
292866÷2	146433	0
146433÷2	73216	1
73216÷2	36608	0
36608÷2	18304	0
18304÷2	9152	0
9152÷2	4576	0
4576÷2	2288	0
2288÷2	1144	0
1144÷2	572	0
572÷2	286	0
286÷2	143	0
143÷2	71	1
71÷2	35	1
35÷2	17	1
17÷2	8	1
8÷2	4	0
4÷2	2	0
2÷2	1	0
1÷2	0	1