

# Template Week 1 – Bits & Bytes

Student number: 586377

## Assignment 1.1: Bits & Bytes intro

What are Bits & Bytes?

Een bit is de kleinste eenheid van digitale informatie.

Het kan alleen twee waarden hebben: **0** of **1**.

Een byte bestaat uit 8 bits.

Met 8 bit kun je 256 verschillende combinaties maken ( $2^8 = 256$ ).

Bytes worden gebruikt om tekst, cijfers, afbeeldingen en programma's op te slaan.

What is a nibble?

Een nibble is een groepje van 4 bits (dus een halve byte).

What relationship does a nibble have with a hexadecimal value?

Elke nibble (4 bits) komt precies overeen met een hexadecimaal cijfer.

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

Omdat binaire getallen (zoals 1010111010010110) lang en moeilijk te lezen zijn.

Met hexadecimale waarden kun je diezelfde informatie korter en overzichtelijker weergeven.

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

Een byte = 8 bits = 2 nibbles = 2 hex-tekens.

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

Een IPv4-adres bestaat uit 4 getallen (bijv. 192.168.0.1), gescheiden door punten.

Elk van die getallen is 1 byte (8 bits) groot en kan een waarde hebben tussen 0 en 255.

4 delen  $\times$  8 bits = 32 bits

IPv4 = 32 bits totaal

= 4 bytes

=  $2^{32}$  mogelijke adressen

= 4.294.967.296 unieke IP-adressen

### **Assignment 1.2: Your favourite color**

Hexadecimal color code: #152dcf

### Assignment 1.3: Manipulating binary data

Color	Color code hexadecimaal (RGB)	Big Endian	Little Endian
RED			
GREEN			
BLUE			
WHITE			
<b>Favourite</b> (previous assignment)			

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.

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

586377 to hex: 0x8F289

Je deelt het getal steeds door 16 en noteert de resten.

Lees de resten van onder naar boven → **8 F 2 8 9**.

586377 to binary: 10001111001010001001

Je deelt het getal steeds door 2 en noteert de resten.

Lees de resten van onder naar boven → **10001111001010001001**.

Ready? Save this file and export it as a pdf file with the name: [week1.pdf](#)