ERHVERVSAKADEMI
AARHUS

Filer

Agenda:

- Repetition bits & bytes
- Flere bytes -> filer
- Fil formater
- Se ned i en fil (hex editor)
- Filformater
- Ændre data i en fil
- Tekst Filer
- Billede Filer
- Space invaders
- Opgave: Design et filformat

ERHVERVSAKADEMI
AARHUS

Codelabby

Base64

	z	v	X	o	D	j	
Oct	63	57	27	50	03	43	2 cifre = 6 bits (2x3)
Bin	110 011	101 111	010 111	101 000	000 011	100 011	
	1100 1110	1111 0101	1110 1000	0000 1110	0011		
Hex	C E	F 5	E 8	0 E	3		

Base 2 numerals

✅ A 36-bit number has been Base64-encoded using the letters zvXoDj. Make the machine show the corresponding decimal value by translating to a 9-digit hex numeral and entering that into the memory editor.

You may need a piece of paper for this problem, but it can be solved using base 2^n numerals only. The following lookup table should be helpful:

A 00	B 01	C 02	D 03	E 04	F 05	G 06	H 07
I 10	J 11	K 12	L 13	M 14	N 15	O 16	P 17
Q 20	R 21	S 22	T 23	U 24	V 25	W 26	X 27
Y 30	Z 31	a 32	b 33	c 34	d 35	e 36	f 37
g 40	h 41	i 42	j 43	k 44	l 45	m 46	n 47
o 50	p 51	q 52	r 53	s 54	t 55	u 56	v 57
w 60	x 61	y 62	z 63	0 64	1 65	2 66	3 67
4 70	5 71	6 72	7 73	8 74	9 75	+ 76	/ 77

Some guidance follows, should you need it.

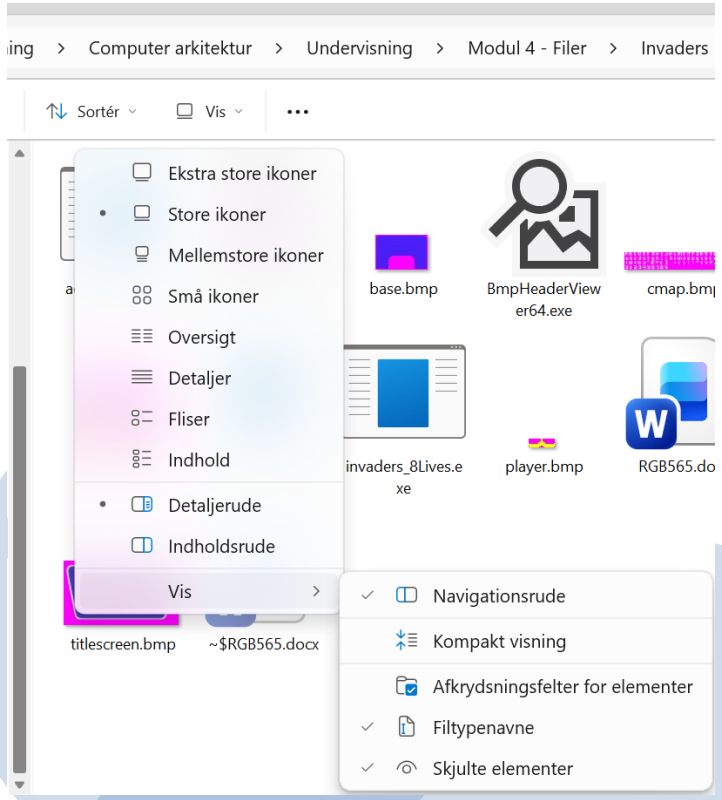
The six Base64 digits each corresponds to six bits for a total of 36. To translate to hex, you need to find out what those six bits are using the octal table above. Then you need to regroup the 36 bits into nine groups of four bits, each corresponding to a single hex digit.

Filtypenavn er ikke forbundet direkte med indholdet,
men en information til Windows om hvad den skal gøre med filen.
Hvilket program associeres filen med.

For at få vist filetypenavn i Windows, skal det aktiveres:

Så hvad sker der når vi ændrer filtypenavnet.

Demo.



ERHVERVSAKADEMI
AARHUS

Hvad er der i filer?

HexEd.it - Gratis, browser-baseret, online og offline hex-redigering

Hexed.it er en hex-editor, som er en fælles betegnelse for et værktøj til oprettelse, læsning og redigering af filer som rå data.

Data er primært repræsenteret som hexa-decimale værdier og som ASCII værdier.

Data er inddelt i bytes.

Hexed.it kan yderligere vise forskellige talværdier som består af flere bytes:

Feks. 16 Bit, 32 Bit, base64, Forskellige tegntabeller.

ERHVERVSAKADEMI
AARHUS

Filetyper

Filnavne og filtypenavne.

Vi tager udgangspunkt i Windows OS.

Filer er samlinger af data (flere bytes)

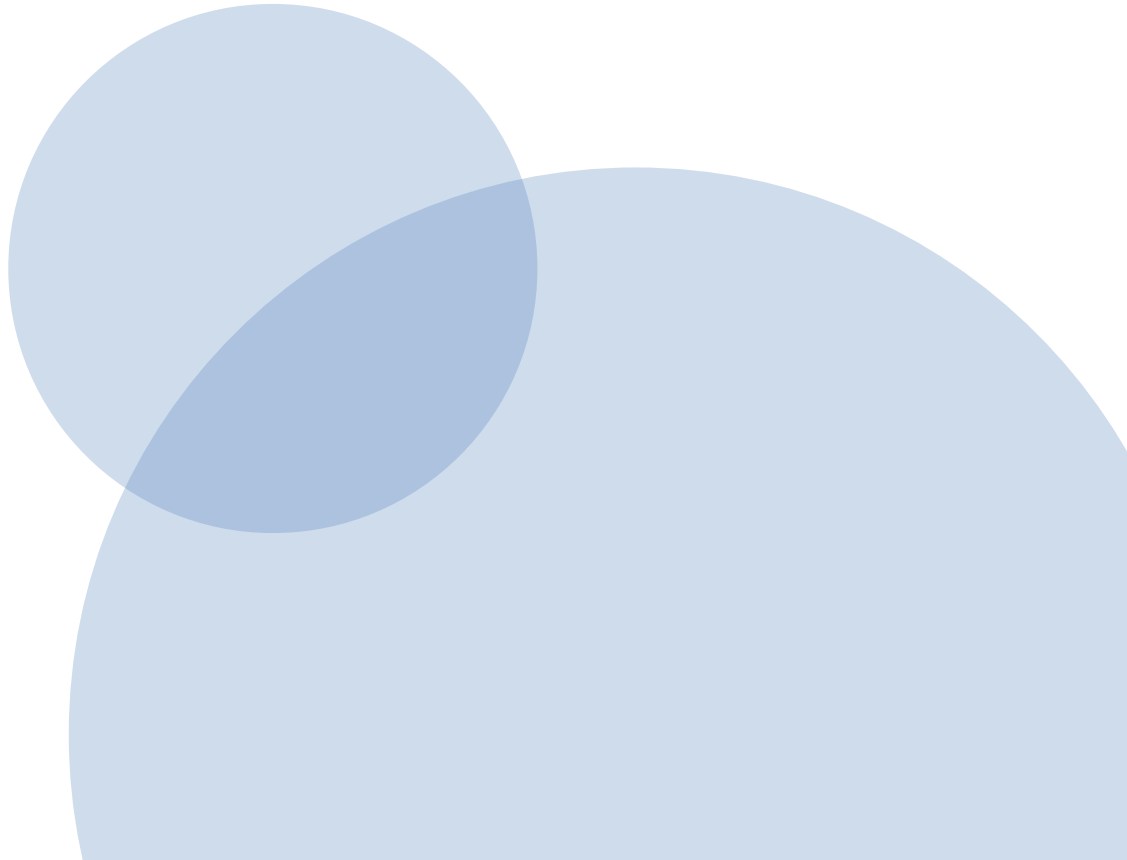
Filnavnet angiver navnet på filen

Filtypenavn (3 eller fire tegn) angiver hvilken type af data filen repræsenterer.

Ex: mitDokument.**docx**

Hvilke andre filtypenavne er almindelige?

Hvad tror i filtypenavnet betyder for indholdet i en fil?



Endianess

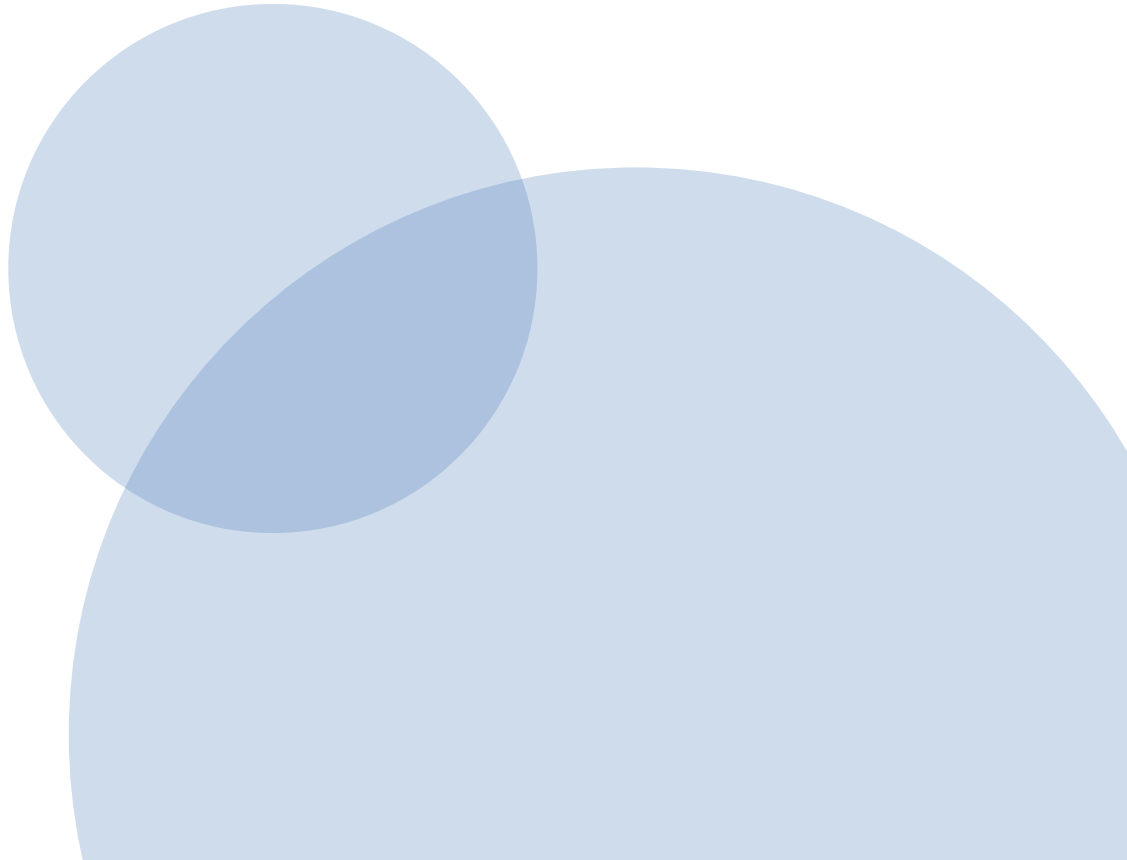
Når man sætter flere bytes sammen, for at kunne repræsentere en større talbredde end 0-255, skal vi taget stilling til i hvilken rækkefølge bytes placeres.

Eksempel:

Vi har brug for at kunne bruget talværdier mellem 0 og 65535.
Dette kræver 16 bit. Fordi 2^{16} = 65535.

0b0000000000000000 = 0
0b1111111111111111 = 65535

Vi deler op i 2 bytes
16 bits = 2 bytes



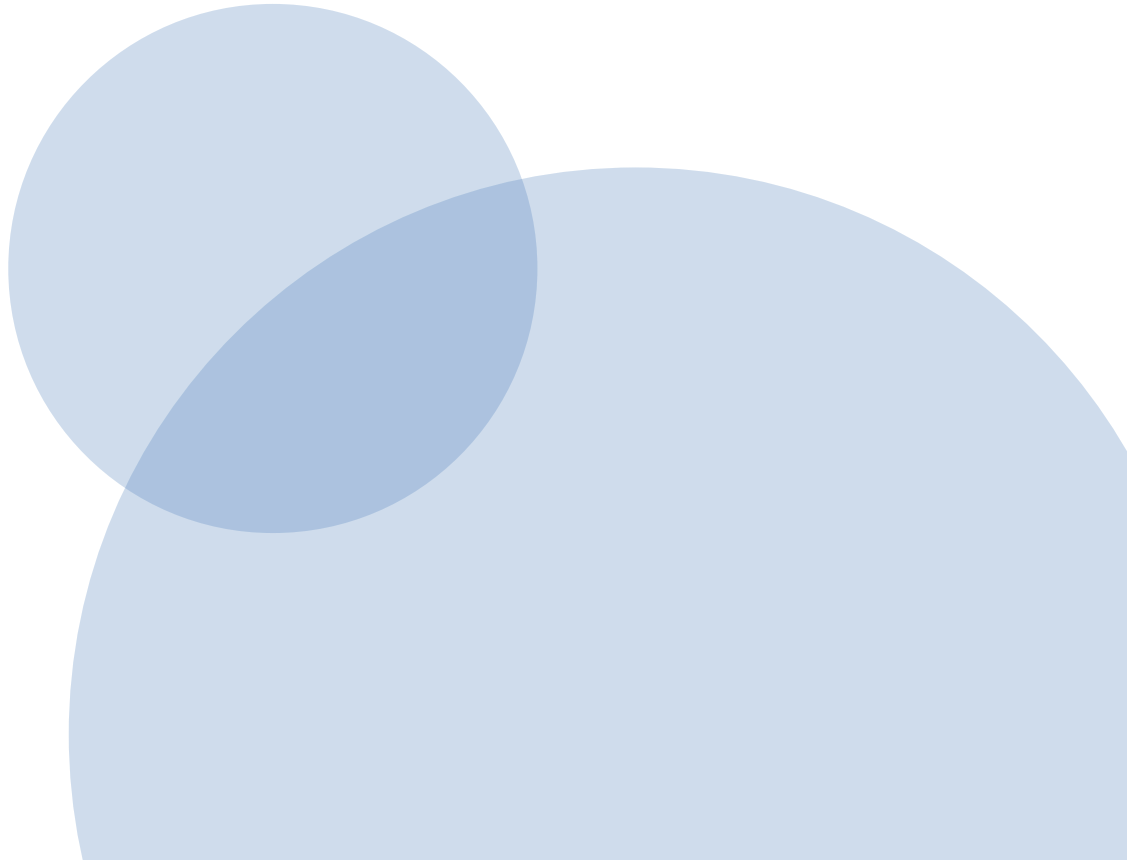
Endianness

Byte 1 (MSB)	Byte 2 (LSB)
0b00000110	0b11000000

MSB = Mest betydende byte
LSB = Mindst betydende byte

Big Endian
Byte 1 -> Byte 2

Little Endian
Byte 2 -> Bytes 1



Endianness

Byte 1 (MSB)	Byte 2 (LSB)
0b00000110	0b11000000

MSB = Mest betydende byte
LSB = Mindst betydende byte

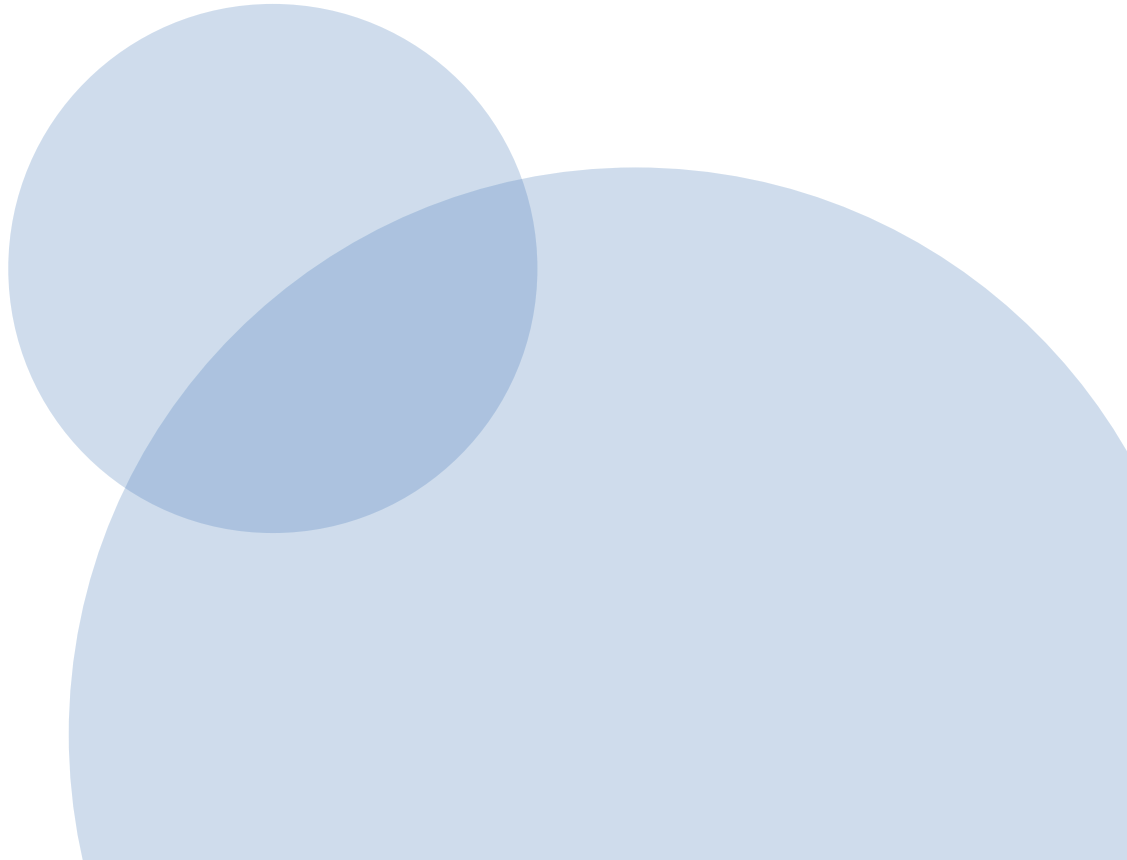
Big Endian = (Byte 1 * 256) + (Byte 2 *1) = 0b00000110 + 0b110000000 = 0x06C0 = **1728**

Little Endian = (byte 2 256) + (Byte 1) = 0b11000000+0b00000110 = 0xC006 = **49158**

Endianness

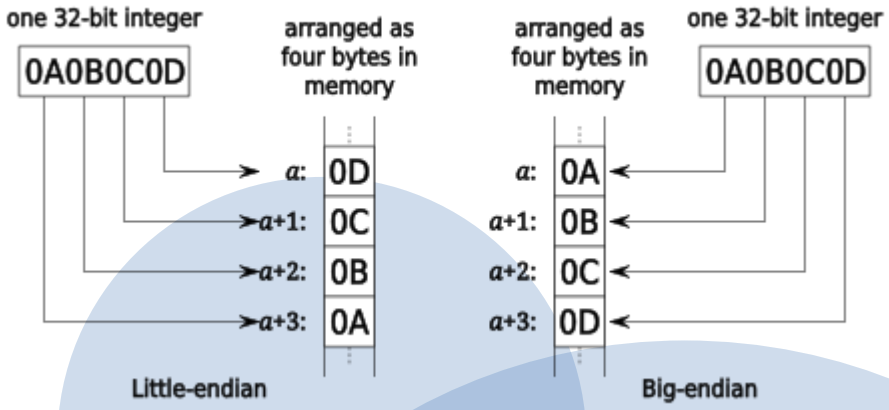
0x499602D2

Brug Hexedit.it til at se ovenstående hexadecimale 32 bit værdi, som henholdsvis Big Endian og Little Endian.



Endianness

In computing, **endianness** is the order in which bytes within a *word* data type are transmitted over a data communication medium or addressed in computer memory, counting only byte significance compared to earliness. Endianness is primarily expressed as **big-endian** (**BE**) or **little-endian** (**LE**).



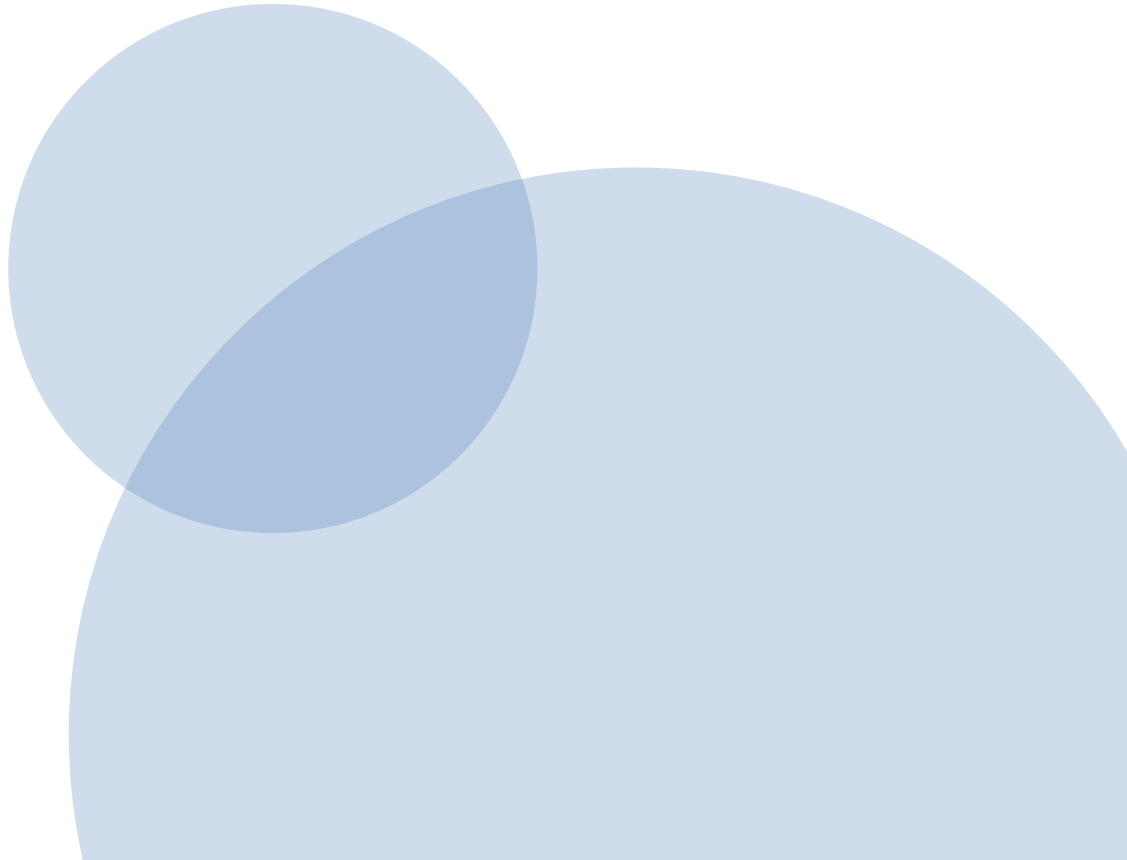
ERHVERVSAKADEMI
AARHUS

Filen

Nu ved vi lidt om hvordan data kan organiseres.

Når vi taler sikkerhed er ændringer, sletning og skabelse af data i filer forbundet med stor risiko.

Så hvad er der i en fil og kan vi se det?



ERHVERVSAKADEMI
AARHUS

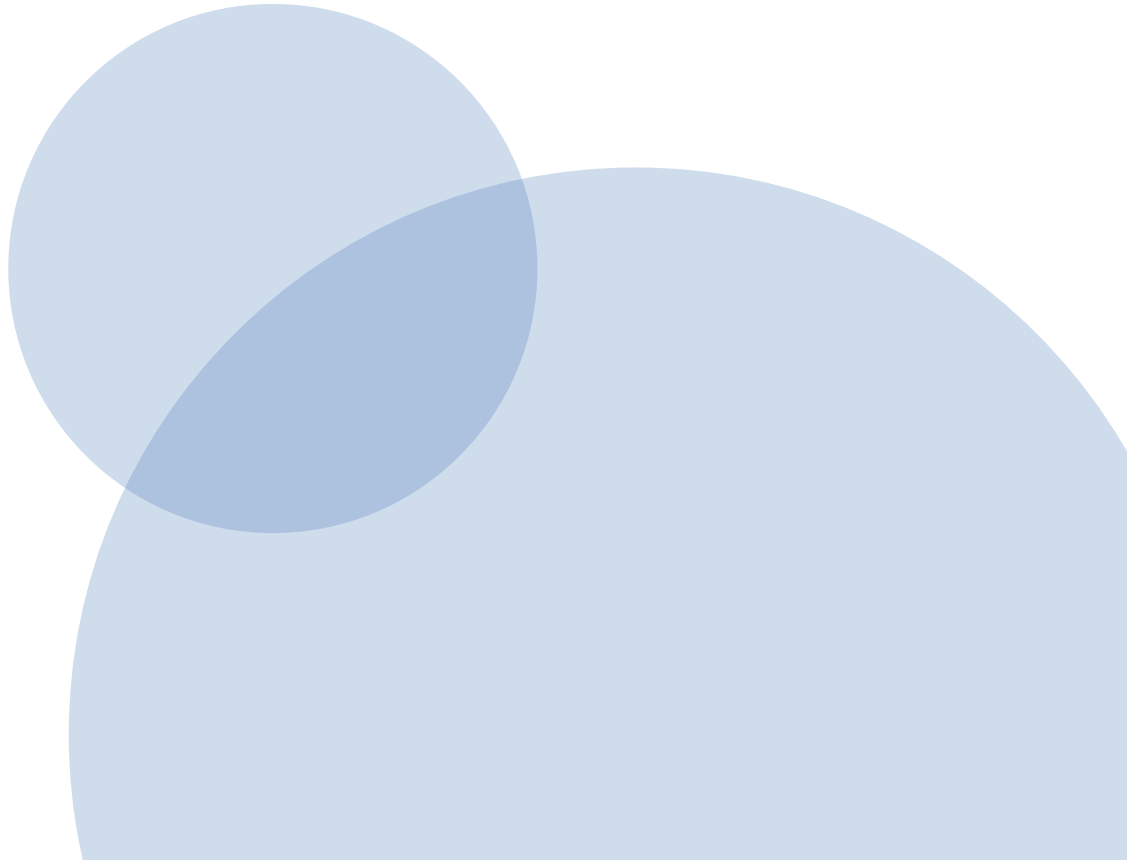
Filen

Forsøg:

Opret en ny fil i notesblok.
Skriv to linier tekst.

Gem filen, som .txt
Åben filen i hexed.it

Hvad ser i?

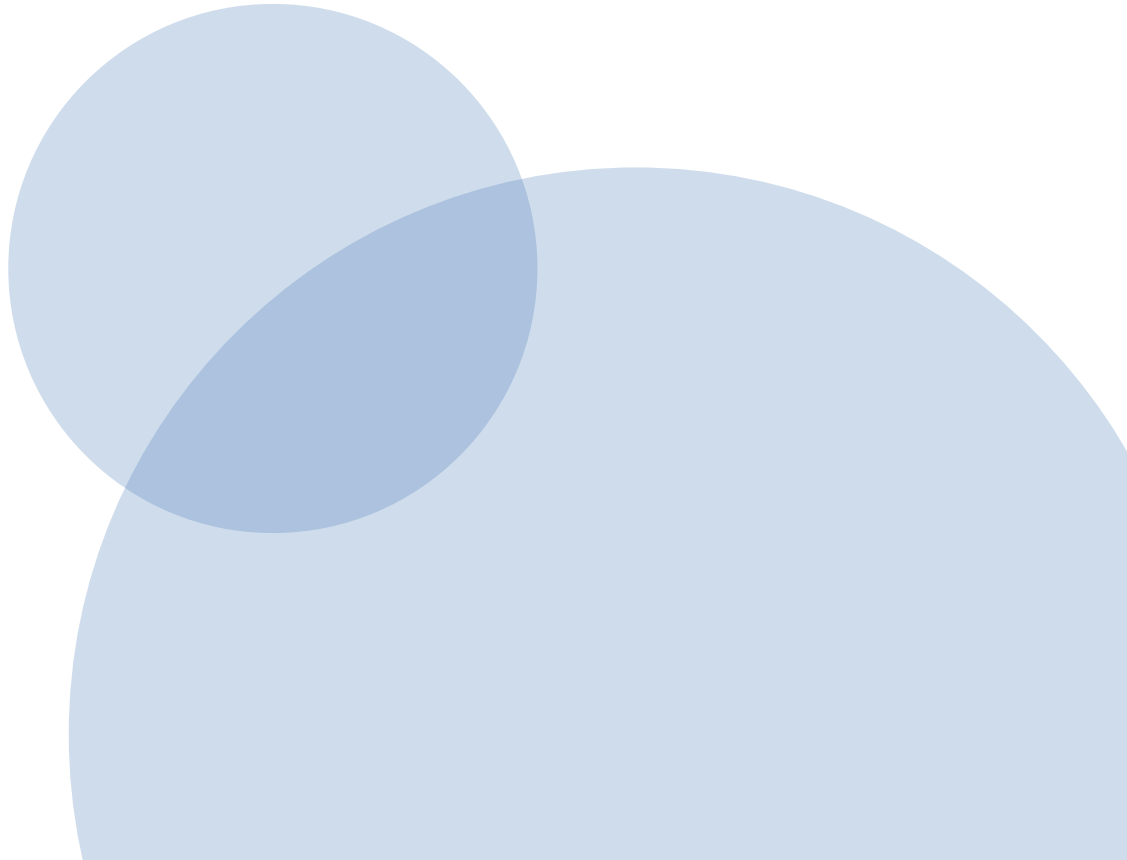


Filen

Prøv at ændre i hex-værdierne i hexed.it og gem filen.

Åben filen i notesblok.

Hvad sker der.

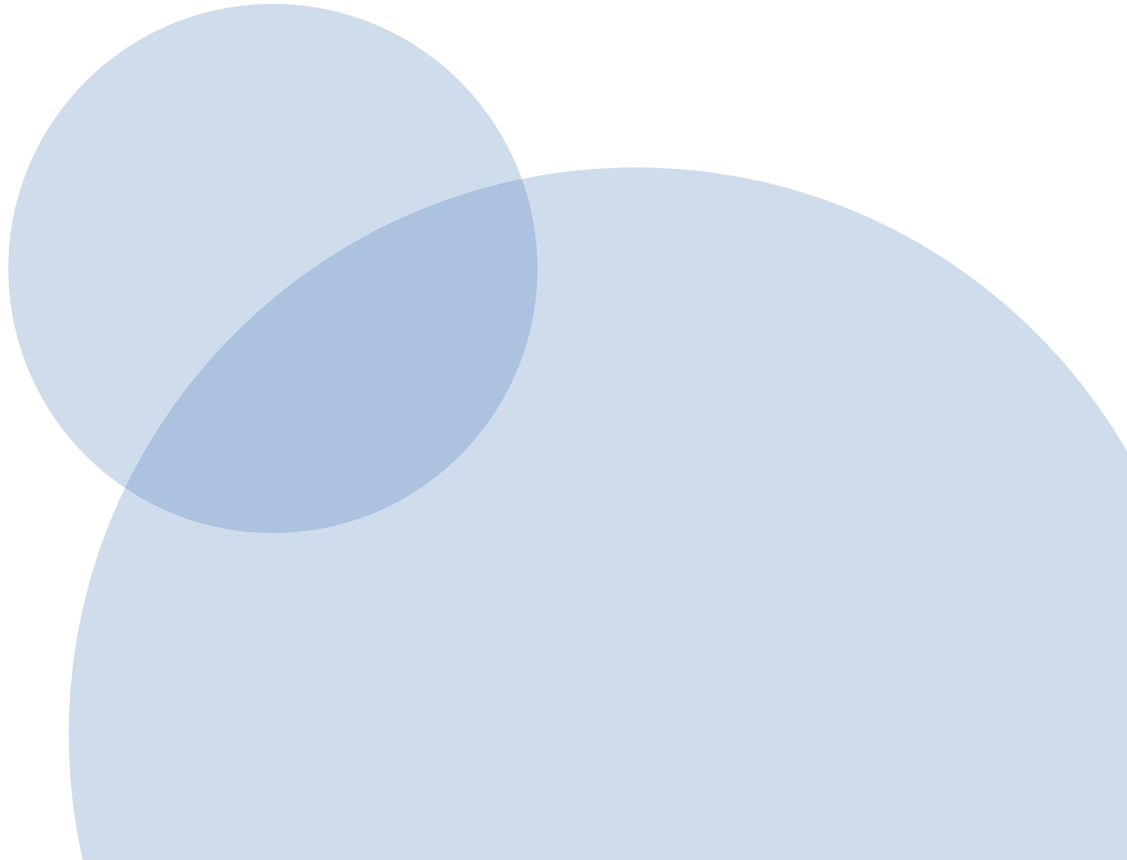


ERHVERVSAKADEMI
AARHUS

Filen

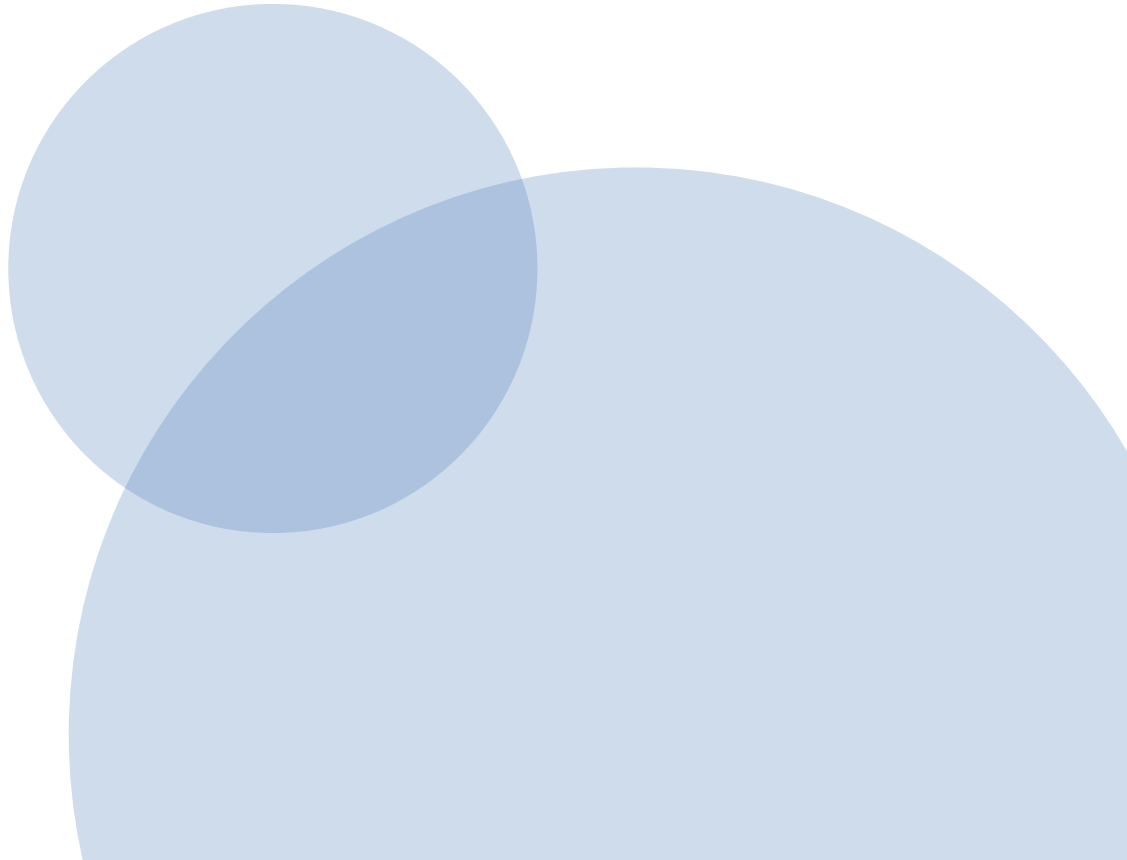
Gem nu en tekst i .rtf format.
Åben filen i hexed.it og undersøg indholdet.

Hvad kan vi se?



Magic bytes

Der er ofte en signatur gemt i de første bytes i en fil.
Denne signatur kaldes også magic bytes.
Den er en indikation på hvilket indehold filen har.



ERHVERVSAKADEMI
AARHUS

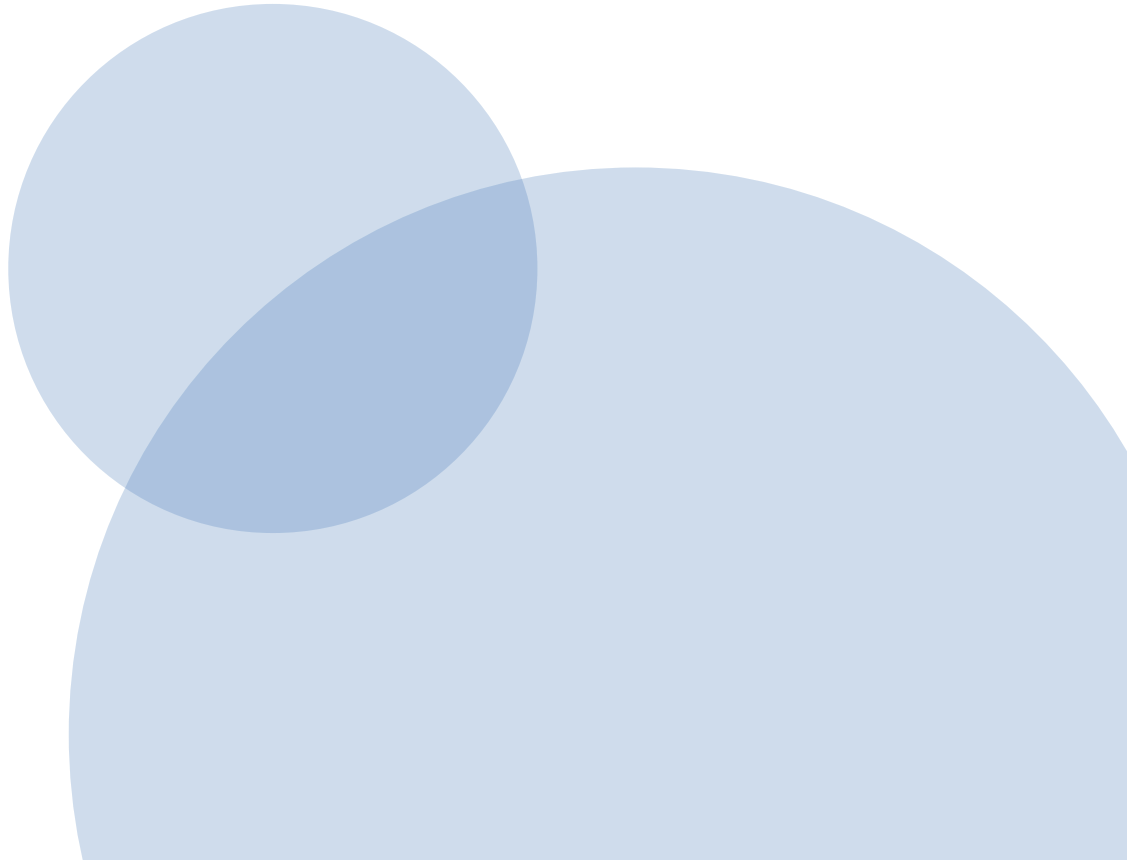
Magic bytes - Øvelse

Der er en medarbejder der ved en fejl er kommet til at slette filtypenavne på filer i en mappe.

I skal undersøge filernes indhold os se om I kan koble der rigtige efternavne på.
I skal ændre filtypenavnet til det i i tror det skal være.
I skal åbne filen efterfølgende for at se om der er det rigtige indhold.

I kan bruge hexed.it som værktøj.

Filerne er på Canvas.

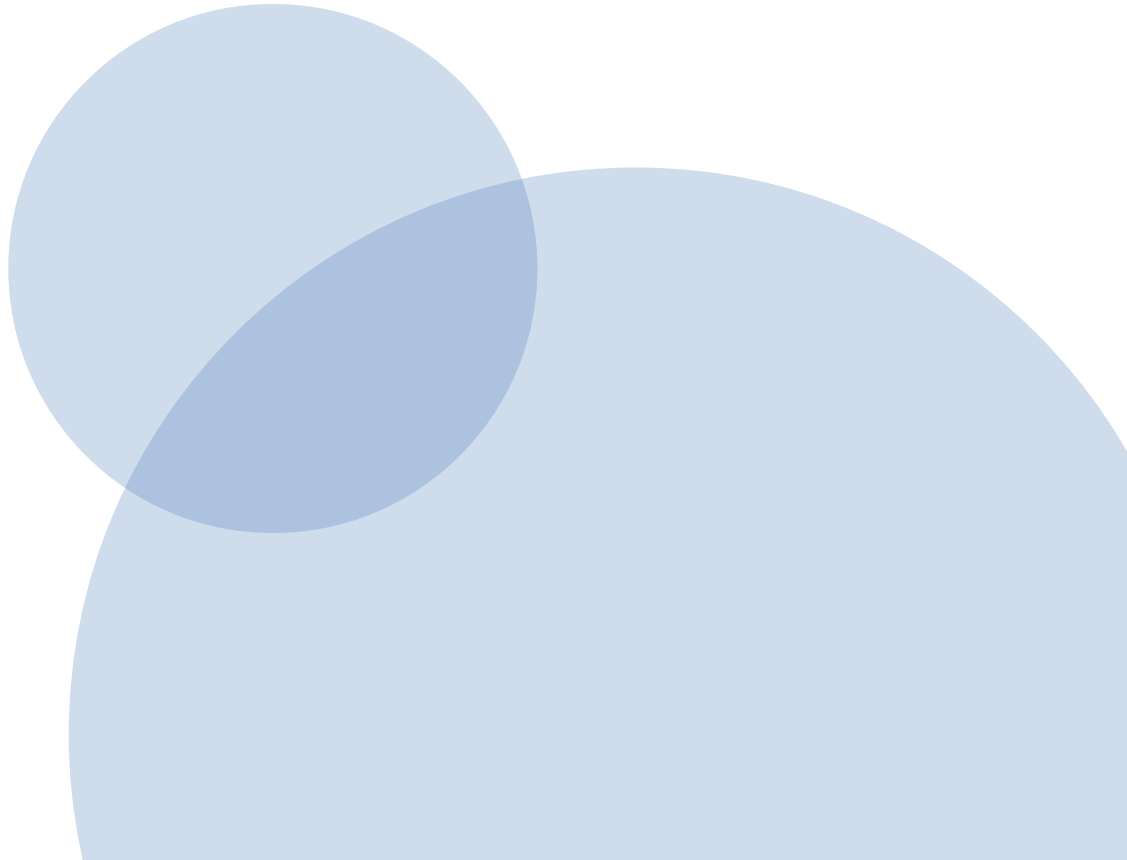


Filstruktur

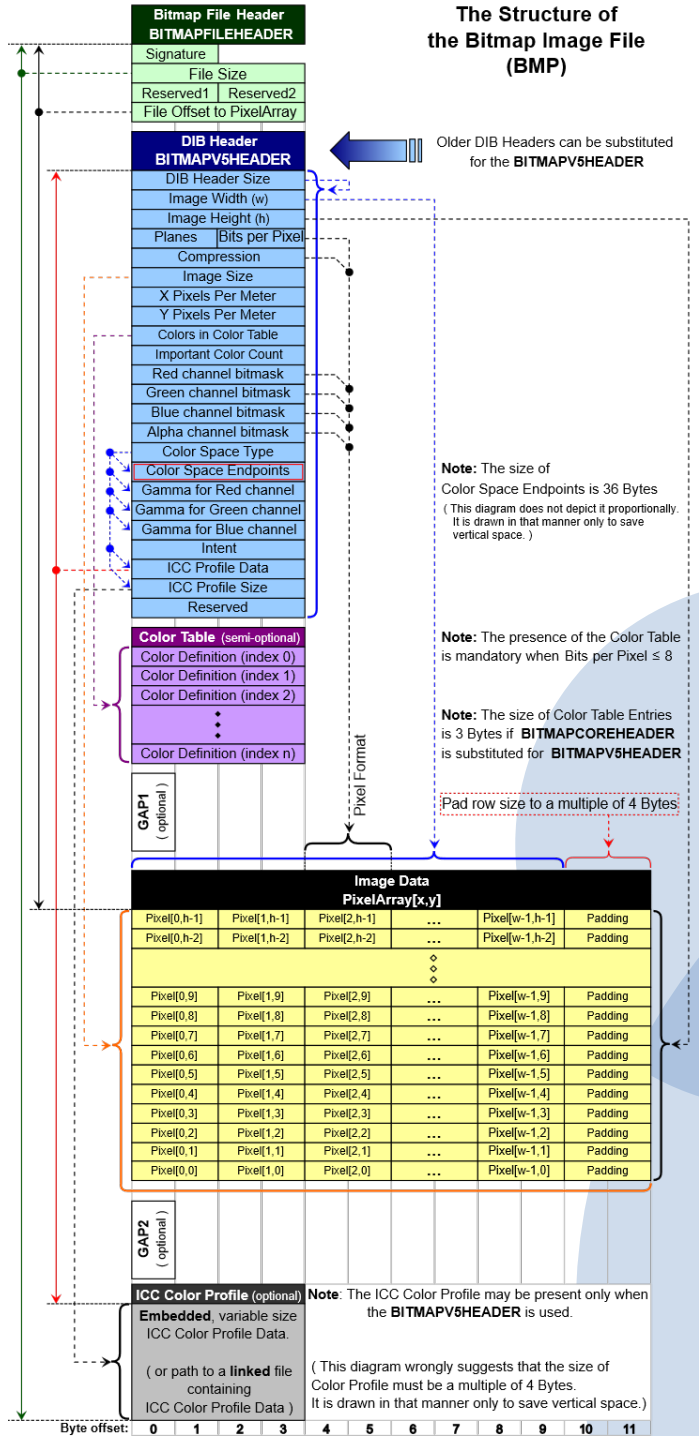
Udover magic bytes er der i nogle filer en header der i detaljer beskriver hvordan filen er opbygget.

Hvilken type data er der i filen og hvor ligger de i filen (på hvilke addresser☺)

Der er ikke en bestem måde at designe en header på.



BMP Header



ERHVERVSAKADEMI AARHUS

BMP Header eksempel

```
# BITMAPFILEHEADER (14 bytes)
42 4D             # 'BM'
7E 00 00 00       # bfSize = 126 bytes
00 00             # bfReserved1
00 00             # bfReserved2
3E 00 00 00       # bfOffBits = 62

# BITMAPINFOHEADER (40 bytes)
28 00 00 00       # biSize = 40
10 00 00 00       # biWidth = 16
10 00 00 00       # biHeight = 16 (bottom-up)
01 00             # biPlanes = 1
01 00             # biBitCount = 1 (monochrome)
00 00 00 00       # biCompression = BI_RGB
40 00 00 00       # biSizeImage = 64 (16 rows × 4 bytes/row)
13 0B 00 00       # biXPelsPerMeter ≈ 72 DPI
13 0B 00 00       # biYPelsPerMeter ≈ 72 DPI
02 00 00 00       # biClrUsed = 2 colors
00 00 00 00       # biClrImportant = 0

# Color Table (2 × RGBQUAD = 8 bytes)
00 00 00 00       # Color 0: black  (B,G,R,0)
FF FF FF 00       # Color 1: white  (B,G,R,0)
```

Bitmap Pixeldata

I vores header er der angivet at billedet er 16x16 pixels og der er to farver (1 bit)

Hardware begrænsninger i ældre computere gør at hver linje i et billede skal være i minimum 32 bit (4 bytes).

Det betyder at antallet af bytes pr linie skal kunne divideres med 4 og give et helt tal.

Så hvis vores billede har pixels pr linie = 16 bit , skal der ”paddes” med 2 bytes, så vi får 4 bytes.

Det betyder at hvis vores billedes første linie skal indeholde en streg med farve 1 skal data være:

	Pixels	fyld-bytes (padding)
Linje 1	0xFF	0x00
Linje 2	0xXX	0x00

ERHVERVSAKADEMI
AARHUS

Bitmap Pixeldata

16×16 1-bpp BMP Pixelmatrix (Printvenlig)

Opgave: Farv firkanterne for at tegne dit billede (1 = farvet pixel, 0 = tom).
Format: 16×16 pixels, 1 bit per pixel, 4 bytes per række (2 data + 2 fyld `00 00`).

```
15 | □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □
14 | □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □
13 | □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □
12 | □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □
11 | □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □
10 | □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □
 9 | □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □
 8 | □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □
 7 | □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □
 6 | □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □
 5 | □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □
 4 | □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □
 3 | □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □
 2 | □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □
 1 | □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □
 0 | □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □
```

Skriv her dine værdier for hver række:

Række	Bits (16)	Databytes (2)	Fyld (00 00)
15	____	__ __	00 00
14	____	__ __	00 00
13	____	__ __	00 00
12	____	__ __	00 00
11	____	__ __	00 00
10	____	__ __	00 00
9	____	__ __	00 00
8	____	__ __	00 00
7	____	__ __	00 00
6	____	__ __	00 00
5	____	__ __	00 00
4	____	__ __	00 00

ERHVERVSAKADEMI
AARHUS

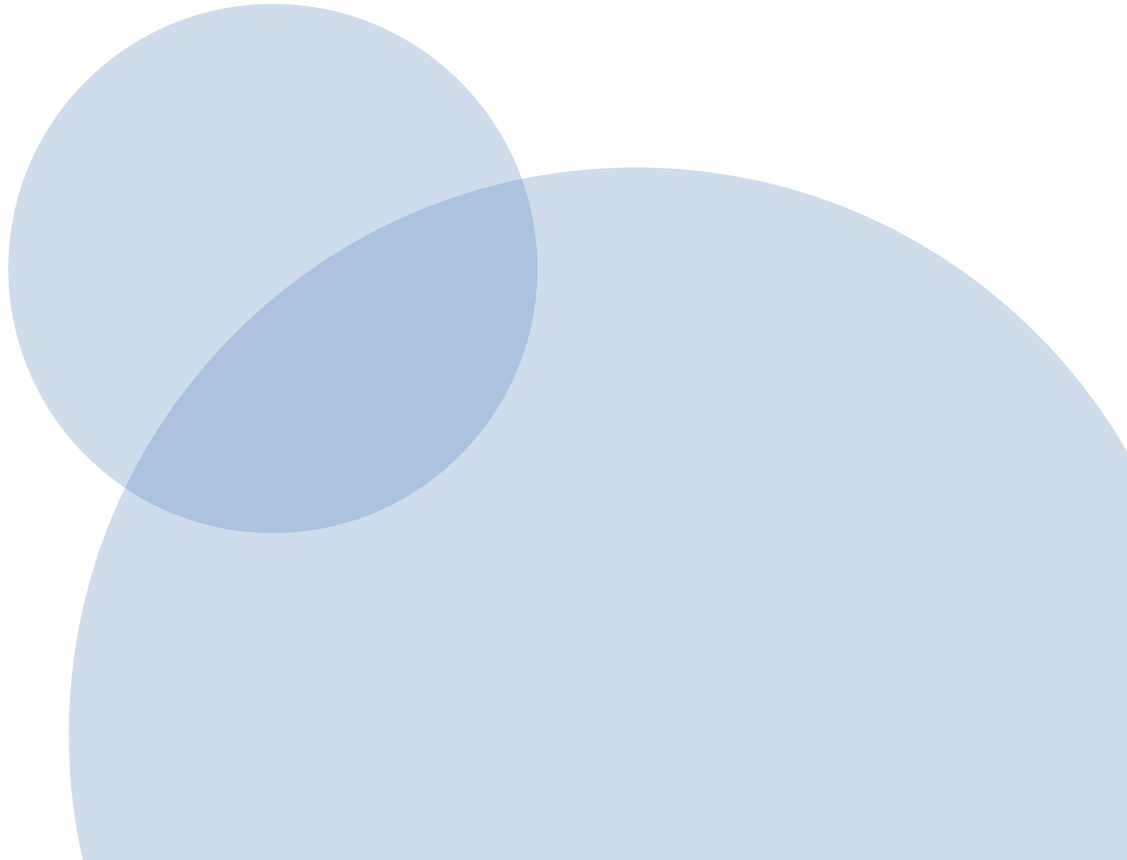
BMP Fil Øvelse

- Headeren i forrige slide udgør beskrivelsen af en bitmap fil på
- 16x16 pixels i to farver. (husk at hver linie er 4 bytes)
- I skal skrive headeren ind i en ny fil i hexed.it
- I skal finde adressen hvor pixel data starter og tilføje pixeldata.
- Brug evt papir og blyant for at lave et billede.

ERHVERVSAKADEMI
AARHUS

Steganografi Lite

Steganografi (af græsk steganos, "dækket", og gráphein, "at skrive") er et underemne inden for kryptologien, der beskæftiger sig med at skjule beskeder i en eller anden form for kontekst. (Wiki)



ERHVERVSAKADEMI
AARHUS

Steganografi Lite

Da vi har adgang til vores data i en hex-editor, er det muligt at tilføje en hemmelig meddelelse efter fildata.

Øvelse:

- Åbn 'hemmelig.png' i hex-editor.
- Navigér til filens slutning (efter IEND).
- Tilføj ASCII-teksten: 'HEMMELIG:BESKED-123'.
- Gem og verificér at billedet stadig kan åbnes.
- Send billedet til din nabo☺

Hex hacking

En af de store sårbarheder i software er muligheden for at ændre compilerede programmers virkemåde.

Hvis man har kendskab til, eller kan søge sig frem til vigtige variabel addresser, kan de ændres og dermed ændre programmets virkemåde.

I spilbranchen har det i de tidligere år været brugt at omgå beskyttelse eller spil variabler som liv og score.

De typiske værktøjer er en hex editor og en disassembler.

ERHVERVSAKADEMI
AARHUS

Hack the administration☺

Download adminsystem.exe fra Canvas modulet.’

Start programmet og se om i kan få adgang.

Hvis det ikke lykkes, prøv at åbne filen i hexed.it

Se om I kan finde det rigtige password.

ERHVERVSAKADEMI
AARHUS

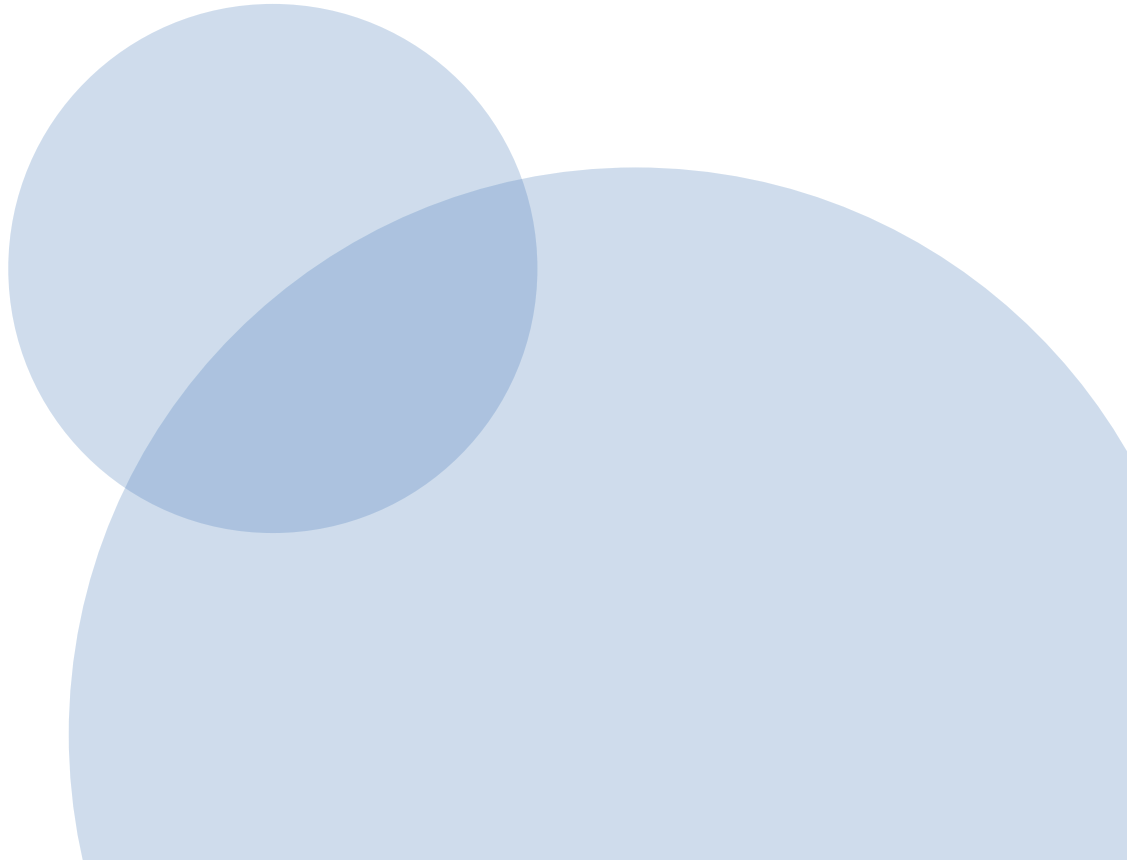
Eternal Life

Download spillet Invaders.exe (i skal udpakke .rar filen)

Test om det virker☺

Åben spillet (.exe filen) i hex editor.

Led efter mønstre og se om i kan ”cracke” spillet og opnå flere liv☺



ERHVERVSAKADEMI
AARHUS

Opgave:

I skal designe jeres eget filformat og lave et eksempel i hex-editoren.

Find ud af hvad jeres filformat skal indeholde.

Find ud af hvad jere filtype-betegnelse skal være (filnavn.abc)

Tilføj en header.

Start evt. med magic bytes.

Lav en beskrivelse af headeren. Hvad betyder de forskellige bytes i headeren.