

Codierung: Bilder

1. Schwarz-Weiß-Pixelbilder

Einfache schwarz-weiß-Pixelbilder lassen sich sehr einfach darstellen. Hierbei muss lediglich angegeben werden, welche Pixel eingeschaltet (weiß) bzw. ausgeschaltet (schwarz) sind. Ein Bild kann so als lange Bit-Kette geschrieben werden.

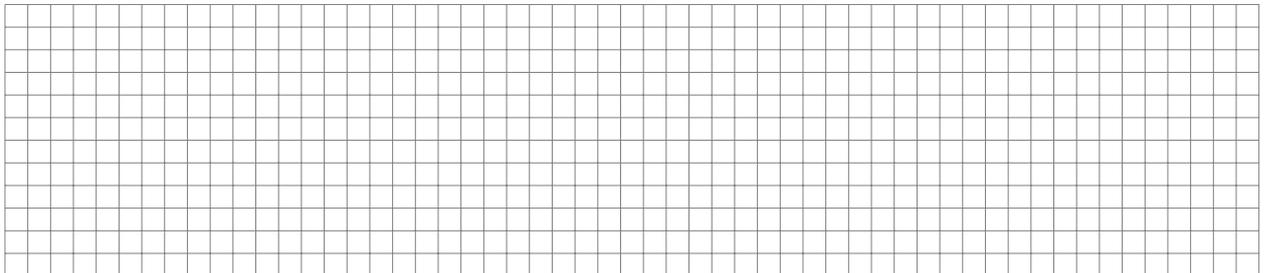
Damit der Computer jedoch weiß, wann eine neue Zeile beginnt, muss zunächst die Breite des Bildes angegeben werden. Damit ein Bild richtig abgespeichert werden kann und auch wieder richtig gelesen werden kann müssen wir uns ein eigenes Dateiformat definieren:

- Im ersten Byte der „Datei“ steht die Breite des Bildes
- Im zweiten Byte steht die Höhe des Bildes
- Anschließend folgen die Bits, die angeben, ob ein Pixel an oder aus ist.
- Ist die Länge der Bitkette kein Vielfaches von 8, so werden die restlichen Bits mit Nullen aufgefüllt.

2. Aufgabe

Entschlüssele mit obigen Angaben folgendes „Bild“:

00000110 00001001 00000001 00100100 10010010 00000010 00011000 01010010 00110000



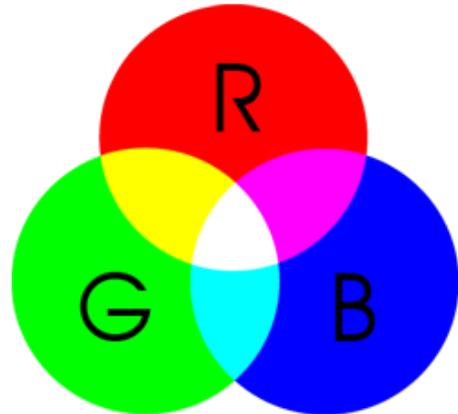
3. Aufgabe

Erstelle selbst ein Bild, codiere es in eine Bitfolge und gib es deinem Nachbarn zur Entschlüsselung.

4. Farbige Bilder

Auf Dauer werden Schwarz-Weiß-Bilder recht langweilig und es muss etwas Farbe ins Spiel kommen. Hier nutzen wir das RGB-System: Jeder Pixel besteht dabei aus 3 Farben rot, grün und blau und können folgendermaßen gemischt werden:

- Rot
- Grün
- Blau
- Rot + Grün = Gelb
- Rot + Blau = Magenta
- Blau + Grün = Cyan
- Rot + Grün + Blau = Weiß



Damit definieren wir ein neues Dateiformat:

- Im ersten Byte der „Datei“ steht die Breite des Bildes
- Im zweiten Byte steht die Höhe des Bildes
- Anschließend folgen die Pixel, dabei gehören immer 3 Bits zu einem Pixel:
 - das erste Bit gibt an, ob der rote Anteil dabei angeschaltet ist
 - das zweite Bit gibt den grünen Anteil an
 - das dritte Bit gibt den blauen Anteil an
 - z. B., eine Bitfolge 000 bedeutet schwarz, 001 bedeutet blau, 110 bedeutet gelb, 111 bedeutet weiß.
- Ist die Länge der Bitkette kein Vielfaches von 8, so werden die restlichen Bits mit Nullen aufgefüllt.

5. Aufgabe

Entschlüssele mit obigen Angaben folgendes „Bild“:

```
00000110 00000110 00100000 00000000 01000111 11011011 10000001 10100100 11000000 01100100
10110000 00011111 01101110 00001000 00000000 00010000
```

