Solutions

Each answer shows the steps, then the **result**.

2.1

- colour depth = the number of **bits used per pixel**
- size = width × height × colour depth

2.2

- size in bits = width × height × colour depth

$$1024 \times 768 \times 8 = 6\,291\,456 \text{ bits}$$

- ÷8 for bytes, ÷1024 for KiB

$$\frac{6\,291\,456 \text{ bits}}{8 \times 1024} = \mathbf{768} \text{ KiB}$$

2.3

- size in bits = sampling rate × resolution × duration × channels

$$16\,000 \text{ Hz} \times 8 \text{ bits} \times 30 \text{ s} \times 1 = 3\,840\,000 \text{ bits}$$

- ÷8 for bytes, ÷1024 for KiB

$$\frac{3\,840\,000 \text{ bits}}{8 \times 1024} = \mathbf{468.75} \text{ KiB}$$

2.4

- any one: it stays **sharp at any size** (a logo is used large and small); the file is small

3.1

(a) size in bits = 1920 × 1080 × 24

$$1920 \times 1080 \times 24 = 49\,766\,400 \text{ bits}$$

- $\div 8 \div 1024 \div 1024$ for MiB

$$\frac{49\,766\,400 \text{ bits}}{8 \times 1024 \times 1024} \approx \mathbf{5.93} \text{ MiB}$$

(b) e.g. **lower the colour depth** (or the resolution)

- smaller file, but fewer colours / less detail

3.2

(a) $t = 3$ min = 180 s; stereo = 2 channels

- size in bits = rate × resolution × t × channels

$$44\,100 \text{ Hz} \times 16 \text{ bits} \times 180 \text{ s} \times 2 = 254\,016\,000 \text{ bits}$$

- $\div 8 \div 1024 \div 1024$ for MiB

$$\frac{254\,016\,000 \text{ bits}}{8 \times 1024 \times 1024} \approx \mathbf{30.28} \text{ MiB}$$

(b) fewer samples per second

- the file is roughly **half the size**, but high-pitched sounds are lost / quality drops

3.3

(a) any two of: centre position (x, y); radius; line/fill colour; line width

(b) **bitmap**

- a painting has continuous, per-pixel tonal detail that cannot be described as a few shapes

3.4

- a bitmap has a fixed number of pixels; enlarging it makes each pixel bigger, so the squares show
- a vector stores shapes and is **re-drawn** at the new size, so edges stay smooth

3.5

(a) $t = 45$ s; stereo $= 2$ channels

- size in bits = rate $\times$ resolution $\times$ t $\times$ channels

$$44\,100 \text{ Hz} \times 16 \text{ bits} \times 45 \text{ s} \times 2 = 6.35 \times 10^7 \text{ bits}$$

- $\div 8$ for bytes

$$\frac{6.35 \times 10^7 \text{ bits}}{8} = 7.94 \times 10^6 \text{ bytes}$$

$$\frac{7.94 \times 10^6 \text{ bytes}}{1\,048\,576 \text{ bytes MiB}^{-1}} = \mathbf{7.6} \text{ MiB}$$

(b) halving both the sampling rate and the resolution divides the size by **four**, to about 1.9 MiB

- the lower rate captures fewer samples per second, so high frequencies are lost
- the smaller resolution records each sample more coarsely, so the waveform is a poorer match and quantisation noise increases

(c) size in bits = width $\times$ height $\times$ colour depth

$$600 \times 400 \times 24 = 5\,760\,000 \text{ bits} = 720\,000 \text{ bytes}$$

$$\frac{720\,000 \text{ bytes}}{1024 \text{ bytes KiB}^{-1}} = \mathbf{703} \text{ KiB}$$

(d) the file also stores **metadata** in a header — the dimensions, the colour depth, the file format

- it may include a colour palette or padding, none of which the pixel calculation counts

(e) run-length encoding replaces each run of identical pixels with **one colour value and a count**

- large flat areas of one colour become very short records, so the file shrinks a great deal
- in a photograph adjacent pixels almost always differ slightly
- so runs are one pixel long and the encoding stores a count beside every pixel, which can make the file **bigger** than the original