Storing Text – ASCII and Unicode

We now know a lot about how to store numbers (integers and fixed point decimals) in binary form. There are lots of other types of data we need to store as well.

Text is usually stored in one of two formats –

- ASCII (American Standard Code for Information Interchange)
- Unicode (usually uses UTF-8 encoding)

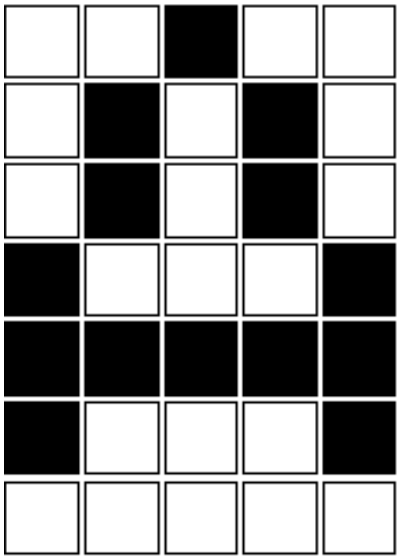
Both methods use a lookup table to match characters with a numeric code. ASCII uses 7 bits, allowing 128 values of which 94 are printable characters.

UTF-8 can be 1 byte in length when matching ASCII characters or up to 4 bytes in length if necessary.

Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
32	20	040		Space	64	40	100	@	@	96	60	140	`	`
33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

Binary Representation of Bitmap Images

All bitmap images are stored as an array of pixels. A monochrome bitmap will store a 1 for a black pixel and 0 for a white pixel (or vice-versa depending on the encoding protocol).



This image could be represented by the following 35 binary digits (5 bytes):

00100 01010 01010 10001 11111 10001 00000

It would also be necessary to store the dimensions of the image.

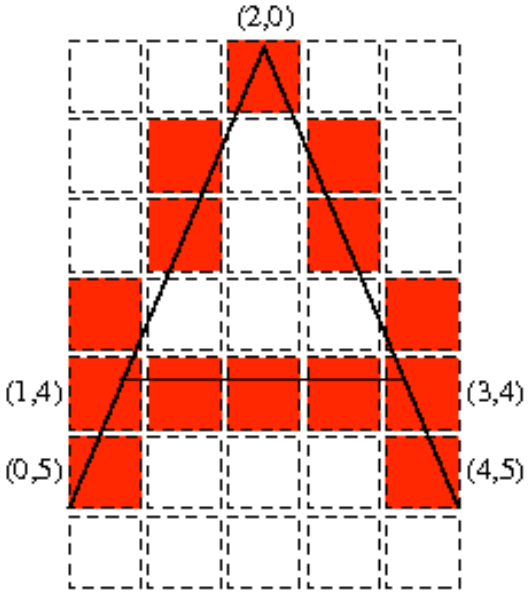
If the image were in colour, using a colour palette of 256 possible shades, each pixel would need to translate to a value between 0 and 256 (8 bits). Thus we would need 35 bytes to store the image.

It is common for colours to be recorded by quantity or Red, Green and Blue (RGB) and this is stored using 3 bytes per pixel – so we would need 105 bytes to store the image.

Bitmap images can be encoded in order to reduce the file size. Examples of encoded bitmaps in clued JPG, PNG and GIF file types.

Binary Representation of Vector Images

Vector images do not store the data by examining pixels, but are a set of instructions for drawing a geometric shape.



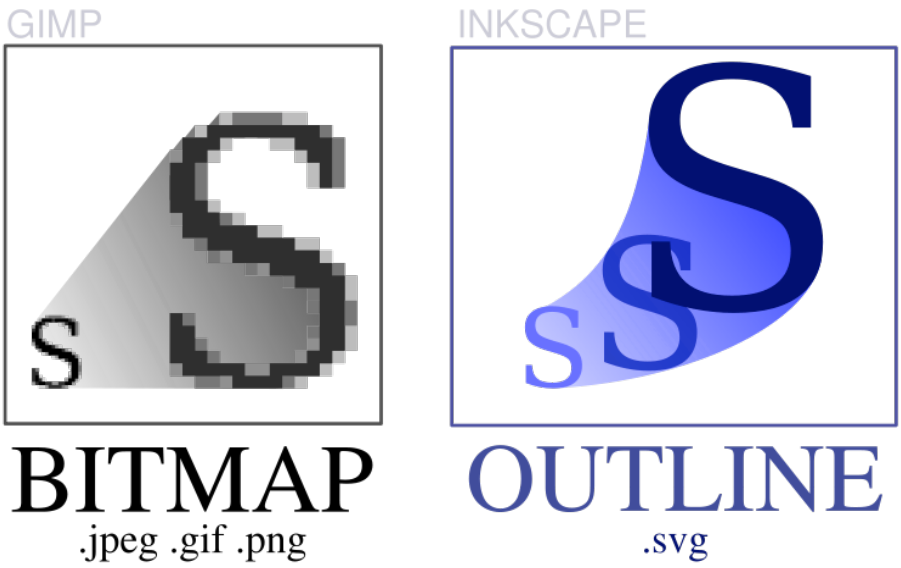
This image shows how a vector graphic (the 3 black lines) could be converted to a bitmap.

Existing images (including photographs) cannot be easily converted to a vector graphic, and any attempt to do so will spoil the realism of the image.

Vector graphics can be resized by any amount without losing quality – making them ideal for fonts, logos and clip-art – but are not suitable for realistic images.

They must also be converted into bitmaps during printing (unless printed to a plotter) or displaying on a screen.

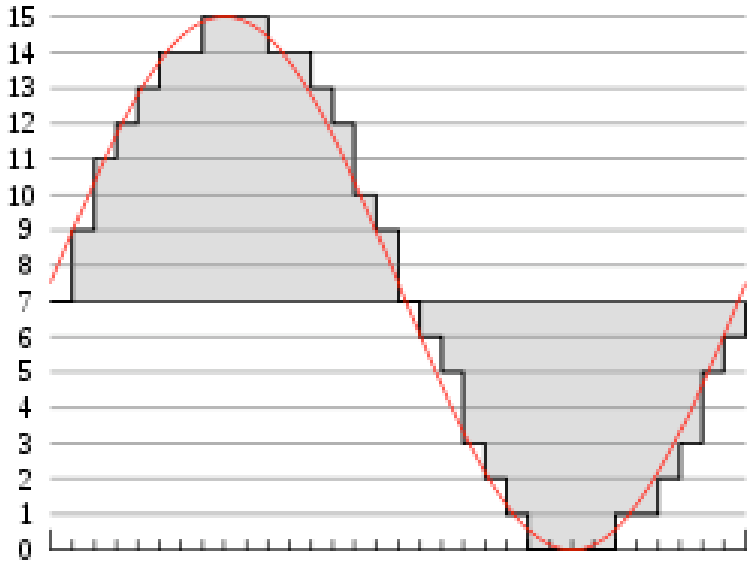
Vector graphics take up less storage space than bitmaps, generally, but require more processing power to repeatedly redraw the image.



Binary Representation of Sound

In order to digital record sound, a microphone is used, a device that converts the analogue sound into a digital form by altering the output signal from the device.

In order to store this digitally, the voltage is **sampled** at frequent intervals (typically 48 000 times per second, or 48kHz) and stored as a binary code (typically 16 or 32 bits per sample).



This equates to slightly over 1.5 million bits per second, or 88 MB per minute. An 80 minute album stored in such a manner would require 6.9 GB of storage.

There are three ways of altering this:

Sample Rate: By reducing the sample rate (e.g. to 22kHz), you reduce the amount of data you need to store. This has quite serious effects on the quality of the audio (see the next page).

Sample Depth: Much as with colour, you can reduce the precision of each datum, using only 8 bits per sample instead of 32, for example.

Compression: Ideally you would sample at the highest practical rate you can and these use an encoding algorithm to reduce the output file size (this is why Audacity projects are much bigger in size than the files output). This can be

Lossless compression (e.g. FLAC [and potentially MP3/AAC/Ogg Vorbis files]) or

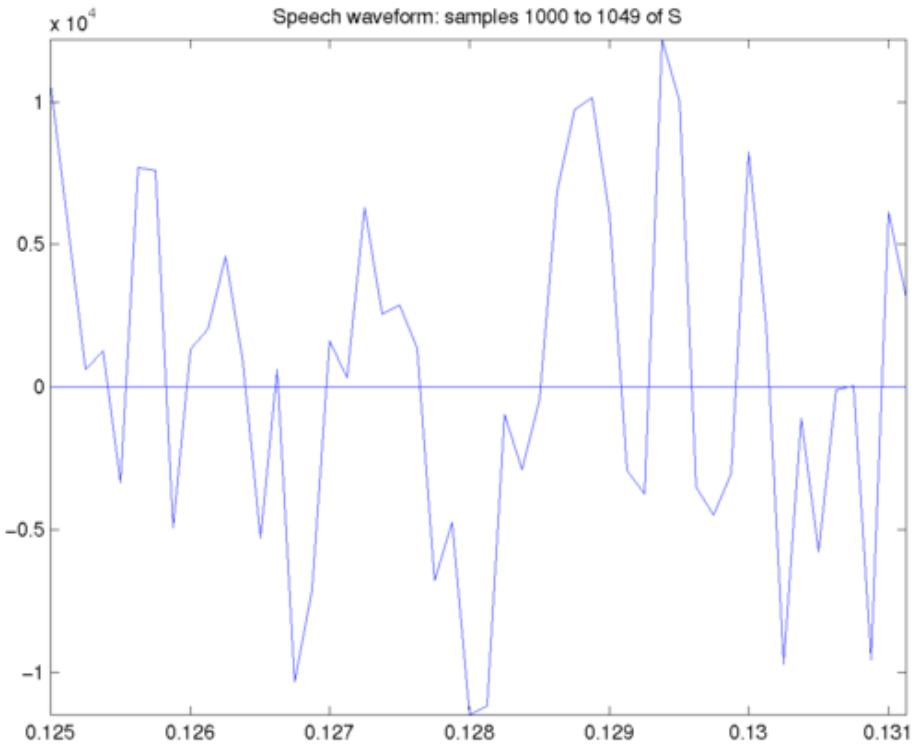
Lossy compression (e.g. most MP3, AAC and Ogg Vorbis files).

WAV and AIFF files are used for storing uncompressed audio.

Nyquist's Theorem

There is a very good reason why we sample music at either 44.1KHz (CD audio) or 48KHz (professional audio). Humans can generally hear sounds in the frequency range of 20Hz – 20,000Hz (20KHz). To make sure we capture all of the sound, Nyquist realised you would have to take a sample AT LEAST twice as often as the frequency of that sound (so double it and add a bit) in order to make sure you don't miss an entire cycle.

Since the maximum human hearing range is about 20KHz, doubling it and adding a bit gives us 40-odd KHz. 44.1KHz is a bit of a quirk of fate as that fits nicely into a PAL or NTSC recording and 48KHz is a round number, digitally.



The telephone network generally uses 8KHz sampling, as this is plenty to hear speech, but reduces the bandwidth required massively. This is why music in the background sounds rubbish over the phone.

VoIP (e.g. Skype) typically uses 8KHz too, sometimes 16KHz, for similar reasons and with similar results.

DVDs often include audio sampled at 96KHz. If you have a soundproof studio with a professionally set-up audio system and you concentrate REALLY hard, then you might just be able to tell the difference.