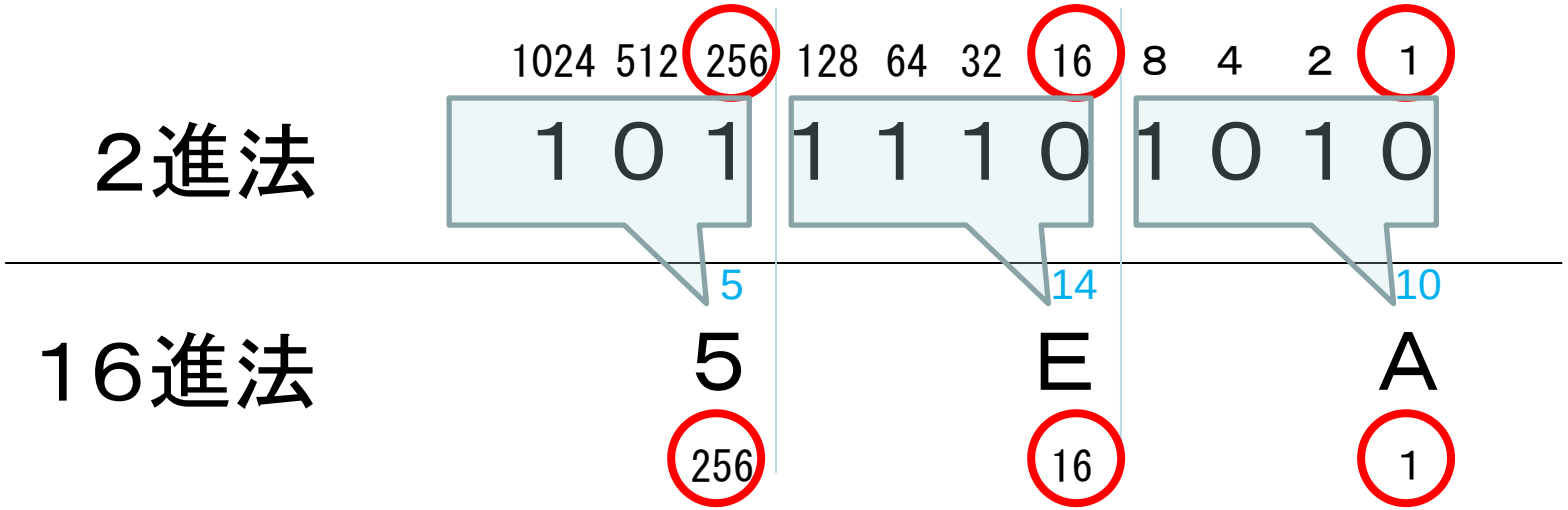
文字のデジタル化

情報Ⅰ　第31回授業

05コンピュータによる情報の表現

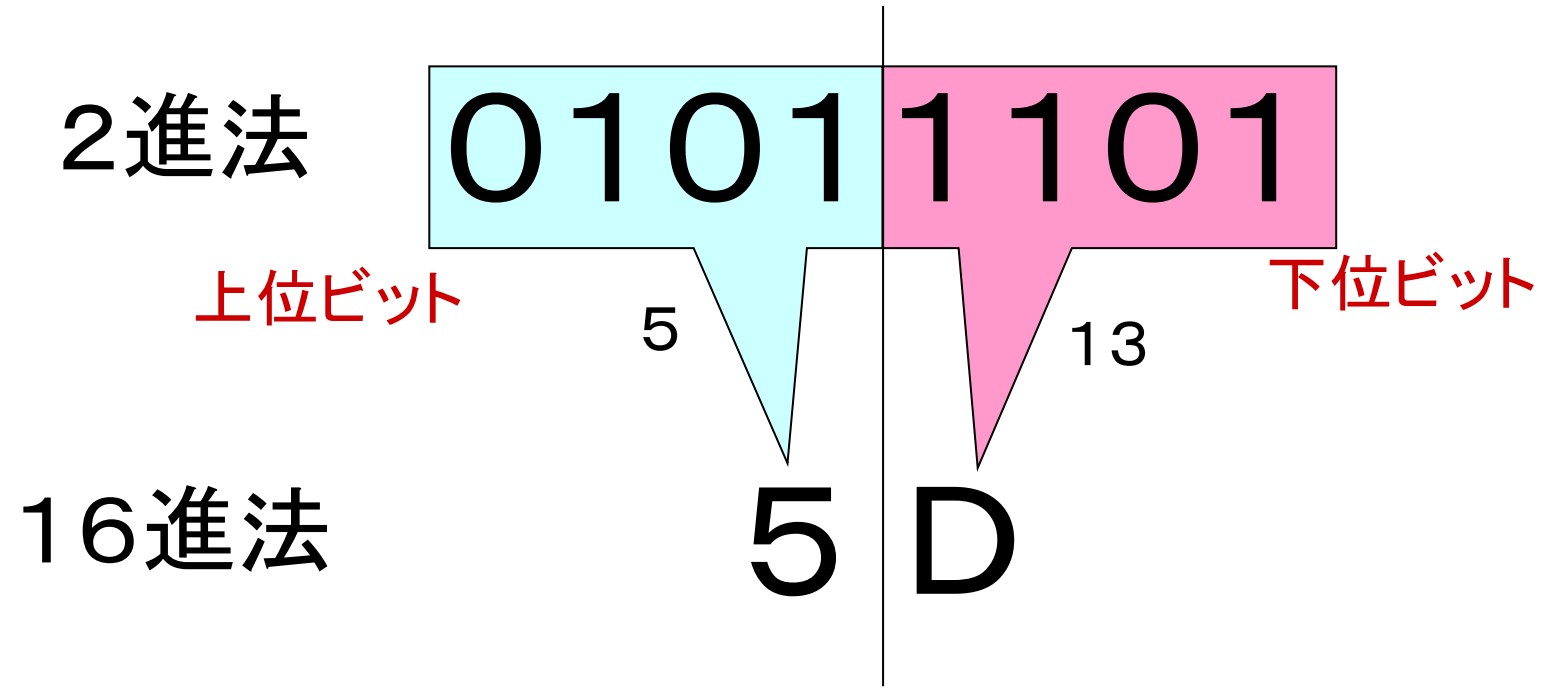
対応ファイル：22exp31.xls

＜復習＞2進法と16進法



２進数と16進数では、繰り上がりのタイミングが同じ！
→　２進4ケタをそのまま16進に変換して表せられる！

＜復習＞ 2進法と16進法



- このように、2進8文字は、16進2文字で表すことができる。
- 人間にとっては、2進法よりも16進法の方が扱いやすい。

＜復習＞情報の量

- コンピュータでは、0と1の電気信号に情報を変換、すなわち2進法で処理をしている。

- 2進法の数1ケタを「1bit（ビット）」とし、情報の量の単位とする。

＜復習＞「2進法」と「場合の数」

1ケタにつき
0 or 1の
2パターン

2進　　1　1　1　1　1

場合の数　　2×2×2×2×2　通り

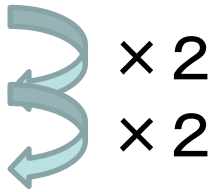
5bitの情報量 ⇐ 2^{5}

2進法での
ケタ数と
同じ

2進法での「ケタ数」が情報の量（bit数）と考えて良い

＜復習＞ 2進法と場合の数

bit	場合の数
1	2
2	4
3	8
4	16
5	32
6	64
7	128
8	256
9	512
10	1024



「一つ上のセル」を2倍することを
コピーしていけばよい

例）（L5の場所）・・　=L4＊2
　　L5を「コピー」、
　　L6からL28まで「貼り付け」

これらのことから

　5bitの情報量では、 32　通り
のものが区別でき、

　512通りのものを区別するには　9　bit

　100通りのものを区別するには　7　bit
の情報量が必要であることがわかる

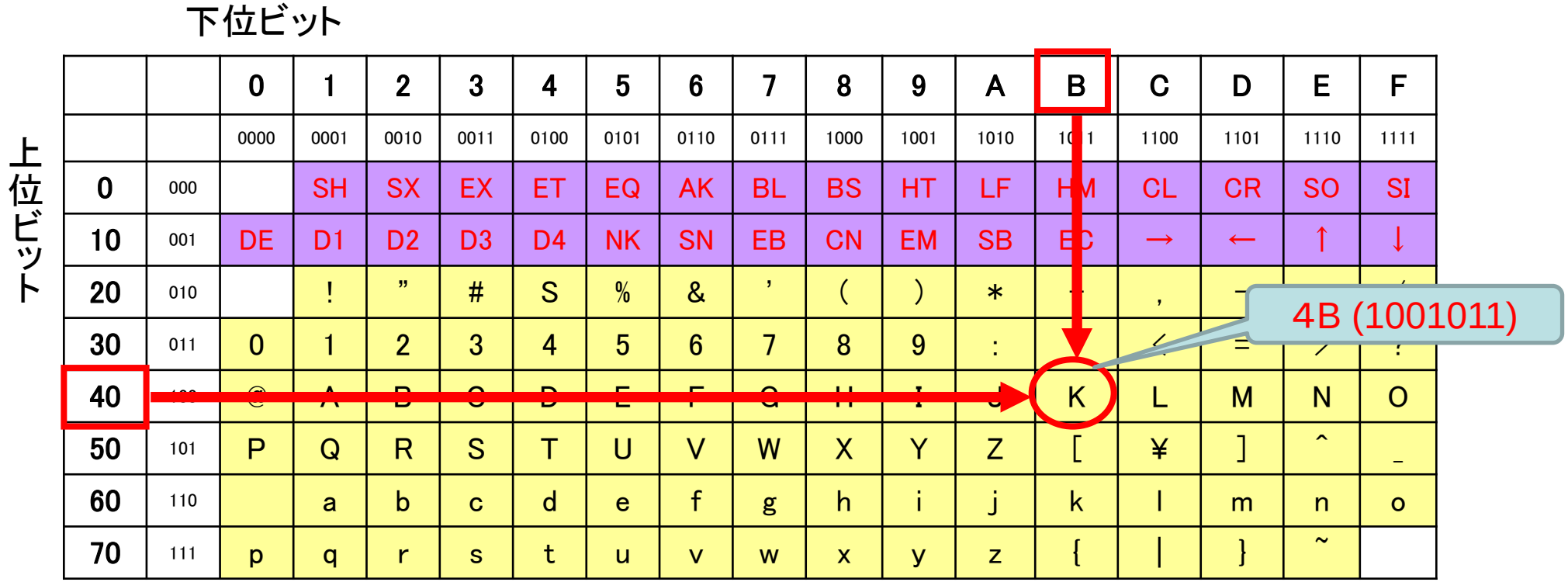
文字コード（p.78）

文字を、それぞれ一定の長さの「数値（0と1）の列」に割り当てる。
一つひとつの文字に割り当てられた「数値の列」を文字コードという。
文字と文字コードの対応関係を文字コード体系という。

2^7＝128通り

1963年　7bit・・・ASCIIコード（英数字記号・制御記号）

ASCIIコード（p.78）



		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
0	000		SH	SX	EX	ET	EQ	AK	BL	BS	HT	LF	HM	CL	CR	SO	SI
10	001	DE	D1	D2	D3	D4	NK	SN	EB	CN	EM	SB	EC	→	←	↑	↓
20	010		!	”	#	S	%	&	’	(	)	*	+	,	−	.	/
30	011	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	100	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	101	P	Q	R	S	T	U	V	W	X	Y	Z	[	¥	]	^	_
60	110		a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	111	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	

※5Cについては、本来は「＼」（バックスラッシュ「/」の向きが逆のもの）」が割り当てられているが、
　日本語環境では、「￥」（円マーク）が表示される。
※教科書の表は、「上位ビット」と「下位ビット」の縦横が逆なので注意すること。

文字コード（p.78）

文字を、それぞれ一定の長さの「数値（0と1）の列」に割り当てる。
一つひとつの文字に割り当てられた「数値の列」を文字コードという。
文字と文字コードの対応関係を文字コード体系という。

2^7＝128通り

1963年　7bit・・・ASCII（アスキー）コード（英数字記号・制御記号）

1969年　1Byte（ 8bit）・・・JIS X 0201＜半角＞

2^8＝256通り

（ASCIIコード＋カナ文字）

JIS X 0201（拡張ASCIIコード）

下位ビット

上位ビット		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
	0000																
10	0001	DE	D1	D2	D3	D4	NK	SN	EB	CN	EM	SB	EC	→	←	↑	↓
20	0010		!	”	#	S	%	&	’	(	)	*	+	,	−	.	/
30	0011	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	0100	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	0101	P	Q	R	S	T	U	V	W	X	Y	Z	[	¥	]	^	_
60	0110		a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	0111	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	
80	1000																
90	1001																
A0	1010		。	「	」	、	・	ヲ	ァ	ィ	ゥ	ェ	ォ	ャ	ュ	ョ	ッ
B0	1011	ー	ア	イ	ウ	エ	オ	カ	キ	ク	ケ	コ	サ	シ	ス	セ	ソ
C0	1100	タ	チ	ツ	テ	ト	ナ	ニ	ヌ	ネ	ノ	ハ	ヒ	フ	ヘ	ホ	マ
D0	1101	ミ	ム	メ	モ	ヤ	ユ	ヨ	ラ	リ	ル	レ	ロ	ワ	ン	゛	゜
E0	1110																
F0	1111																

拡張部分

「漢字」について

常用漢字

・・・日常の使用に必要なものとして定められた漢字

小学校（1006字）

中学校（1130字）　　計2136字（2010年）

JIS第一水準（2965字）、第二水準（3390字）

・・・「日本工業規格（JIS）」で決められた、主にコンピュータでの利用を想定した漢字の規格。

文字コード（p.78）

文字を、それぞれ一定の長さの「数値（0と1）の列」に割り当てる。
一つひとつの文字に割り当てられた「数値の列」を文字コードという。
文字と文字コードの対応関係を文字コード体系という。

2^7＝128通り

1963年　7bit・・・ASCIIコード（英数字記号・制御記号）

1969年　1Byte（ 8bit）・・・JIS X 0201＜半角＞

2^8＝256通り （アスキーコード＋カナ文字）

1976年　2Byte（16bit）・・・JIS X 0208＜全角＞

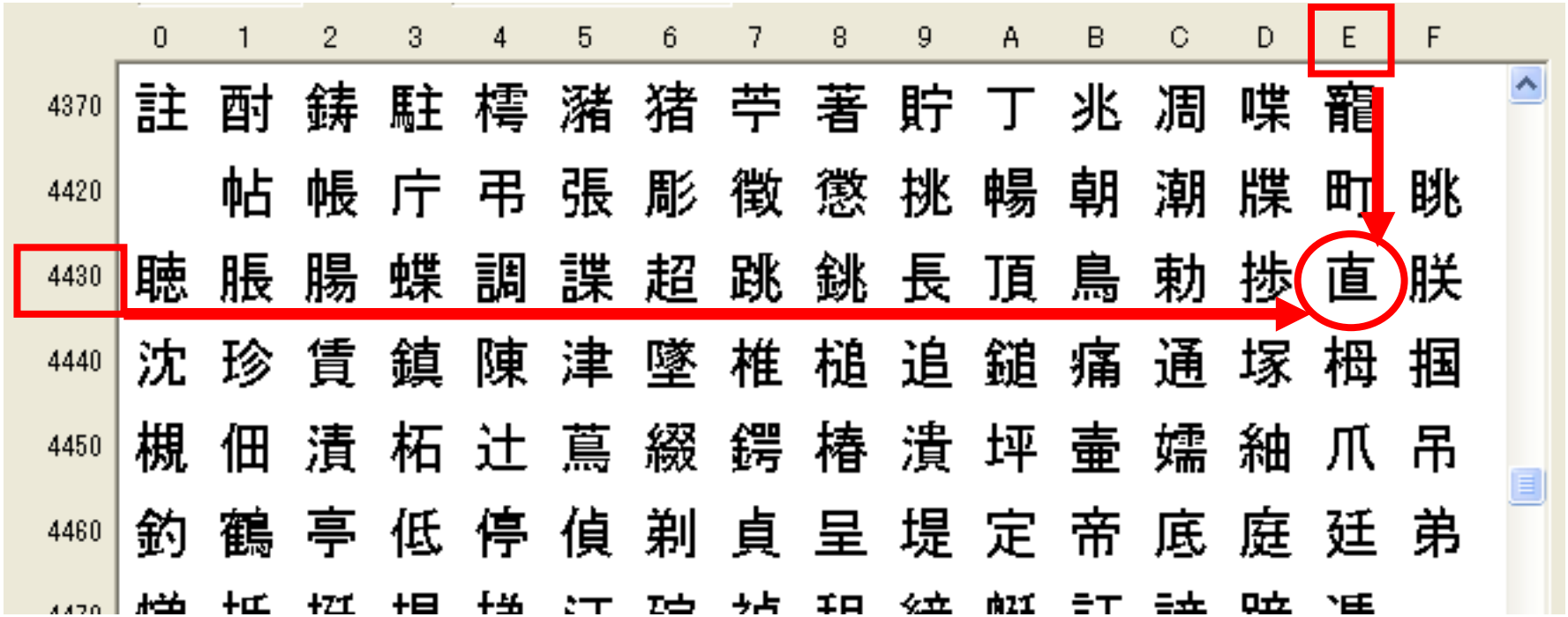
2^{16}＝65536通り （漢字など）

JIS X 0208

始めの3ケタが左側の数字に、最後の1ケタの数が上の数字に対応している。

→　始めの3ケタで「行」を探し、残りの1ケタを横に探せばよい。

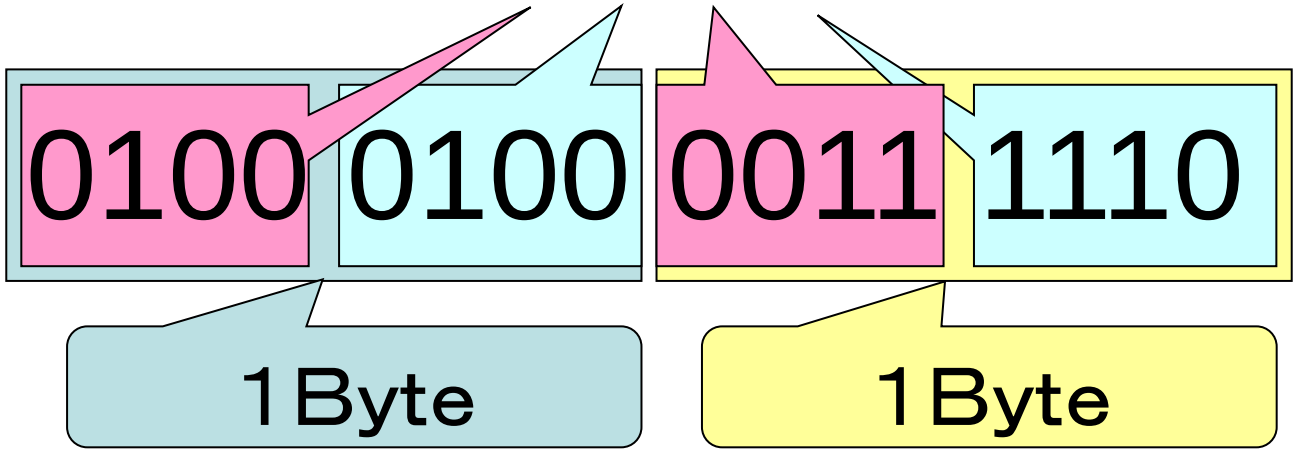
例）　443E



漢字の文字コード

直

443E



漢字１文字について、2Byteの情報量で区別している

漢字の文字コードと16進・2進

塞	翁	が	馬
3A49	3227	242C	474F
0011101001001001	0011001000100111	0010010000101100	0100011101001111

「半角」と「全角」

日本語入力を「オン」にすると、
「半角（直接：1Byte）入力」から
「全角（日本語変換：2Byte）入力」モードへ

見た目はほとんど同じでも、番号が違う！！
→ コンピュータ内では、「全く違う文字」として認識される！！
普段から「半角」と「全角」の違いに気をつけよう！！

エンコーディング（符号化）

- データを0と1の文字列にすることをエンコーディングという。
- 特に、文字を0と1の文字列に当てはめる時、さまざまな方式（エンコーディング方式）が考案され、利用された。
  - JIS X 0208　をもとに、いくつかの組織が独自の文字コード体系を作った。
  - ISO-2022-JP、Shift_JIS、EUC-JP　など
- 0と1の文字列を元のデータに戻すことをデコーディングという。

文字コード（p.78）

文字を、それぞれ一定の長さの「数値（0と1）の列」に割り当てる。
一つひとつの文字に割り当てられた「数値の列」を文字コードという。
文字と文字コードの対応関係を文字コード体系という。

2^7＝128通り

1963年　7bit・・・ASCIIコード（英数字記号・制御記号）

1969年　1Byte（ 8bit）・・・JIS X 0201＜半角＞

2^8＝256通り（アスキーコード＋カナ文字）

1976年　2Byte（16bit）・・・JIS X 0208＜全角＞

2^{16}＝65536通り（漢字など）

その後、ISO-2022-JP,Shift_JIS,EUC-JPなどの異なる方式が考案

1993年　・・・Unicode（UCS-2）＜多国語処理＞

2000年　2Byte（16bit）・・・JIS X 0213：2000＜全角＞

現在　1～6Byte（8～48bit）・・・Unicode（UCS-4）、UTF-8、UTF-16

フォント（p.79）

文字の「形（＝グリフ）」を集めた「種類」
大きさや色情報等を含めて呼ぶこともある

「明朝体（みんちょうたい）」
「ゴシック体」
「行書体(ぎょうしょたい)」

【発展】プロポーショナルフォント

☆プロポーショナルフォント

文字本来の形に合わせ、横幅をバランスよく変えたもの。（→ 欧米の文化）
（「i」や「j」に注目）

例）ＭＳＰゴシック abcdefghijklmnopqrstuvwxyz

☆等幅フォント

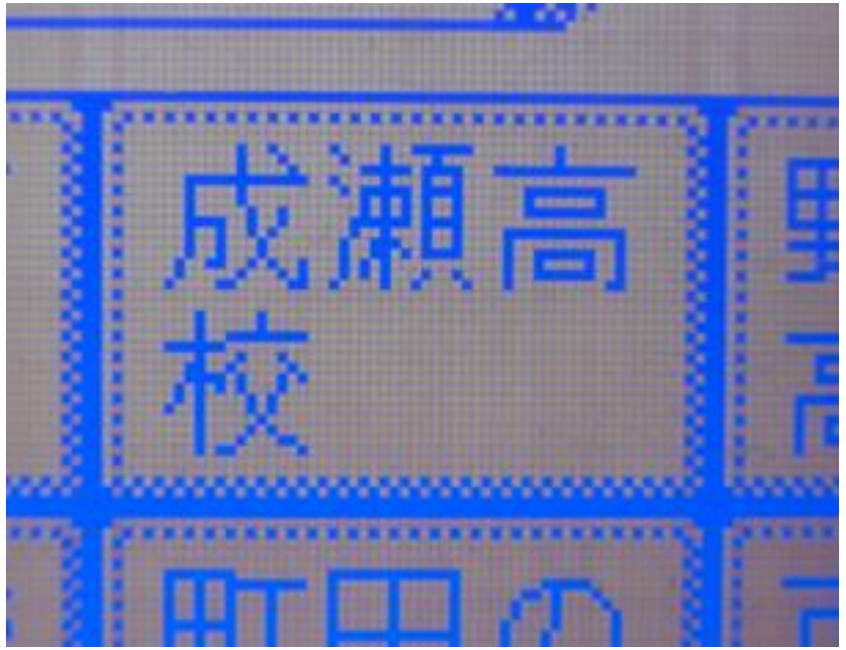
文字の形に関係なく、１つひとつの文字を同じ幅にしたもの。（→ 日本の文化）

例）ＭＳゴシック abcdefghijklmnopqrstuvwxyz

☆プロポーショナルフォント利用時に行頭をあわせたい場合は、「タブ」や「インデント」といった機能を使うとよい。

文字の形を記録するしくみ

ビットマップフォント



アウトラインフォント

成瀬
高校

「点の集まり」として記録	複雑な関数などの計算式として記録
負荷が小さいため、機械への組み込みとして良く使われる	大きくしても再計算し滑らかに表示できる為、画面表示や印刷など広く使われている
大きくするとギザギザが目立つ	都度計算するので比較的負荷が高い （高性能化に伴い現在のPCではほぼ無視できる）

まとめ

- データを0と1の列に変換することを「エンコーディング」といい、逆に0と1の列を元のデータに変換することを「デコーディング」という。
- コンピュータでは文字1つ1つに背番号のような「文字コード」をつけて処理しており、これらの対応関係を文字コード体系という。
- 英数記号カナ文字を1バイトで割り振った「半角」と、漢字などを含めた文字を2バイトで割り振った「全角」がある。
- 文字コード体系には、歴史的・国際的な理由からたくさんの種類があり、送信側と受信側が合っていないと「文字化け」がおこることがある。