



# ビットと符号化

*(C)TERAKOYA 2014*

# ビットとバイト [1]

## [1] ビットとバイト

《あるか、ないか》《YES か、NO か》《スイッチが ON か、OFF か》  
のような<sup>データ</sup>情報は、すべて《0 か、1 か》という 2 通りの数に対応させる  
ことができます。コンピューターは、この《0 か、1 か》の操作を、1 秒  
間に数十億回繰り返すことで、高度な作業を行っているのです。

《0 か、1 か》の操作 1 回分のことを、**1 ビット** [bit: binary digit] とい  
います。001101 は 6 ビットの情報を、10101100 は 8 ビットの情報を表  
しています。0 と 1 だけで記す数のことを **2 進数** と呼びます。

コンピューター内では、処理はすべてビット単位で行われます。数や  
文字のデータ、画像や音声のデータ、さらに動画のデータなど、すべて  
コンピューターで扱うデータは、ビット化 [2 進数化] されて解釈・実行  
されるわけです。

[問題 1] 1 ビットは、0 か 1 で、2 通りの情報が表せます。2 ビットは、  
00 か 01 か 10 か 11 で、4 通りの情報が表せます。  
それでは、3 ビットは何通りの情報が表せるでしょうか。

000 001 . . . . .

実際に使うデータの単位としてビットでは小さすぎるので、8 ビットを  
ひとまとまりとしてデータを扱います。この 8 ビットのことを **1 バイト**  
[Byte] といいます。

1 バイト = 8 ビット

[問題 2] 1 ビットは、0 か 1 で、2 通りの情報が表せます。  
では、1 バイトは何通りの情報が表せるでしょうか。

## ビットとバイト [2]

### [2] 数とビット

ふだん 私たちが 使っている数 [10 進数] とビット [2 進数] とは、次のように対応しています。

[10 進数]	[2 進数]
0 .....	0
1 .....	1
2 .....	10
3 .....	11
4 .....	100
5 .....	101
6 .....	110
7 .....	111
8 .....	1000
9 .....	1001
10 .....	1010
11 .....	1011
12 .....	1100

[問題 3] 10 進数で 21 は、2 進数でどのように表せますか。

<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/>	<input type="checkbox"/>	

[問題 4] 10 進数で 129 は、2 進数でどのように表せますか。

## 文字コード [1]

### [3] 文字とビット

コンピュータでは、データ処理はすべて **ビット** [0 か 1 かという情報] を単位にして行われます。数や文字のデータ、画像や音声のデータ、動画のデータなど、コンピュータで扱うデータは、すべて **ビット化** されて処理されます。

ここでは、文字をどのようにビット化しているのかを見てみましょう。次の表は、<sup>アスキー</sup> **ASCII** の **文字コード** [文字や記号を数値化したもの] です。<sup>アスキー</sup> **ASCII** は、アルファベットを表現するための **7 ビット** の文字コードです。

	000	001	010	011	100	101	110	111
0000	NUL	DLE		<b>0</b>	@	<b>P</b>	`	<b>p</b>
0001	SOH	DC <sub>1</sub>	!	<b>1</b>	<b>A</b>	<b>Q</b>	<b>a</b>	<b>q</b>
0010	STX	DC <sub>2</sub>	"	<b>2</b>	<b>B</b>	<b>R</b>	<b>b</b>	<b>r</b>
0011	ETX	DC <sub>3</sub>	#	<b>3</b>	<b>C</b>	<b>S</b>	<b>c</b>	<b>s</b>
0100	EOT	DC <sub>4</sub>	\$	<b>4</b>	<b>D</b>	<b>T</b>	<b>d</b>	<b>t</b>
0101	ENQ	NAK	%	<b>5</b>	<b>E</b>	<b>U</b>	<b>e</b>	<b>u</b>
0110	ACK	SYN	&	<b>6</b>	<b>F</b>	<b>V</b>	<b>f</b>	<b>v</b>
0111	BEL	ETB	'	<b>7</b>	<b>G</b>	<b>W</b>	<b>g</b>	<b>w</b>
1000	BS	CAN	(	<b>8</b>	<b>H</b>	<b>X</b>	<b>h</b>	<b>x</b>
1001	HT	EM	)	<b>9</b>	<b>I</b>	<b>Y</b>	<b>i</b>	<b>y</b>
1010	LF	SUB	*	:	<b>J</b>	<b>Z</b>	<b>j</b>	<b>z</b>
1011	VT	ESC	+	;	<b>K</b>	[	<b>k</b>	{
1100	FF	FS	,	<	<b>L</b>	¥	<b>l</b>	
1101	CR	GS	-	=	<b>M</b>	]	<b>m</b>	}
1110	SO	RS	.	>	<b>N</b>	^	<b>n</b>	~
1111	SI	US	/	?	<b>O</b>	_	<b>o</b>	DEL

[問題 5] **M** の文字コードは **1001101** です。

- ① **S** の文字コードは いくつですか。
- ② **x** の文字コードは いくつですか。
- ③ 文字コード **1101110** の文字は何ですか。

## 文字コード [2]

### [4] 文字コード

《本来は数値でない情報》を数値に置き換えたものを<sup>コード</sup>符号 [code] と呼びます。そして、どのような数値に置き換えるかを定める作業のことを<sup>ふごうか</sup>符号化といいます。

コンピューターでは、文字を扱うため《文字ひとつひとつに》固有の番号（コード）を付けるという方法を採用しています。これを**文字コード**と呼びます。

コンピューターの普及にともない、統一的な文字コードが必要になり、1963年に<sup>アスキー</sup>ASCII [American Standard Code for Information Interchange] が制定されました。

[問題 6] 1 バイト (8 ビット) では、 $2^8 = 256$  種類の文字を表せます。  
では、2 バイトでは、何種類の文字が表せるでしょう。

[問題 7] <sup>アスキー</sup>ASCII でみたように、アルファベット 1 文字は 1 バイト以内で表せます。このことを「アルファベットは 1 バイト文字である」といいます。では、日本語で使われる文字は何バイト文字といえるでしょうか。

文字コードの種類の中で漢字を表せるものは、おもに次の 4 つです。

- <sup>ジス</sup>JISコード ..... JIS<sup>\*</sup>で考案され、数度改訂されている
- <sup>シフト ジス</sup>Shift-JISコード ..... 日本の多くの PC で使用されている
- <sup>ユニコード</sup>Unicode ..... 世界中の文字を統一的に扱える
- <sup>イーユーシー</sup>EUC ..... 主に UNIX で使用されている

<sup>\*</sup>JIS → 日本工業規格 (Japan Industrial Standards)

## 記憶容量の単位 [1]

### [5] 単位の接頭語

《100,000,000 円》と書くより《1 億円》と書く方が分かりやすいですね。  
コンピューターの世界でも、 $M$ 、 $G$  といったひとまとまりでの表し方がよく使われます。これらの記号を単位の接頭語と言います。

数	指数	記号	読み方	日本語
1,000,000,000,000,000	$10^{15}$	P	ペタ	千兆
1,000,000,000,000	$10^{12}$	T	テラ	一兆
1,000,000,000	$10^9$	G	ギガ	十億
1,000,000	$10^6$	M	メガ	百万
1,000	$10^3$	k	キロ	千
1,00	$10^2$	h	ヘクト	百
10	10	da	デカ	十
1	$10^0$			一
0.1	$10^{-1}$	d	デシ	十分の一
0.01	$10^{-2}$	c	センチ	百分の一
0.001	$10^{-3}$	m	ミリ	千分の一
0.000001	$10^{-6}$	$\mu$	マイクロ	百万分の一
0.000000001	$10^{-9}$	n	ナノ	十億分の一
0.000000000001	$10^{-12}$	p	ピコ	一兆分の一

[元の単位] → [k キロ] → [M メガ] → [G ギガ]  
1,000 倍      1,000 倍      1,000 倍

記憶装置の容量の場合は、 $k=1,000$ 、 $M=1,000,000$ 、 $G=1,000,000,000$ ではなく、次の数値が使われることがあります。

- 1kB = 1024B [2<sup>10</sup>=1024]
- 1MB = 1024kB = 1024×1024B = 1,048,576B
- 1GB = 1024MB = 1024×1024×1024B = 1,073,741,824B

## 記憶容量の単位 [2]

[問題 8] 次の計算問題を解きましょう。

1GB = 1,000MB 1MB = 1,000kB 1kB = 1,000B とします。

- ① ワード文書は、1 枚の CD-R に何ページ保存できるでしょう。
  - ・ワード文書 1 ページの容量を 200kB とします。
  - ・CD-R の容量を 700MB とします。
  
- ② デジカメ写真は、USB メモリに何枚保存できるでしょう。
  - ・USB メモリの容量を 4GB とします。
  - ・デジカメ写真 1 枚の容量を 1.8MB とします。
  
- ③ 新聞の情報を CD-R に保存する場合、  
1 枚の CD-R [700MB] には 新聞何日分が記録できますか。
  - ・新聞 1 日分は朝刊と夕刊の合計で 44 ページとします。
  - ・新聞 1 ページの容量は (圧縮して) 40kB とします。

## 画像の符号化 [1]

### [6] 色コード

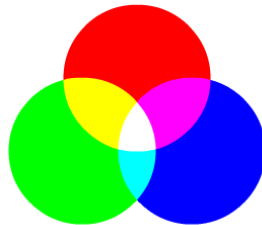
《文字》と同様、《色》も符号化できます。ここでは、赤 [Red] 緑 [Green] 青 [Blue] を 3 原色とする RGB コードについて説明します。R, G, B の濃度をそれぞれ 8 ビットで表した 24 ビットのコードです。00000000 なら色成分なし、11111111 ならその成分が最大になります。

・ 00000000 11111111 00000000 は、明るい緑を表します。

R            G            B

・ 11111111 11111111 00000000 は、明るい黄色を表します。

R            G            B



[問題 9] 次の RGB コードは何色なを表していますか。

① 00000000 00000000 00000000

② 11111111 11111111 11111111

③ 00000000 11111111 11111111

③ 00100000 00100000 00100000

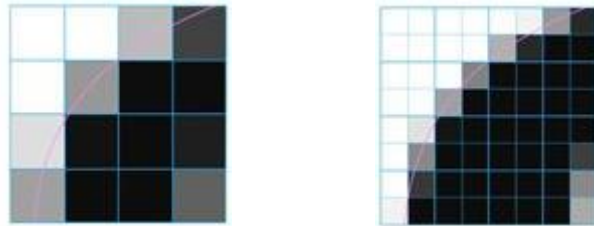
[問題 10] RGB コードの値を変更することで、多彩な色を表現することができます。では、何種類の色を表示することができるのでしょうか。



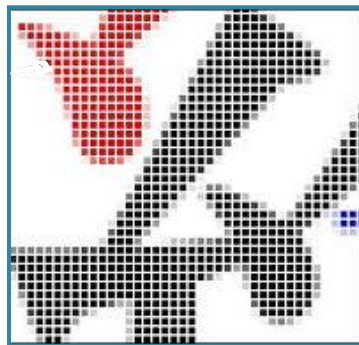
## 画像の符号化 [2]

### [7] ドットとピクセル

ディスプレイやプリンタでは、文字や画像を点 [dot] の集合として表します。同じ表示領域で比べると、ドット数が多いほど細やかできれいな表示ができます。ディスプレイでは、横方向×縦方向のドット数できめ細やかさを表します。



ドットに色の情報を加えたものを画素 [pixel、単位 : px] といいます。1 つ 1 つのピクセルが、24 ビットの色の情報を持つものを True Color といいます。



[問題 11] 700MB の CD-R に「横 1024×縦 768px の True Color 画像」を何枚保存できますか。

[問題 12] 4.7GB の DVD-R に、「640×480px の TV 動画」を約何分保存できますか。 ※TV 動画は 1 秒間に 30 枚の画像を表示します。