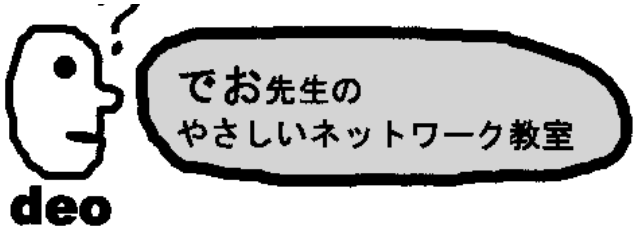
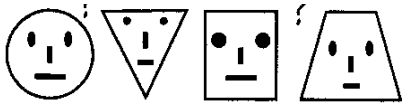




第3回　二進数で動くコンピュータ

コンピュータの原理も知っておこう

前回までに、コンピュータネットワークの原理についてご説明しました。今回は、そもそもコンピュータはどのような原理で動くのかというところまでさらに立ち返ってみましょう。

昨今のコンピュータ（特にパソコン）は、専門外の人にもかなり使いやすくなってきています。しかし、専門外の人といえども、コンピュータの原理を知っておくのは良いことです。それを知っているかどうかで、いざトラブルに陥った時に危機を脱することができるかどうかが違ってきます。

十進数	二進数
0	0
1	1
2	10
3	11
4	100
8	1000
16	1 0000
32	10 0000
64	100 0000
128	1000 0000
2,048	1000 0000 0000
32,768	1000 0000 0000 0000

図1　二進数

二進数

我々が数を表現するのに日常使うのは十進数です。これはもちろん、人の指が両手で10本であることに由来します。

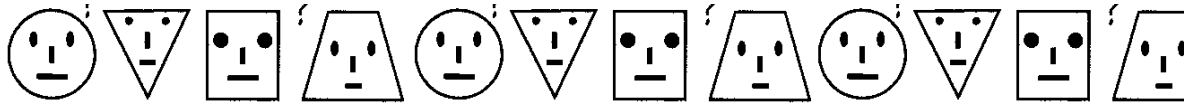
コンピュータは、内部で取り扱う数を表現するのに二進数を使います（図1）。二進数は、0と1だけの2個の数字を用います。各桁は、1からさらに1増えると繰り上がりを起こします。

二進数を使えば、両手の指（二進10桁）で1023まで数えることができます。桁数が増えるにしたがって、表せる数は驚くほどの大きさになります。32桁だと、1秒に1回ずつ数えて数え切るのに約136年、64桁だと約5845億年かかる数になります。

二進数は、電気で計算や電信を行うのに都合の良い形式です。スイッチ（コンピュータでは半導体素子がスイッチの役割を果たします）が通じているか切れているかという2種類の状態で一桁を表すことができるからです。

コンピュータは、文字、数値、表示画面上の各画素の明るさや色、さらには音声など、あらゆるデータを二進数で符号化して取り扱います。

ビットとバイト



二進数の１桁をビット（bit）といいます。bitは、binary digit（二進の桁）を縮めた造語です。

英数字１文字を表すのに必要なビット数のまとまりの単位をバイト（byte）といいます（byteという語にはまともな由来はありません。おそらくbitに似た語として恣意的に作られたものでしょう）。これは、記憶容量の単位として使われます。今のコンピュータは８ビットを１バイトとして取り扱っていますが、１バイトは8ビットと決められているわけではありません。昔は６ビットを１バイトとするコンピュータがありました。今の国際標準の文字符号は７ビットで定義されています。８ビットの組みであることを明示するには、オクテット（octet：８つの組みの意味）という別の用語があります。これはデータ通信の分野でよく使われます。

ブール代数

二進数で動くコンピュータの動作原理の基礎となるのがブール代数という数学です。これは、真（true）か偽（false）かという二値論理を扱うもので、真と偽それぞれを二進数１と０で表します。

ブール代数には次のような演算があります（図２）。

●論理積（logical AND）

二つの入力がともに真であるとき、出力が真になります。「かつ」で表される論理演算です。

●論理和（logical OR）

二つの入力のうち少なくとも一つが真であるとき、出力が真になります。「または」で表される論理演算です。

論理積

入力	出力
0 0	0
0 1	0
1 0	0
1 1	1

論理和

入力	出力
0 0	0
0 1	1
1 0	1
1 1	1

排他的論理和

入力	出力
0 0	0
0 1	1
1 0	1
1 1	0

論理否定

入力	出力
0	1
1	0

図２　ブール代数の演算

●排他的論理和（exclusive OR）

二つの入力のうちいずれか一方のみが真であるとき、出力が真になります。入力がともに真であれば出力が偽になることだけが論理和とは異なります。

●論理否定（logical NOT）

入力を反転した値が出力になります。「…ではない」で表される論理演算です。

ブール演算はほかにもありますが、コンピュータの仕組みを理解するにはこれだけ知っていれば十分です。二進数のあらゆる計算（四則演算など）は、これらのブール演算の組み合わせで実現できます。

世の中には、真か偽かで割り切れない事柄もたくさんあります。ブール代数は、そのようなものには適用できません。しかし、この単純な二値論理の体系は、様々な計算を機械化するには好都合なのです。

次回は、二進数の取り扱いを電気的に行うための原理についてご説明します。