

第9章

基礎理論

この章からいよいよテクノロジー系分野の内容に入ります。本章では、基数や集合の基本的な考え方を理解します。具体的には、コンピュータで扱う数値やデータに関する基礎的な理論や2進数に関する表現や演算、集合と論理演算の基本的な考え方について学んでいきます。

1 離散数学

(1) 数と表現

① 10進法と10進数

10進法とは、0～9の10種類の数字で表現され、1桁に10個の数字が集まるときに桁が1つ上がる表記法です。10進数は、10進法に基づいて具体的に表された数値のことです。10進法は我々の生活の中でも馴染みのある数の表し方ですが、コンピュータの世界では10進法以外の表記法も利用されます。10進数では、「10」を桁上がりの基準とし、値が「10」になるときに桁が1つ上がります。

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13

↑ 桁上がり

② 2進法と2進数

2進法とは、数を「0」と「1」の2種類の数字で表現する数の表し方です。2進法では「2」を桁上がりの基準とし、値が「2」になるときに桁が1つ上がります。例えば、0, 1, 10と表すように、「2」ではなく「10」になります。2進数は、2進法に基づいて具体的に表された数値のことです。2進数では「2」のときに桁上がりし、10（読み方：イチゼロ）となりますので覚えておきましょう。

0, 1, 10, 11, 100, 101, 110, 111, 1000, 1001, 1010

↑ 桁上がり ↑ 桁上がり

↑ 桁上がり

コンピュータは、人間と違って「0」と「1」だけで数表現します。

その理由は、コンピュータの基本的な構造と動作原理に関連しています。コンピュータは電気信号で動作し、これらの信号はオン（1）またはオフ（0）の状態を取ります。このような特性から、コンピュータ内部でデータを効率的に処理し、論理演算を行うのに適している2進数が用いられます。

③ 16進数

16進数とは、0～9の10種類の数値とA～Fのアルファベットを用いて表現され、10進数の「16」のときに、桁が1つ繰り上がる数です。

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, 10

↑ 桁上がり

(2) 基数

基数とは、桁上がりの基準となる数値のことで、10進数であれば、基数は10で、2進数であれば基数は2になります。

① 基数の表記について

ある数があったときに、その数が2進数なのか10進数なのか見分けが付きません。そのため、基数の表記として、以下のようにかっこを書いて、そのかっこの右下に添字を記載し進数を示します。

$(10)_2$: 2進数での「10」

$(10)_{10}$: 10進数での「10」

$(10)_{16}$: 16進数での「10」

同様に、他の数値にもこの表記を用いることができます。例えば、2進数での「1011」は $(1011)_2$ 、16進数での「1A」は $(1A)_{16}$ のように表されます。

② 基数変換

基数変換とは、ある進数で表された数を別の進数で表し直すことです。例えば、2進数で表現されている数値を10進数や16進数などに変換することを指します。いくつかの基数変換について解説しますが、まずは2進数から10進数へ基数変換する方法を見ていきましょう。

■ 数値の対応表

2進数	10進数	16進数
0	0	0
1	1	1
10	2	2
11	3	3
100	4	4
101	5	5
110	6	6
111	7	7
1000	8	8
1001	9	9
1010	10	A
1011	11	B
1100	12	C
1101	13	D
1110	14	E
1111	15	F
10000	16	10

■ 2進数から10進数への基数変換の計算方法

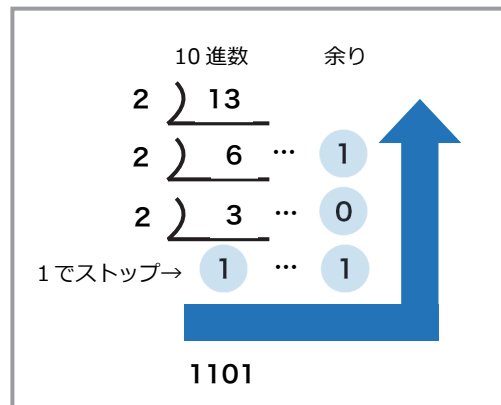
まずは2進数を並べます。次に各行の数字 $\times 2^n$ を計算します。ここで n は桁数 -1 になります。その後、掛け算した各行の数を足し合わせます。実際に2進数「1101」を例に、10進数に変換してみましょう。ここで 2^0 は1です。2に限らず N^0 (N の0乗)は常に1になります。

$$\begin{array}{cccc}
 1 & 1 & 0 & 1 \\
 \times & \times & \times & \times \\
 2^3 & 2^2 & 2^1 & 2^0 \\
 8 & + & 4 & + & 0 & + & 1 & = & 13
 \end{array}$$

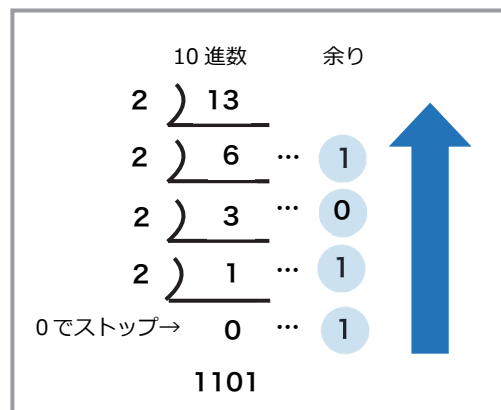
上記の計算により、2進数「1101」は10進数で「13」になることがわかります。

■ 10進数から2進数への基数変換の計算方法

・10進数から2進数に変換するには、10進数の数を2で割り算し、商が0または1になるまで計算します。商が1の場合は右図のように考えます。



・商が0になるまで割る場合は右図のように考えます。余りを下から上に並べると、先ほど同様に「1101」になります。



上記の計算によって、10進数「13」は2進数では「1101」であることがわかります。

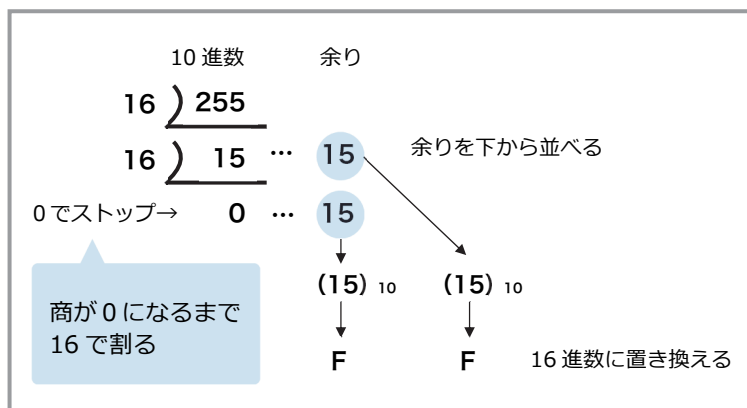
■ 10 進数から 16 進数への基数変換の計算方法

基本的な考え方は 10 進数から 2 進数への変換と同じです。

まず 255 を 16 で割ります : $255 \div 16 = 15 \dots \text{余り } 15$

次に, 15 を 16 で割ります : $15 \div 16 = 0 \dots \text{余り } 15$

変換の流れは, 余りを下から上へと読みます。余りが 10 以上の場合, 対応するアルファベット (A = 10, B = 11, C = 12, D = 13, E = 14, F = 15) を使用します。この場合の余りは「15, 15」となります。したがって, 10 進数の「255」は 16 進数では「FF」となります。



■ 16 進数から 10 進数への基数変換の計算方法

基本的な考え方は 2 進数から 10 進数への変換と同じです。

最初に各行の数 $\times 16^n$ を行います。n は桁数 - 1 です。

ここでポイントですが, 16 進数のアルファベットを 10 進数に変換してから計算する必要があります。16 進数の「1A3」を 10 進数に変換する方法を例に解説します。

16 進数「1A3」の各桁は右から順に, 16 の 0 乗, 16 の 1 乗, 16 の 2 乗を掛ける必要があります。そして, 16 進数の「A」は 10 進数で「10」に相当します。それ以外はそのまます。

右の桁 (3) : $3 \times 16^0 = 3 \times 1 = 3$
 次の桁 (A, 10 進数で 10) : $10 \times 16^1 = 10 \times 16 = 160$
 左の桁 (1) : $1 \times 16^2 = 1 \times 256 = 256$
 これらをすべて合計すると : $3 + 160 + 256 = 419$

したがって, 16 進数の「1A3」は 10 進数で「419」となります。

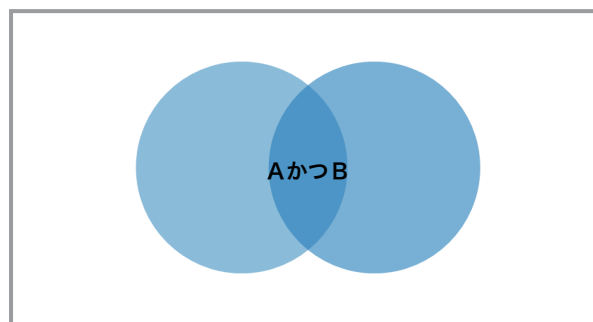
16進数を並べる	1	A	3	
	↓	↓	↓	
	(1) ₁₀	(10) ₁₀	(3) ₁₀	各行の16進数を 10進数に置き換える
	×	×	×	
各行に16進数の 重みを掛け合わせる	16 ²	16 ¹	16 ⁰	
	256	+	160	
		+	3	
	= 419			

(3) 集合と命題

集合とは、区別できる複数の要素を1つにまとめたものです。例えば、「赤、青、黄」の色を集合として「色の集合」と表すことができます。命題とは、真か偽かを明確に判定できる文や式のことです。例えば、「このカードは青色である」という命題は、そのカードの色を見ることで真偽を判断できるため命題です。この命題の場合、そのカードを見ることで色が青色かどうかを確認できます。もしカードが青色であれば命題は真（正しい、True）、青色でなければ偽（正しくない、False）となります。命題の特徴は、具体的な事実に基づき、その真偽が確認できる点にあります。

(4) ベン図

ベン図とは、複数の集合の関係を図で表したものです。これにより、異なる集合の共通点や相違点が一目でわかるようになります。



複数の集合がある場合、その関係を文章で表したい場合は、キャップ（ $\cap$ ）やカップ（ $\cup$ ）などの記号を用います。

キャップ（ $\cap$ ）は、2つの集合に共通する要素のみを含む集合を指します。集合Aと集合Bのキャップ集合は、AとBの両方に属する要素のみから成ります。

カップ（ $\cup$ ）は、2つの集合のいずれかに属するすべての要素を含む集合です。集合Aと集合Bのカップ集合は、AまたはBのどちらか一方にでも属するすべての要素を含みます。

(5) 論理演算

論理演算とは、演算の一種で、真理値（真または偽）を扱います。これらは条件判断やデータの処理に広く使用されます。

① 論理演算の種類

論理演算の種類にはAND、OR、NOT、XOR（排他的論理和）などがあります。ANDは両方が真の場合に真を返し、ORはいずれかが真の場合に真を返します。NOTは真偽を反転し、XORはどちらか一方のみが真、つまり互いが一致しないの場合に真を返します。

■ 基本的な論理演算の種類

演算子	説明
AND	・論理積とも呼ばれる ・AとBの両方が真のときに、結果が真（1）になる
OR	・論理和とも呼ばれる ・AとBの少なくとも一方が、つまりどちらが真（1）の場合に、結果が真（1）になる
XOR	・排他的論理和とも呼ばれる ・AとBが異なる場合、つまりどちらも一致しない場合は、真（1）になり、それ以外は偽（0）となる
NOT	・論理否定とも呼ばれる ・入力反対が結果になる 例えば、Aが真（1）のときは偽で（0）、偽（0）のときは真（1）となる

② 真理値表

真理値表とは、論理演算における入力値とその出力結果を一覧表にまとめたものです。真理値表を確認することで、各入力の組み合わせに対して出力がどのようなになるかを理解できます。以下は代表的な論理演算の真理値表です。

■ AND 演算の真理値表

A	B	A AND B
1	1	1（真）
0	1	0（偽）
1	0	0（偽）
0	0	0（偽）

■ OR 演算の真理値表

A	B	A OR B
1	1	1 (真)
0	1	1 (真)
1	0	1 (真)
0	0	0 (偽)

■ XOR 演算の真理値表

A	B	A XOR B
1	1	0 (偽)
0	1	1 (真)
1	0	1 (真)
0	0	0 (偽)

■ NOT 演算の真理値表

A	NOT A
1	0 (偽)
0	1 (真)

③ その他の論理演算

IT パスポート試験では、先ほど解説した AND, OR, XOR, NOT 以外の論理演算として、NAND (否定論理積) と NOR (否定論理和) も出題されます。NAND (ナンド: NOT AND) とは、論理積と否定を組み合わせた論理演算です。NAND は、AND 演算の答えをすべて反転した数値になります。NOR (ノア: NOT OR) とは、論理和と否定を組み合わせた論理演算です。NOR は、OR 演算の答えをすべて反転した数値となります。

■ NAND 演算の真理値表

A	B	A NAND B
1	1	0 (偽)
0	1	1 (真)
1	0	1 (真)
0	0	1 (真)

■ NOR 演算の真理値表

A	B	A NOR B
1	1	0 (偽)
0	1	0 (偽)
1	0	0 (偽)
0	0	1 (真)