第 9 章　基礎理論

この章からいよいよテクノロジ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（10）

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（10）　↑桁上がり（100）　↑桁上がり（1000）

コンピュータは，人間と違って「０」と「１」だけで数を表現します。

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

⑵　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× 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。

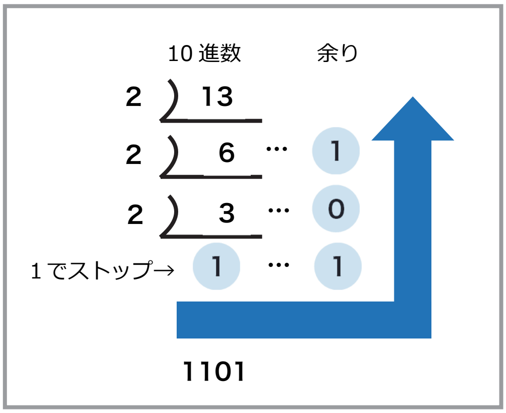
1	1	0	1
×	×	×	×
2^3	2^2	2^1	2^0

$8 + 4 + 0 + 1 = 13$

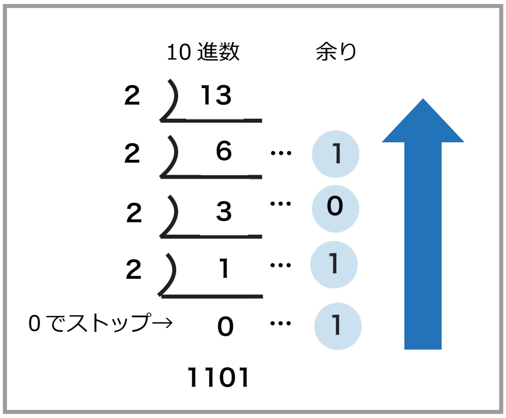
上記の計算により，2 進数「1101」は 10 進数で「13」になることがわかります。

■ 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。



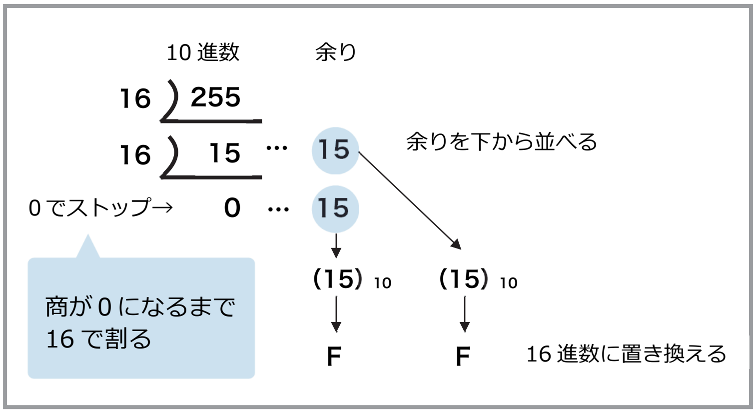
上記の計算によって，10 進数「13」は 2 進数では「1101」であることがわかります。

■ 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 ÷ 16 ＝ 15 ... 余り15
次に，15を16で割ります	：15 ÷ 16 ＝ 0 ... 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以上の場合は，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 ×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です。

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

各行の 16 進数を 10 進数に置き換える　$(1)_{10}$　$(10)_{10}$　$(3)_{10}$

各行に 16 進数の重みを掛け合わせる　$\times 16^2$　$\times 16^1$　$\times 16^0$

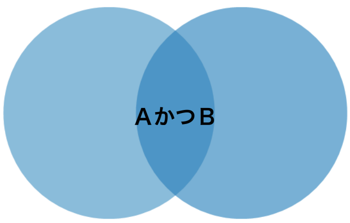
$$256 + 160 + 3$$
$$= 419$$

⑶　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。

⑷　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（∩）やカップ（∪）などの記号を用います。

キャップ（∩）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（∪）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。

⑸　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 ・入力の反対が結果になる 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）