

易しくない 算術のお話

2015.08 初版:2021.2 改訂

0. まえがき

この文書は、**数値計算法**の解説です。トピックとして、**補数**(compliment number)と**負数**(negative number)の概念を説明することを目的としてまとめたものです。小学校低学年の**算数**で**引き算**を習うとき、負数は習いません。補数は、小さな数の桁から大きな数を引きたいとき、前の桁位置から10を借りてくるときの算法に応用されています。小学生には難しい概念です。補数は、コンピュータを利用する計算技術を説明するとき、言わば突然出てきます。しかし、それまで、教わる機会が全くないのです。ところで、庶民が社会生活をする上で、最低限必要な実用技能の種類を言う慣用語に「**読み・書き・そろばん**(算盤)」があります。英語にも同じような言い方があって、Three R's で辞書に載っています。reading, writing, arithmetic の3語の組みを言い、子供に教育する基礎的な学科のことです。Rの音が3語にあることから、慣用されている言い方です。「読み」は、文字の並びをどう言うかの約束です。「書き」の技能は、**お習字**で習います。漢字を覚えさせる意義もあり、想像以上に教育効果があります。漢字を眼で見て読めても、書けないことも多いのです。算盤は、算術技能を覚えるときの比喩的な言い換えです。不思議なことに、「**話し**」が抜けていることです。現代は、電話、それもテレビ付き通信も実用できるようになりました。しかし、以前は、少し遠くに居る人と、顔を合わせて、相手との対話で情報を交換することが不便でした。文字を介して間接的に伝えなければならなかったのです。その「文字並び」は、声に出して読み上げるときの言い方と、かなりの違いがありました。**明治維新**以降、日本語は、英語を始めとした外国語の影響を大きく受けるようになりました。**言文一致**の運動は、今日の口語文を普及させる原動力になりました。敗戦(1945)後、声に出してコミュニケーションをする英会話、つまり、**話し**が重要な技能として認められるようになったことも理解しておく必要があります。この文書は、**算術**を表題に挙げましたが、数に関する雑学的な話題も紹介します。

この文書は、「**易しくない幾何のお話**」の姉妹編です。合わせてお読み下さい。なお、前のバージョンで説明した論理演算のできる関数電卓(図14)は、Windows7以降のパソコンではサポートされなくなりました。これは別途プログラミングをして提供しようと思っています。

島田 静雄

目 次

まえがき

0. はじめに

1. 数の読み書き

- 1.1 実用技術としての算術
- 1.2 数詞と助数詞で数える
- 1.3 単位と表記法
- 1.4 個数と順序数の区別
- 1.5 順序数を一意に決めないことがある
- 1.6 順序数に文字も加えるコード化
- 1.7 数字で書くか文字で書くか
- 1.8 十進数以外の数表記法
- 1.9 文字付きの数字と読み

2. 混乱を起こす数

- 2.1 言い方と書き順とに混乱がある場合
- 2.2 デジタル時計の数表示
- 2.3 満と「かぞえ」の使い方がある
- 2.4 電話機とパソコンのテンキー文字配列

3. 小学校の算数教育を再確認する

- 3.1 一桁の足し算の教育から始まる
- 3.2 補数の概念が使われている
- 3.3 負数は座標の考えを使う
- 3.4 掛け算の基礎に九九を覚える
- 3.5 掛け算は、九九と足し算を応用する
- 3.6 整数を扱う割り算は二種類ある
- 3.7 引き算と割り算は交換律が成立しない

4. 数式の編集と読み書き

- 4.1 数式を文章に表す
- 4.2 数式の種類

4.3 数式はグラフィックスの性質がある

4.4 数式の組版は特殊な技能である

4.5 数式の書き換え

4.6 数式は計算の優先手順がある

4.7 大数と小数

5. 数の丸め

5.1 有効桁数

5.2 覚え易くする呼び数を使う

5.3 大小の寸法系列を合理的に決めたい

5.4 お金の計算で使う丸め

5.5 お金の小数単位にセントがよく使われる

5.6 設計計算書に見る数の丸め

6. 算術計算の道具

6.1 固定小数点数の計算専用機械

6.2 算盤

6.3 手回し計算機

6.4 電動計算機

6.5 電卓

6.6 普通の電卓の使い方

7. 論理演算の約束を覚える

7.1 数値計算に応用する2値論理学

7.2 論理演算の用語と記号に混乱がある

7.3 論理用語と論理記号

7.4 集合論の考え方も使う

7.5 論理演算の基礎知識

7.6 論理演算を幾何モデリングに応用

索引

英字		オブジェクト 4.3	算法 1.1	手回し計算機 6.4	文 4.1
AND 7.2		お習字 0.	算用数字 6.1	定位点 6.2	ベン図 7.4
BASIC 4.2			シフト 3.5	定義 4.5	編集記述言語
character 1.7	カ〜コ		シフトレジスタ 3.5	電子算盤 6.5	4.3
digit 1.7	かぞえ 2.3		四捨五入 5.1	電卓 6.2	補数 0.
divide by 3.6	可算名詞 1.2		四則演算 4.2	電動計算機 6.4	翻訳 4.2
divide into 3.6	仮数部 4.7		指数部 4.7	デジタル 1.2	
ID 1.6	関係演算子 4.2		自然数 1.2	デジタル時計 2.2	マ〜モ
FORTRAN 4.1	寒暖計 3.3		実数 4.7	ドリル学習 3.1	丸め 5.
HTML 4.3	含意 7.2		集合論 7.4	等式 4.1	満 2.3
IMP 7.3	数え唄 3.4		循環小数 4.7	同値 7.2	未満 4.2
INT 5.1	数え年 2.3		順序数 1.2	道具・技法・技能 1.1	無理数 4.7
Knuth 4.3	干支 1.8		除数 3.5		命題 7.1
LateX 4.3	キロ・メガ・テラ 1.7		助数詞 1.2	ナ〜ノ	明治維新 0.
letter 1.7			序数詞 1.4	内含 7.2	文字型 1.3
MOD 3.6	幾何モデリング 7.6		商 3.2	二値論理学 7.1	物 4.3
NOT 7.2	技術 1.1.		小数 4.7		ヤユヨ
OR 7.2	金銭登録機 6.1		乗数 3.6	ハ〜ホ	優先順位 4.6
symbol 1.7	切り捨て 5.1		剰余 3.6	パーセント 1.9	有効桁数 5.1
spell out 1.7	切り上げ 5.1		スラッシュ 1.3	バイト 1.8	有効数字 4.7
Tex 4.3	グロス 1.3		推論 7.1	パスワード 1.6	有理数 4.7
Three R's 0.	九九 3.4		数学 1.1	排他的選言 7.3	呼び数 5.3
XOR 7.2	組版 4.3		数学記号 4.4	排反 7.2	読み上げ算 1.2
	位(くらい) 1.2		数式 4.2	話す 0.	読み・書き・算盤 0.
ア〜オ	繰り上がり 3.1.		数値解析 1.1	ビット 1.8	
アイウエオ順 1.8	桁上がり 3.1.		数値型 1.3	引き算 0.	ラ〜ロ
アクセサリ 6.5	結論 7.1		数値計算法 1.1	否定 7.2	量 1.4
アナログ 1.4	言語学 7.1		整数 1.2	被除数 6.3	レジスタ 3.5
アナログ時計 2.2	言文一致 0.		ゼロ 1.2	被乗数 3.5	レドシヨツセ 2.3
アラビア数字 1.2	ご破算 6.6		セント 5.5	非両立 7.2	連言 7.2
アンダーフロー 6.6	五十音順 1.8		設計計算書 5.6	百分率 1.9	連想記憶 3.4
余り 3.6	固定小数点数 4.7		宣言 4.5	標準数 5.3	ローマ数字 1.8
暗号化 1.6	交換法則 3.5		選言 7.2	表音文字 1.7	論理学 7.1
イロハ順 1.8	交換律 3.5		前提 7.1	評価 4.1	論理積 7.2
以下 4.2			算盤 6.2	ブール代数 7.1	論理和 7.2
以上 4.2	サ〜ソ	タ〜ト		負数 0.	ワ〜を
忌み数 1.5	座標 3.3	ダース 1.2		浮動小数点数 4.7	ワード 1.8
演算子 4.1	左辺値 4.1	大数 4.7		復唱 3.4	割り算 3.6
右辺値 4.1	算術 1.1	代数 1.2		物質名詞 1.4	を超え 4.2
オーバーフロー 6.6	算数 1.1	代入文 4.1		分数 1.3	
		テンキー 2.4			

1. 数の読み書き

1.1 実用技術としての算術

数学(mathematical science)は硬い用語です。数を扱う学問に当てる語です。自然科学(natural science)に含めます。その中身を区別する専門分野は、代数・幾何・微積分…などに分けて学習します。現実には、数値計算に応用することで、始めて身近の問題の解決に使うことができます。これが算術(arithmetic)、つまり計算の技術(technology)です。戦前、小学校の低学年の学科名に使われていた庶民的な言葉です。戦中・戦後から、算数と言い換えるようになりました。高校での学科名は数学です。数を直接扱う算術の意義が薄れ、抽象化した学問として学びます。多くの専門分野を持ちますので、学科名は、数学1、数学2、数学3に分けていて、数学1以外は選択科目です。数の代わりに文字や記号を使うことから、以前は、代数と言いました。算術は、計算を実行する技術ですので、三つの構成要素で考えます。道具・技法・技能です。道具として、そろばん(算盤)を使う計算場面を考えると分かり易いでしょう。技法は、算法の言い方があります。技能は練習で覚えます。算術と言うと、初歩的な意味合いに取られることを嫌って、学術用語は、数値計算法、数値解析などと言います。大学での学科編成では、数学科は理学部に属し、数学の応用を扱う応用数学科(applied mathematics)は、工学部に属する、などと仕分けをしています。

1.2 数詞と助数詞で数える

物(もの)を声に出して一つ二つと数えられるとき、自然数(natural number)で言います。正の整数(integer)です。近年は、デジタル値とも言います。我々が眼にしている日常的な物(もの:thing)は、形が同じで、一つ二つと数えられる物と、数え難い物とがあります。数えられる物の名前は、英語で言う普通名詞であって、文法では可算名詞です。日本語では、物の特徴を言い分ける助数詞を付けて「一個・二個…」、切り紙ならば「一枚・二枚…」、本ならば「一冊・二冊…」のように数え、また文字でも書きます。何も無いとき、ゼロを使う、例えば0個、0枚、0冊などとは言いません。商品などでは、無い・品切れ、などと言います。この表し方は、品詞の区別で言うと、数詞と助数詞の並びです。日本語の助数詞は種類が多く、500種類以上あるそうです。文字としての数字は、漢数字です。数の意義で使う数詞は、一から九までです。古典的なアラビア数字も1から9までであって、0がありませんでした。漢数字の十・百・千・万・億・兆・京などは、数単位を言う助数詞です。したがって「一十・一百・一千」と使うことがありますが、慣用では省きます。領収証などでは、書き換えができないように「一・二・三」に代えて「壹・貳・参」を当てることがあります。日常生活環境では気がつきませんが、数値を書くときも言うときも、零(れい)と負数(マイナス)を使いません。漢字の負に、数の意味はありません。ゼロ(zero)はアラビア語に原義のある言い方です。意味は「空っぽ」であって、数の意義はありませんでした。丸(○)は、ゼロを意味する漢数字ではなく、記号です。数学史で言う「0の発見」は、七世紀頃のインドである、とされています。日本語の環境では、「八百八町、百八つの鐘撞き」のような慣用語があります。位(くらい)取りに無い数を言いません。例えば、アラビア数字の表記で、0を含む103または2005を漢数字で書くと、百三または二千五で済ませます。算盤の読み上げ算では、「百、トンで三、二千トビトビの五」のように言うことで、0の桁が幾つ続くかを知らせます。

1.3 単位と表記法

フランス語が起源のダース(dozen)は、12個を単位とする助数詞として使います。さらに、12ダース(144)をグロスとします。12グロス(1728)は大グロスと言うことがあります。これより多くなるときの単位名はありません。大きな数を表すとき、昔は、6、12、16、36、60などを数の区切り単位に使うこともありました。現代でも慣用されている時間や日付の表示方法がそうです。長さでは、フィートとインチとを混ぜて使うとき、単位記号を添えて、例えば「5'3"」と表します。時間の場合にも、日本語では漢字を区切り符号(助数詞)にして「8時5分30秒」のような表記を使います。コンピュータの利用では、ファイルの更新日時をシステム側で決めています。スラッシュ(/)を区切り記号にした、例えば「2020/5/21 16:11:25」のようです。しかし、スラッシュは、数学では分数表示の略記ですので、日付の表記法は大小の数字順が逆であると誤解されます。表計算のソフトである MS-EXCEL では、日時の表示方法に種々選択ができます。このとき、一桁の月、また一桁の時間を書くとき、0を頭に加えるか省くかの選択を必要とすることがあります。これには、セルの書式設定を、数値型か文字型か、をユーザ側で決めます。時間を表示する「時・分・秒」は、「hh:mm:ss」の文字型を使いますので、一桁の数も頭に0を付けます。ところが、手紙など、日常文書で日付を書くとき、日本では年・月・日・曜日の順ですが、欧米では逆順です。また、国によって習慣が異なり、間違っ理解されることがあります。アメリカ式は「month、day、year」の順で並べ、イギリス式は「day、month、year」の順です。後者の方式が一般的です。例えば、簡単な表記で、「4, 5, 2020」とあると、4月5日なのか、5月4日であるかの区別ができません。これらの表記法は、個数と順序数との混用になっています。また、時間軸で見ると、昇順または逆順に揃いません。

1.4 個数と順序数の区別

整数は、「多い・少ない」の違いを説明する個数を意味する場合と、順番を決める場合との、二つの使い分けがあります。例えば、学校のクラス編成が3組あるというときの3は個数です。第3組と言うときの3は順序数 (ordinal number)です。物は人や動物も含めます。固有名詞は、個別に名前がある物です。小学校教育の場で、名簿を作成するとき、生徒に番号を割り振ります。これは順序数です。個人の名前は、漢字コードを使って並べ変えができますが、順序数を決めることは代名詞としての使い方です。小学校で教える算数は個数を扱い、順序数は使いません。日本語では、順番は「第一・第二…」のよう頭に第を付ける方法と、接尾辞に番、番目、位と付ける、また両方付けることもします。この語は、読みが同じで漢字違いの序数詞と言います。このときの数
が順序数です。英語では定冠詞 (the)をつけ、「first, second, third」と使い、4以上は「fourth, fifth, …」のように接尾辞 (th)を付けます。20 以上、例えば31番目は thirty first とし、一桁目の順序数の表し方を踏襲します。英語では、hundred, thousandなどは可算名詞です。2以上では複数形を使い、助数詞の使い方をしています。穀物のように、個数が多く、数え難くなって、水や油のような液体と同じような流動性を持つ物、または、大きさを決め難い土や鉄鉱石などの材料の名前は、英語では物質名詞とし、文法上は不可算名詞です。多い・少ないの認識は感覚的に理解し、量(りよう、かさ、quantity)と言い分けます。分量とも言います。個数と分量との全体を言う用語が数量です。不可算名詞の物は、多い・少ないの程度が曖昧になります。これは、アナログ量であるとも言います。英語の文脈では単数扱いをしますが、入れものを工夫して、それを単位として個数、つまり整数で言うことがあります。「a cup of water」が一例です。

1.5 順序数を一意に決めないことがある

或る範囲の長さや領域を区別したいとき、擬似的な順序数を使います。擬似的というのは、文字を含む場合、文字コード順で並べるときなどを言います。この文書の表題に使う番号は、一つの例です。そこで使う整数が一意に付けられていないことと、住所を言うとき、町名を使って「何々一丁目・二丁目…」のように順序数を当てます。より詳しい位置を言いたいときは、下位に番地番号などを使います。集合住宅では、建物番号、階番号などを形容詞的に加えて、順序数の室番号を付けます。鉄道駅のホームには、上り線・下り線などの線別を区別するとき、順序数で「1番線・2番線…」のように区別します。しかし、線数が1番線の反対側に増設されることもあって、0番線、00番線と付けることがありました。番地名に、例外的に0番地や無番地の言い方を使うことがあります。ただし、負の番号を当てることはありません。人名では「一郎・二郎・三郎…」のような順序数も見られますが、連続した兄弟の順番を区別する使い方ではないことも少なくありません。欧米の住居表示は、道路の左・右別で、偶数順・奇数順に振ります。ホテルで部屋番号を割り振るとき、4や13を忌み数として、その番号を飛ばします。実生活の場では、数学的な正確性にはこだわらない使い方もあります。

1.6 順序数に文字も加えるコード化

順序数は、固有名詞の代わり、つまり、代名詞の使い方とき、コード(code)とも言います。代表的な順序数には、郵便番号、電話番号、自治体名を付けた自動車登録番号もそうです。順序数として発行されますが、郵便番号以外は、多少の制限の下で、利用者側で番号を選択できることもあります。これらを識別用文字並びとして使うときは記号番号(ID: identification number)とも言います。コンピュータを利用するときのファイル名は、名前の文字列の後ろに点(.)を付けて、拡張子(extension)名を繋ぎ、ファイルの種別が判るようにします。普通は英3字です。工業標準化(JIS)では、種々のコード系が規格化されていて、工業以外にも利用し、さらに世界共通に使えるような国際化が合意されているものもあります。これらのコードは、字数を揃え、分かり易さも考えられています。文字並びは、英字・記号・漢字などを含めることもできます。したがって、並べ変え、検索に使うことができます。個人の名前と、代名詞的な使い方をする順序数との対の代表的な例には、銀行の口座番号があります。銀行名・支店名も含めて幾つかの順序数の組で表しています。個数に上限があるとき、数字並びの桁数を決めます。欠番もあり、また、頭に0が並んだ数でも0を省くことをしません。クレジットカードには、4桁の数字並びごとに空白(スペース)を挟む数字並びで表示をしますが、管理上は一続きです。表計算ソフトのMS-EXCELは、セルの書式が何種類かあって、数値表示と文字表示とを区別できます。個人名が漢字表記であっても、漢字コードの大小で判断できます。名前と順序数との対は個人を特定する情報ですので、全体を暗号化(cypher)する工夫が必要になりました。利用者は、個人ごとに追加のパスワードを決めて、第三者の侵入を防ぐ方法を使いますが、それが破られる被害が多く知られるようになりました。順序数は、その番号が本人のものと証明できるように、写真付きのカードを発行することがあります。公的機関が発行したものは、身分証明書としての使い方ができます。免許証、総務省主導の個人カード(通称で言うマイナンバーカード)、外務省発行の旅券もそうです。一種類の公的カードに統一したいとの役所側の希望があります。

1.7 数字で書くか文字で書くか

日本語では、「漢字・かな・数字」、この全体をまとめて**文字**と言います。江戸時代までは、数字は漢数字「一、二、三…」だけでした。明治以降、欧米文化から輸入したアラビア数字「1, 2, 3…」も使うようになりました。数の大小を区別したいときに半角のアラビア数字を使います。文章の中では全角の漢数字を使う習慣ですが、文字種として全角のアラビア数字を使うこともあって、いささか複雑です。欧米語は、書き言葉に**表音文字**のアルファベット(ラテン文字)を使います。ラテン文字を **letter**, アラビア数字は **digit**、コンマなどは記号 **symbol** と区別し、その全体を **character** とくります。英語の環境から見たとき、日本語のカナは表音文字とされ、Kana Letter と説明し、漢字は、Kanji Character と区別します。漢数字のような使い方に **spell out** があります。例えば、数字の12と書く代わりに twelve と書くことです。個数をアラビア数字並びで書くことは、客観的に数の大小を区別できます。日本語の文字表記は縦書きが標準です。右横書きは、一列一文字の縦書き表記法です。漢数字は縦書きにも横書きにも対応できます。敗戦前までは、書画の表題、絵葉書の説明などに、右からの横書きが使われていました。この表記方法は、アラビア数字や、英語を混ぜて使うときに混乱します。現代の数の表記法は左からの横書きが普通です。見易さ、また読みを助けるため、小数点(ピリオド)以外に、コンマなどの区切り記号を挟むことがあります。欧米では3桁ごと、日本語では4桁ごとです。桁数の多い数は、日本では1から9999までを循環的に言い、これより大きい数を4桁ごとに区切り、万・兆・京…の単位名を助数詞として使います。欧米では、1から999までの3桁ごとに区切ります。単位名は、可算名詞の、千(**thousand**)、百万(**million**)、十億(**billion**)、…です。ギリシャ語原義の**キロ(kilo)**、**メガ(mega)**、**テラ(tera)**、…は、単位記号文字に K、M、G、…を使っています。こちらの品詞は形容詞です。なお、小数点記号には、ピリオドではなく、**コンマ**を使う国もあります。これは、**ピリオド**を full stop とも言いますので、文の切れ目を表示する、と誤解されることを避ける意義を持っています。

1.8 十進数以外の数表記法

数字に代えて文字を使う古典的な方法の例を幾つかを挙げます。第一は**ローマ数字**です。時計の文字盤表記に見られます。最大で 4000 までしか表すことができません。第二は、**干支**(かんし: 十干十二支: えと)です。十二支(**子丑寅卯辰巳午未申酉戌亥**)は、古くから使われていて、1から12までの数に当てます。方位、時刻などの表し方の熟語として残っています。子午線・戌亥(乾)門・丑三つ時、などがあります。第三は、十干(**甲乙丙丁戊己庚辛壬癸**)です。第四は、**イロハ順**です。「いの一 番」の慣用句があります。大正時代まで、名簿はイロハ順が普通にみられました。明治以降、**アイウエオ順**(または**五十音順**)が標準になりました。コンピュータで使う漢字コードを決めるときの参考にされました。明治時代の日本語辞書(例えば言海)は五十音順で編集されていましたが、イロハ順の索引も付いていました。

コンピュータが内部的に扱う数は、複数本の信号線をセットにして、1本ごとに電気的信号の on/off で送受信します。これを(1,0)で数値化して**ビット**と言い、**2進数**です。ビットを並べて**バイト**、さらに**ワード**も以前は数単位名としました。1バイトの構成は4ビットが標準ですが、3ビット並びも使われています。1ワードは、 $2 \times 3 = 6$ ビット、または $2 \times 4 = 8$ ビットを当てることもしましたが、メーカーによって違いがありましたので一般に利用できる単位化には進みませんでした。2進数は、0と1だけの数字並びです。文字数が多くなりますし、10進数に直して大きさを理解することも必要です。そこで、3ビット並びを数表示の単位に置き換えた**8進数**、4ビット並びを1バイトとして数表示に置き換えた**16進数**の表記法が使われるようになりました。ただし16進数では、10~15を一字の英字 A~F に代えます。ここでも、10進数に直して理解することが必要になります。なお、英語の文書では、文頭を大文字の英字にする約束がありますので、文頭に数字を使わない表し方が **spell out** です。

1.9 文字付きの数字と読み

数は、それが使われる場面によって、文字や記号を追加します。大きな桁数の数は、3桁または4桁ごとにコンマで区切って見易い表示をします。空白(スペース)を使うこともあります。金銭記号の \$、¥、& は、可算名詞の記号化であって、頭に付けます。声に出して読むときの順は、数の後ろです。物理的な単位記号の kg、m、などは数字並びの後ろに付け、その順で読みます。10 のべき乗表記は数の後ろに書く記号扱いをします。小数点以下に数字が並ぶ実数を**実用**に使うときは、小数部分を一定の桁数に揃えます。固定小数点数がそうです。これが数の**丸め**です。小数の桁位置を言うとき、日本語では「割・分・厘・毛…」を使い分けます。例えば、1を基準として 0.35 の比率を言うとき、「三割五分」のように言います。分数を最も普通に使う方法に**百分率**(%)があります。上の 0.35 は 35% としますが、100 を分母にした分数比率の分子を言い、小数点を使わない整数表示にする方法です。記号の % は数字の後ろに書き、その順で声に出します。百分率の英語は percent(**パーセント**)です。cent はラテン系の言語で百のことです。パーセントは、普通、二桁までの整数で表記し、小数を含む実数表記、例えば 35.4% のような使い方を、できれば避けます。

2. 混乱を起こす数

2.1 言い方と書き順とに混乱がある場合

英語を代表として、欧米語では、20までは個別の言い方があります。注意することは、13から19までは、英語は「…teen」と言います。これは、1の位の数を先に、10を後に言う語順ですので、アラビア数字で書く文字順とは逆順です。20以上の二桁の数、例えば、53では、「50、3」の順で言います。ところが、ドイツ語では、接続詞を挟んで、逆順に「3と50」の言い方をします。この習慣は、右からの横書きであるアラビア語の習慣に影響されたのではないかと筆者は想像しています。もう一つ、逆順の言い方を挙げます。お金の額を英語風の横向きに書くとき、例えば、「\$34」「¥1500」のように通貨記号を頭に書きます。しかし、それを読むとき、記号の読みを数字の後に言い、複数名詞扱いです。数をアラビア数字で書くときと、読みのアルファベット文字綴りで書く区別があります。例えば、1965 とアラビア数字で書いてあるとき、数学的に、千九百六十五の意味に取ります。千単位に区切って読む数字であることを区別したいとき、区切り記号を入れて、1,965 のように書くこともします。年号として読むときは、二桁単位で言い、nineteen sixty-five と発声します。

2.2 デジタル時計の数表示

身近に普通に見られ、また利用の多い数の表示器は、時計です。伝統的な時計は、針の回転位置を見る丸型ですが、これは**アナログ時計**と言うようになりました。数字が全部書いてない表示もあります。少し離れて見る時計は、アナログ式の表示が見易いようです。大都市では、シンボルマークとして親しまれているものがあります。日本では、銀座の(旧)服部時計店、札幌市時計台、海外では、ロンドンのビッグベンがあります。アナログ時計の文字盤は、1から12までの数が書いてありますが、12時から1時の間の呼び方が、0時なのか12時なのか判りますか？実は、個数も順序数も、0からではなく、1から始めるのが一つの生活習慣であるからです。航空や鉄道などの時刻表は24時制を使って間違いを避ける方法を取ります。日常使う12時制でも、0の用語を使いません。**デジタル時計**では、12時制と24時制の切り替えができるようになっています。12時制の場合、正午の12時になったとき、文字表示が午前(AM)から午後(PM)に変わり、深夜の12時では逆に午前になります。そのまま12時の表示が続き、12時59分の次に1時になります。ただし、NHKの正午のニュースでは、0時で画面に表示されています。24時制では、夜中の23時59分の次は0時00分から始まります。アナログ時計は、ちらりと眺める見方もできますが、デジタル時計では数を読むことで理解しています。筆者の経験ですが、3時半に面会する約束を、デジタル表示の時計で3:50になったときに出向いた失敗をしたことがあります。

2.3 満と「かぞえ」の使い方がある

連続したアナログ的な事象で、0と決めた位置の前後をデジタル的に丸めるときの約束で混乱することがあります。数学的な座標系を考えると、目盛り表示の数値は(… -3, -2, -1, 0, 1, 2, 3 …)とします。しかし、負数と0とを使わない「かぞえ」の方式が普通でした。人の年齢を言うとき、生まれた年を1歳とし、正月になると1歳を加える数え方を**数え年**と言います。12月生まれは、すぐに2歳になります。1月生まれは、まるまる一年経ってから2歳になります。年齢に使う数は順序数ですが、或る幅を持った範囲を一つの数で表しています。時間軸をアナログ数として、整数化する丸めです。第5章を参照してください。満年齢で言うときは、生まれた月日からの経過年齢を、(満)2歳6カ月のように言います。数学的な正確さで言いたいときは、実数を使って2.5歳のよう言うことがあります。

一方、階数の多いビルでは、階に順序数を割り付けます。日本とアメリカでは、地表と同じ高さの階を1階とし、そこから上に2階・3階…と順序数を当てる「かぞえ」方式です。ところが、ヨーロッパでは、日本で言う1階を、英語は**グランドフロア**、フランスでは**レドショッセ**と言い、2階から上の階番号を1からの順序数で数えます。数学的には、0から始める数え方ですが0階とは使いません。地階は別系列の番号とし、負数を使いません。

2.4 電話機とパソコンのテンキー文字配列

電話機とテレビのリモコンでは、俗称で言う**テンキー**があって、指で押す、または叩くデータ入力装置です。1～9までと0を加えて10個(テン)の数字キーを並べてありますので、テンキーと呼びますが、英語は numeric keypad です。電話機の数値キーの文字並びは、左上から右下に右書き、3行4列になっていて、0が4行目の中央に置かれています。ところが、電卓そしてパソコンのキーボードでの文字並びは上下の行並びが反対になっていて、左上から(7, 8, 9)、(4, 5, 6)、(1, 2, 3)、(*, 0, #)です。後者のキー配列は、金融業務で、コンピュータに数値データを送信する、数字入力専用のキーボードとして製作されています。5のキーには小さな突起があります。数字入力のオペレータは、数値キーボードを見ないで、伝票の数字だけを見て作業をする、いわゆる**タッチタイピング**の訓練をして作業効率を上げます。電話機との共通利用を考えたキー配列の統一化はできませんでした。

3. 小学校の算数教育を再確認する

3.1 一桁の足し算の教育から始まる

俗に言う「1足す1は2」の算術が、教育の始めです。結果が2桁になると、一つ上の桁に1を加えます。これが桁上がり、または繰り上がりです。暗算ができるようになるまでには日数が掛かります。指を使って計算する場合、指は10本ですので、桁上がりの計算は案外難しいのです。図1のような簡単な算盤(アバカス)は、10以上になる数の理解を助けます。練習にたっぷり時間を割かないと、複数桁の足し算と引き算の理解ができません。この技能を習得するための反復練習を、ドリル学習と言い、問題集を使う必要があります。アメリカの心理学者ソーンダイク(E. L. Thorndike; 1874-1949)が使ったのが始まりとされています。引き算のドリル学習では、結果が負数になる問題を含ませていません。負数は、小学生には難しい概念です。桁数の大きな二つの数の足し算と引き算とを筆算で進めるとき、位取りを上下に揃えて数字を書き、下位(右側)の桁から一桁ずつ順に計算します。繰り下がりが必要になることを覚えます。



図1 アバカス(Wikipedia)

3.2 補数の概念が使われている

引き算を習うとき、同じ桁位置の引き算で、小さな数から大きな数を引くとき、直前の桁から10を借りてくる方法を教わります。意識をしていますが、補数の考え方を使っています。例えば12-7の計算をしなければならないとき、2から7を引くことができませんので、直前の桁から1を引き、引く数7の、10の補数3と、2とを足す計算をして5を求めます。補数の解説は、高校までの数学の教科書では取り上げていません。コンピュータを利用する数値計算を勉強すると、2進数・8進数・16進数の補数表現の説明があります。しかし、そもそも、補数の概念を習っていなかったことが災いして、この部分についての理解が難しいようです。

3.3 負数は座標の考えを使う

アナログ的な事象を理解する方法に、数字を添え、物差しのような目盛りを付けて理解する方法があります。これを順序数で区別するとき、「1, 2, 3, …, 9, 0」が繰り返して現れます。身近には、この順序数を眼にすることが多いのですが、マイナス方向にも順序数を割りつける例に、寒暖計の目盛があります。図2の寒暖計(Wikipediaより)は、摂氏(°C)と華氏F:ファーレンハイト)の二種の表示目盛が付いています。日本では華氏目盛を使いませんが、英語圏では、今でも親しみやすい単位として日常的に使っています。華氏の温度目盛は、人が感じることでできる実用的な温度範囲で、判り易い正の整数を割り付け、マイナスの数を使わないように決めたものです。図2の寒暖計は座標の表示法を使っています。摂氏の表示は、0を境にして逆向きに目盛が振っています。マイナス符号(-)が書いてないことに注意します。最近の温度計はデジタル表示が多くなっていて、0度以下をマイナスの記号を表示しています。摂氏の温度系列の方を使うのが科学的だと思うのですが、必ずしも実用の面で合理的とは言えない面もあります。

何かの形の寸法を測るとき、物差しを使います。物差しの目盛は、0を起点として一方向に割り付けます。マイナスで表示する寸法は使いません。真っすぐに距離を測るだけではなく、曲面に沿わせて、凸多角形の状態で測ります。女性の体形を言うとき、バスト・ウエスト・ヒップの寸法がそうです。図形を図に描いて説明の寸法数値を記入するとき、実物に物差しを当てて寸法を測る作業に使うことを考えて、計算しなくても済むようにします。例えば、外形が円形のパイプであれば、半径ではなく、直径か周長を使うのです。対称な図形ならば、中心から片側の寸法表示で済ませることができるとは思うのですが、実物で、中心位置に物差しの目盛を当てるのが便利になっていなければ、意味がありません。つまり、マイナス符号付き、またはマイナスの概念を含む数を使うことを避けます。

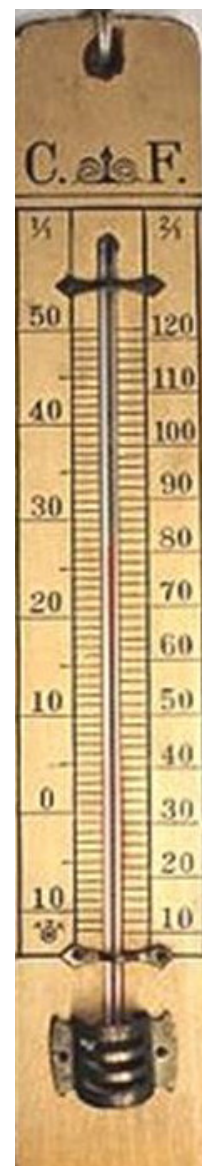


図2 寒暖計(Wikipedia)

3.4 掛け算の基礎に九九を覚える

日本では、小学二年生から、掛け算計算の基礎に「 $2 \times 2 = 4$ (ニンガシ)」ではじまる**九九**を朗読させて覚えさせます。日本語では、一桁の数を一文字のカナで言うことができます。「イチ、サン、ロク、シチ、ハチ」はカナ2字の表記です。日本語での発声単位は、英語の音節(シラブル)の区切り方とは違いがあって、1音節と2音節の読みを使い分けます。九九の計算規則は、空を含めて、口調の良いカナ8音にまとめています。歌にして覚えることもします。声に出すとき、息継ぎの間(ま)も勘定に入れて、4 または8の音節単位に揃えて覚え易くします。子供が計算に九九を応用しているときは、声を出し、自分の耳で聞き直して確かめています。子供は、声を出さないときも、口がもぐもぐと動くのを見ることがあります。大人になると声に出すことをしませんが、頭の中で九九の音節を再現していて、正確さを確認するためのフィードバックを掛けています。つまり、自分に語り掛けているのです。理屈で覚えて利用していると、忘れたり、語順を間違えたりしますが、**復唱**は自分の声を聞いて確認する行為ですので、正しい対応につながります。歴史上の重要な事項の年代、電話番号の数字並び、無理数である2や3の平方根の値などを、語呂合わせで覚えることもそうです。逆に、数を決めることにも応用します。他のことと関連を付けた覚え方を応用するのが**連想記憶**です。例えば、言葉遊びに**数え唄**があります。連想記憶は、情報技術の課題としても研究されています。しかし、ここで言いたいことは、声に出すことの有効性です。

3.5 掛け算は九九と足し算を応用する

掛け算の原理は、同じ数を何回も足す処理です。手回しの計算機があると、掛け算が繰り返して足す計算であることを実感として理解できます。掛け算を代数式($A \times B$)と書くとき、掛ける方の数 A を**乗数**(multiplier)、掛けられる数 B を**被乗数**(multiplicand)とします。掛け算は、掛ける順番を換えて($B \times A$)としても同じ結果になります。掛け算規則は**交換律**(commutative law; **交換法則**とも言います)が成立します。手計算で複数桁の整数の掛け算をするときは、乗数を先に書いて、その下に被乗数を書き、一桁ごとの被乗数で計算した結果をずらし(シフトと言います書いて、最後に全体を縦に加算します(図3)。手回し計算機(図8)、または電動計算機(図9)では、乗数と被乗数とは別々のレジスタを使います。乗数を、定数入力用レジスタにセットしておいて、ハンドルを回して複数回の足し算で実行させます。ハンドルを回す回数を節約するため、被乗数が二桁以上のときは、計算結果の入る**加算用レジスタ**全体を右にずらして足し算の入る位置を変えます。これを**シフトする**と言います、この処理のできるレジスタを**シフトレジスタ**(shift register)と言います。電動計算機は、モータが回してくれます。その回数を、被乗数レジスタで確認します。

	1 2 3
×	4 5 6
	7 3 8
	6 1 5
	4 9 2
=	5 6 0 8 8

図3 筆算の掛け算

3.6 整数を扱う割り算は二種類ある

日本語の環境では、助詞の「で」と「を」とを使って「 A を B で割る」を「 B で A を割る」と言い換えても計算間違いをしません。しかし、英語では、割り算を言葉で説明する表現が前置詞違いで二種類あります。数学的な意義で割り算を言うとき、「divide A **by** B 」と言います、 A/B を意味します。もう一つ、「divide B **into** A 」の言い方がありません。意味は、「 A を B で割りつける」です。割り算の答えは整数(**商**: quotient)で表し、**余り**(remainder)も必要になる計算です。例えば、ロール状の長さの長い紙を、一定長さに切り分けると、最後に半端な長さが残るとき、これが余りです。普通の電卓を使って商と余りの計算をしたいときは、まず普通の計算で代数的な割り算をして、答えの小数以下を切り捨てて商とします。それから、元の数から商×被除数の値を引いて余りを計算します。プログラミング言語では、余り(**剰余**)だけを求める関数名は**MOD**です。英語の moderate が原義のようです。

	4 5 6	…答え
1 2 3)	5 6 0 9 0	…序数・被除数
— (× 4)	4 9 2	
	6 8 9 0	
— (× 5)	6 1 5	
	7 4 0	
— (× 6)	7 3 8	
	2	…余り

図4 筆算の割り算

3.7 引き算と割り算は交換律が成立しない

引き算を代数表記で($A - B$)と書くとして、書き順を変えた($B - A$)は、絶対値が同じで、正負の符号を逆に付けます。代数的な計算では、数は負の数を引く計算もできます。小学校では負の整数を扱いません。算術計算は、結果が正の整数になる場合だけを演習します。小さい数から大きい数を引くことはできない、と約束し、答えが無い、または 0 とします。この約束は、お金を扱う計算のときに困ります。そのため、引き算の順序を逆にして計算し、その数を赤字で書くか、三角記号△を結果数字並びの前に書く表し方をします。

4. 数式の編集と読み書き

4.1 数式を文章に表す

科学技術の専門書は、普通の文章による説明に加えて、数式を使います。**数式**(mathematical expression; mathematical formula)とは、数または数を表す文字(**被演算子**; operand)を、**算術演算子**(operator)または**論理演算子**で繋いで、数を**評価**(evaluate)する**文**(statement)です。論理演算については、第 7 章で改めて解説しますので、この第4章は算術式についてまとめました。数式は、特殊な文字や記号を使い、二次元の抽象画のようにも見えます。数式は書き順があり、声に出して言うことでコミュニケーションをします。一次元の文章にして相手に説明する必要があります。その相手は、人だけでなく、コンピュータも含まれます。数値計算をさせるための算術演算のプログラミングを書くことが、文章化の実践的な応用です。元の式の本質を損なわないように新しい数式を提案する、または、式の形を変更することの研究は、計算精度を落とさず、計算時間を短くする目的もあって、**数値計算法**と言う専門分野が育つことになりました。数式を読み上げる、つまり声に出して伝え、正しく理解してもらえるようにします。前節3の初めに引用した「1 足す 1 は 2」、そして**九九**を朗読するとき、例えば「さぶろくじゅうはち」は、文としては舌足らずの句構造です。主語述語を備えた文に言い換えると「1 足す 1 は、2 です」、「3 掛ける 6 は、18 です」と書きます。この文は**恒等式**(identity)の言い方になっています。数式、例えば「 $A=B+C$ 」は、イコール記号(=)の左右が等しいという意味で使う場合は**等式**(equation)と言います。しかし、比較して異なることがあります。この時の使い方は、関係演算子です。プログラミング言語では**代入文**(assignment expression)と言い、右辺の式を先に計算(評価)してから変数 A に代入すると定義しています。したがって、左右を入れ替えた表現は許されません。**電卓**は、最も単純化された算術計算のコンピュータです。その使い方は、キーの押す順を覚えることです。元の数式が、「 $A=B+C$ 」であるとき、キーを押す順は「 $B + C =$ 」であって、イコール記号(=)の右側を先に計算します。数値表示窓に計算結果が表示され、必要に応じて変数 A に代入します。実は「 $A=B+C$ 」の書き方は、英文の「A is equal to B plus C」を記号化した代入文であって、式を声に出して読み上げるときもそのように言います。計算手順の並びとは左右が逆です。これは欧米人にとっては理解の混乱を招きます。そのため、C 言語の解説には**左辺値・右辺値**の用語があり、詳しい説明があります。日本語の環境では、助詞の(で)と(に)を使って、「A は B 足す C である」、または、その逆順で「B 足す C は A になる」との言い方もできますので、混乱は起きません。

4.2 数式の種類

数学で言う**数式**、または単に**式**の字を持つ用語は幾つかあります。数値計算と関係のあるものは、**算術式**と**論理式**の二つです。式を評価すると言うのは、算術式では数値を求めることです(表 1)。論理式では、**関係演算子**($= < >$)の左右にある式、または数値の、大小関係を比較して、文字表現では(true/false)、数値表現では(1,0)の2値のどちらかを求めます。この値の判定を「if, then, else」文に組み込み、プログラムの実行順の選択を行わせます。比較をする式を**条件式**(conditional expression)、判定の部分を含めた全体を**制御文**(control statement)と言います。イコール記号(=)は、算術式では、表 1 に示したような**代入文**で使う場合と、関係演算子として使う表 2 の場合があります。プログラミング言語では、後者の場合、別の記号表記を使います。

基本的な**四則演算**は、表 1 に、文字と記号の並びの書き方の区別をまとめました。算術演算子(+ - × ÷)は、電卓のキー表示に見られます。日本語のワードプロセッサでも使うことができます。プログラミング言語では半角の英数字で書きますので、記号も(+ - * /)を使います。コンピュータで数値計算をさせるためには、**機械語**に書き直す必要があります。プログラマは、元にした数式を、コンピュータに知らせる**プログラミング文書**(document)に書き直します。コンピュータは、それを読み込んで、機械語に**翻訳**(compile)してから計算を実行します。この目的に使う代表的なソフト名が **FORTRAN** です。Formula Translation を詰めた命名であって、1954-1958 にかけて開発されました。計算結果は、モニタ、またはプリンタに書き出して読むことができます。この機能も FORTRAN に持たせてあって、**文書編集**の書式もプログラミングしなければなりません。この一連の作業はかなり面倒です。初心者向けのプログラミング言語 **BASIC** は、1960 年に発表されました。基礎的を意味する形容詞に、Beginner's All-Purpose Symbolic Instruction Code の頭字語を当てた語になっています。

表 1 基本的な四則演算の書き方数値計算例

算法	代数式	算術式(日本語)	算術式(英文)	電卓のキーを押す順	計算結果 A
加算(足し算)	$A=B+C$	$A=B+C$	$A=B+C$	$B + C =$	5
減算(引き算)	$A=B-C$	$A=B-C$	$A=B-C$	$B - C =$	-1
乗算(掛け算)	$A=BC$	$A=B \times C$	$A=B * C$	$B \times C =$	6
除算(割り算)	$A=B/C$	$A=B \div C$	$A=B / C$	$B \div C =$	0.6666...
備考:	右端の計算結果は、 $B=2$ 、 $C=3$ を代入して得られた値です。				

表2 大小関係の判定に使う関係演算子

JIS 記号	日本語	対応する英語	意味と使い方
=	等しい	equal to	代入文にも使います。
≠	等しくない	not equal to	条件式に使います。
>	を超え	greater than	3を超え、6未満”と云えば、整数ならば4と5であって、3と6は含めません。
<	未満	less than	
≥	以上	greater than or equal to	3以上6以下”は、数値の3と6を含めて範囲を指定します。
≤	以下	less than or equal to	
備考:	演算子記号(JIS)は、全角文字です。パソコンのキーボードでは半角の<, >, = しか利用できませんので、プログラミング言語ごとに固有の記号が使われています。		

4.3 数式はグラフィックスの性質がある

理工系の文書では、数式を説明に使います。図5はその一例です。この図には式が三つあります。鉛の活字を使っていた頃は、式の部分は特別な組版(markup)をしますので、それに対応できる経験のある植字工(typesetter)を抱えた、個性的な出版会社がありました。パソコンを利用する DTP が普及する前、数式の編集ができるワードプロセッサが無かった頃、数式の部分は、手描きでイラストを作り、写真に撮って貼り付けるような対応が必要でした。普通のワードプロセッサで原稿を作成するとき、図5の式(1)は、なんとか対応できますが、式(2)は数式編集のできるソフトを使います。この編集に取り組んだ最初が、D. Knuth(1938-)の

二次方程式

$$y = ax^2 + bx + c \quad \dots (1)$$

において、 $y = 0$ のときの解は

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad \dots (2)$$

図5 数式を含む文の例

開発した TeX(1978)です。その後、改良版の LaTeX(1984)として普及しました。数式を特殊な物(ここではグラフィックス)であるとする考え方は、コンピュータのOS(オペレーティングシステム)が、オブジェクト(物)指向の扱いをするようになってからです。文字と記号並びの数式は、声に出して説明することもあります。声の情報は、一次元の情報並びです。しかし、図5の式(2)に示した根の公式は、二次元的な並びですので、二次元のグラフィックスの性格があります。一次元並びの文字列で説明するには、工夫が要ります。どのように声に出して(読み)言えば、相手に間違いなく伝えられるかを考えます。根の公式は、口調の良い言い方で覚えませんが、式の表し方を思い出す方法の一つであって、厳密に式を再現する目的の言い方ではありません。コンピュータを使って電子的な編集ができるようになって、根の公式のような複雑な式であっても、コンピュータに、文字並びのデータとして書き直して編集処理をする方法が研究されました。その言語を編集記述言語(ML; Markup Language)と言います。インターネットで利用している HTML ファイル(Hyper Text Markup Language)がその一例です。実際の作業は、対話型で進めても、内部的には文字並びのデータが作成されます。数式は、数値計算に利用することが目的ですので、数式解読と実行のできるプログラミング言語が別に必要です。こちらは、声に出して読み上げるには不向きです。改めて、数式そのものの分類や構造についても理解しておく必要があります。

4.4 数式の組版は特殊な技能である

理工系の文書に数式を書く場合は、変数や定数の定義、などから始めます。変数や定数を表す英字は、一文字を当てます。図5では英字の説明がありませんが、暗黙の約束で、小文字の a, b, c...は定数、x, y...は変数です。記号の意味は、「JIS Z 8201 数学記号」にあります。式は、途中で式の合成や分解などの変換があっても、整理した最後の式だけを書くことをします。図5では、式番号(1)から(2)が得られる過程は省いてあります。この省略ができるのは、数学そのものが非常に論理的な道具であるためです。数式は、一並びの文に書けないこともあります。連立方程式は、幾つかの数式を縦に複数行並べます。マトリックスを表したいとき、同じ種類の変数文字を多く使いますので、数字や他の文字を下付きにして区別し、べき乗は上付き文字で表す、などをします。このような約束を使って、掛け算記号を省きます。割り算は、「/」で書くと文字並びの行数を節約できます。しかし、横線を使った分数形式が標準ですし、見易くなります。括弧も、入れ子の関係が分かるように、種類を変え、また、分数式を含むなどのときは、複数行をまとめるため、高さの高い括弧を書きます。数式を版組みするときには、特別な技能を持った植字工が作業しました。したがって、科学技術書を扱う出版社は、独特の存在価値があります。

4.5 数式の書き換え

科学技術関係の文章で説明用の数式を書くとき、掛け算記号を省くことが特徴の一つです(図5)。数値計算に使う場合には、別の形の数式に直します。図5の式(2)は、±記号がありますので、二つの式があることを示しています。また、英字 a が分母にありますので、 $a=0$ の場合は考えないか、エラー扱いをします。さらにルート記号 $\sqrt{\quad}$ の中の式が負数になるときも、数値計算では解が無い、エラー扱いをします。したがって、数値計算に使う実用数式は、演算記号を明示した**算術式**(arithmetic expression)に変え、制御文などを加えた複数の式を並べます。変数や定数には数を当てますが、大きさを持った数値とともに、物理的な**次元**(dimension)と**単位**(unit)とが有ることを**宣言**(declaration)しておいて、中身の**定義**(definition)を決めます。技術的な習慣は、なるべく桁数の少ない固定小数点数で表し、それに合うような単位系を選択します。使う数の値が決まって、始めて、数値だけを扱う数値計算ができます。数学が抽象的な学問であると言うことの意義は、次元と単位とを捨象するからです。

4.6 数式は計算の優先順序がある

数学を扱う教科書や参考書に書かれた代数式は、文字並びの表記を短くします。変数文字は英字1字にします。同じ英字名が複数必要であるときは、一字の数字または英字を下付きにして添えて区別します。掛け算記号は省きます。割り算記号も省き、分数形式を採用します。この数式を、数値計算に使う場合には、演算子記号を補った式に書き直します。このとき、演算子に優先順位があることに注意が必要です。さらに、括弧を補って、先に評価する式の範囲が分かるように書き直します。演算子には**優先順位**が決められていて、数式の文字並びの、どの部分から先に計算するかを決めます。「乗除」の演算は、「加減」の演算よりも順位が高い、と約束します。乗除算単位また関数があるときは、前もって計算値を求めて、それを仮の変数記号に代入しておきます。最後に、加減算だけを使った式を計算します。ここで、括弧 $()$ は、その内側の数式が最高の優先順位を持つ、と約束します。括弧を入れ子状に使うこともあって、最も内側から計算を進めます。括弧の内側の値が確定したときに括弧を外した式に変更します。最後に加減算だけの式が残ります。

4.7 大数と小数

学問としての数学では、数の性質を考えて、何種類かの分類をします。数値計算に実用する数は、小数点を挟んで、上位を大数、下位を小数とする、或る桁数に制限した数字並びで表します。小数を持たない数が整数です。コンピュータ用語では、**整数**と**実数**(real number)に分類します。**有理数**(rational number)とは、分数で表すことのできる数を言います。整数は分母が1の場合です。小数を持つ数は、分母に10のべき乗数を使って、整数を成分とする分数の形に直すことができます。循環小数は、整数成分を使う分数で表せば正確な数として扱うことができます。しかし、実数化すると、無理数と同じように、無限に数字が並ぶ数です。したがって、実用上は或る桁数の数字並びに抑えます。一般的な処理は**丸め**と言います。意味を持つ数の桁並びを**有効数字**と言います。何桁にするかは、実用場面で決めます。円周率は、代表的な**無理数**です。三角関数、対数、平方根などには無理数も混じりますが、適当な桁数のところで丸めて有理数で扱います。正確な値で使いたいとして、有効数字を大きくしたいと思うのが普通の感覚です。すると、例えば「円周率を3と教えるのは間違っている」と主張する人もいますが、迷惑な正義です。実数を正確に使いたいときは、コンピュータのメモリやレジスタの寸法を大きくして有効数字が大きい数値を扱うことができるようにします。この精度を区別したいとき、場面に応じて、単精度・倍精度・四倍精度のような数を使い分けます。数の入れ物であるメモリやレジスタでは、整数の数字並びは右詰め、実数は左詰めです。コンピュータ内部では、ビット並びの2進数で表した**仮数部**と、小数点位置がどこに在るかを示す**指数部**との二つの数の組で表します。この形で扱う実数を**浮動小数点数**(floating point number)と言います。これに対して、数字並びを右詰めにして、小数点位置を或る桁位置に固定しておく表し方があります。この数を**固定小数点数**(fixed point number)と言います。算盤(そろばん)は、固定小数点数の計算道具です。固定小数点数は整数の計算方法を利用しますので、加減算は高速で処理できます。一方、**乗除算**では、小数点位置を考えない整数としての計算をしておいて、小数点がどの桁位置に**移動**するかの計算が追加されます。関数電卓、そしてパソコンも、浮動小数点数の計算をする装置です。しかし、普通に市販されている簡易電卓は、算盤と同じ固定小数点数の計算器です。このため電子算盤の愛称で呼ばれます。移動と浮動の区別は、紛らわしいのですが、元の英語を翻訳した用語です。

5. 数の丸め

5.1 有効桁数

数を数字並びで書き表すとき、数の精度の指標が**有効桁数**です。整数は0を頭に書きません。実数では、絶対値が1より小さな数は、小数点以下に0が詰まることがあります。この0は、有効桁数に含ませません。例えば、無理数の円周率を数値計算に使うとき、3.14 とすると3桁、3.1416 とすると5桁の**有効数字**を持たせると言い、最後の桁に±0.5の誤差がある、と判断して使います。小数の、小数点以下の数字並びの後ろに0が続くとき、どこまでの0が数の精度に関係を持つか、の判断が必要です。整数は、暗黙の約束では、すべての桁が有効であると判断します。実数では、数の精度を示す必要がある0は残します。例えば、3.0、3.00、3.000は、有効数字の桁数が、2、3、4桁であることを示しています。数値計算では、できるだけ、数の精度が保障できる桁数に整えます。その具体的な処理が**丸め**(rounding)です。実用する方法は、**切り上げ**(round up)、**切り捨て**(round down)、**四捨五入**(round off)の3通りです。小数点以下に数字が並ぶ実数を、整数に丸めることは、元の実数の疑似的な整数化です。1の位(くらい)での丸めがどのようになるかを、図6に示しました。**切り上げ**は、0の範囲がありません。これは「**かぞえ**」の方法です。日本では、建物の階数の数え方に応用しています。建物は高さ方向の寸法を持ちますが、床から天井までが階の範囲です。切り捨ては、0の範囲が2倍です。四捨五入は、マイナスからプラスまで、0を含めて、実数範囲がすべて同じ長さです。これは、実数の数値計算に応用すると、計算誤差を最小にできる丸めです。コンピュータでは、実数型の数を整数に変換して使いたいときの方法が幾つかあります。実数型の変数を整数型の変数に代入すると、四捨五入で丸められます。切り上げまたは切り捨てにしたいときは、元の実数に0.5 または-0.5 を足してから代入します。整数化にする関数に、MS-EXCELには、INTとして定義されている関数があります。負の実数は、整数部の絶対値が大きくなることに注目して下さい。

実数座標値	...	-3	-2.5	-2	-1.5	-1	-0.5	0	0.5	1	1.5	2	2.5	3	...
座標目盛		:	:	:	:	:	:	:	:	:	:	:	:	:	:
切り上げ	...	-3	-2	-1	0	1	2	3	...						
切り捨て	...	-2	-1	0	0	1	2	3	...						
四捨五入		-3	-2	-1	0	1	2	3							
INT 関数	...	-3	-2	-1	0	1	2	3	...						

図6 実数を丸めた整数値の範囲

5.2 覚え易い呼び数を使う

数学で言う無理数は、正確に表したいならば、小数点以下の桁に無限に数が並ぶのですが、或る位どりを決めて**丸め**ます。つまり、実質的には整数扱いをします。例えば、円周率(π)の表し方があります。常識的には、と小数点以下二桁の数 3.14 を使います。分数表示の 22/7 も便利です。このような、実用に使い、正確な理論値にこだわらない数値を**呼び数**(nominal number)と言います。設計計算などで「正確な計算が要求される場合」、「桁数が多くなる数、材料寸法に種々の種類がある」などのとき、覚え易い数値に丸めることをします。日本は、アメリカに多くの技術を学びましたので、元の寸法がインチ系列であったものを、ミリ単位の呼び数で使う例があります。例えば、1インチは 25.4mm ですが、呼び寸法を 25mm にします。7/8 インチは 22mm、3/4 インチは 19mm とします。元のインチ寸法の材料を測ると、呼び寸法のミリとは違うことがあります。そのため、材料検査をする段階で、「寸法をごまかした」、と問題にされたことも起きました。インチ系列とミリ系列の寸法とが混在して、混乱する例に自動車などで使うネジがあります。見かけは同じ寸法に見えても、ネジとして合わない組み合わせができない悲劇が起きます。これを避けるために、国際的な話合いでISO規格が合意され、日本ではその取り決めに尊重して工業標準(JIS)として採用しています。

5.3 大小の寸法系列を合理的に決めたい

我々の身の周りには、寸法違いで大きささまざまな品物が使われています。一つの例として、お皿を考えます。食器棚には、小皿から大皿まで、各種の寸法を用意しますが、寸法を選ぶとき、一回り大きい、または小さい、の言い方をします。工学的に判断するときは、等比数列になるような寸法系列で、大小の種類を決めます。上の項で触れた、工業で使う大小違いで多くのネジを必要とするとき、等比数列になるような寸法系列を決めておくと、寸法種類の増加を抑えることができます。等比数列を決める基準に、JIS Z 8601 **標準数**があります。これは1から10までの尺度を常用対数目盛(0~1)で表示しておいて、それを 10、20、40 等分した座標値の数値を、3 桁に丸めて使うように提案したものです。10 分割のときの数値列は、「1.00、1.25、1.60、2.00、2.50、3.15、4.00、5.00、6.30、8.00、10.00」です。

5.4 お金の計算で使う丸め

お金絡む計算は、切り上げと切り捨てが実用されています。お小遣いをもらう場面で、1253円が必要な金額であると、100 円未満を切り上げて 1300 円を渡すのがそうです。桁数が多くても、正確な数値を扱っている業務の代表は、銀行などの金融機関です。利子や税率の計算は、元の金額に利率の掛け算処理が入りますが、その計算では1円未満は切り捨てる約束です。日常的には、消費税の計算があります。銀行預金の利子の計算も、元金に利率を掛けた利子は、1円未満を切り捨てます。利子との合計を次の元金にします。そうすると、例えば、年利率が3%であるとき、30円だけを預けると、利子は1円未満の 0.9 円ですので、いつまで預けておいても利子は加算されません。この規則は、想像している以上に厳格な計算法です。経理の計算のとき、最後の桁に1円でも誤差が発見されると、どこで誤差が出たかを徹底的に調べます。桁数の多い整数をまとめて説明に使いたいとき、有効桁数を小さくするように、後の桁を 0 に丸めることをします。例えば、お金の計算では、千円以下を切り捨てる、などの処理をします。そうすると、下三桁は、いつも 0 が並びます。このとき、(×千円)の単位を書いて、0 の数を省いた数を書くことがあります。

5.5 お金の小数単位にセントがよく使われる

セントは、ラテン系の言語で 100 の意味があります。セントの言い方は、通貨の下位単位に多くの国で採用されています。アメリカでは1ドルの 100 分の1です。それ以下の単位名はありません。日本も、戦前までは円の下位単位に銭と厘とがありました。銭とセントとは音が似ています。敗戦直後のインフレで通貨の円の価値が約百分の一に下がり、銭以下の単位を使わなくなりました。数の桁数が増えましたが、小数を使う場面が無くなったことの利点もあります。しかし、お金に下位単位があると、小数点以下に2桁までを扱う場面で実数を説明するときに判り易くなります。例えば、職人言葉では、長さをメートル法で言うとき、1m50 cm を 1 円 50 銭のように言うこともしました。

5.6 設計計算書に見る数の丸め

図7は、橋の設計計算書の一部を例示したものです。1950年代の計算書は、手書きでしたが、図7は、それをMS-EXCELで再現したものです。その年代の实在橋梁の耐荷力を調査するとき、当時の設計計算書がどのように作成されていたかを知るために作成したものです。単純な四則演算が使われています。どこから参照されたかの情報は、数値セルを参照すると分かります数式通りに計算された数値は内部データとしてセルに保存されますが、印刷用書式が設定されていますので、イコール記号(=)の右に、丸めた数値として表示されています。数式の表現方法とは逆順です。なお、力学単位が助数詞として追加されています。

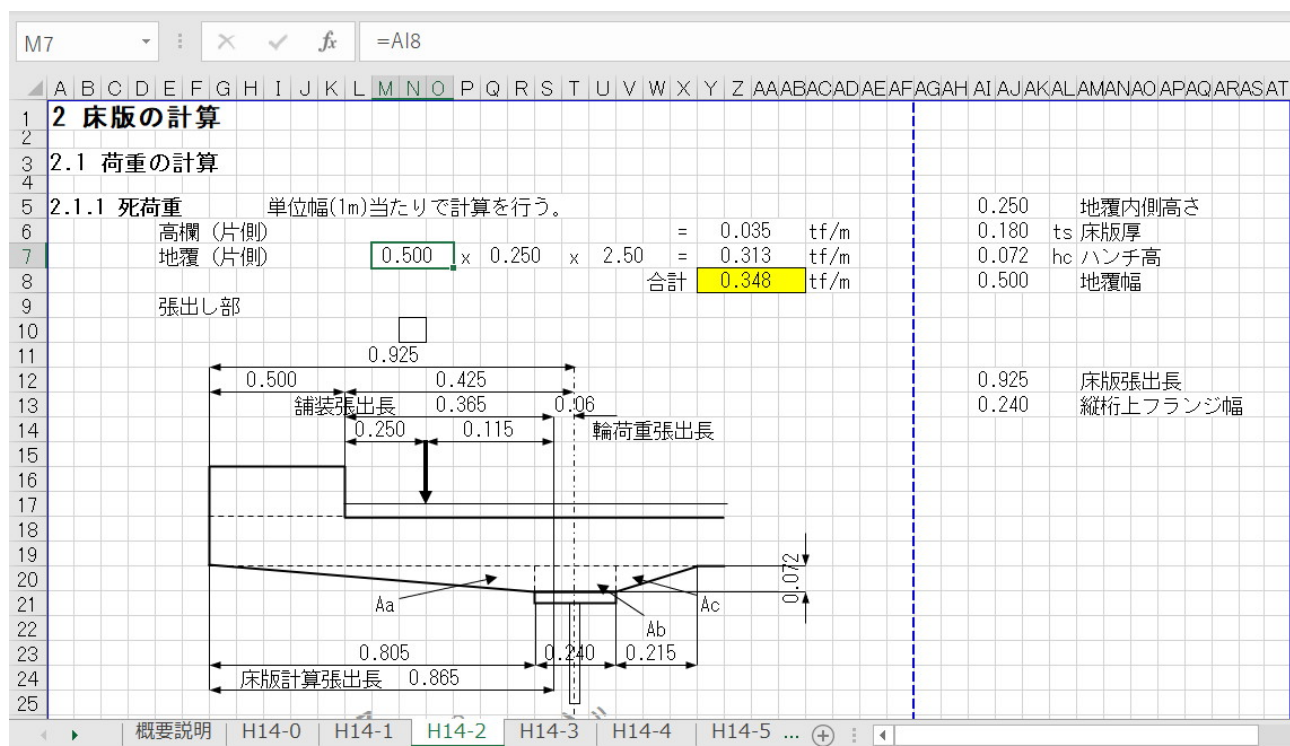


図7 古い橋の設計計算書の MS-EXCEL による再現計算の例

6. 算術計算の道具

6.1 固定小数点数の計算専用機械

社会生活の場で、物の個数を数えるときの日常的な行動は、指(ゆび)を使います。「指折り数える」の慣用句があるように、人の指は、最も基本的な計算道具です。英語の digit はラテン語原義で指のことです。古くは、1から9までのアラビア数字(算用数字)を言う語です。現代では0も含めます。両手を使うと、10までの数量を簡単に数えることができます。しかし、10を超えると、幼児には「手に負えない」数になります。第3.1節の図1に示したアバカスは幼児に数を教える最も単純な道具です。この章では、小学校教育でも扱われるようになった四つ珠算盤の話から始めます。数値計算専用の装置は英語では Calculator と言い、多機能に対応できるパソコンは Computer であると言い分けます。パソコンは、広い用途に応用できますので、プログラミングをして作成したソフトウェアで間接的に計算に利用します。電卓は数値計算用に特化した小型のコンピュータであって、卓上計算機(desktop calculator)と言い分けています。パソコンのアクセサリソフトには電卓があって、市販の電卓と同じようなキー操作で利用できますが、浮動小数点数専用の使い方になっています。一般商店では金銭登録機(cash register)が必需品ですが、これには最も単純な固定小数点用計算機が組み込まれています。パソコンが普及する前には算盤と並んで利用していた機械式の装置でした。銀行を初めとした金融機関では、10桁×10桁=20桁の計算ができる手回しの機械式の計算機が使われ、効率の上がる電動計算機もありました。これらは科学技術計算にも多く利用されていました。これらの計算道具は、算盤と同じ原理の、整数の算術計算の道具であることを解説します。

6.2 算盤

現代は、電卓やパソコンの利用が便利ですが、算術計算の原理を理解させる教育では、四つ珠算盤は非常に重要な意義を持っている道具です。戦前までは、中国で使用されていた十六進数計算用算盤の影響を受けた五つ玉算盤が使われていました。これは、上段に珠二つ、下段は珠五つです。



図8 標準的な四つ玉算盤

上段を珠一つにした算盤が大正年代に使われていました。図8は十進数の計算用として1930年代中頃に標準化された製品です。図8の算盤は、23桁あって、全体が一続きの整数表示用レジスタです。途中をいくつかの小範囲のレジスタに分けて、数値計算に使う部分と、数値をメモする場所に使い分けます。単位を合わせるときの目印に、4桁ごとにマーク(定位点)が付いています。下の段は1~4の数表示用、上の段は5の数専用であって、この構成は五進数の原理です。計算の実践では、暗算で5の補数と10の補数とを素早く計算しています。算盤を使う計算では、小さな数から大きな数を引くことになるのを避けます。そのようなときは「別の桁位置を使って正負を逆にした引き算をする」と算盤に達者な人が説明してくれました。電卓が普及して、機械式の計算機を眼にしなくなりましたので、補数を説明する方法がなくなりました。

6.3 手回し計算機

手回しの機械式計算機(図9)は、10桁×10桁=20桁の計算ができる有限桁数の正の整数用の計算機械です。有限桁数化した実数(固定小数点数)の計算にも使います。小数点位置を表示するマークは、利用者側で位置を設定します。レジスタが三つあります。最上段の数値セット用10桁レジスタ①は、被乗数用です。割り算のときは除数用です。ハンドルの回転数が表示される10桁レジスタ②は掛け算の乗数表示用、割り算では商が得られます。20桁表示できる加算レジスタ(アキュムレータ)③は、割り算のときは被除数を入れておきます。レジスタ②と③と合わせた全体は、左右に動かすことができます。これをシフトすると言い、標準位置は、本体の左に寄せます。。小数点の位置合わせに使います。ハンドルを右に回すと加算、左に回すと減算が実行され、レジスタ②のシフト位置の数が増減します、ハンドルを右に回すと①が③に加算され、その回数が②に加算されます。左に回すと③から①を引く減算です。レジスタの表示数値が③<①であると、引き算の実行によって③は左側数字列の頭に9999…が詰まります。これが補数です。

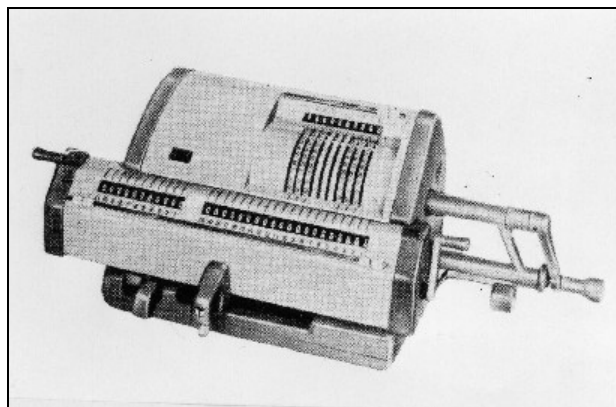


図9 手回しの機械式計算機(タイガー)

6.4 電動計算機

図10は、図9の手回し計算機のハンドル部分をモータに代えて効率を上げた計算機です。当時の価格は、小型自家用乗用車と同じほどの高価な機械でした。手回し計算機のレジスタ番号①②③と機能を合わせて説明します。数値データの入力用レジスタ①は、各種スイッチと合わせたキーボードから、9行 10列に並べた数字キーを選び、確認のためのレジスタ表示用窓がキー並びの上にあります。レジスタ②と③とは本体上部にあって、左右にシフトします。計算の実行命令は、すべてキーボードのスイッチで指示します。次節で説明する電卓ではテンキーを使いますが、機械式で数字をセットしますので数字キーが賑やかになっています。

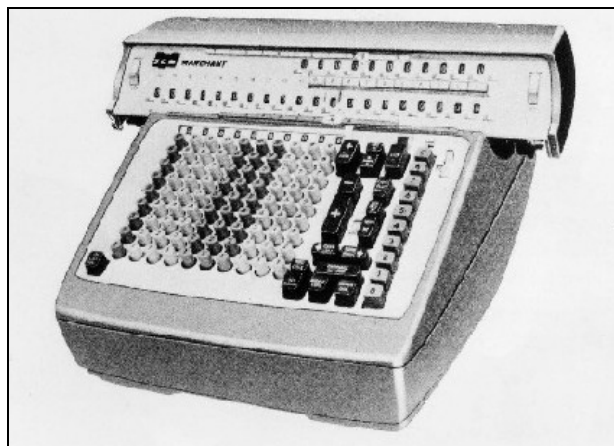


図10 電動計算機(フリーデン)

6.5 電卓

電卓とは、卓上電子計算機を詰めた用語です。机の上に置けると言う意味ですが、かなり大きな装置でした。カシオが1972年に製品化した電卓は、持ち運ぶができる携帯電話程度の小型製品であって、電卓の名称はそぐはないのですが、用語としてはそのまま使うようになりました。算盤と似たところがありますので、通称で電子算盤と呼びました。多くの会社が競いましたので、価格も文房具店で購入できるほどに大衆化しました(図11)。機械式の計算機と同じように複数のレジスタを使い分けますが、12桁までの数値表示用窓が一つしかありません。初期の電卓は、6桁×6桁までの掛け算ができましたが、表示窓も6桁でしたので、12桁までの結果を確認する方法が少し面倒でした。現在は表示窓が12桁になりましたが、掛け算の結果が12桁を超えるとエラーとなり、キー操作はできなくなります。機械式の計算機との違いは、小さな数から大きな数を引く計算では、補数表示ではなく、マイナス符号付きの負数の数表示になって、エラー扱いにならないことです。

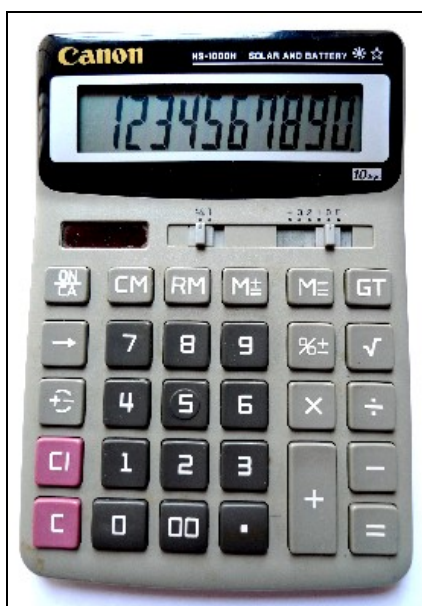


図11 市販の普通の電卓(電子算盤)



図12 パソコンのアクセサリにある普通の電卓

図12は、小型の電卓の機能をエミュレート(疑似下モデル化)してモニタ上に表示したものです。図12と同じような計算ができますが、こちらは浮動小数点数計算用になっていて、大数から小数までの広い数値範囲の掛け算に使うことができます。したがって、科学技術計算には利用できますが、数の小数位置を決めて四捨五入などの丸めの機能がありません。そのため、商用計算には向きません。なお、電卓の種類として、数学関数が見える関数電卓があります。これは扱う数が固定小数点数ではなく、浮動小数点数です。図11のような固定小数点数を扱う電卓は、エラーも起こしますが、このことは、教育用として計算原理を覚えさせる意義があります。

6.6 普通の電卓の使い方

市販の普通電卓は、12桁までの固定小数点数の計算ができる4つの演算レジスタを持っています。数字表示用窓は一つです。それまでのキー操作がどうなっているかは直ぐには分かりません。機械式の計算機の説明で使ったレジスタ番号を踏襲して説明します。一般に、算術計算の道具は、標準状態にする操作から始めます。その計算手順の始めは、算盤用語では**ご破算**です。クリアまたは**消去**と言います。機械式の計算機では、ご破算用の複数のレバーがあります。電気機械式の計算機は消去用ボタンがあります。電卓にも消去キー(C)があります。計算の手順は、数学的な数式を実際計算に応用するときにはキー操作で行いますので、キー操作次第で結果が変わります。これを表3にまとめました。基本的なキー操作を箇条書きで示します；

- 最初にご破算状態にします。幾つかの消去キーがありますので、どのキーがどのレジスタを消去するかをたしかめておきます。
- 現在表示されているのはレジスタ①の内容です。最初は0になっています。
- ユーザが数値キーを押す、数値が入力され、表示窓に数字が並びます。
- 演算記号(+、-、×、÷)のどれかのキーを押すと、①のコピーがレジスタ②に入ります。表示窓はそのままです。
- 次いで数値キーを押すと、前の数値は消去され、入力された数値並びがレジスタ①に入ります。数値の入力を省くこともできます。
- イコールキー(=)を押すと演算が実行され結果が表示され、レジスタ①が書き替えられます。そして、レジスタ③に加算されます。これはメモリ用のレジスタであって、**GT**(Grand Total)のキーでデータをレジスタ①に読み出して、確認のため表示させることができます。
- 4番目のレジスタ④は、データの一時的保存用のメモリに使います。**M**の字を付けたキーが幾つかあって、消去・読みだしレジスタ①との間で加算・減算が実行されます。

図11の電子算盤では、計算結果の数値に「切り上げ・切り捨て・四捨五入」の選択ができる切り替えスイッチ「cut,up,5/4」があります。小数点以下何桁目で丸めるかの桁位置の指定スイッチとセットにして使います。数値を入力するとき、小数点位置が表示画面の右端の外に出るような、大きな整数になると、**オーバーフロー**(**桁あふれ**)のエラーになります。足し算や掛け算の結果で、小数点位置が大きくなって、整数部分が12桁を超えてもエラーになります。又、GT・M 用レジスタも、処理次第でオーバーフローのエラーを起こします。一方、図8のパソコンのアクセサリ電卓では、表示画面の桁数いっぱいを超えると、指数表示に変わり、大きな実数であってもエラーにはなりません。なお、普通電卓でも、平方根の計算ができる関数を持つ電卓器種があって、筆者は愛用しています。

表3 普通の電卓のキー操作で得られる計算結果

代数式	計算結果	電卓のキーを押す順	計算結果
$A+A+A+A\cdots$		(clear) $A+=+=+=\cdots$	2, 4, 6, 8, ...
$A-A-A-A\cdots$		(clear) $A-=-=-=\cdots$	-2, -4, -6, -8, ...
$A*A*A*A\cdots$		(clear) $A\times=\times=\times=\cdots$	2, 4, 8, 16, ...
$A/A/A/A\cdots$		(clear) $A\div=\div=\div=\cdots$	1, 0.5, 0.125, 0.0625, ...
$A+B+B+B\cdots$		(clear) $A+B=+=+=\cdots$	5, 8, 11, 14, ...
$A-B-B-B\cdots$		(clear) $A-B=-=-=-\cdots$	-1, -4, -7, -10, ...
$A*B*B*B\cdots$		(clear) $A\times B=+=+=\cdots$	6, 12, 24, 48, ...
$A/B/B/B\cdots$		(clear) $A\div B=+=+=\cdots$	0.666, 0.222, 0.074, 0.0247 ...
$A+B*C$	13	(clear) $A+B\times C=$	20
$A+B/C$	2.75	(clear) $A+B\div C=$	1.25
$(A+B)*C$	20	(clear) $A+B\times C=$	20
$(A+B)/C$	1.25	(clear) $A+B\div C=$	1.25
$A*(B+C)$	14	(clear) $A\times B+C=$	10
$A/(B+C)$	0.285814	(clear) $A\div B+C=$	4.666666...
備考:	代数式は掛け算記号 * を書かない。計算結果はA=2、B=3、C=4として代入した値。		

7. 論理演算の約束を覚える

7.1 数値計算に応用する2値論理学

論理学(logic)の歴史は、紀元前、ギリシャのアリストテレスにまで遡る歴史の古い文科系の学問です。論理学は、言語学(linguistics)の一分野です。主語・述語の整った文(命題と言います)が正しいか(真 true; 1を当てる)、そうでないか(偽; false; 0 を当てる)、を判断する推論に使います。推論の最も単純な構造は、「前提」の命題と「結論」の命題の組みです。これに代数的な方法を応用するようになったのは20世紀からの記号論理学です。代数学と同じように、文字や記号を使って文の性質や関係を扱います。或る命題が「真であるか偽であるか」の二つの性質だけで判断に使う論理学を、二値論理学と言います。この性質を数学的に扱うとき、2進数の(0 または 1)を持つ変数を当てます。この変数間の算法を決めた代数学をブール代数と言います。

7.2 論理演算の用語と記号に混乱がある

論理学が文科系の学問であった歴史がありますので、論理演算の種類を言う用語は、難しい漢語も使われています(表 4.1)。日本語のワープロ環境では、漢字を使うことができますし、特別なグラフィックス字形の論理記号も使うことに不便はありません。しかし、英語の環境では、論理記号の表し方に苦勞があります。

表4 種々の論理用語と論理記号

論理学用語	論理学の記号 (*1)	集合論の記号	計算機言語 (*2)	英文 (*3)	和文 (*3)
否定 (*1)	$P, \neg P,$	記号に上線を引く	NOT	not	…でない
選言、論理和	$P \vee Q$ (*4)	$P \cup Q$ (*5)	OR	or, and/or	または
排他的選言、排反、非両立	$P \underline{\vee} Q$ (*6)	$P \neq Q$	XOR	exclusive or	どちらか
連言、論理積	$P \wedge Q, P \& Q$	$P \cap Q$	AND	and	かつ
内含、含意	$P \rightarrow Q,$	$P \supset Q$ $Q \subset P$	IMP	if~then, imply	ならば
可逆的内含、対等、同値	$P \equiv Q$	$P \Leftrightarrow Q,$	EQV	equivalence	等しい
全称記号		$\forall x, \Pi x$		all	すべての…
存在記号		$\exists x, \Sigma x$		some	あ(或)る…
帰属		\ni, \in		inclusion	(部分を)含む
包摂		\supseteq, \subseteq		subsumption	(全部を)含む

注

(*1)この列の説明では、論理変数の記号(取り消し線を使う \neg)の方を表記に用いました。式全体を否定にする場合は、式を括弧()で括って、その前に \neg の記号を書きます。

(*2)BASIC 言語で用いる論理演算子の表し方を例示します。

(*3)日常言語のなかで使われる表現と、論理学での定義とでは少し異なることがあります。

(*4)全角英字のVと字形が似ていますので注意して区別して下さい。

(*5)全角英字のUと字形が似ていますので注意して区別して下さい。

(*6)選言記号 \vee にアンダーラインを付けた表記です。

7.3 論理用語と論理記号

算術は加減乗除の4通りの演算規則があります。二つの論理変数間の論理演算を扱うとき、その組み合わせを数え上げると16通りもあります(図13ベン図参照)。個別に演算名を当てることもしますが、基本とするのは4通りに絞ります。表4の計算機言語で言うと、「OR, XOR, AND, IMP」です。一つの論理変数に適用する演算に**否定**(NOT)があります。これは、数の計算の符合変換に相当する、と考えることができます。否定の演算は、論理値の0を1に、1を0に変換します。他の演算則に名前を当てることもしますが、上の五つの演算子を組み合わせれば同じ演算ができます。ORとANDは、算術の和と積の演算と性質が似ていることから、「論理和、論理積」の用語が実用されています。日常見る作文の中で眼にする「または」と「と」の使い方に似ています。「PまたはQ」の言い回しは、「PまたはQ, または、その両方」の意義で使います。XORは、「PかQかどちらか」に限定する意義ですので、**排他的選言**の専門用語があります。IMP(**内含**)は、用語も特殊ですし、演算の約束も難しく、また、引き算や割り算と同じように、交換律がありません。EQVは、イコール記号「=」で代用できますが、場面しだい違った解釈が起きます。

7.4 集合論の考え方も使う

同じ性質を持つ複数の物を、まとめて扱う考えが集合です。これに数学的な方法を応用したものが、狭い意味での集合論(set theory)です。論理演算を説明する図に**ベン図**(図13)があります。白黒写真は、小さな黒い点の集まりです。この個別の点の、或るまとまり(集合)ごとにP, Qの名前を付けて仕分けしておいて、その重なり合いの性質を分類してグラフィックスで表したものです。個別の点に2進数の(1, 0)を当てると約束をすることで、数学の顔をもつようになります。コンピュータの一単位のメモリは複数のビット並びですので、個別のビット位置を定義して使えば、相互の区別ができる複数の点の集合を定義できます。このメモリは、データ型として整数の性質をもちます。数値計算で論理演算をさせるときは、論理変数に整数型のデータを使います。

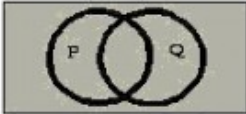
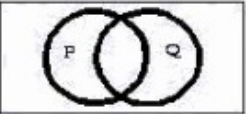
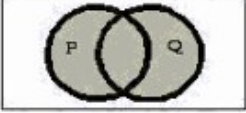



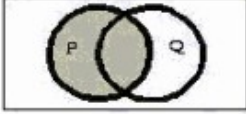
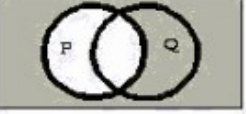
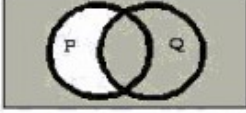
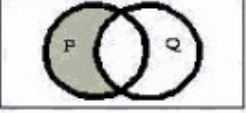
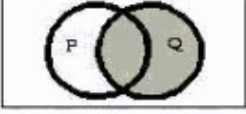
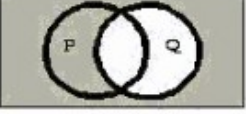

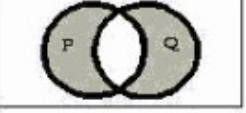
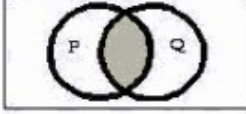
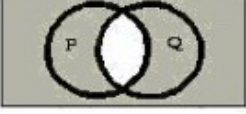
	-15- PとQのトートロジー		-0- PとQの矛盾
	-14- $P \vee Q$ PとQの論理和		-1- ($P \vee Q$) の否定
	-13- $Q \supset P$ 「QならばPである」		-2- ($Q \supset P$) の否定
	-12- 「Pである」		-3- 「Pの否定である」
	-11- $P \supset Q$ 「PならばQである」		-4- ($P \supset Q$) の否定
	-10- 「Qである」		-5- 「Qの否定である」
	-9- $P \equiv Q$ 「PとQは同値である」		-6- 論理的選言、排反 <u>exclusive or</u>
	-8- $P \wedge Q$ PとQの論理積		-7- $P \mid Q$ PとQの非両立

図13 ベン図

7.5 論理演算の基礎知識

小学校で教える算術は、「1 足す1は2」から始め、順を追って掛け算九九を覚え、割り算に進むことを、第3章で説明しました。論理演算も、数値計算(算術)をして演習できると具体的に理解が進みます。その道具には、パソコンのアクセサリから参照できる電卓を、関数電卓の表示モードで使うのがよいでしょう(図14)。市販の関数電卓は、数値計算に使うことを目的としていますので、論理演算の機能がありません。パソコンの関数電卓では、「AND, OR, XOR, NOT」の演算キーがあります。IMPがありませんが、教育利用には十分です。また数の入力でも「2進、8進、10進、16進」の選択ができます。切り替えをすることができて、同じ数を別形式で見ることができます。演算指定のNOTは、10進数以外は補数を計算してくれます。10進数の表示のときは、符号変換として動作します。ただそ。正の整数を符号変換しても、必ずしも補数のビット並びにはなりません。

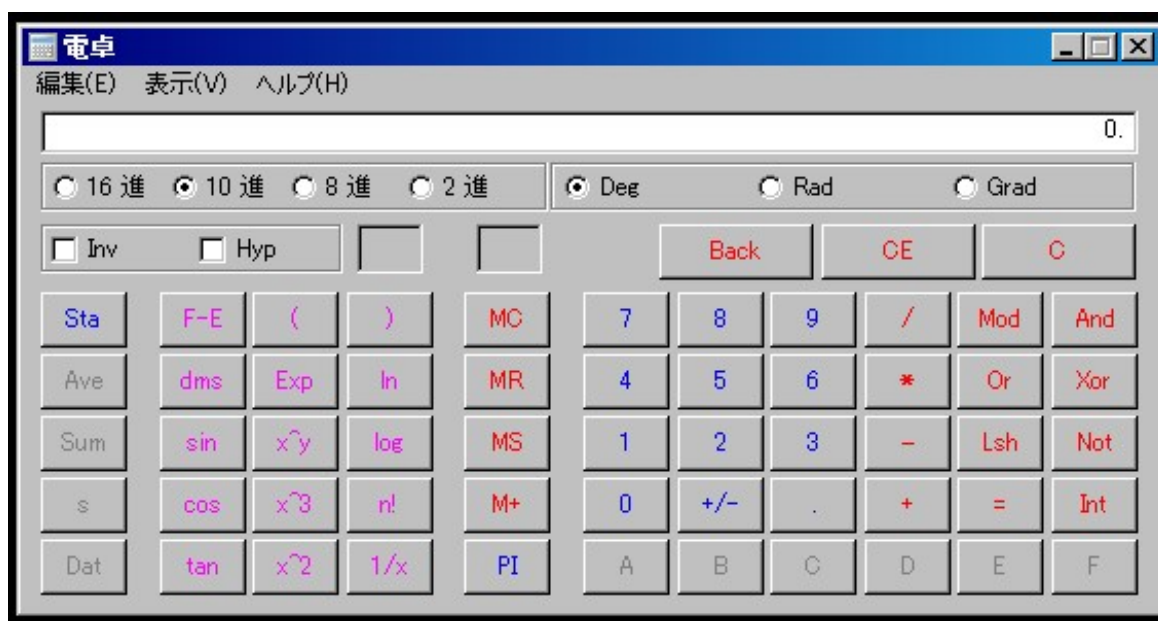


図14 パソコンで利用できる関数電卓の画面(現在はサポートされていません)

7.6 論理演算を幾何モデリングに応用

図13に示したベン図は、論理演算の種類を判り易く説明したものです。二つの論理値の集合P, Qは、円周の外枠で囲っています。この外枠は、別の実体(オブジェクト)であって、論理演算には関係がありません。論理値の集合が在る**実**の部分と**空**の部分の境界を判り易く示すことが目的です。したがって、この境界の両側がどちらも実であるか空であるとき、消去すれば、実の、在る部分の図形が判り易くなります。図15は、二つの正方形図形に**干渉処理**を施して、新しい平面図形を合成する**幾何モデリング**を例示したものです。干渉処理は、足し算(論理和)、掛け算(論理積)、引き算に分類してあります。このデータ構造は、図形要素の辺(オブジェクト)にトポロジー的な特性を持たせていて、辺が図形の実の部分と空の部分とを分けると定義します。二つの図形の干渉処理をするとき、相手の図形に埋もれる辺と、辺の両側が空になる辺を消去します。

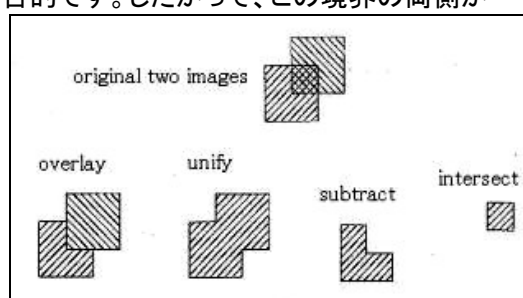


図15 平面図形の干渉処理