第6章　初等関数の計算

> しかし著者がとくに強調したいのは，初等関数の数値計算というきわめて「初等的」な，狭い，そしてすでに研究しつくされていると思われる分野でさえも，対象を見る目をかえていじくれば，単なる落ち穂拾いというのではすまされないようなおもしろい話題が，たくさんあるということである。
>
> 一松信「初等関数の数値計算」(教育出版)

本章では，多項式関数 $p(x) = \sum_{i=0}^{n} a_i x^i$，平方根 $\sqrt{x}$，三角関数 $\sin x, \cos x$，指数関数 $\exp(x)$，(自然) 対数関数 $\log x$ の計算方法について解説する。三角関数，指数関数，対数関数については Taylor 展開に基づいた計算方法を解説するが，現実に使用されているアルゴリズムに比べて効率が悪いことが知られている。従って，これらの方法はあくまで数値計算のエッセンスを理解するための一例題としてとらえて欲しい。

6.1　Horner 法による多項式関数の計算

定義 6.1.1 (多項式関数)
以下のような関数を，複素係数の多項式関数 (polynomial function) と呼ぶ。

$$p(x) = \sum_{i=0}^{n} a_i x^i \quad (a_i \in \mathbb{C},\ x \in \mathbb{C}) \tag{6.1}$$

多項式関数をプログラム言語で扱うには，係数のみを一次元配列に格納しておくだけで済む。特定の実数値 $x = \alpha$ における多項式関数の値 $p(\alpha)$ が必要になれば，以下の Horner 法を使って計算すればよい。このアルゴリズムは四則演算の範囲内で実行可能である。

アルゴリズム 1 (Horner 法)

1. $val := a_n$ とする。
2. 以下 $i = n-1, \ldots, 1, 0$ に対し，以下を繰り返す。
   (a) $val := val \times x + a_i$
3. $p(x) = val$ となる。

前述したように，数値計算においては一般に，加減算より乗除算の方が多くの計算時間を必要とする。Horner 法は乗算の回数が最小で済むため，多項式関数の値を評価する標準的な方法となっている。

問題 6.1.1
n 次多項式関数 $p(x)$ の値 $p(\alpha)$ を求めるのに必要な計算量を求めよ。またこの多項式関数の微係数 $p'(\alpha)$ を計算するためには，アルゴリズム 1 をどのように変更すればよいか？

6.2　Newton 法に基づく平方根の計算

ここでは正の引数 $a \in \mathbb{R}$ に対する平方根 $\sqrt{a}$ の計算方法を考える。平方根の計算方法は様々なものが提案されているが，最も頻繁に取上げられるのは Newton 法に基づく方法である。

2 次方程式 $x^2 - a = 0$ の正の解は $\sqrt{a}$ である。従って，この方程式の解を求めるアルゴリズムがあれば，それが $\sqrt{a}$ を求めるアルゴリズムとなる。

Newton 法について第 12 章で詳しく述べるが，この場合は次のような数列 $\{x_i\}_{i=0}^{\infty}$ を生成するアルゴリズムになる。このように同じ計算を繰り返し実行して近似値の精度を高めていく手法を総称して**反復法** (iteration method, iterative method) と呼ぶ。これを使用する際には停止則 (第 5 章参照) に注意を払う必要がある。

アルゴリズム 2 (Newton 法による平方根の計算)

1. 初期値 x_0 を決める。例えば $x_0 := a$ とする。
2. $i = 0, 1, \ldots$ に対して次の計算を行う。

$$x_{i+1} := \frac{1}{2}\left(x_i + \frac{a}{x_i}\right) \tag{6.2}$$

このアルゴリズムに基づいた $\sqrt{2}$ を計算した例を以下に示す。

例題 6.2.1 ($\sqrt{2} = 1.41421356237309504\cdots$ の計算)
アルゴリズム 2 を使って，初期値を $x_0 = 2$ としたもの (2 列目) と，$x_0 = (1+2)/2$ としたもの (3 列目) を，IEEE754 倍精度浮動小数点計算を用いて計算した結果が以下の表である。また相対誤差をプロットしたものを図 6.1 に示す。

x_i	$x_0 = 2$ の場合　　(相対誤差)	$x_0 = (1+2)/2$ の場合　(相対誤差)
x_0	2.00000000000000000(4.14×10^{-1})	1.50000000000000000(6.07×10^{-2})
x_1	1.50000000000000000(6.07×10^{-2})	1.41666666666666674(1.73×10^{-3})
x_2	1.41666666666666674(1.73×10^{-3})	1.41421568627450989(1.50×10^{-6})
x_3	1.41421568627450989(1.50×10^{-6})	1.41421356237468987(1.13×10^{-12})
x_4	1.41421356237468987(1.13×10^{-12})	

$x_0 = (1+2)/2$ とした方がより早く真値 $\sqrt{2}$ に接近していることが分かる。

問題 6.2.1
$a = 10, 100, 1000$ に対してアルゴリズム 2 を適用し，10 進 5 桁程度の近似解を求めよ。

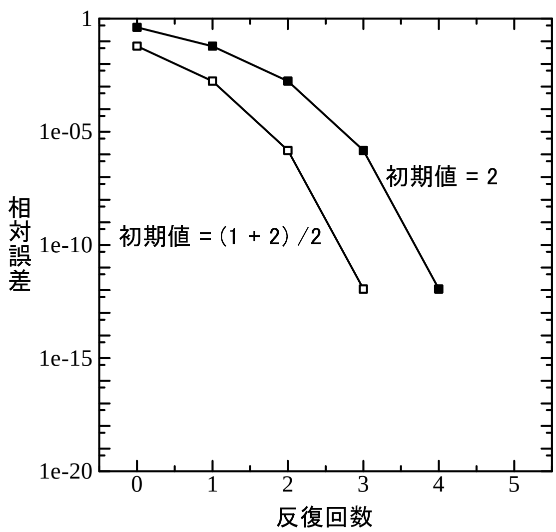

図 6.1: Newton 法による近似値の相対誤差

6.3 Taylor 展開に基づく初等関数の計算

まず微分積分の復習として Talyor の定理を示す。

定理 6.3.1 (Taylor の定理, Taylor 展開, Maclaurin 展開)
実関数 $f(x)$ が閉区間 $[a, b]$ で m 回連続微分可能かつ開区間 (a, b) で $m+1$ 回微分可能である時, $\forall x \in (a, b)$ に対して

$$f(x) = f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \cdots + \frac{f^{(m)}(a)}{m!}(x-a)^m + \frac{f^{(m+1)}(a+\theta(x-a))}{(m+1)!}(x-a)^{m+1} \quad (6.3)$$

を満足する $0 < \theta < 1$ が存在する。特に無限回微分可能であれば

$$f(x) = f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \cdots + \frac{f^{(m)}(a)}{m!}(x-a)^m + \cdots \quad (6.4)$$

と x に関する無限級数で表わすことができ，この式 (6.4) を関数 $f(x)$ の Taylor 展開 (Taylor expansion) と呼ぶ。また特に $a = 0$ の時を Maclaurin 展開と呼ぶ。

この定理は平均値の定理を繰り返し適用することで証明出来る。極めて綺麗な定理であり，応用も範囲も広い。最初の節で述べたように，多項式関数は Horner 法によって四則演算だけでその値を計算することが出来る。即ち，この定理の条件に当てはまる関数で，微係数 $f^{(i)}(a)$ が判明しているものであれば，その関数の近似値を (6.3) 式の多項式関数によって，四則演算の範囲で得ることが可能となる。そして，一般に初等関数と呼ばれる三角関数，指数関数，対数関数は全てこれらの条件に当てはまる上，微係数も容易に求めることが出来るのである。

但し，章の最初にも述べたように，Taylor 展開 (Maclaurin) 展開をそのまま適用して初等関数の値を計算するという手法は計算量の観点からあまり得策ではないことが知られている。実用に供

されている初等関数のアルゴリズムはこれとは別のもの (最良近似多項式，有理近似式，CORDIC 等) であることを付け加えておく。

例題 6.3.2 (代表的な初等関数の Maclaurin 展開)
一般に初等関数 (elementary functions) と呼ばれる，三角関数，指数関数，対数関数は $\mathbb{R}$ 全体もしくは特定の区間で無限回連続微分可能である。従って，Taylor 展開 (Maclaurin 展開) が存在する。

$$\exp(x) = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \cdots + \frac{x^n}{n!} + \cdots \quad (6.5)$$

$$\sin x = x - \frac{x^3}{3!} + \cdots + (-1)^{n-1}\frac{x^{(2n-1)}}{(2n-1)!} + \cdots \quad (6.6)$$

$$\cos x = 1 - \frac{x^2}{2!} + \cdots + (-1)^n\frac{x^{2n}}{(2n)!} + \cdots \quad (6.7)$$

$$\log(1+x) = x - \frac{x^2}{2} + \cdots + (-1)^n\frac{x^{n+1}}{n+1} + \cdots \ (\text{ここで } (1+x) > 0) \quad (6.8)$$

但し，これらの初等関数の Maclaurin 展開式を多項式関数として計算するには，引数 x に応じた配慮をする必要がある。以下，$\exp(x)$, $\sin(x)$, $\log(x)$ について具体的に計算方法を詰めていくことにする。

6.3.1　$e = \exp(1)$ の計算と誤差解析

丸め誤差が実数を有限桁の浮動小数点数で近似した結果生じた誤差であるのに対し，打ち切り誤差は無限級数や極限値のような無限回の演算を必要とする解析表現を，有限回の演算で打ち切る (truncate) ことによって生じる誤差である。丸め誤差は数値によって変動し精密な予測が難しいのに対し，打ち切り誤差は解析表現が明らかであれば，それに基づいて予測することが可能である。故に，打ち切り誤差は理論誤差とも呼ばれる。$e = \exp(1)$ の計算を例に，この打ち切り誤差について見ていくことにしよう。

e は指数関数 $\exp(x)(= e^x)$ の Maclaurin 展開式 (6.5) 式によって

$$e = 1 + \frac{1}{1!} + \frac{1}{2!} + \cdots + \frac{1}{n!} + \cdots$$

という無限級数の形で表現される。しかし，いかに高速なコンピュータといえども無限級数を計算することはできないため，どこかの項 $1/m!$ で計算を打ち切る必要がある。この項までの有限和を $\hat{e}_m$ と書くことにする。すなわち，

$$\hat{e}_m = 1 + \frac{1}{1!} + \frac{1}{2!} + \cdots + \frac{1}{m!}$$

である。この時，打ち切り誤差は

$$e - \hat{e}_m = \frac{1}{(m+1)!} + \frac{1}{(m+2)!} + \cdots$$

となる。

この形では評価が難しいので有限和の Maclaurin 展開公式

$$e = 1 + \frac{1}{1!} + \frac{1}{2!} + \cdots + \frac{1}{m!} + \frac{\exp(\theta)}{(m+1)!}$$

を用いることにする。ここで θ は $0 < \theta < 1$ となる定数である。

これを用いると打ち切り誤差は

$$e - \hat{e}_m = \frac{\exp(\theta)}{(m+1)!}$$

となる。右辺の絶対値を取れば

$$\left|\frac{\exp(\theta)}{(m+1)!}\right| \leq \frac{e}{(m+1)!}$$

となるので，相対打ち切り誤差を取ると

$$\left|\frac{e - \hat{e}_m}{e}\right| \leq \frac{1}{(m+1)!} \tag{6.9}$$

となり，m が決まれば打ち切り誤差の上限を評価することが可能となる。

一般に，打ち切り誤差は計算回数さえ増やせば減らすことができるが，使用する浮動小数点数の丸め誤差の最小単位より過度に小さくしても，コンピュータ資源の無駄遣いにしかならない。実際，IEEE754 倍精度計算を行い，項数を増やしつつ計算してその相対誤差をプロットしてみると図 6.2 のようになり，20 項以上取ってもマシンイプシロン ε_M 以上の精度を得ることは出来ないことが分かる。

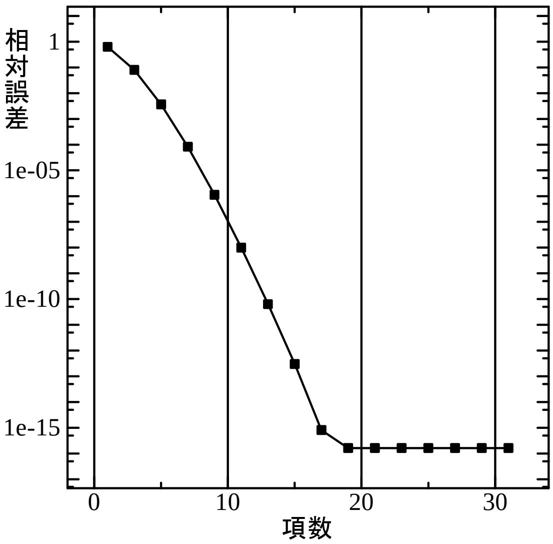

図 6.2: Maclaurin 展開に基づく exp(1) の近似値の相対誤差

例えば，10 進 7 桁の浮動小数点数を用いて e を計算するのであれば，先の評価式 (6.9) を用いて

$$\left|\frac{e - \hat{e}_m}{e}\right| \leq \frac{1}{(m+1)!} \approx \frac{1}{2} \cdot 10^{-6}$$

程度になる m まで計算するのが最適と言える。この場合，

$$\frac{1}{(8+1)!} \approx 2.8 \times 10^{-6},\ \frac{1}{(9+1)!} \approx 2.8 \times 10^{-7}$$

であるから，$m = 9$，すなわち

$$\hat{e}_9 = 1 + \frac{1}{1!} + \frac{1}{2!} + \cdots + \frac{1}{9!}$$

程度まで計算しておけば十分である。実際に計算してみると

$$\hat{e}_9 = \underline{2.7182815}2557\cdots$$

であり，下線部の 7 桁分が真値と一致していることが分かる。

問題 6.3.1

e を 10 進 15 桁の精度を得るために必要な項数 m を求め，実際に $\hat{e}_m$ を計算せよ。

6.3.2　exp(x) の計算

exp(x) を Maclaurin 展開級数 (6.5) に基づいて計算するには次の 2 点を勘案しなくてはならない。

$x < 0$ の場合　引数 x が負の場合，$\exp(x)$ の絶対値は小さくなる。特に $|x|$ が大きくなると，Maclaurin 展開式の項の絶対値 $|x^i/i!|$ も大きくなる。しかしそれらの和を取った結果の絶対値は小さくなるのだから，これは和の計算において $\max_i |x^i/i!|$ と $|\exp(x)|$ の order の差だけ「桁落ち」が起きることを示している。従って，引数が負の時は，$\exp(x) = 1/\exp(-x)$ という関係を使い，正の引数 $\exp(-x)$ を計算し，しかる後にその逆数を取る，という手順を取る必要がある。

$x >> 1$ の場合　理論上，この無限級数は任意の $x \in \mathbb{R}$ について収束することになっているが，数値計算上は $x >> 1$ の場合収束が遅くなり，必要な項数が増えてしまう。即ち $x^n/n! << 1$ となる $n \in \mathbb{N}$ が大きくなってしまうことになる。従って，無限級数の計算を行う x の範囲を，例えば $0 < x - \lfloor x \rfloor < 1$ に限定し，それを越える分については別途 $\exp(\lfloor x \rfloor)$ を計算して掛け合わせるようにすればよい。

以上をまとめると，次のようなアルゴリズムとなる。

アルゴリズム 3 (exp(x) の計算)

(a) $0 \le x \le 1$ の時には Maclaurin 展開級数 (6.5) を使用する。但し $x = 0$ の時は 1 を，$x = 1$ の時は $2.718281\cdots$ を返すようあらかじめ定数を設定しておく。

(b) $x > 1$ の時は，$x' := x - \lfloor x \rfloor$ として，$\exp(x')\exp(\lfloor x \rfloor)$ を計算する。当然 $\exp(\lfloor x \rfloor)$ の部分は定数 $2.71828\cdots$ の $\lfloor x \rfloor$ 乗として計算する。

(c) $x < 0$ の時は $\exp(|x|)$ を (a) 及び (b) を用いて計算し，その逆数 $1/\exp(|x|)$ を取る。

結局，実際に計算するのは $0 < x < 1$ の範囲における $\exp(x)$ の Maclaurin 展開級数である。では，どの程度の項数を取れば「収束」し必要な精度を得られるのか。(6.5) の右辺を m 項で打ち切った有限和を

$$\widehat{\exp}_m(x) = \sum_{i=0}^{m} \frac{1}{i!} x^i$$

と書くことにすれば，e の計算同様，打ち切り誤差は

$$\exp(x) - \widehat{\exp}_m(x) = \frac{\exp(\theta x)}{(m+1)!} x^{m+1}$$

となるから，右辺は

$$\frac{\exp(\theta x)}{(m+1)!} x^{m+1} \le \frac{e \cdot \exp(x)}{(m+1)!} x^{m+1}$$

と抑えられるので，相対打ち切り誤差を取り

$$\left| \frac{\exp(x) - \widehat{\exp}_m(x)}{\exp(x)} \right| \le \frac{e \cdot x^{m+1}}{(m+1)!} \tag{6.10}$$

に基づき，$0 < x < 1$ が決まれば打ち切り誤差の上限を評価することが可能となる。

問題 6.3.2
10 進 15 桁の有効桁を持つよう exp(3.2) を計算せよ。(6.10) 式を用いて第何項まで計算すればよいかも予め評価し，実際に計算してその相対誤差を求めよ。

6.3.3 $\sin x$ の計算

正弦関数 $\sin x$ の計算も，なるべく小さい x を使って Maclaurin 展開式 (6.6) を計算できるように配慮する必要がある。そのため，$\sin x$ の性質を利用して次のように計算すると良い。

アルゴリズム 4 ($\sin(x)$ の計算)

(a) $0 \le x \le \pi/2$ の時には Maclaurin 展開級数 (6.6) を使用する。但し $x = 0$ の時は 0 を，$x = \pi/2$ の時は 1 を返すようあらかじめ定数を設定しておく。

(b) $\pi/2 < x \le \pi$ の時は，$x' := \pi - x$ として，$\sin x := \sin x'$ を計算する。

(c) $\pi < x \le 2\pi$ の時は，$x' := x - \pi$ として，$\sin x := -\sin x'$ を計算する。

(d) $x > 2\pi$ の時は $x' := x - 2\pi \cdot \lfloor x/2\pi \rfloor$ として，(a)〜(c) を用いて $\sin x := \sin x'$ を計算する。

(e) $x < 0$ の時は $\sin(-x)$ の値を (a)〜(d) を使って求め，$\sin x := -\sin(-x)$ とする。

但しこれでも不十分な部分がある。図 6.3 は $\sin x$ の値 (実線) と，Maclaurin 展開式 (6.6) を用いた近似値の相対誤差 (破線) をプロットしたものであるが，ちょうど $\sin x \approx 0$ となる部分で，相対誤差が著しく増大しているのが分かる。これは展開式の計算で桁落ちが発生していることを示している。従って，この部分の近似値を計算する際には特別の配慮をする必要がある。

問題 6.3.3

1. 三角関数 $\cos x$ を Maclaurin 展開を元に計算する場合，どのような配慮が必要か，$\sin x$ の例を元に述べよ。

2. $\sin 1.5$, $\cos 1.5$ を IEEE754 倍精度程度 (10 進 15〜16 桁) 求めるには最大何項まで足し込んでいく必要があるか？

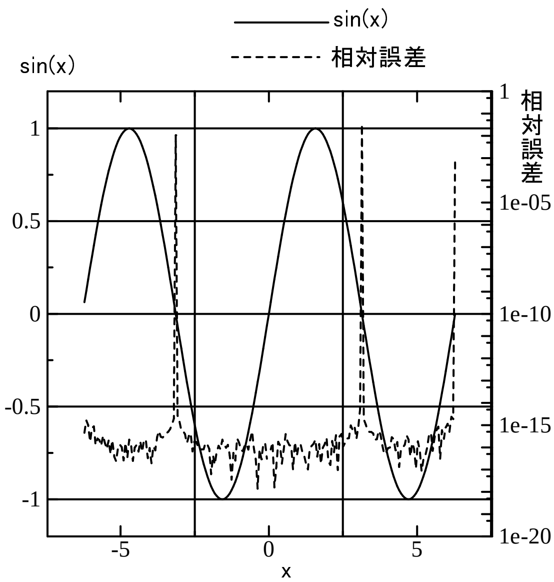

図 6.3: $\sin x$ と Maclaurin 展開に基づく $\sin x$ の相対誤差

6.3.4 $\log x$ の計算

$\log x$ の計算は，収束が遅い展開式を如何に改善させるか，という話題が出る所でよく取り上げられる。先に挙げた $\log(x+1)$ の Maclaurin 展開式 (6.8) の項を，$\exp(x)$ や $\sin x$ のそれを比べてみると，分母の階乗がない分，高次項の係数は $\log(1+x)$ の方がずっと大きい。そのため，収束するために必要となる項数も多くなる。また収束半径もごく限られた範囲に留まるため，(6.8) 式は実用性に乏しい。

そこで，収束を早める工夫が提案されている。高次項をぐっと小さくさせるために展開式を変更するのである。例えば

$$[\text{展開式 } 1]\ \frac{\log x}{2} = \left(\frac{x-1}{x+1}\right) + \frac{1}{3}\left(\frac{x-1}{x+1}\right)^3 + \cdots + \frac{1}{2n-1}\left(\frac{x-1}{x+1}\right)^{2n-1} + \cdots \tag{6.11}$$

$$[\text{展開式 } 2]\ \log x = \left(\frac{x-1}{x}\right) + \frac{1}{2}\left(\frac{x-1}{x}\right)^2 + \cdots + \frac{1}{n}\left(\frac{x-1}{x}\right)^n + \cdots \tag{6.12}$$

等が有名である。収束範囲は展開式 1 の方が広く，収束も早いことは一目瞭然であろう。実際，これらの展開式を使って $\log(2)$ の近似値を求めてみると，展開式 1 の方がずっと早く収束していることが分かる (図 6.4)。

問題 6.3.4

1. $a \in \mathbb{R}, a > 0$ に対して，a^x を計算する方法を考えよ。
2. 展開式 1 が展開式 2 よりも収束が早い理由を説明せよ。
3. $\log 100$ を展開式 1 を使って IEEE754 倍精度程度 (10 進 15〜16 桁) 求めたい時，何項目まで足し込めばよいか？

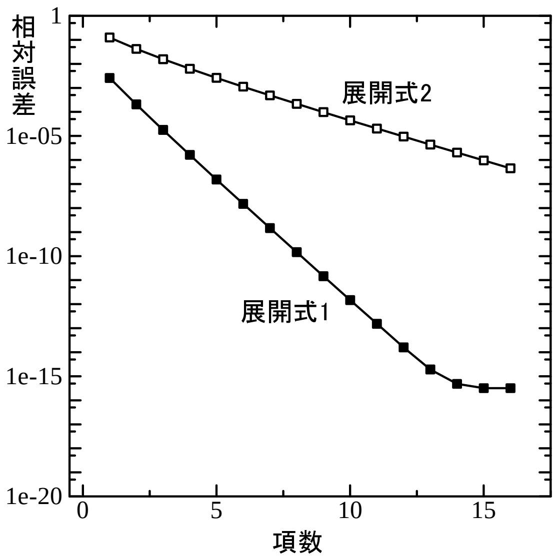

図 6.4: log(2) の近似値の相対誤差

6.4 その他の関数

以上で取り上げた初等関数を初めとして，実用上必要な関数は数多くある。ここでは C 言語の標準規格である C99[14] と，フリーのコンパイラである GCC に規定されている関数群 (表 6.1, 6.2) のうち，あまり馴染みのない関数を簡潔に紹介する。

6.4.1 誤差関数

誤差関数 $\mathrm{erf}(x)$ は，次の式で定義される。

$$\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp(-t^2)dt \tag{6.13}$$

これはちょうど，確率密度関数 $f(t)$ が

$$f(t) = \frac{1}{\sqrt{\pi}} \exp(-t^2)$$

であるような連続型確率変数 X における，確率 $P(-x \le X \le x)$

$$\begin{aligned} P(-x \le X \le x) &= \frac{1}{\sqrt{\pi}} \int_{-x}^x \exp(-t^2)dt \\ &= \frac{2}{\sqrt{\pi}} \int_0^x \exp(-t^2)dt \end{aligned}$$

と等しい。これは Gauss が天文観測値の誤差の精度を推定するために導入した関数で [44]，そのためこのように命名されている。

表 6.1: C99 に規定されている数学関数の一部

関数名	C99 における型指定	変数の範囲	数式表記
逆三角関数	`acos(x)`	$x \in [-1, 1]$	$\cos^{-1}(x)$
	`asin(x)`	$x \in [-1, 1]$	$\sin^{-1}(x)$
	`atan(x)`		$\tan^{-1}(x)$
三角関数	`cos(x)`		$\cos(x)$
	`sin(x)`		$\sin(x)$
	`tan(x)`		$\tan(x)$
逆双曲線関数	`acosh(x)`	$x \in [1, \infty)$	$\cosh^{-1}(x)$
	`asinh(x)`		$\sinh^{-1}(x)$
	`atanh(x)`	$x \in [-1, 1]$	$\tanh^{-1}(x)$
双曲線関数	`cosh(x)`		$\cosh(x)$
	`sinh(x)`		$\sinh(x)$
	`tanh(x)`		$\tanh(x)$
指数関数	`exp(x)`		$\exp(x) = e^x$
	`exp2(x)`		2^x
対数関数	`log(x)`	$x \in (0, \infty)$	$\log x = \log_e x = \ln x$
	`log10(x)`	$x \in (0, \infty)$	$\log_{10} x = \lg x$
	`log2(x)`	$x \in (0, \infty)$	$\log_2 x$
平行根	`sqrt(x)`	$x \in [0, \infty)$	$\sqrt{x}$
立方根	`cbrt(x)`		$\sqrt[3]{x} = x^{1/3}$
べき乗	`pow(x, y)`		x^y
誤差関数	`erf(x)`		$\mathrm{erf}(x)$
	`erfc(x)`		$1 - \mathrm{erf}(x)$
ガンマ関数	`tgamma(x)`		$\Gamma(x)$
対数ガンマ関数	`lgamma(x)`		$\log \lvert\Gamma(x)\rvert$

表 6.2: GCC 3.2 で規定されている数学関数

関数名	型指定	数式表記
第一種 Bessel 関数	`j0(x)`	$J_0(x)$
	`j1(x)`	$J_1(x)$
	`jn(int n, x)`	$J_n(x)$
第二種 Bessel 関数	`y0(x)`	$Y_0(x)$
	`y1(x)`	$Y_1(x)$
	`yn(int n, x)`	$Y_n(x)$

誤差関数を無限級数で表現すると

$$\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \sum_{i=0}^{\infty} \frac{(-1)^i x^{2i+1}}{i!(2i+1)}$$

となることが知られている [18]。

6.4.2　ガンマ関数

ガンマ関数 $\Gamma(x)$ は，次の式で定義される。

$$\Gamma(x) = \int_0^{\infty} t^{x-1} \exp(-t) dt \tag{6.14}$$

統計学では，この関数によって確率密度関数が規定されるガンマ分布を取り扱う。また，これを用いて定義されるベータ関数 $B(x,y)$

$$B(x,y) = \frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)}$$

も，統計ではよく利用される (ベータ分布)。

ガンマ関数は，対数ガンマ関数 $\log\Gamma(x)$ を，次の漸近展開 ($x \to \infty$ の時収束する無限級数)

$$\log\Gamma(x) \sim (x - \frac{1}{2}) \log x - x + \frac{1}{2} \log(2\pi) + \sum_{i=1}^{\infty} \frac{B_{2i}}{2i(2i-1)x^{2i-1}}$$

を用いて計算し，$\Gamma(x) = \exp(\log\Gamma(x))$ として求める [31]。x は次の漸化式

$$\Gamma(x+1) = x\Gamma(x)$$

を用いて，なるべく大きくしてから漸近展開の計算を行う。ここで B_i は Bernoulli 数で，展開式

$$\frac{t}{\exp(t) - 1} = \sum_{i=0}^{\infty} B_i \frac{t^i}{i!} \tag{6.15}$$

に基づいて定義される有理数[1]である。

6.4.3　Bessel 関数

Bessel の常微分方程式

$$x^2 \frac{d^2y}{dx^2} + x\frac{dy}{dx} + (x^2 - n^2)y = 0 \ (n \in \mathbb{Z})$$

の解として定義されるのが，Bessel 関数である。この解は複数あり，

$$\lim_{n\to\infty} J_n(x) = 0$$

[1] $B_0 = 1, B_1 = -1/2, B_2 = 1/6, B_3 = 0, \ldots$

となる解 $J_n(x)$ を，単なる Bessel 関数，あるいは第一種 Bessel 関数と呼ぶ。また

$$\lim_{n\to\infty} Y_n(x) = \infty$$

となる解 $Y_n(x)$ を，第二種 Bessel 関数，あるいは Weber 関数と呼ぶ。

第一種，第二種 Bessel 関数はどちらも

$$\begin{aligned} J_{n+1}(x) &= \frac{2n}{x} J_n(x) - J_{n-1} \\ Y_{n+1}(x) &= \frac{2n}{x} Y_n(x) - Y_{n-1} \end{aligned}$$

という漸化式が成立し，これに基づいて計算を行う。但し，$J_n(x)$ は n が大きい方から小さい方へ，$Y_n(x)$ は逆に n が小さい方から大きい方へと計算しなければならないことが知られている [28]。

演習問題

1. Newton 法を用いて，$a^{1/3}$ を求めるための漸化式は次のようになる。

$$x_{i+1} := \frac{1}{3}\left(2x_i + \frac{a}{{x_i}^2}\right)$$

これを使って $3^{1/3} = 1.442249570307408...$ を求めよ。

2. ユーザが x の値を入力した時，$\cos x$ の値を求めるプログラムを作りたい。この時，次の問に答えよ。なお π は円周率とする。

(a) $\cos x$ の Maclaurin 展開は

$$\cos x = \sum_{i=0}^{\infty} (-1)^i \frac{x^{2i}}{(2i)!}$$

である。$0 < x < \pi/4$ である時，$\cos x$ の値に含まれる打ち切り誤差が 10^{-5} 以下になるようにするには，第何項まで計算する必要があるか？

(b) $\alpha = \pi/7$ を，10 進 5 桁の浮動小数点数 $\widetilde{\alpha}$ に丸めた。近似値 $\widetilde{\alpha}$ を書け。またその近似値に含まれる絶対誤差と相対誤差もあわせて求めよ。

(c) $\cos \widetilde{\alpha}$ を求めよ。

3. $\sinh x, \cosh x, \tanh x$ を計算するプログラムを作りたい。

(a) $\sinh x = (\exp(x) - \exp(-x))/2$, $\cosh x = (\exp(x) + \exp(-x))/2$, $\tanh x = \sinh x / \cosh x$ という関係式に基づいて計算するプログラムを作れ。

(b) 展開式

$$\begin{aligned} \sinh x &= x + \frac{x^3}{3!} + \frac{x^5}{5!} + \cdots + \frac{x^{2n+1}}{(2n+1)!} + \cdots \\ \cosh x &= 1 + \frac{x^2}{2!} + \frac{x^4}{4!} + \cdots + \frac{x^{2n}}{(2n)!} + \cdots \end{aligned}$$

に基づいてプログラムを作れ。

4. 逆三角関数 $\sin^{-1} x, \cos^{-1} x, \tan^{-1} x$ を計算するプログラムを作れ。
5. 円周率 π を計算するプログラムを作れ。(Hint: $\pi/4 = \tan^{-1} 1$ である。) また，多倍長計算可能なソフトウェア (Mathematica 等) を使って，円周率 π を 1000 桁，10000 桁, 100000 桁 ··· 計算した際に要した時間を計測し，その増加率を求めよ。また，それだけの桁数を求めるために必要となる項数はどの程度か？
6. erfc(x) は，次のように定義されることを示せ。

$$\mathrm{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty \exp(-t^2) dt$$

7. 漸化式 $\Gamma(x+1) = x\Gamma(x)$ が成立することを示せ。また，n が正の整数であるとき，$\Gamma(n+1) = n!$ となることも示せ。
8. 漸化式に基づいて $n = 20$ までの $J_n(x)$ と $Y_n(x)$ を求めよ。

参考図書

初等関数は殆ど全ての数値計算で必要となるものだけに，その研究の歴史は長い。古い文献は沢山あるが，ある程度，理論的な内容をまとめたものとしては

初等関数の数値計算
一松信
教育出版
1974 年

が優れている。また，長年この研究に携わってきた著者が著した

近似式のプログラミング
浜田穂積
培風館
1995 年

も，各種近似式の係数が掲載されており役に立つ。また，

Software Manual for the Elementary Functions
W.J.Cody and Jr., W. Waite
Prentice-Hall
1980 年

は，各初等関数のテスト Fotran77 プログラムも掲載されていて，有用である。

なお，初等関数も含め，応用上よく利用される特殊関数についての性質，数表をまとめた

Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables
M. Abramowitz and I.A.Stegun 編
Dover
1965 年

は，大判で太い本であるが，手元に一冊あると便利である。初版は古いが，着実に版を重ねているので，入手は容易い。