C 言語の学習 数学関数

山本昌志*

2006 年 6 月 20 日

概 要

数学関数の取り扱い方法を学ぶ．はじめは，実数型の関数の取り扱いについて，説明する．それに慣れたならば，複素数型の取り扱い方法を学習する．

1 本日の学習内容

本日の内容は，教科書 [1] の 17 章の実数型の数学関数 (p.351–) と付録にある複素数型 (p.471) の取り扱いについて学習する．ただし，教科書には，複素数型の関数についての説明がないので，このプリントで補わなくてはならない．技術者にとって，複素数の取り扱いは，非常に重要である．

本日の学習のゴールは，以下の通りである．

- 実数型の数学関数の使い方が分かる．ヘッダーファイルの書き方と関数の使い方，マクロ定数，コンパイル方法を理解する必要がる．
- 複素数型の数学関数の使い方が分かる．ヘッダーファイルの書き方と複素数の変数宣言，複素数の表し方，複素関数の使い方，コンパイル方法を理解する必要がある．

2 実数の関数

2.1 数学関数の例

C 言語では，ヘッダーファイル math.h をインクルードすることにより，おなじみの数学の初等関数を使うことができる．具体的には，リスト 1 のようにする．

リスト 1: 実数型の関数の使用例

```
#include <stdio.h>
#include <math.h>

int main(void){
  double x, s, c, t, l, e;

  x=M_PI;
```

*独立行政法人　秋田工業高等専門学校　電気工学科

```
8
9     s=sin(x);
10    c=cos(x);
11    e=exp(x);
12    l=log(x);
13
14    printf("sin(pi)=%f\n",s);
15    printf("cos(pi)=%f\n",c);
16    printf("tan(pi)=%f\n",t);
17    printf("exp(pi)=%f\n",e);
18    printf("log(pi)=%f\n",l);
19
20    return 0;
21  }
```

実行結果

```
sin(pi)=0.000000
cos(pi)=-1.000000
tan(pi)=0.000000
exp(pi)=23.140693
log(pi)=1.144730
```

このプログラムの各行の内容は，次の通りである．

- 2 行目 `#include <math.h>`

  数学関数を使うために，ヘッダーファイル `math.h` をインクルードしている．数学関数を使う場合，必ず必要である．

- 7 行目 `x=M_PI;`

  `M_PI` は円周率を表すマクロである．`math.h` をインクルードすると，`M_PI` はコンパイル時に，$3.1415926\cdots$ に置き換わる．

- 9–12 行目 `s=sin(x);` など

  おなじみの数学関数が並んでいる．右辺の関数の戻り値を左辺の変数に代入している．

2.2　数学関数の使用方法

記述方法　数学関数を使うためには，`math.h` をインクルードすることを忘れてはならない．まずは，これを書く．

C 言語の数学関数は，数学で使う初等関数とほとんど同じ記述のため簡単である．必要な関数を数学で学習したように記述すればよい．`math.h` に用意されている関数は，表 1 のとおである．引数も戻り値も倍精度実数型である．

数学の計算でしばしば使われる定数は，`math.h` でマクロとして定義されている．定義されているマクロを表 2 に示す．いろいろ定義されているが，円周率を表す `M_PI` を憶えておけば，ほとんどの場合事足りる．次に重要なのは，ネピア数を表す `M_E` くらいである．

コンパイル方法　数学関数を含んだソースファイルをコンパイルする場合には，`libm` というライブラリーをリンクする必要がある．このライブラリーが数学関数の実体である．数学関数が使われているときには，

```
gcc -lm -o fugafuga hogehoge.c
```

のようにする．`hogehoge.c` がソースファイルで，`fugafuga` が実行ファイルである．オプション `-lm` をつけることにより，数学関数のライブラリー `libm` をリンクしている．

表 1: `math.h` で定義されている関数．関数の引数は倍精度実数である．戻り値も倍精度実数である．滅多に使わない関数—`fmod,ldexp,modf`—は省略．

数学関数名	C 言語関数	引数 x	戻り値
三角関数	`sin(x)`	単位はラジアン	$\sin x$ の値
	`cos(x)`	単位はラジアン	$\cos x$ の値
	`tan(x)`	単位はラジアン	$\tan x$ の値
逆三角関数	`asin(x)`	範囲 $[-1,+1]$	範囲 $[-\pi/2,+\pi/2]$ ラジアン
	`acos(x)`	範囲 $[-1,+1]$	範囲 $[0,\pi]$ ラジアン
	`atan(x)`		範囲 $[-\pi/2,+\pi/2]$ ラジアン
	`atan2(x,y)`		$\arctan(x/y)$ の値で範囲 $[-\pi,\pi]$ ラジアン
双曲線関数	`sinh(x)`		$\sinh x$ の値
	`cosh(x)`		$\cosh x$ の値
	`tanh(x)`		$\tanh x$ の値
指数関数	`exp(x)`		e^x の値
対数	`log(x)`	0≤x	自然対数 $\log_e x$ の値
	`log10(x)`	0≤x	常用対数 $\log_{10} x$ の値
絶対値	`fabs(x)`		$\|x\|$
平方根	`sqrt(x)`		$\sqrt{x}$
べき乗	`pow(x,y)`	x も y も実数可	x^y の値．複素数の場合エラー
整数部	`floor(x)`		x 以下の最大の整数値を `double` 型で返す
	`ceil(x)`		x 以上の最小の整数値を `double` 型で返す

表 2: `math.h` で定義されているマクロ定数.

数学定数名	数学記号	C 言語マクロ	値
円周率	π	`M_PI`	3.14159265358979323846
	$\pi/2$	`M_PI_2`	1.57079632679489661923
	$\pi/4$	`M_PI_4`	0.78539816339744830962
	$1/\pi$	`M_1_PI`	0.31830988618379067154
	$2/\pi$	`M_2_PI`	0.63661977236758134308
	$2/\sqrt{2}$	`M_2_SQRTPI`	1.12837916709551257390
ネピア数	e	`M_E`	2.7182818284590452354
	$\log_2 e$	`M_LOG2E`	1.4426950408889634074
	$\log_{10} e$	`M_LOG10E`	0.43429448190325182765
対数	$\log_e 2$	`M_LN2`	0.69314718055994530942
	$\log_e 10$	`M_LN10`	2.30258509299404568402
平方根	$\sqrt{2}$	`M_SQRT2`	1.41421356237309504880
	$1/\sqrt{2}$	`M_SQRT1_2`	0.70710678118654752440

2.3 練習問題

[**練習 1**] キーボードより変数の値を読み込み，以下の関数の値を表示せよ．

- 三角関数 $(\sin, \cos, \tan)$
- 指数関数
- 自然対数関数と常用対数関数
- 平方根と立方根

3　複素数と複素関数

3.1　複素関数を使った例

以前の C 言語は複素数がサポートされていなかった．数値計算をする場合，複素数が使えないとかなり不便を強いられる．そのため，複素数が使える FORTRAN から抜け出せない人が多くいた．新しい C 言語では，複素数がサポートされている．これは非常にありがたい．

実際に複素数や複素関数を使った例をリスト 2 に示す．これは，オイラーが発見した式

$$e^{i\pi} = -1 \tag{1}$$

の計算結果である．この式はとても不思議で，25 年くらい前にはじめてみたときには大変驚いた記憶がある．それまではなんの関係もないと思っていた円周率 π とネピア数 e と虚数単位 i が，こんなにも簡単な式で結ばれるのでる．

リスト 2: 複素数型の関数の使用例

```
1  #include <stdio.h>
2  #include <complex.h>
3  #include <math.h>
4
5  int main(void){
6    double _Complex z, x;
7
8    x=I*M_PI;
9    z=cexp(x);
10
11   printf("real=%f\timag=%f\n", creal(z),cimag(z));
12
13   return 0;
14 }
```

実行結果

```
real=-1.000000   imag=0.000000
```

このプログラムの各行の内容は，次の通りである．

- 2 行目 `#include <complex.h>`
  複素数おほび複素関数を使うために，ヘッダーファイル `complex.h` をインクルードしている．複素数を使う場合，必ず必要である．

- 6 行目 `double _Complex z, x;`
  倍精度複素数型の変数の宣言である．複素数型の変数 `z` と `x` が使えるようになる．

- 8 行目 `x=I*M_PI;`
  先に述べたように，`M_PI` は円周率を表すマクロである．`I` は虚数単位である．したがって，C 言語の `I*M_PI` は，数学の $i\pi$ を表す．

- 9 行目 `z=cexp(x);`

`cexp(x)`は，数学の e^x を表す．ただし，変数も関数の値も複素数となる．

- 11 行目 `creal(z),cimag(z)`

  `creal(z)`で複素数 `z` の実数部を，`cimag(z)`で虚数部を取り出している．

3.2 複素関数の使い方

3.2.1 記述方法

ヘッダーファイル ヘッダーファイル `complex.h` をインクルードする必要がある．プログラムの前の方に，

```
#include <complex.h>
```

と書く．

変数宣言 複素数の計算では，複素数型の変数宣言が必要である．変数宣言の例を，以下に示す．

```
float _Complex a, b, hoge;
double _Complex c, d. fuga;
long double _Complex e, f, foo;
```

のようにする．通常は，`double _Complex` を使うこと．C 言語で実数を扱う場合は `double`，複素数を扱う場合は `double _Complex` とするのが無難である．

複素数 虚数単位は `I` である．数学は小文字を使うが，C 言語では大文字である．複素数型の変数に値を代入するためには，次のようにする．

```
z=x+I*y;
w=3.1415+I*2.718281828;
```

四則演算 四則演算は特に気にすることもなく，普通に演算子 (`+,-,*,/`) が使える．

複素関数 表 3 のような関数が用意されている．よほどのことがない限り，倍精度型を使うこと．

表 3: `complex.h` で定義されている関数．関数の引数と戻り値は同じ型である．引数が倍精度複素数であれば，戻り値は倍精度複素数または倍精度実数である．

数学関数名	倍精度	単精度	拡張倍精度	戻り値
三角関数	csin(x)	csinf(x)	csinl(x)	複素数
	ccos(x)	ccosf(x)	ccosl(x)	複素数
	ctan(x)	ctanf(x)	ctanl(x)	複素数
逆三角関数	casin(x)	casinf(x)	casinl(x)	複素数
	cacos(x)	cacosf(x)	cacosl(x)	複素数
	catan(x)	catanf(x)	catanl(x)	複素数
双曲線関数	csinh(x)	csinhf(x)	csinhl(x)	複素数
	ccosh(x)	ccoshf(x)	ccoshl(x)	複素数
	ctanh(x)	ctanhf(x)	ctanhl(x)	複素数
逆双曲線関数	casinh(x)	casinhf(x)	casinhl(x)	複素数
	cacosh(x)	cacoshf(x)	cacoshl(x)	複素数
	catanh(x)	catanhf(x)	catanhl(x)	複素数
指数関数	cexp(x)	cexpf(x)	cexpl(x)	複素数
自然対数	clog(x)	clogf(x)	clogl(x)	複素数
絶対値	cabs(x)	cabsf(x)	cabsl(x)	実数
平方根	csqrt(x)	csqrtf(x)	csqrtl(x)	複素数
べき乗	cpow(x,y)	cpowf(x,y)	cpowl(x,y)	複素数 (x^y)
実部	creal(x)	crealf(x)	creall(x)	実数
虚部	cimag(x)	cimagf(x)	cimagl(x)	実数
偏角	carg(x)	cargf(x)	cargl(x)	実数
複素共役	conj(x)	conjf(x)	conjl(x)	複素数
リーマン球の射影	cproj(x)	cprojf(x)	cprojl(x)	複素数

3.2.2 コンパイル方法

実数型と同じように，オプション`-lm`をつける．

```
gcc -lm -o fugafuga hogehoge.c
```

3.3 練習問題

[**練習 1**] 複素数 $z_1 = i$ と $z_2 = 1 + i$ について，以下の値を計算せよ．

– 2 乗と 3 乗
– 平方根と立方根

– 絶対値
– 複素共役

[練習 2] オイラーの関係式 $e^{i\theta} = \cos\theta + i\sin\theta$ が成り立つことを，$0 \leqq \theta \leqq 2\pi$ の範囲で調べよ．この間を 360 等分して，両辺の値を出力する．

参考文献

[1] 林春比古. 新訂 C 言語入門 シニア編. ソフトバンク パブリッシング, 2004.