数値解析　(塩田)　2010 年 10 月 6 日

— 誤差のいろいろ —

1. 丸め誤差

全ての数値は「コンピュータで扱える数値」に近似される。

例 1　$e^{\pi\sqrt{163}}$ は、本当は無理数

262537412640768743.99999999999925...

なのだが、有効数字 29 桁までは整数値

262537412640768744.00000000000

しか出力されない。

2. オーバーフロー・アンダーフロー

大き過ぎる数は表現できずにエラーを起こす。小さ過ぎる数は 0 になってしまう。

例 2　1 を 10 で n 回割ってから、10 を n 回掛けると ...

```
#include<stdio.h>
main()
{
    int i, n;
    float x;
    for(n = 35; n <= 50; n++){
        x = 1.0;
        for(i = 1; i <= n; i++) x /= 10;
        for(i = 1; i <= n; i++) x *= 10;
        printf("n = %d のとき %f\n",n,x);
    }
}
```

```
n = 38 のとき 1.000000
n = 39 のとき 1.000000
n = 40 のとき 0.999995
n = 41 のとき 0.999967
n = 42 のとき 1.000527
n = 43 のとき 0.994922
n = 44 のとき 0.980909
n = 45 のとき 1.401299
n = 46 のとき 0.000000
n = 47 のとき 0.000000
```

3. 累積誤差

塵も積もれば山となる。

例 3　$\displaystyle\sum_{i=1}^{n} \sin\left(\frac{2\pi i}{n}\right)$ の理論値は 0 だが ...

```
#include<stdio.h>
#include<math.h>
#define pi 3.14159265358979324
main()
{
    int i,j,n;
    float x;
    for(j=1;j<=7;j++){
        n=(int)pow(10,j);
        x=0.0;
        for(i=1;i<=n;i++) x+=sin((2*pi*i)/n);
        printf("n = %8d のとき %9.6f\n",n,x);
    }
}
```

```
n =       10 のとき -0.000000
n =      100 のとき  0.000001
n =     1000 のとき -0.000015
n =    10000 のとき -0.000050
n =   100000 のとき -0.000020
n =  1000000 のとき -0.028050
n = 10000000 のとき  0.078272
```

例 4 $\displaystyle\sum_{n=1}^{m}\left\{\frac{1}{n}-\frac{1}{n+1}-\frac{1}{n(n+1)}\right\}$ の理論値は 0 だが ...

```
#include<stdio.h>
#include<math.h>
main()
{
    int n, j, m;
    float x = 0.0;
    for(j = 4; j <= 8; j++){
        m = (int)pow(10,j);
        for(n = 1; n <= m; n++)
            x += (1.0/n - 1.0/(n+1) - 1.0/(n*(n+1)));
        printf("%9d 項目までの和 %9.6f\n",m,x);
    }
}
```

```
    10000 項目までの和 -0.000000
   100000 項目までの和  0.000025
  1000000 項目までの和  0.009984
 10000000 項目までの和 -0.088248
100000000 項目までの和 -0.145553
```

4. 積み残し

$|a| \gg |b|$ のときに $a+b$ を計算すると b の下の方の桁が無視される。

例 5 $\dfrac{1}{1^2}+\dfrac{1}{2^2}+\dfrac{1}{3^2}+\cdots$ の理論値は $\dfrac{\pi^2}{6}=1.6449340668\cdots$ だが ...

```
#include<stdio.h>
#define pi 3.14159265358979324
main()
{
    int i,j,n;
    float x;
    printf("理論値は %f\n\n",pi*pi/6);
    printf("昇順に加えると\n",n);
    for(j = 2; j <= 7; j++){
        n = (int)pow(10,j);
        x = 0.0;
        for(i = 1; i <= n; i++) x += 1.0/i/i;
        printf("%8d 項目までの和 = %f\n",n,x);
    }
    printf("\n");
    printf("降順に加えると\n",n);
    for(j = 2; j <= 7; j++){
        n = (int)pow(10,j);
        x = 0.0;
        for(i = n; i >= 1; i--) x += 1.0/i/i;
        printf("%8d 項目までの和 = %f\n",n,x);
    }
}
```

```
理論値は 1.644934

昇順に加えると
     100 項目までの和 = 1.634984
    1000 項目までの和 = 1.643935
   10000 項目までの和 = 1.644725
  100000 項目までの和 = 1.644725
 1000000 項目までの和 = 1.644725
10000000 項目までの和 = 1.644725

降順に加えると
     100 項目までの和 = 1.634984
    1000 項目までの和 = 1.643934
   10000 項目までの和 = 1.644834
  100000 項目までの和 = 1.644924
 1000000 項目までの和 = 1.644933
10000000 項目までの和 = 1.644934
```

例 6　$\frac{1}{1}+\frac{1}{2}+\frac{1}{3}+\cdots=+\infty$ だが ...

```
#include<stdio.h>
#include<math.h>
main()
{
    int c = 0, i = 0;
    float x = 0.0, y = -1.0;
    while(x != y){
        i++; c++;
        y = x;
        x += 1.0/i;
        if(c == 100000){
            printf("1/1 + 1/2 + ... + 1/%7d = %9.6f\n",i,x);
            c = 0;
        }
    }
    printf("ここから和の値が増えなくなりました:\n");
    printf("1/1 + 1/2 + ... + 1/%7d = %9.6f\n",i,x);
}
```

```
1/1 + 1/2 + ... + 1/ 100000 = 12.090851
1/1 + 1/2 + ... + 1/ 200000 = 12.782757
1/1 + 1/2 + ... + 1/ 300000 = 13.195325
(中略)
1/1 + 1/2 + ... + 1/1800000 = 15.120297
1/1 + 1/2 + ... + 1/1900000 = 15.215665
1/1 + 1/2 + ... + 1/2000000 = 15.311032
ここから和の値が増えなくなりました:
1/1 + 1/2 + ... + 1/2097152 = 15.403683
```

5. 桁落ち

a　b のときに $a-b$ を計算すると著しく精度が落ちる。

例 7　2 次方程式 $ax^2+x+1=0$ の解は、公式どおり計算してはいけないときがある。

```
#include<stdio.h>
#include<math.h>
main()
{
    int j;
    float a,b=1.0,c=1.0,x,y,s;
    printf("\t\t\t 正しい値\t 公式どおり\n");
    for(j=1;j<=8;j++){
        a=pow(10.0,-j);
        s=sqrt(b*b-4*a*c);
        x=(-b+s)/(2.0*a);  // 公式どおり
        y=(2.0*c)/(-b-s);  // 工夫した式 (正しい値)
        printf("a = 10^(-%d) のとき \t%9.6f\t%9.6f\n",j,y,x);
    }
}
```

```
                        正しい値        公式どおり
a = 10^(-1) のとき      -1.127017       -1.127017
a = 10^(-2) のとき      -1.010205       -1.010206
a = 10^(-3) のとき      -1.001002       -1.001000
a = 10^(-4) のとき      -1.000100       -1.000166
a = 10^(-5) のとき      -1.000010       -1.001358
a = 10^(-6) のとき      -1.000001       -1.013279
a = 10^(-7) のとき      -1.000000       -0.894070
a = 10^(-8) のとき      -1.000000        0.000000
```