**数値解析法演習，平成29年4月24日レポート**

情報工学科3年　学生番号　u276156　　　　　名前　根本貴大

1. 理論

1.1 桁落ち

同符号で値の近い数の減算や異符号で絶対値の近い数の加算を行うときに，有効数字の桁数が急激に減少する．それにより相対誤差が極めて大きくなるため計算の精度が低下する．

1.2 情報落ち

　絶対値の差が大きな2数の加減算で，小さい値が無視される．

2. 課題

課題1．

次の式(1)を計算するプログラムを作成せよ。と小さくしたとき結果はどのようになるか。

　 (1)

次の式(2)についても同様に実験し、結果を比較せよ。

 (2)

注意：

出力するときには，次のように%e型書式指定子を使うこと．%fを使うと有効桁数が足らなくなる．

printf("result: %e\n", result);

課題１の結果

x=0.1としたときのプログラムは以下のようになる．

float result = 0, result2 = 0, x = 0.1;

result = (sqrt(1 + x) - 1) / (sqrt(1 + x));

result2 = x / ((1 + x) + sqrt(1 + x));

printf("x = %.2f のとき\n", x);

printf("result = %e\n", result);

printf("result = %e\n\n", result2);

1. result = 4.653742e-02
2. result2 = 4.653741e-02

x=0.01としたときの実行結果は以下のようになる．

x = 0.01;

result = (sqrt(1 + x) - 1) / (sqrt(1 + x));

result2 = x / ((1 + x) + sqrt(1 + x));

printf("x = %.2f のとき\n", x);

printf("result = %e\n", result);

printf("result = %e\n\n", result2);

1. result = 4.962805e-02
2. result2 = 4.962810e-02

演習環境：Visual Studio2015

考察

(1)と(2)の式は変形しただけで同じ計算結果を出す式である。しかし、計算結果はいずれの場合も違う結果を出力した。x = 0.1のときの誤差は少数以下6桁目で発現したが、x=0.01のときの誤差は少数以下5桁目で発現した。このことよりxの値が小さくなるほど誤差の割合が大きくなるといえる。

課題2．

次の式を計算するプログラムを作れ。



ただし、である。として計算した時、結果はと一致するか。

課題2の結果

プログラムの本体は以下のようになる

float result = 0,result2 = 0, a = 0.1;

for (i = 1; i <= n; i++)

result += a;

result2 = n \* a;

printf("result = %e\n",result);

printf("result2 = %e\n",result2);

　　(1)result = 2.097152e+06

(2)result2 = 1.000000e+07

演習環境：Visual Studio2015

考察

(1)と(2)の結果を確認すると誤差の割合が大きいことが分かる。(1)の答えは桁数までもが変動してしまっている。

　暗算で計算をおこなっても答えが同じことが分かるが、プログラムを実行すると結果は違ってしまっている。このことから加算を行う際はループ処理で行うだけでなく、ひと工夫をして乗算でおこなうことも大切だと感じた。