

=====

5 月 22 日（金）第 4 回数値解析 I 提出課題 19TM054 浅野 駿介

提出日：2020/06/02

=====

<作成プログラム>

#define _USE_MATH_DEFINES//円周率を表す M_PI を使えるようにする

#include<stdio.h>

#include<math.h>//数学関数を取り込む

double func(double x) {

 return(4.0 * sqrt(1.0 - x * x));//sqrt()は平方根を返す

}

void main() {

 double h = 0.0001, b = 1.0, a = 0.0, result = 0.0, t,r;

 long cycle = (b - a) / h;

 for (long i = 0; i < cycle; i++) {

 t = a + (double)i * h;

 /*第 4 回の講義内容の式を参照*/

 result = result + (func(t) + func(t + h) + 4.0 * func(t + h / 2.0)) * h / 6.0;

 }

 printf("h=%lf,result=%lf¥n", h, result);

 /*誤差の相対的な割合を求める*/

 r = fabs((result - M_PI) / M_PI);//fabs()は絶対値を求める

 printf("円周率に対する誤差の相対的な割合は%.10lf¥n", r);//小数点以下の表示桁数を増やす

}

<出力結果>

$h=0.1$ のとき

$h=0.100000$, result=3.136447

円周率に対する誤差の相対的な割合は 0.0016378920

$h=0.01$ のとき

$h=0.010000$, result=3.141430

円周率に対する誤差の相対的な割合は 0.0000516949

$h=0.001$ のとき

$h=0.001000$, result=3.141588

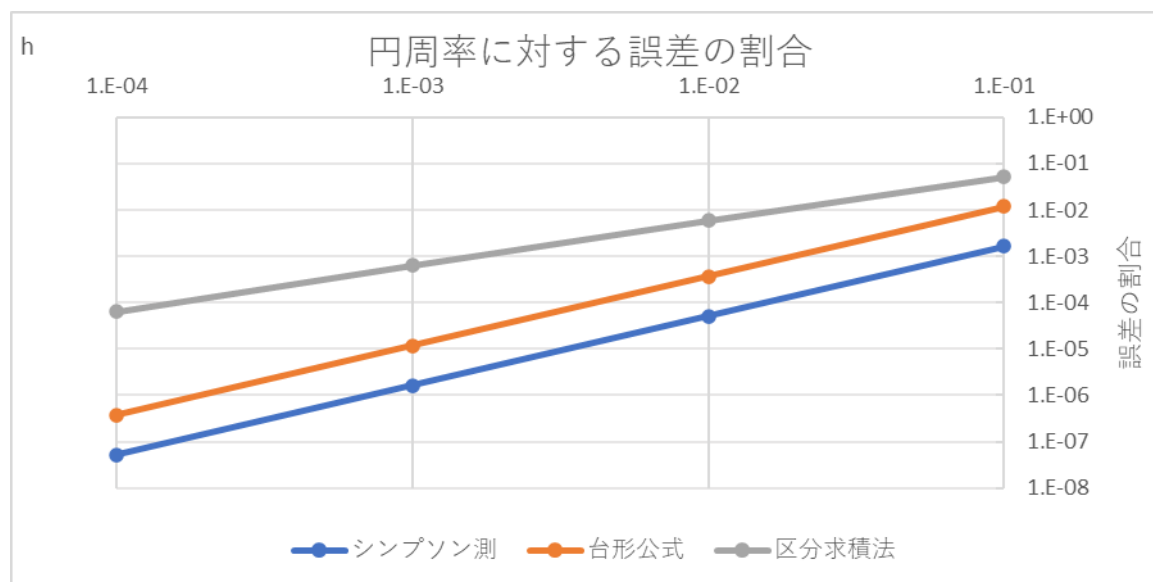
円周率に対する誤差の相対的な割合は 0.0000016344

$h=0.0001$ のとき

$h=0.000100$, result=3.141592

円周率に対する誤差の相対的な割合は 0.0000000517

円周率に対する誤差の割合を縦軸, h を横軸として両対数グラフを作成した. 台形公式と区分求積法の計算誤差の結果は課題 3 の結果を利用した.



<理解した内容, 感想, 注意点など>

- ・シンプソン則, 台形公式, 区分求積法の順に円周率に対する誤差の割合が大きくなった.
- ・どの方法でも共通して h が小さいほうが円周率に対する誤差の割合が大きくなった.