陽解法

　　計算法

xn+1 – xn/ δt = -λxn より

xn+1 = (1 – λδt )xn

　　ソース

#include <stdio.h>

int main(void)

{

int lambda = 5; // λ

double delta = 0.01; // δt

double x = 10; // x

int i; //counter

for (i=0; i<100; i++) {

printf("%f\n",x);

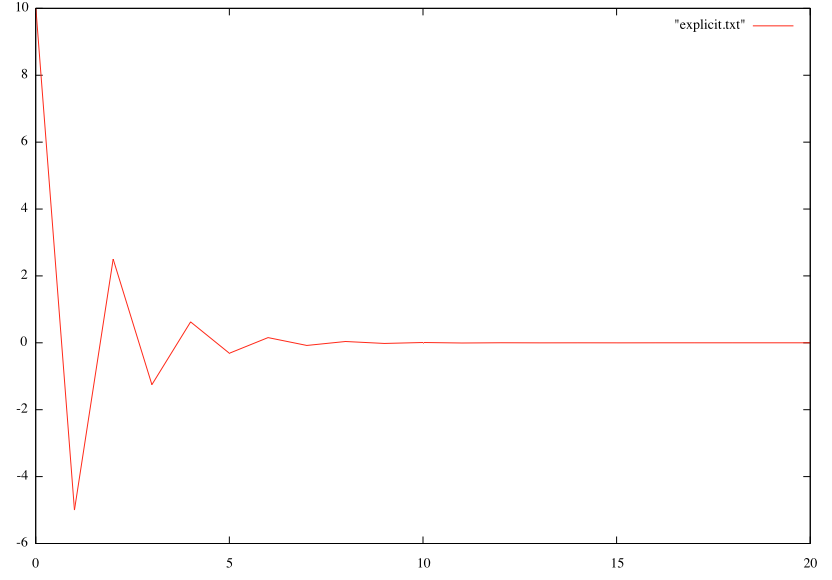
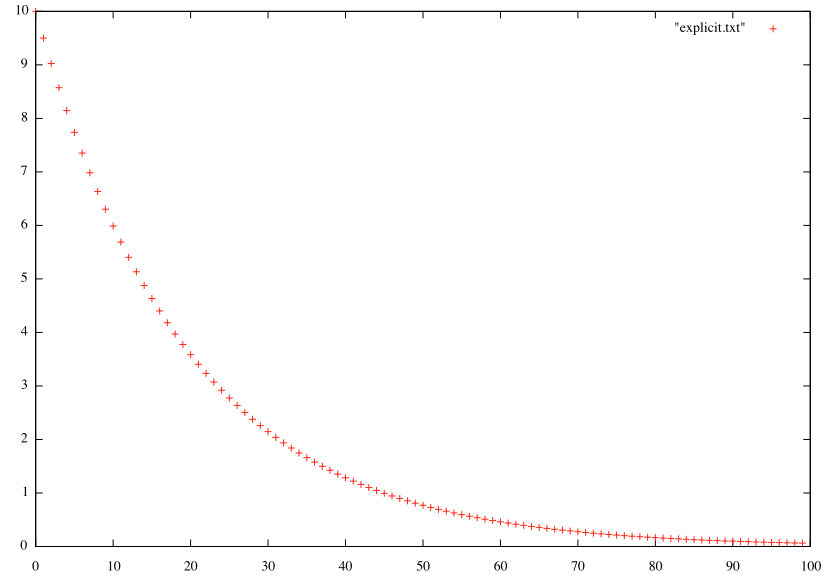
x \*= (1 - lambda \* delta);

}

return 0;

}

グラフ



　　δt = 0.01 δt = 0.3

陰解放

計算法

xn+1 – xn/ δt = -λxn+1 より

xn+1 = xn / (1 + λδt)

ソース

#include <stdio.h>

int main(void)

{

int lambda = 5; // λ

double delta = 0.01; // δt

double x = 10; // x

int i; //counter

for (i=0; i<100; i++) {

printf("%f\n",x);

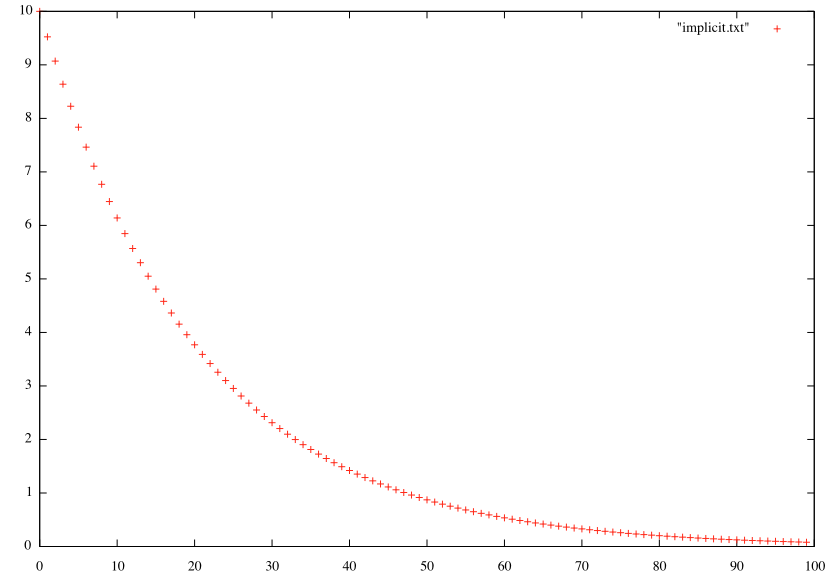
x /= (1 + lambda \* delta);

}

return 0;

}

グラフ



　　　　　　　　　　　　　　　　t = 0.01

中央差分

計算法

(xn+1 - xn-1) / 2δt = -λxnより

xn+1 = xn-1 - 2δt xn

ソース

#include <stdio.h>

int main(void)

{

int lambda = 5; // λ

double delta = 0.001; // δt

double x1 = 10;

double x2 = x1 \* (1 - lambda \* delta);

double x3 = x1 - 2 \* delta \* lambda \* x2;

double tmp;

int i; //counter

printf("%f\n",x1);

printf("%f\n",x2);

for (i=0; i<1000; i++) {

tmp = x3;

x3 = x1 - 2 \* delta \* lambda \* x2;

x1 = x2;

x2 = tmp;

printf("%f\n",x3);

}

return 0;

}

グラフ

