

2006年7月6日(木) 実施

前々回の追加

例題

$0 \leq x \leq 1$, $0 \leq y \leq 1$, $0 \leq z \leq 1$ の範囲で乱数を発生させて、回転放物面体の体積を求めらる。C言語のプログラムを翻訳編集した後、複数回実行して、その都度、体積の値が異なることを確認すること。

ex7-6.c

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

#define M 100000

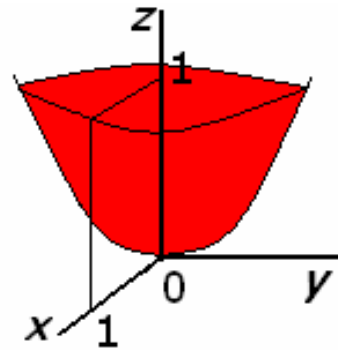
int main(void)
{
    int i, n=0;
    double x, y, z;
    srand((unsigned int)time(NULL));

    for (i=0; i<M; i++) {
        x=(double)rand()/RAND_MAX;
        y=(double)rand()/RAND_MAX;
        z=(double)rand()/RAND_MAX;

        if (z>=x*x+y*y) {
            n++;
        }
    }

    printf("V=%f", 4.0*n/M);

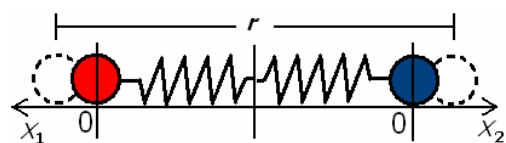
    return 0;
}
```



復元力による運動(第3回)の応用

例題

ばね定数 k のばねを中間で固定し、両端についた質量の異なる重りを、摩擦のない水平面上で自由に運動させることを考える。



自然科学シミュレーション ノート

ex9-1.c

```
#include <stdio.h>

#define M 0.1
#define K 0.2
#define TMAX 10.0

double fx(double);
double fv(double);
double rungex(double, double);
double rungev(double, double);
int i;
double dt=0.01;

int main(void)
{
    double r, x1, v1, x2, v2, t;
    FILE *output;

    x1=0.1;
    v1=0.0;
    x2=0.1;
    v2=0.0;

    output=fopen("spring2.data", "w");

    r=1.0+x1+x2;

    fprintf(output, "%f %f\n", t, r);

    for (t=0.0; t<TMAX; t+=dt)
    {
        i=1;
        x1=rungex(x1, v1);
        v1=rungev(x1, v1);
        i=2;
        x2=rungex(x2, v2);
        v2=rungev(x2, v2);
        r=1.0+x1+x2;
        fprintf(output, "%f %f\n", t+dt, r);
    }

    fclose(output);

    return 0;
}
```

自然科学シミュレーション ノート

```
double fx(double v)
{
    return v;
}

double fv(double x)
{
    if (i==1)
        return -K/M*x;
    else
        return -2*K/M*x;
}

double rungev(double x, double v)
{
    double k0x, k1x, k2x, k3x, k0v, k1v, k2v, k3v;

    k0x=dt*fx(v);
    k0v=dt*fv(x);
    k1x=dt*fx(v+k0v/2.0);
    k1v=dt*fv(x+k0x/2.0);
    k2x=dt*fx(v+k1v/2.0);
    k2v=dt*fv(x+k1x/2.0);
    k3x=dt*fx(v+k2v);
    k3v=dt*fv(x+k2x);
    x=x+(k0x+2.0*k1x+2.0*k2x+k3x)/6.0;

    return x;
}

double rungev(double x, double v)
{
    double k0x, k1x, k2x, k3x, k0v, k1v, k2v, k3v;

    k0x=dt*fx(v);
    k0v=dt*fv(x);
    k1x=dt*fx(v+k0v/2.0);
    k1v=dt*fv(x+k0x/2.0);
    k2x=dt*fx(v+k1v/2.0);
    k2v=dt*fv(x+k1x/2.0);
    k3x=dt*fx(v+k2v);
    k3v=dt*fv(x+k2x);
    v=v+(k0v+2.0*k1v+2.0*k2v+k3v)/6.0;

    return v;
}
```