TI DSP, MCU 및 Xilinx Zynq FPGA 프로그래밍 전문가 과정

강사 – Innova Lee(이상훈)
gcccompil3r@gmail.com
학생 – GJ (박현우)
uc820@naver.com

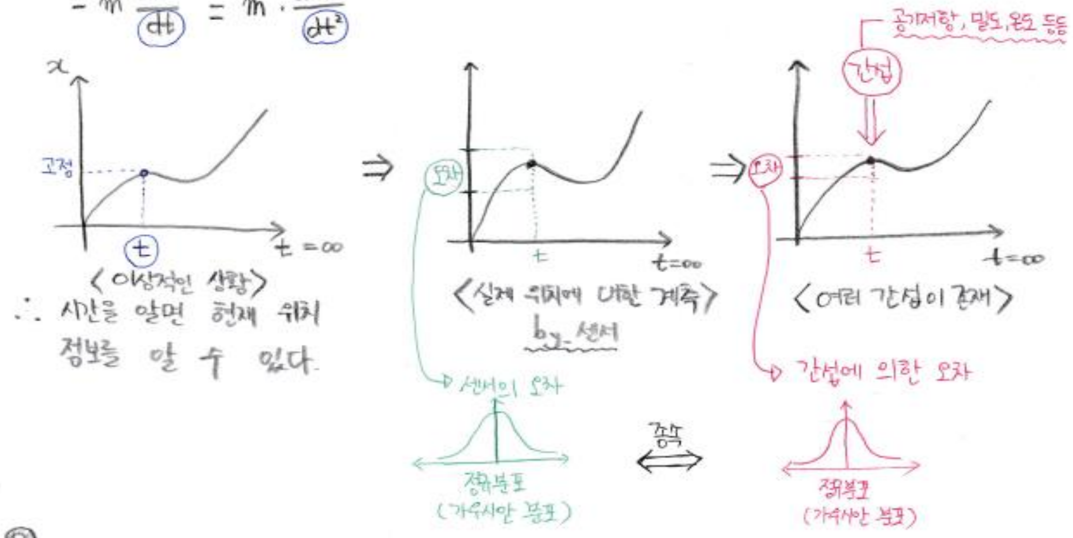
목차

6. 수학

1) 뉴턴의 제 2법칙과 물리모델링이 필요한 이유

$\vec{F} = m\vec{a}$ (뉴턴의 제 2법칙)

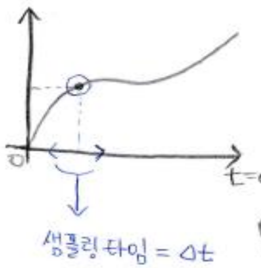
$= m\dfrac{d\vec{v}}{dt} = m \cdot \dfrac{d^2\vec{x}}{dt^2}$



∴ 시간을 알면 현재 위치 정보를 알 수 있다.

? 센서의 오차와 간섭의 오차를 어떻게 줄여야 할까?

1. 보다 정확한 예측을 위한 물리모델링 – 미분 방정식 필수조건.
2. 서로 종속 (독립사건이 아닌)인 정규분포 합성 – 랜덤프로세스
3. *칼만 필터 + P.I.D 제어



1 ~ 3번과 $\left\langle \dfrac{f(x+\Delta t) - f(x)}{\Delta t} \right\rangle$를 이용하여 프로그래밍을 해서 보다 정확한

$\boxed{\text{예측}}$ 값을 얻어낼 수 있다

2) 변수 분리형 미분방정식 / 완전 미분형 미분방정식 / 연쇄법칙

∘ 변수 분리형 미분방정식

$f(x) = g(y)\frac{dy}{dx}$

$\Rightarrow f(x)\,dx = g(y)\,dy$

$\int f(x)\,dx = \int g(y)\,dy + C$

ex) $y' = \frac{x-3}{2+y^2}$, $y(0) = -1$

1) $\int (2+y^2)\,dy = \int (x-3)\,dx$

2) $2y + \frac{1}{3}y^3 = \frac{1}{2}x^2 - 3x + C$

3) $-2 - \frac{1}{3} = C$

$\therefore \underline{\frac{1}{3}y^3 + 2y = \frac{1}{2}x^2 - 3x - \frac{7}{3}}$

☆ 감마함수 → 정규분포 (칼만필터)

$y' = -2xy$, $y(0) = 3$

$\frac{dy}{dx} = -2xy \Leftrightarrow \int -2x\,dx = \int \frac{1}{y}\,dy$

1) $-x^2 + c = \ln y$

2) $e^{-x^2+c} = y$

3) $e^c \cdot e^{-x^2} = y$

상수

$\therefore y = c \cdot e^{-x^2}$

$= 3 \cdot e^{-x^2}$ 감마함수

∘ 완전 미분형 미분방정식

$$\boxed{\frac{P(x,y)}{dy} = \frac{Q(x,y)}{dx}}$$

∨ $u(x,y)$의 전미분 du , $y = y(x)$

∨ $du = P(x,y)\,dx + Q(x,y)\,dy$

∨ $du = 0$ 적분시 $u(x,y) = C$

∨ 조건

$$\frac{\partial P(x,y)}{\partial y} = \frac{\partial Q(x,y)}{\partial x}$$

$$\frac{du(x,y)}{dx} = \frac{du}{dx} + \frac{du}{\partial y}\frac{\partial y}{dx}$$

∘ 연쇄법칙

$\lim_{h\to 0} \frac{f(x+h) - f(x)}{x+h-x}$, $f(g(x))$ 일 때,

$$\frac{f(g(x+h)) - f(g(x))}{g(x+h) - g(x)} \cdot \frac{g(x+h) - g(x)}{x+h-x}$$

$$\therefore \frac{dy}{du} \cdot \frac{du}{dx}$$

3) 완전 미분형 미분방정식

∘ 완전 미분형 미분방정식

$P(x,y)\,dx + Q(x,y)\,dy = 0$ 을 만족하는 $u(x,y)$를 가정, $\dfrac{\partial P(x,y)}{\partial y} = \dfrac{\partial Q(x,y)}{\partial x}$

$$\frac{du(x,y)}{dx} = \frac{\partial u}{\partial x} + \frac{\partial u}{\partial y}\frac{dy}{dx} = P(x,y) + Q(x,y)\frac{dy}{dx} = 0$$

1) $\dfrac{\partial u}{\partial x} = P(x,y) \Rightarrow u(x,y) = \int P(x,y)\,dx + g(y)$

x에 대한 적분이지만 y항이 있으므로 적분상수는 y에 대한 함수가 됨.

2) $\dfrac{\partial u}{\partial y} = Q(x,y) \Rightarrow \dfrac{\partial}{\partial y}\int P(x,y)\,dx + g'(y)$

$$\therefore g'(y) = Q(x,y) - \frac{\partial}{\partial y}\int P(x,y)\,dx$$

ex) $2xy\,dx + (x^2-1)\,dy = 0$

$P(x,y) = 2xy$, $Q(x,y) = x^2-1$

$\dfrac{\partial P(x,y)}{\partial y} = \dfrac{\partial Q(x,y)}{\partial x} \Leftrightarrow 2x = 2x$ (조건성립)

$$g'(y) = (x^2-1) - \frac{\partial}{\partial y}\int 2xy\,dx$$

$$g(y) = -y + c$$

$$\therefore u(x,y) = \int 2xy\,dx - y + c$$

$$= x^2y - y + c$$

$$= (x^2-1)y + c$$

4) 1계 선형 미분방정식

- 일계 선형 미분방정식

$$p(x)y' + q(x)y = h(x)$$

표준형 : $y' + \sim$ 형태로 만들면 됨

$$y' + \frac{q(x)}{p(x)}y = \frac{h(x)}{p(x)}$$ $= 0$ 이면 동차

적분인자 $\mu(x) = e^{\int \frac{q(x)}{p(x)}dx}$, $\frac{q(x)}{p(x)} = k(x)$

$$\overset{D}{\frac{dy}{dx}} \cdot \underbrace{e^{\int k(x)dx}}_{ND} + \overset{ND}{y}\, \underbrace{k(x) \cdot e^{\int k(x)dx}}_{D} = \frac{h(x)}{p(x)} \cdot e^{\int k(x)dx}$$

$$\Downarrow$$

$$\frac{d}{dx}\left(y e^{\int k(x)dx}\right)$$

$$\int \frac{d}{dx}\left(y \cdot e^{\int k(x)dx}\right) = \int \frac{h(x)}{p(x)} \cdot e^{\int k(x)dx}$$

$$y \cdot e^{\int k(x)dx} = \int \frac{h(x)}{p(x)} \cdot e^{\int k(x)dx}$$

$$\therefore\ y = e^{-\int k(x)dx} \cdot \int \frac{h(x)}{p(x)} \cdot e^{\int k(x)dx}$$

ex) $y' + 3y = 6$

1) $k(x) = 3$, $\frac{h(x)}{p(x)} = 6$

2) $\mu(x) = e^{\int 3dx} = e^{3x}$

3) $\frac{d}{dx}(y \cdot e^{3x}) = 6 \cdot e^{3x}$

4) $y \cdot e^{3x} = \int 6 \cdot e^{3x} dx$

5) $y = (2e^{3x} + c) \cdot e^{-3x}$

$\therefore\ y = c \cdot e^{-3x} + 2$

5) 2계 선형 미분방정식

○ 이계 미분 방정식

$y'' + p(x)y' + h(x)y = 0$ ($\because$ y_1을 알고 있음)

찾고자 하는 것: y_2 (y_1이 불안정 함수라 y_2 것도 필요!)

우리가 찾는 y_2가 y_1과 관계가 있다고 가정!

$$y_2 = y_1 u$$

$$y_2' = y_1' u + y_1 u'$$

$$y_2'' = y_1'' u + \underbrace{y_1' u' + y_1' u'}_{2u'y_1'} + y_1 u''$$

$$y_1'' u + 2y_1' u' + y_1 u'' + p(x)(y_1' u + y_1 u') + h(x) y_1 u = 0$$

$$y_1 u'' + 2y_1' u' + p(x) y_1 u' + u\underbrace{(y_1'' + p(x) y_1' + h(x) y_1)}_{=0} = 0$$

$$u'' y_1 + u'(2y_1' + p(x) y_1) = 0$$

$$u'' y_1 = -u'(2y_1' + p(x) y_1)$$

$$\int \frac{u''}{u'} = -\int (2y_1' + p(x) y_1) y_1^{-1}$$

$$\ln u' = -2\ln y_1 - \int p(x)\, dx$$

$$\therefore u' = y_1^{-2} e^{-\int p(x)\, dx}$$

$$\int \frac{1}{x} dx = \ln x$$

$$\int \frac{x'}{x} dx = \ln x$$

$$\int \frac{x''}{x'} dx = \ln x'$$

○ 이계 미분 방정식 치환

$$y'' = y'$$

$$z' = z \quad (\because y' = z)$$

$$\frac{dz}{dx} = z \Rightarrow \int \frac{1}{z} dz = \int dx$$

$$\ln z = x + c$$

$$z = c \cdot e^x$$

$$y' = c \cdot e^x$$

$$\frac{dy}{dx} = c \cdot e^x$$

$$\Rightarrow \int dy = \int c\, e^x\, dx$$

$$\therefore y = c \cdot e^x + D$$

6) 테일러 급수와 오일러 정리

∘ 테일러 급수

다항식 $f(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + \cdots$

$$f(0) = a_0$$
$$f'(0) = a_1$$
$$f''(0) = 2a_2 \quad , \quad a_2 = \frac{f''(0)}{2}$$
$$\vdots$$
$$f^{n}(0) = (n!)\, a_n \quad , \quad a_n = \frac{f^{n}(0)}{n!}$$

$$\therefore f(x) = f(0) + f'(0)x + \frac{f''(0)}{2!}x^2 + \frac{f'''(0)}{3!}x^3 + \cdots + \frac{f^{n}(0)}{n!}x^n$$

① $e^x = 1 + x + \frac{1}{2}x^2 + \frac{1}{3!}x^3 + \cdots \frac{1}{n!}x^n \Rightarrow \sum_{n=0}^{\infty} \frac{x^n}{n!}$

② $\sin x = \underset{0}{\underset{\shortparallel}{\sin 0}} + \underset{1}{\underset{\shortparallel}{\cos(0)}}\, x + \underset{0}{\underset{\shortparallel}{\frac{-\sin 0}{2!}}}\, x^2 + \underset{1}{\underset{\shortparallel}{\frac{-\cos 0}{3!}}}\, x^3 + \cdots \Rightarrow \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n+1)!} x^{2n+1}$

$$= x - \frac{1}{3!}x^3 + \frac{1}{5!}x^5 + \cdots$$

③ $\cos x = 1 - \frac{1}{2!}x^2 + \frac{1}{4!}x^4 - \frac{1}{6!}x^6 + \cdots \Rightarrow \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n)!} x^{2n}$

④ $\log(x+1) = x - \frac{1}{2}x^2 + \frac{1}{3}x^3 - \frac{1}{4}x^4 + \cdots \Rightarrow \sum_{n=0}^{\infty} \frac{(-1)^n}{n!} x^n$

∘ 오일러의 정리

$$e^{ix} = 1 + ix + \frac{1}{2!}(ix)^2 + \frac{1}{3!}(ix)^3 + \cdots$$
$$= \left(1 - \frac{1}{2!}x^2 + \frac{1}{4!}x^4 - \cdots\right) + i\left(x - \frac{1}{3!}x^3 + \frac{1}{5!}x^5 - \cdots\right)$$
$$= \cos x + i \sin x$$

① $\cos x = \dfrac{e^{ix} + e^{-ix}}{2}$

② $\sin x = \dfrac{e^{ix} - e^{-ix}}{2i}$

7) y = 3 * e^(-x^2)의 계측 값과 물리 모델링 y` = -2xy에 대한 오차율 프로그래밍 1

< 접근방식 >

$y = 3 \cdot e^{-x^2}$

1) 테일러 급수

$f(x) = f(0) + f'(0)x + \frac{f''(0)}{2!}x^2 + \frac{f'''(0)}{3!}x^3 + \cdots$

$f(0) = 3 \cdot e^0 = 3$

$f'(0) = -2x \cdot 3 \cdot e^{-x^2} = 0$

$f''(0) = -6 \cdot e^{-x^2} + 4x^2 \cdot 3 \cdot e^{-x^2} = -6$

$f'''(0) = +12x \cdot e^{-x^2} + 8x \cdot 3 \cdot e^{-x^2} - 8x^3 \cdot 3 \cdot e^{-x^2} = 0$

$f''''(0) = 12 \cdot e^{-x^2} - 24x^2 \cdot e^{-x^2} - 24x^2 \cdot 3 \cdot e^{-x^2} + \cdots = 12$

$\left[\sum_{n=0}^{\infty} 3 \cdot \frac{(-2)^n}{(2n)!} x^{2n} \right] \Rightarrow$ 프로그래밍

$\Downarrow$

2) 미분 $\frac{y[1] - y[0]}{\Delta x} = \frac{dy}{dx} = y' \Rightarrow$ 프로그래밍

delta_x = 0.001 , $-5 \le x \le 5$

3) 적분 $\frac{y[1] + y[0]}{2} \cdot \Delta x = \int f(x)\,dx$

4) $y' = -2xy$, $y(0) = 3 \Rightarrow$ 모델링

$\Rightarrow$ 일계 미방 구하는 알고리즘

3) $y = 3 \cdot e^{-x^2}$ 과 4) 모델링과의 오차 비교

< Header >

```c
#ifndef __FIRST_ORDER_EQUATION_H__
#define __FIRST_ORDER_EQUATION_H__

#include<stdio.h>
#include<math.h>
#include<fcntl.h>
#include<string.h>
#include<stdlib.h>

#define DELTA_X     0.001
#define PERIOD  10001
#define START   -5

double y_prime_equation(double x, double y);
void diff_equation(double (*p)(double x,double y), double *y);

void sensor_value(double *y){

    int i;
    for( i=0; i<PERIOD; i++){
        /* y[0] = 3 * e ^ -x^2  , (-5 <= x <= 5 ==  x = -5 + 0.001*i  to 5*/
        y[i] = 3 * exp( -pow( START + DELTA_X * i, 2.0) );
        printf("sensor_value[%d] = %0.12lf ",i, y[i]);
        if( i % 2 ==0 )
            printf("\n");
    }
}

/* 오차율   (y1- y2)/ y1 * 100 */
void error_rate( double *y1, double *y2){

    int i;
    for(i=0;i<PERIOD; i++){
        printf("ERROR RATE[%d] : %lf%% ",i, (y1[i] - y2[i])/y1[i] * 100);
        if(i % 3 ==0)
            printf("\n");
    }
}

/* y` = -2xy */
double y_prime_equation(double x, double y){
    return -2*x*y;
}


void diff_equation(double (*p)(double x,double y), double *y){

    int i;
    double x;

    /*x = 0 ~ 5*/
    for(i=5001; i<PERIOD;i++){
        x = START + DELTA_X * i;
        y[i] = fabs(p(x, y[i-1]) * DELTA_X + y[i-1]);
    }
    /*x = -5 ~ 0*/
    for(i=4999; i>0; i--){
        x = START + DELTA_X * i;
        y[i] = fabs(-p(x, y[i + 1]) * DELTA_X + y[i +1]);

    }
    for(i=0; i<PERIOD; i++){
        printf("modeling_value[%d] = %0.12lf ",i, y[i]);
        if( i % 2 ==0 )
            printf("\n");
    }
}

#endif
```

7) y = 3 * e^(-x^2)의 계측 값과 물리 모델링 y` = -2xy에 대한 오차율 프로그래밍 2

< Main >

```c
#include"first_order_diff_equation.h"


int main(void){
    double y1[PERIOD] = {0}; // sensor_value;
    double y2[PERIOD] = {0}; // modeling_value;

    y2[5000] = 3;   // y(0) = 3;

    sensor_value(y1);
    printf("------------------------------------------------------------------\n\n");
    diff_equation(y_prime_equation, y2);
    printf("------------------------------------------------------------------\n\n");

    error_rate(y1, y2);
    return 0;
}
```

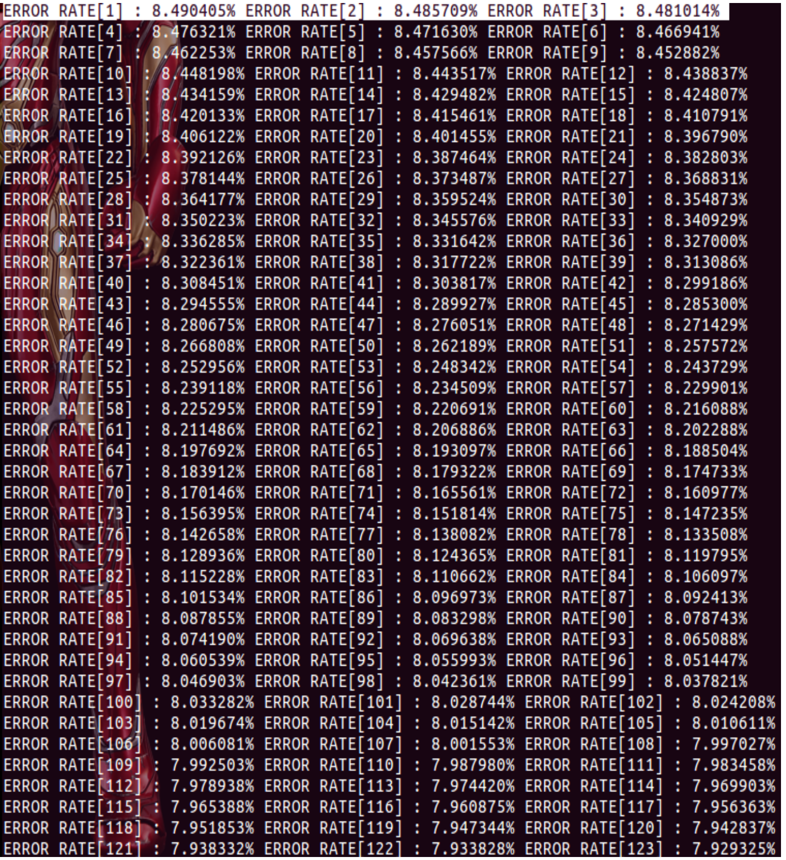
< 결과 >

```
sensor_value[4959] = 2.994961236267 sensor_value[4960] = 2.995203837953
sensor_value[4961] = 2.995440468403 sensor_value[4962] = 2.995671126199
sensor_value[4963] = 2.995895809959 sensor_value[4964] = 2.996114518336
sensor_value[4965] = 2.996327250019 sensor_value[4966] = 2.996534003732
sensor_value[4967] = 2.996734778236 sensor_value[4968] = 2.996929572327
sensor_value[4969] = 2.997118384838 sensor_value[4970] = 2.997301214636
sensor_value[4971] = 2.997478060624 sensor_value[4972] = 2.997648921743
sensor_value[4973] = 2.997813796968 sensor_value[4974] = 2.997972685310
sensor_value[4975] = 2.998125585815 sensor_value[4976] = 2.998272497568
sensor_value[4977] = 2.998413419687 sensor_value[4978] = 2.998548351327
sensor_value[4979] = 2.998677291679 sensor_value[4980] = 2.998800239968
sensor_value[4981] = 2.998917195458 sensor_value[4982] = 2.999028157447
sensor_value[4983] = 2.999133125269 sensor_value[4984] = 2.999232098296
sensor_value[4985] = 2.999325075932 sensor_value[4986] = 2.999412057620
sensor_value[4987] = 2.999493042839 sensor_value[4988] = 2.999568031103
sensor_value[4989] = 2.999637021961 sensor_value[4990] = 2.999700015000
sensor_value[4991] = 2.999757009841 sensor_value[4992] = 2.999808006144
sensor_value[4993] = 2.999853003601 sensor_value[4994] = 2.999892001944
sensor_value[4995] = 2.999925000937 sensor_value[4996] = 2.999952000384
sensor_value[4997] = 2.999973000121 sensor_value[4998] = 2.999988000024
sensor_value[4999] = 2.999997000002 sensor_value[5000] = 3.000000000000
sensor_value[5001] = 2.999997000002 sensor_value[5002] = 2.999988000024
sensor_value[5003] = 2.999973000121 sensor_value[5004] = 2.999952000384
sensor_value[5005] = 2.999925000937 sensor_value[5006] = 2.999892001944
sensor_value[5007] = 2.999853003601 sensor_value[5008] = 2.999808006144
sensor_value[5009] = 2.999757009841 sensor_value[5010] = 2.999700015000
sensor_value[5011] = 2.999637021961 sensor_value[5012] = 2.999568031103
sensor_value[5013] = 2.999493042839 sensor_value[5014] = 2.999412057620
sensor_value[5015] = 2.999325075932 sensor_value[5016] = 2.999232098296
sensor_value[5017] = 2.999133125269 sensor_value[5018] = 2.999028157447
sensor_value[5019] = 2.998917195458 sensor_value[5020] = 2.998800239968
sensor_value[5021] = 2.998677291679 sensor_value[5022] = 2.998548351327
sensor_value[5023] = 2.998413419687 sensor_value[5024] = 2.998272497568
sensor_value[5025] = 2.998125585815 sensor_value[5026] = 2.997972685310
sensor_value[5027] = 2.997813796968 sensor_value[5028] = 2.997648921743
sensor_value[5029] = 2.997478060624 sensor_value[5030] = 2.997301214636
sensor_value[5031] = 2.997118384838 sensor_value[5032] = 2.996929572327
sensor_value[5033] = 2.996734778236 sensor_value[5034] = 2.996534003732
sensor_value[5035] = 2.996327250019 sensor_value[5036] = 2.996114518336
sensor_value[5037] = 2.995895809959 sensor_value[5038] = 2.995671126199
sensor_value[5039] = 2.995440468403 sensor_value[5040] = 2.995203837953
sensor_value[5041] = 2.994961236267 sensor_value[5042] = 2.994712664801
sensor_value[5043] = 2.994458125042 sensor_value[5044] = 2.994197618518
sensor_value[5045] = 2.993931146788 sensor_value[5046] = 2.993658711449
sensor_value[5047] = 2.993380314135 sensor_value[5048] = 2.993095956512
sensor_value[5049] = 2.992805640285 sensor_value[5050] = 2.992509367192
sensor_value[5051] = 2.992207139009 sensor_value[5052] = 2.991898957545
sensor_value[5053] = 2.991584824647 sensor_value[5054] = 2.991264742196
sensor_value[5055] = 2.990938712108 sensor_value[5056] = 2.990606736336
sensor_value[5057] = 2.990268816867 sensor_value[5058] = 2.989924955726
sensor_value[5059] = 2.989575154970 sensor_value[5060] = 2.989219416693
sensor_value[5061] = 2.988857743025 sensor_value[5062] = 2.988490136131
sensor_value[5063] = 2.988116598211 sensor_value[5064] = 2.987737131499
sensor_value[5065] = 2.987351738268 sensor_value[5066] = 2.986960420822
sensor_value[5067] = 2.986563181503 sensor_value[5068] = 2.986160022687
sensor_value[5069] = 2.985750946787 sensor_value[5070] = 2.985335956247
```

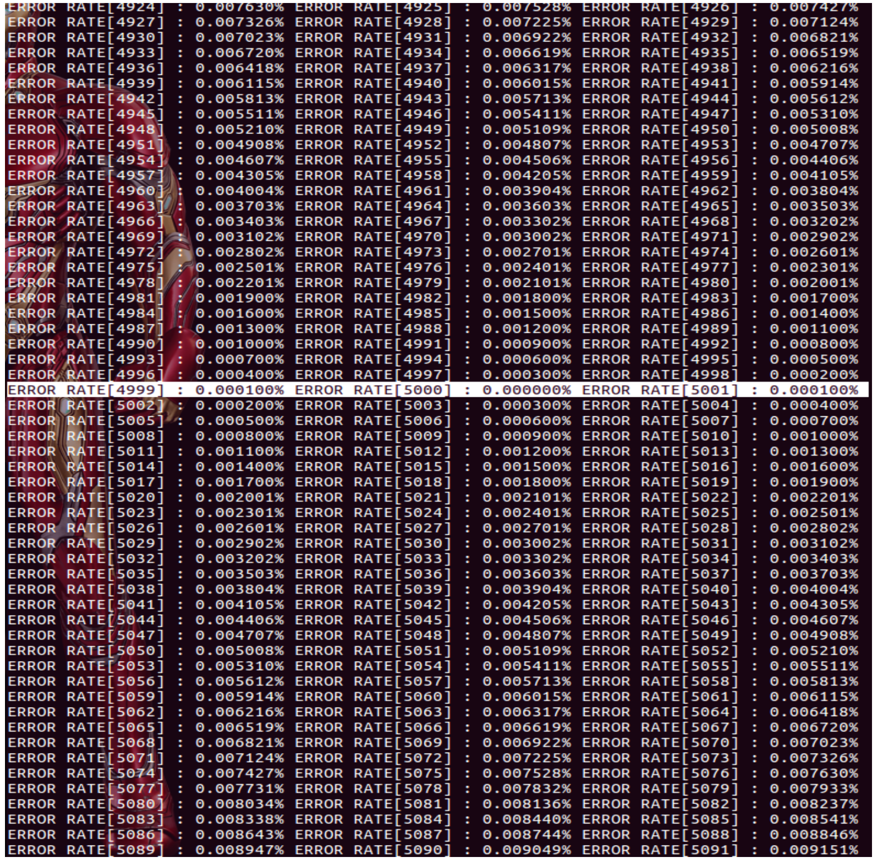
```
modeling_value[4961] = 2.995323525450 modeling_value[4962] = 2.995557178910
modeling_value[4963] = 2.995784858559 modeling_value[4964] = 2.996006563045
modeling_value[4965] = 2.996222291050 modeling_value[4966] = 2.996432041292
modeling_value[4967] = 2.996635812528 modeling_value[4968] = 2.996833603546
modeling_value[4969] = 2.997025413172 modeling_value[4970] = 2.997211240269
modeling_value[4971] = 2.997391083734 modeling_value[4972] = 2.997564942501
modeling_value[4973] = 2.997732815538 modeling_value[4974] = 2.997894701852
modeling_value[4975] = 2.998050600483 modeling_value[4976] = 2.998200510509
modeling_value[4977] = 2.998344431042 modeling_value[4978] = 2.998482361230
modeling_value[4979] = 2.998614300259 modeling_value[4980] = 2.998740247350
modeling_value[4981] = 2.998860201758 modeling_value[4982] = 2.998974162776
modeling_value[4983] = 2.999082129733 modeling_value[4984] = 2.999184101992
modeling_value[4985] = 2.999280078955 modeling_value[4986] = 2.999370060057
modeling_value[4987] = 2.999454044770 modeling_value[4988] = 2.999532032603
modeling_value[4989] = 2.999604023099 modeling_value[4990] = 2.999670015840
modeling_value[4991] = 2.999730010440 modeling_value[4992] = 2.999784006552
modeling_value[4993] = 2.999832003864 modeling_value[4994] = 2.999874002100
modeling_value[4995] = 2.999910001020 modeling_value[4996] = 2.999940000420
modeling_value[4997] = 2.999964000132 modeling_value[4998] = 2.999982000024
modeling_value[4999] = 2.999994000000 modeling_value[5000] = 3.000000000000
modeling_value[5001] = 2.999994000000 modeling_value[5002] = 2.999982000024
modeling_value[5003] = 2.999964000132 modeling_value[5004] = 2.999940000420
modeling_value[5005] = 2.999910001020 modeling_value[5006] = 2.999874002100
modeling_value[5007] = 2.999832003864 modeling_value[5008] = 2.999784006552
modeling_value[5009] = 2.999730010440 modeling_value[5010] = 2.999670015840
modeling_value[5011] = 2.999604023099 modeling_value[5012] = 2.999532032603
modeling_value[5013] = 2.999454044770 modeling_value[5014] = 2.999370060057
modeling_value[5015] = 2.999280078955 modeling_value[5016] = 2.999184101992
modeling_value[5017] = 2.999082129733 modeling_value[5018] = 2.998974162776
modeling_value[5019] = 2.998860201758 modeling_value[5020] = 2.998740247350
modeling_value[5021] = 2.998614300259 modeling_value[5022] = 2.998482361230
modeling_value[5023] = 2.998344431042 modeling_value[5024] = 2.998200510509
modeling_value[5025] = 2.998050600483 modeling_value[5026] = 2.997894701852
modeling_value[5027] = 2.997732815538 modeling_value[5028] = 2.997564942501
modeling_value[5029] = 2.997391083734 modeling_value[5030] = 2.997211240269
modeling_value[5031] = 2.997025413172 modeling_value[5032] = 2.996833603546
modeling_value[5033] = 2.996635812528 modeling_value[5034] = 2.996432041292
modeling_value[5035] = 2.996222291050 modeling_value[5036] = 2.996006563045
modeling_value[5037] = 2.995784858559 modeling_value[5038] = 2.995557178910
modeling_value[5039] = 2.995323525450 modeling_value[5040] = 2.995083899568
modeling_value[5041] = 2.994838302688 modeling_value[5042] = 2.994586736271
modeling_value[5043] = 2.994329201811 modeling_value[5044] = 2.994065700841
modeling_value[5045] = 2.993796234928 modeling_value[5046] = 2.993520805675
modeling_value[5047] = 2.993239414719 modeling_value[5048] = 2.992952063735
modeling_value[5049] = 2.992658754433 modeling_value[5050] = 2.992359488558
modeling_value[5051] = 2.992054267890 modeling_value[5052] = 2.991743094246
modeling_value[5053] = 2.991425969478 modeling_value[5054] = 2.991102895473
modeling_value[5055] = 2.990773874155 modeling_value[5056] = 2.990438907481
modeling_value[5057] = 2.990097997445 modeling_value[5058] = 2.989751146078
modeling_value[5059] = 2.989398355442 modeling_value[5060] = 2.989039627640
modeling_value[5061] = 2.988674964805 modeling_value[5062] = 2.988304369110
modeling_value[5063] = 2.987927842759 modeling_value[5064] = 2.987545387995
modeling_value[5065] = 2.987157007095 modeling_value[5066] = 2.986762702370
modeling_value[5067] = 2.986362476168 modeling_value[5068] = 2.985956330871
modeling_value[5069] = 2.985544268897 modeling_value[5070] = 2.985126292700
modeling_value[5071] = 2.984702404766 modeling_value[5072] = 2.984272607620
```

7) y = 3 * e^(-x^2)의 계측 값과 물리 모델링 y` = -2xy에 대한 오차율 프로그래밍 3

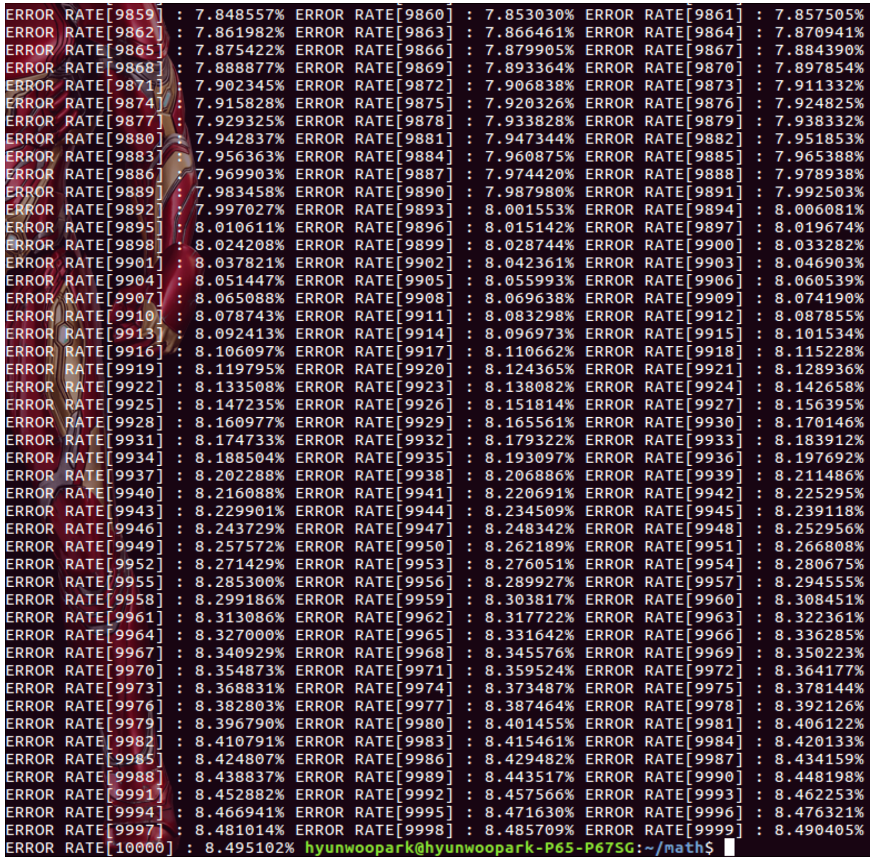
< 오차율 결과 >



```
ERROR RATE[1] : 8.490405% ERROR RATE[2] : 8.485709% ERROR RATE[3] : 8.481014%
ERROR RATE[4] : 8.476321% ERROR RATE[5] : 8.471630% ERROR RATE[6] : 8.466941%
ERROR RATE[7] : 8.462253% ERROR RATE[8] : 8.457566% ERROR RATE[9] : 8.452882%
ERROR RATE[10] : 8.448198% ERROR RATE[11] : 8.443517% ERROR RATE[12] : 8.438837%
ERROR RATE[13] : 8.434159% ERROR RATE[14] : 8.429482% ERROR RATE[15] : 8.424807%
ERROR RATE[16] : 8.420133% ERROR RATE[17] : 8.415461% ERROR RATE[18] : 8.410791%
ERROR RATE[19] : 8.406122% ERROR RATE[20] : 8.401455% ERROR RATE[21] : 8.396790%
ERROR RATE[22] : 8.392126% ERROR RATE[23] : 8.387464% ERROR RATE[24] : 8.382803%
ERROR RATE[25] : 8.378144% ERROR RATE[26] : 8.373487% ERROR RATE[27] : 8.368831%
ERROR RATE[28] : 8.364177% ERROR RATE[29] : 8.359524% ERROR RATE[30] : 8.354873%
ERROR RATE[31] : 8.350223% ERROR RATE[32] : 8.345576% ERROR RATE[33] : 8.340929%
ERROR RATE[34] : 8.336285% ERROR RATE[35] : 8.331642% ERROR RATE[36] : 8.327000%
ERROR RATE[37] : 8.322361% ERROR RATE[38] : 8.317722% ERROR RATE[39] : 8.313086%
ERROR RATE[40] : 8.308451% ERROR RATE[41] : 8.303817% ERROR RATE[42] : 8.299186%
ERROR RATE[43] : 8.294555% ERROR RATE[44] : 8.289927% ERROR RATE[45] : 8.285300%
ERROR RATE[46] : 8.280675% ERROR RATE[47] : 8.276051% ERROR RATE[48] : 8.271429%
ERROR RATE[49] : 8.266808% ERROR RATE[50] : 8.262189% ERROR RATE[51] : 8.257572%
ERROR RATE[52] : 8.252956% ERROR RATE[53] : 8.248342% ERROR RATE[54] : 8.243729%
ERROR RATE[55] : 8.239118% ERROR RATE[56] : 8.234509% ERROR RATE[57] : 8.229901%
ERROR RATE[58] : 8.225295% ERROR RATE[59] : 8.220691% ERROR RATE[60] : 8.216088%
ERROR RATE[61] : 8.211486% ERROR RATE[62] : 8.206886% ERROR RATE[63] : 8.202288%
ERROR RATE[64] : 8.197692% ERROR RATE[65] : 8.193097% ERROR RATE[66] : 8.188504%
ERROR RATE[67] : 8.183912% ERROR RATE[68] : 8.179322% ERROR RATE[69] : 8.174733%
ERROR RATE[70] : 8.170146% ERROR RATE[71] : 8.165561% ERROR RATE[72] : 8.160977%
ERROR RATE[73] : 8.156395% ERROR RATE[74] : 8.151814% ERROR RATE[75] : 8.147235%
ERROR RATE[76] : 8.142658% ERROR RATE[77] : 8.138082% ERROR RATE[78] : 8.133508%
ERROR RATE[79] : 8.128936% ERROR RATE[80] : 8.124365% ERROR RATE[81] : 8.119795%
ERROR RATE[82] : 8.115228% ERROR RATE[83] : 8.110662% ERROR RATE[84] : 8.106097%
ERROR RATE[85] : 8.101534% ERROR RATE[86] : 8.096973% ERROR RATE[87] : 8.092413%
ERROR RATE[88] : 8.087855% ERROR RATE[89] : 8.083298% ERROR RATE[90] : 8.078743%
ERROR RATE[91] : 8.074190% ERROR RATE[92] : 8.069638% ERROR RATE[93] : 8.065088%
ERROR RATE[94] : 8.060539% ERROR RATE[95] : 8.055993% ERROR RATE[96] : 8.051447%
ERROR RATE[97] : 8.046903% ERROR RATE[98] : 8.042361% ERROR RATE[99] : 8.037821%
ERROR RATE[100] : 8.033282% ERROR RATE[101] : 8.028744% ERROR RATE[102] : 8.024208%
ERROR RATE[103] : 8.019674% ERROR RATE[104] : 8.015142% ERROR RATE[105] : 8.010611%
ERROR RATE[106] : 8.006081% ERROR RATE[107] : 8.001553% ERROR RATE[108] : 7.997027%
ERROR RATE[109] : 7.992503% ERROR RATE[110] : 7.987980% ERROR RATE[111] : 7.983458%
ERROR RATE[112] : 7.978938% ERROR RATE[113] : 7.974420% ERROR RATE[114] : 7.969903%
ERROR RATE[115] : 7.965388% ERROR RATE[116] : 7.960875% ERROR RATE[117] : 7.956363%
ERROR RATE[118] : 7.951853% ERROR RATE[119] : 7.947344% ERROR RATE[120] : 7.942837%
ERROR RATE[121] : 7.938332% ERROR RATE[122] : 7.933828% ERROR RATE[123] : 7.929325%
```



```
ERROR RATE[4924] : 0.007630% ERROR RATE[4925] : 0.007528% ERROR RATE[4926] : 0.007427%
ERROR RATE[4927] : 0.007326% ERROR RATE[4928] : 0.007225% ERROR RATE[4929] : 0.007124%
ERROR RATE[4930] : 0.007023% ERROR RATE[4931] : 0.006922% ERROR RATE[4932] : 0.006821%
ERROR RATE[4933] : 0.006720% ERROR RATE[4934] : 0.006619% ERROR RATE[4935] : 0.006519%
ERROR RATE[4936] : 0.006418% ERROR RATE[4937] : 0.006317% ERROR RATE[4938] : 0.006216%
ERROR RATE[4939] : 0.006115% ERROR RATE[4940] : 0.006015% ERROR RATE[4941] : 0.005914%
ERROR RATE[4942] : 0.005813% ERROR RATE[4943] : 0.005713% ERROR RATE[4944] : 0.005612%
ERROR RATE[4945] : 0.005511% ERROR RATE[4946] : 0.005411% ERROR RATE[4947] : 0.005310%
ERROR RATE[4948] : 0.005210% ERROR RATE[4949] : 0.005109% ERROR RATE[4950] : 0.005008%
ERROR RATE[4951] : 0.004908% ERROR RATE[4952] : 0.004807% ERROR RATE[4953] : 0.004707%
ERROR RATE[4954] : 0.004607% ERROR RATE[4955] : 0.004506% ERROR RATE[4956] : 0.004406%
ERROR RATE[4957] : 0.004305% ERROR RATE[4958] : 0.004205% ERROR RATE[4959] : 0.004105%
ERROR RATE[4960] : 0.004004% ERROR RATE[4961] : 0.003904% ERROR RATE[4962] : 0.003804%
ERROR RATE[4963] : 0.003703% ERROR RATE[4964] : 0.003603% ERROR RATE[4965] : 0.003503%
ERROR RATE[4966] : 0.003403% ERROR RATE[4967] : 0.003302% ERROR RATE[4968] : 0.003202%
ERROR RATE[4969] : 0.003102% ERROR RATE[4970] : 0.003002% ERROR RATE[4971] : 0.002902%
ERROR RATE[4972] : 0.002802% ERROR RATE[4973] : 0.002701% ERROR RATE[4974] : 0.002601%
ERROR RATE[4975] : 0.002501% ERROR RATE[4976] : 0.002401% ERROR RATE[4977] : 0.002301%
ERROR RATE[4978] : 0.002201% ERROR RATE[4979] : 0.002101% ERROR RATE[4980] : 0.002001%
ERROR RATE[4981] : 0.001900% ERROR RATE[4982] : 0.001800% ERROR RATE[4983] : 0.001700%
ERROR RATE[4984] : 0.001600% ERROR RATE[4985] : 0.001500% ERROR RATE[4986] : 0.001400%
ERROR RATE[4987] : 0.001300% ERROR RATE[4988] : 0.001200% ERROR RATE[4989] : 0.001100%
ERROR RATE[4990] : 0.001000% ERROR RATE[4991] : 0.000900% ERROR RATE[4992] : 0.000800%
ERROR RATE[4993] : 0.000700% ERROR RATE[4994] : 0.000600% ERROR RATE[4995] : 0.000500%
ERROR RATE[4996] : 0.000400% ERROR RATE[4997] : 0.000300% ERROR RATE[4998] : 0.000200%
ERROR RATE[4999] : 0.000100% ERROR RATE[5000] : 0.000000% ERROR RATE[5001] : 0.000100%
ERROR RATE[5002] : 0.000200% ERROR RATE[5003] : 0.000300% ERROR RATE[5004] : 0.000400%
ERROR RATE[5005] : 0.000500% ERROR RATE[5006] : 0.000600% ERROR RATE[5007] : 0.000700%
ERROR RATE[5008] : 0.000800% ERROR RATE[5009] : 0.000900% ERROR RATE[5010] : 0.001000%
ERROR RATE[5011] : 0.001100% ERROR RATE[5012] : 0.001200% ERROR RATE[5013] : 0.001300%
ERROR RATE[5014] : 0.001400% ERROR RATE[5015] : 0.001500% ERROR RATE[5016] : 0.001600%
ERROR RATE[5017] : 0.001700% ERROR RATE[5018] : 0.001800% ERROR RATE[5019] : 0.001900%
ERROR RATE[5020] : 0.002001% ERROR RATE[5021] : 0.002101% ERROR RATE[5022] : 0.002201%
ERROR RATE[5023] : 0.002301% ERROR RATE[5024] : 0.002401% ERROR RATE[5025] : 0.002501%
ERROR RATE[5026] : 0.002601% ERROR RATE[5027] : 0.002701% ERROR RATE[5028] : 0.002802%
ERROR RATE[5029] : 0.002902% ERROR RATE[5030] : 0.003002% ERROR RATE[5031] : 0.003102%
ERROR RATE[5032] : 0.003202% ERROR RATE[5033] : 0.003302% ERROR RATE[5034] : 0.003403%
ERROR RATE[5035] : 0.003503% ERROR RATE[5036] : 0.003603% ERROR RATE[5037] : 0.003703%
ERROR RATE[5038] : 0.003804% ERROR RATE[5039] : 0.003904% ERROR RATE[5040] : 0.004004%
ERROR RATE[5041] : 0.004105% ERROR RATE[5042] : 0.004205% ERROR RATE[5043] : 0.004305%
ERROR RATE[5044] : 0.004406% ERROR RATE[5045] : 0.004506% ERROR RATE[5046] : 0.004607%
ERROR RATE[5047] : 0.004707% ERROR RATE[5048] : 0.004807% ERROR RATE[5049] : 0.004908%
ERROR RATE[5050] : 0.005008% ERROR RATE[5051] : 0.005109% ERROR RATE[5052] : 0.005210%
ERROR RATE[5053] : 0.005310% ERROR RATE[5054] : 0.005411% ERROR RATE[5055] : 0.005511%
ERROR RATE[5056] : 0.005612% ERROR RATE[5057] : 0.005713% ERROR RATE[5058] : 0.005813%
ERROR RATE[5059] : 0.005914% ERROR RATE[5060] : 0.006015% ERROR RATE[5061] : 0.006115%
ERROR RATE[5062] : 0.006216% ERROR RATE[5063] : 0.006317% ERROR RATE[5064] : 0.006418%
ERROR RATE[5065] : 0.006519% ERROR RATE[5066] : 0.006619% ERROR RATE[5067] : 0.006720%
ERROR RATE[5068] : 0.006821% ERROR RATE[5069] : 0.006922% ERROR RATE[5070] : 0.007023%
ERROR RATE[5071] : 0.007124% ERROR RATE[5072] : 0.007225% ERROR RATE[5073] : 0.007326%
ERROR RATE[5074] : 0.007427% ERROR RATE[5075] : 0.007528% ERROR RATE[5076] : 0.007630%
ERROR RATE[5077] : 0.007731% ERROR RATE[5078] : 0.007832% ERROR RATE[5079] : 0.007933%
ERROR RATE[5080] : 0.008034% ERROR RATE[5081] : 0.008136% ERROR RATE[5082] : 0.008237%
ERROR RATE[5083] : 0.008338% ERROR RATE[5084] : 0.008440% ERROR RATE[5085] : 0.008541%
ERROR RATE[5086] : 0.008643% ERROR RATE[5087] : 0.008744% ERROR RATE[5088] : 0.008846%
ERROR RATE[5089] : 0.008947% ERROR RATE[5090] : 0.009049% ERROR RATE[5091] : 0.009151%
```



```
ERROR RATE[9859] : 7.848557% ERROR RATE[9860] : 7.853030% ERROR RATE[9861] : 7.857505%
ERROR RATE[9862] : 7.861982% ERROR RATE[9863] : 7.866461% ERROR RATE[9864] : 7.870941%
ERROR RATE[9865] : 7.875422% ERROR RATE[9866] : 7.879905% ERROR RATE[9867] : 7.884390%
ERROR RATE[9868] : 7.888877% ERROR RATE[9869] : 7.893364% ERROR RATE[9870] : 7.897854%
ERROR RATE[9871] : 7.902345% ERROR RATE[9872] : 7.906838% ERROR RATE[9873] : 7.911332%
ERROR RATE[9874] : 7.915828% ERROR RATE[9875] : 7.920326% ERROR RATE[9876] : 7.924825%
ERROR RATE[9877] : 7.929325% ERROR RATE[9878] : 7.933828% ERROR RATE[9879] : 7.938332%
ERROR RATE[9880] : 7.942837% ERROR RATE[9881] : 7.947344% ERROR RATE[9882] : 7.951853%
ERROR RATE[9883] : 7.956363% ERROR RATE[9884] : 7.960875% ERROR RATE[9885] : 7.965388%
ERROR RATE[9886] : 7.969903% ERROR RATE[9887] : 7.974420% ERROR RATE[9888] : 7.978938%
ERROR RATE[9889] : 7.983458% ERROR RATE[9890] : 7.987980% ERROR RATE[9891] : 7.992503%
ERROR RATE[9892] : 7.997027% ERROR RATE[9893] : 8.001553% ERROR RATE[9894] : 8.006081%
ERROR RATE[9895] : 8.010611% ERROR RATE[9896] : 8.015142% ERROR RATE[9897] : 8.019674%
ERROR RATE[9898] : 8.024208% ERROR RATE[9899] : 8.028744% ERROR RATE[9900] : 8.033282%
ERROR RATE[9901] : 8.037821% ERROR RATE[9902] : 8.042361% ERROR RATE[9903] : 8.046903%
ERROR RATE[9904] : 8.051447% ERROR RATE[9905] : 8.055993% ERROR RATE[9906] : 8.060539%
ERROR RATE[9907] : 8.065088% ERROR RATE[9908] : 8.069638% ERROR RATE[9909] : 8.074190%
ERROR RATE[9910] : 8.078743% ERROR RATE[9911] : 8.083298% ERROR RATE[9912] : 8.087855%
ERROR RATE[9913] : 8.092413% ERROR RATE[9914] : 8.096973% ERROR RATE[9915] : 8.101534%
ERROR RATE[9916] : 8.106097% ERROR RATE[9917] : 8.110662% ERROR RATE[9918] : 8.115228%
ERROR RATE[9919] : 8.119795% ERROR RATE[9920] : 8.124365% ERROR RATE[9921] : 8.128936%
ERROR RATE[9922] : 8.133508% ERROR RATE[9923] : 8.138082% ERROR RATE[9924] : 8.142658%
ERROR RATE[9925] : 8.147235% ERROR RATE[9926] : 8.151814% ERROR RATE[9927] : 8.156395%
ERROR RATE[9928] : 8.160977% ERROR RATE[9929] : 8.165561% ERROR RATE[9930] : 8.170146%
ERROR RATE[9931] : 8.174733% ERROR RATE[9932] : 8.179322% ERROR RATE[9933] : 8.183912%
ERROR RATE[9934] : 8.188504% ERROR RATE[9935] : 8.193097% ERROR RATE[9936] : 8.197692%
ERROR RATE[9937] : 8.202288% ERROR RATE[9938] : 8.206886% ERROR RATE[9939] : 8.211486%
ERROR RATE[9940] : 8.216088% ERROR RATE[9941] : 8.220691% ERROR RATE[9942] : 8.225295%
ERROR RATE[9943] : 8.229901% ERROR RATE[9944] : 8.234509% ERROR RATE[9945] : 8.239118%
ERROR RATE[9946] : 8.243729% ERROR RATE[9947] : 8.248342% ERROR RATE[9948] : 8.252956%
ERROR RATE[9949] : 8.257572% ERROR RATE[9950] : 8.262189% ERROR RATE[9951] : 8.266808%
ERROR RATE[9952] : 8.271429% ERROR RATE[9953] : 8.276051% ERROR RATE[9954] : 8.280675%
ERROR RATE[9955] : 8.285300% ERROR RATE[9956] : 8.289927% ERROR RATE[9957] : 8.294555%
ERROR RATE[9958] : 8.299186% ERROR RATE[9959] : 8.303817% ERROR RATE[9960] : 8.308451%
ERROR RATE[9961] : 8.313086% ERROR RATE[9962] : 8.317722% ERROR RATE[9963] : 8.322361%
ERROR RATE[9964] : 8.327000% ERROR RATE[9965] : 8.331642% ERROR RATE[9966] : 8.336285%
ERROR RATE[9967] : 8.340929% ERROR RATE[9968] : 8.345576% ERROR RATE[9969] : 8.350223%
ERROR RATE[9970] : 8.354873% ERROR RATE[9971] : 8.359524% ERROR RATE[9972] : 8.364177%
ERROR RATE[9973] : 8.368831% ERROR RATE[9974] : 8.373487% ERROR RATE[9975] : 8.378144%
ERROR RATE[9976] : 8.382803% ERROR RATE[9977] : 8.387464% ERROR RATE[9978] : 8.392126%
ERROR RATE[9979] : 8.396790% ERROR RATE[9980] : 8.401455% ERROR RATE[9981] : 8.406122%
ERROR RATE[9982] : 8.410791% ERROR RATE[9983] : 8.415461% ERROR RATE[9984] : 8.420133%
ERROR RATE[9985] : 8.424807% ERROR RATE[9986] : 8.429482% ERROR RATE[9987] : 8.434159%
ERROR RATE[9988] : 8.438837% ERROR RATE[9989] : 8.443517% ERROR RATE[9990] : 8.448198%
ERROR RATE[9991] : 8.452882% ERROR RATE[9992] : 8.457566% ERROR RATE[9993] : 8.462253%
ERROR RATE[9994] : 8.466941% ERROR RATE[9995] : 8.471630% ERROR RATE[9996] : 8.476321%
ERROR RATE[9997] : 8.481014% ERROR RATE[9998] : 8.485709% ERROR RATE[9999] : 8.490405%
ERROR RATE[10000] : 8.495102% hyunwoopark@hyunwoopark-P65-P67SG:~/math$
```

오차율이 중간 값에서 멀어지면 멀어질수록 점점 커짐.

수학 코딩 어렵네요...