



#include <stdio.h>

#include <math.h>

#define END\_T 2e-6

#define div\_num 100 // (int)(END\_T / H)

#define N 2

#define C 20e-9

#define L 2e-6

#define R 30

double func(double x[N], double t, int n);

void calInt(double x[div\_num], double dt, double y[div\_num]);

double init\_vector[N] = {1000, 0};

int main(void)

{

int i, j;

double t = 0;

double h = END\_T / div\_num;

double x[div\_num][N] = {{0}};

double sub\_x[div\_num] = {0};

double vr[div\_num];

double E[div\_num];

for (i = 0; i < N; ++i)

x[0][i] = init\_vector[i];

for (i = 0; i < div\_num; ++i)

{

for (j = 0; j < N; ++j)

{

x[i + 1][j] = x[i][j] + h \* func(x[i], t, j);

}

t += h;

}

for (i = 0; i < div\_num; ++i)

vr[i] = x[i][1] \* R;

for (i = 0; i < div\_num; ++i)

sub\_x[i] = x[i][1] \* x[i][1];

calInt(sub\_x, h, E);

for (i = 0; i < div\_num; ++i)

{

printf("%.10lf,%lf,%lf\r\n", (double)(h \* i), vr[i], E[i]);

}

return 0;

}

double func(double x[N], double t, int n)

{

switch (n)

{

case 0:

return -1 \* x[1] / C;

case 1:

return (x[0] - x[1] \* R) / L;

}

printf("E");

return 1;

}

void calInt(double x[div\_num], double dt, double y[div\_num])

{

int i;

double sum = 0;

for (i = 0; i < div\_num; ++i)

{

sum += x[i] \* dt;

y[i] = sum;

}

}