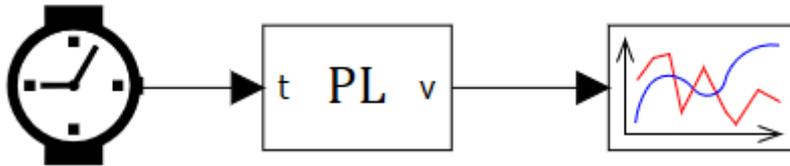


Пояснения к задачам

Задача 2.

Проект *SimInTech* в схеме модели общего вида (Слой «Автоматика»)-



Скрипт проекта (проект в схеме модели общего вида *SimInTech*) -

// Задача - Скорость тела переменной массы (250225)

initialization

var

V0	= 0,	// Начальная скорость м/с
m0	= 200,	// Начальная масса тела, кг
dm	= 1,	// Коэфф. изменения массы, кг/с
F	= 200,	// Тяга двигателя, н
K	= 2,	// Коэфф. сопротивления воздуха, н*сек/м
t_fin	= 50.0;	// Конечное время моделирования, с

end;

Блок PL на языке программирования (язык программирования библиотеки «Динамические») -

// Задача - Скорость тела переменной массы (объект с реактивным двигателем) ((250225)

input t;

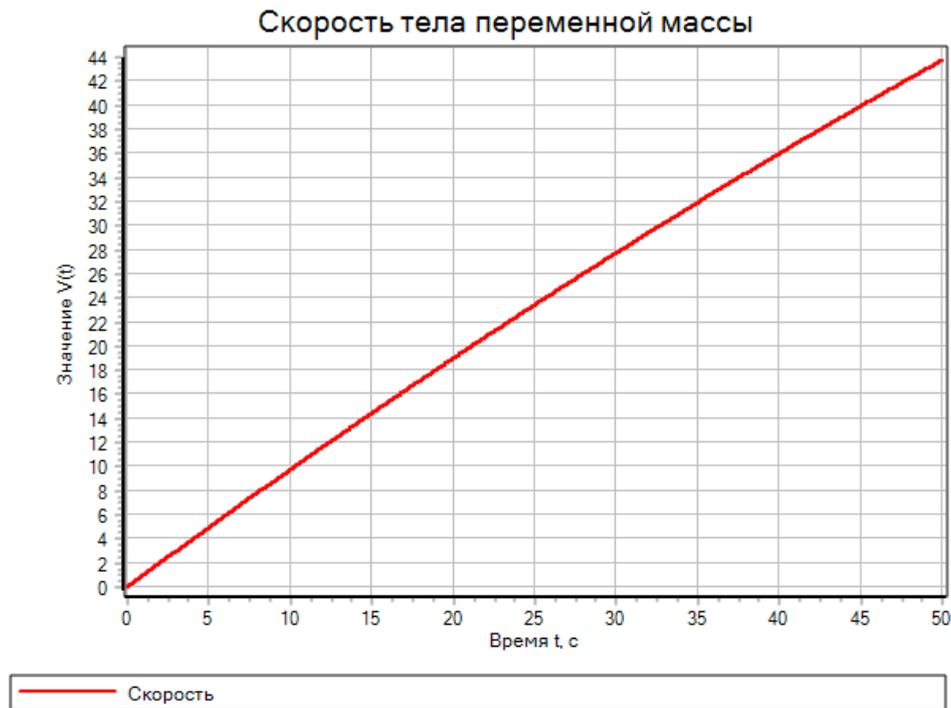
init z=V0;

output V; // Решение

$z' = (F - K * z) / (m0 - dm * time)$

if goodstep then begin V = z; end;

Результаты моделирования



Задача 3.

Проект *SimInTech* в схеме модели общего вида (Слой «Автоматика»)-



Скрипт проекта (проект в схеме модели общего вида *SimInTech*) -

// Задача - модель маятника (250305)

initialization

var

L = 0.25, // длина нити маятника, м

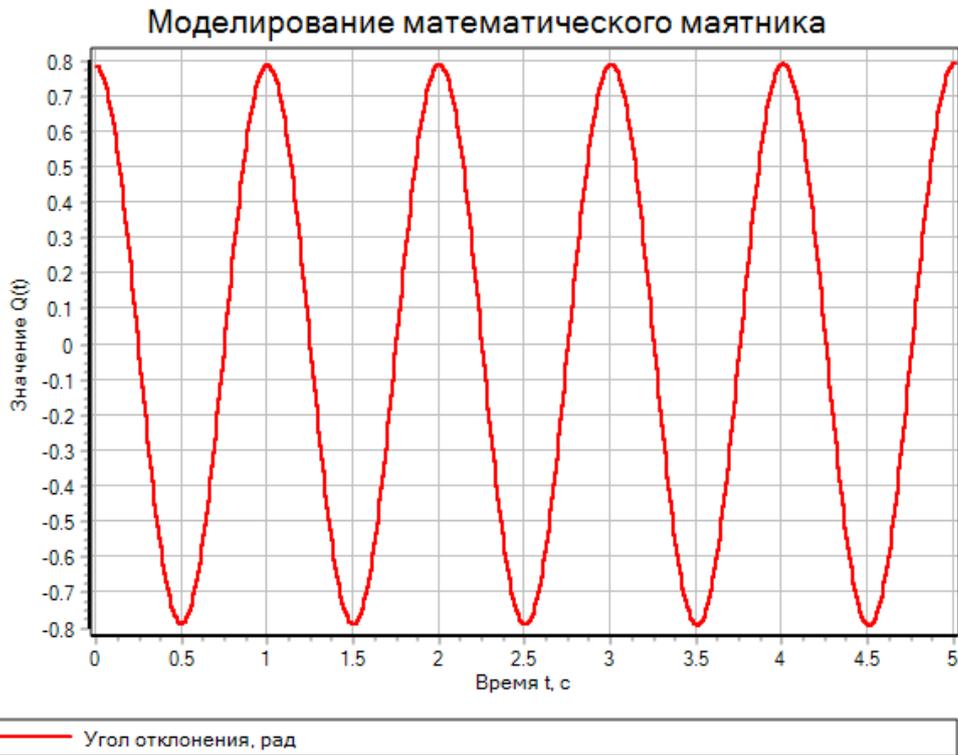
Q0 = pi/4, // угол отклонения маятника при t=0, рад

g = 9.82, // м*м/с

T = 2*pi*sqrt(L/g), // Период колебаний, с

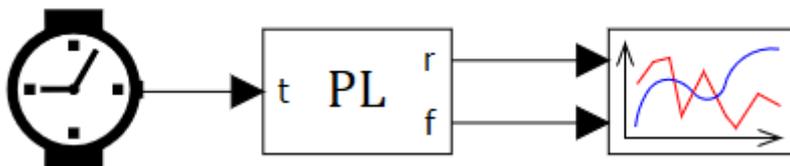
t_fin = 5*T; // Конечное время моделирования,с

end;



Задача 4.

Проект в схеме модели общего вида (Слой «Автоматика»)-



Скрипт (проект в схеме модели общего вида) -

// Задача - Задача - модель Вольтерра (250225)

initialization

var

R0	= 300,	// rabbit,	unit
F0	= 150,	// fox,	unit
alfa	= 0.01,	// alfa	
t_fin	= 30.0;	// Конечное время моделирования	

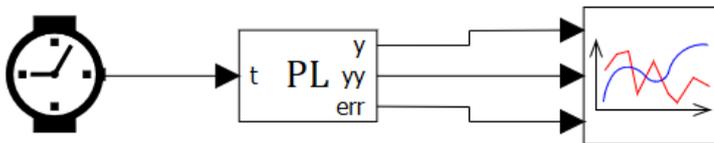
end;

Блок PL на языке программирования (язык программирования библиотеки «Динамические») -

```
// Задача - модель Вольтерра (250225)
input t;
init zr=R0, zf=F0;
output r, f; // Решение
zr' = 2*zr - alfa*zr*zf
zf' = -zf + alfa*zr*zf
if goodstep then begin r = zr; f = zf end;
```

Задача 5.

Проект в схеме модели общего вида (Слой «Автоматика»)-



Скрипт проекта

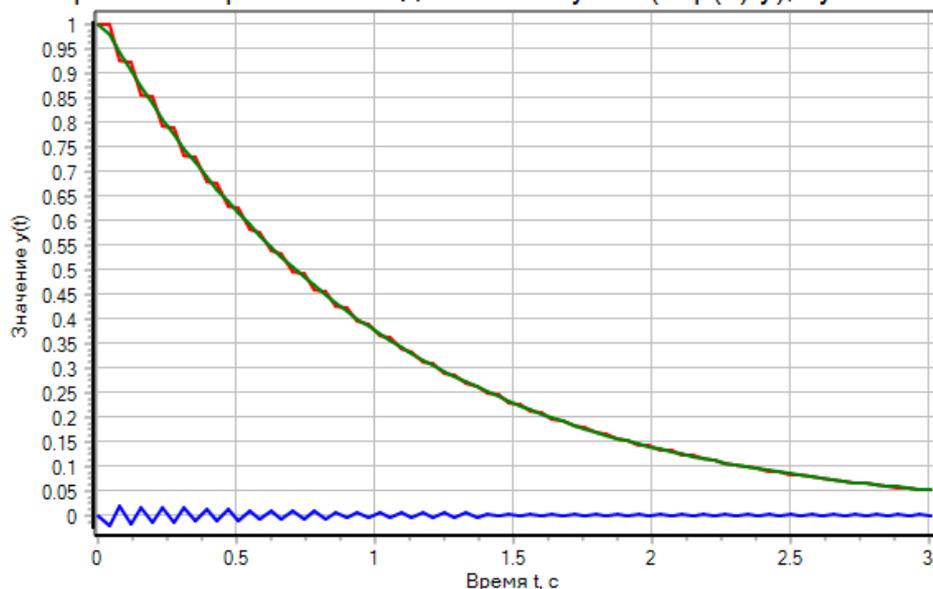
```
// Решение задачи Коши (250227)
initialization
    var
        h      = 0.039, // Шаг интегрирования м
        t_fin  = 3.0,   // Конечное время моделирования, с
        y0     = 1.0;   // Начальное значение искомой функции (при t = 0 )
end;
```

Программа

```
// Задача Kochi 250227
input t;
init z=1;
output y, yy, err; // Численное и аналитическое решения, разность
решений
// Аналитическое решение
function yy_of(t: double):double
    yy_of = (50.0*exp(-t)- exp(-50.0*t))/49.0;
end;
z' = 50*(exp(-t) - z);
if goodstep then begin y = z; yy = yy_of(time); err = yy - y; end;
```

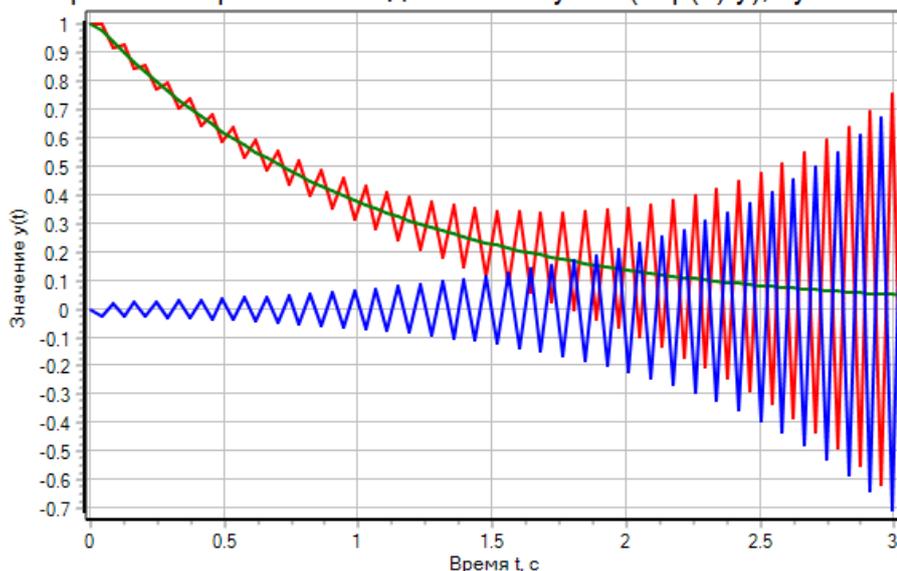
При шаге интегрирования $h=0.039$ решение –

Сравнение решений задачи Коши $y'=50(\exp(-t)-y)$, $y_0=1$.



При шаге интегрирования $h=0.041$ решение

Сравнение решений задачи Коши $y'=50(\exp(-t)-y)$, $y_0=1$.

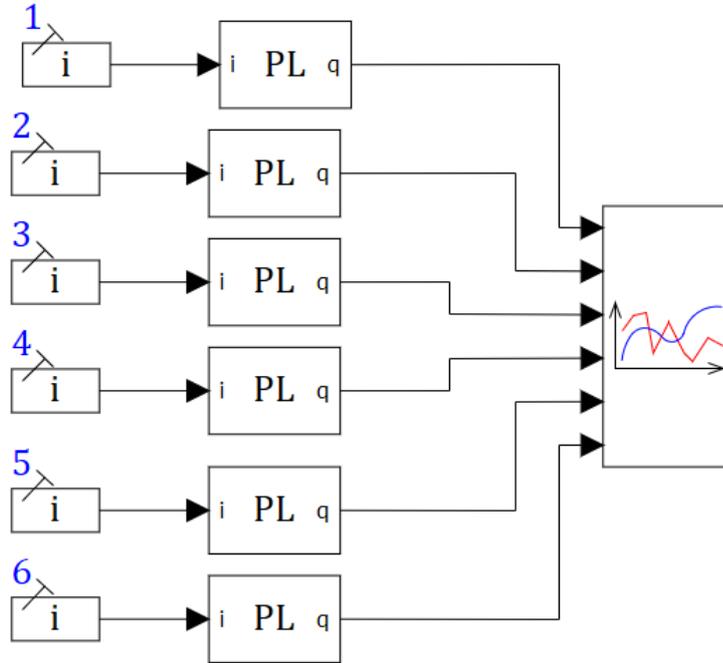


То есть численное решение теряет устойчивость при изменении шага сетки с 0.039 до 0.041. Обычно для конкретной задачи и конкретного метода существует некоторое граничное значение шага h_{\max} , превышение которого приводит к неустойчивости численного решения.

Задача № 6

Ниже представлен один из вариантов решения в схеме модели общего вида (Слой «Автоматика»)-

Порядковый
номер варианта



Параметры расчета

Hmin 1E-5

Hmax 1E-5

Конечное время расчёта t_fin 5

Скрипт проекта (проект в схеме модели общего вида) -

```
// Моделирование движения перевернутого маятника (250306)
```

```
initialization
```

```
var
```

```
g = 9.82, // м/(с*с)
```

```
L = 0.25, // м
```

```
AM = [0, 0.012, 0.25, 0.25, 0.05, 0.012],
```

```
OmegaM = [0, 5.3, 100, 100, 100, 200],
```

```
t_fin = 5; // Конечное время моделирования, с
```

```
end;
```

Для первого варианта заданной таблицы, имеем решение (для остальных вариантов решение с очень небольшими изменениями).

Блок PL на языке программирования (язык программирования библиотеки «Динамические») -

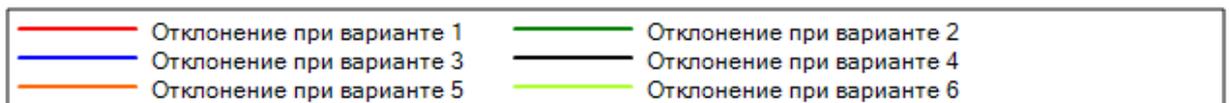
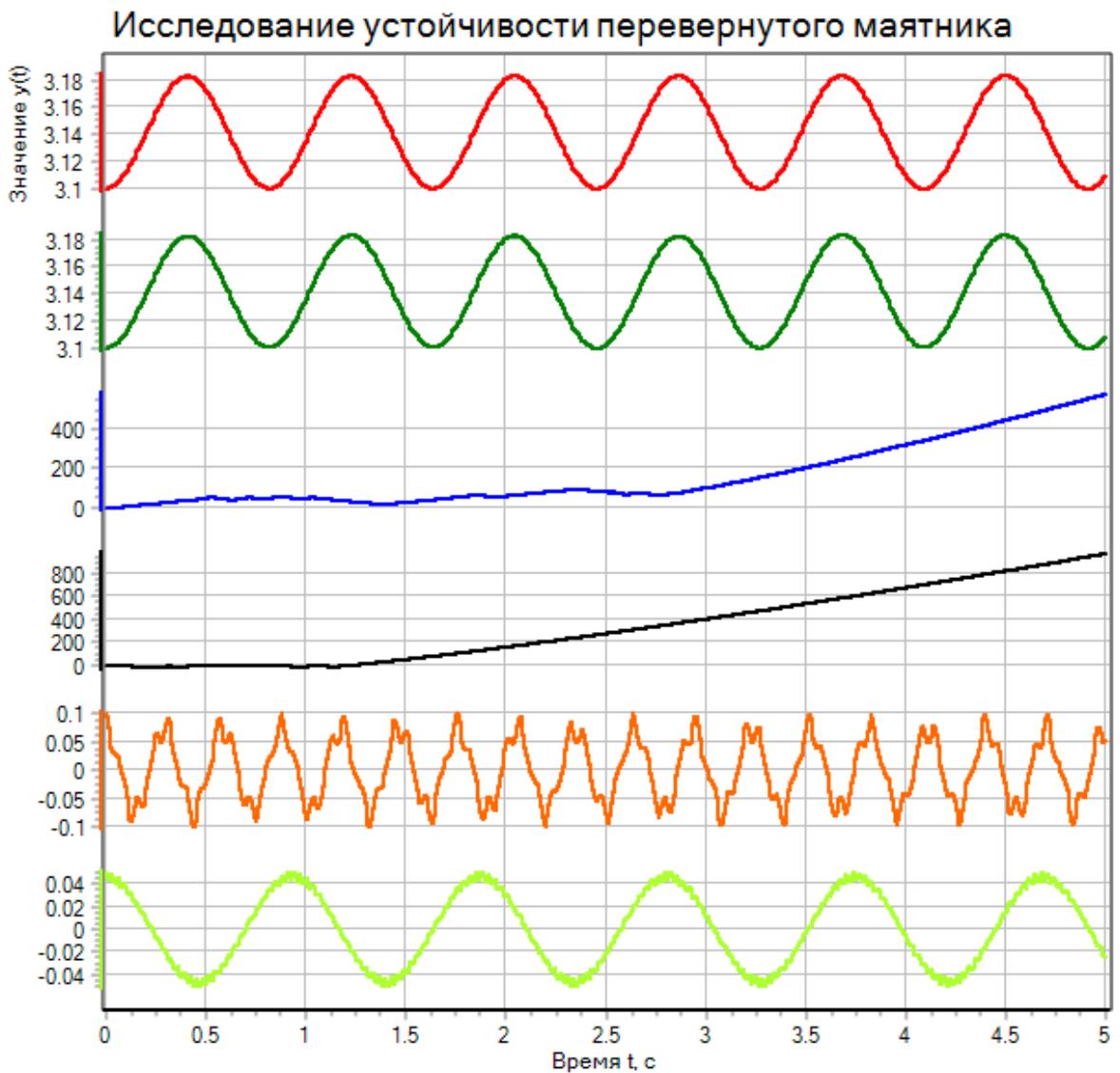
```
// Моделирование движения перевернутого маятника (250306)
```

```

input i;
init zQ = 3.1, // [3.1, 3.1, 3.1, 0.1, 0.1, 0.05],
zV = 0;
output Q;
var
Omega, A;
A = AM[i];
Omega = OmegaM[i];
zQ' = zV;
zV' = (3*(g - A*Omega*Omega*cos(Omega*time))*sin(zQ))/(2*L);
if goodstep then begin Q = zQ; V = zV end;

```

Результаты моделирования



Таким образом, варианты 3 и 4 - неустойчивы и не могут быть реализованы. Варианты 1,2 и 6 устойчивы, вариант 5 условно устойчив.

Задача № 7

Проект в схеме модели общего вида (Слой «Автоматика»)-



Параметры расчета

Hmin 1E-5

Hmax 1E-5

Конечное время расчёта t_fin 1

Скрипт проекта (проект в схеме модели общего вида) -

// Задача - моделирования мех. системы (250311)

initialization

var

t_fin = 1, // Время моделирования, с

m = 0.01, // 1 кг = 0.01 н*с**2/см

K = 2.0, // жесткость пружины, н/см

C = 0.15, // коэффициент демпфирования н*с/см

Kb = 0.2, // коэффициент восстанавливающей силы н/см**3

x0 = 10, // Начальное смещение, см

V0 = 0; // Начальная скорость, см/с

end;

Блок PL на языке программирования (язык программирования библиотеки «Динамические») –

// Задача - моделирования мех. системы (250311)

init zx=x0, zV=V0;

output x, v; // Решение

zx' = zV

zV' = -(C/m)*zV - (K/m)*zx - (Kb/m)*zx^3

if goodstep then begin x = zx; V = zV end;

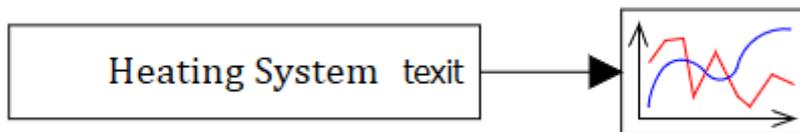
Задача № 8

Движение системы описывается дифференциальным уравнением
 $m\ddot{x} + Kx + f \cdot \dot{x} = 0,$

где m – масса тела, K – коэффициент жесткости пружины, f – коэффициент трения при движении тела.

Задача № 9

Проект в схеме модели общего вида (Слой «Автоматика»)-



Параметры расчета:

hmin 1E-3

hmax 1E-3

Конечное время расчёта t_fin 0.25

Скрипт проекта (проект в схеме модели общего вида) -

// Задача - моделирование системы отопления (250313)

initialization

var

t_fin = 0.25, // Время моделирования, с

T0 = 10, // Начальная температура системы, град

Td = 20, // Заданная температура, град

K_t = 500.0, // Коэффициент потерь тепла в окружающую среду

Q_g = 30000.0, // Подвод тепла от нагревателя, Дж/с

K_e = 250.0, // коэффициент перевода, Дж/град

dT = 1.0; // Допустимый диапазон отклонения температуры, град

end;

Блок PL на языке программирования (язык программирования библиотеки «Динамические») –

// Задача - моделирование системы отопления (250313)

output Texit; // Решение

init zT=T0;

var dQ, dQ_less;

dQ_less = K_t*(zT-T0);

if zT < Td then dQ = Q_g - dQ_less

else dQ = - dQ_less;

zT' = dQ/K_e;

if goodstep then begin Texit = zT end;

Результаты моделирования при заданных параметрах -

