

**САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ПЕТРА ВЕЛИКОГО**

ИНСТИТУТ КОМПЬЮТЕРНЫХ НАУК И КИБЕРБЕЗОПАСНОСТИ

**Направление подготовки: 09.03.04 «Программная инженерия»
65130904/30022**

Курсовая работа, вариант 14.

Выполнил студент: Лютов Александр Владимирович, группа 30022.

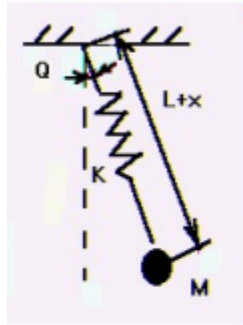
Преподаватель: Воскобойников Сергей Петрович.

Задание

Задание N 14.

Анализ колебаний маятника переменной длины.

Маятник массой M , роль стержня которого выполняет пружина с жесткостью K , совершает сложные колебания относительно положения равновесия. L - начальная длина пружины, x - удлинение пружины относительно положения равновесия, θ - угол отклонения от положения равновесия.



Дифференциальное уравнение движения имеет вид:

$$\begin{aligned} x + \frac{K}{M}x + g(1 - \cos \theta) - (L + x)\left(\dot{\theta}\right)^2 &= 0; \\ \ddot{\theta} + \frac{g}{L + x} \sin \theta + \frac{2}{L + x}x\dot{\theta} &= 0; \\ g &= 9.81. \end{aligned}$$

Начальные условия: $x(0) = A$; $\dot{x}(0) = B$; $\dot{\theta}(0) = C$; $\theta(0) = D$.

Построить графики изменения x и θ для $t \in [0, 4]$ и оценить погрешность результата и влияние на точность погрешности исходных данных. Значения L , M , K , а также A , B , C , D задаются преподавателем.

Вариант N 14С.

Значения A , B , C , D являются решением системы уравнений:

$$\begin{cases} 10A + B + 4C = 16 \\ A + 10B + 5C - D = 20 \\ 4A + 5B + 10C + 7D = 40 \\ -B + 7C + 9D = 28, \end{cases}$$

$$L = 1 + \left(\int_1^5 \frac{dx}{\sqrt{x}(1 + \sqrt[3]{x})} - 1.0507242 \right)^4,$$

$M = 0.7598945 \cdot x^*$ - где x^* - положительный корень уравнения: $e^{-x} + x^2 = 2$; $K = 39.24$.

Результат работы программы

COND= 48697.2695

xstar res 1.31597376

quanc8 res 1.05072427

=====

Начальные условия (решение СЛАУ):

A = 0.00000000 , B = 0.00000000 , C = 4.00000000 , D = -0.00000000

Параметры системы:

L = 1.00000000 , M = 1.00000119 , K = 39.2400017

=====

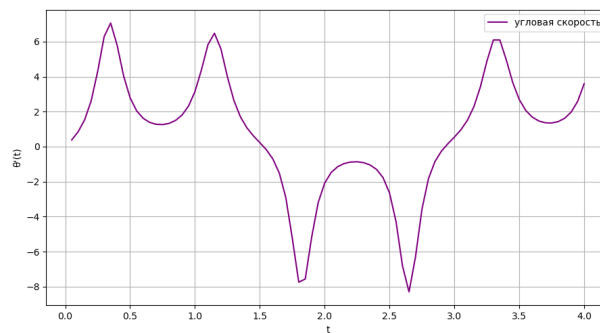
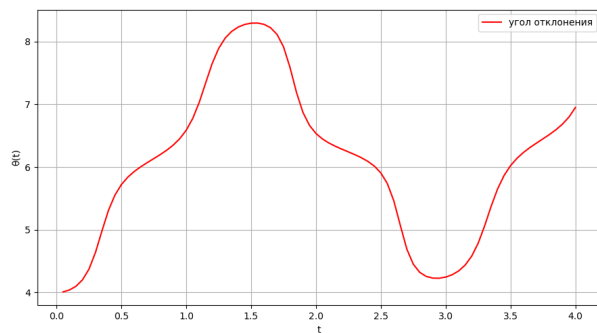
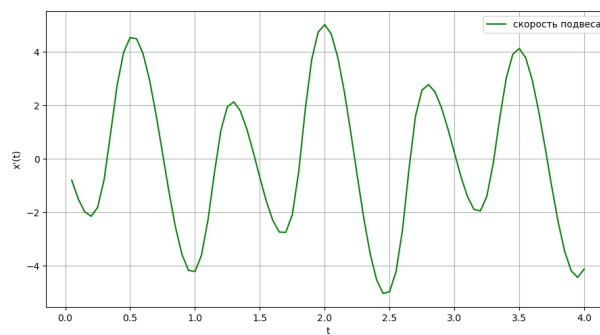
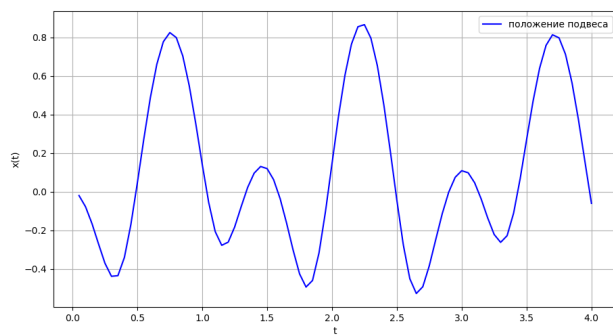
tx(t)x'(t)theta(t)theta'(t)

0.05	-0.020069	-0.794398	4.009452	0.385004
0.10	-0.077764	-1.488173	4.039967	0.859050
0.15	-0.165480	-1.977097	4.098835	1.544083
0.20	-0.270065	-2.139874	4.200951	2.625521
0.25	-0.371247	-1.806826	4.371471	4.300372
0.30	-0.438774	-0.761158	4.637192	6.292617
0.35	-0.435402	0.974244	4.980160	7.064716
0.40	-0.340563	2.760115	5.307593	5.792555
0.45	-0.169060	3.986568	5.552159	4.039142
0.50	0.046849	4.542588	5.720549	2.794355
0.55	0.275165	4.496627	5.839988	2.050471
0.60	0.487886	3.933482	5.930844	1.623980
0.65	0.661492	2.949831	6.005542	1.388782
0.70	0.777820	1.663979	6.071769	1.277568
0.75	0.825170	0.215403	6.134842	1.259693
0.80	0.799196	-1.242947	6.199171	1.328357
0.85	0.703312	-2.555395	6.269331	1.496948
0.90	0.548563	-3.573970	6.351133	1.802703
0.95	0.353047	-4.165108	6.453043	2.316424
1.00	0.141169	-4.209559	6.588075	3.148512
1.05	-0.056995	-3.600455	6.774911	4.393048
1.10	-0.207059	-2.291008	7.031084	5.828738
1.15	-0.277982	-0.513599	7.344727	6.479825
1.20	-0.261661	1.075274	7.652066	5.587688
1.25	-0.182570	1.958458	7.891896	3.993266
1.30	-0.077698	2.133770	8.055927	2.640207
1.35	0.022252	1.794468	8.163060	1.707731
1.40	0.096137	1.117321	8.231732	1.077553
1.45	0.130755	0.245929	8.273640	0.618026
1.50	0.119618	-0.692409	8.294703	0.229132
1.55	0.062392	-1.578258	8.296302	-0.175292
1.60	-0.035427	-2.297334	8.275223	-0.698815
1.65	-0.162645	-2.733902	8.221712	-1.509287
1.70	-0.301767	-2.744434	8.114653	-2.903237

1.75	-0.426124	-2.091760	7.915498	-5.225343
1.80	-0.494396	-0.464043	7.586768	-7.745748
1.85	-0.460406	1.845990	7.188656	-7.566062
1.90	-0.317594	3.727752	6.868705	-5.151933
1.95	-0.102235	4.749188	6.664063	-3.193039
2.00	0.144913	5.024334	6.535082	-2.079368
2.05	0.389885	4.679867	6.447706	-1.476318
2.10	0.604283	3.818902	6.382984	-1.145165
2.15	0.765065	2.556235	6.330700	-0.965307
2.20	0.855502	1.029995	6.284899	-0.879731
2.25	0.866290	-0.602455	6.241569	-0.864321
2.30	0.796250	-2.175508	6.197370	-0.915349
2.35	0.652412	-3.528764	6.148692	-1.047755
2.40	0.449434	-4.519329	6.090569	-1.303254
2.45	0.208502	-5.029331	6.014834	-1.773939
2.50	-0.043873	-4.961658	5.906464	-2.653243
2.55	-0.276330	-4.212052	5.736892	-4.290504
2.60	-0.451303	-2.639884	5.461191	-6.830537
2.65	-0.527876	-0.363716	5.070209	-8.300952
2.70	-0.493025	1.607637	4.694070	-6.297403
2.75	-0.384454	2.576898	4.450827	-3.560879
2.80	-0.247973	2.783985	4.320245	-1.826911
2.85	-0.113949	2.512959	4.255557	-0.850212
2.90	-0.001980	1.922529	4.228947	-0.256073
2.95	0.074864	1.125684	4.227190	0.169535
3.00	0.108933	0.229730	4.245122	0.547353
3.05	0.098058	-0.652705	4.282568	0.963735
3.10	0.045889	-1.401186	4.343577	1.505886
3.15	-0.037630	-1.883083	4.437063	2.283232
3.20	-0.135386	-1.941840	4.577855	3.414499
3.25	-0.221688	-1.394533	4.784318	4.875329
3.30	-0.262797	-0.138354	5.061808	6.097116
3.35	-0.228060	1.549624	5.372889	6.096660
3.40	-0.111361	3.037115	5.651801	4.953013
3.45	0.065292	3.916201	5.866033	3.653548
3.50	0.269219	4.136358	6.023016	2.689577
3.55	0.469288	3.778686	6.140737	2.067996
3.60	0.639262	2.951957	6.233823	1.687903
3.65	0.758721	1.780072	6.312191	1.468772
3.70	0.813836	0.402612	6.382585	1.363565
3.75	0.798066	-1.029733	6.450081	1.351101
3.80	0.712499	-2.364200	6.519218	1.430748
3.85	0.565691	-3.456574	6.595018	1.623122
3.90	0.373003	-4.180066	6.684213	1.977917
3.95	0.155602	-4.429153	6.797045	2.588698
4.00	-0.060528	-4.115520	6.949715	3.596814
4.00	-0.060539	-4.115487	6.949724	3.596879

Графики

Графики изменения x и θ



Выводы

1. Смоделированы колебания маятника переменной длины по заданной системе ОДУ.
2. Система решена методом Рунге–Кутты–Фельберга 4/5 порядка (RK45) с шагом $h=0,05$ на интервале $[0, 4]$.
3. Проведён численный расчёт энергии и её интегрирование с помощью метода QUANC8.
4. Получены устойчивые численные результаты с заданной точностью.
5. Вычислены амплитуда и период колебаний, определены характер и поведение маятника.
6. Погрешности численного метода малы, но чувствительность к исходным данным высока из-за нелинейной природы задачи.

Код программы

```
module pendulum_params
  implicit none
  real :: L, M, K
end module pendulum_params

program variable_length_pendulum
  use pendulum_params
  implicit none

  interface
    subroutine RKF45(F, NEQN, Y, T, TOUT, RELERR, ABSERR, IFLAG, WORK, IWORK)
      implicit none
      external :: F
      integer, intent(in) :: NEQN
      real, intent(inout) :: Y(NEQN), T, TOUT, RELERR, ABSERR
      integer, intent(inout) :: IFLAG
      real, intent(inout) :: WORK(*)
      integer, intent(inout) :: IWORK(*)
    end subroutine RKF45

    subroutine DECOMP(NDIM, N, A, COND, IPV, WORK)
      integer, intent(in) :: NDIM, N
      real, intent(inout) :: A(NDIM, N)
      real, intent(out) :: WORK(N)
      integer, intent(out) :: IPV(N)
      real, intent(out) :: COND
    end subroutine DECOMP

    subroutine SOLVE(NDIM, N, A, B, IPV)
      integer, intent(in) :: NDIM, N, IPV(N)
      real, intent(inout) :: A(NDIM, N)
      real, intent(inout) :: B(N)
    end subroutine SOLVE
  end interface

  external quanc8

  !external :: system_ode, integrand

  integer, parameter :: NEQN = 4
  real :: Y(NEQN), T, TOUT, RELERR, ABSERR
  integer :: IFLAG, IWORK(5)
  real :: WORK(100)
  real :: FINAL, TPRINT
  real :: A_init, B_init, C_init, D_init

  ! Инициализация параметров и начальных условий
  call initialize_parameters(A_init, B_init, C_init, D_init)
  Y = [A_init, B_init, C_init, D_init]

  CALL print_parameters(A_init, B_init, C_init, D_init)

  T = 0.0
  FINAL = 4.0
  TPRINT = 0.05
  TOUT = T + TPRINT
  RELERR = 1.0E-6
  ABSERR = 1.0E-6
  IFLAG = 1

  print *, "t", "x(t)", "x'(t)", "theta(t)", "theta'(t)"
  do while (T < FINAL)
    call RKF45(system_ode, NEQN, Y, T, TOUT, RELERR, ABSERR, IFLAG, WORK, IWORK)
    if (IFLAG /= 2) then
      print *, "Ошибка в RKF45, IFLAG =", IFLAG
      exit
    end if
    print '(F6.2, 4(2X, F12.6))', T, Y(1), Y(2), Y(3), Y(4)
    TOUT = min(TOUT + TPRINT, FINAL)
  end do

contains

  subroutine initialize_parameters(A, B, C, D)
```

```

use pendulum_params
real :: A, B, C, D
real :: MAT(4,4), RHS(4), COND, W(4), X(4)
integer :: IP(4), I
integer :: nfun
real :: xstar, a_integ, b_integ, abserr_integ, relerr_integ, res, esterr, flag

! Система линейных уравнений
MAT(1,:) = [10.0, 1.0, 4.0, 0.0]
MAT(2,:) = [ 1.0, 10.0, 5.0, -1.0]
MAT(3,:) = [ 4.0, 5.0, 10.0, 7.0]
MAT(4,:) = [ 0.0, -1.0, 7.0, 9.0]
RHS = [16.0, 20.0, 40.0, 28.0]

call DECOMP(4, 4, MAT, COND, IP, W)
print *, 'COND=', COND

X = RHS
call SOLVE(4, 4, MAT, X, IP)

A = X(1)
B = X(2)
C = X(3)
D = X(4)

! Найти x*, корень уравнения  $\exp(-x) + x^2 = 2$  методом Ньютона
xstar = 1.0
do
  if (abs(exp(-xstar) + xstar**2 - 2.0) < 1.0e-7) exit
  xstar = xstar - (exp(-xstar) + xstar**2 - 2.0) / (-exp(-xstar) + 2*xstar)
end do

print *, 'xstar res', xstar

M = 0.7598945 * xstar
K = 39.24

! Вычисление L через QUANC8
a_integ = 1.0
b_integ = 5.0
abserr_integ = 0.0
relerr_integ = 1.e-06
call QUANC8(integrand, a_integ, b_integ, abserr_integ, relerr_integ, res, esterr, nfun, flag)
print *, 'quanc8 res', res
L = 1.0 + (res - 1.0507242)**4
end subroutine initialize_parameters

subroutine system_ode(t, y, yp)
  use pendulum_params
  implicit none
  real, intent(in) :: t, y(4)
  real, intent(out) :: yp(4)
  real, parameter :: g = 9.81
  real :: x, xdot, theta, thetadot, denom

  x = y(1)
  xdot = y(2)
  theta = y(3)
  thetadot = y(4)

  yp(1) = xdot
  yp(3) = thetadot

  denom = L + x
  yp(2) = -(K/M)*x - g*(1.0 - cos(theta)) + denom * thetadot**2
  yp(4) = -g/denom*sin(theta) - (2.0/denom)*xdot*thetadot
end subroutine system_ode

real function integrand(x)
  real, intent(in) :: x
  integrand = 1.0 / (sqrt(x) * (1 + x ** (1.0/3.0)))
end function integrand

subroutine print_parameters(A, B, C, D)
  use pendulum_params
  real :: A, B, C, D

```



```
print *, "===== "  
print *, "Начальные условия (решение СЛАУ):"  
print *, "A =", A, ", B =", B, ", C =", C, ", D =", D  
print *, "Параметры системы:"  
print *, "L =", L, ", M =", M, ", K =", K  
print *, "===== "
```

```
end subroutine print_parameters
```

```
end program variable_length_pendulum
```