

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ 1.

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ВЫЧИСЛЕНИЯ ФУНКЦИЙ ОТ МАТРИЦ

Задание. Вычислить функции от матрицы A .

$$A = \frac{1}{6} \begin{bmatrix} p & m & 2S \\ m & 2p + S & m \\ 2S & m & p \end{bmatrix},$$

где

$$p = 2(G + S), \quad m = -p + S,$$

G – последняя цифра зачетной книжки отличная от нуля ($G \neq 0$), S – две последние цифры зачетной книжки отличные от нуля ($S \neq 0$).

Варианты задания

№	$f(A)$	№	$f(A)$
1.	$\cos^2 A^3$	2.	$\sin^4(A + 3A^2)$
3.	$\cos \sqrt{A}$	4.	$\sin(\sqrt[3]{A})$
5.	$A + \sqrt{A^2 + \cos A }$	6.	$\ln(A + \sqrt{A})$
7.	$\ln(\cos^2 A + A^2)$	8.	$\sin(2A^{-1} + \sqrt[5]{A^2})$
9.	$A^0 + A^{-1} + 2A^{-2} + 6A^{-3}$	10.	$(A + A^0)^{-1}$
11.	$\sqrt{(\sin^2 A^{-1} + A^0)}$	12.	$A + \ln(A^2 + 2A^0)$
13.	$\cos(A + \sqrt{A})^3$	14.	$A + \sin \sqrt{A}$
15.	$\arctg(\cos(2A - A^2))$	16.	$\arcsin(\cos^2 A)$

17.	$\arccos(\sin^3(A + \sin A))$	18.	$\operatorname{arctg}\left(e^{-\sqrt[3]{A^2}}\right)$
19.	$\ln(\cos^4(A - \sqrt{A}))$	20.	$(A + \sqrt[4]{A})^{-2}$
21.	$\sin^3(\operatorname{arctg}(A^{-1} + A)) + \frac{1}{2}A$	22.	$\frac{1}{4A}e^{-\sqrt{A}} + \sin A$
23.	$\frac{2A}{A^{-1} + A}$	24.	$\frac{\sin A + \cos 2A}{A^{-1} + A^2}$
25.	$\frac{\operatorname{arctg}(\ln(A + \sqrt{A}))}{e^{-\sqrt[3]{A}} + A}$	26.	$\frac{A^2 + \sin A}{\sqrt{\cos^4 4A + A^0}}$
27.	$3A + \sin^4 5A$	28.	$\arccos(\sin^2 A^3)$
29.	$\cos(\ln(A^{-1} + \sqrt[3]{A}))$	30.	$\sqrt[5]{A^3} + \ln(A^2 + \cos^2 A)$

Пример М-файла для $f(A) = \sqrt[5]{A^3} + \ln(A^2 + \cos^2 A)$ $S=12$ и $G=3$

```

disp('f(A)=(A^3)^(1/5)+log(A^2+(cos(A))^2)');
f=@(x)(x^3)^(1/5)+log(x^2+(cos(x))^2);
s=input('s='); g=input('g=');
p=2*(s+g); m=-p+s;
A=[p m 2*s; m 2*p+s m; 2*s m p]/6;
disp('матрица A');disp(A)
[T,J]=eig(A);p=diag(J);
disp('собственные значения матрицы A');disp(p')
disp('собственные векторы матрицы A');disp(T)
fp=[f(p(1));f(p(2));f(p(3))];
disp('собственные значения матрицы f(A)');disp(fp')
fA=T*diag(fp)*T';
disp('матрица f(A)');disp(fA)

```

Результаты счета (для $S=12$ и $G=3$)

$$f(A) = (A^3)^{1/5} + \log(A^2 + (\cos(A))^2)$$

$$s=12$$

$$g=3$$

матрица A

$$\begin{bmatrix} 5 & -3 & 4 \\ -3 & 12 & -3 \\ 4 & -3 & 5 \end{bmatrix}$$

собственные значения матрицы A

$$\begin{bmatrix} 1.0000 & 6.0000 & 15.0000 \end{bmatrix}$$

собственные векторы матрицы A

$$\begin{bmatrix} 0.7071 & 0.5774 & 0.4082 \\ 0.0000 & 0.5774 & -0.8165 \\ -0.7071 & 0.5774 & 0.4082 \end{bmatrix}$$

собственные значения матрицы $f(A)$

$$\begin{bmatrix} 1.2561 & 6.5390 & 10.4962 \end{bmatrix}$$

матрица $f(A)$

$$\begin{bmatrix} 4.5571 & -1.3191 & 3.3010 \\ -1.3191 & 9.1771 & -1.3191 \\ 3.3010 & -1.3191 & 4.5571 \end{bmatrix}$$

>>

Замечание. Поскольку матрица A симметричная, её собственные векторы ортогональные, поэтому с учетом их нормированности при вычислении с использованием функции $[T, J] = \text{eig}(A)$ получаем $T^{-1} = T^T$.

Варианты задания для ручного счета:

$$A = \frac{1}{2} \begin{bmatrix} 3S & -(S+2G) \\ -(S+2G) & 3S \end{bmatrix},$$

G – последняя цифра зачетной книжки отличная от нуля ($g \neq 0$), S – две последние цифры зачетной книжки отличные от нуля ($S \neq 0$).

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ 2.

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ЧИСЛЕННО-АНАЛИТИЧЕСКОГО РЕШЕНИЯ ЗАДАЧИ ТЕПЛОПРОВОДНОСТИ

Математическая постановка задачи теплопроводности имеет вид:

$$\left\{ \begin{array}{l} \frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2} + f(x, t) \quad - \text{уравнение теплопроводности,} \\ 0 < x < \ell, \quad t > 0 \quad - \text{пространственно-временная область,} \\ \left. \begin{array}{l} u(0, t) = \varphi_0(t) \\ u(\ell, t) = \varphi_\ell(t) \end{array} \right\} t \geq 0 \quad - \text{краевые условия,} \\ u(x, 0) = \psi(x), \quad 0 \leq x \leq \ell \quad - \text{начальные условия.} \end{array} \right.$$

где

x – координата по толщине стены, $0 \leq x \leq \ell$; t – координата по времени, $t \geq 0$;
 $u(x, t)$ – значение температуры в точке x во время t ; α – коэффициент температуропроводности материала; $f(x, t)$ – функция, характеризующая мощность возможного источника тепла.

Варианты задания.

$f(x, t) = 0$ – функция, характеризующая мощность источника тепла;

$\alpha = 1$ – коэффициент температуропроводности материала стены;

$$\left\{ \begin{array}{l} \varphi_0(t) = g \\ \varphi_\ell(t) = s \end{array} \right. \quad - \text{краевые условия;}$$

$\psi(x) = g + (g + 3s)x - 2(g + s)x^2$ – начальные условия;

$\ell = 1$ – толщина стены; g – последняя цифра зачетной книжки отличная от нуля ($g \neq 0$), s – две последние цифры зачетной книжки отличные от нуля ($s \neq 0$).

Текст М-функции

```
function teplo_1_exp  
function U=ut(t)  
    E=eye(n);  
    eAt=expm(t*A);  
    U=eAt*U0-A\ (E-eAt)*F;  
end  
s=input('s=');  
g=input('g=');  
n=input('n=');  
L=input('L=');  
alpha=input('alpha=');  
h=L/(n+1);  
c=alpha/h^2;  
a1=ones(n-1,1);  
A=diag(a1,-1)-2*eye(n)+diag(a1,1),A=c*A;  
u0=g; ul=s;  
F=c*[u0;zeros(n-2,1);ul];  
xi=(0:h:L)';x=xi(2:n+1);  
U0=g+(g+3*s)*x-2*(g+s)*x.^2;  
t=[0 0.15 1.5];  
nt=length(t);res=zeros(nt,n+2);  
fprintf('\n          значения функции температуры U(x,t)\n')  
for i=1:nt  
    res(i,:)= [u0 ut(t(i))' ul];  
  
    fprintf('U(%4.2f):',t(i)),fprintf('%6.2f',res(i,:)),fprintf('\n')  
end  
hold on  
plot(xi,res(1,:),'.-')  
plot(xi,res(2,:), 'o-.r')  
plot(xi,res(nt,:), '*:g')  
grid on  
s1=sprintf('t=%2.0f',t(1));  
s2=sprintf('t=%4.2f',t(2));
```

```

s3=sprintf('t=%4.2f',t(nt));
legend(s1,s2,s3,0)
title(sprintf('U(x,t)=exp(At)*U0-inv(A)*(E-exp(At))*F\n%s  %s  %s',
s1,s2,s3))
end

```

Замечание. Здесь вычисление функции от матрицы e^{At} реализуется с использованием стандартной функции `expm(A)`, в которой используется алгоритм

$$e^{At} = Te^{Jt}T^{-1}$$

Результаты счета

```
s=12
```

```
g=3
```

```
n=7
```

```
L=1
```

```
alpha=1
```

```
A =
```

```

-2      1      0      0      0      0      0
 1     -2      1      0      0      0      0
 0      1     -2      1      0      0      0
 0      0      1     -2      1      0      0
 0      0      0      1     -2      1      0
 0      0      0      0      1     -2      1
 0      0      0      0      0      1     -2

```

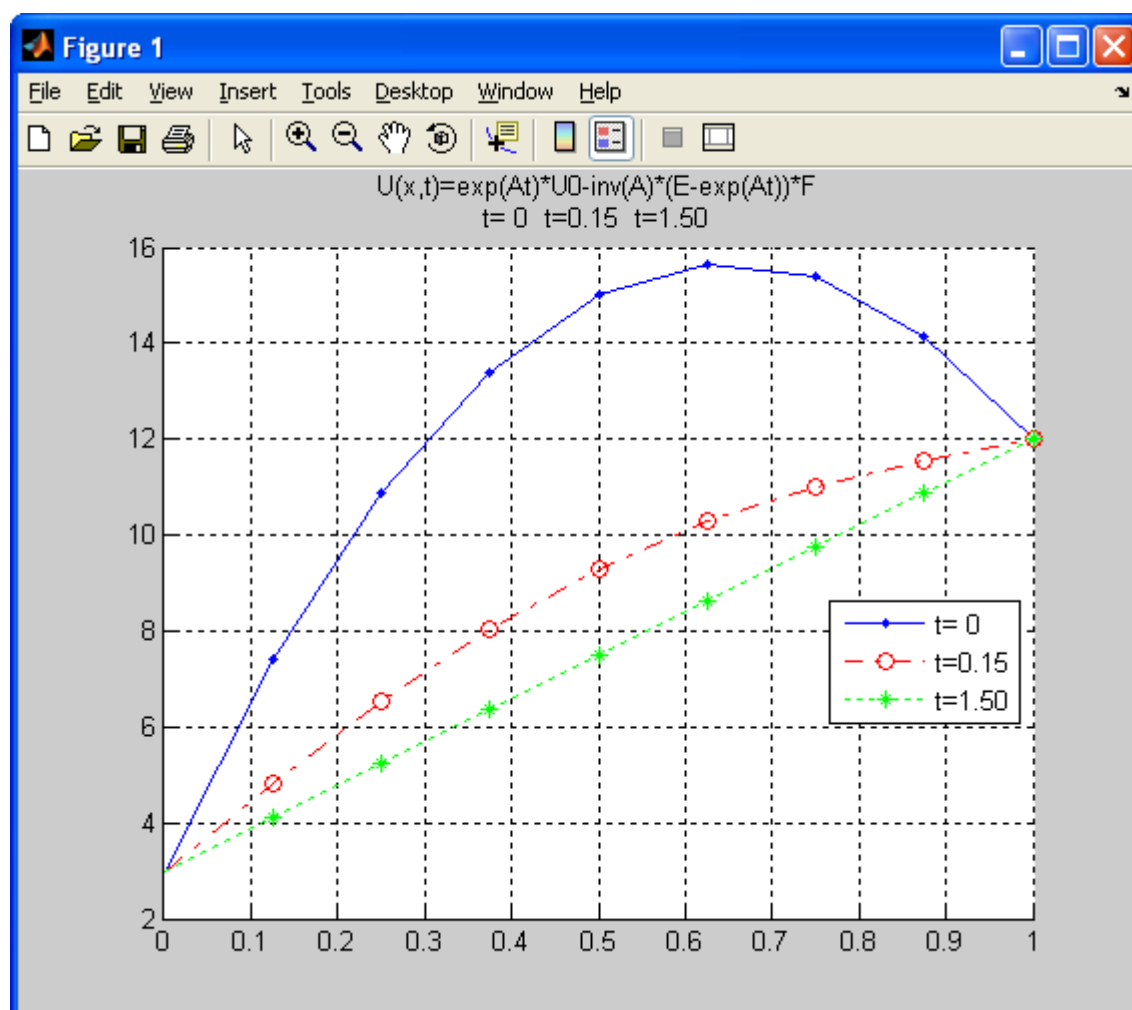
значения функции температуры $U(x,t)$

```

U(0.00):  3.00  7.41 10.88 13.41 15.00 15.66 15.38 14.16 12.00
U(0.15):  3.00  4.81  6.52  8.03  9.29 10.28 11.02 11.56 12.00
U(1.50):  3.00  4.13  5.25  6.38  7.50  8.63  9.75 10.88 12.00

```

График распределения температуры:



ПРАКТИЧЕСКОЕ ЗАНЯТИЕ 3.

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ЧИСЛЕННО-АНАЛИТИЧЕСКОГО РЕШЕНИЯ ЗАДАЧИ О КОЛЕБАНИИ БАЛКИ ПРИ УДАРЕ

Математическая формулировка задачи.

$$\left\{ \begin{array}{l} \frac{\partial^2 y}{\partial t^2} = -a_0 \frac{\partial^4 y}{\partial x^4} + F(x, t) \quad - \text{уравнение колебаний,} \\ 0 < x < \ell, \quad t > 0 \quad - \text{пространственно – временная область,} \\ \left. \begin{array}{l} y(0, t) = y(\ell, t) = 0 \\ y''(0, t) = y''(\ell, t) = 0 \end{array} \right\} - \text{краевые условия, } t \geq 0 \\ \left. \begin{array}{l} y(x, 0) = y_0(x) = 0 \\ \frac{\partial y}{\partial t}(x, 0) = y'_t(x) = 0 \end{array} \right\} - \text{начальные условия, } 0 \leq x \leq \ell \end{array} \right.$$

где $y(x, t)$ – прогиб балки в точке x в момент времени t ; x – координата по длине балки, $0 \leq x \leq \ell$; t – координата по времени, $t \geq 0$; $a_0 = EJ / \rho$; EJ – изгибная жесткость балки; ρ – плотность материала балки; $F(x, t) = P \cdot \delta(x - \ell / 2) \delta(t)$ – функция, моделирующая поперечное ударное воздействие по балке в точке $x = \ell / 2$; $\delta(x - \ell / 2)$ и $\delta(t)$ – дельта-функции.

Варианты задания.

$P = 300$ – величина приложенного сосредоточенного ударного воздействия;
 $a_0 = (100 + G + S) \cdot 10^8$; $\ell = 100$; g – последняя цифра зачетной книжки отличная от нуля ($g \neq 0$), s – две последние цифры зачетной книжки отличные от нуля ($s \neq 0$).

Принять количество внутренних узлов конечно-разностной сетки $N = 7$.

Пример соответствующей М-функции (ниже задано $g = 3$, $s = 12$):

```
function blow_f
g=input('g=');
s=input('s=');
n=input('n=');
L=300;P=300;
h=L/(n+1);
alfa=10^8*(100+g+s);
x=0:h:L;
a0=6*eye(n);a0(1,1)=5;a0(n,n)=5;
a1=ones(n-1,1);
a2=ones(n-2,1);
A=a0-4*(diag(a1,-1)+diag(a1,1))+diag(a2,-2)+diag(a2,2)
A=alfa*A/h^4;
F=zeros(n,1);F((n+1)/2)=P;
sq_A=sqrtm(A);
fJ=sqrt(eig(A));
t0=pi/(4*fJ(n));
tmax=125*t0;
nt=3;
t=[t0,tmax/2,tmax];
res=zeros(nt,n+2);
fprintf('\n      прогиб балки Y(x,t)\n')
for i=1:nt
    Y_t=inv(sq_A)*funm(sq_A*t(i),'sin')*F;
    res(i,2:n+1)=Y_t';

    fprintf('Y(%6.4f):',t(i)),fprintf('%8.4f',res(i,:)),fprintf('\n')
end
plot(x,res(1,:),'.-',x,res((nt+1)/2,:),'o-.',x,res(nt,:),'*:')
grid on
s1=sprintf('t=%6.4f',t(1));
s2=sprintf('t=%6.4f',t((nt+1)/2));
s3=sprintf('t=%6.4f',t(nt));
```

```

legend(s1,s2,s3,0)
title(sprintf('Y(x,t)=inv(sqrt(A))*sin(sqrt(A)t)*F\n%s  %s  %s',
s1,s2,s3))

```

Результаты расчета:

g=3

s=12

n=7

A =

5	-4	1	0	0	0	0
-4	6	-4	1	0	0	0
1	-4	6	-4	1	0	0
0	1	-4	6	-4	1	0
0	0	1	-4	6	-4	1
0	0	0	1	-4	6	-4
0	0	0	0	1	-4	5

прогиб балки Y(x,t)

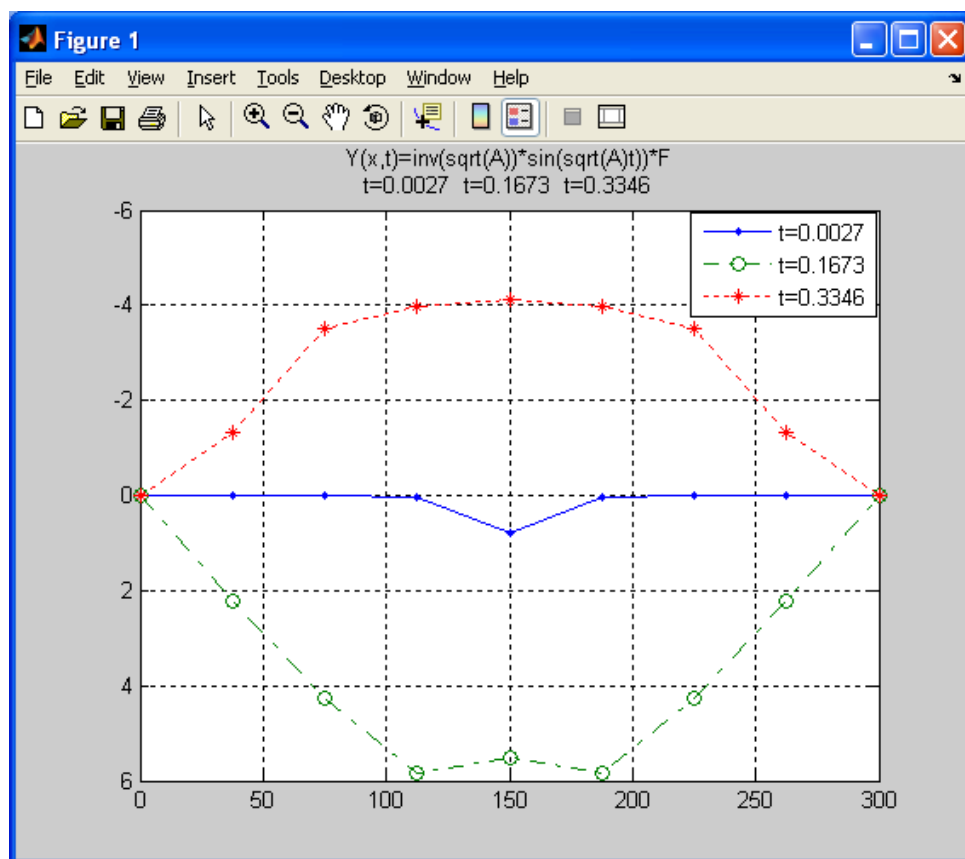
Y(0.0027): 0.0000 -0.0001 -0.0053 0.0217 0.7703 0.0217 -0.0053 -
0.0001 0.0000

Y(0.1673): 0.0000 2.2085 4.2797 5.8565 5.5146 5.8565 4.2797
2.2085 0.0000

Y(0.3346): 0.0000 -1.3399 -3.5149 -3.9787 -4.1294 -3.9787 -3.5149 -
1.3399 0.0000

>>

График колебаний балки при ударе:



Замечание. Здесь вычисление функций от матрицы A реализуются следующим образом:

- $\sqrt{A}$ с использованием стандартной функции $\text{sqrtm}(A)$,
- $\sin(\sqrt{A}t)$ с использованием $\text{funm}(\text{sqrtm}(A), 'sin')$.

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ 4.

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ЧИСЛЕННО-АНАЛИТИЧЕСКОГО РЕШЕНИЯ НА ОСНОВЕ МКЭ ЗАДАЧИ ОБ ИЗГИБЕ БАЛКИ НА УПРУГОМ ОСНОВАНИИ

Общая постановка задачи об изгибе балки на упругом основании (модель Винклера).

Суть модели Винклера состоит в предположении, что реакция основания $r(x)$ в произвольной точке балки x пропорциональна ее прогибу в этой точке: $r(x) = \beta y$. Поэтому графически такую модель можно представить пружинками, не связанными друг с другом, каждая из которых имеет жесткость, пропорциональную прогибу балки в этой точке (рис. 4.1).

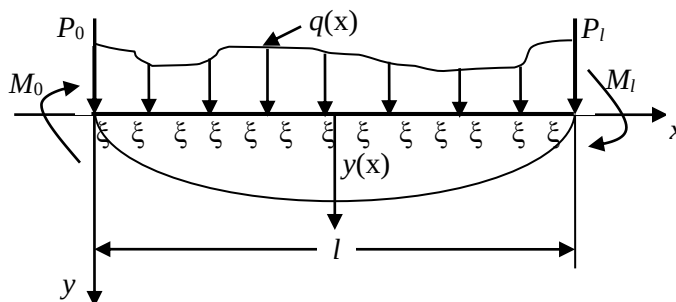


Рис. 4.1. Расчетная схема балки

Напряженно-деформируемое состояние такой балки соответствует решению задачи о минимуме следующего функционала (функционала энергии): функция прогибов балки $y(x)$, вызванных силами P , распределенной нагрузкой q и изгибающими моментами M , является условием минимума функционала энергии балки (т.е. принесет минимальное значение этому функционалу)

$$\Phi(y) = \frac{1}{2} \int_0^l (EJ(y'')^2 + \beta y^2) dx - \int_0^l q(x)y dx - M_0 y'(0) - M_1 y'(l) - P_0 y(0) - P_1 y(l)$$

где

$EJ(x)$ – жесткость балки при изгибе (изгибная жесткость);

$\beta(x)$ – коэффициент упругости основания (коэффициент постели);

$q(x)$, M_0 , P_0 , M_ℓ , P_ℓ – заданные нагрузки.

Замечание. Из курса вариационного исчисления следует, что решение такой задачи совпадает с решением следующей краевой задачи:

$$\left. \begin{aligned} (EJy'')'' + \beta y &= q(x), \quad x \in (0, l) \\ EJy''(0) &= -M_0 \\ EJy'''(0) &= P_0 \\ EJy''(l) &= -M_l \\ EJy'''(l) &= P_l \end{aligned} \right\} - \text{краевые условия},$$

что соответствует исходной постановке, обычно формулируемой в курсе «Сопротивление материалов».

Задание. Получить численное решение и построить графики при $n = 21$.

Варианты задания:

$EJ(x) = c_1$ – жесткость балки;

$\beta(x) = 100$ – коэффициент упругости основания (коэффициент постели);

$q(x) = c\beta(x)x(\ell - x)$ – внешняя нагрузка;

$M_0 = 2cc_1$, $M_\ell = -M_0$ – сосредоточенные изгибающие моменты на краях балки;

$P_0 = 0$, $P_\ell = 0$ – сосредоточенные силы на краях балки;

$\ell = 1$ – длина балки.

При этом

$$c = 4(g + s) / 100 / \ell^2;$$

$$c_1 = 1;$$

g – последняя цифра зачетной книжки отличная от нуля ($g \neq 0$), s – две последние цифры зачетной книжки отличные от нуля ($s \neq 0$).

Пример М-файла:

```
function beam_FEM_cubic
% ввод, задание и формирование исходных данных
    s=input('ввести s=');
    g=input('ввести g=');
    L=1;
    c=4*(s+g)/100/L^2;
    c1=1;
    EJ=c1;
    bt=100;
    n=input('ввести число точек n=');
    ne=n-1; %количество элементов
    h=L/ne; % длина элемента
    xs=h/2:h:L; %массив координат средних точек элементов
    q=c*bt*xs.*(L-xs); %массив нагрузок в средних точках
    элементов
    P0=0; M0=2*c*c1; % краевые условия слева: x=0
    PL=0; ML=-M0; % краевые условия справа: x=L
    mp=2; %число неизвестных в узле: y(xi), y'(xi)
    me=2*mp; %число неизвестных на элементе:
    y(x_i), y'(x_i), y(x_i+1), y'(x_i+1)
    ns=mp*n; %общее количество неизвестных
    xi=0:h:L; % координаты узлов

% формирование вспомогательных матриц и векторов
    Ag=[1 0 0 0;0 1 0 0;1 1 1 1;0 1 2 3]
    A=inv(Ag)
    S=diag([1 2 3]',',-1)
    T0=[ 1 1/2 1/3 1/4
        1/2 1/3 1/4 1/5
        1/3 1/4 1/5 1/6
        1/4 1/5 1/6 1/7]
    TT=420*T0
```

```

Di=diag([1 h 1 h]')
t0=[1 1/2 1/3 1/4]'

% формирование локальных матриц и векторов
K0=h*Di*A'*T0*A*Di
K2=1/h^3*Di*A'*S^2*T0*(S^2) '*A*Di
Ke=EJ*K2+bt*K0 %локальная матрица жесткости
R0=h*Di*A'*t0 %локальный вектор нагрузки

% формирование глобальной матрицы жесткости
Kg=zeros(ns);
for i=1:mp:ns-mp-1
    Kg(i:i+me-1,i:i+me-1)=Kg(i:i+me-1,i:i+me-1)+Ke(:, :);
end

% формирование глобального вектора нагрузки
Rg=zeros(ns,1);
ie=0;
for i=1:mp:ns-mp-1
    ie=ie+1;
    Rg(i:i+me-1)=Rg(i:i+me-1)+q(ie)*R0(:);
end

% учет граничных условий
Rg(1)=Rg(1)+P0;
Rg(2)=Rg(2)+M0;
Rg(ns-1)=Rg(ns-1)+PL;
Rg(ns)=Rg(ns)+ML;

%Kg
%Rg
U=Kg\Rg;
yi=U(1:mp:ns)';
dyi=U(2:mp:ns)';

% вычисление y, y', y'' и y''' в центрах элементов
for i=1:ne
    ys(i)=df(i,0,0.5);
    dy(i)=df(i,1,0.5);
    d2y(i)=df(i,2,0.5);

```

```

        d3y(i)=df(i,3,0.5);
end
% вычисление  $y''$  и  $y'''$  в узлах
for i=1:ne
    d2yi(i)=df(i,2,0);
    d3yi(i)=df(i,3,0);
end
    d2yi(n)=df(ne,2,1);
    d3yi(n)=df(ne,3,1);

disp('прогиб балки'),disp(yi)
disp('угол поворота'),disp(dyi)
disp('изгибающий момент'),disp(-EJ*d2yi)
disp('поперечная сила'),disp(-EJ*d3yi)
figure(),plot(xs,ys,xs,dy,xs,-EJ*d2y,xs,-EJ*d3y),title('v centrah
elementov'),grid on
legend('y','dy','M=-EJ*d2y','Q=-EJ*d3y')
figure(),plot(xi,yi,xi,dyi,xi,-EJ*d2yi,xi,-EJ*d3yi),title('v
uzlah'),grid on
legend('y','dy','M=-EJ*d2y','Q=-EJ*d3y')
    function dpyi=df(i,p,t)
        % вычисление  $dpy$  в точке  $t$  на  $i$ -м элементе
        Dp=(S^p)'*A*Di;
        ti=[1;t;t^2;t^3];
        in=1+mp*(i-1);ik=in+me-1; z=U(in:ik);
        dpyi=dot(Dp*z,ti)/h^p;
    end

end

```

Результаты счета при $g=3$ и $s=12$:

```

ввести s=12
ввести g=3
ввести число точек n=21

```


Ag =

1	0	0	0
0	1	0	0
1	1	1	1
0	1	2	3

A =

1	0	0	0
0	1	0	0
-3	-2	3	-1
2	1	-2	1

S =

0	0	0	0
1	0	0	0
0	2	0	0
0	0	3	0

T0 =

1.0000	0.5000	0.3333	0.2500
0.5000	0.3333	0.2500	0.2000
0.3333	0.2500	0.2000	0.1667
0.2500	0.2000	0.1667	0.1429

TT =

420	210	140	105
210	140	105	84
140	105	84	70
105	84	70	60

Di =

1.0000	0	0	0
0	0.0500	0	0
0	0	1.0000	0
0	0	0	0.0500

t0 =

1.0000
0.5000
0.3333
0.2500

K0 =

0.0186	0.0001	0.0064	-0.0001
0.0001	0.0000	0.0001	-0.0000
0.0064	0.0001	0.0186	-0.0001
-0.0001	-0.0000	-0.0001	0.0000

K2 =

1.0e+004 *

9.6000	0.2400	-9.6000	0.2400
0.2400	0.0080	-0.2400	0.0040
-9.6000	-0.2400	9.6000	-0.2400
0.2400	0.0040	-0.2400	0.0080

Ke =

1.0e+004 *

9.6002	0.2400	-9.5999	0.2400
0.2400	0.0080	-0.2400	0.0040
-9.5999	-0.2400	9.6002	-0.2400
0.2400	0.0040	-0.2400	0.0080

R0 =

0.0250
0.0002
0.0250
-0.0002

прогиб балки

0.0003	0.0288	0.0542	0.0767	0.0961	0.1126
0.1261	0.1366	0.1440	0.1485	0.1500	0.1485
0.1366	0.1261	0.1126	0.0961	0.0767	0.0542
0.0003					

угол поворота

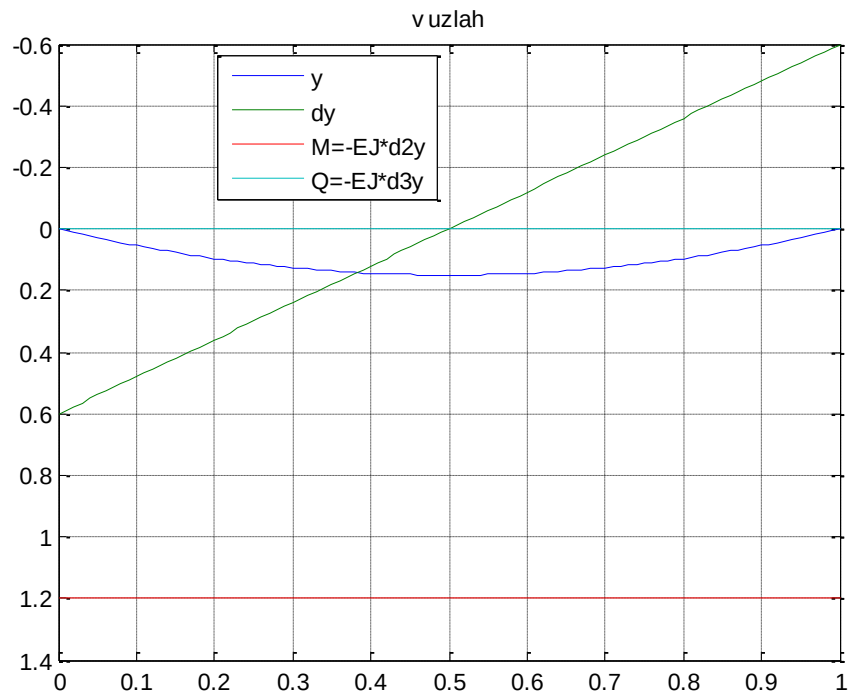
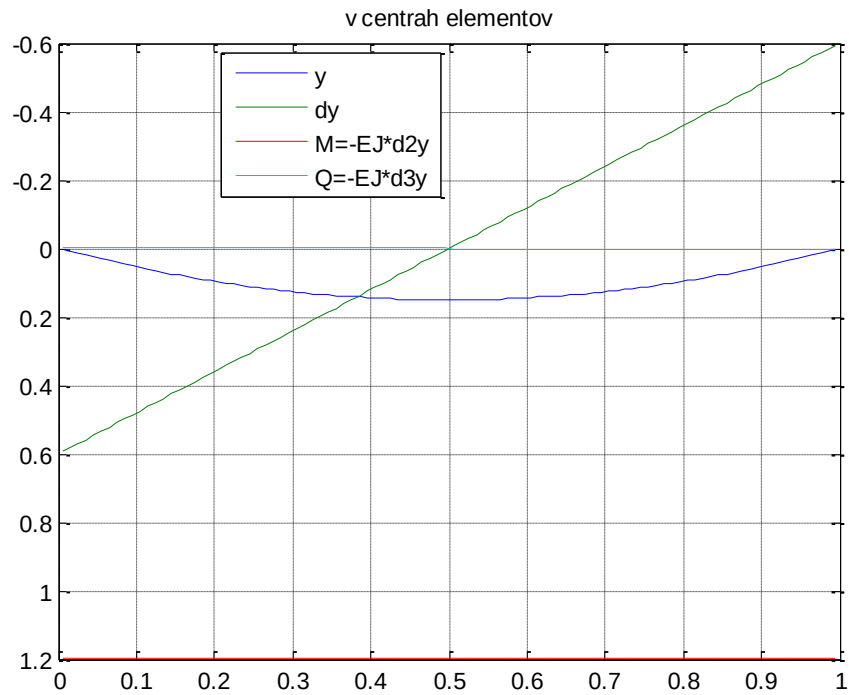
0.5991	0.5391	0.4792	0.4192	0.3593	0.2994
0.2395	0.1796	0.1197	0.0599	-0.0000	-0.1197
-0.1796	-0.2395	-0.2994	-0.3593	-0.4192	-0.4792
-0.5991					

изгибающий момент

1.2001	1.1995	1.1990	1.1986	1.1983	1.1980
1.1978	1.1977	1.1975	1.1975	1.1975	1.1975
1.1976	1.1978	1.1979	1.1982	1.1985	1.1989
1.2001					

поперечная сила

	-0.0138	-0.0117	-0.0097	-0.0080	-0.0064	-0.0050	-
0.0038	-0.0026	-0.0015	-0.0005	0.0005	0.0015	0.0026	
0.0038	0.0050	0.0064	0.0080	0.0097	0.0117	0.0138	
0.0138							



ПРАКТИЧЕСКОЕ ЗАНЯТИЕ 5.

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ АНАЛИТИЧЕСКОГО МЕТОДА ВЫЧИСЛЕНИЯ ГЕОМЕТРИЧЕСКИХ ХАРАКТЕРИСТИК ПОПЕРЕЧНЫХ СЕЧЕНИЙ ЭЛЕМЕНТОВ КОНСТРУКЦИЙ,

Задание.

Определить геометрические характеристики сечения, приведенного на рис. 1 /заштрихованная область/.

Варианты задания.

Координаты точек сечения следует принять по таблице 5.1, где g – последняя цифра зачетной книжки отличная от нуля ($g \neq 0$), s – две последние цифры зачетной книжки отличные от нуля ($s \neq 0$).

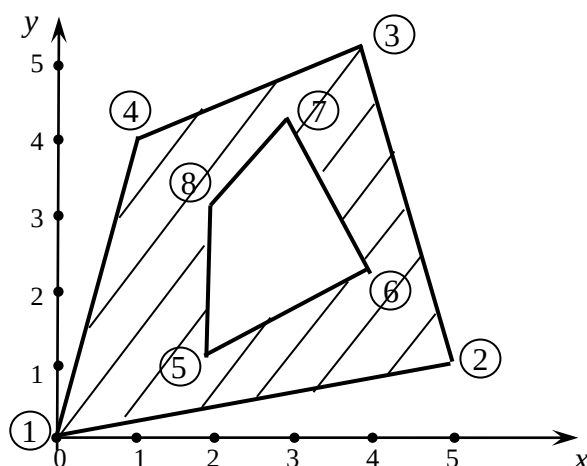


Рис. 5.1. Рассматриваемый многоугольник.

Таблица 5.1

Координаты вершин многоугольника

Координаты точки	1 контур				2 контур			
	1	2	3	4	5	6	7	8
x	$0 \cdot H$	$5 \cdot H$	$4 \cdot H$	$1 \cdot H$	$2 \cdot H$	$4 \cdot H$	$3 \cdot H$	$2 \cdot H$
y	$0 \cdot H$	$1 \cdot H$	$5 \cdot H$	$4 \cdot H$	$1 \cdot H$	$2 \cdot H$	$4 \cdot H$	$3 \cdot H$

Пример выполнения задания (ниже задано $g = 3$, $s = 12$).

Текст М-файла.

```

H=input(' Введите значение параметра H=');
M=input(' Введите количество контуров =');
for k=1:M
    fprintf('\n Порядковый номер контура =%2d \n',k);
    fprintf('\n Введите знак контура');
    fprintf('\n (1 - внешний, -1 - внутренний)');
    znak(k)=input(' Знак контура =');
    N(k)=input(' Введите количество вершин контура =');
end
NMAX=max(N);
x=zeros(M,NMAX+2); y=zeros(M,NMAX+2);
for k=1:M
    fprintf('\n Рассматривается контур номер %2d \n',k);
    x_t=input(' Введите абсциссы точек контура =');
    y_t=input(' Введите ординаты точек контура =');
    x(k,2:N(k)+1)= x_t(1:N(k));
    y(k,2:N(k)+1)= y_t(1:N(k));
    x(k,1)=x(k,N(k)+1);
    x(k,N(k)+2)=x(k,2);
    y(k,1)=y(k,N(k)+1);
    y(k,N(k)+2)=y(k,2);
    xx(k,1:N(k)+1)= x(k,2:N(k)+2);
    yy(k,1:N(k)+1)=y(k,2:N(k)+2);
end

```

```

x=x*H; y=y*H;
xx=xx*H; yy=yy*H;
hold on
for k=1:M
    plot(xx(k,1:N(k)+1),yy(k,1:N(k)+1),'-k');
end
F=0; Sx=0; Sy=0; Jx=0; Jy=0; Jxy=0;
for k=1:M
    for i=2:N+1
        x1(k,i)=(x(k,i+1)+x(k,i))/2;
        y1(k,i)=(y(k,i+1)+y(k,i))/2;
        dx(k,i)=x(k,i+1)-x(k,i);
        dy(k,i)=y(k,i+1)-y(k,i);
        dox(k,i)=x(k,i+1)-x(k,i-1);
        doy(k,i)=y(k,i+1)-y(k,i-1);
    end
    F=F+znak(k)*sum(x1(k,2:N(k)+1).*dy(k,2:N(k)+1));
    Sx=Sx+znak(k)*sum(x(k,2:N(k)+1).*y(k,2:N(k)+1).*...
        doy(k,2:N(k)+1)+4*x1(k,2:N(k)+1).*...
        y1(k,2:N(k)+1).*dy(k,2:N(k)+1))/6;
    Sy=Sy+znak(k)*sum(x(k,2:N(k)+1).^2.*...
        doy(k,2:N(k)+1)+4*x1(k,2:N(k)+1).^2.*...
        dy(k,2:N(k)+1))/12;
    Jx=Jx+znak(k)*sum(x(k,2:N(k)+1).*...
        y(k,2:N(k)+1).^2.*doy(k,2:N(k)+1)...
        +4*x1(k,2:N(k)+1).*y1(k,2:N(k)+1).^2.*...
        dy(k,2:N(k)+1))/6;
    Jy=Jy+znak(k)*sum(x(k,2:N(k)+1).^3.*...
        doy(k,2:N(k)+1)+4*x1(k,2:N(k)+1).^3.*...
        dy(k,2:N(k)+1))/18;
    Jxy=Jxy+znak(k)*sum(x(k,2:N(k)+1).^2.*...
        y(k,2:N(k)+1).*doy(k,2:N(k)+1)+...
        4*x1(k,2:N(k)+1).^2.*y1(k,2:N(k)+1).*...
        dy(k,2:N(k)+1))/12;
end
fprintf('\n Общая площадь сечения, F=%12.4g \n',F);

```

```

fprintf('\n Статические моменты инерции сечения:');
fprintf('\n - относительно оси X, Sx=%12.4g',Sx);
fprintf('\n - относительно оси Y, Sy=%12.4g \n',Sy);
fprintf('\n Осевые моменты инерции сечения:');
fprintf('\n - относительно оси X, Jx=%12.4g',Jx);
fprintf('\n - относительно оси Y, Jy=%12.4g \n',Jy);
fprintf('\n Центробежный момент инерции сечения, ');
fprintf('\n Jxy=%12.4g \n',Jxy)

```

Результаты счета

Общая площадь сечения, $F=$ 16.53

Статические моменты инерции сечения:

- относительно оси X, $Sx=$ 44.11

- относительно оси Y, $Sy=$ 46.89

Осевые моменты инерции сечения:

- относительно оси X, $Jx=$ 154.1

- относительно оси Y, $Jy=$ 170.7

Центробежный момент инерции сечения,

$Jxy=$ 134.2

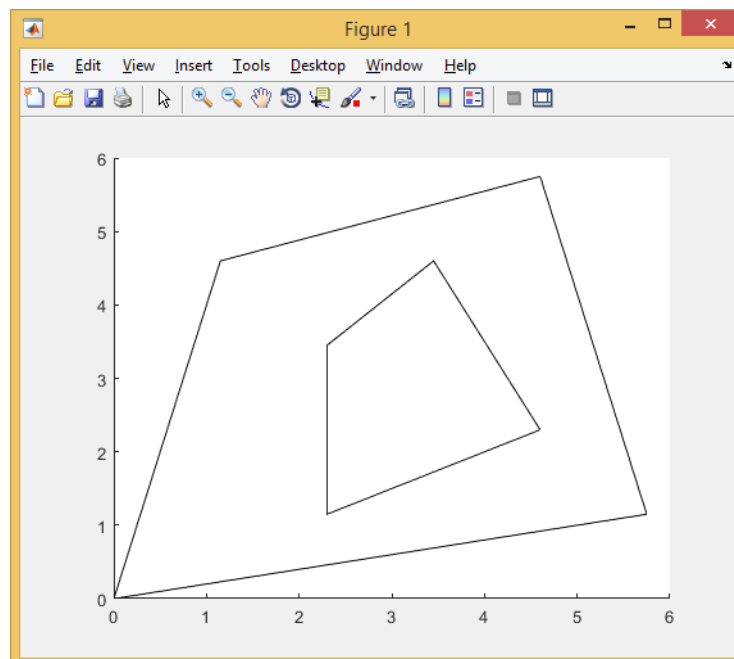


Рис. 5.2. Визуализация многоугольника в системе MATLAB

Пояснения к тексту программы.

Порядок ввода исходных данных для рассматриваемого примера показан ниже (для значений $g = 3$, $s = 12$):

Введите значение параметра $H=1.15$

Введите количество контуров $=2$

Порядковый номер контура $= 1$

Введите знак контура

(1 - внешний, -1 -внутренний) Знак контура $=1$

Введите количество вершин контура $=4$

Порядковый номер контура $= 2$

Введите знак контура

(1 - внешний, -1 - внутренний) Знак контура $=-1$

Введите количество вершин контура $=4$

Рассматривается контур номер 1

Введите абсциссы точек контура $= [0 \ 5 \ 4 \ 1]$

Введите ординаты точек контура $= [0 \ 1 \ 5 \ 4]$

Рассматривается контур номер 2

Введите абсциссы точек контура $= [2 \ 4 \ 3 \ 2]$

Введите ординаты точек контура $= [1 \ 2 \ 4 \ 3]$

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ 6.

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ИСПОЛЬЗОВАНИЯ СТАНДАРТНЫХ РЕШАТЕЛЕЙ ODE45 и ODE15s ДЛЯ РЕШЕНИЯ ЗАДАЧИ КОШИ

Задача Коши для обыкновенного дифференциального уравнения (ОДУ) n -го порядка имеет вид:

$$y^{(n)} = F(x, y, y', \dots, y^{(n-1)}) , \quad x > x_0 \quad (5.1)$$

где

$$\left. \begin{array}{l} y^{(n-1)}(x_0) = g_{n-1} \\ y^{(n-2)}(x_0) = g_{n-2} \\ \dots\dots\dots \\ y'(x_0) = g_1 \\ y(x_0) = g_0 \end{array} \right\} \text{ -- начальные условия.} \quad (5.2)$$

Задачу Коши для ОДУ n -го порядка можно свести к задаче Коши для системы n ОДУ 1-го порядка следующим образом. Введем $(n-1)$ дополнительные неизвестные функции по правилу:

$$z_1(x) = y'(x), \quad z_2(x) = y''(x), \quad \dots, \quad z_{n-1}(x) = y^{(n-1)}(x) . \quad (5.3)$$

Тогда вместо уравнения (5.1) получим систему ОДУ 1-го порядка:

$$\left. \begin{array}{l} z'_{n-1} = F(x, y, z_1, \dots, z_{n-1}) \\ z'_{n-2} = z_{n-1} \\ \dots\dots\dots \\ z'_1 = z_2 \\ y' = z_1 \end{array} \right\} , \quad (5.4)$$

Соответственно, начальные условия приводятся к виду:

$$\left. \begin{array}{l} z_{n-1}(x_0) = g_{n-1} \\ z_{n-2}(x_0) = g_{n-2} \\ \\ z_1(x_0) = g_1 \\ y(x_0) = g_0 \end{array} \right\} - \text{начальные условия.} \quad (5.5)$$

Пример. Задача Коши

$$\left\{ \begin{array}{l} y^{(4)} - \frac{1}{2}xy''' + 2y = 2x^4 - 12x^2 - 4x + 26, \quad 0 < x < 2 \\ y(0) = 1 \\ y'(0) = -2 \\ y''(0) = 0 \\ y'''(0) = 0 \end{array} \right\} - \text{начальные условия}$$

Сведем задачу Коши для ОДУ 4-го порядка к задаче Коши для системы 4-х ОДУ 1-го порядка:

$$\left\{ \begin{array}{l} \bar{z}' = \bar{F}(x, \bar{z}), \quad 0 < x < 2 \\ \bar{z}(0) = \bar{z}_0 \end{array} \right.,$$

где

$$\bar{z}(x) = \begin{bmatrix} z_1(x) \\ z_2(x) \\ z_3(x) \\ z_4(x) \end{bmatrix}, \quad z_1 = y, \quad z_2 = y', \quad z_3 = y'', \quad z_4 = y''', \quad \bar{z}_0 = \begin{bmatrix} 1 \\ -2 \\ 0 \\ 0 \end{bmatrix}$$

$$\bar{F}(x, \bar{z}) = \begin{bmatrix} z_2 \\ z_3 \\ z_4 \\ 2x^4 - 12x^2 - 4x + 26 + \frac{1}{2}xz_4 - 2z_1 \end{bmatrix}$$

Использование встроенной функции ode45.

В данной функции реализуются одношаговые методы Рунге-Кутты 4-го и 5-го порядков. Это классический подход, рекомендуемый для начальной пробы решения. Во многих случаях (если система уравнений не является жесткой) можно получить удовлетворительные результаты. Таким образом, данную функцию применяют довольно часто, особенно тогда, когда неизвестны характеристики задачи.

При обращении к `ode45` входными параметрами будут дескриптор имени функции, реализующей вычисление вектор-функции $\bar{F}(x, \bar{z})$, массив, содержащий значения левого и правого конца диапазона изменения переменной x , а также вектор начальных значений $\bar{z}_0$. Выходными параметрами будут массивы X и Y , где

$$X(i) = x_i, \quad Y(i,1) = z_1(x_i) = y(x_i), \quad Y(i,2) = z_2(x_i) = y'(x_i), \quad i = 1, \dots, N.$$

Пример М-файла

```
x0xL=input(' [x0 xL]= ');
z0=input(' [z10 z20 z30 z40]= ');
F=@(x,z) [z(2);z(3);z(4);2*x^4-12*x^2-4*x+26+x*z(4)/2-2*z(1)];
[X Z]=ode45(F,x0xL,z0);
n=length(X);
fprintf('\n'),fprintf('%3s %6s %10s %10s %10s\n',
'i','x','y','dy','d2y','d3y'),fprintf('\n')
for i=1:n
    fprintf('%3d %6.4f %10.4f %10.4f %10.4f\n',i,X(i),Z(i,:))
end
plot(X,Z(:,1),X,Z(:,2),X,Z(:,3),X,Z(:,4))
grid on
legend('y(x)', 'dy(x)', 'd2y(x)', 'd3y(x)',0),title('ode45')
```

Результаты счета

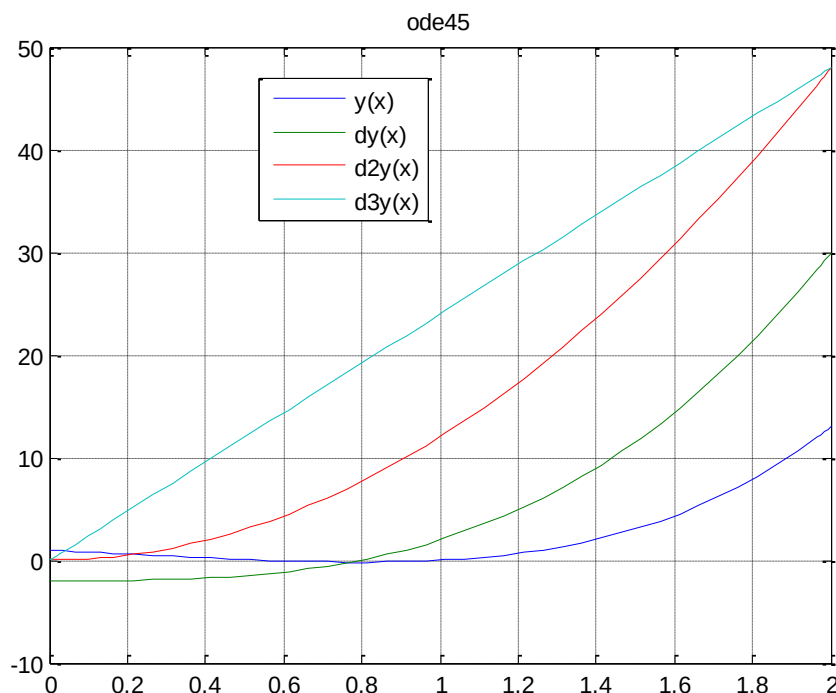
[x0 xL]=[0 2]

[z10 z20 z30 z40]=[1 -2 0 0]

i	x	y	dy	d2y	d3y
1	0.0000	1.0000	-2.0000	0.0000	0.0000
2	0.0000	1.0000	-2.0000	0.0000	0.0001
3	0.0000	1.0000	-2.0000	0.0000	0.0001
4	0.0000	1.0000	-2.0000	0.0000	0.0002
5	0.0000	1.0000	-2.0000	0.0000	0.0002
6	0.0000	1.0000	-2.0000	0.0000	0.0005
7	0.0000	0.9999	-2.0000	0.0000	0.0007
8	0.0000	0.9999	-2.0000	0.0000	0.0010
9	0.0001	0.9999	-2.0000	0.0000	0.0012
10	0.0001	0.9998	-2.0000	0.0000	0.0025
11	0.0002	0.9997	-2.0000	0.0000	0.0037
12	0.0002	0.9996	-2.0000	0.0000	0.0050
13	0.0003	0.9995	-2.0000	0.0000	0.0062
14	0.0005	0.9990	-2.0000	0.0000	0.0125
15	0.0008	0.9984	-2.0000	0.0000	0.0188
16	0.0010	0.9979	-2.0000	0.0000	0.0251
17	0.0013	0.9974	-2.0000	0.0000	0.0313
18	0.0026	0.9948	-2.0000	0.0001	0.0627
19	0.0039	0.9922	-2.0000	0.0002	0.0941
20	0.0052	0.9895	-2.0000	0.0003	0.1255
21	0.0065	0.9869	-2.0000	0.0005	0.1569
22	0.0131	0.9738	-2.0000	0.0021	0.3139
23	0.0196	0.9608	-2.0000	0.0046	0.4709
24	0.0262	0.9477	-1.9999	0.0082	0.6279
25	0.0327	0.9346	-1.9999	0.0128	0.7849
26	0.0654	0.8692	-1.9989	0.0513	1.5699
27	0.0981	0.8039	-1.9962	0.1155	2.3548
28	0.1308	0.7386	-1.9910	0.2054	3.1398
29	0.1635	0.6737	-1.9825	0.3209	3.9248
30	0.2135	0.5750	-1.9611	0.5472	5.1248

31	0.2635	0.4778	-1.9268	0.8334	6.3248
32	0.3135	0.3826	-1.8767	1.1796	7.5248
33	0.3635	0.2904	-1.8078	1.5859	8.7248
34	0.4135	0.2022	-1.7171	2.0521	9.9248
35	0.4635	0.1191	-1.6016	2.5783	11.1248
36	0.5135	0.0425	-1.4583	3.1646	12.3248
37	0.5635	-0.0262	-1.2842	3.8108	13.5248
38	0.6135	-0.0854	-1.0762	4.5171	14.7248
39	0.6635	-0.1332	-0.8315	5.2833	15.9248
40	0.7135	-0.1679	-0.5469	6.1095	17.1248
41	0.7635	-0.1872	-0.2195	6.9958	18.3248
42	0.8135	-0.1890	0.1537	7.9420	19.5248
43	0.8635	-0.1710	0.5757	8.9482	20.7248
44	0.9135	-0.1306	1.0495	10.0145	21.9248
45	0.9635	-0.0651	1.5781	11.1407	23.1248
46	1.0135	0.0282	2.1646	12.3270	24.3248
47	1.0635	0.1523	2.8118	13.5732	25.5248
48	1.1135	0.3104	3.5229	14.8794	26.7248
49	1.1635	0.5057	4.3008	16.2457	27.9248
50	1.2135	0.7417	5.1485	17.6719	29.1248
51	1.2635	1.0218	6.0690	19.1581	30.3248
52	1.3135	1.3498	7.0653	20.7044	31.5248
53	1.3635	1.7296	8.1404	22.3106	32.7248
54	1.4135	2.1652	9.2974	23.9769	33.9248
55	1.4635	2.6608	10.5391	25.7031	35.1248
56	1.5135	3.2206	11.8687	27.4893	36.3248
57	1.5635	3.8492	13.2890	29.3356	37.5248
58	1.6135	4.5511	14.8032	31.2418	38.7248
59	1.6635	5.3311	16.4142	33.2080	39.9247
60	1.7135	6.1942	18.1250	35.2343	41.1248
61	1.7635	7.1453	19.9386	37.3205	42.3248
62	1.8135	8.1898	21.8581	39.4668	43.5248
63	1.8635	9.3329	23.8863	41.6730	44.7247
64	1.9135	10.5803	26.0264	43.9392	45.9248
65	1.9635	11.9375	28.2812	46.2655	47.1248
66	1.9726	12.1973	28.7050	46.6961	47.3436

67	1.9818	12.4609	29.1327	47.1287	47.5624
68	1.9909	12.7285	29.5644	47.5634	47.7812
69	2.0000	13.0000	30.0000	48.0000	48.0000



Использование встроенной функции `ode15s`.

В данной функции реализуется многошаговый метод переменного порядка (от **одного** до **пяти**), использующий формулы численного дифференцирования «назад». Это адаптивный метод, рекомендуемый к применению в случае, когда `ode45` не обеспечивает решения (система уравнений жесткая). При этом, входные и выходные параметры при обращении к `ode15s` полностью совпадают с соответствующими параметрами функции `ode45`.

Пример М-файла

```
x0xL=input(' [x0 xL]= ');
z0=input(' [z10 z20 z30 z40]= ');
F=@(x,z) [z(2);z(3);z(4);2*x^4-12*x^2-4*x+26+x*z(4)/2-2*z(1)];
[X Z]=ode15s(F,x0xL,z0);
n=length(X);
```

```

fprintf('\n'),fprintf('%3s %6s %10s %10s %10s
%10s','i','x','y','dy','d2y','d3y'),fprintf('\n')
for i=1:n
    fprintf('%3d %6.4f %10.4f %10.4f %10.4f
%10.4f\n',i,X(i),Z(i,:))
end
plot(X,Z(:,1),X,Z(:,2),X,Z(:,3),X,Z(:,4))
grid on
legend('y(x)', 'dy(x)', 'd2y(x)', 'd3y(x)',0),title('ode15s')

```

Результаты счета

```

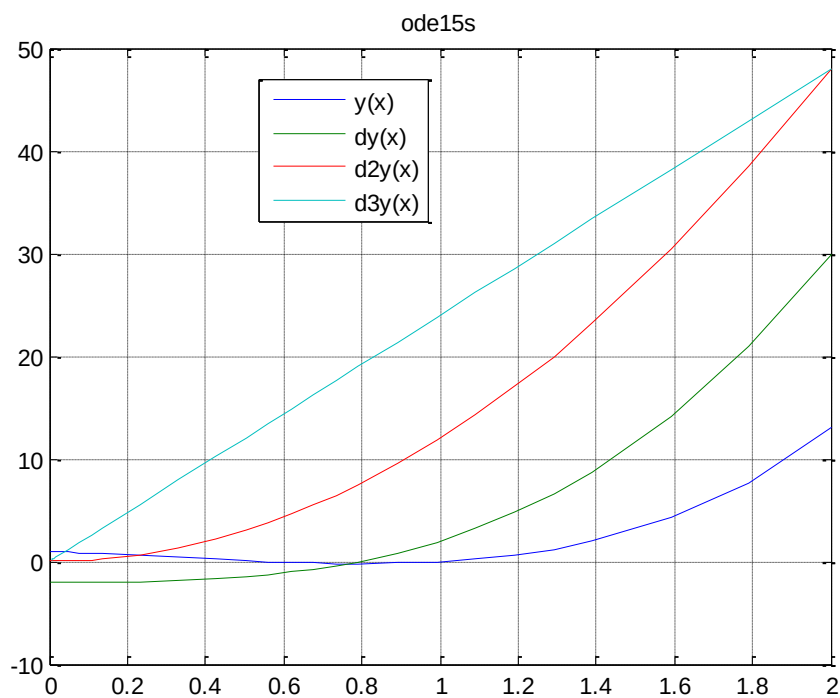
[x0 xL]=[0 2]
[z10 z20 z30 z40]=[1 -2 0 0]

```

i	x	y	dy	d2y	d3y
1	0.0000	1.0000	-2.0000	0.0000	0.0000
2	0.0002	0.9995	-2.0000	0.0000	0.0055
3	0.0005	0.9991	-2.0000	0.0000	0.0111
4	0.0007	0.9986	-2.0000	0.0000	0.0166
5	0.0017	0.9966	-2.0000	0.0000	0.0408
6	0.0027	0.9946	-2.0000	0.0001	0.0650
7	0.0037	0.9926	-2.0000	0.0002	0.0892
8	0.0047	0.9906	-2.0000	0.0003	0.1134
9	0.0077	0.9846	-2.0000	0.0007	0.1851
10	0.0107	0.9786	-2.0000	0.0014	0.2569
11	0.0137	0.9726	-2.0000	0.0023	0.3287
12	0.0167	0.9666	-2.0000	0.0033	0.4004
13	0.0466	0.9068	-1.9995	0.0260	1.1181
14	0.0765	0.8471	-1.9980	0.0702	1.8358
15	0.1064	0.7874	-1.9949	0.1358	2.5535
16	0.1363	0.7278	-1.9894	0.2229	3.2712
17	0.2331	0.5371	-1.9487	0.6520	5.5941
18	0.3299	0.3527	-1.8558	1.3058	7.9170
19	0.4267	0.1808	-1.6887	2.1845	10.2397
20	0.5036	0.0582	-1.4886	3.0430	12.0854

21	0.5612	-0.0222	-1.2922	3.7799	13.4694
22	0.6189	-0.0900	-1.0511	4.5966	14.8534
23	0.6766	-0.1424	-0.7605	5.4932	16.2373
24	0.7343	-0.1766	-0.4159	6.4695	17.6213
25	0.7919	-0.1892	-0.0128	7.5256	19.0052
26	0.8917	-0.1497	0.8370	9.5421	21.4004
27	0.9916	-0.0150	1.9000	11.7976	23.7954
28	1.0914	0.2374	3.2000	14.2921	26.1905
29	1.1912	0.6324	4.7609	17.0257	28.5855
30	1.2910	1.1973	6.6066	19.9984	30.9804
31	1.3908	1.9615	8.7610	23.2101	33.3754
32	1.5908	4.2240	14.1025	30.3650	38.1741
33	1.7908	7.7043	20.9710	38.4797	42.9725
34	2.0000	13.0011	29.9981	47.9947	47.9914

>>



Замечание.

Точность вычисления для функций-решателей **ode45** и **ode15s** , а также другим подобным функциям устанавливается по умолчанию и заключена в значениях следующих параметров:

➤ `RelTol` – относительная погрешность (скаляр), равный по умолчанию $1e-3$ (т.е. точность составляет 0.1%) во всех решателях; оценка ошибки на каждом шаге решения (интегрирования) имеет вид:

$$e(i) < \max\{\text{RelTol} * \text{abs}(y(i)), \text{AbsTol}(i)\} ;$$

➤ `AbsTol` – абсолютная погрешность (скаляр или вектор), равная по умолчанию $1e-6$ (скаляр, если эта погрешность одинакова для всех компонентов решения и вектор, когда она разная).

Имеется возможность вносить изменения с помощью дополнительного параметра `options`, создаваемого функцией `odeset`

Задание.

Численно решить задачу Коши с помощью стандартных решателей `ode45` и `ode15s`.

Номер варианта определяется по двум последним цифрам номера зачетки. Если это число попадает в диапазон 31-60, то из него необходимо вычесть 30, если в диапазон 61-90, то вычесть 60, если в диапазон 91-99, то вычесть 90, для номера 00 использовать вариант №10.

Варианты заданий:

$$1.) \begin{cases} y''' + (y'')^2 = x(26 + 140x^3 + y'), & -1 < x < 1 \\ y(-1) = 4 \\ y'(-1) = -6 \\ y''(-1) = 12 \end{cases}$$

$$2.) \begin{cases} y^{(4)} - \frac{1}{x} y''' + \frac{1}{2x^2} y'' + y = x^4 - 2x + 7, & 1 < x < 2 \\ y(1) = 0 \\ y'(1) = 2 \\ y''(1) = 12 \\ y'''(1) = 24 \end{cases}$$

$$3) \begin{cases} y''' + x^2 y'' = 3x(10 + y'), & 1 < x < 2 \\ y(1) = 0 \\ y'(1) = 2 \\ y''(1) = 12 \end{cases}$$

$$4) \begin{cases} y'' = (x+1)y' - 4(x^3 + y) + 6(2x^2 - x + 1), & -2 < x < 0 \\ y(-2) = 21 \\ y'(-2) = -34 \end{cases}$$

$$5) \begin{cases} y''' + 2xy' = (2x-1)\cos x + 4(2-x)\sin 2x + x^2(\sin x + \cos 2x - y), & -\frac{\pi}{2} < x < \pi \\ y(-\frac{\pi}{2}) = -2 \\ y'(-\frac{\pi}{2}) = 0 \\ y''(-\frac{\pi}{2}) = 5 \end{cases}$$

$$6) \begin{cases} y'' + xy' + y^2 = x(\cos x - 2\sin x) + (\sin x + \cos 2x)^2 - \sin x - 4\cos 2x, & 0 < x < \frac{3\pi}{2} \\ y(0) = 1 \\ y'(0) = 1 \end{cases}$$

$$7) \begin{cases} y''' + y'' \sin x - y' \cos x = \cos x(2 \sin 2x - 1) - 4 \sin x \cos 2x + 8 \sin 2x - 1, & -\pi < x < 0 \\ y(-\pi) = 1 \\ y'(-\pi) = -1 \\ y''(-\pi) = -4 \end{cases}$$

$$8) \begin{cases} y'' + y'y + 3 \cos 2x = (\cos 2x + \sin x)(\cos x - 2 \sin 2x - 1), & -\frac{3\pi}{2} < x < \frac{\pi}{2} \\ y(-\frac{3\pi}{2}) = 0 \\ y'(-\frac{3\pi}{2}) = 0 \end{cases}$$

$$9) \begin{cases} y''' + yy'' + y' - 6 \sin 2x = -(\sin x + \cos 2x)(\sin x + 4 \cos 2x), & 0 < x < \pi \\ y(0) = 1 \\ y'(0) = 1 \\ y''(0) = -4 \end{cases}$$

$$10) \begin{cases} y^{(4)} - \frac{1}{2}(y''')^2 + 2y(y+2) = -2e^{2x} \cos 2x, & 0 < x < 1 \\ y(0) = 1 \\ y'(0) = 1 \\ y''(0) = 0 \\ y'''(0) = -2 \end{cases}$$

$$11) \begin{cases} y^{(4)} + y''' + 2y' + 4(y + e^x \sin x) = 0, & 0 < x < 2 \\ y(0) = 1 \\ y'(0) = 1 \\ y''(0) = 0 \\ y'''(0) = -2 \end{cases}$$

$$12) \begin{cases} y''' + 2y - y''(1+y) = e^{2x} \sin 2x, & 0 < x < 1 \\ y(0) = 1 \\ y'(0) = 1 \\ y''(0) = 0 \end{cases}$$

$$13) \begin{cases} y'' - 2y' + y + e^x \cos x = 0, & 0 < x < 1 \\ y(0) = 1 \\ y'(0) = 1 \end{cases}$$

$$14) \begin{cases} y^{(5)} + 2y''' + y' = 2x + 8(\sin x + \cos x), & 0 < x < \pi \\ y(0) = 0 \\ y'(0) = 0 \\ y''(0) = 4 \\ y'''(0) = 3 \\ y^{(4)}(0) = 10 \end{cases}$$

$$15) \begin{cases} y'' - 4y' = \cos x, & 0 < x < 1 \\ y(0) = 1 \\ y'(0) = 1 \end{cases}$$

$$16) \begin{cases} y'' = (4x+1)xy' + 2y, & 0 < x < 1 \\ y(0) = 0 \\ y'(0) = 1 \end{cases}$$

$$17) \begin{cases} y^{(6)} + y = \sin \frac{3x}{2} \sin \frac{x}{2}, & 0 < x < \pi \\ y(0) = 0 \\ y'(0) = 0 \\ y''(0) = 4 \\ y'''(0) = 3 \\ y^{(4)}(0) = 2 \\ y^{(5)}(0) = 1 \end{cases}$$

$$18) \begin{cases} y'' - \frac{2}{x}y' + 4xy = 0, & 1 < x < 2 \\ y(1) = 0 \\ y'(1) = 1 \end{cases}$$

$$19) \begin{cases} y'' - 4xy = 1, & 0 < x < 1 \\ y(0) = 1 \\ y'(0) = 1 \end{cases}$$

$$20) \begin{cases} y^{(4)} - 12y'' + 12y = 16x^4 e^{x^2}, & 0 < x < 1 \\ y(0) = 1 \\ y'(0) = 1 \\ y''(0) = 0 \\ y'''(0) = -2 \end{cases}$$

$$21) \begin{cases} y^{(4)} + 5y'' + 16y = 0, & 0 < x < \pi \\ y(0) = 0 \\ y'(0) = 1 \\ y''(0) = 2 \\ y'''(0) = 3 \end{cases}$$

$$22) \begin{cases} y^{(4)} + 2y''' - 3y'' - 4y' + 4y = 32\sin 2x - 24\cos 2x, & 0 < x < 1 \\ y(0) = 0 \\ y'(0) = 1 \\ y''(0) = 6 \\ y'''(0) = 10 \end{cases}$$

$$23) \begin{cases} 4y^{(4)} - 12y''' + 11y'' - 3y' = 4\cos x, & 0 < x < 1 \\ y(0) = 0 \\ y'(0) = 1 \\ y''(0) = 6 \\ y'''(0) = 10 \end{cases}$$

$$24) \begin{cases} y''' + \frac{5}{x}y'' + \frac{4}{x^2}y' = \frac{\ln x}{x^2}, & 1 < x < 2 \\ y(1) = 0 \\ y'(1) = 1 \\ y''(1) = 2 \end{cases}$$

$$25) \begin{cases} y''' + 2x^3y = 2x, & 0 < x < 1 \\ y(0) = 1 \\ y'(0) = 2 \\ y''(0) = 3 \end{cases}$$

$$26) \begin{cases} y''' - 4y'' = e^{4x} \sin^2 x, & 0 < x < 1 \\ y(0) = 1 \\ y'(0) = 2 \\ y''(0) = 3 \end{cases}$$

$$27) \begin{cases} y''' - 3y'' + 3y' - y = e^x, & 0 < x < 1 \\ y(0) = 1 \\ y'(0) = 2 \\ y''(0) = 10 \end{cases}$$

$$28) \begin{cases} 4y''' - 8y'' - 11y' - 3y = -18e^x, & 0 < x < 1 \\ y(0) = 1 \\ y'(0) = 4 \\ y''(0) = 8 \end{cases}$$

$$29) \begin{cases} y''' + \frac{3}{2x} y'' + y = 4, & 1 < x < 2 \\ y(1) = 0 \\ y'(1) = 1 \\ y''(1) = 1 \end{cases}$$

$$30) \begin{cases} y''' + \frac{2}{x} y'' - (x+1)y' = 8, & 1 < x < 2 \\ y(1) = -1 \\ y'(1) = 2 \\ y''(1) = 1 \end{cases}$$

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ 7.

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ИСПОЛЬЗОВАНИЯ СТАНДАРТНОГО РЕШАТЕЛЯ BVP4C ДЛЯ РЕШЕНИЯ КРАЕВОЙ ЗАДАЧИ

Простейшая форма обращения к функции `bvp4c` имеет вид:

```
sol=bvp4c(odefun,bcfun,solinit)
```

- **odefun** – функция, вычисляющая вектор правых частей;
- **bcfun** – функция, вычисляющая вектор граничных условий (аргументами этой функции являются **ya** и **yb** – векторы решений **y** в точках **a** и **b**; две компоненты вектора граничных условий представляют собой выражения, обращающиеся в нуль в точках **a** и **b** соответственно;
- **solinit** – выходная структура функции **bvpinit**;
- **sol** – аналогичная структура, содержащая решение краевой задачи (помимо полей **sol.x** и **sol.y** она имеет поле **sol.yp**, в котором содержатся значения производной решения **(sol.y)'** в точках **sol.x**).

Задание начальных условий

Начальные значения **xi** и **yi** задаются с помощью функции **bvpinit**:

```
solinit=bvpinit(xinit,yinit),
```

где

xinit – вектор-строка **xinit(1)=a<xinit(2)<...<xinit(n)=b**,

который может быть задан с помощью функции **linspace**:

```
xinit=linspace(a,b,n);
```

yinit – гипотетические значения для **y(i)**.

Сглаживание результата

В случае недостаточного количества точек в векторе результата **sol.x** для обеспечения гладкости результата применяется функция **deval** с использованием интерполяционных сплайнов для заданного вектора пробных точек **xx**.

Обращение к функции **deval** имеет вид:

$$\mathbf{yy} = \text{deval}(\mathbf{sol}, \mathbf{xx}),$$

где

sol – выходная структура функции **bvp4c**;

xx – вектор пробных точек, который может быть задан с помощью функции **linspace**: **xx=linspace(a,b,m)**, при этом, если число точек **m** пропущено, оно по умолчанию берется равным 100;

yy – значения сплайн-функции в пробных точках **xx**.

Пример решения.

$$\begin{cases} y'' + 3xy' + (2 + |x|)y + x - 2 = 0, & |x| < 3 \\ y(-3) + 2y'(-3) = 0 \\ y(3) = 2 \end{cases} \text{ - краевые условия}$$

Переходим от дифференциального уравнения 2-го порядка к системе двух уравнений 1-го порядка

$$\begin{cases} z_1 = y \\ z_2 = y' \end{cases}$$

Тогда исходная краевая задача преобразуется к виду

$$\begin{cases} z_1' = z_2 \\ z_2' = -3xz_2 - (2 + |x|)z_1 - x + 2 \end{cases}, \quad -3 < x < 3 \\ \begin{cases} z_1(-3) + 2z_2(-3) = 0 \\ z_1(3) - 2 = 0 \end{cases} \text{ - краевые условия} \end{cases}$$

Пример М-файла

```
xg=input(' [a b]=');  
z_gip=input(' [z1_gip z2_gip]=');  
n=input(' ввести начальное количество точек на интервале [a b] n=');  
solinit = bvpinit(linspace(xg(1),xg(2),n),z_gip);  
ode1=@(x,z) [z(2);-3*x*z(2)-(2+abs(x))*z(1)-x+2];  
bc1=@(za,zb) [za(1)+2*za(2)-0;zb(1)-2];  
sol = bvp4c(ode1,bc1,solinit);  
x = linspace(xg(1),xg(2));  
y = deval(sol,x);  
figure(),plot(x,y(1,:),x,y(2,:)),grid on, legend('y','dy');
```

Результаты счета

```
[a b]=[-3 3]
```

```
[z1_gip z2_gip]=[0 0]
```

```
ввести начальное количество точек на интервале [a b] n=5
```

