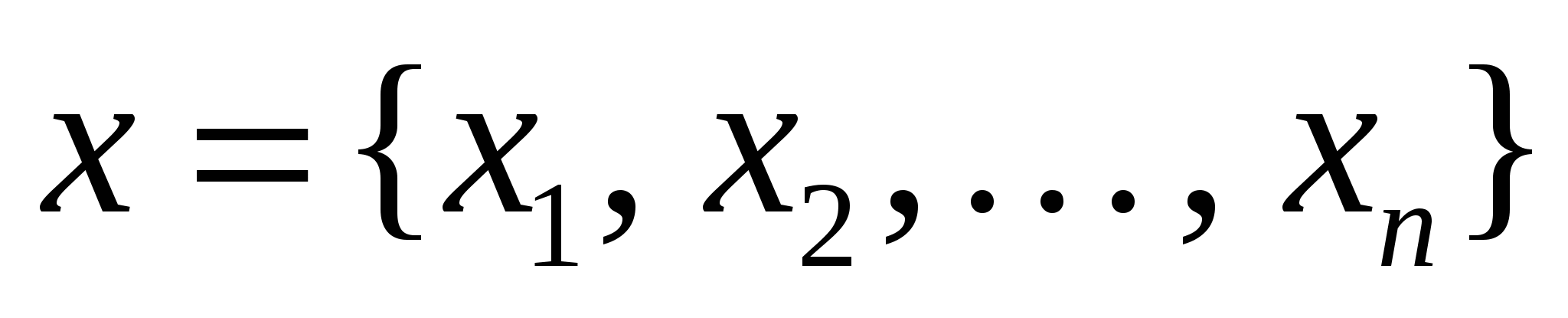
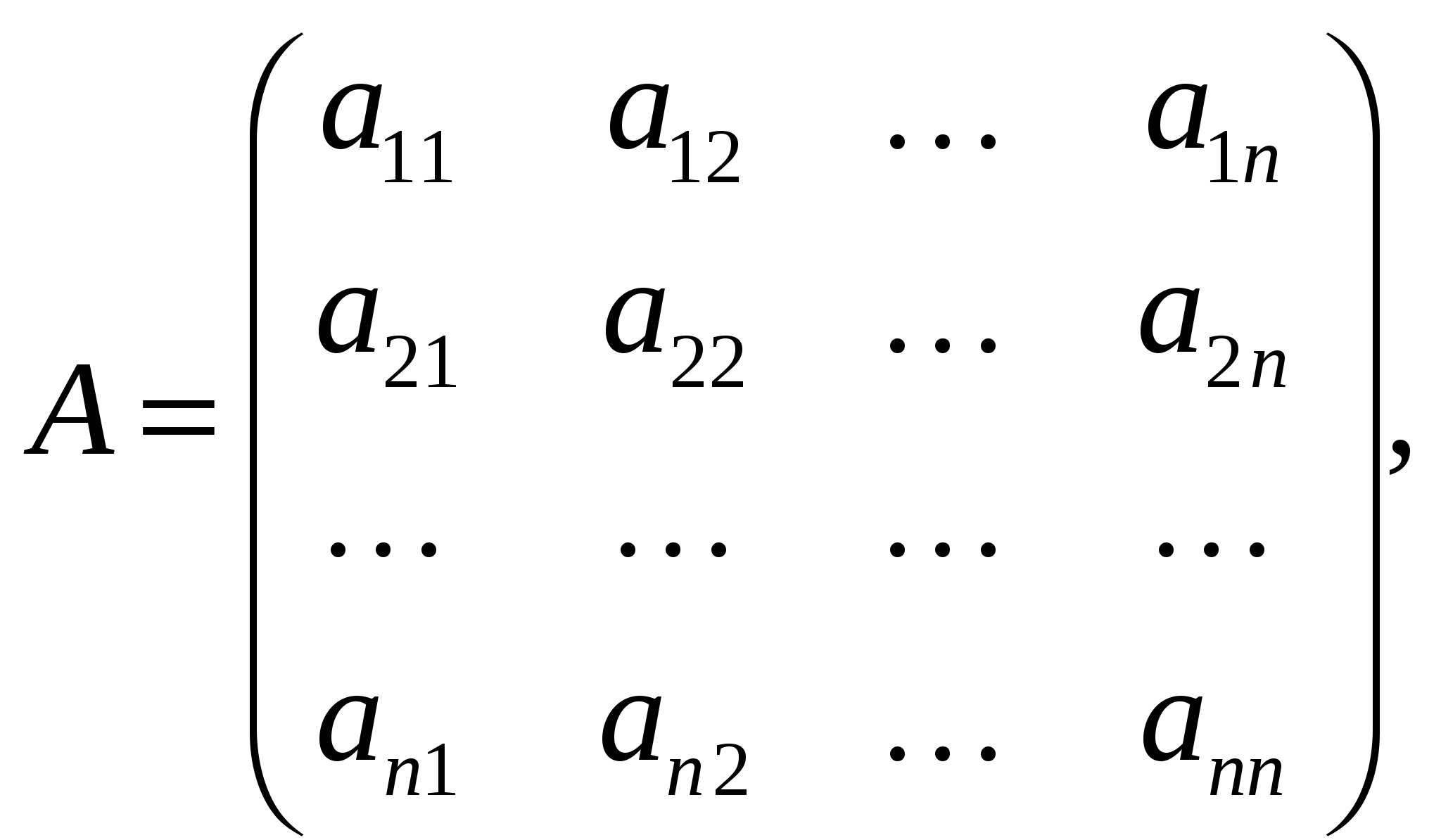
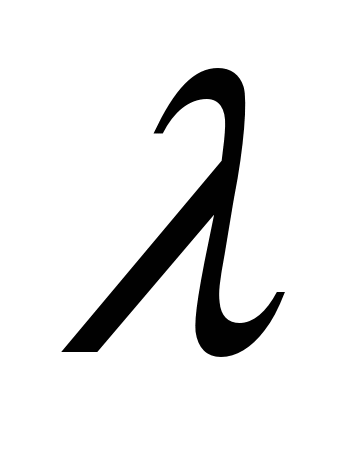
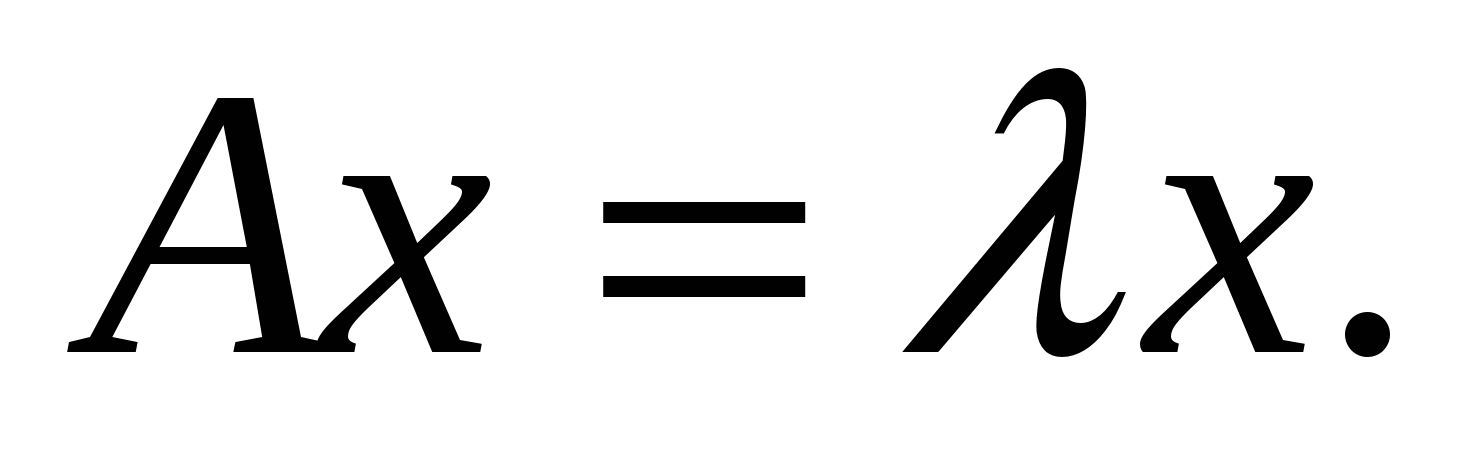
**МЕТОД ЛЕВЕРЬЕ-ФАДЕЕВА**

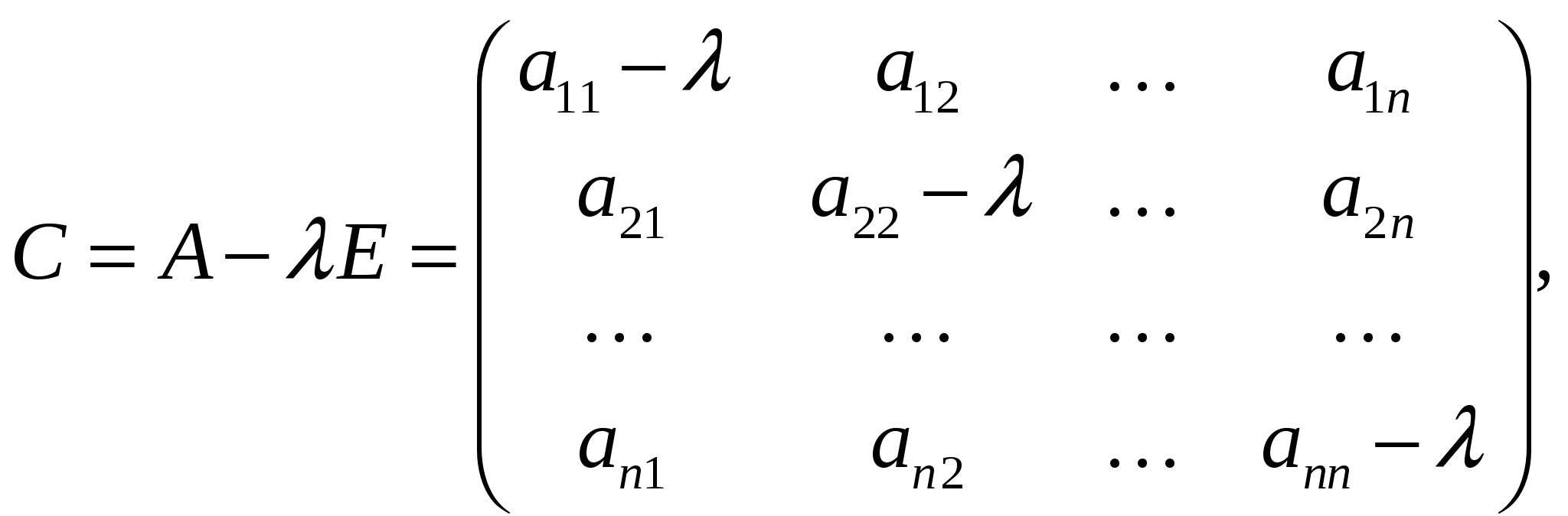
Вектор называется собственным вектором матрицы



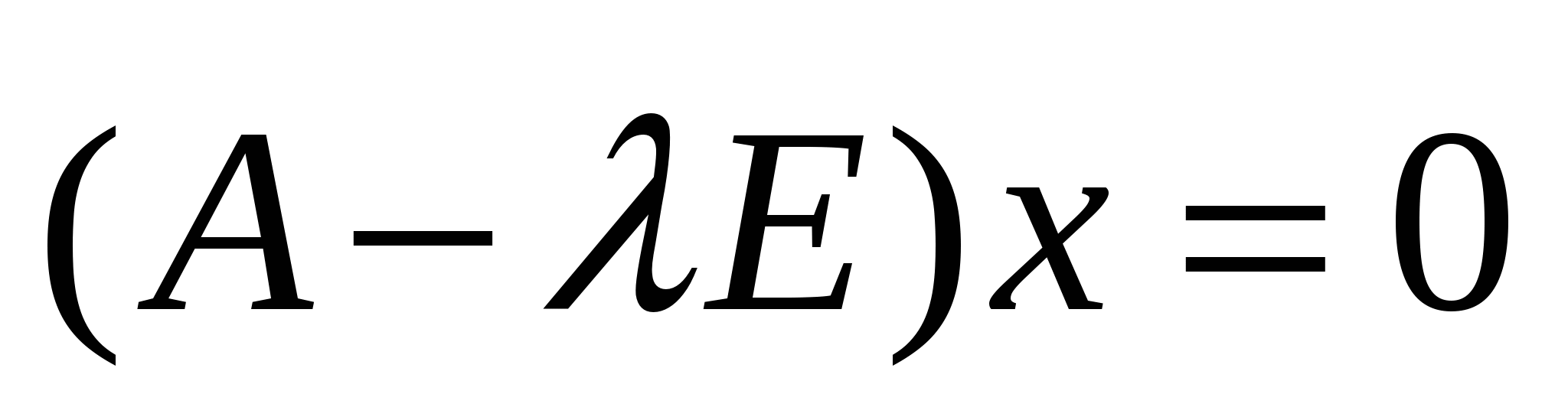
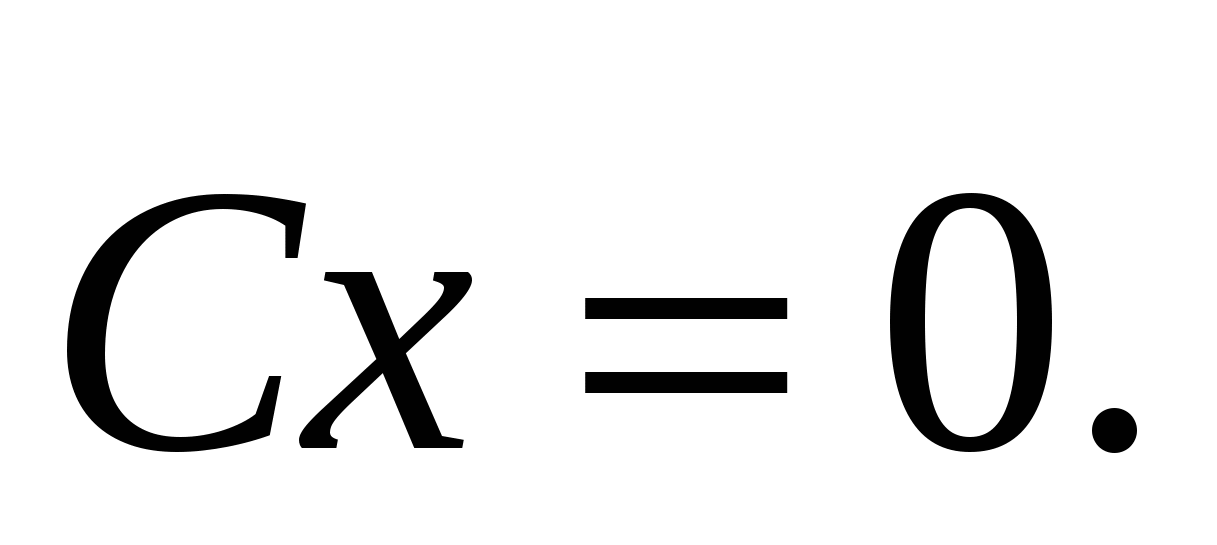
соответствующим собственному значению , если он удовлетворяет системе уравнений

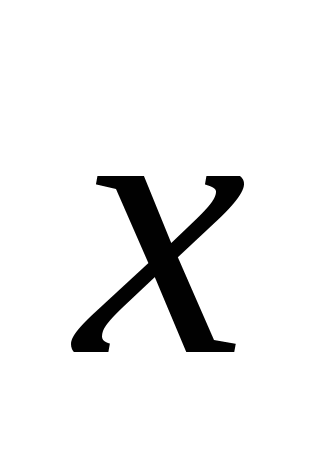
(1)

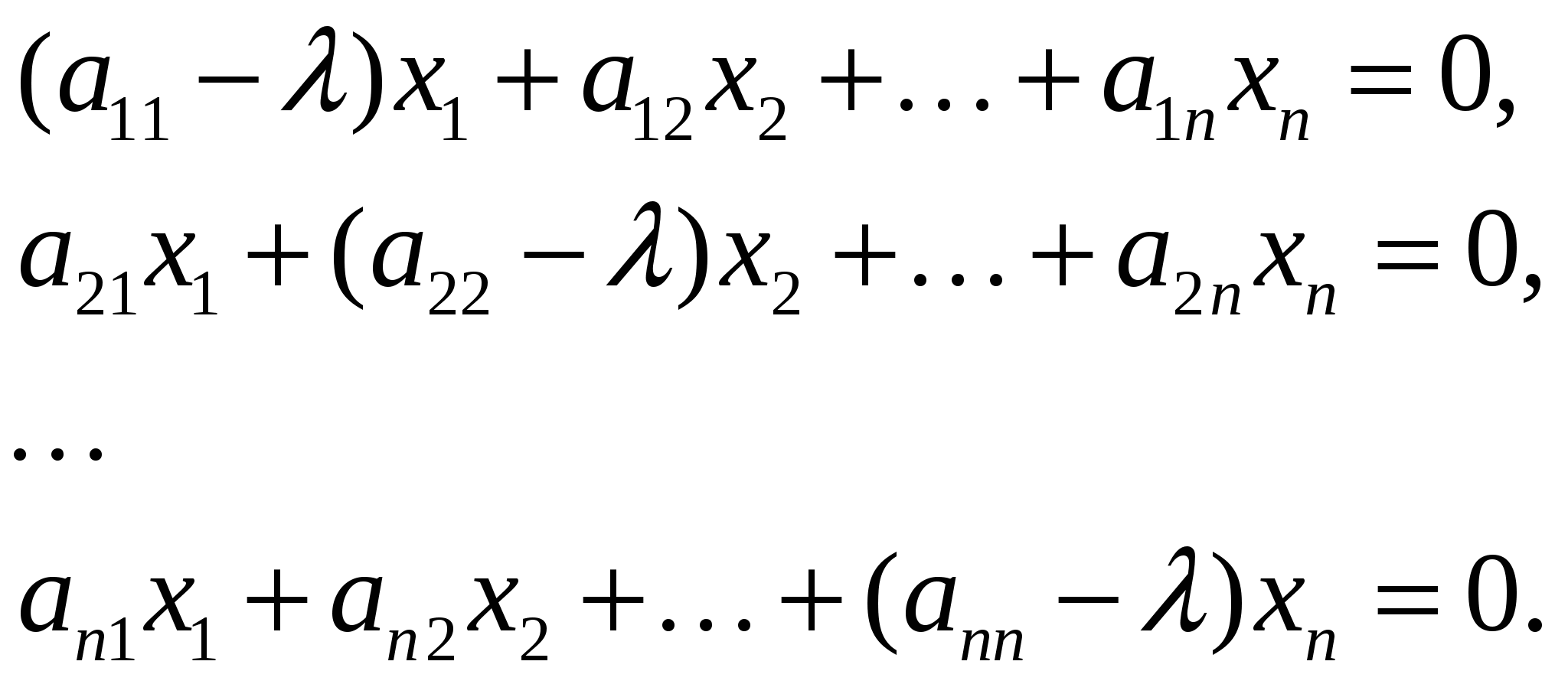
Характеристической матрицей *C* данной матрицы *A* называется матрица вида

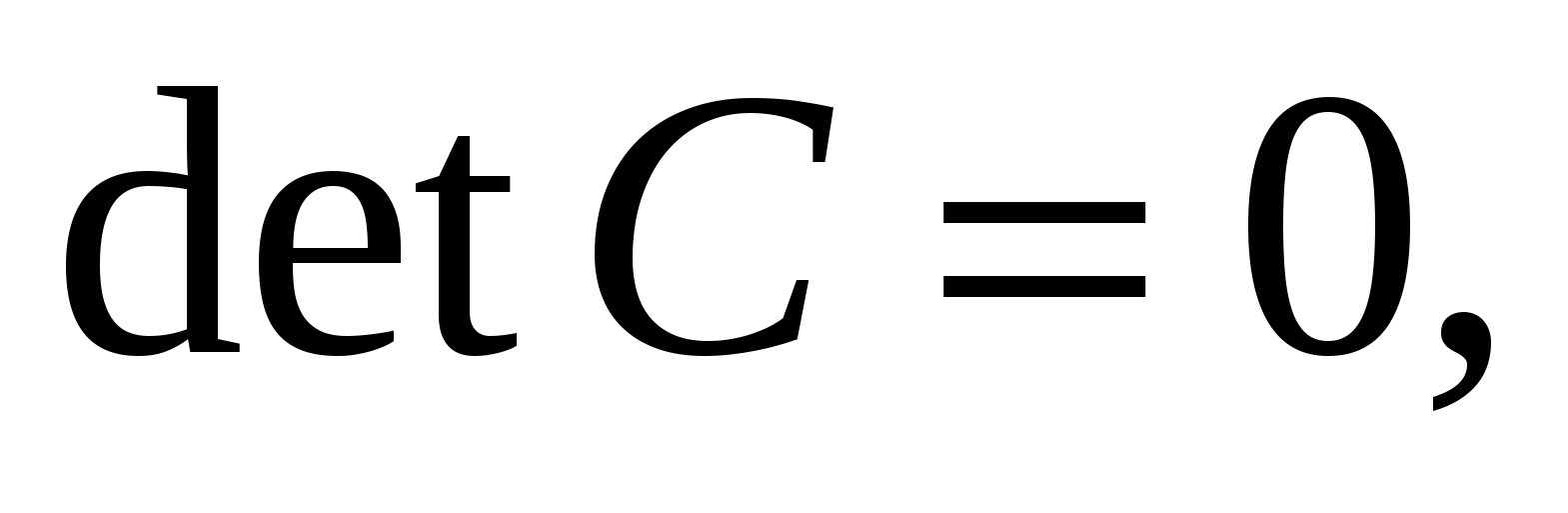
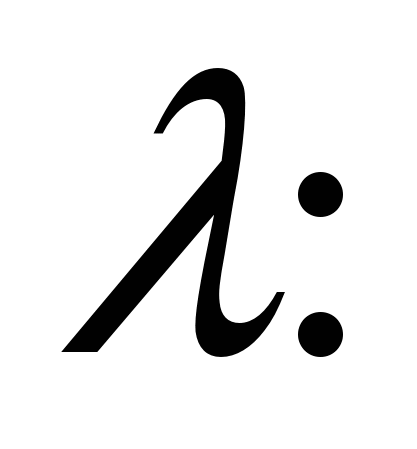


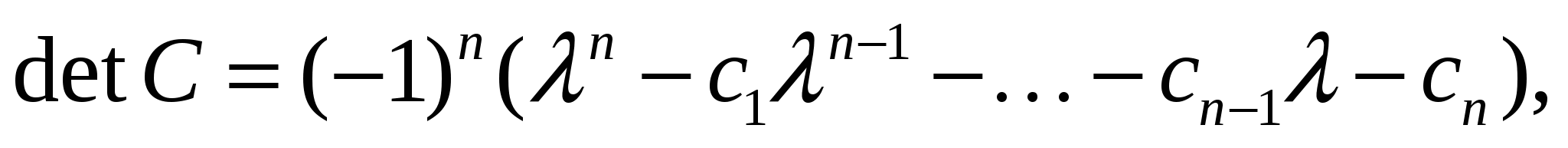
где *E*– единичная матрица. Легко видеть, что систему (1) можно записать в виде

или(2)

Если перейти к координатной форме записи вектора , то систему (2) можно представить, как

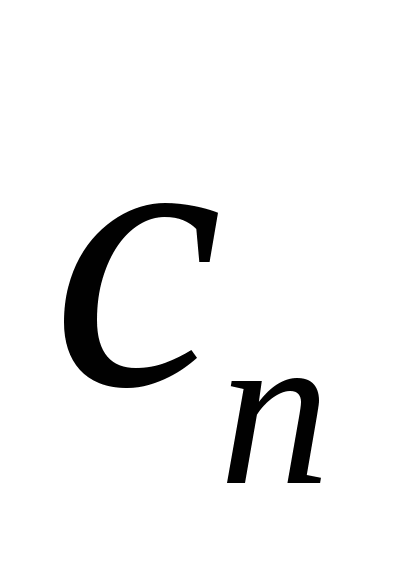
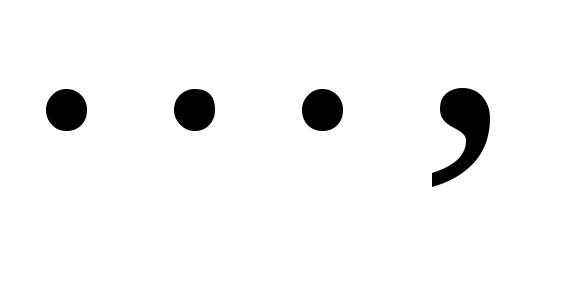
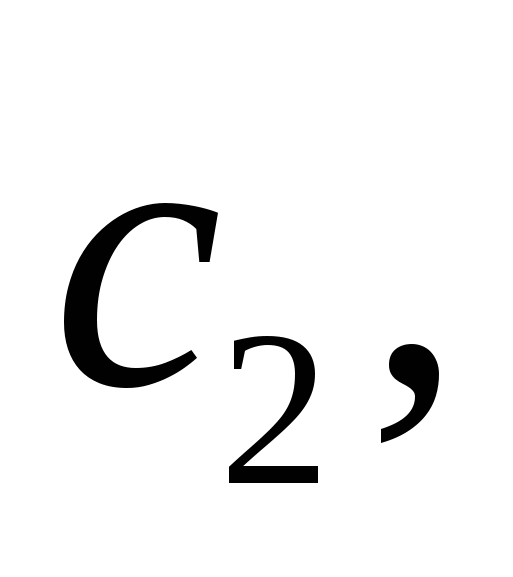
(3)

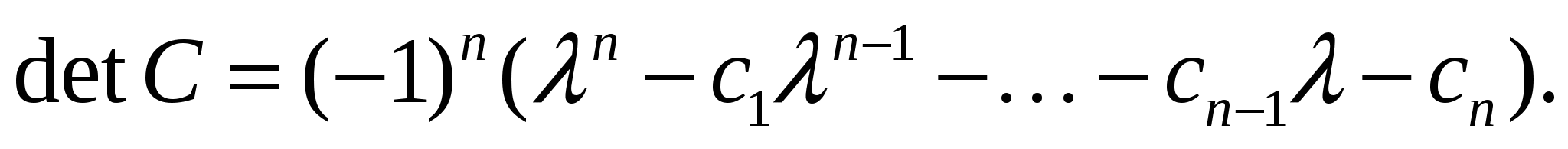
Система (2) или (3) является однородной системой *n* линейных уравнений с *n* неизвестными. Она имеет ненулевые решения лишь тогда, когда ее определитель равен нулю:причем решение не единственно. Определитель *C* является многочленом *n*-й степени относительно

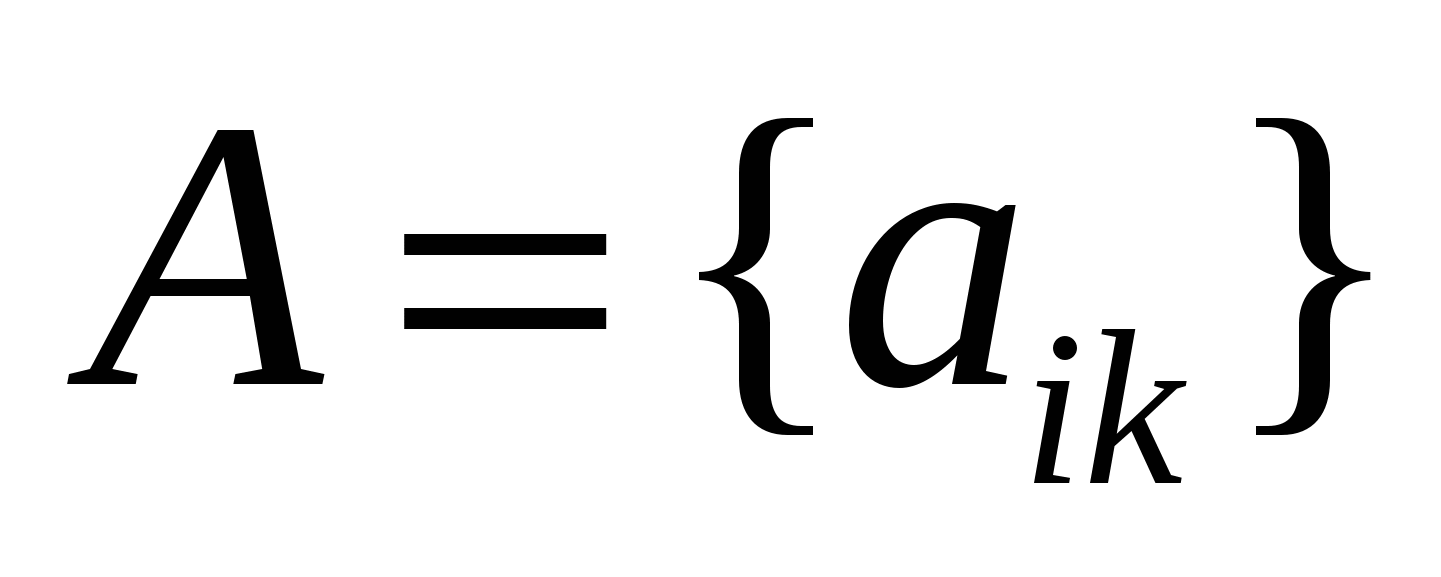
(4)

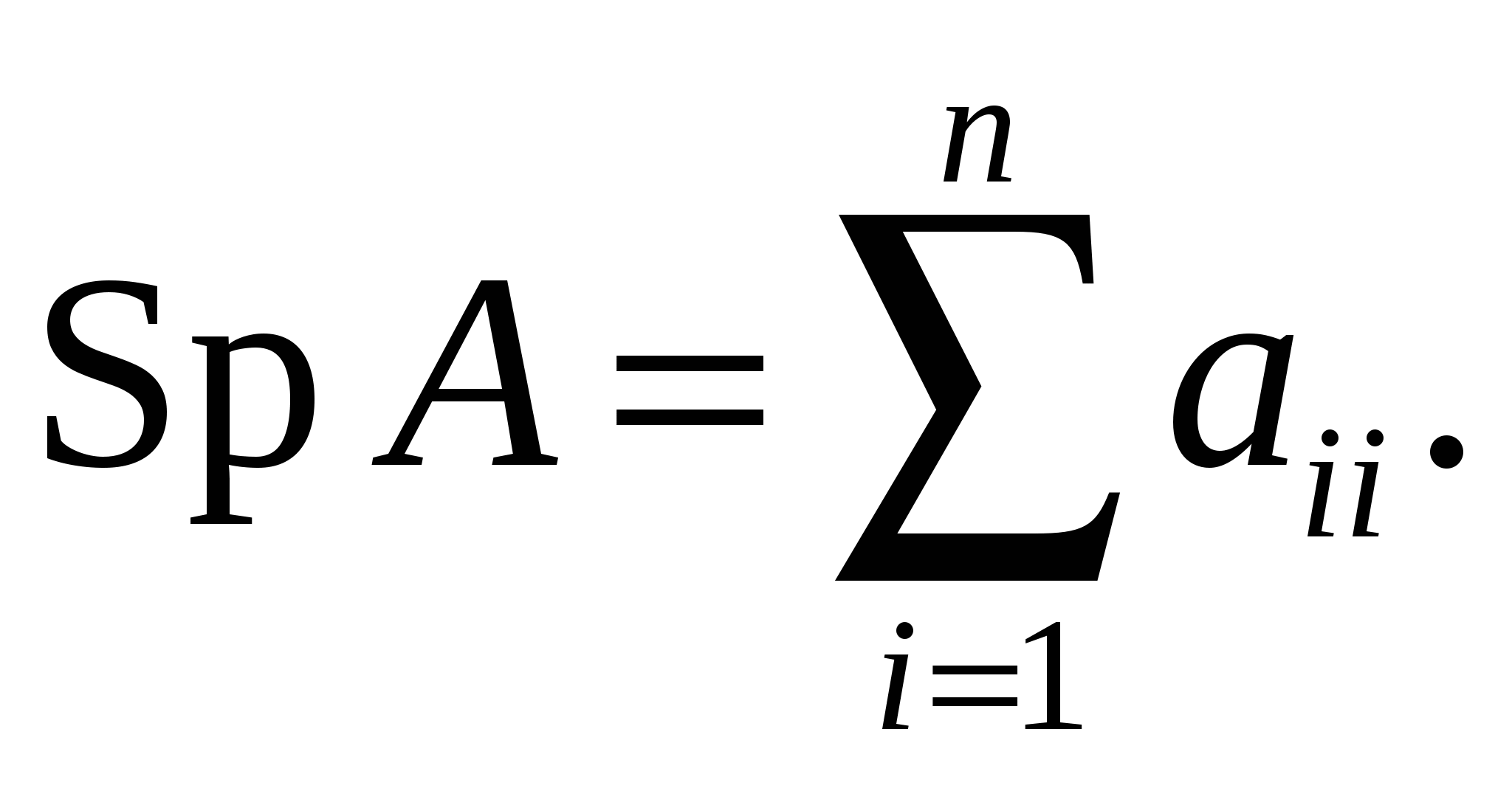
называемым характеристическим многочленом. Корни этого многочлена являются собственными значениями матрицы *A*.

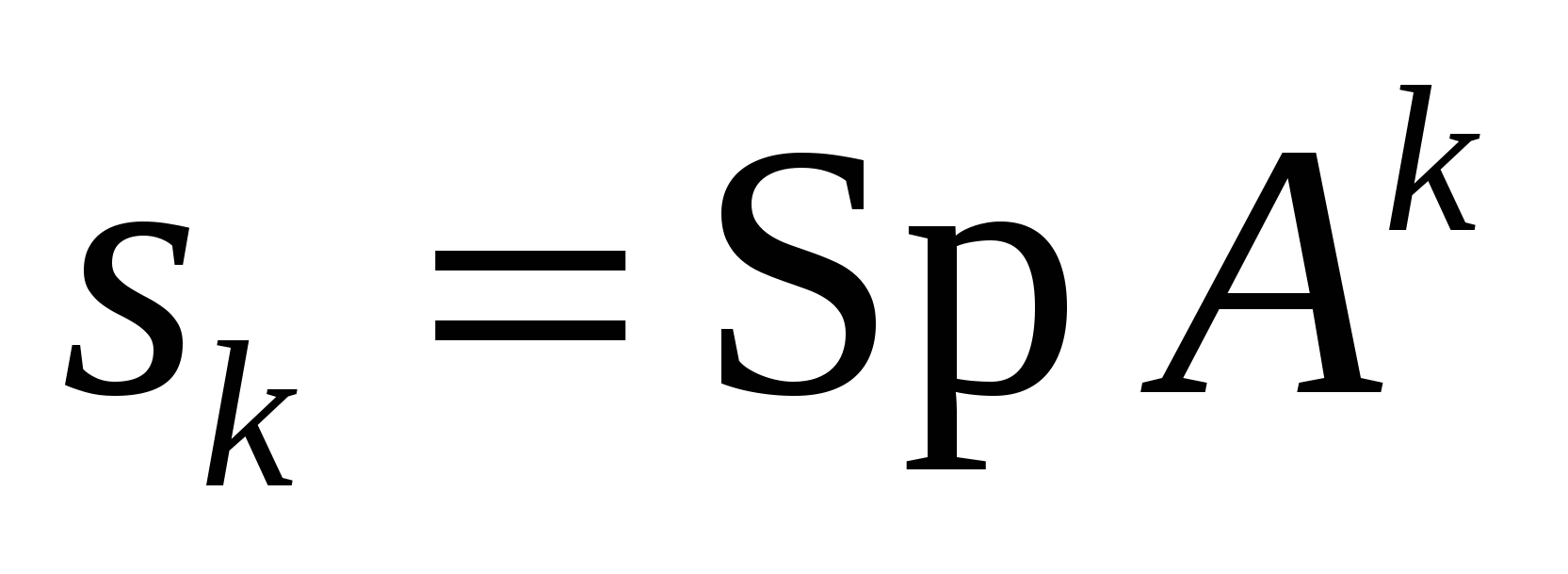
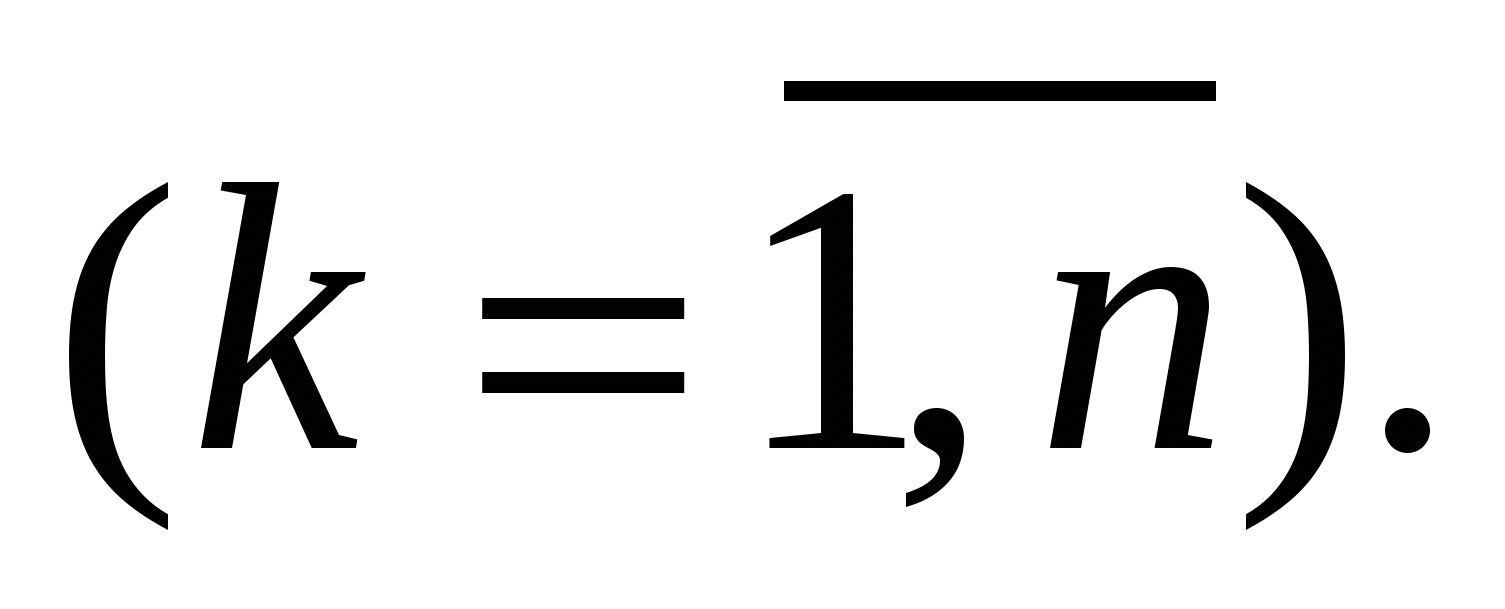
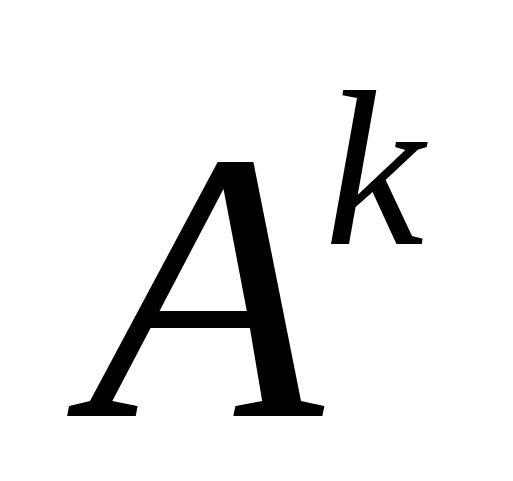
Для нахождения собственных векторов матрицы требуется решить систему линейных алгебраических уравнений, решение которой не единственно. Из линейной алгебры известно, что в этом случае структура общего решения системы имеет следующий вид: одно или несколько неизвестных, называемых свободными, могут принимать любые значения, а остальные неизвестные выражаются через свободные. Число свободных неизвестных равно числу уравнений системы, являющихся следствием остальных уравнений. На практике, если свободное неизвестное одно (что часто и бывает), его полагают равным некоторому числу, например единице. После этого остальные неизвестные (компоненты вектора) находятся однозначно из подсистемы линейно независимых уравнений, в которой отброшено уравнение, являющееся следствием остальных. Эта процедура не влияет на результат решения задачи, поскольку собственные векторы находятся с точностью до постоянного множителя.

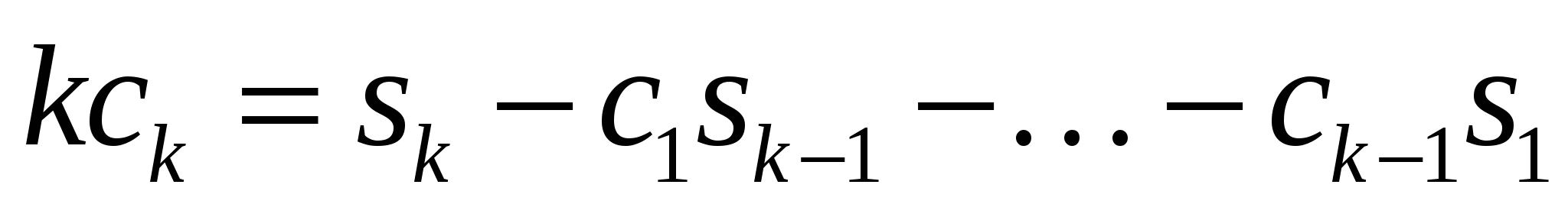
Метод Леверье-Фадеева состоит в нахождении коэффициентов характеристического многочлена

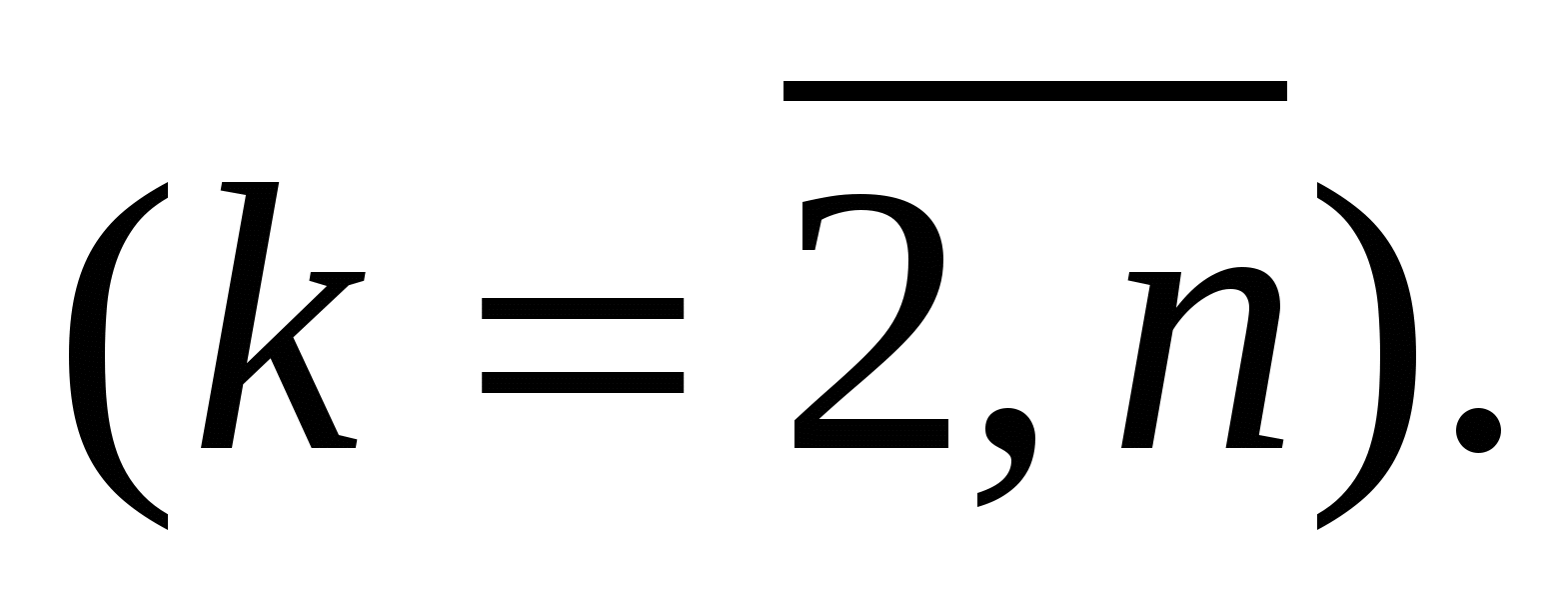
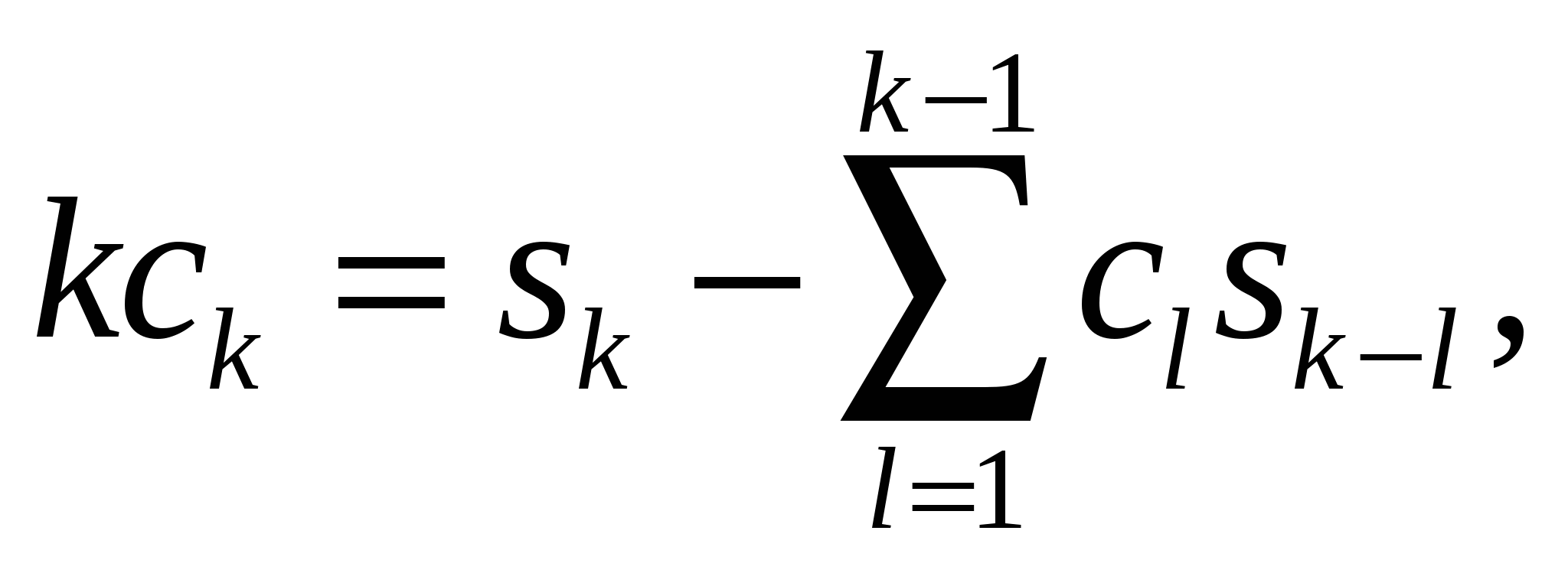
(1)

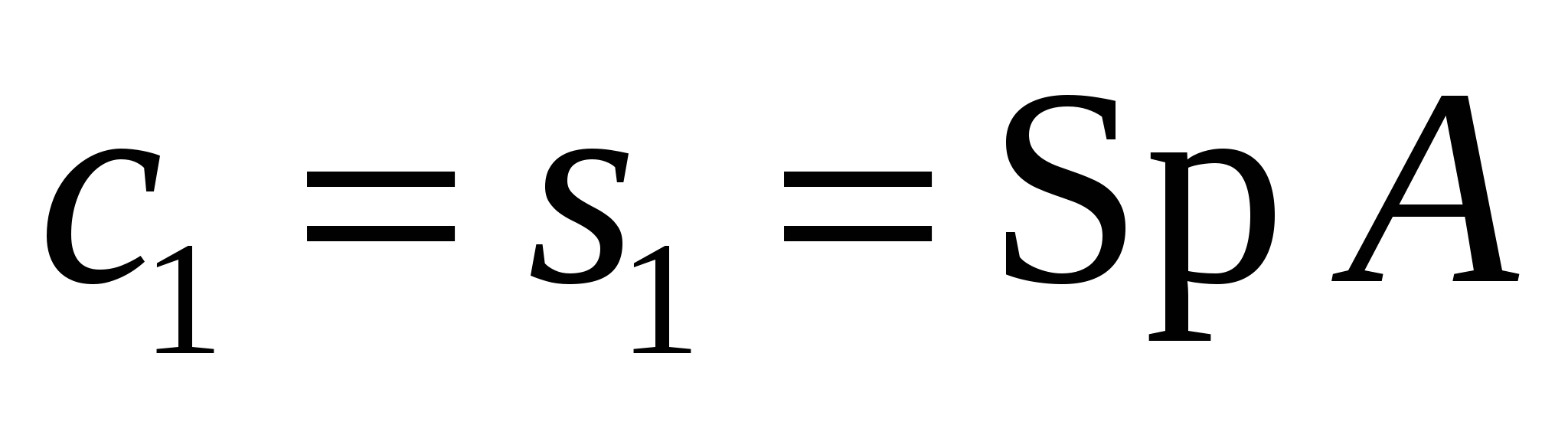
Для изложения метода введем понятие о следе матрицы. Под следом матрицы понимают сумму диагональных элементов этой матрицы:



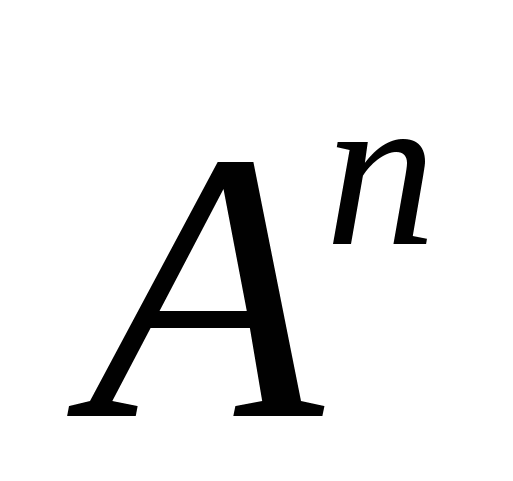
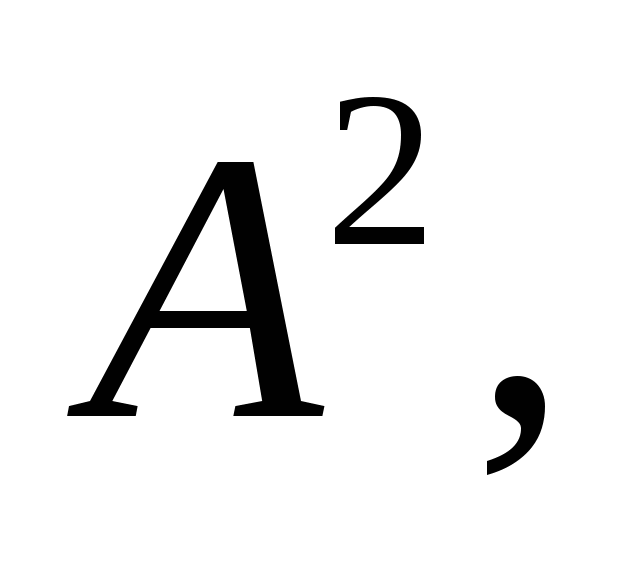
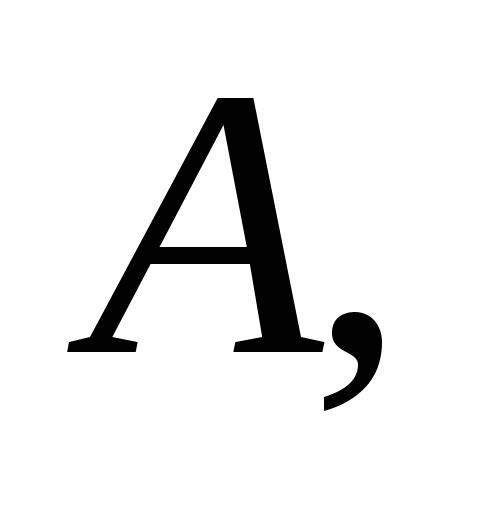
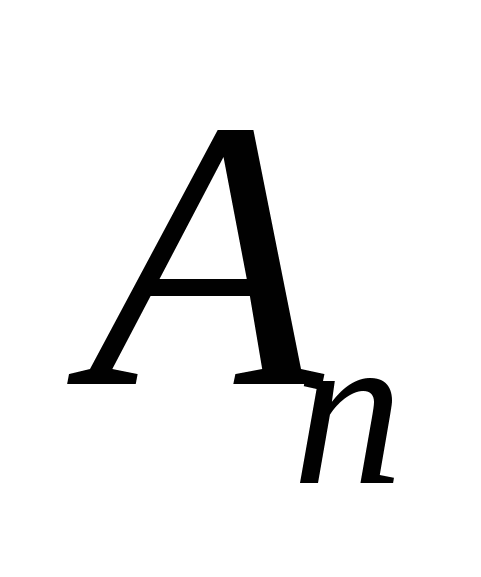
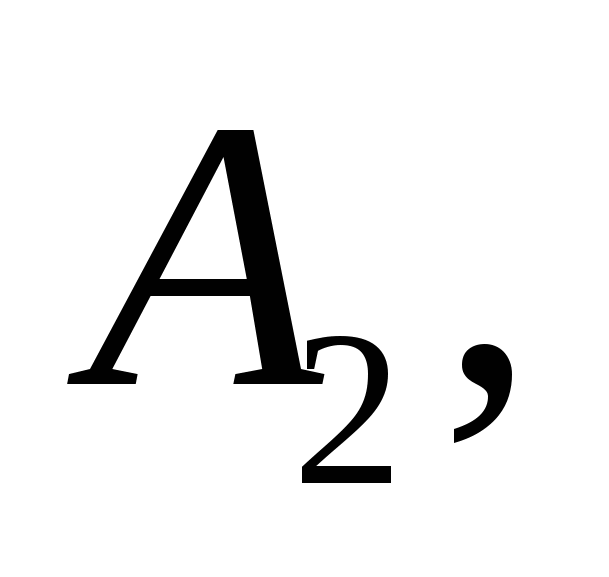
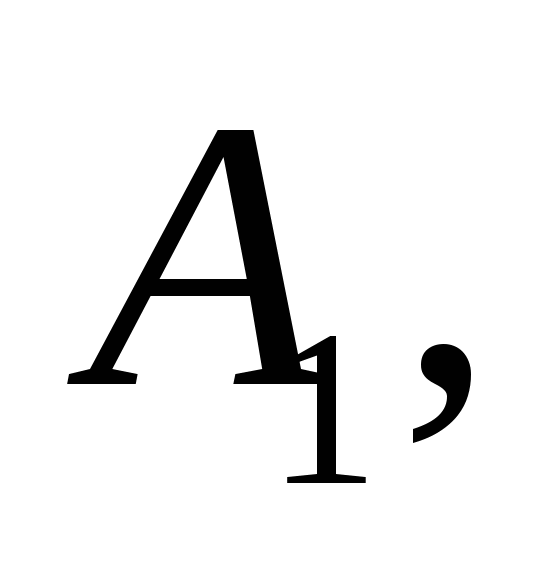
Обозначим через след матрицыТогда справедливы соотношения, известные под названием формул Ньютона:

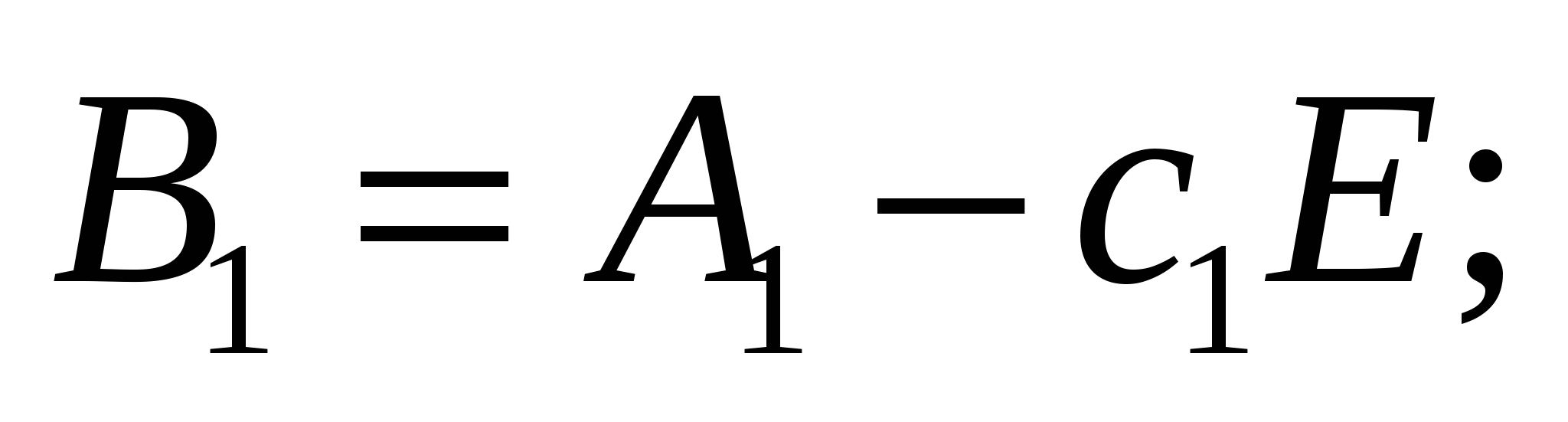
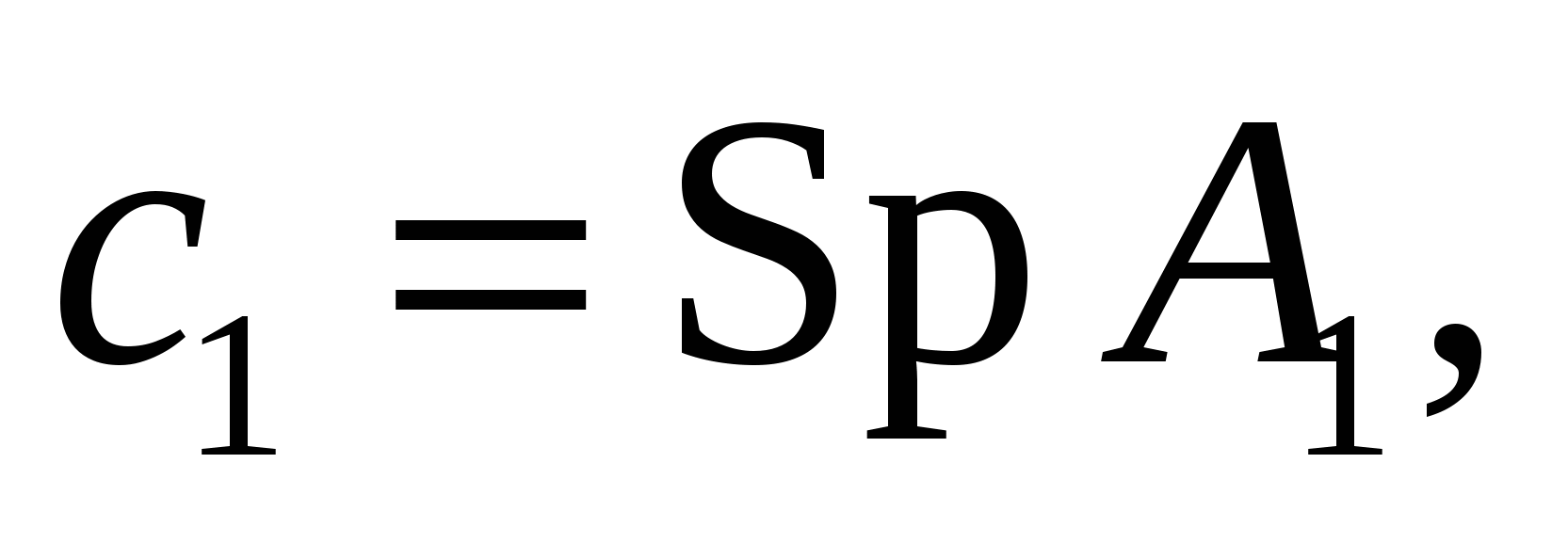
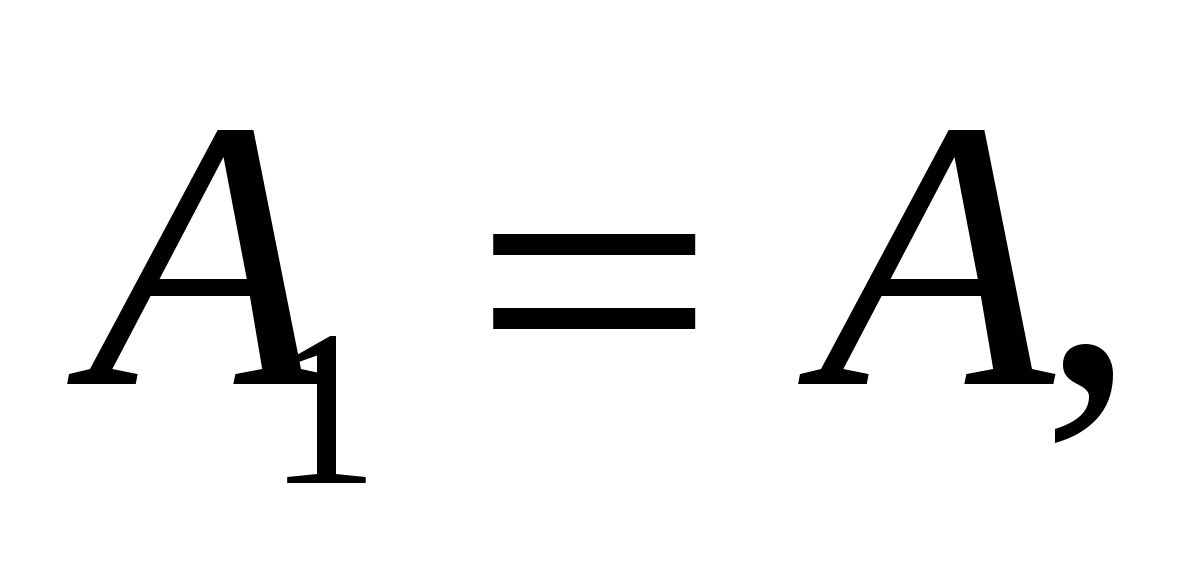
или

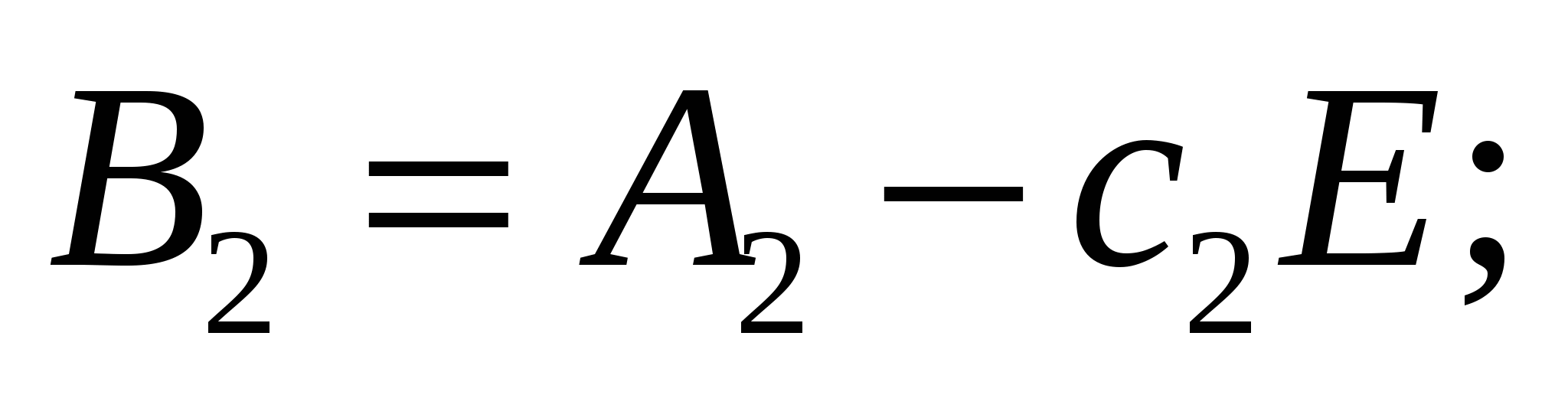
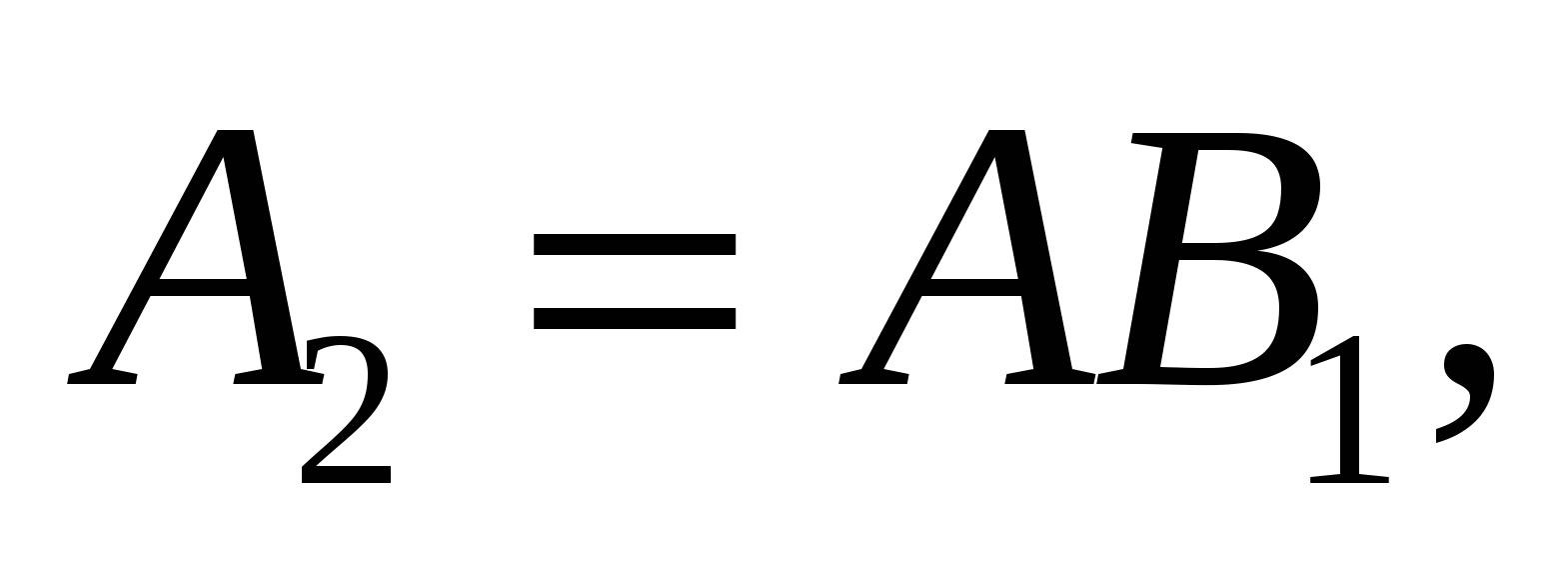
(2)

Первый коэффициент находится как . Остальные из рекуррентного соотношения (2).

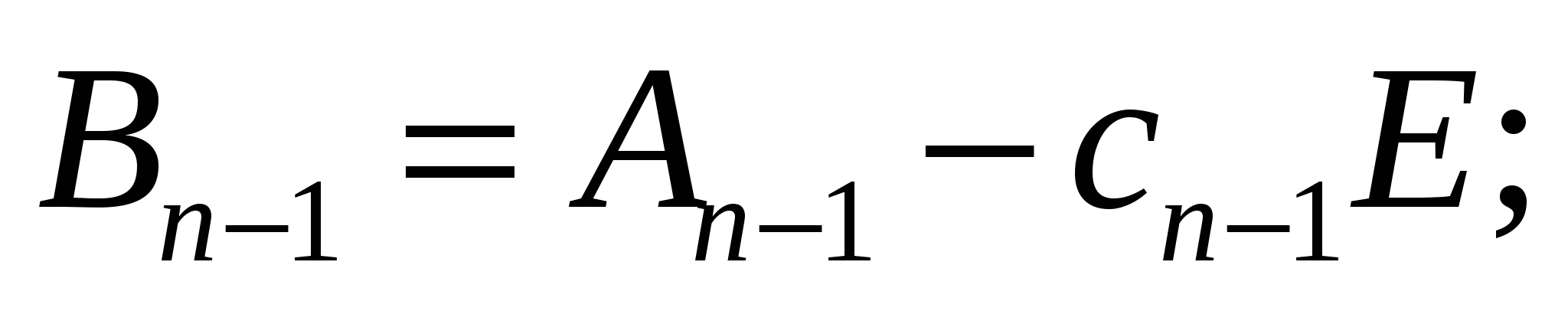
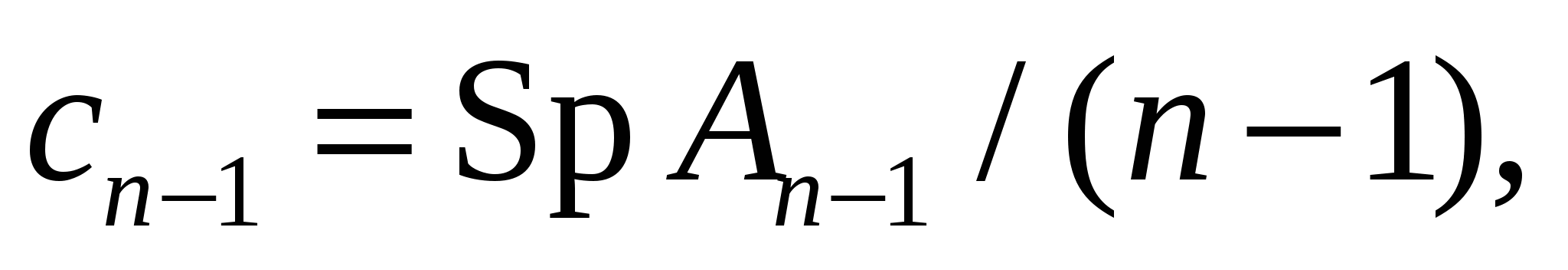
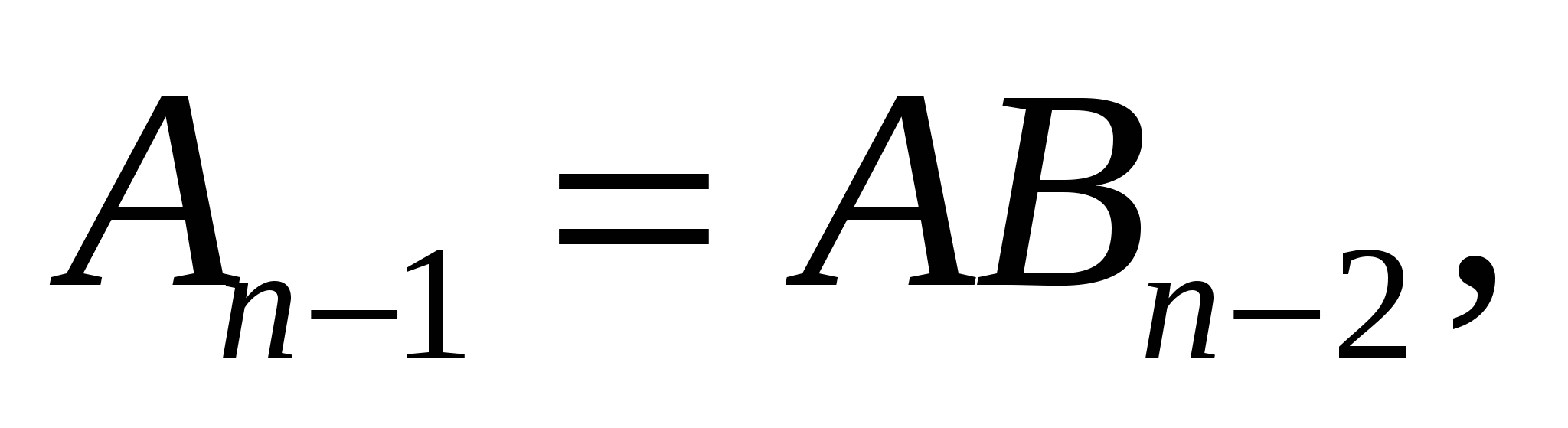
Таким образом, процесс вычисления сводится к последовательному вычислению степеней матрицы *A*, затем к вычислению их следов и, наконец, к решению рекуррентной системы (2).

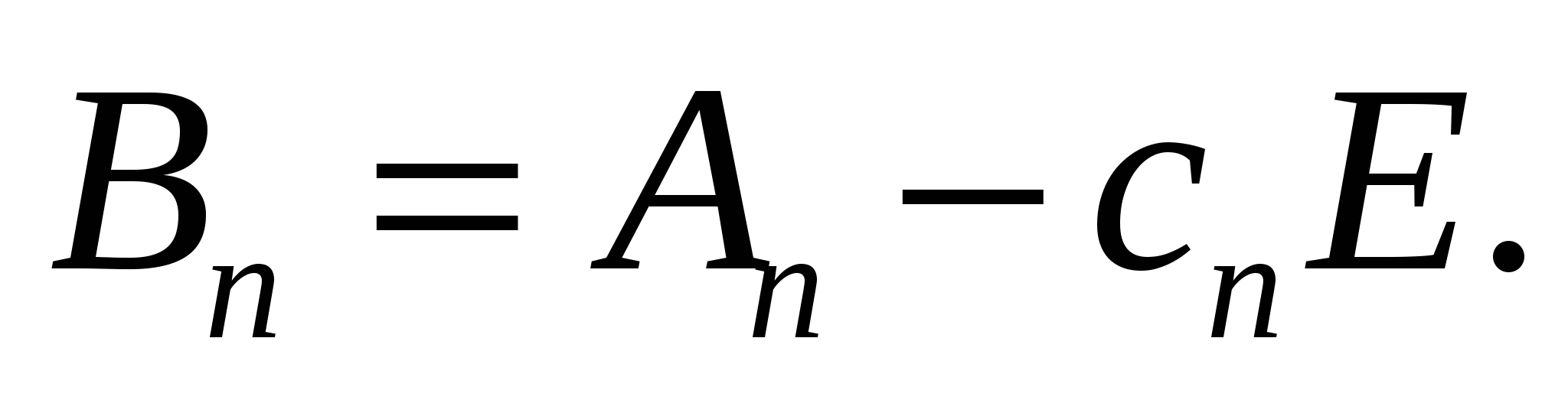
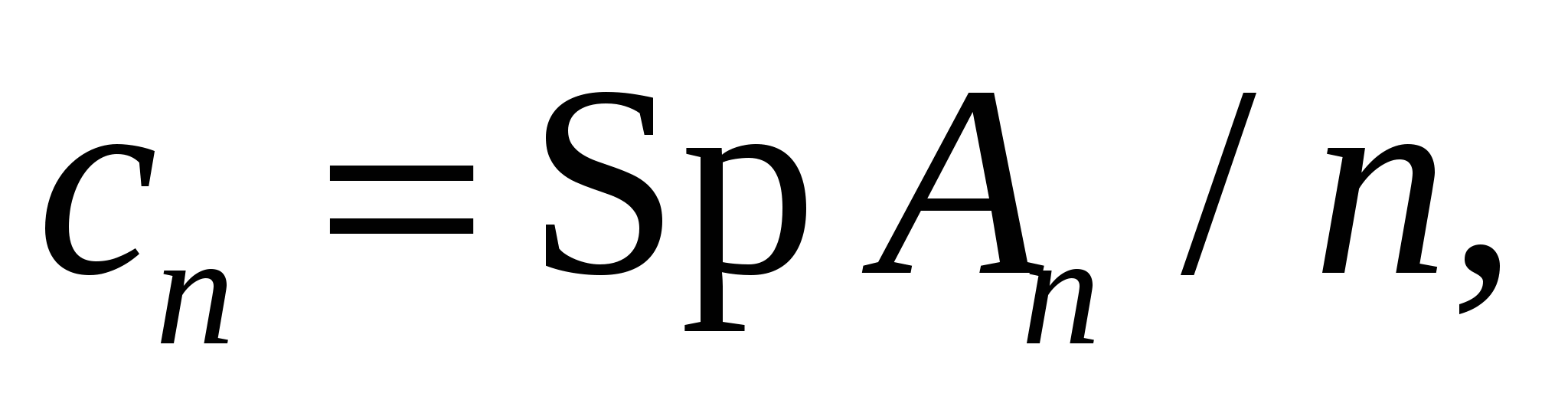
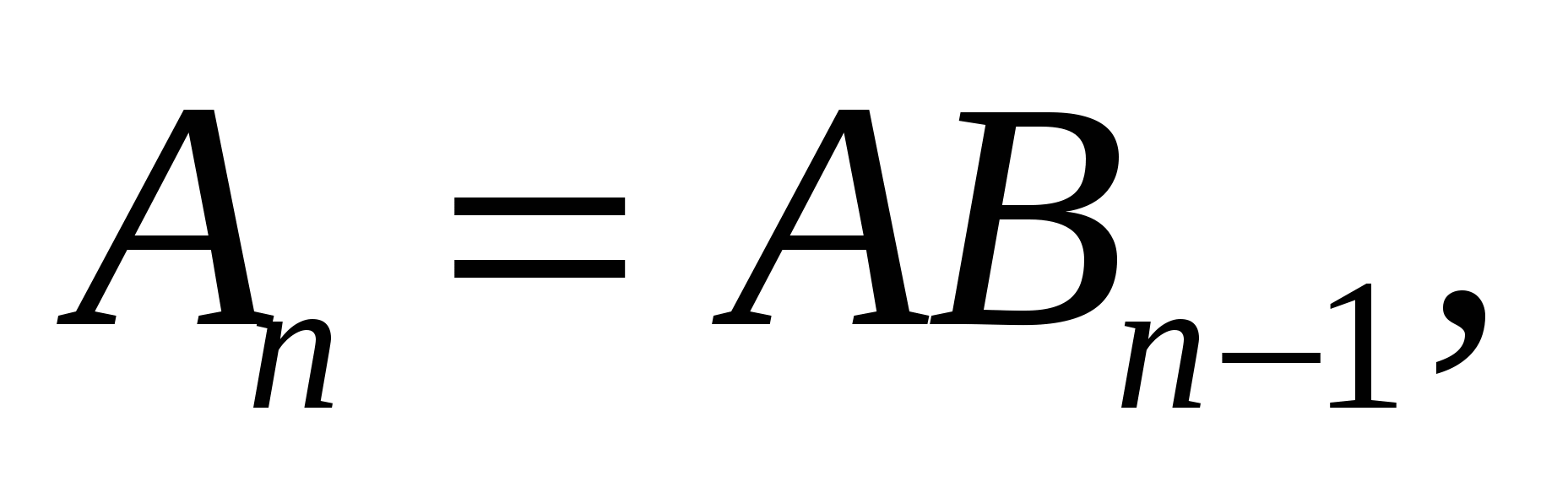
В методе Фаддеева предлагается вместо следов степеней вычислить последовательно следы некоторых других матриц, построенных следующим образом:



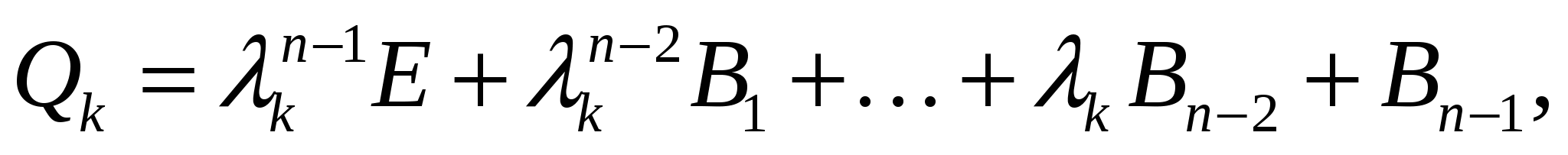


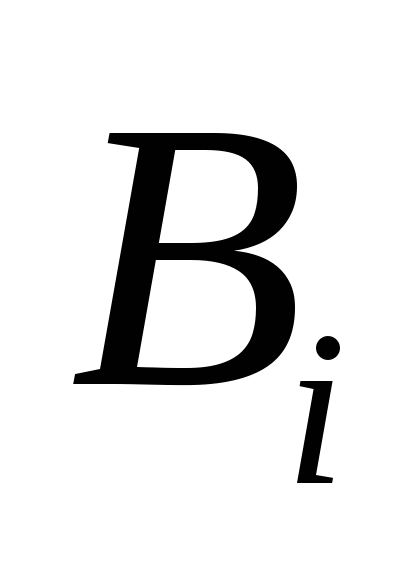
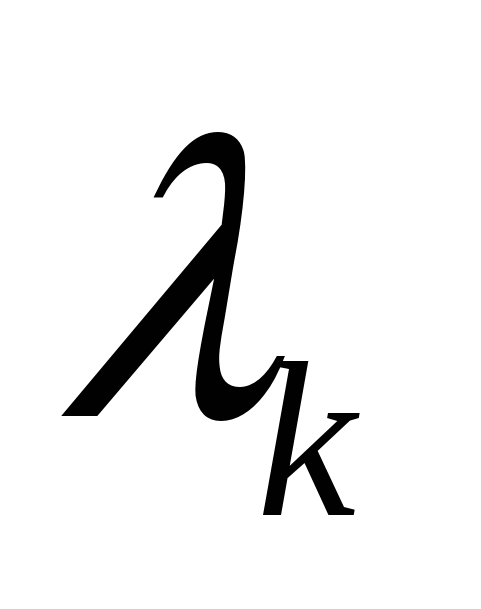
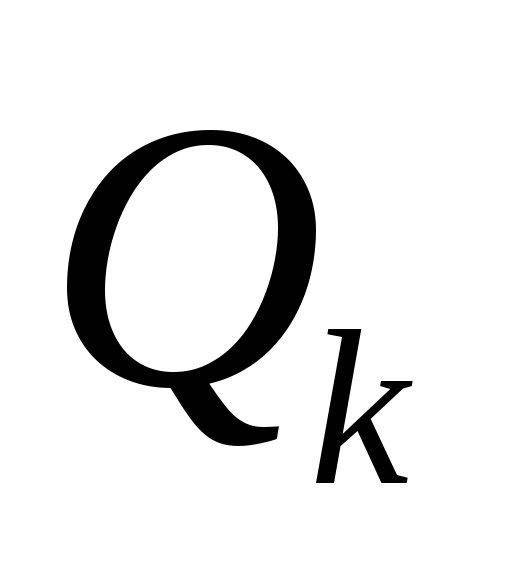
…



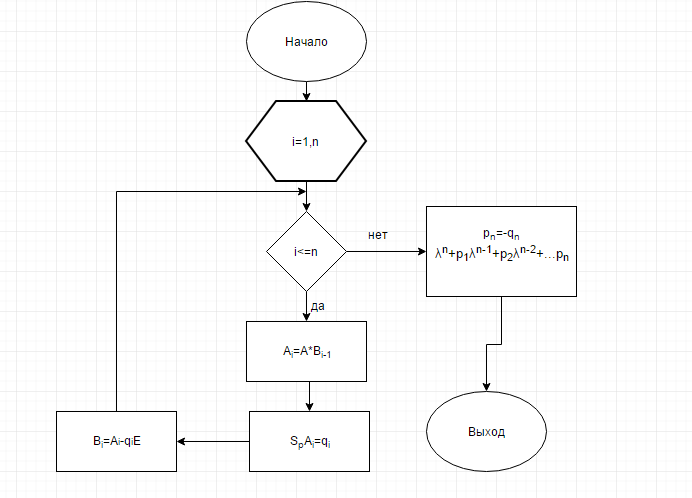


Данный метод позволяет найти не только коэффициенты характеристического многочлена, но и дает возможность определить обратную матрицу и собственные векторы матрицы. Найдем собственные вектора матрицы *A*. Пусть собственные числа уже вычислены и при этом оказались различными. Построим матрицу



где – матрицы, вычисленные в процессе нахождения коэффициентов характеристического многочлена, аесть *k*-е собственное число матрицы *A*. Каждый столбец матрицысостоит из компонент собственного вектора, принадлежащего собственному числу.

БЛОК-СХЕМА МЕТОДА ЛЕВЕРЬЕ-ФАДЕЕВА



ИСХОДНЫЙ КОД МЕТОДА ЛЕВЕРЬЕ-ФАДЕЕВА

namespace LeverieFadeev

{ class Program

{ double[] equationCoefs = new double[3];

public double F(double x)

{return Math.Pow(x, 3) + (equationCoefs[0] \* Math.Pow(x, 2)) + (equationCoefs[1] \* x) + equationCoefs[2];}

public double SolveEquation(double a, double b, double h)

{ double n = Math.Round(Math.Log(((b - a) / 0.001), Math.E) / Math.Log(2, Math.E));

int i = 2;

Console.WriteLine("----------------------Итерация 1---------------------");

double x = (a + b) / 2;

Console.WriteLine("xє[" + Math.Round(a, 5) + ";" + b + "]");

Console.WriteLine("x = " + Math.Round(x, 5));

while (Math.Abs(b - a) > h)

{ if (F(a) \* F(x) < 0)

{

Console.WriteLine("f(" + Math.Round(a, 5) + ") = " + Math.Round(F(a), 5) + " и f(" + Math.Round(x, 5) + ") = " + Math.Round(F(x), 5));

b = x;

Console.WriteLine("Выберем отрезок [a,x] = [" + Math.Round(a, 5) + ";" + Math.Round(x, 5) + "]");}

if (F(b) \* F(x) < 0)

{

Console.WriteLine("f(" + Math.Round(b, 5) + ") = " + Math.Round(F(b), 5) + " и f(" + Math.Round(x, 5) + ") = " + Math.Round(F(x), 5));

a = x;

Console.WriteLine("Выберем отрезок [x,b] = [" + Math.Round(x, 5) + ";" + Math.Round(b, 5) + "]");

}

x = (a + b) / 2;

Console.WriteLine("----------------------Итерация " + i + "---------------------");

Console.WriteLine("x = " + Math.Round(x, 5));

i++;

}

return x;

}

static void Main(string[] args)

{

int[][] a = new int[3][];

a[0] = new int[] { 2, 3, 2 };

a[1] = new int[] { 4, -6, -4 };

a[2] = new int[] { -1, 4, 7 };

int[][] a2 = new int[3][];

a2[0] = new int[] { 2, 3, 2 };

a2[1] = new int[] { 4, -6, -4 };

a2[2] = new int[] { -1, 4, 7 };

int[][] b = new int[3][];

for (int i = 0; i < b.Length; i++)

b[i] = new int[3];

int[][] e = new int[3][];

e[0] = new int[] { 1, 0, 0 };

e[1] = new int[] { 0, 1, 0 };

e[2] = new int[] { 0, 0, 1 };

int[] q = new int[3];

Console.WriteLine("----------------Исходная матрица А--------------");

for (int i = 0; i < 3; i++)

{ for (int j = 0; j < 3; j++)

{Console.Write(a[i][j] + "\t");}

Console.WriteLine();

}

for (int z = 0; z < 3; z++)

{

int sp = 0;

for (int i = 0; i < 3; i++)

{sp += a2[i][i];}

Console.WriteLine("SpA" + (z + 1) + " = " + sp);

q[z] = sp / (z + 1);

Console.WriteLine("q" + (z + 1) + " = " + sp + " / " + (z + 1) + " = " + q[z]);

b = raznost(a2, q[z]);

Console.WriteLine("------------------B " + (z + 1) + "-------------------");

for (int i = 0; i < 3; i++)

{

for (int j = 0; j < 3; j++)

{

Console.Write(b[i][j] + "\t");

}

Console.WriteLine();

}

a2 = proizvedenie(a, b);

Console.WriteLine("-------------------A " + (z + 2) + "-------------------------");

for (int i = 0; i < 3; i++)

{

for (int j = 0; j < 3; j++)

{

Console.Write(a2[i][j] + "\t");

}

Console.WriteLine();

}

}

Program ex = new Program();

for (int i = 0; i < 3; i++)

{

ex.equationCoefs[i] = -q[i];

}

Console.WriteLine("x^3+({0}\*x^2)+({1}\*x)+({2})", ex.equationCoefs[0], ex.equationCoefs[1], ex.equationCoefs[2]);

ex.SolveEquation(-10, 10, 0.001);

}

public static int[][] proizvedenie(int[][] a, int[][] b)

{

int[][] c = new int[3][];

for (int i = 0; i < 3; i++)

{

c[i] = new int[3];

for (int j = 0; j < 3; j++)

{

c[i][j] = a[i][0] \* b[0][j] + a[i][1] \* b[1][j] + a[i][2] \* b[2][j];

}

}

return c;

}

public static int[][] raznost(int[][] b, int q)

{

int[][] razn = new int[3][];

for (int i = 0; i < 3; i++)

{

razn[i] = new int[3];

for (int j = 0; j < 3; j++)

{razn[i][j] = b[i][j];

}}

for (int i = 0; i < 3; i++)

{

razn[i][i] = razn[i][i] - q;}

return razn;}}}

РУЧНОЙ ПРОСЧЕТ МЕТОДА ЛЕВЕРЬЕ-ФАДЕЕВА

*Матрица студента Рузняева Д.А*

А = 2 3 2

4 -6 -4

-1 4 7

А = А1

q1 = 2 -6 + 7 = 3 = Sp(A)

B1 = A1 - q\*E = 2 3 2 - 3 0 0 = -1 3 2

4 -6 -4 0 3 0 4 -9 -4

-1 4 7 0 0 3 -1 4 4

A2 = A\*B1 = 2 3 2 \* -1 3 2 = 8 -13 0

4 -6 -4 4 -9 1 -24 50 16

-1 4 7 -1 4 4 10 -11 10

SpA2 = 8 + 10 + 50 = 68 q2 = 68/2 = 34

B2 = A2 - q2\*E = 8 -13 0 - 34 0 0 = -26 -13 0

-24 50 16 0 34 0 -24 -16 16

10 -11 10 0 0 34 10 -11 -24

A3 = A\*B2 = 2 3 2 \* -26 -13 0 = -104 0 0

4 -6 -4 -24 16 16 0 -104 0

-1 4 7 10 -11 -24 0 0 -104

SpA3 = -104-104-104 = -312 q3 = -104

B3 = ( A3 – q3\*E) = -104 0 0 - -104 0 0 = 0 0 0

0 -104 0 0 -104 0 0 0 0

0 0 -104 0 0 -104 0 0 0

p1 = -3 ; p2 = -34 ; p3 = 104

В результате получаем характеристическое уравнение:

λ^3 - 3\* λ ^2 - 34\* λ + 104 = 0

Решим его методом дихотомии:

----------------------Итерация 1---------------------

xє[-10;10]

x = 0

f(-10) = -856 и f(0) = 104

Выберем отрезок [a,x] = [-10;0]

----------------------Итерация 2---------------------

x = -5

f(-10) = -856 и f(-5) = 74

Выберем отрезок [a,x] = [-10;-5]

----------------------Итерация 3---------------------

x = -7,5

f(-5) = 74 и f(-7,5) = -231,625

Выберем отрезок [x,b] = [-7,5;-5]

----------------------Итерация 4---------------------

x = -6,25

f(-5) = 74 и f(-6,25) = -44,82812

Выберем отрезок [x,b] = [-6,25;-5]

----------------------Итерация 5---------------------

x = -5,625

f(-6,25) = -44,82812 и f(-5,625) = 22,34961

Выберем отрезок [a,x] = [-6,25;-5,625]

----------------------Итерация 6---------------------

x = -5,9375

f(-5,625) = 22,34961 и f(-5,9375) = -9,20679

Выберем отрезок [x,b] = [-5,9375;-5,625]

----------------------Итерация 7---------------------

x = -5,78125

f(-5,9375) = -9,20679 и f(-5,78125) = 7,06808

Выберем отрезок [a,x] = [-5,9375;-5,78125]

----------------------Итерация 8---------------------

x = -5,85938

f(-5,78125) = 7,06808 и f(-5,85938) = -0,94375

Выберем отрезок [x,b] = [-5,85938;-5,78125]

----------------------Итерация 9---------------------

x = -5,82031

f(-5,85938) = -0,94375 и f(-5,82031) = 3,09339

Выберем отрезок [a,x] = [-5,85938;-5,82031]

----------------------Итерация 10---------------------

x = -5,83984

f(-5,85938) = -0,94375 и f(-5,83984) = 1,08264

Выберем отрезок [a,x] = [-5,85938;-5,83984]

----------------------Итерация 11---------------------

x = -5,84961

f(-5,85938) = -0,94375 и f(-5,84961) = 0,07141

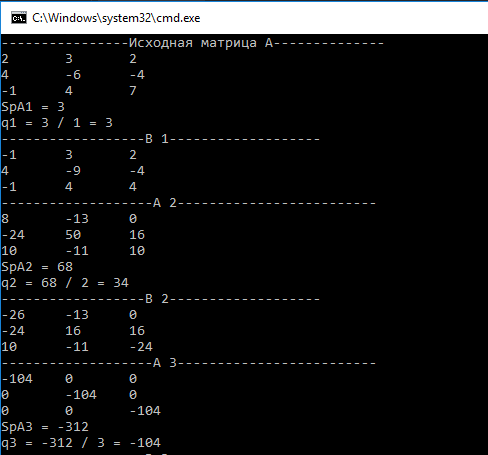
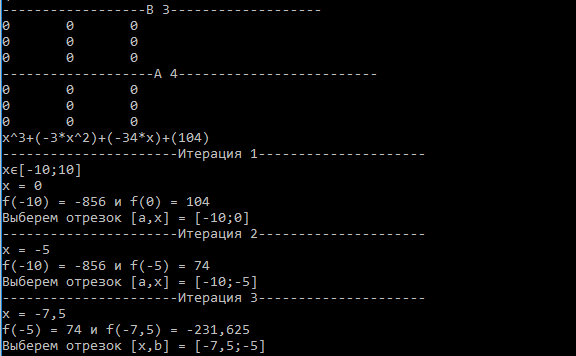
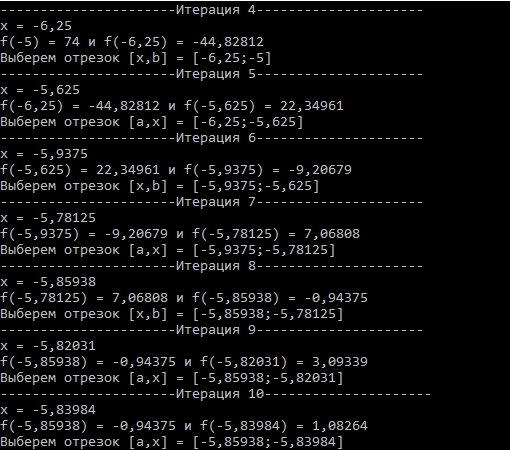
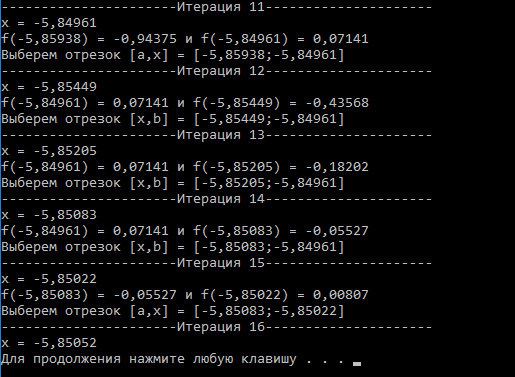
Выберем отрезок [a,x] = [-5,85938;-5,84961]

----------------------Итерация 12---------------------

x = -5,85449

ПРОСЧЕТ ПО ПРОГРАММЕ МЕТОДОМ ЛЕВЕРЬЕ-ФАДЕЕВА

*Студента Рузняева Д.А*

МЕТОД ЛЕВЕРЬЕ

Метод: Пусть det( A - ʎE) = ʎ^n + p1ʎ^n-1 + p2ʎ^n-2+ …+ p^n (3)- это характеристический многочлен матрицы А. ʎ1 ... ʎn полная совокупность корней многочлена (3).

Рассмотрим суммы:

Sk = ʎ1^k + ʎ2^k …ʎn^k , k= от 1 до n

S1 = ʎ1^1 + ʎ2^1 …ʎn^1

S2 = ʎ1^2 + ʎ2^2 …ʎn^2

Sn = ʎ1^n + ʎ2^n …ʎn^n

Каждая сумма - следующая матрица А в степени k.

Тогда,при k ≤ n справедливы формулы Ньютона:

Sk + p1Sk-1 + p2Sk-2+ …+ pk-1S1 = -kpk

Откуда:

При k=1: p1 =-S1

При k= 2: p2 = -1/2(S2 +p1S1) (4)

При k = n : pn = -1/n(Sn + p1Sn-1 + p2Sn-2 + … + pn-1S1)

Коэффициенты характеристического многочлена p1…pnможно оценить, если известны суммы S1…Sn .

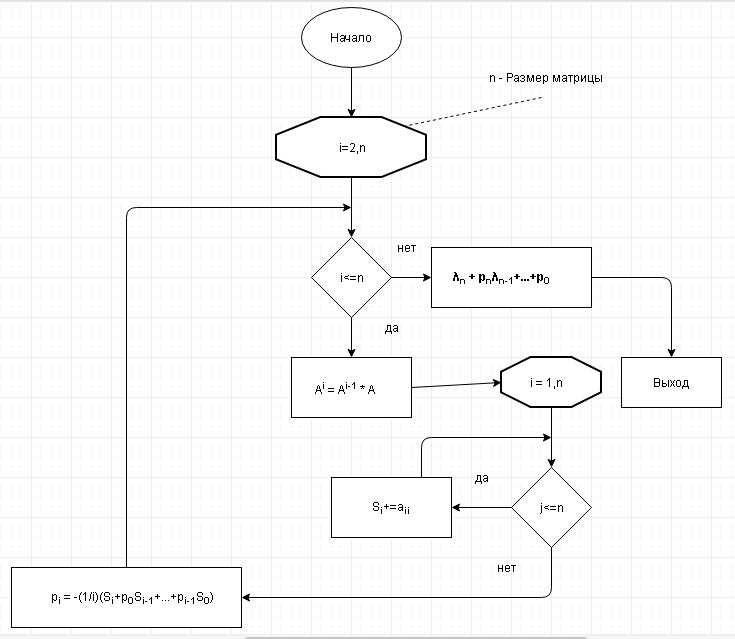
Схема раскрытия характеристического определителя методом Леверье:

1 Вычислим степени матриц A^k = A^k-1A, k = от 1 до n

2 Определяем Sk как суммы элементов главных диагональных элементов матриц Ak

3 По формулам (4) находим коэффициенты pi , i=1,n

БЛОК-СХЕМА МЕТОДА ЛЕВЕРЬЕ

****

ИСХОДНЫЙ КОД МЕТОД ЛЕВЕРЬЕ

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Leverie

{

class Program

{

double[][] aR;

double[] equationCoefs;

public Program(int i)

{

aR = new double[i][];

for (int j = 0; j < aR.Length; j++)

{

aR[j] = new double[i];

for (int z = 0; z < aR.Length; z++)

{

aR[j][z] = 0;

}

}

equationCoefs = new double[3];

}

void Solve()

{

Program[] aa = {new Program(3), new Program(3), new Program(3)};

double[][] source = new double[3][];

for (int z = 0; z < source.Length; z++)

{

source[z] = new double[3];

}

source[0][0] = 5; source[0][1] = 2; source[0][2] = 5;

source[1][0] = -2; source[1][1] = -1; source[1][2] = -8;

source[2][0] = 2; source[2][1] = -3; source[2][2] = -2;

aa[0].aR = source;

double[] s = new double[3];

double[] p = new double[3];

for(int g = 1;g<aR.Length;g++){

//Console.WriteLine("------------Получим матрицу А" + g + "-----------");

for (int i = 0; i < aa[0].aR.Length; i++)

{

for (int j = 0; j < aa[0].aR.Length; j++)

{

for (int z = 0; z < aa[0].aR.Length; z++)

{

aa[g].aR[i][j] += aa[g-1].aR[i][z] \* aa[0].aR[z][j];

//Console.Write(aa[g].a[i][j] + "\t");

}

}

//Console.WriteLine();

}

}

for (int j = 0; j < aR.Length; j++)

{

for (int i = 0; i < aR.Length; i++)

{

s[j] += aa[j].aR[i][i];

}

}

p[0] = -1\*s[0];

p[1] = (-1\*(1.0 / 2.0)) \* (s[1] + (p[0] \* s[0]));

p[2] = (-1\*(1.0 / 3.0)) \* (s[2] + (p[0] \* s[1]) + (p[1] \* s[0]));

Print(aa, 0, s, p);

Print(aa, 1, s, p);

Print(aa, 2, s, p);

equationCoefs[0] = p[0];

equationCoefs[1] = p[1];

equationCoefs[2] = p[2];

}

public double F(double x)

{

//for(int i=0;i<3;i++)

//Console.WriteLine("equationCoefs[" + i + "] = " + equationCoefs[i]);

return Math.Pow(x, 3) + (equationCoefs[0] \* Math.Pow(x, 2)) + (equationCoefs[1] \* x) + equationCoefs[2];

}

public double SolveEquation(double a, double b, double h)

{

Console.WriteLine("Получи характеристическое уравнение");

Console.WriteLine("x^3" + (equationCoefs[0] + "x^2") + "+" + (equationCoefs[1] + "x") + "+" + equationCoefs[2]);

double n = Math.Round(Math.Log(((b - a) / 0.001), Math.E) / Math.Log(2, Math.E));

Console.WriteLine("Необходимое количество итераций n > " + n);

int i = 2;

Console.WriteLine("----------------------Итерация 1---------------------");

double x = (a + b) / 2;

Console.WriteLine("xє[" + Math.Round(a, 5) + ";" + b + "]");

Console.WriteLine("x = " + Math.Round(x, 5));

while (Math.Abs(b - a) > h)

{

if (F(a) \* F(x) < 0)

{

Console.WriteLine("f(" + Math.Round(a, 5) + ") = " + Math.Round(F(a), 5) + " и f(" + Math.Round(x, 5) + ") = " + Math.Round(F(x), 5));

b = x;

Console.WriteLine("Выберем отрезок [a,x] = [" + Math.Round(a, 5) + ";" + Math.Round(x, 5) + "]");

}

if (F(b) \* F(x) < 0)

{

Console.WriteLine("f(" + Math.Round(b, 5) + ") = " + Math.Round(F(b), 5) + " и f(" + Math.Round(x, 5) + ") = " + Math.Round(F(x), 5));

a = x;

Console.WriteLine("Выберем отрезок [x,b] = [" + Math.Round(x, 5) + ";" + Math.Round(b, 5) + "]");

}

x = (a + b) / 2;

Console.WriteLine("----------------------Итерация " + i + "---------------------");

Console.WriteLine("x = " + Math.Round(x, 5));

i++;

}

return x;

}

void Print(Program[]aa,int num,double[]s,double[]p)

{

Console.WriteLine("------------Получим матрицу А" + num + "-----------");

for (int i = 0; i < aa[0].aR.Length; i++)

{

for (int j = 0; j < aa[0].aR.Length; j++)

{

Console.Write(aa[num].aR[i][j] + "\t");

}

Console.WriteLine();

}

Console.WriteLine("s[" + (num+1) + "] = " + s[num]);

Console.WriteLine("p[" + (num+1) + "] = " + p[num]);

}

static void Main(string[] args)

{

Program ex = new Program(3);

ex.Solve();

ex.SolveEquation(-3.16,4.5,0.001);

}

}

}

РУЧНОЙ РАСЧЕТ МЕТОДА ЛЕВЕРЬЕ

*Студента Костюка И.Ю.*

А = 5 2 5

-2 -1 -8

2 -3 -2

S1 = 5 – 1 -2 = 2

A2 = A \* A = -31 -7 -1

-24 21 14

12 13 38

S2 = 90

A2 = A2\*A = 167 72 213

-134 -111 -316

110 -103 -120

S3 = -64

P1 = -2

P2 = -1/2(90-2\*2) = -43

P3 = -1/3(-64 – 2\*90 -43\*2)=110

В результате получаем характеристическое уравнение:

λ^3 - 2λ^2-43λ +100 = 0

Решим его методом Дихотомии:

f’(λ) = 3λ^2 - 4λ - 43

3λ^2 - 4λ - 43= 0

λ1 = -3,16

λ2 = 4,5

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X | -∞ | -3,16 | 4,5 | ∞ |
| Sign f(x) | - | + | - | + |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X | -4 | -3,16 | 4,5 | 5 |
| Sign f(x) | - | + | - | + |

[-4;-3,16] ; [-3,16;4,5] ; [4,5;5]

Уточним корень отрезка [-3,16;4,5]

----------------------Итерация 1---------------------

xє[-3,16;4,5]

x = 0,67

f(4,5) = -32,875 и f(0,67) = 80,59296

Выберем отрезок [x,b] = [0,67;4,5]

----------------------Итерация 2---------------------

x = 2,585

f(4,5) = -32,875 и f(2,585) = 2,7541

Выберем отрезок [x,b] = [2,585;4,5]

----------------------Итерация 3---------------------

x = 3,5425

f(2,585) = 2,7541 и f(3,5425) = -22,9702

Выберем отрезок [a,x] = [2,585;3,5425]

----------------------Итерация 4---------------------

x = 3,06375

f(2,585) = 2,7541 и f(3,06375) = -11,75629

Выберем отрезок [a,x] = [2,585;3,06375]

----------------------Итерация 5---------------------

x = 2,82438

f(2,585) = 2,7541 и f(2,82438) = -4,87201

Выберем отрезок [a,x] = [2,585;2,82438]

----------------------Итерация 6---------------------

x = 2,70469

f(2,585) = 2,7541 и f(2,70469) = -1,14654

Выберем отрезок [a,x] = [2,585;2,70469]

----------------------Итерация 7---------------------

x = 2,64484

f(2,70469) = -1,14654 и f(2,64484) = 0,78253

Выберем отрезок [x,b] = [2,64484;2,70469]

----------------------Итерация 8---------------------

x = 2,67477

f(2,64484) = 0,78253 и f(2,67477) = -0,1874

Выберем отрезок [a,x] = [2,64484;2,67477]

----------------------Итерация 9---------------------

x = 2,6598

f(2,67477) = -0,1874 и f(2,6598) = 0,29623

Выберем отрезок [x,b] = [2,6598;2,67477]

----------------------Итерация 10---------------------

x = 2,66729

f(2,67477) = -0,1874 и f(2,66729) = 0,05408

Выберем отрезок [x,b] = [2,66729;2,67477]

----------------------Итерация 11---------------------

x = 2,67103

f(2,66729) = 0,05408 и f(2,67103) = -0,06674

Выберем отрезок [a,x] = [2,66729;2,67103]

----------------------Итерация 12---------------------

x = 2,66916

f(2,66729) = 0,05408 и f(2,66916) = -0,00635

Выберем отрезок [a,x] = [2,66729;2,66916]

----------------------Итерация 13---------------------

x = 2,66822

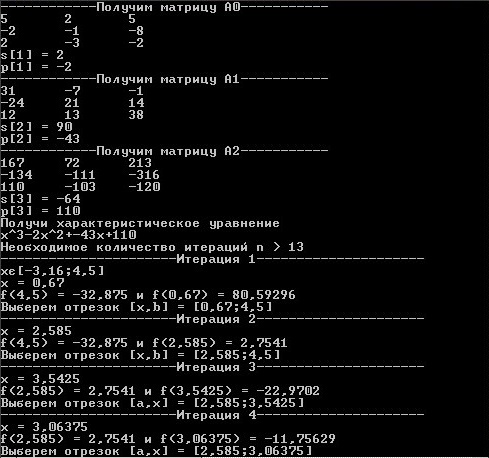
f(2,66916) = -0,00635 и f(2,66822) = 0,02386

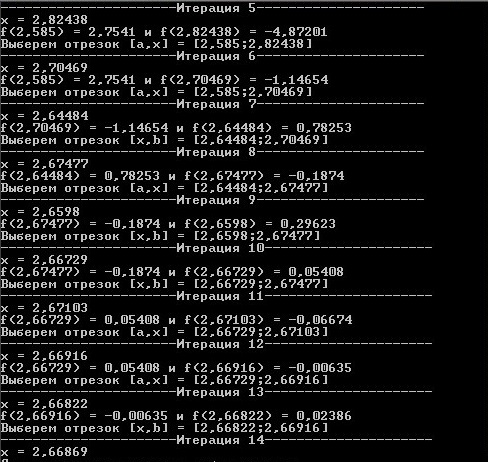
Выберем отрезок [x,b] = [2,66822;2,66916]

----------------------Итерация 14---------------------

x = 2,66869

РЕЗУЛЬТАТЫ ЧИСЛЕННЫХ РАСЧЕТОВ МЕТОДА ЛЕВЕРЬЕ





Вывод: Был изучен метод нахождения характеристического уравнения на основе матриц.