Міністерство освіти та Науки України

Національний технічний університет “Харківський політехнічний інститут”

Кафедра «Програмна інженерія і інформаційні технології управління»

Індивідуальне завдання №3

з дисципліни

«Чисельні методи»

Виконали:

студенти групи КН-34б

Рузняєв Д.А.

Костюк I.Ю.

Перевірив:

Проф. Гужва В.О.

Харків

2016

**МЕТОД ЛЕВЕРЬЕ-ФАДЕЕВА**

Метод Д.К. Фадеева является видоизменением метода Леверье. Помимо упрощений при вычислении коэффициентов характеристического полинома он позволяет определить обрат- ную матрицу и собственные вектора матрицы. Метод Леверье 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 матрицы   Тогда 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

…

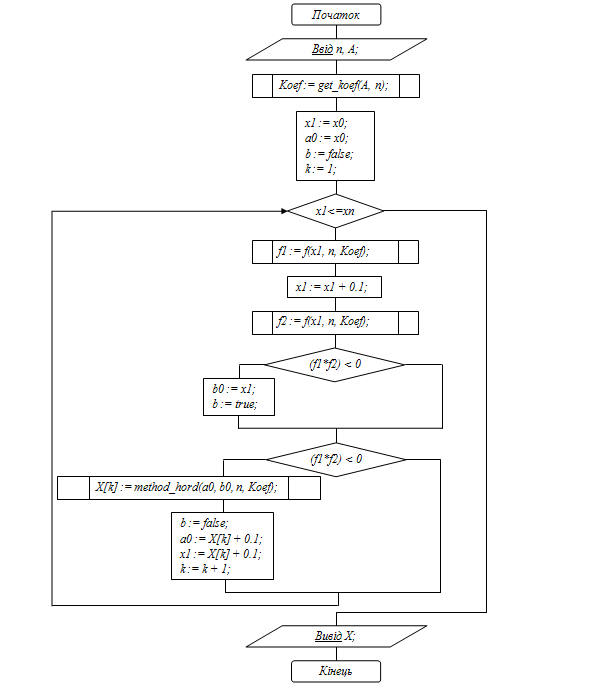
  

Данный метод позволяет найти не только коэффициенты характеристического многочлена, но и дает возможность определить обратную матрицу и собственные векторы матрицы. Найдем собственные вектора матрицы *A*. Пусть собственные числа уже вычислены и при этом оказались различными. Построим матрицу



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

БЛОК-СХЕМА МЕТОДА ЛЕВЕРЬЕ-ФАДЕЕВА

****

ИСХОДНЫЙ КОД МЕТОДА ЛЕВЕРЬЕ-ФАДЕЕВА

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace lever\_eFadeeva

{ 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){

Console.WriteLine("Получи характеристическое уравнение");

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;}

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(-4.05, 2.71, 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:

α^3 - 3\*α^2 - 34\*α + 104 = 0

Решим его методом дихотомии:

Необходимое количество итераций n > 13

----------------------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

f(-5,84961) = 0,07141 и f(-5,85449) = -0,43568

Выберем отрезок [x,b] = [-5,85449;-5,84961]

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

x = -5,85205

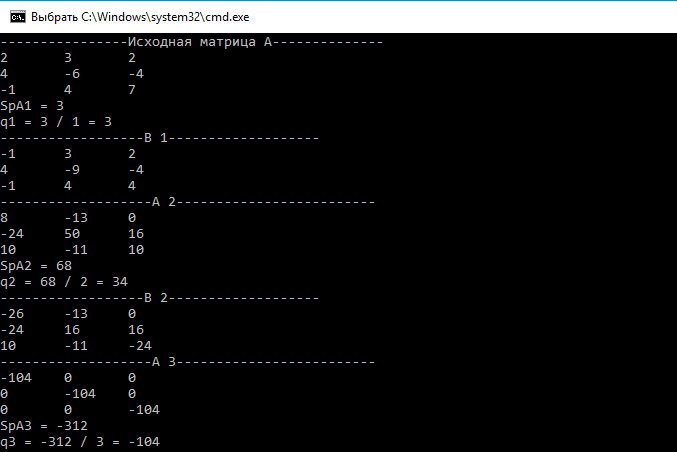
f(-5,84961) = 0,07141 и f(-5,85205) = -0,18202

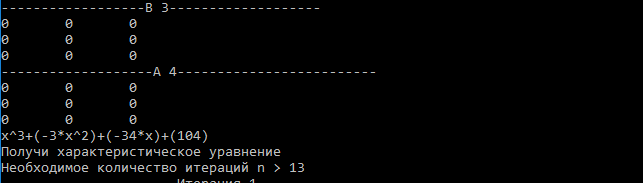
Выберем отрезок [x,b] = [-5,85205;-5,84961]

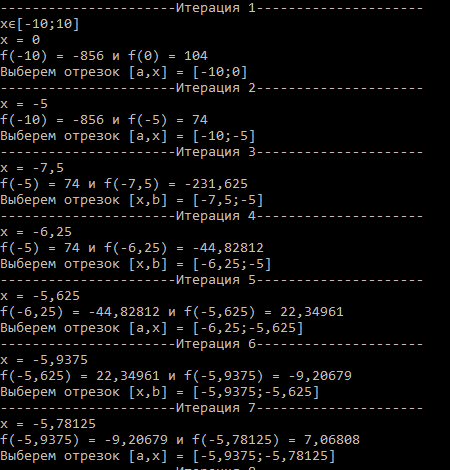
----------------------Итерация 14---------------------

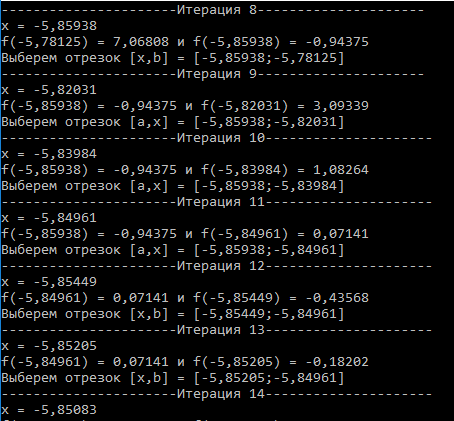
x = -5,85083

ПРОСЧЕТ ПО ПРОГРАММЕ МЕТОДОМ ЛЕВЕРЬЕ-ФАДЕЕВА









**МЕТОД ЛЕВЕРЬЕ**

Метод Леверье основан на формулах Ньютона для сумм степеней корней алгебраического уравнения (19).

Пусть Qn(λ) = (−1)n h λ n − p1λ n−1 − · · · − pn i (20) характеристический полином матрицы A и λ1, . . ., λn его корни, среди которых могут быть равные.

Тогда характеристический полином можно разложить на множители: Qn(λ) = (λ − λ1)(λ − λ2). . .(λ − λn). (21)

Перемножая скобки, стоящие справа в (21), а затем приведя подобные члены и сравнивая с коэффициентами из (20) получим, так называемые формулы Виета, выражающие коэф- фициенты многочлена через его корни:

25p1 = σ1, p2 = −σ2, . . . , pn−1 = (−1)n−2σn−1, pn = (−1)n−1σn, σ1 = λ1 + λ2 + · · · + λn, σ2 = λ1λ2 + λ1λ3 + · · · + λn−1λn, σ3 = λ1λ2λ1 + · · · + λn−2λn−1λn, · · · σn = λ1λ2 · · · λn — элементарные симметрические функции корней характеристического урав- нения.

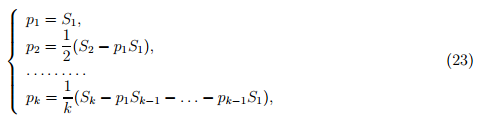
Рассмотрим еще следующие симметрические функции корней:

Sk = λ k 1 + λ k 2 + · · · + λ k n , k = 1, 2, . . . , n.

Теорема единственности, известная из курса высшей алгебры, утверждает: любой сим- метрический многочлен можно единственным образом представить в виде многочлена от элементарных симметрических многочленов. Это представление выражается для степенных сумм по формуле Ньютона

Sk − p1Sk−1 − p2Sk−2 − · · · − pk−1S1 − kpk = 0, (k = 1, . . . , n). (22)

Отсюда получаем:



и можно найти все pk, если будут известны Sk.

Эти суммы вычисляются следующим образом:

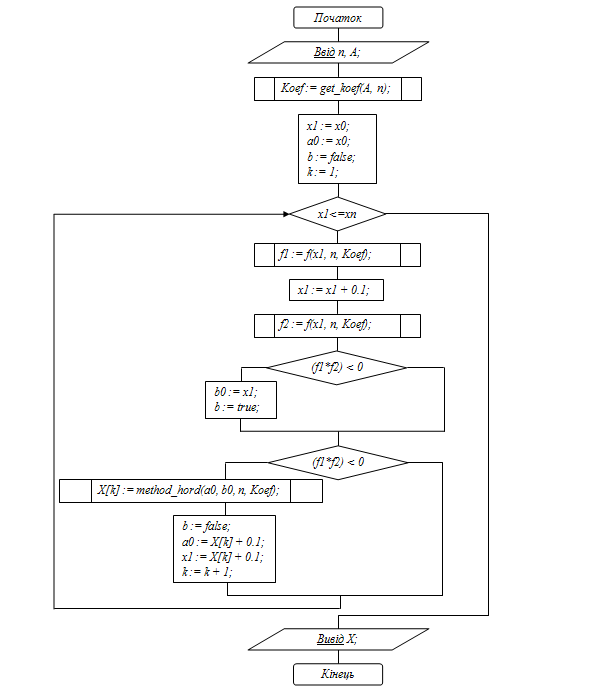
S1 = λ1 + λ2 + . . . + λn = SpA, т.е. S1 = a11 + a22 + . . . + ann.

Как известно, λ k 1 , λk 2 , · · · , λk n являются собственными значениями матрицы Ak .

Поэтому Sk = λ k 1 + λ k 2 + . . . + λ k n = SpAk , т. е. если Ak = [a (k) ij ], то Sk = a (k) 11 + a (k) 22 + . . . + a (k) nn .

Таким образом, схема раскрытия векового определителя по методу Леверье весьма про- стая, а именно: сначала вычисляются A k (k = 1, 2, . . . n) - степени матрицы A, затем нахо- дятся соответствующие Sk - суммы элементов главных диагоналей матриц Ak и, наконец, по формулам (23) определяются искомые коэффициенты pk (k = 1, 2, . . . , n).

БЛОК-СХЕМА МЕТОДА ЛЕВЕРЬЕ



ИСХОДНЫЙ КОД МЕТОДА 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++)

{

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];

}

}

}

}

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)

{

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 ПРОСЧЕТ МЕТОДА ЛЕВЕРЬЕ-ФАДЕЕВА

*Студента Костюка И.Ю.*

А = 6 3 8

3 2 3

-1 0 -4

А = А1

q1 = 6 +2-4 = 4 = Sp(A)

B1 = A1 - q\*E = 6 3 8 4 0 0 2 3 8

3 2 3 - 0 4 0 3 -2 3

-1 0 -4 0 0 4 -1 0 -8

A2 = A\*B1 = 6 3 8 \* 2 3 8 = 13 12 -7

3 2 3 3 -2 3 9 5 6

-1 0 -4 -1 0 -8 2 -3 24

SpA2 = 13 + 5 + 24 = 42

q2 = 42/2 = 21

B2 = A2 - q2\*E = 13 12 -7 - 21 0 0 = -8 12 -7

9 5 6 0 21 0 9 -16 6

2 -3 24 0 0 21 2 -3 3

A3 = A\*B2 = -5 0 0

0 -5 0

0 0 -5

SpA3 = -5-5-5 = -15

q3 = -5

B3 = ( A3 – q3\*E) = -5 0 0 - -5 0 0 = 0 0 0

0 -5 0 0 -5 0 0 0 0

0 0 -5 0 0 -5 0 0 0

p1 = 4 ; p2 = 21 ; p3 = -5

В результате получаем характеристическое уравнение:

α^3+( - 4\*α^2)+(- 21\*α) + 5 = 0

Решим его методом дихотомии:

Необходимое количество итераций n > 13

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

xє[-4,05;2,71]

x = -0,67

f(-4,05) = -41,99012 и f(-0,67) = 16,97364

Выберем отрезок [a,x] = [-4,05;-0,67]

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

x = -2,36

f(-4,05) = -41,99012 и f(-2,36) = 19,13734

Выберем отрезок [a,x] = [-4,05;-2,36]

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

x = -3,205

f(-2,36) = 19,13734 и f(-3,205) = -1,70494

Выберем отрезок [x,b] = [-3,205;-2,36]

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

x = -2,7825

f(-3,205) = -1,70494 и f(-2,7825) = 10,92031

Выберем отрезок [a,x] = [-3,205;-2,7825]

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

x = -2,99375

f(-3,205) = -1,70494 и f(-2,99375) = 5,18699

Выберем отрезок [a,x] = [-3,205;-2,99375]

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

x = -3,09938

f(-3,205) = -1,70494 и f(-3,09938) = 1,88939

Выберем отрезок [a,x] = [-3,205;-3,09938]

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

x = -3,15219

f(-3,205) = -1,70494 и f(-3,15219) = 0,12976

Выберем отрезок [a,x] = [-3,205;-3,15219]

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

x = -3,17859

f(-3,15219) = 0,12976 и f(-3,17859) = -0,77815

Выберем отрезок [x,b] = [-3,17859;-3,15219]

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

x = -3,16539

f(-3,15219) = 0,12976 и f(-3,16539) = -0,32185

Выберем отрезок [x,b] = [-3,16539;-3,15219]

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

x = -3,15879

f(-3,15219) = 0,12976 и f(-3,15879) = -0,09546

Выберем отрезок [x,b] = [-3,15879;-3,15219]

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

x = -3,15549

f(-3,15879) = -0,09546 и f(-3,15549) = 0,0173

Выберем отрезок [a,x] = [-3,15879;-3,15549]

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

x = -3,15714

f(-3,15549) = 0,0173 и f(-3,15714) = -0,03904

Выберем отрезок [x,b] = [-3,15714;-3,15549]

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

x = -3,15631

f(-3,15549) = 0,0173 и f(-3,15631) = -0,01086

Выберем отрезок [x,b] = [-3,15631;-3,15549]

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

x = -3,1559

ПРОСЧЕТ С ПОМОЩЬЮ ПРОГРАММЫ МЕТОДОМ ЛЕВЕРЬЕ

*Студента Костюка И.Ю.*

