*Дубкова Валерия КС-26*

*Вариант 5*

**Отчет по 6 лабораторной работе**

**Код**

**fun.m:**

function f = fun(x)

f(1) = 0.1 \* x(2)^2 - 10 \* x(1) - x(2);

f(2) = -3 - x(2)^3 + x(2) + exp(x(1));

end

**Lab6.m:**

clc

clear all variables

E=0.001;

syms y1 x1 x2;

%0.1 \* x(2)^2 - 10 \* x(1) - x(2)

%-3 - x(2)^3 + x(2) + exp(x(1))

s2 = 0.1 \* (x2)^2 - 10 \* x1 - x2 == 0;

s1 = -3 - (x2)^3 + x2 + exp(x1) == 0;

t2 = -3 + exp(x1);

t1 = ((x2)^2 / 100) - (1/10) \* x2;

f1 = @(x,y)double(subs(subs(t1, x1, x), x2, y));

f2 = @(x,y)double(subs(subs(t2, x1, x), x2, y));

t1 = t1 - x1;

t2 = t2 - x2;

p1 = @(x,y)double(subs(subs(t1, x1, x), x2, y));

p2 = @(x,y)double(subs(subs(t2, x1, x), x2, y));

df1x = @(x,y) double(subs(subs(diff(t1, x1), x1, x), x2, y));

df1y = @(x,y) double(subs(subs(diff(t1, x2), x1, x), x2, y));

df2x = @(x,y) double(subs(subs(diff(t2, x1), x1, x), x2, y));

df2y = @(x,y) double(subs(subs(diff(t2, x2), x1, x), x2, y));

fprintf( 'Решение стандартными методами Matlab: \n');

s = vpasolve(s1,s2);

disp([s.x1,s.x2]);

%f0 = [ p1(s.x1,s.x2) ; p2(s.x1,s.x2)];

%disp(f0);

fprintf( '\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n');

fprintf( 'Решение графически: \n');

figure

ezplot(s1, [-6 45]);

hold on

ezplot(s2, [-6 45]);

grid on

a1 = 11;

b1 = 38.5;

a2 = 0.2;

b2 = -1.5;

[xr1, fr1, ex1] = fsolve(@fun,[a1, b1],optimset('TolX',1.0e-2));

[xr2, fr2, ex2] = fsolve(@fun,[a2, b2],optimset('TolX',1.0e-2));

fprintf( 'Решение: \n');

fprintf('x1 = %f; y2 = %f;\n', xr1);

fprintf('x2 = %f; y2 = %f;\n', xr2);

fprintf( 'Проверка: \n');

fprintf('\tПроверка для x1: %f, %f\n', p1(xr1(1),xr1(2)), p2(xr1(1),xr1(2)));

fprintf('\tПроверка для x2: %f, %f\n\n', p1(xr2(1),xr2(2)), p2(xr2(1),xr2(2)));

hold on;

plot(xr1(1), xr1(2), '\*r');

plot(xr2(1), xr2(2), 'xb');

xlabel('x'); ylabel('y'); legend('0.1 \* y^2 - 10 \* x - y = 0', '-3 - y^3 + y + exp(x) = 0', 'answer 1', 'answer 2');

%plot(s.x1, s.x2, '+g');

fprintf( '\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n');

x0 = transpose(xr1);

x01 = transpose(xr2);

% Нахождения решения для первой точки

f0 = [ p1(x0(1),x0(1)) ; p2(x0(1),x0(1))];

w = [df1x(x0(1),x0(2)), df1y(x0(1),x0(2)); df2x(x0(1),x0(2)), df2y(x0(1),x0(2))]; %матрица Якоби

det(w);

x = x0 - inv(w) \* f0;

i = 0;

while (i < 200 && max(abs(f0)) > E && abs(det(w)) > E) %пока точность не будет достигнута

x0 = x;

f0=[ p1(x0(1),x0(2)); p2(x0(1),x0(2))];

w = [df1x(x0(1),x0(2)), df1y(x0(1),x0(2)); df2x(x0(1),x0(2)), df2y(x0(1),x0(2))]; %матрица Якоби

x = x - inv(w) \* f0; % новое решение

i = i + 1;

end;

%Нахождение решения для второй точки

f01 = [p1(x01(1),x01(1)); p2(x01(1),x01(1))];

w1 = [df1x(x01(1),x01(2)), df1y(x01(1),x01(2)); df2x(x01(1),x01(2)), df2y(x01(1),x01(2))];

det(w1);

x1 = x01 - inv(w1) \* f01;

i1 = 0;

while (i1 < 200 && max(abs(f01)) > E && abs(det(w1)) > E) %пока точность не будет достигнута

x01 = x1;

f01 = [ p1(x01(1),x01(2)); p2(x01(1),x01(2))];

w1 = [df1x(x01(1),x01(2)), df1y(x01(1),x01(2)); df2x(x01(1),x01(2)), df2y(x01(1),x01(2))]; %матрица Якоби

x1 = x01 - inv(w1) \* f01; % новое решение

i1 = i1 + 1;

end;

fprintf( 'Решение методом Ньютона: \n');

fprintf('x1 = %f; y2 = %f;\n', x);

fprintf('x2 = %f; y2 = %f;\n', x1);

fprintf('Проверка: \n');

fprintf('\tПроверка для x1: %f, %f\n', f0);

fprintf('\tПроверка для x2: %f, %f', f01);

fprintf('\n\nВывод:\n');

if(max(abs([p1(x(1),x(2));p2(x(1),x(2))]))<E)

fprintf( '\tПервая СЛАУ решена с помощью метода Ньютона за %d итераций с погрешностью (Е=%d)\n', i, E);

end

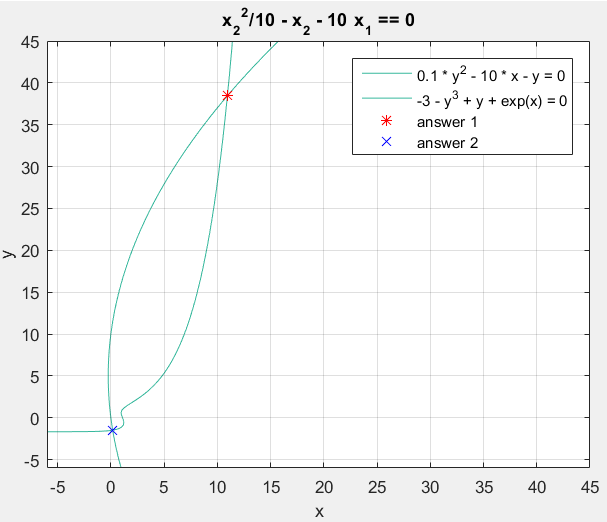
if(max(abs([p1(x1(1),x1(2)); p2(x1(1),x1(2))]))<E)

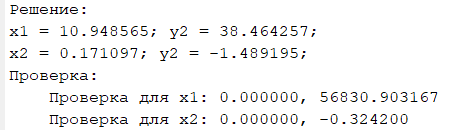
fprintf( '\tВторая СЛАУ решена с помощью метода Ньютона за %d итераций с погрешностью (Е=%d)\n', i1, E);

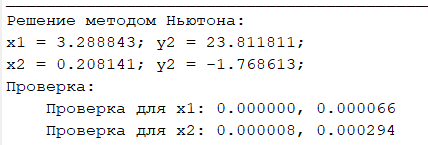
end

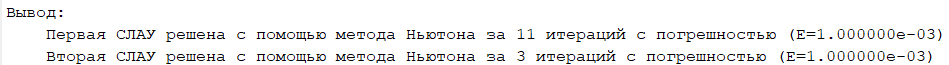
**Результат выполнения программы**

Решение графически:









**Выводы**

По результатам выполнения программы можно сказать, что в среде MatLab есть много эффективных способов для решения НУ. Так же метод Ньютона, который так же был реализован в MatLab, может довольно точно и быстро определить верный ответ с заданной точностью.

**Блок-схема для метода Ньютона**

