***Отчет о проделанной лабораторной работе № 2.4***

По предмету **: Компьютерный практикум по математическому анализу**

На тему**: Принцип сжимающих отображений. Решение уравнений и систем линейных уравнений методом итераций**

**Выполнила студентка ПИН-14**

**Марина Алина**

Зеленоград 2020

**Упражнение 1.**

Создать M-функции, которые вычисляют расстояние между точками  в различных метриках. Проверить их работу для расстояний между точкой  и точками  и  Вычислить расстояния между точками  и  в различных метриках.

function metr(x,y)

n=length(x);

p0=abs(x(1)-y(1));

for i=2:1:n

if (abs(x(i)-y(i))>tc0)

tc0=abs(x(i)-y(i));

end

end

tc1=0;

for i=1:1:n

tc1=tc1+abs(x(i)-y(i));

end

tc2=0;

for i=1:1:n

tc2=tc2+(x(i)-y(i))^2;

end

tc0, tc1, tc2=sqrt(tc2),

end

disp('Task 1')

metr(([0,0]),([3,4]))

metr(([0,0]),([4,3]))

metr(([1,2,3,4]),([7,3,4,200]))

**Если проверим А, то получим** :

tc0 = tc1 = 7tc2 = 5

где tc0 – 1 метрика, tc1 – 2 метрика, tc2 – 3 метрика

**Если проверим В , то получим :**

tc0 =4tc1 = 7tc2 = 5

**Расстояние между А и В:**

tc0 =196tc1 = 204tc2 = 196.0969

…………………………………………………………………………………………………………………………

**Упражнение 2.**

Создать M-функцию, строящую изображение замкнутого шара в  для различных метрик. Построить шары ** в метриках 

Function f= Shar(x0,r)

subplot(1,3,1);

hold on, grid on

plot([ -2.5\*r 2.5\*r], [0 0], 'k-');

plot([0 0],[ -2.5\*r\*3 2.5\*r\*3], 'k-');

for x=-r:0.1:r;

for y=-r:0.1:r;

if (abs(x-x0(1))<=r && abs(y-x0(2))<=r)

plot(x,y,'y.');

end

end

end

axis equal

subplot(1,3,2);

hold on, grid on

plot([ -2.5\*r 2.5\*r], [0 0], 'k-');

plot([0 0],[ -2.5\*r\*3 2.5\*r\*3], 'k-');

for x=-r:0.05:r;

for y=-r:0.05:r;

if (abs(x-x0(1))+abs(y-x0(2))<=r)

plot(x,y,'y.');

end

end

end

axis equal

subplot(1,3,3);

hold on, grid on

plot([ -2.5\*r 2.5\*r], [0 0], 'k-');

plot([0 0],[ -2.5\*r\*3 2.5\*r\*3], 'k-');

for x=-r:0.05:r;

for y=-r:0.05:r;

if sqrt((x-x0(1))^2+(y-x0(2))^2)<=r

plot(x,y,'y.');

end

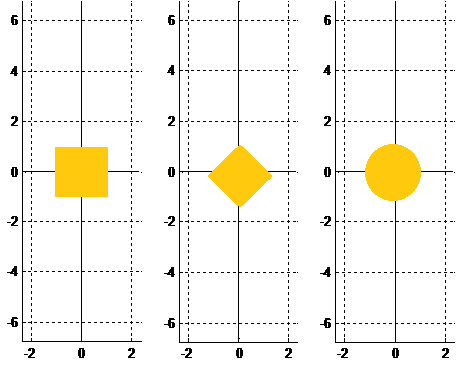
end

end

axis equal

end

Shar([0 0],0.9)



……………………………………………………………………………………………………………………………………………………………

**Упражнение 3.**

Положив ****вывести 10 первых членов последовательности  заданной рекуррентной формулой  Сделать вывод.

clear, clc

n=1, xn=3;

format long

while(n<11)

xn=atan(xn)+pi

n=n+1;

end

Command window

xn = 4.390638425988048

xn = 4.488451644891208

xn = 4.493175250270999

xn = 4.493398404996493

xn = 4.493408936315932

xn = 4.493409433294847

xn = 4.493409456747508

xn = 4.493409457854249

xn = 4.493409457906477

xn = 4.493409457908943

………………………………………………………………………………………………………………………………………

**Упражнение 4.**

СоздатьM-функцию для решения уравнения  с заданной точностью с выводом последовательности приближений. Входными параметрами являются функция  параметр сжатия  начальное приближение  точность решения  Проверить работу для уравнения из примера 2. С точностью 0.0001 решить уравнение (см. ниже, номер задания равен номеру компьютера). Сравнить с ответом, полученными при непосредственном решении в МatLab



function [ ] = num(f,a,x0,e )

A=zeros(1,2);

x1=subs(f,x0);

x2=subs(f,x1);

p0=abs(x1-x2);

n=(log(e\*(1-a))-log(p0))/log(a);

for i=1:1:(n+1);

A(i)=x0;

x0=subs(f,x0);

end

fprintf('Ряд приближений:');

disp(A);

end

disp('Task 4')

f1=xsym('atan(x)+pi');

num(f1,0.2884,3,0.001)

f2=xsym('-sin(x)');

num(f2,0,0045,1,0.0001)

Ряд приближений arctg(x)+pi: 3.0000 4.3906 4.4885 4.4932

Ряд приближений -sin(x): 1.0000 -0.8415 0.7456 -0.6784

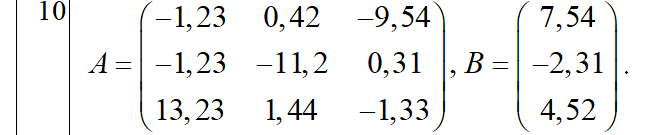
fzero(‘x+1 = x\*log(x)',1)

ans = 1.6571

………………………………………………………………………………………………………………………………….

**Упражнение 5.**

Записать систему уравнений в виде . Проверить выполнение условия сжатости матрицы . Создать M-функцию для решения методом итераций системы уравнений  с точностью  взяв в качестве начального приближения решения  Выходные параметры: приближённое решение и количество итераций.



clear,clc

A=[ -1.23 0.42 -9.54;

-1.23 -11.2 0.31;

13.23 1.44 -1.33]

B=[7.54;-2.31;4.52]

% ----------------Проверка

E=eye(3);

C=(-A+E)/22.2

%2)

for i=1:3

sum=0;

for j=1:3

sum=sum+abs(C(j,i));

end

sum

end

%1)

fori=1:3

sum=0;

for j=1:3

sum=sum+abs(C(i,j));

end

sum

end

%3)

fori=1:3

for j=1:3

c=C(i,j)^2;

end

if c>=1

disp('false')

end

end

Проверку прошёл при C=(-A+E)/22.2

function f=resh\_mat(A,B,e,C)

A1=A/C; B1=B/C;

X=B1;

Y=zeros(3);

E=eye(3);

while(abs(max(X)-max(Y))<e)

for i=1:5

Y=X;

X=B1+(-A1+E)\*X;

end

f=;

end

**X = 0.7534; -0.0246; 0.6495**