NA Codes

Bisection method

```
clc
clear all
%ingredients
f= @(x) -4*x+cos(x)+2
a=0.5;
b=0.75;
n=5
if f(a)*f(b) < 0
    for i=1:n
        c= (a+b)/2
    if f(a)*f(c)< 0
        b=c;
    elseif f(b)*f(c)< 0
        a=c;
    end
    end
else
    disp ('No root between given brackets')
end
```

Fixed-point

```
clc
clear all
%ingredients
g= input ('Enter your function:');
x0= input ('Enter initial guess:');
e= input ('Enter the tolerance:');
n= input('Enter the number of iterations:');
for i=1:n
    x1=g(x0)
    if abs(x1-x0)<e
        break
    end
    x0=x1
end
```

Newton Raphson

```
%ingredients
f=@(x) 2^x-5*x+2;
df=@(x) log(2)*2^x-5
x0=0;
e=0.0001;
n=10;
%processing
if df(x0)~=0
for i=1:n
    x1=x0-f(x0)/df(x0)
    if abs(x1-x0)<e
        break
    end
    if df(x1)==0
        disp ('Newton Raphson failed')
    end
    x0=x1;
end
else
    disp ('Newton Raphson failed')
end
```

Secant

```
%ingredients
clc
clear all
f=@(x) -4*x+cos(x)+2;
x0=0;
x1=0.8;
e=0.0001;
n=5;
%processing
for i=1:n
    x2= (x0*f(x1)-x1*f(x0))/ (f(x1)-f(x0))
    x0=x1;
    x1=x2;
end
```

Gauss Elimination

```
clc
clear
n=input('enter order of matrix');
a=[1 0 1 1 ;-1 1 2 1 ;2 1 0 0 ]
% forward elimination
for j=1:n
for i=1:n
if i>j
xmult = a(i,j)/a(j,j);
for k=1:n+1
a(i,k) = a(i,k)-xmult*a(j,k);
end
end
end
end
a
% back substitution.
x(n) = a(n,n+1)/a(n,n);
for i=n-1:-1:1
sum = 0;
for j=i+1:n
sum = sum+a(i,j)*x(j);
end
x(i) =(a(i,n+1)- sum)/a(i,i);
end
x
```

Gauss seidel

```
clc;
clear all;
a=[10 -1 2 0;-1 11 -1 3;2 -1 10 -1;0 3 -1 8];
b=[6;25;-11;15];
x0=[0;0;0;0];
x=[0;0;0;0];
n=size(a,1);
k=1;
while k<=100
for i=1:n
    sum=0;
    sum1=0;
    for j=1:i-1
        sum=sum+a(i,j)*x(j);
    end
    for j=i+1:n
        sum1=sum1+a(i,j)*x0(j);
    end
    x(i)=(1/a(i,i))*(b(i)-sum-sum1);
end
if abs(x-x0)<0.0001
    break;
end
x0=x;
k=k+1;
end
x
```

Power method

```
clc
clear all
x0=[1;-1;2]
A=[2 1 1;1 2 1;1 1 2]
tol=10^-3;
N=1000;
 y=A*x0;
     m1=max(abs(y));
i=0;
while i<=N
i=i+1;
    x=(1/m1).*y;
     y=A*x;
     m2=max(abs(y));
    if abs(m2-m1)<tol
        fprintf(' max eigenvalue =%f',m2);
        break;
    end
    m1=m2;
end
   x
```