Praveen Kumar Yadav

(MTech VLSI Sem 1, EE22M308)

Lab no. : 3

Used version: Matlab 2022

Q.1:- Use finite difference method to solve the Schrodinger's equation for a particle in a 1D box with mass same as that of an electron and well width of 10 nm. Use 101 points for discretization. Plot the energy values vs the eigenvalue number and compare it with the analytical results. Also plot the probability distribution for eigenvalues 1 and 25. Instead of 101 points, use 1001 points for discretization and redraw the plots. What effect do you see? Also check if the wavefunction you are getting is normalized.

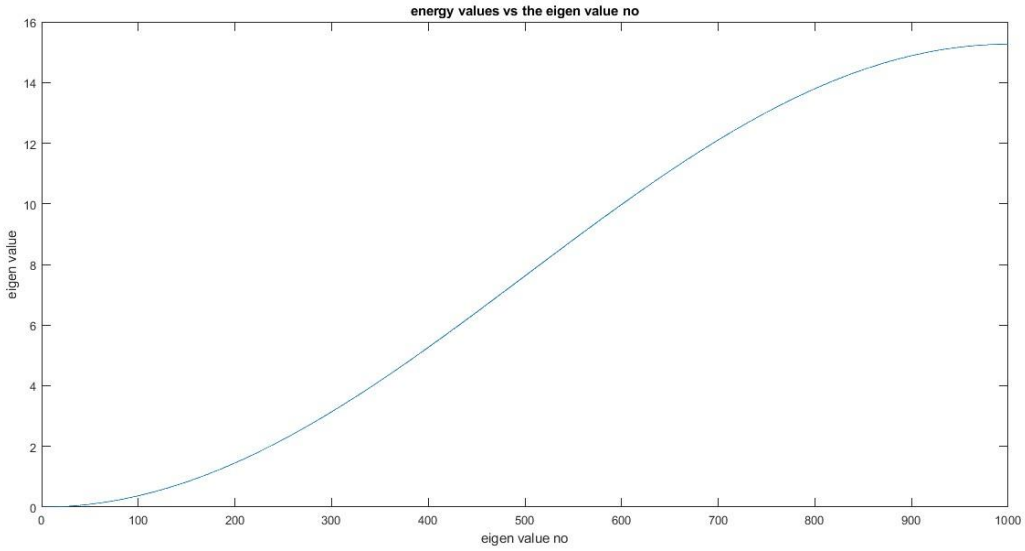
Ans:

FDM:

For 101 points

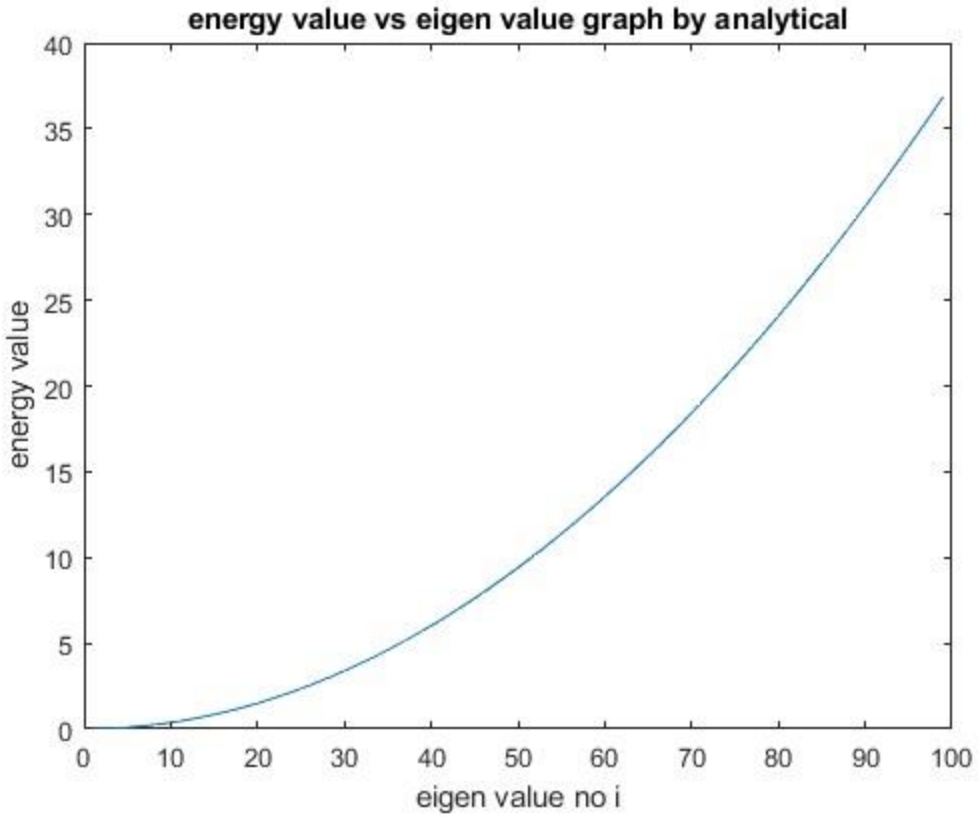
```
clear all;
clc;
q=1.6*(10)^(-19);
m=9.1*10^(-31);
pie=3.14;
h=6.62*1*10^(-34);
a=10*10^(-9);
j=a/100
t0=-((h/(2*pie))^2)/(2*m*((j)^2))/q;
EVN=[];
for i=1:1000
    H(i,i)=-2*t0 ;
    EVN = [EVN, i];
end
for i=1:999
    H(i,i+1)=t0;
end
for i=1:999
    H(i+1,i)=t0;
end
[V,D] = eig(H);
```

```
EV = diag(D);
EVT = EV;
plot(EVN,EV);
xlabel('eigen value no');
ylabel('eigen value');
title('energy values vs the eigen value no');
```



Analytical :

```
clc;
m=9.1*10^(-31);
q=1.6*10^(-19);
h=6.62*1*10^(-34);
a=10*10^(-9);
EN=[];
for i=1:99
    E=(((i*h)/a)^2)/(8*m*q);
    EN =[EN,E];
end
disp(EN);
i=1:99;
plot(i,EN)
xlabel('eigen value no i');
ylabel('energy value');
title('energy value vs eigen value graph by analytical');
```

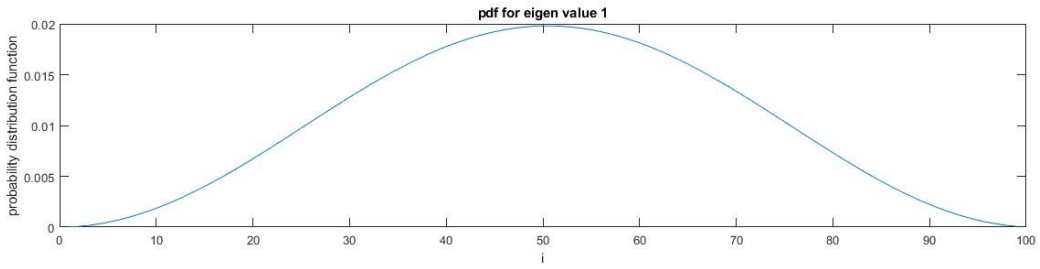


Pdf:

For eigen value 1

```
clear all;
clc;
q=1.6*(10)^(-19);
aifsilon=8.85*(10)^(-12);
m=9.1*10^(-31);
h=6.62*1*10^(-34);
h_fn=h/(2*3.14);
a=1*10^(-10);
t0=-(h_fn^2)/(2*m*a*a);
for i=1:100
    H(i,i)=-(2*t0);
end
for i=1:99
    H(i,i+1)=t0;
end
for i=1:99
    H(i+1,i)=t0;
end
[V,D] = eig(H);
EV = diag(D);
i=1:100;
plot(i,EV)
```

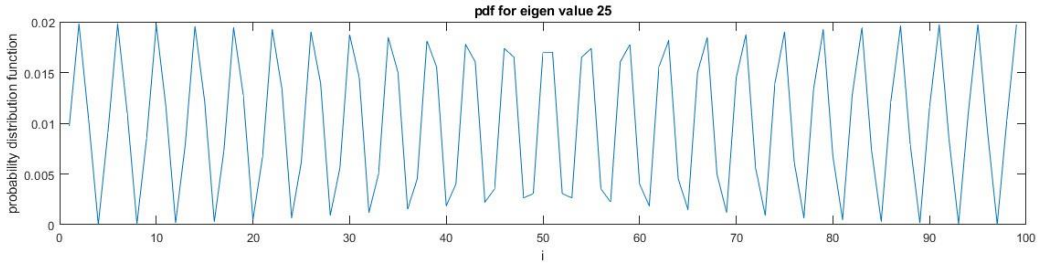
```
f1=[];
V(1:99,1);
f1=[f1,V(1:99,1)];
subplot(2,1,1)
i=1:99;
plot(i,f1.*f1)
xlabel('i');
ylabel('probability distribution function');
title('pdf for eigen value 1');
```



Pdf for eigen value 25

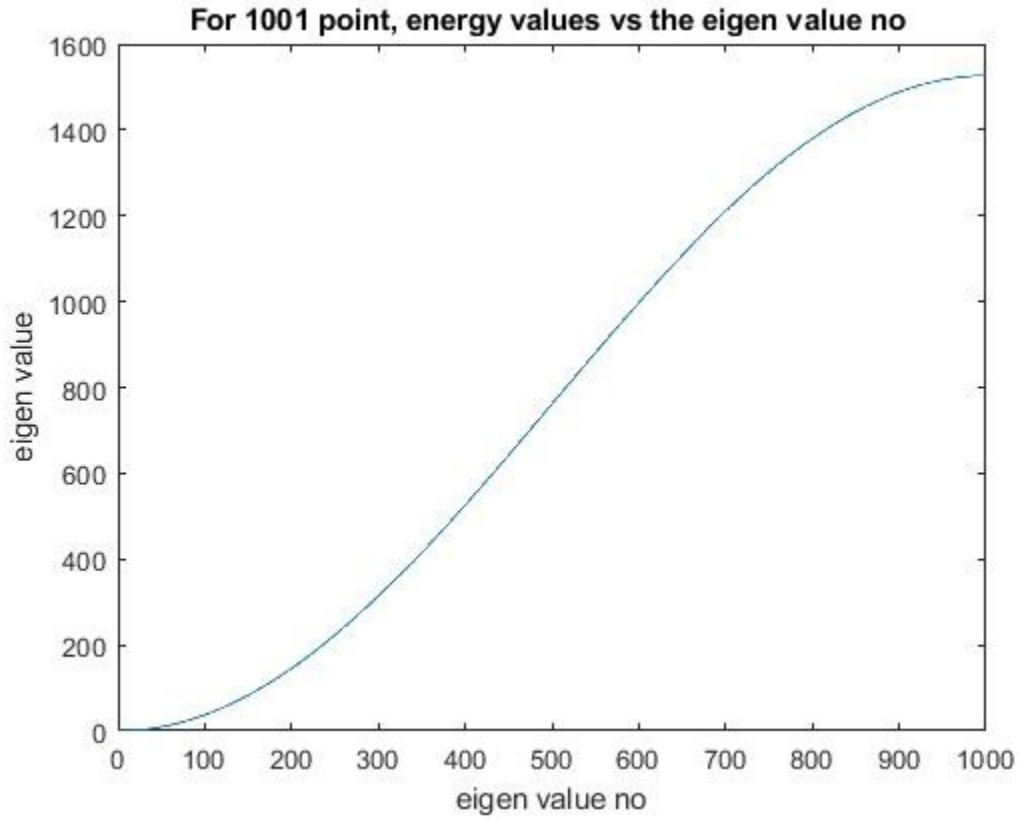
```
clear all;
clc;
q=1.6*(10)^(-19);
aifsilon=8.85*(10)^(-12);
m=9.1*10^(-31);
h=6.62*1*10^(-34);
h_fn=h/(2*3.14);
a=1*10^(-10);
t0=-(h_fn^2)/(2*m*a*a);
for i=1:100
    H(i,i)=-(2*t0);
end
for i=1:99
    H(i,i+1)=t0;
end
for i=1:99
    H(i+1,i)=t0;
end
[V,D] = eig(H);
EV = diag(D);
i=1:100;
plot(i,EV)
f1=[];
```

```
V(1:99,25);
f1=[f1,V(1:99,25)];
subplot(2,1,1)
i=1:99;
plot(i,f1.*f1)
xlabel('i');
ylabel('probability distribution function');
title('pdf for eigen value 25');
```



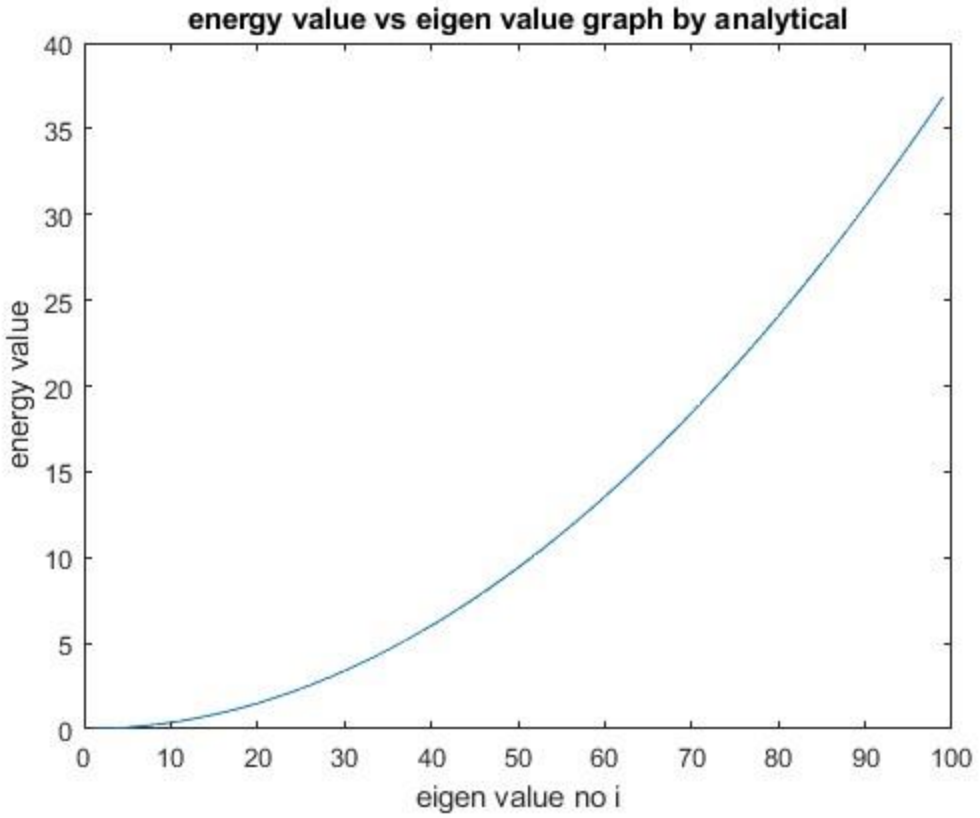

```
FOR 1001 POINTS
ENERGY VS EIGEN VALUE GRAPH
clear all;
clc;
q=1.6*(10)^(-19);
m=9.1*10^(-31);
pie=3.14;
h=6.62*1*10^(-34);
a=10*10^(-9);
j=a/1000
t0=-((h/(2*pie))^2)/(2*m*((j)^2))/q;
EVN=[];
for i=1:1000
    H(i,i)=-2*t0 ;
    EVN = [EVN, i];
end
for i=1:999
    H(i,i+1)=t0;
end
for i=1:999
    H(i+1,i)=t0;
end
[V,D] = eig(H);
EV = diag(D);
```

```
EVT = EV;
plot(EVN,EV);
xlabel('eigen value no');
ylabel('eigen value');
title('For 1001 point, energy values vs the eigen value no ');
```



ANALYTICAL

```
clc;
m=9.1*10^(-31);
q=1.6*10^(-19);
h=6.62*1*10^(-34);
a=10*10^(-9);
EN=[];
for i=1:99
    E=(((i*h)/a)^2)/(8*m*q);
    EN =[EN,E];
end
disp(EN);
i=1:99;
plot(i,EN)
xlabel('eigen value no i');
ylabel('energy value');
title('energy value vs eigen value graph by analytical');
```

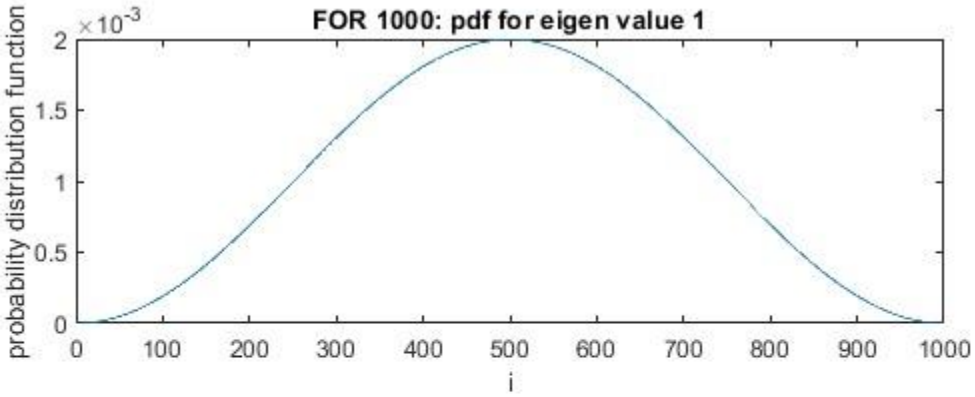


PDF FOR 1001 POINTS

FOR EIGEN VALUE 1

```
clear all;
clc;
q=1.6*(10)^(-19);
aifsilon=8.85*(10)^(-12);
m=9.1*10^(-31);
h=6.62*1*10^(-34);
h_fn=h/(2*3.14);
a=1*10^(-11);
t0=-(h_fn^2)/(2*m*a*a);
for i=1:1000
    H(i,i)=(-(2*t0));
end
for i=1:999
    H(i,i+1)=t0;
end
for i=1:999
    H(i+1,i)=t0;
end
[V,D] = eig(H);
EV = diag(D);
i=1:1000;
plot(i,EV)
```

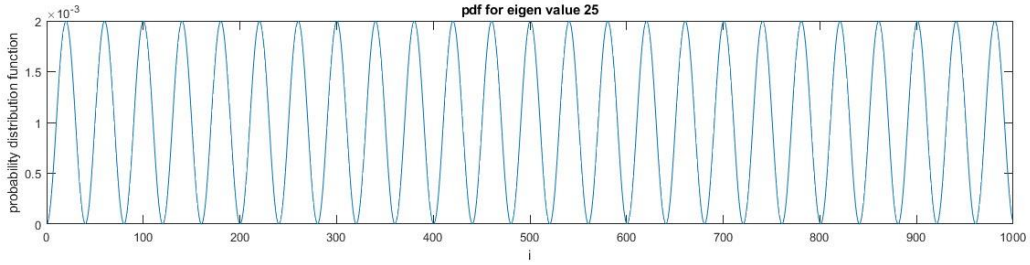
```
f1=[];
V(1:999,1);
f1=[f1,V(1:999,1)];
subplot(2,1,1)
i=1:999;
plot(i,f1.*f1)
xlabel('i');
ylabel('probability distribution function');
title('FOR 1000: pdf for eigen value 1');
```



PDF FOR EIG 25

```
clear all;
clc;
q=1.6*(10)^(-19);
aifsilon=8.85*(10)^(-12);
m=9.1*10^(-31);
h=6.62*1*10^(-34);
h_fn=h/(2*3.14);
a=1*10^(-11);
t0=-(h_fn^2)/(2*m*a*a*q);
for i=1:1000
    H(i,i)=-(2*t0);
end
for i=1:999
```

```
    H(i,i+1)=t0;
end
for i=1:999
    H(i+1,i)=t0;
end
[V,D] = eig(H);
EV = diag(D);
i=1:1000
plot(i,EV);
f1=[];
V(1:999,25);
f1=[f1,V(1:999,25)];
subplot(2,1,1);
i=1:999;
plot(i,f1.*f1)
xlabel('i');
ylabel('probability distribution function');
title('pdf for eigen value 25');
```

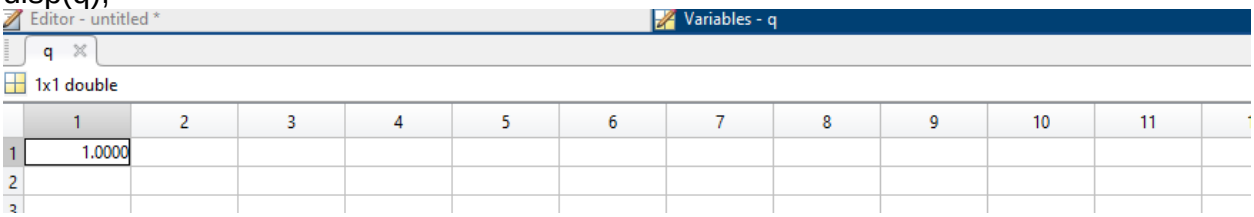


Discretization effect: discretization effects in finite-difference simulations of blowup solutions of the nonlinear Schrödinger equation (NLS) initially accelerate self focusing but later arrest the collapse, resulting instead in focusing–defocusing oscillations.

Normalization:

```
clear all;
clc;
q=1.6*(10)^(-19);
aifsilon=8.85*(10)^(-12);
m=9.1*10^(-31);
```

```
h=6.62*1*10^(-34);
h_fn=h/(2*3.14);
b=10^(-9);
a=1*10^(-11);
t0=-(h_fn^2)/(2*m*a*a*q);
for i=1:1000
    H(i,i)=-(2*t0);
end
for i=1:999
    H(i,i+1)=t0;
end
for i=1:999
    H(i+1,i)=t0;
end
[V,D] = eig(H);
EV = diag(D);
i=1:1000;
plot(i,EV);
f1=[];
V(1:999,25);
f1=[f1,V(1:999,25)];
subplot(2,1,1);
i=1:999;
plot(i,f1.*f1);
y=(f1.*f1);
q = sum(y);
disp(q);
```



Yes wavefunction is normalized.

Q.2:- The radial wavefunction for an electron inside the H atom can be given by

$$Ef(r) = \left(-\frac{\hbar^2}{2m}\frac{\mathrm{d}^2}{\mathrm{d}r^2} + U(r) + \frac{l(l+1)\hbar^2}{2mr^2} \right) f(r)$$

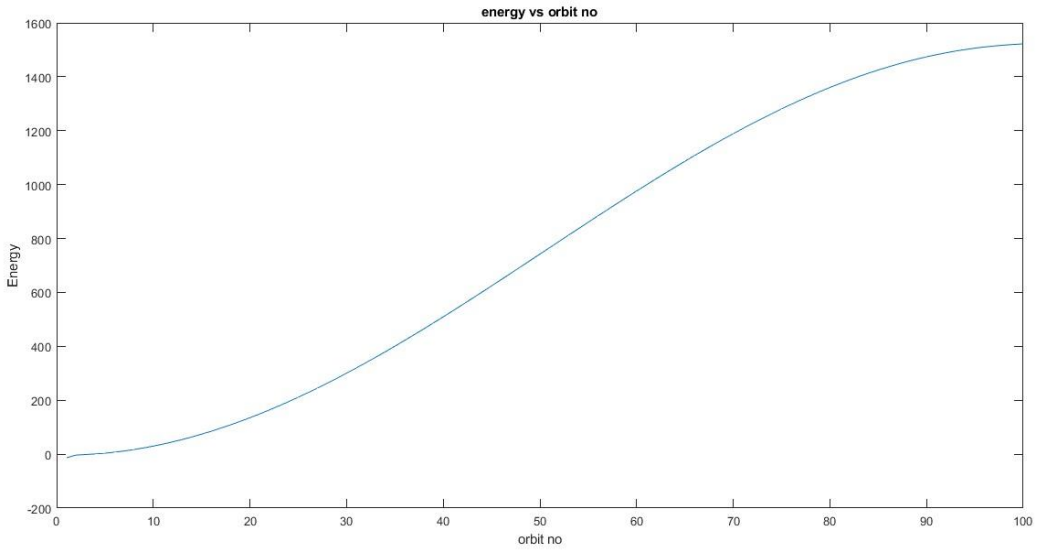
Use the finite difference method to find out the energy of the 1s & 2s levels. Plot the radial probability as a function of the radius for the mentioned energy levels. Can you

obtain the bohr's radius from these graphs? Compare the value of bohr's radius you obtain from these graphs to the analytical value. Also check if the wavefunctions you are getting are normalized.

Ans:

1S AND 2S ENERGY USING FDM:

```
clear all;
clc;
q=1.6*10^(-19);
epsilon=8.85*10^(-12);
m=9.1*10^(-31);
h=6.62*1*10^(-34);
h_fn=h/(2*pi);
a=0.1*10^(-10);
t0=-(h_fn*h_fn)/(2*m*a*a*q);
for i=1:100
    H(i,i)=-(2*t0)-(q*q/(4*pi*epsilon*a*i*q));
end
for i=1:99
    H(i,i+1)=t0;
end
for i=1:99
    H(i+1,i)=t0;
end
[V,D] = eig(H);
EV = diag(D);
i=1:100;
plot(i,EV);
xlabel('orbit no');
ylabel('Energy');
title('energy vs orbit no');
```



Probability as a function of radius:

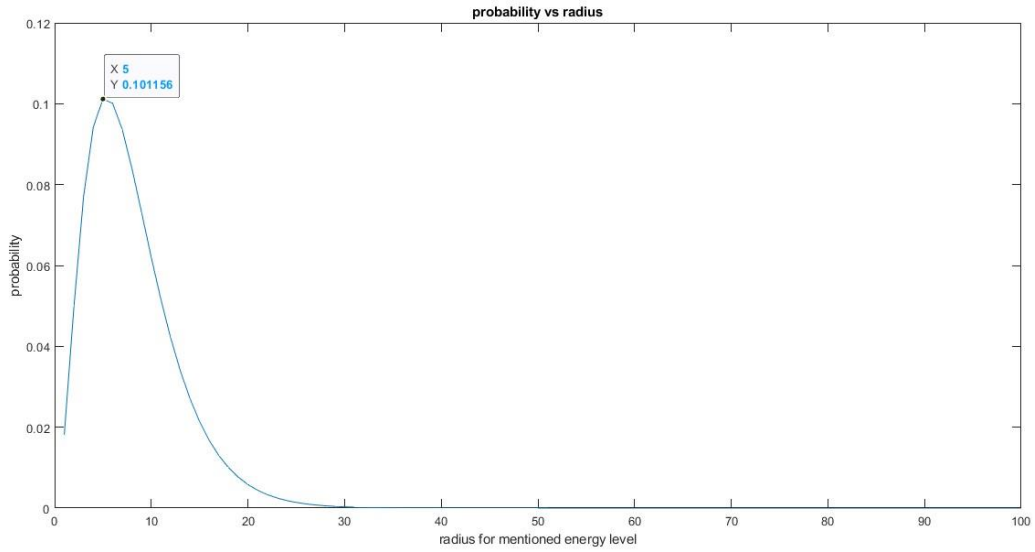
```
clear all;
clc;
q=1.6*10^(-19);
epsilon=8.85*10^(-12);
m=9.1*10^(-31);
h=6.62*1*10^(-34);
h_fn=h/(2*pi);
a=0.1*10^(-10);
t0=-(h_fn*h_fn)/(2*m*a*a*q);
for i=1:100
    H(i,i)=-(2*t0)-(q*q/(4*pi*epsilon*a*i*q));
end
for i=1:99
    H(i,i+1)=t0;
end
for i=1:99
    H(i+1,i)=t0;
end
[V,D] = eig(H);
EV = diag(D);

f1=[];

f1=[f1,V(:,1)];
%subplot(2,1,1)
i=1:100;
plot(i,f1.*f1)
```

```
f2=[];

 f2=[f2,V(:,2)];
%subplot(2,1,2)
i=1:100;
plot(i,f2.*f2)
xlabel('radius for mentioned energy level');
ylabel('probability');
title('probability vs radius');
```



Radius=5*spacing=0.5Angustron

ANALYTICAL:

```
clear all;
clc;
q=1.6*10^(-19);
epsilon=8.85*10^(-12);
m=9.1*10^(-31);
h=6.62*1*10^(-34);
h_fn=h/(2*pi);
a0=(4*pi*epsilon*(h_fn)^2)/(m*(q)^2);
disp(a0);
```

compare: we got almost same radius

NORMALIZATION:

```
clear all;
clc;
q=1.6*10^(-19);
epsilon=8.85*10^(-12);
m=9.1*10^(-31);
h=6.62*1*10^(-34);
h_fn=h/(2*pi);
```

```
a=0.1*10^(-10);
t0=-(h_fn*h_fn)/(2*m*a*a*q);
for i=1:100
    H(i,i)=-(2*t0)-(q*q/(4*pi*epsilon*a*i*q));
end
for i=1:99
    H(i,i+1)=t0;
end
for i=1:99
    H(i+1,i)=t0;
end
[V,D] = eig(H);
EV = diag(D);

f1=[];

f1=[f1,V(:,1)];
%subplot(2,1,1)
i=1:100;
plot(i,f1.*f1)

f2=[];

f2=[f2,V(:,2)];
%subplot(2,1,2)
i=1:100;
%plot(i,f2.*f2)
y=(f1.*f1);
q = sum(y);
disp(q);
```

yes wavefunction is normalized.

Q.3:- Consider a finite potential well with α0a=π/4. Use graphical method to solve for the energy value of an electron subjected to this potential well. Assume that the well is 1 nm wide. Use the finite difference method to solve for the energy of the electron and compare the result with the graphical method. Also plot the wavefunction of the electron obtained using the finite difference method. Is the wavefunction normalized?
ANS:
GRAPHICAl:

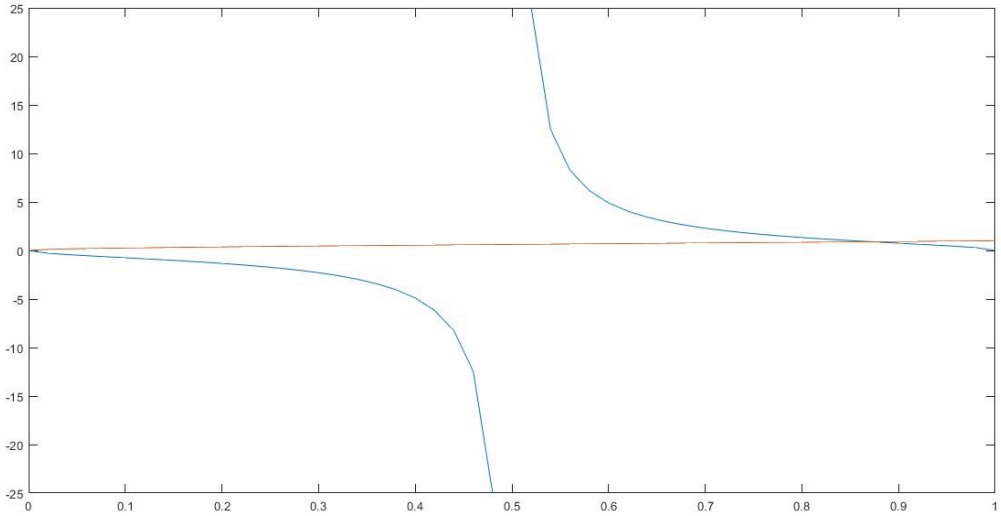
```
clear all;
clc;
q=1.6*10^(-19);
epsilon=8.85*10^(-12);
m=9.1*10^(-31);
h=6.62*1*10^(-34);
```

```
h_fn=h/(2*pi);
a=0.01*10^(-10);
t0=-(h_fn*h_fn)/(2*m*a*a*q);
for i=1:1000
    H(i,i)=-(2*t0)-(q*q/(4*pi*epsilon*a*i*q));
end
for i=1:999
    H(i,i+1)=t0;
end
for i=1:999
    H(i+1,i)=t0;
end
for i=1001:1099
    H(i,i)=-2*t0 ;

end
for i=1001:1099
    H(i,i+1)=t0;
end
for i=1001:1099
    H(i+1,i)=t0;
end
for i=1100:2100
    H(i,i)=-2*t0-(q*q/(4*pi*epsilon*a*i*q)) ;
end
for i=1100:2099
    H(i,i+1)=t0;
end
for i=1100:2099
    H(i+1,i)=t0;
end
[V,D] = eig(H);
EV = diag(D);
eign1=V(:,1);
i=1:2100;
plot(i,eign1);
x=0:0.02:1;

f1=tan((pi/4)*(x).^(0.5));
f3=[];
a=(10^(-9)/2100);
for i=0:0.02:1;
    f2=(2*(i.*(1-i)).^0.5/((2*i)-1));
    f3=[f3,f2];
end
% plot(x,f1);
```

```
plot(x,f3,x,f1);
zeeta=.88;
alpha=pi/(4*a);
k=alpha*(zeeta)^0.5;
E=(((k*h_fn)^2)/2*m);
```

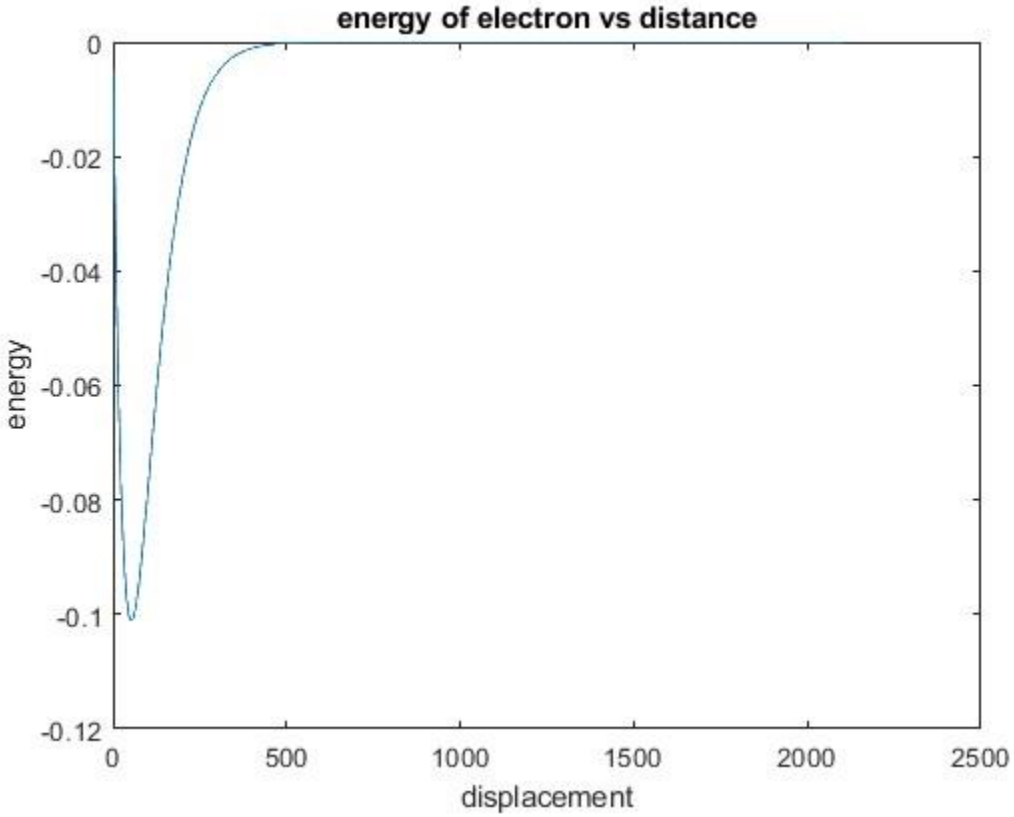


FDM:

```
clear all;
clc;
q=1.6*10^(-19);
epsilon=8.85*10^(-12);
m=9.1*10^(-31);
h=6.62*1*10^(-34);
h_fn=h/(2*pi);
a=0.01*10^(-10);
t0=-(h_fn*h_fn)/(2*m*a*a*q);
for i=1:1000
    H(i,i)=-(2*t0)-(q*q/(4*pi*epsilon*a*i*q));
end
for i=1:999
    H(i,i+1)=t0;
end
for i=1:999
    H(i+1,i)=t0;
end
for i=1001:1099
    H(i,i)=-2*t0 ;

end
```

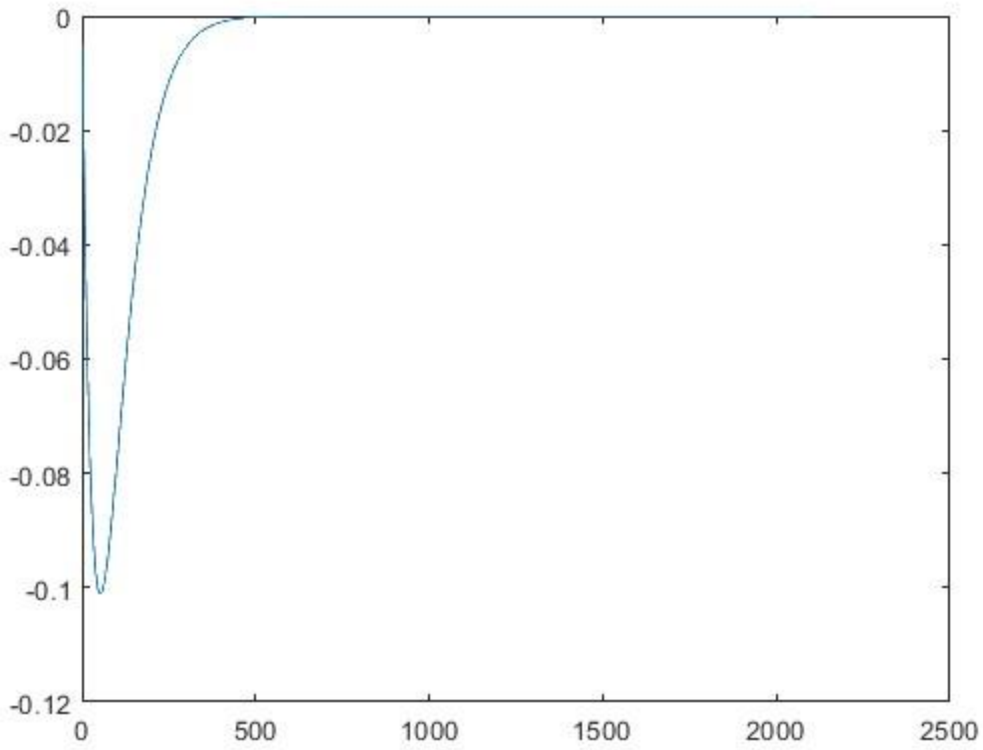
```
for i=1001:1099
    H(i,i+1)=t0;
end
for i=1001:1099
    H(i+1,i)=t0;
end
for i=1100:2100
    H(i,i)=-2*t0-(q*q/(4*pi*epsilon*a*i*q)) ;
end
for i=1100:2099
    H(i,i+1)=t0;
end
for i=1100:2099
    H(i+1,i)=t0;
end
[V,D] = eig(H);
EV = diag(D);
eign1=V(:,1)
i=1:2100;
plot(i,eign1)
xlabel('displacement');
ylabel('energy');
title('energy of electron vs distance');
```



WAVEFUNCTION:

```
clear all;
clc;
q=1.6*10^(-19);
epsilon=8.85*10^(-12);
m=9.1*10^(-31);
h=6.62*1*10^(-34);
h_fn=h/(2*pi);
a=0.01*10^(-10);
t0=-(h_fn*h_fn)/(2*m*a*a*q);
for i=1:1000
    H(i,i)=-(2*t0)-(q*q/(4*pi*epsilon*a*i*q));
end
for i=1:999
    H(i,i+1)=t0;
end
for i=1:999
    H(i+1,i)=t0;
end
for i=1001:1099
    H(i,i)=-2*t0 ;

end
for i=1001:1099
    H(i,i+1)=t0;
end
for i=1001:1099
    H(i+1,i)=t0;
end
for i=1100:2100
    H(i,i)=-2*t0-(q*q/(4*pi*epsilon*a*i*q)) ;
end
for i=1100:2099
    H(i,i+1)=t0;
end
for i=1100:2099
    H(i+1,i)=t0;
end
[V,D] = eig(H);
EV = diag(D);
eign1=V(:,1);
i=1:2100;
plot(i,eign1)
```



NORMALIZATION:

```
clear all;
clc;
q=1.6*10^(-19);
epsilon=8.85*10^(-12);
m=9.1*10^(-31);
h=6.62*1*10^(-34);
h_fn=h/(2*pi);
a=0.01*10^(-10);
t0=-(h_fn*h_fn)/(2*m*a*a*q);
for i=1:1000
    H(i,i)=-(2*t0)-(q*q/(4*pi*epsilon*a*i*q));
end
for i=1:999
    H(i,i+1)=t0;
end
for i=1:999
    H(i+1,i)=t0;
end
for i=1001:1099
    H(i,i)=-2*t0 ;
```

```
end
for i=1001:1099
    H(i,i+1)=t0;
end
for i=1001:1099
    H(i+1,i)=t0;
end
for i=1100:2100
    H(i,i)=-2*t0-(q*q/(4*pi*epsilon*a*i*q)) ;
end
for i=1100:2099
    H(i,i+1)=t0;
end
for i=1100:2099
    H(i+1,i)=t0;
end


 [V,D] = eig(H);
EV = diag(D);
i=1:2100;
plot(i,EV);
f1=[];
V(1:2099,25);
f1=[f1,V(1:2099,25)];
subplot(2,1,1);
i=1:2099;
plot(i,f1.*f1);
y=(f1.*f1);
q = sum(y);
disp(q);
YES wavefunction normalized.
```