Computational Physics Assignment 8

Dara Corr 18483836

In this assignment we look to solving the Schroedinger Equation numerically. The problem we look at is a potential well problem where Schroedinger's Equation is of the following form:

$$\frac{d^2\psi(x)}{dx^2} = -\frac{2m}{\hbar^2}\left[E - V(x)\right]\psi(x)$$

Considering an electron in a box of size 1nm and using energy units in eV, the equation reduces to:

$$\frac{d^2\psi(x)}{dx^2} = -26.2\,(E - V)\,\psi(x)$$

The first part of the assignment is to solve for the first even and odd states of the infinite potential well where V = 0 inside the well and v is very large outside the well so that $\psi(x) = 0$ outside the well. We apply the boundary conditions $\psi(0) = 1$, $\frac{d\psi}{dx} = 0$, $\psi(a) = \psi(-a) = 0$ and $a = 1$. Using the Euler method or Runge-Kutta method we can find an array of psi values over a range of x values from -a to a which we can plot ψ vs x and the probability distribution $|\psi|^2$ vs x. Using the Shooting method, it is possible to find the value of E of the ground state by substituting different values of E until $\psi(1) = 0$ is found.

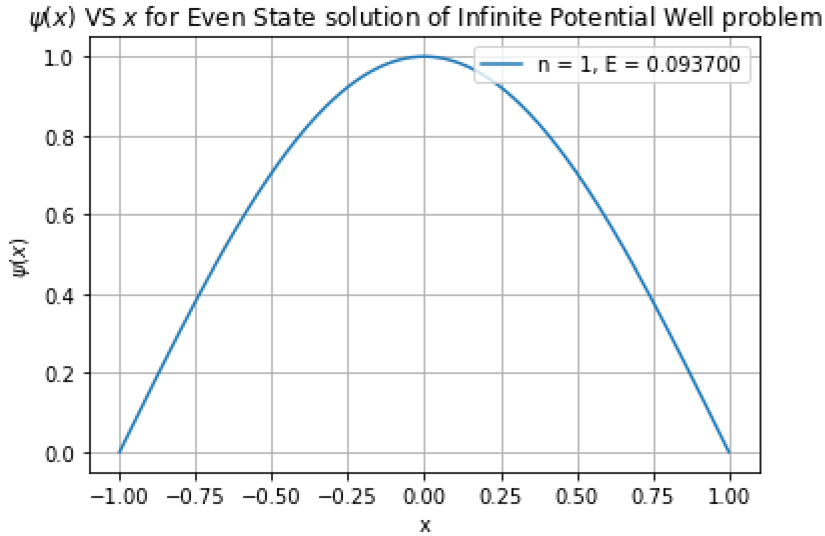
By changing boundary conditions $\frac{d\psi}{dx} = 1$ and $\psi(0) = 0$, it is possible to find the first odd state for the infinite potential well using the shooting method. By increasing the energy values it is then possible to find 2nd, 3rd,... even and odd states for this problem. From this exercise we find energy for each state is $E = E_0 n^2$ where E_0 is the ground state energy and n is the number corresponding to each state.

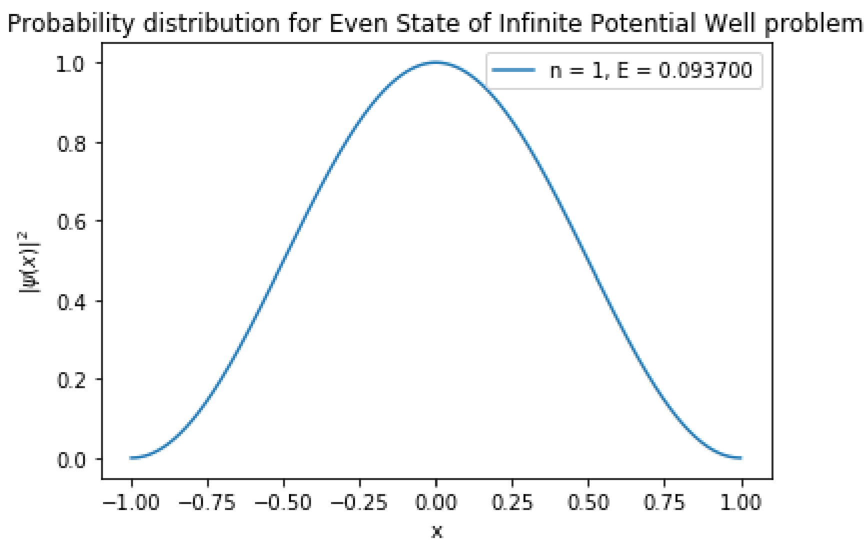
The second part of this assignment is to solve for the finite potential well where for $x > |a|$, $V = 1$ eV. Here the Euler method must be altered to account for the V term when x > |a|. We use the shooting method again to find values for E where the solution dies down to 0 for x > |a| instead of exponentially increasing or decreasing. We find that bound states only exist for E < V and solutions where E > V, the particle can go beyond the potential barrier because it has more energy than the potential barrier of V.

In [170]:
```
import numpy as np
import matplotlib.pyplot as plt

#psi(x)'' = -26.2(E-V)psi(x) schroedinger eqn for electron in box of size 1 nm
in eV units



#[
#solve seperately for regions of constant potential (i.e. inside box, outside,
box, boundaries?)

#set d(psi)/dx = Y, dY/dx = -26.2(E-V)*psi ]

def Euler_method(a,dx,E,psi_0,dpsi_0):
    #a = x_max, dx = step size, E = Energy eigenvalue, psi_0 = initial value o
f psi, dpsi_0 =  initial change in psu
    N= int(a/dx)# Number of steps
    X = np.zeros(N+1) #x values
    psi = np.zeros(N+1) #psi values
    dpsi = np.zeros(N+1) #d(psi)/dx values

    psi[0] = psi_0 #set initial value of psi (psi(0))
    dpsi[0] = dpsi_0 #set initial value of d(psi)/dx (psi'(0))

    for x in range(N):
        d2psi = -26.2*E*psi[x] #using Euler's Method to identify psi values fo
r each time step
        dpsi[x+1] = dpsi[x] + d2psi*dx
        psi[x+1] = psi[x] + dpsi[x+1]*dx
        X[x+1] = X[x] + dx

    return psi, X




def infinite_well_even_plot(a,dx,E,psi_0,dpsi_0,n): #plots even psi soln again
st x

    #function uses euler's method

    psi = Euler_method(a,dx,E,psi_0,dpsi_0)[0]
    X = Euler_method(a,dx,E,psi_0,dpsi_0)[1]

    #make plot symmetric by flipping the X and y arrays (X and Psi) and then a
ppending the origninal arrays
    X2 = -X
    Xtot = np.append(X2[::-1],X)
    psi_tot = np.append(psi[::-1],psi)

    plt.plot(Xtot,psi_tot, label='n = {0:1}, E = {1:4f}'.format(n, E))
    plt.title("$\psi (x)$ VS $x$ for Even State solution of Infinite Potential
Well problem")
    plt.xlabel("x")
    plt.ylabel("$\psi (x)$")
    plt.legend(loc='upper right')
```

```
    plt.grid()


def infinite_well_even_probs(a,dx,E,psi_0,dpsi_0,n): #plots psi**2 against x

     #function uses euler's method

    psi = Euler_method(a,dx,E,psi_0,dpsi_0)[0]
    X = Euler_method(a,dx,E,psi_0,dpsi_0)[1]

    #make plot symmetric by flipping the X and y arrays (X and Psi) and then a
ppending the orignial arrays
    X2 = -X
    Xtot = np.append(X2[::-1],X)
    psi_tot = np.append(psi[::-1],psi)


    plt.plot(Xtot, psi_tot**2,label='n = {0:1}, E = {1:4f}'.format(n, E) )#plo
t
    plt.title("Probability distribution for Even State of Infinite Potential W
ell problem".format(n))
    plt.xlabel("x")
    plt.ylabel("$|\psi (x)|^2$")
    plt.legend(loc='upper right')




#ground state even soln:
#psi(0) = 1, dpsi_0/dx = 0
#psi(-a) = psi(a) = 0

#guess E so that it satisfies boundary psi(a) = 0 = psi(-a) [shooting method],
which gives us our value for a
#a = 1 since box is size of 1nm and wave eqn is in terms of nm and eV


print("The Energy Eigenvalue I found for the Ground State Electron is E = 0.09
37 eV")

infinite_well_even_plot(1,0.005,0.0937,1,0,n=1)
plt.show()

infinite_well_even_probs(1,0.005,0.0937,1,0,n=1)
plt.show()
```

```
The Energy Eigenvalue I found for the Ground State Electron is E = 0.0937 eV
```

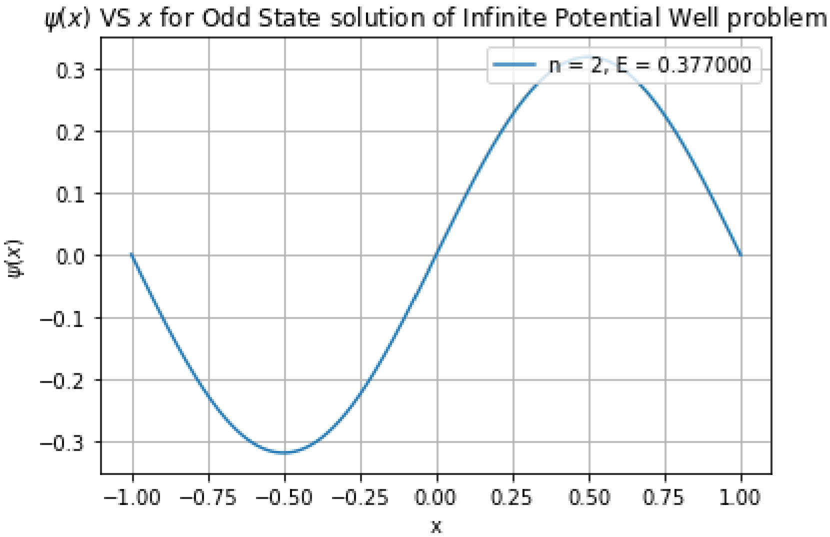


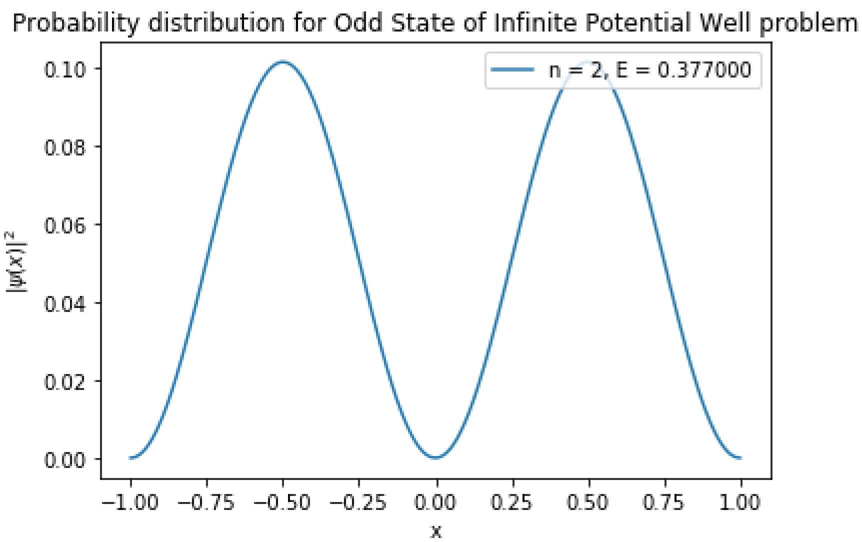


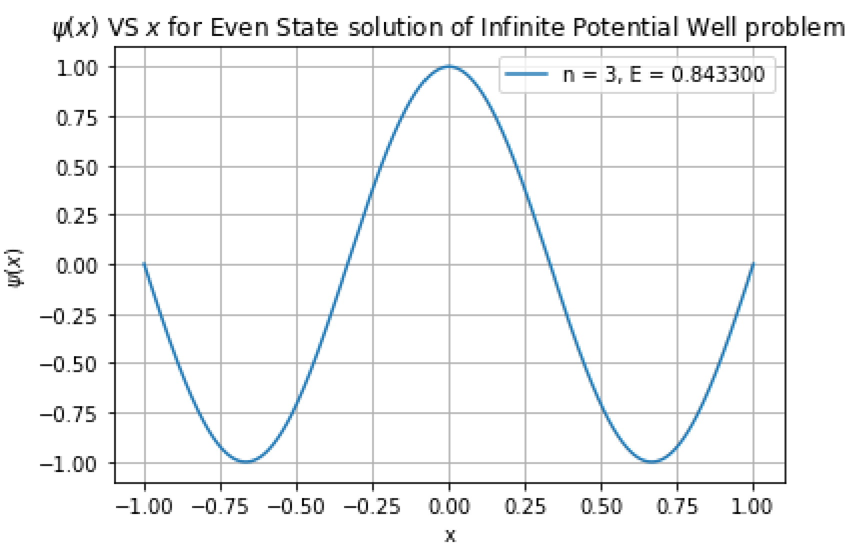
In [158]:

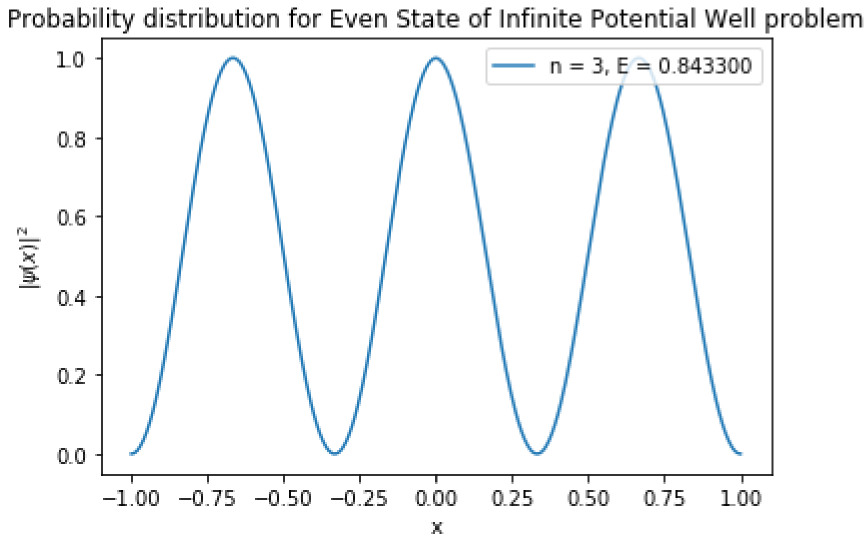
```
#task 2


def infinite_well_odd_plot(a,dx,E,psi_0,dpsi_0, n):

    #function uses euler's method

    psi = Euler_method(a,dx,E,psi_0,dpsi_0)[0]
    X = Euler_method(a,dx,E,psi_0,dpsi_0)[1]

    #plot negative region, note psi(-x) = -psi(x)
    X2 = -X
    Xtot = np.append(X2[::-1], X)
    psi_tot = np.append(-psi[::-1], psi)#minus sign is included on psi for x>0
since it is an odd function here

    plt.plot(Xtot,psi_tot, label='n = {0:1}, E = {1:4f}'.format(n, E))
    plt.title("$\psi (x)$ VS $x$ for Odd State solution of Infinite Potential
 Well problem")
    plt.xlabel("x")
    plt.ylabel("$\psi (x)$")
    plt.grid()
    plt.legend(loc='upper right')


def infinite_well_odd_probs(a,dx,E,psi_0,dpsi_0, n):

    #function uses euler's method

    psi = Euler_method(a,dx,E,psi_0,dpsi_0)[0]
    X = Euler_method(a,dx,E,psi_0,dpsi_0)[1]

    #plot negative region, note psi(-x) = -psi(x)
    X2 = -X
    Xtot = np.append(X2[::-1], X)
    psi_tot = np.append(-psi[::-1], psi)#minus sign is included on psi for x>0
since it is an odd function here


    plt.plot(Xtot, psi_tot**2, label='n = {0:1}, E = {1:4f}'.format(n, E))
    plt.title("Probability distribution for Odd State of Infinite Potential We
ll problem")
    plt.xlabel("x")
    plt.ylabel("$|\psi (x)|^2$")
    plt.legend(loc='upper right')

infinite_well_odd_plot(1,0.005,0.377,0,1,n=2)
plt.show()
infinite_well_odd_probs(1,0.005,0.377,0,1,n=2)
plt.show()

infinite_well_even_plot(1,0.005,0.8433,1,0,n=3)
plt.show()
infinite_well_even_probs(1,0.005,0.8433,1,0,n=3)
plt.show()
```

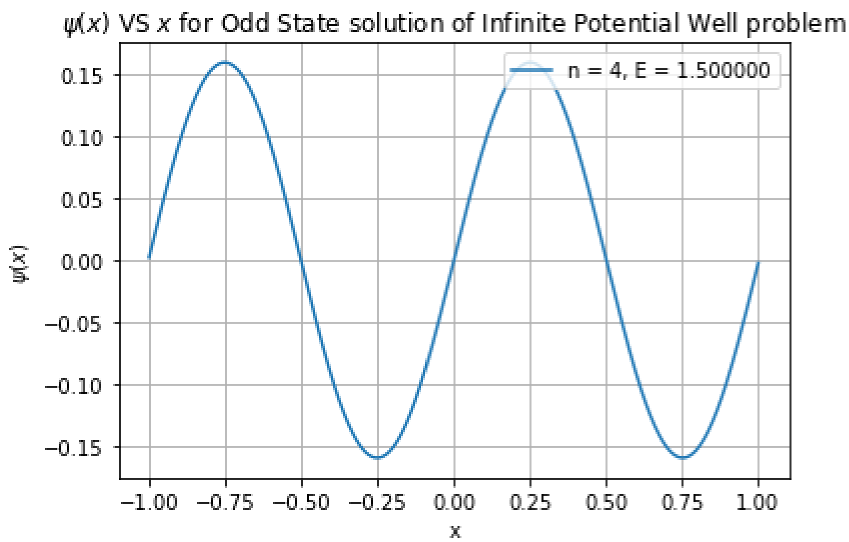
```
infinite_well_odd_plot(1,0.005,1.5,0,1,n=4)
plt.show()
infinite_well_odd_probs(1,0.005,1.5,0,1,n=4)
plt.show()



infinite_well_even_plot(1,0.005,0.0937,1,0,n=1)
infinite_well_odd_plot(1,0.005,0.377,0,1,n=2)
infinite_well_even_plot(1,0.005,0.8433,1,0,n=3)
infinite_well_odd_plot(1,0.005,1.5,0,1,n=4)
plt.title("$\psi (x)$ VS $x$ for n=1..4 Infinite Potential Well problem")
plt.show()

infinite_well_even_probs(1,0.005,0.0937,1,0,n=1)
infinite_well_odd_probs(1,0.005,0.377,0,1,n=2)
infinite_well_even_probs(1,0.005,0.8433,1,0,n=3)
infinite_well_odd_probs(1,0.005,1.5,0,1,n=4)
plt.title("Probability distribution for n=1..4 Infinite Potential Well problem")
plt.show()
```

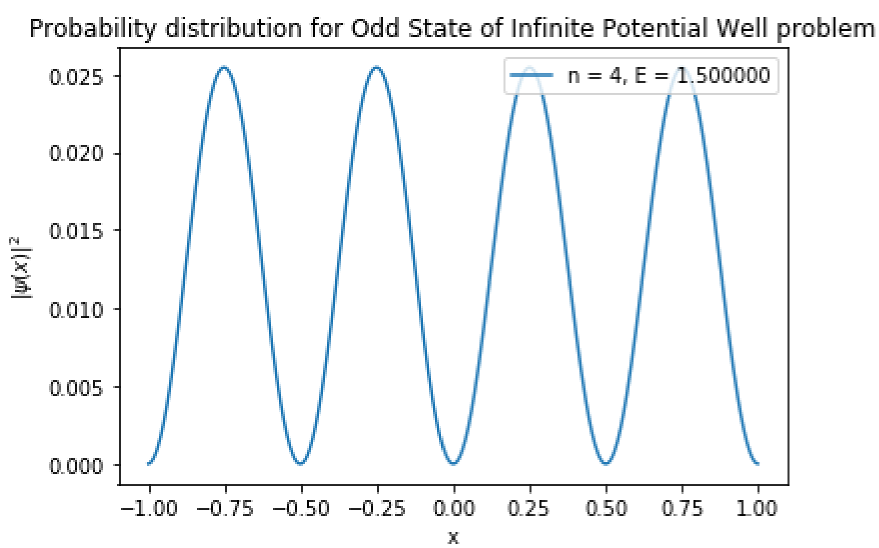

$\psi(x)$ VS x for Odd State solution of Infinite Potential Well problem
n = 2, E = 0.377000
$\psi(x)$
x

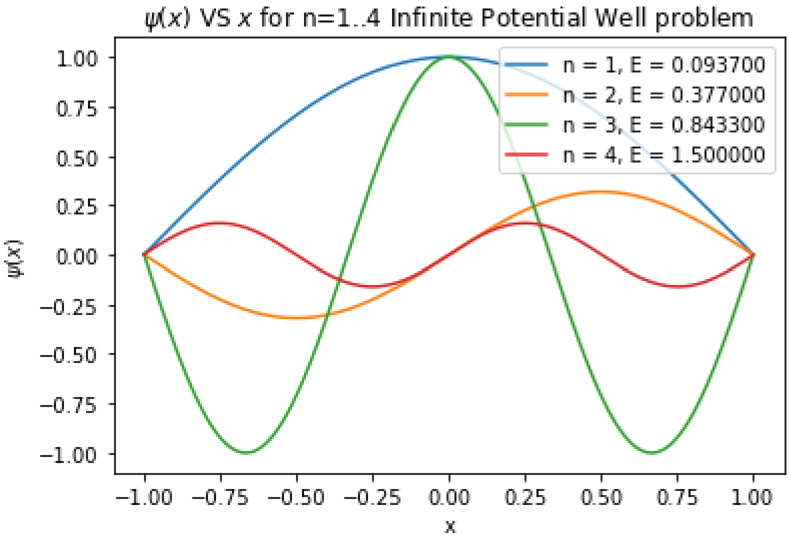

Probability distribution for Odd State of Infinite Potential Well problem
n = 2, E = 0.377000
$|\psi(x)|^2$
x

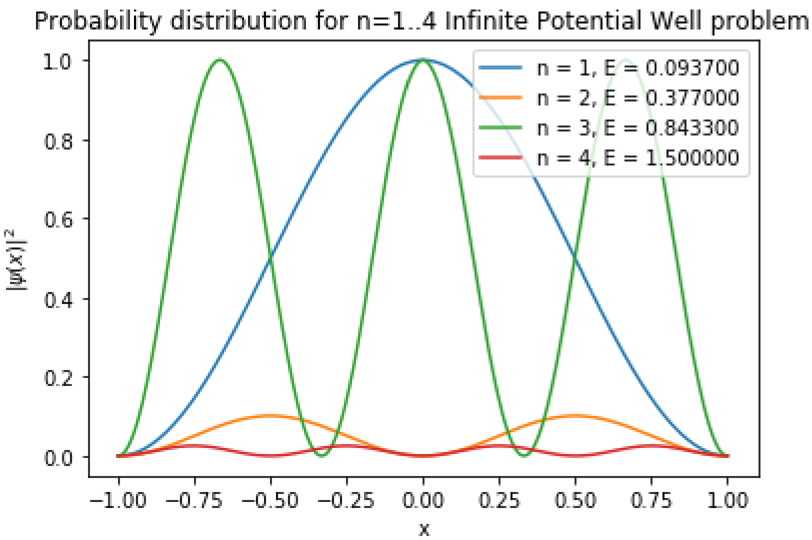

$\psi(x)$ VS x for Even State solution of Infinite Potential Well problem
n = 3, E = 0.843300
$\psi(x)$
x


Probability distribution for Even State of Infinite Potential Well problem
n = 3, E = 0.843300
$|\psi(x)|^2$
x


$\psi(x)$ VS x for Odd State solution of Infinite Potential Well problem
n = 4, E = 1.500000
$\psi(x)$
x


Probability distribution for Odd State of Infinite Potential Well problem
n = 4, E = 1.500000
$|\psi(x)|^2$
x





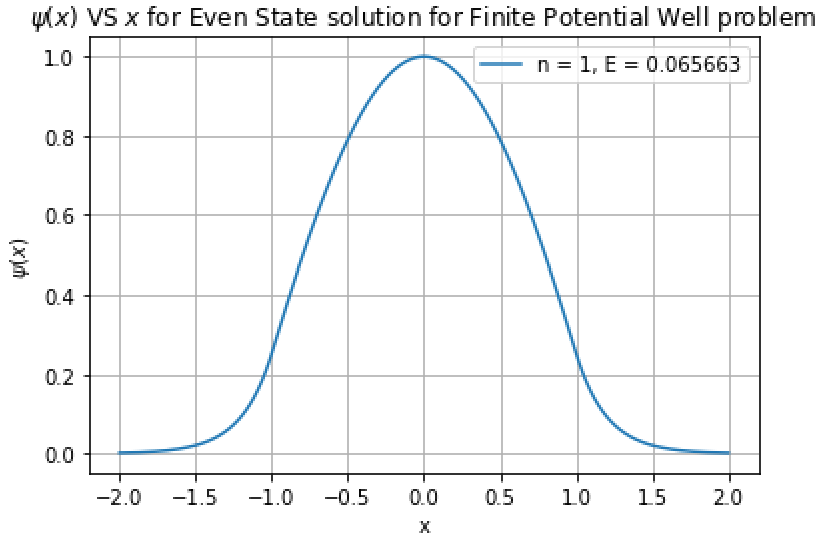
Here we find the Energy E of the electron is related to the quantum number n in the following way

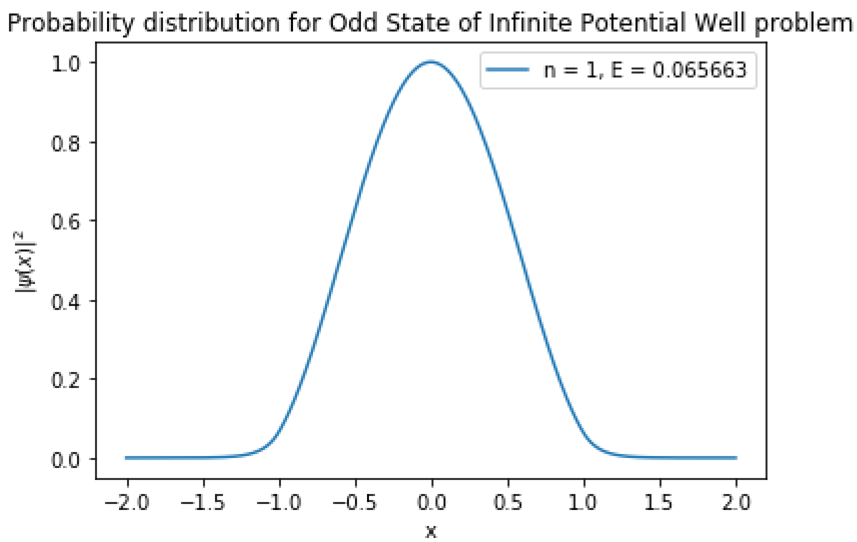
$$E = E_0 n^2$$

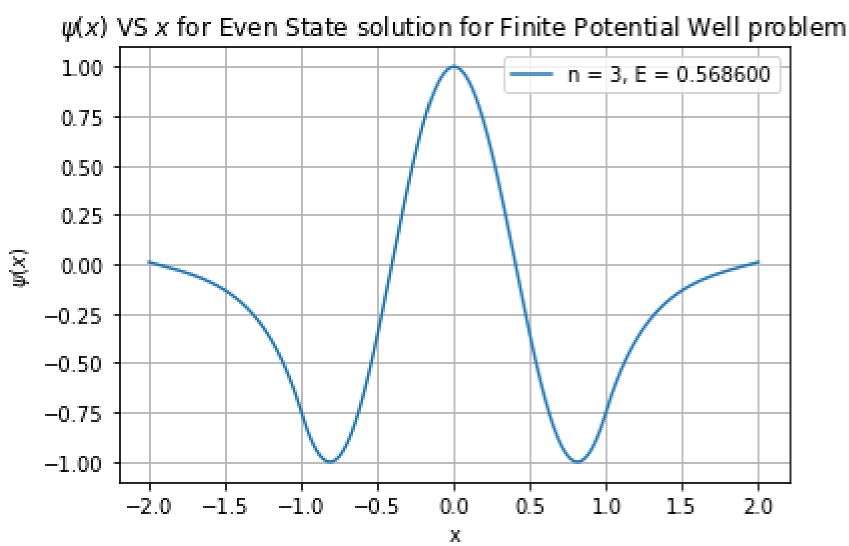
Where E_0 is the Ground State Energy of the Electron for n = 1

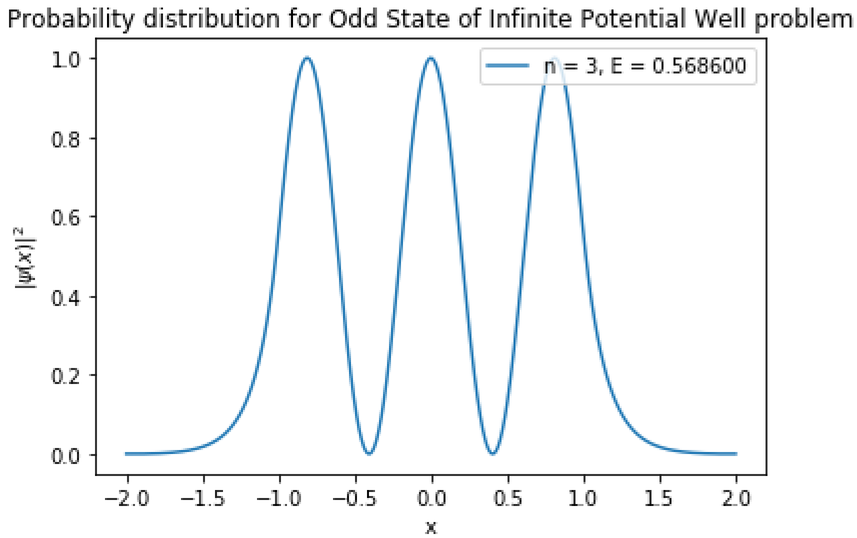
In [168]:
```
#task 3
#psi(x)'' = -26.2(E-V)psi(x)
# V = 1.0 eV for |x|> a

#use shooting method again here to find E such that psi is roughly equal to 0
 outside of the well (psi > a, a = 1)

def Euler_method2(a,dx,E,psi_0,dpsi_0):
    #a = x_max, dx = step size, E = Energy eigenvalue, psi_0 = initial value o
f psi, dpsi_0 =  initial change in psu
    N= int((a+1)/dx)# Number of steps -> beyond a in this case
    X = np.zeros(N+1) #x values
    psi = np.zeros(N+1) #psi values
    dpsi = np.zeros(N+1) #d(psi)/dx values

    X[0] = 0
    psi[0] = psi_0 #set initial value of psi (psi(0))
    dpsi[0] = dpsi_0 #set initial value of d(psi)/dx (psi'(0))


    #if statement determines whether V is 1ev or 0eV based on whether position
is outside well or
    #inside well respectively.

    for x in range(N):
        if X[x] > a:
            d2psi = -26.2*(E-1)*psi[x] #using Euler's Method to identify psi v
alues for each time step
            dpsi[x+1] = dpsi[x] + d2psi*dx
            psi[x+1] = psi[x] + dpsi[x+1]*dx
            X[x+1] = X[x] + dx

        elif X[x] < a:
            d2psi = -26.2*E*psi[x] #using Euler's Method to identify psi value
s for each time step
            dpsi[x+1] = dpsi[x] + d2psi*dx
            psi[x+1] = psi[x] + dpsi[x+1]*dx
            X[x+1] = X[x] + dx


    return psi, X


def finite_well_even_plot(a,dx,E,psi_0,dpsi_0,n): #plots even psi soln against
x

    #function uses euler's method

    psi = Euler_method2(a,dx,E,psi_0,dpsi_0)[0]
    X = Euler_method2(a,dx,E,psi_0,dpsi_0)[1]

    #make plot symmetric by flipping the X and y arrays (X and Psi) and then a
ppending the orignal arrays
    X2 = -X
    Xtot = np.append(X2[::-1],X)
    psi_tot = np.append(psi[::-1],psi)
```

```
    plt.plot(Xtot,psi_tot, label='n = {0:1}, E = {1:4f}'.format(n, E))
    plt.title("$\psi (x)$ VS $x$ for Even State solution for Finite Potential
 Well problem")
    plt.xlabel("x")
    plt.ylabel("$\psi (x)$")
    plt.legend(loc='upper right')
    plt.grid()

def finite_well_even_probs(a,dx,E,psi_0,dpsi_0, n):

    #function uses euler's method

    psi = Euler_method2(a,dx,E,psi_0,dpsi_0)[0]
    X = Euler_method2(a,dx,E,psi_0,dpsi_0)[1]

    #plot negative region, note psi(-x) = -psi(x)
    X2 = -X
    Xtot = np.append(X2[::-1], X)
    psi_tot = np.append(psi[::-1], psi)#minus sign is included on psi for x>0
 since it is an odd function here


    plt.plot(Xtot, psi_tot**2, label='n = {0:1}, E = {1:4f}'.format(n, E))
    plt.title("Probability distribution for Odd State of Infinite Potential We
ll problem")
    plt.xlabel("x")
    plt.ylabel("$|\psi (x)|^2$")
    plt.legend(loc='upper right')



E_0 = 0.065663 #energy found from shooting method for ground state finite well
soln

finite_well_even_plot(1,0.005,E_0,1,0,n=1) #first even bound solution
plt.show()
finite_well_even_probs(1,0.005,E_0,1,0,n=1)
plt.show()

finite_well_even_plot(1,0.005,0.5686,1,0,n=3) #second even bound solution
plt.show()
finite_well_even_probs(1,0.005,0.5686,1,0,n=3)
plt.show()
```

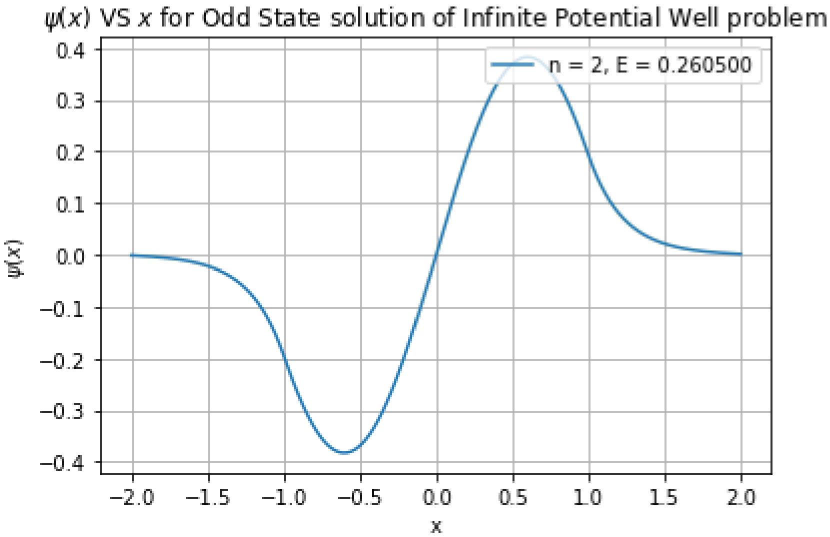

$\psi(x)$ VS x for Even State solution for Finite Potential Well problem
n = 1, E = 0.065663
$\psi(x)$
x

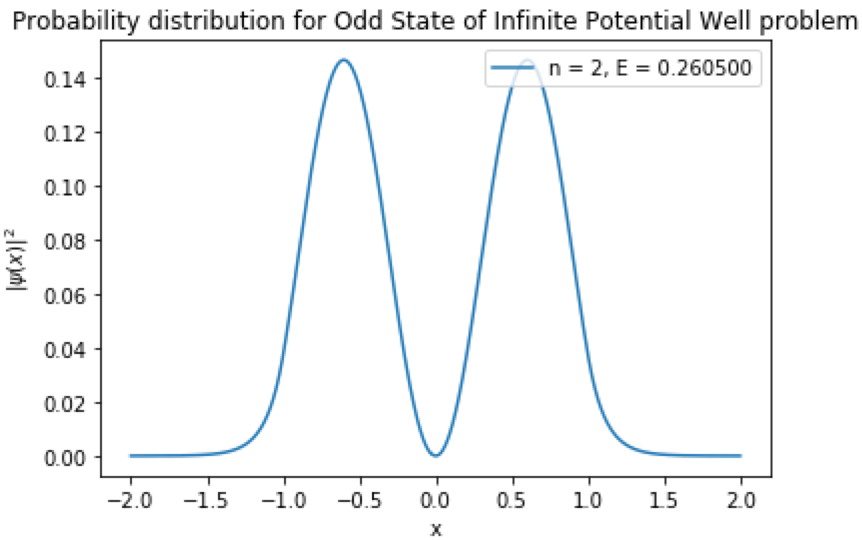

Probability distribution for Odd State of Infinite Potential Well problem
n = 1, E = 0.065663
$|\psi(x)|^2$
x

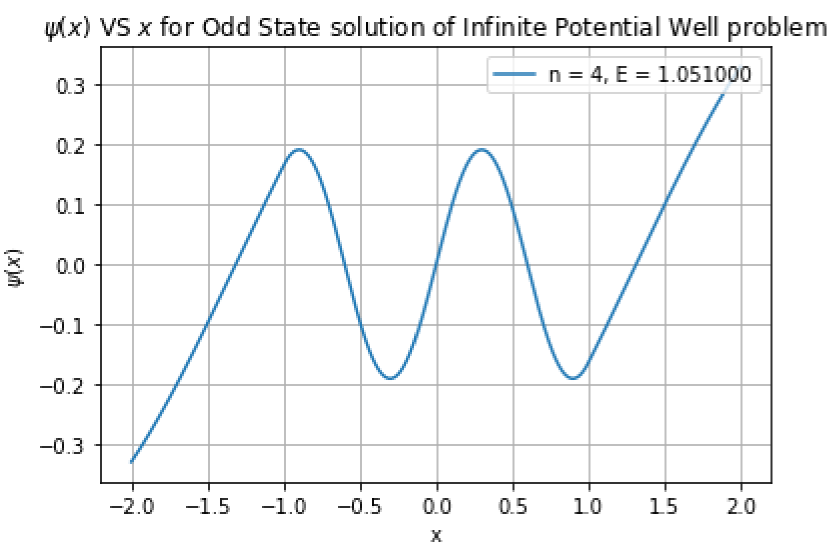

$\psi(x)$ VS x for Even State solution for Finite Potential Well problem
n = 3, E = 0.568600
$\psi(x)$
x

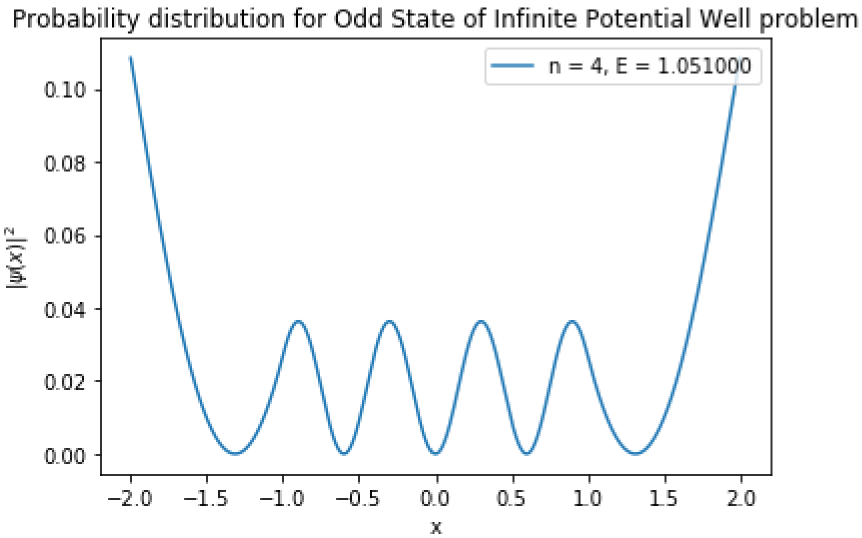

Probability distribution for Odd State of Infinite Potential Well problem
n = 3, E = 0.568600
$|\psi(x)|^2$
x
1.0
0.8
0.6
0.4
0.2
0.0
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0

In [169]:
```
def finite_well_odd_plot(a,dx,E,psi_0,dpsi_0, n):

    #function uses euler's method

    psi = Euler_method2(a,dx,E,psi_0,dpsi_0)[0]
    X = Euler_method2(a,dx,E,psi_0,dpsi_0)[1]

    #plot negative region, note psi(-x) = -psi(x)
    X2 = -X
    Xtot = np.append(X2[::-1], X)
    psi_tot = np.append(-psi[::-1], psi)#minus sign is included on psi for x>0
since it is an odd function here

    plt.plot(Xtot,psi_tot, label='n = {0:1}, E = {1:4f}'.format(n, E))
    plt.title("$\psi (x)$ VS $x$ for Odd State solution of Infinite Potential
 Well problem")
    plt.xlabel("x")
    plt.ylabel("$\psi (x)$")
    plt.grid()
    plt.legend(loc='upper right')


def finite_well_odd_probs(a,dx,E,psi_0,dpsi_0, n):

    #function uses euler's method

    psi = Euler_method2(a,dx,E,psi_0,dpsi_0)[0]
    X = Euler_method2(a,dx,E,psi_0,dpsi_0)[1]

    #plot negative region, note psi(-x) = -psi(x)
    X2 = -X
    Xtot = np.append(X2[::-1], X)
    psi_tot = np.append(-psi[::-1], psi)#minus sign is included on psi for x>0
since it is an odd function here


    plt.plot(Xtot, psi_tot**2, label='n = {0:1}, E = {1:4f}'.format(n, E))
    plt.title("Probability distribution for Odd State of Infinite Potential We
ll problem")
    plt.xlabel("x")
    plt.ylabel("$|\psi (x)|^2$")
    plt.legend(loc='upper right')


finite_well_odd_plot(1,0.005,0.2605,0,1,n=2)#first odd state
plt.show()
finite_well_odd_probs(1,0.005,0.2605,0,1,n=2)
plt.show()

finite_well_odd_plot(1,0.005,1.051,0,1,n=4)#second odd state
plt.show()
finite_well_odd_probs(1,0.005,1.051,0,1,n=4)
plt.show()
```


ψ(x) VS x for Odd State solution of Infinite Potential Well problem
n = 2, E = 0.260500
ψ(x)
x
0.4
0.3
0.2
0.1
0.0
−0.1
−0.2
−0.3
−0.4
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0


Probability distribution for Odd State of Infinite Potential Well problem
n = 2, E = 0.260500
|ψ(x)|²
x
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0


ψ(x) VS x for Odd State solution of Infinite Potential Well problem
n = 4, E = 1.051000
ψ(x)
x
0.3
0.2
0.1
0.0
−0.1
−0.2
−0.3
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0



There are 3 bound state solutions to the finite well of length a = 1 and V = 1eV for |x| > a. This is because, we calculate the energy of the 4th state to be approximately 1.051 eV, which exceeds the energy needed to pass through the barrier so particles can escape the well. The free particle then has a wavefunction that oscillates sinusoidally.

There is a difference between the energy of the even ground state for the infinte potential well and the even ground state solution for the finite potential well. For the infinite potential well I found the energy was 0.0937 eV whereas for the finite potential well I found the energy was 0.065663 eV for the ground state.

In [ ]: