In [3]:

```
import matplotlib.pyplot as plt
import numpy as np
```

In [4]:

```
h=1
m=1
```

In [5]:

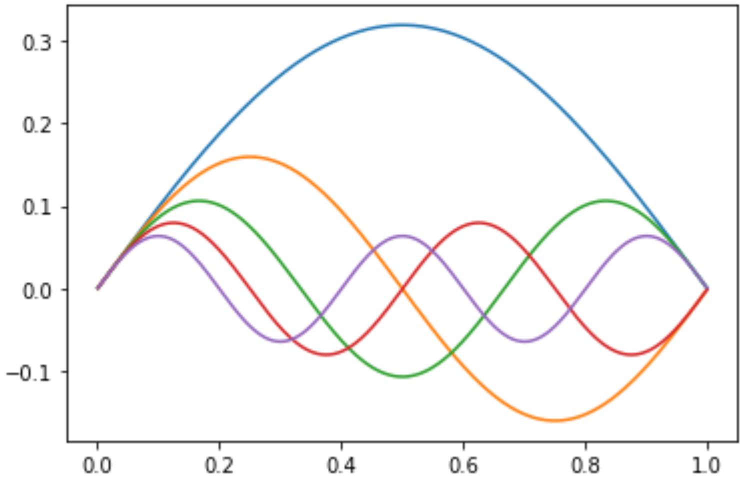
```
a=1
V=0
```

In [6]:

```
psi=1
dpsi=1
```

In [9]:

```
counter=1
while counter<=nmax:
    plt.plot(xlist,eigenfunctions[counter-1][0])
    counter=counter+1
```



In [8]:

```
eigenfunctions=[]
eigenfunctionsxlist=[]
eigenenergies=[]
E=0
dE=0.01
dx=a*0.001
counter=1

nmax=5

psi=1
dpsi=1

while counter<=nmax:
    psi=1

    while abs(psi)>0.001:
        E=E+dE
        psi=0
        x=0
        dpsi=1
        xlist=[]
        psilist=[]

        while x<=a:
            ddpsi=2*m/h**2*(V-E)*psi
            dpsi=dpsi+ddpsi*dx
            psi=psi+dpsi*dx
            x=x+dx
            xlist.append(x)
            psilist.append(psi)

    eigenfunctions.append([psilist])
    eigenfunctionsxlist.append([xlist])
    eigenenergies.append([E])
    counter=counter+1
    E=E+1
```

In [32]:

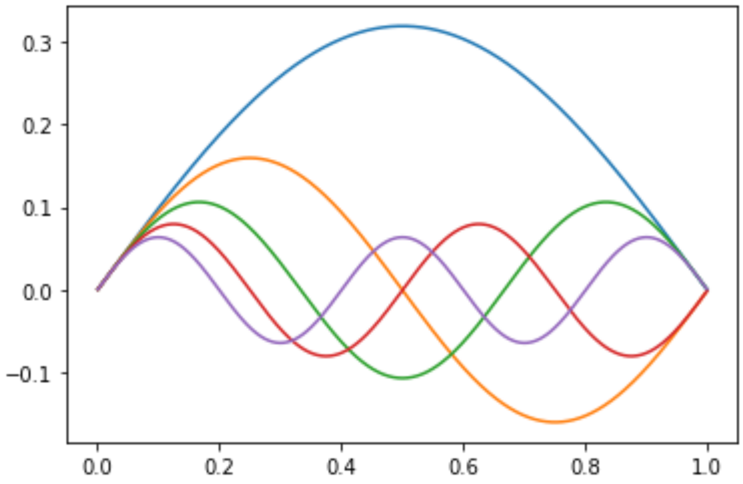
```
eigenenergies
```

Out[32]:

```
[[4.929999999999939],
 [19.70000000000003],
 [44.329999999999615],
 [78.80000000000346],
 [123.13000000002563]]
```

In [9]:

```
counter=1
while counter<=nmax:
    plt.plot(xlist,eigenfunctions[counter-1][0])
    counter=counter+1
```



In [ ]:

```
#Quantum- Physics-Shooting method to find Eigen values, executed by
Bhadale IT
```

In [12]:

```
np.dot(eigenfunctions[2][0], eigenfunctions[2][0])*dx
```

Out[12]:

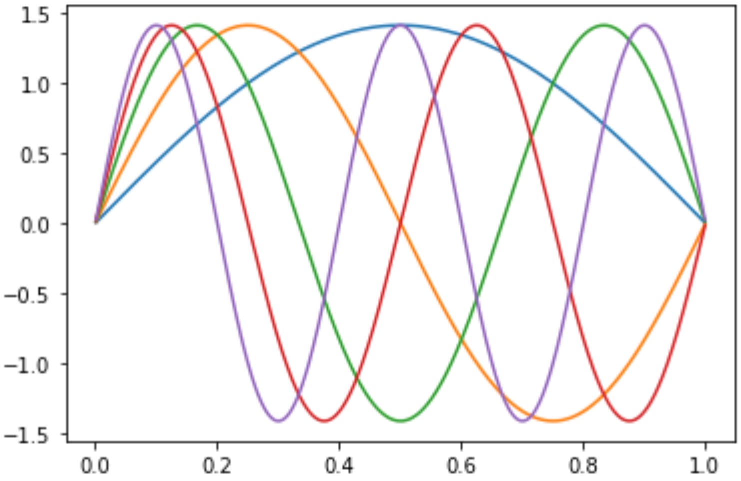
```
0.0056449170415890015
```

In [19]:

```
counter=1
while counter<=nmax:
    norm=np.dot(eigenfunctions[counter-1][0], eigenfunctions[counter
-1][0])*dx
    eigenfunctions[counter-1][0][:] = [x / np.sqrt(norm) for x in ei
genfunctions[counter-1][0]]
    counter=counter+1
```

In [20]:

```
counter=1
while counter<=nmax:
    plt.plot(xlist,eigenfunctions[counter-1][0])
    counter=counter+1
```



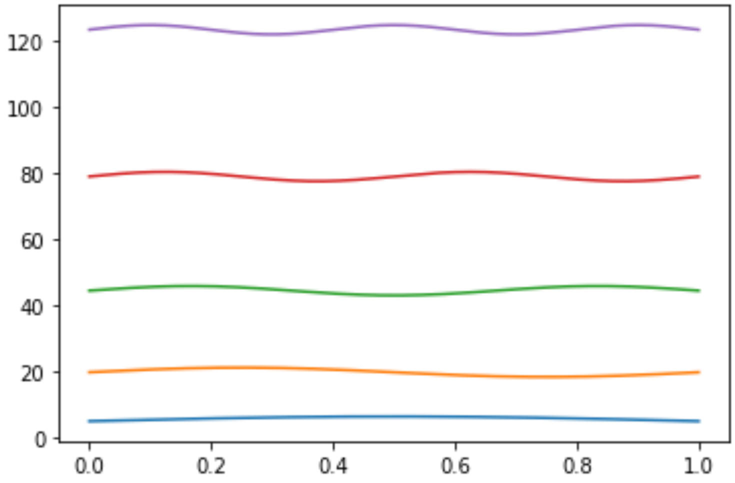
In [ ]:

In [21]:

```
counter=1
while counter<=nmax:
    eigenfunctions[counter-1][0][:] = [x +eigenenergies[counter-1] for x in eigenfunctions[counter-1][0]]
    counter=counter+1
```

In [22]:

```
counter=1
while counter<=nmax:
    plt.plot(xlist,eigenfunctions[counter-1][0])
    counter=counter+1
```



```
In [23]: eigenenergies
```

```
Out[23]: [[4.929999999999939],
          [19.7000000000003],
          [44.329999999999615],
          [78.80000000000346],
          [123.13000000002563]]
```

```
In [ ]:
```