In [1]:

```
import numpy as np
from numpy.linalg import norm
from mpl_toolkits.mplot3d import Axes3D
import matplotlib.pyplot as plt
```

In [2]:

```
# BCC Crystal
a_conv = 1
a1 = a_conv / 2 * np.array([-1,  1,  1]);
a2 = a_conv / 2 * np.array([ 1, -1,  1]);
a3 = a_conv / 2 * np.array([ 1,  1, -1]);
```

In [3]:

```
# Shortest lattice vectors
for a in range(-1,2):
    for b in range(-1,2):
        for c in range(-1,2):

            if a == 0 and b == 0 and c == 0:
                continue

            vector = a*a1 + b*a2 + c*a3
            norma = norm(vector)
            if norma < 0.87:
                print('({}{}{}) -> norm = {:.2f}'.format(a,b,c,norma))
```

```
(-1-1-1) -> norm = 0.87
(-100) -> norm = 0.87
(0-10) -> norm = 0.87
(00-1) -> norm = 0.87
(001) -> norm = 0.87
(010) -> norm = 0.87
(100) -> norm = 0.87
(111) -> norm = 0.87
```

In [4]:

```
# FCC Crystal
a  = 1
a1 = a/2*np.array([0, 1, 1]);
a2 = a/2*np.array([1, 0, 1]);
a3 = a/2*np.array([1, 1, 0]);
```

In [5]:

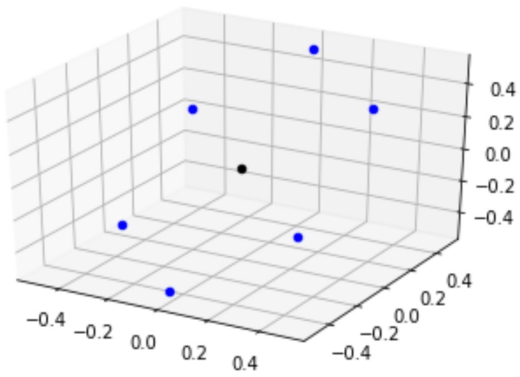
```
# Shortest lattice vectors
for a in range(-1,2):
    for b in range(-1,2):
        for c in range(-1,2):

            if a == 0 and b == 0 and c == 0:
                continue

            vector = a*a1 + b*a2 + c*a3
            norma = norm(vector)
            if norma < 0.71:
                print('({}{}{}) -> norm = {:.2f}'.format(a,b,c,norma))
```

```
(-100) -> norm = 0.71
(-101) -> norm = 0.71
(-110) -> norm = 0.71
(0-10) -> norm = 0.71
(0-11) -> norm = 0.71
(00-1) -> norm = 0.71
(001) -> norm = 0.71
(01-1) -> norm = 0.71
(010) -> norm = 0.71
(1-10) -> norm = 0.71
(10-1) -> norm = 0.71
(100) -> norm = 0.71
```

In [6]:

```
# Plot 2d cross section hexagonal lattice
fig = plt.figure()
ax = fig.gca(projection='3d')

def plotter(a,b,c,color='b'):
    vector = a*a1 + b*a2 + c*a3
    ax.scatter(vector[0], vector[1], vector[2], c=color)

def distance(c1,c2,c3,d1,d2,d3):
    v1 = c1*a1 + c2*a2 + c3*a3
    v2 = d1*a1 + d2*a2 + d3*a3
    print(norm(v1 - v2))

vector0 = [ 0, 0, 0]
vector1 = [ 1, 0, 0]
vector2 = [ 0, 1, 0]
vector3 = [-1, 1, 0]
vector4 = [-1, 0, 0]
vector5 = [ 0,-1, 0]
vector6 = [ 1,-1, 0]

plotter(*vector0,'k')
plotter(*vector1)
plotter(*vector2)
plotter(*vector3)
plotter(*vector4)
plotter(*vector5)
plotter(*vector6)

distance(*vector1, *vector2)
distance(*vector2, *vector3)
distance(*vector3, *vector4)
distance(*vector4, *vector5)
distance(*vector5, *vector6)
distance(*vector6, *vector1)

plt.show()
```

```
0.7071067811865476
0.7071067811865476
0.7071067811865476
0.7071067811865476
0.7071067811865476
0.7071067811865476
```



In [ ]: