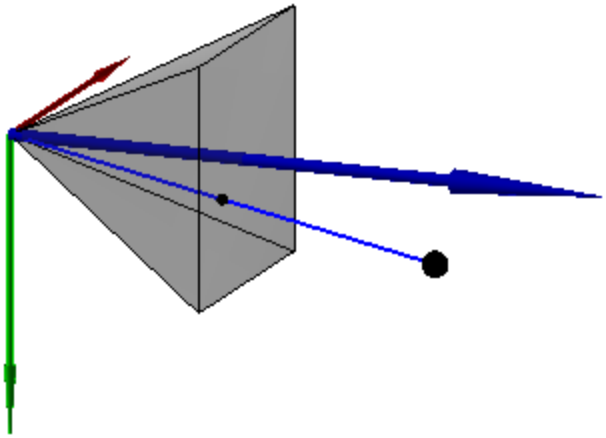
Image size: 512 x 512 pixels

Skew factor: 0

Principal point: (256, 256)

Focal distances: (200,200)



Question 1

What are the image coordinates $X = (x, y)^T$ of the 3-D point $w = (-50, 20, 200)^T$ projected under the camera?

In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d.art3d import Poly3DCollection
```

In [2]:
```
#skew factor
gamma = 0

#focal distances
phi = np.array([200,200])

#Principal point
delta = np.array([256,256])



w = np.array([-50,20,200])

x = (phi[0]*w[0]+gamma*w[1])/w[2] + delta[0]
y = (phi[1]*w[1])/w[2] + delta[1]

print("Coordinates:\nx:", x ,"\ny: ", y)
```

```
Coordinates:
x: 206.0
y:  276.0
```

$$C = \begin{bmatrix} 0 & 50 & 50 & 0 & 0 & 50 & 50 & 0 \\ 0 & 0 & 50 & 50 & 0 & 0 & 50 & 50 \\ 200 & 200 & 200 & 200 & 250 & 250 & 250 & 250 \end{bmatrix}$$

Question 2

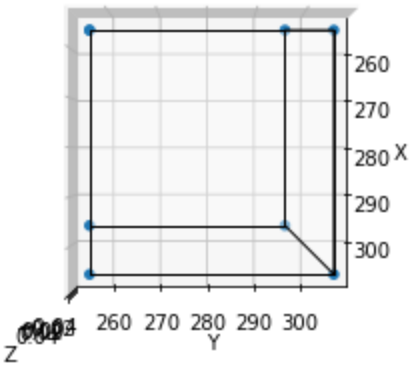
What are the coordinates of the vertices of the cube as represented by the matrix?

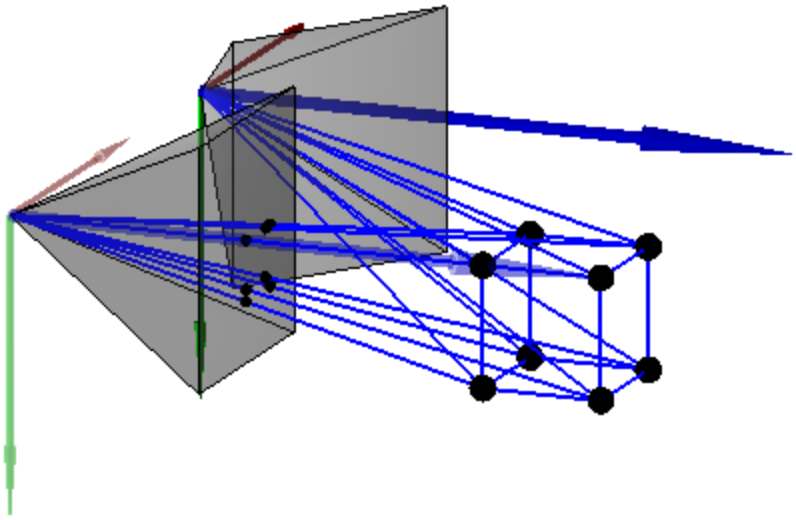
In [3]:

```
C = np.array([[0, 50, 50, 0, 0, 50, 50, 0],
              [0, 0, 50, 50, 0, 0, 50, 50],
              [200, 200, 200, 200, 250, 250, 250, 250]])

xvector = (phi[0]*C[0]+0*C[1])/C[2] + delta[0]

yvector = (phi[1]*C[1])/C[2] + delta[1]

print("Coordinates:\nx:", xvector ,"\ny: ", yvector)

box = np.array([xvector, yvector, np.zeros(8)]).T

# box.(addedbox)

# box = np.vstack((box, addedbox))

print("\n",box)

fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')


ax.scatter3D(box[:, 0], box[: ,1], box[:, 2])
ax.set_xlabel('X')
ax.set_ylabel('Y')
ax.set_zlabel('Z')

# set verts connectors
verts = [[box[0],box[1],box[2],box[3]], [box[4],box[5],box[6],box[7]],
[box[0],box[1],box[5],box[4]], [box[3],box[2],box[6],box[7]]]
         #[box[1],box[2],box[6],box[5]], [box[1],box[2],box[5],box[7]]]


# plot sides
ax.add_collection3d(Poly3DCollection(verts, facecolors='w', linewidths=1, edgecolors='k',
alpha=.25))
ax.view_init(90,0)
```

```
# ax.add_collection()
plt.show()
```

```
Coordinates:
x: [256. 306. 306. 256. 256. 296. 296. 256.]
y:  [256. 256. 306. 306. 256. 256. 296. 296.]

 [[256. 256.   0.]
 [306. 256.   0.]
 [306. 306.   0.]
 [256. 306.   0.]
 [256. 256.   0.]
 [296. 256.   0.]
 [296. 296.   0.]
 [256. 296.   0.]]
```





Question 3

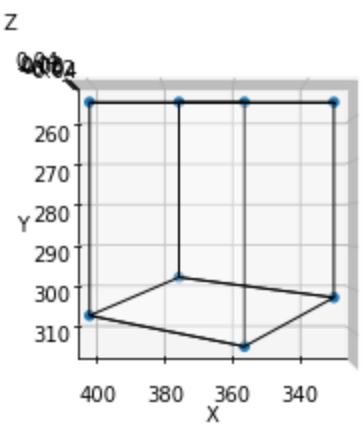
Considering the same camera given in the previous questions and the cube vertices from Question 2, calculate the image coordinates of the vertices given that the camera now rotated by 35 degrees about the v-axis and translated along the u-axis by the following vector $\tau = (200, 0, 0)^T$.

In [4]:

```
#given value
tau = np.array([200,0,0])
```

```
omegav = np.array([[np.cos(30 * np.pi/180), 0, -np.sin(30 * np.pi/180)],
                   [0, 1, 0],
                   [np.sin(30 * np.pi/180), 0 , np.cos(30 * np.pi/180)]])

#transpose the box
box = C.T

xvector = np.empty(len(C[0]))

yvector = np.empty(len(C[1]))


#calculate every x and y point using the equation in the book, degree is 30
for i in range(len(C[0])):

    xvector[i]  = (phi[0] * (np.dot(box[i], omegav[0]) + tau[0]) + gamma*(np.dot(box[i],
omegav[0]) + tau[1]))/(np.dot(box[2], omegav[2]) + tau[2]) + delta[0]

    yvector[i] = (phi[1] * (np.dot(box[i], omegav[1]) + tau[1]) / (np.dot(box[i],
omegav[2]) + tau[2])) + delta[1]


print("Coordinates: \nX:", xvector.round(), "\nY: ", yvector.round())


box = np.array([xvector, yvector, np.zeros(8)]).T

fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')


ax.scatter3D(box[:, 0], box[: ,1], box[:, 2])
ax.set_xlabel('X')
ax.set_ylabel('Y')
ax.set_zlabel('Z')


verts = [[box[0],box[1],box[2],box[3]], [box[4],box[5],box[6],box[7]],
[box[0],box[1],box[5],box[4]], [box[3],box[2],box[6],box[7]]]
         #[box[1],box[2],box[6],box[5]], [box[1],box[2],box[5],box[7]]]

# plot sides
ax.add_collection3d(Poly3DCollection(verts, facecolors='w', linewidths=1, edgecolors='k',
alpha=.25))
ax.view_init(90,90)
```

```
Coordinates:
X: [357. 401. 401. 357. 332. 375. 375. 332.]
Y:  [256. 256. 306. 314. 256. 256. 297. 302.]
```



In [ ]: