**3D Plotting**

This part of code generates a 3D plot using the matplotlib library. It creates a figure and a 3D axes object, and plots the visited nodes (visited (red), visited2 (green) and visited3 (yellow)) and obstacles (obstacles (blue)) on the plot.

The visited nodes are plotted as red, green and yellow lines, while the obstacles are plotted as blue points. The plot function is used to plot the lines, and the scatter3D function is used to plot the points.

The axes labels are set using the set\_xlabel, set\_ylabel, and set\_zlabel functions to label each axes, and the plot is displayed using the show function.

The code used to visualize the path taken by an algorithm as it navigates through a 3D space, with the obstacles represented as blue points on the plot.

**Source code**

# Create a figure and a 3D axes  
fig = plt.figure()  
ax = fig.add\_subplot(111, projection="3d")  
  
# Plot the visited nodes  
ax.plot([x[0] for x in visited], [x[1] for x in visited], [x[2] for x in visited], "ro-")  
ax.plot([x[0] for x in visited2], [x[1] for x in visited2], [x[2] for x in visited2], "go-")  
ax.plot([x[0] for x in visited3], [x[1] for x in visited3], [x[2] for x in visited3], "yo-")  
  
#Plot obstacles  
obstacles\_coordinates = []  
  
# Convert to coordinates as an array for the plotting  
for obst in obstacles:  
 for x in range(obst[0], obst[1] + 1):  
 for y in range(obst[2], obst[3] + 1):  
 for z in range(obst[4], obst[5] + 1):  
 obstacles\_coordinates.append([x, y, z])  
  
ax.scatter3D([x[0] for x in obstacles\_coordinates], [x[1] for x in obstacles\_coordinates], [x[2] for x in obstacles\_coordinates], "b")  
  
# Set the axes labels  
ax.set\_xlabel("X")  
ax.set\_ylabel("Y")  
ax.set\_zlabel("Z")  
  
# Show the plot  
plt.show()

**Pseudo code**

# Create a figure and a 3D axes

fig = create a figure

ax = add a 3D axes to fig

# Plot the visited nodes

plot visited nodes on ax as red lines

plot visited2 nodes on ax as green lines

plot visited3 nodes on ax as yellow lines

# Plot the obstacles

obstacles\_coordinates = []

for each obst in obstacles:

for x in range from obst[0] to obst[1]:

for y in range from obst[2] to obst[3]:

for z in range from obst[4] to obst[5]:

add [x, y, z] to obstacles\_coordinates

scatter obstacles\_coordinates on ax as blue points

# Set the axes labels

set x-axis label of ax to "X"

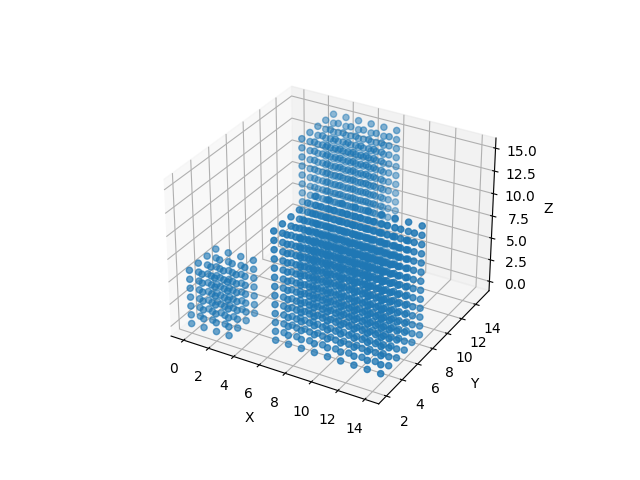
set y-axis label of ax to "Y"

set z-axis label of ax to "Z"

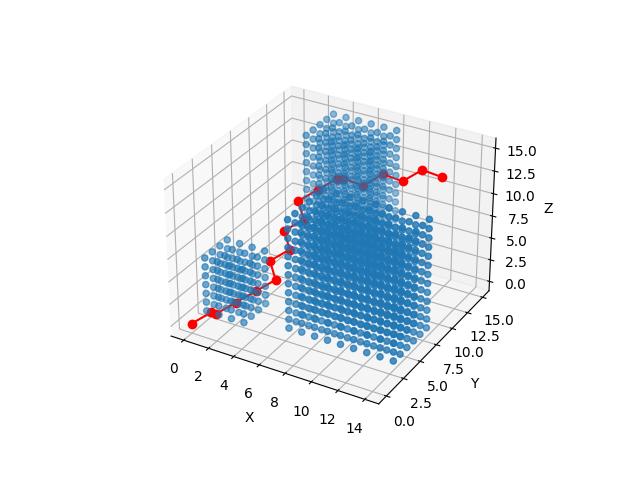
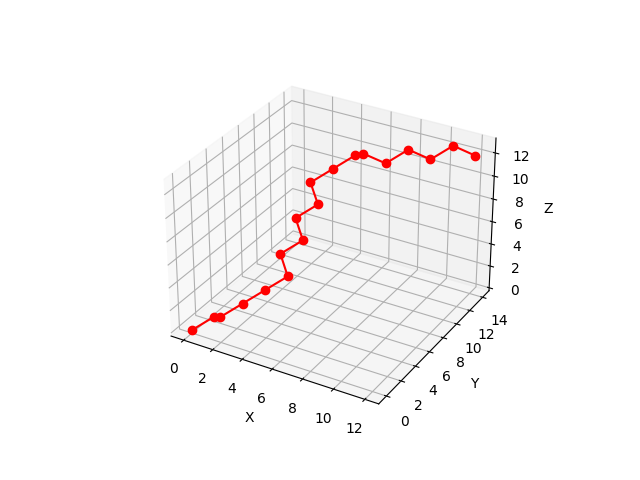
# Show the plot

show fig

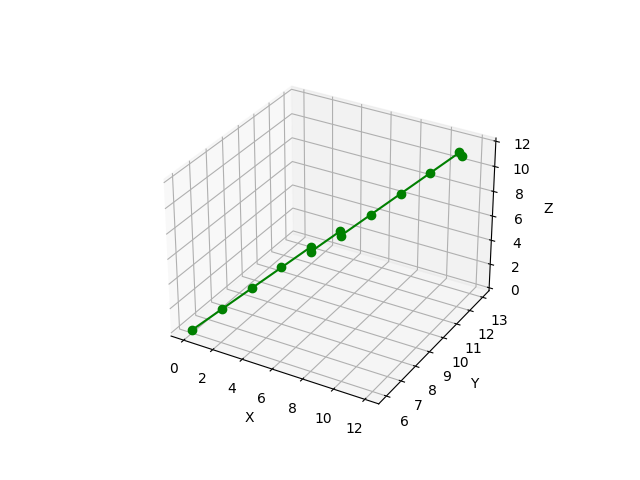
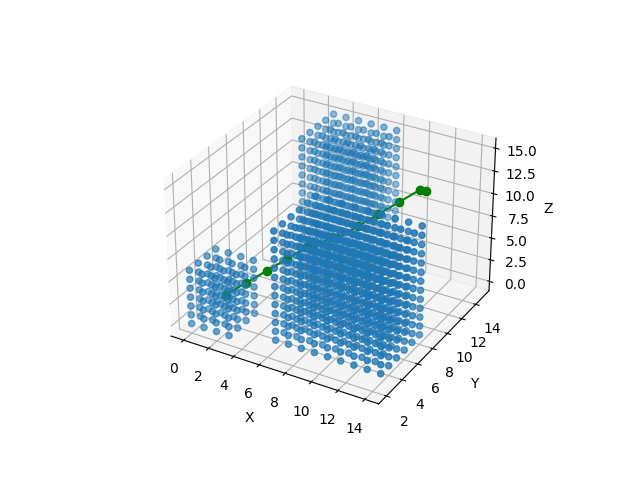
Plot the obstacles:



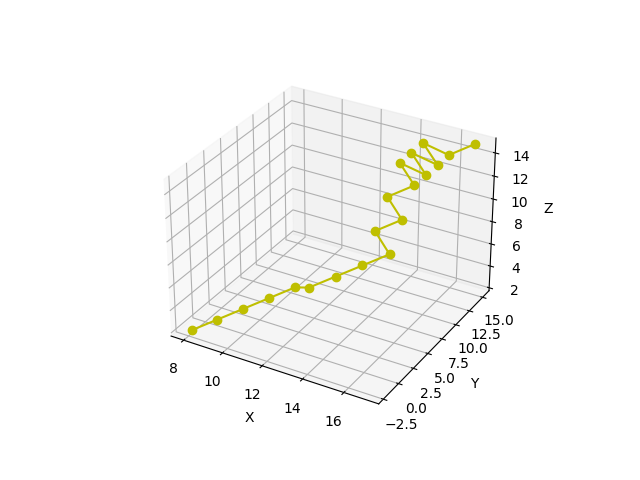
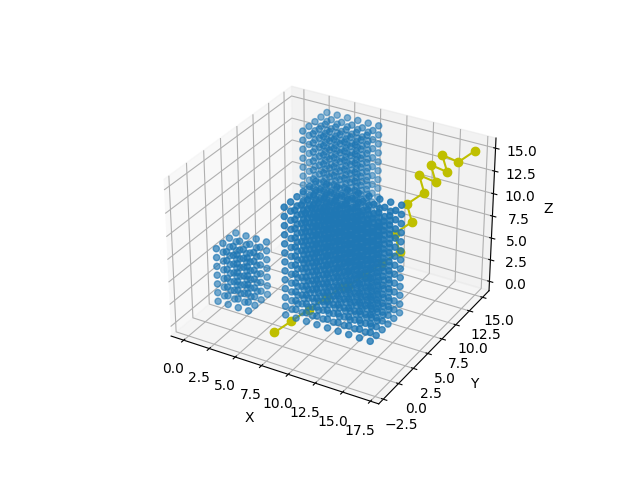
Plot the path 1:

Plot the path 2:



Plot the path 3:



Plot with the obstacles and the paths:

