# **EXPERIMENT - 2**

**Implement the scan conversion of a polygon and use flood-fill algorithms to fill areas in a graphics window.**

import numpy as np

import matplotlib.pyplot as plt

from collections import deque

polygon = np.array([[2, 6, 6, 2, 2],

[2, 2, 5, 5, 2]])

plt.plot(polygon[0], polygon[1], 'b-', linewidth=2)

plt.fill(polygon[0], polygon[1], 'lightgray', label="Scan Converted Fill") # scan conversion (simple fill)

def flood\_fill(image, start, fill\_color, boundary\_color):

rows, cols = image.shape

x, y = start

q = deque()

q.append((x, y))

while q:

x, y = q.popleft()

if 0 <= x < rows and 0 <= y < cols:

if image[x, y] != boundary\_color and image[x, y] != fill\_color:

image[x, y] = fill\_color

q.extend([(x+1, y), (x-1, y), (x, y+1), (x, y-1)])

grid = np.zeros((10, 10), dtype=int)

grid[2:6, 2] = 1

grid[2:6, 6] = 1

grid[2, 2:7] = 1

grid[6, 2:7] = 1

flood\_fill(grid, (3, 3), 2, 1)

plt.figure()

plt.imshow(grid, cmap="viridis")

plt.title("Flood Fill Result (Grid View)")

plt.colorbar()

plt.show()



