****

**LAB MID**

**Artificial Intelligence**

**Submitted To:**

MAM YASMEEN JANA

**Submitted By:**

FAHAD RASOOL

FA20-BSE-008

**(B)**

**COMSATS UNIVSERSITY ISLAMABAD,**

**VEHARI CAMPUS**

**Date:** 18 April, 2023­

**PYTHON MAZE WORLD**

**from pyamaze import maze,agent,COLOR,textLabel**

**def BFS(m):**

**start=(m.rows,m.cols)**

**frontier=[start]**

**explored=[start]**

**bfsPath={}**

**while len(frontier)>0:**

**currCell=frontier.pop(0)**

**if currCell==(1,1):**

**break**

**for d in 'ESNW':**

**if m.maze\_map[currCell][d]==True:**

**if d=='E':**

**childCell=(currCell[0],currCell[1]+1)**

**elif d=='W':**

**childCell=(currCell[0],currCell[1]-1)**

**elif d=='N':**

**childCell=(currCell[0]-1,currCell[1])**

**elif d=='S':**

**childCell=(currCell[0]+1,currCell[1])**

**if childCell in explored:**

**continue**

**frontier.append(childCell)**

**explored.append(childCell)**

**bfsPath[childCell]=currCell**

**fwdPath={}**

**cell=(1,1)**

**while cell!=start:**

**fwdPath[bfsPath[cell]]=cell**

**cell=bfsPath[cell]**

**return fwdPath**

**if \_\_name\_\_=='\_\_main\_\_':**

**m=maze(5,7)**

**m.CreateMaze(loopPercent=40)**

**path=BFS(m)**

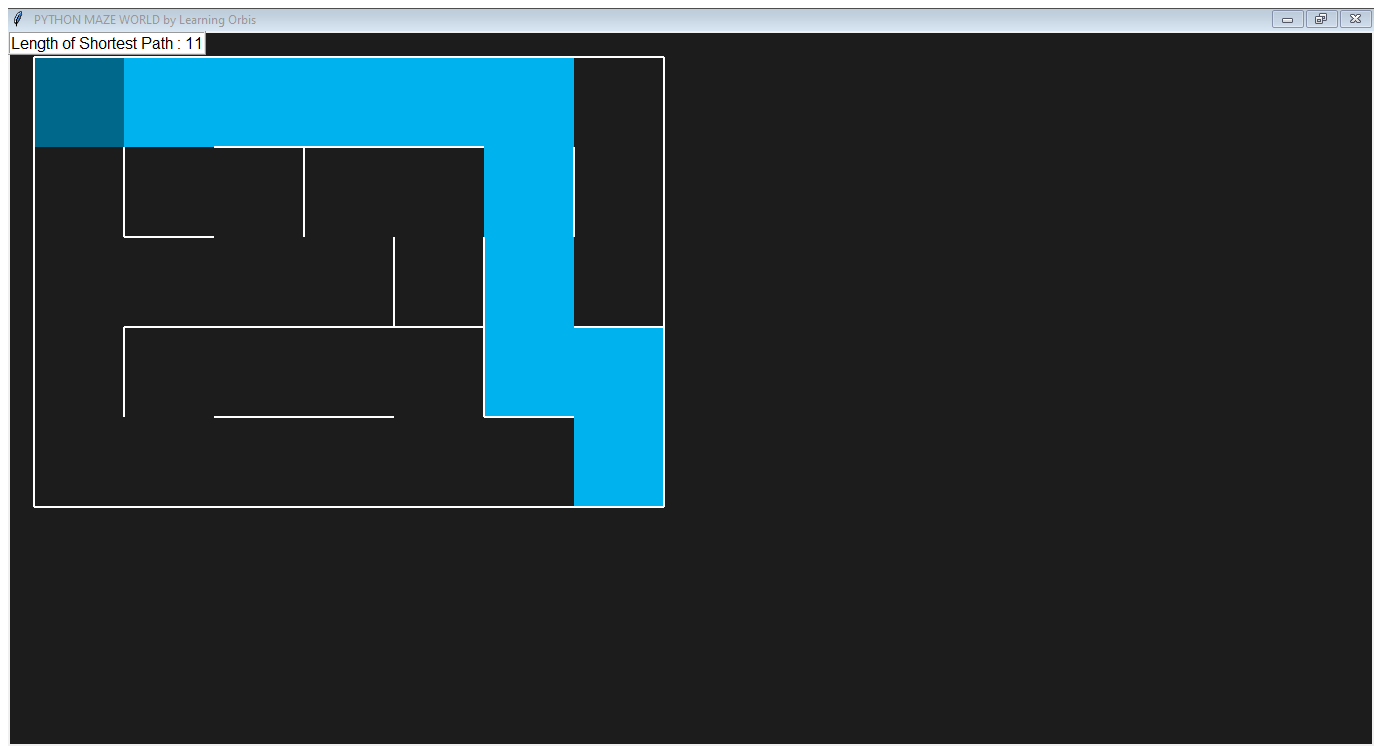
**a=agent(m,footprints=True,filled=True)**

**m.tracePath({a:path})**

**l=textLabel(m,'Length of Shortest Path',len(path)+1)**

**m.run()**

**Output:**

****

**Explain:**

This project uses a module called Pyamaze for drawing the maze board. It uses BFS algorithm to find shortest path between the start and goal node. It uses a two dimensional maze board in which some obstacles are also present. We have to avoid these obstacles and find a way to the goal node. There are 4 directions defined as ‘East, West, South, North’. Once the path is obtained, its time to make it colorfull. It is done by using Pyamaze module.

**Reference:**

https://www.youtube.com/watch?v=D14YK-0MtcQ