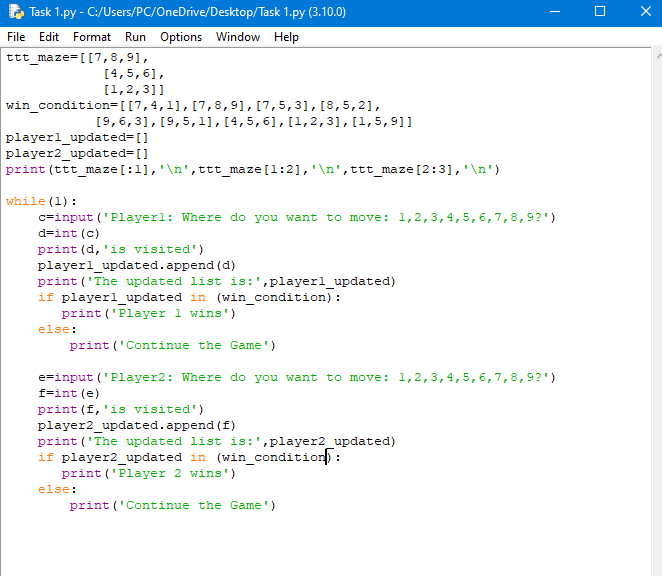
**AI MIDTERM**

**Manaal Waseem**

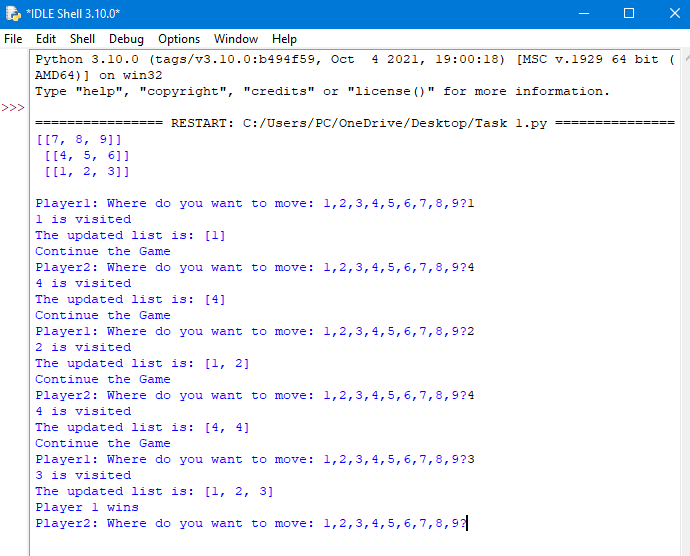
**FA18-BCE-074**

**Task 1:**

**Code:**



**Output:**



**Task 2:  
Code:**

w, h = 5, 5

b=[]

a = [[0 for x in range(w)] for y in range(h)]

a[0][0] = 1

a[0][1] = 3

a[0][2] = 4

a[0][3] = 5

a[0][4] = 2

a[1][0] = 3

a[1][1] = 2

a[1][2] = 1

a[1][3] = 0

a[1][4] = 6

a[2][0] = 6

a[2][1] = 7

a[2][2] = 4

a[2][3] = 2

a[2][4] = 7

a[3][0] = 2

a[3][1] = 3

a[3][2] = 6

a[3][3] = 1

a[3][4] = 3

a[4][0] = 3

a[4][1] = 2

a[4][2] = 4

a[4][3] = 1

a[4][4] = 1

shortest=[]

print(a[:1],'\n',a[1:2],'\n',a[2:3],'\n',a[3:4],'\n',a[4:5],'\n')

start\_node=a[0][0]

print('The start node is:',start\_node)

goal\_node=a[1][3]

print('The goal node is:',goal\_node)

print('SHORTEST PATH TO GOAL NODE IS: \n')

z2=a[1][0]

z3=a[1][3]

def edges():

print('THE EDGES OF EACH NODE IS: \n')

t1=(a[0][0],a[0][1],a[1][0])

t2=((3,2),(3,3))

t3=(4,6)

t4=[(0)]

t5=[(2,4),(2,7)]

t6=((3,3),(3,0))

t7=((2,0),(2,3))

t8=((1,2),(1,0),(1,4),(1,4))

t9=((0))

t10=((0))

t11=((0))

t12=((0))

t13=((0))

t14=((2,1),(2,5))

t15=((0))

t16=((2,6),(2,3))

t17=((3,3),(3,3))

t18=((0))

t19=((1,2),(1,1))

t20=((3,2),(3,3))

t21=((3,3),(3,1))

t22=((2,2),(2,1))

t23=((4,4))

t24=((1,1),(1,1),(1,4))

t25=((1,1),(1,3))

print('EDGE[0][0] : ',t1)

print('EDGE[0][1] : ',t2)

print('EDGE[0][2] : ',t3)

print('EDGE[0][3] : ',t4)

print('EDGE[0][4] : ',t5)

print('EDGE[1][0] : ',t6)

print('EDGE[1][1] : ',t7)

print('EDGE[1][2] : ',t8)

print('EDGE[1][3] : ',t9)

print('EDGE[1][4] : ',t10)

print('EDGE[2][0] : ',t11)

print('EDGE[2][1] : ',t12)

print('EDGE[2][2] : ',t13)

print('EDGE[2][3] : ',t14)

print('EDGE[2][4] : ',t15)

print('EDGE[3][0] : ',t16)

print('EDGE[3][1] : ',t17)

print('EDGE[3][2] : ',t18)

print('EDGE[3][3] : ',t19)

print('EDGE[3][4] : ',t20)

print('EDGE[4][0] : ',t21)

print('EDGE[4][1] : ',t22)

print('EDGE[4][2] : ',t23)

print('EDGE[4][3] : ',t24)

print('EDGE[4][4] : ',t25)

print('\n\n')

shortest.append(z2)

shortest.append(z3)

print(shortest)

print('The length of the shortest path from the start node to the goal node is:')

length=len(shortest)

print(length)

def path\_cost():

print('The shortest path of start node to a[0][0] is :',0)

print('The shortest path of start node to a[0][1] is :',1)

print('The shortest path of start node to a[0][2] is :',3)

print('The shortest path of start node to a[0][3] is ',2)

print('The shortest path of start node to a[0][4] is ',2)

print('The shortest path of start node to a[1][0] is ',1)

print('The shortest path of start node to a[1][1] is ',3)

print('The shortest path of start node to a[1][2] is ',2)

print('The shortest path of start node to a[1][3] is ',2)

print('The shortest path of start node to a[1][4] is',2)

print('The shortest path of start node to a[2][0] is ',2)

print('The shortest path of start node to a[2][1] is ',2)

print('The shortest path of start node to a[2][2] is',2)

print('The shortest path of start node to a[2][3] is',2)

print('The shortest path of start node to a[2][4] is',3)

print('The shortest path of start node to a[3][0] is ',3)

print('The shortest path of start node to a[3][1] is ',2)

print('The shortest path of start node to a[3][2] is ',2)

print('The shortest path of start node to a[3][3] is ',3)

print('The shortest path of start node to a[3][4] is ',3)

print('The shortest path of start node to a[4][0] is ',3)

print('The shortest path of start node to a[4][1] is ',2)

print('The shortest path of start node to a[4][2] is ',3)

print('The shortest path of start node to a[4][3] is ',3)

print('The shortest path of start node to a[4][4] is ',3)

edges()

path\_cost()

**Output:**

[[1, 3, 4, 5, 2]]

[[3, 2, 1, 0, 6]]

[[6, 7, 4, 2, 7]]

[[2, 3, 6, 1, 3]]

[[3, 2, 4, 1, 1]]

The start node is: 1

The goal node is: 0

SHORTEST PATH TO GOAL NODE IS:

[3, 0]

The length of the shortest path from the start node to the goal node is:

2

THE EDGES OF EACH NODE IS:

EDGE[0][0] : (1, 3, 3)

EDGE[0][1] : ((3, 2), (3, 3))

EDGE[0][2] : (4, 6)

EDGE[0][3] : [0]

EDGE[0][4] : [(2, 4), (2, 7)]

EDGE[1][0] : ((3, 3), (3, 0))

EDGE[1][1] : ((2, 0), (2, 3))

EDGE[1][2] : ((1, 2), (1, 0), (1, 4), (1, 4))

EDGE[1][3] : 0

EDGE[1][4] : 0

EDGE[2][0] : 0

EDGE[2][1] : 0

EDGE[2][2] : 0

EDGE[2][3] : ((2, 1), (2, 5))

EDGE[2][4] : 0

EDGE[3][0] : ((2, 6), (2, 3))

EDGE[3][1] : ((3, 3), (3, 3))

EDGE[3][2] : 0

EDGE[3][3] : ((1, 2), (1, 1))

EDGE[3][4] : ((3, 2), (3, 3))

EDGE[4][0] : ((3, 3), (3, 1))

EDGE[4][1] : ((2, 2), (2, 1))

EDGE[4][2] : (4, 4)

EDGE[4][3] : ((1, 1), (1, 1), (1, 4))

EDGE[4][4] : ((1, 1), (1, 3))

The shortest path of start node to a[0][0] is : 0

The shortest path of start node to a[0][1] is : 1

The shortest path of start node to a[0][2] is : 3

The shortest path of start node to a[0][3] is 2

The shortest path of start node to a[0][4] is 2

The shortest path of start node to a[1][0] is 1

The shortest path of start node to a[1][1] is 3

The shortest path of start node to a[1][2] is 2

The shortest path of start node to a[1][3] is 2

The shortest path of start node to a[1][4] is 2

The shortest path of start node to a[2][0] is 2

The shortest path of start node to a[2][1] is 2

The shortest path of start node to a[2][2] is 2

The shortest path of start node to a[2][3] is 2

The shortest path of start node to a[2][4] is 3

The shortest path of start node to a[3][0] is 3

The shortest path of start node to a[3][1] is 2

The shortest path of start node to a[3][2] is 2

The shortest path of start node to a[3][3] is 3

The shortest path of start node to a[3][4] is 3

The shortest path of start node to a[4][0] is 3

The shortest path of start node to a[4][1] is 2

The shortest path of start node to a[4][2] is 3

The shortest path of start node to a[4][3] is 3

The shortest path of start node to a[4][4] is 3