Graph theory and applications
Practise Problem – 2

Find if the Graph is Hamiltonian

Name: Ananya Rao
SRN: PES1UG20CS046

Code-

```
class Graph():
	def __init__(self, vertices):
		self.graph = [[0 for column in range(vertices)]
					for row in range(vertices)]
		self.V = vertices


	def isSafe(self, v, pos, path):
		# Check if current vertex and last vertex
		# in path are adjacent
		if self.graph[ path[pos-1] ][v] == 0:
			return False

		# Check if current vertex not already in path
		for vertex in path:
			if vertex == v:
				return False

		return True

	def hamCycleUtil(self, path, pos):

		# base case: if all vertices are
		# included in the path
		if pos == self.V:
```

```
                # Last vertex must be adjacent to the
                # first vertex in path to make a cycle
                if self.graph[ path[pos-1] ][ path[0] ] == 1:
                        return True
                else:
                        return False

        for v in range(1,self.V):

                if self.isSafe(v, pos, path) == True:

                        path[pos] = v

                        if self.hamCycleUtil(path, pos+1) == True:
                                return True

                        # Remove current vertex if it doesn't
                        # lead to a solution
                        path[pos] = -1

        return False

def hamCycle(self):
        path = [-1] * self.V

        path[0] = 0

        if self.hamCycleUtil(path,1) == False:
                print ("Solution does not exist\n")
                return False

        self.printSolution(path)
        return True

def printSolution(self, path):
        print ("Solution Exists: Following",
                        "is one Hamiltonian Cycle")
        for vertex in path:
                print (vertex, end = " ")
        print (path[0], "\n")
```

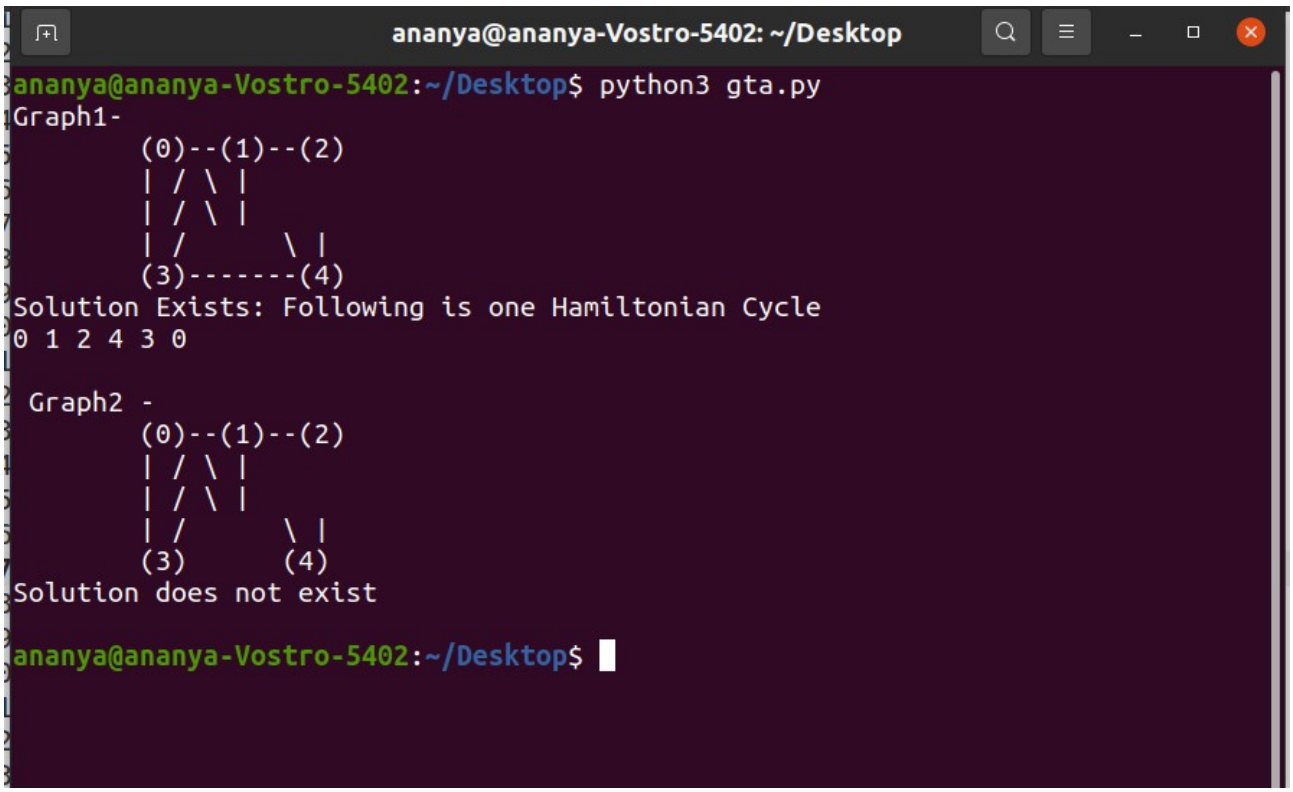
```
print( '''Graph1-
        (0)--(1)--(2)
        | / \ |
        | / \ |
        | /       \ |
        (3)-------(4) ''')
g1 = Graph(5)
g1.graph = [ [0, 1, 0, 1, 0], [1, 0, 1, 1, 1],
                        [0, 1, 0, 0, 1,],[1, 1, 0, 0, 1],
                        [0, 1, 1, 1, 0], ]

# Print the solution
g1.hamCycle();

print(''' Graph2 -
        (0)--(1)--(2)
        | / \ |
        | / \ |
        | /       \ |
        (3)      (4) ''')
g2 = Graph(5)
g2.graph = [ [0, 1, 0, 1, 0], [1, 0, 1, 1, 1],
                [0, 1, 0, 0, 1,], [1, 1, 0, 0, 0],
                [0, 1, 1, 0, 0], ]

# Print the solution
g2.hamCycle();
```

Output Screenshot-


ananya@ananya-Vostro-5402: ~/Desktop
ananya@ananya-Vostro-5402:~/Desktop$ python3 gta.py
Graph1-
(0)--(1)--(2)
| / \ |
| / \ |
| / \ |
(3)-------(4)
Solution Exists: Following is one Hamiltonian Cycle
0 1 2 4 3 0
Graph2 -
(0)--(1)--(2)
| / \ |
| / \ |
| / \ |
(3) (4)
Solution does not exist
ananya@ananya-Vostro-5402:~/Desktop$