```
def Hamiltonian(n, G):
    x = [0] * (n + 1)
    x[1] = 1

    def NextVertex(k):
        while True:
            x[k] = (x[k] + 1) % (n + 1)
            if x[k] == 0:
                return
            if G[x[k - 1] - 1][x[k] - 1] != 0:
                for j in range(1, k):
                    if x[j] == x[k]:
                        break
                else:
                    if k < n or (k == n and G[x[n] - 1][x[1] - 1] != 0):
                        return

    def HamiltonianCycle(k):
        while True:
            NextVertex(k)
            if x[k] == 0:
                return False
            if k == n:
                if G[x[n] - 1][x[1] - 1] != 0:
                    return x[1:n + 1]
            else:
                result = HamiltonianCycle(k + 1)
                if result:
                    return result

    cycle = HamiltonianCycle(2)
    if cycle:
        print("Hamiltonian Cycle:", cycle)
    else:
        print("No Hamiltonian Cycle exists.")

G1 = [
    [0,1,1,0,1],
    [1,0,1,1,0],
    [1,1,0,1,0],
    [0,1,1,0,1],
    [1,0,0,1,0]
]
G2 = [
```

```
    [0,1,1,0,1],
    [1,0,1,1,0],
    [1,1,0,1,1],
    [0,1,1,0,1],
    [1,0,1,1,0]
]

print("Graph 1:")
Hamiltonian(5, G1)
print("\nGraph 2:")
Hamiltonian(5, G2)
```