

```

In [5]: import networkx as nx
import matplotlib.pyplot as plt

def greedy_coloring(G):
    colors = {}
    for node in G.nodes():
        neighbor_colors = set(colors.get(neigh, None) for neigh in G.neighbors(node))
        for color in range(len(G.nodes())):
            if color not in neighbor_colors:
                colors[node] = color
                break
    return colors

def visualize_graph(G, colors):
    node_colors = [colors[node] for node in G.nodes()]
    nx.draw(G, with_labels=True, node_color=node_colors, cmap=plt.cm.rainbow)
    plt.show()

# Create an empty graph
G = nx.Graph()

# Get number of nodes and edges from user
num_nodes = int(input("Enter the number of nodes: "))
num_edges = int(input("Enter the number of edges: "))

# Add nodes to the graph
for i in range(num_nodes):
    G.add_node(i)

# Add edges to the graph
print("Enter edges (format: node1 node2):")
for _ in range(num_edges):
    edge = input().split()
    G.add_edge(int(edge[0]), int(edge[1]))

# Greedy coloring
colors = greedy_coloring(G)

# Visualize the graph with colors
visualize_graph(G, colors)

```

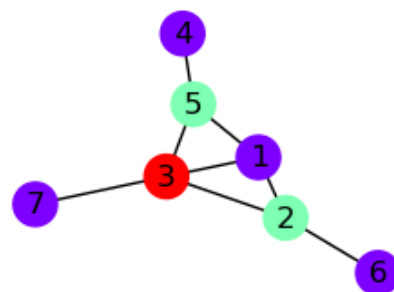
```

Enter the number of nodes: 10
Enter the number of edges: 9
Enter edges (format: node1 node2):
1 2
2 3
3 2
4 5
5 3
3 1
1 5
6 2
7 3

```

9

8



0

In [ ]: