5 Graph Planarity and Directed Graphs

This module covers planarity, Kuratowski's theorem, Euler's formula, and directed graph properties. The exercises below will help you implement and understand these concepts in Python.

5.1 Exercise 1: Planarity Testing (Kuratowski's Theorem)

Task: Implement a function to check if a given graph is planar using Kuratowski's theorem.
Hint: A graph is non-planar if it contains a subgraph homeomorphic to K_5 or $K_{3,3}$.

```
import networkx as nx

def is_planar(graph):
    return nx.check_planarity(graph)[0]
```

5.2 Exercise 2: Euler's Formula Verification

Task: Given a planar graph, verify Euler's formula: $V - E + F = 2$.

5.3 Exercise 3: Planar Graph Embedding

Task: Implement a function to draw a planar embedding of a given planar graph.

5.4 Exercise 4: Directed Acyclic Graph (DAG) Detection

Task: Implement a function to check if a directed graph is acyclic.
Hint: Use topological sorting or cycle detection algorithms.

5.5 Exercise 5: Strongly Connected Components (Kosaraju's Algorithm)

Task: Implement Kosaraju's algorithm to find strongly connected components in a directed graph.

5.6 Exercise 6: Eulerian and Hamiltonian Digraphs

Task: Implement algorithms to check for Eulerian and Hamiltonian paths in directed graphs.

5.7 Bonus Challenge 1: Graph Drawing with TikZ

Task: Use TikZ to draw planar embeddings of different graphs.

5.8 Bonus Challenge 2: Planarity Testing with Random Graphs

Task: Generate random graphs and test their planarity.