

Empecé creando el constructor `__init__` y usé un diccionario para guardar los vértices con sus vecinos. Después hice el método para agregar vértices y otro para agregar aristas, cuidando que si el grafo no era dirigido, la arista se agregara en ambos sentidos.

Luego hice los métodos para obtener vecinos y verificar si existe una arista. Con eso ya podía probar si se agregaban bien las conexiones.

Después programé los recorridos BFS y DFS. Usé un deque para el BFS y recursión para el DFS. Me costó un poco entender cómo evitar repetir vértices, pero lo resolví usando un conjunto (set) de visitados.

Para terminar, hice los métodos `es_conexo` y `encontrar_camino`, usando el BFS para verificar si todos los nodos se pueden alcanzar y para reconstruir el camino entre dos nodos.

Probé todo con los ejemplos del ejercicio y funcionó bien. Al final agregué algunos print para ver que todo saliera correcto y bonito.