

1 Projekt

Ziel war es, in kleinen Gruppen eigene Aufgabenstellungen in Python zu lösen. Verschiedene Texte über graphentheoretische Probleme waren die Grundlage für die Themenfindung. Bei manchen Problemen war es zuerst erforderlich, sich zu überlegen, wie man es graphentheoretisch darstellen kann. Um den Erfolg des Projekts zu sichern, haben wir zuerst die Ein- und Ausgabe spezifiziert. Außerdem braucht man eine Erfolgsmetrik, um zu prüfen, ob das Programm wie gewünscht funktioniert. Um Chaos während der Entwicklung zu vermeiden, entwarfen wir einen Entwicklungsplan, der das Problem in einzelne Komponenten unterteilt.

2 Beispielprojekt

2.1 Aufgabe

Es soll ein Algorithmus programmiert werden, der prüft, ob zwei gegebene Graphen isomorph sind.

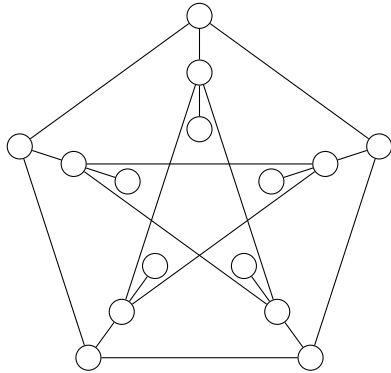
2.2 Umsetzung

Der erste Ansatz war, ein Programm zu schreiben, das das Problem in polynomieller Laufzeit löst. Während dessen haben wir herausgefunden, dass das Problem zu der Klasse NP^1 gehört. Deshalb verfolgten wir die Strategie, die eingegebenen Graphen auf bestimmte Eigenschaften zu prüfen und zu vergleichen. Zu diesen Eigenschaften zählen die Anzahl der Knoten und Kanten, der Grad der Knoten, Grade von Nachbarknoten, Kreise und Längen von kürzesten Wegen im Graphen.

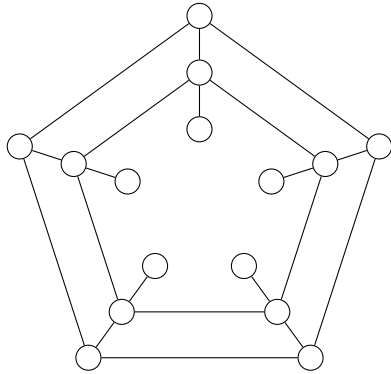
Wenn die Graphen isomorph sind, kann man mit Hilfe der Eigenschaften jedem Knoten des einen Graphen einen des anderen Graphen zuordnen. Wenn die Werte nicht zueinander passen, sind die Graphen nicht isomorph. Es gibt einige Paare von Graphen, für die die Laufzeit des Algorithmus extrem hoch ist, was im Folgenden veranschaulicht wird.

Beispiel:

¹Es sind bisher nur Algorithmen bekannt, die das Problem in nicht polynomieller Zeit lösen.



Der erste Graph ist nicht zum zweiten isomorph. Es ist aber möglich, die Knoten des einen Graphen bijektiv denen des anderen Graphen zuzuordnen, sodass die Grade der Knoten, Nachbarknoten, Nachbarnachbarknoten und Nachbar...nachbarknoten für jeden Knoten des ersten Graphen mit dem zugeordneten Knoten des anderen übereinstimmen.



Das Problem hat sich während des Entwicklungsprozesses als schwerer als erwartet herausgestellt, trotzdem ist der finale Algorithmus zufriedenstellend. Neben diesem Projekt wurden in anderen Gruppen verschiedene Anwendungen des Dijkstra-Algorithmuses ² behandelt.

²Der Dijkstra-Algorithmus bestimmt den kürzesten Weg zu jedem Knoten in einem Graph ausgehend von einem Startknoten.