

Entwickeln Sie eine Klasse zur Verwaltung von ungerichteten bewerteten Graphen zur Speicherung von kurzen Texten mit folgenden Methoden:

1. leeren Graph erzeugen mit Konstruktor
2. Knoten hinzufügen mit Knotentext
3. Kante hinzufügen mit Bewertung
4. Graph anzeigen (alphanumerisch als Werteliste oder grafisch im Dialogfenster)
5. Kante löschen
6. Knoten löschen
7. Graph in externe Datei speichern (Text- oder Binärdatei)
8. Graph aus externer Datei lesen (Text- oder Binärdatei)
9. zufälligen Graph mit vorgebbbarer Knoten- und Kantenzahl erzeugen

Schreiben Sie ein Konsolenprogramm oder eine Dialoganwendung, die in der Lage sind, einen Graphen interaktiv einzugeben, anzuzeigen und in einer externen Datei abzuspeichern. Bei einem weiteren Testlauf ist der Graph einzulesen, anzuzeigen und weitere Veränderungen an Knoten und Kanten sind durch Nutzereingaben zu ermöglichen. Zur Verwaltung des Graphen ist eine Adjazenzmatrix oder eine Adjazenzzliste zu verwenden (siehe Vorlesung/Übung).

Entwerfen Sie zwei Testbeispiele für Graphen mit mindestens 10 Knoten und 15 Kanten. Simulieren Sie folgende Veränderungen und zeichnen Sie das Endergebnis auf Papier (Entfernen von mindestens zwei Knoten / Kanten, Hinzufügen von mindestens drei neuen Knoten / Kanten).

Entwerfen Sie ein Testbeispiel aus dem realen Berliner Straßen- oder Wasserstraßennetz (Bewertung der Kante = ungefähr der Länge der Verbindung) mit mindestens 10 Knoten.

Die Kurzdokumentation soll die Testbeispiele als Skizze, eine Bildschirmkopie wichtiger Ausgaben des Programms (Menü, Masken, Graphanzeige) sowie die wichtigsten aufgetretenen Probleme enthalten.

Konsolenlösung

15 Pkt

Fakultative Erweiterungen:

- a) Entwickeln Sie eine grafische Oberfläche für die Graphoperationen mit Dialogfeldern und einer zeichnerischen Darstellung des Graphen max. **8 Pkt**
 - b) Schreiben Sie eine Methode zur Bestimmung eines Weges zwischen zwei Knoten mit Hilfe der Tiefensuche. **1 Pkt**
 - c) Schreiben Sie eine Methode zur Bestimmung des kürzesten Weges zwischen zwei Knoten mit Hilfe der Breitensuche. **1 Pkt**
- oder
- d) individuelle Erweiterungen der Funktionalität max. **2 Pkt**

Bewertet werden der Programmquelltext 50%, die korrekte Arbeitsweise 30%, die Dokumentation 10% sowie die vorbereiteten Testbeispiele 10%