

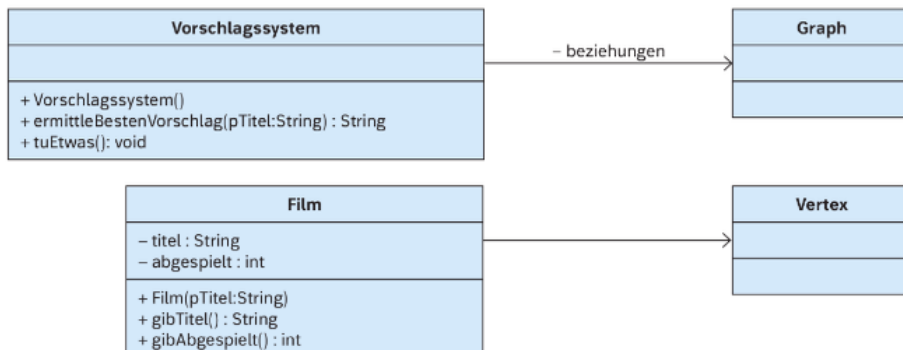
Aufgabe 2

Die Videostreaming-Plattform Netflix bietet den Usern eine große Anzahl an Filmen und Serien in einem Abomodell an. Nachdem ein Video geschaut wurde, bietet die App eine Auswahl von fünf Titeln an, die den Nutzenden interessieren und zum Weiterschauen anregen könnten.

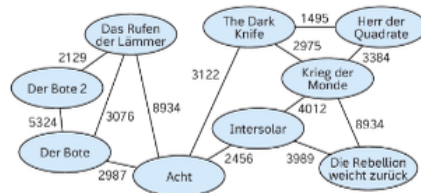
Die Plattform nutzt einen Algorithmus, der basierend auf dem Verhalten der User die beste Auswahl für die Vorschläge trifft. Wenn beispielsweise viele User Film A und Film B geschaut haben, dann wird Film B häufig als Vorschlag nach Film A angeboten und umgekehrt. Der Film, der am häufigsten in Kombination mit dem gerade geschauten vorkommt, gilt als der beste Vorschlag.

- a Geben Sie an, welchen Film das Vorschlagssystem jeweils vorschlägt, nachdem
- der Film „Intersolar“,
 - der Film „Der Bote“ geschaut wurde.

- b Nach einer Aktualisierung der Filmbeziehungen sind folgende Daten hinzugekommen: 3611 Nutzende, die den Film „Krieg der Monde“ und 2045, die „Die Rebellion weicht zurück“ gesehen haben, schauten auch den Film „Vorwärts in die Vergangenheit“. 321 neue Nutzende, die „Herr der Quadrate“ gesehen haben, schauten auch „The Dark Knife“. Stellen Sie die Änderungen durch diese neuen Daten im Graphen dar. Es müssen nur die Knoten und Kanten mit Änderungen dargestellt werden.



Die Beziehung zwischen den Filmen wurde für das Vorschlagssystem als ungerichteter, gewichteter Graph modelliert. Die Kantengewichte geben an, wie viele Nutzenden beide der beteiligten Filme geschaut haben. Der folgende Graph zeigt einen Ausschnitt des Modells.



- c Stellen Sie den gezeigten Graphen als Adjazenzmatrix dar und erläutern Sie, welche Bedeutung Einträge auf der Diagonalen für den Sachkontext hätten.

- d Das Implementationsdiagramm unten zeigt einen Ausschnitt der Modellierung für das Vorschlagssystem. Die Methode `public String ermittleBestenVorschlag(String pTitel)` der Klasse `Vorschlagssystem` ermittelt den Titel des besten Vorschlags für den Film mit dem Titel `pTitel`. Kommen zwei Filme gleich oft vor, wird einer der beiden Filme zurückgegeben. Gibt es keinen besten Vorschlag, ist das Ergebnis `null`. Implementieren Sie die Methode `ermittleBestenVorschlag` nach der Vorgabe.

e Der Quelltext der Methode `public Graph tuEtwas()` der Klasse `Vorschlagssystem` ist unten zu sehen.

```

public Graph tuEtwas() {
    Graph g = new Graph();
    List<Vertex> vertices = beziehungen.getVertices();
    vertices.toFirst();
    while( vertices.hasAccess() ) {
        Vertex v1 = vertices.getContent();
        Vertex v2 = beziehungen.getVertex(
            ermittleBestenVorschlag(v1.getID())
        );
        Edge e = beziehungen.getEdge(v1, v2);

        g.addVertex(v1);
        g.addVertex(v2);
        g.addEdge(e);

        vertices.next();
    }

    return g;
}
  
```

Analysieren Sie die Methode, indem Sie die Funktionsweise des Algorithmus erklären und den erzeugten Graphen für den Beziehungsgraphen auf der vorherigen Seite skizzieren. Beschreiben Sie dann die Funktion der Methode im Sachzusammenhang.

- f Die User des Anbieters wünschen sich eine Funktion, mit der sie alle Filme sehen können, die 2, 3 oder 4 Empfehlungen von einem Film entfernt liegen. Beschreiben Sie eine algorithmische Strategie, mit der eine solche Funktion umgesetzt werden könnte.

- g Netflix möchte seinen Nutzenden eine neue Funktion anbieten, bei der zwei Filme A und B angegeben werden können und automatisch eine Wiedergabeliste mit Filmen angelegt wird, die eine Verbindung zwischen den

beiden Filmen darstellt. A ist immer der erste Film, der zweite Film wurde von vielen Nutzenden geschaut, die A geschaut haben, der dritte Film von vielen, die den Zweiten gesehen haben, und so weiter, bis als letztes Film B erreicht ist.

Ein Entwickler schlägt vor, jedes Kantengewicht von 10000 abzuziehen (aus 3076 wird also $10000 - 3076 = 6924$), um so stärkeren Verbindungen ein kleineres Gewicht zuzuordnen und schwächeren ein Höheres. Nun seien die Voraussetzungen geschaffen, um mit dem Algorithmus von Dijkstra die kürzeste Strecke von Film A zu Film B zu suchen und damit eine Liste von Filmen mit den höchsten Übereinstimmungen. Beurteilen Sie den Vorschlag des Entwicklers hinsichtlich der Problemstellung und der Umsetzbarkeit.