# Datenstrukturen & Algorithmen

Louis Graßmann, Joshua Stripp, Simon Hain

## Historische Funde

### Beschreibung

Beim Ausheben der Wege während des Straßenbaus wurde ein antiker Feuerwerksplan gefunden. Die Lage der pyrotechnischen Effekte und die Zündschnüre sind noch sehr gut zu erkennen. Wie aber ist die Choreografie des Feuerwerks? In welcher Reihenfolge zünden die Bomben? Können Sie den Bürgern der Graphschaft Schilda helfen? (Unter der Annahme, dass die Zündschnur immer mit gleichbleibender Geschwindigkeit abbrennt…)

### Welche Daten sollen verarbeitet werden?

Graph mit Feuerwerkskörper, die mit unterschiedlichen Zündzeiten verbunden sind

### Was sind die Eingaben?

* 2-Dimensionales int-Array, welches einen gewichteten, ungerichteten Graphen darstellt
  + Knoten = Feuerwerkskörper
  + Gewichtete Kanten = Zündverzögerung
  + Knoten 0 wird gezündet

### Was sind die Ausgaben?

* int-Array mit Knoten, sortiert nach der Reihenfolge in denen diese Zünden
* Ausgabe wird auch auf Konsole gedruckt

### Welcher Algorithmus eignet sich?

* Dijkstra-Algorithmus sucht sich immer den Knoten mit kürzester Distanz (muss unbesucht sein)
* Wiederholung, bis alle Knoten hinzugefügt wurden

### Welche Datenstruktur eignet sich?

* Graph-Objekt mit Knoten und Kanten
* Selbst erstellt

### Was ist die Laufzeit?

Die Laufzeit des Dijkstra-Algorithmus ist O(n2)

## Aktivitätsdiagramm

### Beschreibung

* Weil es nicht nur viele Projekte gibt, sondern in den Projekten auch noch viel zu tun  
   ist, wünscht sich die Graphschaft auch Unterstützung bei der Planung der einzelnen  
   Aktivitäten in den Projekten. Die Erfahrung zeigt, dass ohne Planung, viele Projekte  
   im Chaos enden.
* Es gibt Aktivitäten, die erst ausgeführt werden können, wenn andere Aktivitäten  
   bereits beendet sind, andere Aktivitäten hängen nicht voneinander ab. Aber wie die  
   ganzen Aktivitäten in die richtige Reihenfolge bringen?
* Wieder ein Auftrag für Sie: Sie werden beauftragt, das Planungstool so zu erweitern,  
   dass damit auch die Aktivitätenreihenfolge für die einzelnen Projekte geplant werden  
   kann.

### Welche Daten sollen verarbeitet werden?

Graph, der anzeigt, welche Aktivitäten, vor anderen gemacht werden muss

### Was sind die Eingaben?

* 2-Dimensionales int-Array, welches einen ungewichteten, gerichteten Graphen darstellt
  + Knoten = Aktivität
  + Gerichtete Kante von Knoten 1 zu Knoten 2 = Aktivität 1 muss vor Aktivität 2 ausgeführt werden

### Was sind die Ausgaben?

* int-Array mit Knoten, sortiert nach der Reihenfolge in denen die Aktivitäten ausgeführt werden sollten
* Ausgabe wird auch auf Konsole gedruckt

### Welcher Algorithmus eignet sich?

* DFS-Algorithmus
  + Wird von jedem Knoten ausgeführt, zu dem keine Kante führt

### Welche Datenstruktur eignet sich?

* Graph-Objekt mit Knoten und Kanten
* Selbst erstellt

### Was ist die Laufzeit?

Die Laufzeit des vorliegenden Algorithmus hängt von der Anzahl der Knoten (|V|) und Kanten (|E|) im Graphen ab. Die Schleife while (!remainingVertices.isEmpty()) wird höchstens |V| Iterationen durchlaufen, da in jeder Iteration mindestens ein Knoten aus remainingVertices entfernt wird. Die Methode getVertexWithMostConnections durchläuft alle verbleibenden Knoten und sucht nach dem Knoten mit den meisten ausgehenden Kanten, was O(|V|) Zeit in Anspruch nimmt. Innerhalb der Schleife wird ein Knoten und seine ausgehenden Kanten entfernt, wobei das Entfernen eines Knotens im Allgemeinen O(1) dauert und das Entfernen aller ausgehenden Kanten in O(k), wobei k die Anzahl der ausgehenden Kanten des entfernten Knotens ist.

Zusammengefasst ergibt sich eine geschätzte Gesamtlaufzeit von O(|V|^2 + |E|).

## Empfehlungssystem

### Beschreibung

* In den letzten Jahren wurde die Graphschaft Schilda zum Touristenmagneten. Die Stadt hat über die Jahr genau verfolgt, welche Touristen welche Stätten gemeinsam besuchen:
  + Beispielsweise besuchen Touristen, die den Burgfried besuchen häufig auch die historische Ausstellung im Schloss. Touristen, die eine architektonische Führung gebucht haben, gehen anschließend gern im Schlosskeller etwas trinken und so weiter, einige besuchen danach auch noch das Restaurant Hexenkeller. So konnte zwischen allen Sehenswürdigkeiten und Veranstaltungsorten eine mehr oder weniger starke Verbindung festgestellt werden (mit Werten zwischen 1 und 10 belegt).
* Die Stadt wünscht sich nun ein Empfehlungstool, das aufgrund individueller Wünsche (Kurzbesuch, Wochenendbesuch, offen für Neues, etc) Empfehlungen für Freizeitaktivitäten gibt.

### Welche Daten sollen verarbeitet werden?

* Matrix, von wo was am meisten empfohlen wurde
* Standort?
* Liste von empfohlenen Orten

### Was sind die Eingaben?

* Empfehlungsmatrix
* Standort?

### Was sind die Ausgaben?

* Liste von empfohlenen Orten

### Welcher Algorithmus eignet sich?

* K-Means?

### Welche Datenstruktur eignet sich?

### Was ist die Laufzeit?

## ProjektAufgaben

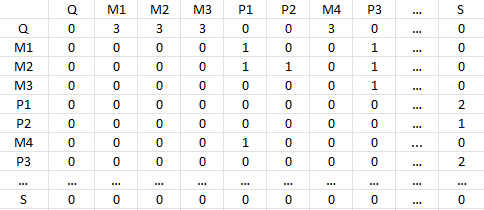
### Beschreibung

* In der Graphschaft Schilda gibt es viel zu tun. Die Graphschaft hat viele Mitarbeiter:innen mit unterschiedlichen Kompetenzen (Finanzen, Planung, Straßenbau, Festorganisation, und so weiter).
* Viele Projekte sind zu bewältigen, viel mehr, als es Mitarbeitende gibt. Gleichzeitig müssen einige Projekte von mehreren Mitarbeitenden gemeinsam bearbeitet werden um erfolgreich zu sein.
* Die Graphschaft hat nun, um die Mitarbeitenden vor Überlastung zu schützen, die Policy aufgestellt, dass jede Mitarbeiterin und jeder Mitarbeiter an maximal 3 Projekten arbeiten darf.
* Damit nun die Kräfte möglichst gut eingesetzt werden, alle Mitarbeier:innen entsprechend ihrer Kompetenzen Projekten zugeordnet sind, aber kein Mitarbeiter/keine Mitarbeiterin mehr als 3 Projekte bearbeitet, bittet die Stadt Schilda Sie ein Planungstool für das Matching zwischen Mitarbeiter:innen und Projekten zu erstellen.

### Welche Daten sollen verarbeitet werden?

* Mitarbeiter mit Kompetenzen
* Projekte mit Kompetenzbereich
* Anzahl an benötigten Mitarbeiter pro Projekt

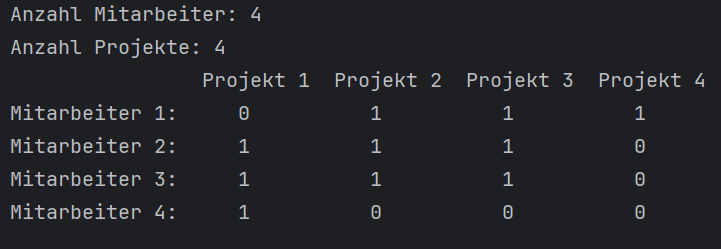
### Was sind die Eingaben?

* Ein Zweidimensionales Array, welches Informationen über die Kompetenzen der Mitarbeiter und der Projekte enthält
  + Aufbau Matrix: 
* Erklärung Der Input Matrix:
  + Q: Superquelle, hat nur Verbindungen zu den Mitarbeitern, Falls Verbindung besteht, Gewichtet mit 3, da ein Mitarbeiter max. 3 Projekte gleichzeitig bearbeiten darf
  + S: Supersenke, Verbindungen von jedem Projekt zu der Supersenke. Gewicht gibt an, wie viel ein Projekt minimal an Mitarbeitern benötigt
  + M...: steht hier für Mitarbeiter
  + P...: steht hier für Projekt
    - Beispiel Input:
      * {0, 3, 3, 3, 0, 0, 3, 0, 0, 0},
      * {0, 0, 0, 0, 1, 1, 0, 1, 1, 0},
      * {0, 0, 0, 0, 1, 1, 0, 1, 1, 0},
      * {0, 0, 0, 0, 1, 1, 0, 1, 1, 0},
      * {0, 0, 0, 0, 0, 0, 0, 0, 0, 3},
      * {0, 0, 0, 0, 0, 0, 0, 0, 0, 3},
      * {0, 0, 0, 0, 1, 0, 0, 0, 0, 0},
      * {0, 0, 0, 0, 0, 0, 0, 0, 0, 3},
      * {0, 0, 0, 0, 0, 0, 0, 0, 0, 3},
      * {0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
      * };

### Was sind die Ausgaben?

Die Ausgabe ist ein zweidimensionales Array. Jede Zeile entspricht ein Mitarbeiter. Jede Spalte entspricht ein Projekt. Falls ein Mitarbeiter einem Projekt zugeordnet ist, so beträgt der Wert in deren Kreuzfeld 1. Wenn ein Mitarbeiter dem jeweiligen Projekt nicht zugeordnet ist, da die benötigten Kompetenzen nicht übereinstimmen, so befindet sich in deren Schnittfeld eine 0.

* Beispiel Output :



### Welcher Algorithmus eignet sich?

* Für diese Aufgabe eignet sich der Ford Fulkerson Algorithmus, der den maximalen Fluss eines Graphen berechnet.
* Unsere Klasse Ford Fulkerson gibt einen MaxFlow int - Wert aus, sowei einen Flowgraph und einen Restgraph des Datentyps graph zurück.
* Anhand des Flowgraphen, können dann die einzelnen Zuordnungen ausgelesen werden und in ein geordnetes zweidimensionales Array eingefügt werden.

### Welche Datenstruktur eignet sich?

* Benutzte Datenstrukturen:
  + Array
  + List

Beide Datenstrukturen helfen die Datenstruktur bipartiter Graph (Hier: Zeilen = Mitarbeter; Spalten = Projekte) umzusetzen

### Was ist die Laufzeit?

* Die Laufzeit des Ford Fulkerson Algorithmus ist abhängig von der Anzahl der Kanten und Knoten im Graphen.
  + Ford Fulkerson Alg.: O(|E| \* f)
    - |E|: Anzahl Kanten im Graphen
    - f: maximaler Flusswert
  + Methode projekteZuteilen: O(m\*n);
    - m: Anzahl Mitarbeiter
    - n: Anzahl Projekte

Gesamtlaufzeit: O(|E| \* f + m \* n)

## Winterwege

### Beschreibung

* Im Winter sind die Wege in Schilda immer sehr verschneit, vereist und glatt. Damit die Touristen nun gut und sicher vom Parkhaus zu den vielen Sehenswürdigkeiten und historischen Plätzen kommen können, hat sich die Stadt Schilda etwas besonderes überlegt: Vom Parkhaus aus werden beheizte Wege eingerichtet, die die Straßen eisfrei halten.
* Dieses Vorhaben ist natürlich sehr teuer: ein Meter Weg kostet mehrere 100 Schilda- Gulden. Die Stadt Schilda hat daher die Bitte an Sie, in Ihr Planungstool eine Funktionalität einzufügen, die basierend auf einem bestehenden Stadtplan die günstigste Auswahl an zu beheizenden Wegen identifiziert, so dass alle Plätze und Sehenswürdigkeiten über beheizte Wege erreicht werden können, die Kosten dabei aber möglichst gering gehalten werden.

### Welche Daten sollen verarbeitet werden?

* Gewichteter, ungerichteter Graph mit Länge der Wege zwischen verschiedenen Orten.

### Was sind die Eingaben?

* Graph mit Wegen.

### Was sind die Ausgaben?

* Wege, welche beheizt werden sollten.

### Welcher Algorithmus eignet sich?

* Prim Algorithmus

### Welche Datenstruktur eignet sich?

* Graph, gewichtet und ungerichtet

### Was ist die Laufzeit?

## Woerterbuch

### Beschreibung

Die Stadt Schilda floriert und hat sich zu einem ordentlichen Publikumsmagneten entwickelt. Leider kommt es immer wieder zu Problemen, da die Touristen die Bürger Schildas nicht verstehen – kein Wunder, denn die Sprache Schildas ist sehr ungewöhnlich: Der Sprachschatz Schildas ist immens. Die Bürger Schildas sind sehr kreativ und erfinden täglich neue Wörter und erweitern damit ihren Wortschatz. Sie sollen nun ein Wörterbuch entwickeln, welches sehr effizient neue Wörter aufnehmen und die Bedeutung von Wörtern suchen kann.

### Welche Daten sollen verarbeitet werden?

* Wörter in ihren Übersetzungen Deutsch und Schilda

### Was sind die Eingaben?

* Wörter in Schilda und ihre Übersetzung in Deutsch
* Oder nur Wörter in Schilda
* Nur Wörter in Deutsch

### Was sind die Ausgaben?

* Übersetzung auf Deutsch
* Übersetzung auf Schilda

### Welcher Algorithmus eignet sich?

* /

### Welche Datenstruktur eignet sich?

* HashMap -> Jedes Wort auf Schilda ist Key und Value ist die Übersetzng
* Auch andersrum Erlaubt

### Was ist die Laufzeit?

* Normalerweise O(1), kann in seltenen Fällen auch O(n) sein