

Zusammenfassung Treffen 18.12.2017

- **Nächstes Treffen voraussichtlich am 22.1.2018**

Allgemein:

- Unsere bisherige Erarbeitung ist eine gute Basis für die zukünftigen Aufgaben
- Die nächsten Aufgaben sollten sein:
 - Tagesablaufprogramm verfeinern
 - Datenbanken für bestimmte Typen (Restaurant, Friseur, Zuhause, Arbeit, etc.) aus OSM mit Parser-Programm von Thomas und Ehab erstellen
 - Routing-Algorithmen/-Programme entwickeln
- Es besteht keine Datenbank für Personenknoten und Patrick war sich noch nicht ganz sicher, ob wir überhaupt eine benötigen
 - Idee von unserem Treffen am 15.12.17 war ja eine Personen-Datenbank mit Personen-IDs zu erstellen und diese IDs für den Tagesablauf zu verwenden
 - Es ist nicht schlimm (sondern eher erwünscht), wenn wir die Simulation/unser Programm mit zum Beispiel 500 Personenknoten starten und diese Personen in einer anderen (später gestarteten) Simulation dann andere Personen im Vergleich zur ersten Simulation sind
 - Daher steht im Folgenden hinter dieser Idee ein (?)

Aufgabe: Tagesablauf verfeinern

- Zeiteinheit verkleinern (Minuten oder double-Werte)
- Tagesablauf sollte am besten die Koordinaten von den Orten aus OSM ausgeben, die an dem Tag besucht wurden (lässt sich leicht anpassen)
- Restriktionen erweitern
 - nach 20 Uhr nicht mehr zum Friseur
 - vielleicht Öffnungszeiten aus OSM ziehen
 - nicht 2 gleiche Typen hintereinander
 - Sonderfälle: Supermarkt, Kneipe,...
 - deutlich verfeinern, z.B. mit einer booleschen Funktion in Datenbank namens *Wiederholung*
- Routinen reinbringen
 - jeder Personenknoten sollte eine ID besitzen (?)
 - diese ID dann dem Programm als Input geben, sodass wir Orte und Zeiteinheiten speichern können (sind jetzt noch zufällig) (?)
 - jeder Personenknoten sollte ein (oder mehrere) konstante(s) Zuhause und eine (oder mehrere) konstante Arbeitsstelle(n) besitzen
 - ebenfalls mit ID der Personenknoten festlegen (?)
 - Tagesablauf zu Wochenablauf erweitern
 - andere Tagesabläufe unter der Woche als am Wochenende
- Initialisierung von Personen
 - Input von z.B. 500 Personen
 - eine Wahrscheinlichkeitsfunktion soll festlegen, ob die Person ein Student, Arbeiter, Kind, Schüler, Rentner, etc. ist
 - z.B. $\frac{1}{4}$ Studenten, $\frac{1}{3}$ Arbeiter, $\frac{1}{12}$ Kinder, $\frac{1}{6}$ Schüler, $\frac{1}{6}$ Rentner

- Personentypen sollten also am besten *abstrakt* erstellt werden
 - jeder Typ soll sich anders Verhalten
- Falls Personendatenbank erstellt wird, könnten hier auch die konstanten Zuhause und Arbeitsstellen eingefügt werden (?)
- Tagesablauf an Personentyp anpassen
 - für jeden Personentypen soll es möglich sein ein eigenes Tagesablaufverhalten zu erstellen
 - Verhalten sollte also versucht werden *abstrakt* erstellt zu werden
 - Idee von Sven: Typ von Person dem Tagesablauf als Input geben
- In Tagesablauf sollte eine *Traveltime* (bezüglich des Routings) integriert werden

Aufgabe: Datenbanken erstellen

- Datenbankerstellung erweitern
 - eine Datenbank (Lise, csv, dict, ...) für Orte vom Typ „Zuhause“ und „Arbeit“
 - jedes Restaurant ist auch eine Arbeit
 - jedes Restaurant könnte auch ein Zuhause sein
 - Idee
 - Für jeden Typ eine Datenbank erzeugen
 - Diese Datenbanken können dann immer noch zu einer großen Datenbank zusammengesetzt werden

Aufgabe: Routing

- Soll auf der Ausgabe der Koordinaten aus dem Tagesablauf basieren
 - Routing zwischen diesen Koordinaten
- Gibt es schon einen Routing-Algorithmus/ ein Routing-Programm für OSM?
 - Falls nicht: Dijkstra
- Routing soll erstmal ohne weitere Präferenzen stattfinden
 - also zunächst keine Wahrscheinlichkeiten für Straßenbenutzung einbringen
 - erstmal nur einen Weg finden/gehen
- Routing soll offline stattfinden
 - nicht auf Dienste wie GoogleMaps zurückgreifen
 - also Routing soll komplett ohne Online-Ressourcen funktionieren