Paper Computer Science Experiment ❻

Great Principles of Computing	**Computation (NP-Vollständigkeit)**
Thema	**Traveling Salesman**
Unterrichtsform	Lernen am Modell
Voraussetzung	Wahrscheinlich kennen viele Schüler/innen das Problem des Handlungsreisenden aus dem Alltag. Es modelliert die Aufgabe, mehrere Orte nacheinander und nur einmal auf dem kürzesten Weg zu besuchen und zum Anfangsort zurückzukehren. Handelsreisende, die verschiedene Orte besuchen und den kürzesten Weg gehen möchten, müssen dieses Problem lösen können, daher der Name. Auch beim Postbeamten, der die Zeitungen austrägt, tritt dieses Problem auf.
Material	• Landkarte (z.B. Kopiervorlage) • Pinwandnadeln und entsprechende Pinwand, Styropor o.ä. • Schnur und Gummifaden • Messrad oder Schrittzähler und Kreide für draussen

Zeitdauer	Erklärungen plus Versuche und Messungen: Outdoor und drinnen je 15 Min.
Vorgehen	Das Traveling Salesman Problem ist verwandt mit dem Hamilton Kreis Problem (Ein Hamilton Kreis ist ein Pfad, der jeden Knoten genau einmal besucht). Den Kanten sind nun aber Gewichte zugewiesen. In der Entscheidungsvariante beantwortet man die Frage, ob in einem Graphen eine Tour existiert, die kürzer als k ist. Im Optimierungsproblem wird die kürzeste Travelling Salesman Tour gesucht. Eine Tour besucht jede Stadt genau einmal. Das Problem rund um diese Frage ist ein schwieriges Problem und NP-vollständig.

Die Schüler/innen versuchen sich selbst als Handlungsreisende:

In der Outdoor-Variante treten mehrere Teams auf dem Pausenplatz gegeneinander an. Jeder Schüler darf mit Kreide eine Stadt an einem beliebigen Ort auf den Boden malen. Die Teams suchen nun eine optimale Reiseroute. Welche Gruppe findet die kürzeste? Die zurückgelegte Strecke wird mit einem Messrad oder einem Schrittzähler bestimmt.

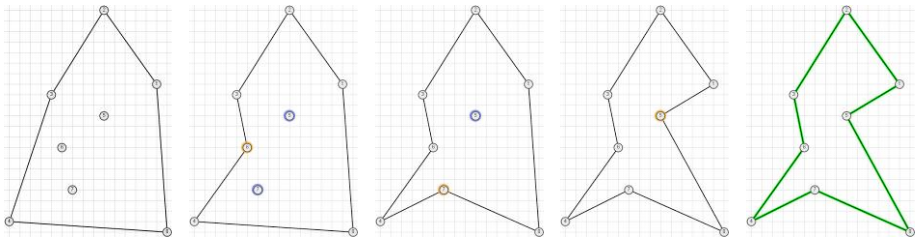
Sales(wo)men auf Pausenplatzreisen

Bei Regenwetter oder zur Vertiefung verschiedener Heuristiken kann der/die Handlungsreisende auch auf der Karte der Kopier-vorlage unterwegs sein.

Auf der Karte sind alle 15 Hauptstädte, die besucht werden sollen, mit einer Pinwandnadel versehen. Die Tour startet und endet in Amsterdam. Wer findet die kürzeste Route?

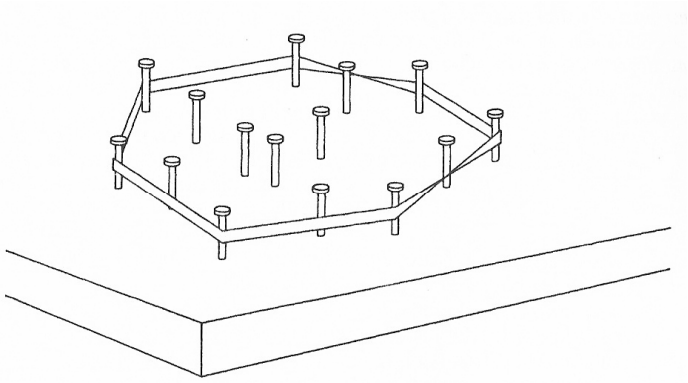
Lösung: Amsterdam – Brüssel – Luxemburg – Paris – London – Dublin – Lissabon – Madrid – Rom – Athen – Wien – Berlin –

	Helsinki – Stockholm – Kopenhagen – Amsterdam (Länge der Route: 10'494 km)
Varianten	Wir betrachten nun Algorithmen für das euklidische Traveling Salesman Problem. Dieses Problem ist gut untersucht und es existieren auch zahlreiche Heuristiken. Im euklidischen TSP Problem entspricht das Gewicht einer Kante seiner Länge. Die Nearest Neighbour Heuristik beginnt beim ersten Knoten und wählt immer den nächsten (dessen Verbindungskante das kleinste Gewicht hat) unbesuchten Knoten bis kein Knoten mehr übrig bleibt. Das geht natürlich sehr schnell, ist aber dementsprechend ungenau. Etwas besser schneidet die konvexe Hülle Heuristik ab. Sie erzeugt zuerst die konvexe Hülle aller Knoten. Dieser Algorithmus lässt sich am Fadenmodell demonstrieren. Die konvexe Hülle kann auch mit einem Gummiband anstatt einer Schnur erzeugt werden. Die folgenden Knoten lassen sich dann sehr einfach hinzufügen. Dann wird der Knoten ausgewählt, welcher der Tour am nächsten liegt und hinzugefügt. Der Algorithmus terminiert, wenn alle Knoten Teil der Tour sind.

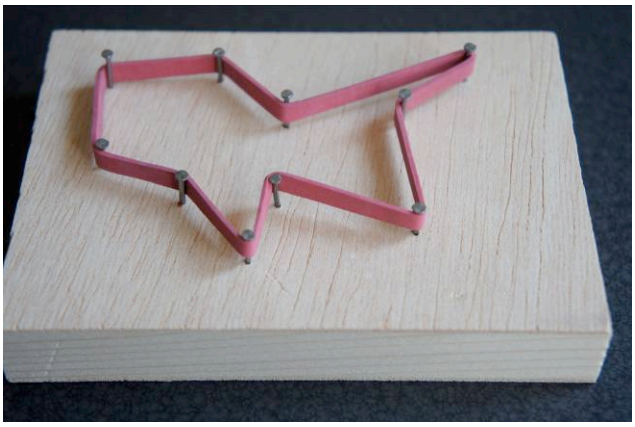


Quelle:
http://www.swisseduc.ch/informatik/theoretische_informatik/hard_problems/

Der Algorithmus der konvexen Hülle lässt sich auch sehr anschaulich mit einem Brett, Nägeln und einem Gummiband nachspielen.



Die konvexe Hülle wird mittels eines Gummibandes erzeugt



Diese „dehnt“ man zur kürzesten Tour mit obigem Algorithmus

Keiner zu klein, ein handlungsreisender Informatiker zu sein!

Die Greedy Heuristik wählt die Kante mit dem kleinsten Gewicht und fügt sie zur Tour hinzu, sofern die Kante keinen Zyklus erzeugt, bis kein Knoten mehr übrig bleibt.

Two-Optimization generiert zuerst eine zufällige Tour. Danach sucht sie zwei Kanten und tauscht deren Endknoten, falls dies die Tour verkleinert. Dies macht der Algorithmus, bis keine zwei solchen Kanten mehr gefunden werden.

Nochmals zur Erinnerung: das Problem des Handlungsreisenden ist NP-vollständig. Das bedeutet, es ist zwar lösbar, jedoch mit erheblichem Aufwand. Alle bekannten korrekten Algorithmen sind so langsam, dass sie schon bei Graphen mit mehreren hundert Knoten nicht mehr in nützlicher Zeit ein Resultat generieren. Ein Algorithmus, der alle Pfade durchsucht, müsste in einem Graphen mit n Knoten n! Touren durchlaufen (bzw. (n-1)! Möglichkeiten gegebenem Startknoten). Konkret gibt es für 5 Städte also 5! = 120 Möglichkeiten, für eine zusätzliche Stadt bereits 6! = 720 Touren und der Handlungsreisende, der unsere 15 Hauptstädte besuchen will, muss bereits 15! = 1'307'674'368'000 Routen berücksichtigen.

Um nur durch Ausprobieren die beste Tour durch die 30 Städte des obigen Länderpuzzles zu finden, müssten wir im Zweifelsfall ca $2.7 \cdot 10^{32}$ Touren berechnen. Nehmen wir an, die Berechnung einer einzelnen Rundreiselänge kann in der Zeit von 10^{-9} Sekunden erfolgen (also etwa im Gigahertz - Bereich). Dann benötigt das „brute force“ Verfahren $30! \cdot 10^{-9}$ Sekunden. Dies sind ca. $8 \cdot 10^{15}$ Jahre. Das Alter des Universums wird auf 10^{10} Jahre geschätzt. Deshalb braucht man die Heuristiken, die als Abschätzung oftmals genügen.

Weitere Ideen

Eine höhere Auflösung der Karte der Kopiervorlage
http://tashgoose.files.wordpress.com/2011/02/0_map_europe_political_2001_enlarged.jpg

Java Applet mit den 15 Hauptstädten dieses Experimentes
http://watchmaker.uncommons.org/examples/salesman.php

Schwierige Probleme in der Informatik auf swisseduc
http://www.swisseduc.ch/informatik/theoretische_informatik/hard_problems/

	GraphBench Lernumgebung für NP-Vollständigkeit http://www.swisseduc.ch/informatik/graphbench/

Quelle: http://tashgoose.files.wordpress.com/2011/02/0_map_europe_political_2001_enlarged.jpg