

Einführung in die lineare und kombinatorische Optimierung

Weihnachtsaufgabe

Sven-Maurice Althoff (FU 4745454)

Michael R. Jung (HU 502133)

Felix Völker (TU 331834)

20. Januar 2015

Alle Aufgaben wurden mit einer Breitensuche implementiert, da diese optimal für Graphen mit uniformen Kantenlängen ist, also das gleiche Ergebnis wie ein Kürzeste-Wege-Algorithmus liefert.

a)

Der Graph D_4 ist nicht zusammenhängend, er besitzt 205 Zusammenhangskomponenten. Die größten 5 Zusammenhangskomponenten besitzen folgende Kardinalität:

Komponente 1: 1195 Knoten

Komponente 2: 6 Knoten

Komponente 3: 5 Knoten

Komponente 4: 5 Knoten

Komponente 5: 5 Knoten

b)

Der größte Grad eines Knotens des Graphen D_4 ist 16. *Maus* ist das einzige Wort, dessen Knoten diesen Grad besitzt und besitzt daher die größte Nachbarschaft. Es existieren viele isolierte Knoten, meist die Knoten dessen Worte Symbole aus anderen Sprachen besitzen (bspw. Apostroph) oder eventuell Namen repräsentieren. Beispiele: *Ulks*, *Ryti*, *Rącz*, *Onyx*, *Ezio*, *Abbé*

c)

Es existiert nur eine Zusammenhangskomponente mit mindestens 20 Knoten und zwar diejenige, die den Knoten *Aale* enthält. Für diese Zusammenhangskomponente gilt, dass die längsten kürzesten Wege 21 Kanten besitzen, also gilt $L' = 21$. Beispiele sind:

Whip->Chip->Clip->Flip->Flop->Floß->Kloß->Klon->Klan->Kran->Gran->Gras->Grus->Gaus->Maus->Maas->Maat->Saat->Spat->Spot->Slot->Plot

Chic->Chip->Clip->Flip->Flop->Floß->Kloß->Klon->Klan->Kran->Gran->Gras->Grus->Gaus->Taus->Tals->Talk->Kalk->Kali->Kuli->Juli->Juni

d)

Da alle Bögen in D_4 uniforme Länge besitzen, ist das Ergebnis einer Breitensuche bis zum erstmaligen Auftreten des Knotens mit dem Nomen *Baum* äquivalent zur Lösung eines Kürzeste-Wege-Algorithmus. Das Ergebnis lautet:

Keks->Koks->Loks->Lots->Lote->Lobe->Labe->Laue->Baue->Baum