

informatiCup 2011 • Aufgabe 1

Einkaufsoptimierung

Einführung

Beim wöchentlichen Großeinkauf einer Familie kann viel Geld gespart werden, wenn zuvor die Artikelpreise verglichen und die Einkaufsrouten zu den Geschäften optimiert wird. Diese Aufgabe soll durch ein Programm unterstützt werden, in das die Preise der gewünschten Artikel für die einzelnen Geschäfte eingegeben werden.

Aufgabenstellung

Gegeben sind n Artikel a_1, a_2, \dots, a_n , die in k Geschäften g_1, g_2, \dots, g_k angeboten werden. Jeder Artikel a_i wird in mindestens einem Geschäft g_j angeboten, wobei die Artikelpreise in jedem Geschäft unterschiedlich sein können. Es wird angenommen, dass jeder Artikel, der in einem Geschäft angeboten wird, auch in der gewünschten Menge vorrätig ist. Jedes Geschäft ist vom Wohnort h (evtl. indirekt) erreichbar. Die Fahrtkosten zu und zwischen den Geschäften sind vorgegeben. Es ist ein Algorithmus zu entwickeln, der die Einkäufe der n Artikel so optimiert, dass die Gesamtkosten aus Fahrtkosten und Summe der Artikelpreise minimiert werden. Ermittelt werden sollen außerdem die Fahrtroute und die Einkaufslisten für die ausgewählten Geschäfte.

1. Runde

Bevor Sie die globale Optimierung durchführen, sollte der Suchraum bezüglich folgender Vereinfachungen untersucht werden:

- Es entstehen keine Fahrtkosten (z.B. Sozial/Behindertenkarte), d.h. Sie ermitteln Einkaufslisten für die absolut günstigsten Angebote.
- Alle Artikel werden überall angeboten und es soll nur in einem Geschäft eingekauft werden, d.h. es ist das Geschäft zu bestimmen, in dem unter Berücksichtigung der Fahrtkosten der Einkauf am günstigsten ist.
- Die Fahrtkosten zwischen den Geschäften sind größer als der Preisvorteil, d.h. es kann bei den nächstgelegenen Geschäften (nicht alle Artikel werden in allen Geschäften angeboten) eingekauft werden.
- Alle Artikel werden überall angeboten, d.h. es kann ohne Einschränkung auf die Verfügbarkeit der Artikel der Gesamtpreis (= Summe Artikelpreise + Fahrtkosten) optimiert werden.

Im letzten Schritt ist eine Gesamtoptimierung für den allgemeinen Fall zu erstellen.

Ihre Lösungen sollten jeweils an einem überschaubaren, jedoch nichttrivialen Beispiel (z.B. 6-10 Artikel und 3-4 Geschäfte) illustriert werden. Entwickeln Sie auch Testfälle von denen Sie annehmen, dass ihre Lösung im Vergleich zu anderen gut bzw. besser abschneidet.

2. Runde

Erstellen Sie eine graphische Benutzeroberfläche, mit der Sie Ihre Testfälle und das Demoszenario

visualisieren. Zeigen Sie in einer Graphik, wie Ihr Algorithmus das Einkaufsproblem schrittweise optimiert, indem für jede Iteration die momentanen Einkaufs- und Fahrtkosten dargestellt werden. Untersuchen Sie das Laufzeitverhalten Ihrer Lösung bezüglich der Artikelmenge n und der Anzahl k der Geschäfte und stellen Sie die Ergebnisse graphisch dar.

Außerdem: Erstellen Sie für Ihre Implementierung bitte eine Bedienungs- und Installationsanleitung. Dokumentieren Sie die von Ihnen getroffenen Entscheidungen bei der Auswahl verwendeter Algorithmen und Datenstrukturen und in der Software-Entwicklung.

Eingabeformat

Es werden kommaseparierte Textdateien zur Eingabe der Artikelpreise und der Fahrtkosten verwendet. Die Datei für die Artikelpreise hat folgendes Format:

```
<Anzahl Geschäfte>, <Anzahl Artikel><cr lf>
Artikel/Geschäft, Menge, <g1>, <g2>, ..., <gk><cr lf>
<a1>, m1, Preis(a1 in g1), Preis(a1 in g2), ..., Preis(a1 in gk) <cr lf>
<a2>, m2, Preis(a2 in g1), Preis(a2 in g2), ..., Preis(a2 in gk) <cr lf>
...
<an>, mn, Preis(an in g1), Preis(an in g2), ..., Preis(an in gk)<cr lf>
```

Legende: $\langle g_i \rangle$ Name des Geschäfts g_i , $\langle a_i \rangle$ Bezeichnung des Artikels a_i , m_i ist die Menge/Anzahl des zu kaufenden Artikels a_i , $\text{Preis}(a_i \text{ in } g_j)$ ist der Preis des Artikels a_i im Geschäft g_j , wenn kein Preis angegeben ist, wird der Artikel in diesem Geschäft nicht angeboten, $\langle \text{cr lf} \rangle$ sind die ASCII-Zeichen CR (13) und LF (10).

Die Datei für die Fahrtkosten beschreibt in Form einer Adjazenzmatrix die Fahrtkosten für jede mögliche Strecke.

```
Fahrtkosten, <g0>, <g1>, <g2>, ..., <gk><cr lf>
<g0>, 0, k(g0,g1), k(g0,g2), ..., k(g0,gk) <cr lf>
<g1>, k(g1,g0), 0, k(g1,g2), ..., k(g1,gk) <cr lf>
...
<gk>, k(gk,g0), k(gk,g1), ..., k(gk-1,gk), 0 <cr lf>
```

Legende: $\langle g_0 \rangle$ bezeichnet den Ausgangspunkt (Wohnung), $k(g_i, g_j)$ sind die Fahrtkosten zwischen g_i und g_j . Es wird von einem ungerichteten Graphen ausgegangen, d.h. Hin- und Rückweg kosten gleich viel, daher gilt $k(g_i, g_j) = k(g_j, g_i)$. Wenn keine Kosten angegeben sind, gibt es keine direkte Verbindung zwischen den Geschäften.

Beispiel 1: Mengen m_1 bis m_6 jeweils 1 (siehe Datei Artikelpreise.txt)

3			6		
Artikel\Geschäft	Menge	g_1	g_2	g_3	
a_1	1	14.95	16.99	11.50	
a_2	1	4.99	4.59	4.99	
a_3	1		12.90	11.99	
a_4	1	5.99	4.55	4.95	
a_5	1	19.90	19.95		
a_6	1		8.19	7.99	

Beispiel 2: Mengen m_1 bis m_7 jeweils 1 (siehe Datei Artikelpreise.txt)

3			7		
Artikel\Geschäft	Menge	g_1	g_2	g_3	
a_1	1	14.95	16.99	11.50	
a_2	1	4.99	4.59	4.99	
a_3	1		12.90	11.99	
a_4	1	5.99	4.55	4.95	
a_5	1	19.90	19.95		
a_6	1		8.19	7.99	
a_7	1	5.95	9.90		

Fahrtkosten (siehe Datei Fahrtkosten.txt)

Fahrtkosten	g_0	g_1	g_2	g_3
g_0	0	3	4	6
g_1	3	0	2	
g_2	4	2	0	3
g_3	6		3	0

Optimaler Einkauf für Beispiel 1

Route	Fahrtkosten	Einkaufsliste	Einkaufspreis	Gesamtkosten
$g_2 + g_3$	13	$g_2(a_2 + a_4 + a_5)$ $g_3(a_1 + a_3 + a_6)$	29,09 31,48	73,57

Optimaler Einkauf für Beispiel 2

Route	Fahrtkosten	Einkaufsliste	Einkaufspreis	Gesamtkosten
$g_1 + g_2$	9	$g_1(a_1 + a_5 + a_7)$ $g_2(a_3 + a_2 + a_4 + a_6)$	40,80 30,23	80,03