technische universität dortmund

Fakultät für Informatik
Lehrstuhl 11 / Algorithm Engineering

Prof. Dr. Petra Mutzel
Carsten Gutwenger

Wintersemester 2008/09

Algorithmen und Datenstrukturen
Übungsblatt 4

Ausgabe: 4. Dezember — Besprechung: 16./18. Dezember

Aufgabe 4.1 (Lineare Programmierung) [***Präsentation von Team: 4, 8***]

(a) [***Ausarbeitung von allen***]

Eine Whisky-Importgesellschaft hat zwar einen unbeschränkten Absatzmarkt für ihre Produkte, jedoch limitieren Importbeschränkungen den monatlichen Einkauf auf maximal

- 2000 Flaschen Macallan zu 35 Euro,
- 2500 Flaschen Cragganmore zu 25 Euro und
- 1200 Flaschen Bruichladdich zu 20 Euro.

Hieraus werden drei Mischungen A, B und C zu den Preisen 34 Euro, 28,50 Euro, bzw. 22,50 Euro hergestellt, die folgenden Anforderungen gerecht werden müssen:

- A enthält wenigstens 60% Macallan und höchstens 20% Bruichladdich.
- B enthält wenigstens 15% Macallan und höchstens 60% Bruichladdich.
- C enthält höchstens 50% Bruichladdich.

Welche Mischungen ergeben den größten Profit?

Formulieren Sie die Fragestellung als Lineares Optimierungsproblem.

Hinweis: Eine der Mischungen kommt von vornherein nicht in Betracht. Bezeichnen Sie die Mengen von Macallan, Cragganmore, bzw. Bruichladdich in der ersten Mischung mit x_1, x_2, bzw. x_3 und in der zweiten Mischung mit x_4, x_5, bzw. x_6.

(b) Lösen Sie das Lineare Optimierungsproblem aus Teil (a), indem Sie eine beliebige Implementierung des Simplex-Algorithmus verwenden. Welche Ecken werden dabei vom Simplex-Algorithmus betrachtet? Wie lautet die Lösung des LPs?

Aufgabe 4.2 (Lineare Programmierung) [***Präsentation von Team: 9***] [***Ausarbeitung von Team: 10***]

Das Polyeder $P(A, b)$ sei gegeben durch:

$$A = \begin{pmatrix} -1 & -1 \\ 1 & 0 \\ 0 & 1 \\ -1 & 0 \\ 0 & -1 \end{pmatrix}, \qquad b = \begin{pmatrix} -1 \\ 1 \\ 2 \\ 0 \\ 0 \end{pmatrix}$$

(a) Stellen Sie $P(A, b)$ graphisch dar und bestimmen Sie alle Ecken.

(b) Formen Sie $P(A, b)$ um in ein Polyeder der Form $A^{=}(\bar{A}, \bar{b})$, welches den Lösungsraum des dualen Problems beschreibt.

(c) Bestimmen Sie alle regulären (3,3)-Untermatrizen von $\bar{A}$.

(d) Berechnen Sie alle Basislösungen und ordnen Sie diese den Ecken zu. Welche Ecken bzw. Basislösungen sind degeneriert, welche sind zulässig?

Aufgabe 4.3 (Lock-free Datenstrukturen) [***Präsentation von Team: 1, 6, 7***]

Greifen mehrere, parallel ausgeführte Programmteile (*Threads*) gleichzeitig auf eine gemeinsame Datenstruktur zu, so muss diese häufig durch einen Lock (z.B. *Mutex*) geschützt werden, um Inkonsistenzen und Korruption der Datenstruktur zu vermeiden. Versucht ein Thread einen Lock zu erhalten, den bereits ein anderer Thread inne hat, so wird er vom Betriebssystem in einen Wartezustand versetzt, bis der Lock wieder freigegeben wird. Dadurch entsteht allerdings ein großer Overhead, der den Leistungsgewinn durch Parallelisierung zunichte machen kann, da das Warten eines Threads bedeutet, dass der entsprechende Prozessor nicht genutzt wird.

Eine Alternative stellen so genannte lock-free Datenstrukturen da. Diese erlauben den gleichzeitigen Zugriff, ohne dass die Verwendung eines Locks notwendig ist. Dies wird in der Regel durch Verwendung der atomaren CAS (*Compare-and-Swap*) Operation erreicht.

Stellen Sie eine Datenstruktur vor, die einen lock-free Stack realisiert. Dieser soll über die üblichen Operationen `push` und `pop` verfügen. Sie können sich an den folgenden Originalartikeln orientieren, oder eigene Ideen verwenden.

D. Hendler, N. Shavit, and L. Yerushalmi, *A scalable lock-free stack algorithm*, Proceedings of the sixteenth annual ACM symposium on Parallelism in algorithms and architectures, 2004, pages 206–215, `http://doi.acm.org/10.1145/1007912.1007944`

R. Colvin and L. Groves, *A scalable lock-free stack algorithm and its verification*, Proceedings of the Fifth IEEE International Conference on Software Engineering and Formal Methods, 2007, pages 339–348, `http://dx.doi.org/10.1109/SEFM.2007.2`