Prof. Dr. A. Goerdt
Julian Pape-Lange

TU Chemnitz
Wintersemester 2022/2023
25.01.2023

Theoretische Informatik I

13. Übung

Abgabe: Lösen Sie bitte die Aufgaben **2** und **4**. Ihre Lösungen geben Sie bitte entweder

- am 31.01.2023 während der Vorlesung oder
- bis zum 31.01.2023 um 9:00 Uhr per Mail
  an `julian.pape-lange@informatik.tu-chemnitz.de`
  mit *Betreff:* `TI1 Hausaufgaben`

ab.

1. Aufgabe:
Zeigen Sie, dass man in einer Liste mit n Zahlen die k-kleinste Zahl auch mit den Medianen aus jeweils 7 Elementen (statt der 5 Elemente, die in der Vorlesung verwendet wurden) in $\mathcal{O}(n)$ Zeit bestimmen kann.
Kann man diese Laufzeitkomplexität auch mit den Medianen aus jeweils 3 Elementen erreichen?

2. Aufgabe: 5P
Wir wenden den Davis-Putnam-Algorithmus (Backtracking-Algorithmus) aus der Vorlesung auf die Formel

$$F = (\neg A \vee B \vee C) \wedge (A \vee B \vee C) \wedge (A \vee B \vee \neg C)$$

an.

3. Aufgabe:
Die Formeln

$$F = (\neg A_1 \vee A_2) \wedge (A_2 \vee A_3) \wedge (\neg A_3 \vee A_4) \wedge (\neg A_4 \vee A_5) \wedge (\neg A_4 \vee \neg A_5)$$

und

$$G = (\neg A_1 \vee \neg A_2) \wedge (A_3 \vee A_4) \wedge (\neg A_3 \vee \neg A_4) \wedge (\neg A_3 \vee A_4) \wedge (A_3 \vee \neg A_4)$$

sind gegeben. Bestimmen Sie mit Hilfe des Davis-Putnam-Algorithmus, ob die Formeln erfüllbar sind. Nutzen Sie dabei die folgende *Teilmengenheuristik*: Wenn nach einer Belegung in der Restformel nur Klauseln der ursprünglichen Formel übrigbleiben, dann ist die Restformel genau dann erfüllbar, wenn auch die ursprüngliche Formel unerfüllbar ist. (Solche Belegungen wurden in der Vorlesung autark genannt)
Zeigen Sie, dass sich mit dieser Heuristik ein Polynomialzeitalgorithmus für 2-KNF ergibt.

4. Aufgabe: 5P
Übersetzen Sie folgende aussagenlogische Formel in eine *erfüllbarkeitsäquivalente* Formel in 3-KNF.

$$F = (X \vee \neg Y) \leftrightarrow (Y \wedge Z)$$

5. Aufgabe:
Ein *Eulerscher Kreis* in einem ungerichteten Graphen ist ein geschlossener Weg, in dem jede Kante des Graphen genau einmal vorkommt. Ein ungerichteter Graph G hat genau dann einen Eulerkreis, wenn G zusammenhängend ist und alle Knoten einen geraden Grad haben.

(a) Konstruieren Sie aus dem Beweis dieser Aussage einen Algorithmus, der zu einem gegebenen Graphen $G = (V, E)$ einen Eulerkreis ausgibt, falls ein solcher in G existiert.

(b) Um die Laufzeit $O(|V|+|E|)$ zu erreichen, muß der Algorithmus eine gefundene Kante in $O(1)$ aus dem Graphen löschen können. Warum ist dies mit der herkömmlichen Adjazenzlistendarstellung kaum möglich?

(c) Entwickeln Sie die Adjazenzliste zu einer Datenstruktur weiter, die es ermöglicht, den Algorithmus mit Laufzeit $O(|V| + |E|)$ zu implementieren.

(d) Geben Sie ein Verfahren an, wie die gegebene Adjazenzliste des Graphen in Ihre Datenstruktur umgewandelt werden kann. Beachten Sie, dass dafür nur Zeit $O(|V| + |E|)$ zur Verfügung steht.