

# Theoretische Informatik I

## 10. Übung

Geben Sie die Lösungen der Aufgaben 1 und 3 a) bitte bis zum 06.01.2006 bei Ihrem Übungsleiter ab oder senden Sie sie an [ti1-hausaufgaben@informatik.tu-chemnitz.de](mailto:ti1-hausaufgaben@informatik.tu-chemnitz.de).

1. Aufgabe:

Finden Sie ein möglichst einfaches Verfahren, um eine  $k$ -KNF in eine 3-KNF umzuwandeln.

2. Aufgabe:

Transformieren Sie die folgende Formel mit dem Polynomialzeitalgorithmus der Vorlesung in eine erfüllbarkeitsäquivalente 3-KNF.

$$(x \vee \neg y) \iff (y \wedge z)$$

Wenden Sie die Davis-Putnam-Prozedur

- a) ohne Verwendung der Heuristiken pure-literal und unit-clause
- b) mit den beiden Heuristiken

auf die erhaltene 3-KNF an. Zeichnen Sie jeweils den Aufrufbaum.

3. Aufgabe:

Lösen Sie folgende Rekursionsgleichungen.

a)

$$\begin{aligned} T(0) &= 2 \\ T(1) &= 10 \\ T(n) &= 10 \cdot T(n-1) - 16 \cdot T(n-2) \end{aligned}$$

b) Bei dieser Aufgabe genügt es, das Ergebnis in  $O$ - und  $\Omega$ -Notation anzugeben.

$$T(n) = \begin{cases} d & \text{für } n < n_0 \\ T(n-1) + T(n-2) & \text{für } n \geq n_0 \end{cases}$$

c)

$$T(0) = 0$$

$$T(1) = 1$$

$$T(n) = 2 \cdot T(n-1) - T(n-2)$$

#### 4. Aufgabe:

Betrachten Sie folgenden Algorithmus *DPT* für das Erfüllbarkeitsproblem von 2-KNF Formeln.

*DPT*(*F*)

1. Ist *F* eine leere Formel
2.     Ausgabe: erfüllbar
3. Enthält *F* eine leere Klausel
4.     Ausgabe: unerfüllbar
5. Wähle ein beliebiges Literal *l*, das in *F* vorkommt
6. *F'* = *setTrue*(*F*, *l*)
7. Falls *F'* ⊂ *F*
8.     *DPT*(*F'*)
9. Ansonsten
10.    *DPT*(*setTrue*(*F*, ¬*l*))

*setTrue*(*F*, *l*)

1. *F'* = *F*
2. Entferne aus *F'* alle Klauseln, die *l* enthalten
3. Entferne ¬*l* aus allen Klauseln von *F'*
4. Wenn es eine unit-clause *c*=(*l*) gibt
5.     *F'* = *setTrue*(*F'*, *l*)
6. return(*F'*)

Zeigen Sie, daß der Algorithmus korrekt ist und polynomielle Laufzeit hat.