

**Blatt 4**

Kai Großjohann, André Schaefer

Abgabe bis 20. Mai 2003

**Aufgabe 1:    Boolesche und Nicht-Boolesche Funktionen  
                 Funktionale Vollständigkeit**

- Definiere den Begriff *funktionale Vollständigkeit*. Verwende dazu deine eigenen Worte.
- Wie viele binäre Boolesche Funktionen gibt es? (Binär bedeutet zweistellig.) Begründe deine Antwort. (Eine mögliche Begründung wäre das Aufzählen aller Möglichkeiten.)
- Wir definieren folgende Boolesche Funktionen:

$$\overline{\wedge}(a, b) := \begin{cases} 0 & \text{falls } a = b = 1 \\ 1 & \text{sonst} \end{cases}$$
$$\overline{\vee}(a, b) := \begin{cases} 1 & \text{falls } a = b = 0 \\ 0 & \text{sonst} \end{cases}$$

Zeige, dass  $\{\overline{\wedge}\}$  funktional vollständig ist, indem du für alle binären Booleschen Funktionen logische Ausdrücke angibst, die nur  $\overline{\wedge}$  als Verknüpfung verwenden.

- Verwende die funktionale Vollständigkeit von  $\{\overline{\wedge}\}$ , um die funktionale Vollständigkeit von  $\{\overline{\vee}\}$  zu zeigen.
- Die Booleschen Funktionen sind nach George Boole benannt. Eine Funktion  $f: \{0, 1\}^n \rightarrow \{0, 1\}$  heißt  $n$ -stellige Boolesche Funktion. Weniger bekannt sind die Schaeferischen Funktionen: eine Funktion  $f: \{0, 1, 2\}^n \rightarrow \{0, 1, 2\}$  heißt  $n$ -stellige Schaeferische Funktion.  
Wie viele binäre Schaeferischen Funktionen gibt es? Begründe deine Antwort.
- Gib eine möglichst kleine Menge von funktional vollständigen Schaeferischen Funktionen an. Definiere die hierzu nötigen Schaeferischen Funktionen nach Bedarf. Beweise die funktionale Vollständigkeit der von dir gefundenen Menge.

10 Punkte

**Aufgabe 2:    Schaltungen**

- Welche Eigenschaften sind bei logischen Schaltungen wünschenswert? Begründe bei jeder Eigenschaft, warum diese wünschenswert ist.
- Was haben Schaltungen mit Graphen zu tun? (Beschreibe dazu die Komponenten, aus denen Schaltungen bestehen, und die Komponenten, aus denen Graphen bestehen, und setze sie jeweils in Beziehung.)

5 Punkte

### Aufgabe 3: Schaltgraphen

Gib einen Algorithmus an, der zu jeder Booleschen Funktion einen Graphen ausrechnet, der dieser Schaltung entspricht.

Was bedeutet es, „einen Graphen auszurechnen“? Nun,  $G = (V, E)$  heißt ja Graph über der Knotenmenge  $V$ , wenn  $V$  eine endliche Menge ist und  $E \subseteq V \times V$  gilt. Wenn man nun zu einer Booleschen Funktion  $f$  eine Knotenmenge  $V$  und eine zugehörige Kantenmenge  $E$  angeben kann, dann hat man „einen Graphen ausgerechnet“.

Weil  $\{\bar{\phantom{x}}\}$  funktional vollständig ist, wollen wir uns hier auf Boolesche Funktionen beschränken, die nur  $\bar{\phantom{x}}$  als Operator verwenden. (Der Operator  $\bar{\phantom{x}}$  kann natürlich mehrfach vorkommen! So wäre beispielsweise  $f(a, b, c) = a\bar{(b\bar{(c\bar{c}})})$  eine mögliche Schaltfunktion, und ich habe nur  $\bar{\phantom{x}}$  als Operator verwendet.

In dem Graphen stehen einige Knoten für die Inputs  $a, b, c$  der Funktion, und andere Knoten stehen für die  $\bar{\phantom{x}}$ -Operationen. Deswegen definieren wir immer zusätzlich eine sogenannte *Labelfunktion*  $L$  zu so einem Graphen, die zu jedem Knoten angibt, wofür der Knoten steht. Wenn  $L(v) = a$  gilt, dann steht der Knoten  $v$  für den Input  $a$ .

Es ist günstig, wie folgt vorzugehen. Zunächst betrachtet man alle Booleschen Funktionen, wo  $\bar{\phantom{x}}$  nur einmal vorkommt. (Das sind  $f_1(a) := a\bar{a}$  und  $f_2(a, b) := a\bar{b}$ .) Man rechnet für solche Funktionen die Graphen aus.

Anschließend betrachtet man Ausdrücke der Form  $X\bar{Y}$ , wo  $X$  und  $Y$  jeweils kompliziertere Ausdrücke sein können. Man nimmt an, dass der Graph  $G_X$  zu  $X$  und der Graph  $G_Y$  zu  $Y$  bereits bekannt seien. Dann konstruiert man aus den Graphen  $G_X$  und  $G_Y$  einen neuen Graph  $G_{X\bar{Y}}$ .

Der Algorithmus, den ich mir ausgedacht habe, würde *zum Beispiel* für die o. g. Funktion  $f(a, b, c) = a\bar{(b\bar{(c\bar{c}})})$  folgenden Graphen  $G = (V, E)$  und folgende Labelfunktion  $L$  ausrechnen:

$$\begin{aligned} V &= \{ 1, 2, 3, 4, 5, 6, 7 \} \\ E &= \{ (1, 7), (2, 6), (3, 5), (4, 5), (5, 6), (6, 7) \} \\ L(1) &= a \\ L(2) &= b \\ L(3) &= L(4) = c \\ L(5) &= L(6) = L(7) = \bar{\phantom{x}} \end{aligned}$$

Man beachte, dass die Aufgabe ist, einen Algorithmus anzugeben. Der Algorithmus soll für alle Fälle funktionieren, nicht nur für den hier angegebenen Fall. Es ist nicht erforderlich, dass der Algorithmus das gleiche Ergebnis produziert wie meiner, sondern nur, dass das Ergebnis „äquivalent“ ist (für geeignete Definition von Äquivalenz).

15 Punkte