

1. Präsenzübung „Algorithmen und Datenstrukturen“

Sommersemester 2009

1 Algorithmenbegriff

1.1 SimpleBoolExpression

Als einfache boolesche Ausdrücke (`SimpleBoolExpression`) werden im folgenden Ausdrücke bezeichnet, in denen nur die Werte `true` und `false`, die Operatoren `&` und `|` sowie Klammern vorkommen und die vollständig geklammert sind. Damit ist folgende Syntax gemeint:

```
SimpleBoolExpression =  
true  
| false  
| "(" SimpleBoolExpression "&" SimpleBoolExpression ")"  
| "(" SimpleBoolExpression "|" SimpleBoolExpression ")" .
```

Unter I bis V sind Verfahren zur Berechnung des Wertes einfacher boolescher Ausdrücke beschrieben. Geben Sie für jedes dieser Verfahren an

- ob es sich um einen Algorithmus handelt

und wenn ja, ob er

- deterministisch
- determiniert bzw.
- terminierend

ist. Begründen Sie Ihre Antworten.

I. Da einfache boolesche Ausdrücke ein Spezialfall von Java-Ausdrücken sind, kann der Wert mit Hilfe eines Java-Programms berechnet werden.

II. (Java-Verfahren):

1. Falls der Ausdruck die Form `true` hat, gib den Wert `true` zurück.
2. Falls der Ausdruck die Form `false` hat, gib den Wert `false` zurück.
3. Falls der Ausdruck die Form

```
"(" SimpleBoolExpression "&" SimpleBoolExpression ")"
```

hat, berechne zunächst durch rekursive Anwendung der Schritte 1. bis 4. den Wert des linken `SimpleBoolExpression`, dann analog den Wert des rechten `SimpleBoolExpression` und verknüpfe dann beide mit Hilfe der Wahrheitstabelle für `&`.

4. Falls der Ausdruck die Form

`"(" SimpleBoolExpression "|" SimpleBoolExpression ")"`

hat, verfare analog zu 3.

- III.
1. Suche im Ausdruck einen Teilausdruck der Form `"(" x "&" y ")"`, bei dem `x` und `y` konkrete Werte (also `true` oder `false`) sind.
 2. Überprüfe, ob es nur einen solchen Teilausdruck gibt. Wenn ja, muss Schritt 3 ausgeführt werden. Andernfalls entscheide zufällig, ob du Schritt 3 auf den bereits gefundenen Ausdruck anwendest oder zunächst mit Schritt 1 einen anderen solchen Ausdruck suchst (zufällig steht dabei für eine 50-50 Chance) und gehe dann zu Schritt 3.
 3. Ersetze den in Schritt 1 gefundenen Ausdruck durch seinen Wert laut Wahrheitstabelle.
 4. Wiederhole die Schritte 1 bis 3 so lange, bis nur noch `true` oder `false` übrig bleibt und gib diesen Wert zurück.
- IV. gleiches Verfahren wie III., jedoch werden als Eingaben `ExtendedSimpleExpression` betrachtet, welche zusätzlich Variablen und den Operator `=` enthalten dürfen (Syntax und Berechnung wie in Java). Ausgewertet werden können die unter III angegebenen sowie Ausdrücke der Formen (`a` ist eine Variable): `a` (Wert entspricht Wert von `a`), `"(" a = true ")"` und `"(" a = false ")"`.

Gehen Sie in Ihrer Antwort davon aus, dass die Überprüfung, ob ein Ausdruck eine bestimmte Form hat, sowie die Verknüpfung von Wahrheitswerten laut Tabelle ausführbare elementare Verarbeitungsschritte sind, und dass die Formulierungen ausreichend präzise sind.

Begründen Sie schlüssig. Falls einer der Algorithmen nicht determiniert ist, geben Sie bitte ein Gegenbeispiel an. Wenn sich Begründungen wiederholen, genügt ein kurzer Verweis.

1.2 Labyrinth

- a) Für die Suche nach dem Ausgang aus einem Labyrinth wird folgender Algorithmus verwendet:
- 1) Entscheide dich willkürlich für Schritt 2a) oder Schritt 2b).
 - 2a) Gehe in der einzig möglichen Richtung los.
 - 2b) Trinke zunächst etwas Quellwasser und gehe dann in der einzig möglichen Richtung los.
 - 3) Sobald du an eine Kreuzung kommst, gehe entlang des nächsten Weges im Uhrzeigersinn weiter (wenn du an das Ende einer Sackgasse kommst, gehe wieder zurück)
 - 4) Wiederhole Schritt 3 solange bis du den Ausgang erreichst.

Eingaben des Algorithmus sind das konkrete Labyrinth und ein Startpunkt (grundsätzlich am Ende einer Sackgasse gelegen). Ausgabe ist die Nummer des Ausgangs. Ist der Algorithmus

- deterministisch (im Ablauf)?
- determiniert (im Ergebnis)?
- terminierend?

Begründen Sie jeweils kurz.

- b) Beziehen Sie sich wieder auf das Problem der Suche nach dem Ausgang aus einem Labyrinth und skizzieren Sie für jede der Eigenschaften deterministisch, determiniert und terminierend einen Algorithmus mit der zu a) entgegengesetzten Eigenschaft. Wenn Sie also beispielsweise für a) zu der Lösung deterministisch, determiniert und nicht terminierend gekommen sind, so sollen Sie jetzt drei Algorithmen angeben: einen nichtdeterministischen, einen nichtdeterminierten und einen terminierenden. Alternativ dürfen Sie natürlich auch einen einzigen Algorithmus angeben, der alle drei Eigenschaften besitzt, aber das ist schwieriger.