

## Vorlesung “Compilerbau” WS 2011/2012

### 3. Übungsblatt

Abgabe: 16. November 2010 bis 10.45 Uhr

#### Aufgabe 1 (7 Punkte)

Beweisen oder widerlegen sie das folgende Sprachen von einem DEA erkannt werden können:

- Zeichenketten über  $\{a,b\}$  mit mehr b's als a's.
- Zeichenketten über  $\{p,a,l\}$  die palindrome sind, d.h. vorwärts und rückwärts gelesen ergeben sie das gleiche Wort.
- Menge der syntaktisch korrekten C++ Programme.

#### Aufgabe 2 (6 Punkte)

Geben Sie zu jeder der folgenden regulären Sprachen über dem Alphabet  $\{0, 1\}$  einen Deterministischen Endlichen Automaten an.

- $\{w \mid w \text{ beginnt mit } 1 \text{ und endet mit } 0\}$
- $\{w \mid w \text{ beginnt mit } 0 \text{ und hat eine ungerade Länge oder beginnt mit } 1 \text{ und hat eine gerade Länge}\}$
- $\{w \mid w \text{ enthält den Teilstring } 110 \text{ nicht}\}$
- $\{w \mid w \text{ enthält den Teilstring } 0101 \}$

#### Aufgabe 3 (3 Punkte)

Schreiben Sie reguläre Ausdrücke für folgende Sprachen:

- Alle Zeichenketten aus 0'en und 1'en, die nicht den Teilstring 011 enthalten.
- Kommentare, die aus einer in `/*` und `*/` eingeschlossenen Zeichenkette bestehen, ohne `*/` dazwischen - es sei denn, innerhalb von doppelten Anführungszeichen (d. h. `“*/”` ist möglich).

#### Aufgabe 4(4 Punkte)

Ein lineares Optimierungsproblem besteht aus einer zu maximierenden oder minimierenden linearen Zielfunktion, einem System von Gleichungs- bzw. Ungleichungsnebenbedingungen und unteren bzw. oberen Schranken für die Werte der (kontinuierlichen) Variablen. Zur Speicherung solcher Optimierungsprobleme verwendet man das sogenannte **LP-Format**. Eine Datei in diesem Format ist wie folgt aufgebaut (Leerzeilen werden ignoriert):

- **Problemname (optional)**  
Die erste Zeile kann den Namen des Problems enthalten. Sie beginnt dann mit `“\Problem name:”` gefolgt vom Namen des Problems.

```
\Problem name: afito.lp
```

- **Zielfunktion**

Zunächst gibt es eine Zeile mit dem Eintrag “Maximize”, “Minimize”, “Max” oder “Min” (bzw. auch “maximize”, etc.), die angibt, ob maximiert oder minimiert werden soll. Ab der nächsten Zeile folgt die Zielfunktion als Folge von Variablen mit Koeffizienten. Die Funktion beginnt mit einem Namen, der der Zielfunktion vorangestellt und mit einem Doppelpunkt abgeschlossen wird. Fehlt ein Koeffizient vor einer Variablen, so wird er als 1.0 angenommen. Es sind alle Variablenamen erlaubt (auch Zahlen!).

Minimize

COST: - 0.4 X02 - X14 - 0.6 Y123 - 0.48 X36 + 10 X39

- **Nebenbedingungen**

Die Spezifikation der Nebenbedingungen beginnt mit einer Zeile, die “Subject To” oder “s.t.” enthält. Dann folgen die Nebenbedingungen. Die Nebenbedingungen sind analog zur Zielfunktion aufgebaut. Sie enthalten zusätzlich eine Angabe über die Art der Nebenbedingung (“=” für “=”, “<=” für “≤” und “>=” für “≥”) und die rechte Seite der Ungleichung als numerische Konstante. Die linke Seite der Ungleichung (bzw. Gleichung) kann auch leer sein.

Subject To

R10: - 1.06 X01 + X04 = 0

X05: X01 <= 80

X21: - X02 + 1.4 X14 >= 0

10002A: = 0

- **Schranken (optional)**

Dieser Abschnitt beginnt mit einer Zeile “Bounds” oder “bounds”. Schranken für eine Variable können auf die folgenden drei Arten angegeben werden:

Bounds

DMBOSONT >= -7

DMBOSONT <= 0

-3 <= DMBURYVR <= 0

Variablen stehen also entweder auf der linken Seite oder in der Mitte. Variablen können auch auf einen festen Wert gesetzt werden:

ULWPO8 = 0

Ferner kann auch der Fall auftreten, dass eine Variable unbeschränkt ist. Dies wird durch das Schlüsselwort **Free** gekennzeichnet:

RGAP1 Free

Ist für eine Variable keine untere Schranke angegeben, so wird diese als 0 angenommen. Ist keine obere Schranke angegeben, so wird diese als Unendlich angenommen.

- **Dateiende (optional)**

Eine Zeile “End” oder “end” zeigt das Dateiende an.

Beschreiben Sie die Syntax einer Datei im LP-Format mit einer Grammatik in EBNF.