

2. Übung zu „Grundlagen des Compilerbaus“, WS 2007/08

Abgabe der Aufgaben: Mi, 31.Oktober 2007 (vor der Vorlesung)

Besprechung der Aufg.: Fr, 2.November 2007 in der Übung

Aufgaben

2.1 DFA

| |
|----------|
| 5 Punkte |
|----------|

Gegeben seien die folgenden Grunddefinitionen zur Behandlung von deterministischen endlichen Automaten in Haskell:

```

type Sigma      = Char
type Alphabet   = [Sigma]

type Word       = String

type Delta state = state -> Sigma -> state
type DFA state = ([state],      -- Liste der Zustände
                  Alphabet,     -- Alphabet
                  (Delta state), -- Transitionsfunktion
                  state,         -- Anfangszustand
                  [state])      -- Menge der Endzustände

```

- (a) Definieren Sie einen DFA `bspDFA :: DFA Int`, der vorzeichenlose Dezimalzahlen, die durch den regulären Ausdruck

$$(0 \mid (1 - 9)(0 - 9)^*).(0 - 9)^+$$

definiert sind, erkennt.

- (b) Schreiben Sie eine Funktion

```
checkWord :: Eq state => (DFA state) -> Word -> Bool
```

die zu einem gegebenen DFA und einem Wort überprüft, ob das Wort von dem Automaten erkannt wird. Testen Sie Ihre Implementierung an dem Beispielautomaten.

2.2 Durchlauf durch einen Backtrack-DFA

3 Punkte

Gegeben seien die regulären Ausdrücke

$$\alpha_1 = ab$$

$$\alpha_2 = ac$$

$$\alpha_3 = (ab)^*c$$

- Geben Sie einen DFA zu jedem Ausdruck an und konstruieren Sie den Produktautomaten für die lexikalische Analyse per Backtrack-DFA.
- Geben Sie die Zustandsfolgen des Backtrack-DFA während der Analyse des Eingabestrings `ababacca` an.

2.3 Lexikalische Analyse

4 Punkte

- Geben Sie (über einem geeigneten Alphabet) reguläre Definitionen für die folgenden Symbolklassen an. Sie können zur Übersicht Hilfsdefinitionen, Aufzählungen und Negation verwenden.

Beispiel: `[1-7]`=Ziffer 1,2,3,4,5,6 oder 7. `[^123]`= bel. Zeichen *außer* 1,2,3.

- **AttValue:** Eine von " umschlossene Zeichenkette ohne < und &
- **CharData:** Eine nichtleere Zeichenkette ohne < und &
- **Comment:** eine von <!-- und --> umschlossene Zeichenkette, welche *kein -- enthält*
- **ERef:** Das Zeichen &, gefolgt von einem Bezeichner, danach das Zeichen ;

Bezeichner beginnen mit einem Buchstaben und enthalten nur Buchstaben und Ziffern.

- Für die in a) definierten Symbolklassen (in der angegebenen Reihenfolge) werde ein Scanner erstellt. Welches Resultat liefert der Scanner für die folgende Eingabe? (begründen Sie Ihre Antwort)

'In <!-- diesem Spezial-> XML hat "---->"&bes;ondere" Bedeutung-->'