

Lösningar till KS i Programmeringsparadigm, del 3: syntax
2012-11-02

1. (2 p) Numrera stegen i parsningsprocessen kronologiskt från 1 till 5 (vartannat steg beskriver dataform och vartannat steg beskriver en process).

5 Syntaxträd

1 Tecken

3 Slutsymboler

2 Lexikal analys

4 Syntaktisk analys

2. (3 p)

- a) Vilken mängd språk kan beskrivas med reguljära uttryck?

Reguljära språk

- b) Vilken mängd språk kan kännas igen av DFA, deterministiska ändliga automater?

Reguljära språk

- c) Vilken mängd språk kan kännas igen av NPDA, ickedeterministiska ändliga automater med stack?

Kontextfria språk

3. (3 p) Ett visst banksystem har följande krav på lösenord:

Lösenordet ska bestå av små bokstäver a-z och siffror. Minst en bokstav och en siffra måste ingå i lösenordet.

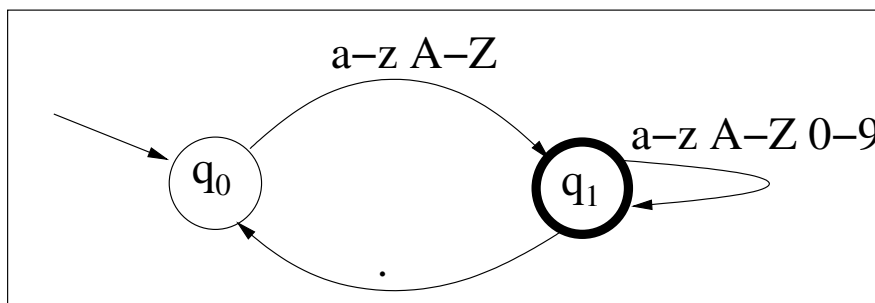
Konstruera ett Regex som beskriver lösenord enligt den definitionen.

$[a-z]^+[0-9][a-zA-Z0-9]^* \mid [0-9]^+[a-z][a-zA-Z0-9]^*$

4. (3 p) En lärobok rekommenderar att namn på package i Java följer denna definition:

$[a-zA-Z][a-zA-Z0-9]^*(\.[a-zA-Z][a-zA-Z0-9]^*)^*$

Rita upp en ändlig automat som känner igen detta språk.



5. (2 p) Språken $\{0^n, n > 0\}$ och $\{1^n, n > 0\}$ är reguljära. Konkateringen av två reguljära språk är fortfarande reguljärt.

Förklara varför detta *inte* innebär att språket $\{0^n 1^n, n > 0\}$ är reguljärt.

Det är inte nödvändigtvis samma n i 0^n och 1^n , så $0^n 1^n$ är inte konkateringen av språken.

6. (3 p) Skriv om följande grammatik så att den blir LL(1). Startsymbolen är *start* och slutsymboler står med fetstil.

$$\begin{array}{lll} start & \rightarrow & x \mid y \\ x & \rightarrow & \mathbf{A} \, x \mid \mathbf{B} \\ y & \rightarrow & \mathbf{AC} \end{array}$$

Omskriven grammatik:

$$\begin{array}{lll} start & \rightarrow & \mathbf{A} \text{ } rest \mid \mathbf{B} \\ rest & \rightarrow & x \mid \mathbf{C} \\ x & \rightarrow & \mathbf{A} \text{ } x \mid \mathbf{B} \end{array}$$

7. (4 p) Skriv en BNF-grammatik för chokladkaksbeskrivningar av typen
MINT-, RUSSIN-, NÖT- & MANDELCHOKLAD

Raden avslutas med **CHOKLAD** och det kan ingå valfritt antal smaksättare, även bara en: **MINTCHOKLAD**, eller ingen: **CHOKLAD**. Slutsymboler är **SMAKSÄTTARE**, **CHOKLAD**, **BINDESTRECK**, **KOMMA**, **OCH**. Strunta i blanktecknen i chokladkaksbeskrivningarna. Startsymbolen är *kaka*. Rita också ett syntaxträd för det första exemplet ovan.

```
kaka ::= CHOKLAD
      | SMAKSÄTTARE CHOKLAD
      | lista OCH SMAKSÄTTARE CHOKLAD
lista ::= SMAKSÄTTARE BINDESTRECK KOMMA lista
        | SMAKSÄTTARE BINDESTRECK
```

Syntaxträd med förkortade namn på slutsymbolerna:

