Formelle Språk

La $A \neq \emptyset$ være en endelig mengde.
Vi har definert A^*, mengden av endelige ord eller strenger over A.

Eks: $abbcca \in \{a,b,c\}^*$

Et formelt språk over A er da en delmengde av A^*.

Denne uken skal vi definere en type formelle språk som heter regulære språk.

La s, t være to strenger.
Da er st konkatineringen av de to.

Eks: La $s = ab$, $t = ba$, da er

$$st = abba$$
$$ts = baab$$

La A være et alfabet og B, C språk over A. Vi definerer konkatineringen av B og C som følger:

$$BC = \{st \in A^* \mid s \in B, t \in C\}$$

Eks: La $A = \{0, 1\}$. La $B = \{0, 00, 000, \ldots\}$ og $C = \{1, 11, 111, \ldots\}$

Da er $BC = \left\{ \begin{array}{l} 01, 011, 0111, 01111, \ldots \\ 001, 0011, 00111, \ldots \\ 0001, 00011, 000111, \ldots \\ \vdots \end{array} \right\}$

BC er altså alle strenger som kan skrives som en konkatinering av ting i B og ting i C.

Merk at unionen av to språk også er et språk:

$$B \cup C = \{0, 1, 00, 11, 000, 111, \ldots\}$$

La $B \subseteq A^*$ være et vilkårlig språk, og la $n \geq 1$ være et naturlig tall.

$$B^n = \{s_1 s_2 s_3 \ldots s_n \in A^* \mid \forall i \in \{1, 2, \ldots, n\} \, s_i \in B\}$$

Eks: La $A = \{0, 1\}$ og $B = \{00, 11\}$

Da er $B^2 = \{0000, 0011, 1100, 1111\}$

Vi definerer $B^0 = \{\Lambda\}$

Kleene-tillukning

La A være et alfabet. Vi har definert A^* induktivt.

Merk at $A \subseteq A^*$ også er et språk.

Merk også at $A^0 \cup A^1 \cup A^2 \cup A^3 \cup \dots = A^*$.

<u>Def</u> Kleene-tillukningen av et språk $B \subseteq A^*$ er definert til å være

$$B^0 \cup B^1 \cup B^2 \cup B^3 \cup \dots = \bigcup_{i \in \mathbb{N}} B^i$$

<u>Eks</u>: $\{0,1\}^* = \{\Lambda, 1, 00, 100, 001, 1001, \dots\}$

<u>Fakta</u> La $B \subseteq A^*$. Da er $(B^*)^* = B^*$

Bevis: $B^* = (B^*)^1 \subseteq (B^*)^*$

La nå $s \in (B^*)^*$. $\exists n$ slik at $s \in (B^*)^n$.

Da er $s = s_1 s_2 s_3 \dots s_n$ der $s_i \in B^{k_i}$.

Da er $s \in B^{(k_1+k_2+k_3+\dots+k_n)}$

Så $s \in B^*$ som ønsket. $\blacksquare$

<u>Eksempel:</u> La $B = \{0,1\}$.

Da er $B^* = \{\lambda, 0, 1, 00, 01, \dots\}$

$(B^*)^*$ inneholder da for eksempel

$$s = (00)(011)(\lambda)(1)(00)$$

$$\in B^*$$

men vi kan skrive s med "kun" bits som bokstaver:

$$s = 00011100$$

$$\in B^*.$$

Regulære språk

Vi har nå maskineriet til å definere regulære språk.

La A være et alfabet (endelig, ikke-tom mengde). Merk at mengden av alle språk over A er lik $\mathcal{P}(A^*)$. Mengden regulære språk er definert induktivt som følger:

Basis: $\{\Lambda\}$ er et regulært språk, og $\{a\}$ er et regulært språk for hver $a \in A$.

Ind. Step: Hvis R er et regulært språk er R^* et regulært språk. Hvis L, R er to regulære språk er $L \cup R$ og LR regulære.

Mengden av regulære språk over A er altså tillukningen av basis mengden under operasjonene

$$R \mapsto R^*$$

$$L, R \mapsto L \cup R$$

$$L, R \mapsto LR$$

Eks La $A = \{0,1\}$. Delmengden av A^* med kun ord uten "1" er da et regulært språk: $\{0\}$ er regulært (basis) og da er $\{0\}^*$ regulært.

Eks La A være et alfabet. A er alltid regulært som språk over seg selv: Vi kan skrive

$$A = \{a_1\} \cup \{a_2\} \cup \dots \cup \{a_n\}$$

for alle bokstaver i A. Ind. steg lar oss ta union.

<u>Eks</u> La $A = \{0, 1\}$. La $S \subseteq A^*$ være språket av ord med partalls lengde. Da er S regulært:

$$S = (AA)^*$$

Konkatinering

Regulært som over.

Kleene-tillukning av regulært språk er regulært.

Regulære uttrykk

Regulære uttrykk gir oss en praktisk måte å beskrive regulære språk på.

I andre kontekster brukes regulære uttrykk (ofte "regex") for å søke i tekst.

Def La A være et alfabet.

De er regulære uttrykk over A definert induktivt:

Basis: Λ er et regulært uttrykk, og a er et regulært uttrykk for alle $a \in A$.

Ind. Steg:

$$R \mapsto R^*$$

$$R, L \mapsto (R \mid L)$$

$$R, L \mapsto (RL)$$

Er de gyldige operasjonene vi har.

Eks la $A = \{a\, b\, c\}$

Da er $(ab)^*$ et regulært uttrykk.

Regulære uttrykk beskriver regulære språk rekursivt. Vi tolker et uttrykk R som et språk $L(R)$ definert rekursivt:

__Basis__: Uttrykket Λ gir språket $\{\Lambda\}$, Uttrykket x for en bokstav $x \in A$ gir språket $\{x\}$

Så $L(\Lambda) = \{\Lambda\}$ og $L(x) = \{x\}$.

Videre definerer vi

$$L(R^*) = L(R)^*$$

$$L((R_1 R_2)) = L(R_1) L(R_2)$$

$$L((R_1 | R_2)) = L(R_1) \cup L(R_2)$$

Om parenteser:

Merk $L(((R_1|R_2)|R_3))$

$= L((R_1|(R_2|R_3)))$

Så vi kan droppe parenteser mellom flere "|" (leses "eller")

Samme med uttrykkene

$(R_1(R_2R_3))$ og $((R_1R_2)R_3)$

Vi skriver heller

$(R_1R_2R_3)$

Vi dropper også ofte de ytre parentesene.

Eks Språket av bitstrenger som begynner på 1 og slutter på 0 beskrives av det regulære uttrykket

$$1(0|1)^*0$$

Merk: Alle språk som beskrives av regulære uttrykk er regulære.

Kan raskt sjekkes med strukturell induksjon.