Plan

- Hvad er Regularitet og Automater
- Praktiske oplysninger om kurset
- **Regulære udtryk**
- Induktionsbevis
- Frokost
- Endelige automater
- Skelnelighed, Produktkonstruktion
- Præsentation af Java projekt

Alfabeter, strenge og sprog

- Et ***alfabet*** Σ er en endelig mængde (af tegn/symboler)
  - eks.: $\Sigma=\{a,b,c\}$
- En ***streng*** x er en endelig sekvens af tegn fra alfabetet
  - eks.: $x=abba$
  - Λ repræsenterer *den tomme streng* (strengen af længde 0), $\Lambda \notin \Sigma$
- Et ***sprog*** L er en (vilkårlig) mængde af strenge
  - eks.: $L=\{\Lambda,cab,abba\}$
- Σ^* er ***mængden af alle strenge over*** Σ
  - dvs. $L\subseteq\Sigma^*$ hvis L er et sprog over Σ
  - eks.: hvis $\Sigma=\{a,b,c\}$ så er $\Sigma^*=\{\Lambda,a,b,c,aa,ab,ac,aaa,aab,...\}$

Konkatenering af strenge

- Hvis $x,y \in \Sigma^*$, så er $x \cdot y$ (*konkateneringen* af x og y) den streng, der fremkommer ved at sætte tegnene i x før tegnene i y

- Eks.: hvis $x=abb$ og $y=a$, så er
  - $x \cdot y = abba$
  - $y \cdot x = aabb$
- Bemærk: $x \cdot \Lambda = \Lambda \cdot x = x$ for alle x

- $x \cdot y$ skrives ofte xy (uden "·")

Konkatenering af sprog

- Hvis $L_1, L_2 \subseteq \Sigma^*$, så er $L_1 \cdot L_2$ (*konkateneringen* af L_1 og L_2) defineret ved

$$L_1 \cdot L_2 = \{x \cdot y \mid x \in L_1 \wedge y \in L_2\}$$

- Eks.: Hvis $\Sigma = \{0,1,2,a,b,c\}$ og
  - $L_1 = \{\Lambda, 10, 212\}$
  - $L_2 = \{cab, abba\}$

  så er $L_1 \cdot L_2 = \{cab, 10cab, 212cab, abba, 10abba, 212abba\}$
- Bemærk:
  - $L \cdot \{\Lambda\} = \{\Lambda\} \cdot L = L$ for alle L
  - $L \cdot \varnothing = \varnothing \cdot L = \varnothing$ for alle L
- $L_1 \cdot L_2$ skrives ofte $L_1 L_2$ (uden "$\cdot$")

Kleene stjerne

- $L^k = \underbrace{LL\cdots L}$
  konkatenering af k forekomster af L
- $L^0 = \{\Lambda\}$
- $L^* = \bigcup_{i=0\ldots\infty} L^i$ (Kleene stjerne af L)
- $L^+ = L^*L$

Rekursive definitioner

- En definition er *rekursiv*, hvis den refererer til sig selv

- Eks.: *Fibonacci* $f\colon \mathbb{N} \to \mathbb{N}$

$$f(n) = \begin{cases} 1 & \text{hvis } n=0 \text{ eller } n=1 \\ f(n-1)+f(n-2) & \text{ellers} \end{cases}$$

- Enhver selv-reference skal referere til noget ”mindre” og føre til endeligt mange selv-referencer

En rekursiv definition af strenge

- x er en ***streng*** over alfabetet Σ, dvs. $x \in \Sigma^*$, hvis
  - $x = \Lambda$, eller
  - $x = y \cdot a$ hvor $y \in \Sigma^*$ og $a \in \Sigma$

  (underforstået: Σ^* er den *mindste* mængde, der opfylder dette)

- Eksempel:
  $abc = (((\Lambda \cdot a) \cdot b) \cdot c) \in \Sigma^*$ (hvor $\Sigma = \{a, b, c, d\}$)

Syntaks af regulære udtryk

Mængden R af ***regulære udtryk*** over Σ er den mindste mængde, der indeholder følgende:

- $\boldsymbol{\emptyset}$
- $\boldsymbol{\Lambda}$
- $\boldsymbol{a}$ for hver $a \in \Sigma$
- $(\boldsymbol{r_1} + \boldsymbol{r_2})$ hvor $\boldsymbol{r_1}, \boldsymbol{r_2} \in R$
- $(\boldsymbol{r_1} \boldsymbol{r_2})$ hvor $\boldsymbol{r_1}, \boldsymbol{r_2} \in R$
- $(\boldsymbol{r}^*)$ hvor $\boldsymbol{r} \in R$

Semantik af regulære udtryk

Sproget $L(r)$ af et regulært udtryk r defineres i strukturen af r:

- $L(\boldsymbol{\emptyset}) = \emptyset$
- $L(\boldsymbol{\Lambda}) = \{\Lambda\}$
- $L(\boldsymbol{a}) = \{a\}$
- $L(\ \mathbf{(}\boldsymbol{r}_1\mathbf{+}\boldsymbol{r}_2\mathbf{)}\ ) = L(\boldsymbol{r}_1) \cup L(\boldsymbol{r}_2)$
- $L(\ \mathbf{(}\boldsymbol{r}_1\boldsymbol{r}_2\mathbf{)}\ ) = L(\boldsymbol{r}_1)L(\boldsymbol{r}_2)$
- $L(\ \mathbf{(}\boldsymbol{r}^{\mathbf{*}}\mathbf{)}\ ) = (L(\boldsymbol{r}))^*$

Regulære sprog

Definition:

Et sprog S er ***regulært*** hvis og kun hvis der eksisterer et regulært udtryk r hvor $L(r)=S$

Parenteser i regulære udtryk

- Forening og konkatenering er **associative**, så vi vælger at tillade f.eks.
  - at **(*a*+(*b*+*c*))** kan skrives ***a*+*b*+*c***
  - at **(*a*(*bc*))** kan skrives ***abc***
- Vi definerer **præcedens** for operatorerne:
  - * binder stærkest
  - konkatenering binder middel
  - **+** binder svagest
  - eks.: **(*a*+((*b**)*c*))** kan skrives ***a*+*b***c***

Eksempel

- Betragt følgende regulære udtryk r over alfabetet {0,1}:

$$r = (1+\Lambda)001$$

- På grund af parentesreglerne er dette det samme som

$$r = ((((1+\Lambda)0)0)1)$$

- Så sproget for r er

$$L(r) = ((((\{1\} \cup \{\Lambda\})\{0\})\{0\})\{1\})$$
$$= \{1001, 001\}$$

Quiz!

1. Hvad betyder {a,bc}* ?

3. Hvad er betingelsen for at et sprog *S* er *regulært*?

Øvelser

- [Martin] Opg. 3.2
- [Martin] Opg. 3.9 (a-e)
- [Martin] Opg. 3.10 (a-b)