

# Begreper idag

<p><b>Strenger</b> over alfabet A Den tomme strengen <math>\Lambda</math> Konkatenering av strenger</p>	<p><b>Tuppel</b>/sekvens vs. mengde Kartesisk produkt av mengder Aritet av <b>relasjon</b></p>
<p><b>Språk</b> over alfabet A Produkt LM av språk L og M <math>L^*</math> <math>L^+</math> <math>L^0, L^1, L^2, L^3, \text{ etc.}</math></p>	<p><b>Funksjoner</b> Argument, verdi Domene, kodomene Aritet av funksjon Funksjoner vs. relasjoner Eksempel: Sannhetsfunksjoner</p>

# Mengder versus sekvenser

$$\{2,5,4,7,7,10,9,12\} = \{2,4,5,7,7,9,10,12\} = \{2,4,5,7,9,10,12\}$$

$$(2,5,4,7,7,10,9,12) \neq (2,4,5,7,7,9,10,12) \neq (2,4,5,7,9,10,12)$$

Altså: Antall og plassering teller

Sekvens av lengde 2 kalles (ordnet) par

(2,5) (5,2) (2,2)

# Begreper idag

<p><b>Strenger</b> over alfabet <math>A</math></p> <p>Den tomme strengen <math>\Lambda</math></p> <p>Konkatenering av strenger</p>	<p><b>Tuppel</b>/sekvens vs. mengde</p> <p>Kartesisk produkt av mengder</p> <p>Aritet av <b>relasjon</b></p>
<p><b>Språk</b> over alfabet <math>A</math></p> <p>Produkt <math>LM</math> av språk <math>L</math> og <math>M</math></p> <p><math>L^*</math></p> <p><math>L^+</math></p> <p><math>L^0, L^1, L^2, L^3, \text{ etc.}</math></p>	<p><b>Funksjoner</b></p> <p>Argument, verdi</p> <p>Domene, kodomene</p> <p>Aritet av funksjon</p> <p>Funksjoner vs. relasjoner</p> <p>Eksempel: Sannhetsfunksjoner</p>

# Kartesisk produkt, $\times$

$$A \times B = \{(x,y) \mid x \in A \ \& \ y \in B\}$$

(dette er definisjonen av kartesisk produkt)

## Eksempler:

$$\{1,3,6,8\} \times \{1,4\} = \{(1,1),(1,4),(3,1),(3,4),(6,1),(6,4),(8,1),(8,4)\}$$

# Begreper idag

<p><b>Streng</b> over alfabet A Den tomme strengen <math>\Lambda</math> Konkatenering av strenger</p>	<p><b>Tuppel</b>/sekvens vs. mengde Kartesisisk produkt av mengder Aritet av <b>relasjon</b></p>
<p><b>Språk</b> over alfabet A Produkt LM av språk L og M <math>L^*</math> <math>L^+</math> <math>L^0, L^1, L^2, L^3, \text{ etc.}</math></p>	<p><b>Funksjoner</b> Argument, verdi Domene, kodomene Aritet av funksjon Funksjoner vs. relasjoner Eksempel: Sannhetsfunksjoner</p>

# Eksempel: Alfabet $A = \{a,b,c\}$

Fra dette kan vi lage strengene

$\Lambda$

a b c

aa ab ac ba bb bc ca cb cc

aaa aab aac aba abb abc aca acb acc baa bab bac

aaaa aaab aaac aaba aabb aabc aaca acb aacc osv.

aaaaa aaaab aaaac aaaba aaabb aaabc aaaca aabaa

aaaaaa aaaaaab aaaaaac aaaaaba aaaaabb aaaaabc ;



# Konkatenering av strenger

$$ab \cdot bc = abbc$$

$$abbc \cdot \Lambda = abb \cdot c = ab \cdot bc = abb \cdot c = abbc \cdot \Lambda = abbc$$

# Begreper idag

<p><b>Strenger</b> over alfabet <math>A</math></p> <p>Den tomme strengen <math>\Lambda</math></p> <p>Konkatenering av strenger</p>	<p><b>Tuppel</b>/sekvens vs. mengde</p> <p>Kartesisk produkt av mengder</p> <p>Aritet av <b>relasjon</b></p>
<p><b>Språk</b> over alfabet <math>A</math></p> <p>Produkt <math>LM</math> av språk <math>L</math> og <math>M</math></p> <p><math>L^*</math></p> <p><math>L^+</math></p> <p><math>L^0, L^1, L^2, L^3, \text{ etc.}</math></p>	<p><b>Funksjoner</b></p> <p>Argument, verdi</p> <p>Domene, kodomene</p> <p>Aritet av funksjon</p> <p>Funksjoner vs. relasjoner</p> <p>Eksempel: Sannhetsfunksjoner</p>



# WFF – Well formed formula

Streng av utsagnsvariabler (P,Q,R...), sannhetssymboler, konnektiver og parenteser, bygd opp etter følgende *induktive* regler:

- ❖ true, false, P,Q,R... er wff'er
  - ❖ Hvis A er en wff, er  $\neg A$  og (A) wff'er
  - ❖ Hvis A og B er wff'er, er også  
 $A \rightarrow B$ ,  $A \wedge B$  og  $A \vee B$  wff'er
- Noe er en wff bare hvis det følger av reglene over

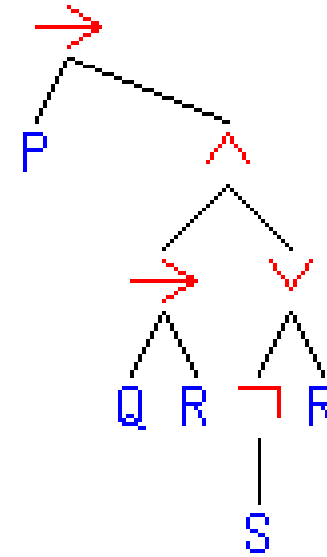
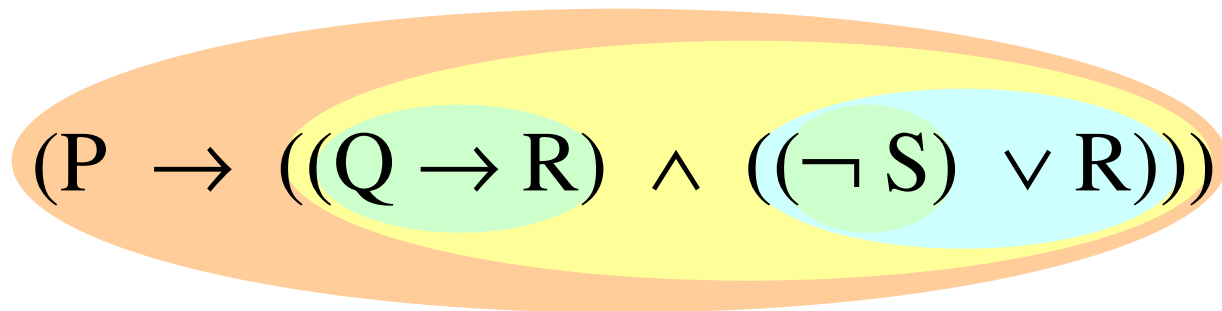
# Velformet formel (vff)

... sier man gjerne på norsk. I praksis blir dette tungvint, og vi/jeg kommer ofte til å si *formel* eller *utsagn* når vi mener vff/wff.

# Eksempler på wff'er

- $P$
- $Q$
- $R$
- true
- false
- $P \wedge Q$
- $(P \wedge Q)$
- $Q \vee R$
- $P \wedge \neg(Q \vee R)$
- $(P \wedge \neg Q) \vee R$

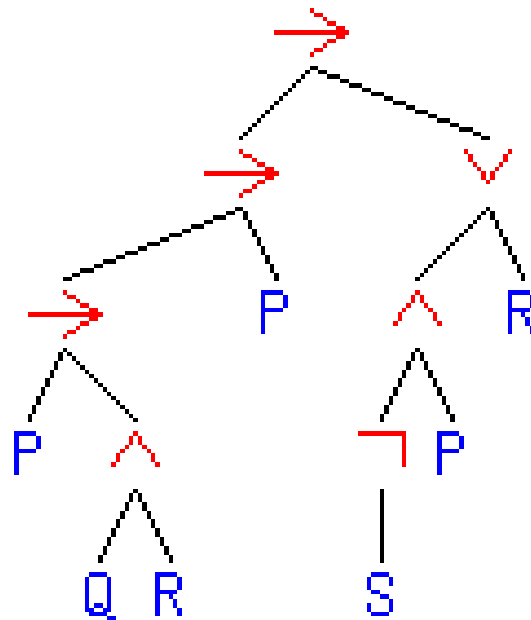
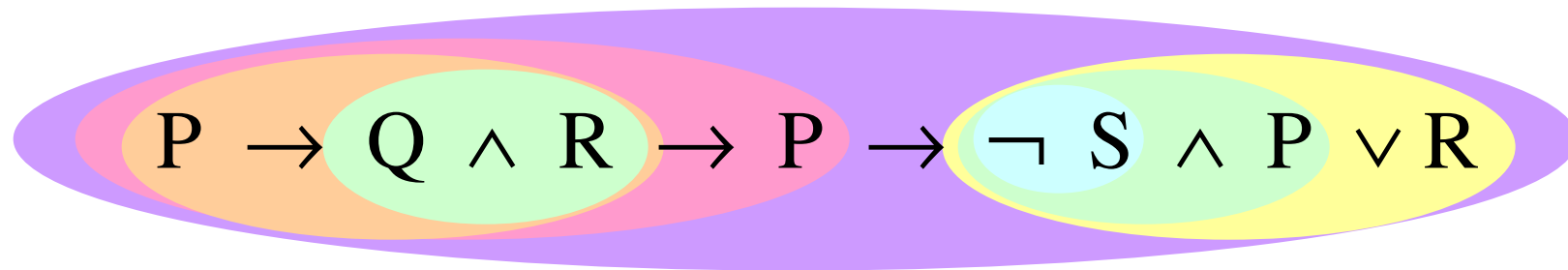
# Syntakstre



Et syntakstre viser strukturen i en wff,  
dvs. historien om hvordan wff'en ble bygd opp

(Treet over laget jeg hos [Gateway to logic](#))

# Hva med følgende?



# Presedens-regler

Vi går innenfra og ut, og “anvender” konnektiver i rekkefølgen

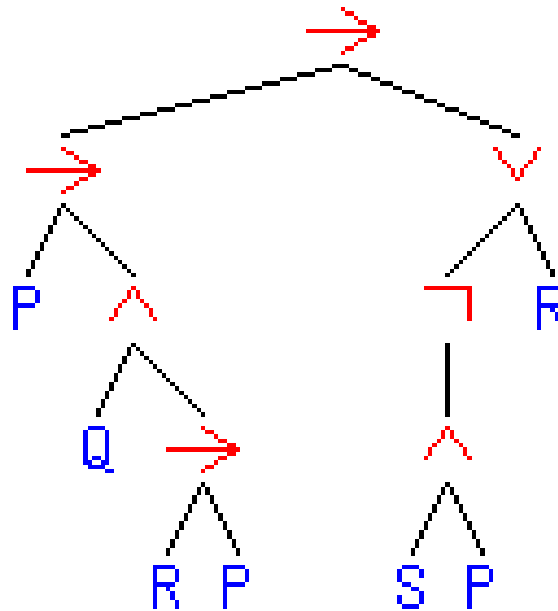
1.  $\neg$
2.  $\wedge$
3.  $\vee$
4.  $\rightarrow$

Binære konnektiver er dessuten venstre-assosiative, det vil si vi “anvender” forekomster (av samme konnektiv) lengst til venstre først.

- $(P \wedge (\neg Q)) \vee R$
- $((\neg P) \vee Q) \rightarrow ((\neg Q) \wedge R)$
- $(P \wedge Q) \wedge R$
- $(P \vee Q) \vee R$
- $(P \rightarrow Q) \rightarrow R$

# Hva med følgende?

$$P \rightarrow Q \wedge (R \rightarrow P) \rightarrow \neg (S \wedge P) \vee R$$





# Begreper idag

<p><b>Strenger</b> over alfabet A Den tomme strengen <math>\Lambda</math> Konkatenering av strenger</p>	<p><b>Tuppel</b>/sekvens vs. mengde Kartesisk produkt av mengder Aritet av <b>relasjon</b></p>
<p><b>Språk</b> over alfabet A Produkt LM av språk L og M <math>L^*</math> <math>L^+</math> <math>L^0, L^1, L^2, L^3, \text{ etc.}</math></p>	<p><b>Funksjoner</b> Argument, verdi Domene, kodomene Aritet av funksjon Funksjoner vs. relasjoner Eksempel: Sannhetsfunksjoner</p>