ZIM
UNI GRAZ

Grundlagen der Informatik
VO und KV

Mengenlehre und Prädikatenlogik

Überblick

Mengen

Mengenoperationen

Venn-Diagramm

Prädikatenlogik

Mengenlehre

- Eine der absoluten Grundlagen der Mathematik.
- Wird damit auch viel in der Informatik – vor allem in der theoretischen Informatik – verwendet.
- Mengen sind Datenstrukturen in der Programmierung.
- Georg Cantor = Erfinder der Mengenlehre

“*Eine Menge ist eine Ansammlung von wohl unterscheidbaren Objekten unserer Vorstellung oder der Realität*.” → “naive Mengenlehre”

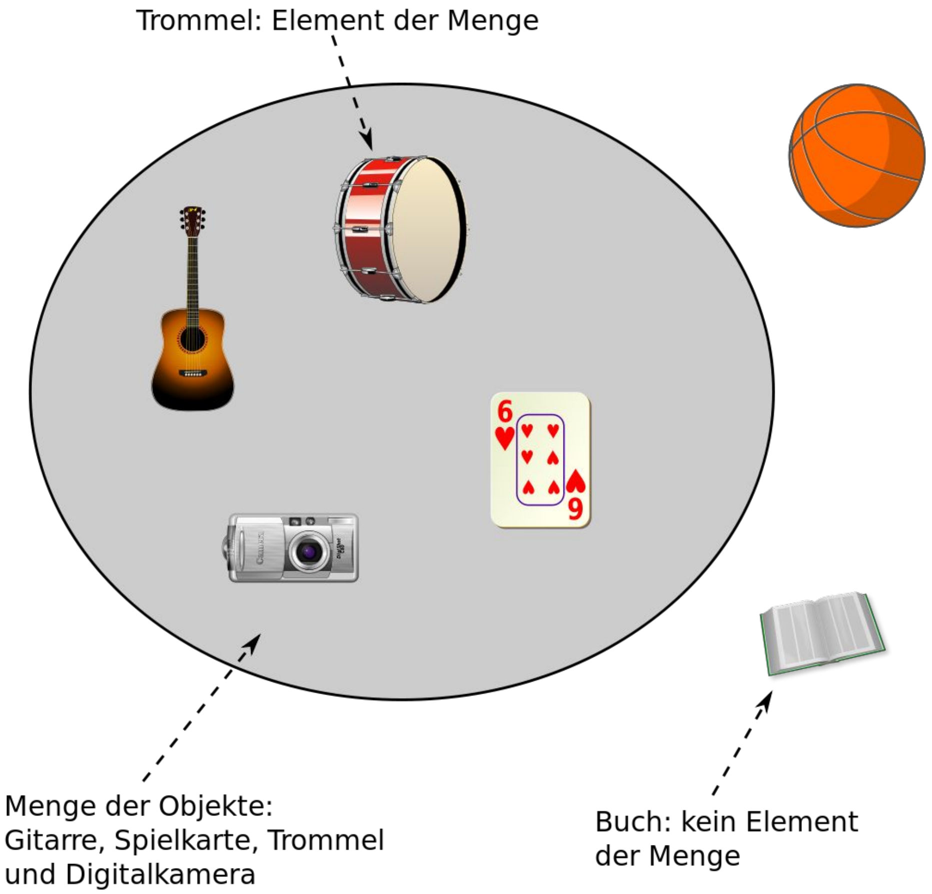
- Die Menge der 5 Eier.
- Die Menge der natürlichen Zahlen.
- Die Menge, der Mengen, der Mengen...

https://www.youtube.com/watch?v=AvVq2TfGQIg

Mengen

A = {
 Spielkarten, Trommel,
 Gitarre, Kamera
}

- T(12)={3, 4, 12, 6, 2, 1}
- |A| = 4   (Kardinalität)
- Mengen sind distinkt
- $\varnothing = \{\}$; $|\varnothing| = 0$



https://de.wikipedia.org/wiki/Menge_(Mathematik)

Teilmengen vs. echte Teilmenge

$A = \{2, 20, 200\}$

$B = \{8, 2, 5, 20, 100, 200, 456\}$

Jedes Element von A ist auch Element von B.

$2 \in A,\ 20 \in A,\ 200 \in A,\ 8 \notin A$

$A \subseteq B$

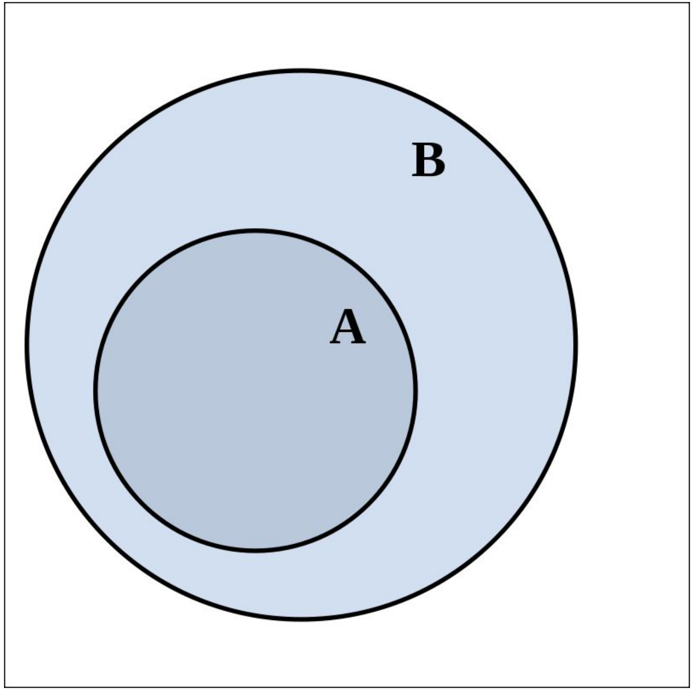
A **ist Teilmenge** von B

$A \subseteq B \text{ iff } \forall x \in A,\ x \in B$

A ist **echte Teilmenge** von B

$A \subset B$

$A \neq B$ (Operator $\subset$)



https://de.wikipedia.org/wiki/Teilmenge

Mengennotation = “set-builder”

Die Mengennotation erlaubt es, Kriterien festzumachen, die definieren, ob ein Element einer Menge zugehörig ist. So können auch unendliche Mengen definiert werden:

- $C = \{x^2 \mid x \in N\}$ … $1^2, 2^2, 3^2, 4^2$ …
- $D = \{x \mid x^2 \bmod p = 0,\ x \in N\}$ … $4^2 \bmod 2 = 0,\ 4 \in N$
- $E = \{x \in N \mid x \leq 3 \text{ or } x \geq 5\}$ … 3,4,5

http://gdi.ist.tugraz.at/skriptum.pdf
https://www.youtube.com/watch?v=H1NoUBFv3FA

Abgrenzung von Begriffen

Liste … eine Liste ist eine geordnete Sammlung von beliebig vielen Elementen. `[10, 2, 2, 3, 56, 3, 3, 3, 10]`

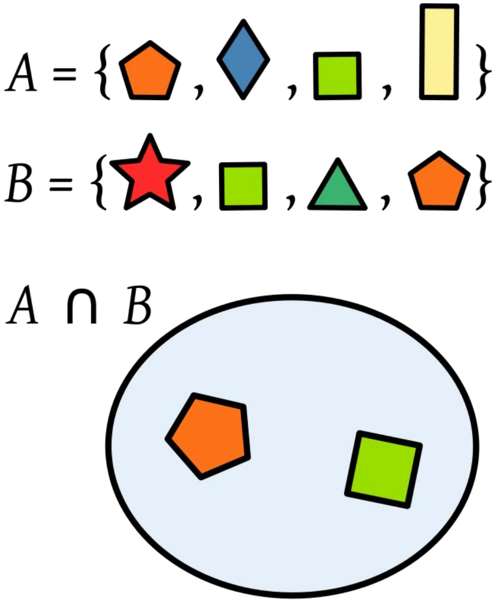
Sequenz … eine Sequenz ist eine geordnete Sammlung von beliebig vielen Elementen, die meistens auf eine endliche Größe beschränkt sind. `[1, 2, 3, 4]`

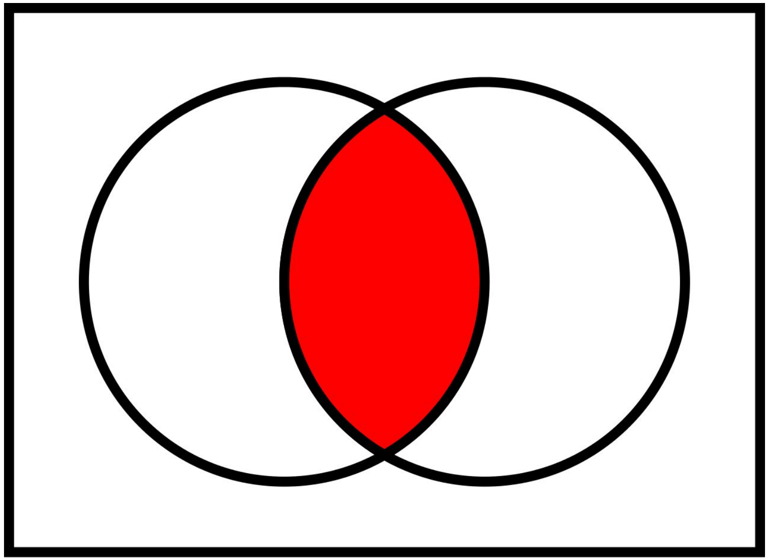
Tupel … eine geordnete Sammlung von homogenen Elementen endlicher Anzahl. `{name: “Christopher”, age:”30”}`

Menge … ungeordnete Sammlung von unterschiedlichen Objekten `myset = {"apple", "banana", "cherry"}`

http://gdi.ist.tugraz.at/skriptum.pdf

Schnittmenge - INTERSECTION - Konjunktion

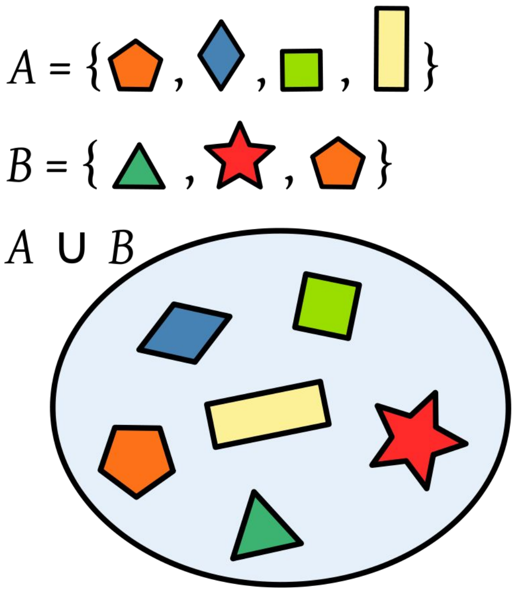


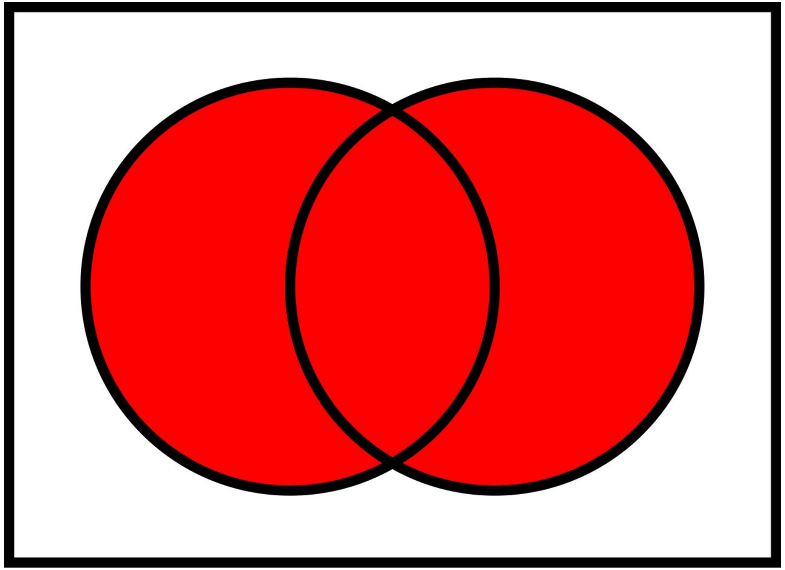


VENN-Diagramm

$$A \cap B = \{x \mid x \in A \land x \in B\}$$

Vereinigung - UNION - Disjunktion

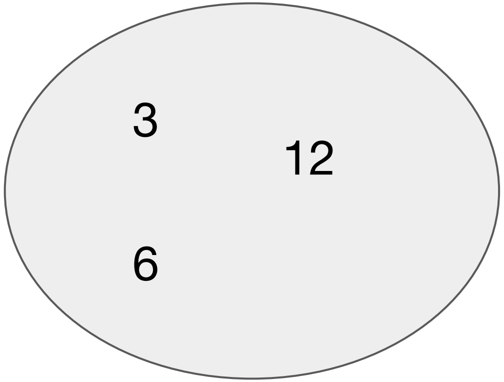


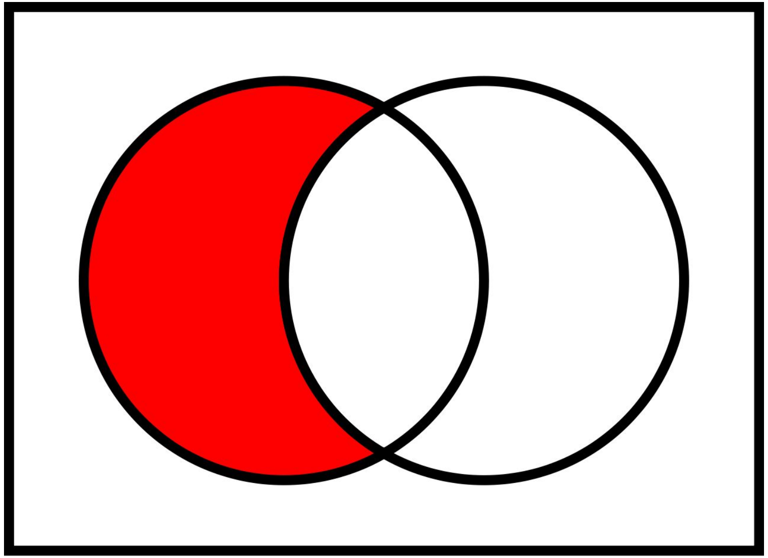


$$A \cup B = \{x \mid x \in A \vee x \in B\}$$

Komplement und Differenz

- T(12)={3, 4, 12, 6, 2, 1}
- T(4)={4, 1, 2}





T(12)\T(4)

$A \setminus B = \{x \mid (x \in A) \wedge (x \notin B)\}$

Venn Diagram

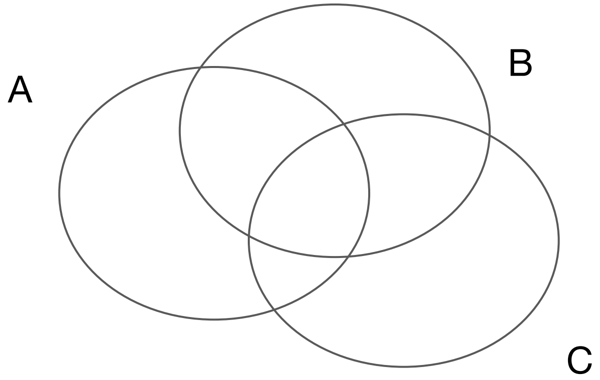
Zeichne:

- $S \cap (A \cup B)$ https://www.wolframalpha.com/input/?i=S+intersect+%28A+union+B%29
- $S' \cap (A \cup B)$ https://www.wolframalpha.com/input/?i=%28complement+S%29+intersect+%28A+union+B%29&lk=3

Übung

Zeige mittels VENN-Diagramm, ob die folgenden zwei Terme, in denen Mengenoperationen auf 3 Mengen (A,B,C) angewandt werden, gleich sind. Zeichne dazu die VENN-Diagramme (Kreise die sich schneiden) auf und schraffiere/färbe die Mengen, die durch die Operatoren definiert werden, ein.

$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$$



Lösung:

https://www.wolframalpha.com/input/?i=A%E2%88%AA%28B%E2%8B%82C%29+%3D+%28A%E2%88%AAB%29%E2%8B%82%28A%E2%88%AAC%29±

Abfragesprachen und Mengen

Das Ergebnis einer Abfrage (Query) ist eine Teilmenge des zugrundeliegenden Informationsbestandes. Man spricht daher auch von einer Filterung der Daten.

Es existieren Mengenoperationen in Abfragesprachen wie SQL, SPARQL etc.

https://glossar.hs-augsburg.de/Mengenoperatoren_in_SQL

Prädikatenlogik

Die Prädikatenlogik erweitert die Aussagenlogik um Quantoren und Prädikate

Wenn Sokrates ein Mensch ist, dann ist er sterblich

Sokrates ist ein Mensch … p

Sokrates ist sterblich … q

Alle Menschen sind sterblich.

Sokrates ist ein Mensch.

→ Sokrates ist sterblich

Manche Menschen sind faul.

Alle Faulen schlafen viel.

→ Manche Menschen schlafen viel

www.fb10.uni-bremen.de/khwagner/grundkurs2/kapitel4.aspx

Prädikatenlogik

- "*x is president of the USA*" → **`P(x)`**

  x ist eine Variable

  "*is president of the USA*" *das Prädikat*

- Für unterschiedliche Werte für x kann P(x) *true* oder *false* sein

  `P("Donald Trump") =` *`false`*

  `P("Joe Biden") =` *`true`*

- Kann beliebig viele Variablen enthalten → `P(x,y)`

  `A(x1, x2)` entspricht `x1 + x2 = 10` ist `true` wenn `x1 = 2` und `x2 = 8`

Quantoren

- $\forall$ … *universal quantifier* $\forall x$ “für jedes x”
- $\exists$ … *existential quantifier* $\exists x$ “es existiert mindestens ein x”

- Quantoren dürfen nie alleine stehen

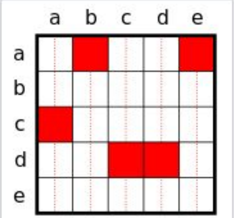
`P(x)` … “x liebt jemanden”

$\forall$`xP(x)` … für alle x gilt, dass sie jemanden lieben
→ Sepp liebt niemanden → `P(“Sepp”) = false`
→ dann ist auch $\forall$`xP(x) = false`
Weil wir jemanden gefunden haben, für den das nicht gilt.

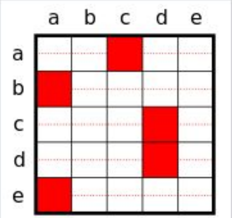
$\exists$`xP(x)` … “es existiert ein x, das jemanden liebt”

https://sites.math.washington.edu/~aloveles/Math300Winter2011/m300Quantifiers.pdf https://de.wikipedia.org/wiki/Pr%C3%A4dikatenlogik

Keine Spalte/Zeile ist leer:



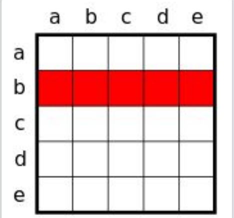
1. $\forall x \exists y Lyx$: Jeder wird von jemandem geliebt.



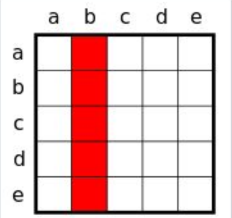
2. $\forall x \exists y Lxy$: Jeder liebt jemanden.

“Die Richtung getauscht”

Eine Zeile/Spalte ist voll:

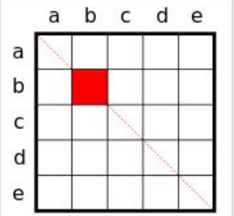


3. $\exists x \forall y Lxy$: Jemand liebt alle.

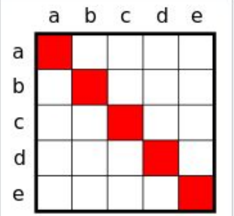


4. $\exists x \forall y Lyx$: Jemand wird von allen geliebt.

Die Diagonale ist nichtleer/voll:

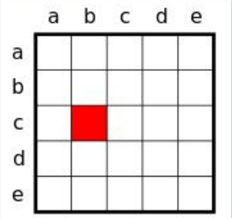


5. $\exists x Lxx$: Jemand liebt sich selbst.



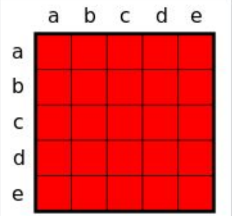
6. $\forall x Lxx$: Alle lieben sich selbst.

Die Matrix ist nichtleer/voll:



7. $\exists x \exists y Lxy$: Einer liebt einen.

8. $\exists x \exists y Lyx$: Einer wird von einem geliebt.



9. $\forall x \forall y Lxy$: Jeder liebt jeden.

10. $\forall x \forall y Lyx$: Jeder wird von jedem geliebt.

L(x,y) …

x liebt y

<table>
<tr><th>Prädikatenlogik – Deutsch</th><th>Erklärung</th></tr>
<tr><td>$\forall x(\mathrm{Katze}(x) \Rightarrow \mathrm{Säugetier}(x))$
„Alle Katzen sind Säugetiere“
(Es kann auch Säugetiere geben, die keine Katzen sind, aber keine Katzen, die keine Säugetiere sind)</td><td><table><tr><td>$\forall x$</td><td>$\mathrm{Katze}(x)$</td><td>$\Rightarrow$</td><td>$\mathrm{Säugetier}(x)$</td></tr><tr><td>Für alle x:</td><td>(Gilt) x sei eine Katze</td><td>dann</td><td>sei x ein Säugetier</td></tr></table></td></tr>
<tr><td>$\forall x(\mathrm{Katze}(x) \wedge \mathrm{Säugetier}(x))$
„Alles ist eine Katze und ein Säugetier“</td><td><table><tr><td>$\forall x$</td><td>$\mathrm{Katze}(x)$</td><td>$\wedge$</td><td>$\mathrm{Säugetier}(x)$</td></tr><tr><td>Für alle x gilt:</td><td>x sei eine Katze</td><td>und</td><td>x sei ein Säugetier</td></tr></table></td></tr>
<tr><td>$\exists x(\mathrm{Stadt}(x) \wedge \mathrm{nördlich}(x, \mathrm{München}))$ [4]
„Es gibt mindestens eine Stadt nördlich von München“</td><td><table><tr><td>$\exists x$</td><td>$(\mathrm{Stadt}(x)$</td><td>$\wedge$</td><td>$\mathrm{nördlich}(x, \mathrm{München}))$</td></tr><tr><td>Es gibt mindestens ein x</td><td>das ist eine Stadt</td><td>und</td><td>nördlich von München liegt</td></tr></table></td></tr>
<tr><td>$\neg\exists x(\mathrm{Stadt}(x) \wedge \mathrm{nördlich}(x, x))$ [5]
„Keine Stadt liegt nördlich ihrer selbst“</td><td><table><tr><td>$\neg\exists x$</td><td>$(\mathrm{Stadt}(x)$</td><td>$\wedge$</td><td>$\mathrm{nördlich}(x, x))$</td></tr><tr><td>Es gibt kein x</td><td>das eine Stadt ist</td><td>und</td><td>nördlich von x liegt</td></tr></table></td></tr>
</table>

https://de.wikipedia.org/wiki/Pr%C3%A4dikatenlogik

Übung

Im Folgenden seien die Quantoren ∀x und ∃x immer auf Menschen beschränkt.

- ∀x bedeutet im folgenden immer
  "*Für alle Menschen x gilt...*"
- ∃ x bedeutet "*Für einige Menschen x gilt...*" oder "*Es gibt mindestens einen Menschen x, für den gilt...*"

Weiters sind folgende Individuenkonstanten, Prädikate und Relationen gegeben :

Joshua	...	j
Dave	...	d
Nick	...	n
Troy	...	t
x ist ein/e Musiker/in	...	M(x)
x ist Musik-Kritiker/in	...	K(x)
x mag y	...	R(x,y)

1. Joshua und Nick mögen sich nicht.
2. Joshua mag Dave und der mag Nick.
3. Jede Musikerin mag Joshua.
4. Keine Kritikerin, die Joshua mag, mag Nick nicht.
5. Jede Kritikerin, die auch Musikerin ist, mag Dave.
6. Einige Kritikerinnen mögen überhaupt niemanden.

Lösung

1. Joshua und Nick mögen sich nicht.
   $\neg R(j,n) \land \neg R(n,j)$ (→ die beiden mögen sich nicht gegenseitig)
   $\neg R(j,j) \land \neg R(n,n)$ (→ keiner der beiden mag sich selbst)
2. Joshua mag Dave und der mag Nick.
   $R(j,d) \land R(d,n)$
3. Jede Musikerin mag Joshua.
   $\forall x(M(x) \rightarrow R(x,j))$
4. Keine Kritikerin, die Joshua mag, mag Nick nicht.
   $\neg\exists x(K(x) \land R(x,j) \land \neg R(x,n))$
5. Jede Kritikerin, die auch Musikerin ist, mag Dave.
   $\forall x((K(x) \land M(x)) \rightarrow R(x,d))$
6. Einige Kritikerinnen mögen überhaupt niemanden
   $\exists x(K(x) \land \forall y \neg R(x,y))$

Allgemein kann man bei $\exists$ eine Konjunktion verwenden und bei $\forall$ eine Implikation. https://math.stackexchange.com/questions/906843/deciding-between-implication-and-conjunction

$\exists s \in S\ \forall b \in B\ \mathrm{stehtIn}(b, s) \Rightarrow \mathrm{istGroß}(b)$ **[TRUE]**

- *s∈S … eine Straße*
- *S … Menge der Straßen*
- *b∈B … Baum*
- *B … Menge der Bäume*
- stehtIn(b, s) … Baum steht in einer Straße
- istGroß(b) … Baum ist groß

"Es gibt eine Straße, in der jeder Baum groß ist".

Was heißt: $\exists s \in S\ \forall b \in B\ \mathrm{stehtIn}(b, s) \land \mathrm{istGroß}(b)$

$\exists s \in S\ \forall b \in B\ \mathrm{stehtIn}(b, s) \Rightarrow \mathrm{istGroß}(b)$ **[TRUE]**

"*Es gibt eine Straße, in der jeder Baum groß ist*". [Dies sollte der Ausgangspunkt sein.]

$\exists s \in S\ \forall b \in B\ \mathrm{stehtIn}(b, s) \land \mathrm{istGroß}(b)$

"Das hieße, dass alle Bäume groß wären und es eine Straße gäbe, in der alle Bäume stünden."

[die Implikation bildet die Abhängigkeit (nicht Kausalität) der Bäume ab, die an der Straße stehen.
Mit UND ist keine Abhängigkeit gegeben und alle Bäume sind groß (auch die, die nicht an der Straße stehen)
Dies gilt für ∀]

Was ist mit: $\exists s \in S\ \neg\forall b \in B\ \neg\mathrm{stehtIn}(b,s) \land \mathrm{istGroß}(b)$?

$\exists s \in S\ \forall b \in B$ stehtIn(b, s) $\Rightarrow$ istGroß(b) **[TRUE]**

"*Es gibt eine Straße, in der jeder Baum groß ist*". [Dies sollte der Ausgangspunkt sein.]

$\exists s \in S\ \forall b \in B$ stehtIn(b, s) $\wedge$ istGroß(b)

"Das hieße, dass alle Bäume groß wären und es eine Straße gäbe, in der alle Bäume stünden."

[die Implikation bildet die Abhängigkeit (nicht Kausalität) der Bäume ab, die an der Straße stehen.
Mit UND ist keine Abhängigkeit gegeben und alle Bäume sind groß (auch die, die nicht an der Straße stehen)
Dies gilt für $\forall$]

$\exists s \in S\ \neg\forall b \in B\ \neg$stehtIn(b,s) $\wedge$ istGroß(b)

"Das hieße, dass es eine Straße gäbe sowie einen Baum, der in dieser Straße steht oder klein (nicht groß) ist."
"Das hieße, dass es eine Straße gäbe sowie einen Baum, der nicht in dieser Straße steht und groß ist."

- einen Baum …weil $\neg\forall$ nicht alle;
- Negationsregel: $\neg\forall x\ A(x) \equiv \exists x \neg A(x)$
- **$\exists s \in S\ \neg\forall b \in B\ \neg$stehtIn(b,s) $\wedge$ istGroß(b)**
  $\equiv \exists s \in S\ \exists b \in B\ \neg(\neg$stehtIn(b,s) $\wedge$ istGroß(b)) | **$\neg(\neg a \wedge b) \equiv (a \vee \neg b)$**
  $\equiv \exists s \in S\ \exists b \in B\ \neg$stehtIn(b,s) $\vee\ \neg$istGroß(b)
  - https://www.wolframalpha.com/input/?i=not%28not%28s%29+AND+g%29
  - https://www.wolframalpha.com/input/?i=s+OR+not%28g%29

Negationsregel

- $\neg\forall x\, A(x) \equiv \exists x\, \neg A(x)$
- $\neg\exists x\, A(x) \equiv \forall x\, \neg A(x)$

∀s∈S ¬∃b∈B ¬stehtIn(b,s) ∨ ¬istGroß(b)

“Das hieße, dass alle Bäume in allen Straßen stünden und groß wären.”

- ¬∃b∈B = es existiert nicht einer, der nicht an der Straße steht
- **¬∃b∈B ¬stehtIn(b,s) ∨ ¬istGroß(b)**

  ≡ ∀b∈B ¬(¬stehtIn(b,s) ∨ ¬istGroß(b)) | ¬(¬a ∨ ¬b) ≡ a ∧ b

  ≡ ∀b∈B stehtIn(b,s) ∧ istGroß(b)
    - https://www.wolframalpha.com/input/?i=not%28not%28y%29+OR+not%28b%29%29
    - https://www.wolframalpha.com/input/?i=y+and+b

∃s∈S ¬∃b∈B stehtIn(b, s) ∧ ¬istGroß(b) **[TRUE]**

“Es gibt keine Straße an der kein Baum steht; der nicht groß ist (klein) ist

“*Es gibt eine Straße in der jeder Baum groß ist”*

∃s∈S ∀b∈ B stehtIn(b, s) ⇒ istGroß(b) | Term doppelt negieren und Implikation auflösen

≡ ∃s∈S ¬¬ ∀b∈ B ¬stehtIn(b, s) ∨ istGroß(b) | ¬ ∀ … ∃; ¬¬ ∀ … ¬∃

≡ ∃s∈S ¬ ∃b∈ B ¬(¬stehtIn(b, s) ∨ istGroß(b)) | ¬(¬a ∨ b) ≡ (a ∧ ¬b)

≡ ∃s∈S ¬ ∃b∈ B stehtIn(b, s) ∧ ¬istGroß(b))