CSE581
Computer Science Fundamentals: Theory

Professor Anita Wasilewska

P1 LOGIC: LECTURE 2b

CHAPTER 2 REVIEW

Mathematical Statements Translations

Our goal now is to "translate " mathematical and natural language statement into correct formulas of the predicate language $\mathcal{L}$.

Let's start with some **observations**.

O1 The quantifiers in $\forall_{x \in N}, \ \exists_{y \in Z}$ **are not** the one used in logic.

O2 The predicate language $\mathcal{L}$ **admits only** quantifiers $\forall x, \ \exists y$, for any variables $x, y \in VAR$.

O3 The quantifiers $\forall_{x \in N}, \ \exists_{y \in Z}$ are called **quantifiers with restricted domain**.

The **restriction** of the quantifier domain can, and often is given by more complicated statements.

Quantifiers with Restricted Domain

The quantifiers $\forall_{A(x)}$ and $\exists_{A(x)}$ are called quantifiers with **restricted domain** , or **restricted quantifiers** , where $A(x) \in \mathcal{F}$ is any formula with a free variable $x \in VAR$.

Definition

$\forall_{A(x)} B(x)$ stands for a formula $\forall x(A(x) \Rightarrow B(x)) \in \mathcal{F}$.

$\exists_{A(x)} B(x)$ stands for a formula $\exists x(A(x) \cap B(x)) \in \mathcal{F}$.

We write it as the following **transformations rules** for **restricted quantifiers**

$$\forall_{A(x)} B(x) \equiv \forall x(A(x) \Rightarrow B(x))$$

$$\exists_{A(x)} B(x) \equiv \exists x(A(x) \cap B(x))$$

Translations to Formulas of $\mathcal{L}$

Translations to Formulas of $\mathcal{L}$

Given a mathematical statement **S** written with logical symbols.

We obtain a formula $A \in \mathcal{F}$ that is a **translation** of **S** into $\mathcal{L}$ by conducting a following sequence of steps.

Step 1 We **identify** basic statements in **S**, i.e. mathematical statements that involve only relations. They are to be translated into atomic formulas.

We **identify** the relations in the basic statements and **choose** the predicate symbols as their names.

We **identify** all functions and constants (if any) in the basic statements and **choose** the function symbols and constant symbols as their names.

Step 2 We **write** the basic statements as **atomic formulas** of $\mathcal{L}$.

Translations to Formulas of $\mathcal{L}$

Remember that in the predicate language $\mathcal{L}$ we write a function symbol in front of the function arguments not between them as we write in mathematics.

The same applies to relation symbols.

For example we re-write a basic mathematical statement $x + 2 > y$ as $> (+(x, 2), y)$, and then we write it as an **atomic formula** $P(f(x, c), y)$

$P \in \mathbf{P}$ stands for two argument relation $>$,

$f \in \mathbf{F}$ stands for two argument function $+$, and $c \in \mathbf{C}$ stands for the number 2.

Translations to Formulas of $\mathcal{L}$

Step 3 We **write** the statement **S** a formula with restricted quantifiers (if needed)

Step 4. We **apply** the **transformations rules** for **restricted quantifiers** to the formula from Step 3 and **obtain** a proper formula A of $\mathcal{L}$ as a result, i.e. as a **translation** of the given mathematical statement **S**

In case of a translation from mathematical statement written without logical symbols **we add** a following step.

Step 0 We **identify** propositional connectives and quantifiers and use them to re-write the statement in a form that is as close to the structure of a logical formula as possible

Translations Examples

Exercise

Given a mathematical statement **S** written with logical symbols

$$(\forall_{x \in N}\ x \geq 0\ \cap\ \exists_{y \in Z}\ y = 1)$$

1. Translate it into a proper logical formula with restricted quantifiers i.e. into a formula of $\mathcal{L}$ that **uses** the restricted domain quantifiers.

2. Translate your restricted quantifiers formula into a correct formula **without** restricted domain quantifiers, i.e. into a proper formula of $\mathcal{L}$

A **long** and **detailed solution** is given in Chapter 2, page 28.

A short statement of the exercise and a short solution follows

Translations Examples

Exercise

Given a mathematical statement **S** written with logical symbols

$$(\forall_{x \in N}\, x \geq 0 \;\cap\; \exists_{y \in Z}\, y = 1)$$

Translate it into a proper formula of $\mathcal{L}$.

Short Solution

The basic statements in **S** are: $x \in N,\; x \geq 0,\; y \in Z, y = 1$

The corresponding atomic formulas of $\mathcal{L}$ are:
$N(x),\; G(x, c_1),\; Z(y),\; E(y, c_2)$, for
$n \in N,\; x \geq 0,\; y \in Z,\; y = 1$, respectively.

The statement **S** becomes restricted quantifiers formula

$$(\forall_{N(x)} G(x, c_1) \;\cap\; \exists_{Z(y)}\, E(y, c_2))$$

By the **transformation rules** we get $A \in \mathcal{F}$:

$$(\forall x(N(x) \Rightarrow G(x, c_1)) \cap \exists y(Z(y) \cap E(y, c_2)))$$

Translations Examples

Exercise

Here is a mathematical statement **S**:

"For all real numbers x the following holds: If $x < 0$, then there is a natural number n, such that $x + n < 0$."

1. Re-write **S** as a symbolic mathematical statement SF that only uses mathematical and logical symbols.

2. Translate the symbolic statement SF into to a corresponding formula $A \in \mathcal{F}$ of the predicate language $\mathcal{L}$

Translations Examples

Solution

The statement **S** is:

"For all real numbers x the following holds: If $x < 0$, then there is a natural number n, such that $x + n < 0$."

S becomes a symbolic mathematical statement SF

$$\forall_{x \in R}(x < 0 \Rightarrow \exists_{n \in N}\ x + n < 0)$$

We write R(x) for $x \in R$, N(y) for $n \in N$, a constant c for the number 0. We use $L \in \mathbf{P}$ to denote the relation $<$ We use $f \in \mathbf{F}$ to denote the function $+$

The statement $x < 0$ becomes an **atomic formula** L(x, c).
The statement $x + n < 0$ becomes L(f(x,y), c)

Translations Examples

Solution c.d.

The symbolic mathematical statement SF

$$\forall_{x \in R}(x < 0 \Rightarrow \exists_{n \in N}\ x + n < 0)$$

becomes a restricted quantifiers formula

$$\forall_{R(x)}(L(x, c) \Rightarrow \exists_{N(y)} L(f(x, y), c))$$

We apply now the **transformation rules** and get a corresponding formula $A \in \mathcal{F}$:

$$\forall x(N(x) \Rightarrow (L(x, c) \Rightarrow \exists y(N(y) \cap L(f(x, y), c)))$$

PART 3: Translations to Predicate Languages

Translations Exercises

Exercise 1

Given a Mathematical Statement written with logical symbols

$$\forall_{x \in R}\ \exists_{n \in N}(x + n > 0 \Rightarrow \exists_{m \in N}(m = x + n))$$

1. Translate it into a proper logical formula with restricted domain quantifiers

2. Translate your restricted domain quantifiers logical formula into a correct logical formula **without** restricted domain quantifiers

Exercise 1 Solution

1. We translate the Mathematical Statement

$$\forall_{x \in R} \, \exists_{n \in N} (x + n > 0 \Rightarrow \exists_{m \in N} (m = x + n))$$

into a proper **logical formula** with restricted domain quantifiers as follows

Step 1

We identify all **predicates** and use their **symbolic** representation as follows:

R(x) for $x \in R$

N(x) for $n \in N$

G(x,y) for relation $>$, E(x,y) for relation $=$

Exercise 1 Solution

Step 2

We identify all **functions** and **constants** and their **symbolic** representation as follows:

f(x,y) for the function $+$, c for the constant 0

Step 3

We write **mathematical** expressions in as **symbolic logic** formulas as follows:

G(f(x,y), c) for $x+n>0$ and E(z, f(x,y)) for $m=x+n$

Step 4

We identify logical **connectives** and **quantifiers** and write the **logical formula** with restricted domain quantifiers as follows

$$\forall_{R(x)} \exists_{N(y)} (G(f(x,y),c) \Rightarrow \exists_{N(z)} E(z, f(x,y)))$$

Exercise 1 Solution

2. We translate the **logical formula** with restricted domain quantifiers

$$\forall_{R(x)} \exists_{N(y)}(G(f(x,y),c) \Rightarrow \exists_{N(z)}E(z,f(x,y)))$$

into a correct **logical formula** **without** restricted domain quantifiers as follows

$$\forall x(R(x) \Rightarrow \exists_{N(y)}(G(f(x,y),c) \Rightarrow \exists_{N(z)}E(z,f(x,y))))$$

$$\equiv \ \forall x(R(x) \Rightarrow \exists y(N(y) \cap (G(f(x,y),c) \Rightarrow \exists_{N(z)}E(z,f(x,y)))))$$

$$\equiv \ \forall x(R(x) \Rightarrow \exists y(N(y) \cap (G(f(x,y),c) \Rightarrow \exists z(N(z) \cap E(z,f(x,y)))))$$

Correct **logical formula** is:

$$\forall x(R(x) \Rightarrow \exists y(N(y) \cap (G(f(x,y),c) \Rightarrow \exists z(N(z) \cap E(z,f(x,y))))))$$

Translations Exercises

Exercise 2

Here is a mathematical statement **S**:

For all natural numbers n the following holds:

If $n < 0$, **then** *there is a natural number m, such that* $m + n < 0$

P1. Re-write **S** as a Mathematical Statement "formula" **MSF** that only uses **mathematical** and **logical symbols**

P2. Translate your Mathematical Statement "formula" **MSF** into to a correct **predicate language formula** **LF**

P3. Argue whether the statement **S** it true of false

P4. Give an **interpretattion** of the **predicate language formula** **LF** under which it is **false**

Exercise 2 Solution

P1. We **re-write** mathematical statement **S**

For all natural numbers n the following holds:

If $n < 0$, **then** *there is a natural number m, such that* $m + n < 0$

as a Mathematical Statement "formula" **MSF** that only uses mathematical and logical symbols as follows

$$\forall_{n \in N}(n < 0 \Rightarrow \exists_{m \in N}(m + n < 0))$$

Exercise 2 Solution

P2. We translate the **MSF** ”formula”

$$\forall_{n \in N}(n < 0 \Rightarrow \exists_{m \in N}(m + n < 0))$$

into a correct **predicate language** **formula** using the following **5** steps

Step 1

We identify **predicates** and write their **symbolic** representation as follows

We write N(x) for $x \in N$ and L(x,y) for relation $<$

Step 2

We identify **functions** and **constants** and write their **symbolic** representation as follows

f(x,y) for the function $+$ and c for the constant 0

Exercise 2 Solution

Step 3

We write the mathematical expressions in **S** as atomic formulas as follows:

L(f(y,c), c) for $m + n < 0$

Step 4

We identify logical **connectives** and **quantifiers** and write the **logical formula** with restricted domain quantifiers as follows

$$\forall_{N(x)}(L(x, c) \Rightarrow \exists_{N(y)} L(f(y, c), c))$$

Exercise 2 Solution

Step 5

We translate the above into a correct **logical formula**

$$\forall x(N(x) \Rightarrow (L(x,c) \Rightarrow \exists y(N(y) \cap L(f(y,c),c)))$$

P3 Argue whether the statement **S** it true of false

Statement $\forall_{n\in N}(n < 0 \Rightarrow \exists_{m\in N}(m + n < 0))$ is TRUE as the statement $n < 0$ is FALSE for all $n \in N$ and the classical implication FALSE $\Rightarrow$ Anyvalue is always TRUE

Exercise 2 Solution

P4. Here is an **interpretation** in a non-empty set X under which the **predicate language formula**

$$\forall x(N(x) \Rightarrow (L(x,c) \Rightarrow \exists y(N(y) \cap L(f(y,c),c)))$$

is false

Take a set $X = \{1,2\}$

We **interpret** N(x) as $x \in \{1,2\}$, L(x, y) as $x > y$, and constant c as 1

We **interpret** f as a two argument function f_I defined on the set X by a formula $f_I(y,x) = 1$ for all $y, x \in \{1,2\}$

The mathematical statement

$$\forall_{x\in\{1,2\}}(x > 1 \Rightarrow \exists_{y\in\{1,2\}}(f_I(y,x) > 1))$$

is a **false statement** when $x = 2$

In this case we have $2 > 1$ is **true** and as $f_I(y,2) = 1$ for all $y \in \{1,2\}$ we get that $\exists_{y\in\{1,2\}}(f_I(y,2) > 1))$ is **false** as $1 > 1$ is **false**

Predicate Tautologies

The notion of **predicate tautology** is much more complicated then that of the **propositional** one

We **introduce** it intuitively here and **define** it formally in later chapters

Predicate tautologies are also called valid formulas, or laws of quantifiers to distinguish them from the **propositional** case

We provide here a motivation, some examples and an intuitive definitions

We also list and discuss the most used and useful **predicate tautologies** and **equational laws** of quantifiers

Interpretation

The formulas of the **predicate** language $\mathcal{L}$ have a meaning only when an **interpretation** is given for its symbols

We **define** the **interpretation** I in a set $U \neq \emptyset$ by interpreting predicate and functional **symbols** of $\mathcal{L}$ as concrete **relations** and **functions** defined in the set U

We interpret constants symbols as **elements** of the set U

The set U is called the **universe** of the **interpretation** **I**

Model Structure

We define a **model structure** for the predicate language $\mathcal{L}$ as a pair

$$\mathbf{M} = (U,\ I)$$

where the set U is called the structure **universe** and of the I is the structure **interpretation** in the universe U

Given a formula A of $\mathcal{L}$, and the **model structure** $\mathbf{M} = (U, I)$

We **denote** by

$$A_I$$

a statement defined in the structure $\mathbf{M} = (U, I)$ that is **determined** by the formula A and the interpretation I in the universe U

Model Structure

When the formula A is a **sentence**, it means it is a formula without free variables, the **model structure** statement

$$A_I$$

represents a proposition that is **true** or **false** in the universe U, under the interpretation I

When the formula A **is not** a sentence, it contains free variables and may be **satisfied** (i.e. true) for some values in the universe U and **not satisfied** (i.e. false) for the others

Lets look at few simple examples

Examples

Example

Let A be a formula $\exists x P(x, c)$

Consider a **model structure** $\mathbf{M}_1 = (N, I_1)$

The universe of the interpretation I_1 is the set N of natural numbers

We **define** I_1 as follows:

We **interpret** the two argument predicate P as a relation $<$ and the constant c as number 5, i.e we put

$P_{I_1} := $ and $c_{I_1} : 5$

Examples

The formula A: $\exists x P(x,c)$ under the interpretation I_1 becomes a mathematical statement

$$\exists x\ x = 5$$

defined in the set N of natural numbers

We write it for short

$$A_{I_1} : \exists_{x \in N}\ x = 5$$

A_{I_1} is obviously a **true** mathematical statement in the model structure $\mathbf{M}_1 = (N, I_1)$

We write it **symbolically** as

$$\mathbf{M}_1 \models \exists x P(x,c)$$

and say: $\mathbf{M}_1$ is a **model** for the formula A

Examples

Example

Consider now a model structure $\mathbf{M}_2 = (N, I_2)$ and the formula A: $\exists x P(x, c)$

We **interpret** now the predicate P as relation $<$ in the set N of natural numbers and the constant c as number 0

We write it as

$$P_{I_2} : < \quad \text{and} \quad c_{I_2} : \; 0$$

Examples

The formula A: $\exists x P(x, c)$ under the interpretation I_2 becomes a mathematical statement $\exists x\ x < 0$ defined in the set N of natural numbers

We write it for short

$$A_{I_2}: \quad \exists_{x \in N}\ x < 0$$

A_{I_2} is obviously a **false** mathematical statement.

We say: the formula A: $\exists x P(x, c)$ is **false** under the interpretation I_2 in $\mathbf{M}_2$, or we say for short: A is **false** in $\mathbf{M}_2$

We write it **symbolically** as

$$\mathbf{M}_2 \not\models \exists x P(x, c)$$

and say that $\mathbf{M}_2$ is a **counter-model** for the formula A

Examples

Example

Consider now a **model structure**

$\mathbf{M}_3 = (Z, I_3)$ and the formula A: $\exists x P(x, c)$

We **define** an interpretation I_3 in the set of all integers Z exactly as the interpretation I_1 was defined, i.e. we put

$$P_{I_3} : < \quad \text{and} \quad c_{I_3} : 0$$

Examples

In this case we get

$$A_{I_3}: \ \exists_{x \in Z}\ x < 0$$

Obviously A_{I_3} is a **true** mathematical statement

The formula A is **true** under the interpretation I_3 in $\mathbf{M}_3$ (A is **satisfied, true** in $\mathbf{M}_3$)

We write it symbolically as

$$\mathbf{M}_3 \models \exists x P(x, c)$$

$\mathbf{M}_3$ is yet another **model** for the formula A

Examples

When a formula A **is not** a closed, i.e. is not a sentence, the situation gets more complicated

A can be **satisfied** (i.e. true) for some values in the universe U of a $\mathbf{M} = (U, I)$

But also and can be **not satisfied** (i.e. false) for some other values in the universe U of a $\mathbf{M} = (U, I)$

We explain it in the following examples

Examples

Example

Consider a formula

$$A_1 : R(x, y),$$

We define a model structure

$$\mathbf{M} = (N, I)$$

where R is **interpreted** as a relation $\leq$ defined in the set N of all natural numbers, i.e. we put $R_I : \leq$

In this case we get

$$A_{1\,I} : \; x \leq y$$

and $A_1 : \; R(x, y)$ is **satisfied** in model structure $\mathbf{M} = (N, I)$ by all $n, m \in N$ such that $n \leq m$

Examples

Example

Consider a following formula

$$A_2 : \ \forall y R(x, y)$$

and the same model structure $\mathbf{M} = (N, I)$, where R is **interpreted** as a relation $\leq$ defined in the set N of all natural numbers, i.e. we put

$$R_I : \leq$$

In this case we get that

$$A_{2I} : \forall_{y \in N} \ x \leq y$$

and so the formula $A_2 : \ \forall y R(x, y)$ is **satisfied** in $\mathbf{M} = (N, I)$ **only** by the natural number 0

Examples

Example

Consider now a formula

$$A_3 : \exists x \forall y R(x, y)$$

and the same model structure $\mathbf{M} = (N, I)$, where R is **interpreted** as a relation $\leq$ defined in the set N of all natural numbers, i.e. we put $R_I : \leq$

In this case the statement

$$A_{3I} : \exists_{x \in N} \forall_{y \in N}\ x \leq y$$

asserts that there is a smallest number

This is a **true** statement and we call the structure $\mathbf{M} = (N, I)$ ia **model** for the formula $A_3 : \exists x \forall y R(x, y)$

Predicate Tautology Definition

We want the predicate language **tautologies** to have the same property as the **tautologies** of the propositional language, namely to be **always true**

In this case, we intuitively agree that it means that we want the **predicate tautologies** to be formulas that are **true** under **any** interpretation in **any** possible universe

A rigorous definition of the **predicate tautology** is provided in Chapter 8

Predicate Tautology Definition

We construct the rigorous definition of a **predicate tautology** in a following sequence of steps

S1 We define **formally** the notion of **interpretation** I of symbols of the language $\mathcal{L}$ in a set $U \neq \emptyset$, i.e. in a **model structure** $\mathbf{M} = (U, I)$ for $\mathcal{L}$

S2 We define **formally** a notion

" a formula A of $\mathcal{L}$ is **true** in the structure $\mathbf{M} = (U, I)$"

We write it symbolically $\mathbf{M} \models A$ and call thestructure $\mathbf{M} = (U, I)$ a **model** for the formula A

Predicate Tautology Definition

S3 We define a notion "A is a predicate tautology" as follows

Defintion

For any formula A of predicate language $\mathcal{L}$,

A is a **predicate tautology** (valid formula) if and only if

$$\mathbf{M} \models A$$

for all model structures $\mathbf{M} = (U, I)$ for the language $\mathcal{L}$

Predicate Tautology Definition

Directly from the above definition we get the following definition of a notion " A is not a predicate tautology"

Defintion

For any formula A of predicate language $\mathcal{L}$,

A **is not** a predicate **tautology** if and only if

there is a model structure $\mathbf{M} = (U, I)$ for $\mathcal{L}$, such that

$$\mathbf{M} \not\models A$$

We call such model structure **M** a **counter-model** for A

Predicate Tautology Definition

The definition of a notion

" A is not a predicate tautology"

says that in order to prove that a formula A **is not** a predicate tautology one has to show a **counter- model** for it

It means that one has to **define** a non-empty set U and **define** an interpretation I, such that we can prove that

$$A_I$$

is **false**

Predicate Tautology Definition

We use terms **predicate** tautology or **valid** formula instead of just saying a **tautology** in order to distinguish tautologies belonging to two very different languages

For the same reason we usually reserve the symbol $\models$ for **propositional** case

Sometimes we use symbols

$$\models_p \quad \text{or} \quad \models_f$$

to denote **predicate** tautologies

p stands for **predicate** and f stands **first order**

Predicate tautologies are also called **laws of quantifiers**

We will use **both** names

Predicate Tautologies Examples

Here are some examples of **predicate** tautologies and **counter models** for formulas that are **not** tautologies

Example

For any formula $A(x)$ with a free variable x:

$$\models_p \ (\forall x\, A(x) \Rightarrow \exists x\, A(x))$$

Observe that the formula

$$(\forall x\, A(x) \Rightarrow \exists x\, A(x))$$

represents an infinite number of formulas.

It is a **tautology** for **any** formula $A(x)$ of $\mathcal{L}$ with a free variable x

Predicate Tautologie Examples

The **inverse** implication to $(\forall x\, A(x) \Rightarrow \exists x\, A(x))$ **is not** a predicate tautology, i.e.

$$\not\models_p \ (\exists x\, A(x) \Rightarrow \forall x\, A(x))$$

To **prove it** we have to provide an **example** of a **concrete formula** $A(x)$ and construct a **counter-model** $\mathbf{M} = (U, I)$ for the formula

$$F : (\exists x\, A(x) \Rightarrow \forall x\, A(x))$$

Let the **concrete** $A(x)$ be an **atomic** formula $P(x, c)$

We define $\mathbf{M} = (N, I)$ for N set of natural numbers and $P_I : <, \quad c_I : \ 3$

The formula F becomes an obviously **false** mathematical statement

$$F_I : (\exists_{n \in N} n < 3 \Rightarrow \forall_{n \in N} n < 3)$$

Restricted Quantifiers Laws

We have to be very careful when we deal with **restricted domain** quantifiers

For example, the most basic predicate tautology

$$(\forall x\, A(x) \Rightarrow \exists x\, A(x))$$

fails when written with the **restricted domain** quantifiers, i.e.

We show that

$$\not\models_p \ (\forall_{B(x)}\, A(x) \Rightarrow \exists_{B(x)}\, A(x))$$

To **prove** this we have to show that corresponding formula of $\mathcal{L}$ obtained by the restricted quantifiers transformations rules **is not** a predicate tautology, i.e. to prove:

$$\not\models_p \ (\forall x(B(x) \Rightarrow A(x)) \Rightarrow \exists x(B(x) \cap A(x))).$$

Restricted Quantifiers Laws

We construct a **counter-model** **M** for the formula

$$F: \quad (\forall x(B(x) \Rightarrow A(x)) \Rightarrow \exists x(B(x) \cap A(x)))$$

We take

$$\mathbf{M} = (N, I),$$

where N is the set of natural numbers

We take as the **concrete** formulas $B(x)$, $A(x)$ atomic formulas

$$Q(x,\ c) \quad \text{and} \quad P(x,\ c),$$

respectively, and the interpretation I i defined as

$$Q_I : <, \qquad P_I : >, \quad c_I :$$

Restricted Quantifiers Laws

The formula

$$F: \quad (\forall x(B(x) \Rightarrow A(x)) \Rightarrow \exists x(B(x) \cap A(x)))$$

becomes a **mathematical statement**

$$F_I: \quad (\forall_{n \in N} (x < 0 \Rightarrow n > 0) \Rightarrow \exists_{n \in N}(n < 0 \cap n > 0))$$

The satement F_I is a **false**

because the statement $n < 0$ is **false** for all natural numbers and the implication $false \Rightarrow B$ is **true** for any logical value of B

Hence $\forall_{n \in N} (n < 0 \Rightarrow n > 0)$ is a **true** statement and $\exists_{n \in N}(n < 0 \cap n > 0)$ is obviously **false**

Restricted Quantifiers Laws

Restricted quantifiers law corresponding to the predicate tautology

$$(\forall x\, A(x) \Rightarrow \exists x\, A(x))$$

is

$$\models_p \ (\forall_{B(x)}\, A(x) \Rightarrow (\exists x\, B(x) \Rightarrow \exists_{B(x)}\, A(x)))$$

We remind that it means that we prove that the corresponding proper formula of $\mathcal{L}$ obtained by the restricted quantifiers **transformations rules** is a predicate tautology, i.e. that

$$\models_p (\forall x(B(x) \Rightarrow A(x)) \Rightarrow (\exists x\, B(x) \Rightarrow \exists x\, (B(x) \cap A(x))))$$

Quantifiers Laws

Another basic **predicate tautology** called a dictum de omni law is

$$\models_p \ (\forall x\, A(x) \Rightarrow A(y))$$

where $A(x)$ are any formulas with a free variable x and $y \in VAR$

The corresponding **restricted quantifiers law** is:

$$\models_p \ (\forall_{B(x)}\, A(x) \Rightarrow (B(y) \Rightarrow A(y))),$$

where $A(x),\ B(x)$ are any formulas with a free variable x and $y \in VAR$

Quantifiers Laws

The next important laws are the **Distributivity Laws**

Distributivity of existential quantifier over conjunction holds only in **one direction**, namely the following is a predicate tautology

$$\models_p \ (\exists x\,(A(x) \cap B(x)) \ \Rightarrow \ (\exists x A(x) \cap \exists x B(x))),$$

where $A(x), B(x)$ are any formulas with a free variable x

The **inverse** implication **is not** a predicate tautology, i.e.

$$\not\models_p \ ((\exists x A(x) \cap \exists x B(x)) \Rightarrow \exists x\,(A(x) \cap B(x)))$$

Quantifiers Laws

To **prove** it we have to find an example of **concrete** formulas $A(x),\ B(x) \in \mathcal{F}$ and a model structure $\mathbf{M} = (U, I)$ with the interpretation I, such that **M** is **counter- model** for the formula

$$F: \ ((\exists x A(x) \cap \exists x B(x)) \Rightarrow \exists x\ (A(x) \cap B(x)))$$

We define the **counter - model** for F is as follows

Take $\mathbf{M} = (R, I)$ where R is the set of real numbers

Let $A(x),\ B(x)$ be **atomic** formulas $Q(x, c),\ \P(x, c)$

We define the interpretation I as $Q_I : >, \quad P_I : <, \quad c_I : 0.$

The formula F becomes an obviously **false** mathematical statement

$$F_I : ((\exists_{x \in R}\ x > 0 \cap \exists_{x \in R}\ x < 0) \Rightarrow \exists_{x \in R}\ (x > 0 \cap x < 0))$$

Quantifiers Laws

Distributivity of universal quantifier over disjunction holds only on **one direction**, namely the following is a predicate tautology for any formulas $A(x), B(x)$ with a free variable x.

$$\models_p ((\forall x A(x) \cup \forall x B(x)) \Rightarrow \forall x\,(A(x) \cup B(x))).$$

The inverse implication **is not** a predicate tautology, i.e.

$$\not\models_p (\forall x\,(A(x) \cup B(x)) \Rightarrow (\forall x A(x) \cup \forall x B(x)))$$

Quantifiers Laws

To **prove** it we have to find an example of **concrete** formulas $A(x),\ B(x) \in \mathcal{F}$ and a model structure $\mathbf{M} = (U, I)$ that is **counter- model** for the formula

$$F:\ (\forall x\ (A(x) \cup B(x)) \Rightarrow (\forall x A(x) \cup \forall x B(x)))$$

We take $\mathbf{M} = (R, I)$ where R is the set of real numbers, and $A(x),\ B(x)$ are **atomic** formulas $Q(x, c),\ \ R(x, c)$

We define $Q_I :\geq$ and $R_I :<,\ c_I : 0$

The formula F becomes an obviously **false** mathematical statement

$$F_I : (\forall_{x \in R}\ (x \geq 0 \cup x < 0) \Rightarrow (\forall_{x \in R}\ x \geq 0 \cup \forall_{x \in R}\ x < 0))$$

Logical Equivalence

The most frequently used laws of quantifiers have a form of a **logical equivalence**, symbolically written as $\equiv$

Remember that $\equiv$ is not a new logical connective

This is a very useful symbol

It says that two formulas always have the same logical value

It can be used in the same way we the equality symbol $=$

Logical Equivalence

We formally define the **logical equivalence** as follows

Definition

For any formulas $A, B \in \mathcal{F}$ of the **predicate language** $\mathcal{L}$,

$$A \equiv B \quad \text{if and only if} \quad \models_p \ (A \Leftrightarrow B).$$

We have also a similar definition for the propositional language and propositional tautology

Equational Laws for Quantifiers

De Morgan

For any formula $A(x) \in \mathcal{F}$ with a free variable x,

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

Definability

For any formula $A(x) \in \mathcal{F}$ with a free variable x,

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

Equational Laws for Quantifiers

Renaming the Variables

Let $A(x)$ be any formula with a free variable x

and let y be a variable that **does not occur** in A(x).

Let $A(x/y)$ be a result of **replacement** of each occurrence of x by y, then the following holds.

$$\forall x A(x) \equiv \forall y A(y), \quad \exists x A(x) \equiv \exists y A(y)$$

Alternations of Quantifiers

Let $A(x, y)$ be any formula with a free variables x and y.

$$\forall x \forall y\ (A(x, y)\ \equiv\ \forall y \forall x\ (A(x, y),$$

$$\exists x \exists y\ (A(x, y)\ \equiv\ \exists y \exists x\ (A(x, y)$$

Equational Laws for Quantifiers

Introduction and Elimination Laws

If B is a formula such that B **does not contain** any free occurrence of x, then the following logical equivalences hold.

$$\forall x(A(x) \cup B) \equiv (\forall x A(x) \cup B),$$

$$\exists x(A(x) \cup B) \equiv (\exists x A(x) \cup B),$$

$$\forall x(A(x) \cap B) \equiv (\forall x A(x) \cap B),$$

$$\exists x(A(x) \cap B) \equiv (\exists x A(x) \cap B)$$

Equational Laws for Quantifiers

Introduction and Elimination Laws

If B is a formula such that B **does not contain** any free occurrence of x, then the following logical equivalences hold.

$$\forall x(A(x) \Rightarrow B) \equiv (\exists x A(x) \Rightarrow B),$$

$$\exists x(A(x) \Rightarrow B) \equiv (\forall x A(x) \Rightarrow B),$$

$$\forall x(B \Rightarrow A(x)) \equiv (B \Rightarrow \forall x A(x)),$$

$$\exists x(B \Rightarrow A(x)) \equiv (B \Rightarrow \exists x A(x))$$

Equational Laws for Quantifiers

Distributivity Laws

Let $A(x)$, $B(x)$ be any formulas with a free variable x

Distributivity of universal quantifier over conjunction.

$$\forall x\, (A(x) \cap B(x)) \equiv (\forall x A(x) \cap \forall x B(x))$$

Distributivity of existential quantifier over disjunction.

$$\exists x\, (A(x) \cup B(x)) \equiv (\exists x A(x) \cup \exists x B(x))$$

Equational Laws for Quantifiers

We also define the notion of logical equivalence $\equiv$ for the formulas of the **propositional language** and its semantics

For any formulas $A, B \in \mathcal{F}$ of the **propositional language** $\mathcal{L}$,

$$A \equiv B \quad \text{if and only if} \quad \models (A \Leftrightarrow B)$$

Moreover, we prove that any substitution of **propositional tautology** by a formulas of the predicate language is a **predicate tautology**

The same holds for the logical equivalence

Equational Laws for Quantifiers

In particular, we transform the **propositional tautologies** into the following corresponding predicate equivalences.

For any formulas A, B of the **predicate language** $\mathcal{L}$,

$$(A \Rightarrow B) \equiv (\neg A \cup B),$$

$$(A \Rightarrow B) \equiv (\neg A \cup B)$$

We use them to prove the following De Morgan Laws for **restricted quantifiers**.

Equational Laws for Quantifiers

Restricted De Morgan

For any formulas $A(x), B(x) \in \mathcal{F}$ with a free variable x,

$$\neg\forall_{B(x)}\, A(x) \equiv \exists_{B(x)}\, \neg A(x), \qquad \neg\exists_{B(x)}\, A(x) \equiv \forall_{B(x)} \neg A(x)$$

Here is a poof of first equality. The proof of the second one is similar and is left as an exercise.

$$\neg\forall_{B(x)}\, A(x) \equiv \neg\forall x\, (B(x) \Rightarrow A(x))$$

$$\equiv \neg\forall x\, (\neg B(x) \cup A(x))$$

$$\equiv \exists x\, \neg(\neg B(x) \cup A(x)) \equiv \exists x\, (\neg\neg B(x) \cap \neg A(x))$$

$$\equiv \exists x\, (B(x) \cap \neg A(x)) \equiv \exists_{B(x)}\, \neg A(x))$$