**Presentation on Predicate Logic and Connectives - Instructions**

**Defintion:** The purpose of Predicate Logic is to avoid ambiguity by forming formulae out of natural language.

We distinguish between World and Predicate Logic.

**Real World:** green (or italic in the course book)

**Predicate Logic/logical language:** orange (or bold in the course book)

**First we need to define our “World”**

1. **U** = {Elisabeth, Marthe, Isabelle, Lisa}
2. **Defining properties:**

brown-eyes: {x| x has brown eyes} = {Elisabeth}

blonde: {x| x blonde} = {<Marthe>,<Isabelle>}

female: {x| x is female} = U

male: {x| x is male) = {}

1. **Defining relations between individuals:**

Love-eachother: {<x,y> | x likes y} = {<Marthe, Isabelle>, <Isabelle,Marthe>}

Fight-with-each-other: {<x,y> | x fights with y}= {<Lisa, Isabelle>,<Isabelle,Lisa>}

Facebook-friends: {<x,y> | x is facebook friend of y}={<Elisabeth, Marthe>, <Marthe,Elisabeth>,<Isabelle,Marthe>,<Marthe, Isabelle>}

Older-than: {<x,y>|x is older than y} = {<Marthe,Lisa>,<Marthe,Isabelle>,<Marthe,Elisabeth>, <Lisa,Isabelle>,<Lisa,Elisabeth>,<Elisabeth,Isabelle>}

**Second: I Function**

The function I maps each name of the logical language to one individual from the universe. It maps individuals to names and properties and sets of individuals to relations. 🡪 Connection/Transformation between the real world and predicate logic/logical language.

**Names:**

I (elisabeth) = {elisabeth}

**Properties:**

I (female) = {<Elisabeth>, <Marhte>, <Isabelle>, <Lisa>}

**Relations:**

I (fight-with -each-other) = {<Lisa, Isabelle>.<Isabelle,Lisa>}

**Formulae:**

**Definition:** Formulae are expressions of logical language that can be interpreted as true or false according to the definition of our “World”. Formulae can be formed out of sentences of natural language.

Elisabeth and Lisa are facebook friends.

Facebook-friends (Elisabeth, Lisa)

[[ Facebook-friends (Elisabeht, Lisa)]] = false

**Task 1:**

Which of the following expressions are formulae? Why or why not?

Isabelle 🡪 no formulae, only a name

Love-each-other (Marthe) 🡪 no formulae, as “love-each-other” is a relation and therefore needs two individuals in brackets in order to be interpreted

Fight-with-each-other (Isabelle, Lisa) 🡪 formulae!

Female 🡪 no formulae, as “female” is a property and needs therefore one individual in brackets in order to be interpreted

Brown-eyes (Marthe) 🡪 formulae!

Blonde(Lisa,Isabelle) 🡪 no formulae, as “blonde” is a property and needs only one individual in brackets in order to be interpreted

**Interpretation of formulae**

Facebook-friends (Elisabeth, Lisa)

[[ Facebook-friends (Elisabeht, Lisa)]] = false,

because

I(elisabeth) = elisabeth,

I (lisa) = lisa

and <elisabeth, lisa> is NOT in the set of I(Facebook-friends).

**Task 2:**

Read this story and try to create two formulae out of it. Then interpret them according to our World as true or false.

These are Lisa, Marthe, Isabelle and Elisabeth with the brown eyes. Isabelle bought ugly shoes at Zalando. Therefore, Lisa and Isabelle are fighting. Elisabeth and Lisa are Facebook friends. They gossip about Isabelle´s ugly shoes. Marthe visits Isabelle and comforts her. She loves her despite her ugly shoes.

**Note:** In order to interpret your formula, you have to keep in mind the Model we have created in the beginning.

**Examples**:

Brown-eyes(elisabeth) = true,

because

I(elisabeth) = elisabeth and <elisabeth> is an element of I(brown-eyes).

Facebook-friends(elisabeth,lisa) = false,

because

I(elisabeth) = elisabeth,

I(lisa) = lisa,

and <lisa,elisabeth> is not in the set of I(Facebook-friends).

Fight-with-each-other(lisa,isabelle) =true

because

I(lisa) = lisa,

I(isabelle) = isabelle

and <lisa,isabelle> is in the set of I(fighting-with-each-other).

Love-eachother(marthe,isabelle) = true

because

I(marthe) = marthe,

I(isabelle) = isabelle,

And <isabelle,marthe)> are in the set of I(love-eachother).

**G Function**

The function g maps variables to individuals.

Facebook-friends (Elisabeth, Lisa)

Facebook-friends (x,y)

g(x)= Elisabeth

g(y)= Lisa

She is a facebook friend of her.

* In order to interpret that as right or wrong you need to know to whom the “She” and “her” refers. So you need the g function there.

**Logical Connectives**

**and**

Symbol: Ʌ   
Sentence: Lisa wears braces and Elisabeth is female.   
Formulae: [[blonde(lisa) Ʌ female(elisabeth)]] = true/false?

**Truthtables**

1=true, 0 = false

Satz 1 (p) Satz 2 (q) Satz 1 Ʌ Satz 2 (p Ʌ q)

1 1 1

1 0 0

0 1 0

0 0 0

**or**

Symbol: V   
Sentence: Lisa wears braces or Elisabeth is female.   
Formulae: [[blonde (lisa) V female-(elisabeth)]] = true/false?

Satz 1 (p) Satz 2 (q) Satz 1 Ʌ Satz 2 (p V q)

1 1 1

1 0 1

0 1 1

0 0 0

**if/then**

Symbol: 🡪  
Sentence: If Lisa wears braces then Elisabeth is female.  
Formulae: [[blonde(lisa) 🡪 female(elisabeth)]] = true/false?

Satz 1 (p) Satz 2 (q) Satz 1 🡪 Satz 2 (p 🡪 q)

1 1 1

1 0 0

0 1 1

0 0 1

**not**

Symbol: ¬   
Sentence: Lisa wears no braces.   
Formulae: [[¬blonde (lisa)]] = true/false?

The original formulae has to be false, that the overall statement is true.

If blonde(lisa) is false, ¬blonde(lisa) has to be true.

**Interpretation of formulae with connectives**

Lisa wears braces or Elisabeth is female.

Formulae: [[blonde(lisa) V female(elisabeth)]] = ?,

🡪To interpret the whole sentence, each part of the sentence needs to be interpreted alone first.

**First sentence**

Lisa is blonde.

[[blonde(lisa)]] = false,

because

I (lisa) = lisa

and lisa is not an element of I(blonde).

**Second sentence**

[[female(elisabeth)]] = true,

because

I (elisabeth) = elisabeth

and <elisabeth> is an element of I(female).

**First sentence:** false

**Second sentence**: true

**Whole sentence:** true