**PH133 Logic**  Lecture 8

Lecturer: s.butterfill@warwick.ac.uk

**Quantifier equivalences**

P→Q ╡╞ ¬Q→¬P

∀x ( Square(x) → Broken(x) )

╡╞ ∀x (¬Broken(x) → ¬Square(x) )

P→Q ╡╞ ¬P∨Q

∀x ( Square(x) → Broken(x) )

╡╞ ∀x (¬Square(x) ∨ Broken(x) )

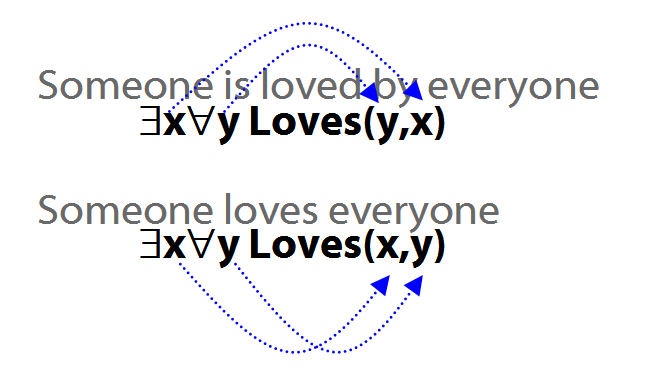
¬P∨Q ╡╞ ¬(P∧ ¬Q)

∃x ¬F(x) ╡╞ ¬∀x F(x)

∀x (¬Square(x) ∨ Broken(x) )

╡╞ ∃x¬( Square(x) ∧ ¬Broken(x) )

∃x ¬Created(x) ╡╞ ¬∀x Created(x)



Quantifiers and identity

Ahura Mazda created everything:  
∀x Created(a,x)

Ahura Mazda created everything apart from himself: ∀x ( ¬(x=a) → Created(a,x) )

**Quantifiers and negation**

Some person is dead.

∃x(Person(x) ∧ Dead(x))

Some person is not dead.

∃x(Person(x) ∧ ¬Dead(x))

No person is dead.

¬∃x(Person(x) ∧ Dead(x))

Every person is dead.

∀x(Person(x) → Dead(x))

Every person is not dead.

∀x(Person(x) → ¬Dead(x))

Not every person is dead.

¬∀x(Person(x) → Dead(x))

Multiple quantifiers

‘There is a store for everything’

∃y∀x StoreFor(y,x)

∀y∃x StoreFor(x,y)

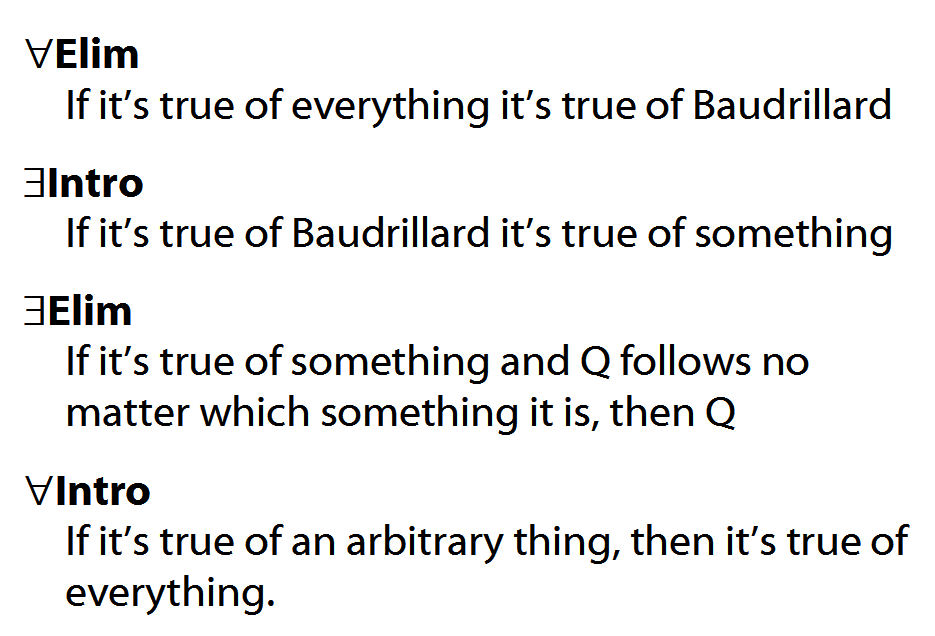
Other sentences to translate:

‘Wikipedia has an article about everything’

‘Everyone hurts someone they love’

‘Someone hurts everyone she loves’

Intuitve summary of quantifier rules



Number

There are at least two squares:  
∃x ∃y ( Square(x) ∧ Square(y) ∧ ¬x=y)

At least two squares are broken:  
∃x ∃y ( Square(x) ∧ Broken(x) ∧ Square(y) ∧ Broken(y) ∧ ¬x=y)

There are at least three squares:  
∃x ∃y ∃z ( Square(x) ∧ Square(y) ∧ Square(z) ∧ ¬x=y ∧ ¬y=z ∧ ¬x=z)

There are at most two squares:   
¬There are at least three squares  
¬∃x ∃y ∃z ( Square(x) ∧ Square(y) ∧ Square(z) ∧ ¬x=y ∧ ¬y=z ∧ ¬x=z)

There are exactly two squares:  
There are at most two squares ∧ There are at least two squares

Number: Alternatives

There is at most one square:   
∀x∀y( (Square(x) ∧ Square(y)) → x=y )

There are at most two squares:   
∀x∀y∀z( (Square(x)∧Square(y)∧Square(z))  
 →   
 (x=y∨y=z∨x=z)   
 )

There is exactly one square:   
∃x ( Square(x) ∧ ∀y( Square(y) → x=y ))

There are exactly two squares:   
∃x∃y (Square(x) ∧ Square(y) ∧ ¬x=y ∧  
 ∀z( Square(z) → (z=x ∨ z=y) ))

The

‘The’ can be a quantifier, e.g. ‘the square is broken’. How to formalise it?

The square is broken

There is exactly one square and it is broken

There is at most one square and there is at least one square and it is broken

There is at most one square and there is at least one square and all squares are broken

¬ ∃x ∃y ( Square(x) ∧ Square(y) ∧ ¬x=y )   
∧ ∃x Square(x)   
∧ ∀x ( Square(x) → Broken(x)