Lesson #6: Boolean Connectives (Sections 3.1 - 3.3)

<u>Assigned reading pp. 67-78</u>

POWERPOINT SLIDE #1

We've discussed the role of both *predicates* and *terms* in an FOL, but so far we've only dealt with *atomic* sentences (i.e., where a single predicate takes one or more terms as its arguments). To create more complex sentences in FOL, we need to introduce a third type of component of logical languages: **Boolean connectives**.

POWERPOINT SLIDE #2

There are *three* basic Boolean connectives: **negation**, **conjunction**, and **disjunction**. Here's a simple table showing how they compare:

Connective	Symbol	English equivalent
Negation	¬	***not***, *it is not the case that, non-, un-*
Conjunction	∧	***and***, *moreover, but*
Disjunction	∨	**or**

These connectives 'connect' multiple atomic sentences together, or otherwise create a complex FOL sentence out of what (without the connective) was a simple atomic sentence. This is easiest to see with **conjunction** and **disjunction**; for example:

POWERPOINT SLIDE #3

Tet(a) ∧ Cube(b) *translation:* "*a* is a tet and *b* is a cube"
Between(a,d,e) ∨ Between(a,d,f) *translation:* "*a* is between *d* and *e* or it's between *d* and *f*"

POWERPOINT SLIDE #4

In the case of **negation**, the result of negating an atomic sentence is also a complex sentence, but a special sort of (simpler) complex sentence that we still find useful to sometimes group with atomic sentences as both being '***literals***' (i.e., **literals** can be either **atomic sentences OR** the **<u>negation of</u> atomic sentences**).

Small(a) *translation:* "*a* is small"
¬Small(a) *translation:* "*a* is not small"

Negation can also apply to complex sentences:

¬(Tet(a) ∧ Cube(b)) *translation:* "It's not the case that (both) *a* is a tet and *b* is a cube"
¬¬Small(a) *translation:* "It's not the case that *a* isn't small" (i.e., *a* is small)

The book mentions some other **peculiarities of the Boolean connectives** that make them different from their normal English counterparts.

POWERPOINT SLIDE #5

- You just saw that FOL allows *multiple negatives* stacked together (English would normally not allow this; *"I don't not want no more cookies").

POWERPOINT SLIDE #6

- Conjunction and disjunction apply in FOL *to entire atomic sentences* (not just to simple nouns or verbs, as in "Thom and Jonny play guitar" and "Fred slipped and cracked his head." In FOL:
  PlayGuitar(Thom) ∧ PlayGuitar(Jonny)
  Slipped(Fred) ∧ CrackedHead(Fred)

POWERPOINT SLIDE #7

- When an *object has multiple properties*, FOL has to break these out as separate predicates in separate atomic sentences, whereas English doesn't:
  Intelligent(Neil) ∧ Handsome(Neil) ∧ Young(Neil) ∧ Man(Neil) vs.
  "Neil is an intelligent, handsome, young man."

POWERPOINT SLIDE #8

- Disjunction always has an '*inclusive*' sense in FOL, never 'exclusive'.

  Bob or Tony smashed the bug.

  Typical English 'exclusive' sense = implies that only *one* of the men applied fatal pressure to the bug.

  Smashed(bob, bug) ∨ Smashed(tony, bug)

  FOL 'inclusive' sense = could be that one man applied the fatal pressure, or could be that *both* men did.

POWERPOINT SLIDE #9

So there are ***seven*** different ways for the sentence Dodec(a) ∨ Dodec(b) ∨ Dodec(c) to turn out true!

POWERPOINT SLIDE #10

Note that the *negation of a disjunction* may be translated with ***neither … nor***:

¬(Tet(a) v Cube(a))

"Object *a* is neither a tet nor a cube"

POWERPOINT SLIDE #11

There is one other important thing you need to remember from today's reading:

POWERPOINT SLIDE #12

Truth Tables:

Truth tables highlight an important fact about the Boolean connectives, namely, that they are ***truth-functional*** connectives, meaning that the *truth value of the complex sentence* formed when these connectives join together two or more atomic sentences *depends strictly on a* ***combination of the truth values of the atomic sentences*** *from which it is built*.

POWERPOINT SLIDE #13

For example, here is the truth table for **disjunction:**

P	Q	P v Q
TRUE	TRUE	TRUE
TRUE	FALSE	TRUE
FALSE	TRUE	TRUE
FALSE	FALSE	FALSE

The way you read such a table is as follows: Look at what is under the heading '**P**' (which is a label for some atomic sentence; it doesn't matter exactly what the atomic sentence says, it could be anything like Cube(a) or LeftOf(b,c) . . . it doesn't matter, so we'll just label it 'P' here). In the column under 'P' there are four truth values . . . *representing four different worlds or situations*, two worlds in which atomic sentence P is 'True' and two worlds in which atomic sentence P is 'False'. Now, notice that the other atomic sentence, labeled '**Q**', also has four truth values under it (i.e., the middle column of the table), again two worlds for which Q would be true and two worlds for which it would be false. Notice also that the truth values in the four rows under P and Q (remember that rows run left to right) cover all four of the possible combinations of P and Q in terms of their truth values: the top row represents the world or situation in which P and Q are both true. The row under that represents the world or situation in which P is true but Q is false. The third row represents the world where P is false but Q is true. Finally, the bottom row represents the world where both P and Q are false. Keeping all that in mind, look at the last column