Predicate Logic

The Formation Rules of Predicate Logic

The Language of Predicate Logic

- It's important to understand that predicate logic is not a brand new language, but an extension of propositional logic.

- That means not only that all of the inference rules from propositional logic will apply, but that we can still use our letters P, Q, R, S, etc. to stand for entire propositions if we want to.

- So, for example, if P is understood to stand for an entire proposition, these are going to be perfectly well formed sentences in predicate logic.

  - $P \rightarrow \forall x Fx$
  - $\exists y(P \vee \forall x Fxy)$

- We would produce a translation like this when the English proposition P represents doesn't have a predicate logic equivalent.

Vocabulary of Predicate Logic

- So this is the complete vocabulary of Predicate Logic:
  - Sentence letters: P, Q, R, S, P_1, Q_1,…
  - n-ary predicates: A, B, C…O, $A_{1…}$
  - names: a, b, c, ..., t, a_1, b_1…
  - Individual variables: u, v, w, x, y, z, t_1, u_1…
  - Sentential connectives: ~, &, v, →, ↔
  - Quantifiers ∀, ∃
  - Grouping Indicators (,)

Formation Rules

- And these are the formation rules.

1. Any atomic formula is a wff.*
2. If $\mathcal{A}$ is a formula, then ~$\mathcal{A}$ is a formula.
3. If A and B are formulas, then $(\mathcal{A} \rightarrow \mathcal{B})$, $(\mathcal{A} \vee \mathcal{B})$, $(\mathcal{A} \,\&\, \mathcal{B})$, $(\mathcal{A} \leftrightarrow \mathcal{B})$ are formulas.
4. If $\mathcal{A}c$, is a formula with name c, and v is a variable that does not occur in $\mathcal{A}c$, then $\exists v \mathcal{A}v$ and $\forall v \mathcal{A}v$, are formulas.
5. Every formula can be constructed from a finite number of applications of these rules.

- *Remember that we have two kinds of atomic formulas. Propositional logic formulas are just capital sentence letters. Predicate logic formulas are predicates followed by names.

Wffs. in predicate logic

- The most important thing to understand about the formation rules is what they tell you about the relation between variables and names.
- Rule 4 tells you that something with a quantifier in it is a wff. if you can begin with a wff. that contains a name, and then substitute a variable for that name and put one of the quantifiers followed by that variable at the beginning of the formula. And Rule 5 tells you that the only things that are wffs. are those that can be constructed by this process.
- So, e.g., this formula is a wff: ∀x(Fx v ∃zGzx). And the reason is that it can be proven to be a wff from the rules as follows.

1. Fa is a wff	(Rule 1)
2. Gba is a wff	(Rule 1)
3. ∃zGza is a wff	(Rule 4, line 2)
4. (Fa v ∃zGza) is a wff	(Rule 3, line 2 and 3)
5. ∀x(Fx v ∃zGzx) is a wff	(Rule 4, line 4)

Wffs. in predicate logic 2

- On the other hand, this is not a wff:

  - ∀xFxy

  because there is no way to produce it from the formation rules. For example, this is legitimate:

  1. Fab (Rule 1)
  2. ∀xFxb (Rule 4, line 1)

  - But no rule would allow you to introduce y here for b without also introducing a quantifier.

Wffs. in predicate logic 3

- We also can't have multiple quantifiers binding the same variable. This is not a wff:

  - $\exists x \forall x Fxx$

  again because there is no way to produce it. (This is always the reason.) An attempt would like this:

  1. Faa (Rule 1)
  2. $\forall x Fax$ (Rule 4, line 1)

- But at this point another application of Rule 4 requires you to introduce a variable not already in the formula you are operating on.

Wffs. in predicate logic 4

- You know that quantifiers bind variables, not names. And now you can say why by reference to the formation rules.
- This is not a wff
  - ∀xFa

  because rule 4 would require you to substitute a variable for ‘a’ when you introduced the quantifier.
- You also know that we never substitute variables for predicates:
  - ∀xXb
- this is just because there is no rule that permits you to do so.

Some semantic clarifications

- There are some other important things to understand about the meaning of predicate logic statements. First, formulas containing multiple variables and quantifiers can result from formulas containing multiple names, like so:
- Lab >>> ∃yLay >>> ∀x∃yLxy
- But they can also be constructed from formulas containing the same constant. For example, this series of substitutions is perfectly permissible:
- Laa >>> ∃yLay >>> ∀x∃yLxy
- So when you see a formula containing multiple variables, don't assume that the formulas must have originated from formulas containing different names.

Some semantic clarifications 2

- A related point is that the fact that the same variables are used with two different quantifiers doesn't mean that the same object is being designated in each case. For example, this formula:
- (∃xLx & ∃xFx)
- might be a translation of the English sentence: "Something is lucky and something is funny." The English sentence suggests to us that there are two distinct things being indicated but that is not necessarily case. The thing that is funny might be the very same thing that is lucky. We can also see this by reference to the formation rules:

1.	La	(Rule 1)
2.	∃xLx	(Rule 4, line 1)
3.	Fa	(Rule 1)
4.	∃xFx	(Rule 4, line 3)
5.	(∃xLx & ∃xFx)	(Rule 3, lines 2 and 4)

Some semantic clarifications 3

- The order of quantifiers in a wff. doesn't affect the meaning if they are all the same quantifier.

- $\forall x \forall y Lxy$ is equivalent to $\forall y \forall x Lxy$
- $\exists y \exists x Lxy$ is equivalent to $\exists x \exists y Lxy$

- But the order <u>does</u> matter when the quantifiers are different. For example:

- $\forall x \exists y Lxy$ is <u>not</u> equivalent to $\exists y \forall x Lxy$

- If L means 'loves', these two formulas would translate respectively into English as:

- Everything loves something.
- There is something that everything loves.

Some semantic clarifications 4

- Finally, whenever you have an expression with quantifiers that are distributed throughout a formula, they are always equivalent to one in which the quantifiers are all on the outside of the formula. E.g., these are equivalent expressions

  - $\exists x(Fx \,\&\, \forall y(Fy \rightarrow Gy))$
  - $\exists x\, \forall y(Fx \,\&(Fy \rightarrow Gy))$

- When doing translations, distributing the quantifiers as in the first formula often makes them easier to understand. For example, these formulas both could represent the English sentence:

- Something is fried, and anything that is fried is good.

- But this is more obviously true of the first formula than the second. When quantifiers are all placed on the outside as in the second formula, they are said to be in <u>normal</u> form.