Lecture 5. Logic

Friedrich Ludwig Gottlob Frege (b. 1848, d. 1925) was a German mathematician, logician, and philosopher who worked at the University of Jena. Frege essentially reconceived the discipline of logic by constructing the first predicate calculus. In this formal system, Frege developed an analysis of quantified statements and formalized the notion of a proof in terms that are still accepted today. Frege then demonstrated that one could use his system to resolve theoretical mathematical statements in terms of simpler logical and mathematical notions.

Quoted from `http://plato.stanford.edu/entries/frege/#BasFreTerLogPreCal`

The modern version of Frege's system is called "first-order logic." It is the main and most important topic in mathematical logic. First-order logic is an artificial language that models many of the essential features of the conventional language of mathematics.

Why am I suggesting that math teachers study this? When teaching mathematics, we need to pay attention to the meanings that learners are trying to convey and to the things they say and write in order to convey those meanings. We also need to pay attention to the conventions that govern the standard language of mathematics and the extent to which learners conform to it. In an attempt to express a reasonable idea, learners often use symbols in ways that ignore or violate conventions and therefore fail to communicate. It is the job of the teacher to see that students master the standard language.

Logic provides a model of the standard language, in which all parts—the symbols, the rules for combining them and the ways to interpret their meanings—are stated explicitly, like the rules for a a board game such as monopoly or checkers. By making the rules explicit in an artificial "language game," first order logic provides the teacher with a useful tool—a rigid and perfectly precise model to which other language can be compared. No one actually speaks first order logic, of course. The language that we use when we do mathematics is just an extension of ordinary English. But logic helps us to understand that language through comparison to an artificial one.

In order to distinguish expressions that we make within the artificial language of logic from expressions that we make in everyday mathematical talk (or writing), we call the former "formal symbols," "formal expressions," "formal sentences," etc.

Symbols

The basic building blocks of the language of first order logic are symbols of various kinds. The first family of symbols includes those from which we build expressions.

- *Constant symbols.* These are symbols that denote specific mathematical objects. The symbols displayed on the following line are constant symbols.

$$0 \qquad 1 \qquad 2 \qquad 3 \qquad 50 \qquad 2.17 \qquad \pi$$

- *Variable symbols.* These are letters such as x, y, z that may appear in expressions in the places where we might expect to see numbers.
- *Function symbols.* The most commonly used function symbols are $+$ (the addition symbol), and $\times$ (the multiplication symbol). Every function symbol has blank spaces associated with it that are filled in when the symbol is used. For example, the sign $+$ has a blank space before

and after it. (If we could see them, the blank spaces would look like this: $\_\_ + \_\_$.) We typically fill in those spaces with constant symbols or variable symbols, e.g., $0+1$, $1+x$, $x+y$. Other commonly used function symbols (with blank spaces shown) include:

- The absolute-value symbol: $|\_\_|$
- The symbol for the natural logarithm: $\ln(\_\_)$
- Exponentiation with base b: b^{-}

Now we can make a precise definition of what an expression is. *An expression is a string of symbols which is either a constant symbol or a variable symbol or is created from other expressions by placing them "in the blanks" of a function symbol.*

Examples of expressions

- 1 (a constant symbol)
- x (a variable symbol)
- $1+x$ (the symbols above placed in the blanks of the function symbol $+$)
- $(1+x) \times (2+y)$ ($1+x$ and $2+y$ placed in the blanks of the function symbol $\times$)
- $\big((1+x) \times (2+y)\big) + \big((1+x) \times (3+z)\big)$ (two expressions placed in the blanks of $+$)
- $\Big(\big((1+x) \times (2+y)\big) + \big((1+x) \times (3+z)\big)\Big) \times (5+w)$

Comments on constant symbols

- A constant symbol is NOT the same thing as a constant. The word "constant" is sometimes used to refer to symbols and sometimes it refers to numbers. The term "constant symbol" refers to a symbol that is used to denote a number. A symbol is not a number.
- Each constant symbol denotes a specific thing. The constant symbol "3" denotes a number, which we typically envision as lying on the number line, three units to the right of 0.
- When talking about constant symbols, we need to be careful to distinguish between the symbol and the thing it denotes. Some people—and some software programs, such as *Mathematica*—use quotation marks to make the distinction clear: "3" is a symbol; 3 is a number. The symbol "3" is a name for the number 3. "Lincoln" is a name symbol; Lincoln was a president. The symbol "Lincoln" is the name of the president.
- How do you know if a symbol is a constant symbol? There are some symbols that are almost always used that way: 1, 2, 3, etc. We can add to this list, if we wish, for anyone is free to create his or own constant symbols. For example, you might decide to use K to stand for $1,000$. Obviously if you create some new constant symbols, you had better inform the people with whom you hope to communicate.
- Sometimes we create constant symbols whose reference is defined indirectly. For example, let a denote the number which, when added to 5 gives 9.
- Constant symbols may refer to things other than numbers (e.g., lines, planes, transformations). However, we will restrict attention to numbers; unless we say otherwise, we will assume that all constant symbols are the names of numbers.

Comments on variable symbols

- In school math, when variable symbols are first introduced they are sometimes described as "place-holders." This is exactly idea that occurs in logic. A variable symbol is simply a symbol that stands in an expression to keep a blank space in a function symbol open.

- Variable symbols don't denote anything. This seems to be a stumbling block for learners who cannot use x without wondering what x denotes. Symbols that do *not* denote anything but that occur in the same contexts as constant symbols (which always do denote something) are hard to get used to.
- The Common Core State Standards for grade six (page 44) say that students will "[u]se variables to represent numbers and write expressions when solving a real-world or mathematical problem; understand that a variable can represent an unknown number, or, depending on the purpose at hand, any number in a specified set." This quotation shows that the Common Core Standards use the word "variable" with a broader range of meanings than "variable symbol." If I say, "w is a real number and $w^3 = w+1$," then is w a variable? According to the statement from the Common Core, the answer would appear to be, "Yes.". But $x^3 = x + 1$ has only one real solution, so w refers to a specific number, and therefore "w" is a constant symbol.

Comments on function symbols

- Function symbols stand for—surprise—*functions*. A function is a rule that assigns outputs to inputs. For example, addition takes a pair of numbers as inputs and produces their sum as an output. When we fill in the blanks of a function symbol with constant symbols, we create a new expression that denotes the output that the function produces when it is fed the inputs that those constants denote. For example, "$3 + 5$" is an expression that denotes 8.
- In the language of first order logic, function symbols do not occur meaningfully by themselves. To create a meaningful phrase, we need to fill in the blanks.

Real-World Applications

Both the invention of programmable computers in the 1940s and the design of programming languages (which began in the 1940s and continues to the present) were dependent on mathematical logic. A more focused application of the ideas presented in this lecture is in the design of input interfaces for calculators and symbolic processing software.

A pocket calculator must respond to the keystrokes of the user to produce the calculation that the user intends. Suppose we push the following buttons:

$$\boxed{1}\,\boxed{2}\,\boxed{+}\,\boxed{3}\,\boxed{\times}\,\boxed{4}\,\boxed{=}$$

Then the screen will show 60. In contrast, under the standard order of operations:

$$12 + 3 \times 4 = 12 + 12 = 24$$

Problems

1. Why doesn't a pocket calculator use the standard order of operations?
2. When using a pocket calculator, what key strokes would you push to compute:
$$(34)(47) + (55)(79) + (26)(88)?$$
3. A standard pocket calculator accepts digits one at a time as a way to input multi-digit numbers. The digits go into the input buffer. When an operation key is pushed, the calculator recognizes the the digits in the buffer as a multi-digit number, and it then proceeds to operate with that number in a way that is dependent upon the internal state. What are the actual rules that a the standard inexpensive pocket calculators presently use? See `http://en.wikipedia.org/wiki/Calculator_input_methods`.
4. What is "Polish notation"? Why was it a useful format for input to calculators (before scientific calculators became widely available)?