

1. There's an interesting logic colloquium at Berkeley this Friday: [click here](#).
2. In this outline we start *using set theory to study logic*
3. *Syntax and semantics*. First I'll show how we can formulate many *syntactical* notions in terms of basic set theory. They have to do with the grammar of phrases constructed in specified ways from a specified collection of symbols. They have nothing to do with possible *meanings* of those phrases. Handling those is called *semantics*. We'll do that later.
4. *Strings*
 - a. Choose any set A , which we'll call our *alphabet*; its members we call our *symbols*.
 - b. If $n \in \mathbf{N}$ and $s : \{m \in \mathbf{N} : m < n\} \rightarrow A$, then s is called a *string* of symbols, and n , its *length*. Thus, the length of a string is the cardinal of its domain.
 - c. The set of all strings is called A^* .
 - d. An element $\langle m, a \rangle$ of a string $s \in A^*$ is called an *occurrence* of the symbol a in s .
 - e. $\phi \in A^*$.
 - f. $a \mapsto a^* = \{\langle 0, a \rangle\}$ is a bijection from A to the set of strings of length 1. It's hard to remember to distinguish between a and a^* , and most authors don't.
 - g. If s, t are strings of lengths m, n , then their *concatenation* is the string $r = s \cdot t$ of length $m + n$ defined by setting $r_i = s_i$ when $0 \leq i < m$ and $r_{m+j} = s_j$ when $0 \leq j < n$. This word stems from the Latin *catena* for *chain*. We often omit the dot in $s \cdot t$, thus denoting concatenation by juxtaposition.
 - h. $\phi s = s = s\phi$ for all $s \in A^*$.
 - i. $s(tu) = (st)u$ for all $s, t, u \in A^*$. We often omit the parentheses in these expressions.
 - j. $\langle A^*, \cdot, \phi \rangle$ is a semigroup with identity, generated by its subset A . It has properties that entitle it to be called *the free semigroup with identity with $\#A$ generators*. I won't use any of those properties unless I mention them.
5. As symbols for expressing some sentences in Boolean logic I choose an alphabet A that contains
 - a. a specific symbol for denoting implication,
 - b. some other symbols.I won't specify these symbols further until I have to. Just think of them as some sets, used as programmers use various codes in low-level programs. I won't even say how many variables I must have, until I need to. Later, as we need additional symbols for logical manipulations, I'll instruct you to consider new, larger, alphabets.
 - c. The string whose single member is an occurrence of the implication symbol is denoted here by $\Rightarrow$.

- d. A string whose single member is an occurrence of one of the other symbols is called a *variable*.
- 6. The set $F \subseteq A^*$ of *implicational formulas* is defined recursively:
 - a. All variables are implicational formulas.
 - b. If p, q are implicational formulas, then $\Rightarrow pq$ is an implicational formula.Recursive definitions like this are common. It means that F is the smallest set meeting requirements (a,b). Clearly, *some* set does: the set A^* itself; F is thus the intersection of the family of all sets meeting those requirements.
- 7. I will often write $(p \Rightarrow q)$ for $\Rightarrow pq$, and often omit the parentheses. But remember, I do so only in the English of these notes, which is independent of the symbols under discussion. I never have to write those—programmers do that—so I haven't even specified them!