## CARNAP ON LOGICAL SYNTAX

*The Logical Syntax of Language.* By Rudolf Carnap. (International Library of Psychology, Philosophy, and Scientific Method.) Translated from the German by Amethe Smeaton (Countess von Zeppelin). New York, Harcourt, Brace, 1937. 16 +352 pp.

The logical syntax of a symbolic language is a study of the formal properties of sentences of that language. It includes the *formation rules* which determine how the symbols of the language can be combined to form sentences, the *transformation rules* which specify when one sentence of the language can be deduced from other sentences, and the other properties of and relations between sentences which can be defined on the basis of these rules. Syntax is a combinatory analysis of expressions, that is, of finite ordered series of symbols. Hence syntax never refers to the meaning of these symbols. Hilbert showed that a clear, formal presentation of the foundations of mathematics must use a metamathematics which is really a syntax of mathematics. The notions of syntax are of central importance for the current growth of mathematical logic.

The present book systematically develops the concepts of syntax, first for two specific Languages I and II, then for an arbitrary language. The specific Language I is a definite ("constructivist" or "finitist") language. It contains the usual calculus of propositions (not, and, implies, $\cdots$ ) and a Peano arithmetic, with a symbol for 0 and for successor, and with the customary axioms. Variables representing numbers are included, but the quantifiers like "there exists an $x$" occur only in a limited form, such as "$(\exists x)3(P(x))$," meaning "there exists an $x$ with $x\leqq 3$ such that $P(x)$," and "$(Kx)5(Q(x))$," denoting the smallest $x\leqq 5$ with the property $Q$.

Language II is a much richer language, and contains everything usually included in a symbolic logic: all of Language I, plus variables for sentences (that is, propositions), variables for predicates, and variables for functors. Such "functors" are functions with any number of arguments of any type. Quantifiers "there exists an $x$" and "for all $x$" are used with all these variables. The predicates, which serve also as classes, are classified by the usual (unbranched) type theory, so that a class of numbers is of lower type than a class of classes of numbers. The language so obtained is of interest because it strives for a maximum of flexibility and not, as is often the case, for a minimum of primitive ideas.

Such symbolic languages are ordinarily restricted to symbols defined by means of the primitive symbols of logic and mathematics. Here, in order to make clearer the nature of language and to prepare for a subsequent discussion of the language of science, Carnap allows Languages I and II to contain not only predicates defined in logical terms, but also *descriptive* predicates and functors. One such descriptive symbol is the temperature functor "*te*," which is to be used so that "$te(3)=5$" means "the temperature at the position 3 is 5." Carnap contends that all sentences of physics can be similarly rendered by a "coordinate" language in which the basic symbols are numbers and not names. The general contention seems to neglect the necessity of specifying by name the coordinate system and the scale of measurement to be used.

The syntax of Languages I and II includes the definitions of such important terms as "directly derivable," "demonstrable," and "refutable." In Language I, the specifications under which one sentence is directly derivable from other sentences include the usual rule, that "$A_2$" and "$A_2$ implies $A_3$" give "$A_3$", in the following form: If the sentence $S_1$ consists of a partial sentence $S_2$ followed by an implication symbol followed by a partial sentence $S_3$, then $S_3$ is directly derivable from $S_1$ and $S_2$. A *derivation* is a finite series of sentences, such that every sentence of the series is either one of the primitive sentences, or a definition-sentence, or is directly derivable from sentences which precede it in the series. A sentence $S$ is *demonstrable* if there is a derivation in which $S$ is the final sentence. A sentence $S$ is *refutable* if each free variable-symbol in $S$ can be replaced throughout $S$ by a constant number-symbol in such a way that the negate of the resulting sentence is demonstrable. These definitions of syntactical terms may indicate how syntax has to do only with the order and arrangement of symbols into expressions, sentences, and groups of sentences.

These syntactical terms all have to do with an enumerable set of objects, the expressions of the language. If a fixed correlation of these expressions to the natural numbers is chosen, then each syntactical property of expressions becomes a property of the corresponding natural numbers, and can usually be defined by a recursive definition. But natural numbers and recursive definitions can be formulated within the symbolism of Language I (or II). Hence the syntax of either language can be arithmetically formulated within that language. This arithmetized syntax, due to Gödel, makes possible the construction of an arithmetic sentence which, syntactically interpreted, asserts its own indemonstrability. If the language is consistent,* this sentence can be neither demonstrable nor refutable. This discussion in English of Gödel's theorem and its striking consequences should prove valuable to many readers.

Logical positivists formerly distinguished between logic (including mathematics) and empirical science, on the ground that the sentences of logic are always resoluble (either demonstrable or refutable), while sentences of science need not be resoluble (on the basis of logical rules). Gödel's construction of a mathematical sentence which is neither demonstrable nor refutable made this distinction untenable. Apparently in order to reintroduce the distinction, Carnap defines a class of *analytic* sentences, wider than the class of demonstrable sentences. A class of contradictory sentences is also defined, the fundamental result being the theorem that every sentence built up out of logical symbols only is either analytic or contradictory. The definition of "analytic" in Language II is involved, since it includes an (apparently extraneous) reduction process due to the Hilbert school. The more essential features can be illustrated by the (demonstrable) sentence

$$(F)\ (\exists x)\ (F(0) \vee \sim F(x)).$$

By the definition, "this sentence is analytic" can be shown to mean† "for every class $B$ of number symbols there is at least one number symbol such that either this number symbol does not belong to $B$, or else the symbol 0 belongs to $B$." The latter sentence seems to be practically a translation of the given sentence into the auxiliary language which is being used for syntax. Similar results hold for other sentences. Thus, to prove that the principle of mathematical induction in Language II is analytic, Carnap must assume the same principle in the syntax language. The statement that a certain sentence is analytic amounts essentially to a careful statement, in the syn-

* Gödel's proof requires also that the language be "$\omega$-consistent." Carnap's discussion slurs over this point. J. B. Rosser has since shown how the assumption of $\omega$-consistency can be avoided, *Extensions of some theorems of Gödel and Church*, Journal of Symbolic Logic, vol. 1, p. 87.

† The definition itself does not use the "either $\cdots$ or" in the syntax language, but instead the usual truth value table.

tax language, of the usual "meaning" of the sentence. Furthermore, the definition of "analytic in language $S$" would seem to require a syntax language at least as strong as the language $S$ being studied. The utility of this notion of "analytic" thus might be open to doubt, especially as Carnap has adduced no positive evidence of the impossibility of a construction according to Gödel of a sentence asserting its own non-analyticity.

Carnap next turns to the difficult and far reaching subject of general syntax. This is a syntactical investigation of any symbolic language whatever. The methods used are essentially those of abstract mathematics. Of the language it is assumed only that there are rules of formation and rules of direct consequence. The latter rules specify when a sentence $S$ is a direct consequence of a class $K$ of sentences. In the special case when the class $K$ is finite, we have a rule of inference of the usual sort. On this narrow basis Carnap succeeds in defining many different general syntactical terms applying to the language: variable, constant, universal operator, arithmetic in a language, predicate, functor, translation into another language, the level of a symbol (as in the theory of types), and the like.

Variables, it has long been recognized, are not variable *things* of some mysterious sort; they are rather symbols for which, under certain circumstances, various other symbols, called "constants," may be substituted. Carnap gives a detailed analysis of this situation, defining such terms as "variable expression," "open expression," "variable," and "constant" in any language.* Here an expression is "open" if it contains at least one free variable. Following the definition of variable expression in Language I, we note that if the symbol "0" in the sentence "$0=0$" be replaced throughout by any other symbol for a constant number, the result is still a demonstrable sentence. According to the definition given, this fact apparently makes the symbol "0" a variable expression.† Furthermore, the sentence "$0=0$" turns out to be an open sentence, contrary to Carnap's previous usage in Language I.

To determine, by the definition, whether a symbol is a variable, one must know all the other variables of the language. This is because one cannot substitute an expression for a variable if the expression contains some other variable which would become bound (governed by a quantifier) after the substitution. This would indicate that Carnap's definition does not define the phrase, "this symbol is a variable in the given language." It defines rather "this class of symbols can be considered as a class of variables." There might well be many such classes of variables in a language, and in this event the term "variable" and other terms defined from it would have no fixed meaning.

Another fundamental concept of general syntax is that of the "logical" sentences of a language $S$. In Language I and II Carnap classes the usual primitive symbols of logic and mathematics, plus all symbols defined exclusively in terms of these primitive symbols, as *logical* symbols. In general syntax this classification by enumeration is to be replaced by a criterion based on the theorem that every logical sentence is determinate, that is, is either analytic or contradictory. The following definition is offered. "Let $K_1$ be the product of all expressional classes $K_i$ of $S$, which fulfill the following four conditions. (1) If $A_1$ belongs to $K_i$, then $A_1$ is not empty and there exists a sentence which can be subdivided into partial expressions in such a way that all belong to $K_i$ and one of them is $A_1$. (2) Every sentence which can be thus subdivided into expres-

* The definitions cannot be significant for every language, for H. B. Curry has developed a language, "combinatory logic," which contains no ordinary variables.

† The contrary assertion is made without proof on p. 195.

sions of $K_i$ is determinate. (3) The expressions of $K_i$ are as small as possible, that is to say, no expression belongs to $K_i$ which can be subdivided into several expressions of $K_i$. (4) $K_i$ is as comprehensive as possible, that is to say, it is not a proper sub-class of a class which fulfills both (1) and (2). An *expression* is called *logical* if it is capable of being subdivided into expressions of $K_1$; otherwise it is called *descriptive.*"

The reviewer fails to understand the rôle of condition (4). For suppose that a class $K_i$ contains some expression $A_1$ which is not a sentence. By the first condition, $A_1$ is then contained in a sentence $S_1$. According to condition (4), $S_1$ must be added to the class $K_i$, although by condition (3), $S_1$ cannot be added to $K_i$. This conflict between conditions (3) and (4) could be avoided by requiring in (4) that $K_i$ is not a proper subclass of a class satisfying (1), (2), and (3) (not merely (1) and (2)). The so modified definition still would not seem to agree with previous usage. For in Language I let $K_2$ be the class of all logical symbols in the usual sense, while $K_3$ is the class consisting of all expressions of the form "$(\exists x)$" and all logical symbols, except the existence-operator "$\exists$." These classes satisfy conditions (1) to (3). If necessary, they can be extended to larger classes $K_2'$ and $K_3'$, respectively, which also satisfy (4). Then $K_3'$ cannot contain the symbol "$\exists$," while $K_2'$ cannot contain the symbol "$(\exists x)$," so that $K_1$, which is part of the intersection of $K_2'$ and $K_3'$, can contain neither "$\exists$" nor "$(\exists x)$." It follows that sentences containing "$\exists$", and logical in the usual sense could not be logical according to this general definition.

This difficulty arises because condition (3) fails to have its intended effect upon the compound expression "$(\exists x)$." The definition of "logical" might be naturally modified as follows: Consider those classes $K_i$ which satisfy (1) and (2) and are maximal with respect to these conditions. For each class $K_i$ denote by $L_i$ the class of those expressions of $K_i$ which cannot be subdivided into several expressions of $K_i$, and let $K_1$ be the intersection of all $L_i$. This avoids the previous difficulty, only to meet another. For in Language I, consider a descriptive functor $f$ (in ordinary usage, $f$ is an empirically defined function $y=f(x)$, where $y$ and $x$ represent integers). Let the class $K_4$ contain the expressions "$f(0)$," "$=$," and "$\sim$," and all sentences constructed from these expressions. The only such sentences are

$$f(0) = f(0), \sim [f(0) = f(0)], \sim\sim [f(0) = f(0)], \cdots.$$

As "$\sim$" is the symbol for negation, all these sentences are either demonstrable or refutable, and hence are determinate. Thus $K_4$ satisfies (1) and (2), and so can be embedded in a maximal class satisfying (1) and (2). No numeral, such as 3, can be contained in this class, for "$f(0)=3$" is not a determinate sentence. Hence numerals are not logical symbols under the modified definition, contrary to the usage in Language I. Could the definition of "logical" symbols be further modified to avoid such difficulties?

Such technical points might raise doubts as to the philosophical thesis Carnap wishes to establish here: that in any language whatsoever one can find a uniquely defined "logical" part of the language, and that "logic" and "science" can be clearly distinguished.

Many of the other ingeniously defined concepts of Carnap's general syntax are free from objection. However, the points discussed above show how difficult is the task of defining so many relatively specific concepts in an absolutely arbitrary language. The notion of "any language" may be just as treacherous as was the notion of "any curve" before the critique of analysis situs. Might it not be possible to develop the concepts of general syntax in a more postulational manner? Thus, one might postulate that in the language there are certain symbols, designated as "logical" (or as "variables"), satisfying certain conditions analogous to those used in the definitions discussed above. Such an approach would recognize the obvious fact that such general syntax, though formulated for any language, is relevant chiefly for languages of the same general type as the Whitehead-Russell calculus, and it would make possible rigorous proofs that the terms defined do in fact agree with corresponding terms as applied to special languages. If such a postulational approach were possible, it would follow the general lines indicated by Carnap's far-reaching and pioneering investigations.

Many confusions and misunderstandings in logic, mathematics, and philosophy can be cleared up, as Carnap shows, by an understanding of the nature and possibilities of syntax. The notion of strict implication, as used in the Lewis logic of modalities, yields one such instance. Usually "strict implication" must be defined in terms of "necessity," so that the postulates must be chosen as the natural properties, if any, of this abstruse and perhaps fuzzy notion. But "$A$ strictly implies $B$" can be translated into the clean-cut syntactical statement, "The sentence '$B$' is a logical consequence of the sentence '$A$,' according to the rules of such and such a language." With this reformulation we can now unambiguously determine the properties of strict implication. We also recognize that these properties depend on the language concerned. In the same fashion, any special "logic of modalities" could be replaced by a syntactical translation. Carnap does not assert that it must be so replaced; he follows here and elsewhere a principle of tolerance in syntax: "It is not our business to set up prohibitions, but to arrive at conventions."

Throughout the book, Carnap makes a meticulous and clear-cut distinction between symbols and designations of symbols. The sentences "$\omega$ is an ordinal type" and "$\omega$ is a letter of the alphabet" appear to have the same subject. Actually, the first sentence is about the object denoted by "$\omega$," the second about the symbol "$\omega$," which is thus to be written in quotations. In this case the distinction is not essential, but in studying syntax it is requisite, for the sentences of syntax are precisely those which speak about symbols. For instance, to say that a sequence is calculable is to make a syntactical assertion about the sequence. Hence it must always be calculable with reference to a certain language.

In philosophical discussion it is important to recognize pseudo-syntactical sentences which do not appear to belong to syntax, but which can be translated into syntax. Carnap, in the last section of this book, shows how many fake problems and misunderstandings can be cleared up by such an analysis of sentences. For instance, "time is continuous" can be translated as "the real number expressions are used as time coordinates." "The world is a totality of facts, not of things" becomes "Science is a system of sentences, not of names." Some of his philosophical distinctions, such as that (p. 289) between the meaning of an expression and the object designated by an expression, are essentially dependent on the definition of "logical" sentences analyzed above. Such philosophical distinctions may therefore be untenable. The book ends with an eloquent discussion of two related theses: Any philosophy is either meaningless or is simply the logic of science; the logic of science is the syntax of the language of science.

The book contains many other illuminating discussions of various aspects of symbolic logic. In particular, we find an extraordinarily general statement of Gödel's theorem for an arbitrary language (unfortunately no proof and no reference to any printed proof is given); a discussion of various famous antinomies, syntactical and otherwise; a discussion of a paradox in certain axiomatic set-theories, according to which all sets are denumerable—syntactically denumerable, but not denumerable within the language of the set-theory itself. These three questions, and some of the other topics, were not included in the German edition of the book. As a whole, the book is a stimulating and fruitful discussion of syntax, a subject not yet in a definitive form but even now having a wide range of application in mathematics, science, and philosophy.

The following minor corrections might be noted. On page 40, Theorem 14.3 cannot be directly proven by induction. One must rather prove by induction that every logical sentence with $n$ distinct free variables either is contradictory or is analytic with not more than $n$ uses of the non-finite rule DC 2. On page 104, RR 9, read "unlimited operators" for "unlimited sentential operators." On page 21 replace the definition of an open expression by "If a variable which is free at some position in $A_1$ occurs in $A_1$ at that position, then $A_1$ is called open."

SAUNDERS MACLANE