FORMAL REDUCTIONS OF THE GENERAL COMBINATORIAL DECISION PROBLEM.*

By Emil L. Post.

1. Introduction. It is not new to the literature that the usual form of a symbolic logic with its parenthesis notation and infinite set of variables can be transformed into one in which the *enunciations*, i. e., formulas of the system, are finite sequences of letters,[1] the different letters constituting a once-and-for-all given finite set. If the primitive letters of such a system are represented by $a_1, a_2, \cdots, a_\mu$, an arbitrary enunciation of the system will take the form $a_{i_1} a_{i_2} \cdots a_{i_n}$, $n = 1, 2, 3, \cdots$, $i_j = 1, 2, \cdots, \mu$. In describing the basis of such a system it is convenient to use new letters to represent finite sequences of the above primitive letters. If then $A, B, \cdots, E$ represent the sequences $a_{i_1} a_{i_2} \cdots a_{i_\rho}$, $a_{j_1} a_{j_2} \cdots a_{j_\sigma}, \cdots, a_{m_1} a_{m_2} \cdots a_{m_\phi}$ respectively, $AB \cdots E$ will represent the sequence $a_{i_1} a_{i_2} \cdots a_{i_\rho}\, a_{j_1} a_{j_2} \cdots a_{j_\sigma} \cdots a_{m_1} a_{m_2} \cdots a_{m_\phi}$.

We shall say that such a system is in *canonical form* if its basis has the following structure.[2] The *primitive assertions* of the system are a specified finite set of enunciations of the above form. The operations of the system are a specified finite set of *productions*, each of the following form:

$$
\begin{array}{l}
g_{11} P_{i'_1}\, g_{12} P_{i'_2} \cdots g_{1m_1} P_{i'_{m_1}}\, g_{1(m_1+1)} \\
g_{21} P_{i''_1}\, g_{22} P_{i''_2} \cdots g_{2m_2} P_{i''_{m_2}}\, g_{2(m_2+1)} \\
\cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot \\
g_{k1} P_{i_1^{(k)}}\, g_{k2} P_{i_2^{(k)}} \cdots g_{km_k} P_{i_{m_k}^{(k)}}\, g_{k(m_k+1)} \\
\qquad\qquad \textit{produce} \\
g_1 P_{i_1}\, g_2 P_{i_2} \cdots g_m P_{i_m}\, g_{m+1}.
\end{array}
$$

* Received November 14, 1941; Revised April 11, 1942.

[1] More exactly, "strings" of "marks," to use terms of C. I. Lewis (*A Survey of Symbolic Logic*, Berkeley, 1918: chapter VI, sec. III).

[2] This formulation stems from the "Generalization by Postulation" of the writer's "Introduction to a general theory of elementary propositions," *American Journal of Mathematics*, vol. 43 (1921), pp. 163-185 (see p. 176). We take this opportunity to make the following *Emendation*: Lemma 1 thereof (pp. 177-178) requires the added condition that the expressions replacing the r's do not involve any letter upon which a substitution is made in the given deductive process. This necessitates several minor changes in the proof of the theorem there following. Actually, both Lemma 1 and its companion Lemma 2 admit of further simplification, with the proof of the theorem then being valid as it stands.

In this display the g's represent specified sequences of the primitive a's, including the null sequence, while the P's represent the operational variables of the production, and, in the application of the production, may be identified with arbitrary sequences of this type. In this notation, the distinct operational variables of a given production are to constitute the finite set of symbols $P_1, P_2, \cdots, P_M$ for some positive integral M. We then add the restriction that each operational variable in the conclusion of a production is present in at least one premise of that production, it having been understood that each premise and conclusion has at least one operational variable. We further assume that no identification of the operational variables is permitted which would lead to the conclusion being null.[3] The *assertions* of the system are then the primitive assertions, and all enunciations obtainable by the repeated application of the given productions starting with the primitive assertions.[4] More precisely, the class of assertions is the smallest subclass of the class of enunciations which contains the primitive assertions, and which, for each admissible assignment of values to the operational variables of each of the given productions, contains the enunciation represented by the conclusion of

[3] For the proof of Section **2** to be universally valid it is necessary that the operations themselves exclude the possibility of a null assertion. Since a null conclusion could arise only from an operation whose conclusion consists of operational variables only, while under our restriction at least one of these operational variables is not to be null, we can achieve the desired automatic exclusion of the null assertion by replacing each such operation by the following equivalent finite set of operations. For each operational variable P_i in the conclusion of such an operation, and each primitive letter a_j form the operation obtained by replacing P_i by $a_j P_i$ throughout the given operation. This modification need only be made on the given system in canonical form; for the productions introduced in Section **2** all have their single premises consist of more than just operational variables, so that the nonexclusion of the null assertion would have no further effect on the system.

[4] That this leads to the constructive generation of the class of assertions is readily verified. In particular, in trying to identify the premises of a production with corresponding previously obtained assertions, an explicit hypothesis on the "rank" of the operational variables involved, rank of a sequence being the total number of letters therein, would immediately lead to their unique determination or impossibility of realization, and hence correspondingly to a unique conclusion or impossibility of derivation. Since the sum of the ranks of the fixed g's and the fixed number of operational variables of a premise must equal the fixed rank of the assertion that premise is to be identified with, only a finite number of such hypotheses are admissible, and all can be uniformly tried out. In practice, a system in canonical form will usually be so constructed that a given assertion can be written in the form of a given premise in one and only one way, if at all. This uniqueness is automatically achieved by systems using the parenthesis notation, and is, of course, obviously attained in the systems in normal form about to be mentioned.

the production whenever it contains the enunciations represented by the several premises of the production.

A very special case of the canonical form is what we term the normal form. A system in canonical form will be said to be in *normal form* if it has but one primitive assertion, and, each of its productions is in the form

$$g\,P$$
produces
$$P\,g'.$$

The main purpose of the present paper is to demonstrate that every system in canonical form can formally be reduced to a system in normal form. The two forms may therefore in fact be said to be *equipotent.* More precisely, we prove the following

THEOREM. *Given a system in canonical form with primitive letters* $a_1, a_2, \cdots, a_\mu$, *a system in normal form with primitive letters* $a_1, a_2, \cdots, a_\mu$, $a'_1, a'_2, \cdots, a'_{\mu'}$ *can be set up such that the assertions of the system in canonical form are exactly those assertions of the system in normal form which involve no other letters than* $a_1, a_2, \cdots, a_\mu$.

As a result of this theorem the decision problem for a system in canonical form is reduced to the decision problem for the corresponding system in normal form. For an enunciation of the former system is an assertion when and only when it is an assertion of the latter system. Hence any procedure which could effectively determine for an arbitrary enunciation of the system in normal form whether it is or is not an assertion thereof would automatically do the same for the system in canonical form. Now by methods such as those referred to in the opening sentence of this introduction, it can be shown that the problem of determining for an arbitrary well-formed formula in the λ-calculus of Church whether it has or has not a normal form (Church) [5] can be reduced to the decision problem for a particular system in our canonical form. While Church has proved the above problem unsolvable in a certain technical sense, in the interest of economy we invoke his identification of λ-definability with effective calculability to conclude that as a result the decision problem for that particular system in canonical form, and hence for the class of systems in canonical form, is unsolvable. We are thus led to the more surprising result

[5] Alonzo Church, "An unsolvable problem of elementary number theory," *American Journal of Mathematics*, vol. 58 (1936), pp. 345-363.

that *there can be no effective procedure for determining for an arbitrary system in normal form and arbitrary enunciation thereof whether that enunciation is or is not an assertion of the system.* That is, the decision problem is unsolvable for the class of normal systems, and indeed, by the previous argument, for a certain particular one of them.[6]

The present paper is not the place to review the reasons why the equivalent mathematical definitions of combinatory solvability based on the technical concepts of λ-definability, general recursive function, and computability [7] can confidently be accepted as being the complete equivalent of combinatory solvability in the intuitive sense. Granting the initial establishment of the unsolvability of a particular decision problem by virtue of its being directly coextensive with the technical definition of solvability adopted, the chief method of establishing the unsolvability of further removed decision problems is by reducing the known unsolvable problem, by more or less ingenious formal devices, to those other problems.[8] Our reduction of the decision problem for the complicated canonical form to that of the simple normal form illustrates this in some measure. And it may be that because of its formal simplicity, the normal form may lend itself more readily to representation in specialized mathematical developments, and the unsolvability of its decision problem thus lead to the unsolvability of various hitherto unsolved decision problems of classical mathematics.

Of more immediate promise is the fact that the concepts of the present paper, with the help of its basic theorem, easily lead to an independent approach to unsolvable problems which may be far simpler than, say, the λ-calculus of Church. In this connection we may note that if we define a *normal set* of

---

[6] Absolutely unsolvable, that is, to use a phrase due to Church. By contrast, the undecidable propositions of Gödel's epoch making paper of 1931 (see footnote 7) are but relatively undecidable, the very proof of their undecidability in the given logic leading to an extension of that logic in which they are, indeed, proved to be true. A fundamental problem is the question of the existence of absolutely undecidable propositions, that is, propositions which in some *a priori* fashion can be said to have a determined truth-value, and yet cannot be proved or disproved by any valid logic.

[7] For the first two see the paper referred to in footnote 5, for the third see A. M. Turing, "On computable numbers, with an application to the Entscheidungsproblem," *Proceedings of the London Mathematical Society* (2), vol. 42 (1937), pp. 230-265. We might also add the writer's "Finite combinatory processes-formulation I," *Journal of Symbolic Logic*, vol. 1 (1936), pp. 103-105. The basic paper is, of course, that of Kurt Gödel, "Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I," *Monatshefte für Mathematik und Physik*, vol. 38 (1931), pp. 173-198.

[8] A very important instance of such a reduction is Gödel's transformation of the iterative recursive proposition into the non-iterative arithmetical proposition.

sequences on $a_1, a_2, \cdots, a_\mu$ as the set of assertions on those letters only of any system in normal form with primitive letters $a_1, a_2, \cdots, a_\mu$ and a finite number of additional letters, and a *canonical set* similarly via a system in canonical form, then the above theorem has as an immediate consequence the

COROLLARY. *The class of canonical sets is identical with the class of normal sets.*

For every normal set is *ipso facto* a canonical set; while if a canonical set is given by a certain system in canonical form, the theorem shows that it is also given by the corresponding system in normal form, and hence is a normal set. Now the canonical form naturally lends itself to the generating of sets by the method of definition by induction, while redefining the resulting canonical sets as normal sets makes it easy to use them as building blocks in further constructions. As a result of this alternating use of the canonical form as method, normal set as object, the Church development is easily paralleled.[9] And since at each step only normal sets of sequences are obtained, we are led to identify the intuitive concept of *generated set* with normal set for much the same reasons that led Church to identify effective calculability with λ-definability. Under this identification, the intuitive concept of a *solvable set* of sequences on $a_1, a_2, \cdots, a_\mu$, i. e., one for which there is an effective procedure for determining whether a given sequence on those letters is or is not in the set, becomes precisely the *binormal set,* i. e., a set such that both it, and its complement with respect to the set of all finite sequences on $a_1, a_2, \cdots, a_\mu$, are normal. The resulting definition of solvability then easily leads to the unsolvability of the decision problem for the class of normal systems, as well as for a particular one of them. We may note this interchange of primary and secondary concept as compared with the Church development; for normal set corresponds to recursively enumerable set, binormal set to (general) recursive set.[10]

[9] More completely, the Gödel, Church, Kleene, Rosser development.

[10] While this equivalence undoubtedly follows from the reduction of the λ-calculus to a system in normal form, it would probably be more easily established by way of Turing's concept of computability. A few initial properties of normal and binormal sets may here be noted. With the ordinary Boolean operations on classes in question, the class of all normal sets constitutes a (distributive) lattice, of all binormal sets, a Boolean ring, of all binormal sets on a given finite set of letters, a Boolean algebra. Every infinite normal set contains an infinite binormal set. Query: Does there exist an infinite set which is the complement of a normal set, relative to the given set of letters, and does not contain an infinite binormal set? [Added in proof: yes]. There is no theoretical loss of generality in restricting ourselves to normal sets on a single letter

Before turning to the proof of our basic theorem given in the next section, we wish to mention a further transformation of the normal form which is of interest for its juxtaposition of the solvable and unsolvable, and state a problem which largely determined the direction taken by the reductions of the next section, and may offer further opportunities for unsolvability proofs. By making the question of whether a given sequence has a certain succession of primitive letters at one end depend on a related sequence having a corresponding succession of letters at the other end, the following result can be proved. Given any system in normal form on primitive letters $a_1, a_2, \cdots, a_\mu$, an enunciation P thereof will be an assertion when and only when $\alpha P \alpha$ is an assertion in a corresponding effectively derivable system in canonical form on letters $a_1, a_2, \cdots, a_\mu, \alpha, a'_1, a'_2, \cdots, a'_\mu$, having a finite number of primitive assertions, and a finite number of operations of the following forms,

gP	Pg	g_1P	Pg_1
produces	*produces*	g_2P	Pg_2
$g'P$,	Pg' ,	*produce*	*produce*
		$g'P$,	Pg'.

A solution of the decision problem for the derived system then immediately yields a solution of the decision problem for the given system. Much of the simplicity of the normal form is thus given up in order to have the g' of the conclusion of each production on the same side of the operational variable as the g, or now g's, of the premise, or premises. But it is this very fact that leads to a solution of the decision problem for certain classes of these systems given *ab initio*. Indeed, for those systems in which all the productions have the g's on the same side a solution of the decision problem follows almost immediately from the solution of a decision problem given by the writer in a former paper.[11] This solution has been extended by the writer to those of the above systems having only first order productions, i. e., productions with but one premise, and it has at least been seen by the writer how to extend these

---

a—normal sets of natural numbers essentially. The following is then probably the analogue for normal systems, i. e., systems in normal form, of Turing's universal computing machine. A fixed finite set of normal operations involving a and a single additional letter b can be set up such that by varying a single primitive assertion on a and b all normal sets on a are obtained.

[11] "On a simple class of deductive systems," abstract, *Bulletin of the American Mathematical Society*, vol. 27 (1921), pp. 396-7. The systems in question are those systems of the formulation referred to in footnote 2 whose primitive functions are all functions of one argument.

solutions to those of the above systems in which only the second order productions are restricted to having their g's all on the same side, e. g., to systems having only productions of the first three of the above four types. For the class of all of the above systems, as indeed for a certain particular one of them, the decision problem is of course unsolvable as a consequence of the corresponding result for systems in normal form.

The problem referred to above takes two related forms. Both forms employ the following " tag " operations as we shall call them. Given a positive integer ν, and μ symbols which may be taken to be $0, 1, \cdots, \mu - 1$, we associate with each of the μ symbols a finite sequence of these symbols as follows.

$$0 \rightarrow a_{0,1}\, a_{0,2} \cdots a_{0,\nu_0}$$
$$1 \rightarrow a_{1,1}\, a_{1,2} \cdots a_{1,\nu_1}$$
$$\cdot\quad \cdot\quad \cdot\quad \cdot\quad \cdot\quad \cdot\quad \cdot\quad \cdot\quad \cdot\quad \cdot$$
$$\mu - 1 \rightarrow a_{\mu-1,1}\, a_{\mu-1,2} \cdots a_{\mu-1,\nu_{\mu-1}}.$$

It is understood that in each sequence the same symbol may occur several times, and that a particular associated sequence may be null. Given any non-null sequence

$$B = b_1\, b_2 \cdots b_l$$

on the symbols $0, 1, \cdots, \mu - 1$, a unique derived sequence B' on those symbols is determined as follows. To the right end of B adjoin the sequence associated with b_1, the first symbol of B, and from the left end of the resulting augmented sequence remove the first ν symbols—all if there be less than ν symbols. Starting with a given tag operation, and a given sequence A on its primitive symbols, we can then iterate the tag operation to yield $A_1 = A, A_2 = A'_1, A_3 = A'_2, \cdots$, the process terminating when and only when the null sequence is thus obtained. The first form of the problem of "tag" for a given tag operation is then to find an effective procedure for determining of an arbitrary initial sequence whether the above iterative process does or does not terminate.[12] In the second form of the problem we assume both the tag operation and the

[12] In an earlier formulation of this problem we merely checked off each successive ν-th letter of the sequence starting with b_1, at the same time adding the corresponding associated sequences to the right end of the sequence. Whether the iterative process terminated or not then depended on whether the constantly advancing check mark did or did not overtake the monotonically advancing right end of the sequence, whence the suggestive name of " tag " given the problem by B. P. Gill.

initial sequence to be given, and ask for an effective procedure for determining of an arbitrary sequence whether it is or is not one of the sequences obtained from the given sequence by the iteration of the given tag operation.

The first form of the problem of tag was intensely studied by the writer.[13] Extended by the further dichotomy of the non-terminating cases to the periodic (sequences bounded), and divergent (sequences unbounded), the problem was completely solved for all cases in which both μ and ν are 2. But little real progress can be reported for μ or ν greater than 2, the problem for such a simple basis as $0 \rightarrow 00$, $1 \rightarrow 1101$, $\nu = 3$ having proved intractable.[14] In its second form the problem is almost the decision problem for a special type of normal system. In fact, we may define a *monogenic normal system* as one in which the g's of the premises form a *complete set,* i. e., a set $g_1, g_2, \cdots, g_k$ such that each of the sequences of length equal to the maximum length ν of the g's can be written in the form g_iP for one and only one i. Except for the tag operation being applicable to sequences of length less than ν, a system of tag in its second form is then a monogenic normal system in which the g's constitute all of the μ^ν sequences in question, while the corresponding g''s are identical for all g's having the same initial symbol.

For a given tag operation the solution of the first form of the problem of tag probably leads to the solution of the second form of the problem. This is immediately so for those initial sequences which lead to termination or periodicity; and, while the mere hypothesis of divergence seems insufficient to guarantee a corresponding solution, the actual proof of divergence would probably make the definition of divergence effective, in which case the solution of the second form of the problem would again follow. For the writer, the little progress made in the solution of the first form of the problem make both forms, in their full generality, candidates for unsolvability proofs. Even more so, therefore, the decision problem for the class of monogenic normal systems. Among normal systems there is a "complete normal system" to which every normal system can be reduced, and whose decision problem is consequently unsolvable. A most interesting situation would obtain should it be shown that the complete normal system cannot be reduced to a monogenic normal system,

[13] During the writer's tenure of a Procter fellowship at Princeton University, 1920-1921.

[14] Numerous initial sequences actually tried led in each case to termination or periodicity, usually the latter.

while the decision problem for the class of monogenic normal systems is otherwise shown to be unsolvable.[15]

2. Reduction of the canonical form to the normal form. Our reduction of the canonical form to the normal form is the result of four successive reductions.[16] Each of these reductions yields a formulation which is included in the preceding formulation, but eliminates some formal complexities allowed in that preceding formulation. For a given system this simplification is achieved at the expense of an increase in the number of primitive letters employed, and in the number of productions appearing in its bases.

Our first reduction of an arbitrary system in canonical form is to one in which there is but one primitive assertion, and in which each production involves but a single premise. That one premise, and corresponding conclusion, however, may have all the complexity allowed for in the general canonical form. The general plan of the method involved is to formally introduce the logical products of arbitrary assertions of the given system, and operate within such products.

Let then S_1 be a system in canonical form with primitive letters $a_1, a_2, \cdots, a_\mu$, S_2 the system, about to be described, to which S_1 is to be reduced. With $a_1, a_2, \cdots, a_\mu$ also primitive letters of S_2, introduce two new primitive letters u and a_0 in S_2. When the logical product of assertions, $P_1, P_2, P_3, \cdots, P_n$ of S_1 is asserted in S_2, it will appear in the form

$$ua_0P_1\,a_0uua_0P_2\,a_0uuua_0P_3\,a_0\ldots\underbrace{u\ldots u}_{n}\,a_0P_na_0\,\underbrace{u\ldots u}_{n+1},$$

each P being flanked on either side by a_0. The separating u sequences are thus made to increase left to right by one each to enable us by the mere form of a premise to insure that certain operational variables therein must represent assertions of S_1, if that premise is to be identified with an assertion in S_2. The final basis for S_2 will reveal the necessary source of that insurance, i. e., that the only assertions of S_2 involving u are those of the above form. We shall call such an expression a product, the P's therein the factors of the product.

[15] It is easy to talk of obtaining a property of all normal solutions which could not be satisfied by a solution of a given decision problem; but this is probably equivalent to finding one of those not immediately obvious effectively calculable invariants of conversion which Church reports as still unfound in 1936. (See p. 358 of the paper referred to in footnote 5).

[16] Not counting a minor reduction needed to validate the last of the four.

We first introduce in the basis of S_2 certain productions whereby from the assertion of a product may be obtained the assertion of all products obtainable from the given product by a mere permutation of its factors. It suffices to allow for the interchange of any two consecutive factors. For the first two factors of a product this is achieved by

$$ua_0P_1a_0uua_0P_2a_0uuua_0S \quad \textit{produces} \quad ua_0P_2a_0uua_0P_1a_0uuua_0S,$$

our system being so devised that each product appearing therein has at least three factors. This allows that last a_0 to be assumed. The u, uu, uuu of the premise are then "maximal" u sequences. As these u sequences differ by one each, P_1 and P_2 must be free from u's, and hence, by our induction, be the two initial factors of the product. The interchange then results via the production. For two consecutive factors neither starting nor ending the product the result is achieved by

$$Ra_0uQua_0P_1a_0uQuua_0P_2a_0uQuuua_0S$$
$$\textit{produces}$$
$$Ra_0uQua_0P_2a_0uQuua_0P_1a_0uQuuua_0S.$$

Here Q must consist of u's only. For otherwise a_0uQua_0 and a_0uQuua_0 would have their initial a_0's followed by identical maximal u sequences. The u sequences uQu, $uQuu$ and $uQuuu$ are then maximal, and differ in length by one each. P_1 and P_2 again then are consecutive factors of the product. Finally, for two factors ending a product, the last production, rewritten with a_0S deleted, suffices.

The next production to be added to the bases of S_2 allows us to pass from the assertion of a product to the assertion of the first factor of a product, and hence, with the help of the previous three productions, to the assertion of an arbitrary factor of a product. The production is simply

$$ua_0Pa_0uua_0R \quad \textit{produces} \quad P.$$

In translating the operations of S_1 into operations within products of S_2, we allow for passing from a product whose initial factors can be identified with the premises of an S_1 operation, to that product with the conclusion of the S_1 operation as additional factor. That additional factor must end the new product so as not to disturb the progression of the maximal u sequences. Let "$G_1, G_2, \cdots, G_k$ produce G" represent any one of the S_1 operations. Let H represent

$$ua_0G_1a_0uua_0G_2a_0\ldots\underbrace{u\ldots u}_{k}a_0G_ka_0\underbrace{u\ldots u}_{k+1}.$$

Then the corresponding S_2 operation may be represented by

$$Ha_0Ra_0uQua_0Sa_0uQuu \quad \textit{produces}$$
$$Ha_0Ra_0uQua_0Sa_0uQuua_0Ga_0uQuuu.$$

Note that the operational variables of this production are those of the S_1 production, and Q, R, S. Since each operational variable in G occurs in at least one of the G_i's, our new production will indeed have the same operational variables in its conclusion as in its premise. The portion of the premise following H insures that Q consists of u's only. This, with the form of H, insures that $G_1, G_2, \cdots, G_k$ are determined factors of the premise and G of the conclusion. Hence our transformation of the S_1 production is valid. The additional operational variables R and S require an assertion to which this production is applied to have at least $k+2$ factors, a requirement secured below. Of course, the basis of S_2 is to have the correspondent of each of the operations in the basis of S_1.

With S_1 having κ productions, the above $\kappa+4$ productions constitute all of the productions in the basis of S_2. Its sole primitive assertion is then formed as follows. Let L be the largest number of premises occurring in any production of S_1. If S_1 has λ primitive assertions, let each be repeated L times to give λL sequences each involving no other letters than $a_1, \cdots, a_\mu$. If $\lambda L < L+2$, or $\lambda L < 3$, again duplicate one of these sequences the one or two times needed to avoid these inequalities. If then $k_1, k_2, \cdots, k_M$ are these duplicated primitive assertions of S_1, the primitive assertion of S_2 will be their product

$$ua_0k_1a_0uua_0k_2a_0\ldots\underbrace{u\ldots u}_{M}\,a_0k_Ma_0\underbrace{u\ldots u}_{M+1}.$$

Now it is readily proved by induction that if at a certain point of the process for obtaining assertions in S_1 a certain finite set of assertions has been obtained, then there will be asserted in S_2 a product among whose factors are each of the above assertions repeated L times. For the primitive assertions of S_1, this is insured by the primitive assertion of S_2. Assume it to be true for the deductive process in S_1 at an arbitrary point, let P_2 be the corresponding assertion in S_2, P_1 the next assertion obtained in $S_1, P_{11}, P_{12}, \cdots, P_{1k}$ the premises of the production of S_1 yielding conclusion P_1. Then each P_{1j} appears as a factor of P_2 indeed L times at least. Hence from P_2, by the first three productions of S_2, an assertion P'_2 can be obtained in which the first k factors are $P_{11}, P_{12}, \cdots, P_{1k}$ respectively, whatever repetitions may occur among those P's. The production of S_2 corresponding to the one of S_1 in question will then add P_1 as factor to P'_2. Mere repetition of the application of this production will then yield P''_2, which will be P'_2 with L additional factors equal to P_1. The induction is thus established. It follows that for each assertion P_1 in S_1 there will be an assertion P_2 in S_2 having P_1 as factor. By the first three productions of S_2 this factor can be made the first factor of an assertion in S_2, and hence, by the fourth production of S_2, P_1 itself will be an assertion of S_2. That is, every assertion of S_1 is an assertion of S_2. Our basis for S_2 shows that the only other assertions of S_2 are products of assertions of S_1, and so not wholly written on the letters of S_1. Hence, an enunciation of S_1 is an assertion of S_1 when and only when it is an assertion of S_2, whence the reduction of S_1 to S_2.

In our second reduction of the canonical form the productions, all with single premises by the previous reduction, now take the more special form

$$g_1 P_1 g_2 P_2 \cdots g_m P_m g_{m+1}$$
$$\textit{produces}$$
$$\bar{g}_1 P_1 \bar{g}_2 P_2 \cdots \bar{g}_m P_m \bar{g}_{m+1}$$

where, however, m, and of course the g's, may vary from operation to operation. By contrast, in the previous productions P's could be repeated, have different arrangements in premise and conclusion, and in part be missing from the conclusion while present in the premise.

Again let the primitive letters of the given system be symbolized $a_1, a_2, \cdots, a_\mu$. Let its i-th production be

$$g_1 P_{j_1} g_2 P_{j_2} \cdots g_m P_{j_m} g_{m+1}$$
$$\textit{produces}$$
$$g'_1 P_{j'_1} g'_2 P_{j'_2} \cdots g'_{m'} P_{j'_{m'}} g'_{m'+1}$$

where it is understood that each letter except P has i for additional subscript. The subscripts of the P's need not be distinct in premise or conclusion, while the different subscripts of the P's in the conclusion all appear in the premise. However, the letter P occurs exactly $m + m'$ times in the production.

We introduce a new primitive letter u, and for each such production two new primitive letters v_i, w_i. In obtaining the effect of the i-th production we shall, as above, leave this subscript i understood. v_i will be used in passing from an assertion involving a's only that could be the premise of the i-th production to one which has both that premise and corresponding conclusion recognizable within it; w_i in passing from such a composite assertion to the desired conclusion only. The efficacy of our method will depend on each assertion in the new system which involves v or w having that letter only at the beginning of the assertion, and in the first case always involving exactly $2m + m'$ u's, in the second, m' u's. Our new productions will in every case explicitly exhibit this v and $2m + m'$ u's, or w and m' u's, so that we can be sure that in their application the operational variables can represent sequences of a's only. Except for a minor preliminary type, all of our "v-assertions" will be in the form

$$vug_1P_1uQ_1ug_2P_2uQ_2 \cdots ug_mP_muQ_mg_{m+1}ug'_1Q_{m+1}ug'_2Q_{m+2} \cdots ug'_{m'}Q_{m+m'}g'_{m'+1},$$

and when so asserted will have the following properties. The sequence of a's $g_1P_1Q_1g_2P_2Q_2 \cdots g_mP_mQ_mg_{m+1}$ is an assertion of the given, and indeed new system, while the sequences of a's $Q_1, Q_2, \cdots, Q_m, Q_{m+1}, \cdots, Q_{m+m'}$ can, in order, be identified with $P_{j_1}, P_{j_2}, \cdots, P_{j_m}, P_{j'_1}, \cdots, P_{j'_{m'}}$, that is, any two Q's corresponding to P's with identical subscripts are equal. Note that with all $2m + m'$ u's exhibited, the g's being given, the P's and Q's of such an assertion are uniquely identifiable in the assertion. Our method depends on the fact that when such an assertion is obtained in which the P's are null, then, due to the equalities forced on the Q's, $g_1Q_1g_2Q_2 \cdots g_mQ_mg_{m+1}$ becomes an assertion on a's only that can be identified with the premise of the i-th production of the given system, and hence $g'_1Q_{m+1}g'_2Q_{m+2} \cdots g'_{m'}Q_{m+m'}g'_{m'+1}$ an expression on a's only that will be the corresponding conclusion. Of course, each production about to be described is directly seen to be in the desired newly simplified form.

Since a null assertion has been excluded from our systems, each assertion of the given system is of the form a_jP, $j = 1, 2, \cdots \mu$. The productions

$$a_jP \quad \textit{produces} \quad va_jPu \ldots u$$

with $2m + m'$ u's in $u \ldots u$ changes each "a-assertion," i. e., assertion involving a's only, into what we shall call the intermediate v form. As all other assertions of our new system will begin with v or w, these productions will be inapplicable to them. If now an a-assertion can be the premise of the i-th production, its intermediate v form will be put into primary v form, or just v form, by the production

$$vg_1P_1g_2P_2 \cdots g_mP_mg_{m+1}u \ldots u$$

$$\textit{produces}$$

$$vug_1P_1uug_2P_2uu \ldots g_mP_mug_{m+1}ug'_1ug'_2u \ldots ug'_{m'}g'_{m'+1}.$$

Of course this production may be applicable without the P's being identifiable with those of the premise of the i-th production. But, comparing this conclusion with our general v form, we see that it satisfies the requirement thereof with all Q's null. Now any set of a-sequences that could be identified with the $P_{j_1}, P_{j_2}, \cdots, P_{j_m}, P_{j'_1}, \cdots, P_{j'_{m'}}$ of the i-th production can be built up as follows. Start with the set of null sequences. Let $Q_1, Q_2, \cdots, Q_m, Q_{m+1}, \cdots, Q_{m+m'}$ be any such derived set of a-sequences. Let $Q_{j_1}, Q_{j_2}, \cdots, Q_{j_\nu}$, j's increasing, be any subset thereof corresponding to all P's with subscripts equal to a given subscript, a_j any one of the primitive a's. Then $\cdots, a_jQ_{j_1}, \cdots, a_jQ_{j_2}, \cdots, a_jQ_{j_\nu}, \cdots$, all other Q's unchanged, will also be such a set of a-sequences. Rewrite the subscript sequence $j_1, j_2, \cdots, j_\nu$ in the form $j_1, \cdots, j_\lambda, j_{\lambda+1}, \cdots, j_\nu$ so that $j_\lambda \leqq m$, $j_{\lambda+1} > m$, and let $j_{\lambda+1} - m = j'_1, \cdots, j_\nu - m = j'_{\lambda'}$. Of course we may have $\lambda = \nu$. Now for each such choice of original P subscript, and each a_j, introduce the production

$$v \ldots ug_{j_1}P_{j_1}a_juQ_{j_1} \cdots ug_\lambda P_{j_\lambda}a_juQ_{j_\lambda} \cdots ug'_{j'_1}Q_{j_{\lambda+1}} \cdots ug'_{j'_{\lambda'}}Q_{j_\nu} \cdots$$
produces
$$v \ldots ug_{j_1}P_{j_1}ua_jQ_{j_1} \cdots ug_\lambda P_{j_\lambda}ua_jQ_{j_\lambda} \cdots ug'_{j'_1}a_jQ_{j_{\lambda+1}} \cdots ug'_{j'_{\lambda'}}a_jQ_{j_\nu} \cdots$$

all of the rest of both premise and conclusion being as in the type v form above. Such a production will then change a valid v form into a valid v form, the effect being however to "drain" the P's of such a form and "swell" the Q's. If then an assertion of the given system can be put in the form of the premise of the i-th production, the corresponding intermediate v form will pass into a v form such that successive application of the above productions will completely drain the P's thereof; and, indeed, conversely. This marks the end of the first half of the passage from a-assertion to a-assertion in the new system. While the second half could be set up by means of similar w productions in reverse, with interchange of emphasis on premise and conclusion of the i-th production, the following method is simpler. With P's all null, the v form determines the desired a-conclusion as described above. The w forms, about to be introduced, each have exactly m' u's all explicitly appearing in the productions. From such a v form with P's all null the first w form is obtained via

$$vug_1uQ_1ug_2uQ_2 \cdots ug_muQ_mg_{m+1}ug'_1Q_{m+1}ug'_2Q_{m+2} \cdots ug'_{m'}Q_{m+m'}g'_{m'+1}$$
produces
$$wg_1Q_1g_2Q_2 \cdots g_mQ_mg_{m+1}ug'_1Q_{m+1}ug'_2Q_{m+2} \cdots ug'_{m'}Q_{m+m'}g'_{m'+1}.$$

We can now get rid of the no longer interesting part of this w form, i. e., the part between w and the first u thereof, by the μ productions

$$wa_jPug'_1P_1ug'_2P_2\cdots ug'_{m'}P_{m'}g'_{m'+1}$$
$$produces$$
$$wPug'_1P_1ug'_2P_2\cdots ug'_{m'}P_{m'}g'_{m'+1}$$

iteratively applied till letter by letter what was the original a-assertion disappears. The desired a-conclusion then would be obtained via

$$wug'_1P_1ug'_2P_2\cdots ug'_{m'}P_{m'}g'_{m'+1}$$
$$produces$$
$$g'_1P_1g'_2P_2\cdots g'_{m'}P_{m'}g'_{m'+1}.$$

Our final system will then be on the primitive letters $a_1,\cdots,a_\mu, u, v_1, w_1, v_2, w_2,\cdots, v_\kappa, w_\kappa$, κ being the number of productions of the given system. The one primitive assertion of the new system will be the one primitive assertion of the given system, the productions of the new system, all of the above productions for each of the κ productions of the given system. Our above analysis then easily shows that the assertions of the new system involving no other letters than $a_1,\cdots,a_\mu$ are exactly the assertions of the given system, and the desired reduction has been effected.

Our third and penultimate simplifying reduction of the canonical form is to one where the operations are of the form

$$g_1Pg_2$$
$$produces$$
$$\bar{g}_1P\bar{g}_2,$$

i. e., involve but a single operational variable. Again let a system in the previous simplified form have primitive letters $a_1, a_2,\cdots, a_\mu$, and κ operations, the number of P's in the premise, and hence conclusion, of the i-th operation being m_i. For the i-th operation, with $i=1,2,\cdots,\kappa$, and each primitive letter a_j we introduce $2m_i+1$ new primitive letters $a'_{ji}, a''_{ji},\cdots, a^{(2m_i)}{}_{ji}$, $a^{(2m_i+1)}{}_{ji}$. We also introduce the primitive letter a_{0i} and its $2m_i+1$ primed equivalents. With one such operation in mind at a time we shall, as above, omit the extra subscript i. Apart from the use of a_0 and $a^{(j)}{}_0$'s, needed to take care of g's or P's that are null, the essence of our method is to pass from an a-assertion in the form $g_1P_1g_2P_2\cdots g_mP_mg_{m+1}$ to an assertion $g'_1g'''_2\cdots g^{(2m+1)}{}_{m+1}P''_1P^{\mathrm{IV}}{}_2\cdots P^{(2m)}{}_m$ where the superscript k say indicates that each a_j in the corresponding expression is here written $a_j{}^{(k)}$. As a result our premise will now have the form gP with $g=g'_1g'''_2\cdots g^{(2m+1)}{}_{m+1}$, $P=P''_1P^{\mathrm{IV}}{}_2\cdots P^{(2m)}{}_m$.

In detail, we first introduce μ productions

$$a_j P \quad \textit{produces} \quad a_0 a_j P, \qquad j = 1, 2, \cdots, \mu,$$

which will be applicable in fact only to assertions on $a_1, a_2, \cdots, a_\mu$, and changes any such assertion Q into $a_0 Q$. We then introduce a finite series of finite sets of production depending in number on m and μ. The first set has the one production

$$a_0 g_1 P \quad \textit{produces} \quad a'_0 g'_1 a_0 P a''_0.$$

Inductively let the conclusion of the sole production in the $(2k-1)$-st set be in the form $G_k a_0 P a^{(2k)}{}_0$. Then the $(2k)$-th set has the μ productions

$$G_k a_0 a_j P \quad \textit{produces} \quad G_k a_0 P a^{(2k)}{}_j, \qquad j = 1, 2, \cdots, \mu,$$

the $(2k+1)$-st set the sole production

$$G_k a_0 g_{k+1} P \quad \textit{produces} \quad G_k a^{(2k+1)}{}_0 g^{(2k+1)}{}_{k+1} a_0 P a^{(2k+1)}{}_0.$$

This is to hold for $1 \leqq k < m$, while for $k = m$ the sole production of the $(2m+1)$-st set is to be

$$G_m a_0 g_{m+1} a''_0 P \quad \textit{produces} \quad G_m a^{(2m+1)}{}_0 g^{(2m+1)}{}_{m+1} a''_0 P.$$

We then readily see that starting with an assertion on $a_1, \cdots, a_\mu$ in the form $g_1 P_1 g_2 P_2 \cdots g_m P_m g_{m+1}$, one can, with the aid of these productions, obtain as an assertion

$$a'_0 g'_1 a'''_0 g'''_2 \cdots a^{(2m-1)}{}_0 g^{(2m-1)}{}_m a^{(2m+1)}{}_0 g^{(2m+1)}{}_{m+1} a''_0 P''_1 a^{\mathrm{IV}}{}_0 P^{\mathrm{IV}}{}_2 \cdots a^{(2m)}{}_0 P^{(2m)}{}_m.$$

Furthermore, note that starting with an assertion on $a_1, a_2, \cdots, a_\mu$, flanked on the left by a_0 as above, one can apply the above operations only in the following order, if at all. First, the sole operation of the first set; and inductively, if the operation in the $(2k-1)$-st set has last been applied, the next applicable operation can only be an operation in the $2k$-th set or the operation in the $(2k+1)$-st set, if an operation in the $(2k)$-th set has last been applied, the next applicable operation can only be an operation in the same set, or the operation in the next set. Furthermore, the last operation in its premise explicitly indicates the a''_0, first introduced into an assertion only as a result of the first operation. It readily follows that if the last operation does enter into a possible sequence of operations, the conclusion thereof can have no letter a_j in it without a superscript. The entire given assertion has thus been translated; and it is readily seen that that last assertion, and hence given assertion, are and can be put in the forms above given.

The actual correspondent of the original i-th operation in translated form may then be written simply

$$a'_0g'_1a'''_0g'''_2 \cdots a^{(2m-1)}{}_0g^{(2m-1)}{}_ma^{(2m+1)}{}_0g^{(2m+1)}{}_{m+1}P$$
$$\textit{produces}$$
$$a'_0\bar{g}'_1a'''_0\bar{g}'''_2 \cdots a^{(2m-1)}{}_0\bar{g}^{(2m-1)}{}_ma^{(2m+1)}{}_0\bar{g}^{(2m+1)}{}_{m+1}P;$$

and the passage from this translated conclusion to the actual conclusion can be effected by a set of productions the reverse of those above given. That is, in each of the above productions prior to the actual correspondent of the i-th production replace all g_j's by $\bar{g}_j$'s, and *interchange hypothesis and conclusion.* The resulting productions then clearly suffice to yield the conclusion yielded by the original i-th production. True, the complete set of productions thus set up to take the place of the original i-th production may now allow other paths than from assertion on $a_1, \cdots, a_\mu$, down the first group of productions, through the intermediate production, and up the second group of productions to new assertion on $a_1, \cdots, a_\mu$.[17] But it is readily seen that any departures from this progression merely constitute unravelings of parts of such a progression, or, apart from such unravelings, constitute shortcuts of valid full progressions of this type. Since, furthermore, one can change the set of productions one is working with only when an assertion on $a_1, \cdots, a_\mu$ alone is obtained, the validity of our reduction follows.

Our final reduction is to a system whose operations are in the form

$$gP$$
$$\textit{produces}$$
$$Pg'.$$

The present method assumes that in the productions of the previous system, all in the form

$$g_1Pg_2$$
$$\textit{produces}$$
$$g'_1Pg'_2,$$

g_1 and g_2 are never null. We therefore actually first need the following preliminary reduction. Introduce a new primitive letter a_0, and if h is the sole primitive assertion of the given system let a_0ha_0 be the sole primitive assertion of the new system. Replace each of the above operations of the given system by

$$a_0g_1Pg_2a_0 \quad \textit{produces} \quad a_0g'_1Pg'_2a_0$$

and finally add the production

[17] This could have been avoided say by the v, w method used earlier.

$$a_0Pa_0 \quad produces \quad P.$$

Except for the last production the new system may be said to be simply isomorphic with the old, P being an assertion in the given system when and only when a_0Pa_0 is an assertion in the new system. The last operation then merely recovers the assertions of the given system. Note that even that last operation is in the desired form with neither g_1 nor g_2 null.

Assume then that such is our given system with primitive letters again $a_1, a_2, \cdots, a_\mu$. We introduce new primitive letters $\bar{a}_1, \bar{a}_2, \cdots, \bar{a}_\mu$, and "translating productions"

$$a_jP \quad produces \quad P\bar{a}_j, \quad \bar{a}_jP \quad produces \quad Pa_j, \qquad j = 1, 2, \cdots, \mu.$$

Starting with an assertion of the form $a_{i_1} \cdots a_{i_j}\, a_{i_{j+1}} \cdots a_{i_n}$, these productions will yield only assertions of the form $a_{i_{j+1}} \cdots a_{i_n}\bar{a}_{i_1} \cdots \bar{a}_{i_j}$, $\bar{a}_{i_1} \cdots \bar{a}_{i_j}\bar{a}_{i_{j+1}} \cdots \bar{a}_{i_n}$, $\bar{a}_{i_{j+1}} \cdots \bar{a}_{i_n}a_{i_1} \cdots a_{i_j}$, in addition to the original assertion. Only one of these $2n$ distinct forms consists wholly of unbarred letters, i. e., the original form, while continued application of the above operations merely keeps deriving these $2n$ "equivalent forms" cyclically, so that anyone can thus be obtained from any other.

Our reduction will then be effected if for each operation "g_1Pg_2 *produces* $g'_1Pg'_2$" of the given system we introduce in the new system the operation

$$\bar{g}_2g_1P$$
$$produces$$
$$Pg'_2\bar{g}'_1,$$

where $\bar{g}_2$, for example, is g_2 with each letter replaced by the corresponding barred letter. Of course, the one primitive assertion of the given system is also the one primitive assertion of the new system. Note that if at any point an assertion without barred letters appears, then if it can be written g_1Pg_2, the first given translating operations derive from it $\bar{g}_2g_1P$, hence the above yields $Pg'_2\bar{g}'_1$, and so finally $g'_1Pg'_2$ is obtained as desired. That is, the new system contains all of the assertions of the given system. It further follows that the assertions of the new system consist only of the assertions of the given system and their equivalents. For, proceeding inductively, this clearly remains true under the translating operations. Now suppose it is true of an assertion in the form $\bar{g}_2g_1P$. Since $\bar{g}_2$ and g_1 are not null, $\bar{g}_2g_1$ alone exhibits a change from barred to unbarred letters. P therefore must consist of unbarred letters only. $\bar{g}_2g_1P$ is therefore a translation of the assertion g_1Pg_2 of the original system, and hence the conclusion $Pg'_2\bar{g}'_1$ is a translation of $g'_1Pg'_2$, also an assertion of the original system. The desired reduction has thus been effected.

The original system in canonical form has thus been reduced to a system in normal form. At each stage in that reduction the primitive letters of the new system are the primitive letters of the preceding system and a finite number of additional letters, while the assertions of that preceding system are exactly those assertions of the new system which involve only primitive letters of the preceding system. The same is then true of the original system in canonical form and the final system in normal form, whence the theorem of the introduction.[18]

THE CITY COLLEGE,
COLLEGE OF THE CITY OF NEW YORK.

---

[18] While the present paper is presented as a contribution to the literature as it now exists, its main development, that of Section **2**, as well as the further transformation mentioned in the introduction, was obtained by the writer in substantially the form here given in the summer of 1921. The larger development of which this was a part is no longer essentially new, but may be worth a brief résumé. In the fall of 1920, starting with the formulation referred to in footnote 2 as canonical form A, the operation of substitution of that formulation was weakened in two successive stages to yield canonical forms B and C, the last only allowing any 1-1 replacement of variables by variables. It was first shown in detail that each of these forms was reducible to the other two, and then the project that led to the above changes of canonical form A was carried through, namely the reduction of what would now be termed the restricted functional calculus of *Principia Mathematica* to a particular system in canonical form C. Much of this work has since been found to be seriously in error, but easily corrected by the methods then employed. As a result of the last reduction it appeared obvious to the writer that all of *Principia Mathematica* could likewise be reduced to a system in canonical form C. In the summer of 1921, the intervening work on the problem of tag suggested the reduction of canonical form C to the canonical form of the present paper, and this reduction was followed by the successive reductions to normal form essentially as given in Section **2**. The added methods thus revealed led us to conclude that not only *Principia Mathematica*, but any symbolic logic whose operations could effectively be reproduced in *Principia Mathematica*, and hence probably any (finitary) symbolic logic could be reduced to a system in canonical form, and consequently to a system in normal form. But now the entire direction of our thought, that of solving the decision problem for arbitrary systems, was reversed. Having noted the identity of canonical sets and normal sets referred to in the introduction, our last conclusion was transformed into the generalization that every generated set of sequences on a finite sets of letters was a normal set. The seeming counter example furnished by the diagonal method then led to an informal proof that the decision problem for the class of systems in normal form was unsolvable. In the early fall of 1921, the formal proof of this unsolvability, referred to in the introduction, was outlined, and led to the further conclusion that not only was every (finitary) symbolic logic incomplete relative to a certain fixed class of propositions (those stating that a given sequence was or was not an assertion in a given normal system) but that every such logic was extendable relative to that class of propositions. Since the earlier formal work made it seem obvious that the actual details of the outline could be supplied, the further efforts of the writer were directed towards establishing the universal validity of the basic identification of generated set with normal set.