A VARIANT OF A RECURSIVELY UNSOLVABLE PROBLEM

EMIL L. POST

By a string on a, b we mean a row of a's and b's such as *baabbbab*. It may involve only a, or b, or be null. If, for example, g_1, g_2, g_3 represent strings *bab*, *aa*, *b* respectively, string $g_2g_1g_1g_3g_2$ on g_1, g_2, g_3 will represent, in obvious fashion, the string *aababbabbaa* on a, b. By the *correspondence decision problem* we mean the problem of determining for an arbitrary finite set (g_1, g_1'), (g_2, g_2'), $\cdots$, (g_μ, g_μ') of pairs of corresponding non-null strings on a, b whether there is a solution in $n, i_1, i_2, \cdots, i_n$ of equation

$$(1) \qquad g_{i_1}g_{i_2}\cdots g_{i_n} = g'_{i_1}g'_{i_2}\cdots g'_{i_n}, \qquad n \geqq 1,\ i_j = 1, 2, \cdots, \mu.$$

That is, whether some non-null string on $g_1, g_2, \cdots, g_\mu$, and corresponding string on $g_1', g_2', \cdots, g_\mu'$, represent identical strings on a, b.

In special cases, of course, the question posed by (1) may be answerable. Thus, if, with $\mu=3$, (g_1, g_1'), (g_2, g_2'), (g_3, g_3') are (bb, b), (ab, ba), (b, bb) respectively, $g_1g_2g_2g_3 = bbababb = g_1'g_2'g_2'g_3'$, and (1) has a solution. Again, if each g_i is of greater length than the corresponding g_i', or if each g_i starts with a different letter than the corresponding g_i', (1) has no solution. We proceed to prove, on the other hand, that in its full generality *the correspondence decision problem is recursively unsolvable*,[1] and hence, no doubt, unsolvable in the intuitive sense.

We start with the known recursive unsolvability of the decision problem for the class of normal systems on a, b.[2] A normal system S on

Received by the editors October 20, 1945.

[1] It suffices here to consider "recursively unsolvable" to mean unsolvable in the sense of Church [1]. Of course the general problem remains recursively unsolvable if we allow null g's and g''s. Numbers in brackets refer to the references cited at the end of the paper.

[2] See [4, §2] for an informal proof. As far as the printed literature is concerned, we must refer to [2] for a formal proof, though there then remains the actual verification, via Gödel representations, that the reduction effected is indeed recursive. This verification, at least for the reduction of S' to S''' [2, p. 51], is immediate if we use the following simpler method of reducing S' to a system S'' in canonical form than that there given by Church. The primitive symbols of our S'' are those of S' and one additional primitive symbol α. The basis of S'' in part consists of the two primitive assertions αI, αJ, and the operation αP, αQ *produce* $\alpha(PQ)$. It will follow that αP is asserted in S'' when and only when P is a combination without free variables. The remainder of the basis of S'' consists of the primitive assertion of S' as primitive assertion, and the thirty-eight operations of S' each modified as follows. For each operational variable P occurring in the operation, αP is introduced as additional premise.