Use of a Computing Machine as a Mechanical Dictionary

In considering the application of an automatic calculator to problems of mechanical translation, the first need which arises is the storage of a dictionary[1]. It has been shown[2] that the naïve scheme in which the translation is contained in that storage location the address of which is given by the number that results from numerical encoding of the foreign-language word is not practical, and the procedure which has been adopted up to the present has been to store the dictionary in a sequence of locations. The order of storage is such that the code number of the foreign language portions of the words are in ascending order of magnitude. To select the entry corresponding to a given foreign-language word, the code number of that word is subtracted successively from the dictionary entries starting from the first. The result of such a subtraction will always be negative until the required position is reached, when the searching process is discontinued and the next phase of the translation operation is initiated.

It is evident that, for a dictionary which contains *N* words, an average of *N/2* subtractions will be required to locate a word chosen at random. Since *N* may be between 10^4 and 10^5, and the subtraction and access times on many machines amount to about 10^{-3} sec., the time to locate a given word will lie between 5 and 50 sec.

To reduce this time, a possible method is to store words which start with the same letter in storage locations the code numbers of which are related to those of the initial letters. This reduces the time of search by a factor of about 26, but has the disadvantage that some blocks of locations will be only partially filled.

An alternative approach to the problem of dictionary search is as follows. For convenience it will be assumed that entries are stored in locations 1 to *N*, and the unknown word is first subtracted from the dictionary entry in location *N/2*. If the result is positive, the required entry is either that contained in position *N/2*, or it is contained in one of the locations *N/2* to *N*. If the result of the subtraction is negative, the required entry lies in locations 1 to $(N/2 - 1)$. In the former case the comparison is repeated on the entry contained in location $(N/2 + N/4)$, and in the latter case on the entry in $(N/2 - N/4)$. This process is repeated until the unknown position is isolated.

The importance of the method lies in the very small number of comparisons which are needed to isolate an entry. Thus, for a dictionary of 10^4 words, less than 14 comparisons will isolate any given word, and for 10^6 words 20 comparisons should suffice. The time of search required for the example quoted earlier would thus be reduced to less than 20 m.sec.

In practical application it is not usual for *N* to be of the form 2^m, so that instead of the numbers *N/2, N/4,* . . ., etc., it is sufficient to take the nearest integer to these values.

An amusing lecture, demonstration is based upon this process. A member of the audience selects a word at random from the "Concise Oxford Dictionary"; by asking a maximum of 14 questions the lecturer is then able to find the word selected. A simple form of the question is "Does the word come before.... ?" in the dictionary. Partition need only be approximate and a rough division of the dictionary into ½, ¼, etc., by eye is sufficient.

ANDREW D. BOOTH
Birkbeck College Computational Laboratory,
21 Torrington Square,
London, W.C.1.
May 12.

[1] Booth, A.D., and Booth, K. H. V., "Automatic Digital Calculators", 15 (Butterworths, London, 1953).
[2] Booth, A. D., "Computers and Automation", **2**. 6 (1953).