produced by the method can be increased by increasing the number of tests required to complete the separation of each taxon pair. In the four sets of data analysed the simple sequential method found in every case a diagnostic table with the fewest possible tests.

Acknowledgements

This work was supported by a grant from the Department of Health and Social Security. Thanks are due to Miss J. Stevens for the preparation of the manuscript.

References

BARNETT, J. A. (1971). Selection of tests for identifying yeasts, *Nature, New Biology*, Vol. 232, pp. 221-223.
DARWOOD, N. (1971). Implementing a decision table, *Computer Weekly*, No. 262, p. 6.
GORRY, G. A. (1968). Strategies for computer-aided diagnosis, *Mathematical Biosciences*, Vol. 2, pp. 293-318.
GOWER, J. C., and BARNETT, J. A. (1971). Selecting tests in diagnostic keys with unknown responses, *Nature, London*, Vol. 232, pp. 491-493.
GYLLENBERG, H. G. (1963). A general method for deriving determination schemes for random collections of microbial isolates, *Annales Academiae Scientiarum Fennicae*, A, IV, Vol. 69, pp. 1-23.
HALL, A. V. (1970). A computer-based system for forming identification keys, *Taxon*, Vol. 19, pp. 12-18.
HILL, L. R., and SILVESTRI, L. G. (1962). Quantitative methods in the systematics of Actinomycetales. III. The taxonomic significance of physiological-biochemical characters and the construction of a diagnostic key, *Giornale di Microbiologia*, Vol. 10, pp. 1-28.
LAPAGE, S. P., BASCOMB, S., WILLCOX, W. R., and CURTIS, M. A. (1970). Computer identification of bacteria. In: *Automation, Mechanization and Data Handling in Microbiology*, Society for Applied Bacteriology Technical Series No. 4, edited by A. Baillie and R. J. Gilbert, London: Academic Press, pp. 1-22.
MACCACARO, G. A. (1958). La misura della informazione contenuta nei criteri di classificazione, *Annali di Microbiologia ed Enzimologia*, Vol. 8, pp. 231-239.
MÖLLER, F. (1962). Quantitative methods in the systematics of Actinomycetales. IV. The theory and application of a probabilistic identification key, *Giornale de Microbiologia*, Vol. 10, pp. 29-47.
MORSE, L. E. (1971). Specimen identification and key construction with time-sharing computers, *Taxon*, Vol. 20, pp. 269-282.
NIEMELÄ, S. I., HOPKINS, J. W., and QUADLING, C. (1968). Selecting an economical binary test battery for a set of microbial cultures, *Canadian Journal of Microbiology*, Vol. 14, pp. 271-279.
PANKHURST, R. J. (1970a). A computer program for generating diagnostic keys, *The Computer Journal*, Vol. 13, pp. 145-151.
PANKHURST, R. J. (1970b). Key generation by computer, *Nature, London*, Vol. 227, pp. 1269-1270.
PIGUET, J. D., and ROBERGE, P. (1970). Problèmes posés par le diagnostic automatique des bâtonnets gram-négatifs, *Canadian Journal of Public Health*, Vol. 61, pp. 329-335.
RYPKA, E. W., CLAPPER, W. E., BOWEN, I. G., and BABB, R. (1967). A model for the identification of bacteria, *Journal of General Microbiology*, Vol. 46, pp. 407-424.
SNEATH, P. H. A. (1969). Computers in bacteriology, *Journal of Clinical Pathology*, Vol. 22, suppl. 3, pp. 87-92.

Correspondence

To the Editor
The Computer Journal

Sir,

I must take issue with Mr. Finn (*The Computer Journal*, Vol. 15, No. 1, p. 12, 1972) over his suggested extension to the DO loop in FORTRAN IV. One of the few redeeming features of the language is that it can deal with DO loop controlled variables in a reasonably efficient way, while the example he quotes of a complicated ALGOL 60 for statement shows how unpleasantly messy that construction can become. It also shows one of its major weaknesses; to give a simple example, assuming i is not modified inside the controlled statement, how many times will the controlled statement in the following be executed:

for $i := 1$ **step** 1 **until** 10, $i + 1$ **while** $i < 20$ **do** . . .?

Does $i + 1$ mean 'add one to the last value of i for which the controlled statement is executed' (in which case the answer is 20) or 'add one to the last value of i tested' (in which case the answer is 19, the case $i = 11$ being omitted)?

It is significant that ALGOL 68 is much more restrictive than ALGOL 60 over the matter of for loops. If FORTRAN is to be improved (other than by the most effective means, i.e. total annihilation), an ALGOL 68 kind of extension would be preferable than one based on ALGOL 60, for example

DO $i = m$ TO n BY k WHILE l

where i stands for any INTEGER variable m, n and k for any arithmetic expressions delivering an INTEGER value, and l for any LOGICAL (Boolean) expression. If l is the logical constant .TRUE. the 'WHILE l' may be omitted; if k is the integer constant 1 'BY k' may be omitted; and (in order to preserve the 'upward compatibility' so beloved of Fortranites and which has frustrated so many improvements in the past) 'TO' and 'BY' may be replaced by commas.

I agree with Mr. Finn that k should be allowed to deliver a negative increment; but allowing REAL controlled variables, which his suggestion of REAL increments also presumably implies, would do no more than encourage bad programming practice among a group of programmers already too exposed to bad influences.

Yours faithfully,

B. L. MEEK

Computer Unit
Queen Elizabeth College
Campden Hill Road
London W8
20 March 1972