depth first search technique. The algorithm is presented in ALGOL-like terms and involves recursion. It is easily implemented using GRAAP and this coding is shown in Fig. 7 which, to emphasise the ease of implementation, includes statements of the original algorithm as comments between the pairs of symbols ¢{ and }¢ alongside the GRAAP source code. Other comments appear between the normal ¢ symbols.

For a graph with N nodes and E edges the algorithm requires $O(N + E)$ time. Table III shows the average time taken to find the biconnected components of a graph on N nodes with edge density ρ (i.e. $N^2 \rho 2$ edges) for $N = 10(10)50$ and $\rho = 0.3(0.2)$ 0.9. Each time is obtained by averaging over 50 randomly generated graphs. The timings confirm that the algorithm has $O(N + E)$ time complexity.

Average time, in seconds, to find the biconnected components of a connected graph with a given number of nodes and edge density.

4. Conclusions

The package described in the previous sections possesses the following attributes:
1. Extensibility.
2. Transparency of internal structures.
3. Operations which are natural to graph theory.
4. Ability to accommodate large graphs.
5. Easy implementation of graph algorithms.

Table 3

Edge density, ρ	0.3	0.5	0.7	0.9
No. of nodes, N				
10	0.053	0.063	0.076	0.089
20	0.189	0.255	0.305	0.349
30	0.405	0.522	0.649	0.735
40	0.706	0.920	1.125	1.385
50	1.099	1.475	1.784	2.154

Experience with the package has shown these attributes to be extremely useful in solving a wide variety of problems. Algorithms which have been implemented include (i) finding spanning trees, (ii) computing transitive closures, (iii) graph colouring, (iv) cycle generation, (v) finding shortest paths, (vi) finding blocks and cutnodes, (vii) solving partitioning problems, (viii) pathfinding in electrical networks, coming from many different scientific disciplines. The algorithms have been easily implemented using the GRAAP facilities and running times show the package to be efficient.

References

BRON, C. and KERBOSCH, J. (1973). Algorithm 457: Finding All Cliques of an Undirected Graph, *CACM*, Vol. 16 No. 9, pp. 575-577.

CHASE, S. M. (1970). Analysis of Algorithms for Finding All Spanning Trees of a Graph, Report No. 401, Department of Computer Science, University of Illinois.

CHRISTOFIDES, N. (1975). *Graph Theory—An Algorithmic Approach*, Academic Press, London.

CRESPI-REGHIZZI, S. and MORPURGO, R. (1968). A Graph Theory Oriented Extension to Algol, *Calcolo*, Vol. 5 No. 4, pp. 1-43.

DEO, N. (1974). *Graph Theory with Applications to Engineering and Computer Science*, Prentice-Hall.

EVEN, S. (1973). *Algorithmic Combinatorics*, Macmillan, New York.

FRIEDMAN, D. (1968). GRASPE Graph Processing: A LISP Extension, Report TNN-84, Computation Center, University of Texas, Austin.

GARSIDE, G. R. and PINTELAS, P. E. (1978). *GRAAP Reference Manual*, Computing Laboratory, University of Bradford.

GERHARDS, L. and LINDENBERG, W. (1979). Clique Detection for Nondirected Graphs: Two New Algorithms, *Computing*, Vol. 21, pp. 295-322.

JOHNSTON, H. C. (1976). Cliques of a Graph—Variations on the Bron-Kerbosch Algorithm, *Internat. J. Computing Information Sci.*, Vol. 5 No. 3, pp. 209-238.

KING, G. A. (1970). *A Graph-theoretic Programming Language*, Ph.D. thesis, University of West Indies.

PAULL, M. C. and UNGER, S. H. (1959). Minimizing the Number of States in Incompletely Specified Sequential Switching Functions, *IRE Trans. Electronic Computers*, EC-8, pp. 356-367.

PINTELAS, P. E. (1976). *An Algol 68-R Package for Handling Sets and Graphs*, Ph.D. thesis, University of Bradford.

RHEINBOLT, W. C., BASILI, V. R. and MESZTENYI, C. K. (1972). On a Programming Language for Graph Algorithms, *BIT*, Vol. 12, pp. 220-241.

TARJAN, R. (1972). Depth-first Search and Linear Graph Algorithms, *SIAM J. Comput.*, Vol. 1, pp. 146-160.

Book reviews

Automatic Speech and Speaker Recognition, edited by N. Rex Dixon and Thomas B. Martin, 1978; 428 pages. (*IEEE Press*)

This book is a collection of 38 papers on various aspects of speech and speaker recognition, originally published between 1972 and 1978. For someone wishing to get an indepth view of the field it would certainly save a lot of work in the library. The papers range widely and there are contributions from the UK, Japan and the Netherlands as well as from the United States.

The reader will find he needs to have a wide ranging preparation if he is to read and understand everything (acoustics, anatomy, phonetics, linguistics, mathematics, computing) but this is typical of the field. Five review papers on speech recognition and two on speaker recognition will provide a good overview of the main developments in the field for the non-specialist. If one believes, as does the reviewer, that progress in this area is crucial to a more effective use of computers in our society as a whole, the book contains evidence of solid if not dramatic progress in the last decade.

P. G. RAYMONT (Manchester)

Computer Security, by D. K. Hsiao, D. S. Kerr and S. E. Madnick, 1979; 299 pages. (*Academic Press*, $18.00)

This is another of a series of inter-related American research publications on computer security, and nearly 50 percent of it consists of references to other similar works. For once, however, the topic is seen as a wider ranging problem than teleprocessing access control, although this is still the major issue considered, whilst other important areas, such as power supply, flooding, corrupt input information and incorrectly timed use of file information, get but passing mention.

Some parts of the book are encouragingly written in layman's language, whilst others go into some depth on software matters. However technical the cause of problems, they have to be appreciated and evaluated by ordinary managers so that effective action may be taken. This volume provides a helpful survey of many aspects of computer security, but adds little new to the available information on the topic.

A. J. THOMAS (Sunbury on Thames)