UNIVERSITY OF QUEENSLAND
COMPUTER CENTRE

# COMPUTER CENTRE BULLETIN

Volume 2, Number 12.
1st December, 1969.

Editor:
H. L. Smythe.

# THIS EDITION

This month's Bulletin contains an *Index* which references all major articles and items of computing information in both Volumes 1 and 2. We hope this Index will become a yearly feature, and advise readers that back copies of the Bulletin are readily available if desired. In this issue, a postgraduate student of the Department of Computer Science takes a look at the provisions for legal protection for programs, and a brief note on magnetic tape subroutines on the GE 225, should be useful to clients.

With Volume 3 of the Bulletin already on the drawing board, we are revising and updating the mailing list for the New Year. As mentioned in the last issue, individual readers who wish to receive Volume 3, should complete the application form on the last page of this Bulletin. Organizations and departments, however, are not required to reapply. Any change of address should be notified promptly to the Editor.

The Editor would like to thank sincerely all those badgered and harassed people who have contributed to the preparation of the Bulletin this year. Their efforts were much appreciated, and the quality of their work is indeed worthy of a repeat performance next year. In particular, thanks are due to readers and clients outside the Computer Centre who have contributed ideas and articles. Their encouragement and response were especially heartening.

# INDEX OF VOLUMES 1 AND 2, COMPUTER CENTRE BULLETIN

*v = volume* *p = page*

## GE 225 FORTRAN IV

Clients have been experiencing some difficulties with specification statements on the GE 225, which are not always reliable in practice.

The following points may assist clients with such programming problems:

(a) When assigning variables to COMMON, assignment takes place in the order of appearance of names which is not necessarily their order in the COMMON statement.
For example, if the first two statements referencing A, B and C are:

```
COMMON A, B, C
DIMENSION B(10), C(5), A(5,10)
```

assignment is in the order that the variables appear in the COMMON statement, i.e., A, B then C. If the statements are reversed, thus:

```
DIMENSION B(10), C(5), A(5,10)
COMMON A, B, C
```

the variables are first encountered in the DIMENSION statement, and are assigned to COMMON in the order B, C then A.

(b) To avoid possible error in the assignment of locations for real variables in COMMON, always place integer quantities first, e.g., COMMON I,J,K,A,B,C is correct, but COMMON A,B,C,I,J,K would be incorrect.

(c) It is safest to use identical COMMON statements in the main program and in all subroutines. If a number of COMMON statements are involved, they must be kept in the same order in all routines.

(d) When using the EQUIVALENCE statement for arrays, it is advisable to use constant subscripts, e.g., (A(1), B(1)) rather than (A,B). As arrays are stored column by column, careful equivalencing of the appropriate elements automatically equivalences the following elements of both arrays.

# MAGNETIC TAPE SUBROUTINES ON THE GE 225

## CALLING ALL CLIENTS

A number of different tape subroutines are used in the GE 225 system, and include subroutines supplied with GE software, those written by Computer Centre staff, and special routines written by various customers. These routines handle read and write error conditions in several different ways.

The Computer Centre general purpose, magnetic tape, input and output subroutine, MTIO, (library number E2.200), has been revised to conform with GE standards regarding tape error conditions and recovery. The library number of this subroutine is now E2.200 rev. 1.

These modifications have been made to utilize a hardware facility which allows a variable amount of tape to be erased by means of an erase command. If bad spots exist on tape, causing errors on writing, then the faulty part of the tape can be erased, so that information may be written successfully on the tape. Tapes containing bad spots can thus be utilized without any problems, instead of being rendered unserviceable.

Customers using programs containing the Computer Centre subroutine, MTIO (E2.200), should now use the revised version. Customers with their own special versions of magnetic tape routines, should check to ensure that the routines will recover correctly from write errors. Computer Centre staff will advise any customer of the correct procedure to be followed when error conditions arise.

# LIBRARY ACCESSIONS

This section lists the books and periodicals relating to computing, that have been acquired by the Libraries of the University of Queensland in June and July 1969.

## JUNE ACCESSIONS

Fox, Leslie. *Computing Methods for Scientists and Engineers.* 1968. (517.6 FOX, Engin.Lib.)

Larson, Robert E. *State Increment Dynamic Programming.* 1968. (519.92 LAR, Maths.Lib.)

Welbourne, Daniel. ed. *Analogue Computing Methods.* 1965. (510.782 WEL, Physics Lib.)

Goodspeed, M.J. *Report on the Symposium on the Use of Analogue and Digital Computers in Hydrology.* Tuscon, Arizona; December 1968. 1969. (551.49018 GOO, Engin.Lib.)

Deutsch, Ralph *System Analysis Techniques.* 1969. (620.7 DEU, Physics Lib.)

Mischke, Charles R. *An Introduction to Computer-Aided Design.* 1968. (620 MIS, Engin.Lib.)

Trickett, E.S. *Computer Applications in Irrigation and Drainage.* 1969. (631.7 TRI, Agriculture Lib.)

Davis, Gordon B. *Auditing and EDP.* 1968. (658.15 DAV, Main Lib.)

Houghton, Bernard ed. *Computer Based Information Retrieval Systems.* 1968. (651.8 HOU, Main Lib.)

U.S. Department of Defence. *DOD and NASA Guide: PERT COST Systems Design.* 1962. (Q658.5 UNI, Accountancy Seminar Room)

## JULY ACCESSIONS

Burkhalter, Barton R. ed. *Case Studies in Systems Analysis in a University Library.* 1968. (025.1 BUR, Main Lib.)

U.S. Federal Council for Science and Technology. Committee on Scientific and Technical Information. *Guide-lines for the Development of Information Retrieval Thesauri.* 1967. (029.5 UNI, Main Lib.)

I.B.M. Scientific Computing Symposium on Combinatorial Problems, Yorktown Heights, N.Y., 1964. *Proceedings.* 1966. (510 IBM, Engin.Lib.)

Panel on the Role of Computers in Radiotherapy, Vienna, 1967. *Role of Computers in Radiotherapy.* 1968. (615.842 PAN, Physical Education Lib.)

Conference on Systems and Computer Science, University of Western Ontario, London, Ontario, 1965. *Systems and Computer Science.* 1967. (621.38195 CON, Engin.Lib.)

Blumenthal, Sherman C. *Management Information Systems.* 1969. (658 BLU, Main Lib.)

Cleland, David I. *Systems Analysis and Project Management.* 1968. (658.502 CLE, Main Lib.)

Conference on Graduate Academic and Related Research Programs in Computing Science, State University of New York, Stony Brook, 1967. *University Education in Computing Science.* 1968. (651.8 CON, Engin.Lib.)

## LEGAL PROTECTION FOR COMPUTER PROGRAMS

*K. Fitzgerald*

*In this article, Miss Kate Fitzgerald discusses the provisions that exist for the legal protection of programs in the United States and the United Kingdom. Australia, however, she finds, is lamentably slow to legislate in this increasingly-significant field.*

*Miss Fitzgerald holds a Bachelor of Science degree from the University of Queensland, and is a student in the Postgraduate Diploma in Automatic Computing.*

### INTRODUCTION

Protecting computer programs for better or worse has become a matter of importance because of the speed of software development. So much money is tied up in the industry that there are many people with axes to grind. In this article, I intend to look at the possibilities proposed by various groups, some of the actualities attained, and the advantages and disadvantages of protection.

## COPYRIGHTS

In the United States, two kinds of copyright are available. The first is referred to as a *Common Law Copyright*, which an author obtains the moment he produces a copyright-able work. Generally speaking, this must be original and, in some measure, creative. A *Statutory Copyright* is also available if the work is produced with a notice appended (usually in a prominent place) stating that the work is copyrighted. In addition, the program must be published. An author may register his copyright with the Copyright Office, and this will prove a help in establishing ownership, transferring rights in the copyrighted program, and bringing suit against a person for violation of copyright.

Copyright protects a program from outright copying, copying with minor changes, translation into other languages and more than likely from publication in other forms, e.g. magnetic, tape, punch cards, etc. The life of the copyright is twenty-eight years, and is normally renewable for one term.

However, this form of protection has some severe shortcomings which cannot be ignored. An author is not able to protect the ideas and techniques involved in his program from abuse, nor can he protect himself in any way against a similar work being produced independently by another worker.

The benefits, such as they are, allow companies to release programs for general use in return for a reasonable fee. These companies write programs aimed at small users and develop programs beyond their immediate needs, encouraged by the knowledge that at least part of the cost will be carried by somebody else.

In April 1964, in a reversal of a previous decision, the United States Copyright Office declared that programs were copyright-able. The case for copyrighting computer programs had been discussed widely prior to this decision because of the phenomenal growth in the volume of software being produced. By mid 1966, there were fifty-two computer programs copyrighted by individuals.

In the United Kingdom, the situation with the Copyright Law is much the same as in the United States, although copyright is obtained automatically. Some people in the United Kingdom feel that adequate protection for programs could successfully be written into a new part of the Copyright Law being proposed at the moment.

## PATENTS

The first program patent was awarded in the United States in mid 1968 to a sorting program filed by a company in April 1965, after the withdrawal of previous legislation. The question of patenting computer programs has been before the Patents Examining Staff of the U.S. Patents Office for many years.

The major difficulty lies in the fact that, if a computer program does not fall within one of the statutory classes laid down by the Patents Law, i.e. (1) manufacturers, (2) machines, (3) compositions of matter and (4) processes, then the program is not patent-able. The problem is to fit a computer program into one of these classes.

A *process* patent is probably the best means to seek, for one may view a computer as a giant plug-board that the program wires and rewires to different jobs. Process patents contain "method claims" defined as "steps for performing a specified function". Others, however, feel that a program represents a system of knowledge, not a process. While it is true that mathematics are used to write a program, this is also the case in the logic design of hardware where nobody quibbles about patents. A computer may be viewed as a collection of hardware, hamstrung until it receives a program. Consequently, a program is, in this sense, part of a machine, and a machine patent should be sought.

Tests of patent-ability include: is the program new, useful and novel in the sense of not being obvious to some person skilled in the art of programming? Patent Law requires that a written description of the invention be submitted, explaining to any person skilled in the art of programming, how the program works and the best way to use it.

The advantages of the patent are that developers of programs are given protection of a higher order than allowed by copyright, and are entitled to income from their invention. The patent takes much longer to obtain, however, and is expensive, giving the developer the problem of protecting the program prior to the award of a patent.

In the United Kingdom, patent-ability of computer programs is currently being investigated with a view to new legislation with respect to the Patents Law. As far as can be ascertained, no patents have been awarded, and the situation is much the same as in the United States.

## TRADE SECRET PROTECTION

Trade secret protection for a program requires that:

(1) The program is kept secret, that is, under lock and key.
(2) It, and any documents related to it, are clearly marked 'secret' or 'confidential'.
(3) Only a limited number of authorized personnel have access to it.

Should a case go to court, it is essential the above provisions are observed, for the strength of the injured party's case depends much on his efforts to keep his secret. Generally speaking, the greater the amount of work it represents and the greater the loss to him as the result of its disclosure, the more serious is the harm done to him, and damages are awarded accordingly.

Maintaining a program as a trade secret affords best protection for owners, but many cases of guarding trade secrets have been lost for the Court ruled that there was no trade secret, i.e., the three conditions above had not been adhered to adequately. Some part of a program could possibly be 'locked' into a computer system so that it would be very difficult, or even impossible, for anybody to have access to it. The person seeking protection must also show that a great deal of an allegedly infringing program has been copied from his trade secret. Such protection is useless to owners who wish to distribute programs as a part of their business, a situation which frequently arises. Such people will have to seek protection in some other form.

A similar situation prevails in the United Kingdom with respect to trade secrets.

## OTHER KINDS OF PROTECTION

One other significant method of computer program protection lies in *contractural* arrangements made between program developers and users. Programs are made available to users (at a price) under a contract that obliges the user not to disclose the nature of the program to anybody else. A disadvantage of this method is that there is, in the eyes of the law, no obligation on the user to restrain any third person who violates the stipulation of the contract.

Each of the methods so far discussed has its advantages and drawbacks. Perhaps a logical step is to begin to combine them, and hope for the best of both worlds and forget the worst. This is successful in certain cases, e.g., copyright can be linked with contractural obligations, and patents may be sought if the material qualifies for one. Many patent attorneys recommend that a program seeking patent protection should be treated as a trade secret until a patent is awarded.

Consider, however, an advanced program, incorporating important new methods and techniques, representing many hours of work, ready for the developer to distribute, and still unable to be patented. In this case, the possibilities of contract and copyright scarcely seem adequate. New laws relating to unfair competition are needed to cover situations like this in the computer industry. Some people believe, also, that there is protection in the unique complexity of their program; this would greatly depend on the value of the program to other people, but may very well work in cases where most of the value is for the owner.

## CONCLUSION

Another aspect that must be considered is the kind of rights companies, seeking protection, have in establishing monopolies. While they are entitled to reward for investing effort in a program, and should be empowered legally to deal with those who refuse to pay for fair use of the material, companies are prohibited from abuse of private and property rights by laws like the Restrictive Practices or the Competition Law, the branch of the law which establishes free and fair competition. How far can companies tie up their employees so that they can secure protection for their programs? There is a multitude of questions that only time will resolve through court decisions and legislation.

It is apparent that Australian law has made few provisions for the protection of computer programs. Not only does this reflect the general lack of public awareness about the issue, but it may also be explained by the way software is tied up in Australia. It is not the huge industry here that it is in the United States, and programs are released by computer manufacturers to go with their machines as compared with software developed by companies specializing in software alone.

Protection in the forms of copyright and patent, is generally held in the computer industry to be in the interests of furthering development because of the financial reward that is a spur to encourage people to put their work in a common pool.

# BULLETIN MAILING LIST

The Bulletin is distributed according to two mailing lists, one for organizations and departments which is permanent, and the other for individuals, maintained on an annual basis.

To receive Volume 3 of the Bulletin, please complete the appropriate section of the following form, and return it to -

The Editor,  
Computer Centre Bulletin,  
Computer Centre,  
University of Queensland,  
ST. LUCIA,  
Queensland, 4067.

---

Application for Individual Circulation

NAME: ..............................

POSITION: ...........................

ADDRESS: ............................

....................................

....................................

Application for Organizational Circulation

POSITION: ...............................

ORGANIZATION: ...........................

ADDRESS: ................................

........................................

........................................

NUMBER OF COPIES: .......................

---

SIGNED: .................................... DATE: ...............................

---