National Law School Journal

Volume 1 | Issue 1

Article 3

1-7-1989

Copyright in software

Kanwal Puri

Copyright in software

KANWAL PURI

Recent developments in the law in some overseas countries and in the computer and related industries have focused increasing attention on copyright as a vehicle for protection of proprietary rights in software. These include, for example:

(i) The "untieing" of software/hardware packages and the emergence of packaged computer programs.

(ii) The mass marketing of and demand for software brought on by the personal computer in home, school, university ("courseware") and small-business computers.

(iii) The tremendous success of the video game industry.

(iv) Changes effected by the Australian Copyright Amendment Act, 1984, the United Kingdom's Copyright (Computer Software) Amendment Act, 1985, the United States' Software Copyright Act, 1980, and the judicial confirmation of copyrightability of computer programs in some other countries, e.g., Canada: *Apple Computer Inc* v *Macintosh Computers Inc*.[1]

The Copyright Act, 1962 (NZ) does not expressly enumerate computer programs as copyrightable subject-matter. No consideration was given to the special needs of the computer industry when this statute was passed, no doubt because the computer industry was still in its relative infancy. It is therefore, not possible to discern clearly whether the legislature intended to include computer programs within the scope of copyrightable subject-matter in the Act of 1962. Today, computer software is extremely important, yet there is no clear law on its protection. In New Zealand, the Industrial Property Advisory Committee (IPAC) has recommended that the Copyright Act, 1962 be amended to confirm protection of computer programs.[2] Since it is unknown whether and when IPAC's recommendations will be implemented by the legislature this paper will consider the state and effect of the present law in New Zealand. For this purpose some knowledge of copyright system is required. Also, a description of the operation of a computer sufficient to provide the background to a determination of the issues seems desirable.[3]

[1] (1986) 28 DLR 178.

[2] See *The Legal Protection in New Zealand for Computer Programs* (Preliminary Report 10 March, 1984 and Final Report 18 March, 1986) Department of Justice, New Zealand.

[3] See below the appendix to this paper, Glossary of Computer Jargon.

A copyright is automatically born the instant thoughts and ideas are transferred from an author's mind to paper or some other material form. There is absolutely nothing to be done in order to have a legally enforceable copyright. Protection by copyright is automatic, immediate and free from formalities. The protection is not perpetual, however. The general rule is that copyright subsists for the life of the author plus 50 years. At the expiration of this period, the work falls into the public domain. The Copyright Act, 1962 confers copyright protection on eight general categories of copyrightable creations, viz., literary, dramatic, musical, artistic, sound recordings, cinematograph films, broadcasts and published editions. Once entitlement to protection under one of these categories is satisfied, the copyright owner has certain exclusive rights in regard to that particular work. The exclusive rights include the right to reproduce the work in any material form and the right to re-create the work in a different version (adaptation). In a broad sense, the copyright law is designed to safeguard the copyright owner against unauthorised making by others of reproductions that might otherwise have been sold by the legitimate owner

What exactly is copyrightable? Copyright law protects the expression of ideas, not the ideas themselves. The ideas must be in some retrievable form, for example, in handwriting or print, or recorded on magnetic tape. In addition, the Act requires that the creation be original. But copyright requires only a minimal degree of originality to qualify for protection. Almost anything is considered original so long as it is not copied. The expression "original" does not mean that the work must be the expression of original or inventive thought. "The originality which is required relates to the expression of the thought. But the Act does not require that the expression must be in an original or novel form, but that the work must not be copied from another work — that it should originate from the author".[4] Originality is a matter of degree, depending on the amount of skill, judgment or labour that has been involved in making the work.[5]

There is an important limitation placed on copyright. It is subject to the right of all persons into whose possession the work comes, to deal fairly with it. The concept of fair dealing permits the reproduction, for legitimate purposes, of material taken from a copyright work to a limited extent that will not cut into copyright owner's potential market. Thus sections 19 and 20 of the Copyright Act, 1962 limit copyright owner's rights where the use of the work is made for purposes of research or private study. Use of the copyrighted material in a reasonable manner without the owner's consent may also be made for purposes of criticism, review, reporting of current events or reporting of judicial proceedings. The doctrine of fair dealing attempts to balance the rights of the copyright owner against the interests of society. So in an appropriate situation, the copyright owner's interest in a maximum financial return may be subordinated to the greater public interest in the development of education, research and industry.

Besides sections 19 and 20, which permit limited copying for specified purposes by private individuals for their personal or private study, section 21 contains special

[4] *University of London Press Ltd.* v *University Tutorial Press Ltd* (1916) 2 Ch 601, 608-609.

[5] *Ladbroke (Football) Ltd* v *William Hill (Football) Ltd* (1964) 1 WLR 273, 277-278.

exceptions for libraries, universities and schools. Thus, copies of literary or other works may be made or supplied by or on behalf of a teacher or a librarian at these institutions. Section 21(1) lists four conditions which must be fulfilled to make an otherwise illegitimate copying permissible:

(i) the copies may be supplied only to persons who satisfy the teacher or librarian that they require them for research and private study;

(ii) no copy is to extend to more than a reasonable proportion of the work in question;

(iii) no person is to be given more than one copy of the same piece of work; and

(iv) no payment apart from the actual cost is to be taken for supplying the copies.

Another important exception to infringement is to be found in section 21(4). Under this provision, the copyright in a literary or other work is not infringed by reason only that the work is copied, or an adaptation thereof is made either *(i)* in the course of instruction at a university or school or elsewhere by a teacher or student; or *(ii)* as part of a question to be answered in an examination or in answer to such a question. The first part of this exception appears to be very broad as it does not seem to place any limit on the amount of the work or adaptation which may be copied or, indeed, on the number which may be made.

Before discussing copyright protection in software in detail it is necessary to consider briefly the nature of software. "Software" can have various meanings. This paper equates it principally with computer programs. A computer program is a set of instructions designed to cause a computer to perform a particular function or to produce a particular result. The instructions may be written in any of three different levels of computer languages. Computers can "understand" (i.e. execute) only programs in "machine language" — the lowest level. A machine language instruction is a cluster of "O" and "1" symbols called "bits", which are the only symbols recognised by digital computers. A piece of software consisting of a sequence of machine language is described as an "object code". To make it easier for humans to read and write programs, two "higher" levels of languages exist — "assembly language" and "human-readable language". Assembly language instructions consist of alphanumeric labels rather than bits (e.g., HLT = "halt processing"). To be executed by computer, the alphanumeric instructions must be translated into their corresponding clusters of bits by another computer program known as an "assembler". Human-readable language, such as FORTRAN or COBOL, that employs English-like words is easier to use and understand than assembly or machine language. A computer program known as a "compiler" translates human-readable language into the corresponding object code. Programs written in assembly or human-readable languages are referred to as "source programs" or "source code".

To be used in a computer, object code must be stored in a memory device such as a floppy disk or a "read-only memory" (ROM). A ROM consists of a semiconductor "chip" (integrated circuit), about the size of a small fingernail, enclosed in an outer package that is plugged into the circuit boards of the computer. Because ROMs are easy to unplug, they are readily inserted into a laboratory device that can copy the object code into another ROM or on to paper. The "output" of a computer may be in various forms, for example, a paper printout, a visual display on a monitor, or the control of other machinery.

There has so far been no decision of the New Zealand courts relating to the subsistence of copyright in computer software, though in an interlocutory proceeding the Auckland High Court seems to have recognised that computer programs are susceptible to copyright protection: *IBM* v *Computer Imports Ltd.*[6] It is worth noting that the Federal Court of Canada has recently held that under Canadian Copyright Act, 1970 (similar in many respects to the New Zealand Copyright Act, 1962) computer software is entitled to copyright protection and this decision may well be of persuasive value in New Zealand courts: *Apple Computer Inc* v *Macintosh Computers Inc.*[7] In the United Kingdom provisional opinion has been expressed by judges in interlocutory proceedings that a machine code program (object code) is either a reproduction or an adaptation of an assembly code program.[8] The Supreme Court of South Africa has also held that a program in an object code may be a literary work.[9] Given that the United States, United Kingdom, Canada, Australia, Japan, Netherlands, West Germany, South Africa and India have all confirmed (either through legislative amendment or judicial pronouncement) that software is entitled to protection under the copyright law, it seems safe to assume that the courts in New Zealand would also be prepared to hold that copyright subsists in software.

Is a source program entitled to protection under the Copyright Act, 1962? It may be recalled that the written program in source code is directed to a human reader and not to a machine. Its essential function is to record and communicate programming instructions. Is it an original literary work? In interpreting the expression "original literary work" the following principles have been established:

(i) In order to satisfy that description the work must afford either information and instruction or pleasure in the form of literary enjoyment to human beings.[10]

(ii) A literary work includes "any written table or compilation".[11]

(iii) There is no necessity for a literary work to have any literary merit or quality.[12]

[6] *The Dominion Wellington*, 21 July, 1986, p 21.

[7] (1986) 28 DLR 178.

[8] *Sega Enterprises Ltd* v *Richards*, (1983) FSR 73. See also *Thrustcode Ltd* v *W W Computing Ltd* (1983) FSR 502.

[9] *Northern Office Micro Computers Pty Ltd* v *Rosenstein* (1982) FSR 124.

[10] *Exxon Corporation* v *Exxon Insurance Ltd* (1982) Ch 119; *Hollinrake* v *Truswell* (1984) 3 Ch 420, 428.

[11] Section 2(1), Copyright Act, 1962 (NZ). See also at 608 where Peterson, J, said, "In my view the words 'literary work' cover work which is expressed in print or writing, irrespective of the question whether the quality or style is high". This observation was approved by House of Lords in *Ladbroke (Football) Ltd* v *William Hill (Football) Ltd* (1964) 1 WLR 273, 285, 291.

[12] *Robinson* v *Sands & McDougall Pty Ltd* (1916) 22 CLR 124, 133.

(iv) It is sufficient if the work supplies information capable of conveying an intelligent meaning, even if only to a limited group with special knowledge. It is immaterial that this information is not expressed in the form of words, phrases or sentences.[13]

(v) "Originality" means that the author's own skill and labour must be involved, though the degree of such skill and labour may be slight. It is not necessary that new or inventive ideas be contributed by the author, but work must be more than a copy of other material.[14]

It is submitted that source program clearly satisfies these tests.[15] Whether or not a literary program must be in writing, a source program is in writing. Although the substance of the program may be expressed in assembly code, this is a language which is readily intelligible to anyone versed in computer science. Each program is the product of skill, time and effort. It is a particular kind of vehicle for the communication of useful information to persons who may desire it. Before the source program is transformed into another medium (viz object program), it has an existence which is entirely independent of the machine. It is capable of conveying meaning as to the arrangement and ordering of instructions for the storage and reproduction of knowledge. These features of a source program are sufficient to bring it within the scope of literary copyright. However the conclusion that copyright subsists in source program may not have great practical significance because software is distributed only in object form. The underlying source code is retained by the software producer and is not publicly available. This is common practice because the user needs object

[13] Thus a telegraphic code has been held to be a literary work though the words of the code were meaningless: *D P Anderson & Co Ltd* v *Lieber Code Co* (1917) 2 KB 469.

[14] *Pitman* v *Hines* (1884) 1 TLR 39; *University of London Press Ltd* v *University Tutorial Press Ltd* (1916) 2 Ch 601, 608-609; *Ladbroke (Football) Ltd* v *William Hill (Football) Ltd* (1964) 1 WLR 273, 277-278.

[15] In the court of first instance in *Apple Computer Inc & Another* v *Computer Edge Pty & Another* (1984) 1 IPR 353, Beaumont, J, held that copyright did not subsist because a source code was not literary work. The judge argued that a literary work is something which was intended to afford "either information or instruction or pleasure in the form of literary enjoyment". He found support in a statement appearing in *Hollinrake* v *Trustwell* (1894) 3 Ch 420 by Davey, LJ, [recently approved in *Exxon Corporation* v *Exxon Insurance Consultants International Ltd* (1982) Ch 119 (CA)]. Except Deane, J, all other judges (four High Court and three Full Federal Court) held that a source program is protectable because it is an original literary work. Deane, J, considered this unnecessary to decide (see the High Court's judgment reported in (1986) 6 IPR 1 and Full Federal Court's in (1984) 2 IPR 1); *Hollinrake* was a case in which a cardboard sleeve chart, i.e., a representation of a sleeve designed for a lady's arm with certain scales of measurements on it intended for practical use in dressmaking, was held not to be a literary work. Davey, LJ, said at 428: "a literary work is intended to afford either information and instruction, or pleasure, in the form of literary enjoyment". In *Exxon*, the English Court of appeal denied the character of a literary work to the single work "*Exxon*" standing alone : "It is a word which though invented and therefore original, has no meaning and suggests nothing in itself". It is submitted that writings in the form of mnemonics of alphanumeric code are included within the definition of compilation. Several authorities may be cited in support, e.g. *Pitman* v *Hines* (1984) 1 TLR 39 (Ch D) (shorthand notations), *Ladbroke (Football) Ltd* v *William Hill (Football) Ltd* (1964) 1 WLR 273 (HL) (football betting coupons), *University of London Press Ltd* v *University Tutorial Press Ltd* (1916) 2 Ch 601 (examination papers). It is submitted that the statement made in *Hollinrake* was not, nor was it intended to be, exhaustive. But even if that statement were applicable, the source programs did convev information and instruction.

program for execution on a computer and may not have a compiler or assembler to convert from source to object form. Moreover, software producers restrict the dissemination of the source code to shield the structure and logic of the software from disclosure to competitors and potential competitors.

As stated above, most software is now sold in a more basic form called "object code" that is directly intelligible to the machine. Some people have attempted to circumvent the copyright protection of source programs by copying only the object code version of the program and maintaining that object code is not covered by copyright law. Opponents of copyrighting object code ground their objections both on the language of the statute and on the differences between the policies underlying copyright law and those underlying the patent law. Some would deny copyright to object code on the basis of these arguments:

(i) The copyright statute protects only works intended to communicate to humans. Because the use of object code in machine-usable form conveys nothing of the programmer's original expression to the user, but instead merely controls the computer, the opponents conclude that object code could not qualify for copyright protection. Gibbs, CJ, summed up this argument in a succinct manner:

> "It seems to me a complete distortion of meaning to describe electrical impulses in a silicon chip, which cannot be perceived by the senses and are not intended to convey any message to a human being and which do not represent words, letters, figures or symbols as a literary work; still less can a pattern of circuits be so described".[16]

[16] *Computer Edge Pty Ltd* v *Apple Computer Inc* (1986) 6 IPR 1, 9 (hereinafter referred to as the *Apple* case). In May, 1986, the High Court of Australia in a split decision (3:2) held that copyright did not subsist in software. In all, nine judges were involved from the trial to the highest appellate court: five of them held that copyright protections did not extend to computer programs in object code. The proceedings arose out of the importation into and sale in Australia by Computer Edge (a Victorian Co) of micro-computers under the name "Wombat". Apple Computer Inc (a Californian Co) filed civil action against Computer Edge seeking injunctions, damages and an account of profits under the Trade Practices Act, 1974 and the Copyright Act, 1968. Apple, Manufacturer and seller of micro-computers under the name of "Apple II", claimed that to the knowledge of Computer Edge (formerly an Apple dealer) the Taiwanese manufacturer of "Wombat" had copied Apple's software. Apple alleged that three silicon chips (2 ROMS and 1 EPROM) installed in the Wombat were unauthorised copies of Apple's six ROMS installed in Apple II microprocessor. Apple claimed ownership of copyright *(i)* in the "Applesoft" source and "Autostart" source as original literary works; and *(ii)* "Applesoft" ROM (in machine or object code) and "Autostart" ROM (in machine or object code) as original literary works. The second claim to ownership was supplemented by stating two alternative grounds of ownership: *(a)* Apple's ROMs (i.e. object programs in the silicon chips) were adaptations or translations of the source programs; or *(b)* Apple's ROMs constituted a reproduction in a material form of the Applesoft and Autostart source programs. The main issue was whether copyright subsisted in software. This involved a difficult problem of construction of the Copyright Act, 1968, Beaumont, J, held that neither the Applesoft source nor the object program was an original literary work. On appeal, the Full Court of the Federal Court held by a majority (Fox and Lockhart, JJ, Sheppard, J, dissenting) that Apple ROMs were translations or adaptations of the source code programs and that Wombat. ROMS and EPROM were unauthorised reproductions. The Full Court was unanimous in holding that source code programs were protected by copyright. On further appeal to the High Court, the majority (consisting of Gibbs, CJ, and Brennan and Deane, JJ,) in restoring Beaumont, J's, decision, held that Apple's ROMs (object code) were not protected by copyright. Further, the ROMs were not translations or adaptations of

(ii) Copyright of the source program cannot automatically protect the object code version because the latter is not an "adaptation" of the former in the sense the word "adaptation" is used in the Act. Furthermore, an adaptation of a literary work should itself qualify as a literary work, but an object program is not a "work".

(iii) The notion of reproduction within the meaning of the Copyright Act requires that the infringing work must sufficiently resemble the copyright work. In other words, no infringement would occur if the reproduction does not have a visible similarity to the work reproduced. That is, when the authorities say that there should be "a sufficient degree of objective similarity between the two works",[17] this signifies a similarity of the physical manifestations of the work, not simply the works embodied therein. This view was adopted by the High Court in the *Apple* case.

(iv) Finally, many potential objections to copyrighting object program stem from the concern that copyright protection would undermine the purpose of the patent scheme by granting the functional equivalent of the more extensive patent monopoly to a work that has not been examined under the high standards of patent law. These arguments can rest upon any of three views of the nature of object code: that it is a "machine part", an embodiment of a useful "art", or a "process", all of which are subject-matter of patent law.

It is submitted that object code both is and should be protected by copyright law. It is further submitted that the current Copyright Act, 1962 does provide protection to object form software. Several arguments against copyright protection of object code will be examined in the following paragraphs to show that they are unpersuasive. It is further submitted that the protection of object code is consonant with the purposes of the copyright scheme and well suited to the characteristic needs of the software market.

It is submitted that the existing New Zealand Act is largely sufficient in providing copyright protection to all forms of computer programs though some amendments may be made for the purposes of clarification and removal of doubts. The Intellectual Property Advisory Committee has made recommendations to the same effect.[18] While rejecting the suggestion of patent protection for computer software, the Committee has recommended that urgent action be taken to amend the Act of 1962 to *confirm* that it does cover computer programs: "That the Copyright Act, 1962 should be amended to apply in relation to a computer program as it applies in relation to a literary work and that it should be a retrospective amendment... since it is intended as

the source code programs. Deane, J, went further to hold that Apple's ROMs were not reproductions in a material form of the written expression of the programs expressed in source code

[17] See *Francis Day & Hunter Ltd* v *Bron* (1963) Ch 587, 614; *Lend Lease Homes Pty Ltd* v *Warrigal Homes Pty Ltd* (1970) 3 NSWR 265, 273; *S W Hart & Co. Ltd* v *Edwards Hot Water Systems* (1985) 59 ALJR 729, 730.

[1ε] See *The Legal Protection in New Zealand for Computer Programs* (Preliminary Report 10 March, 1984 and Final Report 18 March, 1986) Department of Justice, New Zealand.

a *confirmation of existing rights*[19]. It is arguable, however, that in the light of the Australian High Court's recent judgment of 6 May, 1986 in *Apple* case, the above italicised statement may lack validity. In other words, the Apple ruling may be said to represent the state of the law in New Zealand prior to the 1984 Australian amendment. It is submitted that that is not the correct inference.

In *Computer Edge Pty Ltd* v *Apple Computer Inc*[20], Apple claimed that Computer Edge's importation of a "Wombat" computer was an infringement of its copyright in two computer programs, known respectively as "Applesoft" and "Autostart". Computer Edge did not deny piracy but contended that Apple's object programs (embodied in ROMs) were not protected by copyright because they were *inter alia* not literary works. At first instance Apple failed to persuade Beaumont, J, that its object programs were protected by copyright under the Copyright Act, 1968 (Australia). The judge held that neither the object programs nor the source programs from which they were derived were original literary works capable of copyright protection. Subsequent to the decision, the Copyright Amendment Act, 1984 was passed which conferred copyright protection on computer software. But since the amendment did not apply retrospectively, the judge's decision remained intact. However, on appeal, the Full Bench of the Federal Court overturned Beaumont, J's, decision by a majority (Fox and Lockhart,. JJ, Sheppard, J, dissenting) and held that the object programs contained in Wombat ROMs and EPROM (erasable programable read only memory) were reproductions of the programs contained in the Apple ROMs, which were themselves adaptations (i.e. translations of their respective source code programs). The Full Court was unanimous in concluding that the source code programs were new and original literary works in which copyright subsisted, but divided on the adaptation issue. On further appeal, the majority of the High Court (Gibbs, CJ, Brennan and Deane, JJ, Mason and Wilson, JJ, dissenting) held that, under the Copyright Act, 1968 (prior to its amendment in 1984 specifically to deal with computer programs), copyright protection did not extend to computer programs in object code. The High Court did not distinguish the position of machine readable object code in ROM or in disk or tape nor did it make any distinction between operating system software and application software (nor, for that matter, dâta in any machine readable medium). The source programs, which were written by human hand on paper, were held by the majority of the Court (Deane, J, not expressing his opinion on this matter) to be literary works and therefore protected under copyright law. In short, the Australian High Court's majority found as follows :

(*i*) To enjoy protection, Apple's object programs should qualify as *literary works* themselves, or *translations* (i.e. adaptations) of the source code, or *reproductions* of the source code in another material form.

(*ii*) Object programs, either as electrical impulses or fixed in a permanent state in ROM, did not constitute literary works and were not subject of copyright protection.

[19] Emphasis supplied. *Ibid* Final Report, p. 16.

[20] (1986) 6 IPR 1 [The High Court of Australia reversed Full Federal Court's decision reported in (1984) 2 IPR 1 and restored Beaumont J's decision reported in (1983) 1 IPR 353]. For a detailed statement of facts, see above note 16.

(*iii*) An object program was not a translation (i.e. adaptation) of the source code.

(*iv*) A reproduction must have a visible similarity to the work reproduced.

It is submitted that the Australian High Court's decision would not be followed in New Zealand. Firstly, that decision does not have any precedent value even in Australia because of the enactment of the Copyright Amendment Act, 1984 which has extended copyright protection to all forms of computer software to the same extent as it applies to literary works. Secondly, the majority's reasoning on some points, it is respectfully submitted, is not unimpeachable (e.g. in relation to the nature of "reproduction"). It is also arguable that the decision ran contrary to the Australian Copyright Act. Thirdly, there are some material differences between the wordings of the Australian Copyright Act, 1968 (pre-amendment) and that of the New Zealand Copyright Act, 1962, which are likely to sway New Zealand courts in favour of copyright protection to object programs.

The High Court's (majority's) main objection to giving copyright protection to the object programs was that they were not literary works. Their findings on this point may be summarised below :

(*i*) Object programs existed in the form of sequences of electrical impulses, or possibly in the pattern of circuits that when activated generated those electrical impulses.

(*ii*) They were not expressed in writing or print. Section 22(1) of the Copyright Act, 1968[21] should be read with section 10(1)[22] to construct "writing". Material form still requires writing. Moreover, ROMs were not within the definition of "work" as stated in section 10(1).[23]

(*iii*) They were not visible or otherwise perceptible, and were not intended to be capable by themselves of conveying a meaning which could be understood by human beings. The fact that the object programs might have been printed out in binary or hexadecimal form was not relevant.

(*iv*) A material form is a form which can be perceived by the senses. But a form which is not perceptible to sight, touch or hearing is not a material form. The electrical charges which constitute the object programs cannot be seen or touched or heard or, if they can, they do not communicate the letters of the original literary work, the source program.

There are several problems with these findings. First, there is the objection regarding lack of accessibility. The copyright law's tradition has been to protect works produced in written text. In the present writer's view, the requirement of "readability" or "appearance to the eye" or "visibility" requires no more than that there be a method by which the work in which copyright is claimed and the work which is alleged to

[21] Equivalent to section 3(8). Copyright Act, 1962 (NZ).

[22] Equivâlent to section 2(1), Copyright Act, 1962 (NZ).

[23] The expression "work" is not defined in the New Zealand copyright legislation.

infringe can be visually compared for the purpose of determining whether copying has occurred. Since, in this case, the programs could be "read" out of ROM and so compared, this requirement is met. Further, the definition of "literary work" is not exhaustive. It may well take account of modern means of communication and of recording information, which have moved so much (and so rapidly) into the electronic field.

Second, there is the problem of medium-message dichotomy. A programmer writing a program is not thinking about the specifications of the ROM chip voltage levels, or logic gates, or conditioning the circuitry of the CPU, or of providing specifications to the manufacturer of the ROM chip for the construction of that chip. Indeed, the medium in which the program eventually finds itself (disk or chip) is not of concern to him or her. The programmer is thinking of the computer as having a certain number of registers and of being able to perform a certain number of limited operations and of moving information between those locations.

The third problem relates to perceptibility-visibility question. The majority relied heavily on the point that an object program could not be perceived by the senses and was not visible. This requirement was said to be found in the definition of expressions like "writing" and "work" in the Act. The inspiration for this requirement comes from an oft-cited (though obiter) remark of Holmes, J, of the Supreme Court of United States in a 1908 case called *White-Smith Music Publishing Co* v. *Appollo.*[24] The learned judge had said:

> "The ground of this extraordinary right [meaning copyright] is that the person to whom it is given has invented some new collocation of visible or audible points,—of lines, colors, sounds, or words."

The case was decided under the Act of 1897 (based on the UK Act of 1842). The judge was concerned about tangibility; the purpose was to exclude ideas or mental conceptions. The judge wanted to ensure that the subject-matter should be durably perceptible — should be something concrete so that it may be possible to trace ownership. It is submitted that in that sense, a ROM chip is tangible, is durably perceptible and there is no difficulty regarding tracing the true owner.

The *Apple* majority conducted the case on the footing that a ROM embodied a series of electrical impulses stored in the chip. The judges should have perhaps conducted it on the footing that object program contained written description in binary or hexadecimal notation. But even if they are to be treated as electrical impulses, those impulses serve to identify a set of instructions which originated from written description.

Fourth, there was the question of writing requirement. The majority's fundamental objection to giving copyright protection to object programs was that they were not in writing. This conclusion was supported by the definition of "literary work" which includes "a written table or compilation".[25] Must a literary work be in writing?

[24] (1907) 209 US 1

[25] Emphasis supplied. Note that the New Zealand analogue in section 2(1) contains the word "any" instead of "a".

Ordinarily and traditionally it is true that a literary work would take a written form. But the Act does not require it to be so. Indeed, section 22(1)[26] identifies the time when the work "was first reduced to writing *or some other material form*".[27] There seems no reason to doubt that a literary work is made and is entitled to copyright protection from the time it is first recorded or transformed on to a disk or a chip, if that is the first material form that the work takes. It is submitted that there is no logical distinction between a magnetic and electronic medium and that an object code stored in a chip is a "material form". Gibbs, CJ, however expressed a contrary view in the *Apple* case and held that the object code was not a translation of the source code because the turning into electrical impulses was not "the expression of rendering of something in another form". In the same Court both Masson and Wilson, JJ, thought otherwise, as did Lockhart and Fox, JJ, in the Full Federal Court. Gibbs, CJ's, view was therefore a minority opinion.

To sum up, it is submitted that an object code falls into the category of "literary works" and is protected by copyright law because it satisfies the "writing" requirement of the statute. It may be recalled that the definition of "writing" includes "any form of *notation*" which the Oxford English Dictionary defines as "representing of numbers, quantities, pitch and duration of sound, etc., by symbols". A straightforward reading of the language of the New Zealand Copyright Act of 1962 indicates that object code, when printed on paper or stored in a memory device, satisfies the material form criterion. Object code written out on paper may certainly be directly perceived. Object code stored in machine-usable media — such as magnetic tape, floppy disk, or ROM — can be perceived or reproduced with the aid of a machine, because a computer can produce a printed copy of the object code. In the case of programs written in assembly or a high-level language, the object code produced by an assembler or compiler is just as much an original work of the author of the source program as is the source program itself, because object code is essentially a mechanical translation of the source program into another language Finally, if there is no copyright in the object programs which are a natural and necessary derivative of the source programs, there is no point in protecting the source program.

The majority of High Court also, however, held that the object program was not a "translation" of the source program and hence not an "adaptation" of a literary work within the meaning of the statute. Adaptation means changing the work's form or shape rather than its essence. According to the Concise Oxford Dictionary, adaptation means to modify, alter, adjust, *make suitable for a purpose*. In the majority's view the programs in object code were not adaptations, i.e. translations[28] of the program in source code for the following reasons :

(*i*) The Act does not contain a definition of "translation". The shorter OED defines "translation" as,—

[26] Equivalent to section 3(8) of the New Zealand Copyright Act, 1962.

[27] Emphasis supplied.

[28] Section 10(1)(*c*)(*i*), Copyright Act. 1968 (Cth).

> "The action or process of turning from one language into another" [described by the majority as the "primary" meaning].
> "The expression or rendering of something in another medium or form" [described by the majority as "transferred and figurative" use].

In the context of the definition of "adaptation", the primary, meaning is intended. If that were not so, the provisions of paras (*a*), (*b*) and (*c*)(*ii*) of the definition would appear to be surplusage. The majority argued that clearly, the programs in ROMs were not a translation because they were not turned into another language, not even into another computer language — they were turned into electrical impulses. Even in the figurative and transferred sense, ROMs did not portray or represent the source code in a different form or medium; rather they were the means of putting into action and making effective the instructions written in source programs.[29]

(*ii*) The definition of "adaptation" in section 10(1)[30] "is an exhaustive one".[31]

It is submitted that there is no authority for the majority's conclusion that the "primary" meaning should be attributed to "translation". The ordinary meaning and the sense in which computer programmers use the word[32] is to "express the sense of (word, sentence, speech, book, poem, etc)... in or to another form of representation or another medium or form.[33] Conversion of one code to another clearly falls within this definition: "It is submitted that such a conversion, however carried out in practice, is a 'translation' of the work within the meaning of section 2(6)(*a*)(*iii*) of the Act [of 1956]...."[34] Nor is the argument based on "context" persuasive. Not only the word "language" is defined in a broad sense (language does not necessarily mean writing or speech), thus "method of expression (finger language, talk by conventional signs with fingers; system of symbols and rules for writing computer programs.[35] Furthermore, since para (*c*)(*ii*), which deals with pictorial version, is not covered by the so-called "primary" meaning, why should para (*c*)(*i*) be ?

Regarding the majority's other objcction, viz ROMs did not represent a source code in a different medium but they were actually the means of putting into action the instructions, two submissions may be made: One, what media is finally chosen for embodiment of the program is irrelevant to the programmer. The copyrightability entitlement does not differ merely on the basis of medium in which the program is found. And, two, when a person converts a text into morse code (a series of dots and dashes), could it be argued that the resultant notations were really instructions to the telegraph operator on how to send the message ?

[29] Brennan, J, thought that the secondary meaning might make it arguable that it is a translation. But an adaptation, according to him, had itself to be a "work".

[30] Equivalent to section 2(1), Copyright Act, 1962 (NZ).

[31] Note that the New Zealand provision uses the expression "includes".

[32] This meaning is supported by the *1979 Report of the United States National Commission on New Technological Uses on Copyright Works (CONTU)*.

[33] *Concise Oxford Dictionary* p 1232.

[34] Laddie, Prescott and Vitoria *The Modern Law of Copyright*, (Butterworths 1980) para 2.143. See also *Sega Enterprises Ltd* v *Richards* (1983) FSR 73.

[35] *Concise Oxford Dicitionary*, p. 606.

Must an adaptation of a literary work itself be a literary work? Four out of the five High Court judges (Deane, J, did not consider it necessary to decide this question) took the view that an adaptation had itself to be a "work" in the light of section 31(1)(*a*)(*vii*).[36] It is submitted that this interpretation would lead to double requirement. Further, the textual wording of this section does not support the view the *Apple* Court took.[37] In any case, as stated earlier, denial of protection to object code would make copyright protection to source code almost redundant.

The High Court's final finding was that the object program was not a reproduction in a material form of the same program. It is submitted that the Court failed to address this question properly. The Chief Justice discussed it with reference to Wombat ROMs, i.e. whether they contained reproductions of Apple's source ROMs. He concluded that there was no reproduction. Mason and Wilson, JJ, also looked at this question from the same perspective but reached the opposite conclusion. Only Brennan and Deane, JJ, addressed the question in the correct manner but came to the conclusion that there was no reproduction in material form of the written expression. Regretfully, the judges erroneously used the expressions "reproduction" and "adaptation" interchangeably, which, it is submitted, is a misconstruction of the statutory scheme. The lower courts did not decide this issue.

Be that as it may, the majority judges concluded that a reproduction must have a visible similarity to the work reproduced. This conclusion was most surprising. The judges relied heavily on *Cuisenaire* v. *Reed*[38] and a few other authorities in support of the conclusion that ROM chips were not reproductions in a material form of source code. *Cuisenaire* involved a publication by the plaintiff of a book in which a new method of teaching arithmetic was described. The method used a number of wooden rods of varying lengths and colours, all having a uniform thickness. The book set out a table describing the number of rods required, their respective lengths and colours. The book did not contain any diagrams or illustrations of the rods. The plaintiff claimed copyright in the rods. Protection was denied because the plaintiff was seeking to protect the idea and not its expression. Another case which was cited in support of majority's conclusion was *Hollinrake* v. *Truswell.*[39]

It is submitted that these cases do not assist. A computer program when written is a literary work. Its embodiment in a chip retains the form of expression of the original work; retrieval is possible. But no part of the text of the book could be retrieved or "read" from the rods. The assembly code was written for the purpose of being encoded on chips — to serve as the operating instructions for the computer. Computer programs are designed primarily to be "communications" to computers. No other form of written text operates quite this way. This uniqueness was the essence of the dispute as to whether copyright existed.

[36] Equivalent to section 7(3)(*g*), Copyright Act, 1962 (NZ).

[37] Note that the New Zealand provision uses a different wording. Compare section 7(3)(*g*) of the New Zealand provision with its Australian counterpart section 31(1)(*a*)(*vii*).

[38] (1963) VR 719.

[39] (1894) Ch 420 (CA). See also above note 15.

There seems to be running through the majority judgments the notion that a reproduction (also true in the case of an adaptation), like the literary work itself, must be capable of being seen or heard. Support was derived from the definition of "writing". In the author's view, this objection based on the specific wording of section 19(1) "writing" is conclusively answered by section 31(1)*(a)(i)* "to reproduce the work in a material form". These words clearly cover the program as embodied in the ROM chip.[40] To find otherwise would require reading words into section 31(1)*(a)(i)* which are not there.

It may be argued that since the New Zealand's definition of "writing" contains the expression "any similar process", the principle of *ejusdem generis* would apply. But it should be remembered that that principle is applicable only to cut down the general scope of a statutory provision when it is clear that the legislature did not intend a broad one. It is submitted that the New Zealand legislature's intention in the case of section 2(1) "writing" and section 3(8) and 7(3)*(a)* "some other material form" and "any material form" (respectively), is very clear. The enumerations are used as examples. The general wording is said to "include" the specifics. Therefore, the principle of *ejusdem generis* can have no role to play.

There are several material differences between the New Zealand and Australian copyright legislations. The main ones are:

(i) The definition of "writing" in the New Zealand statute is not only inclusive but is also very broad:

"Writing" includes any form of notation, whether by hand or printing, typewriting, or any similar process.[41]

On the other hand, the Australian Act contains this definition:

"[W]"riting means a mode of representing or reproducing words, figures or symbols *in a visible* form, and 'written' has a corresponding meaning[42]

(ii) The definition of "adaptation" under the New Zealand statute uses the expression "includes" (implying that it is not an exhaustive definition), whereas the Australian statute contains the word "means'.[43]

(iii) Section 7(3)*(a)* of the New Zealand Copyright Act prohibits reproduction of the work "in *any* material form"[44] whereas the Australian cognate provision reads "to reproduce the work in a material form".[45] This difference in the language influenced the Canadian Federal Court recently to hold that object code had protection under the copyright statute.[46]

[40] See section 7(3) (*a*), Copyright Act, 1962 (NZ). Note that this provision uses the word "any" instead of "a".

[41] Section 2(1), Copyright Act, 1962 (NZ) (emphasis supplied).

[42] Section 10(1), Copyright Act, 1968 (Cth) (emphasis supplied).

[43] See sections 2(1) and 10(1), respectively.

[44] Emphasis supplied.

[45] Section 31(1)*(a)(i)* Copyright Act, 1968 (Cth) (emphasis supplied).

[46] *Apple Computer Inc* v *Macintosh Computers Inc*, (1986) 28 DLR 178. Note that the Canadian provision is the same as the New Zealand provision.

When the present Copyright Act was originally enacted in 1962, no thought could have been given to software protection and whether software would be covered by the provisions of the Act. However, this may not be a relevant consideration since the only question is whether the terms of the Act as drafted can fairly be said to cover such programs encoded in the ROM chip. The legislative history of "material form"[47] may be of some assistance. The 1842 UK Copyright Act, which applied to Australia and New Zealand, provided, "'Copyright' shall be construed to mean the sole and exclusive liberty of printing or otherwise multiplying copies of any . . . 'book'". "Book" was defined as including: "every volume, part or division of a volume, pamphlet, sheet of letter press, sheet of music, map, chart of plan . . " In *Boosey* v. *Whight*,[48] it was held that perforated sheets did not breach copyright in the sheet music from which the copies were made. The 1911 Copyright Act amended this provision thus: "Copyright means the right to produce or reproduce the work or any substantial part thereof in *any material form whatsoever* ...[49] and perforated rolls along with several other things were expressly mentioned. This provision has been carried forward into the Australian and New Zealand legislation and appears in similar form in section 31(1)*(a)(i)* and section 7(3)*(a)*, respectively.

It is apparent that the words of section 7(3)*(a)* were purposely drafted broadly enough to encompass new technologies which had not been thought of when the Act was drafted. It is submitted that the 1911 amendments did away with any requirement that in order to be covered by copyright, the copy or reproduction of the work had to be in a human readable form. Equally, the 1911 amendments did away with any rule which would deny copyright protection to a work merely because the copy or reproduction could be characterised as being part of a machine e.g. section 19(1) of the 1911 Act which provided, "Copyright shall subsist in record, perforated rolls, and other contrivances by means of which sounds may be mechanically reproduced . . . "

For these reasons, it is submitted that the *Apple* decision would not be followed in New Zealand and is not a barrier to granting copyright protection to object programs in New Zealand.

The view that copyright statute should be read to require that a work be intended to communicate an expression to humans is unsound for three reasons. First, such an interpretation is unjustified in light of prior copyright practice. The law reports are full of examples of works which have been copyrighted that were not intended to communicate any original expression, e.g. telephone directories, telegraphic codes. Second, both the "material form" requirement and the definition of "copy" require only that a work be capable of being "reproduced". In fact an "eye-readable" version of the object code program is easily reproduced from the machine usable form. Moreover, machine language is well understood by trained programmers. Thirdly, the view that copyright protects only works intended to

[47] See section 31(1)*(a)(i)*, Copyright Act, 1968 (Cth); section 7(3)*(a)*, Copyright Act, 1962 (NZ).

[48] (1899) 1 Ch. 836, affirmed (1900) Ch. 122 (CA).

[49] Emphasis supplied.

communicate an original expression to humans wrongly forces a program to qualify under two statutory categories of copyrightable creations in order to gain protection. Futhermore, as pointed out earlier, this view appears to confuse the "medium" and the "message".[50]

Copyright law provides a number of mechanisms for limiting the exclusive rights of a copyright owner in order to ensure public access to information. One of these is fair dealing. As noted earlier,[51] fair dealing is an equitable doctrine that limits the scope of copyright when appropriate for the ultimate benefit of society. Fair dealing is primarily a question of fact. There are several factors to be considered in determining whether a particular dealing is fair. No single factor is determinative; all must be weighed.[52]

Is the decompilation or disassembly of object form software a fair dealing? When the purpose of the decompilation and disassembly is to gain access to the structure and logic of the software, the answer is in the affirmative. Although the decompilation or disassembly process may result in the formation of a copy of the *whole* of the object program, it will be a fair dealing if the *ultimate* purpose of the process was not to misappropriate the protected expression but rather to gain access to the underlying ideas "hidden" in the program in order to modify or adapt the program to a new use, or to facilitate the creation of compatible software. The fact that the purpose of developing compatible software has a commercial motive will not by itself preclude a finding of fair dealing if the new work offers some benefit to the public.

The exception of fair dealing has been held to be particularly applicable to compilation works, e.g. data bases. In a United States' case *Dow Jones & Co.* v. *Board of Trade of the City of Chicago* the Court observed, "Because copyright protection for compilations of factual material cannot be reconciled with the general principles of the copyright laws. . . such works should be most conducive to fair [dealing]. Authors of compilations, therefore, must be held to grant broader licenses for subsequent use than persons whose work is truly creative.[53]

The nature of object form software is that any ideas contained in it are not accessible without decompilation or disassembly. The program is not like a book whose ideas can be extracted by inspection. The only access to its ideas is through processes that necessarily involve the creation of an unauthorised copy.

Generally stating, the fair dealing exceptions under sections 19 to 21[54] do not permit copying of the *entire* work. Indeed, section 21 expressly provides that "no copy shall extend to more than a *reasonable proportion* of the work[55] and this restriction

[50] See above note 6.

[51] See above paras 5-6

[52] See Puri "Fair Dealing with Copyright Material in Australia and New Zealand" (1983) 13 VUWLR 277.

[53] 546 F Supp 113, 120 (NY 1982). See also *Sony Corporation of America v. Universal City Studios Inc* 104 S Ct 774 (1984).

[54] Copyright Act, 1962 (NZ).

[55] Emphasis supplied.

may be a major problem for anyone copying an object form of software. Sections 19 and 20 may also be similarly construed, otherwise the reproduction of the entire work may not be a "fair" dealing even for purposes of research or private study. It is submitted that the existing legislative provisions dealing with fair dealing are inappropriate. However it is hoped that the courts in New Zealand would interpret these provisions differently vis-a-vis software use. The copying of a substantial portion of the object program, and in some cases the entire work, may have to be permitted to make full sense of fair dealing doctrine. The process of decompilation or disassembly results in an equivalent source code version of the object form program. Although this version is a copy of the *entire* computer program, it is unavoidable that such a copy is made in the course of "reverse engineering" object form software. It is impossible to determine the meaning of each binary sequence in the object code version of the program without reference to the context in which it is executed by the computer. To understand the context, the entire program must be analysed. Nothing less than complete copying will suffice. Decompilation and disassembly should not therefore be condemned merely because the entire work is copied.

The alleged infringing act that occurs during decompilation or disassembly is the creation of an unauthorised copy, the equivalent source code version of the object form software. In considering decompilation or disassembly, the economic impact of the allegedly infringing article (equivalent source code version) on the market for the object form program must be examined. The object form software is directly usable on a computer while the equivalent source code version is not. Thus, the allegedly infringing article performs a different function from the copyrighted work. The market for the object form program consists of computer users who need an executable program. The "market" for the equivalent source code version consists of competitors and users seeking to understand the structure and logic of the program. Those wanting an executable (object) program would neither need nor want the alleged infringer's equivalent source code version. Since the object program and the copy perform different functions, the latter has no immediate adverse economic impact upon the market for the former. It should however be pointed out that any recompilation or reassembly of the equivalent source code version and sale of the resulting object form software in competition with the original object form program would not be a fair dealing. Nor, for that matter, would marketing of the equivalent source code be protected under this exception. The only purpose for which a copy can be made is to gain access to the structure and logic of the program to permit adaptation or correction of the program for research or private study or to facilitate the development of compatible programs. Absent contractual restrictions and/or trade secrets, the reproduction of an object form of software for the purpose of gaining access to the uncopyrighted ideas in the program is fair dealing. The nature of object form software masks these ideas and requires that the *entire* work be copied as an essential step in accessing them.

The preceding paragraphs have dealt with copyright in software. However, in many instances, it will be at least as important to protect copyright in the program's documentation. This may include instruction manuals and the programmer's notes on how a program was constructed. Usually a detailed set of instructions accompany the software package. Instruction manuals are not outside the scope of copyright protection. Copyright might not protect every minor direction for use of a machine or device, but an instruction manual contains a set of complex and precise instructions for the user. To reproduce in a material form (i.e. to make a photocopy) or to adapt the actual written instructions plainly involves infringement of the copyright in the "literary" work. It is most unlikely that a reproduction of an instruction manual would satisfy the requirements of fair dealing.

As seen above, there are several legal approaches to protecting software, and they may in many situations be combined for maximum results. Usually when an object form of software is sold, a combination strategy of contract law and copyright law is used. A contract is simply a legally enforceable agreement that states the parties' rights and responsibilities. Usually a contract prohibits the customer from using the software in an unauthorised manner or lending it to third parties. However, the major weakness of a contract as a protection method is that it is binding only on those who are parties to it. In contrast, it is one of the features of copyright that it provides protection in the absence of any legal relationship, whether express or implied. Personal computers are nowadays owned and operated by fairly substantial business houses, academic institutions, official organisations and the like. It is a common practice for suppliers of software to seek legal protection not only under the copyright law but also by using the law of contract. The following is an (edited) example of one such contract:

MICROSOFT LICENCE AGREEMENT

> Carefully read all the terms and conditions of this agreement prior to breaking the diskette seal. Breaking the diskette seal indicates your acceptance of these terms and conditions.
>
> 1. LICENCE: You have the non-exclusive right to use the enclosed program. This program can only be used on a single computer. You may physically transfer the program from one computer to another provided that the program is used on only one computer at a time. You may not electronically transfer the program from one computer to another over a network. You may not distribute copies of the program or documentation to others. You may not modify or translate the program or related documentation without the prior written consent of Microsoft.
>
>   YOU MAY NOT USE, COPY, MODIFY, OR TRANSFER THE PROGRAM OR DOCUMENTATION, OR ANY COPY EXCEPT AS EXPRESSLY PROVIDED IN THIS AGREEMENT.
>
> 2. BACK-UP AND TRANSFER: You may make one (1) copy of the program solely for back-up purposes. You must reproduce and include the copyright notice on the back-up copy. You may transfer and license the product to another party if the other party agrees to the terms and

conditions of this Agreement and completes and returns a Registration Card to Microsoft. If you transfer the program you must at the same time transfer the documentation and back-up copy or transfer the documentation and destroy the back-up copy.

3. COPYRIGHT: The program and its related documentation are copyrighted. You may not copy the program or its documentation except as for back-up purposes and to load the program into the computer as part of executing the program. All other copies of the program and its documentation are in violation of this Agreement.

4. TERM: This licence is effective until terminated. You may terminate it by destroying the program and documentation and all copies thereof. This licence will also terminate if you fail to comply with any term or condition of this Agreement. You agree on such termination to destroy all copies of the program and documentation.

Appendix

GLOSSARY OF COMPUTER JARGON

1. "APPLICATION" PROGRAM AND "OPERATING" PROGRAM:

Although, in one sense, all programs instruct the computer, they are frequently categorised by the function they perform. Operating programs are programs that, as a system, instruct the computer to manage its own resources, execute application programs and manage data. Application programs are designed for a specific task, such as the playing of a video game, preparation of a tax return, or, in business, to calculate the payroll or the accounts receivable. Operating programs are usually supplied by hardware manufacturers, whereas application programs can be bought "off-the-rack", or can be "tailor-made" to suit the user's needs. *N.B.* In *Apple* the programs answered the description of operating programs.

2. ASSEMBLY LANGUAGE OR SOURCE CODE:

A second level of language (also referred to as intermediate level) consists of mnemonics (each comprising three letters of the alphabet, e.g., LDY, which stands for "Load index Y with memory") which correspond more explicitly to the operations the computer must perform. It is a language which is directed to a human reader and not to a machine. It can be readily understood by persons skilled in its use.

Assembly language (or code) is converted into machine language by using a computer program called "assembler". This conversion from source to object code consists of a series of electrical impulses which can be stored on a magnetic tape or disk or permanently installed on a ROM. A ROM is embodied in silicon chip.

3. BINARY NOTATION: Binary notation and hexadecimal notation are conventional ways of representing in writing an object program. Binary notation is based on a number system having a base two. A microprocessor cannot store or operate in accordance with a program expressed in source code. As an electrical device it can only register the presence or absence of electrical impulses and perform all its functions according to a process which can be described in terms of binary arithmetic. An "o" (zero) indicates the presence of an electric current, or impulse, at a particular place, and a "1" (one) indicates the lack of a current, or an electric impulse. The digits 0 and 1 are called binary digits or "bits", and represent the only unit of information that can be recognised by a computer, namely, the presence or absence of electric current. In fact, it is the binary form of machine language which the computer "understands".

4. HEXADECIMAL SYSTEM: Hexadecimal notation is based on a number system having a base sixteen. This system is merely a shorthand way of writing the binary code. It is used because it uses less characters and is therefore less cumbersome than binary. The assembly language (source code) can be converted into machine language (object code) in its binary or hexadecimal code by means of a computer program called "assembler". The conversion can be done manually but the process is very tedious. The program can be originally written in hexadecimal code, rather than in assembly language, except for the cumbersomeness and tediousness of doing so.

5. BIT AND BYTE : Each digit of binary code (be it a "0" or "1") is called a "bit". A string of eight digits is called a "byte". A byte is a basic unit of measure of a computer's memory.

6. COMPUTER LANGUAGE: A code for writing a program.

7. COMPUTER PROGRAM: A computer program is a concise series of instructions that directs the computer to do the tasks required of it step by step and to produce the desired result.

8. HARDWARE: The machines in a computer system are called its "hardware". The computer and all the physical devices attached to it (e.g. microprocessor, keyboard, disk drive unit, monitor, printer) are compendiously described as hardware.

9. HIGH LEVEL LANGUAGE: A high level language has symbols and rules that correspond closely enough to ordinary mathematics and Engiish (or other common language) that it may be read and understood with relative ease. Examples are languages such as BASIC, COBOL, PASCAL and FORTRAN. Whichever type of high level language is used by the programer, a "translator" is needed to translate the symbolic statements in that language into machine language. In performing the function the translator is called a "compiler" *N.B.* BASIC cannot be converted into machine language or object code.

10. INPUT DEVICES: The keyboard and disk drive unit are referred to as "input" devices.

11. MACHINE LANGUAGE OR OBJECT CODE: A low level language which has two versions: a system of binary notation and a system of hexadecimal notation. Either notation is a description of the object code program and is intelligible to persons trained in the use of that form of notation. Object code is machine language, i.e. language which can be understood by the microprocessor in the computer. Object code can be written.

12. MICROPROCESSOR OR CENTRAL PROCESSING UNIT (CPU): The CPU is the operating centre of every computer. All the manipulation of information entering a computer system takes place in its CPU. The CPU controls the operation of the entire system by fetching instructions and data from the memory and processing the data in accordance with the instructions. For example, when the computer system is used for the retrieval of legal information — a form of data processing — the CPU is where all the sorting and rearranging takes place. The CPU is a complex and extensive set of electrical circuitry. As such, it knows only the presence or absence of electrical impulses. The semiconductors in the microprocessor can be in one of two possible states, similar to that of an electric

light globe, namely either "on" or "of". This two-state system (known as a binary system) is represented by the digits 0 or 1.

13. ROM (READ-ONLY MEMORY): Some programs can be stored permanently in a memory device, the essence of which is a silicon chip. These are attached inside the computer at the time of manufacture, although they are readily removable and can be replaced. Their "memory" is not erased when the computer is switched off. They cannot be altered, and are therefore sometimes called "firmware". ROMs typically store operating programs for the microcomputer, as these programs are always required whenever the computer is being used. ROMs initialise the computer and lets a person to use it. ROMs hold important programs or data which is available to the computer when the power is first turned on.

14. EPROM (ERASABLE PROGRAMMABLE READ-ONLY MEMORY): A program stored in an EPROM is embodied therein as a pattern of electronic charges stored on insulated "floating gates" (referred to as "MOS" technology). The absence of an electronic impulse is indicated by an uncharged gate. If an EPROM is illuminated through a "window" in its outer surface by ultraviolet light the pattern will be erased, otherwise it remains. An EPROM, unlike a ROM, is a variable memory device.

15. RAM (RANDOM ACCESS MEMORY): Storage on memory chips, in which data and programs can be written in and read from any main storage location readily and speedily, are called RAMs. All RAMs share the feature of being volatile, that is, they require a continuous supply of electric current to maintain the stored data. If the current is stopped, or even interrupted, all the data represented in the memory is lost.

16. SOFTWARE: Programs which do not form part of the microprocessor itself (e.g. magnetic disks, ROM, RAM, EPROM).

17. SILICON CHIPS: A silicon chip is made up of thousands of integrated, complete electronic circuits etched onto a small chip of silicon. The chip contains numerous fusible connectors and insulated capacitors. Using a device referred to as a "ROM Burner", the impulses are passed through the

chips, burning out the connectors and leaving the capacitors electrically charged. The pattern of connectors and capacitors thus affected correspond to the sequence of electrical impulses so that the same sequence of electrical impulses can be regenerated. The impulses are regeneratcd in this way each time the program is used or "run" in the microprocessor. The patterns can be seen, but only with an electronic microscope.

* * * *