The pervasiveness of computers in our daily lives today has brought to the fore many problems relating to the legal protection of computer programs. The issue as to which form of intellectual property protection is appropriate for protecting computer programs has been a hotly debated topic for the international community for a long time. The two contrasting forms of intellectual property rights, which could be used to protect computer programs, are patent law and copyright law. Copyright law aiming to protect all but the most trivial computer programs which satisfy the requirement of originality and patent law aiming to protect software inventions provided the software satisfies the three requirements of novelty, inventiveness and usefulness thus having the potential to produce a ‘further technical effect’.

Under Indian law, however, the position is very clear, with ‘computer programs’ being considered ‘literary works’ solely within the purview of copyright protection as Section 3(k) of the Patents (Second Amendment) Act, 2002 expressly provides that computer programs per se are not patentable. This Article discusses the various tests formulated by the courts in trying to deal with copyright infringement and
thereafter discusses the applicability of such tests when dealing with computer programs in order to facilitate the understanding of the actual concept behind copyright protection of computer programs.

Copyright Infringement of Computer Programs

Since computer programs are protected under the Indian law as ‘literary works’, the essential elements required to be proved by a plaintiff alleging copyright infringement of such existing works are: (i) ownership of a valid copyright; and (ii) copying of constituent elements of the work that are original. Ownership of valid copyright is conceded in almost all cases, however, with regard to the issue of copying, as *Fiest Publications v. Rural Telephone Services Co* illustrates, copying as a factual matter is insufficient, if improper appropriation is lacking. Therefore, copying is ordinarily established indirectly by the plaintiff’s proof of access and substantial similarity.

Access

Although the Indian courts have never defined access, the US courts have defined access as the actual viewing and knowledge of the plaintiff’s work by the person who composed the defendant’s work. It is, however, felt that as it is virtually impossible to offer direct proof of copying, so it is often impossible for a plaintiff to prove the fact of actual viewing and therefore perhaps the more prevailing definition of access is the opportunity to copy. Hence, even if the plaintiff is unable to prove actual viewing by the defendant, but is able to establish the reasonable opportunity to view, this is to be regarded as a showing of access for the purposes of shifting the burden to the defendant to establish independent creation.

Substantial Similarity

The determination and extent of similarity, which becomes substantial, presents one of the most difficult question in copyright law. In *R.G. Anand v. Delux Films & Ors.*, the Supreme Court observed that, “It is not necessary that the alleged infringement should be an exact or verbatim copy of the original in a large measure is sufficient to indicate that it is a copy.”

Therefore, in trying to determine copyright infringement in computer programs, the problem is that between the one extreme of no similarity and the other of complete and literal similarity lies the line marking off the boundaries of substantial similarity, which wherever drawn will seem arbitrary.

It would seem to follow analytically that, more similarity is required when less protectable matter is at issue. Thus, if substantial similarity is the normal measure required to demonstrate infringement, ‘super substantial similarity’ must pertain when dealing with thin works.

Further, computer programs tend to be incomprehensible to a lay judge, and hence evaluating the similarity between two computer programs is often exceedingly difficult. Such difficulties are particularly applicable when the
allegations of infringement go beyond the mere literal copying of the program code\(^{15}\). Thus, in order to determine substantial similarity, the courts have formulated certain tests, which is necessarily to be discussed for the better understanding of this concept of substantial similarity.

Audience Test

As has been previously discussed, in the Indian scenario, computer programs are protected under the copyright law as literary works. The most predominant test in determining substantial similarity between literary works is the ‘audience test’. The Supreme Court in *R.G. Anand v. M/s. Delux Films*\(^{16}\) after considering a number of English, Indian and American authorities observed that, the surest and safest test to determine whether or not there has been a violation of copyright is to see whether a spectator or viewer after having read or seen both the works is clearly of the opinion that the subsequent work appears to be a copy of the original\(^{17}\).

It is, however, felt that, a special challenge is posed in adjusting the traditional categories of substantial similarity to the burgeoning areas of technology protected by copyright\(^{18}\). As applied to computer programs, the ‘audience test’ is not only weak but also facially inapplicable, as it is meaningless to attempt to isolate the ‘spontaneous and immediate reaction’ of the lay observer to two sets of object code\(^{19}\).

The Indian courts have not till date considered the legal implications arising from copyright infringement of computer programs, but the US courts have had the opportunity to deal in this matter a number of times. Some of these US courts have acknowledged the charade inherent in applying the ‘audience test’ to such works of high technology and have forthrightly abandoned the pretense of applying the two-step analysis envisioned by the ‘audience test’\(^{20}\).

Thus, it can be rightly concluded that, in the Indian scenario the applicability of the test laid down in order to determine substantial similarity by the Supreme Court in *R.G Anand v. M/s Delux Films*\(^{21}\) is not applicable to computer programs because the material under scrutiny is of a highly technical nature which an ordinary spectator cannot judge\(^{22}\).

Iterative Test

In *E.F. Johnson v. Uniden Corporation*\(^{23}\), the District Court of Minnesota formulated the ‘iterative test’ to determine copyright infringement of computer programs. Under this test a *prima facie* case of infringement is established by showing:

(i) That the defendant ‘used’ the copyrighted work in preparing the alleged copy, which may be established by proof of access and similarity sufficient to reasonably infer use of the copyright work; and

(ii) That the defendant’s work is one produced by iterative or exact duplication of substantial portions of the copyrighted work.

It is pertinent to mention here that, this test appears to focus on ‘use’ of the plaintiff’s program by the defendant even
in the presence of proof of literal copying. Furthermore, if protection is to be afforded to non-literal elements of a program such as its sequence of operations or structure and organization, the ‘iterative test’ seems completely unworkable\textsuperscript{24}. Therefore, this test can be used only when dealing with computer programs, which involve literal copying of the code or direct translation of the program.

Look and Feel Test
The advancement of technology has made copying of computer programs very simple. If the ‘original’ is a source code program, the ‘copy’ may have been written in a different computer programming language, so a line for line comparison of the programs will not be conclusive.

Making use of an existing program by inspecting to determine the flow and structure of the program will facilitate the writing of another program, intended to perform the same function in a different computer programming language is a daily phenomenon and the Circuit Court of the United States in \textit{Whelan Associates v. Jaslow Dental Laboratory}\textsuperscript{25}, when considering the question as to whether copying of the flow and structure of the program in order to create a new program would be unfair, formulated the ‘look and feel test’.

The Court observed that, in relation to a computer program designed to carry out a mundane task, anything, which was essential to the task, was ‘idea’ whilst that, which was not essential and could have been written in different ways was ‘expression’. If these latter parts were copied then the copyright would be infringed because the expression had been copied. Consequently, not just the actual program code, but the structure of a computer program can be protected because of its ‘look and fee’ and ‘similar structure’.

It is however necessary to note that, the aforementioned case has come under scrutiny in subsequent cases in the United States and the proposition laid in \textit{Whelan}\textsuperscript{26} has been labeled as flawed because the court assumes that only ‘one idea’ in copyright law terms, underlie any computer program and that once a separable idea can be identified, everything must be expression. Further, two programs used to perform the same simple function could have vastly different structures and therefore the hypothesis that a program’s application will always dictate the structure of the program is a false one\textsuperscript{27}.

Abstraction -Filtration-Comparison Test
It is an established principle of copyright law that, it is only the author’s original expression, which is protected, whereas the idea or elements taken from pre-existing works is not protectable.

In \textit{Computer Associates International Inc. v. Altai Inc.}\textsuperscript{28}, the Second Circuit held that, infringement of copyright in computer programs is shown by a substantial similarity of protectable expression, not just an overall similarity between the works. Thus, before evaluating substantial similarity, it is necessary to eliminate from consideration those elements of a program that are not
protected by copyright\textsuperscript{29}. In order to do this the Court formulated the three-stage abstraction-filtration-comparison test (Fig. 1).

The first stage is ‘abstraction’, by which the non-literal elements of the program are determined by a process like reverse engineering, tracing back the programmer’s steps in the process of writing the program. This operation results in the identification of the non-literal elements of varying degrees of detail.

The second stage is one of filtration, filtering out the elements that are not protected by copyright. Such filtration can be brought about by the following methods:

\begin{itemize}
  \item \textbf{Exclusion of Elements Constituting Abstract Ideas}
  
  It is an established principle that only expressions of ideas and not the ideas themselves are entitled to copyright protection\textsuperscript{31}. In practice, a programmer usually will start with a general description of the function that the program is to perform. Then, a specific outline of the approach to this problem is developed usually by studying the needs of the end user. Next, the programmer begins to develop the outlines of the program itself, and the data structures and algorithms to be used. Therefore, it is clear that at the start of the process, the programmer has only a general notion of what the program is supposed to do and of possibly, which algorithms would be desirable-material that falls within the realm of unprotectable ideas and which are to be filtered out. It is only when the program is completed, and the programmer has produced code giving form to his ideas that, such code will constitute protectable expression\textsuperscript{32}.

  \item \textbf{Exclusion of Elements Dictated by Logic and Efficiency}
  
  Under the Merger Doctrine, when an idea can be expressed in only one fashion, that expression is not protected by copyright, as the result would be to provide a monopoly over the idea itself\textsuperscript{33}. In the realm of computer programs,
\end{itemize}

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{fig1.png}
\caption{Diagrammatic representation of the abstraction-filtration and comparison test\textsuperscript{30}}
\end{figure}
merger issues may arise in somewhat unusual ways. Although, theoretically, many ways may exist to implement a particular idea, efficiency concerns can make one or two choices so compelling so as to virtually eliminate any other form of expression. In such a case, applying the Merger Doctrine may dictate the conclusion that a particular way to implement an idea is unprotectable, and thus, should not be considered evidence of substantial similarity of protected expression.

Exclusion of Elements Dictated by External Considerations

Hardware Standards—The computer, on which a certain program is intended to run, may dictate numerous elements of a computer program. To make a program compatible with a particular machine, a programmer must tailor the source code of the program to meet the machine’s design standard. As a result, two programs intended to run on the same computer may contain a number of similarities that are not the product of the programmer’s independent creativity, but are simply necessary for the program to function properly on the desired machine and hence these elements which are dictated by such hardware standards are to be filtered out.

Software Standards—Any end-user program must communicate and cooperate with the computer’s operating system. Compatibility with the operating system also dictates numerous aspects of program design, such as the way in which the program accesses data files on disks, or the way in which the user initially calls it up. Therefore, the software environment in which the program is developed or operates may also govern elements of the program design, and these elements are rendered unprotectable.

Target Industry Practices—The most significant external factors influencing program design are the business practices and technical requirements of the end user. Programs, which are intended to suit the same end user, would necessarily involve a significant degree of similarity, but such similarity is not actionable and elements dictated by target industry practices are to be filtered out.

Computer Industry Programming Techniques—When developing any computer program, most programmers rely on a number of traditional solutions to recurring problems in their programming. When certain common computer programming techniques are used by the programmers in developing their programs, these techniques are not subject to copyright protection and are in the realm of unprotected expression.

Exclusion of Program Elements Taken from Public Domain

An enormous amount of public domain software exists in the computer industry, perhaps to a much greater extent than is true of other fields. In addition, computer-programming texts may contain examples of actual code that the programmers are encouraged to copy. Material in the public domain is not protected by copyright even when incorporated into a copyrighted work. Therefore, if certain elements are taken from the public domain then these elements in a computer
program necessarily need to be filtered out as such material is unprotectable.

The analytic process set forth above successively filters out unprotected elements and what remains is the remaining core of protectable material known as the ‘golden nugget’. The third stage is one of comparison. The elements present in this ‘golden nugget’ are at this stage compared with the defendant’s program and in case of similarity between the two a holding of substantial similarity against the defendant is justified.

Conclusion

After having discussed the various tests laid down by the foreign courts in determining copyright infringement of computer programs, it is felt that the Altai test, is by far the most logical and analytical way of determining such issue of copyright infringement. Of course, not all the tests mentioned above would apply to every program involved in any given case, and only those doctrines supported by the facts and evidence of a particular case should be considered.

In the absence of any Indian precedent on this aspect, it is felt that in case at a future date the courts are faced with such questions of non-literal copying of computer programs, the Indian courts should disregard the ‘audience test’ and welcome the ‘abstraction-filtration and comparison’ test. The courts should also consider the dissimilarities present in the two computer programs and the presence of common errors in both programs before coming to the conclusion of infringement.

It is felt that this ‘abstraction-filtration-comparison test’ is merely a thorough and rigorous application of traditional copyright rules to cases involving computer programs and that this test would help the Indian courts to focus on inquiries and bring some clarity to these cases of computer programs, wherein the subject matter is already very confusing.

References and Notes

3. Section 13 read with Section 2(o) and 2(fic) of the Copyright Act, 1957.
4. Infra note 5, § 13.01[A], at page 13-06.
5. 499 US 340 (1991); Although this case dealt with copyright infringement of databases, the US Supreme Court when deliberating on the issue as to what nature of ‘copying’ would be actionable had made this observation; followed by the Delhi High Court in, Eastern Book Co v. Naveen J Desai, I A No. 3149/2000 (decided on 17.1.2001).
7. Berkic v. Crichton, 761 F. 2d 1289 (9th Cir. 1985) at page 1291.
15  Apple Computer v. Franklin Computer, 714 F.2d 1240 (3rd Cir. 1983).
16  AIR 1978 SC 1513.
18  Apple Computer v. Franklin Computer, 714 F.2d 1240 (3rd Cir. 1983).
19  AIR 1978 SC 1513.
20  National Broadcasting Co. v. Sonneborn, 630 F. Supp 524 (D. Conn) at page 528, this case demonstrates how technology has made some traditional categories of copyright law anachronistic.
22  Whelan Associates Inc. v. Jaslow Dental Laboratory, Inc., 797 F.2d 1222 (3rd Cir. 1986) at page 1232; these courts do not hesitate to admit expert testimony on the key issue of similarity between computer programs. See Plains Cotton Co-Operative Association v. Goodpasture Computer Service, Inc. 807 F.2d 1256 (5th Cir. 1987) at page 1259; Gates Rubber Co v. Bando Chemical Industry Ltd, 9 F.3d 823 (10th Cir. 1993) at page 835.
23  Supra note 11
24  Although 2 computer programs may be similar to look at, but even then they may be written in absolutely different programming languages utilizing different programming techniques, hence a common observer’s audience test is ineffective and is potentially misleading where the subject matter is complex such as in the case of a computer program, David Bainbridge, Software Copyright Law (FT Pitman Publishing, London) 4th edn, 1992, at page 88.
26  Reliance is placed on Nimmer on Copyright (Matthew & Bender & Co Inc) 1999, Vol 4, § 13.03[A][1][d], 13-41.
28  Ibid
30  Ibid
31  The familiar proposition that ideas are as ‘free as air’ has been recognized by computer copyright cases no less than in other types of copyright litigation, Q-Co. Industry v. Hoffman, 625 F. Supp 608 (S.D.N.Y. 1985) at page 615.
32  Supra note 5, § 13.03[F][1], at page 13-121.
34  An example of the application of the Merger Doctrine is that of two choices in a menu bar that can logically be placed only at the top or at the bottom of the screen, Productivity Software International Inc v. Healthcare Technologies Inc, 37 U.S.P.Q. 2d 1036, at page 1040-41.
35  In Computer, some methods of sorting or searching are significantly more efficient than others in handling particular types of data, searching and sorting algorithms provides good example of this phenomenon. It is now recognized even though any numerous methods will work, Basse S, Computer Algorithms, 51-60 (1978); cf: Nimmer on Copyright, (Matthew & Bender & Co Inc), 1999, Vol 4, § 13.03[F][3], at page 13-120.
37  Supra note 5, § 13.03[F][3][a], at page 13-130, 131.
38  The programming language in which the program is written may also control some aspects of the program design, hence the programming language also forms a part of the software environment, Supra note 5, §13.03[F][3][b], at page 13-135.
Eg. Programs intended to trade on the Bombay Stock Exchange necessarily must be designed to comply with the rules and practices of the Exchange, and therefore certain elements of those programs are definitely dictated by target industry practices.

See v. Durang, 711 F. 2d 141 (9th Cir. 1983), at page 143.


Supra note 36

Hartman v. Hallmark Cards, 833 F. 2d 117 (8th Cir. 1987).

It is however necessary to note that, if substantial similarity is found, the defendant will not be immunized from liability by reason of only the addition in his work of different characters or additional and varied incidents, Atai Inc. v. America Philips Consumer Electrics Corporation, 672 F. 2d 607 (7th Cir. 1982).