In this chapter you will learn to:

- recognise the effects of software solutions on society
- identify the impact of inappropriately developed software on users
- identify the effect of the inappropriate use of software on society and individuals
- apply a relevant code of conduct to their own software development
- interpret licence agreements and develop personal practices that reflect current laws
- identify the relationship between copyright laws and software license agreements
- acknowledge all sources in recognition of the intellectual contribution of authors
- identify a range of techniques designed to combat software piracy
- evaluate the usefulness of networks in the development environment
- identify the impact of dominant developers of software on software development
- discuss the reasons for, and consequences of, significant legal actions pertaining to the development of software

Which will make you more able to:

- explain the relationship between emerging technologies and software development.
- identify and evaluate legal, social and ethical issues in a number of contexts.

In this chapter you will learn about:

The impact of software
- inappropriate data structures, for example the year 2000 problem
- computer malware such as viruses
- reliance on software
- social networking
- cyber safety
- huge amounts of information (which may be unsupported, unverifiable, misleading or incorrect) available through the internet

Rights and responsibilities of software developers
- acknowledging the intellectual property of others
- recognition by others of the developer's intellectual property
- producing quality software solutions
- appropriately responding to user-identified problems
- adhering to code of conduct
- neither generating nor transmitting malware
- addressing ergonomic issues in software design
- ensuring software addresses inclusivity issues
- ensuring individuals' privacy is not compromised

Software piracy and copyright
- concepts associated with piracy and copyright, including:
  - intellectual property
  - plagiarism
  - copyright laws
  - licensing issues
  - licence conditions
  - shareware
  - public domain
  - open source
  - ownership versus licensing
  - collaboratively developed software
  - reverse engineering
  - decompilation
- current and emerging technologies used to combat software piracy. Such as:
  - non-copyable data sheet
  - use of serial numbers
  - site licence installation counter on a network
  - encryption key
  - registration code
  - back-to-base authentication

Use of networks
- by the developer when developing software
  - access to resources
  - ease of communication
  - productivity
- by the user when using network based software
  - response times
  - interface design
  - privacy and security issues

The software market
- maintaining market position
- the effect of dominant developers of software
- the impact of new developers of software and new products

Legal implications
- national and international legal action resulting from software development. Identify issues raised in cases at both national and international level, such as:
  - RACV vs Unisys
  - Microsoft vs Netscape
  - NSW T Card system
  - search engines (e.g. Google vs national censorship laws)
  - Metallica vs Napster

1

SOCIAL AND ETHICAL ISSUES

The development of software solutions has a profound impact on the community and the wider world. Software developers have a responsibility to ensure their products function in a socially responsible and ethical manner. Many software products perform tasks that are vital to the operation of essential services. Many of these services rely on computer software to perform their tasks adequately.

The users of computer software also have responsibilities. To ensure the continued production of quality software products requires developers to include systems to minimise the chance of software piracy. Copyright laws are designed to protect intellectual property rights. Such laws provide developers with legal methods of pursuing piracy cases. Software developers use licence agreements in an attempt to enforce their intellectual property rights and to minimise their liability should the software cause unintended problems.

In this chapter, we examine the impact of software on society. This includes the rights and responsibilities of software developers. Software developers are authors; they produce original works and as such are entitled to have their intellectual property protected in the same manner afforded to any other author. Unfortunately, software products are on the whole easy products to copy. What's more, the copy is identical to the original. This has lead to widespread piracy of software products. We examine techniques available to developers to combat piracy and other breaches of copyright.

FORUM

> **GROUP TASK Discussion**
> Do you think it is the responsibility of software developers to consider social and ethical issues. Wouldn't it be easier if they could just concentrate on the task of writing software? Discuss.

THE IMPACT OF SOFTWARE

Software increasingly underlies more and more aspects of our lives. It is used to control the provision of essential services such as electricity, gas and water. Virtually all communication systems including traditional phone lines are largely driven by software. Every message sent across the internet, including email, web access, financial transactions and even many phone conversations utilise a variety of software systems. Most physical products use software during their design and manufacture and most electrically powered products include embedded software. Clearly the effect of software on society is significant.

When software is inappropriately developed or used the impacts have the potential to be significant. Some problems are due to poorly developed software that includes bugs, other problems are intentional such as malware and other issues are due to unforeseen circumstances such as the year 2000 problem. Software has changed the nature of communication and the availability of information. Social networking sites allow us to share virtually any aspect of our lives, but the public nature of such information has lead to widespread concerns with regard to cyber safety and privacy.

The internet allows access to an amazing and almost unlimited store of information, but filtering and verifying the correctness of such a vast store of information remains an issue.

The Year 2000 (Y2K) Problem

In the 1960s and 1970s storage was expensive so software was required to use the absolute minimum amount of storage space. To cater to this need, software developers often used only 2 digits to represent the year within dates. It was never envisaged that these products would still be in use when the Year 2000 came along. As a consequence of this short-sightedness it was anticipated that many computer-based systems would fail as the year changed from 1999 to 2000. That is, the software would potentially think that the year was 1900 and not 2000. Similar problems were expected with regard to other years, such as 1960 being interpreted as 2060 and hence causing a variety of incorrect calculations and computer crashes.

GROUP TASK Discussion
Many new companies emerged during the 1990s with the sole aim of checking software for Year 2000 compliance.
Should consumers and businesses have had to utilise the services and products of these 'Y2K' companies?
Discuss and debate each side.

GROUP TASK Debate
Many individuals and social groups prepared for the worst disaster. They felt that communication systems, power and other utility supplies would fail as a result of the 'Y2K' problem.
Discuss why these people may have had grave concerns.
Do you think that these concerns were justified?

GROUP TASK Discussion
As we now know there were no significant system failures as a result of the 'Y2K' problem. Let's for a moment consider who would have been responsible had there been a complete failure of the electricity grid system on a global scale.
Discuss in terms of social and the ethical considerations.

Computer Malware (including viruses)

Malware or malicious software includes any software which performs an unwanted and uninvited task. Examples of malware include viruses, worms, Trojan horses and spyware. New malicious software appears somewhere in the world on a daily basis.

Malware
Malicious software designed to damage or perform unwanted actions on computer systems.

Anti-virus, anti-malware and anti-spyware utilities are installed to detect and destroy malware and to advise users of potentially unsafe practices or websites.

The ability to replicate itself distinguishes viruses from other malware. Viruses are commonly attached or embedded within apparently legitimate software or files. As the software is executed the virus is also able to execute, duplicate itself and perform its malicious processing. The duplicates are then passed onto other computer systems via email and other forms of electronic communication.

Worms are a type of virus which aims to slow down a system. Many worms exploit a security flaw in the operating system and then install themselves on the hard drive (often as hidden files). The worm then begins to replicate itself over and over again. Over time the worm consumes the entire hard disk. Other worms continue to execute copies of themselves. Typical symptoms of such attacks result in RAM being completely consumed by the worm and hence other software will run slowly or there may be insufficient RAM for software to execute at all.

In Greek mythology, a Trojan horse was a large wooden horse which the Greeks offered as a sign of peace to the city of Troy. Once the Trojan horse entered the city gates the Greek soldiers jumped out and apparently overthrew the city. In computer terms a Trojan horse is malicious software that is masquerading as legitimate software. For instance, you may think you're downloading a game, however when the game is executed it begins to perform unwanted processes, such as deleting or corrupting files.

Trojan horses are often used to install malware that enables an outside user (or program) to take control of your computer. Such Trojan horses are used to create botnets (or zombie armies). Sometimes these botnets include thousands or hundreds of thousands of computers. The owner of the botnet is able to instruct all the computers to perform malicious tasks. For example, denial of service (DoS) attacks occur when all computers in a botnet are instructed to access a single website. The web server is unable to process all the requests and hence the website crashes or is unable to process legitimate requests. Such attacks can come from competitors who wish to cripple the competition. Botnets are routinely used to send spam messages. Often all computers within the botnet are instructed to send the same email to all contacts. The botnet provides a level of cover for the criminal operator as it appears the spam messages are from a legitimate source.

Spyware aims to obtain your information without your knowledge. Some spyware is able to capture your web browsing habits, read your email contacts or even read keystrokes as you enter passwords and credit card details. The information is sent back to the criminal who then fraudulently uses the information for various illegal acts. The information obtained by spyware assists criminals to steal your identity. Once they have sufficient personal information the criminal is able to pretend they are you and gain access to bank accounts or perhaps take out a loan or credit card in your name. Identity theft has become an enormous issue not just because of the money being stolen, but also because it taints an individual's reputation. It can then be difficult for victims of identity theft to obtain credit.

FORUM

GROUP TASK Discussion
Often free software is bundled into installation packages that contain adware. Adware causes annoying advertisements often as banners to appear. Do you think adware is malware? Discuss.

FORUM

GROUP TASK Research
Each class member is to research a different specific example of malware. Determine how the malware is distributed, what malicious processes it performs and methods for detecting and removing the malware. Share your results with the class.

GROUP TASK Discussion
A company purchases a new software application and implements it across its network. A virus penetrates the application and corrupts all the company's data.
Who do you think is ethically responsible for the damage caused – the application developer, management of the company, or the employees?

Reliance on Software

We are often unaware of the impact of computers and software on our everyday lives. Household appliances such as ovens, televisions, videos, washing machines and stereos all rely on software to operate. All types of motorised transport, from cars, buses and trains to aircraft, are controlled by software. Utilities such as electricity, water and gas also rely on software. Government authorities such as the Tax Office and Social Security as well as hospitals, banks, universities and other large institutions are all software reliant in some form or another. The software development industry has a huge responsibility to ensure all these systems are reliable and perform their various functions accurately.

GROUP TASK Discussion
Compile a list of items around your school environment that have the potential to cause problems for students and teachers should the controlling software fail.

GROUP TASK Discussion
An aircraft manufacturer has developed a computer system that automatically compensates, if an engine failure occurs, by increasing boost to the other engine. This system also adjusts the control inputs so that the aircraft's handling attributes remain almost identical to normal. Pilots were not informed of this system on the new aircraft.
Identify scenarios where this new system could actually be counter-productive to the intended purpose and potentially cause accidents.

GROUP TASK Discussion
How often have you heard someone say, "Oh, sorry, we cannot do…., the computer is down"?
Describe occurrences from your experiences when this has happened.
Are these occurrences the fault of the computer systems?
Why are the computer systems blamed?
How could problems like the ones you have described be avoided?

Social Networking

A social network is simply a group of people who associate with each other. Often we have a variety of social networks based on some common association. For instance we have social networks built around family, another for friends, one for work and maybe some for special interests such as sports or hobbies.

Social Networking Site
An online, virtual community where people share their thoughts and activities with family, friends or those with similar interests.

Recently the concept of social networks has rapidly expanded into the online world. Perhaps the most well known social networking sites are Facebook, LinkedIn and Twitter, but there are numerous others. The first recognised social networking site was SixDegrees.com, which operated from 1997 to 2001 and was based on the idea that there are a maximum of six degrees of separation between any two people on the planet. In 2002 Friendster and then just a year later in 2003 MySpace commenced operation. MySpace was created by two friends and was the most visited website during 2006. It was sold to Murdoch's News Corporation in 2005 just two years after its launch for a reported $580 million. However by 2008 Facebook (which commenced in 2004) had overtaken MySpace as the top site.

Facebook was developed by a group of college students in 2004. Initially it was only open to students, but it was when it was released to the world that it really took off. The movie "The Social Network" released in 2010, describes how Facebook was founded and exponentially grew to become the most popular social site of all time.

Twitter, which has its own unique instant messaging approach, commenced in 2006. Essentially short messages (called "tweets"), including SMS messages from mobile devices, are added to live feeds. Subscribers can view these feeds and also group "tweets" based on hashtags. For example, the hashtag #sddhsc might be used to group "tweets" from current SDD students.

LinkedIn is a business contacts networking site. Professionals make business contacts and are able to share these contacts with others. Commonly a chain of contacts evolves yet the various links between ends of the chain remain private. For instance a professional contact can be made without knowing any of the people that form the links between the two ends of the chain. Currently social networking sites are becoming increasingly more integrated, for instance twitter feeds can be included on Facebook and LinkedIn pages and vice versa.

Clearly social networking sites have many varied and worthwhile advantages, however there are also disadvantages. Private and personal information is all too easily shared and this can (and has) led to a variety of problems including identity theft, stalking and online bullying. The ability of a single message to be delivered to thousands of unknown others has the potential for unintended consequences, such as thousands of people arriving at a party. Comments made on an online site can remain accessible indefinitely. Views you may hold now and comments made as a teenager may be embarrassing and inappropriate when viewed by some future employer or by your future children and friends.

FORUM
GROUP TASK Discussion
Brainstorm a list of disadvantages of social networking sites. Discuss strategies that could be used by users of these sites and also by the developers of these sites to minimise the effect of the disadvantages you have identified.

FORUM
GROUP TASK Research
Most social networking sites utilise a database to store all user data. Software then reads from the database and formats the information into a webpage for delivery to the user. Research a particular social networking site to determine the database system used and the programming languages used.

Cyber Safety

The online world of the internet includes a variety of different dangers and security issues. All internet connected devices have risks, this includes laptops, mobile phones and even game consoles. Cyber safety is about minimising the risk of such dangers, particularly for children. Although security software and firewalls can assist in this regard, many online issues are unable to be dealt with using software alone. Cyber safety is about protecting yourself when online. To be cyber safe means observing recommended practices and procedures to protect your personal information when using the internet.

The Australian federal government has setup the Cybersmart website (www.cybersmart.gov.au) to inform users about cyber safe practices and procedures. The information below is a brief summary of some of the recommendations from the Cybersmart website:

- Location based services. Services such as Google latitude and Facebook places, allow your current location to be tracked. This makes it simple for unknown persons to find you and it also alerts them that you are not at home. It is critical to check privacy settings regularly to ensure only those you know and trust are able to view your location. It is probably wise to turn off location tracking unless it is specifically required.

- Unwanted contact. Whilst online you may receive messages or see something that makes you uncomfortable. Do not respond to such messages; rather tell someone you trust, such as a parent or teacher. It can be useful to retain a copy of any unwanted contacts in the event that it continues as such evidence can greatly assist when tracking down the perpetrator.

- Cyberbullying. When online the same standards apply as in real life. Spreading false rumours, teasing or making threats online is unacceptable. The best defence is to ignore or block the cyberbully. Often a friend will be the victim of such attacks. It is best to support them and do not forward such messages to others as this will only exacerbate the problem. Instances of cyberbullying should be reported to the appropriate authority - your school or the police.

- Online friends. Meeting new friends online is fun and a great way to discuss common interests. Be aware that people may not be who they claim to be. It is prudent to not share private information to those you do not know in real life. Try to manage your friends list wisely. Having hundreds of online friends can be exciting, but it is very difficult to keep track of such a wide and varied group. If you have not had contact with an online friend for some time it is best to remove them from your contact list. Should you wish to meet an online friend in person never go alone. Take a trusted adult along, such as a parent, and always arrange to meet in a public place.

- Your digital footprint. What you share with friends is often made public. For instance, a photo or video you post for a friend can easily be forwarded to others who then post it publicly. Furthermore these messages and private information can remain accessible forever. When all your online information is combined a picture of you emerges - your digital footprint. Will this digital footprint cause issues later in life? For example, if read by a future employer, partner or even your future children. It is virtually impossible to remove such information completely. Also, respect your friend's privacy and think about the possible consequences before forwarding their information to others.

- Online purchasing. It is prudent to restrict online purchases to well known organisations, such as eBay or brands that have real shops. Examine the details and conditions for sale. Be wary of ongoing purchases where your credit card will be debited many times. Most trustworthy sites will include a system for contacting sellers should you be unsure about the goods, postage or returns policy. Never enter credit card or bank account details without ensuring the transaction is securely encrypted.

- Identity theft. Criminals can obtain sufficient information about you so they are able to use your credit card, access your bank account or take out loans in your name. Be very careful with your personal information. Use virus and/or malware utilities to protect against spyware. Ensure you use complex passwords and protect these passwords. Do not respond to emails that request passwords or any other financial information. And finally watch your bank accounts for any unusual or unauthorised transactions. Often a small withdrawal of a few cents indicates a criminal testing your bank account before they make much larger withdrawals.

GROUP TASK Research

Research an example of cyberbullying. Outline the nature of the bullying, the consequences for the victim and explain how others could have assisted to prevent the bullying occurring in the first place.

GROUP TASK Discussion

Currently security software and firewalls have limited value in terms of cyber safety. Explain reasons why software solutions are unable to effectively keep us "Cyber Safe"?

Evaluating information available through the internet

Huge amounts of information are publicly available through the internet. Information is uploaded or posted by anybody regardless of their qualifications, expertise or experience and in many cases it is difficult to identify the author. This rarely occurs with published print and broadcast media where the author is clearly identified and all material is checked by a variety of skilled personnel, such as reviewers, editors and proof readers. On the internet biased, unsupported, unverifiable, misleading and often incorrect information is common. Although the internet encourages and enhances freedom of speech for all, it has the unfortunate consequence that determining the correctness of such information is difficult and often impossible. This is a particular issue when performing research as verifying the accuracy of information is a necessity.

The following questions and guidelines can assist when evaluating all information sources, but they are particularly important when the information is sourced through the internet. Unfortunately it is often difficult to obtain definitive answers to these questions for much of the information available on the internet, therefore we often need to uncover evidence and then do a little bit of detective work to make an accurate judgement.

- Who is the author? The author's name should be clearly stated. If it is a person (rather than an organisation) then look for their qualifications and or claimed expertise in the area. Often information on an educational or government institution's website has more authority. Phone numbers and street addresses also

improve confidence in the legitimacy of the author. Try to search for other documents which reference the author to confirm their expertise and other details. Many professionals will maintain their own web pages; such pages can assist in verifying their identity and authority.

- Is the information up-to-date? Look for a date on the page or examine the HTML source for a date. It is better if the page clearly states when and how often it is updated. Depending on the topic, it is often obvious if the information is reasonably up-to-date. For example, it may reference some event that has occurred recently.

- Who is the intended audience? Is the information written for the general public or is it written for experts in the area. It can be difficult to determine the validity of information for the general public, whilst expert information is likely to include a list of references.

- Is the information accurate and unbiased? Is the information supposed to be a factual account of the topic or is it the author's opinion. Be wary of information of a political, religious or cultural nature as it can be biased to a particular view of the world. Commercial sites are also likely to be understandably biased towards their own products and services.

- What is the purpose of the information? Be careful of information whose purpose is to market a product or persuade you to agree with their point of view. The purpose of the information should be clear and the information should achieve this purpose. Some commercial sites may claim to be educational, when in fact the purpose is to advertise and promote.

GROUP TASK Research
Wikipedia is often the first website students visit as they perform internet research. Research and explain how information is obtained for inclusion on Wikipedia and how the accuracy of the information is verified.

GROUP TASK Discussion
Books and published articles in magazine and journals are considered to be more authoritative sources compared to most internet sites. Do you agree that this is generally true? Discuss.

RIGHTS AND RESPONSIBILITIES OF SOFTWARE DEVELOPERS

Software developers have *rights* in regard to how other people may use and duplicate their products. Along with these rights come *responsibilities*. Developers of software products are accountable to their customers to provide solutions that are of high quality, ergonomic, inclusive, free of malware and above all perform their stated task correctly and efficiently. Responsible developers will provide systems that allow them to respond to problems encountered by users. Many developers voluntarily adhere to a professional code of conduct.
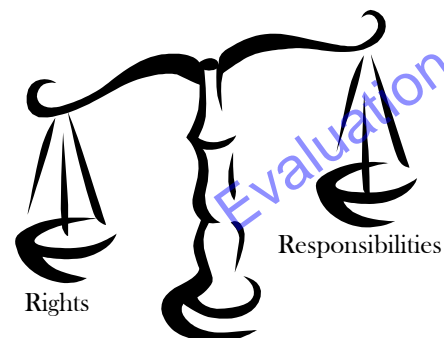

Fig 1.1
A balance between the rights of developers and their responsibilities to customers must be found.

INTELLECTUAL PROPERTY

Intellectual property is property resulting from mental labour. Therefore intellectual property is created by all types of authors, including software developers. An author is the *original creator* of a work, and

> **Intellectual Property**
> Property resulting from the fruits of mental labour.

hence they own the right to control how the work is sold and distributed. With most literary works, such as books, poems and essays it is clear who the author is and therefore it is clear who owns the intellectual property rights. In the case of software products, the situation is often not so clear.

Consider the following:

- A software product is written in a programming language, which was developed by another company.

- A team of developers is utilised to create the software product, with some of those developers being outsourced from another company.

- Some modules of source code are purchased from other software companies.

- The printed documentation is designed and written by a company, which specialises in computer user manuals.

- Finally, the artwork for images, video and animation is designed and created by a graphic design company.

All the people and companies involved in the original creation of this software package are authors and have intellectual property rights in regard to their particular original contribution. *Copyright laws*, discussed later in this chapter, are designed to protect the intellectual property rights of all the authors involved.

Programming Language Creator

Graphic Designers

Developers

Source Code from Outside Companies

Manual Authors

Outsourced Developers
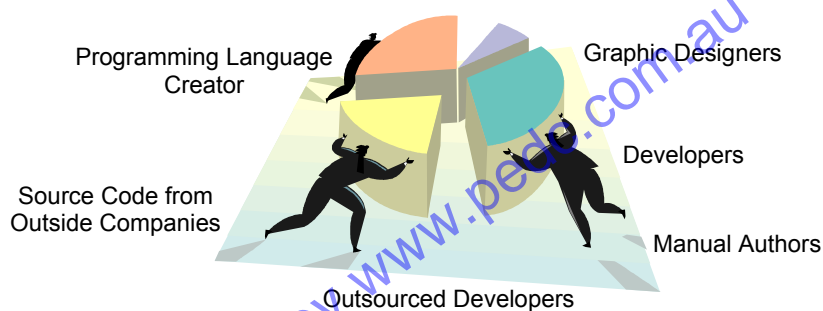
Fig 1.2
All authors have intellectual property rights in regard to their original material.

GROUP TASK Discussion
Software developers expect their intellectual property rights will be recognised. They must also respect the intellectual property rights of others who have contributed to the software they produce. Discuss.

GROUP TASK Discussion
Do you think each of the contributors to a software product should have intellectual property rights? What if the author is employed by a software development company? Discuss.

Different contributors to a software product will have differing requirements in regard to their intellectual property. Most programming languages are either open source or they are sold with a licence requiring the developers to acknowledge use of the programming language without

Royalty
Compensation, usually a portion of proceeds paid to the owner (author) of a right, for the use of a work.

the need for royalties to be forwarded. Original work, written for a specific product, usually requires a royalty to be paid to each contributing author based on the number of units sold of the final product. The particular details of how royalties are to be distributed should be detailed in a contract.

Publishers of software products have a responsibility to ensure the intellectual property rights of all authors involved in the creation of the final product are acknowledged and compensated, as the law requires. As was seen previously, this

Publisher
One whose business is the copying, distribution and sale of works by an author.

process often involves many individuals and companies.

QUALITY

The final quality of a software development project is an important responsibility for all software developers. To develop high quality applications is time consuming and costly. Often compromises have to be made because of financial constraints.

Quality software is developed as a result of thorough planning and testing. Many companies have developed sets of standards to which their employees must adhere.

Fig 1.3
Customer Perception of Quality.

Quality Assurance (QA) is used to ensure that these standards are observed. It is an ongoing process throughout the design and development to ensure and verify the quality of the software. Customers perceive quality according to their expectations about the software product. If the customer's expectations are exceeded, then they will perceive the product to be one of high quality. If the product fails the customer's expectation then the product is perceived to be of low quality.

Quality Assurance attempts to make sure that customer expectations are met and/or exceeded. Different customers will come with differing expectations. However, all customers have a right to have their expectations met. It is the responsibility of the software developer to make sure this occurs. It is in the developer's best interest to know the customer's requirements and to exceed their expectations if they are to continue to operate profitably.

As with most industries and products, quality at a reasonable price is a formula for success. Software developers must strive throughout the software development process to produce high quality software products which exceed their prospective customers' expectations.
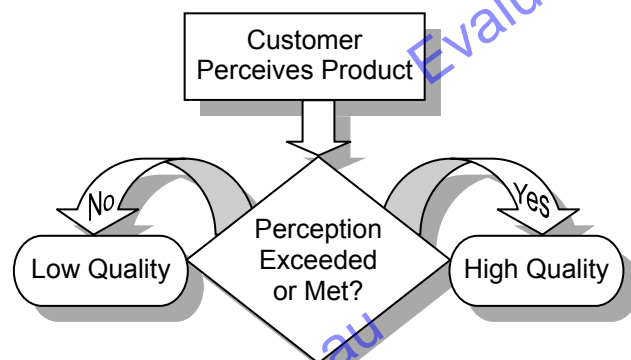
Consider the following list of factors that affect software quality:

- Correctness: Does it do what it is supposed to do?
- Reliability: Does it do it all of the time?
- Efficiency: Does it do it the best way possible?
- Integrity: Is it secure?
- Useability: Is it designed for the end user?
- Maintainability: Can it be understood?
- Flexibility: Can it be modified?
- Testability: Can it be tested?
- Portability: Will it work with other hardware?
- Re-useability: Can parts of it be used in another project?
- Interoperability: Will it work with other software?

FORUM

> **GROUP TASK Discussion**
> Consider the above list of factors affecting software quality. Provide an
> example situation where each factor is not present and hence the quality
> of the software is compromised.

External factors affecting quality

Apart from the design and development process there are other external factors, which need to be considered by developers to maximise the quality and reliability of their products. Such factors include:

- Hardware
- Operating System
- Other software
- Runtime errors

- **Hardware**

The vast array of hardware devices, which includes various network devices, available today makes the job of the software developer difficult. Software developers rely on hardware manufacturers to design and build components that operate reliably on the given platform. The software developer has a right to expect this to be the case. To ensure maximum reliability of their product, software application developers should test their product with a number of

Fig 1.4
A Computer Lab is often used in the testing of software.

different hardware configurations using a variety of different devices. Developers have a responsibility to ensure this testing is as thorough as costs and time allow.

It is inevitable that hardware problems will occur and it is the responsibility of the software developer to build software modules, which react appropriately to these hardware failures. Software, which helps to identify the cause of a hardware failure, or reports the actual failure, will prove to be more reliable and hence higher quality.

Consider the following:

A database for a large business will grow in size over time. It is inevitable that at some time the amount of secondary storage on the hard drive will be insufficient to store the database. The software developer should take this into consideration when designing the system so that warning messages are given when low levels of hard disk storage space occur. This will allow the user to either purchase more hard disk space, or to free up space on the current drive by way of purging old data, which is no longer required.

GROUP TASK Discussion
Should it be the responsibility of the software application to report that the low storage threshold has been reached, or should it be the responsibility of the operating system developers? Discuss.

- **Operating System**

The developer needs to consider the effects that the operating system will have on their software product. Developers have a responsibility to ensure as far as possible that their products interface correctly with the specifications of the intended operating system environments. At the same time, the operating system creators have a responsibility to ensure that specifications published are accurate and that the operating system performs correctly. Developers of software expect the operating system to be reliable and robust. Application software *should not* make changes to the operating system settings. Any changes made to the operating system that are not part of the original specification may cause the operating system to return unexpected results. Application software developers should instead implement checks in their product to ensure that it performs reliably given most operating system settings.

For example, some operating systems allow for the length and format of file names to be altered. Rather than the application changing the operating system settings to suit, the application should access the allowable format of file names and adjust in line with the operating system settings. This will ensure the reliability of not only the operating system but also other installed software applications.

> **Operating System**
> Provides the interface between the software application and the hardware.

GROUP TASK Discussion
Currently the developer's of operating systems have much control over the look and functionality of application software. What responsibilities do operating system developers have to application software developers? Discuss.

Operating systems are continually being improved. It is the responsibility of the software developer to ensure that their product is as *forward compatible* as possible.

Conversely, it is the developer's right to expect that the developers of the operating system will continue to support products written to interface with earlier operating system versions in their upgraded products.

Forward Compatible
The ability of an older system to operate in harmony with a new system.

GROUP TASK Discussion
Most operating systems are regularly updated to correct bugs and security vulnerabilities. Many software applications are now providing a similar facility. Commonly these updates occur automatically via the internet, however it can be annoying for users who are continually bombarded with update notifications. Discuss in terms of developers and also users.

- **Other Installed Software**

Software already installed, or that could be installed in the future, can affect the perceived quality of a developer's product. Software developers have a right to expect that other developed products will not affect the operation of their product. Unfortunately, this cannot be relied upon. Therefore, the developer has a responsibility to ensure that as far as possible other installed software is not able to affect the reliability of their product. Of course, software developers must be careful to ensure that their product also does not adversely affect the operation of other software products.

Consider the following:

A fax program that does not correctly close the serial port connected to the modem will cause other communication programs to encounter errors when accessing the modem. By predicting this as a possible occurrence, the developer should write a module that checks the serial port status before attempting to access the modem. The software can then prompt the user to close the port before attempting to open it and use the modem.

Fig 1.5
Multiple communication programs may access the modem.

GROUP TASK Discussion
If all software developers did the right thing and cleaned up after their applications closed, then problems such as this would not occur. Perhaps standards are required to ensure software developers perform these 'clean-up' tasks. Do you agree? Discuss.

GROUP TASK Discussion
Identify other possible scenarios where installed software applications may cause conflicts with the operation of other applications.

Many problems that jeopardise the integrity of a software product's ability to perform in a reliable manner in conjunction with other installed applications occur at installation time. Shared files can be overwritten without considering the effect on other installed applications. Often file associations are altered at installation time without warning. These actions result in reliability problems for other developers that are difficult to overcome. It is the responsibility of software developers to consider the rights of other software products when coding the installation procedure of their product.

File Association
Defines the application that is executed when a data file is opened, usually by double-clicking.

- **Run Time Errors**

All applications developed must include error checking built into the code. Error checking enhances reliability and minimises *run time errors*. It is arrogant to presume that a program will be completely

Shared File
A file whose code is used by more than one program. Example: A dynamic link library file (.dll file)

error free, and in any case, it is impractical to test for all possible eventualities. However, to minimise the possibility of errors, thereby ensuring greater quality, the developer must include routines that will deal with possible errors in execution. It is the software developer's responsibility to ensure that errors encountered during the execution of the code are dealt with in such a way that execution continues and the product can continue to execute. Fatal errors, which result in execution being terminated, can result in loss of user data. Therefore a developer will find it advantageous, when writing a software application, to code a fatal error detection module which executes a save procedure prior to execution being terminated.

RESPONSE TO PROBLEMS

Software developers have a responsibility to ensure that any problems users encounter with their products are resolved in a timely, accurate and efficient manner. The developer needs to ensure there is a mechanism in place to assist in the identification of errors and their subsequent resolution.

Error messages issued by the application should be unambiguous and should aim to identify the source of the error.

Most large software companies have support departments whose task is to respond to user problems. Even small developers must ensure that they have systems in place to guarantee a quick response to problems encountered with their products. Many support contracts specify a maximum time allowable for a problem to be rectified. Often businesses outsource their customer support to a company that specialises in computer support.

Depending on the severity and how often the error has occurred, different methods of response will be more appropriate.
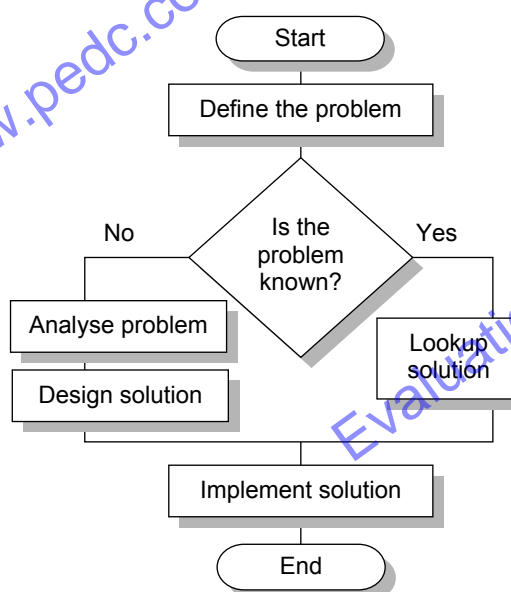

Fig 1.6
The processes undertaken to solve a user problem.

Pages 17 to 35

Not included in this sample chapter

HSC style question:

MineSoft is a software development business that develops, distributes and installs various commercial software applications to the mining industry. The software licence agreement accompanying each of MineSoft's products includes the following clause.

> *13. LIMITATION OF LIABILITY. Should you encounter problems with this software product and MineSoft is found to be at fault then MineSoft agrees to reimburse you the actual value of damages up to the amount you paid MineSoft for the software product.*

Discuss the appropriateness of this clause in terms of the rights and responsibilities of software developers.

Suggested Solution

There are a number of relevant issues with respect to the developer refunding the client up to the purchase price of the product.

If the software application modifies the operating system instructions or contains a virus, or in some way leaves the purchaser vulnerable as a result of the installation of this software, it would be reasonable to expect that the developer accept the responsibility for this. In the same way, if the application calculates values that are critical such as the amount of tax to be paid, or dosages of medicines, for example, and these are calculated incorrectly by the application, clearly the developer should accept the responsibility for this, and the amount to be paid to the client should not be limited just to the refund of the purchase price.

On the other hand, the developer has no control over the environment under which the client will be running their software (including the operating system or hardware used) and therefore cannot be held responsible for issues that become apparent in particular environments. It is assumed that the developer has fully tested their product before distributing it, and that therefore any subsequent issue would not be their responsibility. Having said that, it is often in the interests of goodwill for the developer to look into such issues and advise and fix it for them if at all possible.

It is often the case that licence agreements are the final "fallback" position should legal action be taken. In many cases the developer will spend time and money to help their customers and correct the product. This makes sense for both ethical and marketing reasons.

CURRENT AND EMERGING TECHNOLOGIES USED TO COMBAT SOFTWARE PIRACY

Without some form of protection, software is simple to copy and it is virtually impossible to determine that a copy has been made. This is not the case with physical products, where the effort required to duplicate the product far outstrips the purchase cost. This is why software piracy is such an issue and also is the reason software developers implement various technologies to prevent piracy. Software piracy has been around since the introduction of computers. It was 1976 when Bill gates first encountered piracy. His early versions of BASIC were being stolen and distributed to

other early computer hobbyists. Gates' "An Open Letter to Hobbyists" written in 1976, is reproduced below in *Fig 1.17*.
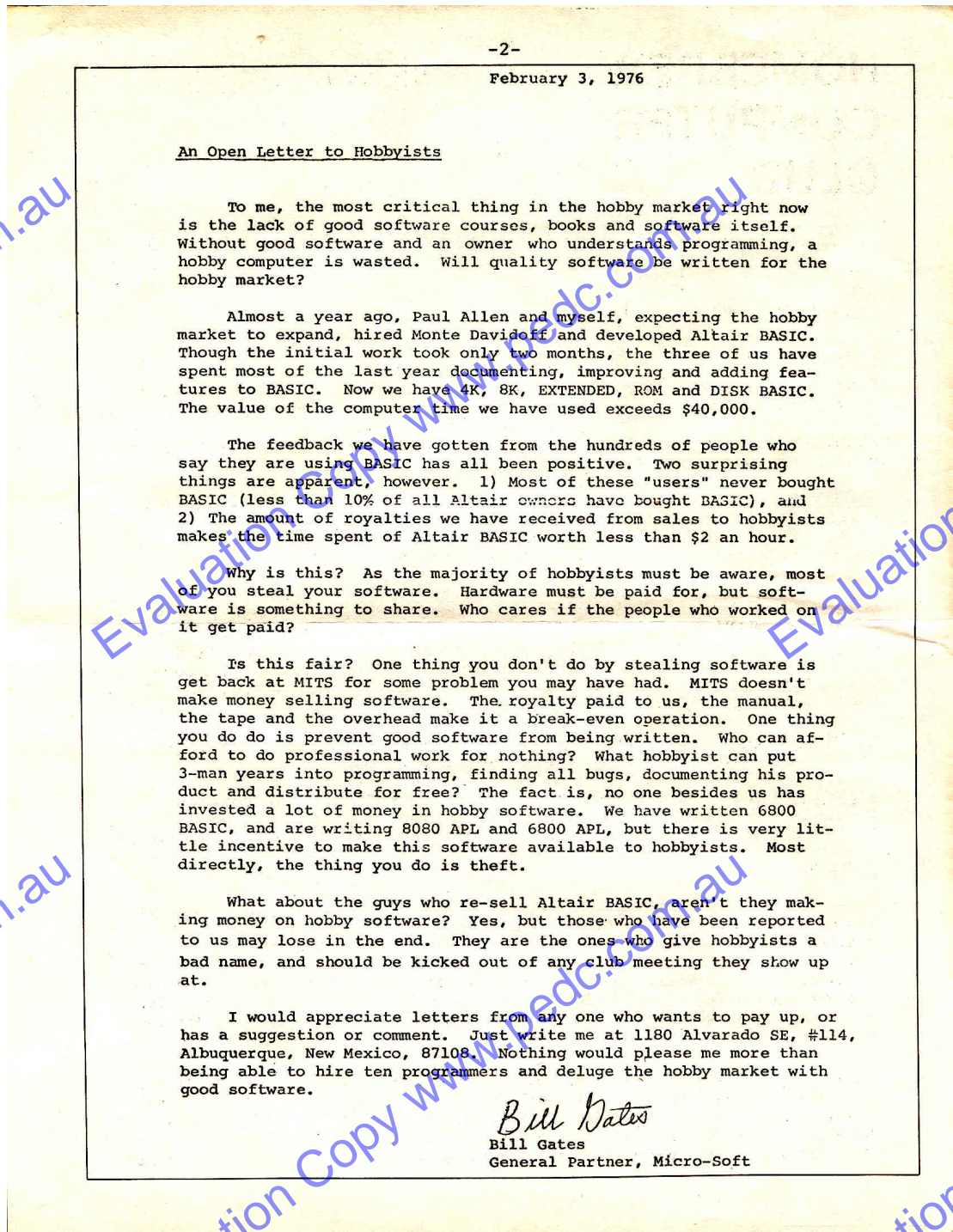


```
                                                       -2-
                                              February 3, 1976


          An Open Letter to Hobbyists


                  To me, the most critical thing in the hobby market right now
          is the lack of good software courses, books and software itself.
          Without good software and an owner who understands programming, a
          hobby computer is wasted.  Will quality software be written for the
          hobby market?

                  Almost a year ago, Paul Allen and myself, expecting the hobby
          market to expand, hired Monte Davidoff and developed Altair BASIC.
          Though the initial work took only two months, the three of us have
          spent most of the last year documenting, improving and adding fea-
          tures to BASIC.  Now we have 4K, 8K, EXTENDED, ROM and DISK BASIC.
          The value of the computer time we have used exceeds $40,000.

                  The feedback we have gotten from the hundreds of people who
          say they are using BASIC has all been positive.  Two surprising
          things are apparent, however.  1) Most of these "users" never bought
          BASIC (less than 10% of all Altair owners have bought BASIC), and
          2) The amount of royalties we have received from sales to hobbyists
          makes the time spent of Altair BASIC worth less than $2 an hour.

                  Why is this?  As the majority of hobbyists must be aware, most
          of you steal your software.  Hardware must be paid for, but soft-
          ware is something to share.  Who cares if the people who worked on
          it get paid?

                  Is this fair?  One thing you don't do by stealing software is
          get back at MITS for some problem you may have had.  MITS doesn't
          make money selling software.  The royalty paid to us, the manual,
          the tape and the overhead make it a break-even operation.  One thing
          you do do is prevent good software from being written.  Who can af-
          ford to do professional work for nothing?  What hobbyist can put
          3-man years into programming, finding all bugs, documenting his pro-
          duct and distribute for free?  The fact is, no one besides us has
          invested a lot of money in hobby software.  We have written 6800
          BASIC, and are writing 8080 APL and 6800 APL, but there is very lit-
          tle incentive to make this software available to hobbyists.  Most
          directly, the thing you do is theft.

                  What about the guys who re-sell Altair BASIC, aren't they mak-
          ing money on hobby software?  Yes, but those who have been reported
          to us may lose in the end.  They are the ones who give hobbyists a
          bad name, and should be kicked out of any club meeting they show up
          at.

                  I would appreciate letters from any one who wants to pay up, or
          has a suggestion or comment.  Just write me at 1180 Alvarado SE, #114,
          Albuquerque, New Mexico, 87108.  Nothing would please me more than
          being able to hire ten programmers and deluge the hobby market with
          good software.

                                          Bill Gates
                                          Bill Gates
                                          General Partner, Micro-Soft
```

Fig 1.17
Bill Gates' "An Open Letter to Hobbyists"

FORUM

GROUP TASK Discussion
In the above letter Bill Gates is trying to appeal to hobbyists to be ethical. At the time, technologies to combat piracy didn't exist and laws were such that it was unclear if software piracy was even a crime. Given the current size, dominance and aggressive business tactics of Microsoft do you think a similar appeal, if made today by Bill Gates, would be justified? Discuss.

Over the years a variety of different technologies and strategies have been used in an attempt to minimise the possibility of piracy. Prior to the internet various anti-piracy strategies were attempted, such as the use of non-copyable data sheets and various forms of disk protection. The widespread use of the internet has made piracy a much more significant issue as software is distributed with ease across the globe. However the internet has also provided the means for software publishers to digitally verify legitimate users of their products. Technologies such as registration codes, serial numbers, encryption keys and back-to-base authentication all aim to verify the user possesses a legal copy of the software product.

Non-copyable datasheet

Commencing in the 1980s non-copyable datasheets were included with some software products. The user needed to enter codes (or answers) from the datasheet to continue using the software. For example, Broderbund software released the game "Where in the World is Carmen Sandiego?" in 1985. The game included a copy of the book "The World Almanac and Book of Facts" published by Pharos Books. During game play the user was required to answer specific questions whose answers could only be found within the accompanying almanac. As the almanac was a large volume it was a significant effort for potential pirates to copy the book – particularly when scanning technology was not widely available. Some other software products included datasheets printed using inks that could not be copied using the photocopiers commonly in use at the time.

Disk Copy Protection

Prior to the widespread use of the internet most software was distributed on floppy disks and then on CD-ROM. A variety of different technologies were utilised to prevent these disks from being copied. For example, some copy protected disks included data on areas of the disk that were not normally able to be read or placed the data on specific sectors of the disk. Most copying software at the time was unable to replicate these features and hence copying the disk was not possible. Over time hackers identified the disk protection methods being used and created software that could either disable the protection system or replicate the disks precisely. New disk protection technologies continue to be developed, however historically all have been cracked fairly rapidly.

GROUP TASK Research
Games consoles, such as PS3, Wii, Xbox 360 and Nintendo DS, use their own proprietary disk or cartridge formats. Although this provides significant protection against piracy it does not solve the problem. Research to determine piracy issues related to games consoles.

Hardware Serial Numbers

A variety of hardware components within every computer include an embedded serial number which cannot be altered. Software can be designed so that it examines these serial numbers and if they don't match then the program will not execute. In effect the software is linked to a particular machine. Many industry specific software products utilise this type of copy protection. Often these software products are sold together with the hardware as a complete system. On a Windows machine, the serial number of the computer can be determined by entering the following command at a command prompt wmic bios get serialnumber. This serial number can be retrieved as the software commences execution and compared to the stored serial number within the executable code. If they do not match then the software will not run.

GROUP TASK Discussion
During initial activation of the Window's operating system details about the hardware on your machine are linked to your copy of Windows. This information allows Microsoft to validate whether Windows is genuine. Non-genuine installations are unable to receive updates from Microsoft. Discuss advantages and disadvantages of Microsoft's validation system.

Site Licence Installation Counter on a Network

In larger organisations software is often installed from a network server. The organisation purchases a site licence which specifies the maximum number of machines who may either install or simultaneously execute the product. The network server includes a utility which monitors the number of times the software has been installed or the number of times the software is currently executing. For installation based site licences the server refuses to provide the installation files if the licensed number of installations has already occurred. If the site licence specifies the number of simultaneous users then each machine must gain permission from the network server before the software will execute. The server maintains a count of the number of current users. This counter is incremented as new users execute the product and it is decremented as users exit the product. Some multi-user applications execute on the server with users accessing the application via a client (often a web browser). In this case the application is executing once on the server and is able to track the number of clients to ensure the terms of the site licence are not exceeded.

GROUP TASK Research
Research to determine the functions of a site licence management application, such as Adobe License Manager. Make a list of advantages for the installing organisation and advantages for the software publisher.

Registration Code

A registration code is used to activate software products during the initial stage in the installation process. For single software licenses the registration code is unique to your installation. Site licenses commonly share a single registration code and also provide some facility for automating the activation process for multiple machines. This allows the installation to be performed without user intervention.

There is a wide variety of techniques used to verify registration codes. Some software applies an algorithm to the registration code so that it can be verified as correct. Although this prevents people entering random codes it does not prevent registration codes being used by others. Today most registration codes are either generated over the internet or are verified over the internet.

If the registration code is generated over the internet then the software developer will have access to some identifying information from the purchaser. For instance, an IP address, the person's email address, or even a serial number from the purchaser's computer. This information can be used to generate a specific registration code which will only activate the software if the other information is also present.

Other registration codes supplied with software must be verified over the internet for the software to continue operating. In many cases failure to verify results in reduced functionality or the software ceasing to function after a set period of time. In this case the verification process contacts a server which records the registration code. Therefore the system is able to detect if a registration code has already been used or in the case of site licences has been used too many times.

GROUP TASK Discussion

Although many computers are permanently connected to the internet there are many which are not. Therefore it is unreasonable for software to not install or cease to function simply because software cannot be activated or registration codes cannot be verified. Discuss.

Encryption Key

Encryption effectively scrambles the data or executable code in such a way that it is virtually impossible to make sense. The decryption code is required to reverse the encryption process. There are two common systems of encryption; single key and two key systems. Single key or secret key encryption uses the same key to both encrypt and decrypt the data. Two key systems use one key (called the public key) to perform the encryption and a different key (the private key) to decrypt. Public/private two key systems operate within the secure sockets layer (SSL) protocol used to perform most financial transactions over the internet.

Single key systems are difficult to implement over networks because the common key must be securely transmitted to the other party if the system is too be secure. Two key systems allow the encryption (or public key) to be shared with anyone as only the owner of the private key will be able to decrypt the message. For example (refer *Fig 1.18*), as the purchaser of a software application, your computer generates a

```
BEGIN Two Key Software Transfer
    Publisher computer requests public key from purchaser computer
    Purchaser computer generates public/private key pair
    Purchaser computer sends public key to publisher computer
    Publisher computer uses public key to encrypt software package
    Publisher computer transmits encrypted package to purchaser computer
    Purchaser computer decrypts package using private key
    Purchaser computer installs software package
END Two Key Software Transfer
```

Fig 1.18
Two key software transfer algorithm.

public/private key pair. The public key is transmitted to the publisher's computer which uses it to encrypt the software package. The package is sent to your computer which then uses your private key to perform the decryption prior to installation. The generation of the keys is performed automatically and transparently without the need for user interaction. Once the transfer is completed the keys are destroyed. It is also possible for a two key system to be used to securely transfer a single key encryption code. The actual package is encrypted using the single key and then the same single key (which was transmitted securely) is also used to decrypt the package.

GROUP TASK Research

Most two key (public key) encryption systems are based on prime numbers and the RSA algorithm. Research the general nature of RSA algorithms including their use of prime numbers.

Back-to-base Authentication

Back-to-base authentication means the application contacts the software publisher's server to verify the user or computer holds a valid software licence. If the licence is verified as correct then the software will execute. Back-to-base authentication can be used when installing the software or it can be used each time the software is run. For software which requires internet connectivity to perform its functions, back-to-base

authentication is an excellent strategy for preventing piracy. However for software whose functions do not use the internet back-to-base authentication restricts user access when an internet connection is not present.
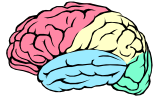
Consider the following ways in which software can be pirated:

- End User Copying. Individuals make copies of software or lend software to friends and workmates.
- Piracy of OEM Products. Manufacturers of Personal Computers or Original Equipment Manufacturers, install operating systems and other applications illegally on new machines they sell. A licence for all software should accompany the newly purchased machine.
- Counterfeiting. Unscrupulous organisations copy original software and then on sell it as legal copies. Often the packaging and manuals are also included and on the surface, the product looks, if not exactly the same, then very similar to the bona fide article.
- Mischannelling. Software sold with discounted licences is resold as a fully licensed product. An example of this would be an academic version being retailed as a fully licensed version.

GROUP TASK Discussion
Do you think the methods outlined on the previous pages for combating software piracy are effective against end user copying, piracy of OEM products, counterfeiting and mischannelling? Discuss.

HSC style question:

(a) Outline the purpose of copyright laws.
(b) Describe the relationship between copyright law and software licence agreements.

Suggested Solutions

(a) Copyright laws are in place to acknowledge the time, effort and intellectual creativity put in to the development of software by software developers. By protecting their authorship through these laws, it encourages others to value the work of the developers, and to encourage others to develop new ideas and products without fear of being taken advantage of. Copyright law aims to afford similar rights to the ownership of intellectual property as are provided for ownership of physical property.

(b) A software licence agreement is a legal contract between the author of the software and the user which defines what a purchaser of the software licence may do with the software whose licence has just been purchased. These defined conditions (such as how many copies may be run concurrently, whether or not the software can be shared with others, etc.) are specified so there is no misunderstanding as to how the software may be used, in a way which protects the author of the software by ensuring that the intellectual property is appropriately acknowledged and protected by copyright laws.

SET 1B

1. Within a software licence agreement there is a clause which restricts the amount of money a user can claim if the software causes some issue. Which of the following best describes this clause?
   - (A) warranty
   - (B) limited use
   - (C) limited liability
   - (D) term

2. Falsely claiming ownership of someone else's work is:
   - (A) piracy.
   - (B) plagiarism.
   - (C) counterfeiting.
   - (D) theft.

3. The purpose of copyright laws is to:
   - (A) provide incentive for creative endeavour.
   - (B) protect the expression of ideas.
   - (C) provide the owner with the sole rights to copy and distribute a work.
   - (D) All of the above.

4. Software where users are encouraged to distribute copies but must pay the owner if they continue to use the product is:
   - (A) commercial
   - (B) shareware
   - (C) open source
   - (D) public domain

5. In terms of software copyright law, which of the following is FALSE:
   - (A) One copy may be made for backup purposes.
   - (B) Derivative works based on the original cannot be made.
   - (C) Distributing copies is not permitted.
   - (D) Backup copies may be retained if the licence is sold.

6. Reverse engineering is the process of:
   - (A) studying the original design of the product.
   - (B) analysing a product to determine its original design.
   - (C) using a product as a major part of a new product.
   - (D) translating executable code back into a higher-level language.

7. Software Piracy can be best described as:
   - (A) theft.
   - (B) copying software.
   - (C) taking from the rich.
   - (D) infringing copyright.

8. The only classification of software where copyright has been relinquished is:
   - (A) commercial.
   - (B) shareware.
   - (C) open source.
   - (D) public domain.

9. Which of the following is TRUE:
   - (A) Works must be registered with the copyright office before copyright laws will cover a work.
   - (B) The copyright symbol © must be displayed on all copyrighted works.
   - (C) Transfers of copyright must be in writing to be legally valid.
   - (D) Copyright laws are designed to safeguard publishers.

10. Selling an academic version of a product as a fully licensed version is known as:
    - (A) mischannelling.
    - (B) counterfeiting.
    - (C) end-user copying.
    - (D) piracy of an OEM product.

11. Define each of the following terms:
    (a) plagiarism
    (b) shareware
    (c) open source
    (d) public domain
    (e) reverse engineering
    (f) decompilation
    (g) backup
    (h) single key encryption
    (i) public/private key encryption
    (j) authentication

12. Explain the connection between intellectual property, copyright laws and licence agreements. Who is responsible for creating each of these?

13. Apparently, the majority of home users have illegal software installed on their computers. Discuss the ethics of this situation in terms of firstly, the home user, and secondly, the software developer.

14. Choose a large software development company with which you are familiar. For this company, research the methods they are utilising to combat software piracy.

15. A software product is to be distributed to customers via the Internet. Devise some strategies the distributor could implement to ensure the owner's copyright rights are protected.

USE OF NETWORKS

The widespread use of networks, including the internet, has revolutionised the development of software. The ability to collaborate and share resources greatly improves productivity. Although network speeds continue to improve they still provide a bottleneck for network based software. Changes to the user interface in terms of how data is entered and retrieved across the network can greatly improve response times. In addition security and privacy concerns (discussed earlier in the chapter) must be addressed when data is transferred across public and often insecure networks.

Network
A collection of computers connected electronically to facilitate the transfer of data and software.

Response Time
The elapsed time between a request (or user input) and a response (or output to the user) from the system.

Use of networks by software developers

Currently all but the smallest software applications are developed collaboratively by a team of developers. In many cases (particularly for open source development) the programmers have never met in person and may live in different countries. Without networks, in particular the internet, many of the applications we use on a daily basis would not exist. In addition to communicating with other developers many programming language libraries and related documentation reside on the web. During coding programmers are able to research and then in an instant download and incorporate existing source code into their application. Clearly both online collaboration and ready access to code and other resources greatly improves the productivity of software developers.

Consider collaborative development

The idea of collaborative development sounds terrific, however having multiple programmers all working on the same or different modules of source code can easily become a management nightmare. A variety of different versions of the same module can and do exist and determining which is superior is not clear. Furthermore each developer requires access to current versions of related modules, yet these related modules may well be currently undergoing their own revisions.

GROUP TASK Discussion
Brainstorm a list of issues that are likely to be encountered when multiple developers are working on a single software project. Can you suggest strategies to help manage these issues?

GROUP TASK Research
There are a number of software tools available to automate the management processes needed to collaboratively develop software. Git is one such product, and is itself an open source collaboratively developed project. Research a collaborative development software tool and list its significant functions.

Use of networks by users when using network based software

No doubt we've all accessed a web site, waited and then given up because the response time is too slow. Even sites that generally respond quickly can experience poor response times during periods of high activity. For software developers response times can be difficult to control with certainty. The network conditions and server load are often beyond the software developer's control. For instance, the slowest network link in the transmission determines the speed of the transmission, but for internet connections this is completely beyond our control. Nevertheless there are many design techniques which software developer's can use to make the most of the available network and server conditions.

Many RAD tools, including Flash, can create unnecessarily large files. Although media rich content can look amazing, it will have little impact if nobody can be bothered waiting for it to download. When embedded video and large images are included then response times will be severely compromised. For example, many websites use a single flash file with embedded video and images for an entire website, when not carefully designed such sites can take an eternity to load. Text based content is many times smaller in terms of file size compared to video and images. Although techniques such as compressing and streaming video can assist, often a link to such media is a more effective strategy.

Most network based software uses a client-server model. The client application is installed on the user's computer or runs within a web browser. Data or requests entered by the user into the client application are transmitted over the network to the server. The server application runs on the server and responds to the client's request by retrieving data from a database, which the server application then formats and transmits back to the user. Each client request and server response must travel over the network. Therefore we need to minimise the number of request and responses that applications perform and also minimise the quantity of data transmitted. Careful design of the order and number of inputs made by users before the data is sent over the network can yield significant performance improvements. Similarly minimising the amount of data retrieved by the server will also improve performance. Many commonly used network applications can provide some clues in this regard.

Consider the following:

- Consider searching for images using Google. The search criterion is entered in its entirety before being sent to Google's servers. The server responds with a limited number of small low resolution images. The user then selects one image at a time to view in higher resolution.

- A car insurance company provides an online quotation system. To determine the type of car, the user is presented with a list of all makes (Ford, Holden, Toyota, etc.). The selected make is sent to the server and a list of models is returned (Fiesta, Focus, Falcon, etc.). The user selects the model, which is sent to the server and a list of model variants is returned (G, GL, GLX, etc.).

GROUP TASK Discussion
Response times and user interface design are closely linked. Use the above two scenarios as examples to assist explaining the relationship between response time and interface design.

THE SOFTWARE MARKET

The software market does have its own unique requirements compared to other industries. However, having said that, general marketing concepts still apply. One common misconception is that marketing and selling are interchangeable terms. This is not the case: marketing is a much broader area that includes sales as well as a great deal more.

> **Marketing**
> The effort to determine and meet the needs and wants of current and potential customers.

Maintaining a software company's standing and position in the market requires that the company consider social and ethical issues when developing marketing techniques and strategies. The effect of not considering social and ethical issues will result in undermining the confidence of consumers to trust the assertions of software companies.

Marketing is often described in terms of the *Four P's of Marketing* namely, product, place, price and promotion. The relative proportion of resources allocated to each of these four efforts is often

Fig 1.19
The Four P's of Marketing.

called the *marketing mix*. Let us consider each of these in terms of the software market with special consideration being given to the social and ethical responsibilities of those involved in the marketing process.
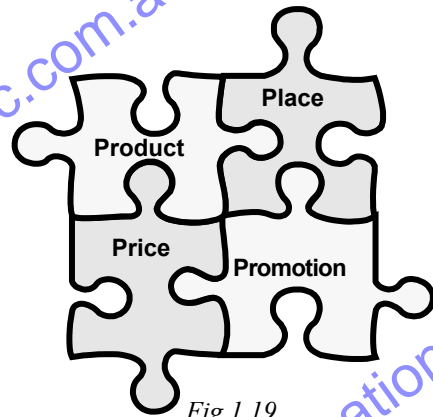
PRODUCT – WHAT DO YOU SELL?

Firstly, you are not so much selling a software product as selling an expectation of the product. The expectations of consumers must be realised in the actual product. It is unethical (and also poor marketing) for a product to not live up to the expected functionality desired by customers.

Consider the following scenario:

A desktop publishing application is advertised as having the ability to import graphics of different types. A potential customer purchases the product for their graphic design business with an expectation that they will be able to import graphics from their favourite photo editing software product. The customer installs the product, undergoes extensive training and then commences using it within their business. They discover that in fact the product is unable to import graphics of the desired format.

> **FORUM**
> **GROUP TASK Debate**
> The user manual contains details of the graphic formats that can be imported, and the one the customer requires is not mentioned. Debate the ethical issues arising for both the customer and the software company's marketing department.

Many software development companies specialise in writing custom software. These developers must ensure that the objectives of the system are realised once the final product has been developed and implemented. Not doing this makes no sense from a marketing point of view as well as from an ethical point of view.

Products are often produced with insufficient knowledge of the requirements of the marketplace. The sales people are then confronted with the unenviable task of selling a product that does not conform to the requirements, and hence the

> **Custom Software**
> A product designed to solve a particular problem or set of problems.

expectations and needs of the potential client base. This can lead to sales people making exaggerated and false assertions in regard to the product's capabilities. This, of course, is unethical and in the longer term will result in industry dissatisfaction with the product and reduced sales. It is vital that customer needs, requirements and expectations are fully understood before software products are designed.

PLACE – WHERE DO YOU SELL IT?

The physical place where a product is sold is related to the type of product and its intended audience. Software is distributed in various locations using various methods. For example:

- Shop fronts, software retailers, e.g. computer stores, book stores.
- Industry specific distributors, e.g. businesses specialising in the sale of software to a particular industry.
- Direct sales, e.g. over the Internet, by mail order, direct mail.

The place where the software is sold can have an effect on the perceived ethical status of the product and the expectations of potential customers.

The software developer has a social and ethical responsibility to their distributors in regard to pricing. Often distributors will be given discounts for volume sales. These discounts can make it difficult for smaller distributors or distributors offering enhanced services to compete with the large volume distributors. It is the responsibility of the software company to ensure their pricing structure achieves a balance that is fair and equitable for all.

Consider the following:

A product that automates the storage and retrieval of legal documents for a law company is to be marketed. It would be inappropriate to sell this product through retail chains of computer stores. In fact attempting to sell via computer retailers could potentially reduce the perceived status of the product in the eyes of the legal profession. A company that specialises in legal software distribution would be a better marketer of this product. Although the product is the same regardless of the distribution method, the perceived ethical value of such software is enhanced when it is distributed using an industry specific distributor.

> **GROUP TASK Activity**
> There are numerous industry specific software applications. Make a list of any industry specific software used at you or your parents' place of work. Can you determine where this software is sold?

PRICE – WHAT DO YOU SELL IT FOR?

There are two commonly used methods for pricing a product. The first is based on a 'cost-plus' basis, and the second is referred to as 'consumer-based pricing'. The cost-plus method is relatively straightforward. Simply take the cost of producing the product and add the profit margin. Consumer-based pricing is not so straightforward. This method involves looking at what the consumer wants and how much they are prepared to pay for it. From both a marketing and ethical perspective, the second method makes more sense. By examining consumer needs and how much they are prepared to pay for that need to be satisfied, products will be produced which not only supply the consumer with an economical product but also enhance the social and ethical standing of the software development industry as a whole.

PROMOTION – HOW DO YOU SPREAD THE WORD?

Promotion consists of all the ways used to persuade people to buy products and services. Software marketers have an ethical responsibility about what they say and how they say it when selling to prospective customers. Advertising can be helpful to consumers who are investigating what is available in the marketplace. Promotional activities should help consumers to make informed choices. These advertising activities should provide accurate information that is inoffensive and not misleading.

Word of mouth is a powerful promotional technique. It is important to consider existing customers' needs as part of a promotional strategy. In regard to software, it is especially important to provide support both in terms of maintenance upgrades and *bug* fixes, as well as ongoing training support.

Consider the following:

A software company has developed a new computer game and is now in the process of developing a marketing strategy. Part of this strategy is a television advertising campaign using real life video footage with human actors depicting various scenes within the game. However, the actual game is entirely computer generated animation.

GROUP TASK Discussion
Discuss the ethical issues that arise with the television campaign being employed as part of the marketing strategy.
Is the consumer being given misleading information?
How great an influence would the television campaign have on the consumer's decision to purchase this product?

THE EFFECT OF DOMINANT DEVELOPERS OF SOFTWARE

In any market if one player's product dominants then it becomes the default purchase. This is largely what has occurred in the operating system market and also in the word processor and spreadsheet market. Microsoft's Windows family of operating systems is sold preinstalled on the large majority of personal computers and laptops. It requires effort on the part of the user to even purchase a computer without Windows. Furthermore most users are relatively familiar with Windows way of doing things, regardless of its performance or other limitations. Similar arguments are also true for Microsoft Office. The open source community has produced Open Office with similar functionality to MS Office but free of charge. Despite Open Office being free, MS Office continues to retain the majority of users.

Microsoft is not the only software development company that dominates - although its dominance in the operating system market is extreme. Apple has made a largely marketing decision when they include their operating system and other applications as part of any hardware purchase. This includes not only their computer products but also portable devices such as iPhones, iPads and iPods. In addition, for portable Apple devices software can only be purchased (or acquired) through Apple's iTunes App Store. Currently Apple receives a significant portion of all revenue (reportedly 30%) from every App sold.

There are also some surprising dominant players. Currently Apache dominates the web server market (with Microsoft's IIS a close second). Apache is an open source product and hence is distributed free of charge. The open nature of the internet is what made it possible for Apache to gain favour. Those who operate web servers are technically able and hence they are better positioned to ignore advertising and marketing campaigns in favour of functionality and performance. Apache's story demonstrates that in the software market (and largely due to the internet) it is possible for David to beat Goliath.

GROUP TASK Debate
"If a product is dominant within the market then it must have the best functionality and performance" Do you agree? Debate both sides.

IMPACT OF NEW DEVELOPERS AND SOFTWARE PRODUCTS

Although there are companies and products which dominate existing software markets there is still room for new players. Often new developers emerge due to inventiveness. That is, they invent a new software product that breaks new ground. Both Google and Facebook are prominent examples of companies which began with a bright idea. In early 1997 Google was nothing more than a Stanford University research project by Larry Page and Sergey Brin. By 2004 Google was handling some 85% of worldwide search requests. Facebook which officially began in 2004 was also a student project that took the world by storm. These are extreme examples and there are numerous others who have forged a successful living from new and innovative software.

Currently (2011), small software applications for mobile phones (known as Apps) are creating a significant market. Apps for Android phones and for Apple iPhones can sell in the millions. Otherwise unknown developers are able to access an enormous worldwide market using the Android Market (now owned and operated by Google) or through Apple's App Store. Most Apps sell for just a few dollars (and many are free) however when there are millions of potential customers a popular App can make significant profits. For example, the popular game "Angry Birds" currently has revenue exceeding $70 million for a very modest initial outlay. The "Angry Birds" development company Rovio Mobile is another business begun by university students – Niklas Hed, Jarno Väkeväinen, and Kim Dikert at Helsinki University Technology.

As China and other countries begin to lift their standard of living the market for software continues to grow at an enormous rate. The future for new and innovative software developers looks very healthy!

GROUP TASK Research
Research current examples of new and innovative software. List any inventive functions performed by the software and outline the history of the people who developed the software.

SET 1C

1. The effort to determine and meet the needs of current and potential customers is called:
   (A) Selling.
   (B) Marketing.
   (C) Promotion.
   (D) Researching.

2. Product, place, price and promotion are known as:
   (A) Marketing principles.
   (B) Marketing strategies.
   (C) The four P's of marketing.
   (D) The marketing mix.

3. A product is expected by consumers to have a particular feature. It in fact does not contain this feature. This is an example of:
   (A) a product failing to meet customer expectations.
   (B) false advertising.
   (C) unethical selling practices.
   (D) All of the above.

4. Which of the following is TRUE in terms of pricing:
   (A) Customers will always purchase a superior product, regardless of price.
   (B) 'Cost-based' pricing is more successful in terms of marketing.
   (C) Pricing should be based on perceived economic value for the consumer.
   (D) Discounting for volume sales will increase the overall volume sold.

5. A software distributor decides to sell its new product over the internet. Which of the following marketing elements are they considering?
   (A) Product.
   (B) Place.
   (C) Price.
   (D) Promotion.

6. Which of the following best describes response time?
   (A) The time taken for a user to understand the interface.
   (B) The time taken by a web server to respond to a request.
   (C) The time between a request and a response.
   (D) The response received by a timing device.

7. The proportion of resources allocated to each of the *Four P's of Marketing* is called:
   (A) Marketing principles.
   (B) Marketing strategies.
   (C) The four P's of marketing.
   (D) The marketing mix.

8. Which is NOT open source software?
   (A) Apache web server
   (B) MySQL database
   (C) Open Office
   (D) Angry Birds App

9. The best time to consider potential customers' requirements is:
   (A) when the marketing plan is being developed.
   (B) once the product has a large client base.
   (C) before the product is designed and produced.
   (D) during the testing of the product.

10. Software developers who develop custom applications:
    (A) must know their customers' expectations and requirements precisely.
    (B) need not be so concerned with the quality of their products.
    (C) do not require sales personnel to sell their products.
    (D) usually are more interested in the needs of their customers.

11. Name a dominant software application and its developer. Explain the effect of this dominant application on potential and actual competitors.

12. Imagine you are a small software developer. You have written an application that is of particular use to home users. Discuss how you would go about marketing this product yourself.

13. Choose a large software development company with which you are familiar. For this company, research the methods they are using to market their products. Can you suggest any improvements they could make to these marketing strategies?

14. Competition often has a significant effect on marketing. Do you think it is ethical to examine competitors' products before designing, producing and marketing new products? Explain.

15. A monopoly is when only one company sells a particular type of product - in other words, consumers have no choice. What do you think are the marketing consequences of a monopoly in the software development industry?

LEGAL IMPLICATIONS

As we have seen there are many significant social and ethical issues that need to be considered by those in the business of creating and distributing software. In this section, we consider a range of significant legal case studies both within Australia and also internationally.

Software implemented on systems throughout a country (or throughout many countries) can result in significant legal action if the software contravenes the law of the country in some way or its development does not comply with the legal contract between the software developer and the customer. The legal action may be the result of copyright breaches or can arise when software does not perform as intended.

NATIONAL LEGAL ACTION

Following are two well known cases that resulted in legal action within Australia:

- RACV versus Unisys
- NSW Tcard system

RACV versus Unisys

The RACV is a motor vehicle and general insurance company similar in nature to the NSW NRMA. In 1993 the RACV issued a Request for Proposal (RFP) also known as a Request for Tender (RFT) after making the decision to convert from its current paper-based claims management system to an electronic storage and retrieval system.

Unisys Systems won the proposal to supply RACV with their new system, one that would lead to more efficient handling of claims, reduced costs and an overall improvement of customer satisfaction.

During presentations of the new system to RACV, Unisys assured RACV that the response times and functionality shown during demonstration would be indicative of the live system. Acceptance testing was signed off in March 1995 and the project continued.

When the system was delivered however, the expected response times and functionality failed to eventuate. RACV ceased using their new system and reverted back to the paper-based system. Unisys' system would need to be redesigned. Even after a redesign, the new system failed to deliver on expectations and in fact was reported to be slower than the original paper-based system.

In June 1996, RACV terminated their contract with Unisys and sought damages through the courts citing false representation and misleading conduct with regards to the functionality of the system.

One critical issue that was to be a key-point in the court case was that the RACVs RFP was not included in the contract signed with Unisys. The RFP supplied by RACV stipulated that response times should be between a 2 and 4 second retrieval. Unisys argued that the contract that they signed, only stated a delivery of functional specifications, not response times as requested by the RFP.

The court over ruled Unisys arguments and awarded damages of $4 million to RACV stating that the RFP even though not part of the signed contract, were incorporated by necessary implication.

FORUM

GROUP TASK Debate
Based on the above information (and your own research) make up a list supporting Unisys and another list supporting the RACVs case.

NSW Tcard system

The NSW Tcard system was a proposal put forward by the NSW State Government in 1996 to replace the automated ticketing system that was being used. It was hoped that the Tcard system, based on Smart Card technology would be operational in time for the Sydney 2000 Olympic Games.

The Sydney Olympic timeline passed however with various legal wrangling between parties preventing the start of development. Finally in 2002 the contract for the development of the system was finally awarded to the ERG Group and development commenced with a 2005 deadline given for full ticket integration.

The project was plagued by delays, with both the NSW State Government and the ERG Group blaming each other for failure to deliver on particular milestones. The deadline for a full commuter trial to take place was set for the end of 2004 but that deadline passed without the milestone being reached.

A trial by school students of the new technology was commenced in 2005 and by 2006 had expanded to include some private bus companies.

Still the delays continued and when the second deadline for commuter trials expired in 2007 without delivery being met, the State Government terminated the contract with the ERG group and sought damages of $95 million.

Following the termination of the contract, the school student Tcard system that had been in operation since 2005 was also brought to an end.

In its initial submissions, lawyers for the ERG Group allege that the NSW State Government refused to allow live testing of its system and that it had also stipulated that nothing could be installed until the system was 100% perfect. The ERG Group have indicated that this was one of the main reasons why the project failed.

The lawsuit has been set down to be heard by the end of this year (2011).

- ERG alleges that it did not receive sufficient cooperation from operators such as RailCorp and that they were not provided with reliable system testing reports throughout the development of the project.

- ERG has successfully developed smart card transport systems for Hong Kong, San Franciso, Stockholm, Singapore and a host of other countries.

- ERG have filed a counterclaim in the NSW Supreme court for $200 million for illegal termination of contract by the NSW State Government.

GROUP TASK Research
Research using the internet, to determine the outcome of the NSW State Government lawsuit. Also determine the status of ERG's counterclaim.

GROUP TASK Debate
Based on the above information (and your own research) make up a list supporting the NSW State Government and another list supporting ERGs case.

GROUP TASK Discussion
Is it reasonable for new software to be 100% perfect, particularly when no live testing has taken place? Discuss.

INTERNATIONAL LEGAL ACTION

Following are three well known cases that resulted in international legal action:

- Microsoft versus Apple Computer (GUI)
- Microsoft versus Netscape (Web browser)
- Search engines (Google versus national censorship laws)
- Napster versus Metallica (and RIAA)

Microsoft versus Apple Computer (GUI)

Apple computer released the Macintosh operating system with their range of Macintosh Personal Computers. This operating system was the first commonly available Graphical User Interface (GUI). Microsoft soon after released Windows, which in many ways looked and felt like the Apple Macintosh operating system. Both the Macintosh and Windows systems were distributed globally before any legal action began. Apple claimed that Microsoft had stolen their idea. After a long, drawn out court battle, a settlement was reached allowing both companies to continue to distribute their GUI operating systems.

FORUM

> **GROUP TASK Discussion**
> Let's for a minute imagine that Apple Computer had won this case and Microsoft Corporation was ordered to withdraw the Windows operating system product from the global market.
> Discuss the effects this decision would have had on Software Developers, Governments, Software Retailers and other businesses as well as the home computer user.

FORUM

> **GROUP TASK Debate**
> With your knowledge of copyright law do you think that Apple Computer had a case to argue against Microsoft Corporation?
> Debate both sides of the case and explain your reasoning for, firstly the Apple Computer side, and secondly the Microsoft Corporation side.

Microsoft versus Netscape (Web browser)

During 1990 to 1992, the Internet was primarily of academic interest. Most information was text-based being distributed via FTP and Gopher servers. It was not until 1993 that web browsers (and websites) as we know them today began to emerge.

The first widely available browser was Mosaic, the forerunner to Netscape. The first version was written for UNIX machines in 1993. Other browsers were under development such as the open source based Amaya, which is still in existence today. By the end of 1993, versions of Mosaic were available for Windows and Macintosh machines. At that time, there were some 500 web servers in existence, the first commercial site being Digital Equipment Company. In 1994, the web really took off and by the end of that year some 3 million servers were operating. Mosiac Netscape became a company with more than 80 percent of the browser market. Because of Netscape's educational roots, their browser was freely distributed to students, teachers and researchers.

In 1995, the potential of the World Wide Web was obvious and Microsoft could see this. Windows 95 was released with Microsoft Network (MSN) client software included, the aim being to establish the Microsoft Network as a parallel network to the

World Wide Web. MSN as a separate network was not successful and Netscape continued their domination. The release of Netscape Navigator 2 further secured Netscape's position. Navigator 2 included Email, frames, support for progressive JPEGs, Java support as well as support for SSL (Secure Sockets Layer) encryption. Microsoft's Internet Explorer 2 existed with a small yet significant following.

The battle for market share between Netscape and Microsoft began in 1996. Microsoft released Internet Explorer 3 and gave it away for free! Internet Explorer 3 had similar functionality to Netscape's product. Both Netscape and Microsoft were developing parallel technologies, each using their own standards. Web designers at that time often produced two versions of their site, one for Netscape browsers and one for Microsoft browsers. The World Wide Web Consortium (W3C) was supposedly the official standards body; however neither Netscape nor Microsoft adhered to these standards. In 1996, there were some 10 million web servers in operation.

The final crunch for Netscape came in 1997 when Microsoft packaged Internet Explorer 4 with Windows 95. Microsoft integrated the web into Windows 95 using the 'Active Desktop' concept. Netscape filed lawsuits against Microsoft, alleging Microsoft was taking illegal advantage due to their vast share of the operating system market. Microsoft's response was that Internet Explorer was an integral part of their operating system. No real resolution was ever reached.

In 1998, Netscape had lost most of its market share as a result of Microsoft's domination. Netscape's browser became an 'open source' product. Source code was released to the open source community as the Mozilla project. In 1999 America Online (AOL) purchased Netscape and then in 2003 AOL provided the initial funds to setup the Mozilla project. The Mozilla project now develops the popular Firefox browser together with a variety of other open source applications for the web.

GROUP TASK Debate
In your view were Microsoft's actions in regard to its marketing of Internet Explorer reasonable? Debate both sides.

GROUP TASK Discussion
How have web browsers altered the way in which we communicate? What makes web browsers such useful applications compared to other methods of communication? Discuss in relation to computer-based communication as well as other forms of communication.

Search engines (Google versus China's national censorship laws)

China has some of the most restrictive censorship laws in the world. Google on the other hand represents itself as being a bastion of free speech and free press. It was therefore somewhat inevitable that a Google search engine located in China would cause issues for both the Chinese Government and Google.

Google china first began operations in 2005 and in early 2006 google.cn was launched. The results from google.cn searches were filtered and censored to comply with China's local censorship laws and regulations.

In early 2009 China blocked access to Google's YouTube site and in January 2010 a number of US tech companies operating in China claimed to have been hacked. Google responded with an announcement that they were no longer willing to censor searches in China and that they may pull out of China altogether. On March 23, 2010 Google began redirecting all google.cn searches to Google.com.hk – its Hong Kong

search engine. Although Hong Kong is part of China, it is not subject to China's restrictive censorship laws.

On March 30. 2010 China banned access to all Google sites. Apparently any attempt to access a Google site resulted in a DNS error. However the ban lasted just 24 hours. The redirection to the uncensored Hong Kong site then continued.

Google was due to have its licence to operate in China renewed on June 30, 2010. When the day came, Google restored its China search engine complete with censorship. They did however include a link to their Hong Kong site. Both Google and the Chinese Government have claimed victory. Complete information is not currently (2011) available, however it would appear Google backed down under pressure that it would be thrown out of China.

GROUP TASK Research
China is not the only country to censor the internet. Research and summarise internet censorship occurring in different countries.

Napster versus Metallica (and RIAA)

Napster was a Peer-to-Peer (P2P) file sharing internet service that allowed people to share music without having to purchase an original CD. Napster's application, once downloaded allowed users to access and download songs that are stored on other users hard drives. Such was the success of Napster that in the first few months, it was estimated that more than 85 million people around the world used the service.

In 1999, the Recording Industry Association of America (RIAA) filed a lawsuit on behalf of five major US record labels alleging Napster violated both Federal and State copyright laws and stated that *"Napster knowingly and wilfully set out to build a business based on copyright infringement on an unprecedented scale."* and in 2000 the heavy metal band Metallica files a similar lawsuit against Napster.

Napster during their defence argued that they are simply an Internet Service Provider but are also protected by the 1992 Home Recording Act which allows the general public to make a copy of music for personal use. The judge disagreed saying the service did violate copyright and issued an injunction to have the site shutdown. The US government also states that Napster's service falls outside of the definitions of the Home Recording Act.

Just hours before the site is due to be shut down the US Circuit Court of Appeals issues a stay of the injunction until Napster's full appeal can be heard. During the appeal Napster's defence team insist that it is the end-users that are in violation of copyright and not Napster, saying the company only provides the means for users to trade music, which in itself, is not an infringement of copyright.

In September of 2001, Napster reaches a partial settlement of the lawsuits filed against it and agrees to pay US$26 million to some music publishers and songwriters. Napster also states that it intends to become a subscription-based service with a percentage of royalties being distributed to the copyright owners of the music.

Agreement with the major record labels could not be reached however and in 2002 Napster filed for Chapter 11 bankruptcy.

GROUP TASK Research
Peer-to-Peer (P2P) software is still in widespread use. Research to determine its operation and legal status.

CHAPTER 1 REVIEW

1. Software developers need to achieve:
    (A) a respect for the marketing departments in software companies.
    (B) a balance between their own rights and their responsibilities to their clients.
    (C) a relationship between their copyright and royalty rights.
    (D) strong personal relationships with clients.

2. Which of the following classifications of software cannot be copied and distributed?
    (A) Commercial
    (B) Open Source
    (C) Shareware
    (D) Public Domain

3. Responsible software developers will ensure:
    (A) their products do not influence the execution of other installed applications.
    (B) they make copies of competitor's products.
    (C) they make changes to any modules of code they use from other products.
    (D) All of the above.

4. Owning a software licence usually means:
    (A) you can freely copy, modify and distribute the product.
    (B) you can use the software and make one copy for archival purposes.
    (C) there is no need to enter into a contract with the copyright owner.
    (D) the product can be installed on any machines that you own.

5. Security measures to combat software piracy include all the following EXCEPT?
    (A) encrytion
    (B) registration coes
    (C) back-to-base authentication
    (D) heat sensitive ink

6. A set of standards to which professional software developers agree is called:
    (A) a Code of Conduct.
    (B) Quality Assurance.
    (C) Beta testing.
    (D) a Contract.

7. Copying software and then selling it under the guise of being the original is known as:
    (A) plagiarism.
    (B) piracy.
    (C) counterfeiting.
    (D) theft.

8. Operating system developers:
    (A) must maintain their market share at all costs.
    (B) should introduce completely new products as new technologies arrive.
    (C) have a responsibility to ensure their products are well documented and reliable, as software must interface with these products.
    (D) cannot be expected to consult with application developers when designing new products.

9. When a runtime error occurs:
    (A) the program should display a meaningful error message to assist in isolating the problem.
    (B) well written applications will do their best to save any data before execution halts.
    (C) error routines should attempt to isolate and correct the problem so that execution can continue.
    (D) All of the above.

10. Support for software products is the responsibility of software developers. Software support includes all of the following EXCEPT:
    (A) Isolating problems.
    (B) Creating bug fixes.
    (C) Updating.
    (D) Training.

11. Define the terms Intellectual property, copyright, and software licence agreement. Explain the relationship between each of these terms in regard to software products.

12. As a software developer, you have ethical responsibilities to uphold for the good of the industry. Describe these ethical responsibilities. Include areas related to the quality and reliability of the product, upholding the intellectual property rights of others, and software marketing in your response.

13. A school wishes to purchase a multimedia CD-ROM product. The teacher purchases one copy and then installs the product on the school's network where it can be used by anyone with network access. Discuss the legality of this situation in terms of copyright.

14. A group of software developers are creating products to automate tasks within the Real Estate industry. These software developers form an association to help maintain standards within their particular section of the market. Design a code of conduct for this association.

15. A software developer creates a new product. This product accepts object code as its input. The product analyses this code and outputs a listing similar to a high-level language. The software developer decides to market the product as 'Shareware'. Discuss the possible social and ethical issues involved in this scenario in terms of the developer and the users.