COMPUTER ETHICS: CAUTIONARY TALES AND ETHICAL DILEMMAS IN COMPUTING

By Tom Forester[1] and Perry Morrison.[2]
Cambridge, Massachusetts: The MIT Press. 1990. Pp. 193.
$19.95.

> Every new technology carries with it an opportunity to invent
> a new crime (p. 10).[3]

Technological change penetrates society faster than we can form new attitudes, reach new consensuses, or adapt our legal and ethical codes. Adaptation must occur if we are to cope adequately with the new problems — or to recognize old problems in new garb — that the technologies bring. Since the pace of technological advance in the area of computers is unlikely to abate, our ability to understand, discuss, and decide computer issues must leapfrog forward if we are to live rationally and peacefully with our silicon neighbors.

Forester and Morrison's book is an attempt to spur this sort of discoursc. They survey the gamut of computer-engendered problems: health risks, invasions of privacy, confusions over ownership and authorship, alterations of workplace mores, crimes, uncertainties of liability, and crises in national defense. The list of issues is long, and virtually all our societal institutions are implicated. In demarcating the playing field, the authors raise many provocative questions, and suggest the shape and scope of debates of coming years.

Several topics will be of particular concern to those with an interest in the near-term legal ramifications of computerization. These include discussions of hacking and computer crime, privacy, expert systems, liability for faulty programs, and software ownership and piracy. Since both authors are Australian, their legal orientation tends toward Australian and British law. They explicitly consider few American statutes or policies. The book is not intended as an introduction to computer law, much

[1] Forester teaches in the School of Computing and Information Technology, Griffith University, Queensland, Australia.

[2] Morrison is Lecturer in Computing at the University of New England, New South Wales, Australia.

[3] Comment of Laurence A. Urgenson, Chief Assistant U.S. Attorney for the Eastern District of New York, on a case in which New Yorkers defrauded a local cellular phone company by reprogramming memory chips in their mobile phones in order to make free calls. Buder, *18 Are Seized in Illegal Use of Mobile Phones*, N.Y. Times, Mar. 27, 1987, at A1, col. 3.

less American computer law. Irrespective of jurisdiction, the authors are adept at pointing out gray areas of the law, places where traditional doctrine stretches into flawed analogies inadequate to new conflicts and dilemmas. The book, then, is most valuable as a general guide to brave new frontiers in the law of computers, communication, and information.

Perhaps the grayest area that the authors explore is that of intellectual property rights for that "wholly new kind of entity," computer software (p. 31). The authors evoke the foggy nature of software: not quite machine, not quite process, not quite literary work. Because of this indeterminacy, it comes as no surprise that "the law on intellectual property as it applies to computer software is in a mess" (p. 33). No analogy from traditional doctrine is complete, and all have serious defects from either a mechanical- or policy-based perspective, or both, so that "we are not sure whether copyright, patents or trade secrets apply or should apply to this strange new thing called software."[4] "Copyright law does not wholly protect a program," and the generalization of copyright "could become a serious and costly obstacle to standardizing software applications" (p. 33). Patent law offers more protection, but could grind the software industry to a halt.[5] The inherently leaky nature of information cuts against use of trade secret law, which is, in any case, "in contradiction to the notion of the widest possible dissemination of innovations and would make marketing a program . . . virtually impossible" (p. 33). Of course, many suggestions have been made to help reduce the confusion, ranging from modifications of traditional mechanisms to Professor Paul Marett's[6] proposed development of a new field of "informatics" law (p. 35). The point is that software law awaits new social consensuses and requires new legal approaches.

Other ethical ambiguities and inchoate laws arise from the act of breaking into computer systems, the art of "hacking." The ontological status of hacking is as uncertain as that of software. Part of the uncertainty results from the diversity of motivations for breaking into computers, including outright theft of funds, "theft" of information, theft of computing resources,[7] vengeance against an employer by a disgruntled

[4] The authors conclude that, as a result of the legal system's inability to form new doctrines fast enough, confusion spreads and "the gap between legal precedent and everyday behaviour on the part of computer professionals and users grows . . . wider" (p. 31).

[5] For a discussion of the move toward software patents and their detrimental effects on innovation in the software industry, see Kahin, *The Software Patent Crisis*, TECH. REV., Apr. 1990, at 52.

[6] Loughborough University, UK.

[7] Early "hackers" specialized in breaking into the telephone company network. The most famous of these "phone phreaks" is John T. Draper, known as "Captain Crunch," who discovered that a toy plastic whistle supplied in a breakfast cereal box emitted a tone which gained access to toll-free phone services (p. 42). For a chronicle of hacking, see S. LEVY, HACKERS: HEROES OF THE COMPUTER REVOLUTION (1984).

employee,[8] vengeance against society, espionage, vandalism, economic competition, curiosity, intellectual challenge, and mere mischief.[9] Crime on a new frontier clearly presents new questions of culpability, intent, and liability. Criminal intent and premeditation are particularly difficult to assess when, as is almost always the case, the hacker fails to come into physical proximity to the scene of the crime.

Where theft is the motivation, difficult questions arise as to the nature and value of the items removed. "When a file has been copied or selectively viewed, what has been stolen?" (p. 60). Or when a hacker breaks into a commercial database to correct information about himself, what crime, if any, is committed? Who owns such personal data? (p. 62). Is it the person who is the subject matter of the information, or is it the company that has paid to assemble it? Can such an act be termed theft?[10] What about a hacker who simply "walks through" a system without disturbing anything? (p. 60). Only recently have courts and legislatures begun to address such issues.[11]

When saboteurs broke into California congressman Ed Zschau's Capitol Hill computer system and destroyed his records and mailing lists in 1986, the police could only recommend better controls for the future. Outraged, he said, "The entering of my computer was tantamount to someone breaking into my office, taking my files and burning them.... Because people don't see the files overturned or a pile of ashes outside the door, it doesn't seem as bad.... But it is equally devastating."[12] Yet others would hesitate to apply traditional criminal doctrine to electronic break-ins. The authors also suggest that hackers may be socially useful because they improve computer security, counteract a dangerous tendency toward centralization of information, and even help foil terrorist attacks.

[8] Keith Hearnden's study found this "battlezone theory" to be the most prevalent reason of all. Four-fifths of computer crimes were found to be carried out by employees rather than outsiders, particularly by clerks and cashiers (p. 19) (Hearnden, *Computer Criminals Are Human Too*, in COMPUTERS IN THE HUMAN CONTEXT 415–42 (T. Forester ed. 1989)).

[9] For a discussion of the motivations for computer crimes, see BloomBecker, *Introduction to Computer Crime*, in COMPUTER SECURITY (J. Finch & E. Dougall eds. 1984).

[10] This debate frequently arises in the context of credit-rating agency databases. Most recently the same issues were focused by Lotus Company's attempt to introduce its consumer database, Marketplace Households, said to contain data on 120 million Americans. The project was withdrawn in response to public pressure. *See New Data Base Ended by Lotus and Equifax*, N.Y. Times, Jan. 24, 1991, at D4, col. 1.

[11] Massachusetts has been in the vanguard with recently proposed controversial legislation that sought a rational approach to computer crime. *See* An Act to Prevent Computer Crime, Mass. S. 1543 (1990). *See* Kay, *Computer-Crime Law Could Be a Model*, MACWEEK, Dec. 4, 1990, at 36.

[12] *Two Cases of Computer Burglary*, N.Y.Times, Mar. 21, 1986, at B4, col. 4.

The authors argue that "the legal basis of system break-ins languishes in the dark ages of real locks and doors and physical forms of information such as blue prints and contracts" (p. 60). A modem and password are not the same as a padlock, and "the highly mutable forms of information that computer files represent" are not the same as information in paper form (p. 60). The analogies are weak, doctrines are lacking, and new conceptualizations have not kept pace with technological developments.

As in other fields where computers introduce new methods and standards, the application of existing legal frameworks to expert system is problematic. An expert system emulates the informed decision-making processes of human specialists through computer software. While expert systems are still relatively crude, and rare on the commercial scene, they portend a host of legal problems if used for anticipated applications such as medical diagnosis, legal advice, structural design, and manufacturing control. The authors pose a hypothetical case where a doctor uses an expert system containing the codified knowledge of experts in his field. Due to a flaw in the program, the patient dies (p. 123). Who is liable? The doctor who provided the treatment? The programmer? The software company? Is the system a product or a service? One suggestion is that "sources" of expert knowledge require software companies to indemnify them against the experts' own errors![13] It may be difficult for doctors simply to muddle through. Possibly, as standards evolve, a doctor would be found negligent for *failure* to consult an expert system (p. 123). Uncertainty and fear as to what their potential liability might be has made companies more cautious in their expert-system ventures. The field is new and nobody wants to be a test case (p. 122).

Since expert systems are embryonic, the discussion of expert systems remains speculative. To get around this problem, Forester and Morrison resort to a device they employ throughout the book, a fictional scenario or parable that attempts to embody future ethical problems. They look a decade into the future and envision an overburdened judiciary using expert systems to take over the job of sentencing in order to reduce the workload of judges.

The system is based on a database of benchmark cases for which a number of judges have suggested sentences. When a criminal is convicted, the computer automatically matches his case to the nearest analogous case in the database, and imposes the mean sentence calculated

[13] R. Lucash, *Legal Liability for Malfunction and Misuse of Expert Systems*, 18 SIGCHI BULL. 39 (1986).

from its "conscience" (p. 134). In addition to saving time, the system would avoid the "one-person-lottery" of current sentencing. Yet it would also create many concerns. Would we standardize defects and freeze evolution in the area of sentencing? Would we "hard-wire" class, sex, race, and other biases? Would the system too frequently miss key nuances of a case, miss the human texture and context that a judge might consider? More generally, assuming we can implement such a system, would we really have done anything to increase fairness and justice? The intuition in the last question points to a more general problem in the philosophy of technology: Should we always adopt a technical solution where one may exist? That is, when is an "improvement" merely a "technofix," an attempt "to concoct a superficial technological solution to what is essentially a human problem?" (p. 136).[14] The fear is that we will implement a system with significant human consequences before we have thought through what the consequences will be, before we understand its implications, and before we obtain the "consent" of those whose lives will be affected.

If the major theme of *Computer Ethics* is that the development of ethical and legal understanding lags behind the bold and swift computerization of society, then the minor theme is that of our misplaced faith in computers. The authors have a rich sense of the fickle, fragile nature of digital-based computer systems, which lack the depth and resilience of analog-based or human-based systems. A digital system characteristically fails all at once, catastrophically, or behaves aberrantly with no warning. In an analog system, such as a thermostat, there are few discontinuities, and total failure is rare. In a digital system, by contrast, each state is dependent on its predecessor, so that a failure at any point can be fatal: digital systems fail in a large variety of ways (p. 81).

Analog is concrete, tangible. Engineers historically have been able to design in a "fudge factor" or margin of safety for an analog device, so that errors within some tolerable range do not cause a system failure. This builds in robustness. Unfortunately, digital technology's abstract nature is far less conducive to engineering against error: "To a large degree, the behaviour of complex systems is at the outer edge of our intellectual understanding, so that our ability to know or predict all the possible states (including error states) that a system might take is severely restricted" (p. 77). The result is that, "[a]s digital technology infiltrates almost all aspects of our lives ... our involvement in

[14] The technofix critique is expanded in Morrison, *Limits to Technocratic Consciousness: Information Technology and Terrorism as Example*, 11 SCI., TECH., & HUM. VALUES 4 (1986).

mystifying technological failure becomes more common" (p. 74).

Examples are rife, as a small sample shows: the Audi 5000 that, due to a flawed electronic idle control, is given to sudden uncontrollable surges of acceleration (p. 74); the presidential 747 airplane that closes electronically-controlled garage doors every time it operates out of an air-base in California (p. 73);[15] the man who, because a computer database registered him as dead, was unable to cash checks, receive social security payments, or process medical claims for a period of months (pp. 75–76); the angry *Encyclopedia Britannica* employee who entered the database for the new edition and changed "Jesus Christ" references to "Allah" (p. 3);[16] the squirrel that wandered into the NASDAQ computer, shutting down the system and halting stock trading for more than an hour (p. 3);[17] the guidance system that causes fighter planes to fly inverted whenever they cross the equator (p. 73); and the video pirate who overrode a Chicago television broadcast for ninety seconds with a transmission of a man in a Max Headroom mask smacking his exposed buttocks with a fly swatter (p. 41).[18]

One consequence of the "bugginess" of software and the glitch-prone nature of digital systems is the failure of software developers to provide significant warranties for their products. One firm warranted only that "the diskette(s) on which the program is furnished . . . [will] be of black color and square shape under normal use for a period of ninety (90) days from the date of purchase" (p. 76). Because of a propensity toward precipitous error, digital systems require new thinking about responsibility and liability for products. The resounding question about expert systems applies as well to inexpert devices: Who is to blame when things go wrong?

> [A]re programmers unobligated in the event of a substantial system failure? If such a system (say a robot) kills someone, is the programmer a murderer? If a patient dies on an operating table because software running the life support equipment fails, is the programmer guilty of manslaughter or malpractice? Is he or she excused if they [sic] provide a disclaimer or inform the surgeon of potential configurations that could cause problems? Or is the programmer guilty simply because they

[15] *See* SOFTWARE ENGINEERING 7 (April 1986).

[16] *See Laid-off Worker Sabotages Encyclopedia*, San Jose Mercury News, Sept. 5, 1986 (cited in SOFTWARE ENGINEERING 28 (October 1986)).

[17] *See Stray Rodent Halts NASDAQ Computers*, N.Y. Times, Dec. 10, 1987, at D21, col. 1.

[18] *See* 13 SOFTWARE ENGINEERING NOTES 7 (1988).

> provided a system that (both theoretically and practically) could not be guaranteed for application in a life-critical situation? After all, if a manufacturer of heart pacemakers knowingly supplied defective equipment, surely [it] would be required to answer in court! (p. 79).

Computer Ethics leaves us with many questions and few answers. The book sets out to highlight problems we face in our relationship to computer, communication, and information technologies. It does a good job of using well-drawn examples and closing each chapter with entertaining hypothetical scenarios. However, the book lacks significant assessments of our various institutional attempts to deal with many of the issues raised. Moreover, among the few statutes and legal cases mentioned, the American system is underrepresented. I sense that the authors found present solutions so inadequate that they were unwilling to consider them at much length. Reading the book is similar to being part of brainstorming conversation in which the agenda is set for more deliberation. Rather than attempting to lay down policy, the book emphasizes the need for a great deal more discussion and participation in a subject toward which the general population is too complacent.

Steven Bercu