

Zur gesellschaftlichen Bedeutung der Informatik
Eike Best, 14.12.1988

Sehr geehrter Herr Dekan, meine Damen und Herren!

Ich danke für die Gelegenheit, einige Gedanken zu einem Thema vortragen zu können, das wie kaum ein anderes in der öffentlichen Dauer-Diskussion steht und meine Wissenschaft, die Informatik, betrifft:

- Stellt der durch die sogenannte ‘elektronische Revolution’ herbeigeführte Wandel in unserer Gesellschaft eher eine erhebliche Gefahr dar oder liegt in ihm eine große Chance für eine bessere Zukunft?

Diese Frage ist Kristallisationspunkt für eine ganze Reihe weiterer Fragen, nicht zuletzt der nach dem Selbstverständnis der Wissenschaft Informatik.

Wenn Sie mir zunächst eine persönliche Bemerkung gestatten: Ich habe mich regelrecht zwingen müssen, mir diese Fragen zu stellen und meine eigene Position zu klären, denn es wäre um sovieles bequemer gewesen, einen Vortrag über meine wissenschaftlichen Hobbies zu geben. Ich habe mich deswegen dazu gezwungen, weil ich die Wichtigkeit der Fragen sehe und es für verkehrt halte, ihre Beantwortung Politikern zu überlassen.

Noch vor 20 Jahren gab es keine deutsche Universität, an der Informatik als Hauptfach gelehrt wurde. Heute hat sich dieses Bild gewandelt (in München sind z.B. 10 Prozent aller Studienanfänger Informatikstudenten), was die Tatsache widerspiegelt, daß die elektronische Datenverarbeitung in fast alle Bereiche des gesellschaftlichen Lebens gründlich eingegriffen hat, und umgekehrt, daß in fast allen Bereichen ein Bedarf entstanden ist, der durch die Möglichkeiten der Datenverarbeitung gedeckt werden konnte und ihre Entwicklung mitbeeinflußt hat.

Ich möchte meinen Vortrag in zwei Teile gliedern.

- (1) Im ersten Teil werde ich einige Beispiele aufzählen, um Ihnen einen Eindruck von der Verschiedenartigkeit und der Universalität des Einsatzes von Datenverarbeitung zu vermitteln. Anhand dieser Beispiele möchte ich Ihnen gleichzeitig einige der konkreten Gefahren verdeutlichen, die der Einsatz von EDV mit sich bringen kann.

- (2) Im zweiten Teil möchte ich einige Besonderheiten der Wissenschaft Informatik, auch im Vergleich zu anderen Wissenschaften, herausarbeiten. Daraus will ich dann Schlußfolgerungen im Hinblick auf das Thema meines Vortrags ziehen.

TEIL I)

In den vergangenen Jahren war ich in der theoretischen Forschung tätig und möchte zunächst kurz erzählen, woraus mein persönlicher Bedarf an EDV bestand. Da ich annehme, daß meine Zuhörerschaft zum Teil aus Nicht-Sachverständigen besteht, werde ich während meines Vortrags auch auf die Funktionsweise von Computern in hoffentlich allgemeinverständlicher Weise eingehen.

Mein persönlicher EDV-Bedarf erschöpft sich momentan in Texterstellung und Kommunikation. Texterstellung sieht so aus, daß ich meine Arbeiten, also englische oder deutsche mathematische Texte, in einer Codesprache in einen Computer, der auf meinem Schreibtisch steht, eintippe. Ein Programm kann diese Codesprache verstehen und meinen Text als qualitativ hochwertige Druckseiten auf einem Druckgerät ausgeben. Das Programm kann z.B. eine mathematische Formel aus meinem eingegebenen Code erzeugen, aber es kann natürlich nicht die Richtigkeit dieser Formel nachprüfen, sodaß es mir die Verantwortung für den Inhalt meiner Arbeiten nicht abnimmt. Trotzdem komme ich ohne diese Hilfsmittel der Texterstellung kaum mehr aus, bin abhängig davon geworden. Diese Abhängigkeit empfinde ich uneingeschränkt als positiv, denn außer in trivialen Ärgernissen (Druckerausfall gerade dann, wenn es eilt usw.) sehe ich in der Benutzung von Textsystemen keine große Gefahr.

Etwas anders sieht es bei der Kommunikation aus, die ich per Computer betreibe. Dazu muß ich einiges über unsere Arbeitsweise vorausschicken. Wir stehen natürlich in Verbindung mit Kollegen an anderen Forschungsstellen, auch im Ausland. Bis auf einige wenige sehr theoretisch eingestellte Kollegen, die sich standhaft weigern, Hand an eine Maschine zu legen, hat mittlerweile jeder von uns eine Benutzernummer auf einem der Rechner seines Instituts.

Diese Maschinen haben einen Namen und eine Adresse, die sie international eindeutig identifizieren, ähnlich Ihrer Privatadresse, sofern Sie eine haben. Post- und private Kabel sind inzwischen längst so durchgeschaltet, daß es möglich ist, von jeder beliebigen ans Netz angeschlossenen Maschine zu jeder beliebigen anderen solchen Maschine zu gelangen, von wenigen Ausnahmen abgesehen.

Konkret sieht das so aus, daß ich auf unserer Maschine ein Programm namens *send-mail* aufrufen kann, dem ich als Parameter den Namen der Zielmaschine, die Benutzernummer des Empfängers auf dieser Maschine sowie die zu übertragenden Daten geben kann. Die elektronische Post nimmt ihren

Weg über eine unbekannte Anzahl von Zentralknoten, d.h. Maschinen, die als Sammel- und Verteilerstellen fungieren. (Nebenbei gesagt, wäre es ein leichtes, meine und überhaupt alle Privatpost auf einem der Zentralknoten zu speichern und auszuwerten.) Diese Art des Briefeschreibens bietet außer stark erhöhter Geschwindigkeit (bei moderaten Kosten) auch noch weitere Vorteile wie z.B. den, daß der Quellcode einer gemeinsamen Arbeit ohne große Umstände zwischen den Autoren hin- und hergeschickt werden kann. Aus dieser Beschreibung ist schwer ersichtlich, welche Gefahren die Kommunikation per elektronischer Post bergen kann. Und doch beruht der sogenannte Computer-Wurm, der Anfang November viele Computerzentren in den Vereinigten Staaten lahmlegte – eine Tatsache, die in der Weltpresse großes Aufsehen erregt hat – unter anderem auf einer Lücke in dem unscheinbaren Programm *send-mail*. Ich will nun auf dieses Vorkommnis etwas näher eingehen und tue dies aus zwei Gründen: Einmal wegen der großen Publizität des Ereignisses, und dann auch weil es typisch ist für die Probleme, die in Bezug auf die Beherrschbarkeit großer und verteilter Computersysteme auftreten können.

Um annähernd die Funktionsweise dieses Wurms zu erklären, muß ich ein wenig ausholen und einiges über die Interna der betroffenen Maschinen erzählen. Es handelt sich um Mehrbenutzermaschinen. Das bedeutet, daß die Anforderungen vieler Benutzer quasi-gleichzeitig bearbeitet werden können. Die Maschine ist vergleichbar einem dauernd beschäftigten Koch, der für viele im Speisezimmer befindlichen Gäste (die Benutzer in ihren Büros) deren Speisewünsche gemäß den jeweiligen Rezepten erfüllt. Während die Kartoffeln für den einen Gast gerade garen, kann der Koch mittlerweile den Nachtisch für einen anderen Gast zubereiten. Genauso springt die Maschine von der Bedienung eines Benutzers zum nächsten Benutzer. Es gibt eine weitere Kategorie von sogenannten Systemprogrammen für allgemeine Verwaltungsarbeiten, statistische Auswertungen und ähnliches.

Natürlich geht es nicht an, daß der Gast, der den Nachtisch bestellt hat, dem Koch aus einer Laune heraus aufträgt, die Kartoffeln des anderen Gastes vom Feuer zu nehmen. Genausowenig dürfen ein durchschnittlicher Benutzer des Systems oder ein normales Systemprogramm einander oder andere Benutzer stören, weswegen in Mehrbenutzersystemen stets ein Protektionsmechanismus eingebaut ist. Es gibt allerdings einen speziellen Benutzer namens *super-user*, vergleichbar vielleicht mit dem Inhaber der Gaststätte, der genau das tun darf. Er darf die Systemdaten sowie alle Benutzerdaten lesen und schlimmstenfalls auch zerstören.

Jedes Computerprogramm ist ein intellektuelles Produkt von großer bis sehr großer Komplexität; hierauf werde ich später noch näher eingehen. Es ist leider allzu leicht, falsch zu programmieren. Deshalb muß jedes Programm idealerweise verifiziert, mindestens aber ausgiebig getestet werden. Ein Programmierer tut gut daran, sich selbst zu mißtrauen und den Fall vorherzuse-

hen, daß sein Programm sich einmal anders verhält als von ihm gewünscht. Der Autor des Programms *send-mail* hat aus diesem Grund zunächst eine Test-Version seines Programms erstellt, die es ihm im Fehlerfall ermöglichen sollte, schnell die Quelle des Fehlers zu entdecken. Als einfachste, wenn auch etwas brutale Methode hat er die Möglichkeit eingeplant, im Fehlerfall einen *super-user*-Prozeß zu starten und damit schnell Zugriff auf alle zur Fehlersuche benötigten Daten zu haben.

Diese Testversion des Programms *send-mail* lief aber offenbar doch so gut, daß sie (und nicht eine endgültige Version) in den betroffenen Installationen übernommen wurde.

Der Programmierer des sogenannten ‘Wurm’s hat nun herausgefunden, was es mit *send-mail* auf sich hat und hat sich das folgendermaßen zunutze gemacht. Der Wurm wurde ursprünglich auf einer allgemein zugänglichen Maschine gestartet. Zunächst hat er die Liste aller durch *send-mail* erreichbaren Maschinen durchsucht und hat sich dann selbst wie eine Art Serienbrief auf alle diese Maschinen geschickt. Ein Programm ist nämlich, bevor es übersetzt und ausgeführt werden kann, ein ganz normaler Text. Als Text ist es unschädlich. Nun wurde der *super-user*-Status ausgenutzt (wie genau, ist noch nicht öffentlich geworden), so daß auf den Zielmaschinen der Wurm übersetzt und in die Liste der auszuführenden Programme als Systemprogramm getarnt aufgenommen wurde. Außerdem hat sich der Wurm auf jeder Maschine in periodischen Abständen repliziert, was die Maschinen stark verlangsamt und zu seiner Entdeckung führte.

Dieser Wurm – der Name erklärt sich aus seiner Funktionsweise – hat vor allen Dingen durch das, was er **nicht** tat, Aufsehen erregt. Er hätte offenbar ohne weiteres die Programme und Daten der Zielmaschinen (unter ihnen mehrere geheime Einrichtungen der Armee) zerstören können oder auch weniger offensichtliche Effekte wie das heimliche Einschleusen von Spionageprogrammen haben können; der Phantasie sind kaum Grenzen gesetzt.

Über diesen Vorfall wird in Bälde ein Bericht veröffentlicht werden. Die beiden wesentlichen Schlußfolgerungen des Berichts sind vorab bekannt geworden:

- (1) Wir benötigen bessere Programme und bessere Programmierer.
- (2) Die Informatik-Ausbildung muß eine Ethik-Ausbildung mit einschließen.

Ich habe diese Geschichte deswegen so ausführlich erzählt, um auf die weite Kluft zwischen Machbarkeit und Beherrschbarkeit von Computersystemen hinzuweisen. Das Programm *send-mail* ist nicht außergewöhnlich kompliziert; es ist ein Programm, wie es Hunderte im Betriebsumfang jedes mittleren Computersystems gibt. Und doch genügte die kleinste Lücke, um potentiell ungeheure Wirkungen hervorzurufen.

Es gibt eine große Anzahl dokumentierter Fehlerfälle, von denen viele auf die unbezwungene Komplexität eines Systems zurückzuführen sind. Programmfehler sind in der Vergangenheit schon für die Verschwendung von Zeit und Geld und auch für den Verlust menschlichen Lebens verantwortlich gewesen. Vor den Gefahren der Diskrepanz zwischen Machbarkeit und Beherrschbarkeit ist nicht nur in der Informatik eindringlich gewarnt worden. Auch die Produktion von Hunderten neuer Chemikalien, darunter viele Giftstoffe, pro Tag ist ja eine machbare, aber auf absehbare Zeit nicht beherrschbare Technologie.

Beherrschbarkeit ist ein wichtiger Gesichtspunkt bei der Beurteilung der gesellschaftlichen Bedeutung der Datenverarbeitung, aber es ist nicht der einzige.

Der vielleicht am tiefsten im öffentlichen Bewußtsein verankerte gesellschaftliche Aspekt der EDV ist der über Rationalisierungsmaßnahmen bewirkte Arbeitsplatzverlust in vielen industriellen Bereichen. Die vollautomatische Fabrikhalle und das halbautomatisierte Büro sind längst keine Zukunftsvisionen mehr. Man kann diese Vorgänge auf zweierlei Weisen betrachten, abstrakt und unabhängig von der Gegenwart, bzw. konkret und eingebettet in unsere Gesellschaft.

Abstrakt geht es bei Automatisierungen des Produktionsprozesses darum, menschliche Arbeit durch maschinelle zu ersetzen. Die EDV spielt hier eine besondere Rolle, indem sie nämlich einen Teil derjenigen Arbeitsprozesse zu automatisieren gestattet, die mit dem Begriff 'Kopfarbeit' umschrieben werden können. Auch in der Entwicklung kann die EDV auf sinnvolle Weise eingesetzt werden, etwa durch die Simulation neuer Verfahren im Rechner, die mechanisch nur auf aufwendige und umweltschädigende Weise erprobt werden könnten.

Abstrakt gesehen gibt es über die Rolle der EDV im Produktionsprozeß sicherlich viel Positives zu sagen. Leider beruhen Rationalisierungsmaßnahmen normalerweise nicht auf Menschenfreundlichkeit, sondern auf konkreten Rechnungen: Lohnkosten gegen Maschineninvestitions- und Wartungskosten. Fallen dabei Arbeitsplätze weg, dann ist das Pech für die Betroffenen, die schlimmstenfalls arbeitslos werden. Insofern ist, was längerfristig eine Arbeitserleichterung für alle bedeuten könnte, oft mit konkreten Härten für die Betroffenen verbunden.

Ich möchte im Rahmen meines Vortrags auf die sehr reale Problematik der Arbeitslosigkeit durch Rationalisierung infolge EDV-Einsatzes nicht näher eingehen, außer auf eine in der Diskussion dieser Frage enthaltene Gefahr aufmerksam zu machen. Es ist leicht und wird manchmal versucht, dem abstrakten Begriff des technischen Fortschritts, der EDV insgesamt oder den am sogenannten Fortschritt beteiligten Wissenschaftlern die Schuld für Arbeitslosigkeit und zunehmenden Druck im Arbeitsleben zuzuschieben. Dabei

wird aber vergessen, daß vor so gut wie jeder konkreten Rationalisierungsmaßnahme in einem Betrieb eine Management-Entscheidung steht. Hier wäre in erster Linie die Verantwortung für resultierende Arbeitslosigkeit zu suchen.

Ich möchte noch auf eine weitere Art der Anwendung von EDV in Betrieben zu sprechen kommen, die nicht unmittelbar mit Rationalisierungen zu tun hat. Die Rede soll kurz von sogenannten Management-Informationssystemen und Personal-Informationssystemen sein. Erstere enthalten unter anderem Daten über die Struktur des Betriebs und die Abschlüsse der einzelnen Betriebsteile. Sie erleichtern die Möglichkeit, den Betrieb umzuorganisieren und deswegen, mittelbar, auch Rationalisierungen. Personal-Informationssysteme enthalten persönliche Daten, zum Teil sehr privater Natur, zum Teil das Verhalten im Betrieb betreffend, der Beschäftigten. Sie ermöglichen es, nach verschiedenen Kriterien Personalentscheidungen durchzuspielen und letztendlich zu treffen.

Für diese Systemarten ist es charakteristisch, daß wichtige Informationen in einseitiger Richtung fließen, nämlich vom Betrieb als ganzem zur Geschäftsleitung. Personal-Informationssysteme und Management-Informationssysteme bewirken daher eine Konzentration von Macht in wenigen Händen in einem Ausmaß, wie es besonders in großen multinationalen Konzernen nötig ist.

Nur im Vorübergehen sei angemerkt, daß die EDV im staatlichen und polizeilichen Einsatz oft eine ähnliche Funktion erfüllt. Auch hier ist die Richtung des Informationsflusses einseitig: hin zu den Zentren der Macht.

Um den ersten Teil meines Vortrags und meine Beispielsammlung abzuschließen, möchte ich noch kurz auf militärische Anwendungen der EDV eingehen. Die Diskussion dieses Themas halte ich trotz neuester Bewegungen in Richtung Abrüstung für aktuell, sie leidet aber besonders stark unter den Voreingenommenheiten festgefügtter Freund- und Feindbilder. Ich bitte Sie trotzdem, mir in einem Versuch zu folgen, zu belegen, daß die besondere Rolle der Mikroelektronik in neuen Waffensystemen zwei spezielle Gefahrenmomente mit sich bringt.

Es ist üblich, die Kräfteverhältnisse in der Welt vereinfachend in West und Ost einzuteilen. Bis vor einigen Jahren war es akzeptierte Doktrin, daß zwischen beiden Blöcken ein ungefähres militärisches Gleichgewicht herrsche; was ein Mehr von Ost an landgestütztem Potential darstelle, werde auf West's Seite durch ein Plus an see- und luftgestütztem Material sowie durch eine höher entwickelte Technologie ausgeglichen. Das sogenannte Gleichgewicht des Schreckens besagte, daß der Einsatz von Atomwaffen unbedingt das Ende der Zivilisation bedeuten würde und daß deshalb sowohl West als auch Ost vor einem Ersteinsatz zurückschrecken würden, was dann logischerweise bedeuten würde, daß solche Waffen nie zum Einsatz kommen

könnten.

Einige Stichworte neuerer militärtechnischer Erfindungen sind: Integrierte Warnsysteme; Miniaturisierung; Zielgenauigkeit; Flexibilität. Alle diese Charakteristika sind mit Hilfe des massiven Einsatzes von EDV erreicht worden oder erreichbar geworden.

Ein integriertes Warnsystem ist zum Beispiel geplant als Vernetzung von unterseeischen Überwachungssystemen über die bereits existierenden automatisierten Radarsysteme mit eventuell im Weltraum zu stationierenden Warnsystemen und den davon abhängigen Abwehrwaffensystemen. Ein verteiltes System dieser Größenordnung ist zur Zeit und auf absehbare Zeit außerhalb jeder Beherrschbarkeit. Und daraus ergibt sich das erste große, mit EDV verbundene Gefahrenmoment: die Gefahr eines unerklärten 'Krieges aus Versehen' durch Fehlfunktionen der EDV-Anlagen. Es gibt ausführliche Dokumentationen von mindestens zehn Fällen in den letzten zehn Jahren, in denen ein Krieg nur noch nach Minuten bemessen an uns vorübergegangen ist. Die Dunkelziffer wird auf ein Vielfaches davon geschätzt.

Miniaturisierung, Zielgenauigkeit und Flexibilität betreffen einzelne Raketen, nicht ein globales System. Es gibt Flugkörper, deren Abschluß- und Einschlagstellen kurz vor dem Start einprogrammiert werden können. Aufgrund mitgeführter Landkarten wird die Einschlagstelle mit großer Genauigkeit getroffen. Es können sehr 'kleine' Atombomben oder auch chemische Bomben mitgeführt werden, die z.B. unterirdisch (in der Nähe einer Kommandozentrale) gezündet werden können und dann wahrscheinlich sogar nur relativ wenige Menschenleben kosten.

Das zweite große Gefahrenmoment in dieser Entwicklung liegt darin, daß die Öffentlichkeit sich ganz allmählich daran gewöhnen könnte, Atomwaffen und chemischen Waffen als 'ganz normale Munition' (soweit man davon sprechen kann) anzusehen, deren Einsatz in einem eventuellen Krieg möglich ist, ohne daß dieser Krieg gleich zur Vernichtung der Zivilisation führt. In der Tat gibt es bereits offizielle Dokumente, die von einem gewinnbaren Krieg ausgehen, in dem zu bestimmten Zeitpunkten auch der Einsatz von Atomwaffen vorgesehen ist. Das wäre eine klare Abkehr von der Idee des 'Gleichgewichts des Schreckens'. Hat sich die Öffentlichkeit erst einmal an diesen Gedanken gewöhnt, wächst die Gefahr, daß sich Politiker dazu entschließen könnten, einen gewinnbaren Krieg auch zu führen.

TEIL II)

Ich möchte nun im zweiten Teil meines Vortrags zunächst auf einige Gründe für die doch vielleicht erstaunliche Universalität und Einsatzbreite der EDV zu sprechen kommen, die es ermöglicht hat, daß sie innerhalb kurzer Zeit eine bedeutende Stellung in vielen gesellschaftlichen Bereichen einnehmen konnte. Die Funktionsweise eines Computers kann man sich vereinfacht so vorstellen, daß er aus einer langen Reihe von Ziffern (dem Speicher) besteht

und daß der in der Maschine befindliche Prozessor planmäßig einige dieser Ziffern zueinander addiert oder andere einfache Operationen ausführt – und das tagein, tagaus bis abgeschaltet wird. Der Plan, den der Prozessor befolgt, heißt Programm und ist ebenfalls als Ziffernfolge gespeichert (die allerdings tunlichst nicht während der Ausführung verändert werden darf).

Wesentliches Merkmal sind nun die Dimensionen dieses eigentlich sehr simplen Geschehens. Es lassen sich riesengroße Mengen von Ziffern in unvorstellbarer Geschwindigkeit verarbeiten. Dadurch wird Flexibilität und Anpaßbarkeit gewonnen. Eine Maschine, die nur wenige Ziffern langsam zusammenaddiert, wäre höchstens für eine Grundschule zu gebrauchen. Maschinen, die das gleiche schnell und mit vielen Ziffern tun – eben Computer – lassen sich in fast jede Umgebung sinnvoll einfügen, denn die Besonderheiten der Umgebung können in Zahlen umcodiert und mit geeigneter Geschwindigkeit abgefragt und verändert werden. Um ein Alltagsbeispiel zu nennen: Ein Bildschirm läßt sich beispielsweise so im Speicher darstellen, daß jedem Bildpunkt eine Zahl zugeordnet ist, die die Farbe des Punktes codiert; dazu braucht man nur einige zehntausend Zahlen. Soll ein neues Bild erzeugt werden, werden für jeden Bildpunkt eine Reihe geeigneter Additionen ausgeführt, zu schnell für das Auge des Betrachters.

Größe und Geschwindigkeit solcher Systeme erlauben trotz ihrer grundlegenden Einfachheit fast beliebige Verhaltenskomplexität. Für den Programmierer stellt sich die Aufgabe, Speichermuster und Verhaltensmuster derart zu definieren, daß er die Komplexität des Maschinenverhaltens im Griff behält. Es ist nur allzu leicht, unter die Abermillionen Operationen pro Sekunde aus Versehen (durch Tippfehler, Denkfehler oder ähnliches) nur eine einzige zu schmuggeln, die etwas falsches bewirkt. Das Ausmaß solcher Komplexität ist nur durch geeignete Strukturierungsmethoden zu bewältigen. Die heutige Informatik ist zu großen Teilen die Lehre von speziellen und allgemeinen Methoden zur Strukturierung der in einem Computer stattfindenden Abläufe.

Am eben erwähnten Beispiel des Bildschirms lassen sich zwei Strukturebenen ablesen: Die Ebene der Spezifikation, auf der ein Übergang zwischen zwei Bildern als Farbveränderung beschrieben wird, und die Ebene des Programms oder der Implementation, auf der die gleiche Veränderung als eine Reihe von Operationen (Addieren usw.) beschrieben wird, die auf den Speicherinhalt wirken. Man nennt ein Programm korrekt, wenn es seiner Spezifikation genügt, d.h. das tut, was die Spezifikation von ihm verlangt.

Korrektheit bedeutet Fehlerfreiheit. Ist eine Spezifikation in einer mathematisch exakten Formulierung vorgegeben, dann hat die Behauptung, daß ein Programm korrekt sei, den gleichen Status wie eine mathematische Behauptung. Ihre Richtigkeit kann durch mathematische Methoden bewiesen werden (und die Behauptung ist dann ein Satz), oder ihre Unrichtigkeit kann durch ein Gegenbeispiel nachgewiesen werden.

Korrektheitsbeweise sind zur Zeit die einzige Chance, die Fehlerfreiheit von Programmen wirklich sicherzustellen. Es nimmt daher nicht wunder, daß in einigen Bereichen zum Beispiel in England von den Programmierern kritischer Software (das heißt: lebensgefährlicher Software) bereits mindestens eine Beweisskizze verlangt wird. Die tägliche Arbeit eines durchschnittlichen Programmierers kann auch so gesehen werden, daß er fortwährend mathematische Behauptungen aufstellt, ohne die Möglichkeit zu haben, auch deren Beweise anzutreten, oft sogar ohne sich der mathematischen Natur seiner Tätigkeit bewußt zu werden. Jeder Mathematiker würde es unter seiner Würde finden, Behauptungen ohne Beweis aufzustellen; die Informatiker müssen aber tatsächlich damit leben, und ihre Behauptungen kommen auch meist ohne Beweis zum Einsatz.

Ein Systementwickler, der einen Programmierauftrag annimmt, verlangt vom Auftraggeber zunächst einmal eine Spezifikation. In aller Regel ist diese nicht formal abgefaßt, sondern womöglich nur mündlich und auch noch in sich widersprüchlich. Um ein korrektes Resultat abliefern zu können, muß der Entwickler sich mit dem Auftraggeber zusammensetzen, bis die gesamte Spezifikation in einer unzweideutigen Form vorliegt. Das kann ein langwieriger und für beide Seiten frustrierender Prozeß sein, der aus Zeitgründen nicht in der nötigen Gründlichkeit durchgeführt werden kann. Ist ein System einmal im Einsatz, dann müssen die unbewiesenen Behauptungen, d.h. die Programme des Systems, sich ganz konkret im Wirkungszusammenhang des Systems bewähren, mit allen Eventualitäten, wie ich sie Ihnen vorher geschildert habe.

Lassen Sie mich aus diesem gewiß nicht ganz vollständigen Berufsbild des Informatikers einige Rückschlüsse für die Bedeutung der Informatik als Hochschuldisziplin ziehen.

Das ‘Kernstück’ der Informatik – die Bereitstellung von Problemlösungs- und Beweismethoden, die von einer Spezifikation zu einer Implementation führen, ist in seinem allgemeinen Teil deutlich eine mathematische Disziplin. Deshalb ist eine gute mathematische Grundausbildung für jeden Informatiker zu fordern. Ich möchte hinzufügen, daß die mathematischen Inhalte der Informatik sich allerdings schwerlich als Untergebiet traditioneller Mathematik auffassen lassen. Es gibt vielmehr ureigene Schwerpunkte, die von neuen Gebieten der diskreten Mathematik über Algorithmen- und Strukturtheorie bis hin zur Anwendung von Logikkalkülen reichen.

Es gibt Kollegen, die den Begriff ‘formale Spezifikation’ als eine Art Mauer ansehen, und zwar sowohl im fachlichen Sinn (als Trennung zwischen mathematisch-präzisem Kalkül und informeller Argumentation) als auch im persönlichen Sinn als eine Grenze für ihre Interessen. Diese Haltung ist auch unter dem Gesichtspunkt zu verstehen, daß es wesentlich mehr Spaß macht, sich auf die mathematisch-exakte oder auf die technische Seite unserer Wissenschaft zu konzentrieren, als sich um die nicht so exakt faßba-

ren Auswirkungen unserer Tätigkeit zu kümmern. Diese beiden Seiten sind aber, ob wir es wollen oder nicht, untrennbar miteinander verknüpft. (Dies steht im Gegensatz zur Mathematik, wo jemand, der etwa nach den sozialen Konsequenzen eines Theorems aus einer abstrakten Theorie früge, zu recht Kopfschütteln ernten würde.)

Ich will nun etwas näher auf die erwähnte Verknüpfung eingehen.

Diese eingeschränkte Sichtweise: daß nämlich das Gebiet diesseits der Mauer namens formale Spezifikation die eigentliche exakte Informatik ausmache, während das jenseitige Gebiet zu den Anwendungen gehört und daher von der Informatik nicht zu betrachten sei, unterbewertet zumindest die Tatsache, daß das Erstellen von Spezifikationen ein Prozeß zwischen Anwender und Informatiker ist, der meist den größeren Teil der Systementwicklungszeit in Anspruch nimmt. Die Definition formaler Spezifikations Sprachen ist deswegen eine wichtige (und allgemein anerkannte) Aufgabe der Informatik. Keiner kann erwarten, daß ein Anwender seine Spezifikation problemlos in einer solchen Sprache ausdrücken kann, und umgekehrt wirken die Anwendungen auf der Entwurf solcher Sprachen zurück. Zumindest dieser interaktive Prozeß sollte ebenfalls als Gegenstand der Informatik aufgefaßt werden.

Die Verknüpfung kann aber auch noch weiter gehen.

Natürlich versteht ein Systementwickler auch ganz oder teilweise, zu welchem Zweck sein System eingesetzt werden wird. Spätestens durch dieses Verständnis wird er gezwungen, entweder die auftretenden Fragen nach den Auswirkungen seiner Tätigkeit zu ignorieren oder dazu Stellung zu beziehen. Es gibt Fallstudien, die zeigen, daß ein Systementwickler den Entwicklungsprozeß dazu nutzen kann, diese Auswirkungen zu beeinflussen. Denn zwei verschieden aufgebaute Systeme können zu ganz verschiedenen Konsequenzen in ihrer Auswirkung auf die Betroffenen führen, selbst wenn sie die gleiche Spezifikation erfüllen. Hier besteht, in Grenzen, eine Möglichkeit für den Informatiker, sein Bewußtsein für die Belange der durch sein System eventuell betroffenen Menschen in sein Handeln einfließen zu lassen.

Ich hoffe, daß ich Ihnen darlegen konnte, daß die Informatik ein neuartiger Typ Wissenschaft ist, worin technische, mathematische, methodische und soziale Komponenten fest miteinander verknüpft sind. Aus einer ähnlichen Sichtweise heraus haben sich einige Universitäten dazu entschlossen (andere sind gerade dabei, sich zu entschließen), alle ebengenannten Komponenten in den Studiengang Informatik einfließen zu lassen und diesem dadurch einen interdisziplinären Charakter zu geben.

Zum Schluß möchte ich einige meiner Bemerkungen aufgreifen und in der Form von drei Thesen zusammenfassen und werten.

These 1: Im Gegensatz zu anderen neuen Technologien (z.B. molekularen) ist die Beherrschbarkeit informatischer Systeme zwar grundsätzlich durch

Beweismethoden gegeben; das nützt aber trotzdem nichts, da Herstellung eines Systems von anderen Kriterien als von seiner Beherrschung geleitet wird und sich daran auf absehbare Zeit nichts ändern wird.

Beleg:

In der Praxis wird ein System, das von einem Entscheidungsträger als notwendig angesehen wird, auch hergestellt, egal ob die Theorie soweit ist, grünes Licht für die Beherrschung solcher Systeme zu geben. Die Informatik-Wissenschaftler können noch soviel warnen, daß nur eine formale Spezifikation und sehr sorgfältiges Programmieren und Beweisen die absolute Sicherheit eines Computersystems garantieren können: Es wird munter weiter mit halbformalen Spezifikationen und überkommenen Methoden programmiert, diese werden zum Teil sogar kultiviert. Häufigste Ausrede ist der Zeitdruck: das System müsse bis dann und dann fertig sein. Die Folge ist bei kritischen Systemen (Kernkraftwerks-Sicherheitssysteme, militärische Systeme, etc.) eine sehr reale Gefahr der Katastrophe durch ‘Versehen’ – man könnte auch sagen, durch Nachlässigkeit bei der Herstellung der Systeme.

These 2: Die hauptsächliche Bedeutung der Informatik in unserer Gesellschaft ist im betrieblichen Bereich zu sehen, dort wo Produktion und Wertschöpfung in großem Stil stattfindet. Die EDV hilft dort, die Gewinnspanne der Unternehmen und Konzerne zu vergrößern und damit die ökonomische und ökologische Ausnutzung, um nicht zu sagen Ausbeutung, der Ressourcen der Erde zu beschleunigen.

Beleg:

In der Informatik gibt es den Begriff der Terminierungsgrößen eines Prozesses. Das sind Größen, die sich monoton in der Zeit verändern, jedoch nur bis zu einer gewissen Schranke hin. Durch solche Größen kann die Terminierung eines Prozesses bewiesen werden.

Unsere Gesellschaft befindet sich in einem Prozeß, der entscheidend von der Massenproduktion in Betrieben und Konzernen beeinflusst wird. Die Unternehmen sind gezwungen, die Gewinnmaximierung zum obersten Ziel ihrer Politik zu machen. Wie in letzter Zeit immer deutlicher wird, hat dies direkte und indirekte negative Auswirkungen auf die lebensnotwendigen irdischen Ressourcen. Die Endlichkeit dieser Ressourcen kann als eine Terminierungsgröße für diesen Prozeß angesehen werden.

Der Einsatz der EDV in Unternehmen unterstützt das Ziel der Gewinnmaximierung – und damit auch die beschleunigte Veränderung der eben genannten Terminierungsgröße – auf verschiedenerelei Arten.

Der Einsatz der EDV in der Produktion steigert die Produktivität und damit auch die Gewinnspanne durch die Ersetzung von menschlicher Arbeitskraft durch EDV-Systeme. In dieser Rolle ähneln EDV-Systeme anderen maschinellen Hilfsmitteln, die in der Geschichte oft für Arbeitslosigkeit verantwort-

lich gemacht wurden, außer daß es auch um die Ersetzung von 'Kopfarbeit', nicht nur um die Ersetzung manueller Tätigkeiten geht.

Der Einsatz der EDV in der Verwaltung hat neben Rationalisierungen in der Verwaltung auch das Ziel, der Betriebsleitung etwa durch Informationssysteme einen guten Überblick über den Betrieb zu verschaffen, damit Personal- oder Sachentscheidungen besser (mit Gewinnmaximierung als globalem Ziel) beeinflußt werden können. Diese Rolle der EDV dürfte neuartig sein und ist besonders auf die Bedürfnisse multinationaler Konzerne zugeschnitten.

Ich möchte hier nicht als jemand stehen, der nicht auch die positiven Seiten des DV-Einsatzes würdigt. Es gibt genügend Beispiele dafür, daß der Einsatz von EDV tatsächlich Arbeitserleichterungen schafft. Auch dies verbindet die EDV mit anderen maschinellen Hilfsmitteln. Längerfristig ist die Rolle der EDV als Arbeitserleichterer (und vielleicht Wegbereiter der 4-Tage-Woche) uneingeschränkt positiv zu bewerten. Solange aber die Arbeitsplatzsicherheit mitbedroht ist, sollte meiner Ansicht nach diese Rolle der EDV nicht losgelöst von den anderen betrachtet werden.

These 3: Die Informatik kann sich zu einer ganzheitlichen Wissenschaft entwickeln.

Die Informatik benötigt, wie ich vorhin ausführlicher erklärt habe, Einflüsse aus anderen Gebieten, insbesondere aus der Mathematik und aus den Gesellschaftswissenschaften.

Ein kurzer Blick in die Geschichte der Physik zeigt, daß die moderne Physik sich erst von den Beschränkungen theologischer Vorurteile und Dogmen befreien mußte, um überhaupt zu einer Wissenschaft zu werden. Die Trennung von wissenschaftlich-technischen Aspekten und allgemein-menschlichen oder -gesellschaftlichen Aspekten gehört seither zur Standardausrüstung abendländischer Wissenschaftler (nicht nur Physiker, sondern erstaunlicherweise auch Biologen).

Neuerdings wird angesichts der negativen Folgen von Forschung und Technologie wieder verstärkt darüber diskutiert, ob eine solche Trennung heutzutage noch sinnvoll ist. Aus vielen Fachgebieten werden Stimmen laut, die eine ganzheitliche Wissenschaft fordern, die die Belange der Menschen und der Natur in ihre Betrachtungen mit einbeziehen soll, mehr noch: zum Maßstab ihrer Betrachtungen machen soll.

Ich habe Ihnen aufgezeigt, daß die Informatik von ihrer Anlage einige Voraussetzungen mitbringt, eine solche ganzheitliche und segensreiche Wissenschaft zu werden. Ob sie sich dazu entwickeln kann, hängt davon ab, ob wir und unsere Kinder von großem Unglück, Krieg oder Zerstörung unserer Lebensgrundlagen, bewahrt bleiben – und auch davon, ob genug Menschen, Politiker wie Wissenschaftler bereit sind, eine humane und nicht rein technologische Ausrichtung unserer Wissenschaft zu unterstützen.

Ich danke Ihnen für Ihre Aufmerksamkeit.