

Inger Lytje

# Formaliseringskravet i menneske-maskine interaktionen

Forudsætninger og konsekvenser

## 1. Introduktion

Datamater og edb-systemer er områder man traditionelt beskæftiger sig med inden for datalogi. Her samler interessen sig om *kendsgerninger* om edb og edb-udvikling, idet disse kendsgerninger udsiger noget om potentialer og begrænsninger i relation til informatisering.<sup>1</sup> Datalogi er således forankret i de *eksakte videnskaber*, lige som naturvidenskab er det.

Styrken i de eksakte videnskaber ligger i bestræbelsen efter *sandheden*, underforstået en forkastelse af relativismen. Det er en bestræbelse efter, via abstraktions-konkretiseringsbevægelsen, at finde frem til tingenes *væsen*. Og styrken ligger i en tradition, hvor viden operationaliseres og omsættes til praktisk handling. Begrænsningerne ligger i deres *strukturalistiske* og *ahistoriske* tilgang og i deres *moral-frihed*, som til dels hænger sammen med ideologien om *universalitet* og *almengyldighed* i de propositioner, som udsiges om virkeligheden. Det blokerer for en dynamisk helhedsforståelse, som kun opnås ved at sætte tingene ind i deres historiske og sociale kontekst. For at gøre det må vi gå uden for de eksakte videnskabers tradition.

Datalogi rummer også sit halvferdseroprør, som består i at universaliteten og almengyldigheden problematiseres og forskningens værdifrihed drages i tvivl. Anledningen til oprøret var den måde hvorpå datalogisk viden blev nyttiggjort gennem udvikling af edb-teknologi. Gennem *konsekvensanalyser* blev det påvist at magthavernes interesser blev fremmet gennem udvikling af teknologi, på bekostning af arbejderes og jævne folks interesser.<sup>2</sup> Og senere viste kvindeforskningen at kvindearbejdet blev misfortolket i forbindelse med indførelse af ny teknologi.<sup>3</sup> Set ud fra én synsvinkel kunne man sige at systemerne var gode. Set ud fra en anden synsvinkel at de var dårlige. Denne relativisme i vurderingen af systemerne aktualiserer spørgsmålet om, hvad der er viden og hvordan viden opdyrkes inden for datalogi. Dermed fødes en problematisering af hele den datalogiske videnskab.

Begrebet *computerkultur* er primært forankret i humaniora og knyttet til en hermeneutisk tradition, hvor interessen samler sig om den *betydning*, mennesker tillægger edb-systemer, og om den måde systemerne *tolkes* på.<sup>4</sup> I denne sammenhæng vil jeg fortolke computerkultur inden for rammerne af *humanistisk datalogi* forstået som en ny tværvidenskabelig forståelsesramme, som skabes i en *dialog* mellem en humanistisk og en datalogisk tradition. Denne dialog udspiller sig på flere niveauer fra det generelt videnskabsteoretiske til det egentlig tværfaglige, hvor et genstandsområde belyses ud fra flere faglige vinkler, og viden skabes gennem en form for syntesedannelse. Kun gennem en sådan syntesedannelse mellem datalogi og humaniora er det muligt på én gang at studere edb og studere det menneskeliv, som i bred forstand udfolder sig i forbindelse med udvikling og anvendelse af edb-systemer.

Inden for den ovenfor skitserede forståelsesramme vil mit udgangspunkt være, at ét og samme system kan tolkes på flere forskellige måder afhængig af den historiske og sociale kontekst, hvori systemet bringes i anvendelse. Set fra en datalogi-vinkel kan man imidlertid godt synes, at den betydning, som edb-systemer tillægges »har meget lidt hold i virkeligheden«, underforstået at der med »i virkeligheden« menes faktiske kendsgerninger om edb-systemer. Mit ærinde er at påvirke betydningsdannelsen gennem *oplysning* om formaliseringskravet i menneske-maskine interaktionen og de potentialer og begrænsninger, som følger heraf. Med udgangspunkt heri vil jeg beskæftige mig med den forandring i kultur, som er knyttet til udbredelsen af datamater til stadigt flere mennesker og områder. Jeg mener således at der er nogle træk ved datamater, som på én gang cementerer nogle kulturtræk, som allerede er ganske udtalte i vores samfund, og samtidig implicerer kulturelle forandringer, som i høj grad vil ske gennem et opgør med de kulturtræk, som datamaterne cementerer.

De træk i kulturel forandring, som jeg vil interessere mig for er forandringer i *viden*, *sprog* og *tænkning*. Desuden vil jeg interessere mig for de forandringer i dagligdags *rutiner* og *vaner*, som edb-teknologien implicerer. Begrundelsen for at rette opmærksomheden mod disse områder findes i træk ved edb-systemer og i brugernes interaktion med systemerne.

## 2. *Menneske-maskine relationen*

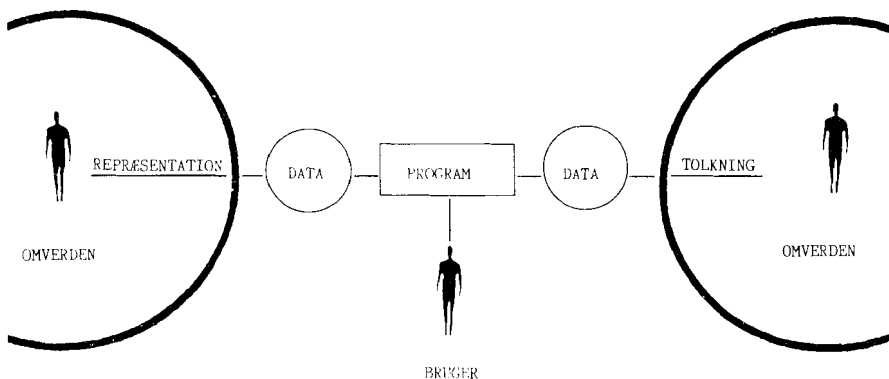
Problemstillingen i denne artikel omfatter to nøglebegreber, nemlig formalisering og menneske-maskine interaktion. Vi skal diskutere det sidste først, og indleder afsnittet med en terminologisk oversigt.

Computere kan bl.a. forstås som maskiner, som *behandler* information, og som *styres* af information. Den information, som behandles, kalder vi *data* og den information, som styrer, kalder vi et *program*. Der er tale om information på to forskellige niveauer, idet programmet indeholder information *om* data. Programmet aktiveres af en *bruger*, som på den måde starter *programudførelsen*. Programmet kan være udformet således, at udførelsen sker i interaktion med brugeren, som *overfører* styreinformation til programmet og modtager systemmeddelelser undervejs. Et program kan forstås som en *maskine*, som udfører informationsbehandling for brugeren og derved genererer ny information. Menneske-maskine interaktionen består i at overføre information mellem omverden og maskine. Betragter vi denne informationsoverførsel i lys af det andet nøglebegreb, formalisering, bliver *kodningen* interessant. Informationen skal have en *form*, så den kan overføres til og behandles af et program. Samtidig må brugeren stille sig tilfreds med information som er genereret af et program.

Når vi kører tekstbehandling styres computeren af et tekstbehandlingsprogram, som udføres i interaktion med brugeren, som løbende overfører styreinformation til programmet, f.eks. gennem anvendelse af funktionstaster. Data er den indtastede tekst, og databehandling er tekstredigering.

I det følgende vil vi skelne mellem 3 former for menneske-maskine relation. Heri er programmeringsprocessen ikke medtænkt, idet den vil blive behandlet i afsnit 4.

Den første form vil jeg beskrive med udgangspunkt i følgende model:



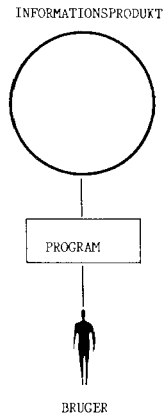
I midten af figuren har vi et program, som anvendes af en bruger. Programmet læser data ind i maskinen og genererer data, som stam-

mer fra en bearbejdning af de indlæste data. Data skal i denne sammenhæng forstås som en formel *repræsentation* af kendsgerninger eller sagsforhold fra »den virkelige verden«. Data er således fremkommet gennem en formaliseringsproces, og til grund for denne formalisering ligger en tolkning af »den virkelighed«, som repræsenteres. Resultatet af bearbejdningen skrives ud og fortolkes af brugeren som kendsgerninger om »den virkelige verden«. Der sker en reduktion i den mængde af information, som er knyttet til virkeligheden selv. På baggrund af den reducerede information ræsonneres der så om »hele virkeligheden«.

Som eksempel på den skitserede model kan nævnes bibliotekssystemer. Data i det system fremkommer gennem katalogiseringsprocessen, hvorved data registreres i et givet format, f.eks. MARC formatet, og lagres i maskinen. Gennem anvendelse af et søgesprog, kan låneren anmode systemet om at søge bestemte bøger frem. Disse data tolkes så af låneren som handler derefter.

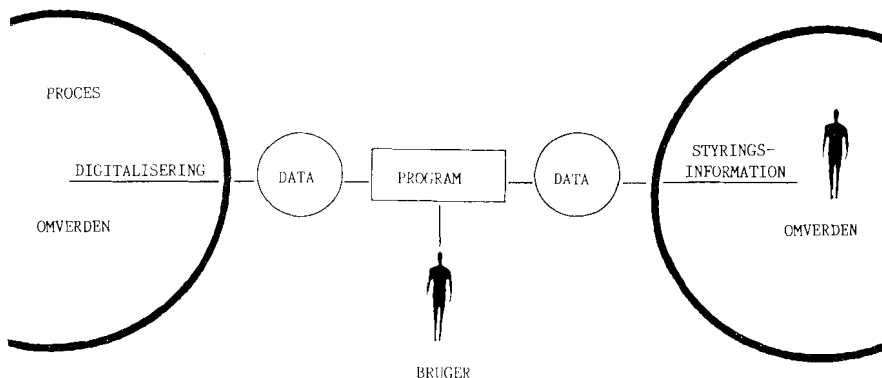
Det som kendetegner denne model er at de data som behandles frembringes *bevidst* af mennesker og at de data, som genereres af systemet tolkes bevidst af mennesker.

Den næste model er udtryk for, at information, som behandles eller generes af en datamat, ikke behøver at være information om noget:



I det tilfælde vil vi ikke tale om data, men om et *informationsprodukt*. Der tænkes f.eks. på tekster, billeder, computerkunst og computerspil. Fælles for disse informationsprodukter er at brugeren af programmerne kan tillægge produkterne en betydning. Ikke på den måde at de udsiger noget om en anden virkelighed, for de er selv selve virkeligheden.

I den sidste model bearbejder maskinen data, som stammer fra naturlige processer, f.eks. biologiske processer.



Signalerne *digitaliseres*, hvilket betyder at de kodes i alfabetet  $\{0,1\}$ . Data bearbejdes af et program, som tillægger data en betydning og programmet genererer information, som anvendes i »den virkelige verden«. Data produceres ikke som i den første model af den menneskelige bevidsthed. Den menneskelige tolkning kommer kun til udtryk i det program, som bearbejder data. Som eksempel på den type af menneske-maskine relation kan nævnes edb-baserede styrings-systemer i procesindustrien, patientovervågning og militære overvågningssystemer.

### 3. Algoritmer

Ved interaktion med en datamaskine interagerer brugeren med et *program* og kernen i det program er *algoritmen*. Algoritmer udføres automatisk af maskinen, og styrer således selve informationsbehandlingen. Det er algoritmer, som læser data ind i maskinen (input), og det er algoritmer, som skriver data ud fra maskinen (output). Algoritmen definerer således formen af input og output, så i en diskussion af formaliseringskravet i menneske-maskine interaktionen står algoritmebegrebet centralt. Derfor skal vi diskutere det begreb nærmere.

I hele det 20. århundrede har matematikere arbejdet med algoritmebegrebet i et forsøg på at finde opskrifter på, hvordan bestemte klasser af matematiske problemer kan løses.<sup>5</sup> Ideen i disse bestræbelser er, at når forskriften for løsning af et problem én gang er fundet, så kan enhver, som er i stand til slavisk at følge den forskrift, løse problemet. Vi kan f.eks. blot tænke på algoritmen for at addere tal, som er noteret i 10-tals systemet. Nu omfatter den nævnte »enhver« ikke længere bare mennesker, men også datamaskiner. I 1930'erne får algoritmebegrebet en præcis definition af den engelske matemati-

ker Allan Turing, som knytter begrebet til en matematisk model af en datamaskine. Det er den definition, jeg vil knytte an til her, idet den kan anvendes til at klargøre, hvori formaliseringen består.<sup>6</sup>

Ifølge denne model hører der til en datamaskine et endeligt (men undertiden meget stort) antal *tilstande*, som maskinen potentielt kan befinde sig i, og til maskinen hører et *alfabet*, som al information skal kodes i. Til maskinen hører også et *informationslager* og i det lager placeres to typer af information, nemlig den information, som definerer det konkrete problem, som skal løses og mere universel information om selve den algoritmiske proces. Denne er defineret gennem en *formel beskrivelse* af processen. Under programudførelsen gennemløber maskinen en sekvens af tilstande, bestemt af den information, som er placeret i informationslageret. Processen udføres af en *automatisk kontrol* med en *føler*, som kan afkode informationen i lageret.

Hensigten med at arbejde med en så abstrakt model af en maskine er at formidle en forståelse af selve maskinens væsen og dermed af dens potentialer og begrænsninger.

#### 4. *Hvori består formaliseringskravet?*

På baggrund af afsnit 2, hvor kontaktfladerne mellem menneske og maskine blev præsenteret og afsnit 3, hvor jeg præsenterede en forståelse af en datamaskine vil jeg i dette afsnit diskutere, hvori formaliseringskravet består, idet jeg med *formalisering* mener repræsentation inden for rammerne af et formelt system.

Lad os først se på overførsel af information fra brugeren til maskinen. En betingelse for at maskinen kan tillægge den information en *betydning* er at informationen er formet som en tekst<sup>7</sup> som er struktureret i overensstemmelse med en formel *grammatik*. En høj grad af strukturering er ensbetydende med megen betydningsinformation, mens en ringe grad af strukturering er udtryk for det modsatte.<sup>8</sup> Tekster skrevet i et programmeringssprog er eksempel på det første, mens almindelige tekster, som skrives ved hjælp af et tekstbehandlingssystem, er eksempel på det andet. Det program, som »fortolker« inputteksten, skal udformes på en måde, så det kan »forstå« alle tekster, som er struktureret i overensstemmelse med den pågældende grammatik.

Der er i princippet to forskellige måder, hvorpå input til et program kan tillægges betydning. Den ene er gennem *navngivning* og *kategorisering*. Hvis f.eks. input indeholder en oplysning om alder på en person, så kan man via programmet knytte navnet »alder« til den oplysning. Den anden er gennem oversættelse af det skriftsproglige

udtryk til en algoritmisk *proces*. Forståelse svarer til *handling*. Hvis f.eks. input er en ordre til et tekstbehandlingssystem, så er det den betydning ordren tillægges af tekstbehandlingsprogrammet.

Det formelle system, som betinger at der udelukkende kan opereres med de nævnte opfattelser af forståelse og betydning er selve datamaskine-konceptet, og formaliseringskravet består i at de kommende brugere af systemet skal følge grammatikreglerne. Sprogriktighed og betydning bliver således tæt forbundne, idet afvigelser fra grammatikreglerne får konsekvenser for den betydning, som programmet tillægger input. Der kan være tale om at systemet afviser brugeren, fordi input ikke kan fortolkes. Der kan være tale om at oplysninger navngives forkert, hvilket kan få de alvorligste følger, eller der kan være tale om, at programudførelsen forløber anderledes end planlagt, hvilket også kan få de alvorligste følger.

Umiddelbart kan de nævnte principper for betydningdannelse synes ganske primitive af flere grunde. Maskinen processer input slavisk efter en forskrift og besidder ikke *viden om* processen. Input fortolkes ens uanset konteksten, etc. Dertil kan siges at metaviden om processen kunne lægges ind i maskinen, blot denne viden formuleres sprogligt og inden for rammerne af en formel grammatik. Kontekstinformation kunne også lægges ind i maskinen under de samme betingelser. Pointen er at formaliseringskravet ligger i kravet til skriftsproglig formulering inden for rammerne af en given formalisme.

Vi har nu vist at de formaliseringskrav, som en edb-bruger underlægges til dels hidrører fra datamaskinens væsen som en maskine, der udfører algoritmer, som opererer på data, som er kodet i et tegnsystem. Vi vil dernæst se på den form for formalisering, der ligger i at reducere processer til algoritmiske processer eller at reducere viden og information til data. Disse former for formalisering realiseres på udviklingstidspunktet af programmører og systemudviklere.

Et vigtigt spørgsmål i den sammenhæng er hvilke processer, som kan informatiseres og hvilke der ikke kan. Er f.eks. beslutningsprocesser algoritmiske processer? Er problemløsningsprocesser algoritmiske processer? Kan den viden og information, som anvendes i arbejdet repræsenteres i et edb-system som data? Beskrivelse af arbejdsprocesser kommer til at spille en central rolle. Formaliseringen ligger i at identificere de algoritmisk løsbare problemer og dermed identificere de algoritmiske processer. Eller det ligger i at identificere formaliserbare kommunikationsprocesser eller at identificere den information og viden, som med fordel kan repræsenteres i en datamat.

Formaliseringsprocesserne er centrale i en udviklingsproces, og systemudvikleren har frit spil inden for givne rammer. Processerne er

centrale, fordi de bliver bestemmende for de rutiner og vaner, som etableres i de sammenhænge, hvori systemerne bringes i anvendelse.

De formaliseringskrav, som systemudvikleren er underlagt, formidles i praksis gennem det *sprog*, som udvikleren har til rådighed. Sproget leverer et sæt af begreber og kategorier, som udvikleren kan tænke i, og vil i praksis være stærkt styrende for tankevirksomheden. Anvender udvikleren f.eks. et *databasesystem* til udvikling af et edb-baseret informationssystem, påtvinges han eller hun at tænke i objekter, relationer mellem objekter, propositioner om objekter etc. De søgemuligheder, som udvikles til brugerne vil i praksis afhænge af, hvilke faciliteter udvikleren har til rådighed til systemudviklingen. Findes der et søgesprog i systemet? Er der mulighed for grafisk kommunikation? etc. Samme type af argumenter gælder for alle andre sprog: objektorienterede programmeringssprog, ekspertsystemskal-ler, logikprogrammeringssprog, forfattersystemer etc. De sætter alle konkrete begrebsmæssige rammer for udviklerens tænkning.

### 5. Formaliseringskrav og kulturel forandring

Af afsnittet ovenfor fremgår det at det ikke er helt uden grund at computere får tillagt en betydning som »maskiner, som forstår sprog«, »maskiner, som kan tænke« og »maskiner, som indeholder viden«. Og det er ikke uden grund at computere betragtes som autonome enheder, som organiserer de sociale processer, hvori de indgår. Jeg mener vi kan opsummere det på den måde, at datamaskiner forstår sprog i det omfang sprog sættes lig med skriftlig information. Datamaskiner kan tænke i det omfang tænkning sættes lig med beregning (forstået som processer, der udføres af algoritmer). Og datamaskiner er vidende i det omfang vi med viden forstår propositioner om objekter og regelbaseret viden. Og datamaskiner organiserer sociale processer i det omfang mennesker betragtes som maskinens slave og ikke som dens herre.

Gennem datamaskinens eksistens og anvendelse tillægges de nævnte begreber altså en betydning i relation til en maskine. Denne betydning ligger i forlængelse af de betydninger af begreberne, som iøvrigt er herskende i vor kultur, hvilket jeg vil vise i det følgende.

Vedrørende begrebet *viden* er det karakteristisk at edb-systemer repræsenterer viden i en eksternaliseret, sproglig og ikke personbunden form. I vores samfund er det en udbredt opfattelse at den artikulerede, formulerede og objektive viden, som opdyrkes gennem videnskaben, er identisk med viden som sådan, og på den måde cementerer de vidensbaserede systemer den herskende vidensopfattelse. I et op-



gør med denne fremhæver mange betydningen af den erfaringsbundne, ikke-sproglige viden, som er knyttet til mennesker og som opdyrkes gennem praksis.<sup>9</sup> Det er en form for viden, som ikke kan opbevares i en datamat. Datamaskinerne har bragt vidensbegrebet til debat.

Vedrørende begrebet *sprog* er det karakteristisk at interaktionen med datamaskiner er *skriftsproglig*, og at det er meget vigtigt at brugeren overholder de regler, som grammatikken tilsiger. Kravene om korrekthed i stavning og i ordsammensætning får en renæssance. Dette krav om skriftsproglighed er i forvejen stærkt udtalt i vor kultur. Men ikke alene stiller datamaskinen yderligere *alfabetiseringskrav* til alle samfundsborgere. Den bliver også bærer af en kultur, hvor sproglige formalismer som erkendelsesmiddel spiller en central rolle (i modsætning til andre kulturer, hvor *intuitionen* spiller en vigtig rolle, f.eks. indenfor Zen-buddismen). Det er en kultur, som er præget af dualisme i den forstand at selvet adskilles fra omverdenen i vores forsøg på at erkende den sprogligt.<sup>10</sup> Den sprogopfattelse, som er knyttet til datamaskiner, modsiges af en række teoretikere, som fremhæver det ikke-verbaliserede sprogs betydning i en kommunikation og det kontekstbundne og situationsbestemte moment i fortolkningen af sproglige udtryk.<sup>11</sup>

Der er en snæver sammenhæng mellem sprog og *tænkning*, idet sproget i en vis forstand sætter rammerne for, hvordan man kan tænke. Computeren har som begreb givet os nogen nye begreber at tænke i, f.eks. at »tænke i information« eller »tænke i algoritmiske processer«. Disse begreber og denne form for tænkning vil igen afspejles i vores sprogbrug. Computeren som en tænkende maskine er en metafor, som bestandig går igen inden for området kunstig intelligens. Allerede Allan Turing skrev kort før sin død i 1954 et berømt paper »Computer Machinery and Intelligence«, hvor han argumenterer for at man godt kan sige at datamaskiner kan tænke.<sup>12</sup> I 1957 skriver John Von Neumann en bog *The Computer and the Brain*, hvor han sammenligner computeren med den menneskelige hjerne.<sup>13</sup> Og også i dag forventer man at finde »det store gennembrud« for kunstig intelligens via hjerneforskningen inden for området »Neurale Netværk«. Datamaskinen har givet os en skabelon inden for hvilken vi kan tænke om tænkning som værende identisk med algoritmiske processer, logiske ræsonnementer, beregninger (hvilket alt sammen på et givet abstraktionsniveau er ét fedt). Og samtidig forskes der i, hvordan den menneskelige tænkning egentlig foregår for derigennem at finde frem til en ny og anderledes model for at bygge datamaskiner.

I forbindelse med systemudvikling har det vist sig at være vigtigt at tænke kultur, organisering og mennesker med, når systemer skal ud-

formes og bringes i anvendelse<sup>14</sup> Ellers sker der næsten automatisk det, at systemernes hovedfunktion bliver at bekræfte den herskende orden. Det betyder bl.a. at de udvikles med det formål at løse problemer, som tidligere systemer har skabt, og det betyder igen at der sker en cementering af samfundets organisering på alle niveauer på bekostning af den produktivitetstigning, som latent er til stede i datamaterne.

Men samtidig er der i forbindelse med systemudvikling en mulighed for at bruge potentialerne i edb-teknologien til en omorganisering og et opbrud ud fra en tese om at edb-teknologien kan udgøre en ny materiel basis for vores dagligliv. Derfor er det vigtigt at tænke i sociale kategorier, når man tænker systemudvikling til gavn for mennesker.

## 6. Afslutning

Gennem en undersøgelse af menneskers interaktion med datamaskiner og gennem en afklaring af formaliseringsbegrebet er vi nået frem til at datamaskiner er et typisk produkt af den vesterlandske kultur, samtidig med at de udfordrer denne kulturs grundlag. Computeren sætter fokus på basale kulturbegreber som sprog, viden og tænkning, hvor datamaskinerne kan betragtes som inkarnationen af vor kulturs rationalistiske menneske- og verdenssyn. I anvendelsen konfronteres det med dagliglivets logik, og i denne konfrontation skabes en erkendelse om vores kultur, som kan blive anledningen til en dybtgående forandring i denne. Teknologiudviklingen understøtter derfor den postmodernistiske tese om at dekonstruere etablerede begreber og skabe nye som grundlag for et nyt verdenssyn, hvor bl.a. begrebet om det teknologiske fremskridt får ny betydning.

Computere udgør en ny materiel basis for omorganisering af samfundet på alle niveauer, formidlet gennem en øgning i den menneskelige produktivitet. En sådan omorganisering vil samtidig være en betingelse for at få frigjort det potentiale, som er knyttet til computeren og at få rigdommene fra produktivitetstigningen fordelt på retfærdig vis. Omorganisering vil medføre opbrud i dagliglivets vaner og rutiner og i vore livsformer som sådan og vil være forbundet med en ændring i det værdisystem, som bærer vor kultur.

## Noter

1. Informatisering. Hermed menes omlægning ved hjælp af edb. Jfr. også det engelsksprogede begreb computerization.
2. Jeg tænker på det svenske DEMOS-projekt, det danske DUE-projekt og det norske NJMF-projekt.
3. Lytje, Inger m.fl., *Kvinder og Teknologi*, Konferencerapport, Aalborg Universitetscenter, 1986
4. Jeg knytter i afsnittet an til den opfattelse af kultur, som bl.a. tegner sig i Gunhild Aggers artikel »Hverdagskultur og Organisationskultur i arbejdersammenhæng 1870-1930«, *Politica*, nr.3 1987.
5. Hofstadter, Douglas R., *Gödel, Escher, Bach. An Eternal Golden Braid*, Vintage Books, 1979.
6. Trakhtenbrot, B.A., *Algorithms and Automatic Computing Machines*, D.C. Heath, 1963.
7. Her skal begrebet tekst ikke forstås i den humanistiske, kvalitative forstand, men i den rent kvantitative som en sekvens af tegn, som hører til et givet alfabet.
8. Der knyttes an til informationsteoriens kvantitative informationsbegreb, som blev introduceret af Claude Shannon i 1940'erne. Se f.eks. Hamming, Richard W., *Coding and Information Theory*, Prentice-Hall, 1980.
9. Jannik, A., *Tacit Knowledge, Working Life and Scientific Method*, Göransson, Bo og Josefson, Ingela (eds.), *Knowledge, Skill and Artificial Intelligence*, Spronger-Verlag, 1988.
10. Grønbæk Hansen, *Erkendelse og Bevidsthed*, Gyldendal, 1987.
11. Winograd, Terry m.fl., *Understanding Computers and Cognition, A New Foundation for Design*, Addison-wesley, 1986.
12. Turing, Allan, »Computing Machinery and Intelligence«, in: Feigenbaum, Edward A. (ed.), *Computers and Thoughts*, McGraw-Hill, 1963.
13. Von Neumann, John, *The Computer and The Human Brain*, Yale University Press, 1957.
14. Lytje, Inger, *A Humanistic Approach to System Development*, Office: Technology and People, udkommer.