

Den tause kunnskapen i IT-studia

Hans Georg Schaathun

⟨georg@schaathun.net⟩

NTNU — Noregs Teknisk-Naturvitskaplege Universitet

Samandrag

Michael Polanyi er kjend for omgrepet *taus kunnskap*. Han viste at me ofte veit meir enn me kan fortelja. Dette ser me tydleg i profesjonsstudia. Studentane treng ein praktisk kompetanse som førelesarane slit med å setja ord på. Denne artikkelen oppsummerer litt av det som har vore skrive om kva taus kunnskap er, kvifor han er taus og korleis han vert formidla. Me argumenterer for at det er den tause kunnskapen som skil menneske frå maskin og dermed er det som studentane våre treng i ein kvar jobb som ikkje snart vert robotisert.

1 Innleiing

Ei av dei største utfordringane for oss i utdanningssystemet er balansen mellom teori og praksis. Me ser det i konflikten mellom industrikompetanse og forskarkompetanse ved tilsetjingar. Me ser det ved prosjektbasert læring, der både studentar og sensorar kan vera usikre på kva vurderingskriterium som ligg til grunn når det innleverte arbeidet skal få karakter. Svært sjelden finn man tydleg formulerte vurderingskriterium som kunne sikra objektive og forutseielege karakterer.

Ei anna utfordring, som slektar på den fyrste, er manglande samanheng mellom teori og praksis. Den teoretiske kunnskapen er proposisjonell, dvs. me kan formulera han i ord, og ein stor del av forskarkompetansen handlar nettopp om å skriva (og lesa) for å omsetja kunnskap i ord. Den praktiske kunnskapen er gjerne taus (*tacit knowledge* som Michael Polanyi kalla det), og mange dyktige praktikarar maktar ikkje å forklara kva dei gjer eller kvifor dei gjer som dei gjer. Slik opererer praktikarane og teoretikarane på kvar sin kognitive arena og kan bidra me svært lite til studentane utover sin eigen arena.

Taus kunnskap er vanskeleg å handtera i pensum. Regelverket for høgare utdanning krev skriftlege læringsutbytebeskrivingar og skriftleg sensorvegledning som skal sikra objektiv og etterprøvbar sensur. Taus kunnskap er nettopp taus fordi me ikkje maktar å forklara kva me veit. Korleis kan me då skriva sensorvegledning og emneskildring?

I denne artikkelen tek me for oss konseptet taus kunnskap saman med det beslektta omgrepet *techne* frå Aristoteles, gjerne omsett som kunst, samt danningomgrepet frå tysk og skandinavisk didaktikktradisjon. Me viser kvifor kunst og taus kunnskap er noko anna enn vitskapleg kunnskap, og kvifor det er minst like viktig i eit profesjonsstudium. Målet er å setja ord på ein del av utfordringane som me har i IT-utdanningane, og vonleg inspirera til vidare diskusjon.

Denne artikkelen vart presentert på konferansen NIK-2019; sjå <http://www.nik.no/>.

2 Tilbake til Aristoteles

Aristoteles (1999, Bok 6, Kap. 1) delte den rasjonelle sjela i to delar, den *vitande* delen (ἐπιστημονικόν, epistêmonikon) og den *overvegande* delen (λογιστικόν, logistikon). Den vitande delen tek seg av kunnskap om ting som ikkje kunne ha vore annleis, medan den overvegande delen søkjer sanning om det som kan vera annleis. Dette skiljet er fundamentalt. Kunnskap eller *episteme* (ἐπιστήμη) handlar om evige sanningar. Overveging, på den andre sida, søkjer innsikt i einskildtinga og dei unike falla som avheng av tid og stad. I informatikkjargon kan me seia at kunnskap handlar om klassene, medan den overvegande delen skal forstå instansane.

Til den overvegande delen høyrer mellom anna *techne* (τέχνη), som gjerne vert omsett til kunst eller kunnen og er evna til å skapa gode og riktige ting. Dei praktiske IT-faga våre handlar nettopp om å skapa. Studentane våre skal skapa programsystem som skal bidra til «det gode livet». Dyden som studentane skal læra seg i slike fag er dermed *techne*. *Techne* dreier seg ikkje om å finna løysingar som alltid er sanne, men om å finna løysingar som er *gode* i den konkrete, unike situasjonen som ein har framfor seg.

I skiljet mellom *episteme* og *techne* kan me kanskje sjå det kjende skiljet mellom teori og praksis, men då skal me vera nøye med kva me meiner med praksis. *Episteme* er kunnskap om ting som ikkje kunne vera annleis. Me kan gjerne bruka praktiske øvingar og studentaktive læringsformar for å formidla *episteme*, t.d. i aritmetikken. Oppgåver som $812 + 373 =$ eller $1243 : 17 =$ har eitt svar som ikkje kunne vera annleis. Når studentane øver på å rekna for hand er det difor stadig *episteme* dei lærer, sjølv om oppgåva på sett og vis er praktisk og i alle fall studentaktiv.

Døme nærare vårt eige fag finn me i algoritmeteorien. Som matematikken, er algoritmeteorien formell. Både objekt og problem er veldefinerte, og algoritmane som me studerer, er universelt gyldige innanfor den formelle modellen. Dijkstras algoritme finn alltid den kortaste stigen i ein graf (på visse veldefinerte vilkår). Vidare kan me finna universelle sanningar om køyretid og kompleksitet. Alt dette er *episteme*, og når me bed studenten om å implementera Dijkstra i python eller i assembly, er *episteme* stadig det primære målet. Studenten implementerer algoritmar for å hugsa algoritmen, ikkje for å læra å programmera.

Techne kjem inn i biletet når me hentar problemet frå røynda, heller enn frå den abstrakte grafmodellen. Tak t.d. eit logistikkproblem. Me skal finna kortaste køyreveg herfrå til Gokk. Utfordringa er ikkje algoritmen; den er universell. Utfordringa er heller å modellera geografien som ein vekta graf slik at den universelle algoritmen kan brukast. Her nyttar det ikkje berre å vita, me er nøyde til å overvega. Kva meiner me med kortast? Tid eller distanse? Kor detaljert skal modellen vera? Presisjon må vegast opp mot køyretid. Kor lenge vil brukaren venta på utrekninga? Korleis presenterer me resultatet for brukaren? Alt dette krev merksemd om enkelttinga i det konkrete logistikkproblemet. Difor er *episteme* ikkje nok.

Aristoteles (1999, Bok VI, Kap. 8) var ikkje særleg optimistisk til utdanning for den overvegande sinnsdelen. Unge mennesker kan nok «bli geometere og matematikere og flinke i slikt,» meinte han,

men klokt kan ikke et ungt menneske være. Grunnen til dette er at klokskap også gjelder enkeltting, noe man får kjennskap til gjennom erfaring, mens den unge jo er uerfaren. For det er bare nok tid som gir erfaring.

Me skal ikkje hevda at skiljet mellom *techne* og *episteme* er like svart/kvitt som

Aristoteles ser ut til å meina i *Etikken*. Det er det grunnleggjande skiljet mellom kunnskap om det universelle og forståing for enkelttinga som er interessant. Konseptta *techne* og *episteme* kan me då sjå på som ytterpunkt og som representantar for hhv. overvegande og vitande sinnsevner. Gråsonen kan ein evt. koma tilbake til seinare. Me skal utdjupa dette skiljet med hjelp av nokre meir moderne tenkjarar, og drøfta kva det kan fortelja om utdanninga i IT-fag i dag.

3 Den tekniske rasjonaliteten

Akademia har tradisjonelt hatt lite tid til *techne*. Idealet for framveksten av moderne vitenskap har vore universelle sanningar og kontekstfrie resultat. Vitskapen har òg vorte idealet for utdanninga. Skular og universitet søkjer å formidla universell sanning. Som me har høyrd ein del sosialøkonomar formulera det, *røynda er berre eit uinteressant spesialtilfelle*. Schön (1983) er ein av dei som har studert problema denne tradisjonen har skapt for profesjonsutdanningane.

Profesjonsutdanningane

Profesjonane i Schöns vokabular omfattar so vidt forskjellige disiplinar som arkitektur, design, ingeniørfag, kliniske fag, osv. Desse profesjonane vert ofte omtalt som design, i tråd med ordbruken etter Simon (1996, p. 111):

Everyone designs who devises courses of action aimed at changing existing situations into preferred ones.

Kjenneteiknet på design er altso at ein skaper nye og betre løysingar eller situasjonar. Dette står som motsats til vitskapen der ein søkjer å forstå eller forklara situasjonen slik han er. Her kjenner me att definisjonen av *techne* som skapande sinnsevner, til skilnad frå *episteme* eller vitskap. I denne ordbruken står då IT-faga fram som designfag.

Der har vore eit sterkt press gjennom meir enn 100 år for å gjera profesjonane mest mogleg vitskaplege. Yrkesgruppene har fått status gjennom den vitskaplege kunnskapen som dei legg til grunn. Moore (1970, p. 55f) skil mellom *avocations* og *professions*. Ein *avocation* er det motsette av ein profesjon, fordi han baserer seg på innarbeidd og tradisjonell praksis (customary practice) og vert utvikla gjennom individuell prøving og feiling.

The profession, on the other hand, involves the application of general principles to specific problems, and it is a feature of modern societies that such general principles are abundant and growing.

Utdanninga i dei generelle prinsippa frå vitskapen fekk forrang, medan bruken på spesifikke problem gjerne vart sett på som triviell. Ingeniørskulane utvikla seg til ingeniørvitskaplege høgskular, der vitskapsfolk kunne nytta den høgaste statusen basert på heilt andre verdiar enn dei som ligg til grunn for profesjonen som studentane vart utdanna til (Schön, 1983, p. 27).

I 1983, då Schön gav ut den banebrytande boka si, kunne han observera ein aukande skepsis til og kritikk mot profesjonane gjennom to tiår. Trass i den suksessen som vitskapane hadde hatt i å frambringa ny kunnskap, var ikkje vitskaplege fundament nok til å sikra tillit til profesjonane. Både profesjonsmiljøa sjølve og folk flest hadde sidan 1963 vortne meir og meir kritiske til profesjonane (Schön, 1983, p. 39). Profesjonelle vurderingar er ikkje alltid nødvendige sanningar som ikkje kunne vera annleis, sjølv om dei byggjer på slike nødvendige sanningar frå vitskapen. I

det konkrete, individuelle tilfellet kan ulike ekspertar gjerne koma fram til ulike konklusjonar, sjølv om ingen tvil herskar rundt den underliggjande og generelle vitskapen. Kva tillit kan me då ha til profesjonskunnskapen?

Reflektert praksis

Ein sentral observasjon hjå Schön (1983) er at dyktige profesjonsutøvarar møter nye problem som unike fall og ikkje som instansar av generelle klasser som er dekt av eksisterande kunnskap. Unntak er regelen. Universelle sanningar strekk ikkje til.

Lækjaren kan kjenna ei lang rekkje sjukdomar og diagnosar som vitskapen har skildra i detalj, men pasientane er ofte ikkje læreboksdøme. Mange pasientar har fleire sjukdomar og atypiske symptom, som er vanskelege å skilja frå kvarandre, og ein kur som verkar positivt på ei av lidingane kan verka negative på den andre (Schön, 1983, s. 64f). Pasienten er ein «enkeltting», og lækjaren må sjå etter det særeigne i den individuelle pasienten. Fleire vitskaplege teoriar kjem saman i ein ny og unik kombinasjon, og lækjaren treng *techne* for å overvega slutningar frå ulike teoriar.

Den rådande epistemologien for profesjonane kalla Schön (1983) for den *tekniske rasjonaliteten*. Etter dette synet dreier profesjonspraksis seg berre om å løysa instrumentelle problem vha. vitskaplege teoriar. Det går greitt so lenge problema er velformulerte, men i røynda, skriv Schön, er dei fleste problem kaotiske og uoversiktlege. Veldefinerte problem er noko som profesjonsutøvararen konstruerer sjølv. Å finna riktig spørsmål er ofte både viktigare og vanskelegare enn å svara på det.

Behovet for å omforma problem er stort i faget vårt. Stadig ser me digitaliserte prosessar som er meir tungvinte og tidkrevjande enn gamle papirbaserte prosessar (Olsen, 2014). Grunnen er at dei digitale løysingane let dei gamle papirbaserte prosessane definera problemet, og gjerne kopierer prosessen steg for steg. Til dømes skriv Olsen (2014):

Da bilprodusenten Ford skulle effektivisere sine innkjøp skaffet de seg et avansert datasystem som sendte ordrer og andre dokumenter elektronisk. Toyota, derimot, fjernet behovet for disse dokumentene! Ved å gi leverandørene innsyn i Toyotas datasystem ble ordrer unødvendige.

Ford har altså gått ut frå ei universell forståing av korleis ordreprosessen foregår. Dei har løyst problemet ut frå den same problemforståinga som låg til grunn då papirprosessen vart etablert. Toyota har på den andre sida vore merksame på enkelttinga, dvs. det som er særeige for den tida som kjem. Ein ordreprosess i det 21. århundredet er ikkje det same som ein ordreprosess i det 20., og dette tek Toyota omsyn til. Toyota har altså omforma problemet ved å sjå på enkelttinga, heller enn å føresetja det universelle. Dei stør seg på *techne* og ikkje berre på *episteme*.

Innanfor designtradisjonen er der kultur for å stilla spørsmål ved problemet (Norman, 2013, s. 220), og mange designarar vert upopulære fordi dei bruker altfor lang tid før dei tek til å løysa problemet. Grunnen er likevel openberr. Altfor ofte gjer me som Ford og løyser eit problem basert på føresetnader frå ei anna tid eller ein annan bransje.

Refleksjon-i-handling

Schön byggjer bøkene sine på ei lang rekkje observasjonar av praksis. Han observerer at utøvarane utviser stor kunnskap og forståing når dei stiller spørsmål ved

problemet, og gjerne forkastar det til fordel for eit anna problem. Denne kunnskapen er i stor grad taus. Dvs. utøvaren kan ofte ikkje setja ord på kva han gjer eller korleis han tenkjer, sjølv om det er tydleg at han *er* svært kyndig i det han gjer. Det er kunnskap *i handling*, eller *techne*, som det ikkje er trivielt å kopiera for dei ukyndige.

Eit av hovudmåla for Schön er å gjera ein del av den tause kunnskapen eksplisitt. Han føresleg ein epistemologi som han kallar *reflection-in-action*. Epistemologi viser etymologisk til *episteme*, men likevel er det ikkje her tale om å skaffa kunnskap om ting som ikkje kunne vore annleis. Schön gjev oss ein «epistemologi» for *techne*¹. Målet hans er kunnskap som er objektiv ved at ein kan oppdaga feil (Schön, 1987, s. 79). Dette er eit objektivitetskriterium som høver for *techne*.

Epistemologien byggjer Schön opp rundt fellesdraga i vidt forskjellige profesjonar, med psykoterapi og arkitektur som dei mest detaljerte døma. Refleksjons-i-handling omfattar to kognitive element:

1. ein eksperimentell metodologi.
2. eit repertoar av døme samla gjennom eigne røynslar.

Eksperimenta i Schöns metodologi er ikkje lab- og felteksperiment som me er mest vane med. Dei har meir til felles med tankeeksperimenta som Einstein var kjend for, og som me finn i litteraturen tilbake til Ørsted i 1811 (Brendel, 2004).

Arkitektar og designarar presenterer gjerne tankeeksperimentet i perspektivskisser og planteikningar. Psykologen i Schöns døme eksperimenterer med hypotetiske forklaringsmodellar som er uttrykt som narrativar. Programutviklarar kunne bruka klassediagram og andre formar for diagram (med eller utan den strenge syntaksen frå t.d. UML). I tankeeksperimentet kan ein testa mange ulike variasjonar i eit konsept på kort tid.

Det som er vesentleg for objektiviteten i prosessen i fylgje Schön, er for det fyrste at ein utbroderer konsekvensane av kvar variasjon i tilstrekkeleg detalj til å kunna sjå inkonsistensar og andre negative og positive fylgjer. Når ein føresleg eit klassediagram i UML må ein t.d. vera nøye med å definera mål og meining ved kvar klasse og samanhengane mellom dei. I tankeeksperimentet kan ein forestilla seg kva som ville skje dersom ein sidan må endre den eine klassa og kva utviklingsbehov det kunne resultera i. Høg kopling og manglande koherens vert då noko røynleg og konkret, fordi ein kan leva seg inn i *konsekvensane* for framtidig arbeid, og ikkje berre prata om arkitekturteoretiske konsept.

Dette tyder ikkje at ein må modellera alt før ein programmerer. Somme tider vert tankeeksperimentet for krevjande slik at ein treng ein prototyp for å vurdere konsekvensane. Vidare er metodikken like relevant ved utviding og refaktorering som ved opprinneleg design.

Denne eksperimentelle metodologien kan lærast (Schön, 1983; Schön, 1987). Kunsten er å utdjupa den tenkte situasjonen i slik detalj at ein verkeleg kan leva seg inn i fylgjene, og leva seg fullt og heilt inn i forestillinga *som om* den tenkte situasjonen var verkeleg. Det er vanskelegare å sjå korleis eksperten vel kvart trekk, for her byggjer han på heile porteføljen av døme som er samla gjennom ein lang karriere.

¹Ein epistemologi for *techne* høyrer ut som ei sjølvmotseiing i det greske vokabularet som Aristoteles brukte. Me kan verta freista til å kalla det ein *technologi*, men diverre, dét ordet er allereie teke til ei anna meining.

Taus kunnskap

Mykje av kompetansen som Schön observerer hos profesjonsutøvaren er taus. Omgrepet «taus kunnskap» vart introdusert av Polanyi (1983), som observerte at *me kan vita meir enn me kan fortelja*. Sjølv om me kan kjenna igjen eit andlet blant millionar, tyder ikkje det at me kan skildra andletet me kjenner att, eller forklara kva me ser etter. Det motsette av taus kunnskap vert gjerne kalla *proposisjonell kunnskap*, dvs. kunnskap som kan formulerast i ord, gjerne i logiske utsegner (proposisjonar).

Hendrix (1947) viste at taus kunnskap, som ho kalla *unverbalized awareness*, enklare let seg overføra til nye problem enn proposisjonell kunnskap. Ho studerte matematikkdiridaktikk på barneskulenivå, og observerte at elevar som lærte å rekna gjennom oppgåver og døme, vart flinkare til å overføra kunnskapen enn dei som lærte generelle reknereglar fyrst. Ved å starta med oppgåver og døme studerer ein enkeltting og byggjer opp ein form for *techné*. Dei som lærer generelle reglar fyrst, får dei universelle sanningane (*episteme*) som proposisjonell kunnskap. Hendrix viste vidare at det skader forståinga om den tause kunnskapen for tidleg vert gjort proposisjonell.

Sjølv om den tause kunnskapen vert opparbeidd gjennom å studera enkeltting, so handlar han ikkje om detaljkunnskap om enkelttinga. Elevane i Hendrix sin studie etablerer ein generell rekneteknikk sjølv om dei ikkje kan setja ord på han, og der er ingen grunn til å tru at dei hugsar dei einskilte oppgåvene dei har vore gjennom. Polanyi (1983) viser òg at inngåande studie av detaljane kan øydeleggja heilskapsforståinga som ligg i den tause kunnskapen.

Den tause kunnskapen er ein svart boks, der me berre kan observera handlingar og skaparverk som kunnskapen leier til. Der kan godt vera generelle mekanismar inni boksen, men me manglar orda til å forklara mekanismane. Taus kunnskap kjem til kort når me ynskjer å forvissa oss om at universelle utsegner er sanne. Enkelttinga kan me observera og studera i verda rundt oss. Universalitet er derimot ein abstraksjon som me konstruerer og uttrykkjer gjennom ein eller annan form for språk. Vitskapleg kunnskap, eller *episteme*, er difor proposisjonell.

4 Danning og utdanning

Utdanningsfilosofien skil gjerne mellom *danning* og *utdanning*. Dette skiljet finst på tysk og alle dei skandinaviske språka, men ikkje på fransk og engelsk der *education* dekkjer bae tydingane (Kemp, 2013). Danningsaspektet har ein tendens til å verta fortrent i utdanningssystemet. I fylgje Kemp (2013, s. 180f) er skiljet mellom danning og utdanning det same som mellom forståing og forklaring:

Forklaring skulle gælde det gentagelige og lovmæssige i erkendelsen, mens forståelse skulle gælde de individuelle éngangsfænomener, der ikke kunne begribes med nogen kausalforklaring.

Her kjenner me att skiljet mellom den vitande og den overvegande sinnsdelen frå Aristoteles. Det går an å vita dei gjentakelege og lovmessige forklaringane. Den overvegande sinnsdelen kan derimot forstå enkeltting der forklaringane kjem til kort. Kemp (2013, s. 181) skriv vidare at

uddannelsens indhold er en viden, der kan oplagres og gemmes væk, hvis ingen har brug for den, og tages op og anvendes, når der bliver behov for det.

Ordet «viden» er den alminnelege omsetjinga av *episteme*, og det er naturleg å lesa han slik at *techne* fell utanfor. Den tause kunnskapen fell tydeleg utanfor, sidan han ikkje kan lagrast opp og gøymast som proposisjonell kunnskap.

Det er danninga som tek for seg den tause kunnskapen, forståing og *techne*. Kemp (2013, s. 181) skriv at danninga «må ses som en historisk proces med utallige momenter av éngangsfænomener» og gjev berre meining om «den utvikles, bæres og overtages af mennesker, der gøres og gør sig dannede». Kemp meiner altso at *techne* ikkje berre vert utvikla av kvar enkelt gjennom røynsler, slik som Aristoteles såg ut til å meina. Me utviklar *techne* (òg) gjennom ein historisk prosess, eller fagtradisjon, som spenner generasjonar. Ein generasjon kan overføra taus kunnskap til neste, men berre gjennom direkte kontakt mellom menneske.

Utdanning må gjerne inngå som ein del av dannelsingsprosessen (Kemp, 2013, s. 181). Når sinnet skal overvega løysingar på eit konkret problem, kan vitenskapleg kunnskap gjerne vera ein del av vurderingsgrunnlaget, ved sidan av taus kunnskap. Som Ricoeur skriv må ein somme tider forklara meir for å forstå betre (Kemp, 2013, s. 182). Det er derimot ikkje gode forklaringar og *episteme* som kjenneteiknar dei dyktige studentane som er klare for kritiske oppgåver i arbeidslivet etter studiet.

Læring som etterlikning

Både Schön (1987) og Kemp (2013) dreg fram etterlikning som læringsmetode for å tilegna seg den tause kunnskapen. Studenten etterliknar meisteren. Dette er, seier dei begge, kontroversielt på eit vis. Ingen liker etterlikningar. Der er noko billig og mindreverdig ved ein kopi. Kva er då bra med etterlikning?

Etterlikning er ikkje det same som etteraping (Kemp, 2013, s. 205). Den etterlikninga som me søkjer er den skapande etterlikninga (Kemp, 2013, s. 200), som Aristoteles (1902) kalla *mimesis*, eit omgrep som er mykje brukt i filosofihistoria. Schön (1987, s. 108) omtalar etterlikninga (*imitation*) som ein selektiv konstruksjon. Når me etterliknar, vel me ut det som er vesentleg for oss og konstruerer det på nytt med vårt eige uttrykk.

Den som aper etter meisteren liknar likevel ikkje meisteren, for meisteren er original. Innan sveinen aper etter er meisteren gått vidare. Utfordringane som sveinen skal løysa er ikkje dei same som meisteren løyste. Som Schön fortel oss er kvar profesjonelle utfordring unik. Sveinen som blindt kopierer meisteren tek ikkje den unike situasjonen på alvor. For å gjera ein god jobb må sveinen *tenkja* som meisteren tenkte og og tolka førebiletet inn i si eiga konkrete og unike situasjon. Tankesettet kan vera ei etterlikning, men løysinga som han finn kan ikkje vera det. Løysinga skal vera sær eigen for si eiga tid og stad.

Når me ser etterlikninga som som ei kognitiv etterlikning, ser me òg langt forbi mykje av kritikken mot mimesistanken, der ein har fokusert på *mimesis* som etterlikning av fysiske handlingar og andre ytre faktorar². Etterlikninga er ikkje ei sovepute for å sleppa å *tenkja* sjølv, for det er nettopp i tanken at ein skal etterlikna.

Live-koding som undervisningsform

Innanfor programutvikling ser me fleire førelesarar som har innført det som dei kallar *live-koding* som førelesingsform. Førelesaren løyser då ei programmeringsoppgåve frå kateteret, med skjermvising på prosjektor. Han arbeider som regel utan manuskript

²For ein gjennomgang av kritikken mot mimesistanken, kan ein sjå Kemp (2013).

og det er ikkje på førehand opplagt kva løysing som vert vald ved kvar krossveg. Det er tvert imot viktig at studentane ser at tvil og overveging er ein naturleg og uunnverleg del av prosessen. Studentane vert gjerne inviterte med på råd.

Det interessante ved *live*-koding er at det viser fram kunnskapsformar som elles sjelden kjem til syne i undervisinga. Studentane vert vitne til meisteren sin kunnskap-i-handling, hans *techne*. Meisteren formidlar prosess heller enn produkt, og studentane vert inviterte til å etterlikna tankeprosessen meir enn sluttproduktet. Når meisteren må overvega ulike designval, kan han formidla at der ikkje finst nokon fasit. Svara han kjem fram til, kunne gjerne ha vore annleis. Ulikt meir teoretiske fag, må studentane søkja etter noko anna enn *episteme*.

Der er ein underliggjande fare i all meisterlæring, for at meisteren opptrer so meisterleg at prosessen vert skjult (Schön, 1983, s. 104), berre resultatet vert synleg. Me ser ofte studentar som trur at læraren forventar at det *veit* svaret. Når det ikkje er mogleg å vita, gissar dei heller enn å overvega. Schön (1987, s. 322) legg vekt på at tradisjonelle disiplinar må undervisast slik at utforskingemetodikken vert synleg. Berre då kan studentane etterlikna prosessen heller enn produktet. Dette kan ein oppnå i *live*-koding med tilstrekkeleg tålmod og omtanke.

Praktisk kompetanse og aktiv læring

Sokalla moderne undervisningsformar legg stor vekt på aktiv læring og praktisk kompetanse. Stundom vert det framstilt som om aktiv læring automatisk gjev praktisk kompetanse. Den føregåande drøftinga gjev derimot grunnlag for å problematisera dette inntrykket.

Den praktiske kompetansen er overvegande og ikkje vitande. Studenten som vert sett til å rekna aritmetikk eller implementera kjende algoritmar, er absolutt aktiv, men han arbeider med problem der me *veit* løysinga. Der kan vera fleire alternative løysingar (t.d. ei rekursiv formulering og ein løkkeimplementasjon), men i dei fleste tilfelle går oppgåva ut på bruka éi løysing for å hugsa ho. Inga overveging trengst for å løysa oppgåva. Studenten endar opp med ny *episteme*, han hugsar ei løysing som (i essens) ikkje kunne vera annleis.

Motsett ser me at *live*-koding ikkje er meir student-aktivt enn (andre) tradisjonelle førelesingar. Likevel er det overvegande kompetanse, *techne*, som vert formidla, i langt større grad enn kunnskap. Studenten får sjå korleis førelesaren tenkjer og overveg i ein praktisk situasjon, og kan etterlikna tankesettet.

Dette skiljet mellom *techne* og *episteme* kan moglegvis kasta nytt lys på eksisterande kontroversar innanfor aktiv læring. Det er ei kjend sak at empirien ikkje gjev eintydig støtte til aktiv læring. Mayer (2004) er ein særleg fyrig kritikar, medan Prince (2004) peiker på blanda resultat når ein ser på fleire ulike læringsutbyte og viser til eit døme frå medisin. Studiar tyder på at prosjektbasert læring (PBL) gjev ei viss forbetring på klinisk kompetanse og ei viss forverring på standardiserte eksamenar. Prosjektbasert læring er ein læringsaktivitet som tek utgangspunkt i enkelttinga, og det er rimeleg å tru at ein då trenar dei overvegande evnene som ein treng i klinisk praksis. Dersom dei standardiserte eksamenane testar kunnskap om ting som ikkje kunne vera annleis, so er det kanskje ikkje overraskande om dei fordrar læringsaktivitetar som fokuserer på generell, proposisjonell kunnskap. Kan henda er styrken og ulempa ved PBL slett ikkje studentaktivitet men fokuset på *techne* over *episteme*.

Hendrix (1961) drøfta tre ulike læringsformar som alle kunne gå under nemninga

discovery learning. Den induktive metoden bruker oppgåver og døme, gjerne i ein nøye planlagd sekvens, med sikte på at studentane sjølv skal oppdaga og formulera dei generelle reglane. Undervisinga fokuserer dermed på enkeltting, medan meininga med læringa er generell og proposisjonell kunnskap. Samantreffsmetoden (*incidental method*) bruker praktiske prosjekt, der ein vonar at studentane skal støyta på og læra seg den generelle teorien ved behov. Igjen er målet generell og proposisjonell kunnskap. Den tredje metoden, som me nemnde tidlegare som *unverbalized awareness*, er den einaste der målet er taus kunnskap. Denne tause kunnskapen er meir effektiv enn proposisjonell kunnskap, sjølv om proposisjonell kunnskap er mogleg, seier Hendrix (1947).

Debatten handlar ofte om det er best å fortelja studenten det han treng vita, eller om det er best å lata han finna det ut sjølv (Lee and Anderson, 2013). Av Schön, Polanyi og Aristoteles har me derimot lært at der finst kunnskap som me kan fortelja til studenten, og annan kunnskap som det ikkje er råd å fortelja. Når me drøftar aktiv læring må me kunna skilja mellom desse to, og me skal for all del ikkje gløyma den kompetansen som ikkje kan forteljast. På ein eller annan måte må den òg formidlast.

5 Vegene mot IT-danning

Som nemnd inngår læring av *techne* i danninga, heller enn utdanninga. Me lærer frå både Kemp (2013) og Gadamer (1960) at danning er noko som vert utvikla gjennom ein kultur- eller fagtradisjon. Utan tradisjon finst der ingenting som studenten kan danna seg inn i. Kva kan me då seia om IT-tradisjonen?

Fagtradisjonen i IT-fag

IT-faga har eit særleg problem. Fagmiljøa våre har ikkje ein felles fagtradisjon som me byggjer på, og dei tradisjonane som evt. finst vert i liten grad formidla. I staden er der mange tradisjonar og ingen semje om kva rolle kvar tradisjon skal ha.

Den gongen datamaskina var ny for rundt 70 år sidan var det naturleg at utviklinga kom i matematikk- og elektronikkmiljø. Ein trengte elektronikingeniørar for å byggja og vedlikehalda maskinvaren og matematikarar til å analysa og programmera samansette aritmetiske prosessar. Det er difor ikkje overraskande at me 30 år seinare fekk informatikkinstitut skilt ut frå matematikkinstitut og dataingeniørstudium skilt ut frå elektrostudium.

Allereie på 1960-talet stod datamaskina fram som eit universelt verkty som alle fag har bruk for. Ved Universitetet i Bergen var det so tydleg at ein gjekk inn for eit datastudium ved Historisk-Filosofisk Fakultet (Bagge, 1996). Svein Nordbotten vart utnemnd til professor i informasjonsvitskap frå 1971. Då hadde fakultetet nettopp vorte delt, og sjølv om informasjonsvitskapen var tenkt å tena både humaniora og samfunnsvitskap, var prof. Nordbotten sosiolog og faget vart fyrst forma i fagtradisjonen hans. Sidan kom der eit eige studium i humanistisk informatikk.

Informasjonsvitskap er rett nok særlege for Bergen, og mange tilsette som sjølve kjem frå informatikktradisjonen har vore med å (om)definera kva informasjonsvitskap skal vera i dag. Situasjonen illustrerer likevel at datafag kan byggja på meir enn éin tradisjon. Andre plassar finn ein datafag med røter i handelsfagstradisjonar, lingvistik og andre fag.

Studieprogramma i datafag vert drege mellom alle desse ulike tradisjonane. Ikkje berre har me den same konflikten som Schön har peikt på, der studentane

søkjer danning innanfor profesjonstradisjonen medan undervisarane står i ein vitskapstradisjon. I tillegg er det uklart kva vitskapstradisjonar som kan vera relevante for profesjonen.

Svært mange av dei som underviser og datafag i dag har sine røtene sine i den matematiske informatikktradisjonen. Dei har lært den logisk-deduktive forskingsmetoden frå matematikken, og forstår faget ut frå det. Når ein utviklar komplekse informasjonssystem i industrien, er der mange andre relevante fag og metodeverk. Frå statsvitskap og handelsfag kan ein henta teoriar for organisasjonen og prosessen som informasjonssystemet skal støtta oppunder. I sosiologien finn ein metodikkar for å evaluera systemet i bruk.

Mange studieprogram søkjer å formidla kompetanse frå alle desse faga, men det er slett ikkje alltid dei har undervisarar som sjølv er danna innanfor den same fagtradisjonen. Dersom det berre hadde vore *episteme* dei skulle formidla, kunne dette kanskje ha gått, fordi episteme kan skrivast ned og hentast fram igjen i ein ny kontekst. Når me derimot skal formidla *techne* og taus kunnskap, må me vera bevisste på den rolla som kvar tradisjon skal ha. Det gjeld både dei ulike akademiske tradisjonane og profesjonstradisjonen.

Ein risiko som ein lauper når tradisjonen vert gløymt er at dagens sanningar, eller *best practice*, framstår som evige sanningar, og ikkje som det dei er, nemleg eit tilfeldig punkt på ei lang utviklingsline. Studenten kan lett misforstå eit resultat av *techne* som *episteme*. Me finn eit døme i smidige metodar. Hohl et al. (2018) peiker på skilnaden på «being agile» og «doing agile». Det er lett å læra seg ein av dei kjende smidige metodikkane, og gjera seg smidig etter boka. Når ein les historia bak smidige metodologiar ser me derimot ein reaksjon mot nettopp boka. Den som *er smidig* frigjer seg frå boka og ser på enkelttinga som det dei er. Difor har me òg fått mange smidige metodologiar som har oppstått i litt ulike historiske kontekstar. Kjennskap til variasjonane i tradisjon bidreg til repertoaret av døme som Schön viser til, og er difor nyttig for å utvikla og velja metodikk til konkrete prosjekt i framtida.

Eksamen som ein turingtest

Eit anna perspektiv på kunnskap kan me få gjennom forkinga på kunstig intelligens. Debatten om kva kunstig intelligens er, har pågått i fleire mannsaldrar. Kva krev me av ei tenkjande maskin?

Eit av dei mest kjende kriteria for ein tenkjande maskin er turingtesten, introdusert i 1950 av Turing (2009) som eit tankeeksperiment. Ein intervjuar stiller spørsmål til eit menneske og til ei maskin, utan på førehand å vita kven som er maskina. Maskina består testen om han kan imitera mennesket utan at intervjuaren kan sjå forskjell. Ei slik maskin må seiast å vera tenkjande.

Lat oss no snu dette på hovudet. Kva krev me av ein tenkjande student? I ei tid der mange yrke vert erstatta av robotar, er det klart at ein student som ikkje kan meir enn ei maskin på eksamen har dårlege sjansar i arbeidslivet. Dersom eksamen ikkje er egna til å skilja menneske frå maskin, vert ikkje studenten vurdert på kompetanse som han treng i yrkeslivet. Ein eksamen som ikkje fungerer som turingtest er dermed irrelevant for vurderinga av avgangsstudentar. Kva er det i utdanninga som me ynskjer å læra studentane og som me ikkje kunne ha lært til ei maskin?

Datamaskina utmerker seg på proposisjonell og generell kunnskap. Alt maskina bruker av program og modellar er i prinsippet proposisjonelle og kan skrivast

ut som logiske proposisjonar. Datamaskina får aldri del i den tause kunnskapen, fordi kunnskap som vert programmert per definisjon vert proposisjonell. Dersom studentane våre skal ha ein plass i arbeidslivet, må styrken deira liggja i den tause kunnskapen og i forståinga av enkelttinga, altså *techne*.

Me har tidlegare kritisert matematikkundervisinga for å utdanna *computers* (Schaathun and Moe, 2019). På den tida, rundt 1950, då Turing skreiv om turingtesten, var nettopp dét ein vanleg fyrste jobb for unge ingeniørar. Kunnskapen dei trong som *computer* var proposisjonell kunnskap, om samanhengar som ikkje kunne vera annleis (i alle fall i stor grad), men teknologien den gongen var ikkje avansert nok til å automatisera alt som var proposisjonelt. I dag kan maskinene meir, men matematikkpensum har til samanlikning endra seg lite i den same perioden. Når me ser på matematikkeksamenar på ingeniørstudia, so er der få oppgåver som testar evner som ikkje maskinene kan gjera betre enn studentane. Kvifor er det slik?

Turingtesten er stadig eit tankeeksperiment. Opprinneleg handla eksperimentet om kva maskina kan gjera i framtida. No handlar eksperimentet vel so mykje om kva plass mennesket har når maskinene vert stadig meir avanserte. Dette er ein viktig tanke i studie- og læringsdesign. Kva lærer studentane som gjer dei betre enn maskiner? Korleis vurderer me denne ikkje-maskinelle kompetansen på eksamen?

Sjølv om det varierer mykje mellom fag og emne, er der skremmande mange eksamenar som kan løysast maskinelt. Det er eit paradoks som krev ein kritisk diskusjon.

6 Konklusjon

Både Aristoteles og Schön fortel oss at den skapande kompetansen byggjer på forståing om enkeltting. Vitskapleg kunnskap fokuserer på universelle sanningar. Det er kunnskap om ting som ikkje kan vera annleis. Slik kunnskap kan vera naudsynt, men det er ikkje tilstrekkeleg for å løysa praktiske problem, og me kan trygt overlata det meste av vitande kompetanse til datamaskiner.

Studentane treng kunnskap og forståing for enkelttinga, som dei møter i praksis, og det er viktig å understreka at det er noko meir enn praktiske øvingar og aktiv læring. Gjennom praksis kan me òg trenna *episteme*. Dersom me skal formidla taus kunnskap må me ha eit tydleg fokus på å forstå enkelttinga og løysa problem som ikkje er dekte av generelle teoriar.

Skiljet mellom vitande, generell og proposisjonell kunnskap på den eine sida, og taus kunnskap og forståing for enkelttinga på den andre, reiser fleire brennaktuelle spørsmål rundt utdanningane våre.

- Lærer me studentane våre noko meir enn datamaskinene kan?
- Kva for ein taus kunnskap treng me å formidla?
- Korleis legg me til rette for formidling av taus kunnskap?
- Korleis vurderer me den tause kunnskapen på eksamen?

Ingen problem er løyst i denne artikkelen, men me vonar at drøftinga og kartleggjinga av dei filosofiske konsept, kan inspirera til ein ny diskusjon rundt (ut)danningsopplegget.

Me skal ikkje hevda at den tause kunnskapen manglar heilt i utdanningane, men det er sjelden tydleg kva taus kunnskap studentane skal ha, korleis han vert vurdert eller korleis han vert formidla. Særleg kan me uroa oss over dei mange studentane som freistar å lesa på eiga hand. Dei finn ikkje den tause kunnskapen i bøkene, men

likevel gjer mange av dei det godt på eksamen. Her er der mange opne spørsmål for framtidig gransking.

Referansar

- Aristoteles. *Poetics*. London. MacMillan and Co., Ltd., 1902.
- Aristoteles. *Etikk*. Gyldendal Akademisk, 1999. Omsett av Anfinn Stigen.
- Sverre Bagge. Samfunnsvitenskapenes historie. In *Universitetet i Bergens historie*, volume 2. Universitetet i Bergen, 1996.
- Elke Brendel. Intuition pumps and the proper use of thought experiments. *Dialectica*, 58(1): 89–108, 2004. doi: 10.1111/j.1746-8361.2004.tb00293.x. URL <https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1746-8361.2004.tb00293.x>.
- Hans-Georg Gadamer. *Wahrheit und Methode*. Tübingen, 1960.
- Gertrude Hendrix. A new clue to transfer of training. *The Elementary School Journal*, pages 197–208, 1947.
- Gertrude Hendrix. Learning by discovery. *The Mathematics Teacher*, 54(5):290–299, May 1961.
- Philipp Hohl, Jil Klünder, Arie van Bennekum, Ryan Lockard, James Gifford, Jürgen Münch, Michael Stupperich, and Kurt Schneider. Back to the future: origins and directions of the “agile manifesto” – views of the originators. *Journal of Software Engineering Research and Development*, 6(1):15, November 2018. ISSN 2195-1721. doi: 10.1186/s40411-018-0059-z. URL <https://doi.org/10.1186/s40411-018-0059-z>.
- Peter Kemp. *Verdensborgeren: pædagogisk og politisk ideal for det 21. århundrede*. Hans Reitzels Forlag, 2. reviderede udgave edition, 2013.
- Hee Seung Lee and John R Anderson. Student learning: What has instruction got to do with it? *Annual review of psychology*, 64:445–469, 2013.
- Richard E Mayer. Should there be a three-strikes rule against pure discovery learning? *American psychologist*, 59(1):14, 2004.
- Wilbert E. Moore. *The professions: roles and rules*. Russel Sage Foundation. New York, 1970.
- Don Norman. *The design of everyday things: Revised and expanded edition*. Constellation, 2013.
- Kai A. Olsen. Låst i papirets byråkrati. *Bergens Tidende*, September 2014. <https://www.bt.no/btmeninger/kronikk/i/m5k3L/laast-i-papirets-byraakrati>.
- Michael Polanyi. *The Tacit Dimension*. Peter Smith, Gloucester, Massachusetts, 1983. First published Doubleday & Co, 1966.
- M. J. Prince. Does active learning work? a review of the research. *Journal of Engineering Education*, 93(3):223–231, 2004.
- Hans Georg Schaathun and Jan Gunnar Moe. Kva er egentleg målet i matematikken? In Reidar Lyng, editor, *MNT-Konferansen*, Nordic Journal of STEM Education. March 2019. <https://www.ntnu.no/ojs/index.php/njse/article/view/2992>.
- Donald A Schön. *Educating the reflective practitioner*. Jossey-Bass San Francisco, 1987.
- Donald A. Schön. *The Reflective Practitioner*. Ashgate Arena, 1983.
- Herbert A Simon. *The sciences of the artificial*. MIT press, 1996.
- Alan M. Turing. *Computing Machinery and Intelligence*, pages 23–65. Springer Netherlands, Dordrecht, 2009. ISBN 978-1-4020-6710-5. doi: 10.1007/978-1-4020-6710-5_3. URL https://doi.org/10.1007/978-1-4020-6710-5_3. Annotated version of the paper from the British journal *Mind* 1950.