



UiO : **Institutt for pedagogikk**
Det utdanningsvitenskapelige fakultet

Computational Thinking in Collaborative Learning and Design

Anders Mørch
Inst. for pedagogikk (IPED)
Universitetet i Oslo
andersm@uio.no

SDU, Kolding, 8. mars 2022

www.uv.uio.no/iped



Innhold

- Ch2: CT i et idéhistorisk perspektiv
 - Is CT an emergent phenomena or skill to be taught?
- Ch8: Algoritmisk dannelse
 - Specific and general knowledge and skills Integrated?
- Ch11: Bodygramming
 - Statiske vs. dynamiske representasjoner av CT i progr.?
- Collaborative learning and design of tools
 - ProSkap project, example from our work on learning CT

Ch2: CT – idehistorisk utvikling

- Tre ulike bakgrunner til CT
 - Logikk
 - Utdanningsteori
 - Teknologifilosofi
- Relevante idehistoriske diskusjoner
- Utfordringer og muligheter for CT i dag og fremover
 - Skal CT forstås som del av naturvitenskap?
 - Skal CT forstås som del av design-vitenskap?
 - Skal CT kunne forklares enkelt eller skal det være kompleks?
 - Vil CT være like relevant om 5 eller 10 år?

Logikk-perspektivet

- Vannevar Bush og Claude Shannon var visjonære
- Boolsk logikk er viktig del av programmeringsspråk (software) og elektroniske kretser (hardware)
- Hansen spør om menneskets tenkning også er en type mekanisk prosess, og refererer til bevissthetsfilosofer som har forsket på dette
- Spørsmål for CT fra dette perspektivet?
 - Bør man abstrahere logikk til et høyere nivå for at skjule den eller bør man forstå logikk som en sentral byggekloss i utviklingen av programmeringsspråk og digital teknologi?

Utdanningsteori-perspektivet

- Higher-order thinking skills (f.eks. Vygotsky)
- Thinking frameworks (gjennomgangstema i kap.)
- Ulike teoretiske perspektiver for å forstå tenkning
 - Konstruksjonisme (Papert) og sosio-kulturelle perspektiver (Vygotsky, Wegerif)
 - Konnektivism (nettverksbaset tenkning)
 - Kognisjon som rammeverk
- Hvordan kan teoretiske rammeverk anvendes i (empirisk) forskning?

Teknologifilosofi-perspektivet

- Dette perspektivet diskuterer relasjonen mellom mennesket og redskap
- Heidegger er sentral filosof i dette perspektivet, da han er opptatt av både tenkning og teknologi
- Papert er sentral forsker ved å anvende teknologien i utdanning
 - Using the computer to program the child vs. enabling the child to program the computer
 - Det første er knyttet til computer-aided instruction (CAI)
- Det andre har ledet til Making and Making culture

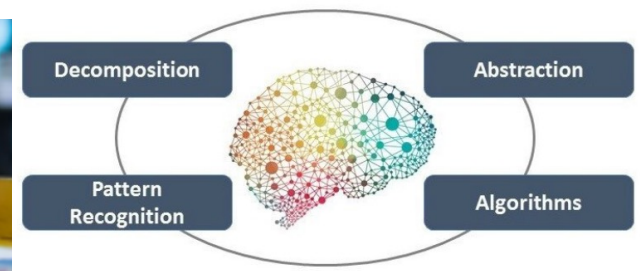
Nåtidens forståelse av CT

- Denne er preget av J. Wing's definisjon og popularisering, som noen mener "utvanner" og forenkler kompleksitet (e.g., diSessa, 2022)
- Wing ønsker å gjøre CT til et "all-emracing framework", der "C"-delen og "T"-delen er integrert
- Flere er uenig i dette, inkl. Børsen Hansen
- Papert brukte begrepet "objects to think with," der er CT adskilt i "C-" og "T"-delen

Dilemma 1 om CT forståelse

- H.A. Simon (en sentral aktør ved CMU som Wing kjente til) gjorde et skille mellom to vitenskaper:
 - Naturvitenskapen: “How things are” (past and present)
 - Design-vitenskapen: “How things ought to be” (present and future)
- Skal CT knyttes til naturvitenskapen eller design-vitenskapene (design science)?
- Det er avhengig av om fokus er “C” eller “T” delen!
 - C → design-vitenskap;
 - T → naturvitenskap

Dilemma 2



- Skal CT være enkel å forstå (abstrahere underliggende logiske, teknologiske og teoretiske bestanddeler)
- Eller skal CT være et kompleks system og trekke på og synliggjøre den utviklingen den er del av?
- Pr. Idag så er det den “enkle” varianten (dvs. Wing sin variant) som har gått seirende ut, men hvor lenge vil den vare?
- *Jeg spurte 25/1*: Er CT som redskap primært abstrakt eller konkret, eller en kombinasjon?

Ch8: Algoritmisk dannelse

- Her sees CT i sammenheng med kunstig intelligens (AI), gjennom historiske perspektiv
- Spesielt problemet med mangel på transparens i å forstå AI algoritmer i interaksjon med teknologi
- Gerde tar kritisk stillingstagen til siktede og utilsiktede virkninger av IT i dette kapittel
- Mestring av teknologi i vår digitaliserte tidsalder, er viktig som del av utdanningen

Anvendelse av CT til kritisk tekning

- Anne Gerdes tilnærming er å anvende CT til kritisk refleksjon om design og bruk av IT gjennom begrepene til CT, f.eks. (se foil 9)
- AI feltet fordrer at man stiller etiske spørsmål, hevder hun
- F.eks. knyttet til bruk av AI i medisinsk behandling og utdanning (learning analytics), som er til tider kontroversiell

Algoritmisk dannelse

- Tilegnelse av tekno-moralske dyder, som setter oss i stand til å navigere i komplekse og foranderlige teknologi-medierte sammenhenger

Tekno-moralske dyder: To typer

- Algoritmisk dannelselse krever forståelse av AI gjennom etisk stillingstagen, som kan lede til:
- Samfunnsmessige dyder
 - Ydmyket, høflighet, moralsk perspektiv, fleksibilitet, rettferdighet, m.fl.
- Psykologiske dyder
 - Omsorg, selvkontroll, empati, ærlighet, m.fl.
- *Spørsmål*: Kan man lære disse dyder uten å gå veien om CT? Uten å bruke AI?

Dilemma 1: AI vs. IA

- Autonomous AI og IA (intelligence augmentation)
 - *AI eksempel*: Selvkjørenede biler (mennesket er lite involvert gjennom interaksjon)
 - Strong AI
 - *IA eksempel*: Netflix anbefalinger (recommender systemer, mennesket er mye involvert gjennom interaksjon)
 - Human-centered AI

Dilemma 2: AI vs. maskinlæring

- Muligheter og utfordringer for AI transparens
- Regelbasert AI (klassisk/symbolic AI) er IT program som består av regler som kan enklere overfører til algoritmer som er forståelig for mennesker
- Maskinlæring, derimot, er mer kompleks da løsninger skapes nedenfra-og-opp, fra data gjennom ulike abstraksjons-lag, og algoritmene bruker mye matematikk, inkl. statistikk, lineær algebra, og sansynlighets-regning

CT as emergent phenomena or skill to be taught?

- Definisjonen av CT i boka (ch. 1, Bonderup Dohn)

Vi undersøker i denne antologi Computational Thinking (CT) med afsæt i følgende definition fra Dohn (kapitel 1): *Computational Thinking er de kognitive processer, som er involveret i udviklingen af it-artefakter og programmer til at leve i verden i dag.*

- Hvordan tolke dette i forhold til emergent vs. instruksjon?
- Mao. hva kommer først:
 - Utvikling av IT-artefakter eller CT prosesser?

Specific vs. General Skills' Practices

- I vår forskning er vi opptatt av to type ferdigheter
 - Fagspesifikke, f.eks. Naturfag, Engelsk, Historie
 - Generiske, f.eks.: samarbeid, kreativitet, problem-løsning, kommunikasjon (her kan passe CT skills?)
- Generiske ferdigheter brukes på tvers av fag, og får gradvis mer betydning i skolen i Norge
 - Demokrati og bærekraft (klima) er tilsvarende fagovergripende ferdigheter på pensum i skolen

TIMEPLAN

Klotteslett	MANDAG		TIRSDAG		ONSDAG (- Sivert)		TORSDAG (- Hilde)		FREDAG (- JB 1130 – 1500)	
0830-0900	Frokost m/ tema		Frokost m/ tema		Frokost m/ tema		Frokost m/ tema		Frokost m/ tema	
0900-1015	ENGELSK (1st)	MATTE (1st)	KRLE (1st)	NORSK (1st)	M&H (1st)	NORSK/ UTDANNINGSVALG (1st)	MATTE (1st)	SAMFUNN (1st)	MATTE (1st)	SAMFUNN (1st)
1015-1130	MATTE (1st)	ENGELSK (1st)	NORSK (1st)	KRLE (1st)	M&H (1st)	NORSK/ UTDANNINGSVALG (1st)	SAMFUNN (1st)	MATTE (1st)	SAMFUNN (1st)	MATTE (1st)
1130-1200	Lunsj		Lunsj		Lunsj		Lunsj		Lunsj *	
1200-1300	K & H (1st og 1st)		ENGELSK (1st)	NATURFAG (1st)	M&H (1st)	LINK/ARBEIDSEFFE/ BEGREPSLÆRING (1st)	VALGFAG (1st)	NATURFAG (1st)	GYM/AKTIVITET (1st og 1st)	
1300-1400	K & H (1st og 1st)		NATURFAG (1st)	ENGELSK (1st)	LINK/ARBEIDSEFFE/ BEGREPSLÆRING (1st)		NATURFAG (1st)	VALGFAG (1st)	GYM/AKTIVITET (1st og 1st)	

Ch11: Kroppsbasert CT

- Didaktisk design med kroppsbasert CT
- Kalles også for unplugged programmering
- Nye måter å forstå abstraksjon på som er enklere enn å programmere for nybegynnere å unge barn
- Hypotesen til Mikkonen er at kroppslige aktiviteter kan understøtte læring av abstrakte prosesser, som er interessante å utforske
- Kan sammenlignes med interaksjon med dagligdagse ting slik som har sensorer i f.eks. kjøleskap

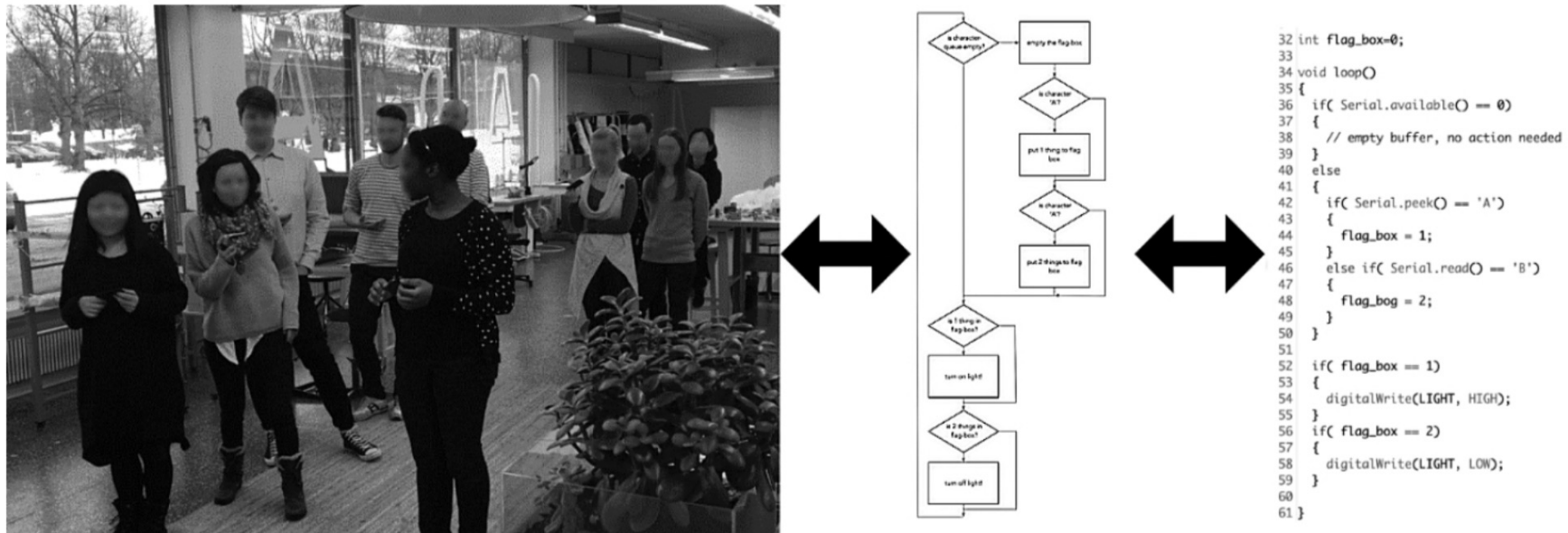
Didaktisk design

- Målet med didaktisk design av CT er å utvikle en forståelse (undervisningsopplegg) for algoritmiske prosesser
- To diaktiske design foreslås av Mikkonen, Bodygramming og The Blind Processor
- Målet er å lære nybegynnere å programmere og forstå abstrakte begreper i CT
- Disse to design har blitt benyttet på gymnas og universitets-nivå

Simulering av IT prosesser

- Studentene skal forklare prosessoren, og det må man abstrahere de fysiske komponenter til et høyere abstraksjonsnivå
- Komplekse systemer som modell for et interaktivt (IT) system
- Ulike abstraksjonsnivå må defineres
- Representasjonskløften mellom kode og adferd
- Statiske vs. dynamiske representasjoner,
 - Diagrammer av kode (statisk) vs. storyboard av adferd (dynamisk)

Bodygram eksempel (s. 301)



Figur 11.1. Ækvivalenter til Bodygramming i flowdiagrammer og i kode.

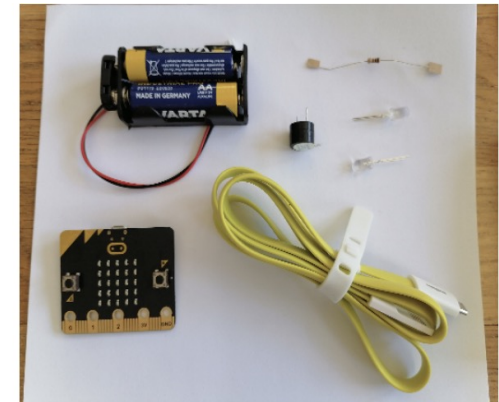
- Bruk av kroppen, trinn-for-trinn utførelse av instruksjoner, som sekvens av et eksekverende (dynamisk) IT program; dette kan lede til læring av CT

Computational Thinking (generic skills) in Collaborative Learning with Tools

- When CT is integrated in other subjects, such as
 - Math, science, music, or arts & crafts (sløjd/håndværk)
- What do the pupils learn about CT in these contexts?
- To what extent does CT work as method to teach and learn domain knowledge?

Programming in a Makerspace (CT i STEM fag gjennom programmering)

- ProSkap project in Oslo (UiO + OsloMet)
- RQ: What characterizes block-based programming (MakeCode/Micro:bit) as an explorative design space to teach STEM topics in an online collaborative setting?
 - 43 pupils, 12-15 years old
 - Focus on integrating programming in school subjects: science, math and physics



Empirical research: Methods

- Data collection
 - Virtual ethnography (Hine, 2000)
 - Screen recordings via zoom
 - Total hours of screen recording: 70
 - Participant observation
 - Interview via Zoom
 - N=23
- Data analysis
 - Thematic analysis
 - Interaction Analysis (Jordan & Henderson, 1995)
 - Visual artifact analysis (method we are developing)

Empirical Data Example

Discursive object →

Table 3. Simulating a die.

Turn	Person	Utterance	Analytic Concepts
1.1	Sophia	It is just like... Can the Microbit be used as a die?	Argumentation base
1.2	Olivia	Nothing more than that?	
1.3	Sophia	No, that is all! ...	
1.4	Olivia	Should we try making a die then, or what?	Negotiation
1.5	Liam	Okay. Mm. We could actually just do what we did earlier.	Example (idea reuse);
1.6	Olivia	Like... On [Microbit] shake... ((pics purple block from pallet)) Then...	Design unit (1)
1.7	Liam	Then you can add “show number”... ((Olivia picks blue block and puts inside the purple block)) and a... In addition, math of course. ((Olivia goes to the math pallet and searches)) Then we have a random number ((Olivia finds the “pick random” block set to default values 0 and 10)) from 1 to 6 or 1 to 10 ((Liam sees the number 10)), but 1 to 6 ((Olivia changes the value from 10 to 6 in the innermost block)). Moreover, this should work like a die.	Rule; Information sharing; Design unit (3)
1.8	Olivia	((Tests Microbit by clicking the “shake” button 5 times)) It works! ((tone of voice is happy))	Design unit (4); Group cognition

Technology object →

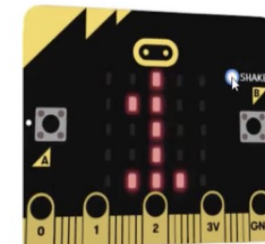
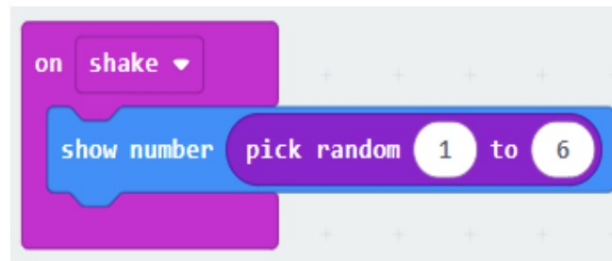


Figure 2. Visual artefacts for extract 1: A program composed of four design units that simulates a die when the user clicks the shake button. *Left:* Code after turn 1.7. *Right:* Output of program.

Preliminary Results

- Technology objects (means)
- Discursive objects (goal)
- The objects evolve in parallel and interrelate
 - Technology objects evolve by composing and later abstracting design units
 - Discursive objects evolve by merging multiple perspectives (from private to shared understanding)
 - Breakdown, problems, and surprise toggle mode
- We also developed an observation protocol

Referanser

- Andersen R., Mørch A.I., Litherland K.T. (2021). Learning Domain Knowledge Using Block-Based Programming: Design-Based Collaborative Learning. In: Fogli D., Tetteroo D., Barricelli B.R., Borsci S., Markopoulos P., Papadopoulos G.A. (Eds). *End-User Development. IS-EUD 2021*. Lecture Notes in Computer Science, vol 12724. Springer, Cham.
- Simon, H.A. (1996). *The Sciences of the Artificial. Third Edition*. Cambridge, MA: MIT Press.