


 MatteliST
 MATEMATIKKSENTERET


arm School Program

FILTRET FORSKNING, FRA CAMBRIDGE MATHEMATICS

I SAMARBEID MED MATEMATIKKSENTERET

KAFFEPRAT:

HVA SIER FORSKNING
OM TILNÆRMINGER
OG VERKTØY SOM KAN
FREMME ALGORITMISK
TENKNING I
KLASSEROMMET?

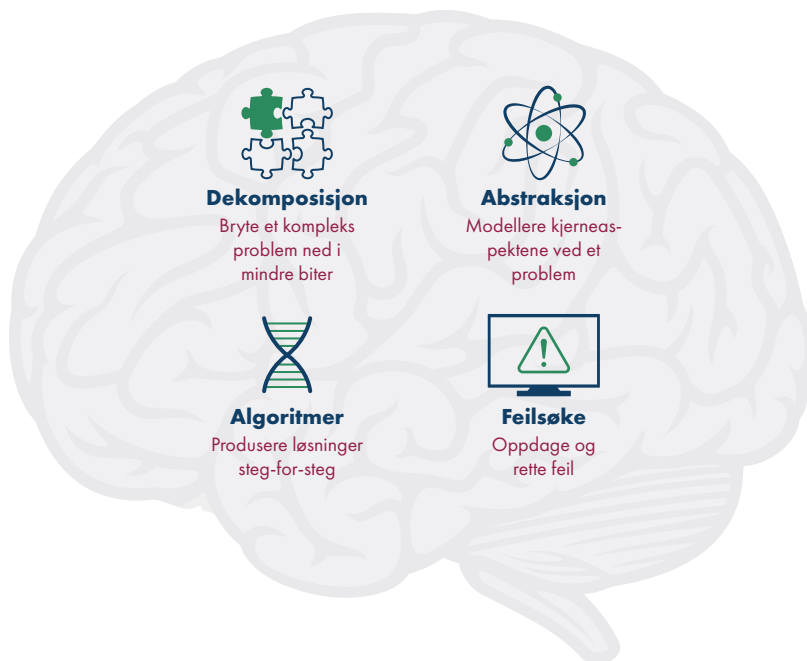
OPPSUMMERING

- AT er en grunnleggende ferdighet som er relevant i mange fag, og kan fremmes gjennom forskjellige tilnærminger og verktøy.
- Det trengs mer forskning som undersøker hvordan disse tilnærmingene og verktøyene kan brukes på en hensiktsmessig måte for å legge til rette for læring og undervisning i AT.
- Barn helt ned i fireårsalderen kan lære grunnleggende AT- og programmeringsbegreper gjennom bruk av pedagogisk programvare. Det har vist seg at en slik tilnærming også styrker forståelsen i matematikk og teknologi.
- Det er viktig å vurdere barns utviklingsstadier hvis roboter eller annen maskinvare tas i bruk for å fremme AT.
- Visuelle programmeringsspråk er gode verktøy for å introdusere grunnleggende begreper i datateknologi.
- Ved bruk av visuelle programmeringsspråk kan lærere enklere oppmuntre elevene til å dele tankeprosesser. Fokuset blir da både på tankeprosesser og på resultat.
- «Frakoblede» aktiviteter er gode og økonomiske alternativer som fremmer AT uten bruk av datamaskiner

Algoritmisk tenkning

'Ikke bare å tenke som en datamaskin'

(Wing, 2006)



Hentet fra Shute, Sun & Asbell-Clarke (2017)

1

Algoritmisk tenkning (AT) krever kognitive prosesser som er nyttige på tvers av fag og i verden utenfor skolen, der problemer ofte er diffuse og vanskelige å definere.³ «Gjennomtenkt bruk av algoritmiske verktøy og arbeidsmåter kan styrke læringen av matematisk og vitenskapelig innhold».⁴ Noen forskere fremmer undervisning om dette som en grunnleggende ferdighet som er anvendelig på tvers av de fleste fag!¹ For å fremme AT finnes det flere tilnærminger og verktøy, med forskjellige følger for elevenes læring. Det er imidlertid behov for ytterligere forskning for å undersøke hvordan de kan brukes hensiktsmessig for å støtte læring og undervisning i klasserommet.

IMPLIKASJONER: Algoritmisk tenkning krever bruk av tankeprosesser som er nyttige i problemløsning på tvers av fag, og i verden utenfor skolen

Videre forskning er nødvendig for å belyse hvordan ulike tilnærminger og verktøy kan fremme AT i klasserommet

“Algoritmisk tenkning er en grunnleggende ferdighet for alle, ikke bare for informatikere. I både lesing, skrijving og regning bør algoritmisk tenkning være en del av ethvert barns analytiske evner”

Wing, 2006

“Algoritmisk tenkning inkluderer et bredt spekter av mentale verktøy og begreper fra informatikken, som hjelper mennesker med å løse problemer, designe systemer og forstå menneskelig atferd”

National Research Council, 2011

2

Pedagogisk robotikk tilbyr engasjerende opplevelser på alle utdanningsnivå,⁷ til og med for små barn på 4–6 år.⁵ Robotverktøyene blir «objekter å tenke med»,⁶ og kan skape et utforskende miljø for elever, slik at de kan designe og løse problemer, programmere og samtidig konstruere kunnskap. Dette forbedrer matematisk og vitenskapelig forståelse, samtidig som det bidrar til å skape interesse for realfag.⁷ Elever som deltok i undervisning med robotikk, hadde en markant forbedring i AT-kunnskap og ferdigheter.⁸ Det er imidlertid viktig at lærere tar i bruk hensiktsmessige robotverktøy for å fremme AT-begreper, praksiser og perspektiver på en god måte, ut fra elevenes forutsetninger.

IMPLIKASJONER: Enkle AT- og programmeringsbegreper kan introduseres på en meningsfull måte til små barn helt nede i 4–6-årsalderen ved hjelp av pedagogisk robotikk

Aktiviteter med robotikk tilbyr en plattform der elever kan lære ved å gjøre, noe som forsterker elevenes motivasjon

Bruk av robotikk kan øke forståelsen i matematikk og vitenskap og samtidig bidra til å bygge opp interesse for realfag

Å lykkes med å utvikle AT ved bruk av robotikk kan avhenge av om verktøyene som tas i bruk, er hensiktsmessige med tanke på elevenes utviklingsmessige forutsetninger

3

Scratch og Alice er visuelle programmeringsspråk som har blitt populære verktøy for å utvikle AT i skolen. På grunn av deres sterke pedagogiske kobling til *konstruktivismen*⁶ oppmuntrer disse verktøyene til å lage objekter (grafikk og animasjon), og bidrar dermed til utviklingen av begreper, praksiser og perspektiver innenfor AT. Forskning tyder på at slike barnevennlige programmeringsspråk er enkle nok for nybegynnere, slik at de kan lære begreper innenfor informatikken (lav inngangsterskel), og samtidig kan de lage avanserte programmer (stor takhøyde).⁹ I prosessen utforsker elever AT-begreper blant annet gjennom prøving og feiling. Elever som presterer på høyere nivå, eksperimenterer vanligvis mer, og det tyder på at denne tilnærmingen er hensiktsmessig for dem.¹⁰ Elever som presterer på lavere nivå, trenger gjerne mer støtte. For å legge til rette for begrepsmessig forståelse bør elevene alltid oppfordres til å forklare tenkningen sin med egne ord, før de koder programmer. Dette sikrer at tankeprosesser er like verdsatt som resultater.^{11,12}

IMPLIKASJONER: Visuelle programmeringsspråk er gode introduksjonsverktøy for elevene når de skal øve på AT-ferdigheter, samtidig som verktøyene legger til rette for læring av grunnleggende begreper innenfor informatikken

Elever som presterer høyt, tjener på å få arbeide med problemer som legger til rette for prøving og feiling, mens elever som presterer lavere, trenger mer støtte i arbeidet

Lærerne bør be elevene om å forklare tankeprosessene sine eksplisitt før de lager blokkbasert kode, slik at tenkning blir både verdsatt og fremmet

4

Tilnærmingene ovenfor vektlegger maskinvare- og programmeringsaspektene ved AT. Frakoblede aktiviteter tilbyr alternative måter å utvikle elevens problemløsningsferdigheter mht AT på.¹³ Det kan gjøres for eksempel ved hjelp av lekbaserte brettspill (eller papirspill) og konkurranser som demonstrerer AT-prosesser (se csunplugged.org).¹⁴ Denne tilnærmingen kan være spesielt nyttig for skoler som ikke har andre ressurser eller infrastruktur på plass (dvs. datamaskiner, teknologisk utstyr, lærere med programmeringskompetanse). AT-begreper som læres gjennom frakoblede aktiviteter, kan bygges inn i ulike fagområder, eller brukes som en overgang til tilkoblede aktiviteter.¹⁵ Lærere og elever anser frakoblede AT-aktiviteter som et like godt alternativ som vanlige, tilkoblede datateknologiaktiviteter,¹⁶ og forskning tyder på at elever som deltok i frakoblede aktiviteter, forbedret AT-ferdighetene sine betydelig sammenlignet med elever som ikke gjorde det.¹⁷

IMPLIKASJONER: Frakoblede aktiviteter, som engasjerende brettspill, er alternative måter å arbeide med AT-begreper på, og krever ikke en datamaskin

De kan være en økonomisk måte å forbedre AT-ferdigheter på, særlig nyttige for skoler med begrensede ressurser

Lærere og elever anser frakoblede aktiviteter som en motiverende måte å lære AT på, en motivasjon som går over i økt interesse og selvillit ved læring av datateknologi og matematikk

REFERENCES

- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33–35.
- National Research Council. (2011). *Report of a workshop on the pedagogical aspects of computational thinking*. Washington, DC: The National Academies Press.
- Cuny, J., Snyder, L., & Wing, J. (2010). Demystifying computational thinking for non-computer scientists. Unpublished Manuscript.
- Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. *Journal of Science Education and Technology*, 25(1), 127–147.
- Bers, M. U., Flannery, L., Kazakoff, E. R., & Sullivan, A. (2014). Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. *Computers & Education*, 72, 145–157.
- Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. New York: Basic Books, Inc.
- Penmetcha, M. R. (2012). *Exploring the effectiveness of robotics as a vehicle for computational thinking* (Master's thesis). Retrieved from ProQuest Dissertations & Theses A&I (Order No. 1529729).
- Witherspoon, E. B., Higashi, R. M., Schunn, C. D., Baehr, E. C., & Shoap, R. (2017). Developing computational thinking through a virtual robotics programming curriculum. *ACM Transactions on Computing Education (TOCTE)*, 18(1), 1–20.
- Iye, S. Y., & Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K–12? *Computers in Human Behaviour*, 41, 51–61.
- Rose, S., Habgood, J., & Jay, T. (2017). An exploration of the role of visual programming tools in the development of young children's computational thinking. *Electronic Journal of E-Learning*, 15(4), 297–309.
- Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 366(1881), 3717–3725.
- Lai, R.P.Y. (in press). Exploring the cognitive mechanism of computational thinking and its educational implications. In *The Proceedings of International Conference on Computational Thinking Education 2019 Conference*. Hong Kong: The Education University of Hong Kong.
- Loai, C.-K., How, M.-L., Longkai, W., Seow, P., & Liu, L. (2018). Analysis of linkages between an unplugged activity and the development of computational thinking. *Computer Science Education*, 28(5), 255–279.
- Bell, T., Alexander, J., Freeman, I., & Grimley, M. (2009). Computer science unplugged: school students doing real computing without computers. *New Zealand Journal of Applied Computing and Information Technology*, 13(1), 20–29.
- Caldwell, H., & Smith, N. (2016). Teaching computing unplugged in primary schools: Exploring primary computing through practical activities away from the computer. *Learning Matters*.
- Faber, H. H., Wierdsma, M. D. M., Doombos, R. P., Ven, J. S. van der, & Vette, K. de. (2017). Teaching computational thinking to primary school students via unplugged programming lessons. *Journal of the European Teacher Education Network*, 12(3), 20–24.
- Brackmann, C. P., Román-González, M., Robles, G., Moreno-León, J., Casali, A., & Barone, D. (2017). Development of computational thinking skills through unplugged activities in primary school. *Proceedings of the 12th Workshop on Primary and Secondary Computing Education*, 65–72.