

Programmering og matematikkundervisning

Rune Herheim, HVL: rher@hvl.no
 Fredrik Eidsvåg, Midtun skole: fredeids@gmail.com
 5.11.2021

1

1

Disposisjon

- Kva er algoritmaskenking, programmering og koding?
- Kva er blokkbasert og tekstbasert programmering
- Døme på programmering av femkant, både med blokk og tekst
- Oppsummering

2

2

Algoritmaskenking–programmering–koding

Algoritmaskenking
 Abstrahering, generalisering, automatisering, algoritmebehandling, dekomponering

Programmering
 Løse problem v.h.a. en algoritme eller program: forarbeid, planlegge, skrive, tegne, bearbeide, undersøke om problem er løst. Eksperimentere, revurdere, argumentere for at algoritmen alltid virker. Generalisere algoritmen til å løse en hel klasse av problemer. Sjekke effektivitet.

Koding
 Selve aktiviteten med å reformulere dette inn i et bestemt programmeringsspråk.

3

3

Den algoritmske tenkeren

Nøkkelbegrep

- 1 Logikk
Analyse og forstå
- 2 Algoritmer
Regler og steg for steg
- 3 Dekomposisjon
Bryte ned i mindre delar
- 4 Mønster
Finne og bruke likheter
- 5 Abstraksjon
Fjerne unødvendige detaljar
- 6 Evaluering
Opne valdelegger

Arbeidsmåter

- ✓ Fikle
Utforske og eksperimentere
- ✓ Skape
Svare og lege
- ✓ Feltiske
Opplyse og sette fel
- ✓ Holde ut
Fortsette og prøve igjen
- ✓ Samarbeide
Dele og jobbe sammen

[Algoritmaskenking](#)

4

4

Algoritmaskenking

Klare parallellar mellom AT og kjerneelement i matematikk:

Algoritmaskenking

- Abstrahering
- Generalisering
- Automatisering
- Algoritmebehandling
- Dekomponering

(Gjøvik & Torkildsen, 2019)

Kjerneelement i matematikk

- Utforsking og problemløsing
- Modellering og anvendelse
- Resonnering og argumentasjon
- Representasjon og kommunikasjon
- Abstraksjon og generalisering

5

5

Algoritmaskenking

Kjerneelement: Utforsking og problemløsing
 Utforsking i matematikk handlar om at elevane leiter etter mønster, finn samanhengar og diskuterer seg fram til ei felles forståing. Elevane skal leggje meir vekt på strategiane og framgangsmåtene enn på løysingane. Problemløsing i matematikk handlar om at elevane utviklar ein metode for å løyse eit problem dei ikkje kjenner frå før. **Algoritmaskenking er viktig i prosessen med å utvikle strategiar og framgangsmåtar for å løyse problem og inneber å bryte ned eit problem i delproblem som kan løysast systematisk. Vidare inneber det å vurdere om delproblema best kan løysast med eller utan digitale verktøy.** Problemløsing handlar òg om å analysere og forme om kjende og ukjende problem, løyse dei og vurdere om løysingane er gyldige.

6

6

Algoritmisk tenking og programmering i LK20

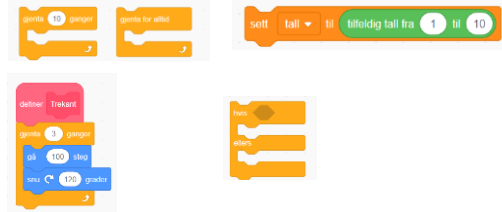
AT og programmering er sentralt i kjerneelementet *Utforskning og problemløsning*

Kompetansemål

2. trinn: lage og følge regler og trinnvise instruksjoner i lek og spel
3. trinn: lage og følge regler og trinnvise instruksjoner i lek og spel knytte til koordinatsystemet
4. trinn: lage algoritmar og uttrykke dei ved bruk av **variablar, vilkår og lykkjer**
5. trinn: lage og programmere algoritmar med bruk av **variablar, vilkår og lykkjer**
6. trinn: bruke **variablar, lykkjer, vilkår og funksjonar** i programmering til å utforske geometriske figurar og mønster
7. trinn: bruke programmering til å utforske data i tabellar og datasett
8. trinn: utforske korleis algoritmar kan skapast, testast og forbeholdt ved hjelp av programmering
9. trinn: simulere utfall i tilfeldige forsøk og berekne sannsynet for at noko skal inntreffe, ved å bruke programmering
10. trinn: utforske matematiske eigenskapar og samanhengar ved å bruke programmering

7

Variabler, vilkår, løkker, funksjoner ...



8

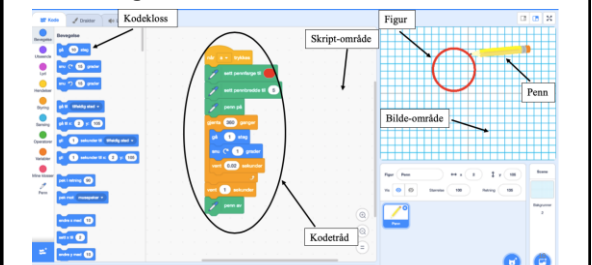
Produktivt strev når ein programmerer

- 7. trinn, elevar arbeidde i par, brukte Scratch på PC
- Oppgåver
 1. Programmer ein firkant
 2. Programmer ein femkant
 3. Programmer ein sirkel
 4. Programmer ei stjerna

Herheim og Johnsen-Høines (2020; 2021)
Pettersen (2019)

9

Scratch grensesnitt



10

Oppgåve 2: Kor mange sider er det i ein femkant?

[Knut viser tegningen til Ida.]

Knut: Er det femkant?

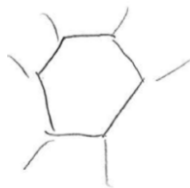
Ida: Ja.

Knut: Den.

Ida: Nei, det er ... Vent litt, få se.

Knut: Jo, det er en, to, tre, fire, fem.

Ida: En, to, tre, fire, fem, seks. Det er en sekskant.



11

11

Oppgåve 2: Kor store er vinklane i ein femkant?

Knut: Hvordan skal vi gjøre det? Vent, hvor mange grader er det i en femkant?

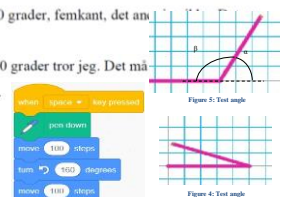
Ida: Jeg aner ikke. En firkant, det er 90 grader, femkant, det anner hvert fall over 90 grader.

Ida: Der. Sann. Ja. Da må vi ha over 90 grader tror jeg. Det må liksom, for alle er stumpe vinkler.

Knut: 120?

Ida: Ja, kanskje det. Jeg vet ikke.

Knut: Vi bare prøver da.

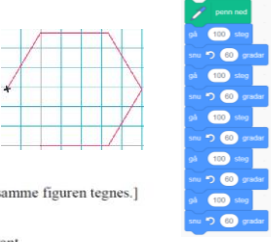


12

12

Oppg ve 2 held fram: pr ver 60 

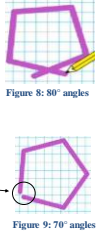
Knut: Vi mangler en. Se.
 Ida: Ja, jeg skj nner ikke helt.
 Knut: Nei ...
 [De trykker start p  nytt, og den samme figuren tegnes.]
 Ida: Da m  vi ta g .
 Knut: Nei, men nei, da blir det en sekskant.
 Ida:  ja, men da kan ikke vi ha 60 grader. Da m  vi ha mindre. Nei mer.



13

Oppg ve 2 held fram: pr ver 80  og 70  ...

Knut Vent da. Er ikke det sekskant? .. Femkant.
 Ida Jo, det er jo det vi skal ha. 70 grader da. 70 er sikkert greit, siden 60 ikke .. ikke, da pr ver vi 70. Hehe.
 Knut Ja, det kan hende ... Hvis det er 70, s  blir jeg glad.
 Ida Ja, jeg ogs  ... Okei. (klikker gr nt flagg)
 Ida Jaa! Nei, hva er det? (peker p  glipen)
 Knut 1, 2, 3, 4, 5, ja, da har vi det!
 Ida Men den er jo ikke helt .. inntil .. (peker p  glipen)
 Knut Jo jo jo, de ser ikke det.



14

Oppg ve 2 held fram: pr ver   endra sidelengder

Knut   hh, arg ... kan pr ve 102, nei 105 .. p  den f rste og den andre .. nei p  den f rste og siste. (justerer antal steg p  de f rste g -klossene)
 Ida Men kanskje det er gradene?
 Knut Nei, det er ikke noe med gradene, vi har pr vd de for mye (kviskrar litt p  slutten)
 Ida Men kanskje noen som er lengre enn andre?
 Og BINGO! Der sat femkanten i sikringsboksen (ikkje regul r)

15

Proessen til Ida og Knut

- Algoritmisk tenking
 - Dei braut ned i mindre delar (antal sider, vinkelstorleik osv.)
 - Logikk: pr va og feila med ei systematisk og logisk tiln rming
 - Dei fikla, s kte etter feil, var utdaldande!
- Programmering
 - F rearbeid, planlegging, teikna femkant osv.
 - Laga forslag til kode, pr vde ut, redigerte og forbetra kode osv.
- Koding
 - Dei kodane dei laga

16

Kva viss dei hadde f tt hjelp og gjort rett med ei gong?

- Rask l sning:
 - Vinkelsum femkant er 540 
 - Delt p  antal kantar: 108 
 - Nabovinkelen: 180  - 108  = 72 
- Kva kunne dei g tt glipp av d ?
 - Eigenskapar til femkantar
 - Rolla til vinklar, vinkelstorleik, vinkeltypar
 - «Du f r det du programmerer» → ikkje nok   velja 2 g -blokker og 1 snu-blokk
 - Samanheng mellom mangelkant-type og vinkelstorleik
 - Kan kantar kryssa kvarandre?
 - M  mangelkantar vera lukka?

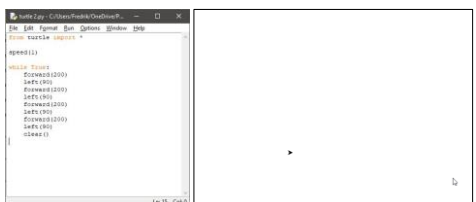


NB: Ulike representasjonar

17

Tekstbasert programmering (Python)

- Likheter med blokkprogrammering
- Koden best r av tekst som m  skrives i stedet for ferdige blokker



18

Python-programmering og matematikk

Hvilke temaer passer?

- Desimaltall (Utforske tallrekker)
- Geometri (Turtle)
- Tallforståelse (Delelighet, faktorisering, regnerekkefølge, algebra)
- Statistikk (Gjennomsnitt, median, typetall etc.)

19

Python-Turtle og geometri

```

1 from turtle import *
2 square(100)
3
4 # Her definerer en funksjon som skriver ut alle brøker i parentes
5 # og oppretter, lagrer, og plott
6 # et 100x100 kvadrat
7 def print_square():
8     length = 100
9
10 # For et kvadrat, lagre all informasjon om kvadratet og sett den som
11 # kvadrat(100, 100)

```

20

Utforske tall med Python – noen eksempler:

- Lag et program som skriver ut 4-gangen opp til 100
- Lag et program som finner alle tall mellom 0 og 100 som er delelig med 17
- Lag et program som skriver ut denne tallrekken opp til hundre: 2, 7, 12, 17, 22 ...

21

Utforske desimaltall med Python?

Hva skjer når man multipliserer et helt tall med hundredele?

Hva skjer fra 0.16 til 0.2 og videre til 0.24?

```

1 from time import sleep
2
3 x = 0.16
4
5 for i in range(100):
6     print(i * x)
7     sleep(1)

```

22

Hvorfor bruke tekstbasert programmering i barneskolen?

Krever en del forarbeid og kunnskap hos lærer, men kan være et spennende utgangspunkt for gode matematikkfaglige diskusjoner og arbeid

Bredt erfaringsgrunnlag – Kan det styrke elevers algoritmiske tenkning?

Matematikk og programmering

<https://sites.google.com/bergenskolen.com/matematikkopprogrammering>

23

Kvifor integrera programmering i undervisning?

- Læra programmering (og rekneark) fordi det er kompetansemål
- Variasjon, kjekt og når dei kan tilby noko du ikkje får til på andre måtar
 - Særleg det dynamiske for funksjonar/grafar og geometri, representasjonar, tempo

For å få til matematikkfagleg fokus er det viktig at:

- Oppgaver er
 - gode med omsyn på å utnytta potensiale i digitale læremiddel
 - har passende kompleksitet og vanskegrad
- Ein diskuterer eksplisitt kva matematikk som er brukt
- Programvaren er transparent nok

Er ikkje gull alt som er digitalt ...

24

Referansar

- Eidsvåg, F. (2019). *Utforskning av delerlighet og rest med tekstbasert programmering* (Masteroppgave). Bergen: Høgskulen på Vestlandet.
- Forsström, S. E. & Kaufmann, O. T. (2018). *A literature review exploring the use of programming in mathematics education. International Journal of Learning, Teaching and Educational Research*, 17(12), 18–32.
- Gjøvik, Ø. & Torkildsen, G. (2019). *Algoritmisk tenkning. Tangenten – tidsskrift for matematikkundervisning*, 30(3), 31–37.
- Herheim, R. & Johnsen-Høines, M. (2020). *A culture perspective on students' programming in mathematics. Journal of Mathematics and Culture*, 14(2), 91–110.
- Herheim, R. & Johnsen-Høines, M. (2021). *Students' productive struggle when programming in mathematics. I Bringing Nordic mathematics education into the future: Proceedings of Norma20, The ninth Nordic conference on mathematics education* (s. 121–128). Göteborg: SMDf, Swedish Society for Research in Mathematics Education.
- Mishra, P. & Koehler, M. J. (2006). Technological pedagogical content knowledge: A framework for teacher knowledge. *Teachers College Record*, 108(6), 1017–1054.
- Pettersen, O. Å. (2019). *Matematisk argumentasjon og programmering. Når elever på 7. trinn arbeider i par med Scratch* (Masteroppgave). Bergen: Høgskulen på Vestlandet.

25

25

Tangenten-artiklar om programmering

- Berg, T. K. (2021). *Analog programmering. Tangenten – tidsskrift for matematikkundervisning*, 32(3), 42–52.
- Fløy, E. E. (2021). *Programmering i LK20 Tangenten. Tangenten – tidsskrift for matematikkundervisning*, 32(1), 3–9. ([Undervisningsopplegg](#))
- Gjøvik, Ø. & Torkildsen, H. A. (2019). *Algoritmisk tenkning. Tangenten – tidsskrift for matematikkundervisning*, 30(3), 31–37.
- Herheim, R. (2019). *Programmering og algoritmisk tenkning? Tangenten – tidsskrift for matematikkundervisning*, 30(4), 1.
- Kravik, R. (2020). *Programmering og geometri. Tangenten – tidsskrift for matematikkundervisning*, 31(2), 7–11.
- Maugesten, M., Stigberg, H. & Stigberg, S. K. (2021). *Programmering på ungdomstrinnet. Tangenten – tidsskrift for matematikkundervisning*, 32(3), 2–7.
- Stenseth, B., Kaufmann, O. T. & Forsström, S. E. (2019). *Programmering og matematikk. Tangenten – tidsskrift for matematikkundervisning*, 30(2), 7–12.
- Taraldsen, L. H. & Myhra, K. S. (2019). *Programmering med Sphærboller. Tangenten – tidsskrift for matematikkundervisning*, 30(3), 2–7.

26