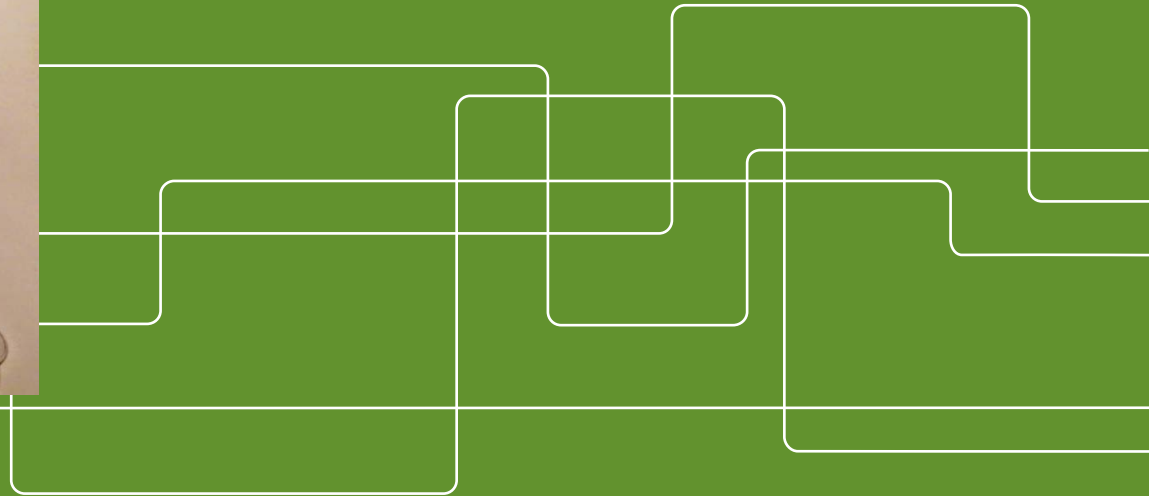




# En diskursanalytisk studie av programmering som innehåll i grundskolans teknikämne

Eva Björkholm och Susanne Engström, KTH





# Bakgrund

Kunskaper och förmågor kopplade till programmering har under senare tid förstärkts i läroplaner och kursplaner för bland annat grundskolans teknikämne. I samband med införandet av detta innehåll genomförs bl.a. en mängd kompetensutvecklingsinsatser för verksamma lärare.

En fallstudie av en kompetensutvecklingskurs i programmering för lärare. Kursen innefattar även ett projekt där lärare undervisar i sina egna klasser.

Syftet med studien är att undersöka dels vilket kunskapsinnehåll i relation till programmering som lyfts fram som centralt i undervisningen, dels vilka underliggande värden som ger innehållet ett berättigande.

## *Forskningsfråga*

*Vilket innehåll och vilka värden i relation till programmering framstår som viktiga i undervisningen i kompetensutvecklingskursen och i lärarnas egen undervisning?*



# Metod

I kursen undervisade programmeringskunniga personer med erfarenhet av undervisning i fortbildningskurser för lärare. I kursen ingick att genomföra ett projekt i programmering i den egna klassen, som redovisades i slutet av kursen. 16 verksamma lärare åk 1-9 deltog i kursen.

**Datainsamling** genom *observation* av undervisningen (två forskare) vid samtliga 6 kurstillfällen under en termin (1 heldag/tillfälle). Dessutom observerades 3 lärares undervisning i sina klasser (åk 3,4 och 7). Två lektioner à 1 timme observerades för varje lärare. Fältanteckningar. Efter varje observation diskuterade och sammanfattade forskarna sina analyser.

**Dataanalys** baserat dels på identifierandet av det innehåll som lyftes fram, dels ett diskursanalytiskt perspektiv (Börjesson & Palmblad 2015; Gee 2014) med fokus på värden och styrningsstrategier i undervisningspraktiken (Öhman, 2015).



# Resultat

## I fortbildningssituationen (undervisningen i kursen)

Vilka normativa värden framkommer?

- (1) Kunskapsinnehållet betraktas som ett självklart innehåll i skolan och innehållet betraktas som något uttalat självklart i sig.
- (2) Förhållningssättet: knapphändigt och dyrkande av programmeringens möjligheter.

Vilka styrningsstrategier används?

- (1) Förminskandets strategi.
- (2) Åtskiljandets strategi.
- (3) Logikens frammaning.



# (1) Kunskapsinnehållet betraktas som ett självklart innehåll i skolan och innehållet betraktas som något uttalat självklart i sig.

## Resultat 1:

Kunskapen består av att:

*Formulera sina tankar. Stegvisa instruktioner. Organisera sina lösningar. Struktur löser problem, väldefinierade tankar. Mer konkret: styra genom att skriva kod.*

Förstå att språket måste vara exakt och tydligt så att alla kan använda det. Kunskapen omfattar att kunna tolka och producera något som inte får missförstås. Ej tolkningsbar kunskap. Programmeringens logik. Kodningens krav. Även exempel på kodning. Förståelse för begreppet programmeringsspråk. Att koden lever sitt eget liv utanför artefakten.

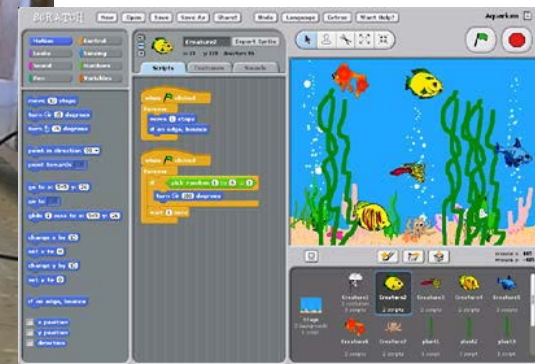
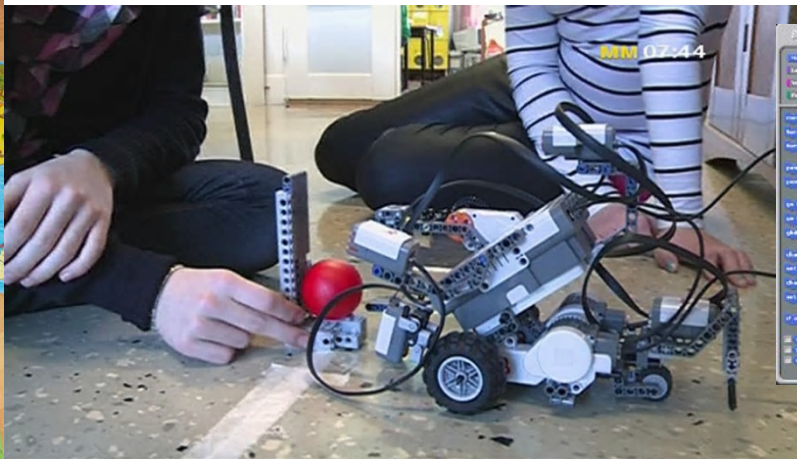
Lära sig hantverket = att programmera. Att pröva och att göra misstag. Att felsöka. Att träna mycket.



## Resultat 2:

Artefakterna som innehåll:

BlueBoot och legorobot blir ett innehåll i sig. Att lära sig styra en specifik artefakt och få *den* att göra saker. Fokus på legoroboten. Finns en progression genom att gå från att styra robotar (BB) till att bara använda dator. Men i scratch fastnade de fortfarande i "artefakten" i verkligheten = akvariet. Kunskapen (att programmera) görs inte tydlig.



## Resultat 3:

Kunskap som lyfts fram av studentlärarna:

Inte bara principen att styra med programmering utan även artefakterna i sig. Datorn, programmet som ska till, roboten... lärarna är främst fokuserade på artefakterna. Det blir otydligt för dem att kunskapen borde ligga mer på programmerandet och inte på att roboten/katten lyckas si och så. *Artefaktens upphöjning* trots att det är programmeringskonsten som ska utvecklas.







## **(2) Förhållningssättet: knapphändigt och dyrkande av programmeringens möjligheter.**

Korthugget (inget onödigt prat, definitivt, ingen öppning för diskussion, korta instruktioner), snabba instruktioner, ”rakt på”, bärare av det rätta sättet, snabba sekvenser, inspirerande, roliga och skapande aktiviteter, utveckla den skapande förmågan, skapa nytt utan att reflektera/ifrågasätta, lekfullt, okomplicerat, enkelt, utan begränsningar. Intentionen är att vara effektiv.

Inbjudande, dela programmeringskunskaper, dela med sig av idéer.

Att programmering kan leda till något utan begränsningar. I detta finns en vördnad och upphöjning av ämnet = att kunna programmera. Man skapar egna världar!







# Styrningstrategier

## ***(1) Förminskandets strategi***

Förminskande av målgruppen

*– hur känns det? Har ni roligt?, sensorer är överdramatiserat, inte så svårt, få inte panik!, roligt och tråkigt, våga prova! var inte rädda, bara som att skriva, det var en gång en robot som...hej, vi ska leka med robotar, vi kan inte gå in i rummet för då vill ni bara testa och titta...så här är det är i skolan!. - Det här är inte svårare än allt annat i skolan. Få eleverna att organisera och strukturera. Som vilket skrivande som helst.*

Bidrar till ett förhållningssätt hos studentlärarna: Lever upp till förväntningarna, blir "elever", applåderar, vad roligt!, skrattar och överdriver, förminskar sig



## (2) **Åtskiljandets strategi**

Ni och vi: åtskillnad mellan "ni och vi",

*Vad menar vi då med programmering?*

*Vi menar...en åtskillnad mellan vi som kan programmering och ni som inte kan men som ska få med er lite verktyg för att hanka er fram och få elever att tycka att det är roligt och att eleverna ska bli duktiga, våga utvecklas och förstå att allt är möjligt.*

## (3) **Logikens frammaning**

Frammanande av logiken:

*ge order för att uppmana till att göra rätt, se till att ni... se till att ni... följ instruktionen, felsök!, lämna inget till slumpen,*



# Resultat – lärarnas egen undervisning

## Årskurs 3:

Kunskapsinnehållet i teknik (det viktiga i programmering: begreppet loop, *styra* BB i relevanta problem med ett språk som alla är överens om) synliggörs, möjliggörs, elever lär sig det avsedda.

Men eleverna bortrationaliserar kodningsmomentet vartefter. Banbygget i fokus. Progressionen uteblir.

Förhållningssätt: undervisning, mål för lektionen, återkoppling etc. här sker ett lärande genom diskussion etc.



## Årskurs 4:

Kunskapsinnehåll: Bryter ned ett problem (beskriva en händelse). Delar upp problemet i delar som tillsammans blir en helhet. Delarna ska tas i en speciell ordning. Bryta ned det stora problemet i olika delar. Göra det ännu tydligare. Det är programmering! Fortsätter med val och strukturer. God progression. Intention att integrera i NT-ämnena.

Men, sammanhanget blir ologiskt och krystat. NT-integreringen irrelevant.

Viktigt: att använda programmering där det finns ett behov. Sammanhanget inte vilket som helst.

Förhållningssätt: tydlighet, framhåller vikten av struktur, seriöst (användbar kunskap – inte "lek och kul"), upprepning av kunskapsinnehållet, påminna om vad eleverna ska göra gällande innehållet, ger korta och tydliga stöd

## Årskurs 7

Kunskapsinnehåll med fokus på artefakter, otydligt om programmering, följa instruktioner, lösa uppdrag med legorobot utan gemensam återkoppling.

Programmera för att styra robotar är teknik.

Förhållningsätt: korthugget, effektivitet, utan diskussioner, "kör på", roligt





# Lärarnas redovisningar

## Normativa värden:

(1) Kunskapsinnehållet kvarstår, tas för givet, inget kritiskt förhållningssätt – underordnar sig en reproduktion:

det abstrakta tänkandet, programmering med ... och loopar, buggar, felsökning, datalogiskt tänkande, vill tydliggöra centrala begrepp.

(2) En tveeggad inställning till undervisning lyfts fram:

å ena sidan måste det vara roligt och en utmaning för eleverna, de lär sig snabbt, de ska bli nyfikna, de vill hela tiden göra mer

å andra sidan en konflikt mot den "vanliga" undervisningen eftersom det inte finns mål, ingen progression, svårt med ämnesintegrering.

(3) Förhållningssätt till eleverna:

en tydlig genuskodning: flickorna är starkare, uthålligare och mer noggranna, vilket passar programmering – pojkarna är mer slarviga och ger upp lättare



## Styrningsstrategier

Att överlåta ansvaret på eleverna.

Det är möjligt att släppa kontrollen! Eleverna lär sig ändå!  
Eleverna är duktiga och klarar sig utan läraren.

”Det blev väldigt lyckat utan att jag kunde någonting!”





# Slutsats

Innehåll och värden som framkommer som viktiga i programmeringsundervisningens diskurs:

Kunskapsinnehållet betraktas som självklart och ifrågasätts inte. Innehållet som lyfts fram uppvisar likheter med resultat från andra studier.

Avsaknad av tydlig progression i undervisningen och problematiskt med att knyta till relevanta sammanhang. En tydlig konflikt med idén om att det måste vara roligt.

Innehåll och förmågor som lyfts fram är noggrannhet, tålamod och struktur, vilka anses passa för den stereotypa bilden av en kvinnlig elev.

Acceptans inom diskursen för idén om att elever kan lära sig själva, utan att ha blivit undervisade.