

Ett undervisningsmaterial för att stödja elevers förståelse för programmering inom ramen för teknikämnet

Niklas Salomonsson

Mattias Wickberg Hugerth

Per Anderhag

Stockholms stad

Stockholms stad och Stockholms universitet

Stockholms stad och Stockholms universitet

I denna artikel diskuterar vi hur vi har använt resultat från ett mindre forskningsprojekt om programmering för att utveckla undervisningen inom teknikämnet. I forskningsprojektet såg vi att eleverna använde olika strategier när de programmerade en robot att följa en böjd linje, och att dessa strategier var olika framgångsrika för att lösa uppgiften. Resultatet från detta projekt samt våra erfarenheter av att undervisa programmering på grund- och gymnasieskolan användes för att ta fram och pröva ett undervisningsmaterial som stödjer elevers förståelse för programmering. Materialet adresserar inte enbart programmeringsspecifika begrepp, utan också för teknikämnet relevanta företeelser såsom återkopplingssystem, ändamålsenlighet och pröva och ompröva. Undervisningsmaterialet, som är grundat i den didaktiska modellen Organiserande syften, utgörs av en PowerPoint med korta lärarkommentarer.

Inledning

Programmering har nu kommit in som ett innehåll i teknikämnets kursplan. Det är dock inte självklart hur detta innehåll kan undervisas på ett sätt som stödjer eleverna i att utveckla sina strategier eller hur man kan skapa en genomtänkt progression. Dessutom är utrymmet begränsat för att genomföra omfattande programmeringsprojekt i teknikämnet. Det är därför viktigt att uppgifter kan knytas till de förmågor som eleverna ska få möjlighet att utveckla när de lär sig programmera inom ramen för teknikämnet. Programmering är en typ av teknisk lösning, liksom att bygga en bro är en teknisk lösning, och även om de är olika kan man i båda fallen titta på dem utifrån syfte, behov, konstruktionsteknik och påverkan på människa, samhälle och miljö.

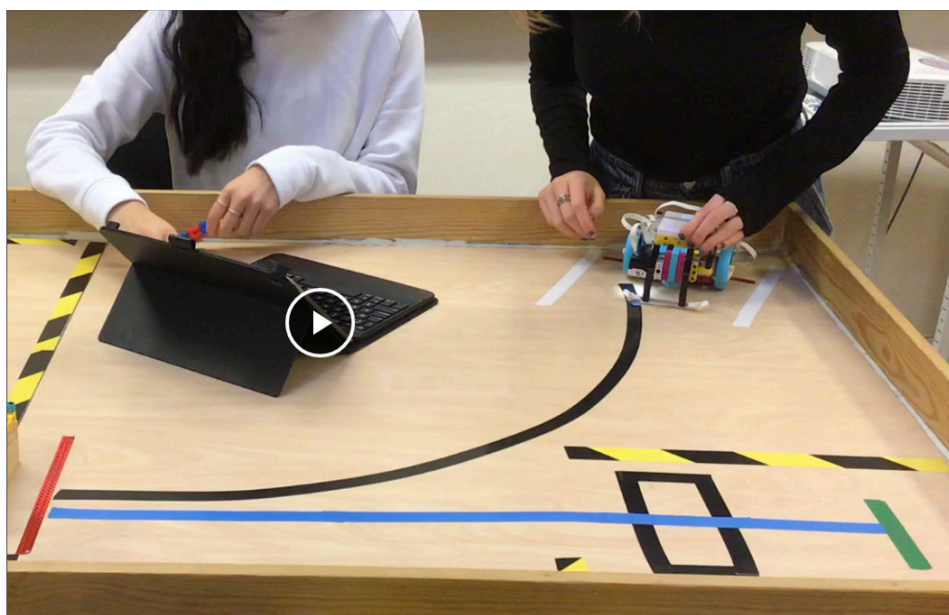
Syftet med denna artikel är att presentera ett undervisningsmaterial, baserat på vetenskaplig grund och lärares undervisningserfarenheter, som vi tagit fram för att adressera frågorna ovan. Materialet är tänkt att ge lärare stöd i hur de kan tänka kring hur teknikundervisning i programmering på högstadiet kan utformas för såväl förståelse för programmering i sig som teknikundervisning i stort. I arbetet med att designa undervisningen och skapa en genomtänkt progression har vi tagit stöd av den didaktiska modellen organiserande syften.

Ambitionen har varit att ta fram ett konkret material som tekniklärare i grundskolan kan använda i sin undervisning. Vi kommer därför presentera hur vi använt resultat från ett mindre forsknings- och utvecklingsprojekt i kombination med erfarenheter av att undervisa om programmering på grund- respektive gymnasieskolan för att utveckla och lättare integrera undervisning kring programmering i teknikämnet. I texten resonerar vi kring de förändringar vi såg som viktiga att göra i den programmeringsuppgift som vi använde i forskningsprojektet i relation till detta undervisningsmaterial. Undervisningsmaterialet, som är i form av en Powerpoint med kortfattade kommentarer till läraren, finns tillgängligt via Figshare [\[1\]](#), ett öppet datarepositorium för forskningsdata och forskningsprodukter. Studien har genomförts inom ramen för Stockholm

Teaching & Learning Studies (STLS).

Programmering för att skapa förståelse för teknik

I forskningsprojektet undersökte vi vilka strategier elever i årskurs 9 använde för att programmera en robot att följa en böjd linje [2,3] (Figur 1). I en av klasserna var denna aktivitet del i ett större sammanhang, att följa en böjd svart linje och stanna vid en röd linje var bara en av flera uppgifter roboten skulle göra i en modell av en automatiserad återvinningsstation. I studien valde vi dock att närmare studera hur eleverna hanterade undervisningssituationen med den böjda linjen, då denna typ av programmeringsuppgift är relativt vanlig [4]. Dessutom såg vi att uppgiftens utformning möjliggjorde för eleverna att prata om och försöka lösa tekniskspecifika utmaningar med relevans för utvecklandet av ett teknikkunnande.



Figur 1. En av uppgifterna består i att eleverna ska programmera en robot att med hjälp av en ljussensor följa en böjd linje för att sedan stanna vid den röda linjen. (Bild: Niklas Salomonsson)

Vi kommer att närmare diskutera sådana aspekter i relation till presentationen av lektionsmaterialet, men till exempel uppstår intressanta frågor och diskussioner kring kodens funktionalitet när man för in en viss grad av komplexitet i uppgiften [5]. Detta gäller också det övergripande systemet kod – robot, till exempel genom att använda sensorer eller inte för att få roboten att följa linjen. Det finns helt enkelt flera olika sätt som man kan programmera en robot att följa en böjd linje på. Dessa olika sätt kan öppna upp för givande samtal kring input-process-output, återkoppling, optimering och reglering.

I forskningsprojektet och i vår undervisning har vi även noterat att elever kontinuerligt utvärderar kodens funktion när de observerar robotens rörelse. Detta sker spontant och är vanligtvis ingenting som vare sig eleverna eller vi lärare uppmärksammar explicit. Vi tänker dock att det kan vara viktigt att uppmärksamma eleverna på vad de faktiskt gör, att analysera och utvärdera tekniska lösningars ändamålsenlighet är centralt i ett teknikkunnande och något som elever potentiellt kan öva på och utveckla när de programmerar. Koden kan alltså ses som en form av teknisk lösning [5].

Slutligen har vi sett, både i forskningsprojektet och i vår reguljära undervisning, att eleverna kontinuerligt prövar och omprövar sina kodsegment. Detta skulle också kunna vara något som man som lärare uppmärksammar och visar på mer systematiska sätt att göra det på. Vi tror således att

det finns en stor outnyttjad potential här som lärare kan använda sig av för att stödja elevernas förståelse för teknik i allmänhet.

Elevers strategier att programmera en robot att följa en böjd linje

Eleverna jobbade i par och använde en Ipad där de använde blockprogrammering för att programmera en Lego Spike robot. Samtliga robotar hade en ljussensor monterad och eleverna uppmuntrades att utgå från en exempelkod som var tänkt att stödja dem i att konstruera en kod där robotens sensor och hjul samverkade i rörelsen framåt utmed linjen. Studien visade på tre olika strategier som eleverna använde när de programmerade sina robotar: (1) *sensor-följa linjen*, sökandet efter ett kodblock som automatiskt skulle få roboten att följa linjen (ett sådant kodblock fanns inte i programmeringsmiljön), (2) *sensor-hjul*, användandet av kod för att skapa ett återkopplingssystem mellan sensor och hjul, och (3) *rotationer-grader-hjul*, användandet av robotens förändrade position för att stegvis finjustera hjulens rörelse så att den följde linjen.

De olika strategierna och de utmaningar som uppstod i relation till strategierna specifikt och programmering av en robot generellt, utgör fokus för denna text och det undervisningsmaterial vi utvecklat. I vad som följer kommer vi därför att kortfattat beskriva fyra centrala poänger som vi har utgått ifrån när vi utvecklat materialet.

1. **Programmeringsspråket och föremålet det styr:** Många elever hade lite erfarenhet av att programmera och generellt hade de en vag förståelse för vad mjuk- och hårdvara kunde göra. Till exempel fanns det en förväntan om att det borde finnas ett *Följa linjen*-kodblock som kunde användas. Kopplingen mellan robotens sensor och dess rörelse var något eleverna behövde mycket stöttning från lärarna för att förstå och kunna realisera i en fungerande kod. Det vill säga, mycket undervisning behövdes kring de processer där input från sensorn omsätts som output i hjulens rörelse.
2. **Programmeringsspråket:** Förståelse för programmeringsspråket. De olika kodblockens funktion och hur de används i relation till hårdvaran (roboten med dess olika komponenter) behöver göras explicit.
3. **Göra aktiviteter kontinuerliga med lärandemålet:** Samtliga elever förstod uppgiften. Det vill säga målet var klart för alla elever (att programmera en robot att följa en böjd linje), vägen dit var det inte. Detta skapade mycket frustration hos eleverna och mycket tid ägnades åt att testa och justera koden [2].
4. **Att komma i gång (och sedan fortsätta):** Eleverna hade svårt att komma i gång, vilket vi ser beror på ovanstående punkter, det vill säga liten programmeringserfarenhet, brist på förståelse för programmeringsspråket, och svårt att bryta ner uppgiften i hanterbara enheter.

Ett undervisningsmaterial för att stödja elevernas kunnande i att använda programmering för att styra och reglera en egen konstruktion med hjälp av programmering

Här följer en beskrivning av hur vi har resonerat i utvecklandet av undervisningsmaterialet. Lektionerna är framtagna för att användas i grundskolans teknikundervisning i årskurs 7 till 9 och är väl anpassat till ämnets kursplan. Ambitionen har också varit att materialet ska vara relevant utifrån en lärandeprocess; vi har därför resonerat kring vilka generella programmeringsaspekter som kan utgöra en grund för den programmering eleverna eventuellt kommer att möta på gymnasiet. Det vill säga, vad i att styra och reglera en robot att följa en böjd linje är relevant bortom den specifika lektionskontext som aktiviteten genomförs i. Salomonssons och Wickberg Hugerths erfarenheter av att undervisa programmering på grundskolan respektive gymnasiet har legat till grund för dessa diskussioner. Vi kommer nu i tur och ordning beskriva hur

vi har hanterat de fyra punkterna som vi presenterade i avsnittet ovan när vi utvecklat materialet. Efter det presenteras materialets delar och upplägg.

Programmeringsspråket och föremålet det styr

Det första momentet i materialet handlar om robotens delar och de kodblock som kommer att användas för att kunna programmera roboten att följa en böjd linje. Detta moment syftar alltså till att introducera programmeringsspråket och den robot de ska konstruera. I ursprungsuppgiften hade vi valt att låta eleverna jobba med färdigmonterade robotar. Här har vi dock valt att införa ett moment där eleverna själva ska montera sin robot. Detta för att eleverna ska få en bättre förståelse för vad de olika delarna gör och hur de interagerar med varandra.

Arbetet med att montera roboten gör också att eleverna får arbeta med kursplansmålet egna konstruktioner, vilket nämns både i teknikämnets centrala innehåll och i de övergripande syftesbeskrivningarna. När de monterar blir det tydligt hur delarna kopplas till hubbens portar som de sedan kommer att se i programmet när de programmerar. Här kontextualiseras också programmeringsuppgiften i ett vidare sammanhang, vad är programmering och vilka erfarenheter har eleverna av programmering? Det finns flera sätt att programmera en robot att följa en böjd linje, följaktligen finns det en rad olika kodblock och komponenter som kan användas. Vi bedömer dock att de delar och kodblock som introduceras i materialet är fullt tillräckliga för att eleverna ska kunna lösa uppgifterna tillsammans med varandra och läraren.

I den ursprungliga undervisningssekvensen var det flera elever som löste uppgiften att följa en böjd linje helt utan att använda sensorn, och därmed skapade en lösning som visserligen löste det aktuella problemet, men som inte nyttjar den tillgängliga tekniken. Därav senarelades dels monteringen av sensorn till ett senare steg, och dels lade in en uppgift att följa en linje utan sensorn. Detta gjordes för att tydliggöra programmering utan omvärldsinformation i form av sensordata och kunna diskutera vad som möjliggörs när sensorn sedan monteras och används för den slutgiltiga uppgiften.

Som beskrivits ovan i relation till de olika strategier vi tidigare identifierat är det lämpligt att läraren uppmärksammar eleverna på relationen mellan mjuk- och hårdvara, det vill säga hur en viss kod kan eller förväntas realisera robotens rörelse genom dess olika komponenter. Vår erfarenhet är att detta är något som läraren kontinuerligt bör uppmärksamma eleverna på och att eleverna uppmanas att diskutera tillsammans. Vad vill vi att roboten ska göra? Vilka komponenter är nödvändiga för att roboten ska kunna göra det vi vill att den ska göra? Vilken kod kan användas för att styra komponenterna så att de gör det vi vill att de ska göra

Programmeringsspråket

Materialet är anpassat för elever med begränsade erfarenheter av att programmera. Det är dock bra om de tidigare kommit i kontakt med blockprogrammering, till exempel genom Scratch eller någon liknande programmeringsmiljö.

Vi tänker att det inledningsvis finns två uppenbara aspekter som undervisningen behöver uppmärksamma eleverna på när de ska börja använda blockprogrammering. Det ena är kopplingen mellan kodblock och de fysiska artefakter som ska programmeras, i det här fallet en robot. Att det föreligger ett återkopplingssystem mellan kod och föremål och att föremålet enbart gör vad som står i koden. Det andra som eleverna behöver få syn på och diskutera är de specifika kodblock som de ska använda. Vad de kallas, hur de ser ut och vad de används till.

Vi rekommenderar att lägga tid inledningsvis att låta eleverna bekanta sig med de olika kodblocken, men det är först i handling när de används och eleverna själva kan se effekterna av dem som de blir meningsfulla och begripliga. Av samma skäl introduceras den centrala komponenten sensorn när den framstår som meningsfull att använda för eleverna. Det vill säga, när

sensorn kan lösa en reell utmaning. När eleverna arbetar med de olika programmeringsuppgifterna i materialet är det därför viktigt att du som lärare återknyter, sammanfattar och påminner eleverna om programmeringsspråkets logik och de delar som bygger upp denna logik. Som nämnts ovan har vi valt att enbart presentera de kodblock som eleverna kommer att använda för att kunna programmera roboten att göra de olika rörelserna.

Göra aktiviteter kontinuerliga med lärandemålet

Det faktum att samtliga elever förstod vad de skulle göra, det vill säga programmera en robot att följa en böjd linje och att sedan stanna på en angiven plats, visar att eleverna hade mål i sikte. Begreppet *mål i sikte* kommer från den didaktiska modellen *Organiserande syften* [6] som används av lärare och forskare för att designa och analysera undervisning (se t.ex. Danielsson Thorell m.fl., 2014 [7]). Modellen är grundad i en pragmatisk teoribildning [8] och utgår från att läraren alltid har ett *övergripande syfte* med undervisningen, eleverna ska lära sig någonting. För att eleverna ska nå det övergripande syftet, detta är ju någonting som de ännu inte kan, behöver läraren organisera aktiviteter som ligger nära elevernas erfarenheter och som de förstår och kan agera på, dessa elevnära aktiviteter är lektionens *närliggande syften*.

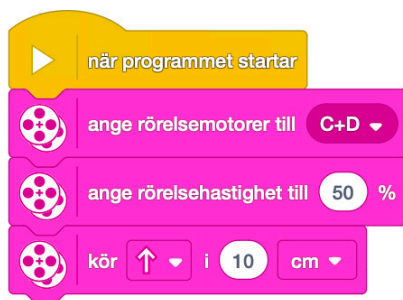
Ett bra närliggande syfte ger eleverna mål i sikte i det att de förstår vad de ska göra och kan själva avgöra när de är klara. Detta kan man bland annat märka genom att det är få frågor från eleverna gällande *vad* de ska göra eller *hur* de ska göra. Vidare behöver det närliggande syftet också skapa ett behov av att veta, det vill säga eleverna behöver skapa nya relationer till det de möter i undervisningen (t.ex. vi behöver en kod som gör att den stannar när den ser en röd färg). Det räcker givetvis inte att de närliggande syftena är mål i sikte eller skapar ett behov av att veta, de behöver också göras kontinuerliga med varandra och därmed kopplas med lektionens övergripande syfte. Detta kan inte eleverna göra själva utan detta behöver läraren hjälpa eleverna med.

I relation till ovanstående modell kunde vi i forskningsprojektet konstatera att aktivitetens övergripande syfte var att eleverna skulle utveckla en förståelse för systemet sensor – rörelse och att det enda närliggande syftet - Programmera roboten så att den följer den böjda linjen och stannar på avsedd plats - var alltför omfattande. Ingen elev frågade vad de skulle göra, däremot ägnade de mycket tid till att diskutera hur de skulle göra och behövde mycket hjälp från lärarna. Aktiviteten att utveckla en förståelse för systemet sensor-rörelse behövde flera närliggande syften och som på ett bättre sätt grundas i elevernas tidigare erfarenheter. I materialet har vi därför skapat aktiviteter som startar från en tänkt elevgrupp med lite erfarenhet av programmering. Varje aktivitet, eller närliggande syfte, är formulerade så att de ger mål i sikte och successivt kopplas samman med varandra mot det övergripande syftet.

Att komma i gång (och sedan fortsätta)

I den tidigare studien hade eleverna tillgång till en exempelkod som uppmuntrade strategi 2 ovan, dvs *sensor-hjul* som innebär att skapa ett återkopplingssystem mellan sensor och hjul. Det visade sig dock att exempelkoden användes sparsamt, förmodligen eftersom den var för komplex givet elevernas förkunskaper. Det var för svårt för eleverna att förstå hur exempelkoden kunde användas för att realisera robotens rörelse.

För elever som till exempel förväntade sig att hitta ett *Följ linjen*-kodblock eller använda rotationer snarare än sensorn, fyllde exempelkoden ingen funktion. I materialet ska alla elever därför utgå från en specifik kod ([Figur 2](#)) för att programmera roboten att göra den första enkla rörelsen att köra framåt i 10 cm. När vi har utvecklat uppgiften har vi sett att det är mycket betydelsefullt att eleverna tidigt får uppleva att det de gör får roboten att reagera på programmet. Det vill säga att de i handling ser att input omvandlas till output.

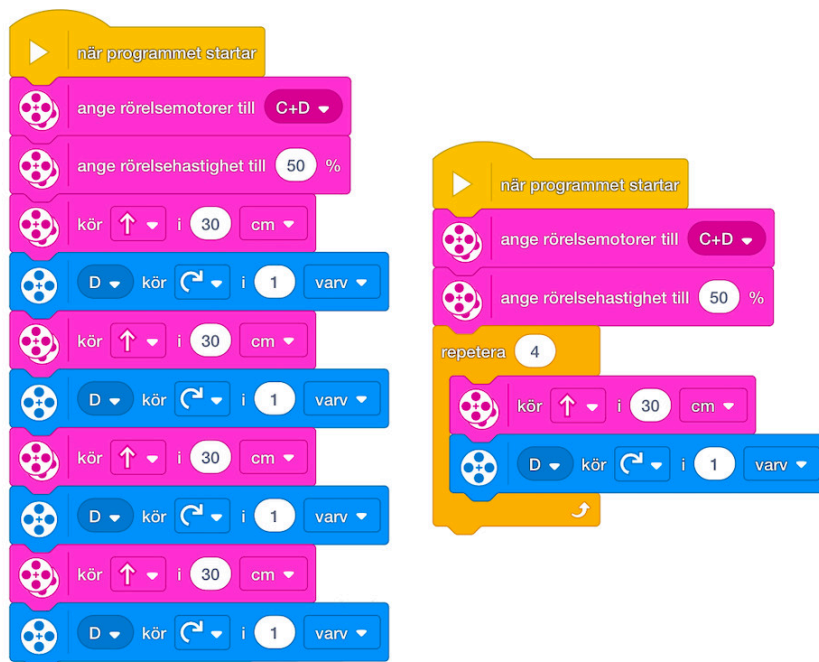


Figur 2. Exempelkod för att roboten ska köra framåt 10 cm.

Det närliggande syfte är *skapa den här koden och testa så att ni får roboten att köra framåt*. Detta syfte ger eleverna mål i sikte och de kan enkelt avgöra när de är klara, koden är också så pass enkel att de lätt kan uppmärksamma, testa och justera saker i den ifall de vill ändra något i rörelsen. Till exempel vad händer om vi ändrar **rörelsehastigheten** eller riktningen i kommandot **kör**?

När alla elever har fått roboten att röra sig framåt introduceras nästa övning. Här ska eleverna, utifrån koden de precis har konstruerat, programmera roboten så att den beskriver en kvadratisk rörelse. Det finns olika sätt att göra detta på, ett sätt är att använda en loop vilket också introduceras som ett möjligt lösningsförslag (Figur 3).

När eleverna har programmerat sin robot och är klara diskuteras därför de olika förslag som har kommit upp. De olika förslagen kan med fördel diskuteras utifrån funktionalitet, när är loop bra att använda? När är det inte? Fungerar loop om roboten skulle utföra en rektangulär rörelse?



Figur 3. Exempelkoder för att roboten ska beskriva en kvadratisk rörelse.

Materialet är genomgående uppbyggt på ovanstående sätt. Det vill säga en förflyttning mot mer komplexa program och där stötningen successivt avtar. På detta sätt skapas en kontinuitet mellan

de olika aktiviteterna där eleverna successivt kan agera mer självständigt i sin programmering.

Undervisningsmaterialets delar och upplägg

Lektionerna är tänkta att genomföras under cirka fyra timmar, fördelat på två eller tre tillfällen och består av nedanstående delar. Vi har använt en variant av den klassiska EPA-modellen när vi planerat de olika aktiviteterna. EPA står för enskilt-par-alla, och syftar till att låta elever växla mellan enskild reflektion, samtal i par eller mindre grupp och samtal i helklass. Genom att aktiviteterna rör sig mellan grupparbete till helklassamtal kan eleverna dela med sig av sina tankar och erfarenheter.

Vi har sett att parprogrammering är att föredra, snarare än att eleverna arbetar i grupper om tre eller fler. Att arbeta på detta sätt där grupperna arbetar parallellt med samma uppgifter ger eleverna möjlighet att se och diskutera alternativa lösningar som deras kamrater använt sig av. Det ger också oss lärare möjlighet att se vad eleverna har svårt med vilket utgör en grund för att kunna sammanfatta och konkretisera vad som är centralt och viktigt för den fortsatta processen framåt.

Materialets övergripande syfte är att stödja elevernas förståelse för hur man med hjälp av programmering kan styra och reglera ett föremål genom ett återkopplingssystem. Utöver de programmeringsspecifika begrepp eleverna möter i övningarna kommer de också använda mer allmänna begrepp och principer kopplade till teknikämnet.

Introduktion till robotprogrammering

Robotprogrammering introduceras där hårdvara och mjukvara diskuteras. Samtalen grundas i elevernas tidigare erfarenheter, både gällande programmering generellt och den specifika programmeringsmiljö som de kommer att arbeta i.

Närliggande syfte: Montera roboten i enlighet med anvisningen.

Övning 0

I den här övningen får eleverna lära sig vilka kodblock som krävs för att starta programmet, aktivera motorerna och att få roboten att röra sig framåt. De lär sig också att överföra koden från mjukvara till roboten.

Närliggande syfte: Skapa den här koden, och testa så att ni får roboten att köra framåt.

Övning 1

I den här övningen övar eleverna på att konstruera ett program som får roboten att följa en förutbestämd bana genom att köra framåt och sedan svänga. Genom begreppet repetera/loop introduceras eleverna till potentiellt mer funktionella lösningar.

Närliggande syfte: Bygg på koden så att roboten åker i kvadrat, och stannar när den kommit tillbaka till ursprungspunkten.

Närliggande syfte: Diskutera för- och nackdelar med de olika lösningsförslag klassen har använt.

Övning 2

I den här övningen får eleverna öva sig på att programmera roboten att med hjälp av motorer följa en böjd linje utan sensorer.

Närliggande syfte: Få roboten att följa den böjda linjen och stanna vid den röda linjen.

Närliggande syfte: Jämför er kod med en annan grupp och diskutera möjliga förbättringar man skulle kunna göra.

Närliggande syfte (hemuppgift): Prata hemma om vilka maskiner/apparater ni har som använder sig av den här typen av kod.

Övning 3

I den här övningen får eleverna montera en ljussensor på sin robot och se hur den fungerar och hur dess input visualiseras i programmet. Eleverna får öva sig på att programmera roboten att köra framåt och stanna när dess ljussensor registrerar röd färg. I den här uppgiften är fokus på att förstå hur återkopplingssystemet sensor-motorer fungerar. I efterföljande uppgift ska eleverna konstruera en kod som kan användas för en mer komplex rörelse.

Närliggande syfte: Montera och aktivera ljussensorn på roboten.

Närliggande syfte: Få roboten att köra framåt med hjälp av en loop och stanna när ljussensorn registrerar röd färg.

Närliggande syfte: Jämför er kod med lösningsförslaget, vilka skillnader finns och när används den här typen av program?

Övning 4

I den här övningen ska eleverna konstruera en kod som får roboten att följa en böjd linje med hjälp av en ljussensor.

Närliggande syfte: Använd lämpliga kodblock för att bygga en kod som med hjälp av ljussensorn följer den svarta linjen och stannar vid den röda.

Närliggande syfte: Jämför er kod med de andra gruppernas lösningar samt lösningsförslaget, vilka skillnader samt för- och nackdelar finns?

Diskussionsfrågor och uppgifter

Utöver dessa övningar, finns det i undervisningsmaterialet vi delar på Figshare [\[1\]](#) förslag på diskussionsfrågor och uppgifter för att relatera programmeringskoncepten till tekniska lösningar i elevens vardag. Ett exempel på en sådan uppgift är: "Titta hemma och prata med familjen. Vad har ni för maskiner/apparater hemma som använder sig av den här typen av kod?" ([\[1\]](#), slide 13). Ett senare förslag på diskussion handlar om programmeringens och sensorers betydelse för samhälle, individ, miljö och arbetsliv. Dessa diskussioner kan användas för att uppfylla kursplanens syfte, där en del kretsar just kring hur tekniken påverkar samhälle, individ och miljö.

Sammanfattande kommentarer

Vi är mycket väl medvetna om att utrymmet är begränsat för att genomföra omfattande programmeringsprojekt i teknikämnet. Vi tänker därför att det är viktigt att uppgifter kan knytas till de förmågor som eleverna ska få möjlighet att utveckla när de lär sig programmera inom ramen för teknikämnet. Som diskuterats i texten ser vi flera områden som kan hanteras när eleverna gör de föreslagna uppgifterna, till exempel att utvärdera tekniska lösningar utifrån dess funktionalitet eller att pröva och ompröva på ett systematiskt sätt. Vi tänker också att programmering kan användas för att introducera eller exemplifiera tekniska system där man får möta begrepp som

input, output, process, återkoppling, mål med systemet. I relation till detta kan det finnas andra förmågor som berörs i uppgiften men som vi ännu inte har noterat.

Utöver programmeringens roll i teknikundervisningen kan också lyftas att undervisningen ger en förståelse för grundläggande programmeringskoncept och begrepp, som variabler, villkorssatser och loopar, vilket i sig är värdefull kunskap som också lägger en god grund för de elever som sedan läser programmering på gymnasiet. Slutligen, vi tror det skulle vara mycket givande att testa materialet i fler sammanhang och det skulle vara mycket intressant att ta del av de justeringar och förbättringar som kan göras.

Författare

Niklas Salomonsson arbetar som teknik- och slöjdlärare i Stockholms stad. Niklas har ett försteläraryrke med inriktning mot teknikämnet.

Mattias Wickberg Hugerth är gymnasielärare i svenska, engelska och IT-ämnen i Stockholms stad. Sedan hösten 2022 är Mattias även doktorand vid Data- och systemvetenskapliga institutionen på Stockholms universitet, med fokus på artificiell intelligens i utbildning.

Per Anderhag arbetar som lektor på Stockholm stads utbildningsförvaltning samt är affilierad forskare på institutionen för ämnesdidaktik, Stockholms universitet.



Figur 4. Författarna Niklas Salomonsson, Mattias Wickberg Hugerth och Per Anderhag.

Referenser

1. Salomonsson N, Wickberg Hugerth M, Anderhag P. Teaching material to support students' understanding of programming in technology education (Version 0). 2024. <https://doi.org/10.17045/sthlmuni.25416862>
2. Anderhag P, Salomonsson N, Bürgers A, Estay Espinola C, Fahrman B, Seifeddine Ehdwall D, Sundler M. What strategies do students use when they are programming a robot to follow a curved line?. *International Journal of Technology and Design Education*. 2023;34(2). <https://doi.org/10.1007/s10798-023-09841-x>
3. Andrée M, Anderhag P, Björnhammer S, Salomonsson N. Aesthetic experience in technology education - the role of aesthetics for learning in lower secondary school robotic

- programming. *Frontiers in Education*. 2024;9. <https://doi.org/10.3389/feduc.2024.1291070>
4. Xia L, Zhong B. A systematic review on teaching and learning robotics content knowledge in K-12. *Computers & Education*. 2018;127. <https://doi.org/10.1016/j.compedu.2018.09.007>
 5. Anderhag P, Björn M, Fahrman B, Lundholm-Bergström A, Weiland M, Wållberg T. Kod som teknisk lösning: En studie om grundskoleelevers uppfattningar av ändamålsenlighet i deras spontana programspråk. *Nordic Studies in Science Education*. 2021;17(1). <https://doi.org/10.5617/nordina.7020>
 6. Johansson A-M, Wickman P-O. The use of organising purposes in science instruction as a scaffolding mechanism to support progressions: A study of talk in two primary science classrooms. *Research in Science & Technological Education*. 2018;36(1). <https://doi.org/10.1080/026535143.2017.1318272>
 7. Danielsson Thorell H, Andersson C, Jonsson A, Holst A. Är det man ser det som sker? - En designbaserad studie av en laboration med elevens perspektiv i fokus. *Forskning om Undervisning och Lärande*. 2014;(13). <https://forskul.se/wp-content/uploads/2014/09/FORSKUL13.s5%E2%80%939329.pdf>
 8. Jakobson B, Lundegård I, Wickman PO. *Lärande i handling: en pragmatisk didaktik*. Studentlitteratur; 2014.