



NATUR-MILJÖ-SAMHÄLLE

**Examensarbete i fördjupningsämnet  
Naturorientering och teknik  
15 högskolepoäng, avancerad nivå**

**Lärares uppfattningar av  
implementeringen av programmering i  
teknik**

*Teachers' Perceptions of the Implementation of Programming in  
the Technology subject*

**Katja Lindblad  
Johanna Nyhlén**

Grundläraresexamen, 240 hp  
2018-03-21

Examinator: Agneta Rehn  
Handledare: Eva Davidsson



# Förord

Detta är ett examensarbete på avancerad nivå i vårt fördjupningsämne Naturorientering och Teknik, utfört under vårterminen 2018 vid fakulteten Lärande och samhälle.

Då vi är två som tillsammans författat examensarbetet valde vi att till viss del fördela arbetsuppgifterna mellan oss för att optimera tiden vi hade till vårt förfogande. Katja ansvarade för att sammanställa studier och avhandlingar på svenska samt bar huvudansvaret för tidigare forskning och begrepp. Johanna ansvarade för att sammanställa studier och avhandlingar på engelska, skapa och distribuera enkäten, samt bar huvudansvaret för sammanfattning, teori, metod och avslutande ord. Vi hjälptes åt med att strukturera och att skriva övriga delar i samråd med varandra.

Vi vill passa på att rikta ett stort tack till de 55 tekniklärare som deltog i vår enkätundersökning och till de lärare och IT-pedagoger/utvecklingsledare som vi intervjuade. Ni bidrog alla med värdefulla åsikter! Vi vill även tacka vår handledare Eva Davidsson.

Malmö, mars 2018

Katja Lindblad och Johanna Nyhlén



# Sammanfattning

I augusti 2018 blir det obligatoriskt att följa den reviderade läroplanen (Skolverket, 2017a) där programmering har implementerats i teknik. Frågan vi ställer oss är hur väl förberedda lärarna är att undervisa i programmering då det tidigare inte funnits med i Lgr 11 (Skolverket, 2017a). Studiens syfte är att undersöka tekniklärares uppfattningar av implementeringen av programmering i teknik samt hur de uppfattar att tillgången till fortbildning och material/programvara ser ut. Detta har undersökts genom en kvantitativ enkätundersökning som 55 tekniklärare deltog i, samt sex kvalitativa intervjuer. En fenomenografisk forskningsansats har inspirerat till studien. Genom en kvalitativ analys och kategorisering av enkätsvaren och intervjuerna har uppfattningarna kring fenomenet programmering i teknikämnet och tillgång till fortbildning samt material analyserats.

Resultatet av den insamlade empirin visar att majoriteten av tekniklärarna och utvecklingsledarna ser positivt på att programmering införs i teknikämnet samt att tillgången till material/programvara är god. Tillgången till fortbildning uppges vara något mindre god än tillgången till material. Strax över hälften av lärarna uppgav att det blivit erbjudna fortbildning och färre än hälften svarade att de redan har gått någon fortbildning.

**Nyckelord:** enkäter, fenomenografi, fortbildning i teknik, intervjuer, lärare i teknik åk 4-6, uppfattningar, programmering, robotar



# Innehållsförteckning

<b>1. Inledning</b>	<b>9</b>
1.1 Syfte och frågeställningar	10
<b>3. Begrepp</b>	<b>11</b>
3.1 Datalogiskt tänkande	11
3.2 Programmering	11
<b>4. Teori</b>	<b>14</b>
<b>5. Tidigare forskning</b>	<b>15</b>
<b>6. Metod</b>	<b>20</b>
6.1 Urval	20
6.2 Genomförande	21
6.3 Forskningsetiska aspekter	23
6.4 Analysmetod	24
<b>7. Resultat och analys</b>	<b>25</b>
7.1 Vilka uppfattningar har tekniklärare av implementeringen av programmering i teknik i årskurs 4-6?	26
7.2 Vilka uppfattningar har lärarna av tillägget gällande programmering i den reviderade kursplanen i teknik?	29
7.3 Hur ser tillgången till fortbildning i programmering ut enligt tekniklärarna?	31
7.4 Hur ser tillgången till resurser/material i programmering ut enligt tekniklärarna?	32
<b>8. Slutsats och Diskussion</b>	<b>35</b>
8.1 Vilka uppfattningar har tekniklärare av implementeringen av programmering i teknik i årskurs 4-6?	35
8.2 Vilka uppfattningar har lärarna av tillägget gällande programmering i den reviderade kursplanen i teknik?	36
8.3 Hur ser tillgången till resurser och fortbildning i programmering ut enligt tekniklärarna?	37
8.4 Metoddiskussion	40
8.5 Avslutande ord	41
<b>Referenser</b>	<b>42</b>
<b>Bilagor</b>	<b>47</b>





# 1. Inledning

Regeringen gav i september 2015 Skolverket i uppdrag att föreslå nationella IT-strategier för skolväsendet som skulle "bidra till ökad måloppfyllelse och likvärdighet genom att den strategiska potential som IT har tillvaratas i hela skolväsendet" (Utbildningsdepartementet 2015, s. 1). Dessa strategier innefattar att: vid behov göra förändringar i läroplan, kursplan eller ämnesplan, att elevers it-användning sker säkert och ansvarsfullt, att anpassa och utveckla undervisningen gällande digitala läresurser utefter varje elevs behov och förutsättningar, it-användning i skolutveckling och lärare och elevers tillgång till digitala verktyg och läresurser (Skolverket, 2015).

I juni 2016 presenterade Skolverket (2016) sitt slutgiltiga förslag och i Regeringens (2017) informationsmaterial *Stärkt digital kompetens i skolans styrdokument* går det att läsa om regeringens beslut att i styrdokument förtydliga och förstärka den digitala kompetensen hos elever. De skriver att programmering ska införas som ett tydligt inslag i framförallt matematik och teknik. Skolverket (2016) betonar att även om programmering i många sammanhang förknippas med kodskrivande, utgår de från ett vidare perspektiv som innefattar logiskt tänkande och kreativ problemlösning, problemformulering, dokumentation, att pröva och ompröva samt förmågan att generalisera och att arbeta strukturerat. Skolverket (2016) lyfter även fram vikten av att belysa programmeringens demokratiska dimensioner, och hur styrning liksom reglering är något programmering kan användas till.

Programmering har tidigare inte haft någon central plats i grundskolans läroplaner. I läroplanen för grundskolan 1980 fanns datalära med som ett inslag i matematik, samhällsorienterande och naturorienterande ämnen, men undervisningen berörde främst användande av datorer och förtrogenhet med för eleven relevanta program (Skolöverstyrelsen, 1982). I kommentarmaterialet till matematik står det att "köpa en dator för att lära alla elever programmera är alltså helt felaktigt. För de allra flesta elever måste programmeringsövningar uppta en mycket liten del av den tid som anslås till undervisning om datorisering" (Skolöverstyrelsen 1982, s. 56).

I det centrala innehållet för kursplanen i teknik i årskurs 4–6 under arbetssätt, är den nya skrivningen "Att styra egna konstruktioner eller andra föremål med programmering"

(Regeringen 2017, s. 2). Den reviderade läroplanen för grundskolan, förskoleklassen och fritidshemmet kommer från och med läsåret 2018/2019 bli obligatorisk att följa (Skolverket, 2017a). Då vi utexamineras sommaren 2018 med intentionen att börja arbeta som lärare hösten 2018 är införandet av programmering i teknik högst intressant för oss. Vi intresserar oss för tekniklärares åsikter och uppfattningar gällande införandet av programmering i kursplanen i teknik. Det vi undrar är ifall lärarna har fått eller erbjuds den kompetensutveckling som är nödvändig för att de ska känna sig trygga i att undervisa i programmering i teknik.

## 1.1 Syfte och frågeställningar

Syftet med studien är att undersöka verksamma tekniklärares uppfattningar av implementeringen av programmering i teknik i årskurs 4–6, samt hur det nya tillägget i kursplanen kan tolkas. Vidare ämnar studien undersöka hur lärare uppfattar att tillgången till fortbildning och resurser ser ut på skolor i Skåne.

Frågeställningar som ämnas besvaras:

1. *Vilka uppfattningar har tekniklärare av implementeringen av programmering i teknik i årskurs 4-6?*
2. *Vilka uppfattningar har lärarna av tillägget gällande programmering i den reviderade kursplanen i teknik?*
3. *Hur ser tillgången till resurser och fortbildning i programmering ut enligt tekniklärarna?*

## 3. Begrepp

Följande kapitel beskriver begreppen datalogiskt tänkande, programmering i en skolundervisningskontext samt förklarar vad analog programmering respektive programmering med dator innebär.

### 3.1 Datalogiskt tänkande

Enligt Wing (2006) är datalogiskt tänkande (computational thinking) en grundläggande färdighet som alla bör ha. Det är en färdighet som innefattar problemlösning, analytiskt och logiskt tänkande, abstraktion, generalisering och kreativitet (Bocconi, Chiocciariello & Earp, 2018; Wing, 2006). Voogt, Fisser, Good, Mishra och Yadav (2015) menar att det innefattar de tankeprocesser som är involverade när komplexa problem löses och generaliseras för att sedan kunna överföras till en mängd olika problem. I de Nordiska länderna fokuserar datalogiskt tänkande ofta på problemlösning och digital kompetens som bland annat innefattar en kreativitet med digitala verktyg samt en förståelse av digital teknik och förmågan att vara en kritisk användare (Bocconi et al., 2018).

### 3.2 Programmering

Programmering är ett relativt nytt begrepp inom den nordiska skolkontexten då programmering inte tidigare haft en klar roll i läroplanen, och det är därmed viktigt att tydliggöra dess innebörd. Programmering associeras ofta med en stereotypisk syn om att det är ett ensamt arbete med fokus på kodning (Kjällander, Åkerfeldt, Mannila & Parnes, 2018), men Skolverket (2017b) betonar i kommentarmaterialet "Få syn på digitaliseringen på grundskolenivå" att programmering handlar om mer än att koda:

*"att skriva kod, vilket har stora likheter med generell problemlösning. Det handlar bland annat om problemformulering, att välja lösning, att pröva och ompröva samt att dokumentera. Men programmering ska ses i ett vidare perspektiv som även omfattar kreativt skapande, styrning och reglering, simulering samt demokratiska dimensioner. Det här vidare perspektivet på programmering är en viktig utgångspunkt i undervisningen och programmering ingår därmed i alla aspekter av digital kompetens"* (Skolverket 2017b, s. 10)

Enligt Kodboken (u.å) kan programmering definieras som att skriva instruktioner för hur problem ska lösas på ett språk som en dator kan förstå. Det finns många olika programmeringsspråk och de olika språken är bra för olika saker. Med programmering går det att skapa många olika saker som spel, musik eller appar (Kodboken, u.å). Nygård (2015) definierar begreppet som en problemanalys och abstraktion som innefattar en strukturplanering av kod.

Programmering som sker utan användning av digitala medier kallas för analog programmering (Skolverket, 2017c). Att använda sig av analog programmering i form av olika lekar kan vara en bra inkörsport för elever innan de ger sig på att programmera med dator. Hermansson och Wickholm (2017) påtalar i sin artikel "*Rutnät, robotar och koder - Om programmering i skolan*" att elever genom olika lekar till exempel robotlekar, danslekar eller lekar med rutnät, får inblick i olika begrepp, så som begreppet "kommando". I dessa lekar gav eleverna valfria, enkla antingen muntliga eller skriftliga kommando till kamraterna, vilka i sin tur skulle utföra den av kamraten önskade handlingen. Projektet som Hermansson och Wickholm (2017) utförde under en termin i en årskurs 4, gick ut på att introducera och arbeta vidare med den grundläggande programmeringen. I början av projektet, då de arbetade med analog programmering, noterade Hermansson och Wickholm (2017) tydliga framsteg av elevernas rumsuppfattning och elevernas förmåga att orientera sig och konstruera algoritmer då de utförde lekar med rutnät. Den analoga programmeringen hjälper eleverna att bygga upp sin förståelse och sitt datalogiska tänkande.

Programmering med dator har vi valt att främst avgränsa till *visuell programmeringsmiljö* såsom *blockprogrammering* (Skolverket, 2017c). Vi upplever att denna typ av programmering, då man använder sig av färdiga pusselbitsliknande block i olika färger (Thoreson, 2016) istället för kodning, är mest aktuell för elever i årskurs 4–6 och passar in i Skolverkets utgångspunkter.

*Blockprogrammering* kan således liknas vid att lägga pussel med olika block (Skolverket, 2017c). När de förkonstruerade blocken byggs ihop bildas det ett program till datorn. Blocken representerar instruktionerna till datorn, det vill säga koden. Varje programspråk har sin egen syntax som måste följas strikt, genom att istället använda färdiga block kan missöden med ofullständig kod på grund av stavfel eller utelämnat/felplacerat tecken undvikas (Skolverket, 2017c). Scratch är exempel på ett *grafiskt* eller *visuellt programmeringsspråk* (på engelska kallas detta "Visual Programming Languages", VPL) vilket ofta används av elever i

utbildningssyfte (Kjällander, Åkerfeldt & Petersen, 2016). När det gäller de yngre barnen används ofta fysiska objekt som till exempel Bee-Bot, som är en insektsliknande robot, vilka eleverna via programmering får att utföra olika rörelser (Kjällander et al., 2016).

## 4. Teori

Studien har inspirerats av en fenomenografisk forskningsansats (Claesson, 2007). I denna studie utgörs fenomenet av uppfattningar av programmering i teknik.

Den fenomenografiska ansatsen utvecklades av Ference Marton och en forskargrupp vid Göteborgs universitet (Lundgren, 2015). Martons forskningsgrupp, som forskade i inläring och omvärldsuppfattningar, engagerade sig i att undersöka samband mellan människors tankar om studier och hur de sedan genomförde sina studier samt hur deras tänkande förändrades då de bedrev sina studier (Claesson, 2007). Senare kom flertalet fenomenografiska studier att fokusera på kartläggning av variationen av uppfattningar av vissa fenomen. Inom fenomenografisk forskning kallas det för ett ”andra ordningens perspektiv” då syftet är att undersöka människors tankar eller uppfattningar om något, det vill säga om ett visst fenomen (Claesson, 2007). Vanligtvis genomförs intervjuer som sedan transkriberas och analyseras så att kvalitativa skillnader åskådliggörs (Claesson, 2007). Olika beskrivningskategorier skapas och förhållandena sinsemellan studeras sedan i ett så kallat fenomenografiskt utfallsrum (Lundgren, 2015).

## 5. Tidigare forskning

Då dagens elever lever i ett alltmer digitaliserat samhälle har regeringen beslutat om förändringar i styrdokumentet. Dessa förändringar ska bidra till att eleverna utvecklar förståelse för digitaliseringens betydelse, både när det gäller individen och samhället, samt att eleverna ska kunna agera på ett ansvarsfullt sätt i digitala sammanhang (Skolverket, 2017d). För att vi i framtiden ska kunna säkra ett demokratiskt samhälle är det viktigt att det skapas en förståelse för samhälls- och teknikutvecklingen, då krävs det att vi alla lär oss hur den digitala världen fungerar (Runeson, Andersson, Heintz, Mannila & Rolandsson, 2015).

Mannila hävdar enligt Lindström (2016, 18 mars) i ett uttalande att det saknas forskning om implementering av programmering i skolan men att programmering numera är en del av allmänbildningen. Även Kjällander et al. (2016) påpekar att forskning om programmering i svensk skola är limiterad.

Resultatet från en undersökning av EU-kommissionen i oktober 2015 visade att 16 av 21 länder i Europa (samt Israel) infört programmering i sina läroplaner (Kjällander et al., 2016). Kjällander et al., (2016) har tittat närmare på implementeringen av programmering i England och Finland, då de båda länderna infört programmering i läroplanen hösten 2014 respektive hösten 2016. Anledningen till att de valde just England och Finland, är för att likheten med den svenska skolan är stor, och de anser att vi kan lära oss mycket av de erfarenheter som gjorts där. En lärdom som Kjällander et al., (2016) menar att vi kan dra är att en stor utmaning vid införandet av programmering, var och kommer fortsättningsvis att vara att fortbilda alla lärare. Finlands utbildningsminister Sanni Grahn-Laasonen menar att lärarnas kompetensutveckling måste stöttas och då måste investeringar göras i lärarfortbildningar (Wallin, 2017, december). När den nya läroplanen trädde i kraft i Finland fanns det ett stort behov av fortbildning för lärare. Staten åtgärdade detta genom att finansiera olika utvecklingsprojekt som genomfördes runt om i landet (Kjällander et al., 2018), och utbildnings- och kulturdepartementet gav bidrag åt en del projekt startade av lärare och utbildningsforskare (Wallin, 2017, december).

En annan utmaning som Grahn-Laasonen lyfter fram har varit allmänhetens ifrågasättande av att lära ut programmering i skola, att många inte förstår varför barn behöver förstå hur kodning fungerar (Wallin, 2017, december). Inför införandet av programmering i den nya

svenska reviderade läroplanen, har många gjort sina åsikter hörda. Majoriteten verkar vara överens och ser positivt på införandet av programmering i skolan. Czajkowski (2017, 7 april) påtalar i sin debattartikel i DN Åsikt att införandet av programmering i skolan är ett beslut i rätt riktning, då det snart är brist på programmerare i Sverige. Czajkowski (2017, 7 april) menar även att om barn tidigt har möjlighet att intressera sig för programmering, kan detta gynna deras framtida yrkesval och förutsättningar att få ett bra jobb. Ett svar på Czajkowski artikel kommer från von Zweigbergk (2017, 10 april), som menar att införandet av programmering inte ska göra alla elever till programmerare, utan han ser det mer som att skolan skapar allmänbildade medborgare.

Statens offentliga utredning (SOU, 2016:89) förespråkar programmering i skolan och menar att detta kan ge eleverna kunskaper för att kunna hantera den digitala vardag och de digitala verktyg som de kommer i kontakt med. Voogt et al. (2015) resonerar i sin artikel om vad datalogiskt tänkande innebär samt vad som bör undervisas och hur. En av deras slutsatser är att programmering är ett viktigt redskap för att utveckla datalogiskt tänkande, och att lärare bör bekanta sig med de grundläggande begreppen i datalogiskt tänkande om de ska undervisa i programmering. Införandet av programmering i skolan ger både lärare och elever ökad möjlighet att få tillgång till ett tekniskt språk, vilket i sin tur ger elever och lärare ökade möjligheter att ställa högre krav på tekniska system.

Kjällander et al. (2016) menar att det finns tre utmaningar som framkommit i internationell forskning som kan vara viktiga aspekter att ha i åtanke gällande införandet av programmering i den svenska skolan.

1. **Att det är brist på forskning.** Kjällander et al. (2016) visar i sin översikt att i flera av de dokument som ligger till grund för beslut av införandet av programmering i skolan, finns det brister gällande kopplingar till forskning. Avsaknaden av vetenskapligt granskad forskning i Sverige kan bero på att programmering har inte funnits i skolan sedan 1980-talet och då var det endast gymnasieelever som hade undervisning i detta.
2. **Att det i Sverige är brist på lärare som kan undervisa i programmering.** Sveriges pedagoger har ännu inte hunnit få den utbildning och kompetens som krävs för att undervisa i programmering (Kjällander et al., 2016). Rolandsson (2015) belyser betydelsen av lärarens kunskaper och centrala roll som avgörande för hur ämnet



programmering utvecklas i klassrummet och menar: *“Som expert på det ämnesdidaktiska ämnesområdet vill jag därför rekommendera den svenska regeringen att skapa möjligheter för lärare att fördjupa sina ämnesdidaktiska kunskaper inom datavetenskap”* (Rolandsson 2015, s. 60).

3. **Brister när det gäller de digitala resurserna.** Både att hitta verktyg för lärande och verktyg för bedömning av elevernas programmering, ser Kjällander et al. (2016) som en stor utmaning för den svenska skolan.

Nya ämnen medför både utmaningar och iver. Lärare behöver utveckla lämplig pedagogik för att undervisa i det nya ämnet, särskilt då det gäller programmering och datalogiskt tänkande, eftersom programmering anses vara den mest utmanande delen av datavetenskap (Sentance & Csizmadia, 2017). Kjällander et al. (2018) anser att det finns ett behov av att utveckla metoder för undervisning och bedömning eftersom många av dagens lärare i Sverige saknar erfarenhet och kunskap kring att utbilda i 2000-talets förmågor såsom programmering. Vidare påpekar de hur viktigt det är att även lärarutbildningar utvecklas för att ge blivande lärare rätt kompetenser. Då universitet och högskolor i stor utsträckning inte tillhandahållit den utbildning som lärare och lärarstudenter behövt har den marknaden öppnats upp för kommersiella företag och organisationer. I en studie av Kjällander et al. (2018) utforskades tre olika nordiska, ett från Finland och två från Sverige, makerspace-initiativ gällande hur maker-rörelsen och lärarutbildning i programmering kan influera varandra. I så kallade makerspaces kan intresserade träffas för att i en workshop-liknande miljö samarbeta med varandra och utbyta kunskaper och idéer kring programmering och teknologi. Deras studie (Kjällander et al., 2018) visar att Maker-rörelsen, en slags utvidgning av gör-det-självrörelsen med en teknikinriktat fokus, kan fungera som ett stöd då programmering ska introduceras i grundskolan och att lärandemiljöer kan utformas så att både elever och lärarstudenter kan lära sig programmering.

Rolandsson (2015) betonar i sin studie kring lärares perspektiv på programmering i skolan att undervisning i programmering verkar vara en utmaning och att många elever inte har en förståelse för ämnet. Vidare pekar Rolandssons (2015) studie på att det finns två typer av lärare; de med synsättet att alla elever kan lära sig programmering och de som anser att

programmering inte är för alla. Det framkom även att lärare tror att elevernas logiska och analytiska förmågor är avgörande för deras läranderesultat.

I en studie i Storbritannien 2014 genomfördes en enkätundersökning med över 300 lärare i syfte av att analysera lärarnas perspektiv på utmaningar och strategier i samband med undervisning av datavetenskap (Sentance & Csizmadia, 2017). Lärarna fick bland annat svara på vilka utmaningar de ansåg fanns och de fem vanligaste utmaningarna var: lärarnas egna ämneskunskaper, elevers bristande förståelse av ämnesinnehållet, tekniska problem i skolan, differentiering för att möta elevers olika kunskapsnivåer samt elevers vilja och förmåga att lösa problem. En åtskillnad gjordes mellan intrinsiska och extrinsiska utmaningar (Sentance och Csizmadia, 2017). Till intrinsiska utmaningar räknades lärarnas ämnesdidaktiska kunskaper, deras förmåga att anpassa undervisningen utefter elevers olika kunskapsnivåer inom samma klass och deras tillvägagångssätt vid undervisning av ämnet. Extrinsiska utmaningar innefattar tekniska problem, bedömning, brist på resurser, tid och support samt skolan och andras förväntningar. Lärare kan självständigt arbeta med sina intrinsiska utmaningar såsom sina ämneskunskaper, men extrinsiska utmaningar såsom brist på resurser och tekniska problem som läraren inte själv kan råda över kan ge upphov till frustration (Sentance och Csizmadia, 2017).

Många lärare behöver få öva mer och därigenom bygga upp sitt självförtroende gällande sina pedagogiska förmågor (Sentance & Csizmadia, 2017). Det krävs stöd från rektorer och arbetslag såväl som tillgång till välfungerande tekniska lösningar.

Sentance och Csizmadia (2017) anser att ett av huvudsyftena med undervisning i datavetenskap bör vara att eleverna ska utveckla sitt datalogiska tänkande (computational thinking). Enligt riktlinjer som framtagits för lärare i England delas datalogiskt tänkande in i fyra överlappande förmågor: abstraktion, algoritmiskt tänkande, generalisation och utvärdering.

Kilbrink (2008) har i sin studie undersökt elevernas uppfattning av lärande vid arbete med att programmera och konstruera legorobotar. Studien utfördes med hjälp av intervjuer av elever i årskurs 4-9 som arbetat med programmerbart legomaterial. Kilbrink (2008) menar att det är viktigt att integrera både teori och praktik vid programmeringsundervisning med robotar. Detta för att eleverna ska ha möjlighet att få en djupare förståelse i teknikämnet och

inte bara vara nöjda och stanna efter att ha löst en specifik uppgift, utan kunna se kopplingen mellan förståelse och görande eller teori och praktik. Lektionsupplägget är således viktigt för att undervisningen med robotar ska tillföra eleverna något och utifrån resultaten från studien menar Kilbrink (2008) att lärare bör uppmuntra till kommunikation och reflektion för att skapa förutsättningar till djupinläring.

## 6. Metod

Vid insamlingen av empirin valde vi att kombinera en kvantitativ enkätundersökning med fem kvalitativa intervjuer. Tillvägagångssättet beskrivs utförligt så att studien ska vara möjlig att upprepa och på så vis nå en hög reliabilitet. Reliabilitet syftar till om forskningsresultaten är upprepningsbara eller ej (Alvehus, 2013).

En metoddiskussion kring konsekvenser av metodval förs i avsnitt 8.4.

### 6.1 Urval

Vi valde att använda oss av flera olika urvalsprinciper. Ett strategiskt urval medför att urvalet “utformas specifikt utifrån de undersökningsfrågor som ställs” (Alvehus 2013, s. 67). Vi valde att enkäten enbart skulle skickas ut till skolor i Skåne eftersom vårt syfte var att utforska vilka uppfattningar som finns i Skåne. Det strategiska urvalet innebar även att enkäten endast skulle skickas ut till behöriga lärare i teknik i årskurs 4-6. Det systematiska urvalet innebar att varannan skola som var listad på respektive kommuns sida skulle väljas, förutsatt att det var en kommunal skola med årskurserna 4-6, tills dess att vi hade fyra skolor från varje kommun (i de fall det fanns så många skolor). Ett subjektivt urval innebär att urvalet “handplockas” av forskaren som medvetet väljer personer utefter sin kännedom om de företeelser som ska undersökas och forskningsfrågans bakgrund (Denscombe, 2009). Vi valde att intervjua två teknicklärare (Lärare A och B i avsnitt 6.2) som arbetar i skilda kommuner och som enligt vår kännedom har olika erfarenheter av programmering och som även besvarade enkäten.

Vi valde även att genomföra intervjuer med IT-pedagoger/utvecklingsledare i tre olika kommuner baserat på respektive kommuns teknicklärares svar i enkätundersökningen samt vår kännedom om kommunerna. Den första kommunen valdes på grund av att vi inte lyckades få in några enkätsvar därifrån trots att vi visste att flera skolor i kommunen redan arbetar med programmering. Valet av den andra kommunen baserades på svaren från enkätundersökningen då vi fick ett väldigt snabbt gensvar från teknicklärarna som var verksamma i den kommunen och teknicklärarna angav att de erbjudits fortbildning kring programmering. Den tredje kommunen valdes för att den enligt vår kännedom inte ligger i framkant vad gäller digitalisering och programmering.

## 6.2 Genomförande

Då vårt examensarbete skrivs under en tidsbegränsad period och vi önskade undersöka uppfattningen bland lärare i hela Skåne ansåg vi att en enkätundersökning var en lämplig datainsamlingsmetod. Ejvegård (1996) och Kylén (2004) menar att när det gäller insamling av data är enkätundersökningar fördelaktiga då de kan nå ut till många personer som alla besvarar samma frågor.

Enligt Bourque och Fielder (1995) samt Eljertsson (2005) är det viktigt att tydligt och kortfattat informera respondenterna om studiens syfte, vem som står bakom den, samt hur enkäten ska besvaras och hur respondenterna kan kontakta de ansvariga. Vidare ska respondenten göras medveten om eventuell anonymitet, att undersökningen är frivillig och att svaren behandlas konfidentiellt (Eljertsson, 2005). Vi valde att i följebrevet (se bilaga 1) informera respondenterna om detta.

Det finns tre huvudtyper av frågor: öppna frågor, frågor med fasta alternativ samt frågor med skalor (Kylén, 1994). Valet av frågetyp påverkar möjligheterna att analysera och statistiskt bearbeta frågorna (Kylén, 2004). Vi har i enkätundersökningen använt oss utav frågor med fasta alternativ (ja eller nej) som har kompletterats med en uppmaning om att respondenterna kortfattat ska utveckla sina svar, med förhoppningen om att då få en bredare bild. Eljertsson (2005) påpekar att frågor med möjlighet för respondenten att utveckla eller fördjupa sitt svar endast till viss del kan ge en möjlighet till fördjupning.

Vid konstruktionen av enkäten i SurveyMonkey, en onlineplattform, valde vi att inspireras av Kyléns (2004) fyra steg; första utkastet, första försöksversionen, slutversionen och skicka ut, samla in och gör en påminnelse. Vi skapade ett första utkast som vi lät två personer samt vår handledare granska. Detta utkast ledde sedan till vår första försöksversion som mailades till sju lärarstudenter och vår handledare.

Vi mailade 130 skolor runt om i Skåne och bad om mailadresser till deras tekniklärare så att vi skulle kunna maila enkäten direkt till lärarna (se bilaga 2). Då vi upptäckte att det var många rektorer och skolassistenter som vi inte fick svar ifrån valde vi att maila till fler skolor än de som slumpmässigt valts men tyvärr är det sammanlagt nio kommuner, av Skånes 33, som inte finns representerade i enkätundersökningen eftersom vi inte fick tag på någon tekniklärare därifrån. Enkäten (se bilaga 3) mailades till 101 tekniklärare i 24 olika kommuner

i Skåne varav 55 besvarade enkäten, vilket ger en svarsfrekvens på 54%. En påminnelse skickades ut till de respondenter som inte hade besvarat enkäten inom fem dagar, och ytterligare en påminnelse skickades ut fem dagar efter den första.

Både enkätfrågor och intervjufrågor utformades med avsikt att ge oss underlag till att besvara våra tre frågeställningar. De fyra stegen gav oss en möjlighet att försäkra oss om att vi verkligen mätte det vi avsåg att mäta, alltså att validiteten var hög. Validitet syftar till om det som avses undersökas verkligen är det som undersöks (Alvehus, 2013).

Vetenskapsrådet (2017) menar att ett vanligt misstag "kan vara att överskatta betydelsen av de resultat man fått och att utvidga deras bärkraft långt utanför det område där man har funnit att de gäller." (s. 28). Då vi inte fick in enkätsvar från Skånes alla kommuner och dessutom inte valde att kontakta alla skolor i varje kommun, anser vi att resultatet inte kan sägas representera hur det ser ut i alla skolor i Skåne och att alla lärare resonerar så som respondenterna i vår studie. Däremot ger resultatet en indikation på hur det kan se ut i Skåne.

Semistrukturerade intervjuer genomfördes som ett komplement till enkätundersökningen med avsikt att nå en djupare förståelse av vårt problemområde eftersom vi då har möjlighet att ställa följdfrågor. Enligt Eljertsson (2005) ger intervjuer möjlighet till fördjupning eftersom mer komplicerade frågor och följdfrågor kan ställas. En semistrukturerad intervju karakteriseras av en flexibilitet gällande ämnenas eller frågornas ordningsföljd. Det viktiga är inte i vilken ordning de förutbestämda frågorna besvaras utan att låta den intervjuade utveckla sina idéer och synpunkter, och att hålla svaren öppna (Denscombe, 2009). Vi valde att ha sex förutbestämda frågor som skulle ställas vid varje intervju (se bilaga 4), detta för att kunna ge utrymme åt passande följdfrågor. Alvehus (2013) förespråkar ett färre antal förutbestämda frågor eftersom intervjun annars riskerar att fokusera mer på att endast få svar på de frågorna än att höra den intervjuades berättelse. De fem intervjuerna (vid intervju 3 deltog två IT-pedagoger) tog cirka en halvtimme var att genomföra och genomfördes på respondenternas respektive arbetsplats, bortsett från intervjun med person 4 som genomfördes via mail.

Vi har valt att kalla lärarna för lärare A respektive lärare B istället för att använda deras riktiga namn. IT-pedagogerna/utvecklarna nämns som person 1, person 2, person 3 och person 4.

1. **Lärare A** har undervisat i NO/teknik i 16 år och arbetar för närvarande på en F-6 skola med omkring 350 elever i sydvästra Skåne.
2. **Lärare B** har undervisat i NO/teknik i cirka 35 år och arbetar för närvarande på en F-9 skola med omkring 700 elever i nordvästra Skåne.

3. **Person 1, i kommun 1**, har arbetat som IT-pedagog/utvecklingsledare i 7-8 år, och är i grunden lärare i Ma/NO i åk 1-7.  
**Person 2, i kommun 1**, har arbetat som IT-pedagog/utvecklingsledare i 5 år, och är i grunden lärare i Sv/SO i åk 1-7.
4. **Person 3, i kommun 2**, har arbetat som IT-pedagog i 1,5 år, och är i grunden gymnasielärare i SO/historia men har arbetat som lärare i ma/No i åk 4-6 i 4-5 år i sin kommun innan han blev IT-pedagog.
5. **Person 4, i kommun 3**, arbetar sedan hösten 2017 som utvecklare inom didaktik på Utbildningsförvaltningen. Dessförinnan arbetade personen som lärare i 17 år.

### 6.3 Forskningsetiska aspekter

Vetenskapsrådet (2002) förespråkar vid forskning inom humanistiska och samhällsvetenskapliga ämnen, fyra allmänna huvudkrav som härrör från det grundläggande individskyddskravet: informationskravet, samtyckeskravet, konfidentialitetskravet och nyttjandekravet (Ibid., s. 6). Informationskravet innebär att de medverkande ska informeras om syftet med den aktuella forskningsuppgiften, att deras deltagande är frivilligt och att de har rätt att när som helst avbryta sin medverkan. Samtyckeskravet innebär att forskaren ska ha de medverkandes samtycke. Konfidentialitetskravet medför att personuppgifter ska lagras på ett säkert sätt som inte ger obehöriga tillträde, och att uppgifter ska avrapporteras så att inga deltagande personer ska kunna identifieras av utomstående. Nyttjandekravet innebär att de uppgifter om enskilda personer som samlats in under forskningen endast får användas för forskningens ändamål och inte användas eller utlånas till andra icke-vetenskapliga syften.

I följebrevet till enkäten (se bilaga 1) informerades respondenterna om syftet med studien och intentionen med enkätundersökningen samt att de upplystes om anonymitet och att medverkan var frivillig. På liknande vis informerades de intervjuade om de fyra forskningsetiska principerna före intervjun, de gav även sitt samtycke till att intervjun ljudinspelades och de gjordes medvetna om att de närsomhelst under intervjun hade rätt att avbryta den. Vidare informerades de om att ljudinspelningen samt transkriberingen endast skulle vara tillgängliga för oss som genomförde studien, och att ljudinspelningen skulle raderas när transkriberingen var gjord. Resultaten redovisas på ett sådant sätt att det inte går att identifiera enskilda personers uttalanden eller svar, och resultaten kommer endast att användas för de ändamål vi informerat berörda om.

## 6.4 Analysmetod

Då enkätsvaren skulle sammanställas gjordes detta efter indelning i olika beskrivningskategorier, såsom; ålder, vilken kommun de undervisar i och om de gått någon fortbildning. Kategoriseringen gjordes för att vi i ett så kallat fenomenografiskt utfallsrum (Lundgren, 2015) skulle kunna se skillnader och likheter av fenomenet tydligare, och för att kunna utforska ifall faktorer som till exempel ålder och undervisningserfarenhet kan ha någon inverkan på inställningen mot programmering och uppfattningarna av tillägget i kursplanen.

Intervjuerna ljudinspelades och ljudinspelningen avlyssnades ett flertal gånger och transkriberades sedan delvis. En kvalitativ innehållsanalys (Alvehus, 2013) hjälpte oss att se mönster och skillnader, vilket var till stor hjälp då vi skulle organisera all empiri. De olika beskrivningskategorierna som framkom då enkätsvaren analyserades sammanfattades sedan i teman, vilket gav en tydlig och övergripande bild av resultatet av intervjuerna och enkäterna samt förenklade arbetet i analysen. Vi valde att dela upp den tredje frågeställningen i två teman, men i övrigt är våra teman enlighet vår studies frågeställningar; *Vilka uppfattningar har tekniklärare av implementeringen av programmering i teknik i årskurs 4-6? Vilka uppfattningar har lärarna av tillägget gällande programmering i den reviderade kursplanen i teknik? Hur ser tillgången till fortbildning i programmering ut enligt tekniklärarna? Hur ser tillgången till resurser/material i programmering ut enligt tekniklärarna?*



## 7. Resultat och analys

Studiens syfte är att undersöka lärares uppfattningar av implementeringen av programmering i teknik samt tillgången till fortbildning och material/programvaror. Utifrån en fenomenografisk analysmetod har svaren från enkätundersökningen samt delar av intervjuerna sammanställts och kategoriserats i fyra teman som skapats utifrån våra tre frågeställningar. Vi har valt att i resultatdelen dela upp den tredje frågeställningen i två teman; 7.3 samt 7.4. Vår första frågeställning besvaras i avsnitt 7.1, medan den tredje frågeställningen besvaras i avsnitt 7.2. I respektive tema går vi djupare in på resultatet och ger exempel på vad respondenterna i enkätundersökningen samt intervjuerna uttryckt. Vi gör även egna kortare reflektioner.

Resultatet i stort (se tabell 1) visar att lärarna är väldigt positiva till att programmering har införts i teknik och menar att det är nödvändigt om skolan ska följa med i samhällsutvecklingen och för att eleverna ska få en bra grund att stå på. 60% av lärarna, alltså 33 av 55, har blivit erbjudna/tillfrågade om fortbildning, men mindre än hälften uppger att de gått någon, vilket bland annat kan bero på att fortbildningen ännu inte ägt rum. Vidare uttrycker ungefär två tredjedelar av lärarna att de inte känner sig säkra på att undervisa i programmering utefter det nya tillägget i kursplanen i teknik och efterfrågar mer fortbildning. Troligtvis känner de sig osäkra för att de kanske har begränsad erfarenhet av programmering och att undervisa i det.

<b>Undervisat i antal år</b>	<b>Positiv syn på programmering</b>	<b>Säker på att undervisa</b>	<b>Gått fortbild.</b>	<b>Erbjuden fortbild.</b>
0-2 år	11/12	4/12	6/12	5/12
3-5 år	12/12	6/12	2/12	6/12
6-8 år	4/4	1/4	2/4	1/4
9-11 år	8/9	1/9	3/9	7/9
15-17 år	8/8	5/8	5/8	7/8
18-20 år	9/9	2/9	4/9	6/9
21-23 år	0/0	0/0	0/0	0/0
Fler än 24 år	1/1	1/1	1/1	1/1
<b>Totalt</b>	<b>53/55</b>	<b>20/55</b>	<b>23/55</b>	<b>33/55</b>

Tabell 1: Övergripande tabell över svarsfrekvensen på fyra av enkätfrågorna.

## 7.1 Vilka uppfattningar har tekniklärare av implementeringen av programmering i teknik i årskurs 4-6?

Enkätsvaren visar att 53 av de 55 tekniklärare som medverkade, ser positivt på införandet av programmering i teknik i årskurs 4-6. De två teman med flest antal åsikter (se tabell 2) är den som handlar om att programmering är en del av samhällsutvecklingen och framtiden, och temat som handlar om att införandet av programmering i skolan ger eleverna en bra grund att stå på.

Åsikter	Antal
Samhällsutveckling/framtiden	17
Ger elever en grund	11
Lärares kompetens/behörighet	5
Undervisning	8
Väcker intresse/nyfikenhet	5
Ger ej elever något	1

Tabell 2: Åsikter kring införandet av programmering i teknik: sorterat i teman

IT-pedagogerna och utvecklingsledaren ser också positivt på införandet av programmering i teknik. IT-pedagogen i kommun 2 lyfter att implementeringen av programmering kan ge teknik en högre status och mer utrymme, men IT-pedagogerna i kommun 1 anser att införandet har förstorats och fått mycket fokus.

**Person 1** *“Programmering har fått ett lagom utrymme i styrdokumentet men det står inte i proportion till det utrymme det fått i samhället. Skolverket har slått på den stora trumman vad gäller programmering också även om det är så mycket mer än bara programmering.”*

I kommun 1 och 2 talar man om att det var rätt beslut att inte göra programmering till ett eget ämne och menar att det annars skulle blivit ett ännu större problem med fortbildningsbehovet. Vidare påtalar de att det kommer att finnas olika utmaningar, och Lärare A påpekar i intervjun att det är viktigt att tänka på hur programmering knyts an till ämnet och att det ska finnas ett tydligt syfte, medan IT-pedagogerna i kommun 1 även framhäver en allmänbildningsaspekt och gör en liknelse med kemi och biologi.

**Person 1 & 2** *“Vi brukar göra jämförelsen med att man har länge läst biologi och kemi men det är inte tänkt att alla ska bli biologer eller kemister. Samma sak är det att inte alla ska bli programmerare eller kodare utan behöver bara förstå omvärlden omkring oss. Till exempel att förstå varför min google-sökning dyker upp i mitt facebook-flöde”*

Majoriteten av lärarna som besvarat enkäten talar om att programmering i skolan behövs i dagens samhälle och menar att skolan måste följa med i utvecklingen. Utvecklingsledaren i kommun 3 menar att revideringen av kursplanen kan se till att eleverna lär sig ett datalogiskt förhållningssätt som behövs för att kunna förstå samhällets digitalisering. Många lärare nämner även att det är viktigt att synliggöra den programmering som finns runt omkring oss och att skapa en viss förståelse för den.

**Enkät svar** *“Då många funktioner i vårt samhälle är programmerade för att fungera, t.ex trafikljus, är det viktigt att vi synliggör detta för våra elever. Samt att dagens yrken utvecklas med olika tekniker och då måste skolan hänga på. Vi utbildar vår framtid!”*

De lärare som menar att programmering i skolan ger eleverna en bra grund att stå på inför framtiden säger till stor del att det handlar om att ge eleverna rätt förutsättningar för deras framtida liv.

De två lärarna som svarade att de inte ser positivt på införandet av programmering i teknik uttrycker en oro över att programmeringen inte kommer att vara nyttig för eleverna som redan spenderar mycket tid framför skärmar, respektive en oro över att den egna kompetensen inte räcker till, en oro som skulle kunna bidra till negativiteten.

**Enkät svar** *“Vi ligger back med rena faktakunskaper och eleverna sitter redan vid skärmar så det räcker o blir över. Jag är övertygad om att arbeta med tex spagettibroar el dyl ger mer i längden.”*

Även lärare B uttrycker i intervjun en viss skepsis gentemot implementeringen och ställer sig frågan varför eleverna måste lära sig att just styra saker.

**Lärare B** *“Det är extremt få som har nytta av att kunna styra andra saker. Även om det skulle vara att du ska styra kylskåpet eller dammsugaren hemma så behöver du inga programmeringskunskaper. Vi har många elever som är praktiska och som säkert skulle tycka att det är jättekul, och därför skulle bli mer intresserade av datorer och programmering. Det är möjligt. Men de som verkligen i det fysiska livet sen, i arbetslivet, som ska använda detta. Det måste vara få personer.”*

Enkätsvaren visar att 19 av 55 lärare känner sig säkra i att undervisa i programmering utifrån det nya tillägget i teknik. Utav dessa 19 har 13 tidigare undervisat i programmering och 10 lärare angett att de gått fortbildning. Faktorer som kan bidra till att lärarna känner sig säkra i att undervisa i programmering kan vara att lärarna har någon i sin närhet som de kan få stöd och hjälp av om det behövs. Det kan även bero på ett eget intresse av programmering eller att de redan gått någon fortbildning i programmering.

**Enkät svar** *“På vår skola har vi en förstelärare som har stor kunskap om programmering. Hon har haft kurser med oss och det har varit lärorikt.”*

**Enkät svar** *“Eftersom mitt intresse bygger mycket på detta och jag har möjlighet att kombinera det med mer fortbildning så ser jag fram emot förändringarna.”*

Majoriteten av de tillfrågade lärarna, 36 av 55 stycken, känner sig osäkra på att undervisa i programmering. Utav dessa 36 var det 10 lärare som tidigare undervisat i det och 12 som gått fortbildning. I enkäten har lärarna i kommentarsfältet angett att osäkerheten till största delen beror på att de inte fått någon/fått för lite fortbildning samt att de saknar undervisningsmetoder för hur programmeringen ska användas i undervisningen.

**Enkät svar** *“Saknar konkreta arbetsexempel för att implementera tillägget i undervisningen varvid en viss osäkerhet uppstår kring mina förkunskaper när jag möter eleverna.”*

22 av 55 lärare anger att de tidigare har undervisat i programmering, vilket är fler än de som uppger att de känner sig säkra. Kan det kanske vara så att de känner sig osäkra när det nu finns ett specifikt innehåll de ska följa? Finns det möjlighet att programmering tidigare endast setts som ett roligt inslag som t.ex. spela spel och programmera drönare? Just detta påstående påtalas också vid intervjun med IT-pedagogerna i kommun 1.

**Person 1 & 2** *“Det är ju den utmaningen som både vi och de själva identifierar - att där är ju ett glapp i styrdokumentet när det inte finns med i kunskapskraven - vilka förmågor är det vi bedömer mot? Det är inte alltid klart för dem. Det har vi fått lägga en del tid på att diskutera - så det inte bara blir lekandet och körandet med robotar. Så blir det ju svårt ibland, man tänker jag är mattelärare och är jättebra på matte men jag kan inte kod och ändå ska jag nu plötsligt kunna använda kod för att kunna omsätta problem till handling”.*

Vi har även tittat på om huruvida antal år som lärarna undervisat påverkar deras säkerhet när det gäller att undervisa i programmering. De lärare som arbetat i 15–17 år känner sig mest säkra på att undervisa (5/8) i programmering i teknik. Samtidigt är det så att 6/8 av dessa lärare angett att de tidigare har undervisat i programmering, vilket kan förklara varför så stor andel känner sig säkra i undervisningen. Enkäten visade även att en del har undervisat i programmering utan att känna sig säkra i ämnet, vilket skulle kunna bero på att lärarna saknar utbildning i programmering. Av de som undervisat i 9–11 år är det endast 1/9 som känner sig säkra på att undervisa i programmering (3/9 har gått fortbildning). De två andra lärarna som har gått en fortbildning uppger ändå att de känner sig osäkra. Detta skulle kunna tyda på att fortbildningarna inte håller tillräckligt hög nivå för att lärarna ska känna sig trygga i att undervisa i programmering.

Då vi undersökt om det finns några klara samband mellan hur länge lärarna undervisat och hur säkra lärarna känner sig i att undervisa, har vi sett att de som arbetat i 3–5 år och 15–17 år känner sig säkrast. En möjlig förklaring till att gruppen 3-5 år visar stor säkerhet i att undervisa i programmering skulle kunna bero på att de hunnit få några års erfarenhet av undervisning och att medelåldern troligtvis inte är särskilt hög i gruppen. Dessa lärare är troligtvis uppväxta i det digitala samhället och har möjligen en större inblick i programmering, som till exempel erfarenheter av kodspråk vid design av bloggar etc. som vi upplever blev väldigt populärt vid början av 2000-talet. En förklaring till att de lärare som undervisat i 15–17 år känner sig säkra är att de förmodligen är rutinerade efter många år i yrket, och troligtvis vana vid både att anpassa sig efter olika sorters förändringar samt att använda dator i läraryrket.

## 7.2 Vilka uppfattningar har lärarna av tillägget gällande programmering i den reviderade kursplanen i teknik?

Lärarna fick frågan om hur de tolkar det nya tillägget gällande programmering i det centrala innehållet i teknik, vad de trodde menades med “andra föremål” och ifall det inkluderade digitala föremål. Deras svar delades in i olika teman och svarsfördelningen uttrycks i tabell 3.

<b>Tolkning</b>	<b>Antal</b>
Digitala och fysiska föremål	26
Digitala föremål räknas in	13
Fysiska föremål, robotar11	
Har inte funderat över det	2
Övriga kommentarer	3

Tabell 3: *Tolkning av tillägget i teknik: sorterat i teman*

Strax över en femtedel av lärarna svarar att de räknar in digitala föremål bland “andra föremål” och skriver i sitt svar inget om fysiska föremål. Majoriteten tolkar det som att det innefattar styrning av både digitala och fysiska föremål, men en femtedel av lärarna nämner inte digitala föremål överhuvudtaget utan de verkar enbart tolka det som att det gäller fysiska föremål, och då främst robotar. Enkätfrågans formulering kan möjligtvis påverkat hur lärarna svarat eftersom en relativt hög andel lärare inom temat “Digitala föremål räknas in” endast har svarat att det räknas in och inte skriver hur de uppfattar att “andra föremål” kan tolkas.

**Enkät svar** *“Jag tänker att digitala föremål räknas in. Det är inte enkelt för skolor att köpa in den praktiska tekniken för att genomföra allt praktiskt.”*

**Enkät svar** *“Jag tolkar att det kan vara digitala saker men även analoga saker som man gör själv. Det kan vara en grund att börja på med analogt för att sedan koppla över till det digitala.”*

**Enkät svar** *“Jag räknar in små robotar som du kan exempelvis låna på av media om det inte finns på skolan.”*

Två lärare svarade kort att de inte hade funderat över frågan och valde att inte heller reflektera över frågan då de genomförde enkäten. Att dessa två lärare svarade så här skulle kunna bero på ett ointresse/okunskap om programmering eller att de ännu inte satt sig in i revideringen.

Övriga kommentarer innefattar kommentarer som inte gick att sortera in i våra huvudteman och som inte ansågs besvara frågan, såsom följande kommentar:

**Enkät svar** *“Programmera gör jag med mina elever inom ramen för teknik”.*

## 7.3 Hur ser tillgången till fortbildning i programmering ut enligt tekniklärarna?

I enkäten svarade 32 av 55 tillfrågade att de blivit erbjudna att gå någon form av fortbildning i programmering och 22 av 55 svarade att de redan har gått någon fortbildning. Nästan en tredjedel anger att de har gått en fortbildning som varit knuten till Skolverket, resterande anger att de har gått andra fortbildningar eller använt sig av sidor på nätet (se bilaga 5).

**Enkät svar** *“Några workshops som kommunen arrangerat, varit på studiebesök på Högskolan i Halmstad, andra fortbildningar som jag själv initierat.”*

**Enkät svar** *“Har deltagit i AV medias fortbildningskväll i programmering och lärde mig om "Code Monkey" (vilket jag också körde med mina elever som blev helt förälskade i det hela) och även eftermiddagsträffar vid två tillfällen med bl.a. introducering till VR.”*

Bland de lärare som undervisat i 15–17 år har hela 7 av 8 lärare blivit erbjudna att gå fortbildning, och 5 av 8 har gått någon form av fortbildning. Det är i denna grupp som störst andel lärare fått erbjudande om att gå fortbildning och även har gått fortbildning. Vi ställer oss frågan vad det kan bero på? Möjligtvis kan det vara så att de med sin höga yrkeserfarenhet fått förtroende att först gå fortbildning och sedan utbilda övriga lärare på skolan. I gruppen av lärare som är födda -57 eller tidigare var det 2 av 3 som blivit erbjudna fortbildning, men ingen utav dem som hade gått någon fortbildning. Personerna i denna grupp, som därmed är 61 år och äldre, börjar närma sig pensionsåldern och det kan med stor sannolikhet vara en bidragande faktor till att ingen ännu gått fortbildning. I gruppen som har undervisat 0–2 år är det 5/12 som erbjudits att gå fortbildning och 6/12 som har gått någon form av fortbildning. Detta scenario ser vi också i den grupp lärare som undervisat i 6–8 år, här är det dubbelt så många som gått fortbildning jämfört med hur många som blivit erbjudna. En möjlig förklaring till att fler än de som blivit erbjudna gått utbildning, kan vara att de själva funnit utbildningar de velat gå vilket i sin tur påvisar ett intresse och engagemang hos lärarna, eller möjligen en oron för vad som komma skall.

En annan siffra som sticker ut är i gruppen som undervisat i 3–5 år, här är det endast 1/6 som gått fortbildning, trots att hälften blivit erbjudna att gå någon fortbildning. Detta skulle kunna bero på att det finns ett stort eget intresse hos dessa personer när det gäller programmering och att de själva skaffar sig kunskap. Det skulle även kunna bero på att det

inte funnits tid att gå någon fortbildning ännu eller att den tänkta fortbildningens startdatum inte infallit. I vissa fall kan det säkert också vara att lärarnas önskemål inte kan uppfyllas för tillfället då fortbildning är en kostnadsfråga, vilket påtalades i intervjun med lärare A.

**Lärare A** *“LIN utbildningen var dyr, jag hittade den själv och fick därför möjlighet att gå. De andra lärarna på skolan har blivit rekommenderade att gå skolverkets webbkurs. Mycket handlar om pengar och att den kursen är gratis vilket gillas av rektorerna”.*

Vi undersökte även regionala skillnader i Skåne gällande fortbildning (se tabell 4). Mellersta Skåne och Nordvästra Skåne låg klart i framkant när det gäller erbjudande av fortbildning, i de båda regionerna hade över 3/4 av lärarna blivit erbjudna fortbildning. I Sydöstra Skåne, Nordöstra Skåne och Sydvästra Skåne var det knappt hälften som blivit erbjudna fortbildning. När vi däremot tittar på i vilken region flest lärare gått fortbildning ligger Sydvästra Skåne i topp, där drygt hälften av de som erbjudits fortbildning har gått någon slags fortbildning. I Nordöstra Skåne är det endast 1/5 av lärarna som har gått någon form av fortbildning gällande programmering.

	<b>Erbjuden fortbild.</b>	<b>Gått fortb.</b>	<b>Finns material</b>	<b>Planerade inköp</b>
Sydöstra Skåne	4/8	3/8	7/8	0/8
Nordöstra Skåne	2/5	1/5	3/5	1/5
Mellersta Skåne	6/7	3/7	5/7	0/7
Nordvästra Skåne	12/16	5/16	11/16	5/16
Sydvästra Skåne	9/19	10/19	16/19	5/19
<b>Totalt</b>	<b>33/55</b>	<b>22/55</b>	<b>42/55</b>	<b>11/55</b>

Tabell 4: Svarsfrekvensen på enkätfrågor gällande fortbildning uppdelade i kommunregioner.

## 7.4 Hur ser tillgången till resurser/material i programmering ut enligt tekniklärarna?

Tillgång till material och programvara är en förutsättning för att undervisningen i programmering ska framskrida väl. En glädjande siffra som framkom i våra enkätsvar var att 42 av 55 lärare ansåg att det på skolan fanns material/programvara att tillgå i



programmeringsundervisningen. Det som övervägande nämndes var ”iPads/datorer/appar” och då ofta Scratch, samt olika robotar, framförallt ”Blue bots/beebot”. En trolig förklaring till att just dessa material/programvaror nämndes skulle kunna vara att de är starkt förknippade med just programmering samt att vi har märkt att de ofta nämns i programmeringssammanhang.

I bilaga 6 ges en kort beskrivning av de mest omtalade hårdvaror och programvaror som de informanter som intervjuats eller besvarat vår enkät har nämnt.

Vi undersökte även om det fanns någon regional skillnad i Skåne då det gäller tillgång till material/programvara (se tabell 5). Kommunerna i Sydöstra Skåne och Sydvästra Skåne, verkar bäst rustade när det gäller material/programvara för att undervisa i programmering. I Sydvästra Skåne uppger även 1/4 att det på skolan finns planerade inköp gällande programmeringsundervisningen.

**Enkät svar** *“Vi ska försöka hitta pengar till MicroBit uppsättningar.”*

	<b>Finns material</b>	<b>Planerade inköp</b>
Sydöstra Skåne	7/8	0/8
Nordöstra Skåne	3/5	1/5
Mellersta Skåne	5/7	0/7
Nordvästra Skåne	11/16	5/16
Sydvästra Skåne	16/19	5/19
<b>Totalt</b>	<b>42/55</b>	<b>11/55</b>

Tabell 5: Svansfrekvensen på enkätfrågor gällande tillgång till/inköp av material uppdelade i kommunregioner.

För att programmeringsundervisningen ska fungera optimalt är det en förutsättning att det finns datorer/Ipads att tillgå. Många kommuner har 1:1, men det finns också kommuner där eleverna inte har personliga datorer/Ipads och det endast finns några datorer/Ipads att tillgå per klass.

**Person 1 & 2** *“Den likvärdiga skolan som vi eftersträvar är svår att uppnå i detta fallet. De som inte har de fysiska förutsättningarna sedan innan har en dubbel väg att gå. Att skolor bara en två datorer i en klass kan indikera att digitaliseringen inte är prioriterad”.*

**Person 4** *”Tyvärr tror jag att det ser mycket olika ut beroende på om kommunerna har haft möjlighet att köpa enheter.....”*

På skolorna bör det i möjligaste mån finnas möjlighet att ha tillgång till en hel klassuppsättning datorer/Ipads så att varje elev har möjlighet att arbeta och programmera med en egen dator/Ipad vilket även IT-pedagogerna i kommun 1 och lärare B framhäver.

**Person 1 & 2** *”Behöver inte vara personliga men du behöver definitivt ha den tillgång - och just när det gäller programmering. Visst vi kan börja med dansprogrammering men nånstans måste vi bygga upp progressionen till något annat”.*

**Lärare B** *”Grunderna är att det åtminstone ska vara en dator per två, det funkar men 1:1 är det ultimata.”*

Något som framkom i flertalet intervjuer var att inköpen av material/programvara är dyrt. Utlåningsdepåer skulle kunna gynna kommunerna ekonomiskt och ge skolorna möjlighet att inte endast använda sig av ett material i undervisningen.

**Lärare A** *”Vår tanke är att låna Lego Wedo av en annan skola i orten. Det är dyrt så skolorna får samsas, ungefär som att det finns ett bibliotek med saker att låna. Det är ju ofta så att det används 4–5 veckor intensivt sedan kan det stå länge utan att det används”.*

Små kommuner kan ha svårare att bygga upp en utlåningsdepå eller att byta material mellan skolorna menar IT-pedagog 3, och säger att de kanske inte har ett centralt samarbete utan att skolorna verkar enskilt. Tillgången till material/programvara i programmeringsundervisningen handlar mycket om kommunernas ekonomi, material som t.ex. Lego är kostsamma, men samtidigt finns det många programvaror som är gratis.

**Person 1 & 2** *”Vi ser många kommuner där politikerna pekar med hela handen men inte skapar förutsättningar men de kanske inte ger ekonomiska medel eller följer upp inte det. Ekonomin är en stor del i det hela - det måste göras en kontinuerlig satsning varje år för att kunna upprätthålla det.”*

**Lärare A** *”En klassuppsättning lego wedo pratar vi om kanske upp mot 40 000 kr. och för skolor som har en ansträngd budget går det inte att köpa in.”*

## 8. Slutsats och Diskussion

I detta kapitel kommer vi att diskutera och problematisera resultatet av studien i förhållande till tidigare forskning och studier som vi berört i kapitel 5. Studiens syfte var att undersöka vilka uppfattningar tekniklärare har kring implementeringen av programmering i teknik samt hur de upplever att tillgången till fortbildning och resurser ser ut. Vår första frågeställning diskuteras i avsnitt 8.1, den andra frågeställningen diskuteras i 8.2 och vår tredje frågeställning för vi en diskussion kring i avsnitt 8.3. Avslutningsvis förs en metoddiskussion, samt förslag ges på vidare forskning inom området.

### 8.1 Vilka uppfattningar har tekniklärare av implementeringen av programmering i teknik i årskurs 4-6?

Majoriteten av lärarna samt de IT-pedagoger/utvecklingsledare vi intervjuade är överens om att det är nödvändigt att undervisa i programmering redan i grundskolan och ser positivt på att programmering har införts i läroplanen. Precis som i tidigare forskning (Runeson et al., 2015; Voogt et al., 2015) och i Statens offentliga utredning (SOU 2016:89) resoneras det som så att eleverna behöver undervisas i programmering för att de ska kunna hänga med i och förstå samhälls- och teknikutvecklingen. IT-pedagogerna i kommun 1 säger att det handlar om att förstå omvärlden och inte om att få alla elever att bli programmerare eller kodare.

Inte alla förstår anledningarna bakom implementeringen av programmering i grundskolan. En vanlig missuppfattning verkar vara att tanken är att fler ska bli programmerare (Wallin, 2017, december), och ämnet har fått uppmärksamhet i samhället och i media. Under intervjun med lärare B, men även i något av enkätsvaren, kunde vi märka att en del lärare inte är övertygade med syftet av införandet. Vi undrar om den förstorade debatten kan bidra till att tekniklärare känner sig osäkra på hur de ska undervisa och upplever programmering som något läskigt? Då Finland införde programmering i läroplanen uttryckte landets utbildningsminister att allmänhetens oförstående till implementeringen var en utmaning (Wallin, 2017, december). Enligt IT-pedagogerna har programmering fått stor uppmärksamhet här i Sverige med tanke på att det endast utgör en liten del av digitaliseringen. I en debattartikel skriver Czajkowski (2017, 7 april) att det är fördelaktigt att

programmering införts eftersom Sverige snart kommer att ha brist på programmerare. Det kan tolkas som Czajkowskis ståndpunkt är att programmering i skolan ska leda till att fler väljer att bli programmerare, vilket enligt Skolverket (2016) inte är det främsta målet. von Zweigbergk (2017, 10 april) är i sin debattartikel däremot tydlig med att han inte ser det som att alla elever ska bli programmera utan han ser programmeringen som en allmänbildning. Sammanfattningsvis kan vi säga att majoriteten av de tekniklärare och IT-pedagoger/utvecklingsledare vi varit i kontakt med är positiva till införandet av programmering i läroplanen och tycker att det är en del av samhällsutvecklingen samt att programmeringsundervisning ger eleverna en bra grund att stå på.

## 8.2 Vilka uppfattningar har lärarna av tillägget gällande programmering i den reviderade kursplanen i teknik?

I läroplanens centrala innehåll för teknik går det att utläsa: *Att styra egna konstruktioner och eller andra föremål med programmering* (Skolverket, 2017a) . Hur tolkas detta tillägg av lärare och då framförallt gällande ”andra föremål”, innefattar andra föremål även digitala föremål?

Majoriteten av lärarna i denna studie tolkade det som att andra föremål innefattar både digitala och fysiska föremål. Några lärare förklarade sin tolkning med att det gällde alla programmerbara föremål, oavsett om det är digitala eller analoga. En stor del av de tillfrågade lärarna tolkar det som enbart fysiska objekt och då framförallt robotar, vilket vi hade en föräning om innan studien påbörjades.

Hermansson och Wickholms (2017) resultat av sin studie visade att analog programmering i form av olika lekar är en bra inkörsport för eleverna innan de börjar programmera digitalt med datorer. Vid analog programmering får eleverna en tydlig inblick i olika begrepp på ett konkret sätt. Att så få lärare nämner analog programmering i enkätsvaren, var lite oväntat. En lärare påpekar vikten av att inte glömma bort att programmering även kan handla om t.ex. dansprogrammering. Denna kommentaren utmärker sig, då mycket av svaren handlar om olika slags robotar. Många lärare verkar ha varit i kontakt med eller är nyfikna på robotar, vilket är positivt i sig. Något som däremot är viktigt att ha i åtanke är att kunna knyta an till något ämne i programmeringsundervisningen, så att det inte blir programmering utan någon

koppling till skolämnet. Vi anser att det är viktigt att programmering inte bara blir att använda sig av robotar och lek för nöjets skull utan att undervisningen även tar upp programmeringens grunder och exempel på hur programmering syns i samhället. För att eleverna ska ha möjlighet att få en djupare förståelse i programmering med hjälp av robotar är det viktigt att integrera både teori och praktik i programmeringsundervisningen. Kilbrinks (2008) studie visade att eleverna annars har svårt att se kopplingen mellan teori och praktik. Kopplingen mellan förståelse och görande måste förstärkas så att det blir tydligt för eleverna hur allt hänger ihop.

Vid intervjun med en lärare ifrågasattes tillägget i läroplanen gällande programmering. Läraren frågade sig varför man måste lära sig att styra andra saker och vad nyttan är med att kunna det? Vidare tror läraren att många elever kommer tycka att det är roligt med programmering, men tror inte att eleverna kommer att ha nytta av det i sitt framtida yrkesliv, mer än om de involveras i dataspelsbranschen. Detta påstående kan tyckas vara lite oväntat i dagens digitaliserade samhälle!

Vilka uppfattningar lärarna har av tillägget *Att styra egna konstruktioner och eller andra föremål med programmering* (Skolverket, 2017a) varierar, men det går att utläsa att majoriteten av tekniklärarna och de IT-pedagoger/utvecklingsledare som deltagit i vår studie tolkar att *andra föremål* innefattar både fysiska och digitala objekt.

### 8.3 Hur ser tillgången till resurser och fortbildning i programmering ut enligt tekniklärarna?

Implementeringen av programmering i teknik medför att många lärare behöver fortbildas för att kunna undervisa. Kjällander et al. (2016) påvisade i sin studie, då de granskade implementeringen av programmering i England och Finland, att den absolut största utmaningen var fortbildning av lärare. I Finland finansierade staten olika utvecklingsprojekt som ett led i att fortbilda lärare. I Sverige ser Skolverkets (2017c) gratis webbkurs i programmering ut att ha blivit populär, den nämns av flera lärare i enkäten men också i våra intervjuer. IT-pedagogen i kommun 2 berättade att de har fullt upp med fortbildningsinsatser för att kunna nå så många som möjligt samt att hjälpa lärarna med hur de ska arbeta. Samtidigt önskade IT-pedagogen att övergångsperioden varit längre så fortbildning av all

personal hunnits med innan programmeringsundervisningen blir obligatorisk, samt att Skolverkets material blivit klart tidigare. IT-pedagogen påtalar även att det hos lärarna de mött finns en stor oro och osäkerhet framförallt gällande de resurser som behövs.

Sentance och Csizmadia (2017) nämner i sin studie två olika sätt på vilka lärarna själva såg utmaningar i programmeringsundervisningen: intrinsiska utmaningarna såsom lärarnas tillvägagångssätt vid undervisning i ämnet eller ämnesdidaktiska kunskaper, och extrinsiska utmaningarna såsom brist på resurser. Vi tror att de extrinsiska utmaningar som lärarna själva har svårt att råda över kan ge upphov till större frustration hos lärarna än de intrinsiska. I vår enkätstudie visar det sig att majoriteten ännu inte hunnit med att gå fortbildning och inte heller känner sig säkra på att undervisa i programmering. En möjlig orsak till detta kan vara att övergångsperioden varit för kort och därför har inte personal hunnit fortbildas. Största skälen till osäkerheten att undervisa anges vara bristen på fortbildning och oförmågan att veta på vilket sätt undervisningen i programmering ska ske. Betydelsen av lärarens kunskaper och dess centrala roll menar Rolandsson (2015) har en avgörande roll när det gäller hur programmering utvecklas i klassrummet och hur undervisningen blir. Sentance och Csizmadia (2017) påtalar också vikten av att lärarna behöver utveckla en lämplig pedagogik samt träna på programmering och därigenom bygga upp sitt självförtroende när det gäller att undervisa. Kontinuerlig fortbildning i programmering tror vi hade underlättat för många lärare, både när det gäller att bygga upp sitt självförtroende och att utvecklas inom programmeringen. I studien av hur Maker-rörelsen kan integreras med lärarutbildning fann Kjällander et al. (2018) fyra olika sätt att göra det på: 1) lärarstudenterna kan gå till valfritt makerspace; 2) lärarstudenter, lärare och rektorer kan bjudas in till makerspaces för diskussioner och praktik; 3) mobila makerspaces kan skapas på övningsskolor; och 4) makerspaces kan byggas upp på universiteten. Vi menar att maker-rörelsen kan vara en väg att gå för lärarstudenter men även för lärare och elever som vill utveckla sina programmeringskunskaper och inspireras av samt få stöd av andra likasinnade. Genom att ta tillvara på de makerspaces som redan finns i närheten eller att bygga upp på universitet/skolor kan fler lättare komma i kontakt med programmering och utvidga sina kunskaper. Precis som makerspaces finns det många andra olika sorters initiativ och organisationer (några exempel återges i bilaga 5) som kan vara till god hjälp.

Vår studie visar att fortbildning av personal ofta är en kostnadsfråga. Att en lärare skickas iväg på utbildning och sedan får utbilda resten av kollegorna har vi sett flera exempel på. Skolverkets webbkurs är populär bland rektorer då den är gratis och många får rekommendationer om att gå just denna. Flertalet lärare upplever dock denna webbkurs som att den ger en bra kontext kring programmering men utan att ge tillräckligt med praktiska exempel på hur undervisning i programmering kan se ut. Det verkar därmed bland lärarna i vår studie finnas en önskan om ett komplement till webbkursen som går djupare in på bedömning och ger praktiska exempel.

Att det finns tillgång till material och programvara anser vi är en förutsättning för att undervisningen i programmering ska framskrida väl. Över 3/4 av lärarna i enkätundersökningen svarade att det fanns tillgång till material. IT-pedagogerna i kommun 1 påtalar att i deras kommun har eleverna 1:1 och de anser, precis som vi gör, att tillgången på datorer/Ipads måste finnas vid undervisning i programmering. Analog programmering kan man använda sig av i början men sedan måste det ske en progression inom programmeringen till mer digital teknik, vilket då påvisar att det ultimata är 1:1. De material och hårdvaror som lärarna i vår enkät nämnde var i första hand Ipads/datorer/appar och olika robotar. Robotar är populära men inköpen av dessa är kostsamma. Att använda sig av utlåningsdepåer skulle kunna gynna kommunerna ekonomiskt, men här tror vi att mindre kommuner kan ha en svårighet då fåtalet skolor kan medföra att det finns en begränsad utlåningsdepå i kommunen. IT-pedagogerna i kommun 1 diskuterar också om vikten av att inte göra robotprogrammeringen till en rolig happening och de tror att detta går att förebygga genom att ofta ha tillgång till robotarna så att de blir ett naturligt inslag i undervisningen. Då det gäller tillgång till resurser och tillgångar verkar flertalet skolor ha tillräckligt med resurser, så när det gäller resurser verkar Skånes skolor ha en bra grund att börja med. Den absolut största utmaningen handlar om att fortbilda lärare, vilket även var Finlands största utmaning vid deras implementering av programmering i skolan (Kjällander et al., 2018). Då Finland ligger ett par år före Sverige när det gäller implementeringen av programmering i skolan, är det möjligt att Sverige i framtiden kan dra nytta av hur implementeringen fortskrider i Finland.

## 8.4 Metoddiskussion

Vårt val av metod gjorde att studiens frågeställningar var möjliga att besvara, men en nackdel vi såg med vår enkätundersökning var att det saknades möjlighet att ställa följdfrågor eller att be om förtydligande. I vissa fall var det svårt för oss att tolka respondenternas svar och vi hade då behövt be om en förklaring eller ett förtydligande för att kunna förstå vad respondenten menade. I andra fall var svaren så kortfattade att vi hade önskat att vi hade kunnat ställa följdfrågor för att få ett mer uttömmande svar.

Om vi enbart hade valt att genomföra intervjuer hade vi haft svårare för att få den geografiska spridning som vi eftersökte, eftersom det inte hade funnits utrymme till flera utspridda intervjuer. Intervjuer är tidskrävande och då vi har begränsat med tid hade detta blivit problematiskt, ett alternativ hade då varit att göra intervjuer via mail. Studiens syfte hade troligtvis kunnat uppnås även med denna metod, men vi tror att resultatet hade blivit än mindre generaliserbart eftersom vi då troligtvis inte hade haft möjlighet att höra lika många lärares uppfattningar. Däremot kunde intervjuer gett oss ett mer uttömmande svar gällande till exempel lärarnas erfarenheter av programmering och fortbildning inom detta.

Det är möjligt att vår studie hade nått en högre kvalitet och varit mer precis ifall vi valt att fokusera på antingen uppfattningar av implementeringen av programmering i teknik eller på den upplevda tillgången till fortbildning och material/programvara, samt om vi valt enbart enkätundersökning eller intervjuer som metod. Vår studie blev mer omfattande och tidskrävande än vi hade väntat oss och det har varit en utmaning att lyckas sammanställa all empiri.

Om vi hade haft mer tid till vårt förfogande och en möjlighet att genomföra en större studie hade det varit önskvärt att intervjua IT-pedagoger/utvecklingsledare i alla kommuner som deltog i vår studie. Då hade vi fått en större variation och hade även haft möjlighet att jämföra tekniklärarnas enkätsvar gentemot hur utvecklingsledarna tänkt sig att fortbildning och inköp ska se ut i respektive kommun.



## 8.5 Avslutande ord

Som Kjällander et al. (2016) påpekar är forskning kring programmering i den svenska grundskolan limiterad eftersom programmering inte tidigare har undervisats i de lägre åldrarna. Genom att belysa lärares uppfattningar av implementeringen av programmering i teknik och deras tolkningar av tillägget i kursplanen kan vår studie bidra till forskningen kring programmering i den svenska grundskolan. Studien bidrar även till att skapa en bild av hur förberedelserna sett ut bland lärare och skolor, vilka robotar och andra hårdvaror som används samt hur lärarna fördjupar sin kompetens i programmering, och kan därmed ligga till grund för framtida utvecklingsarbete.

Vi har i vår kommande yrkesroll fått nytta av studien då vi har fått en ökad kännedom om hur förberedelserna inför implementeringen sett ut, och vi har även blivit medvetna om vilka robotar och andra programvaror som används i skolor samt fått tips på olika resurser som kan vara behjälpliga om vi vill fördjupa våra kunskaper i programmering.

Utifrån empirin kan vi konstatera att majoriteten av lärarna är positiva till att programmering har införts i grundskolan, och en del har på eget initiativ undervisat i programmering trots att det inte tidigare fanns med i Lgr 11 (Skolverket, 2017a). Det är först höstterminen 2018 som det blir obligatoriskt att följa den reviderade läroplanen och många lärare samt skolor är mitt uppe i förberedelsearbetet och kanske ännu inte hunnit sätta sig in i de nya förändringarna. Det hade varit intressant att i en framtida forskningsstudie genomföra en liknande enkätundersökning för att se om tekniklärarnas tankar och uppfattningar kring programmering och programmering i teknikundervisningen förändras då de arbetat med det en längre tid. Kommer majoriteten av lärarna fortfarande vara positivt inställda till att undervisa i programmering och ger den undervisningen någon effekt för eleverna?

Det hade även varit intressant att göra en studie utifrån elevernas perspektiv, att undersöka vilka uppfattningar de har om programmering och vad elever tror det finns för syfte med att programmera i skolan.

# Referenser

- Alvehus, Johan. (2013). *Skriva uppsats med kvalitativ metod*. Stockholm: Liber.
- Balanskat, Anja & Engelhardt, Katja. (2015). *Computing our future: Computer programming and coding—priorities, school curricula and initiatives across Europe*. Bryssel: European Schoolnet.
- Bocconi, Stefania., Chiocciariello, Augusto., & Earp, Jeffrey. (2018). The Nordic approach to introducing Computational Thinking and programming in compulsory education. Report prepared for the Nordic@BETT2018 Steering Group. <https://doi.org/10.17471/54007>
- Bourque, Linda B. & Fielder, Eve P. (1995). *How to Conduct Self-Administered and Mail Surveys*. SAGE Publications, Inc. Thousand Oaks, USA.
- Brinkkjær, Ulf., & Høyen, Marianne. (2013). *Vetenskapsteori för lärarstudenter*. Lund: Studentlitteratur.
- Claesson, Silwa. (2007). *Spår av teorier i praktiken: några skolexempel*. (2., [utökade] uppl.) Lund: Studentlitteratur.
- Czajkowski, Jan. (2017, 7 april). Programmeringskunskaper är inte en fråga om liv och död. *DN* *Åsikt*. Tillgänglig: <https://asikt.dn.se/asikt/debatt/programmeringskunskaper-ar-inte-en-fraga-om-liv-och-dod/>
- Denscombe, Martyn. (2009). *Forskningshandboken: för småskaliga forskningsprojekt inom samhällsvetenskaperna*. (2. uppl.) Lund: Studentlitteratur.
- Ejlertsson, Göran. (2005). *Enkäten i praktiken: en handbok i enkätmetodik*. (2. [omarb.] uppl.) Lund: Studentlitteratur.
- Ejvegård, Rolf. (1996). *Vetenskaplig metod*. (2., [rev.] uppl.) Lund: Studentlitteratur.
- Heintz, Fredrik., Mannila, Linda., Nygårds, Karin., Parnes, Peter., & Regnell, Björn. (2015). *Computing at School in Sweden - Experiences from Introducing Computer Science within*

- Existing Subjects. In A. Brodnik, & J. Vahrenhold (Eds.), *Informatics in Schools. Curricula, Competences, and Competitions /Lecture Notes in Computer Science and General Issues* (Vol. 9378, pp. 118-130). Springer. Doi: 10.1007/978-3-319-25396-1\_11
- Hermansson, Alexander. & Wickholm, Jenny. (2017). Rutnät, robotar och koder. Om programmering i skolan. *Skolporten*, 5. Tillgänglig: <https://www.skolporten.se/app/uploads/2017/03/undervisning-larande-5-2017.pdf>
- Kilbrink, Nina. (2008). Robotar i skolan – elevers uppfattningar av lärandeobjekt och problemlösningstrategier. (Avhandling Karlstad studier I pedagogisk) Tillgänglig: <http://www.diva-portal.org/smash/get/diva2:5392/FULLTEXT01.pdf>
- Kjällander, Susanne., Åkerfeldt, Anna., Mannila, Linda., & Parnes, Peter. (2018). Makerspaces Across Settings: Didactic Design for Programming in Formal and Informal Teacher Education in the Nordic Countries, *Journal of Digital Learning in Teacher Education*, 34:1, 18-30, DOI: 10.1080/21532974.2017.1387831
- Kjällander, Susanne., Åkerfeldt, Anna., & Petersen, Petra. (2016). *Översikt avseende forskning och erfarenheter kring programmering i förskola och grundskola*. Tillgänglig: [http://omvarld.blogg.skolverket.se/wp-content/uploads/sites/2/2016/06/oversikt\\_programmering\\_i\\_skolan.pdf](http://omvarld.blogg.skolverket.se/wp-content/uploads/sites/2/2016/06/oversikt_programmering_i_skolan.pdf)
- Kodboken. (u.å.). *Vad är programmering?* Tillgänglig: <https://www.kodboken.se/start/fattakoden/programmering-och-kod/vad-ar-programmering>
- Kylén, Jan-Axel. (1994). *Fråga rätt – vid enkäter, intervjuer, observationer och läsning*. Kylén Förlag AB: Stockholm.
- Kylén, Jan-Axel. (2004). *Att få svar: intervju, enkät, observation*. (1. uppl.) Stockholm: Bonnier utbildning.
- Lindström, Karin. (2016, 18 mars). Forskare: Programmering i skolan är ett sätt att få eleverna att greppa världen. *Computer Sweden*. Tillgänglig: <https://computersweden.idg.se/2.2683/1.653443/programmering-skolan>

- Lundgren, Ulf P. (2015). Läroplansteori och didaktik - framväxten av två centrala områden. I Lundgren, Ulf P., Säljö, Roger., Liberg, Caroline. (Red.), *Lärande, skola, bildning: Grundbok för lärare*. (3 utg., ss. 139-249). Stockholm: Natur & Kultur.
- Nygårds, Karin. (2015). *Koden till digital kompetens*. Stockholm: Natur & Kultur.
- Passey, Don. (2017). Computer science (CS) in the compulsory education curriculum: Implications for future research. *Education and Information Technologies*, 22(2), 421-443. doi: 10.1007/s10639-016-9475-z
- Regeringen. (2017). *Stärkt digital kompetens i skolans styrdokument*. Tillgänglig: <http://www.regeringen.se/493c41/contentassets/acd9a3987a8e4619bd6ed95c26ada236/informationsmaterial-starkt-digital-kompetens-i-skolans-styrdokument.pdf>
- Rolandsson, Lennart. (2015). *Programmed or Not -A study about programming teachers' beliefs and intentions in relation to curriculum*. KTH, School of Education and Communication in Engineering Science. Tillgänglig: <https://www.diva-portal.org/smash/get/diva2:791197/FULLTEXT02.pdf>
- Runeson, Per., Andersson, Stefan., Heintz, Fredrik., Mannila, Linda., & Rolandsson, Lennart. (2015). *Hur skall det svenska utbildningssystemet möta framtidens utmaningar? Helhetsyn på svensk utbildning om digitalisering och programmering*. Tillgänglig: <http://swedsoft.se/wpcontent/uploads/sites/7/2015/05/2015-01-Utbildning-whitepaper.pdf>
- Sentance, Sue., & Csizmadia, Andrew. (2017). Computing in the curriculum: Challenges and strategies from a teacher's perspective. *Education and Information Technologies*, 22(2), 469-495. doi: 10.1007/s10639-016-9482-0
- Skolverket. (2016). *Uppdrag om nationella it-strategier i skolväsendet*. Tillgänglig: <http://www.skolverket.se/publikationer?id=3668>
- Skolverket. (2017a). *Läroplan för grundskolan, förskoleklassen och fritidshemmet 2011*. ([Ny, rev. utg.]). Hämtad 2018-02-01 från <https://www.skolverket.se/publikationer?id=3813>

Skolverket. (2017b). *Få syn på digitaliseringen på grundskolenivå*. Tillgänglig: [https://www.skolverket.se/om-skolverket/publikationer/visa-enskild-publikation?\\_xurl\\_=http%3A%2F%2Fwww5.skolverket.se%2Fwtpub%2Fws%2Fskolbok%2Fwpubext%2Ftryck-sak%2FBlob%2Fpdf3783.pdf%3Fk%3D3783](https://www.skolverket.se/om-skolverket/publikationer/visa-enskild-publikation?_xurl_=http%3A%2F%2Fwww5.skolverket.se%2Fwtpub%2Fws%2Fskolbok%2Fwpubext%2Ftryck-sak%2FBlob%2Fpdf3783.pdf%3Fk%3D3783)

Skolverket. (2017c). *Om programmering - webbkurs*. Tillgänglig: <https://www.skolverket.se/kompetens-och-fortbildning/lorare/om-programmering-webbkurs-1.263574>

Skolverket. (2017d). *Tydligare om digital kompetens i läroplaner, kursplaner och ämnesplaner*. Tillgänglig: <https://www.skolverket.se/skolutveckling/resurser-forlarande/itiskolan/styrdokument>

SOU 2016:89. *För digitalisering i tiden*. Stockholm: Elanders Sverige AB.

Skolöverstyrelsen. Sverige. (1982). *Läroplan för grundskolan. Kommentarmaterial. Att räkna : en grundläggande färdighet*. Stockholm: LiberLäromedel/Utbildningsförlaget.

Thoresson. Anders. (2016) *Barnhack – kom igång med programmering!* Version 2.0. Tillgänglig: <https://www.iis.se/fakta/barnhack/>

Utbildningsdepartementet. (2015). Uppdrag att föreslå nationella it-strategier för skolväsendet. Regeringsbeslut U2015/04666/S. Stockholm: Utbildningsdepartementet, 2015  
<http://docplayer.se/4215291-Uppdrag-att-foresla-nationella-it-strategier-for-skolvasendet.html>

Vetenskapsrådet. (2002). *Forskningsetiska principer inom humanistisk-samhällsvetenskaplig forskning*. Hämtad 2018-01-02, från <http://www.codex.vr.se/texts/HSFR.pdf>.

Vetenskapsrådet. (2017). *God forskningsed*. Hämtad 2018-01-02, från <https://publikationer.vr.se/produkt/god-forskningssed/>

von Zweigbergk, Per. (2017, 10 april). Programmering är lika viktigt som matematik och naturvetenskap. *DN* *Åsikt*. Tillgänglig:

<https://asikt.dn.se/asikt/debatt/programmeringskunskaper-ar-inte-en-fraga-om-liv-och-dod/programmering-ar-lika-viktigt-som-matematik-och-naturvetenskap/>

Voogt, Joke., Fisser, Petra., Good, Jon., Mishra, Punya., & Yadav, Aman. (2015). *Computational thinking in compulsory education: Towards an agenda for research and practice*. *Education and Information Technologies* 20, Issue 4, pp 715–728. Hämtad 2018-01-23, från <https://link.springer.com/article/10.1007/s10639-015-9412-6>

Wallin, Fredrik. (2017, december). Det här kan vi lära av Finland. *Skolvärlden*. Hämtad 2018-01-30, från <http://skolvarlden.se/artiklar/det-har-kan-vi-lara-av-finland>

Wing, Jeannette M. (2006). Computational thinking. *Communications of the ACM- self managed systems*. 49(3). pp. 33-35.

# Bilagor

## Bilaga 1. Följebrev till enkät

Hej,

Vi är två lärarstudenter, vid namn Johanna och Katja, som läser till grundlärare i årskurs 4-6 på Malmö Universitet. Inom vårt fördjupningsämne NO/teknik ska vi nu skriva ett examensarbete med fokus på teknik och programmering.

Vi ber Dig att fylla i denna enkät som är en del av vårt arbete. Enkäten består av 10 frågor och tar ca 10 minuter att fylla i. Enkäten vänder sig endast till behöriga lärare i teknik i årskurs 4-6.

Som du vet har läroplanen för grundskolan, förskoleklassen och fritidshemmet reviderats. Den nya reviderade läroplanen kommer från och med läsåret 2018/2019 bli obligatorisk att följa.

I det centrala innehållet för kursplanen i teknik i årskurs 4–6 under arbetsätt har följande lagts till: “Att styra egna konstruktioner eller andra föremål med programmering” (Skolverket 2017, s. 285).

Syftet med vår studie är att få en samlad bild av hur lärares syn på implementeringen av programmering i teknik i årskurs 4–6 ser ut i skolor i Skåne, samt hur det nya tillägget i kursplanen kan tolkas.

Om Du accepterar att delta i vår studie ber vi att Du besvarar enkäten inom en vecka, senast den 30:e januari.

Ditt deltagande är helt frivilligt, men är mycket uppskattat för att bidra till en samlad bild av teknicklärares syn på implementeringen av programmering i teknik. Du kommer i examensarbetet att vara anonym och dina svar kommer inte att kunna härledas i studiens resultat.

Om Du vill veta mer om studien är du välkommen att kontakta oss eller vår handledare. Du kan även kontakta oss för att erhålla den färdiga studien.

Johanna Nyhlén, xxxxxxxxx@xxxxxx

Katja Lindblad, xxxxxxxxx@xxxxxx

Handledare: Eva Davidsson, xxxxxxxxx@xxxxxx

## Bilaga 2. Mail till skolorna

Hej,

Johanna och Katja heter vi och vi skriver just nu ett examensarbete i vårt fördjupningsämne NO och teknik vid Malmö Universitet.

Syftet med vår studie är att få en samlad bild av hur tekniklärare på skolor runt om i Skåne ser på införandet av programmering i åk 4-6, samt att få en bild av hur det nya tillägget i kursplanen kan tolkas.

Vi ber er om en mejladress till tekniklärare i årskurs 4-6 på er skola så att vi kan maila en kort enkät, och hoppas att ni kan hjälpa oss med detta.

Enkäten som vänder sig till behöriga lärare i NO/teknik för årskurs 4-6 består av 10 frågor och tar max 10 minuter att fylla i.

Det är naturligtvis frivilligt att delta i vår enkätundersökning - vi blir otroligt tacksamma för all medverkan.

Vänliga hälsningar

Johanna Nyhlén och Katja Lindblad



# Bilaga 3. Enkätfrågor

## 1. Inom vilket åldersspann är du född?

- |                                 |  |
|---------------------------------|--|
| <input type="radio"/> 1997-1990 | <input type="radio"/> 1973-1966                |
| <input type="radio"/> 1989-1982 | <input type="radio"/> 1965-1958                |
| <input type="radio"/> 1981-1974 | <input type="radio"/> Född 1957 eller tidigare |

## 2. I hur många år har du undervisat i teknik i årskurs 4-6?

- |                                |                                     |
|--------------------------------|-------------------------------------|
| <input type="radio"/> 0-2 år   | <input type="radio"/> 15-17 år      |
| <input type="radio"/> 3-5 år   | <input type="radio"/> 18-20 år      |
| <input type="radio"/> 6-8 år   | <input type="radio"/> 21-23 år      |
| <input type="radio"/> 9-11 år  | <input type="radio"/> Fler än 24 år |
| <input type="radio"/> 12-14 år |                                     |

## \* 3. Hur tolkar du det nya tillägget i det centrala innehållet i teknik: Att styra egna konstruktioner och eller andra föremål med programmering?

Exempelvis, vad tror du menas med "andra föremål"? Räknas även digitala föremål in?

\* 4. Ser du positivt på införandet av programmering i teknik i årskurs 4-6?

- Ja  
 Nej

Utveckla ditt svar.

\* 5. Känner du dig säker i att undervisa utifrån det följande nya tillägget i teknik:  
Att styra egna konstruktioner och eller andra föremål med programmering?

- Ja  
 Nej

Utveckla ditt svar.

\* 6. Har du tidigare undervisat i programmering?

- Ja  
 Nej

Om JA: Beskriv dina erfarenheter kortfattat.

\* 7. Har du blivit erbjuden fortbildning i programmering?

- Ja
- Nej

Om JA: Beskriv kortfattat omfattning och typ av fortbildning.

\* 8. Har du gått någon fortbildning i programmering?

- Ja
- Nej

Om JA: Beskriv kortfattat omfattning och typ av fortbildning.

\* 9. Finns det på din skola material/programvara att tillgå för att undervisa i programmering?

- Ja  
 Nej

Om JA: Beskriv vilka material/programvaror.

\* 10. Finns det enligt din kännedom några planerade inköp av material/programvara på din skola?

- Ja  
 Nej

Om JA: Beskriv vilka material/programvaror.

## Bilaga 4. Intervjufrågor

1. I hur många år har du undervisat i NO/teknik i årskurs 4-6?
2. Hur ser du på införandet av programmering i kursplanen i teknik i årskurs 4-6?
3. Vilka erfarenheter har du av programmering? Både privat och i yrket.
4. Vad tänker du att “Att styra egna konstruktioner eller andra föremål med programmering” innebär för din undervisning?
5. Hur väl förberedda tror du att dagens skolor är avseende nödvändiga kompetenser och resurser som krävs för undervisning i programmering?
6. Hur hade du önskat att förberedelserna av lärare och skolor sett ut vid införandet av programmering i läroplanen?

# Bilaga 5. Fortbildning inom programmering

## **LIN-education**

LIN-education är en serviceorganisation som hjälper skolor, organisationer och företag att digitalisera lärandet. De arbetar för att göra lärandet mer tillgängligt och roligare och har utvecklat egna produkter för lärande (t.ex. appen Pluttra). LIN-education arrangerar årligen över 1000 utbildningsinsatser (LIN-education, u.å.)

## **Programmera Mera**

En programserie på tio avsnitt från Utbildningsradion (UR). Programserien riktar sig till barn som är nybörjare inom programmering. Att barnen ska lära sig skriva kod är inget som är i fokus när det gäller programserien, utan syftet är istället att barnen ska förstå tankesättet (Utbildningsradion, 2017).

## **Skolverkets Webbkurs**

Webbkursen, som beräknas ta runt 15 timmar att utföra, är i första hand inriktad mot lärare som undervisar i grundskolan, fritidshem och förskola. Här ges pedagoger konkreta exempel på hur man kan använda programmering som verktyg i undervisningen och här finns också material att tillgå. Målet med skolverkets webbkurs är att ge pedagoger tillfredsställande grundkunskaper för att kunna undervisa i programmering, eftersom programmering tillkommit i kursplanen (Skolverket, 2017c).

## **Webbstjärnan**

2008 startades Webbstjärnan av Internetstiftelsen i Sverige (IIS). Webbstjärnan går ut på att vara ett stöd för skolan och ge lärare och elever möjlighet att lära sig mer om internet, webb och webbpublicering. Webbstjärnan består även av en tävling, vilken går ut på att publicera ett skolarbete på internet. På Webbstjärnan finns det möjlighet för lärare att delta i kostnadsfria seminarier och workshops (Internetstiftelsen i Sverige IIS, 2008).

---

LIN-education. (u.å.). *Det är vi som är LIN-education*. Tillgänglig: <http://www.lineducation.se/start>  
Utbildningsradion. (2017). *Programmera mera*. Tillgänglig: <https://urskola.se/Produkter/196673-Programmera-mera/Visa-alla>  
Skolverket. (2017c). *Om programmering - webbkurs*. Tillgänglig: <https://www.skolverket.se/kompetens-och-fortbildning/larare/om-programmering-webbkurs-1.263574>  
Internetstiftelsen i Sverige, IIS. (2008). *Om Webbstjärnan*. Tillgänglig: <https://www.webbstjarnan.se/om-webbstjarnan/>

# Bilaga 6.

## Hårdvara/programvara för programmering

### Programmering online och av digitala objekt

#### **Code.org och Hour of code**

Code.org skapades av ingenjörer från Google, Facebook och Twitter, för att alla elever i alla skolor skulle få möjlighet att lära sig datavetenskap. Här finns många exempel på övningar, instruktionsvideor och olika nivå- och åldersanpassade kurser för elever. Hour of code är en av dessa och började som en timmes programmering för att avdramatisera "att koda" och har i dagsläget utvecklats till en mängd olika initiativ.Handledning och lektionsförslag finns tillgängliga på över 45 språk (Code, 2018).

#### **Java**

Java är ett säkert, snabbt och tillförlitligt programmeringsspråk och datorplattform. Flertalet webbplatser och applikationer måste ha Java installerat för att fungera. Java finns bl.a. i mobiltelefoner, internet och spelkonsoler (Java, u.å.).

#### **Kojo**

Kojo är en utvecklingsmiljö för programmeringsspråket Scala. På enkel nivå innebär det att styra en sköldpadda med hjälp av kommando, vilket gör att sköldpaddan ritar olika mönster. I Kojo görs riktig programmering med kod och Kojo appen är gratis (Kogics foundation, u.å.).

#### **Scratch**

Scratch kan beskrivas som ett grafiskt programmeringsverktyg. I Scratch använder man sig av ett blockbaserat språk, vilket går ut på att pusselbitsliknande block länkas samman. Dessa block har olika funktioner och man behöver inte koda själv. I Scratch går det att skapa t.ex. berättelser, spel och musik (Scratch, u.å.).

---

Code. (2018). *Hour of Code*. Tillgänglig: <https://code.org/>

Java (u.å.). *Vad är Java-teknik och varför behöver jag det?* Tillgänglig: [https://www.java.com/sv/download/faq/whatis\\_java.xml](https://www.java.com/sv/download/faq/whatis_java.xml)

Kogics foundation (u.å.). *Play, create, learn*. Tillgänglig: <http://www.kogics.net/kojo>

Scratch (u.å.). *Vad är Scratch?* Tillgänglig: <https://sites.google.com/site/scratchprogrammering/vad-er-scratch-1>

## Programmering av hårdvara och fysiska objekt

### **Arduino**

Arduino är en fysisk elektronikplattform, bestående av ett kretskort, som är baserad på lättanvänd hårdvara och mjukvara. För att programmera plattformen använder man sig av Arduinos programmeringsspråk (Arduino, 2018).

### **Blue-bot och Bee-bot**

Blue-bot och Bee-bot är färgglada små robotar, främst avsedda för små barn. De kodas genom att trycka på piltangenterna som sitter på ovansidan och dessa kan förflytta roboten framåt, bakåt, vänster och höger. Kommando kan även ges via dator eller surfplatta (Bee-bot, 2016).

### **Dash & Dot**

Dash & Dot är robotar som kan utföra många olika saker, vilka de utför efter kommando som ges via surfplattan och överförs till roboten via Bluetooth. Det finns olika svårighetsgrad, detta beror på vilken app du använder (Dash & Dot, u.å.).

### **Makey Makey**

Makey Makey är ett kretskort som, genom att koppla den till elektriskt ledande föremål, gör det möjligt att styra program på datorn via de elektriskt ledande föremålen. Exempelvis att kunna spela musik på ett riktigt äpple! (MakeyMakey, 2018).

### **Micro:Bit**

Micro:Bit är en "liten enkel dator" som kan göra olika saker genom att programmeras. Micro:biten har många olika funktioner såsom kompass, accelerometer, LED-display och två programmerbara knappar. Micro:biten programmeras, via en särskild hemsida, genom att den kopplas till en dator via en USB kabel (Micro:bit, u.å.).

### **LEGO**

Utifrån LEGOs klassiska små byggklossar har mängder av olika LEGO typer utvecklats, så som programmeringsbara robotar, t.ex. LEGO WeDo, LEGO Mindstorms. LEGO education är ett undervisningsmaterial med läroplansrelevanta uppgifter, och med hjälp av LEGO WeDo och LEGO Mindstorms, bygger eleverna programmeringsbara robotar och får dem sedan att utföra olika uppgifter (LEGO, 2017).

---

Arduino. (2018). *What is Arduino?* Tillgänglig: <https://www.arduino.cc/en/Guide/Introduction>  
Bee-Bot. (2016). *Meet Bee-Bot and Blue-Bot.* Tillgänglig: <https://www.bee-bot.us/>  
Dash & Dot. (u.å.). *Dash & Dot.* Tillgänglig: <https://hos.se/dash--dot>  
Makey Makey. (2018). *Makey Makey classic.* Tillgänglig: <https://makeymakey.com/>  
Micro:bit (u.å.). *Get creative, Get connected, Get coding.* Tillgänglig: <http://microbit.org/>  
LEGO. (2017). *Prova mer, upptäck mer.* Tillgänglig: <https://translate.google.se/translate?hl=sv&sl=en&u=https://education.lego.com/en-us&prev=search>