



Høgskulen på Vestlandet

Masteroppgave

MASIKT-OPG

Predefinert informasjon

Startdato:	07-09-2018 09:00	Termin:	2018 HØST1
Sluttdato:	14-09-2018 12:00	Vurderingsform:	Norsk 6-trinns skala (A-F)
Eksamensform:	MasIKT-opg: Masteroppgave	Studiepoeng:	45
SIS-kode:	203 MASIKT-OPG 1 OM-1 2018 HØST1		
Intern sensor:	(Anonymisert)		

Deltaker

Kandidatnr.: 108

Informasjon fra deltaker

Tittel *: Hvilke ulike praksiser eksisterer for faget programmering i ungdomsskolen, og hvilke innholdskomponenter legger lærerne vekt på i utformingen av faget?

Antall ord *: 32022

Tro- og loverklæring *: Ja

Jeg bekrefter at jeg har Ja
registrert oppgavetittelen
på norsk og engelsk i
StudentWeb og vet at
denne vil stå på
vitnemålet mitt *:

Jeg godkjenner avtalen om publisering av masteroppgaven min *

Ja

MASTEROPPGAVE

Hvilke ulike praksiser eksisterer for faget programmering i ungdomsskolen, og hvilke innholdskomponenter legger lærerne vekt på i utformingen av faget?

Which different practices exists in the subject of programming in our secondary schools, and which aims do teachers utilize when implementing programming in their classrooms?

Ellen Marje Thorland

Master IKT Læring

Høgskulen på Vestlandet

Anders Grov Nilsen

14. september 2018

Forord: Fra 1984 via Kodèmus til Black Mirror

«Sometimes a scream is better than a thesis». - Manfred Eigen

Det har vært en utfordring å studere og jobbe samtidig disse to årene, selv om jeg fikk nedsatt stilling andre året. Det å bevege seg mellom forsker/studentrolle samtidig som man skal være lærer og kollega har ført til lett frustrasjon, og læringskurven har vært særdeles bratt innimellom. Det er en del som nok kunne blitt gjort bedre og annerledes ved arbeidet av denne oppgaven, men jeg håper likevel at den kan tenne en aldri så liten inspirasjonsgnist for andre som er interessert i å tittle litt på hvilken rolle programmering og IKT kommer til å spille i den norske skolen framover.

Det er flere som bør takkes for god hjelp og støtte i arbeidet med denne oppgaven. Først og fremst vil jeg takke informantene og deres arbeidsplasser for å være så positive og villige til å være med på studien. Videre vil jeg takke rektor og gode kolleger for forståelse for fravær, både rent fysisk når vi hadde samlinger, og mentalt de dagene jeg har surra rundt i en egen boble. Videre vil jeg rette en takk til min meget tålmodige veileder, Anders Grov Nilsen, samt andre lærere på høyskolen som har hjulpet til i arbeidet. En ekstra takk til Elen Instafjord som fikk meg til å puste rolig igjen etter en liten «meltdown» over forskningsspørsmålene, samt en stor takk til Hein Berdinesen som har sett over oppgaven og kommet gode råd i innspurten. Jeg vil også rette en takk til min bedre halvdel, Thorbjørn, som hatt bistått teknisk når ting ikke virker, samt støttet og oppmuntret meg hele veien. Jeg vil også på forhånd beklage en noe rufsete måte å referere kilder på, da jeg i god tro har brukt samme metoder i lengre tid uten kommentarer. Jeg ser i ettertid at ved et så stort arbeid, burde jeg nok ha satt meg grundigere inn i referanselogistikken.

Til sist vil jeg takke alle sciencefiction-forfatterne der ute som tente interessen min for teknologi og vidløftige drømmer. Som en selverklært bokorm og sci-fi nerd var jeg en av dem som drømte om flyvende biler og ferier i verdensrommet. Science fiction forfattere har alltid hatt en tendens til å «spå» framtiden både på godt og vondt, fra George Orwells 1984 til dagens «film noveller» fra Channel 4s serie Black Mirror. Går vi mot et samfunn der alt er kodet og programmert? Det kan være litt søkt å ha med dette i forordet og i oppgaven generelt, men som en som synes det er mye kjekkere med fiction enn fakta, drister jeg meg likevel til å ta med noen verk som har inspirert og til

tider skremt vannet av meg. Det kan også være til inspirasjon til gøyale lesning/titting for den som måtte finne på å lese denne oppgaven.

1984 ble utgitt i 1949 og skildrer et totalitært samfunn der «Storebror» ser alt du gjør. Hovedpersonen Winston Smith jobber som redaktør for «The Ministry of Truth», der han skriver om historiske hendelser slik at de skal bli godkjent av regimets ledelse. Det er skjerm i alle hus, som gir «Storebror» anledning til å se og høre innbyggerne i Oceania 24 timer i døgnet. (Orwell, 1949). All media er manipulert og folket får bare lese det som blir godkjent av regimet. Som en kuriositet har salget av denne boken økt en god del siden høsten 2016.

I novellen *Kodèmus*, skrevet av den norske forfatteren Tor Åge Bringsværd i 1968, møter vi et samfunn der menneskene blir styrt av en sentral datahjerne. Denne hjernen har en filial, Lillebror, hos hvert enkelt menneske. Lillebror er en liten datamaskin som styrer hovedpersonen i novellen, *Kodèmus*, og det er ikke plass for personlige initiativ av noe slag. (Bringsværd, 1968). Informasjonsteknologi spiller en viktig rolle i *Kodèmus*, og det er spennende å sammenligne beskrivelsen av IT fra 1968 med dagens IT i 2018. I dag går vi jo alle rundt med lillebrødre (smarttelefonen) på oss.

Black Mirror er en britisk science fiction serie skapt av Charlie Brooker. Serien tar for seg det moderne samfunnet, der man kommenterer uforutsette konsekvenser ved utviklingen og bruken av ny teknologi. Episodene er ofte satt i en alternativ versjon av dagens samfunn eller vår nære framtid (Channel 4, 2018).

Når vi ser på den rivende teknologiske utviklingen og implementeringen og gjennomsyringen av den i alle deler av samfunnet vårt, synes jeg det er et spennende tankesprang å se på litteraturen som bakteppe for oppgaven. Selv om vi fremdeles ikke har flyvende biler.

Sammendrag

I 2016 kom programmering som valgfag på lærerplanen for grunnskolen i Norge for første gang. I første omgang skulle faget tilbys til utvalgte skoler som ville være med på prosjektet, og det var 155 skoler som satte i gang høsten 2016. Høsten 2017 ble det åpnet for at alle skoler som ønsket å tilby programmering som valgfag, kunne gjøre dette.

Hovedproblemstillingen i denne oppgaven er som følger:

Hvilke ulike praksiser eksisterer for faget programmering i ungdomsskolen, og hvilke innholdskomponenter legger lærerne vekt på i utformingen av faget?

Jeg har forsøkt å fokusere på lærerne i faget og hvordan skolene organiserer og implementerer undervisningen i koding, samt at jeg også har sett på hvilke læringsmål i faget lærerne mener er viktige og hvordan de jobber praktisk med faget. Jeg har også sett på hvem som er lærere i faget, hvilken holdning som finnes til IKT faget i skolene og hvordan lærerne ser på mulighetene for at koding kan bli et eget fag i den norske skolen. I arbeidet med oppgaven har jeg forsøkt å finne og lese så mye relevant litteratur og tidligere forskning jeg kunne finne om emnet. Det at det hele tiden publiseres ny forskning på løpende bånd, har gjort dette arbeidet komplisert og tidkrevende da det har vært vanskelig å begrense hvilke arbeider som er verdt å lese og hvilke som er mindre relevante. Det er også vanskelig å klare å stoppe opp og slutte å lete etter nytt og relevant materiale.

Metoden jeg har brukt i arbeidet med oppgava er hovedsakelig kvalitative dybdeintervjuer, men jeg har også foretatt en liten kvantitativ undersøkelse der 42 lærere deltok. Blant funnene i arbeidet kan det nevnes at det er en forståelse for viktigheten av å jobbe med programmering i skolen, men at det er uenighet om faget skal være et eget fag eller om det skal fortsette å være et valgfag. Andre funn viser at det er få lærere som har utdanning og erfaring i koding, noe som kan bli et problem når vi vet hvor fort utviklingen i IKT feltet går. Jeg har et håp om at denne oppgaven kan inspirere andre til å se nærmere på programmeringsfaget og dets plass i den norske skolen i framtiden.

Abstract

In 2016 programming as a chosen subject was introduced to secondary schools in Norway for the first time. Primarily programming was to be offered to selected schools interested to join the project, and 155 schools across Norway offered programming starting in the autumn of 2016. From the start of the new school year in 2017, the government announced that any secondary school in Norway could offer programming if they had the resources. The main research question of this study is: What different practices exist for programming as subject in secondary education, and what components and learning targets teachers choose to teach when planning their teaching?

In this study, I have tried to focus on the teachers and how the schools organise and implement the teachings in coding, as well as having a look at the different competences teachers feel is important and how the teachers work practically with coding. I have also tried to find out who the teachers are, as well as the attitude towards ICT in their schools. I have also tried to gauge the possibilities for making programming a core subject in the Norwegian curriculum. During the work, I have tried to read as much relevant literature as possible, as well as previous published research on this topic. The challenge here was that new, relevant and important research are published all the time, thus making it somewhat difficult to limit what works to read and basically knowing when to stop.

I have used mixed methods when working on this paper, but mainly my findings are found within a qualitative framework based on in depth interviews with four teachers from different schools in the country. I also conducted a short quantitative survey, where 42 of 155 teachers participated. Some of my findings include an understanding for the importance of implementing programming into our schools, however there is an argument about whether programming should continue to be an elective subject or become a part of the common core curriculum. Other findings show that there are currently few teachers that have formal qualifications in programming, something that can become a problem when we know how fast technology changes.

In hindsight I see that there are several things I could have done differently and no doubt better, but for better and worse, this is my work and with any luck, my study will

inspire others to take a closer look at programming as a subject and its place within the Norwegian school system.



Innhold

Forord: Fra 1984 via Kodèmus til Black Mirror	ii
Sammendrag.....	iv
Abstract	v
1. Innledning	3
1.1 Bakgrunn for valg av tema	3
2. Problemstilling	8
2.1 Problemstilling	8
F1. Hvordan organiserer lærerne faget?	9
F2 Hvilket nivå – og hvilke kompetansemål legges det vekt på i kodeundervisningen?	9
3. Begrepsforklaring.....	10
3.1 Vurdering	10
3.2 Læreplanteori	10
3.3 Motivasjon.....	11
3.4 Paradigme.....	11
3.5 Programmering og koding.....	11
3.6 Algoritmisk tankegang	12
4. Tidligere forskning.....	13
4.1 Forskningsoversikt (litteraturreview).....	14
4.2 Kodefaget – hvorfor skal man undervise i koding, hva skal man undervise, hvilke vanskeligheter er det, og hvordan gjør vi det?	16
4.2.1 Hvorfor skal man undervise i koding?	19
4.2.2 Hva skal man lære?	23
4.2.3 Hva kan hindre læring i IKT?	25
4.2.4 Hvordan skal man undervise i programmering?	30
5. Teori	32
5.1 Konstruktivismen	34
5.2 Konstruksjonisme og det sosiokulturelle synet på læring i den moderne skolen.	35

5.3 Sosiokulturell teori	38
5.4 Vurdering for læring.....	40
5.5 Pragmatismen og positivismen.....	47
6. Metode.....	49
6.1 Kvantitativ metode	49
6.2 Kvalitativ metode	50
6.3 Fordeler og begrensinger i valg av metode	51
7. Funn, analyse og drøfting.....	52
7.1 Metodekritikk	52
7.1.1 Om utvalget	52
7.1.2 Bortfallsanalyse og svarrespons	53
7.1.3 Innhenting av data – survey	53
7.1.4 Transkribering av intervjuer.....	54
7.1.5 Reliabilitet og validitet	55
7.2 Funn og drøfting.....	56
F.1 Hvordan organiserer lærerne faget?.....	65
F.2 Hvilket nivå – og hvilke kompetansemål legges det vekt på i kodeundervisningen?.....	75
8. Konklusjon og veien videre.....	81
Problemstilling:	81
Forskningsspørsmål 1: Hvordan organiserer lærerne faget?.....	81
Forskningsspørsmål 2: Hvilket nivå – og hvilke kompetansemål legges det vekt på i kodeundervisningen?.....	83
Bibliografi	86
Vedlegg:	93

1. Innledning

1.1 Bakgrunn for valg av tema¹

Hvor befinner det norske samfunnet seg generelt og den norske skolen seg spesielt når det gjelder teknologi, programmering og koding? Storbritannia innførte koding på læreplanen allerede i 2014 for elever helt ned i 5 års alderen og i Finland satte de i gang med det samme i 2016. I følge en fersk rapport fra 2018, «The Nordic approach to introducing Computational thinking and Programming in compulsory education», befinner Norge seg et godt stykke etter våre nordiske naboland når det gjelder innføring av koding i grunnskoleløpet (Bocconi, 2018).

«Teknologi i praksis» har allerede vært et valgfag i skolen siden 2012. Formålet med dette faget var at elevene skulle utvikle et mer aktivt forhold til en teknologisk hverdag. I fagets læreplan heter det: «Valgfaget teknologi i praksis skal motivere elevene til å utvikle teknologiske produkt med utgangspunkt i lokale behov og problemstillinger» (Utdanningsdirektoratet, 2012).

Fra høsten 2016 har altså programmering og koding kommet inn på læreplanen om enn bare som forsøksfag. Det finnes lite forskning på koding som fag her hjemme ennå, men man kan likevel finne artikler som indikerer at IKT satsingen i skolen generelt bør ses nærmere på. Resultater fra ICILS 2013 (International Computer and Information Literacy Study) viser at Norge scorer høyere enn det internasjonale gjennomsnittet, og 30% av norske elever karakteriseres som digitalt kompetente. Det er likevel viktig å understreke at denne studien viser kun konsumet av digitale hjelpemidler og medier og ikke produksjonen av disse (Ottestad, 2014).

Faget programmering/koding ble introdusert som et nytt i valgfag på ungdomstrinnet fra høsten 2016 som et prøveprosjekt som skal gå over tre år. 155 skoler over hele landet er med på prosjektet der både kommunale og private skoler er inkludert. Det undervises allerede i programmering og datafag ved videregående skoler, samt at det er startet opp et nytt forsøk i videregående opplæring høsten 2017. Faget; Programmering og modellering X, bygger på matematikk 1T og er spesielt beregnet for elever på Vg2 som velger matematikk R1 (Utdanningsdirektoratet, 2018). I tillegg gjennom arbeidet med

¹ Innledningen er bearbeidet og omskrevet fra tidligere innlevert forskningskisse.

fagfornyelsen, er det indikert at programmering skal innføres i blant annet matematikk gjennomgående fra 1. trinn og ut grunnopplæringen. De nye lærerplanene skal være klare for elevene høsten 2020. (Utdanningsdirektoratet, 2017).

Stortinget og regjeringen lenge vært opptatt av å sette lys på hvordan IKT brukes på skolene i landet. Allerede i 2004 kom Stortingsmelding nummer 30; Kultur for læring- En skole for kunnskap, mangfold og likeverd, der det ble hevdet at: «Gode opplevelser, erfaringer og læring innenfor det digitale området (IKT) kan motivere barn til å lære mer og kan være et viktig bidrag i et livslangt læringsperspektiv» (Kunnskapsdepartementet, 2003/2004).

I Stortingsmelding 31 (2007/2008) har Kunnskapsdepartementet med et kapittel som drøfter bruk av variert undervisning og bruk av IT i klasserommet. Her hevdes det at det er kvaliteten på undervisningen som fører til gode resultater og ikke nødvendigvis arbeidsformene. Forskning viser likevel at lærere som bruker og behersker ulike metoder, har elever som oppnår bedre resultater. «Læringsplakaten legger vekt på at opplæringen skal fremme tilpasset opplæring og varierte arbeidsmåter». (Kunnskapsdepartement, 2008).

I Stortingsmelding 27 (2015/2016) Digital agenda for Norge – IKT for en enklere hverdag og økt produktivitet, skrives det at formålet med meldingen er å presentere regjeringens overordnede politikk for hvordan vi i Norge kan utnytte IKT til samfunnets beste. Videre heter det at man ønsker en styrket digital kompetanse og deltakelse.

Digital kompetanse, fra grunnopplæringen og gjennom alle faser i livet, skal styrkes for å sikre deltakelse og tillit til digitale løsninger. Digitale tjenester skal være lette å forstå og lette å bruke. Avansert IKT-kompetanse og IKT-forskning er grunnleggende forutsetninger for digitalisering av Norge. (Kommunal – og moderniseringsdepartementet, 2015/2016).

Digitale ferdigheter er en av de fem grunnleggende ferdighetene elevene i norsk skole. Grunnleggende ferdigheter beskrives som en forutsetning for læring og utvikling i skolen, samfunn og arbeidsliv. Dermed blir det å kunne bruke digitale verktøy en ferdighet på lik linje med lesing, regning, muntlige ferdigheter og skriving, og skal integreres i læreplanene på tvers av fagene.

Digitale ferdigheter defineres slik:

Digitale ferdigheter vil si å kunne bruke digitale verktøy, medier og ressurser hensiktsmessig og forsvarlig for å løse praktiske oppgaver, innhente og behandle informasjon, skape digitale produkter og kommunisere. Digitale ferdigheter innebærer også å utvikle digital dømmekraft gjennom å tilegne seg kunnskap og gode strategier for nettbruk.» (Anders Sanne, 2016). Utdanningsdirektoratet definerer programmeringsfaget slik:

Valgfaget programmering handler om å lage programkode, det vil si et sett med regler og uttrykk for å styre digitale enheter. I dette inngår prosessen fra å identifisere problemer og utforme mulige løsninger, til å lage kode som kan forstås av en datamaskin, systematisk feilsøke og forbedre denne koden, og dokumentere løsningen på en forståelig måte. Det omfatter alle nivåer fra å forutse og analysere hva et program skal gjøre, til å kjenne igjen mønstre, eksperimentere og evaluere mulige løsninger, og samarbeide med andre. Summen av disse ferdighetene kalles algoritmisk tankegang. Opplæringen skal legge til rette for at elevene lærer å løse problemer på nye måter. Elevene skal få muligheten til å utvikle sin kreativitet og skape produkter ved hjelp av programmering. Gjennom å lage programmer oppøver elevene også ferdigheter i å vurdere eget og andres arbeid, gi konstruktive tilbakemeldinger og samarbeide med andre. Å skape og produsere digitalt, krever forståelse og kompetanse i programmering. (Utdanningsdirektoratet, 2016).

Ved lanseringen av faget høsten 2016 sa daværende utdanningsminister Thorbjørn Røe Isaksen blant annet at:

Prosjektet med programmering som valgfag må sees i sammenheng med den nye realfagstrategien «Tett på realfag». Dette er en overordnet strategi for arbeidet med matematikk, naturfag og teknologi i skole og barnehage fra 2015-2019. Koding har vært etterspurt som fag i skolen. Nå kan skoler over hele landet søke om å delta i et prøveprosjekt med programmering som valgfag. (Regjeringa, 2015).

Denne satsingen vil sannsynligvis være en utfordring da man kan ikke regne med at alle lærerne som skal ha faget har utdanning eller omfattende kunnskap om koding. På bakgrunn av dette har man opprettet et samarbeid med Senter for IKT i utdanningen.

Det er verdt å merke seg at Senter for IKT i utdanningen har fusjonert med Utdanningsdirektoratet.

På Utdanningsdirektoratet sider heter det:

I den forbindelse har Senter for IKT i utdanningen laget en MOOC for grunnleggende innføring i programmering for lærere. MOOCen er en åpen ressurs, slik at alle som ønske det kan få innføring i noen grunnleggende begreper og konkrete tips til hvordan man kan gjennomføre undervisning i faget, støttet med gode eksempler på bruk av teknologi. (Utdanningsdirektoratet, 2016).

Pilotprosjektet, vil som nevnt tidligere, gå over tre år, og både kommunale og private skoler kunne søke for å være med på satsinga. En del av skolene fikk økonomisk støtte til gjennomføring av faget. Det ble utviklet en egen forsøkslæreplan for faget som skulle gjelde for de skolene som ble med, og regjeringen satte av 15 millioner kroner for å finansiere prosjektet for de tre årene forsøket vil vare. Det viste seg interessen for kodefaget var så stort at det var flere skoler som ville henge seg på enn de som i utgangspunktet fikk være med, så den opprinnelige planen ble revidert. Regjeringen åpnet derfor for at alle kommuner kan tilby programmering i forsøksperioden. De skolene som tilbyr faget fra høsten 2017, vil ikke være en del av selve forsøket og de vil heller ikke kunne søke om økonomisk støtte til faget. (Utdanningsdirektoratet, 2017).

I følge statistikk fra Barne-, ungdoms og familiedirektoratet fra 2014 finner vi at datavanene og mediebruken til barn og ungdom har endret seg mye de siste årene. De bruker nye medier, de sosiale rammene er endret og de bruker mye mer tid på teknologi enn tidligere. Selv om tallene viser at tidsbruken barn og unge bruker på internett øker, bruker de en god del tid sammen med familie og å treffe venner. Bruken av internett øker med alder; 5–16-åringer bruker i snitt over 3,5 timer per døgn på nettet. (Medietilsynet, 2016). Undersøkelsen viser samtidig at unge mennesker er ikke bare er konsumenter av innholdet de finner på nettet, men de produserer også innhold selv. Ikke minst gjelder det bilder og videoopptak som deles med andre. Her kan nevnes applikasjoner som Instagram, Snap Chat og Youtube.

I Norge har undervisning og opplæring av digitale ferdigheter i grunnskolen hovedsakelig vært rettet mot det å bruke teknologi. «Fra 2006 har digitale ferdigheter blitt definert som en av de fem grunnleggende ferdighetene barn og unge skal tilegne seg i løpet av skoleløpet, fra barnehagen til og med videregående». (Kommunal - og moderniseringsdepartementet, 2013). Innenfor EU er den digitale kompetansen en av åtte nøkkelkompetanser og ifølge beskrivelsene skal det vektlegges en «*confident, critical and creative*» bruk av IKT. Videre står det at digital kompetanse ikke er et eget skolefag for EU eller Norge, men heller en tverrfaglig kompetanse som inngår i alle fag. I følge Digiutvalget er det gjort mye godt arbeid med innføringen av digitale ferdigheter i skolen. Men de hevder også at det : Ensidig er blitt lagt for stor vekt på verktøykompetanse, kommunikasjon i digitale medier og for lite vekt på den kritiske refleksjonen rundt teknologisamfunnet og på det å skape teknologi.» (Kommunal - og moderniseringsdepartementet, 2013). Med andre ord er det «en ubalanse hvor digitale ferdigheter er definert med for stor vekt på kommunikasjon, tekst og det humanistiske, og med for lite vekt på algoritmer, tall, matte og teknologi.» (Kommunal - og moderniseringsdepartementet, 2013).

Det har vært lite fokus på utvikling og eksperimentering med teknologi i skolen tidligere, selv om det har foreligget planer om dette i ulike stortingsmeldinger som jeg referert til tidligere. Opplæring i utvikling, koding og lek med teknologien har i stor grad vært overlatt og ivaretatt av frivillige organisasjoner som «Lær kidsa koding» og Digitalpedagogene. Sammen med arbeidstakerorganisasjonene NITO og Tekna, samt arbeidsgiverorganisasjonen Abelia mener IKT - Norge at det bør et nasjonalt løft til for å heve IT-kompetansen i Norge. Disse organisasjonene vil gjøre koding obligatorisk i skolen, samt at de vil opprette 1.000 flere studieplasser innenfor IKT og kartlegge behovet for arbeidskraft. «IKT- Norge vil berømme regjeringen på at de vil innføre koding som en obligatorisk del i skolen, samt deres satsing på lærernes digitale kompetanse. Teknologiforståelse er avgjørende for å ruste norske barn for fremtiden,» sier IKT-Norges administrerende direktør Heidi Austlid. (NTB, 2017).

Koding er på frammarsj og det finnes nå kodeklubber over hele verden. Sammen med flere kjente mennesker fronter Malala Yousafzai og Barack Obama en nettkampanje i regi av organisasjonen Code.org. Kampanjen fremmer at alle kan og bør lære seg å

kode, om enn bare litt. Code.org organiserer blant annet hvert år en «EU Code Week», der målet er å få eksperter innen programmering eller andre IT kyndige til å besøke skoler der de kan gi elevene en kort, enkel og inspirerende innføring i koding og programmering. I Norge kalles dette «Kodetimen» og frontes av Lær Kidsa Koding og ifølge nettsidene deres øker antallet elever som er med på Kodetimen i Norge for hvert år. I 2013 deltok ca. 10 000 norske elever og i 2017 deltok over 100.000 elever. (Lær kidsa koding, 2016).

Med dette som bakgrunnsteppe bør det være både interessant og relevant å se nærmere på valgfaget programmering i grunnskolen og jeg har derfor valgt å forske på dette i oppgaven min. I tillegg er skolen min en av skolene som har vært med på prosjektet fra starten av, og vi skal nå i gang med det tredje året. Faget har vist seg å være såpass populært blant elevene våre og vi vil fra høsten 2018 ha to kodelærere hos oss og vi kan da tilby enda flere elever faget. Vi tilbyr faget på alle tre klassetrinn og vil ha aldersblandede grupper. Siden elevene kan velge nye fag hvert år, vil vi til høsten ha to grupper, en for nybegynnere og en gruppe for de som har hatt faget tidligere og er klar for mer avansert jobbing med faget.

2. Problemstilling²

2.1 Problemstilling

Hvilke ulike praksiser eksisterer for faget programmering i ungdomsskolen, og hvilke innholdskomponenter legger lærerne vekt på i utformingen av faget?

Problemstillingen ovenfor er kjernen i det arbeidet jeg ønsker å jobbe med i denne oppgaven. Jeg vil forsøke å finne ut hvordan undervisningen i faget er organisert og hva som blir undervist sett i lys av læreplanen som ble publisert høsten 2016. Dette er interessant da det kan være ulike måter å organisere faget på, samt at kunnskaper og utdanning hos den enkelte lærer vil påvirke hva som blir undervist, og hvilke kompetansemål som legges vekt på i undervisningen. Ut fra denne problemstillingen har jeg kommet frem til følgende to forskningsspørsmål:

² Dette er bearbeidet og omskrevet arbeid fra tidligere innlevert forskningskisse.

F1. Hvordan organiserer lærerne faget?

Dette vil kunne si noe om hvordan lærerne organiserer faget på skolen. Hvor store klasser har man, er det elever fra alle trinn i samme klasse, hvor mange timer i uka undervises det i og hvor mange lærere er inne i klassen samtidig. Jeg vil også undersøke om man foretrekker individuelle oppgaver eller gruppearbeid, hvordan elevene jobber og hva lærerne tenker om fagets framtid i skolen. Det er også naturlig å finne ut noe om hvilke plattformer det jobbes på og hvilken holdning lærerne/skolen og eventuelt skoleeier ser på IKT i skolen.

F2 Hvilket nivå – og hvilke kompetansemål legges det vekt på i kodeundervisningen?

Dette vil kunne si noe om hvilke nivå læreren legger innholdet på og hvilken rolle produkt og prosess har i faget. Dette gir igjen indikasjoner på didaktiske valg og begrunnelser lærerne av disse. Her vil det også være naturlig å undersøke hvordan man vurderer koding da man i utgangspunktet kan få tre ulike karakterer fordelt over tre år om eleven velger faget i alle de tre årene de går på ungdomsskolen. Hvordan underviser man i et fag der man i utgangspunktet i andre året og tredje året av prosjektet kan få en klasse bestående av elever som har gjennomført et eller to år av faget mens noen elever er helt uerfarne og har aldri forsøkt å programmere før?

Andre elementer det kan være naturlig å se på her er hvordan man gjennomfører et vellykket undervisningsopplegg og hvordan lærerne vektlegger hovedområdene i faget; modellering og koding. Selv om det er to hovedområder, må de sees i sammenheng, da modellering og koding ikke er adskilte aktiviteter. Det kan tenkes at lærerne i faget vektlegger disse komponentene ulikt etter hvilken erfaring og utdanning den enkelte lærer har. I tillegg vil man kunne finne ut noe om hva lærerne mener om elevenes muligheter til å utvikle nye ferdigheter ved å lære koding og om elevene evner å utvikle kritisk tenkning gjennom arbeidet med faget. Jeg vil også finne ut om lærerne bruker egenproduserte eller eksterne ressurser, og hvilke ressurser de eventuelt bruker.

3. Begrepsforklaring

3.1 Vurdering

I oppgaven skal jeg blant annet se på vurdering og vurderingsformer lærerne bruker i kodefaget på ungdomsskolen. Derfor kan det være nyttig forklare hva vurdering er.

Skolebasert vurdering har ifølge Lars Monsen ingen lang historie i Norge. (Haaland Ø. H., 2011, s. 176). Det første forsøket av en viss størrelse startet i 1998 av Rådet for videregående opplæring som satte i gang en debatt om manglende kunnskap i utdanningssystemet i Norge. Senere har det vært flere debatter rundt dette, og det ble da særlig lagt vekt på at man burde utvikle en skolebasert vurdering i motsetning til ulike former for ekstern vurdering. Jeg vil komme nærmere inn på VFL, altså vurdering for læring i teorikapittelet.

3.2 Læreplanteori

Noe bakgrunnskunnskap om læreplanteori kan være med på å gi en forståelse for hvordan det er mulig å endre praksis gjennom utdanningsreformer. Begrepet læringsplan brukes som et overordnet begrep for planlegging av det som skjer i skolen. Det er elevenes læring som skal stå i sentrum. (Gundem, 1990). Ordet læringsplan kommer fra det latinske ordet curriculum (lat – currere) kan forstås som et løp der reglene for løpet er forhåndsbestemt. Dette indikerer at fagene i skolen har et innhold som er fastlagt på forhånd og som beskriver hvordan lærerne skal undervise og hva elevene skal lære.

I den generelle delen av kunnskapsløftet, som er det styrende dokumentet i den norske skolen, heter det at:

«Læreplanens generelle del utdyper formålsparagrafen i opplæringsloven, angir overordnede mål for opplæringen og inneholder det verdimessige, kulturelle og kunnskapsmessige grunnlaget for grunnskolen og videregående opplæring. Generell del av læreplanverket er videreført fra R-94 og L-97, men skal oppdateres i forbindelse med fornyelsen av fagene i skolen.» (Utdanningsdirektoratet, 2015).

3.3 Motivasjon

Siden motivasjon er blitt diskutert ut fra forskningsspørsmålene, kan det være formålstjenlig å kort definere begrepet. Motivasjonsforskningen skiller som regel mellom indre og ytre motivasjon. Indre motivasjon forstås som at motivasjonen oppnås i selve aktiviteten, altså det man gjør er spennende og tilfredsstillende i seg selv. Ytre motivasjon blir oftest forstått som en aktivitet som man utfører for å få ros og anerkjennelse for arbeidet som er utført. (Skaalvik, 2015). Wormnes og Manger skriver at motivasjon er en nødvendig forutsetning for læring, samtidig som de sier at en del motivasjonsforskere har tatt i bruk andre begreper enn indre og ytre motivasjon. I stedet blir begrepene mestringsorientering og prestasjonsorientering brukt. (Wormnes, 2005) Mestringsorientering handler om et fokus på å øke egen kompetanse, mens prestasjonsorientering handler om det å oppnå resultater som enn kan vise til andre.

3.4 Paradigme

Krumsvik refererer til Guba og Lincoln som forklarer paradigme som et allment akseptert system som skal hjelpe forskeren til å forstå ontologi, epistologi og metode i forskningsarbeidet. (Krumsvik, Forskningsdesign og kvalitativ metode, ei innføring, 2014).

3.5 Programmering og koding

I denne oppgaven bruker jeg begrepene programmering og koding om hverandre, da jeg tenker at begrepene dekker det samme. Det kan likevel være noe usikkerhet i hva man legger i disse begrepene. Da jeg sendte ut forespørsel til skolene til å delta i surveyen, fikk jeg en melding fra en skole at de ikke hadde koding, men programmering som valgfag. Svaret fikk meg til å stoppe opp litt, og jeg har forsøkt å finne ut om det er noen forskjeller. Det er vanskelig å finne noen skiller, men det kan tenkes at en programmerer er en som har formell utdanning innenfor faget. I følge en artikkel publisert av Differencebetween.net hevdes det at:

While both the terms are synonymous with each other and are often used interchangeably, they are quite different from each other. Coding simply means writing codes from one language to another such as from English to Java. It is less intimidating

and less intensive. Programming, on the other hand, means to program a machine with a set of instructions to run. (Difference between.net, 2018). De skriver vidare at en som koder er en som oversetter logikk til et språk som datamaskinen vil forstå. Kodere er mer språkorienterte, mens programmerere er mer opptatt av sammenhengen og det store bildet. En programmerer har ofte formell utdanning og erfaring, mens en koder er en programmerer med mindre ekspertise som ikke er like detaljorienterte. (Difference between.net, 2018).

3.6 Algoritmisk tankegang

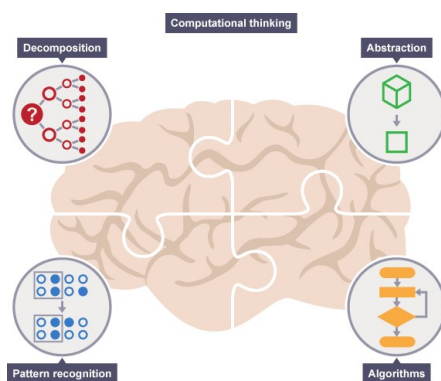
Begrepet algoritmisk tankegang kommer fra det engelske uttrykket «Computational thinking skills» og ble først brukt av Seymour Papert i boken «Mindstorms; Children, Computers and Powerful Ideas» (Papert, 1993).

En god definisjon på hva begrepet innebærer er som følger: «Computational thinking allows us to take a complex problem, understand what the problem is and develop possible solutions. We can then present these solutions in a way that a computer, a human, or both, can understand». (BBC, 2018).

Videre skriver de at det er fire hjørnesteiner som må på plass for å beherske algoritmisk tankegang:

1. Forminske et vanskelig problem eller system i mindre og håndterlige deler
2. Gjenkjennelse av mønster, se etter likheter mellom og inni problemer
3. Det å kunne fokusere kun på viktig informasjon og overse unødvendige detaljer
4. Algoritmer, utvikle steg for steg løsninger til et problem eller et sett regler som man må følge for å løse problemet.

Hver hjørnestein er like viktige, de er som bein på et bord. Hvis et bein er borte er sannsynligheten for at bordet vil kollapse. Om man behersker og bruker disse teknikkene, vil det være til hjelp når man skal programmere en datamaskin. (BBC, 2018).



Figur 1 – De fire hjørnesteinene i algoritmisk tenking - <https://www.bbc.com/education/guides/zp92mp3/revision/1>

«Simply put, programming tells a computer what to do and how to do. Computational thinking enables you to work out exactly what to tell the computer to do». (BBC, 2018).

4. Tidligere forskning³

I denne oppgaven vil det være hensiktsmessig å undersøke hvordan den norske skolen plasserer seg når det gjelder kodefaget i skolen sammenlignet med andre land det er naturlig å sammenligne oss med. Etter omfattende søk i ERIC, Oria og andre søketjenester, har jeg oppdaget at det er lite forskning på dette området i Norge på nåværende tidspunkt. Det er heller ikke overveldende mye tidligere forskning fra utlandet, men jeg har funnet noe forskning fra Norge, samt en del fra andre land, fortrinnsvis fra Europa. I denne delen av oppgaven vil jeg vise til en oversikt over litteratur og kilder jeg har jobbet med. Arbeidet med å finne relevant litteratur og tidligere forskning har vært en pågående prosess og det publiseres stadig ny forskning og litteratur som omhandler undervisning av teknologi i skolen, både her hjemme og i utlandet. Jeg har derfor måttet begrense søkene mine til forskning publisert etter 2014 til og med 2017 slik at jeg kunne finne mest mulig relevant arbeid. En annen grunn til at jeg valgte 2014, er at European Schoolnet (EUN) startet i 2014 The Coding Initiative på oppdrag fra EU-kommisjonen. I følge et notat fra Senter for IKT i utdanningen var målet å fremme koding og informatikk i Europa, blant annet via nettstedet «*all you need is {C<3DE}*». Dette er et nettsted for elever, lærere og andre som vil utforske kodingens

³ Dette omskrevet og bearbeidet materiale fra tidligere innlevert forskningskisse.

muligheter. Siden 2014 har «CodeWeek» satt koding på agendaen en hel uke gjennom en serie arrangementer i ulike land. (Sevik, 2015).

Videre kan vi lese i notatet at European Schoolnet i oktober publiserte rapporten «Computing our future. Computer programming and coding - Priorities, school curricula and initiatives across Europe».(Anja Balanskat, 2015). Denne rapporten viser en oversikt over status for koding i skolen i 21 europeiske land. Rapporten forteller om hvordan de forskjellige landene prioriterer programmering og koding i skolen, både som del av pensum og andre initiativ. Kort oppsummert viser det seg at 16 av de 21 landene som svarte på undersøkelsen Allerede hadde koding som en del av pensum, enten på nasjonalt eller regionalt nivå. Mellom disse landene finner vi Polen, Israel, Litauen, Danmark og Spania. England innførte koding som obligatorisk del av opplæringen allerede høsten 2014, mens Frankrike, Spania og Polen har besluttet innføring i løpet av det siste året. Det er interessant å merke seg at Norge, sammen med Nederland og den belgiske regionen Vallonia, er de eneste landene/regionene som hverken har koding som en del av pensum eller konkrete planer om å innføre dette. Kunnskapsdepartementet i Norge annonserte utprøving av programmering som valgfag i norske ungdomsskoler fra skoleåret 2016/17 for utvalgte skoler, og dette ble utvidet høsten 2017. Da kunne alle skoler som ønsket dette, innføre programmering som valgfag. Jeg ser at jeg nok med fordel kunne utvidet søket til 2018, da det nok sannsynligvis har blitt publisert mer forskning etter jeg avsluttet mine søk, men for å i det hele tatt komme i gang med skrivingen, besluttet jeg å sette strek ved stoff utgitt senest 2017.

4.1 Forskningsoversikt (litteraturreview)

For å kunne etablere et rammeverk for arbeidet i denne oppgaven, har jeg forsøkt å lage et Forskningsreview for å få en oversikt over forskningsfeltet. Søket har blitt gjennomført ved å bruke ulike databaser som ERIC, Idunn, Google Scholar, Science Direct og Academic Search Direct. Figur 2 viser en oversikt over de kriteriene som er brukt. Som nevnt ovenfor har jeg valgt i hovedsak å se bort fra forskning publisert før 2014. Både kvantitative og kvalitative arbeider er inkludert for å sikre så relevant og ny forskning som mulig. Et unntak her er en studie gjennomført i Nederland i 2012. Dette er arbeidet er tatt med da den ser på forkunnskaper IKT lærerne hadde i den

nederlandske skolen før man eventuelt skulle sette i gang med programmering i skolen der. Det er verdt å merke seg at Nederland er et av landene der koding ikke er en del av pensum ennå. Jeg kan nevne at selv om jeg foruten norsk og engelsk, også søkte etter svenske og danske arbeider, men jeg fikk kun opp engelske og norske arbeider da søket ble gjennomført.

Tema	Inkludert	Ekskludert
Database/bibliotek	ERIC, Idunn, Google Scholar, Oria, Science Direct, Academic Search Elite	
Søkeord	Programmering, koding, grunnskole, coding, programming, secondary schools, implementation, core subjects, coding and kids, improving classroom teaching, programming languages, assessment, ICT and education, IKT i skolen	
Tidsramme	2012 - 2018	Arbeider publisert før 2012
Fokus	Artikler og forskning som fokuserer på innføring av koding/programmering i grunnskolen	
Type aktivitet	Undervisning og implementering av programmering i skolen	
Språk	Norsk, engelsk, dansk, svensk	
Metode	Kvantitative og kvalitative arbeider	Mixed Methods
Nivå	Grunnskole	Videregående skole, høyskoler og universitet

Figur 2. Oversikt over kriterier for inkludering og ekskludering i litteraturreview

Nedenfor er en oversikt over studiene og litteraturen jeg har tatt med i oppgaven. Som man kan se er dette artikler, masteroppgaver og rapporter fra ulike land, både i Europa og verden for øvrig.

Navn på studien	Forfatter/e	Land	År publisert
Teaching programming for secondary school: a pedagogical content knowledge based approach	Saeli, M et al	Nederland	2012
Taking Computer Science and Programming into Schools: The Glyndŵr/BCS Turing Project	Grout, V	Storbritannia	2014
Hva kan skolene lære av kodeklubbene?	Egil Bjørnevoll	Norge	2015
A new way of teaching programming skills to K 12 – students:Code.org	Kaleliogu, F.	Tyrkia	2015
The story of computing at school	Simon Humphreys et al	Storbritannia	2014
Review on teaching and learning of computational thinking through programming: What is next?	Sze Yee Lye, Joyce Hwee Ling Koh	Singapore	2014

Koding i skolen	Notat fra Senter for IKT i utdanningen	Norge	2015
Learning basic Programming Concept By Creating Games With Scratch Programming Environment	Ibrahim Ouahbi et al	Morocco	2015
Visual programming languages integrated across the curriculum in elementary school: A two year case study using "Scratch" in Five Schools	Jose-Manuel Saez-Lopez, Marcos Roman-Gonzalez, Esteban Vazquez-Cano	Spania	2016
Learning and Teaching Programming Skills in Finnish Primary Schools – The Potential of Games	Tuomas Hiltunen	Finland	2016
Correlates of success in Introductory Programming: A study with Middle School Students	Yizhou Qian & James D. Lehman	USA/Canada	2016
Exploring Media Literacy and Computational Thinking: A Game Maker Curriculum Study	Jennifer Jenson og Milena Droumeva	Canada	2016
Code to learn: Where does IT belong in the K-12 curriculum?	Jesus Moreno – León, Gregorio Robles & Marcos Román - Gonzales	Spania	2016
Task-based assessment of students 'computational thinking skills developed through visual Programming or tangible Coding Environments	Takam Djambong and Viktor Freiman	Canada	2016
The influence of game-based programming education on the algorithmic thinking	Gabor Kiss & Zuzanna Arki	Slovakia	2017

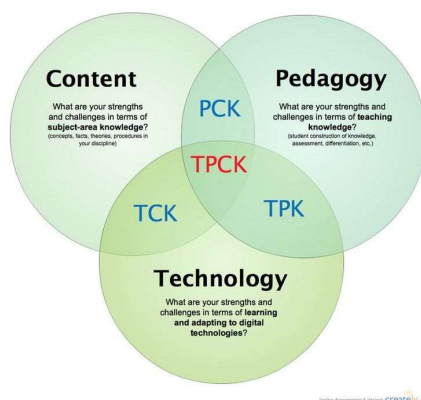
Figur 3. Oversikt over litteratur/studier jeg har brukt i arbeidet med oppgaven

4.2 Kodefaget – hvorfor skal man undervise i koding, hva skal man undervise, hvilke vanskeligheter er det, og hvordan gjør vi det?

I denne delen av oppgaven vil jeg se litt nærmere på dette. Det kan være interessant å finne ut hvilken utdanning og erfaring lærere sitter med når man skal introdusere et nytt fag i skolen, samt hva man skal undervise, hvilke vanskeligheter som kan oppstå og hvordan man skal undervise.

I en rapport skrevet av Saeli, M, et al (Saeli, 2012): «*Teaching programming for secondary school: a pedagogical content knowledge based approach*», var et av målene å finne ut om hvilken pedagogisk kunnskap lærere hadde innenfor IKT med fokus på koding (PCK - Pedagogical Content Knowledge). PCK er definert som den

kunnskapen læreren har om emnet som det skal undervises i og kunnskapen om hvordan en skal videreformidle denne til elevene de underviser.



Figur 4 – Modell av PCK

https://www.google.no/search?q=pck+education&source=lnms&tbn=isch&sa=X&ved=0ahUKEwiloNWg7aXcAhVDkSwKHXC0CSQQ_AUICigB&biw=1347&bih=633#imgrc=8hdE_D6HG0mSJM:

Dette innbefatter altså planlegging, gjennomføring og vurdering av faget. Saeli sammenfatter dette med et sitat av Lee Schulman, professor ved Stanford University og som er kreditert for å ha popularisert begrepet PCK i pedagogikken: «The ways of representing and formulating the subject that make it comprehensible to others». Schulman, 1986, p.9 (Saeli, 2012).

Målet med studien var å se på programmering som et fag, og de stilte spørsmål som hvorfor man skal undervise i koding, hva som skulle undervises, eventuelle lærevansker og hvordan elevene skulle undervises. Disse spørsmålene ble besvart sett fra lærernes standpunkt. Dermed er denne studien relevant med tanke på min egen problemstilling, der jeg ønsker å finne ut hva lærerne tenker om kodefaget og hvordan de organiserer undervisningen og hva det skal undervises i. Det er også interessant å se hvilke forkunnskaper lærerne i Nederland hadde når denne studien ble utført, sammenlignet med kunnskapene de norske lærerne hadde når valgfaget ble introdusert hos oss.

Vic Grout og Nigel Houlden skrev en rapport publisert i Science Direct i 2014, som diskuterer et prosjekt som ble satt i gang i sju skoler i England og Wales. Prosjektet fikk navnet: The Glyndwr/BCS Turing Project. (Grout, 2014). Bakgrunnen for at prosjektet kom i gang var at man innså at IKT undervisningen i den engelske skolen, sammen med mange andre land i verden, hadde blitt merkbart dårligere. I stedet for at elevene jobbet

med programmering og problemløsning, fant man at undervisningen bestod av arbeid med Software der det avsluttende prosjektet var at elevene skulle produsere en video eller en presentasjon. Dette gjenspeiles her hjemme da, undervisningen og opplæringen av digitale ferdigheter i grunnskolen hovedsakelig har vært rettet mot det å bruke teknologi ved å lage presentasjoner, filmer og å bruke Excel i matematikkfaget.

I en rapport skrevet av Takam Djambong og Viktor Freiman blir det vist til en undersøkelse bland kanadiske elever i 2012. Det viser seg at: «A number of Canadians, even those who belong to so-called “born digital” generations show the lowest levels of competence in problem solving in technology-rich environments». (Freiman, 2016). Dette bekrefter at selv om elevene i dag har vokst opp i en digital verden, er det slett ikke sikkert at de behersker den. Yizhou Qian & James D. Lehman publiserte et arbeid i Canadian Center of Science and Education der de skriver:

The demand for computing professionals in the workplace has led to increased attention to computer science education, and introductory computer science courses have been introduced at different levels of education. (Lehman, 2016).

Forfatterne undersøkte sammenhengen mellom kjønn, akademiske prestasjoner innenfor fag som ikke brukte programmering og prestasjoner hos ungdomsskoleelever som ikke hadde noe programmeringserfaring, men som fikk et introduksjonskurs i koding/programmering. De oppdaget at det var liten forskjell mellom kjønn, men at de elevene som gjorde det godt i andre fag, spesielt i matematikk og engelsk, klarte seg mye bedre i programmeringsfaget enn de elevene som ikke presterte så bra. Videre skriver de at begynnerkurs i programmering har blitt introdusert på mange nivå i skolen i USA, men at hovedvekten har vært på ungdomsskolen og videregående skole. De sammenligner dette med hva som skjer i UK der: «Even children as young as kindergarteners have been exposed to learning about robotics and programming». (Lehman, 2016).

De spekulerer om at det kunne vært en fordel å starte med programmering tidligere i skoleløpet. En av mine informanter mener også at det kan være en fordel, i alle fall om en skal nå flere jenter. Han sier at:

Det er jo litt av det samme, altså. Så med litt sånn forsiktighet. Og jeg ser og at det å lage animasjoner i Scratch, det tror jeg kunne egnet, og det har jeg sett på kurset, kunne egnet seg veldig godt på mellomtrinnet. Og, og, hvis de da har hatt det litt på mellomtrinnet kunne de gjerne fått oppgavene der hos oss. Og det vil jo komme inn under flere fag. Eg trur nok det om vi skal få jentene mer interessert i dette, så må vi begynne tidligere. Kanskje de bør ha litt om koding på mellomtrinnet. Så eg føler kanskje at dette bør inn i, litt inn i de vanlige fagene, inn som tema. (informant 4).

4.2.1 Hvorfor skal man undervise i koding?

Saeli et al siterer ulike «respekterte menneskers» meninger når de svarer på hvorfor man skal undervise programmering til unge elever. Seymour Papert & Alan Kay, hevder at: «In learning to program one learns powerful problem-solving/design/thinking strategies». (Saeli, 2012). Videre i rapporten skriver man at når elevene behersker problemløsning lært via programmering, kan man overføre disse kunnskapene til andre fag. Spesielt i matematikkfaget tenker en del forskere at programmering kan være med på å få elevene til å forstå konsepter som variabler, funksjoner, komposisjon og andre generelle områder i faget. Dette gjenspeiles i en av mine informanter når han sier at:

Programmering skal bli ein del av naturfag og matte, og det er eg for. Eg ser at det kan være nyttig for koordinatsystem med x og y og negative tall og positive tall, dette bruker me, dette er kvardagsting innanfor koding, så eg ser koding har ein plass innanfor matte. (informant 2).

En annen av mine informanter er ikke enig i dette, hun sier at:

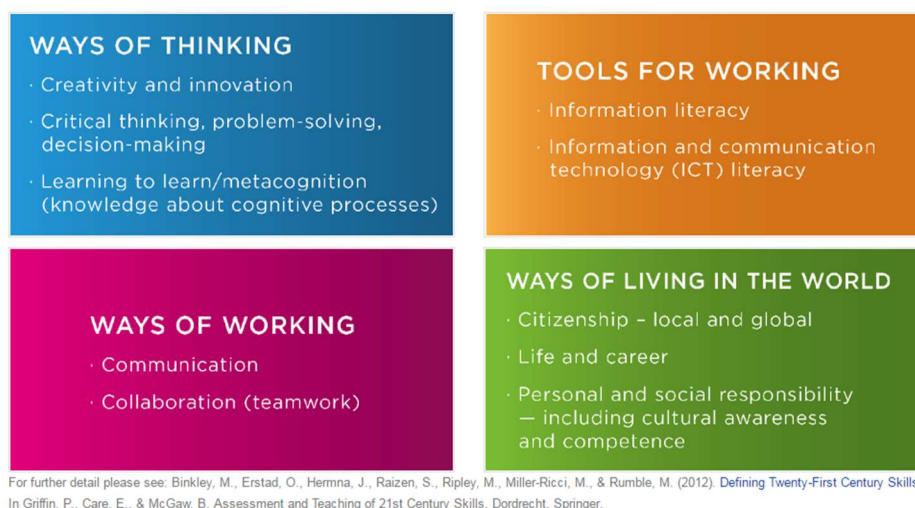
Så vet jeg jo eller jeg tror at koding skal inn i matte i den nye læreplanen. Det tror jeg ikke er det lureste, men er ikke jeg som skal bestemme det. Jeg vil helst hatt et eget fag som man fokuserer på koding og Nettvett og sosiale normer på Internett og hele denne... jeg føler det er et behov der som ikke er dekket. (informant 1).

En tredje informant sier det slik:

Nei, jeg håper jo at det kunne blitt et eget fag. Det tror jeg samfunnet ville tjent på. Fordi at jeg mener at det trengs programmerere på så mange ulike nivå og jeg tror at veldig mye av jobbene fremover vil bli å på en måte programmere, legge til rette i ulike type yrker rett og slett. Og jeg tror det vil være mange nye typer jobber som vi ikke har hatt som berører koding på en eller annen måte. (informant 3).

Eg kunne tenkt meg at noe av det kom inn som, i fagene. Ja, fordi at altså.. Vil jo tro at store deler av norske mattelærere vil jo da tatt naglene på veggen, men eg føler at koding har litt med noe av det som mange elever mangler i matematikken. Dette at du gjør noe i flere ting. Det er algoritmen. Altså de fleste av våre elever sliter i matematikk. De ser på oppgaven og ser svaret og aner ikke hva de har gjort. Hvis du finner koding som passer til matematikkfaget, så kan vi nemlig på en måte få frem trinnene. På samme måten som de og må tenke mer bevisst som når de skal ha det inn i Excel. (informant 4)

Senter for IKT i utdanningen skriver i et notat; Koding i skolen, publisert i 2015 at: «Koding eller programmering i skolen ofte begrunnes med at det er nødvendig kompetanse for å lære, arbeide og leve i dagens og morgendagens samfunn». (Senter for IKT i utdanningen, 2015). Det er mange ulike definisjoner på hva som ligger i begrepene kompetanser og ferdigheter, men de fleste forsøk inneholder begreper som kreativitet, innovasjon, kritisk tenkning, metakognisjon, kommunikasjon, samarbeid, digital dannelse og digital kompetanse, samt medborgerskap, karriere og arbeidsliv. Senter for IKT i utdanningen refererer til Binkley et al, som med prosjektet *Assessments and teaching of 21st Century skills*, definerte disse ferdighetene i fire store kategorier, som vist nedenfor.



Figur 5 – 21st Century skills - <https://resources.ats2020.eu/resource-details/LITR/ATC21s>

Notatet siterer også en rapport Coding our Future (European Schoolnet 2015); der det slås fast at selv om skolen skal utdanne elevene til framtidens yrker, er det enda viktigere å kunne:

Utstyre dagens unge med den nødvendige kompetanse for å mestre og skape egne digitale teknologi; «We believe that teaching and learning how to code, in formal and non-formal education settings, will play a significant role in this process». (Senter for IKT i utdanningen, 2015).

Filiz Kalelioğlu (Baskent University, Ankara) skriver at: «In the 21st century, people are expected to not only be consumers, but also producing individuals». (Kaleliogu, 2015). Han mener derfor at det er det kan være nødvendig å lære elevene koding tidligere I skoleløpet enn det som er tilfellet i de fleste land i dag. Han begrunner dette med at siden barna som begynner på skolen i dag er «*digital natives*», har de lettere for å ta til seg ny kunnskap innen IKT. Han siterer videre fra Grout & Houlden, 2014 at det er nødvendig å forandre undervisningen innen teknologien fordi:

According to the developments in computer science, as the very nature of computer sciences is dynamic, flexible and innovative. Then society in general have to give sufficient importance to computer science in order to create innovators. (Kaleliogu, 2015).

I en masteroppgave skrevet av Tuomas Hiltunen i 2016, hevdes det blant annet fra utviklerne av Tynker; en plattform for programmering og undervisning, at barn bør lære programmering fordi programmering er en grunnleggende ferdighet i en digital verden, programmering kan endre verden og det trenger ikke være vanskelig å lære. (Hiltunen, 2016). Han peker også på en artikkel skrevet av Wing- Kostner i 2012, der man peker på den økonomiske fordelene med å lære barn å kode;

«The idea that has been expressed in the form of code can potentially generate value if the idea behind it is unique. » (Hiltunen, 2016) Som i Norge og Storbritannia begrunnes innføringen av IKT i skolen med utviklingen i samfunnet:

The strength of the services sector and the wide adoption of ICT reduce the labour input for routine cognitive and manual tasks; technological change also influences the organization of work towards more flexible forms, which affects demand for different levels of skills. (Finnish National Agency for Education, u.d.).

I artikkelen «*The story of Computing at School*», (Humphreys, 2014), publisert i forbindelse med CodeWeekEU2016, slås det fast at i nesten alle land i verden har man forstått viktigheten ved å lære barn koding i skolen. Forfatterne mener at det er viktig at barna ikke bare blir undervist i bruken av IT, men også hvordan IT virker, samt grunnleggende prinsipper. De hevder videre at:

Lacking such knowledge renders them powerless in the face of complex and opaque technology, disenfranchises them from making informed decisions about the digital society, and deprives our nations of a well-qualified stream of students enthusiastic and able to envision and design new digital systems. (Humphreys, 2014).

Videre skriver de om hvordan IKT fra 2014 skal være en del av det obligatoriske utdanningsforløpet i den britiske skolen. Det er interessant at de hevder at denne endringen er ikke kommet på bakgrunn av arbeid satt i gang av regjeringen, men at det er en «grasrotbevegelse» bestående av foreldre, lærere, akademikere, programmerere og andre som har jobbet hardt for å få dette til. Dette gjenspeiler også hva som har skjedd i Norge, der kodeklubber i regi av frivillige har vokst fram i mange lokalsamfunn over

hele landet. Dette arbeidet har vært med på å promotere interessen for koding blant barn og unge her hjemme.

José-Manuel Sàez-Lopez et al skriver i en artikkel at det er allmenn støtte for å innføre programmering i skolen, men at elevene ofte gir opp fordi de tenker det er for vanskelig å få til. De hevder videre at mange lærere ser på programmering kun som et fag som skal gjøre elevene i stand til å jobbe innen teknologibransjen. På denne måten blir fordelene ved å innføre koding i barn – og ungdomsskolen ignorert. (Josè-Manuel Sàez-Lòpez, 2016). Artikkelforfatterne gjennomførte en undersøkelse der 107 elever fra 5. og 6. klasse fra fem ulike skoler i Spania deltok. Studien foregikk over to år der de analyserte hvordan koding og blokkprogrammering kunne integreres i fag som kunst og håndverk og realfagene. Programmet de brukte i studien var Scratch, som er et visuelt programmeringsspråk spesielt utviklet for barn ved MIT i USA.

Resultatet av denne studien viser at:

There are positive values obtained mainly from factors such as active learning, art and history contents, computational concepts, useful and fun.” (Josè-Manuel Sàez-Lòpez, 2016). Videre skriver de at: “An active pedagogical approach using a Visual Programming Language significantly improves several elements; learning programming concepts, logic and computational practices. (Josè-Manuel Sàez-Lòpez, 2016).

4.2.2 Hva skal man lære?

Saeli et al mener at kjernen i programmering handler om problemløsning og skape et program som en løsning. De skriver videre at det er to typer kunnskap; program utvikling og forståelsen av disse programmene. Dette fordrer da at elevene bør bli opplært i: The process of problem solving, reflection on this process, and in the development of algorithmic ways of thinking». (Saeli, 2012). Med andre ord bør opplæringen i ungdomsskolen fokusere på utvikling og forståelse av programmeringen.

I 2012 ble det innført en ny læreplan i England der fokuset er at skolen skal gjøre elevene i stand til å bruke algoritmisk tankegang og kreativitet for å kunne forstå og bruke teknologi. De viktigste kjerneelementene i faget skal være informatikk, hvordan

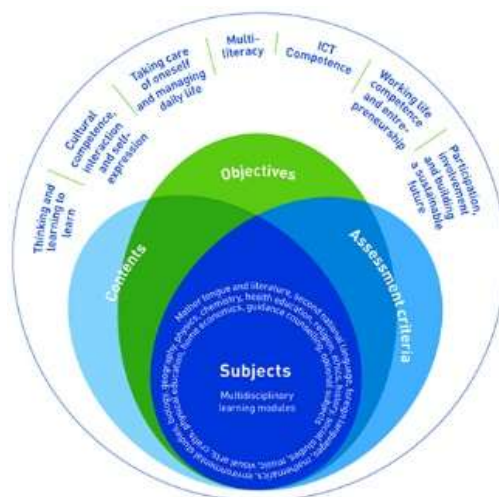
digitale systemer virker, samt kunnskap om ulike prinsipper innenfor informasjon og databehandling. Det presiseres at: «Computing has deep links with mathematics, science and design and technology, and provides insights into both natural and artificial systems». (Department for Education UK, 2013).

Videre skal faget gjøre elevene digitalt kompetente, altså de skal bli i stand til å bruke, uttrykke seg og utvikle egne ideer gjennom informasjons – og kommunikasjonsteknologi slik at de skal kunne være aktive og finne sin plass i en digital verden. Dette stemmer godt med det som Utdanningsdirektoratet her hjemme skriver under formålet for programmeringsfaget i den norske skolen.

Programmering er en viktig ferdighet i dagens samfunn, som inngår i de fleste fagområder, fra digital musikk til naturvitenskap og matematikk. Å gi elever grunnleggende ferdigheter i programmering er med på å forberede dem for fremtidige realfaglige jobber, i tillegg til å øke forståelsen for naturvitenskapelige og matematiske problemer. (Utdanningsdirektoratet, 2018).

I Finland har de fra høsten 2016 innført undervisning i programmering i skolen fra første klasse. Det blir ikke et eget fag, men skal bli implementert i andre fag, og da særlig i matematikkfaget. Vanlige lærere skal etter planen lære elevene programmering i alle finske skoler fra første til niende klasse.

The grades in which the instruction of certain subjects were also changed. For example, social studies and second national language are introduced earlier. The freedom to choose optional lessons in artistic and practical subjects is expanded to lower grades. There is also more focus on ICT skills, well-being and daily life management in all subjects. (Finnish National Agency for Education, u.d.).



Figur 6 – oversikt over fag i den finske skolen fra høsten 2016
<https://www.oph.fi/download/1022d0b2f3dd3b30aff898ac4aa61e03f9b96161.jpg>

4.2.3 Hva kan hindre læring i IKT?

Qian & Lehman mener at uansett på hvilket nivå man begynner med programmering, så er faget veldig vanskelig for elevene å få tak på. De hevder at:

«Students' learning of computer science can be understood through a lens of constructivist theory, which posits that students actively construct understanding by building on the knowledge that they already possess. » (Lehman, 2016). De mener altså at siden elevene ikke har noen referanser om hvordan en datamaskin er bygget opp eller hvordan den virker, har de ikke modeller eller referanser å henge ny kunnskap på.

Jennifer Jenson og Milena Droumeva skriver i sin rapport; «Exploring Media Literacy and Computational Thinking: A Game Maker Curriculum Study», at det er en utfordring i det 21 århundre å sikre at alle har den kunnskapen i IKT som trengs for å kunne delta i en digital kunnskapsbasert økonomi. De hevder videre at det største problemet her er at selv om det er delt entusiasme om viktigheten av IKT i skolen, er det slik at man ikke klarer å enes i hva som må læres: «Researchers and policy makers around the globe are still trying to articulate exactly what constitutes 21st century learning, while public education generally is being criticised for not adopting it». (Droumeva, 2016, Volume 14 issue 2).

En annen faktor som kan være til hinder for undervisningen er vidløftige og avanserte mål som settes for elevene. Ser vi på hovedmålene i læreplanen i England ser vi at elevene skal:

- Forstå og kunne ta i bruk de fundamentale prinsipper og konsepter fra informatikkfaget. Dette inkluderer abstraksjon, logikk og algoritmer. .
- At elevene skal kunne analysere problemer ved å bruke faguttrykk, samt gjentatte praktiske erfaringer i å skrive dataprogrammer for å kunne løse disse problemene
- Elevene skal kunne evaluere og ta i bruk informasjonsteknologi, inkludert ny og ukjent teknologi, slik at de kan løse problemer analytisk
- Elevene skal være ansvarlige, kompetente, trygge og kreative brukere av informasjon og kommunikasjonsteknologi. (Department for Education UK, 2013).

Dette er svært vidløftige mål, særlig når vi ser at to av disse målene knyttes opp mot algoritrisk tankegang og at det skal programmeres på alle trinn. Allerede fra første klasse, altså fra elevene er bare fem år gamle skal de programmere. Det er ulike mål innenfor de forskjellige «*Key stages, altså ulike trinn i skolen*», men planen er likevel ambisiøs.

Ser vi på målene i valgfaget programmering her i Norge, kan vi argumentere med at disse er like ambisiøse, særlig når vi tenker på at dette er et valgfag kun tilbudt på ungdomstrinnet i grunnskolen. Elevene kan velge et nytt valgfag hvert år, og at elevene da skal komme gjennom alle målene på kun ett år, kan synes uoverkommelig.

Saeli et al sier at man også må tenke på at elevene vil ha ulik kunnskapsnivå og ulike utfordringer, slik at man må legge opp undervisningen deretter. De siterer flere kilder som hevde at programmering er vanskelig å lære. (Saeli, 2012, ss. 79 -80). Blant annet kan lærerne gjøre det vanskelig for elevene hvis målene og ambisjonene blir for store. Det er også viktig at man varierer undervisningen slik at elevene kan ha noe å strekke seg etter. I masteroppgaven «Hva kan skolen lære av kodeklubbene», siterer forfatteren en studie der man fant: «At elever som bare jobber med Scratch og bare har Scratch – samfunnet som sitt nettverk, mister interessen og slutter ganske fort». (Bjørnevoll, Hva kan skolene lære av kodeklubbene?, 2016, s. 28). En av mine informanter forteller at når

hun skulle sette i gang med koding i klassen, så var det mange som allerede hadde jobbet med Scratch gjennom lokal kodeklubb, så hun valgte å jobbe med Micro:bit.

Jeg har jobbet med Micro: bit, og det er jo kjempespennende. Jeg vurderte Scratch også, men, i og med at såpass mange var veldig opptatt av at de ikke ville at det skulle være et barnslig opplegg og sånt, og de hadde hatt Scratch gjennom kodeklubben som er her på Kolbotn, en del av de, så valgte jeg Micro: bit. (informant 1).

Nå er det jo slik at hos oss er programmering et valgfag og vil kanskje tiltrekke seg elever som allerede er interessert i faget, men blir programmering et eget fag er det viktig at skolen da legger til rette for hele elevgruppen. Da er det viktig med håndterbare mål og variasjon i undervisningen. Utdanningsdirektoratet deler målene i faget inn i to hovedområder, modellering og koding.

Modellering: mål for opplæringen er at eleven skal kunne:

- gjøre rede for hvordan datamaskiner og programmer fungerer, inkludert et utvalg utbredte programmeringsspråk og deres bruksområder
- omgjøre problemer til konkrete delproblemer, vurdere hvilke delproblemer som lar seg løse digitalt, og utforme løsninger for disse
- dokumentere og forklare programkode gjennom å skrive hensiktsmessige kommentarer og ved å presentere egen og andres kode

Koding: mål for opplæringen er at elevene skal kunne:

- bruke flere programmeringsspråk der minst ett er tekstbasert
- Bruke grunnleggende prinsipper i programmering, slik som løkker, tester, variabler, funksjoner og enkel brukerinteraksjon
- utvikle og feilsøke programmer som løser definerte problemer, inkludert realfaglige problemstillinger og kontrollering eller simulering av fysiske objekter
- overføre løsninger til nye problemer ved å generalisere og tilpasse eksisterende programkode og algoritmer (Utdanningsdirektoratet, 2016).

Ikke bare er målene mange og vidløftige for elevene, men det kan også være en bekymring for lærerne som skal undervise faget. Har kodelærerne i grunnskolen kvalifikasjoner og erfaring nok til å kunne få elevene gjennom alle målene som er satt? I følge mine informanter er det en del lærere som har direkte utdanning i koding og programmering, men de aller fleste har ikke dette. Jeg sendte ut en enkel survey til alle 155 skolene som hadde koding høsten 2016 og jeg fikk svar fra 42 av disse, samt at jeg har hatt dybdeintervjuer med fire lærere, en fra Akershus, to fra Rogaland og en fra Hordaland. Av de 42 som svarte på surveyen var det 16 som hadde utdanning i koding, mens de resterende 26 ikke hadde dette. Av de fire lærerne jeg intervjuet hadde to av dem noe utdanning innen koding, men de to andre kun hadde noen kurs og lignende.

Jeg er utdannet faglærer i kunst og håndverk og master med tema design og forbruk. Jeg har noen kurs i koding fra før som ikke er fra universitet eller høyskolen, men fra privatpersoner. Jeg tok over en etablert kodeundervisning av en kollega eller fra en kollega som skal slutte og han, vel han, skal jeg si, hadde laget sitt opplegg, men jeg har vært på oppfølging med Utdanningsdirektoratet og de kursene, også de samlingene de har. Også har jeg han tidligere kodelæreren jeg kan spørre om ting, samt at mannen min er programmerer noe som hjelper litte grann, hehe. (informant 1).

Kodemessig så har eg et kurs på videregående en gang og det var i gamle QBasic så eg måtte skrive 30 sider tekst i QBasic og det var veldig gøy. Men noen av språkene er det samme, den har ikke utvikla seg så veldig, men passer for det eg er vant til å jobbe med data. Derfor gjekk det kjapt å lære koding. (informant 2).

Før det så har jeg vært utdannet telemontør. Så det har jeg fagbrev i. I skolen så er det realfag som er mitt felt. Og så har jeg 60 studiepoeng i IT fra HSH på Stord. Men, vi programmerte jo veldig mye i jobben som telemontør. Fordi vi satte opp telefonanlegg og dataanlegg i bedrifter og privat. Da kodet vi jo hvem som skulle ha samtale først. Hvem de skulle sette over til og så videre. Så koding har det egentlig vært mye av i yrket mitt tidligere, men ikke på den måten som jeg har i dag. Det er mer, hva skal jeg si, praktisk. (informant 3).

Eg har universitetsutdanning med matematikk, kjemi og geografi. Og så har eg eit kurs i programmering på universitetet fra 1973. Ja, så da lærte vi programmering og tankegangen bak programmering, så det var starten på det. Det var forkant fire med hullkort og en UNI vak 1110 på Universitetet i Bergen. Det var ikkje skjerm, vi spilte, spilte det første dataspillet mitt, som heitte «Moonlander», der du ga tekstkommandoer til maskinen og den svarte på papirrull. Sånn at spillet foregikk med tastaturet også svarer maskinen på papirrull. (informant 4).

Kun 15 av de 42 lærerne som svarte på surveyen hadde fått tilbud om etterutdanning i programmering og av informantene fra dybdeintervjuene har tre av dem fått tilbud om etterutdanning i ulike former.

Og så har jeg kommet inn på NTNU på nettstudier, så jeg skal gå der fra høsten av. Det er deltid og da jobber jeg 100% i tillegg, Men jeg får stipend og det er nettbasert utdanning i moduler så vidt jeg har forstått det, med oppgaver som skal leveres med jevne mellomrom. (informant 1).

Nei, eg har ikkje fått noko tilbud om det. Når eg blei med på programmering som valfag som eit pilotprosjekt så fikk eg tilbud på ein sånn MOOC, men det fikk alle tilbud på, men det er ikkje studiepoeng knytt til dette, men akkurat no er det ikkje i koding eg ønskjer etterutdanning i, men eg føler litt meir innanfor didaktikk og det har eg ikkje sett nokon som tilbyr det enno. (informant 2).

Ja, det har jeg, så.. Ikke nå. Men jeg har lyst. På sikt. Nei, jeg ville hvert fall ta matematikk for jeg mangler 30 studiepoeng i forhold til kravene som kommer. Så det ønsker jeg. Og så har jeg og vurdert, for det har kommet et nytt fag på HSH som heter IT, som jeg og har vurdert da. Men, jeg må først på en måte gjøre ferdig matten, for da har jeg på en måte undervisning i de fagene jeg skal ha. Så da får vi se hva vi har lyst til. (informant 3).

Ja, nå har dei jo lagt ut desse sentrale oppgåvene på programmering, så eg har vore inne og gjort litt på nokre av dei. Med det eg ser er at, det viktigaste her er jo det når ein får dette utstyret, så er det rett og slett det å eksperimentere litt sjølv. Altså eg har jo har jo heldt på med Arduino, der har eg kunne spurt sonen min ein del og så har

eg prøvd meg fram. Og på samme måten med den der Mikrobitten, der ligger det jo veldig mye bra på nettet. Og så har vi jobba med Scratch, og der og er det jo mye bra nettressurser. Det likør elevene veldig godt. (informant 4).

Av de 15 som rapporterte positivt om etterutdanning fra surveyen, var det ulike typer etterutdanning som var tilbudt; MOOC, 15 studiepoeng innen IKT fra HVL, 30 studiepoeng i programmering for lærere ved NTNU, 60 studiepoeng som lærerspesialist ved UIO og annen type utdanning innen matematikk og naturfag. Noen har også fått tilbud om små kurs gjennom lokale kodeklubber

4.2.4 Hvordan skal man undervise i programmering?

På spørsmålet om hva skal læres, altså hvilke kjerneoppgaver innenfor programmering som elevene trenger å lære, sier Saeli et al at det er ofte de som bestemmer hvilken læreplan som skal gjelde for skolene som bestemmer dette. Likevel er det ofte slik at de som fastsetter en læreplan vil overlate metode og innhold til den enkelte lærer så lenge kunnskapsmålene som er satt følges. Videre hevder Saeli et al at: «Among secondary teachers there seems to be heterogeneity in the choice of programming languages/paradigms». (Saeli, 2012).

I en artikkel publisert av Gabor og Arki hevder forfatterne at etter deres egen personlige erfaring i å undervise programmering, viser det seg at elevene ikke klarer å oppnå algoritmisk tenking ved å bruke tradisjonelle opplæringsmetoder. Forfatterne fant at elever som jobbet med koding og programmering oppnår bedre resultater ved å benytte seg av spillorienterte oppgaver sammenlignet med tradisjonell opplæring i ulike dataspråk. I denne studien fulgte de to grupper; der den ene gruppen jobbet med programmering og koding ved tradisjonelle metoder, og den andre gruppen ble undervist ved å bruke spillbasert undervisning. De fant at elevene som fikk spillbasert undervisning oppnådde i snitt en hel karakter bedre enn den andre gruppen ved sluttevalueringen i faget.

«The traditional way of teaching programming was not so successful than the game-based examples. The students were more motivated to write game programs and more students would pass the test». (Arki, 2016).

I en studie fra Marokko viser det seg også at elevene lærer bedre ved å bruke blokkbasert programmeringsmetoder enn tradisjonelle metoder (bokstavbaserte programmeringsspråk). Forfatterne av denne studien jobbet med en gruppe på 69 realfagselever. Gruppen ble delt i tre mindre grupper, der en gruppe jobbet med å lage spill der de brukte Scratch og de andre to gruppene ble undervist ved å bruke programmeringsspråket Pascal. Etter endt prosjekt fant de at:

Using an environment for learning programming such as Scratch highly motivate students and empower them to pursue their studies in programming. In fact, when learners were asked about their desire to continue their studies in programming, 65% of students who experienced using Scratch considered continuing, whereas only 10,3% of the students who used a standard programming environment showed some interest. (Ibrahim Ouahbi, 2015).

José-Manuel Sàez-Lopez et al, mener også at det å bruke blokkbaserte programmeringsspråk, er med på å gjøre elevene mer interesserte i koding. De jobbet med et prosjekt der de introduserte Scratch i undervisningen over en periode på to år og de oppsummerer studien slik:

The values obtained in the study show statistically improvements in the understanding of computational concepts and computational practices in this educational stage, which suggests recommending to the educational authorities to implement programming in educational settings. (José-Manuel Sàez-López, 2016).

Dette er en interessant studie, da den ser på fordeler ved å introdusere kodefaget tidligere i skoleløpet enn det vi tenker her hjemme. Samtidig bekrefter arbeidet noe av det samme som Hiltunen fant ut i sin studie fra Finland. I følge de fleste av mine informanter stemmer dette bra. Nesten alle sier at de hovedsakelig bruker Scratch, Micro: bit eller andre blokkprogrammer i undervisningen, da dette er enkle språk som

nybegynnere innenfor programmering lett klarer å lære seg, samt at de motiverer til videre læring.

Det er derfor eg jobbar, for å seie det sånn eg jobbar mykje med visuelt basert koding. Det finnest fleire av desse, Blockie er ein, Scratch er ein annan. Det er sånn klosskoding. Eg har lyst til at dei skal vite kva desse klossane betyr og kva som funker og kva som ikkje funker. Og i min erfaring, eg har satsa mest på dette. (informant 2).

Jeg kan gjerne ta. Vi har hatt mange vellykkete, holdt jeg på å si som var veldig kjekke. Vi hadde en koding i Scratch der vi skulle lage julekort til jul. Og da skulle vi lage et animert julekort. Og det var veldig kjekt. (informant 3).

Og på samme måten med den der Mikroboten, der ligger det jo veldig mye bra på nettet. Og så har vi jobba med Scratch, og der og er det jo mye bra nettressurser. Det liker elevene veldig godt. (informant 4).

En av informantene svarer slik:

Jeg har jobbet med Micro: bit, og det er jo kjempespennende. Jeg vurderte Scratch også, men, i og med at såpass mange var veldig opptatt av at de ikke ville at det skulle være et barnslig opplegg og sånt, og de hadde hatt Scratch gjennom kodeklubben som er her på Kolbotn, en del av de, så valgte jeg Micro: bit. For det er jo samme puslespilløsning. De aller fleste får det til på en god måte. (Informant 1).

5. Teori⁴

“In theory there is no difference between theory and practice. In practice there is.”

(Yogi Berra, amerikansk atlet 1925-2015)⁵

Det er hverken et mål eller plass til å skrive om alle relevante læringsteorier som kan passe innenfor forskningsspørsmålene denne oppgaven stiller. Det finnes omfattende

⁴ Dette kapittelet er omskrevet og bearbeidet fra egen Forskningskisse.

⁵ https://www.brainyquote.com/quotes/yogi_berra_141506?src=t_theory

litteratur som omhandler hvordan mennesket søker ny kunnskap, samtidig som det er verdt å merke seg at mange teorier overlapper noe. Derfor vil det være hensiktsmessig å forholde seg til noen få relevante teorier i denne oppgaven.

Interessen for å skjønne hvordan man kan forbedre hvordan vi tar til oss kunnskap er stor, da læring har en viktig posisjon i kulturen vår. Læring er også knyttet opp til forestillinger om «økonomisk og sosial utvikling og til et ønske om å forbedre våre livsvilkår» (Säljö, 2001). Säljö peker vider på at de fleste i dag er opptatt av hvordan barn og unge skal skaffe seg kunnskap og utdanning som gjør at de er rustet til en fremtid som blir mer og mer teknologisk. Uansett hvor mange teorier som utvikles om hvordan mennesket tar til seg kunnskap, er det vanskelig å tenke seg at det kommer en fasit. Den teknologiske utviklingen og vårt stadig mer globale samfunn gjør at kunnskap og læring ikke er statisk, men er hele tiden i bevegelse og utvikling.

Mennesket er en lærende art, man kan si at læring er en av våre viktige trekk. Vi kan ta vare på erfaringer og senere ta disse frem i passende situasjoner. Vi lærer både sammen med andre og individuelt.

Krumsvik sier at «rett teori til rett empiri er viktig» (Krumsvik, Forskningsdesign og kvalitativ metode, ei innføring, 2014). Det finnes mange ulike definisjoner på hva en teori er. Krumsvik nevner flere av disse i teorikapitlet i boken som det er referert til over. «Ordet teori kan beskrive et system av almene prinsipper eller regler som gjelder for, eller danner grunnlaget for en virksomhet; brukes ofte i motsetning til praksis» (Krumsvik, Forskningsdesign og kvalitativ metode, ei innføring, 2014).

Krumsvik understreker at teoriene en bruker i forskningen, må være teoretisk robuste og at disse teoriene kan være med på å støtte og underbygge det man ønsker å forske på. Det er også viktig å innse at man ikke «finner» opp nye teorier til et hvert nytt studie. Krumsvik sier at: «Vi står på skuldrene til andre kvalitative forskere før oss og bygger videre på de teoretiske linsene deres» (Krumsvik, Forskningsdesign og kvalitativ metode, ei innføring, 2014).

Sett i lys av problemstillingen og forskningsspørsmålene er det naturlig å fokusere på et konstruktivistisk og sosiokulturelt paradigme. Dette passer inn med tidligere forskning

som bekrefter at læring skjer ved at elevene er aktive, og at arbeid sammen med andre ofte bidrar til utvikling av sosiale ferdigheter. Konstruktivismen er dessuten et moderne paradigme, og siden digitale ferdigheter er et relativt nytt krav i skolen kan det tenkes at konstruktivismen er et naturlig valg i denne oppgaven. Koding i skolen vil foregå i en klassesituasjon og det er derfor sannsynlig at elevene vil jobbe sammen. Fagets natur fordrer også at elevene er aktive i egen læring sammen med lærer. Disse tankene kan også overføres til voksne lærere som skal lære seg ny kunnskap: Det kan tenkes at voksne mennesker som allerede har en god del utdanning bak seg, lærer på en annen måte enn uerfarne barn, men prinsippene vil forbli de samme. Vi lærer ved å være aktive, både sammen med andre og i egen læring. Det kan også være grunn til å kort å nevne positivismen og pragmatismen i denne oppgaven, da en liten del av forskningen i oppgaven vil være kvantitativ.

5.1 Konstruktivismen

På slutten av 1900 – tallet begynte man å tenke på hva som egentlig var sikker kunnskap. Postmodernistene mente at det ikke fantes en objektiv virkelighet som kunne måles ved vitenskapelige metoder. De hevdet at det postmodernistiske mennesket konstruerte egen kunnskap ut fra egne erfaringer (Solerød, Pedagogiske grunntanker, 2012) Jean Piaget (1896 – 1980) var en av pioneren innenfor konstruktivismen. Han var biolog, men fikk senere jobb som assistent ved et institutt i Paris som drev med intelligensmåling. Det var her han begynte å interessere seg for kognitiv utvikling, Han mente at intelligenstesting ikke var veien å gå, og han mente at samtaler med elevene for å finne ut hvordan de resonerte og tenkte var mer hensiktsmessig, Piaget var med andre ord opptatt av «*kunnskapens opprinnelse og utvikling*» (Solerød, Pedagogiske grunntanker - i et dannelsesperspektiv, 2012).

Læringsteorien konstruktivisme bygger på hvordan mennesket tilegner seg kunnskap. Et viktig element innenfor denne teorien er at mennesket konstruerer egen kunnskap gjennom å utføre handlinger som resulterer i læring. Innenfor skolen tenker man seg tilpasset undervisning til elevenes mestring – og utviklingsnivå. Dette vil også kunne gjelde lærere som skal tilegne seg ny kunnskap. Denne undervisningen vil være støttet av mentor/lærer, men vil i hovedsak være styrt av eleven/studenten selv. Innenfor

konstruktivismens rammer ses mennesket på som aktivt, ansvarlig og utøvende. I motsetning til behaviorismen som legger vekt på ytre stimuli, oppfattes konstruktivistisk syn på læring en retning der elevene/studentene tolker mottatt informasjon og knytter det så opp mot kunnskap de allerede besitter. Evnen i å tenke og forme nye begreper skjer der de lærende selv er aktive, i motsetning til at de passivt mottar instruksjoner. (Dysthe, Teoretiske perspektiver på dialog og dialogbasert undervisning, 2012). Både Vygotsky og Bruner teorier om barns læring og utvikling står svært sentralt innenfor konstruktivismen.

I en artikkel der man analyserte 27 studier om hvordan programmering har blitt implementert i skolen, fant forfatterne ut at når man brukte blokkprogrammer som Scratch og Logo, var disse både: «low floor; easy for the students to pick up as well as high – ceiling; allowing students to create more sophisticated programs»(Sze Yee Lye, 2014). Disse studiene innbefattet barn fra 5 til 14 år, i ulike utdanningsinstitusjoner, ulik tidsbruk og i ulike fag. Artikkelen konkluderer med at de fleste studiene de hadde sett på rapporterte positive resultater ved bruk av visuelle programmeringsspråk og undervisning som lot elevene å lage digitale fortellinger for å befestet hva de hadde lært, altså ved å implementere en konstruktivistisk læringsteori.

5.2 Konstruksjonisme og det sosiokulturelle synet på læring i den moderne skolen.

Vi mennesker lærer når vi forsøker å oppnå mål som er ønsket, både av samfunnet, skolen og av oss selv. Vi forsøker hele tiden å lære nye ting, men i starten av læringen vet vi ikke alltid hvordan vi skal få det til. Vi har ikke forstått hvordan vi skal komme i mål. Vi prøver, eksperimenterer, feiler og prøver igjen til vi oppnår det resultatet vi ønsker. Fokuset er mer på prosessen og samhandlingen som foregår mens man jobber, enn selve resultatet. Mange vil kanskje si at dette ikke stemmer helt med hvordan undervisning bør foregå, fordi man gjerne tenker at feil svar betyr dårlig resultat, altså en tenking der produktet er mer viktig enn selve læringsprosessen. Siden vi som regel jobber sammen med andre, utvikler vi også visse sosiale ferdigheter som gjør at vi kan begynne å forstå hvordan andre mennesker jobber og tenker. Dette læringsynet bygger i stor grad på Vygotsky sine teorier. Her vektlegges det at vi mennesker ikke lærer i et

vakuum, men at læringen forgår i en sosial kontekst, for eksempel i et klasserom sammen med andre elever. Mennesker gjør mange erfaringer i løpet av livet og blir naturlig nok preget av det som oppleves. Læring handler om erfare og ta del i ny kunnskap og nye ferdigheter, samt evnen til å bruke disse på en måte som produserer ny kunnskap man kan ta med seg videre i livet.

I et sosiokulturelt perspektiv fungerer psykologiske og fysiske redskaper som strukturerende ressurser som gjør det mulig for deltakere i sosiale praksiser å tolke og handle kompetent i nye situasjoner. (...) Resultatet av interaksjon er at vi forandres som individer; det gjelder både vårt intellektuelle og kommunikative repertoar og vår måte å beherske fysiske redskaper på. Vi øker vår evne til å forstå hvordan sosiale aktiviteter er strukturer og hva de innebærer; det er ingenting i det «ytre» som kommer inn i det «indre». (Säljö, Læring i praksis, et sosiokulturelt perspektiv, 2003).

Hovedbudskapet her er at elevenes læring må ses i sammenheng med kultur, språk og fellesskap. Læring skjer ikke bare i skolen, men overalt og hele tiden. Læring skjer oftest når man er en del av et fellesskap og har et felles språk.

Bruk av datamaskiner i skolene startet allerede på slutten av 1960 – tallet og med utviklingen av mindre datamaskiner på slutten av 1970 – tallet, ble undervisning med bruk av datamaskiner mer vanlig i skolen. Men datamaskinene ble da brukt mer til pugging og det man kalte Computer Aided Instruction (CAI). Dette betød at datamaskinene ble brukt på to måter; enten som en ren kilde for informasjon om et tema eller som erstatning for læreren ved at elevene skulle testes på forståelse av ulike tema. Med andre ord når man trengte å drille elevene i et emne, kunne man frigjøre læreren slik at han/hun kunne bruke tiden på elever som trengte ekstra hjelp. En annet positivt aspekt ved å bruke datamaskinen på denne måten var at elevene slapp å svare i plenum, og dermed unngå at hele klassen får med seg at de svarer feil eller at de bruker lengre tid på oppgavene enn de andre. (The Editors of Encyclopaedia Britannica, 2017). Konstruktivistene var kritiske til CAI da ikke bare blir det menneskelige aspektet redusert, men også fordi man hadde tilgang til et verktøy der elevene selv kunne være kreative og at man ikke benyttet seg av dette. Her hadde man muligheter til å la elevene konstruere, lære seg problemløsning og algoritmisk tenkning. Seymour Papert sier om CAI at:

One might say that a computer is being used to program the child. In my vision, the child programs the computer and in doing so, both acquires a sense of mastery over a piece of the most modern and powerful technology and establishes an intimate contact with some of the deepest ideas from science from mathematics and from the art of intellectual model building. (Papert, 1993).

Seymour Papert var en av de mest sentrale skikkelsene i den bevegelsen som så på mulighetene for å lære barn/elever programmering. Seymour Papert var født i Sør-Afrika, men jobbet ved MIT i USA, og var en av pionerene innenfor AI (artifisiell intelligens) og konstruksjonismen. Konstruksjonismen var basert på Piaget og hans konstruktivistiske læringsteorier og Papert jobbet sammen med Piaget i flere år i Genève. (MIT News, 2016).

Han mente at ved å lære barn programmering fikk de muligheten til å være kreative, de ville bli mer engasjert i det de skulle lære og de fikk bruke egen intelligens på en annen måte enn ved tradisjonell undervisning. Ved et besøk ved en skole der han skulle lære elever å jobbe med matematikk i LOGO, som er et dataprogram Papert var med å utvikle, fikk han se noen elever som jobbet med et prosjekt i kunst og håndverk. Elevene skulle lage skulpturer av såpe, og han ble opptatt av hvordan elevene jobbet og hvor forskjellig dette var fra hvordan de jobbet i matematikkfaget.

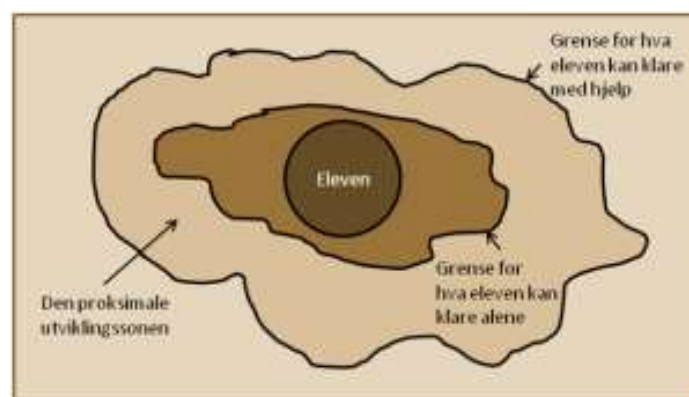
For a while, I dropped in periodically to watch students working on soap sculptures and mused about ways in which this was not like a math class. In the math, class students are generally given little problems, which they solve or do not solve pretty well on the fly. In this particular art class, they were all carving soap, but what each students carved came from wherever fancy they had, and the project was not done and dropped, but continued for many weeks. It allowed time to think, to dream, to gaze, to get a new idea and try it and drop it or persist, time to talk, to see other people's work and their reaction to yours--not unlike mathematics as it is for the mathematician, but quite unlike math as it is in junior high school. (Seymour Papert, 1991).

Papert mente at dette var en hensiktsmessig måte å jobbe på, og fikk ambisjoner om at man kunne jobbe med matematikkundervisningen i ungdomsskolen på denne måten. Papert og hans syn på at elevene kunne utvikle kognitive ferdigheter og refleksjonsinnsikt når de lærte seg programmering, ble møtt med en del skepsis. Noen

av disse skeptikerne mente at de ikke var noen bevis for at de elevene som lærte å programmere, utviklet evner utover det å programmere (Kurland, 1984). Papert mente imidlertid at kritikerne hadde misforstått han i det at det ikke var maskinene som skulle stå for revolusjonen i skolen, men at det var menneskene som forstod teknologien de jobbet med som skulle gjøre undervisningen bedre på sikt. Det var viktig at man forstod at det er hvordan mennesket kan endre skole ved hjelp av teknologi, heller enn å tenke at teknologien skulle endre skolen i seg selv.

5.3 Sosiokulturell teori

Olga Dysthe skriver at det ikke finnes en «sosiokulturell læringsteori», men at det heller er ulike retninger innenfor dette begrepet. Videre skriver hun at læringsteori er avhengig av hvilken synsvinkel en velger for å se på læring. (Dysthe, Dialog, samspel og læring, 2001). Den sosiokulturelle teorien handler om læring som deltager i sosiale praksiser. I følge Vygotsky er sosial samhandling utgangspunktet for læring. Elevene har bedre muligheter til å nå eget læringspotensial når man lærer sammen med andre, enn når man studerer alene. Når man jobber sammen har elevene muligheter å ta til seg og overta kunnskap fra medelever (Säljö, Læring i praksis, et sosiokulturelt perspektiv, 2003). Videre benytter Vygotsky begrepet den *proksimale utviklingszone* for å beskrive avstanden mellom det elevene kan få til på egen hånd, og de kan oppnå med veiledning og støtte fra en lærer eller i samarbeid med mer kompetente medelever. Denne modellen kan sees ikke bare fra elevens ståsted, men også lærere som skal ta innover seg ny kunnskap, slik som mange av lærerne innbefattet i studiet.



Figur 7. Illustrasjon som viser den proksimale utviklingssonen: <https://kennethhorpestad.wordpress.com/2013/10/03/vygotskys-lærings-og-motivasjonsteori-og-konsekvenser-for-undervisning-i-kroppsoving/>

Den sosiokulturelle læringsteorien mener da at utviklingen skjer fra det sosiale til det kulturelle. Den viser hva elevene kan få til alene og hva elevene kan klare sammen med noen som besitter mer kunnskap enn dem selv. Elevene må da arbeide aktivt og få støtte fra andre for å klare å løse nye oppgaver. Etter dette prinsippet er det viktig at undervisningen legges opp slik at elevene har noe å strekke seg etter, men samtidig skal ikke oppgavene være av slik art at de blir umulige å få til. Man tenker seg at den proksimale sonen ikke er fastsatt hos hver elev, men at denne varierer med hvilke evner den enkelte elev og lærer måtte ha innenfor ulike fag. Hvilke arbeidsmetoder som benyttes vil også ha en viss betydning for dette. (Dysthe, Dialog, samspel og læring, 2001). Informantene jeg har intervjuet bekrefter dette med at elevene jobber godt sammen med andre og at det kan være lettere å finne løsninger sammen. Informantene sier det slik:

Men etter det så har vi i hovedsakelig jobbet med individuelle prosjekter, men jeg har delt inn klassen sånn at de har hatt læringspartnere. De har alltid sittet ved siden av noen som enten, altså det er ikke sånn at de som er veldig gode sitter i et hjørne liksom, jeg har fordelt dem jevnt over i klasserommet slik at de har muligheter til å formidle kunnskapen videre til noen og de har muligheten til å spørre hvis de trenger det, og det synes å fungere godt for da ser jeg at de som kan det de holder på med at de blir enda bedre fordi de må forklare og må nesten gå inn i teorien og se på hvorfor skriver jeg koden på den måten egentlig, mens de som ikke kan så mye de får mer motivasjon av å sitte sammen noen som kan opplegget da. (informant 1).

Og så gi dei ei oppgåve som testar dette på forskjellig måtar med variert, liksom ulik måloppnåing. Det skal ikkje vere ja eller nei, men greidde det litt eller greidde det med middels eller mot høg måloppnåing. Men oppgåvene skal alltid vere, mine er alltid, det er meir «*learning by doing*». Alt blir delt. Med resten av klassen. Innleveringer er på eit opent forum, der alle skal gå inn og sjå kva dei andre har laga og gi tilbakemelding på dei. (informant 2).

Og så samarbeider de. Så de sitter i grupper på to eller fire. Litt etter hva type ting de jobber med og da skal de spørre gruppa. Så da har jeg alltid en flink som jeg vet kan ta litt ansvar og som er god å samarbeide. Og så har jeg litt forskjellige nivå ned igjennom. Og da kan de og hjelpe hverandre. (informant 3).

Ibrahim Ouahbi et al skriver at de fant positive reaksjoner fra elevene som jobbet sammen når de skulle programmere. De fant at elevene var interesserte i hva de andre elevene hadde jobbet med og at de var ivrige etter å dele med hverandre. En av elevene sa det slik: «*This is great; we are really playing and learning*». (Ibrahim Ouahbi, 2015). Andre elever som var med i studien presenterte arbeidene sine til venner samt andre elever på skolen. Dette var elever i gruppa som jobbet med Scratch som introduksjon til programmering.

5.4 Vurdering for læring

Vurdering i fag har som formål å:

- Bidra til læring gjennom underveisvurdering
- Gi informasjon om kompetansen til eleven underveis i opplæringen
- Gi informasjon om kompetansen ved avslutninga av opplæringa i faget

På engelsk brukes ofte begrepet «*Class Room assessment*», altså vurdering i klasserommet. Det var forskning utført av Black og William som viste at vurdering kan være med på å fremme læring om elevene blir involvert i vurderingsprosessen. (Black, 1998). Forskningen dette bygger på er blant annet en studie som ble gjort i England og baserer seg på internasjonale forskningsprosjekter som ser på hvordan vurdering kan ha en læringsfremmende effekt. (Black, 1998). Vurderingen skal gi god tilbakemelding og dermed hjelpe elevene til videre læring i faget. Det er også vurdering i orden og adferd. Dette skal være med på å bidra i elevenes sosialisering og sikre at elevene har et godt psykososialt miljø, samt å gi informasjon om elevenes orden og adferd på skolen. (Utdanningsdirektoratet, 2017).

Begrepene evaluering og vurdering blir ofte brukt om det samme. Betydningen av disse begrepene vil avhenge av hvilket språk du bruker og hvilket land du kommer fra. I Norge bruker vi begrepet vurdering (assessment på engelsk), men de siste årene er det blitt vanlig å skille mellom begrepene. Slemmen siterer Bloom (jf. Bloom i Lysne 1999:308) som definerer vurdering som en bedømmelse av elevenes utbytte av den undervisningen og de læreraktivitetene som elevene er en del av. (Slemmen, 2010). Altså, vurdering er en prosess der lærere samler inn informasjon om kompetansen of

ferdighetene til elevene de underviser. Evaluering, (evaluation på engelsk), betyr ifølge Slemmen det å bedømme verdien eller kvaliteten på noe, som for eksempel kan være en læreplan eller en institusjon. (Slemmen, 2010).

Utdanningsdirektoratet satte i gang en storstilt satsing i 2010; Vurdering for læring. Det overordnede målet for satsingen var at skoleeiere, skoler og læringsbedrifter skulle videreutvikle en vurderingskultur og en vurderingspraksis som har læring som mål. (Utdanningsdirektoratet, 2015). Man kom fram til fire hovedprinsipper som sees på som sentrale for å få til læringsfremmende undervisvurdering. Disse prinsippene er basert på forskning og er nå en del av opplæringsloven, og det ble slått fast at alle elever og lærlinger har rett på undervisvurdering under utdanningsforløpet.

Disse fire prinsippene er som følger:

1. Elevene og lærlingene skal forstå hva de skal lære og hva som er forventet av dem.
2. Elevene og lærlingene skal få tilbakemeldinger som forteller dem om kvaliteten på arbeidet eller prestasjonen.
3. Elevene og lærlingene skal få råd om hvordan de kan forbedre seg.
4. Elevene og lærlingene skal være involvert i eget læringsarbeid ved blant annet å vurdere eget arbeid og utvikling.

I Trude Slemmens bok «*Vurdering for læring i klasserommet*» presiserer forfatteren at dette handler om vurdering FOR læring, altså vurderingsprosessene skal ha et læringsfremmende formål. (Slemmen, 2010). Det motsatte, vurdering AV læring betyr at du som lærer gir en vurdering av hva eleven kan i et emne der og da, for eksempel ved en eksamen. Resultatene kan ikke endres og elevene må bruke den kunnskapen de besitter akkurat den dagen. På denne måten har læreren to ulike roller – «Coach» eller «dommer». (Slemmen, 2010).

Boken tar utgangspunkt i det sosiokulturelle perspektivet på læring, noe som passer godt inn med arbeidet i denne oppgaven. Slemmen presiserer at:

Vurdering FOR læring handler ikke om at hele skolen skal ha avanserte planer for hvordan vurdering skal skriftliggjøres og dokumenteres til en hver tid for hver enkelt elev. Vurdering FOR læring handler om de prosessene som skjer i klasserommet. Det handler om at du som lærer innhenter informasjon om elevenes kompetanse og ferdigheter slik at du kan tilpasse opplæringen og involvere elevene i deres egen læring. (Slemmen, 2010).

Det å gi tilbakemeldinger til elevene er vanlig og virkningsfull måte lærere bruker når de underviser. Virkningen av tilbakemeldinger til elevene kan likevel være svært varierte. I følge John Hattie er det viktig at vi forstår avstanden av hvor elevens kunnskap er nå og hvor elevens kunnskap skal være. (Hattie, *Visible Learning for Teachers - Maximizing Impact on Learning*, 2012).

Hattie sier at:

To make feedback effective, therefore, teachers must have a good understanding of where the students are, and where they are meant to be – and the more transparent they make this status for the students; the more students can help to get themselves from the points at which they are to the success points, and thus enjoy the fruits of feedback. (Hattie, *Visible Learning for Teachers - Maximizing Impact on Learning*, 2012). Videre skrive Hattie at vurdering kan hjelpe eleven slik at han/hun kan få et bedre fokus og vise vei til hvordan eleven kan komme fram til løsninger, vurderingen kan vise hvor elevene har misforstått oppgaven og vurderingen kan samtidig motivere eleven til å jobbe mer målrettet for å løse oppgavene. (Hattie, *Visible Learning for Teachers - Maximizing Impact on Learning*, 2012).

Hattie lister opp tre viktige tilbakemeldings spørsmål som er viktige for lærerne å bruke i samtaler med elevene; hva skal de lære, hvordan skal de lære dette og hva skal de lære etterpå. (Hattie, *Visible Learning for Teachers - Maximizing Impact on Learning*, 2012) Det første spørsmålet viser til mål – altså hva skal jeg lære. Her er det viktig at læreren har et mål med læringen og at han/hun kan formidle dette til elevene slik at de skjønner hva som skal læres. Det er mange elever som synes dette er vanskelig, da de tenker at det er resultatet som er viktig og ikke forståelsen av veien og innholdet.

Dette bekreftes av en av mine informanter som sier det slik:

Jeg har instruert de, eller aller først var det en veldig mye veiledning og jeg følte at løp fram og tilbake og jeg fikk ikke gjort annet enn å løpe fram og tilbake i timene, men så endra jeg strategi og la opp en sånn – dette er det en programmerer gjør, først planlegger også skriver koden feilsøke osv. og da ble det liksom det blir tydelig for elevene når de skal jobbe med dette her eller hvis du skal gå videre så er du nødt til å finne ut av grunn problemene sine selv, de må bruke internett, alle svarene du trenger, alle svarene du vil ha ligger på internett på etter eller annet punkt. (informant 1).

Med andre ord, da elevene fikk oppgaven var de mer opptatt av å komme til målet enn å forstå hva de skulle gjøre og hvorfor. Etter hvert forandret læreren strategi, og elevene begynte å jobbe på en annen måte.

Det andre spørsmålet Hattie stiller er: «*Hvordan skal jeg komme dit (til målet)?*». Her viser han til at det er viktig med vurdering på hvor langt elevene er kommet med oppgaven sett i lys av kriteriene og ikke sammenligne elevene med hverandre, men heller gi tilbakemelding på egen innsats. Hattie skriver at:

«Providing feedback that moves learners forward; encouraging students to see themselves as the owners of their own learning; and activating students as instructional resources for one another». (Hattie, *Visible Learning for Teachers - Maximizing Impact on Learning*, 2012).

Informant 2 sier det slik:

Ja, det er mykje prøving og feiling. I alle fall i dei to åra eg har hatt koding. Det kommer eit punkt der de byrjar å snakke med kvarandre og retter opp neven og spør meg og kvarandre, men dette kjem litt seinare. Heldigvis og, det kan diskuterast om det er bra eller ikkje, men eg likar dei, at det er mykje prøving og feiling før de går og hentar svaret hos en annen elev. Ja, så først er det, det er meir feiling hos nokon, men det er læring heile vegen. (informant 2).

Og så må du være kjempekonkret, hvis ikke blir det bare tull. Så du må være veldig konkret. “Hva er det dere skal lage nå?”. Og så har jeg det på ark, og så huker de av når de har gjort oppgave A, og så huker de av når de har gjort oppgave B. Og så gir

de ett vink, og så kommer jeg og så tester jeg programmet de holder på med til enhver tid. Og så samarbeider de. Så de sitter i grupper på to eller fire. Litt etter hva type ting de jobber med og da skal de spørre gruppa. Så da har jeg alltid en flink som jeg vet kan ta litt ansvar og som er god å samarbeide. Og så har jeg litt forskjellige nivå ned igjennom. Og da kan de og hjelpe hverandre. (informant 3).

Informant 4 sier det slik:

Det er derfor de sitter to stykker, og så spør de hverandre og så er jeg rundt hele veien. Så det er ganske travelt. Det er det. For.. Og det er kanskje det å fokusere litt på det av og til er det litt lite fokusert på at koding i for så vidt ikke selve kodingen. Kodingen er tankegangen som ligger før du begynner å kode. Og det prøver vi å snakke om. At du må på en måte, nå må du få en ide først. Akkurat som han der teite svensken som fant på at han skulle lage noe med noen firkanta og noe kuber på skjermen. Høres jo helt dumt ut, nå er han jo steinrik. (informant 4).

Det tredje spørsmålet som Hattie tar for seg er: «*Hvor skal jeg nå?*». Hattie hevder at dette er det spørsmålet som er viktigst for elevene, og målet her blir å vise dem svaret på hvor og hvordan de skal komme seg videre til neste mål, men også å lære elevene å finne egne svar på dette spørsmålet. (Hattie, Visible Learning for Teachers - Maximizing Impact on Learning, 2012). Hattie hevder at elevene vil være bedre i stand til å velge passende oppgaver og at de lærer ulike strategier for å komme til målet.

Altså, har de et spørsmål så må de Google det. Og så fikk de noen nettsider av meg å gå inn å lese på og sånn. Så da etter det så ble det bedre i forhold til løping veiledning, og de de fikk rett og slett ikke lov til å be om veiledning før de hadde søkt minst en gang fordi da vet jeg at de de må prøve å være løsningsorienterte selv også. (informant 1).

Eg begynner ofte med å vise ett eller to begrep. Nå skal me gå inn på kva x og y koordinatar er, nå skal vi gå inn på kva ein variabel er, no skal vi gå inn på korleis me kan få ulike input og output og vise korleis dei skal gjere det praktisk, felles på skjermen. Og så gi dei ei oppgåve som testar dette på forskjellig måtar med variert, liksom ulike måloppnåing. Det skal ikkje vere ja eller nei, men greidde det litt eller

greidde det med middels eller mot høg måloppnåing. Men oppgavene skal alltid vere, mine er alltid, det er meir «*learning by doing*». (informant 2).

Slemmen skriver at diskusjonen i skoledebatten her hjemme har vært om vurderingens effekt. (Slemmen, 2010) Hun stiller spørsmål om vurderingen hemmer eller fremmer læringen i skolen og om vurdering er læring eller kontroll. Hun skriver vidare at undersøkelser de siste årene viser at vurdering kan ha både negative og positive konsekvenser for elevene.

Dersom man ser vurdering som kun karakterer og resultatbaserte prøver, kan det gå på bekostning av elevers motivasjon for å lære. Dersom man ser vurdering som mer enn prøver og karakterer, kan vurdering fremme elevers læring. (Slemmen, 2010).

Slemmen viser til undersøkelser gjort i England på slutten av 90 – tallet der man ønsket å finne ut om vurdering FOR læring kunne være med på å fremme elevers læring. Myndighetene ble bekymret for konsekvensene av resultatvurderingen, der politikerne hadde vært mer opptatt av resultater enn hvilke konsekvenser et slik fokus hadde på elevene. Dermed ble det satt i gang en undersøkelse av en forskergruppe «Assessment Reform Group» ledet av Black og William. (Slemmen, 2010). Forskerne ønsket å finne svar på tre spørsmål:

1. Finnes det bevis på at vurdering FOR læring kan fremme elevers læring?
2. Finnes det bevis på at det er behov for å forbedre praksisen for vurdering FOR læring?
3. Finnes det bevis på hvordan praksisen for vurdering FOR læring kan forbedres?

Gruppen gikk gjennom 580 artikler om vurdering og fant ut at de kunne svare positivt på alle tre spørsmålene.

På de første spørsmålet ble det konkluderte med at systematisk bruk av læringsfremmende tilbakemeldinger i klasserommet er spesielt til hjelp for de svakeste elevene, men også for klassen som en helhet. Slemmen siterer William slik: «Vi tror at

effektiv bruk av læringsfremmede vurderingsmetoder kan forbedre elevenes læringsutbytte med inntil 70 – 100%». (Slemmen, 2010).

På spørsmålet om det finnes bevis på at det er behov for å forbedre praksisen for vurdering FOR læring, fant Black og Williams at det var behov for endring i klasserommet på grunn av mange ulike utfordringer. De mener at tester og prøver kan ha en tendens til å fremme pugging og overfladisk læring, skoler har dårlig delingskultur blant lærerne og resultater kan blir brukt som et konkurranseelement framfor personlig utvikling blant elevene. Dette kan føre til at vektlegging av karakterer blir viktigere enn det å finne ut hvor elevenes kompetanse ligger. (Slemmen, 2010).

På spørsmålet om det finnes bevis på hvordan praksisen for vurdering FOR læring kan forbedres, vektlegger Black og William tre hovedpunkter som man bør tenke på for å forbedre vurderingen i klasserommet: elevenes selvtillit, egenvurdering og utvikling av effektiv undervisning. (Slemmen, 2010). En klassekultur som fokuserer mye på karakterer, kan igjen føre til at mange elever prøver å finne den letteste veien til toppkarakteren, uten å tenke på eget læringsbehov. Videre kan det å sammenligne egne resultater med andres gi elevenes selvtillit en knekk. Som lærer på en ungdomsskole kan jeg si meg enig i det siste utsagnet her. Vi ser dette nesten hver gang vi har prøver og resultatene skal deles ut i klassen. De flinkeste elevene i klassen er ofte veldig høylytte og deler resultatene deres villig vekk, samtidig som de er interessert i å høre hva medelevene har fått. Vi ser at dette ikke alltid er like heldig for de elevene som får dårligere karakterer. Disse elevene får ofte en følelse av at de ikke mestrer, noe som igjen kan føre til mindre innsats. «*Hvorfor skal vi jobbe når vi ikke får det til likevel?*». Disse elevene klarer ikke alltid å se sin egen framgang, noe som skolen og lærerne kunne blitt flinkere til å formidle.

Black og Williams mener at det er viktig at elevene selv vet hva de kan oppnå og viser til tre punkter:

1. Elevene må vite hva de skal lære
2. Elevene må ha kunnskap om hvor de står i forhold til det de skal lære
3. Elevene må vite hva de skal gjøre for å komme til målet fra der de er nå

De skrives videre at oppgaver i undervisningen må bli begrunnet av læringsmål, og at elevene må involveres i planleggingen av undervisningen. (Slemmen, 2010).

Vurdering er et redskap som kan hjelpe deg som lærer å kontrollere om elevene dine har lært det de skal.

5.5 Pragmatismen og positivismen

Pragmatismen er en filosofisk tradisjon som oppstod i USA rundt 1870 –tallet. John Dewey og Charles Sanders Pierce er eksempler på filosofer som var med på å utvikle pragmatismen som teori. Pragmatismen ser for seg tanken som et verktøy for å løse problemer og utforme prognoser og avviser ideen at funksjonen til tanken er å beskrive, representere eller speile virkeligheten. (Wikipedia, 2017).

Pragmatismen kan sees på en teori om handling og konsekvensen av disse og at man dermed bare kan få kunnskap om virkeligheten gjennom praksis og egne handlinger. (Manger, Lillegjord, & Nordahl, 2012). Dewey mente at all kunnskap må kunne vise relevans og betydning i praktiske sammenhenger – «*learning by doing*.» Pragmatismen, også kalt handlingsfilosofien, er opptatt av å studere handlinger som kan bidra til å løse konflikter, finne svar på og forbedre kunnskap. (Solerød, Pedagogiske grunntanker, 2012). I arbeidet med masteroppgaven vil denne teorien være relevant med tanke på min korte empiriske undersøkelse basert på enkle spørreskjema/enquêter med lukkede spørsmål, der avkryssing av innlagte svar er eneste mulighet for besvarelse.

Silverman hevder at positivismen er den mest vanlige modellen som brukes i kvantitativ forskning. (Silverman, 2005). En del av oppgaven min består i en enkel survey og vil da være et kvantitativt arbeide. Surveyen ble sendt ut til samtlige skoler som ble med i første omgang da koding ble en del av valgfagene man kunne velge fra høsten 2016. Respondentene vil alle vil få de samme spørsmålene og de fleste spørsmålene er det ikke mulig for informantene å komme med individuelle og personlige innspill. Utfallet vil da kunne måles i tall, det vil bli brukt et nøytralt språk og det vil bli foretatt sampling av et stort utvalg av den aktuelle populasjonen.

I følge Store norske leksikon er definisjonen på positivisme slik:

«Positivismen er en betegnelse for vitenskapelig tilnæringsmåte som fremhever den menneskelige erkjennelsens sansbare, empiriske grunnlag og avviser all metafysikk». (Sletnes, 2015).

En annen definisjon er som følger:

Mid- century positivistic conceptions of scientific method and knowledge stressed objectivity, generality, replication of research and falsification of competing hypotheses and theories. Their beliefs in scientific logic, objectivity, unitary methods and truths legitimized reducing qualities of human experience to quantifiable variables. (Charmaz, 2014).

Positivismen søker årsaksforklaringer basert på fakta og erfaring, og har derfor vært en hovedtendens innenfor naturvitenskapen, men også innenfor humaniora, spesielt der man vektlegger observasjonsdata. Man baserer seg på målbare iakttagelser, nøyaktig og nøytralt språk og at forskeren har et objektivt og passivt syn på forskningen og resultatene av denne.

«Viten var noe en kunne vinne gjennom erfaring, og vitenskapens oppgave er å finne regelmessigheter og lovmessigheter mellom positivt fitte størrelser.» (Solerød, Pedagogiske grunntanker - i et dannelsesperspektiv, 2012). August Comte (1798 – 1857) ble sett på som opphavsmannen av denne teorien i nyere tid. Solerød skriver videre at Comte mente at rasjonaliteten ikke bare kunne innføres innen naturvitenskapen, men også innen humaniora. Ved å organisere arbeidet i empiriske data, teste ulike hypoteser og vise til regelmessigheter, kunne man kunne både kontrollere og forutse samfunnet.

I dag vil man sannsynligvis referere til postpositivismen, der dette begrepet forstås ved at vi gir uttrykk for at det kan være vanskelig å gjennomføre absolutte og kontrollerte målinger. (Befring, Forskningsmetode med etikk og statistikk, 2007). Befring sier videre at:

«...vi vil søke etter data ved hjelp av mangeartede og indirekte målinger, og vi vil se behov for repeterte undersøkelser (replikasjon) og flere metodiske tilnærminger (triangulering).» (Befring, Forskningsmetode med etikk og statistikk, 2007).

6. Metode

For å finne svar på problemstillingen og forskningsspørsmålene i oppgaven vil det være nyttig å foreta en enkel kvantitativ kartleggingsstudie for å få en oversikt over hvilke praksiser og vurderingsformer som finnes i faget. Disse undersøkelsene ble rettet mot lærerne som har jobbet som kodelærere siden prosjektet startet høsten 2016. En enkel survey ble distribuert digitalt til de aktuelle skolene og lærerne. Denne nettbaserte surveyen fordrer ingen pålogging fra respondentene da den blir sendt som en live link.⁶ Surveyen ble sendt til alle 155 skolene som var registret ved start 2016.

Jeg har også foretatt en kvalitativ intervjustudie med fire lærere for å gå mer i dybden på ulike praksiser, vurdering og motivasjon. Informantene til disse intervjuene ble valgt ut på bakgrunn av nærhet til egen arbeidsplass.

6.1 Kvantitativ metode

Kvantitative metoder består av teknikker og prinsipper der man benytter objektive målinger, statistikk og tallmateriale. Kvantitative metoder brukes som regel i større undersøkelser der man har mange elever, lærere eller skoler som innbefattes i forskningen. Kvantitative metoder har en grunnleggende verdi for forskning som vil finne ut noe om generell kunnskap innenfor et felt. (Befring, Forskningsmetoder i utdanningsvitenskap, 2016).

Et viktig trekk i å bruke kvantitativ metode er inndeling og kategorisering av problemfeltet i spesifikke variabler. Disse variablene omfatter både personlige og kontekstuelle egenskaper. Ved målingene tillegges variablene tallverdier.

⁶ Link til survey legges ved som tillegg

Bearbeidingen og analysene av kvantitative data vil dermed foregå statistisk, med tabeller og grafisk presentasjon, samt beregning av gjennomsnitt, variasjon og korrelasjon. (Befring, Forskningsmetoder i utdanningsvitenskap, 2016).

Som regel er det enkelt å se hva som egner seg for statistiske analyser og hva som bør analyseres ved bruk av andre metoder. (Undheim, 1996). Sammenhengen mellom faget koding og organiseringen av denne er et godt eksempel på forhold som krever kvantitative undersøkelser. Det samme kan sies om sammenhengen mellom elevenes kjønn og valg av fag.

6.2 Kvalitativ metode

Postholm skriver at: «Kvalitative forskere nærmer seg sin forskning med utgangspunkt i et paradigme eller et verdisyn». (Postholm, 2010). Dette betyr at forskeren har med seg syn(*paradigmer*) på verden som er med på å styre forskningen. Slike paradigmer kan være kognitivism, positivisme og konstruksjonismen. Postholm hevder videre at mer eller mindre all tradisjonell kvalitativ forskning hører hjemme innenfor et konstruktivistisk paradigme, noe som tilsier at de ulike paradigmene står for både teori og metode. (Postholm, 2010) Postholm skriver videre at det er tre begreper som kan være nyttige å redegjøre for innenfor kvalitativ forskning; *ontologi*, *epistemologi* og *aksiologi*.

Ontologi dreier seg om virkeligheten og hvordan den er, altså tingenes eksistens og egenskaper. I kvantitativ forskning blir virkeligheten skapt av informanter og andre som er med i studien, og målet med forskningen er å forsøke å forstå og løfte fram meningen informantene har konstruert i sin situasjon og sine erfaringer. (Postholm, 2010). På denne måten er det informantenes perspektiv som blir fokuset i kvantitative studier.

Epistemologi dreier seg om forholdet mellom forskeren og de som med i studien. Det blir opprettet et mer personlig samarbeidsforhold mellom forskeren og informantene. Postholm skriver at siden virkeligheten blir konstruert i møtet mellom informanter og forsker, kan det oppfattes som unødvendig å stille spørsmål om hva som finnes og dermed bli kjent. «Dermed blir skillet mellom ontologi og epistemologi noe uklart». (Postholm, 2010). Likevel kan man skille disse noe, da epistemologi kan innbefatte beskrivelsen av det fysiske miljøet, som antallet pulter i et klasserom og eventuelt annet

utstyr som man ikke kan være noe uenighet om. Hvordan pultene er organisert kan derimot ha noe å si for hvordan læringen i klasserommet foregår. (Postholm, 2010).

Aksiologi handler om læren om verdier. Postholm skriver at en kvalitativ forsker innser også at forskningen er påvirket av hans/hennes subjektive, individuelle teorier. Postholm hevder at: «Det er viktig at forskeren legger frem sine perspektiver og meninger, slik at leseren kan se hvordan forskeren har påvirket forskningsarbeidet». (Postholm, 2010). Dette vil da være med på å kvalitetssikre studiet.

Vi kan også si at kvalitative metoder bygger på hermeneutikk (*fortolkning*) og fenomenologi (*erfaring*). Metoden omfatter innsamling, bearbeiding og analyse av materiale fra intervjuer og observasjoner. Kvalitative forskningsmetoder har ofte relevans for aktuelle, empiriske undersøkelser og man bruker ofte direkte observasjoner og dybdeintervjuer for å innhente data. Kvalitative data kan også omfatte dagbøker, tegninger, lyd – og videoopptak med mer. (Befring, Forskningsmetoder i utdanningsvitenskap, 2016). Dybdeintervjuer med noen få lærere om hvilken motivasjon som ligger bak valget for å være lærer i kodefaget, kan sies å være et eksempel på informasjon som krever en kvalitativ tilnærming.

6.3 Fordeler og begrensinger i valg av metode

Det finnes både sterke og svake sider ved begge metodene. Ved en vellykket kvantitativ undersøkelse får man et mer omfattende resultat enn når man bruker kvalitative metoder. Ved systematisk innsamling og analyse av empiriske data kan en oppdage og kartlegge relevante sammenhenger. Dette kan i neste omgang gi grunnlag for teoriutvikling og praktiske forandringer. (Befring, Forskningsmetode med etikk og statistikk, 2007). På den negative siden er det slik at det kan ofte blir store mengder data å analysere, siden man her arbeider med materiale fra mange informanter. Det kan også være utfordrende å forfatte et spørreskjema som ikke er for omfattende og komplisert da man helst vil ha svar fra så mange som mulig. Tidsbruk ved å sende ut en undersøkelse bør også tenkes på, samt hvordan få flest mulig av den utvalgte populasjonen til å svare på undersøkelsen. (Befring, Forskningsmetode med etikk og statistikk, 2007).

En kvalitativ metode vil kunne gå mer inn i dybden på det man ønsker å finne svar på. I den kvalitative forskningen brukes ofte ord og frie uttrykksformer i stedet for tallmateriale. Dette betyr ofte at nye spørsmål og tentative konklusjoner kan oppstå under selve innsamlingen av materialet. Ved begge metodene er det viktig å tenke objektivitet, man må kunne skille eget yrke og det å være en forsker. Ved å bruke kvalitative metoder; i denne oppgaven dybdeintervju, kan det være vanskelig å gå inn i situasjonen uten å være subjektiv. Det blir viktig å ha utarbeidet en klar spørsmålsguide slik at man kan lettere holde seg innenfor det temaet man vil at det skal svares på. Likevel kan man i slike situasjoner ikke forutse hva informanten velger å fortelle og man kan sitte igjen med lange intervjuer som skal transkriberes. Selv om antallet informanter i en kvalitativ undersøkelse ikke er så omfattende, kan man likevel bruke like mye tid på å analysere dette materialet som man bruker på et kvantitativt arbeid.

7. Funn, analyse og drøfting

7.1 Metodekritikk

7.1.1 Om utvalget

Når man ser på utvalget i den kvantitative undersøkelsen ønsket jeg å få så mange respondenter som mulig, og jeg sendte ut forespørsler til alle skolene som hadde meldt seg på som kodeskoler fra høsten 2016. Av 155 skoler, var det 42 lærere som responderte på surveyen. I tillegg fikk jeg svar fra noen skoler der de fortalte at de hadde begynt med programmering, men hadde avsluttet faget grunnet manglende interesse fra elevmassen eller at de ikke hadde kvalifiserte lærere.

Når det gjelder den kvalitative delen, forsøkte jeg å intervjuere lærere fra ulike fylker, ulik skolestørrelse, samt lærernes erfaring i skolen. Jeg hadde noen tanker om kjønn, men siden jeg sendte forespørselene til ledelsen på hver enkelt skole, var det umulig å vite kjønn på lærerne som hadde programmering på den enkelte skole. I ettertid ser jeg at jeg kunne sent forespørsler om dybdeintervju tidligere i prosessen, samt utvidet området noe. Jeg har dybdeintervjuet fire lærere, en fra Akershus, to fra Rogaland og en fra Hordaland. Tilfeldighetene ville det at to var kvinner og to var menn, og at to var ganske nyutdannede mens de andre to hadde lang erfaring i skolen.

7.1.2 Bortfallsanalyse og svarrespons

Bortfall av respondenser er en viktig feilkilde, da dette kan tyde på at undersøkelsen man har sendt ut ikke blir sett på som relevant eller at den ikke er utformet på en god måte. (Befring, Forskningsmetode med etikk og statistikk, 2007). Som nevnt ovenfor fikk jeg bare 42 av 155 svar, noe som kan tyde på at spørreskjemaet ikke var optimalt eller at en slik undersøkelse ikke var interessant nok for skoleledelsen å sende videre til sine kodelære. Det kan også være at tidspunktet for undersøkelsen var mindre heldig, da den ble sendt ut i april/mai. Dette er en periode som er travel på skolene med forberedelser til tentamener, eksamener og mye rettetarbeid for lærerne.

Hvor mange som deltar i en spørreundersøkelse har mye å si for påliteligheten eller troverdigheten i undersøkelsen, altså denne står i forhold til hvor mange personer som er i populasjonen man undersøker. I denne surveyen var populasjonen 155 og jeg fikk svar fra 42 av disse, noe som tilsvarer litt over 27%. Med et konfidensintervall⁷ på 95% ligger da feilmarginen på +/- 12,9% , noe som tilsier at feilmarginen er ganske høy og man kan argumentere med at surveyen ikke er av stor betydning for arbeidet i denne oppgaven. Jeg vil likevel sammenligne svar fra respondentene med dybdeintervjuene for å se om det er korrelasjon på felles spørsmål.

På den kvalitative delen av undersøkelsen ringte jeg til rektorer ved åtte skoler, og jeg fikk positiv respons fra sju av disse. Rektorene satte meg i kontakt med kodelærene og av ni lærere fikk jeg positivt svar fra fire. Grunnivelsen til de som ikke ville være med var hovedsakelig at de ikke hadde tid, selv om de var positive til prosjektet. Etter diskusjon med medstudenter og veileder, konkluderte jeg med at fire dybdeintervjuer var tilstrekkelig for å få gode svar innenfor forskningsspørsmålene i denne oppgaven.

7.1.3 Innhenting av data – survey

Innhenting av data på surveyen ble gjennomført ved hjelp av et elektronisk spørreskjema utarbeidet i Office 365/forms.⁸ Surveyen ble sendt til alle skolene på listen over kodeskoler og den var anonym. Surveyen består av både ja/nei spørsmål, åpne

⁷ Et **konfidensintervall** er i statistikken en måte å angifeilmarginen av en måling eller en beregning på. Sannsynligheten angis i prosent. <https://snl.no/konfidensintervall>

⁸ Se vedlegg.

spørsmål og noen få spørsmål som skulle besvares ved bruk av Likert-skala. Det er viktig å understreke at bruken av Likert-skala kan være diskutabelt da det ikke finnes et riktig svar på hvor mange kategorier som kan/skal brukes. Det er likevel mest vanlig å bruke fem, noe jeg har holdt meg til i min survey. Johannesen et al peker på at det er uenighet blant forskere om man i det hele tatt skal ha med en nøytral kategori (jeg vet ikke/hverken eller), men jeg har valgt å ha med denne kategorien her. (Johannesen, 2010).

7.1.4 Transkribering av intervjuer

«Å transkribere betyr å transformere, skifte fra en form til en annen». (Kvale, 2009). Dette innebærer altså at man skal oversette talespråk til skriftspråk, lydfiler skal omgjøres til tekst. Målet med å transkribere muntlig materiale er å gjøre informasjonen egnet for analyse. Det kan være vanskelig å gjengi kontekst i en transkripsjon, da man ikke får med seg det sosiale samspillet mellom partene i intervjuene, altså stemmeleie, gester og så videre. (Kvale, 2009).

Innhenting av data for den kvalitative delen av arbeidet ble gjort med intervjuer ansikt til ansikt. Jeg avtalte på forhånd når jeg skulle besøke respondentene og de fikk informasjonsskriv og spørsmålsguide⁹ sendt på mail noen dager før intervjuet fant sted. Det ble gjort lydopptak av intervjuene som senere ble transkribert til skriftlige dokument. Prosessen med å transkribere fra lyd til skrift var utfordrende, ikke bare på grunn av mangel på den sosiale konteksten, men også fordi muntlig språk er så forskjellig fra skriftspråket. Jeg var usikker på om jeg skulle ta med alle «ehh, hmm, tja» og så videre, samt at jeg opplevde at informantene ofte gjentok seg selv. Det var også en utfordring med ulike dialekter, samt at en av mine informanter ikke har norsk som morsmål. Jeg opplevde også at det innimellom var vanskelig å følge spørsmålsguiden kronologisk, da respondentene ofte begynte å snakke om noe annet. Det kunne da være vanskelig å komme tilbake til et spørsmål, og jeg opplevde mer enn en gang at vi begynte å snakke om ting som jeg ikke hadde tenkt på, eller at samtalen sporet fullstendig av. Jeg synes at det var vanskelig å avbryte informantene når dette skjedde, da de tydelig var interessert og oppslukt av det de snakket om.

⁹ Se vedlegg.

I ettertid tenker jeg at en form for observasjonspraksis over tid kunne vært nyttig for å få et mer nyansert bilde av hva som skjer i en kodeklasse, både når det gjelder undervisningsmetoder, samspill mellom lærere og elever, samspill mellom elevene og hvordan elevene og lærerne vurderer arbeidet som blir gjort.

7.1.5 Reliabilitet og validitet

Krumsvik sier at det er viktig at man oppdaterer og kalibrerer metodene man bruker når man forsker etter hvert som samfunnet forandrer seg. (Krumsvik, Forskningsdesign og kvalitativ metode, ei innføring, 2014)

Reliabilitet handler om pålitelighet i forskningen som er gjort; kan en annen forsker få de samme resultatene ved å gjøre nøyaktig de samme undersøkelsene? Dette kan være svært vanskelig når man bruker kvalitative metoder som intervju, da sannsynligheten for at informantene ville svart akkurat likt ved et nytt intervju er små. Krumsvik sier at:

Intervjureliabiliteten er relatert til spørsmålsformuleringene; er spørsmålene klare og tydelige, er spørsmålene med på å forme svarene, er spørsmålene ledende på noen måte og bruker man et vokabular som er kjent for informantene. (Krumsvik, Forskningsdesign og kvalitativ metode, ei innføring, 2014).

Ved å arbeide med transkribering og gjennomgang av intervjuguiden noen ganger, ser jeg at jeg med fordel kunne jobbet mer med den. Jeg ser at noen spørsmål kan være ledende, samt at ikke alle er like godt relatert til forskningsspørsmålene.

Validitet i kvalitativ forskning handler om forskeren undersøker det man hadde planer om å gjøre, mens i kvantitativ forskning er validiteten mer relatert til om en måler det man skal måle. (Krumsvik, Forskningsdesign og kvalitativ metode, ei innføring, 2014). Videre skriver Krumsvik at begreper som knyttes til dette er troverdighet, bekreftbarhet og overføringsverdi. Videre skilles det mellom intern og ekstern validitet. Intern validitet handler om sammenhengen mellom det teoretiske rammeverket og det forskeren har funnet ut, mens ekstern validitet handler mer om funnene i forskningen kan generaliseres på tvers av sosiale settinger. (Krumsvik, Forskningsdesign og kvalitativ metode, ei innføring, 2014).

Når det gjelder validitet i oppgaven, mener jeg at dette er noe som er mer vellykket. Teorien henger sammen med det jeg har forsket på, samt at funnene kan generaliseres på tvers av sosiale settinger.

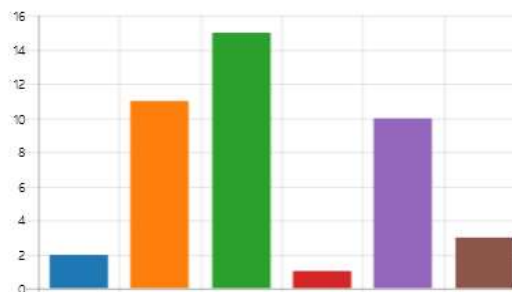
7.2 Funn og drøfting

Hvilke ulike praksiser eksisterer for faget programmering i ungdomsskolen, og hvilke innholdskomponenter legger lærerne vekt på i utformingen av faget.

I denne delen av oppgaven vil jeg forsøke å analysere funn sett i lys av problemstillingen og forskningsspørsmålene i oppgaven. Men før jeg går inn på forskningsspørsmålene tenker jeg at det kan være interessant å se på utdanningsnivå, erfaring i skolen og kjønn i populasjonen/respondentene. Utdanningsnivået blant respondentene på surveyen ses nedenfor, og vi ser at et flertall av lærerne er på adjunktnivå.

3. Utdanningsnivå

[Fleire detaljar](#)



Figur 8 – utklipp fra survey

På oppfølgingsspørsmålet hvilken utdanning har du hvis du har svart annet, var det ni som svarte, noe som ikke stemmer helt med foregående spørsmål da det var bare tre som svarte annet. Blant svarene var det en sivilingeniør, en radiotelegrafist, en telemontør og en høgskolekandidat innen IT og økonomi. Blant lærerne i dybdeintervjuene svarte de slik:

«Faglærer i kunst og håndverk og master med tema design og forbruk». (informant 1)

Eg har fire år på HVL, allmennlærerutdanning med spesialisering i engelsk og naturfag, men eg har og tre år på universitet i Bergen der eg studerte engelsk litteratur og så har eg to år fra Memorial University of Newfoundland i Canada. Det var to år med generelle studier. (informant 2)

Før det så har jeg vært utdannet telemontør. Så det har jeg fagbrev i. Skole så er det realfag som er mitt felt. Og så har jeg 60 studiepoeng i IT. Også har jeg 60 studiepoeng i kroppsøving. (informant 3).

«Eg har universitetsutdannelse med matematikk, kjemi og geografi. Og så har eg eit kurs i programmering på universitetet fra 1973». (informant 4).

Vi ser at det er få lærere som har utdanning innen programmering, og de som har noe er ofte basert på korte kurs og lignende. Dette stemmer godt med tidligere forskning som mener at de enda er få lærere som har utdanning innenfor kodefaget. Tuomas Hiltunen refererer til en av sine informanter som sier at:

A fear that arises, or the greatest challenge, is not the students, but the teachers. We have teachers who are not so enthusiastic or interested in the matter, and then we have those who are. The gap between them is growing. (Hiltunen, 2016, s. 39).

En av mine informanter sier det slik:

Så då er det ein del utfordringar med hensyn til å få andre lærarar til å ta det på alvor. At ein kan berre fungere som ein alltid har gjort og det kan de ikkje. Fordi då føler me ikkje kompetansemåla for digitale ferdigheter, det gjer me ikkje. (Informant 2)

Grout and Houlden skriver at:

Not only were schoolchildren about to be immersed in an entirely unfamiliar syllabus, but so were their teachers. Most estimates put the number of existing IT/ICT teachers unprepared for the new Computing Curriculum in the tens of thousands. (Grout, 2014, s. 681).

Man kan tenke seg at kommunene og skolene som valgte å være med på å prøve programmering som valgfag er udelt positive til IKT i skolen og er dermed villige til å satse på både utstyr og etterutdanning . Når det gjelder etterutdanning viser det seg at det er bare 15 av 42 lærere som har fått tilbud om dette. Disse hadde i hovedsak fått tilbud om IKT- senteret sine MOOC, men det var også noen som skulle begynne på NTNU og HVL.

6. Har du fått tilbud om etterutdanning i kodefaget?

[Fleire detaljar](#)



Figur 9 – utklipp fra survey

7. Hvis du har fått tilbud om etterutdanning, hvilken etterutdanning er det snakk om?

13 Svar

4	anonymous	Ja, skal studere programmering neste skoleår.
5	anonymous	Har deltatt på noen endags kurs. Planleggingsdag og kurs i oslo dagen etter samling i kodefaget
6	anonymous	IKT senteret sin MOOC
7	anonymous	UDIR/LKK sin MOOC (tilbudt alle som er med i forsøksordningen)
8	anonymous	MOOC, UIS ...
9	anonymous	3x5 stp IKT i Læring, NTNU 30 stp Programmering for lærere, deltid, NTNU 60 stp Programmering, UiO (?) 60 stp lærerspesialist i IKT, UiO (?)
10	anonymous	PROfag og kurs og konferanser gjennom utdanningsetaten i Oslo.
11	anonymous	Kursing for forsøkskommuner med Programmering som valgfag + andre småkurs av kodeklubben og programmerings- mooc.
12	anonymous	15 stp via Høgskulen på Stord (Det er alt eg har av kodeutdanning)
13	anonymous	matematikk og naturfag

Figur 10 – utklipp fra survey

Av de fire informantene jeg dybdeintervjuet var det to som hadde fått tilbud etterutdanning, de andre to hadde fått informasjon om MOOC som ligger ute på nettet

for de lærerne som ønsket å gjennomføre disse. Informant 2 hadde fått tilbud, men ville ikke benytte seg av dette enda.

Jeg har noen kurs i koding fra før som ikke er fra universitet eller høyskolen, men fra privatpersoner. Og så har jeg kommet inn på NTNU på nettstudier, så jeg skal gå der fra høsten av. (informant 1).

Nei, eg har ikkje fått noko tilbod om det. Når eg blei med på programmering som valfag som eit pilotprosjekt så fikk eg tilbod på ein sånn MOOC, men det fikk alle tilbod på, men det er ikkje studiepoeng knytt til dette. (informant 2).


«Ja, det har jeg, så.. Ikke nå. Men jeg har lyst. På sikt». (informant 3).

Ja, nå har dei jo lagt ut desse sentrale oppgåvene på programmering, så eg har vore inne og gjort litt på nokre av dei. Med det eg ser er at, det viktigaste her er jo det når ein får dette utstyret, så er det rett og slett det å eksperimentere litt sjølv. Altså eg har jo har jo heldt på med Arduino, der har eg kunne spurt sønnen min en del og så har eg prøvd meg fram. Og på samme måten med den der Mikrobiten, der ligger det jo veldig mye bra på nettet. (informant 4).

Jeg tenkte det kunne være interessant å spørre hvorfor de var kodelærere. Det kan tenkes at de fleste kodelærerne i skolen melder seg frivillig utfra egen interesse og kunnskap, men det kan også tenkes at slike lærere ikke finnes på alle skoler selv om skolene har vist interesse for å starte med faget. Som vi ser av surveyen har de fleste meldt seg frivillig.

8. Hvorfor ble akkurat du lærer i koding ved din skole?

[Fleire detaljar](#)

 Forespørsel fra ledelsen	15
 Meldte meg frivillig	22
 Annet	5

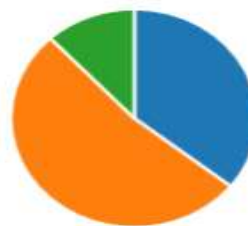


Fig 11 – utklipp fra survey

Informantene i dybdeintervjuene svarte slik:

Jeg ble spurt. Fordi, jeg vet hvordan man bruker Word og sånne ting. Og.. haha, kan bruke smart Board. Og bruker det aktivt. Jeg gikk og spurte om jeg kunne være så snill å få et smart Board da jeg begynte å jobbe her. For da var det ikke smart Board i mitt klasserom, i kunst og håndverk. Da kom de til meg og så lo de, da sa de nei. Du kan bare glemme. Du får ikke noe smart Board du. Nå har jeg det. Så det gikk fint. Hm, og så når jeg har interessen så er det jo greit forså vidt så da, da mente de at det. Nå er jo mannen min som er programmering og, så da kunne jeg jo liksom få litt hjelp der da. (informant 1).

Eg trur det var begge deler. Eg og rektor, me hadde prata mykje om det før det kom noko informasjon om koding som valfag. Eg hadde teknologi i praksis som valfag i to år før dette, og der hadde eg jobba med koding, som ein av oppgåvene i to år før dette. Og eg hadde snakka med rektor ein del og så begge, eller eg hugsar ikkje heilt om det var eg eller han som sa det først, men det var mange samtaler rundt dette og det var openbart at dette noko eg var, han visste allereie at eg var interessert i. (informant 2)

«Jeg ble spurt av ledelsen. Jeg.. rektor hadde sett at jeg hadde programmering i fagkretsen og hun ønsket at jeg skulle ha det. Så jeg ble spurt og da takket jeg ja». (informant 3).

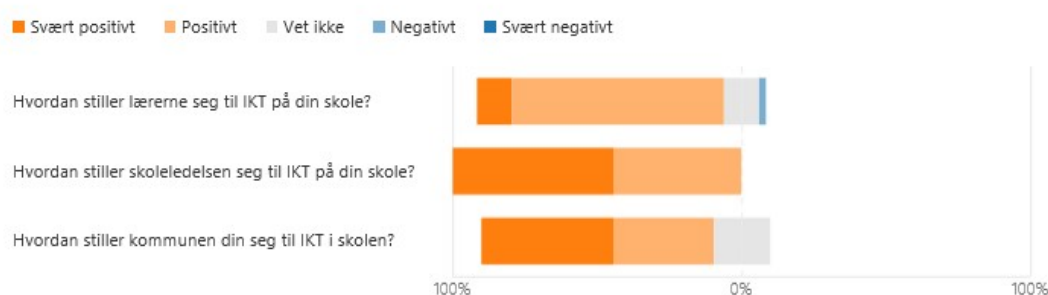
Ja, då fikk eg spørsmålet om eg kunne tenke meg å være med, fordi at for at kommunen skulle kunne søke så måtte dei ha to skuler som måtte vere med. Eg synes jo dette var spennende. For eg var jo òg med 1984, som it-ansvarlig fra den gongen Ped-senteret var på tur til Derbyshire. Så då vi kom tilbake då, så var det en del av oss som eksperimenterte litt med å lage undervisningsprogrammer i BASIC. Fleire av dei har jo jobba vidare med sånne ting. Sånn at eg har liksom hatt litt interessa heile vegen. Og så har eg ein son som er hekta på programmering og har vore på datatreff. (informant 4)

Lærere som svarte på surveyen, oppga ulike grunner til at de ble kodelærere på sin skole. Noen hadde vært pådrivere for å få starte faget på sin skole, noen var allerede IKT ansvarlige på sine skoler og noen oppga at de overtok fra lærer som hadde sluttet.

På spørsmål om holdninger fra kommune og skole, ser vi at de aller fleste er positive til IKT, det er noen som svarer at de ikke vet og det er noen få som mener at kommunen/skolen de jobber for er negative til IKT. Det kan være vanskelig å tolke svarene fra surveyen, da det kan hende at den enkelte lærers holdning kan spille inn. En skulle også tro at når en kommune/skole valgte å ha programmering på skolene at de da var villige til å investere i utstyr og etterutdanning. Man kunne søke om økonomisk støtte fra Utdanningsdirektoratet, men det var ikke alle som fikk dette.

20. Holdning til IKT

[Fleire detaljar](#)



Figur 12 – utklipp fra survey

Informantene som ble dybdeintervjuet mente i hovedsak at kommunen/skolen var positive. Noen savnet mer positivitet fra kolleger, og en savnet også at kommunen investerte i nytt og bedre utstyr for å gjøre undervisningen bedre.

Nå er det et satsningsområde hos oss, så holdningen er jo at kompetansen skal bli sterkere blant alle lærerne, og ledelsen skjønner absolutt at det er noe vi må få tid til å gjøre fordi vi også er veldig mange som ikke har nok kompetanse og det er vi jo på vei til å få. Så hver gang det kommer et kurs, eller vi får tilbud om seminarer og så videre om IKT, så får vi lov å dra på det. Og så blir de veldig glade hvis vi formidler videre til personalet. Vi holder samlinger og sånn og så får vi få tid til å bli bedre. Ja, altså en del er vel litt sånn, det er ikke det at de er negativt innstilt, men det er ah, skal vi gjøre dette igjen liksom, fordi ofte så blir det en del av det samme og så skjønner man ikke helt hvorfor man trenger å lære dette på nytt eller hvorfor trenger jeg å bli bedre i noe jeg tror jeg allerede kan. Men så viser det seg jo at når vi har disse kursene at det er mange som ikke kan det vi skal lære oss. Så det er jo, jeg tror mange tenker at data og IKT er IKT og at det står stille. Men det gjør jo ikke det, det skjer jo noe nytt hele tiden og da

må man jo få en kontinuerlig opplæringen da. Men alt i alt en god holdning da.
(informant 1 – skole i Akershus).

Eg føler at mange av skolene, i alle fall på vår skole har begynt å skjønne at dette noe som burde blitt tatt på alvor for noen år siden. Og dei gjer det dei kan for å ordne opp i det. Så eg meiner de satsar stort på det det tekniske, på ustyrssida, men fordi me jobbar i etterkant sant, dette er noko me burde ha begynt med for la oss seie for 5 – 7 år sidan. Så då er det ein del utfordringar med hensyn til å få andre lærarar til å ta det på alvor. At ein kan berre fungere som ein alltid har gjort og det kan de ikkje. Fordi då føl me ikkje kompetansemåla for digitale ferdigheter, det gjer me ikkje. Men kommunen har vore veldig stor støtte innanfor dette, de, kvar gong eg ville prate med nokon om utstyr eller kurs eller andre ønskje er dei veldig positive til dette. Me fekk ikkje noko økonomisk støtte av UDIR eller staten, når me blei ein del av valfaget, men alt eg har bedt om fra kommunen har eg fått. Så lenge eg held meg innafor rimelighetens grenser, hehe. (Informant 2 – skole i Hordaland).

Altså, det som jeg syntes jo er bra er at, holdt på si, xx ungdomsskole og xx ungdomsskole takket ja til å bli med, men vi fikk ingen ressurser. Og det mener jeg er feil. For det har jo gjort at vi har slitt i forhold til det. Så hadde vi fått de ressursene som de andre fikk så hadde jo dette vært lettere å betjene. Ja, altså nå får vi i Pad-er, så det holdt på å si, så jeg håper jo at vi kan få rustet opp PC parken vår. Da vil det vært, mye ville vært enklere, og i programmeringsfaget.(informant 3 – skole i Rogaland).

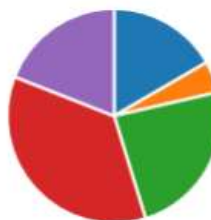
Den (*holdningen*) er jo veldig bra her, altså.. Alle våre elever har nettbrett, og så har vi bærbare PCer til litt over et trinn, så vi har litt over 100 bærbare PCer som står på ei tralle. Så det er aldri et problem, bortsett fra på tentamen, så er det alltid ledig. Så eg kan alltid ta meg ei tralle til disse elevene. Eg fikk bestille inn utstyr frå kommunen. Dei sponsa, så derfor så første året så, så fikk eg, eg sendte ned ei liste med det eg ønska, så fikk eg bestille for 12 000 kroner. (informant 4 – skole i Rogaland).

Det er mye erfaring blant lærerne som deltok i surveyen, de aller fleste har over seks års erfaring i klasserommet. Dette kan tyde på at det er erfarne lærere som underviser i programmering.

2. Hvor lenge har du jobbet som lærer?

[Fleire detaljar](#)

1 - 3 år	7
3 - 6 år	2
6 - 9 år	10
mer enn 10 år	15
Mer enn 20 år	8



Figur 13 – utklipp fra survey

Informantene i dybdeintervjuene bekrefter dette. Her er det kun en lærer som har mindre enn seks års erfaring.

«Begynte i 2013, så da blir det 5 år». (informant 1).

«Eg har jobba i skulen, eg er snart ferdig med mitt sjette år, her på «x» ungdomskule». (informant 2).

«I 12 år». (informant 3).

«36 år». (informant 4).

Når det gjelder kjønn er det en overvekt av menn; det er 31 menn og 15 kvinner av de som er med i undersøkelsen. Da har jeg regnet med dybdeintervjuene. Dette kan stemme med undersøkelser som hevder at det fremdeles er en overvekt av menn som jobber med teknologiske fag, både i skolen og i næringslivet for øvrig. I følge en artikkel publisert av digi.no viser en undersøkelse fra Deloitte viser at:

Kvinneandelen i den globale IT-bransjen antas å være på under 25 prosent ved utgangen av 2016, noe som i beste fall er på nivå med fjoråret. I land som Sverige og USA har kvinneandelen i IT-bransjen falt noe de siste fem til ti årene, men ikke dramatisk. Men andelen kvinner blant studentene som oppnår en universitetsgrad innen informatikk, har falt betydelig på noen tiår. I 1985 var 37 prosent av avgangsstudentene i informatikk i USA kvinner. Dette var redusert til 18 prosent i 2013. Lignende tall ser

man også i Storbritannia og Sverige. IKT-Norge mener at utviklingen i Norge er sammenlignbar. (digi.no, 2016).

1. Mann eller kvinne

[Fleire detaljar](#)



Figur 14 – utklipp fra survey

Ser vi på kjønnsfordelingen blant elevene er dette med på å bekrefte at teknologi fremdeles er en «guttegreie», selv om en artikkel publisert på NRK skriver at:

«*Samordna opptak melder om at jentene strømmer til teknologifag*». (Johannessen, 2018). Artikkelen skriver at 74 prosent flere kvinner har fått tilbud om plass på et av IT-studiene til OsloMet i år sammenlignet med i fjor. Informantene på surveyen er i samsvar her, det er veldig få jenter som velger programmering som valgfag.¹⁰

Nei, jeg har vel fire jenter av de 29, så det er, men det er flere jenter enn i fjor. Og en av, eller når jeg presenterte valgfaget for elevmassen eller elevgruppen så, så viste jeg en av de videoene til IKT-Norge som går på at jenter også bør involvere seg i teknologi eller teknologofag, fordi vi tenker annerledes. Vi bør, eller altså, det er ikke noen grunn til at vi ikke skal søke selv om vi ikke kan det og det er sånn typisk jentegreie. Det er sånn «jeg kan ikke så da kan jeg ikke søke», men guttene kan det jo ikke de heller. Så da fikk jeg hvert fall tre flere søkere enn i fjor da, det var fint. Så utfordrer jeg de til neste år og så ser vi hva som skjer. (informant 1).

Eg hadde ei jente det første året, eg har ingen i år og det er litt synd. Eg har sagt at eg kunne likt ein betre fordeling der, men eg har ikkje fått det. Eg veit det har vore jenter som har søkt, men dei har ikkje kommen inn i valfaget mitt, dessverre. (informant 2).

«Ja, I niende har jeg to jenter. I åttende ingen». (informant 3)

¹⁰ Se vedlegg av survey – spørsmål 14

«Vi har hatt veldig få jenter, i år har eg berre to jenter. I fjor var det vel det same». (informant 4).

I denne delen av arbeidet vil jeg forsøke å kommentere og analysere spørsmål som er med på å belyse forskningsspørsmålene i oppgaven.

F.1 Hvordan organiserer lærerne faget?

Det er noe variasjon på hvordan skolene organiserer faget. Noen skoler har elever fra alle tre trinn i en klasse, mens andre har en gruppe fra hvert trinn. Det er også forskjell på hvor mange elever man har i gruppa. Noen ser ut til å ta inn så mange som mulig, mens andre mener at det bør være et tak på hvor mange elever det skal være i klassen for at undervisningen skal være effektiv.

11. Hvor mange elever er det i kodeklassen?

[Fleire detaljar](#)

● Mindre enn 10 elever	6
● 10 - 15 elever	6
● 15 - 20 elever	14
● mer enn 20 elever	16



Figur 15 – utklipp fra survey

Vi ser av surveyen at de fleste klassene har over 15 elever, noe som stemmer overens med klassestørrelsene hos respondentene i dybdeintervjuene.

«Jeg har en klasse i valgfag og elevtallet er vel nå på 29 stykker». (informant 1).

Eg har satt ein maks på 18 elevar i klassen, så eg er ein av dei minste valfaga på skulen. Så eg vil ha maks 18 elevar, akkurat no har eg berre 16 elevar, men det er fordi nokre elvevar har bytta til andre valfag. Men, ja 16 til 18. (informant 2).

«Jeg har to klasser, så jeg har 18 elever på 8. trinn. Og så har jeg 9. klasse på.. Det var 16. Nå er det 15». (informant 3).

«Ja, i år har eg 15 elever på åttende trinn». (Informant 4).

Det kan være vanskelig å si hva som er det optimale tallet på elever i en klasse. Det er lærere som mener at dess færre elever, dess mer effektiv er undervisningen. Da kan man rekke å hjelpe alle som trenger det, og både undervisningen og vurderingen får en bedre kvalitet. John Hattie, en moderne «guru» innenfor læring, gjennomførte en stor studie som ble presentert i boka *Visible Learning: (Hattie, A Synthesis of over 800 Meta-analysis Relating to Achievement, 2009)*. I denne studien rangerte han over 150 faktorer som skal påvirke elevene i skolen. Denne lista ble utvidet i 2017 der det listes opp over 250 faktorer. (*The Australian Society for Evidence Based Teaching, 2017*). Hattie kom fram til at klassestørrelse var ikke blant det som rangeres som viktigst for elevers læringsutbytte. Hattie sier likevel at klassestørrelse har en positiv effekt på læring, men ikke spesielt sterk. Han mener at færre elever ikke er viktig om ikke lærerne tilpasser måten de underviser på. (*Hattie, A Synthesis of over 800 Meta-analysis Relating to Achievement, 2009*).

Det er ikke alle som er enige med det Hattie hevder. I følge en artikkel Utdanningsforbundet har publisert, er det flere studier som mener at de har bevis på det motsatte. En av flere studier de referer til er prosjektet kalt STAR (Student Teacher Achievement Ratio). Dette var en eksperimentell studie som gikk fra 1985 til 1989 i USA. Dette er en av få studier hvor det har vært en helt tilfeldig fordeling av elevene i små og store klasser, og det er en viktig styrke ved studien. Konklusjonen i STAR-studien var at elever som gikk i små klasser fikk et faglig forsprang på elever som gikk i normalklasser. Gevinsten var størst for ressursvake elever. (Utdanningsforbundet, u.å). En annen faktor når det gjelder klassestørrelse, kan være at det er lettere å spørre om hjelp når det er færre elever å forholde seg til i en gruppe. Slemmen refererer til Hattie som sier at: «Man ser elevenes frykt for å be om hjelp som et hinder for læring, og at trygghet i klasserommet er viktig for å fremme gode læringsprosesser». (Slemmen, 2010). Dette utsagnet er et lite paradoks på det Hattie sier om klassestørrelse, da det kan tenkes at i mindre klasser, der eleven har færre å forholde seg til, kan oppleves som tryggere enn en større klasse. Egen erfaring tilsier at det er ofte lettere for mange elever å spørre om hjelp når de er i mindre grupper.

I følge min survey, hadde noen skoler og lærere mer enn en klasse i koding¹¹. Blant informantene i dypdeintervjuene hadde bare en av lærerne som hadde to klasser, de andre hadde bare en klasse hver.

«Jeg har to klasser, så jeg har 18 elever på 8. trinn. Og så har jeg 9. klasse på.. Det var 16. Nå er det 15». (informant 3).

Av de fire hoved informantene mine hadde to av dem aldersblandede klasser, mens de to andre hadde bare ett trinn i sine klasser. Skolene i Akershus og Hordaland jobbet i klasser der man hadde aldersblanding, mens begge skolene i Rogaland hadde valgt å jobbe trinnvis. Lærerne hadde ulike grunner til hvorfor skolene hadde organisert undervisningen på denne måten.

«Jeg har en klasse i valgfag og elevtallet er vel nå på 29 stykker».

Er det fra alle trinn eller?

«Ja.»

Fordeler eller ulemper med det?

8 trinn er veldig delt da, enten så skjønner de koding og er supermotivert og jobber godt eller så er det helt motsatt. Hvis de først har falt av så er det akkurat som at de, de er liksom ikke er modne nok til å ta en del ansvar for egen læring og sånn sett, ja faktisk gjøre det de skal og da blir det bare vimsing. Så jeg kunne tenkt meg flere 10. klassinger. Spesielt med 9. klassingene går det greit. Der søker de veiledning og får den hjelpen de skal ha de bruker tiden mye bedre enn 8. klasse da. Jeg ville foretrukket færre 8. klassinger rett og slett. (informant 1).

Er det sånn at du har elever fra alle trinn?

Ja, eg har hatt eit ønskje om seks elevar frå kvart trinn. Så det er øyremarka i første omgang, men eg har frå alle tre. Eg har ikkje lyst å ha elevar berre frå åttande eller berre fra 10 trinn. Men det er heller ikkje eit must at de må vere med i tre år, og så liker eg å ha elevar i ulik alder i klassen. På den sosiale sida er det godt å finne ut de er færre og at kan de jobbe samene og hjelpe kvarandre på tvers av alder og trinn. Eigentleg så har alderen veldig lite, altså i alle fall i to år, eg har sett att alderen har ingenting å

¹¹ Se vedlegg av survey – spørsmål 12.

seie i kor stor grad dei klarer dette. Så det er mulig, eg ser ingen ulemper fra min side, så eg likar å ha aldersblanda klasse. (informant 2).

Er det en fordel å ha de separat fra åttende og niende trinn tenker du, eller vil du ha aldersblanding?

Jeg syntes ikke det fordi det er veldig nivå forskjell. Og det fant vi ut nå i høst, fordi at da laget de hjemmeside. Og åttende klasse trengte hjelp om trent til alt. De kunne jo ikke lagre og sende de fonter og alt var nytt for de. Så du måtte ta det helt ABC. Mens niende, som har holdt på ett år først, det var *smukk, smukk*. Så det er klart at det er store forskjell på å få inn nye grupper kontra de som har hatt det nå ett år. Voldsom forskjell. (informant 3).

I fjor var vi to lærarar på 25. Det var ein blanding av åttande og niande trinn. Det var ikkje særlig gunstig. Det var nok litt dei som hadde valt det, men eg følte kanskje litt dette med modningsmessig så spriker det sånn, at det var vanskeleg.

Ok, så da foretrekker du å ha de trinnvis?

Ja, eg vil føretrekke trinnvis, nettopp i forhold med det at skjer så veldig mykje modningsmessig. Eg er ikkje sikker på kva som gjorde at den gruppa fungerte dårlig, det kan jo ha vert nokre av dei elevane som hadde litt utfordringar.

Så, nå har du bare 8.klasse?

«Ja, nå har eg berre 8 klasse og til neste år så tilbyr vi valgfaget berre til 8.klasse.»(informant 4).

Av de 42 lærerne som svarte på surveyen er det et knapt flertall som rapporterer at har aldersblandete klasser.

13. Hvordan er klassene organisert?

[Fleire detaljar](#)

● Aldersblandede klasser	20
● Vi har en klasse pr trinn	19
● Annen organisering	3



Figur 16 – utklipp fra survey

Når det gjelder tidsbruk på kodefaget, er det ulik praksis på den enkelte skole hvordan man fordeler timene man har til rådighet. På Utdanningsdirektoratet sine sider som omhandler valgfag heter det at tidsbruken man har til rådighet skal være 57 timer a 60 minutter pr år. (Utdanningsdirektoratet, 2016). Det er 38 uker i et skoleår, noe som tilsvarer at elevene skal ha 1,5 timer i uken med valgfag.

«Vi har to timer i uka, på 45 minutter så 90 minutter til sammen». (informant 1)

Eg, me har ein tre-vekers periodeplan, der me har a, b og c veker og eg har koding på onsdagar i a og b-vekene. Så det er liksom fra ettermiddagen begge dei dagane. To og ein halv time kvar av desse dagane. Så i løpet av tre veker har dei fem timar. (informant 2)

«Jeg har to timer med 8. i uka, og to timer med 9». (informant 3).

«To». (informant 4).

Jeg var nysgjerrig på om ulike egenskaper/kunnskaper hos elevene påvirket hvordan lærerne organiserte undervisningen. Dette innbefattet individuelle sammenlignet med gruppeoppgaver, veiledete oppgaver sammenlignet med styrte oppgaver. Informantene hadde ulike meninger her, uavhengig om de hadde elever bare fra ett trinn eller om de hadde elever fra alle tre trinnene i gruppa.

Men, er det slik at egenskapene til elevene spiller inn på hvordan du organiserer undervisningen?

Til en viss grad. I starten av undervisningen så hadde vi en undersøkelse om hvor mange som hadde jobbet med programmering tidligere. Og da var det noen som hadde hatt valgfaget før og de fikk mye friere tøyler enn de andre. Men alle sammen har fulgt et opplegg underveis og vi har startet på bar bakke og om jeg har sett at dette kan jo egentlig enkelte av elevene veldig, veldig godt, har de fått lov til å jobbe med egne prosjekter eller lage større, ikke begrense seg så mye, så de er veldig selvgående. Derfor har de valgt valgfaget på nytt da. (informant 1).

Ikkje nødvendigvis organisering, nei. Eg pleier å lage opne oppgåver slik at det trengs ikkje forkunnskaper i mine kodetimer til å delta på lik linje med de andre, men de har nok, eg får veldig mange «gamere», eg kan seie det sånn, elever som er mykje på dataen og dei er, dei har, dei drømmer til å lage det nye skytespillet, det nye «Call of Duty» eller det nye «League of Legends», så resultatet er ofte et spill. Spesielt i begynnelsen bare fordi det er det dei ønsker å gjere. Men, etterkvart så blir det, så begynner dei å gå i ulike retningar. (informant 2).

Veldig. Ja. Fordi at sånn som åttende så er jeg helt nødt for å kopiere og ha en oppskrift hele veien. For at de skal klare å vite hva de skal gjøre. Fordi jeg kan ikke, og det er ikke tid, å gå og hjelpe hver enkelt. Så det må være veldig sånn. Mens i niende så, så har jeg mer kunne sette de i gang også har de vært enda flinkere å samarbeide og da har jeg lagt opp litt annerledes. Men det er jo, i og med at jeg er én voksen på så mange så har jeg vært nødt for å ha selvinstruerende undervisningsopplegg da. (informant 3).

Nei, vi, eg prøver å finne, eg prøver å finne litt gode oppgåver, for det som viser seg med disse om ikkje har nokon erfaring frå før, dei er ikkje i stand til å skrive kode frå starten av. Derfor så finner eg gjerne oppgåver der dei har ein kode dei skriv den inn, og så er på ein måte mye av oppgåva det, for det første å få det til å virke. Nøyaktighet er jo viktig her, det er disse gode på fra før. Og så er det jo då og å kunne forstå den koden og kanskje det viktigste, det å endre ein kode. For eg ser jo at, den måten dei aller fleste lærer seg koding på, er nettopp det å endre ferdig kode. (informant 4).

Jeg ville forsøke å finne ut om det var noen forskjell på hvordan elevene jobbet med styrte oppgaver i forhold til veiledete oppgaver. Jeg ser i ettertid at spørsmålet var noe uklart og informantene tolket spørsmålet forskjellig slik at svarene ikke helt svarte til intensjonen. Svarene indikerer at lærerne bruker en del tid på å veilede elevene i styrte eller oppgaver som lærerne hadde satt på forhånd.

Ja, jeg gjør stort sett det samme i alle fagene mine, med at jeg gir en oppgave som har visse kriterier og visste forventninger. For eksempel skulle de lage en nettside nå, så var et av kriteriene at de skulle lage en nettside om seg selv eller en hobby de har. De skulle ha tekst, overskrift og bilde og de skulle bruke en html CSS og JavaScript og

bortsett fra det så var det sånn; lag en brukervennlig side som man kan lese teksten på, litt sånne ting, men ellers så var det helt fritt. Men det er også viktig å gi dem såpass stor frihet at de kan beholde motivasjonen og samtidig føle at produktet er deres og jobbe ut fra det da. (Informant 1).

Eg begynner ofte med å vise eitt eller to omgrep. Nå skal me gå inn på kva x og y koordinatar er, nå skal vi gå inn på kva ein variabel er, no skal vi gå inn på korleis me kan få ulike input og output og vise korleis dei skal gjere det praktisk, felles på skjermen. Og så gi dei ei oppgåve som testar dette på forskjellig måtar med variert, liksom ulike måloppnåelser. Det skal ikkje vere ja eller nei, men greidde det litt eller greidde det med middels eller mot høg måloppnåing. Men oppgåvene skal alltid vere, mine er alltid, det er meir «*learning by doing*». (Informant 2).

Altså, det er veldig forskjellig. For eksempel så hadde jeg Micro: bit og da hadde jeg laget et hefte til de der de da gjorde oppgave for oppgave. I det heftet. Og noen, de suste igjennom dette, mens andre trengte mange timer. På det. Men, det var jo programmering med ulike ting så de skulle få ut av denne kodebrikka da. Og, om vi var ute, vi hadde kart og kompass, altså på dette, og veldig masse kjekt. Så vi testet i gangene og. Vi hadde, har mye lek og testing da. Også og dette her som de syntes var veldig kjekt, det er at når de har laget de oppgaver og de får den godkjent, så viser vi det på smart Board. Og så blir det mye lott og løye, fordi at de jobber iherdig med at de ikke skal bli for selvhøytidelig. At vi forstår forskjellige av ikke le av, men med. (informant 3)



Eg styrer dei jo litt når dei får oppgåvene, men når dei då skal prøve å endre med når du skal prøve å endre på koden så blir det jo i noen tilfelle veiledning, for noen av de begynner jo å få litt idear sjølv. Mens andre, dei som ikkje klarer å finne på noe, må vi styre litt. Så det er jo ein kombinasjon der. Når det gjelder disse oppgåvene i Scratch, så får de jo ei oppgåve, de må få ei litt konkret oppgåve. Hvis ikke, så sitter de enten bare å roter, eller så finner de noe som er ferdig. Og seier at de har gjort det. Men når de da har kommet i gang så blir de jo en veiledning, og kanskje og type sånn kunne du gjort noe sånt eller, ja. (informant 4).

Jeg ville også finne ut hvordan de vurderte arbeidet til elevene underveis og hvordan lærerne stiller seg til å bruke karakterer i valgfaget programmering.

Surveyen viser at det er et flertall av lærerne som er negativ til bruk av vurdering med karakterer i faget, men vi ser også at det er en del som har svart «vet ikke». Som figur 15 viser, vurderer lærerne elevene på flere måter, både gjennom prosjektarbeid, individuelle arbeider og innsats i timene.

15. Hvordan stiller du deg til karakterer i faget

[Fleire detaljar](#)

	Svært positivt	0
	Positivt	13
	Vet ikke	7
	Negativt	15
	Svært negativt	7

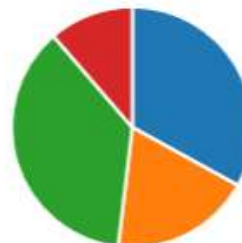


Figur 17 – utklipp fra survey

16. Hvordan vurderer du elevene? Du kan krysse av på flere alternativer

[Fleire detaljar](#)

	Prosjektarbeid	35
	Innsats i timene	20
	Individuelt	39
	Grupper	12



Figur 18 – utklipp fra survey

Slemmen skriver at: «Karakterer bør derfor suppleres med begrunnelse for og veiledning om hvordan eleven kan bli bedre i faget», (Slemmen, 2010). og informantene viser at selv om karakterer ikke bare er positivt, foregår det vurdering og rettleidelse i klassene. Informant 4 sier det slik:

Det som fungerer litt var jo at. Både i fjor og i år så brukte jeg animering av eventyr som på en måte en stor individuell oppgave på slutten der den teller mye. Da er det måten du jobber på, hva du kommer frem til, resultatet, altså diskusjonen med meg. Det blir vurdert. Så kommer vi fram til.. Den type vurdering kan være greit! For de vil

jo gjerne vite noe om det. Og så ser jeg at for eksempel når vi holder på med Arduino Micro: bit der de bygger ting og sånt, så kan det og være dette med, altså basisen er jo det å klare å koble opp og få det til å virke basiskoden. Skal de da ha en femmer, så må de da på en måte kunne endre på, komme med ideer, og da mener jeg at ideen skal vurderes hvis en må sette karakter. Vil ikke ha prøver i praktiske fag. Der går grensen, akkurat som jeg vil ikke ha prøver i sløyd heller. Nei, ja.. Sånn at det jeg gjorde i fjor var jo det at, noen timer så hadde jeg laget meg et ark, sånn type fra Excel, med navnet til venstre og litt sånn stor rute. For da får du 15 linjer ned over på et ark hvor du kan skrive litt. Og så, i noen timer tok jeg litt notater på elver. Så i løpet av noen timer hadde jeg tatt notater på alle elever. Som jeg brukte å sette karakterer underveis. (informant 4).

Informant 2 sier det slik:

Dei tar det som ein slags konkurranse på slutten. Eg har prøvd å dra inn ein eigen eller sånn kvarandre vurdering, altså «*peer assessment*» på slutten av dette. Akkurat den biten har ikkje fungert så bra, dei er så gira, dei er så glade for å sjå roboten sin og sjå dei andre at å skrive tilbake meldingar til de andre har ikkje funket så bra. Men som ein praktisk, som ei oppgåve som testar praktiske løysningar og kreativiteten og kva dei eigentleg klarer innanfor løyningane. For det er ikkje nokon fasit, fordi det er ikkje nokon fasit på sjølve oppgåva. Den fungerer.

I utgangspunktet så skulle jeg ønske at det ikke var karakter, rett og slett at det var bestått ikke bestått, og at det gikk mer på oppmøte og deltakelse på oppgavene fordi det er vanskelig å sette karakter på at et fag som er helt nytt og som er har såpass få kompetansemål i læreplanen. Ja, men jeg også jeg ser jo på sånne ting som at de har fylt i rammene jeg har satt opp, at de kanskje har gjort litt mer enn bare det jeg har sagt at jeg absolutt må gjøre. Også er det progresjonen som teller mest føler jeg, fordi jeg ikke tar aldri noe gjennomsnitt av oppgaver eller noen sånne ting. Det hadde jo blitt tull så progresjonen absolutt og så tenker jeg vipper heller opp, enn å vippe ned. Jeg vipper aldri ned. Det gagnar ingen, synes jeg. (informant 1).

En annen sier det slik:

For dette er som eit språk, du kan komme inn i timen og allereie kan du kode og du kan komme inn utan noen som helst erfaringar, då skal dei ha like gode muligheter til å klare en god karakter. Eg synes det er vanskelig med og vurdere koding som valfag, derfor må oppgåvene være veldig opne med forskjellige løsningsforslag. Det er ikkje berre eit spel, men det skal vere noko som skal teste ferdighetene, ikkje berre ein type resultat. Eg, kompetansemålene i valfag, i alle fall for koding er veldig ambisiøse. Så eg må prioritere kva kompetansemål er verdt meir enn andre for å seie det sann. Eg veit ikkje om det er rett, men det det eg må gjøre. Så eg er kanskje litt snill med noen karakterar, men eg meiner at nokon som kjem inn og viser, og som aldri har koda før, og får til eit godt resultat skal få det, og innsats teller og ja. Ja, det er eit av kjenneteikna til valfaga, innsats skal telle. (informant 2).

En sier det på denne måten:

Jeg syntes det er litt viktig for at da har du en markør som viser at disse tingene må være med i oppgave, da er du kjempegod. Og disse tingene gir middels, og dette gir på et lavt nivå. Så det er på en måte litt viktig for at de skal forstå hva som er nødvendig i akkurat den oppgaven. Kanskje mer enn i et annet fag. Og da har jeg punkter som de krysser ut etterhvert som de har gjort. Og det fungerer bra. For da er de aldri i tvil om hva er det jeg egentlig skal gjøre. Eller hva er det jeg skal lage? Og hva slags resultat skal jeg få ut av dette. Og så er det jo det at da er det lettere for meg og så si det at dette er bra. Dette er kjempegod oppgave. Og når de har fremvisning på tavla så tar jeg en generell oppdatering der jeg sier at her har Per med det og det og det. Som dere har krysset av. Men han har i tillegg gjort det, og han har funnet ut det, og han har med det. Dette her er ei sekser oppgave, for her har han funnet ut ting som faktisk vi ikke hadde tenkt å ha med i oppgave. Så dette er kreativt. Dette er flott. Og så designet. Hvordan ser dette ut? Ser dette ryddig ut? Er det lett manøvrere. Er det selvinstruerende osv.

Synes elevene det er øk med karakterer? «Ja, det syntes de. Og de vet jo om de gjør det de skal eller ikke. På en måte». (informant 3).

Er det sånn at de får vurderingsskjema, at de lager et vurderingsskjema hvis de får en oppgave f.eks. At de krysser av ?

Har hatt minimum av vurdering. I forhold til valgfag så føler jeg det er helt bak mål. Det er helt unyttig. I tillegg så , ja, altså. Jeg vil hvert fall si i programmering så, jeg hadde en prøve i fjor i nøden. På programmering ville jeg vertfall ikke hatt en skriftlig prøve i programmeringsspråk og sånne ting. (informant 4).

F.2 Hvilket nivå – og hvilke kompetansemål legges det vekt på i kodeundervisningen?

Tidligere i oppgaven¹² har det blitt sett på hvilke kompetansemål som Utdanningsdirektoratet har lagt opp til at elevene skal kunne etter endt undervisning i programmeringsfaget. Siden det er et valgfag og at det er et faktum at elevene kan velge et nytt valgfag hvert år, kan det tenkes at ikke alle elevene som velger programmering på 8.trinn vil fortsette med det på 9.trinn. Det kan også tenkes at en elev velger programmering i 10. trinn uten noen erfaring fra før. Dette kan gjøre det vanskelig å nå alle ønskelige mål i faget. (Utdanningsdirektoratet, 2018).

Saeli et al skriver at kjernen i programmeringsfaget er å lære problemløsning og å lage programmer som skal hjelpe til med dette. De hevder videre at det er to ulike kunnskapsnivåer innen programmeringsfaget; det å lage et program og det å forstå programmet. (Saeli, 2012). Når jeg spurte informantene mine om hvilke kompetansemål de fokuserte på, var det vanskelig å få klare svar her. Dette kan tyde på at det er for mange mål, at lærerne ikke har fullstendig oversikt over målene, eller at lærerne rett og slett jobber ut fra egen kunnskap og erfaringer uten å referere til målene i kunnskapsløftet i det hele tatt. Det kan også tenkes at jeg ikke var presis nok i

¹² Kapittel 4.2.3

intervjusituasjonen og jeg ser i ettertid at jeg kunne gått mer inn på de enkelte målene i faget for å få bedre og mer utfyllende svar.

Informantene mine svarer slik på spørsmålet:

Ja, eg meiner ein bør, me må satse meir på fordjuping i færre kompetansemål. Eg har lyst at dei skal kunne eit eller anna språk eller metode, og lære det skikkelig og ikkje berre mekanisk kva ord må skrivast inn, men korleis det fungerer og i kva rekkjefølgje. Dei må tenke, de må klare å tenke algoritmisk og utvikle, liksom, ferdigheter som er overførbart til andre ting. Det er derfor eg jobbar, for å seie det sånn eg jobbar mykje med visuelt basert koding. Og i min erfaring, eg har satsa mest på dette, og når vi går over til tekstbasert koding, som Python, eg bruker lite tid på dette. (informant2).

Hm, ja, skal vi se. Det er vel det å kunne noen programmeringsspråk. Forstå litt sånn hva skal vi bare si, logikken av hvordan programmer og nettsider er bygget opp. Men også det å, altså læringsmål, altså kompetansemålene. Det å kunne litt Basic programmeringsspråk. Det er jo det hovedmålet, altså det jeg fokuserer på. (informant 1).

Jo, jeg tenker. Ja, de er luftige, men samtidig syntes jeg det er nødvendig, fordi vi får et mer og mer teknologisk samfunn og bare du skal, holdt på å si, se på TV, så har du programmert de ulike kanalene på de ulike knappene, så dette er noe som er framtiden vår. Og som det er nødvendig å kunne litt om. Og, og i kunnskapsløftet tar de jo opp det og sidestiller det med, holdt på å si, andre ferdigheter da. Så jeg syntes jo at. Eller jeg er i alle fall glad for at det er et valgfag. For hvis de skal presse og dette inn i matematikk eller naturfag, så vi har Excel, og all den tiden vi bruker på det så tror jeg det vil gå ut over den tiden den vi har i faget. Ja, og det. Og det er såpass tidkrevende og på en måte sette seg inn i det, og få ting til, så jeg personlig mener at det bør være et, eller et eget fag da. (informant 3).

Nei, jeg tenker. Dette å forstå. Dette med at koding, programvare, ligger bak alle de tingene de bruker. F.eks. for når vi har Arduino så snakker vi om at dette er en mikrokontroller. Det sitter en sånn i vaskemaskinen hjemme og det sitter en sånn i

mikrobølgeovnen hjemme, sant? Når vi f.eks. i Arduino bruker en avstandssensor så er det lett og så prate: Ja, det finnes jo en sånn i støtfangeren i bilen nå. Den er og programmert sånn som du gjør med denne. Den sammenhengen, sånn at de på en måte ser at, det er den veien verden går og at det er på en måte og dette med at nå programmerer vi ved hjelp av disse lysene til å blinke, den lyden til å komme. Det programmet som lager noe annet, det er bare lengre og mer komplisert. Og kanskje og å øve opp. Dette med å øve opp tankegangen. Dette å ha tanker i flere ting. Altså algortimetankegangen. (informant 4).

Som vi ser av svarene, er det vanskelig å identifisere akkurat hvilke mål lærerne jobber med i programmeringsfaget, men det er stort sett enighet at det er viktig å lære elevene et kodespråk de kan få bruk for senere selv om de erkjenner at det elevene lærer i grunnskolen nok er utdatert når elevene kommer ut i jobb.

Når det gjelder nivået på undervisningen vil den være forskjellig ut fra lærerens kunnskap og elevenes ståsted. Saeli et al sier av flere undersøkelser viser at programmering er vanskelig å mestre, og at det er både motivasjon og manglende teknisk forståelse som spiller inn her. (Saeli, 2012). Elevene kan stille spørsmål om hvilken nytte det å kunne programmere kan ha for den enkelte, samt problemer med å forstå syntaks og semantikk. Som nevnt tidligere kan selve strukturen ved valgfaget i den norske skolen også ha en innvirkning, da lærerne i utgangspunktet kan få helt nye elever hvert år, eller de kan få en blanding av elever med erfaring og elever som er nybegynnere. Organiseringen av undervisningen på den enkelte skole spiller også inn her, har man en klasse pr trinn eller blandes trinnene i en klasse. Jeg diskuterte dette med nivå på undervisningen med informantene mine og de svarte slik:

Til en viss grad. I starten av undervisningen så hadde vi en undersøkelse om hvor mange som hadde jobbet med programmering tidligere. Og da var det noen som hadde hatt valgfaget før og de fikk mye friere tøyler enn de andre. Men alle sammen har fulgt et opplegg underveis og vi har startet på Scratch og om jeg har sett at dette kan jo egentlig enkelte av elevene veldig godt, har de fått lov til å jobbe med egne prosjekter eller lage større, ikke begrense seg så mye, så de er veldig selvgående. Derfor har de valgt valgfaget på nytt da. Det er jo slik at jeg er helt ny på dette og hadde ikke noe

erfaring da jeg startet, så da brukte jeg elever som kunne komme opp å vise eksempler for å vise; «*hvordan vil du skrevet dette hvis du skulle satt opp en enkel html kode*» for eksempel også går de gjennom og viser selv for klassen. Og da får de jo en del mestringsfølelse og det synes jeg er greit. (informant 1).

Eg pleier å lage opne oppgaver slik at det trengs ikkje forkunnskaper i mine kodetimer til å delta på lik linje med de andre, men de har nok, eg får veldig mange gamere, eg kan seie det sånn, elever som er mykje på dataen og dei er, dei har, dei drømmer til å lage det nye skytespillet, det nye «Call of Duty» eller det nye «League of Legends», så resultatet er ofte et spill. Spesielt i begynnelsen bare fordi det er det dei ønsker å gjere. Men, etterkvart så blir det, så begynner dei å gå i ulike retningar. Ja, og det er mykje prøving og feiling. I alle fall i dei to åra eg har hatt koding. Det kommer eit punkt der de byrjar å snakke med kvarandre og retter opp neven og spør meg og kvarandre, men dette kjem litt seinare. Heldigvis og, det kan diskuteras om det er bra eller ikkje, men eg likar det, at det er mykje prøving og feiling før de går og hentar svaret hos en annen elev. Ja, så først er det, det er meir feiling hos nokon, men det er læring heile vegen. (informant 2).

Ja. Fordi at sånn som åttende så er jeg helt nødt for å kopiere og ha en oppskrift hele veien. For at de skal klare å vite hva de skal gjøre. Fordi jeg kan ikke, og det er ikke tid, å gå og hjelpe hver enkelt. Så det må være veldig sånn. Mens i niende så, så har jeg mer kunne sette de i gang også har de vært enda flinkere å samarbeide og da har jeg lagt opp litt annerledes. Men det er jo, i og med at jeg er én voksen på så mange så har jeg vært nødt for å ha selvinstruerende undervisningsopplegg da. Og så må du være kjempe konkret, hvis ikke blir det bare tull. Så du må være veldig konkret. «*Hva er det dere skal lage nå?*». Og så har jeg det på ark, og så huker de av når de har gjort oppgave A, og så huker de av når de har gjort oppgave B. Og så gir de ett vink, og så kommer jeg og så tester jeg programmet de holder på med til enhver tid. Og så samarbeider de. Så de sitter i grupper på to eller fire. Litt etter hva type ting de jobber med og da skal de spørre gruppa. Så da har jeg alltid en flink som jeg vet kan ta litt ansvar og som er god å samarbeide. Og så har jeg litt forskjellige nivå ned igjennom. Og da kan de og hjelpe hverandre. (informant 3).

Nei, vi, eg prøver å finne, eg prøver å finne litt gode oppgåver, for det som viser seg med disse om ikkje har nokon erfaring frå før, dei er ikkje i stand til å skrive kode frå starten av. Derfor så finner eg gjerne oppgåver der dei har ein kode dei skriv inn, og så er på ein måte mye av oppgåva det òg, for det første å få det til å virke. (informant 4).

Jeg forsøkte også å finne ut om programmering kunne være et verktøy for elevene på den måten at de kan utvikle problemløsning til bruk for andre fag, og om elevene kunne utvikle kritisk tenking ved å jobbe med faget.

Yizhou Qian og James D. Lehman fant i sin studie at matematikk-kunnskapene til elevene hang sammen med hvor flinke de var å programmere, noe som var forventet ut fra tidligere studier. Det som var uventet var sammenhengen mellom engelskkunnskapene deres og programmering. (Lehman, 2016). Det er verdt å notere seg at elevene i denne studien var flinke med over gjennomsnittet gode engelskkunnskaper fra før. José- Manuel Sàez-Lopez et al fant at programmering var med på å hjelpe elevene:

«To understand the elements of logic, mathematics and content creation in art and history». (José-Manuel Sàez-Lòpez, 2016).

Ikke alle er like optimistiske. Kalelioğlu skriver at: «According to the results, programming did not cause any difference in the reflective thinking skills towards problem solving». (Kaleliogu, 2015).

Han fortsetter å konkludere med at det var heller ingen merkbar forbedring i elevenes måte å løse problemer på i andre fag. Denne undersøkelsen ble gjort på barn i fjerde klasse, noe som kan være med på å forklare funnene. Kan det være at elevene får en bedre effekt av kodeundervisningen når de blir eldre og har utviklet bedre kognitive evner? En studie utført av Jesus Moreno – Leòn et al fant at sjetteklassinger viste en merkbart akademisk forbedring sammenlignet med andreklassinger. De fant også at programmering hadde en mer positiv effekt på samfunnsfagsstudier enn matematikkstudier. (Jesus Moreno - Leòn, 2016).

Nevertheless, the effect size was twice as big in the social studies as in math. Our conclusion is that the bigger effect size is due to a better adjustment between cognitive and motivational factors, as learners in social studies showed higher levels of enjoyment and confidence in self – learning. (Jesus Moreno - Leòn, 2016). Forskjellen mellom andreklassingene og sjetteklassingene mener forfatterne er at de eldre elevene er kognitivt mer modne, noe som gjør at der blir lettere å overføre læring fra koding til andre fag.

Mine informanter jobber alle i ungdomsskolen og har elever mellom 12 og 16 år. De er noe uenige om alder har noe å si for om kodekunnskap kan bidra til økt mestring, men de tenker likevel at programmering kan være med på å utvikle kritisk tenking og forbedre kognitiv forståelse for problemløsning i andre fag. Jeg ser også at jeg nok ikke har vært helt klar i spørsmålsformuleringen da svarene kan være noe vanskelige å tolke.

Det å feilsøke. Finne løsninger selv. Bli mer selvstendige. Være mer kritiske til hvor de får informasjonen sin fra. Det ser jeg på som en mangelvare ellers i skolen. Jeg tror at gjennom koding så vil de altså, hvis de blir utfordret litt på å finne løsninger selv. Å bli presset litt på det. Sikkert med litt veiledning på hvor det kan være lurt å søke og litt sanne ting. Ja, det tror jeg er en veldig, veldig god egenskap å ta med seg videre. Men jeg blir jeg blir stadig vekk overrasket over at de ikke har noen begreper om kilder eller noen ting. (informant 1).

Kreativitet, utholdenhet fordi dei kjem til å feile veldig mange gonger. Algoritmisk tenking og forstå at den du snakker med eller snakker til ikke alltid er, eller forstår hva du vil. Du må være presis, at alt må komme i rett rekkefølge, at det ikke bare er greit nok, men den av og til så må du sette ting i rett rekkefølge fra det skal fungere. Dei lærer i alle fall at av og til kan fasiten vere feil, og dei må finne ei eiga løysing. Matte er en ting, naturfag er en annen, testing av en hypotese for eksempel. Dei er vant til å, når vi jobber med forsøk eller praktisk arbeid i naturfag er dei vant til å følge ei oppskrift . Steg en gjør sånn, steg to gjør sånn. (informant 3).

Det var nok litt dei som hadde valt det, men eg følte kanskje litt dette med modningsmessig så spriker det sånn, at det var vanskeleg. Ja, eg vil foretrekke trinnvis, nettopp i forhold med det at skjer så veldig mykje modningsmessig. Og eg ser og at det å lage animasjoner i Scratch, det tror eg kunne egnet, og det har jeg sett på kurset, kunne egnet seg veldig godt på mellomtrinnet. Og, og, hvis de da har hatt det litt på mellomtrinnet kunne de gjerne fått oppgavene der hos oss. Og det vil jo komme inn under flere fag. Dette med å øve opp tankegangen. Dette å ha tanker i flere ting. Altså algortimetankegangen, eg er ikke helt sikker når det gjelder problemløsning for de fleste elevene. Eg tror det er litt for vanskelig? Eg tror kanskje der at vi må fokusere på , eg ser i matematikken så tror eg vi har nok med problemløsningen. De regner mot oppgaven. Altså uten nødvendigvis verktøy. Eg tenker problemløsning er det som skjer før du bruker verktøyet. (informant 4).

Det er interessant å se hva man tenker om kognitiv utvikling og overførbarhet til andre fag ved læring av programmering, og i ettertid ser jeg at dette området nok kunne vært en større del av studiet.

8. Konklusjon og veien videre

Målet med denne studien var å finne ut hvordan skoler og lærere organiserer og jobber med valgfaget programmering sett i lys av formål og læringsmål i faget. Gjennom arbeidet med oppgaven ser jeg at det er flere aspekter jeg kunne jobbet med, samt at jeg ser at problemstilling og forskningsspørsmålene kunne med fordel vært mer gjennomtenkte.

Problemstilling:

Hvilke ulike praksiser eksisterer for faget programmering i ungdomsskolen, og hvilke innholdskomponenter legger lærerne vekt på i utformingen av faget.

Forskningsspørsmål 1: Hvordan organiserer lærerne faget?

Jeg har brukt både kvalitative intervjuer og en kort kvantitativ survey for å finne svar på dette spørsmålet. Skolene har valgt mange ulike løsninger på dette. Noen har aldersblandede klasser, mens andre foretrekker skille trinnene da de tenker at det er en

del modningsforskjeller mellom de yngste og de eldste elevene på ungdomsskolen. De fleste skolene har bare en kodelærer, men det er noen få større skoler som har to. Klassestørrelsen varierer også ganske mye. Noen har klasser med opptil 30 elever, mens andre har mindre grupper. De fleste lærerne rapporterer at de foretrekker mindre klasser, da elevene ser ut til å trenge en del veiledning, særlig i starten. Det viser seg også at de fleste lærerne trives som kodelærer. Tabellen nedenfor gir en viss indikasjon på dette.

10. Trives du som lærer i koding?

Fleire detaljar

● Veldig godt	17
● Godt	18
● Hverken eller	6
● Mindre godt	1
● Trives veldig dårlig	0



Figur 19 – utklipp fra survey

Også informantene fra dybdeintervjuene indikerer at de liker å ha faget.

Og så har jeg tidligere undervist i valgfag design og redesign, men det måtte jeg droppe da til fordel for programmering. Og det er jeg veldig fornøyd med. Kjempedeilig å ha.

Så det er et fag du har lyst å fortsette med videre?

«Hm, absolutt! Ellers hadde jeg jo ikke tatt den i utdanningen tenker jeg». (informant 1). De tre andre svarer kun ja, de liker faget uten å utdype noe mer.

Alle de fire lærerne i den kvalitative delen av undersøkelsen var den eneste kodelæreren på skolen og de fleste savnet noen de kunne planlegge og diskutere opplegg og vurdering sammen med. Et annet aspekt med å være alene om faget er problemet med å finne vikar om han/hun er borte, da de opplever at mange av kollegene ikke føler seg trygge nok i IKT til å ha faget.

Det hadde vært kjempebra hvis vi hadde vært to. Men jeg har også en tidligere kodelæreren jeg kan spørre om ting, samt at mannen min er programmerer noe som hjelper litt grann, hehe. (informant 1).

Ja, det vanskelig å finne vikar om eg ikkje er på jobb, det er mange som, det er skummelt. Me snakka om det i starten at nokon har, nokon er ikkje trygg på pc-en og det er vanlige ting, vanlege bruk av pc-en, så med ein gang vi går inn på koding, eller ordet koding eller programmering blir brukt, så blir dei redd. (informant 2).

Det blir spennende å se hva Utdanningsdirektoratet konkluderer med når det opprinnelige prosjektet er ferdig våren 2019. Skal det fremdeles være et valgfag, skal det bare være et tilbud for ungdomstrinnet og videregående skoler, eller skal man innføre koding allerede på barnetrinnet slik de har gjort i flere andre land vi liker å sammenligne oss med?

Forskningsspørsmål 2: Hvilket nivå – og hvilke kompetansemål legges det vekt på i kodeundervisningen?

Lærerne i denne studien er enige om at læringsmålene i programmeringsfaget er mange og at de spenner for vidt. Når man tenker at dette er et valgfag, og man har i snitt to timer i uken, er det mye som skal gjennomgås. Dette er noe som er diskutert under forskningsspørsmål 2 i denne oppgaven.

Rektorer og lærere har lenge gitt signaler om at dagens læreplaner er for omfattende og at man ikke får tid til å gå i dybden i fagene. Regjeringen Stoltenberg satte i 2013 gang et arbeid ledet av professor Sten Ludvigsen som skulle se på grunnopplæringens fag opp mot kompetansekravene i et framtidig samfunns – og arbeidsliv. (Det Kongelige Kunnskapsdepartement, 2016).

Ludvigsen-utvalget la fram sin sluttrapport i 2015 og det arbeides nå med en fornyelse av innholdet i skolefagene. Dette arbeidet startet i 2016 og dette blir den største forandringen i den norske skolen siden Kunnskapsløftet. Et viktig begrep her er kjerneelement; altså det viktigste og mest sentrale elevene skal lære i hvert enkelt fag.

Kjerneelementene skal gi prioriteringer og være retningsgivende for de nye lærerplanene som skal lages, og kjerneelementene er allerede identifisert og fastsatt i de fleste fagene. Det er verdt å merke seg at man ikke har identifisert kjerneelementer innenfor valgfagene ennå. (Regjeringen.no, 2018). De nye planene skal være klare i 2020, og det blir spennende å se om man også vil ta tak i valgfagene før arbeidet er ferdig.

IKT er et fagfelt i rivende utvikling. Ny teknologi og nye bruksområder er noe som kjennetegner fagfeltet. Dette vil igjen påvirke hvordan faget skal implementeres i skolene i framtiden. Skal programmering fortsette som et valgfag i ungdomsskolen og videregående skole eller skal faget inn som et eget obligatorisk fag? Skal programmering bakes inn i de fagene vil allerede har, og i så fall hvilke fag vil det være naturlig å begynne med? Skal man begynne med programmering på barnetrinnet? Hvordan skal man gjøre dette? Hvem skal være lærere? Kommer programmering til å være en obligatorisk del av pensum for framtidens lærerstudenter? På hvilket nivå skal man legge seg? Hva med mangelen på jenter innenfor IKT faget? Hvordan skal man oppdatere og fornye PC-parken og nødvendig hardware og hvem skal finansiere dette? Kan man alliere seg med næringslivet? Dette er bare noen av spørsmålene som kan være spennende og interessant å jobbe med videre innenfor temaet koding i skolen.

En av mine informanter sier det slik på spørsmålet om programmering bør bli eget fag i den norske skolen:

Ikkje enno nei, eg meiner, fordi me ligger litt bak på noen ting om for eksempel bare digitale ferdigheter me allereie har på kompetansemålene. Eg trur me må bli flinkare til å ta disse på alvor før me opprettar koding som eige fag som er avhengig av desse digitale ferdigheitene. På sikt om kanskje 5 – 7 år, ja. Kanskje det, kanskje nokon synest det vert for lenge å vente, men eg meiner om alle elever skal innom koding som eige fag no, så blir det veldig ulikt fra skule til skule. (informant 2).

Som nevnt tidligere i oppgaven er det mange ting jeg kunne gjort annerledes, og gjennom denne prosessen har jeg oppdaget at det er andre elementer innenfor faget jeg nok har mer interesse for. Her kan jeg nevne kjønn – det vil være spennende å se utviklingen her. Hvorfor er ikke jentene på banen ennå og hva kan vi gjøre for å forandre dette? I den forbindelse vil jeg nevne at jeg skal være kodelærer på min skole dette skoleåret og jeg har allerede hatt mitt første treffpunkt med elevene. I år er vi 35 elever i kodeklassen, hvorav fire er jenter. Bare to av disse hadde koding som førstevalg.

Jeg lar informantene mine få det siste ordet:

Nei, jeg har vel fire jenter av de 29, så det er, men det er flere jenter enn i fjor. Og en av, eller når jeg presenterte valgfaget for elevmassen eller elevgruppen så, så viste jeg en av de videoene til IKT-Norge som går på at jenter også bør involvere seg i teknologi eller teknologofag, det er ikke noen grunn til at vi ikke skal søke selv om vi ikke kan det og det er sånn typisk jentegreie. (informant 1).

«Eg hadde ei jente det første året, eg har ingen i år og det er litt synd». (informant 2).

«Ja, I niende har jeg to jenter. I åttende ingen». (informant 3).

«Vi har hatt veldig få jenter, i år har eg berre to jenter. I fjor var det vel det samme». (informant 4)



Bibliografi

- al, A. S. (2016). *Teknologi og programmering for alle*. Oslo: Utdanningsdirektoratet.
- Anders Sanne, O. B. (2016). *Teknologi og programmering for alle*. Oslo: Utdanningsdirektoratet.
- Anja Balanskat, K. E. (2015). *Computer programming and coding, Priorities, school curricula and initiatives*. Brussels: European Schoolnet, (EUN Partnership AIBSL).
- Arki, G. K. (2016, juni 15). The influence of game-based programming education on the algorithmic thinking. *ScienceDirect*, s. ScienceDirect.
- BBC. (2018, juli 31). *Bitesize*. Henta frå Introduction to computational thinking: <https://www.bbc.com/education/guides/zp92mp3/revision/2>
- Befring, E. (2007). *Forskningsmetode med etikk og statistikk*. Fagernes: Det norske samlaget.
- Befring, E. (2015). *Forskningsmetoder i utdanningsvitenskap*. Latvia: Cappelen Damm AS.
- Befring, E. (2016). *Forskningsmetoder i utdanningsvitenskap*. Latvia: Cappelen Damm as.
- Beitland, K. (2015). *Norske skolelever skal lære koding*.
- Bjørnevoll, E. (2016). Hva kan skolene lære av kodeklubbene? Stord, Norge.
- Bjørnevoll, E. (2016). *Hva kan skolene lære av kodeklubbene? (masteroppgave)*. Stord: Høgskulen på Vestlandet.
- Black, P. o. (1998). *Inside the Black Box - Raising standards through classroom assesment*. London: King's College (School of Education).
- Bocconi, S. C. (2018). *The Nordic approach to introducing Computational thinking and programming in compulsory education*. Italy: Nordic@BETT2018 Steering Group. <https://doi.org/10.17471/54007>.
- Bringsværd, T. Å. (1968). *Kodèmus - eller: Datamaskinen som tenkte at hva faen*. Oslo : Gyldendal.
- Brodersen, R. B. (2013). *Tekstens Autoritet*. Oslo: Universitetsforlaget.
- Channel 4. (2018, mars 25). *Black Mirror*. Henta frå Channel 4: <http://www.channel4.com/programmes/black-mirror/>
- Charmaz, K. (2014). *Constructing Grounded Theory*. London: Sage Publications Ltd.
- Claxton, G. W. (2002). *Sociocultural Perspectives on the Future of Education*. Cornwall, Great Britain: Blackwell Publishers.
- Cochran, K. F. (1997). <https://www.narst.org>. Henta frå <https://www.narst.org>: <https://www.narst.org/publications/research/pck.cfm>

- Datatilsynet.* (2017, april). Henta frå datatilsynet.no: <https://www.datatilsynet.no/verktoy-skjema/Sporsmal-Svar/?question=6190&qlist=2300,6190&tema=9>
- Datatilsynet. (u.d.). *Datatilsynet.* Henta frå Spørsmål og svar: <https://www.datatilsynet.no/verktoy-skjema/Sporsmal-Svar/?question=6190&qlist=2300,6190&tema=9>
- Department for Education UK. (2013, September 11). *GOV.UK.* Henta frå National curriculum in England: computing programmes of study: <https://www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study/national-curriculum-in-england-computing-programmes-of-study>
- Det Kongelige Kunnskapsdepartement. (2008). *Kvalitet i skolen.* Oslo: Det Kongelige Kunnskapsdepartement.
- Det Kongelige Kunnskapsdepartement. (2016, April 15). *Stortingsmelding 28 Fag – Fordypning – Forståelse En fornying av Kunnskapsløftet.* Henta frå Det kongelige Kunnskapsdepartementet: <https://www.regjeringen.no/contentassets/e8e1f41732ca4a64b003fca213ae663b/no/pdfs/stm201520160028000dddpdfs.pdf>
- Difference between.net. (2018, mars 13). *Difference between.net.* Henta frå Difference between Coding and Programming: <http://www.differencebetween.net/technology/difference-between-coding-and-programming/>
- digi.no. (2016, mars 8). *Kvinneandelen i den norske IT-bransjen er svært stabil.* Henta frå digi.no: <https://www.digi.no/artikler/kvinneandelen-i-den-norske-it-bransjen-er-svaert-stabil/348237>
- dn.no. (2015). Henta frå <https://www.dn.no/nyheter/politikkSamfunn/Utdannelse/it-bransjen-kraver-koding-inn-i-skolen>
- Droumeva, J. J. (2016, Volume 14 issue 2). Exploring Media Literacy and Computational Thinking: A Game Maker Curriculum Study. *The Electronic Journal of E-learning*, ss. 114 - 121.
- Dysthe, O. (2001). *Dialog, samspel og læring.* Oslo: Abstrakt Forlag as.
- Dysthe, O. (2012). Teoretiske perspektiver på dialog og dialogbasert undervisning. I B. o. Dysthe, *Dialogbasert Undervisning* (ss. 45 - 79). Bergen: Fagbokforlaget.
- Everett, E. L. (2012). *Masteroppgaven Hvordan begynne og fullføre.* Oslo: Universitetsforlaget.
- Finnish National Agency for Education. (u.d.). *Finnish National Agency for Education.* Henta frå New national core curriculum for basic education: https://www.oph.fi/english/curricula_and_qualifications/basic_education/curricula_2014
- Finnish National Agency for Education. (u.d.). *Finnish National Agency for Education.* Henta frå Background of the WBL-TOI Manual project: https://www.oph.fi/english/education_development/quality_assurance_and_evaluation/wbl-toi/manual/background_of_the_project

- Freiman, T. D. (2016, oktober 11). Task-based Assessment of students' computational thinking skills developed through visual programming or tangible coding environments. *13th International Conference on Cognition and Exploratory Learning in Digital Age*, ss. 41 - 51.
- Grout, V. a. (2014). Taking Computer Science and Programming into Schools: The Glyndŵr/BCS Turing Project. *ScienceDirect*, 680 -685.
- Gundem, B. B. (1990). *Læreplanpraksis og læreplanteori : en introduksjon til læreplanområdet*. Oslo: Universitetsforlaget.
- Haaland, Ø. e. (2011). *Pedagogikk for en ny tid*. Vallset: Oplandske Bokforlag og forfatterne.
- Haaland, Ø. H. (2011). *Pedagogikk for en ny tid*. Vallset: Oplandske Bokforlag og forfattere.
- Hattie, J. (2009). *A Synthesis of over 800 Meta-analysis Relating to Achievement*. Abingdon/New York: Routledge.
- Hattie, J. (2012). *Visible Learning for Teachers - Maximizing Impact on Learning*. Abingdon: Routledge.
- Hiltunen, T. (2016, 15 mai). Learning and Teaching Programming Skills in Finnish Primary Schools. Oulu, Finland.
- (u.d.). <https://code.org/>.
- Humphreys, S. e. (2014, Oktober). *Codeweek.eu*. Henta frå <http://blog.codeweek.eu/post/99387218550/the-story-of-computing-at-school-from-the-uk>: <http://blog.codeweek.eu/post/99387218550/the-story-of-computing-at-school-from-the-uk>
- Husabø, I. (2015, desember). Bør kidsa kode på skulen.
- Ibrahim Ouahbi, F. K. (2015). Learning Basic Programming Concepts by Creating Games with Scratch Programming Environment. *Procedia - Social Behavioral Sciences 191*, ss. 1479 - 1482.
- James W.Pellegrino, N. C. (2001). *Knowing what Students Know*. Washington D.C: NATIONAL ACADEMY PRESS.
- Jesus Moreno - León, G. R.-G. (2016, juni 9). Code to Learn - Where does IT belong in the K-12 Curriculum? *Journal of Information Technology Education - volume 15*, ss. 283 - 303.
- Johannesen, A. T. (2010). *Introduksjon til samfunnsvitenskapelig metode (4.utgave)*. Oslo: Abstrakt forlag.
- Johannessen, E. M. (2018, juli 19). *Kvinner strømmer til teknologifag*. Henta frå NRK: https://www.nrk.no/norge/samordna-opptak-2018_-kvinner-strommer-til-teknologifag-1.14131390

- Josè-Manuel Sàez-López, M. R.-G. (2016, mars 10). Visual Programming languages intergrated across the curriculum in elementary school: A two year case study using "Scratch" in five schools. *Computers and Education*, ss. 129 - 141.
- Kaleliog̃lu, F. (juni, 2015). A new way of teaching programming skills to K-12 students: Code.org. *Computers in Human Behavior*, ss. s 200 - 210.
- Kaleliogu, F. (2015, juni 1). A new way of teaching programming skills to K-12 students:Code.org. *Computers in Human Behaviour*, ss. 200 -210.
- Kommunal - og moderniseringsdepartementet. (2013, Januar 1). *Regjeringen.no*. Henta frå 6. Digital kompetanse: <https://www.regjeringen.no/no/dokumenter/nou-2013-2/id711002/sec8>
- Kommunal – og moderniseringsdepartementet. (2015/2016). *Stortingsmelding 27 Digital agenda for Norge — IKT for en enklere hverdag og økt produktivitet*. Oslo: Kommunal – og moderniseringsdepartementet.
- Krumsvik, R. J. (2013). *Innføring i forskningsdesign og kvalitativ metode(kompendium)*. Bergen: fagbokforlaget.
- Krumsvik, R. J. (2014). *Forskningsdesign og kvalitativ metode, ei innføring*. Bergen: Fagbokforlaget.
- Kunnskapsdepartement. (2008). *Stortingsmelding 31 Kvalitet i skolen*. Oslo: Kunnskapsdepartement.
- Kunnskapsdepartementet. (2003/2004). *Stortingsmelding nr 30. Kultur for læring- En skole for kunnskap, mangfold og likeverd*. Oslo: Kunnskapsdepartementet.
- Kurland, R. D. (1984). On the cognitive effects of learning computer programming. *New Ideas in Psychology*, ss. 137 - 138.
- Kvale, S. B. (2009). *Det kvalitative forskningsintervju (2.utgave)* . Oslo: Gyldendal Akademisk Forlag.
- Larsen, C. S. (2017). *Digital Didaktikk*. Henta frå <http://digitaldidaktikk.no/refleksjon/detalj/laeringsteorier>
- Lehman, Y. Q. (2016, Mars 15). Correlates of Success in Introductory Programming: A study with Middle School Students. *Canadian Center of Science and Education* , ss. 73 - 83.
- Lær kidsa koding. (2016). *kidsakoder.no*. Henta frå kodetimen: <https://kidsakoder.no/kodetimen/>
- Manger, T., Lillegjord, S., & Nordahl, T. &. (2012). *Livet i Skolen*. Bergen: Fagbokforlaget.
- Medietilsynet. (2016). *Barn og Medier 2016*. Oslo:: https://www.medietilsynet.no/globalassets/publikasjoner/barn-og-medier-undersokelser/2016_barnogmedier.pdf.
- Merriam, S. B. (2009). *Qualitative Research*. United States of America: John Wiley & Sons, Inc.

- MIT News. (2016, august 1). *MIT News*. Henta frå Massachusetts Institute of Technology :
<http://news.mit.edu/2016/seymour-papert-pioneer-of-constructionist-learning-dies-0801>
- NTB. (2017, august 25). *Skuffet over «uforpliktende» digital skolesatsing*. Henta frå Dagens Næringsliv:
<https://www.dn.no/nyheter/2017/08/25/1353/Politikk/skuffet-over-uforpliktende-digital-skolesatsing>
- Orwell, G. (1949). *1984*. London: Penguin Publishing Group.
- Ottestad, G. T. (2014). *ICILS 2013: Norske elever har gode digitale ferdigheter*. Oslo: Senter for IKT i utdanningen/https://www.udir.no/globalassets/filer/tall-og-forskning/rapporter/2014/icils_rapport_rettet.pdf.
- Papert, S. (1993). *Mindstorms; Children, computers, and powerful Ideas*. New York: Basic Books.
- Pellegrino, J. C. (2001). *Knowing what Students Know*. Washington, DC: National Academy.
- Pierce, M. (2013). 21st Century Curriculum: Coding for Middle Schoolers. *T.H.E. Journal*, 20 -23.
- Postholm, M. B. (2010). *Kvalitativ metode - en innføring med fokus på fenomeologi, etnografi og kasusstudier 2.utgave*. Oslo: Universitetsforlaget.
- Regjeringa. (2015, August 20). *Koding blir valgfag på ungdomsskolen*. Henta frå Regjeringa.no:
<https://www.regjeringen.no/no/aktuelt/koding-bli-va-vgfag-pa-ungdomsskolen/id2435271/>
- Regjeringen.no. (2016, april 6). *Regjeringen.no*. Henta frå Regjeringen.no:
<https://www.regjeringen.no/no/aktuelt/koding-bli-va-vgfag-pa-146-skoler/id2481962/>
- Regjeringen.no. (2018, juni 26). *Kunnskapsdepartementet*. Henta frå Fornyer innholdet i skolen:
<https://www.regjeringen.no/no/aktuelt/forny-er-innholdet-i-skolen/id2606028/?factbox=factbox2606080>
- Saeli, M. (2012). *Teaching programming for secondary school : a pedagogical content knowledge based*. Eindhoven: Technische Universiteit Eindhoven.
- Senter for IKT i utdanningen. (2015). *Koding i skolen*. Bergen: Senter For IKT i Utdanningen.
- Sevik, K. (2015). *Koding i skolen*. Oslo: Senter for IKT i utdanningen.
- Seymour Papert, & I. (1991). Situating Constructionism. I & I. Seymour Papert, *Constructionism* (ss. 3 - 24). Ablex Publishing Corporation.
- Silseth, T. (2016, september 22). *IT - bransjen trenger flere jenter*. Henta frå NRK Ytring:
<https://www.nrk.no/ytring/it-bransjen-trenger-flere-jenter-1.13142519>
- Silverman, d. (2005). *Doing Qualitative research*. London: Sage Publications, Ltd.
- Skaalvik, E. o. (2015). *Motivasjon for læring*. Oslo: Universitetsforlaget.

- Slemmen, T. (2010). *Vurdering for læring i klasserommet (2.utgave)*. Oslo: Gyldendal Norsk Forlag AS.
- Sletnes, K. B. (2015, mai 12). *Positivisme - vitenskapsteori*. Henta frå Store Norske Leksikon:
https://snl.no/positivisme_-_vitenskapsfilosofi
- Solerød, E. (2012). *Pedagogiske grunntanker*. Oslo: Universitetsforlaget.
- Solerød, E. (2012). *Pedagogiske grunntanker - i et dannesperspektiv*. Oslo: Universitetsforlaget.
- Sze Yee Lye, J. H. (2014, september 30). Review on theaching and learning of computational thinking through programming: What is next for K-12. *Computers in Human Behaviour*, ss. 51 - 61.
- Säljö, R. (2001). *Læring i praksis*. Oslo: J.W Cappelens forlag a.s.
- Säljö, R. (2001). *Læring i praksis*. Oslo: J.W Cappelens forlag a.s.
- Säljö, R. (2003). *Læring i praksis, et sosiokulturelt perspektiv*. Oslo: J.W Cappelens Forlag a.s.
- The Australian Society for Evidence Based Teaching. (2017, september 24). *The Australian Society for Evidence Based Teaching*. Henta frå Hattie's 2017 Updated List of Factors Influencing Student Achievement: <http://www.evidencebasedteaching.org.au/hatties-2017-updated-list/>
- The Editors of Encyclopaedia Britannica. (2017, Mai 30). *Encyclopaedia Britannica*. Henta frå Computer-assisted instruction: <https://www.britannica.com/topic/computer-assisted-instruction>
- Thronsen, I. e. (2015). *Elevenes digitale ferdigheter og bruk av IKT i ICILS 2013*. Oslo: Universitetsforlaget.
- Utdanningsdirektoratet. (2015). *Vurdering for læring*. Henta frå <https://www.udir.no/laring-og-trivsel/vurdering/>: <https://www.udir.no/laring-og-trivsel/vurdering/nasjonal-satsing/omsatsingen/>
- Undheim, J. O. (1996). *Innføring i statistikk og metode for samfunnsvitenskapelige fag*. Oslo: Universitetsforlaget.
- Utdanningsdirektoratet. (2016, 12 14). *Utdanningsdirektoratet*. Henta frå Statistikk om grunnskolen 2016-17: <https://www.udir.no/tall-og-forskning/finn-forskning/tema/elever-og-ressurser-i-grunnskolen-2016-17/>
- Utdanningsdirektoratet. (2012, august 1). *Læreplan i valfaget teknologi og praksis*. Henta frå Kunnskapsløftet: <https://www.udir.no/kl06/TPR1-01>
- Utdanningsdirektoratet. (2015, August 25). *Utdanningsdirektoratet*. Henta frå Generell del av læreplanen: <https://www.udir.no/laring-og-trivsel/lareplanverket/generell-del-av-lareplanen/>
- Utdanningsdirektoratet. (2015, September 1). *Vurderingspraksis*. Henta frå Vurdering for læring: <https://www.udir.no/Vurdering-for-laring/>

- Utdanningsdirektoratet. (2016, August 1). *Forsøkslæreplan i valgfag programmering (PRG1-01)*. Henta frå Utdanningsdirektoratet: <https://www.udir.no/kl06/PRG1-01>
- Utdanningsdirektoratet. (2017, mai 30). *Individuell vurdering*. Henta frå Utdanningsdirektoratet: <https://www.udir.no/regelverk-og-tilsyn/finn-regelverk/etter-tema/Vurdering/individuell-vurdering-udir-5-2016/>
- Utdanningsdirektoratet. (2017, juli 7). *Programmering valgfag på ungdomstrinnet*. Henta frå Utdanningsdirektoratet: <https://www.udir.no/kvalitet-og-kompetanse/nasjonale-satsinger/real-fagsstrategien/forsok-med-programmering-som-valgfag/>
- Utdanningsdirektoratet. (2018, mars 25). *Forsøkslæreplan i programmering og modellering X - programfag i utdanningsprogram for studiespesialisering (PRM1-01)*. Henta frå Utdanningsdirektoratet: <https://www.udir.no/kl06/PRM1-01>
- Utdanningsforbundet. (u.å). *Utdanningsforbundet*. Henta frå Hva sier forskningen?: <https://www.utdanningsforbundet.no/var-politikk/stopp-store-klasser/forskning-og-fakta/stotte-i-forskningen/>
- Utdanningsdirektoratet. (2016, Juni 30). *Senter for IKT i utdanningen - nå fusjonert med Utdanningsdirektoratet*. Henta frå Utdanningsdirektoratet: <https://www.udir.no/laring-og-trivsel/lareplanverket/veiledning-lp/valgfag-programmering/stottemateriell/>
- Utdanningsdirektoratet. (2017, August 29). *Hva er fagfornyelsen*. Henta frå Utdanningsdirektoratet: <https://www.udir.no/laring-og-trivsel/lareplanverket/fagfornyelsen/hva-er-fornyelse-av-fagene/>
- Utheim, E. B. (E24) Publisert: 07:14 - 08.03.2013, Oppdatert: 07:51 - 08.03.2013). <http://e24.no/digital/vi-trenger-dere-jenter/20342993>. E24.
- Visual programming languages integrated across the curriculum in elementary school. (2016, mars 10). *Computers & Education*, ss. 129 - 141.
- Wikipedia. (2017, November 12). *Pragmatism*. Henta frå <https://en.wikipedia.org/wiki/Pragmatism>
- Wikipedia. (2018, mai 10). *Konfidensintervall*. Henta frå Wikipedia: <https://no.wikipedia.org/wiki/Konfidensintervall>
- Wormnes, B. o. (2005). *Motivasjon og Mestring: veier til effektiv bruk av egne ressurser*. Bergen: Fagbokforlaget.

Vedlegg:

Vedlegg 1 – Godkjenning fra NSD

Vedlegg 2 - Intervjuguide

Vedlegg 3 – Informasjonsskriv til informantene

Vedlegg 4 – link til survey/spørsmålsdelen

Vedlegg 5 – link til survey/svar

Vedlegg 6 – 9 – transkriberte intervjuer