



Høgskulen
på Vestlandet

MASTEROPPGAVE

Bruk av programmering i matematikkfaget på mellomtrinnet

En kvalitativ studie av læreres tilrettelegging og erfaringer med ressurser fra Super:bit.

The use of programming in mathematics at the intermediate school

Benjamin Haastø Sunde og Eivind Øen

Emne: MGBMA550. Grunnskolelærerutdanning 1.-7 og 5.-10 trinn

Fakultet for lærarutdanning, kultur og idrett

Veiledere: Frode Olav Haara og Lene Hayden Taraldsen

Innleveringsdato: 16.05.2022

Forord

Nå som vi er ferdig med en 5.årig utdanning ved HVL campus Sogndal, ser vi tilbake på en fin studietid med et bra studentmiljø og mange varierte aktiviteter. Vi er nå klare for nye utfordringer og ser fram til yrkeslivet som lærere i grunnskolen.

Etter fire år med utdanning var vi spente på hvordan ett år med masterskriving ville bli. Det virket skremmende å begi seg ut på en så stor oppgave som ingen av oss noen gang hadde sett for oss at vi skulle skrive. Selv om det har vært ett år med mye arbeid, ser vi tilbake på ett år preget av mye struktur og målsettinger. På denne måten har omfanget på oppgaven vist seg å være oppnåelig. Vi er spesielt glade for at vi har valgt å skrive oppgaven sammen to og to. Det å skrive sammen med en medstudent har krevd god planlegging og kommunikasjon kontinuerlig. I tillegg har det vært sosialt, gøy og en god motivasjon inn i skrivingen av masteroppgaven. Noen ganger har det så klart vært krevende å være to om prosjektet, men vi ser begge to på samarbeidet som et godt valg.

Vi vil først og fremst gi en stor takk til de fire lærerne som tok seg tid til å delta på intervjuene i en travel lærerhverdag. Prosjektet hadde ikke vært mulig uten dem. Videre ønsker vi å takke våre veiledere, Frode Olav Haara og Lene Hayden Taraldsen, for et godt samarbeid, samt positive og konstruktive tilbakemeldinger. Til slutt så vil vi takke medstudenter for godt samhold og sosialt på «masterrommet». Det har bidratt til at det har vært kjekt å gå på skolen.

Sogndal, mai 2022

Benjamin Haastø Sunde og Eivind Øen

Sammendrag

Etter den nye læreplanen (LK20) ble iverksatt høsten 2020 skal programmering ha fått en sentral del i matematikkfaget. Programmering som verktøy for å lære matematikk har blitt diskutert i flere kanaler og vi så på dette som en gyllen mulighet til å undersøke dette temaet. Super:bit-satsingen har hatt som mål å få programmeringsverktøy ut i skolene. Vi har derfor i denne studien undersøkt hvordan lærere på mellomtrinnet bruker dette verktøyet i undervisningen, og hvilke erfaringer lærerne har knyttet til dette. Studien har følgende problemstilling:

Hvordan legger fire lærere til rette for læring av matematikk på mellomtrinnet gjennom bruk av pedagogiske roboter fra Super:bit-satsingen, og hvilke erfaringer gjør de seg knyttet til dette?

For å besvare problemstillingen har vi benyttet et kvalitativt forskningsdesign og intervjuet fire lærere fra fire ulike skoler. Lærerne som er med i studien har blitt valgt ut ifra to gitte kriterier: (1) lærerne skulle være matematikklærere på mellomtrinnet som (2) har undervisningserfaring med Super:bit gjennom programmering. Studien bygger på teori og forskning knyttet til programmering generelt og inn mot matematikkfaget, samt aktuelle teoretiske elementer som gjør seg gjeldene ved slik undervisning.

Funn i denne studien viser at det er stor variasjon i kompetansen til lærerne som underviser i programmering i matematikkfaget. Studien peker på at dette gjør seg spesielt synlig når det gjelder interessen til lærerne, rammefaktorer knyttet til skolens organisering og lærernes tidsbruk til arbeid med programmering. Videre viser studien til lite implementering av matematikk i programmeringsundervisning. Dette begrunnet i at lærerne uttrykker at det er vanskelig med lav kompetanse på feltet. Studien viser samtidig at lærerne har mange gode erfaringer med programmering i matematikkfaget, og de gir uttrykk for at elevene kan ha god nytte av den varierte og praktiske undervisningen. I tillegg synliggjøres det at programmeringsundervisning er lett å differensiere og tilpasse til elevene. Studien bidrar til å løfte fram et nyttig bilde av hvordan lærere arbeider med å implementere nye elementer i matematikkfaget.

Abstract

Since the new curriculum (LK20) was implemented in autumn 2020, programming should now be an essential part of teaching mathematics at school. As a tool for learning mathematics, programming has been discussed in several channels, and we saw this as an opportunity to examine this topic. The Super:bit initiative is aimed to bring programming tools to pupils. Therefore, we wanted to explore how teachers use this tool in teaching and what experiences they have related to this. We have chosen to research the following problem:

How do four teachers facilitate the learning of mathematics at the intermediate level through using pedagogical robots from the Super:bit initiative, and what experiences do they have related to this?

The study uses a qualitative research design where we interviewed four teachers at their respective schools. The teachers chosen for the study have been selected based on two given criteria: (1) the teachers should be mathematics teachers at the intermediate level, and (2) have teaching experience with Super:bit through programming. This thesis is based on theory and research linked with programming, together with theoretical practices in the didactics.

Findings in the thesis indicate that the competence of the teachers who teach programming varies a lot. This affects the interest of the teachers but also the framework for the teachers related to the school's organization and the teacher's prioritization of working hours. Furthermore, the study shows limited implementation of mathematics when teaching programming, as this is difficult with low programming competence among the teachers. On the other hand, the study also shows that there are many good experiences with programming in the subject of mathematics, which express that the students can benefit from variation and practical tasks when learning. In addition, it emphasized that programming is easy to differentiate and adapt to the teaching. The thesis contributes to introduce information regarding how to implement programming in the subject of mathematics.

Innholdsfortegnelse

1.0 INNLEDNING	1
1.1 BAKGRUNN FOR VALG AV TEMA	1
1.2 PROGRAMMERING	1
1.3 SUPER:BIT	2
1.4 PROBLEMSTILLING	2
1.5 OPPBYGGING AV OPPGAVEN	3
2.0 TEORIGRUNNLAG	4
2.1 HVA ER PROGRAMMERING?	4
2.1.1 LK20 og programmering	5
2.2 ALGORITMISK TENKNING	6
2.2.1 Den algoritmiske tenkeren	6
2.2.2 Tre dimensjoner av algoritmisk tenkning	7
2.2.3 Oppsummering av algoritmisk tenkning	10
2.3 SUPER:BIT-PROSJEKTET	11
2.3.1 Micro:bit	11
2.3.2 Satsingen i Norge.....	12
2.3.3 Opplegget fra vitensentrene.....	13
2.4 SOSIOKULTURELL LÆRINGSTEORI	14
2.4.1 Den proksimale utviklingssonen	14
2.4.2 Scaffolding	15
2.4.3 Artefakter	15
2.5 DEN DIDAKTISKE RELASJONSMODELLEN.....	17
2.5.1 Elevforutsetninger	18
2.5.2 Rammefaktorer	18
2.5.3 Mål	19
2.5.4 Innhold.....	20
2.5.5 Læringsaktiviteter.....	21
2.5.6 Vurdering.....	22
2.5.7 Oppsummering didaktisk relasjonsmodell.....	23
2.6 OPPSUMMERING AV TEORIGRUNNLAG	24
3.0 METODE	25
3.1 FORSKNINGSDESIGN OG METODE	25
3.1.1 Kvalitativ tilnærming	25
3.2 DATAINNSAMLING OG VERKTØY	27
3.2.1 Kvalitativt forskningsintervju.....	27

3.2.2 Utvalg av lærere	28
3.2.3 Pilot.....	30
3.2.4 Gjennomføring.....	30
3.3 ANALYSE AV INTERVJU.....	32
3.3.1 Forberedelse- transkripsjon og koding	33
3.3.2 Koding og kobling til teoretisk rammeverk.....	34
3.3.3 Kategorisering	36
3.4 VALIDITET OG RELIABILITET.....	41
3.4.1 Intern validitet	41
3.4.2 Ekstern validitet.....	41
3.4.3 Pålitelighet.....	42
3.4.4 Nøytralitet	42
3.5 ETIKK.....	44
4.0 ANALYSE OG PRESENTASJON AV FUNN	45
4.1 HVORDAN LÆRERE LEGGER TIL RETTE FOR LÆRING AV MATEMATIKK VED BRUK AV SUPER:BIT.....	45
4.1.1 Undervisning.....	45
4.1.2 Læring av matematikk.....	50
4.2 ERFARINGER	55
4.2.1 Elevenes forutsetning og digital kompetanse	55
4.2.2 Læreres forutsetninger og digitale kompetanse.....	56
4.2.3 Erfaring fra lærerens undervisning med Super:bit.....	59
4.3 OPPSUMMERING AV FUNN.....	63
5.0 DISKUSJON	64
5.1 TILRETTELEGGING FOR LÆRING AV MATEMATIKK.....	64
5.1.1 Undervisning.....	64
5.1.2 Læring av matematikk.....	65
5.2 ERFARINGER	67
5.2.1 Læreres og elevens digitale kompetanse.....	67
5.2.2 Læringsituasjon	70
5.3 OPPSUMMERING AV DISKUSJON.....	72
6.0 AVSLUTNING	74
6.1 KONKLUSJON	74
6.2 IMPLIKASJONER	75
6.3 VEIEN VIDERE	76
7.0 REFERANSELISTE.....	77

VEDLEGG	82
VEDLEGG 1: NSD GODKJENNING AV MELDESKJEMA.....	82
VEDLEGG 2: INFORMASJONSSKRIV OG SAMTYKKEERKLÆRING.....	84
VEDLEGG 3: INTERVJUGUIDE.....	87

Liste over figurer og tabeller

FIGUR 1: DEN ALGORITMISKE TENKEREN	6
FIGUR 2: MICRO:BIT KODEBRIKKE.....	11
FIGUR 3: BIT:BOT.....	12
FIGUR 4: DEN DIDAKTISKE RELASJONSMODELLEN	17
TABELL 1: UTVALGET AV LÆRERNE I INTERVJUET, HER BESKREVET MED LÆRERNES ERFARING OG KUNNSKAP.	30
TABELL 2: DEN ANALYTISKE FREMGANGSMÅTEN OG ARBEIDSPROSESSEN I ANALYSEN.	32
TABELL 3: FARGEKODER FOR OPPMERKING AV SITAT PÅ KRYSS AV DOKUMENT.	36
TABELL 4: UTKLIPP FRA SITATENE KNYTTET OPP MOT PUNKTENE I SOSIOKULTURELL LÆRINGSTEORI.	37
TABELL 5: UTKLIPP AV EXCEL-ARK MED INNDELING AV KATEGORIER.....	38
TABELL 6: SITAT KOBLET OPP MOT KATEGORIER.....	39

1.0 Innledning

1.1 Bakgrunn for valg av tema

Siden 1980-tallet har programmering vært en del av læreplanene i Europa, men det har blitt vektlagt i varierende grad (Benton et al., 2017, s. 116). Det har de siste årene blitt en økt politisk interesse for programmering og dens plass inn i skolen i mange land (Johansen, 2020). Målet er at elever som utdannes ikke bare skal være forbrukere, men også ha en forståelse for, og kunne utvikle ny teknologi (Meld. St. 28, 2015-2016, s. 32; NOU 2014: 7, s. 113). Den teknologiske utviklingen i samfunnet har ført til at de digitale ferdighetene har fått et økt fokus (Utdanningsdirektoratet, 2019c). I 2013 kom det ut en studie fra Oxford der det blir anslått at så lenge et problem kan spesifiseres, kan og vil en datamaskin ta over for denne arbeidsplassen i framtiden (Frey & Osborne, 2017, s. 14). Morgendagens arbeidsplasser vil ha arbeidsgivere og arbeidsoppgaver som blant annet krever høyere ferdigheter innenfor digital kompetanse, problemløsning og algoritmisk tenkning (Meld. St. 28, 2015-2016, s. 32). Skolene blir derfor presset til å møte kravene om å utdanne digitalt kompetente samfunnsborgere (Johansen, 2020).

England var forholdsvis tidlig ute med programmering i skolen (Sevik, 2016, s. 22). Dette var en ringvirkning av en rapport som kom i 2011 på bestilling fra ministeren for kultur, kommunikasjon og kreative næringer. Rapporten viser til nedgangen innen produksjon av videospill og visuelle effekter i landet, og skylder på mangelen av programmeringsundervisning i utdanningsløpet (Livingstone & Hope, 2011, s. 5). Etter at England startet med programmering inn i skolen har flere Europeiske land fulgt etter. I England har de valgt å innføre programmering som et eget fag, kalt «computing». I Norge, på lik linje med Finland, har programmering blitt inkludert som en del av matematikkfaget (Sevik, 2016, s. 23–24). Målet med den store satsingen er at elevene får en forståelse for nytteverdien av programmering som et verktøy for å lære matematikk (Johansen, 2020).

1.2 Programmering

Begrepet programmering brukes om å lage programmer som en datamaskin kan utføre (Rossen, 2020; Vihovde, 2020). Dette innebærer å lage en rekke instruksjoner som skal fortelle datamaskinen hva den skal gjøre. En kan kalle det en slags veibeskrivelse for datamaskiner til å utføre sin gitte oppgave. Programmering handler derimot ikke bare om kodingen, men er en sammensatt prosess som består av å kode, løse et problem, analysere, feilsøke og samarbeide (Haraldsrud et al., 2020, s. 189). I skolen i dag er det mest vanlig å bruke blokkprogrammering framfor tekstprogrammering, fordi blokkprogrammering er en lettere måte å komme inn i programmeringen på, også ved lavere årstrinn (Lye & Koh, 2014, s. 53; Piedade et al., 2020, s. 3–4).

Siden programmering er en del av den nye læreplanen (LK20) og de siste årene har fått et økt fokus, er utviklingen i skolen med på å tilrettelegge for endringene som skjer i samfunnet (NOU 2014: 7, s. 112). I LK20 har programmering blitt en sentral del av matematikkfaget, og er lagt opp med programmering eksplisitt nevnt fra 5.trinn (Kunnskapsdepartementet, 2019). Skal en se ut ifra kompetansemålene vil programmering være en sentral del av matematikkfaget framover. I småskolen er det mulig å oppfylle kompetansemålene uten programmeringsundervisning, men vi finner likevel at læringsaktivitetene knyttet til disse kompetansemålene er rettet mot programmering på senere trinn. Elevene skal kunne utforske, lære og prøve nye ting som kan bli sentrale for dem senere i livet. Et av kjerneelementene som er sentralt knyttet til programmering er «utforskning og problemløsning». Her er algoritmisk tenkning et problemløsende tankesett utviklet gjennom programmering (Angeli et al., 2019, s. 90; Brennan & Resnick, 2012, s. 2; Wing, 2006, s. 33). Programmeringsundervisningen i matematikkfaget skal derfor bygges på algoritmisk tenkning. Internasjonal forskning viser til algoritmisk tenkning som en sentral ferdighet som utvikles gjennom arbeidet med programmering (Angeli et al., 2019, s. 86). Programmering og algoritmisk tenkning er et av temaene som er tatt inn i LK20 for å utdanne reflekterte og kritiske elever.

1.3 Super:bit

I 2016 kom BBC med en respons til mangelen på programmerere i samfunnet ved å utvikle prosjektet «Make it Digital». «Make it digital» har som mål at elever skal utvikle seg fra å være passive forbrukere til innovatører av teknologi. For å nå dette målet tilbyr BBC med prosjektet «Make it Digital», utstyr som kan programmeres med. Dette utstyret er kalt Micro:bit og er en liten datamaskin. Denne lille datamaskinen består av enkle funksjoner som kan programmeres. Flere land har fulgt etter med å bruke Micro:bit og har lagd ulike satsingsprosjekt for å få dette ut til unge. I Norge kalles prosjektet Super:bit og er gitt på oppdrag fra regjeringen og den teknologiske skolesekken, til vitensentrene, «Lær Kidsa Koding» og NRK (Moldøen, 2019). De har et mål om å få programmering ut i skolen til elevene slik at de kan få en opplevelse av mestring og forståelse knyttet til den teknologiske verden (Super:bit, u.å.a). I Norge har programmering blitt mer populært gjennom kodeklubbene «Lær Kidsa Koding» (Sevik, 2016, s. 9). Kodeklubbene har brukt blokkprogrammeringen som et enkelt verktøy for å lære unge programmering.

1.4 Problemstilling

Super:bit-satsingen i Norge har som mål å styrke programmeringskunnskapen til elever. Prosjektet har fokus på å skape en mulighet for at elever skal utvikle algoritmisk tenkning gjennom å bruke programmering som et verktøy i arbeidet med læring av matematikk (Moldøen, 2019; Super:bit, u.å.a). Vi ser derfor på Super:bit og programmerings plass i LK20 som en gylden mulighet til å

undersøke hvordan programmering blir brukt i skolen i matematikkfaget. Gjennom praksis, vikariat, forskning og tidsskrifter har vi dannet oss et perspektiv på statusen til programmering skolen. Vi opplever derimot at det er stor variasjon i hva som blir sagt og vektlagt knyttet til programmering og ønsker derfor å få et dypere innblikk i hvordan lærerne bruker programmering i matematikkundervisningen sin. I tillegg har vi et ønske om å se på hvordan lærerne selv jobber med programmering og hvilke erfaringer lærerne har med det nye emnet i LK20. Med bakgrunn i nevnt forskning og interesse ønsker vi å besvare følgende problemstilling:

Hvordan legger fire lærere til rette for læring av matematikk på mellomtrinnet gjennom bruk av pedagogiske roboter fra Super:bit-satsingen, og hvilke erfaringer gjør de seg knyttet til dette?

1.5 Oppbygging av oppgaven

Ut ifra denne problemstillingen ønsker vi i kapittel 2 å belyse teorigrunnet som legger føring for oppgaven videre. Vi setter søkelys på programmering og hvordan programmering kan kobles opp mot algoritmisk tenkning. Videre i teorikapittelet presenterer vi Super:bit, i tillegg til at vi ser på sosiokulturell læringsteori og den didaktiske relasjonsmodellen. I kapittel 3 presenterer vi metodikken for forskingen vi har gjennomført. Det inkluderer forberedelser, gjennomføring og bearbeiding av dataene vi har samlet inn gjennom de kvalitative intervjuene, samt hvilke forskningsetiske vurderinger som er gjort i studien. I kapittel 4 presenterer vi funnene i oppgaven. I diskusjonen i kapittel 5 drøfter vi funnene våre i lys av teorigrunnet vi har presentert i kapittel 2. Oppgaven avsluttes med kapittel 6 der vi konkluderer, trekker fram implikasjoner og tanker om veien videre for forskning på temaet.

2.0 Teorigrunnlag

I dette kapittelet skal vi ta for oss de ulike delene som danner teorigrunnlaget vårt. Vi starter med å presentere hva programmering er, og hvilken plass det har fått i LK20. Videre seg vi på algoritmisk tenkning og koblingen dette har til programmering. Vi skal deretter se på den norske videreføringen og satsingen av Micro:bit, kalt Super:bit. Til slutt tar vi for oss sosiokulturell læringsteori og den didaktiske relasjonsmodellen, som sammen med algoritmisk tenkning danner de tre delene vi baserer teorigrunnlaget på.

2.1 Hva er programmering?

For å kunne forstå hva programmering er, er det viktig å vite hva et program er. Et program er en rekke instruksjoner en datamaskin har fått for å kunne løse en oppgave (Haraldsrud et al., 2020, s. 17). Mange forveksler programmering og koding, men saken er at programmering handler om noe mer enn bare koding (Haraldsrud et al., 2020, s. 189). Programmering er en sammensatt prosess som er bestående av å løse et problem, analysere, kode, feilsøke og samarbeide (Haraldsrud et al., 2020, s. 189). Å programmere handler om å utvikle program gjennom mer enn bare koding (Rossen, 2020; Vihovde, 2020). Ved programmering verifiserer vi et problem, og prøver å komme fram til ulike løsninger. Dette skjer gjennom at vi koder en datamaskin som skal gjøre en oppgave slik at problemet løses. Noen ganger blir ikke dette riktig, og da feilsøker vi for å forbedre kodene og dokumentere løsningene slik at det kan arbeides videre med (Nygård, 2018, s. 7). Koding blir beskrevet som selve prosessen der en skriver instruksjoner om hvordan en kan utføre en oppgave, enten gjennom tekstprogrammering eller blokkprogrammering (Lær Kidsa Koding, 2021, s. 1). Dette er de to typene koding blir kategorisert i. Den ordinære kodingen er tekstkodingen, ofte kjent gjennom program som Python og JavaScript. Det er blokkkodingen som blir mye brukt i skolenene i dag (Angeli et al., 2019, s. 86). Blokkprogrammering er en forenklet versjon der en bruker ferdigprogrammerte ruter eller blokker som en setter sammen, mye likt Lego®, for å slippe en mer utfordrende form for programmering (tekstkoding) (Lindsø, 2020). Programmering kan blant annet foregå i form av koding, bruk av pedagogiske roboter, bruk av regneark eller andre digitale hjelpemidler som skolene har til rådighet. Pedagogiske roboter er vanlige roboter som ved hjelp av forhåndsprogrammerte datamaskiner kan utføre ulike kommandoer som de får beskjed om (Liseter, 2020). Robotere som kan brukes til koding kan være Bit:bot gjennom Micro:bit, LEGO® Mindstorm og SPHERO SPRK+, og dette kan skape en nytteverdi gjennom programmering (Smyrnova-Trybulska et al., 2017, s. 311). Disse pedagogiske robotene er i sammenheng med læring av fag effektive verktøy for å skape interesse, engasjement og motivasjon i læringen av programmering (Grover & Pea, 2013, s. 40). På barneskolen blir disse i stor grad brukt i sammenheng med blokkkodingen (Angeli et al., 2019, s. 86). Denne formen for koding er en effektiv pedagogisk strategi for å lære programmering

gjennom å simplifisere bruken. Dette gjør at en videre slipper store implikasjoner knyttet til bruken av tekstkoding ved lavere skoletrinn (Piedade et al., 2020, s. 3–4). Ifølge Lye og Koh (2014, s. 53) er blokkprogrammering en lettere måte for elever å lære programmering fordi det ligner mer vårt daglige språk. Programmering kan derfor være et viktig verktøy for læring (Angeli et al., 2019, s. 90). I tillegg til at programmering skal være et verktøy for læring er målet i LK20 å utdanne elever med tilstrekkelig digital kunnskap slik at de kan forstå og kunne mene noe om digital teknologi, dens utvikling og rolle i samfunnet. Ifølge Sevik (2016, s. 25) krever det forståelse og kompetanse i programmering for å skape og produsere digitalt.

2.1.1 LK20 og programmering

Programmering har etter innføringen av LK20 blitt implementert i fagene matematikk, naturfag, musikk og kunst og håndverk. Det blir lagt stor vekt på programmering og andre begreper tilknyttet dette feltet i LK20. Dette kommer tydelig fram ved å se på de nye kompetansemålene knyttet til programmering på de ulike trinnene (Kunnskapsdepartementet, 2019, s. 5–14):

- 2.trinn: lage og følge regler og trinnvise instruksjoner i lek og spill.
- 3.trinn: lage og følge regler og trinnvise instruksjoner i lek og spill knyttet til koordinatsystemet.
- 4.trinn: lage algoritmer og uttrykke dem ved bruk av variabler, vilkår og løkker.
- 5.trinn: lage og programmere algoritmer med bruk av variabler, vilkår og løkker.
- 6.trinn: bruke variabler, løkker, vilkår og funksjoner i programmering til å utforske geometriske figurer og mønstre.
- 7.trinn: bruke programmering til å utforske data i tabeller og datasett.
- 8.trinn: utforske hvordan algoritmer kan skapes, testes og forbedres ved hjelp av programmering.
- 9.trinn: simulere utfall i tilfeldige forsøk og beregne sannsynligheten for at noe skal inntreffe, ved å bruke programmering.
- 10.trinn: utforske matematiske egenskaper og sammenhenger ved å bruke programmering.

Kompetansemålene viser at programmering blir eksplisitt nevnt i LK20 fra og med 5.trinn, men kan også finne tilknyttede emner i skolen allerede fra småtrinnet.

2.2 Algoritmisk tenkning

Læreplanen i matematikk består av flere deler. Den ene delen heter etter innføringen av LK20 kjerneelement. Kjerneelementene er med på å beskrive hva som skal være med å gjennomføre hele matematikkundervisningen. Her har algoritmisk tenkning kommet inn som en sentral del. Algoritmisk tenkning blir ofte oversatt fra det engelske ordet «computational thinking». Algoritmisk tenkning har kommet inn i LK20 og blir beskrevet som en problemløsningsmetode gjennom evnen til å kunne bryte ned problemer i mindre og mer overkommelige delproblemer (Utdanningsdirektoratet, 2019a; Kunnskapsdepartementet, 2019, s. 2). Det er en bred faglig enighet om hva algoritmisk tenkning er, men det er ikke enda satt en spesifikk definisjon (Brennan & Resnick, 2012, s. 2). Ifølge Piedade et al. (2020, s. 3) er arbeidet med programmering blant annet tett knyttet til utviklingen av algoritmisk tenkning. Dette gjennom problemløsning, kritisk tenkning, kreativitet, kommunikasjon, innovasjon, samarbeid, og digitale kompetanser (Grover & Pea, 2013, s. 39–40; Piedade et al., 2020, s. 2).

2.2.1 Den algoritmiske tenkeren

Utdanningsdirektoratet har hentet inspirasjon fra Barefoot computing (UK) og videre beskrevet på norsk hva som beskriver den algoritmiske tenkeren. I figuren under er det illustrert hvilke nøkkelbegrep og arbeidsmåter som er forenelig med utviklingen av algoritmisk tenkning (Utdanningsdirektoratet, 2019a).



Figur 1: Den algoritmiske tenkeren [Illustrasjon], av Utdanningsdirektoratet, 27.03.2019, Utdanningsdirektoratet <https://www.udir.no/kvalitet-og-kompetanse/profesjonsfaglig-digital-kompetanse/algoritmisk-tenkning/>

Algoritmisk tenkning blir beskrevet som måten å kunne organisere og analysere informasjon på. Samtidig blir det beskrevet som måten å kunne lage en algoritme, også kalt en framgangsmåte, gjennom å lage abstraksjoner og modeller av den virkelige verden. I denne prosessen arbeider elevene med å fjerne detaljer som er unødvendige og fokusere på de detaljene som er relevante til problemstillingen (Utdanningsdirektoratet, 2019a). Figur 1 beskriver gjennom elleve punkt hvordan en utvikler algoritmisk tenkning. Figuren viser hvilke arbeidsmåter som er forenelige med utviklingen av algoritmisk tenkning. I tillegg viser figuren nøkkelbegrep som er sentrale i løsningen av problem for å utvikle algoritmisk tenkning. Denne figuren til Utdanningsdirektoratet har beskrevet begrep og arbeidsmåter som går igjen i beskrivelsen av algoritmisk tenkning som et bredt tankesett med ferdigheter, strategier og arbeidsmåter for å løse et problem (Cervera et al., 2020, s. 1; Lye & Koh, 2014, s. 52–53; Nouri et al., 2020, s. 1; Wing, 2006, s. 33–34).

2.2.2 Tre dimensjoner av algoritmisk tenkning

Tanken om algoritmisk tenkning er ikke ny. Seymour Papert diskuterte allerede på 1980-tallet prosessuell tenkning (procedural thinking) som en viktig faktor for arbeid med LOGO programmering (Grover & Pea, 2013, s. 38). Vi skal derimot 26 år nærmere vår tid for å komme til en artikkel som endret synet og sparket i gang diskusjonene knyttet til algoritmisk tenkning. I 2006 beskrev Wing (2006, s. 33) algoritmisk tenkning gjennom å vise til hvordan en datamaskin fungerer. Beskrivelsen Utdanningsdirektoratet (2019a) har brukt på algoritmisk tenkning, baserer seg på måten å løse et problem, gjennom å bryte de ned i mindre deler slik at en kan løse de gjennom kjente strategier (Wing, 2006, s. 33–34). Wing (2006, s. 33) beskriver derfor algoritmisk tenkning som en problemløsende tankegang i lys av hvordan en datamaskin fungerer og løser et problem. Det er altså et bredt tankesett som sammenfatter ulike strategier og ferdigheter til å løse et problem (Cervera et al., 2020, s. 2). Denne tilnærmingen kan en sammenligne med å dra på tur. En pakker med seg det man trenger, og det man tror man kan trenge. Det som er problematisk med denne framstillingen er at det kan være vanskelig å knytte opp mot en reel matematikkundervisning og blir noe flytende (Nouri et al., 2020, s. 12–13). Problemet med beskrivelsen til Wing (2006, s. 33) kan ha noe med måten en ser på algoritmisk tenkning. Ifølge en svensk studie utført av Nouri et al. (2020) vil Brennan og Resnick (2012, s. 3) sin beskrivelse av algoritmisk tenkning være lettere i lys av matematikkundervisningen. Studien til Brennan og Resnick (2012) har en bred empirisk dekning og blir mye brukt gjennom beskriving av algoritmisk tenkning (Nouri et al., 2020, s. 1). De mener algoritmisk tenkning er en ferdighet som blir utviklet gjennom arbeidet med programmering. Brennan og Resnick (2012, s. 3) har beskrevet algoritmisk tenkning inn i tre dimensjoner. De kom fram til dette gjennom studien sin der de undersøkte en større gruppe med unge som programmerte i scratch. Disse ble referert til i studien som scratchere. De tre dimensjonene lyder som følger:

algoritmiske begrep, algoritmiske øvelser og algoritmiske perspektiv. Algoritmiske begrep og algoritmiske øvelser er de to dimensjonene som baserer seg på de ulike begrepene og de ulike arbeidsmåtene som en opparbeider seg når en programmerer. En beskrivelse av algoritmisk tenkning ut ifra bare disse to, viste seg å være problematisk da scratchere snakket en del om at de skapte en større forståelse til seg selv sammen med andre, og hvordan forståelsen deres til den teknologiske og digitale verden rundt dem (Brennan & Resnick, 2012, s. 10). Disse tre perspektivene er gode for å kunne måle eller se hvordan elever arbeider/utvikler med algoritmisk tenkning (Nouri et al., 2020, s. 1). Perspektivene blir videre beskrevet av Lye og Koh (2014, s. 53) som essensielle for å kunne beskrive begrepet algoritmisk tenkning, men er ikke nødvendige for å kunne programmere i seg selv. Vi skal nå gå videre inn på hva disse innebærer:

Algoritmiske begrep: De algoritmiske begrepene bygger på begreper som programmerere bruker når de arbeider med programmering (Brennan & Resnick, 2012, s. 3–6). I blokkprogrammeringen sin verden er det ulike begrep som Brennan og Resnick (2012) beskriver:

- Sekvenser: er måten å tenke stegvis eller sette opp stegvise instruksjoner for et program.
- Løkker: er en instruksjon som gjentar seg, enten antall ganger eller til en betingelse er oppfylt.
- Hendelser: kan ligne på betingelser, men handler om når noe skjer, så vil en hendelse starte.
- Parallellisme: er når to sekvenser eller hendelser skjer samtidig.
- Betingelser: er beskrevet nokså lik som hendelser, men handler om at noe om noe skjer så kan det være flere hendelser som kan utfolde seg avhengig av hvilke betingelser som blir oppfylt.
- Operatører: er de regnetekniske funksjonene som beskriver matematiske utregning, logikk og strenguttrykk.
- Data: er måten å lagre, hente og oppdatere verdier som du bruker.

Som en ser gjennom beskrivelsene av de algoritmiske begrepene som Brennan og Resnick beskriver, knyttes disse tett opp til hvordan selve programmeringen tar plass. De legger derimot vekt på at dette også gjelder i situasjoner der en ikke programmerer (Brennan & Resnick, 2012, s. 3). Disse verktøyene er en slags beskrivelse av som kan knyttes opp til Wing (2006) (2006) sitt tankesett om algoritmisk tenkning.

Algoritmiske øvelser: Denne delen av algoritmisk tenkning gjenspeiler de rutinene og øvelsene som en opparbeider seg gjennom å programmere. Det handler om hvordan en beveger seg bort ifra hva man lærer til å tenke hvordan man lærer (Brennan & Resnick, 2012, s. 7). Øvelsene Brennan og Resnick (2012, s. 6–9) beskriver er:

- Å være inkrementell og iterativ: Denne algoritmiske øvelsen beskriver arbeidet som en adaptiv prosess der en kontinuerlig endrer og forbedrer planen ettersom en arbeider med problemet eller oppgaven.
- Testing og feilsøking: Det å kunne gjenkjenne et problem, lage en plan og fullføre den uten hindringer er svært sjeldent mulig, og gjennom arbeidet må en teste og feilsøke. Disse strategiene programmererne opparbeider seg gjennom arbeidet er en sentral faktor i de algoritmiske øvelsene.
- Gjenbruk og remiksing: Det å kunne bygge videre på andre sine prosjekt og henta inspirasjon fra andre sitt arbeid er noe som tett knyttes til de algoritmiske øvelsene. Det å kunne utvide horisonten og potensiale innenfor tidsrammen du selv har er spesielt gunstig når en programmerer for å utvikle større og mer komplekse problem.
- Abstrahere og modulere: Denne øvelsen handler om å sette sammen mindre deler slik at en til slutt for et fullstendig prosjekt.

En ser her på algoritmisk tenkning som en måte å utfolde seg og kan tett knyttes opp mot problemløsning med datamaskiner. Å ha stor takhøyde i arbeidet med dette blir sett på som en essensiell del av didaktikken i matematikkfaget når en programmerer (Grover & Pea, 2013, s. 40). Oppgavene bør derfor legges opp slik at de kan bygges opp gradvis for å differensiere til elevene i klassen slik at de kan jobbe på deres eget nivå (Torkildsen & Stedøy, 2018, s. 21).

Algoritmiske perspektiv: Denne delen handler om de perspektivene programmererne skaper om seg selv og den teknologiske verden rundt dem. Det viser til en selvrefleksjon som handler om hvordan en selv skaper egne ideer og utvikler dem (Brennan & Resnick, 2012, s. 10–11). De deler de algoritmiske perspektivene inn i tre deler. Det handler om å uttrykke seg selv, samspill med andre og være utforskende og spørrende.

- Uttrykke seg selv: Unger er spesielt adaptive i arbeidet med teknologiske og digitale verktøy. Det som derimot skiller det til et perspektiv er hvordan en person videre kan bruke disse verktøyene for å utvikle og skape noe som en kan få uttrykke seg selv med.
- Samspill med andre: Den algoritmiske tenkeren blir beskrevet som en som kan jobbe i samspill med andre. Det å kunne utvikle mer komplekse prosjekter med andre, selv om du ikke kunne lagd det selv, er et viktig perspektiv. Det blir også dratt fram at det å engasjere andre, lære opp andre, lage prosjekter andre kan bygge videre på, er aktiviteter som programmere utvikler gjennom samspill med andre.
- Være utforskende og undrende: En algoritmisk tenker beskrives som en som undrer seg og vil vite mer. Det handler om å ikke ta alt for gitt og prøve å finne ut hvordan ting fungerer.

2.2.3 Oppsummering av algoritmisk tenkning

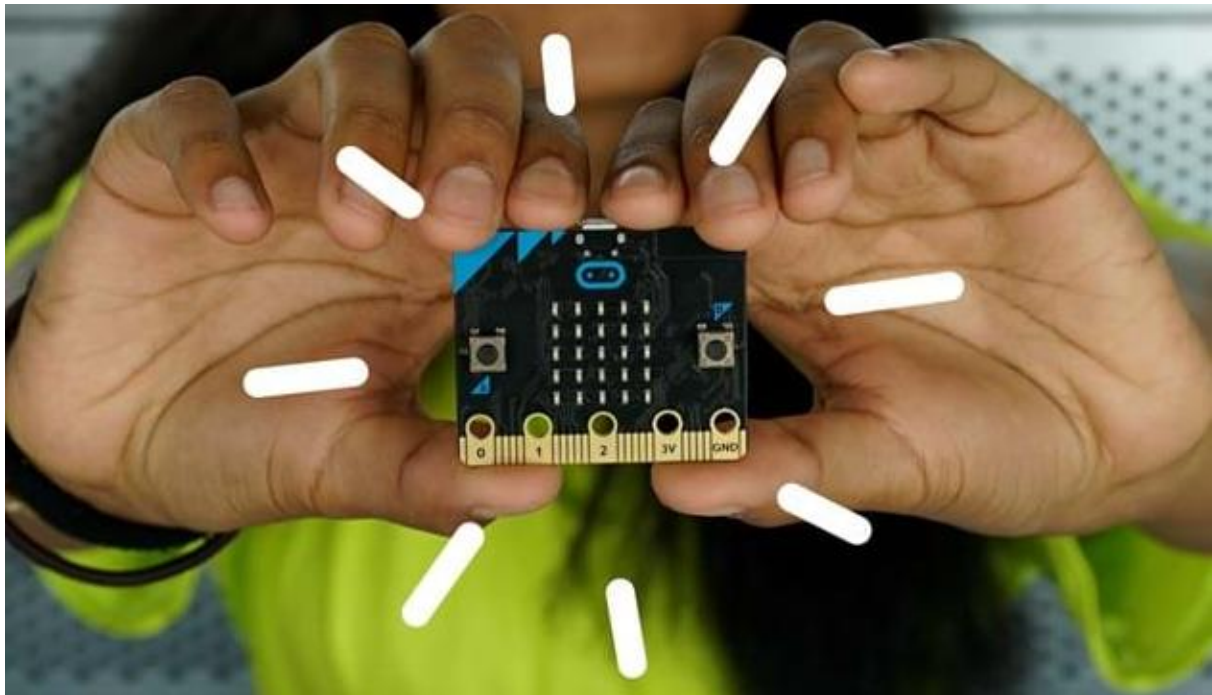
Definisjonen til Brennan og Resnick (2012, s. 3) om algoritmisk tenkning kan være et viktig hjelpemiddel for lærere i matematikkundervisningen (Nouri et al., 2020, s. 1). Likevel er det sentrale begrep og arbeidsmåter som en kan knytte til algoritmisk tenkning gjennom Figur 1. Det er derimot kanskje ikke hele regla da algoritmisk tenkning er vist til å knyttes opp mot: problemløsning, kritisk tenkning, kreativitet, kommunikasjon, innovasjon, samarbeid, og digitale kompetanser (Grover & Pea, 2013, s. 39–40; Piedade et al., 2020, s. 2). Algoritmisk tenkning kan ses i ulike lys gjennom teorien til Brennan & Resnick (2012, s. 3) med en forståelse for hva tanken og beskrivelsen av algoritmisk tenkning som et tankesett bestående av strategier, ferdigheter og arbeidsmåter som en kan opparbeide seg gjennom erfaring (Cervera et al., 2020, s. 1; Lye & Koh, 2014, s. 52–53; Nouri et al., 2020, s. 1; Wing, 2006, s. 33–34).

2.3 Super:bit-prosjektet

2.3.1 Micro:bit

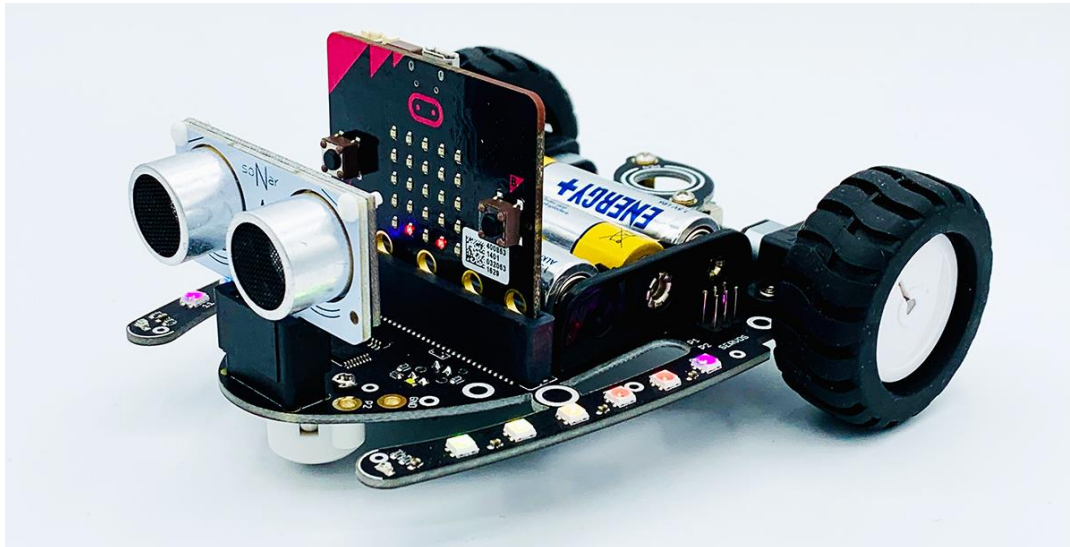
I 2016 ble Micro:bit lansert av Make it Digital, et satsingsprosjekt fra BBC med fokus på å heve programmeringskompetansen hos elever i (Micro:bit Educational Foundation, u.å.b, s. 6, u.å.d).

Micro:bit er en liten datamaskin med et utseende og størrelse til en brikke. Micro:bit består av flere funksjoner.



Figur 2: Micro:bit kodebrikke [Fotografi], av Vitensentrene, u.å., Vitensentrene (<https://www.vitensenter.no/superbit/>)

Figur 2 viser kodebrikken, Micro:bit, med de ulike funksjonene. Micro:bit har LED-lys som kan programmeres, to knapper som kan brukes for å iverksette programmet, utganger for lyd, tilkobling til USB og et strøminntak for batteri. På den nye generasjon av Micro:bit er det touchsensor, mikrofon (lydsensor), en bevegelsessensor, temperaturmåler og en nullstillknapp. Denne kodebrikken kan også kobles til en liten bil kalt Bit:bot. Bit:bot utvider spekteret av aktiviteter med ytterligere sensorer og andre funksjoner (Micro:bit Educational Foundation, u.å.a).



Figur 3: Bit:bot [Fotografi], av n00b.no, u.å., n00b.no (<https://oppgaver.n00b.no/micro-bit/bit-bot>)

På figur 3 kan en se kodebrikken, Micro:bit, som er satt i Bit:bot. Her vil dette gi ytterligere funksjoner som kjøring, lyd, tegning, avstand-, linje- og lyssensor. For å kunne bruke Micro:bit og Bit:bot er det nødvendig å programmere den. I denne oppgaven trekker vi fram pedagogiske roboter. Dette definerer vi med roboter eller maskiner som brukes i undervisning, her inkluderer vi både Micro:bit og Bit:bot.

2.3.2 Satsingen i Norge

Etter Micro:bit ble lansert i Storbritannia har dette spredd seg, og er nå i bruk i over 60 land verden rundt (Micro:bit Educational Foundation, u.å.b, s. 4). I Norge har dette utformet seg i en satsing kalt Super:bit. Satsingen er gitt på oppdrag fra regjeringen gjennom den teknologiske skolesekken (Utdanningsdirektoratet, 2019c). Målet med satsingen her i Norge er å få programmeringsutstyr ut til alle grunnskolene i landet. Slik kan de bruke programmering på en holistisk måte i undervisningen, gjennom å bruke programmering som et verktøy for læring (Micro:bit Educational Foundation, u.å.c). Det skal være med på å inspirere elever og lærere, samt gi kunnskap i den grad det utvikler den algoritmiske tenkningen, og kunnskap knyttet til programmering og teknologi (Super:bit, u.å.b). Prosjektet er styrt av Vitensentrene sammen med NRK og «Lær Kidsa Koding». Disse tre aktørene har forskjellige oppgaver i satsingen, og er følgende:

- Vitensentrene skal arbeide mot og utdanne skoler, samt dele ut utstyr til skolene som deltar på satsingsprosjektet.
- «Lær Kidsa Koding» er en frivillig organisasjon som sikter seg inn mot barn og unge både på skolen gjennom eksempelvis SFO og utenfor skolen som fritidsordning.
- NRK skal nå barn og unge gjennom sine kanaler i NRK Super og NRK Skole.

(Super:bit, u.å.a)

Over en 5-årsperiode (2018-2022) skal alle 6. klasser i Norge ha fått tilbudet om å bli med på dette prosjektet. Skolene får tilbud om å delta på prosjektet gjennom å sende lærere på kurs og delta på opplegget vitensentrene kjører (Super:bit, u.å.c). Når opplegget er ferdig får hver klasse som har hvert med på dette et klassesett som blant annet inkluderer Micro:bit og Bit:bot for å dekke om lag 20-25 elever.

2.3.3 Opplegget fra vitensentrene

Opplegget Vitensentrene kjører med Super:bit er delt i fire deler: forarbeid, Super:bit-oppdraget, etterarbeid og bonusoppdrag. Disse fire delene er en samling av ulike oppgaver som skal gi en progresjon i undervisningen og være en veileder for lærere som skal lære elever programmering.

Forarbeid

Denne delen bygger på tre praktiske oppgaver der det handler om å skape en forståelse for hvordan programmering fungerer gjennom en stegvis tilnærming til en oppgave.

Super:bit-oppdraget

Her starter en først å programmere analogt og går videre over på å programmere enkle oppgaver digitalt. Eksempler på oppgaver lærerne skal gjennomføre er å få Bit:bot til å kjøre en meter, snu og kjøre tilbake.

Etterarbeid

I denne fasen bygger en videre og utvider kunnskapen sin gjennom utforskning og utprøving. Oppgaver som ligger i denne deler er å kjøre Bit:bot gjennom en løype, lage en selvkjørende biler, trafikklys og veibom.

Bonusoppdrag

Den siste delen inneholder mer utfordrende og kreative oppgaver som blant annet kan være å lage roboter som danser eller tegner.

(Super:bit, u.å.b)

Gjennom denne satsingen skal Super:bit være med på å heve kompetansen til lærere og utvikle viljen til å bruke pedagogiske roboter i undervisningen for å skape læring. Elevene skal kunne få økt kunnskapen sin, og sammen skal de skape en nysgjerrighet som i en programmeringssituasjon vil utvikle deres algoritmiske tenkning og programmeringsferdigheter (Vitensentrene, u.å.b).

2.4 Sosiokulturell læringsteori

Vygotsky er en av teoretikerne som står bak den sosiokulturelle læringsteorien, hvor den senere utviklingen av denne teorien i amerikanske og vesteuropeiske land er sterkt inspirert av hans tankegang (Imsen, 2020, s. 196; Skaalvik & Skaalvik, 2021, s. 67). De siste 10-20 årene har interessen for den sosiokulturelle læringsteorien hatt en sterk økning, men den har også tatt ulike retninger. Derfor er det i dag vanskelig å omtale den sosiokulturelle læringsteorien som en enhetlig teori (Skaalvik & Skaalvik, 2021, s. 67). Det vil være mer beskrivende å omtale disse teoriene som et sosiokulturelt perspektiv som kjennetegnes av noen ideer som blant annet; den proksimale utviklingssonen, stillasbygging og artefakter.

Vygotsky hevder at et viktig utgangspunkt i denne læringsteorien er at kulturen som barnet lever i bestemmer hva og hvordan barnet lærer om verden (Skaalvik & Skaalvik, 2021, s. 67). Med andre ord kan dette handle om at mennesker i barnets liv er medaktører som hjelper til å forstå hvordan verden fungerer, og hvordan tolke den. Formidlingen av kunnskapen hvor vi lærer oss å observere, beskrive og handle slik omgivelsene har lært oss, kalles mediering (Säljö, 2020, s. 77). Säljö (2001, s. 13–15) beskriver at barn ikke bare lærer gjennom egen aktivitet og egen utforskning, men at omverdenen blir tolket for oss i samspill med andre i vår kultur. Et sentralt poeng innenfor sosiokulturelt perspektiv er at all intellektuell utvikling og all tenkning har utgangspunkt fra en sosial aktivitet. Ved et sosiokulturelt perspektiv vil barnets individuelle og selvstendige tenkning være et resultat av at barnet gjør ting sammen med andre i et læringsmiljø (Imsen, 2020, s. 196).

2.4.1 Den proksimale utviklingssonen

Vygotsky har en teori om den proksimale utviklingssonen. Denne teorien, sett gjennom et sosiokulturelt perspektiv på læring, kan brukes som et bakteppe for å forstå barns utvikling. Vygotsky skiller mellom to forskjellige mentale utviklingsnivåer hos barnet: barnets nåværende utviklingsnivå og barnets potensielle utviklingsnivå. Barnets nåværende utviklingsnivå viser hva barnet kan utføre på egen hånd uten hjelp og støtte. Vanskegraden på oppgavene som barnet møter på er innenfor barnets mestringssone. Barnets potensielle utviklingssonen er det Vygotsky kaller den proksimale utviklingssonen. Dette er avstanden mellom hva elevene klarer alene og hva de klarer med hjelp fra andre. Det er her det ifølge Vygotsky ligger et læringspotensial (Imsen, 2020, s. 196). Vygotsky peker på at eleven, i samarbeid med voksne og medelever, gjennom samtale og imitasjon, kan få ting til som han/hun ikke kunne gjort på egen hånd (Skaalvik & Skaalvik, 2021, s. 68). Dette er betydningen av å arbeide med den proksimale utviklingssonen, og en populær utforming av dette prinsippet er; hva barnet kan gjøre med assistanse i dag, kan barnet gjøre alene i morgen. Med andre ord handler det om å assistere og gi veiledning til elevene, slik at de kan bli i stand til å løse oppgaver eller gjennomføre aktiviteter på egenhånd. For å kunne arbeide mot den proksimale utviklingssonen i

klasserommet, stilles det store krav til differensiering av undervisningen. På denne måten støtter Vygotskys teori om den proksimale utviklingssone opp under prinsippet om tilpasset opplæring (Imsen, 2014, s. 194).

2.4.2 Stillasbygging

Når innholdet i undervisningen skal knyttes til elevenes nærmeste utviklingssone i et sosiokulturelt perspektiv, trenger elevene veiledning og støtte i læringssituasjonen (Skaalvik & Skaalvik, 2021, s. 70). Begrepet stillasbygging brukes om denne støtten som gis i læringssituasjonen. Dette innebærer at en mer kapabel partner gir støtte til barnet som er avhengig av støtte for å komme seg videre. Denne støtten kan fjernes etterhvert som barnet behersker den aktuelle ferdigheten (Säljö, 2020, s. 79). Med andre ord kan læreren fungere som et stillas for eleven i arbeidet med programmering. For at elevene skal kunne utvikle selvstendighet, og ikke en avhengighet til læreren, bør veiledningen gis på en måte at elevene selv finner løsningen ved at de får tilstrekkelig med hint, forklaringer, korrigeringer og oppmuntringer. I tillegg er eleven ifølge Lyngsnes og Rismark (2007, s. 62) avhengig av noen som kan peke på kritiske faktorer, lage strukturer som hjelper eleven videre i tenkningen, samt noen som minner dem på og hjelper å holde motivasjonen og arbeidsmoralen oppe. For at læreren kan bringe elevene videre i læringsprosessen, må læreren utfordre eleven på oppgaver som er innenfor den nærmeste utviklingssonen. På denne måten kan eleven evne å koble inn sine kognitive funksjoner, ved at svaret ikke blir gitt direkte, men ved at eleven må bygge bro mellom egen kunnskap og kunnskapen til den mer kapable partneren, for å skjønne hvordan eleven skal kunne løse oppgaven (Säljö, 2020, s. 79). Dette kan med andre ord si at etterhvert som eleven mestrer oppgaven på egen hånd, kan en gradvis fjerne stillaset til eleven.

2.4.3 Artefakter

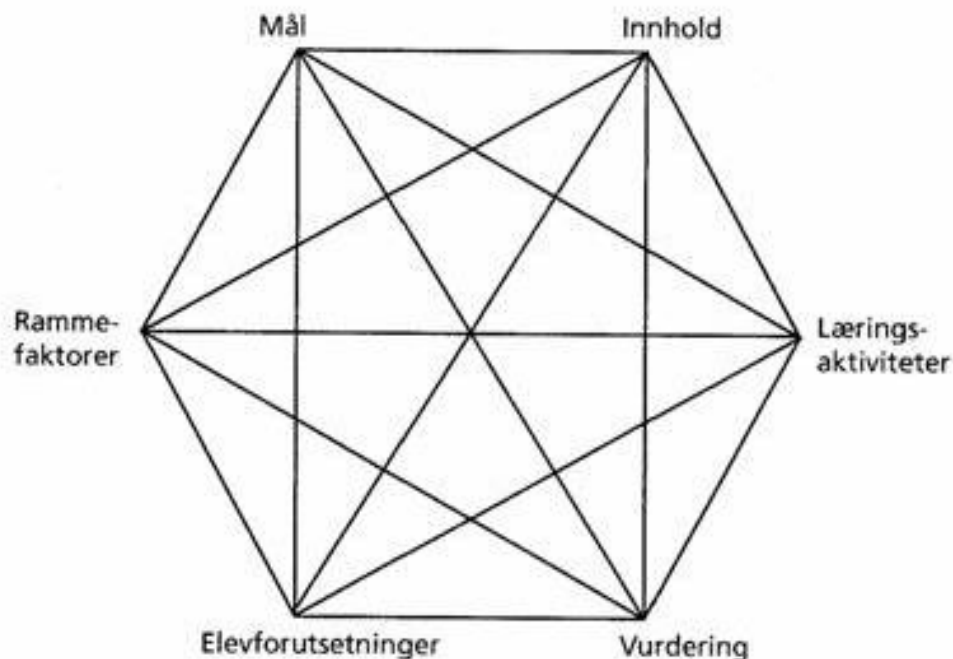
I den sosiokulturelle læringsteorien er utgangspunktet at læring skjer gjennom deltakelse i fellesskap, samt gjennom å bruke medierende artefakter (Säljö, 2020, s. 77). Dette er et sentralt begrep innenfor sosiokulturell læringsteori. Ifølge Säljö (2001, s. 13–15) er det ulike former for fysiske og psykiske artefakter som medierer virkeligheten for mennesker i konkrete sosiale kontekster. Med begrepet mediering menes det hvordan vi mennesker fortolker og formidler verden rundt oss gjennom bruk av artefakter. Artefakter er menneskeskapte redskaper som er skapt for å gi svar på eller løse et menneskelig behov. Medierende artefakter kan både være psykiske redskaper som språk, men også fysiske redskaper som datamaskiner eller annen teknologi som ulike programmeringsverktøy (Säljö, 2020, s. 77). Det at mennesker lærer å ta i bruk og utnytte artefakter kan resultere i utvikling og læring (Säljö, 2001, s. 13–15). Vygotsky hevdet at språket blir et artefakt for læring. Dialogen blir derfor trukket fram som viktig. Gjennom dialog kan elevene komme fram til forståelser og få ideer som ingen av dem ville ha fått om de arbeidet hver for seg. Dialogen har flere

bidragsområder, blant annet å gi forståelse, utvikle begreper, oppklare misforståelser, se sammenhenger og oppdage løsninger (Skaalvik & Skaalvik, 2021, s. 72).

I programmeringssammenheng på skolen kan vi se på interaksjoner, språk og utstyr som sentrale artefakter. Dette er redskaper som blir tatt i bruk i læringsprosessen. Programmeringsverktøyet fungerer som en lærende partner for elevene hvor elevene er i interaksjon med verktøyet. Når elevene arbeider med programmering vil de kontinuerlig få en respons fra programmeringsverktøyet om og hvordan problemet er løst, eller om det finnes feil i programmet som må løses. Videre kan lærere og elever være artefakter innenfor programmering siden elevene får tilbakemelding på om de har gjort oppgaven riktig eller om det er noe som må endres.

2.5 Den didaktiske relasjonsmodellen

Læreplanen skal være et dokument som gjennomsyrrer undervisningen til lærere. Det er altså noe en er lovpålagt å følge, likevel er læreplanen nokså fri. Det ligger vår skoletradisjon til grunne at lærere skal kunne ta egne didaktiske valg til sin undervisning (Bjørndal & Lieberg, 1978, s. 130). Disse didaktiske valgene kan for eksempel være pensum, læringsmål eller læringsmidler som brukes i undervisningen. Som vi presenterte tidligere i oppgaven, er samfunnet i rask endring (NOU 2014: 7, s. 113). Teknologien åpner nye dører og det krever at lærere må utvikle seg i takt med samfunnet for ikke å skape konsekvenser og henge igjen i tiden (Bjørndal & Lieberg, 1978, s. 131; Gundem, 1998, s. 47). For at lærerne skal få tid til å utvikle sin kompetanse er det viktig at de har metoder som er med på å effektivisere planleggingen av undervisning (Bjørndal & Lieberg, 1978, s. 131). Bjørndal og Lieberg (1978, s. 135) lagde derfor en modell som har blitt utviklet til å inneholde seks deler, kalt den didaktiske relasjonsmodellen. Den didaktiske relasjonsmodellen er en modell slått sammen av de elementene de mente var viktig for planleggingsfasen og gjennom erfaringer til den enkelte lærer (Bjørndal & Lieberg, 1978, s. 135). Modellen de lagde er både for å bli brukt i form av planlegging, gjennomføring og analyse av undervisning (Vestøl, 2008, s. 4). Modellen er vist under som figur 4.



Figur 4: Den didaktiske relasjonsmodellen [Illustrasjon], av Siri Randen, 2008, *Rundsanger og kanon* (<http://roundabout-rundsangerogkanon.blogspot.com/2008/11/tennplugger-til-tankevirksomhet.html>)

Undervisningssituasjonen er et dynamisk samarbeid og vekselvirkningen med spenninger mellom. Disse forskjellige didaktiske delene ved undervisningen er illustrert med linjer mellom ordene i illustrasjonen av modellen (Bjørndal & Lieberg, 1978, s. 22; Lyngsnes & Rismark, 2014, s. 81). Modellen tar for seg ulike sider ved undervisningen som elevforutsetninger, rammefaktorer, mål, innhold, læringsaktiviteter og vurdering. Modellen er beskrevet som at den ikke er komplett, men skal ta høyde for at den får fram de viktigste faktorene i undervisningen (Bjørndal & Lieberg, 1978, s. 136). Vi skal nå gå videre inn på de ulike delene ved den didaktiske relasjonsmodellen for å videre få en forståelse for hvordan en kan bruke dette til både planlegging, se på undervisningen og erfaringene lærere har til sin egen undervisningspraksis.

2.5.1 Elevforutsetninger

Elever kommer med ulike forutsetninger gjennom erfaringer, interesser, evner, kunnskap eller ulik kulturell bakgrunn (Lyngsnes & Rismark, 2014, s. 87). Elevenes forutsetninger er sammen med den teknologiske utviklingen i stor grad med å endre selve undervisningen i skolen (Säljö, 2001, s. 13). Det er derfor spesielt viktig at læreren legger opp og planlegger undervisningen etter hvilke læreforutsetninger eleven har og setter realistiske mål (Lyngsnes & Rismark, 2014, s. 87–88). Eleven kan da få benytte sitt kunnskapsnivå, og i samhandling med andre nå den proksimale utviklingssonen (Skaalvik & Skaalvik, 2021, s. 68). Pólya (1985, s. 5) beskriver at læreren må ha en forståelse for erfaringene og kunnskapene til eleven for å kunne være en effektiv veileder i en problemløsingssituasjon. Det må altså være et mål å bruke interessene og forkunnskaper til elevene, slik at differensieringen er nyttig og meningsfullt for hver elev (Polya & Conway, 1985, s. 20). Elevene må da møtes der de er slik at de kan føle seg tilpass, noe som er essensielt for at de skal utfolde seg i undervisningen (Lyngsnes & Rismark, 2014, s. 89). Oppgavene bør legges opp slik at de kan bygges opp gradvis for å differensiere til elevene i klassen slik at de kan jobbe på deres nivå (Torkildsen og Stedøy, 2018, s. 21). Å se på elever på denne måten som utforskende og spørrende vesen er tett knyttet opp til det algoritmiske perspektiv beskrevet av Brennan og Resnick (2012, s. 10–11).

2.5.2 Rammefaktorer

Rammefaktor er de faktorene som begrenser undervisningspraksisen til læreren (Lyngsnes & Rismark, 2014, s. 89). Vi kan dele dette opp i flere deler. En av de er rammene som myndighetene gir oss i form av opplæringsloven og læreplanen. I tillegg hvilke lærebøker, digitale læremidler og annet utstyr som læreren og skolen har tilgang på er sentrale faktorer som er med på å prege valgene som læreren må ta i planlegging og gjennomføring av undervisning (Lyngsnes & Rismark, 2014, s. 90). Det er ekstremt mange faktorer som kan være med å begrense en undervisning. Blant dem kan skolens tilgang på læreverk, midler til innkjøp av pedagogiske roboter, eller mangel på datamaskiner slik at

en må planlegge godt i deling mellom klasser. Hattie (2009, s. 119–120) mener det viktigste rammefaktorer er læreren selv gjennom kompetanse, holdninger, kreativitet og innsatsen en legger i arbeidet en skal bedrive. Det viser seg at lærernes kompetanse og holdninger til digitale verktøy er essensiell i hvor mye de bruker det i undervisningen sin og hvor godt læringsutbyttet er ovenfor elevene (Cuban, 2001, s. 135). Disse erfaringene er sentrale til hvordan læreren vil kunne bruke det den har for å lage et best mulig opplegg som fremmer elevens læring. De siste faktorene som er like viktige er skolen og organiseringen av skolen. De handler om hvordan skolen legger opp til pedagogisk utvikling gjennom kurs og kompetanseheving. Videre er relasjon og dialog mellom lærere og ledelsen viktig og hvilke forventninger de har til hverandre (Lyngsnes & Rismark, 2014, s. 90).

2.5.3 Mål

Mål eller læringsmål vil være det som skal oppnås etter endt time eller periode. Det er særs viktig at læringsutbyttet til eleven står i sentrum (Lyngsnes & Rismark, 2014, s. 92). Dette skal beskrive det elevene sitter igjen med. Når en lærer utarbeider mål for timer eller perioder, er læreplanen en sentral del i dette arbeidet. Mer spesifikt er det kompetansemålene som gir retningslinjer på hvilke ferdigheter eller kunnskaper elevene skal sitte igjen med (Lyngsnes & Rismark, 2014, s. 93). Et eksempel på dette i matematikkfaget kan sees ut ifra kompetansemålet som er rettet mot programmering etter endt 6.trinn: *«Mål for opplæringen er at eleven skal kunne bruke variabler, løkker, vilkår og funksjoner i programmering til å utforske geometriske figurer og mønstre»* (Kunnskapsdepartementet, 2019, s. 10). Ut ifra hvordan dette kompetansemålet er formulert vil ikke dette være tilstrekkelig nok som læringsmål i en time da det er for bredt (Lyngsnes & Rismark, 2014, s. 97). Kompetansemålene er lagt opp i den grad de kan differensieres slik at forutsetningene til elevene kan stå i sentrum når det utarbeides læringsmål (Lyngsnes & Rismark, 2014, s. 94). De bør sees på som en stegvis forståelse der en kan reprodusere kunnskap, se sammenhenger og kombinere kunnskap (Lyngsnes & Rismark, 2014, s. 97). Da vil det være lettere for en elev, sammen med veiledning fra lærer, kunne nå sitt potensiale gjennom den proksimale utviklingssonen (Imsen, 2014, s. 194). Læringsmålene en setter i skolen bør en da styre undervisningen etter og legge til rette for at de kan arbeides mot (Bjørndal & Lieberg, 1978, s. 39). Programmering er en del av matematikkfaget der en jobber mer praktisk og utforskende (Eidslott, 2021). I praktisk og utforskende arbeid bør dette være et fokus når en setter mål, det er da veien og ikke sluttresultatet som er målet (Bjørndal & Lieberg, 1978, s. 38; Lyngsnes & Rismark, 2014, s. 94). Det er ikke alle tilfeller en bør legge opp til at en skal nå alle målene, men derimot jobbe mot dem. I en slik utforskende undervisning bør det differensieres til at eleven kan arbeide på nivået en selv er på (Lyngsnes & Rismark, 2014, s. 98).

2.5.4 Innhold

Innholdet i en undervisning blir sett på som den faglige og den allment faglige kunnskapen som blir regnet til å knyttes mot ett fag (Bjørndal & Lieberg, 1978, s. 41), og sammen med mål er innhold en viktig faktor som viser i hvilken retning undervisningen skal ta form (Lyngsnes & Rismark, 2014, s. 98). Innholdet i undervisning skal være i tråd med rammeverket som er satt utarbeidet etter samfunnets behov. Dette reflekteres i samfunnets behov for utviklingen av programmering og algoritmisk tenkning i tråd med den raske teknologiske utviklingen (NOU 2014: 7, s. 67). Da er det viktig å spille på elevenes interesser for å skape en forståelse ovenfor nytten av de de skal lære (Lyngsnes & Rismark, 2014, s. 99). Om en ikke fenger elevene vil interessen deres dale og de vil ikke ha samme engasjement eller drivkraft for å utføre eller bedrive en oppgave (Polya & Conway, 1985, s. 5). Engelsen (2012, s. 131–133) viser til tre faktorer som er viktige å spille på i skolens innhold i undervisning:

Fagets egenart

Fagets egenart skal skape en forståelse for hvorfor en trenger denne kunnskapen og hva en kan bruke den til (Engelsen, 2012, s. 131; Lyngsnes & Rismark, 2014, s. 99).

Samfunnets interesser

Samfunnets interesser kan være det å motvirke negative tendenser i samfunnet som rasisme eller styrke positive tendenser som teknologisk utvikling og medborgerskap (Lyngsnes & Rismark, 2014, s. 100). Det handler i bunn og grunn om en bevisstgjøring av hvilken retning samfunnet går i. Dette kan være å skape, bruke og mestre digitale verktøy for å mestre samfunnet en skal ut i (Engelsen, 2012, s. 132–133; NOU 2014: 7, s. 36; Wing, 2006a, s. 33).

Elevenes behov

Elevenes interesser og forutsetninger er spesielt viktig i valg av innhold (Engelsen, 2012, s. 131). Om en ikke bygger på interessen vil elevene miste motivasjon (Polya & Conway, 1985, s. 5), og vil ikke kunne nå den proksimale utviklingssonen (Skaalvik & Skaalvik, 2021, s. 68). Elever skal mest sannsynlig ikke få samme innhold, da de ulike forutsetningene, behovene og interessene fører til at en trenger å tilpasse innholdet spesifikt (Lyngsnes & Rismark, 2014, s. 100).

Beskrivelsen til Engelsen (2012) viser at det komplekse innholdet i hvordan en bør bygge opp faget. Det er nok derimot en stor sammenheng mellom disse tre faktorene som Engelsen (2012) presenterer i den grad samfunnets interesser påvirker skolens læreplaner som igjen vil være i tråd med det elevene har ønske om å lære, for eksempel knyttet til programmerings plass inn i skolen og matematikkfaget.

2.5.5 Læringsaktiviteter

Denne delen av den didaktisk relasjonsmodellen omhandler aktivitetene som læreren legger opp til i undervisningen sin (Lyngsnes & Rismark, 2014, s. 101). Hvordan og hvilke aktiviteter som blir brukt i undervisningen er opp til læreren og det er lærerens ansvar for å legge opp til en god læringsaktivitet (Hattie, 2009, s. 34). Da er lærerens kompetanse og holdninger helt avgjørende for utbyttet til elevene i undervisningen (Hattie, 2009, s. 34; Lyngsnes & Rismark, 2014, s. 102). Det er flere måter legge til rette for ulike gode læringsaktiviteter i undervisning. Hattie (2009, s. 210) peker på bruken av teknologi og digitale verktøy, samt praktisk arbeid som positive for elevens læring og danning.

Bruke teknologi og digitale verktøy

Den teknologiske utviklingen bidrar til at en kan variere undervisningen i større grad nå i dag (Hattie, 2009, s. 222). Det er derimot viktig at læreren har kunnskap og en holdning til at de vil lære seg ny dette for å sikre en god undervisning gjennom læringsaktiviteter med digitale verktøy (Hattie, 2009, s. 34). Denne digitale kompetansen er det viktig at læreren benytter, da elevene i stor grad har en høy digital kompetanse ettersom at de har vokst opp med disse verktøyene i større grad enn læreren (Lyngsnes & Rismark, 2014, s. 105). Dette viser igjen i Monitor 2019 der det blir vist til at lærernes egen kompetanse er en avgjørende faktor for at lærerne skal ta i bruk digitale hjelpemidler (Fjørtoft et al., 2019, s. 60).

Praktisk arbeid

Praktisk arbeid blir sett på som en viktig måte å legge opp læringsaktiviteter gjennom fokuset på læring av ferdigheter (Lyngsnes & Rismark, 2014, s. 109). Elever kan observere for å se hvordan ting blir gjort, men en må i stor grad bruke verktøyene selv for å kunne utvikle en erfaring og forstå bruken av det (Lyngsnes & Rismark, 2014, s. 109). En kan sammenligne dette med å lære seg programmering. Da kan en med fordel observere og utvikle en kunnskap om bruken av programmet, men det kreves en øvelse for å utvikle ferdigheten. Elevene kan her jobbe med å observere, øve seg, få tilbakemelding og veiledning i videre arbeid med ferdigheten (Lyngsnes & Rismark, 2014, s. 110).

Å bruke pedagogiske roboter som et digitalt verktøy sammen med praktisk arbeid viser seg gjennom studiet til Hattie (2009, s. 210) å være positivt for elever. Videre skal vi se på måter elever kan jobbe i samspill med andre for å få benytte potensialet sitt.

Arbeid individuelt i et sosialt fellesskap

Det er flere faktorer som kan være avgjørende for en undervisnings der en legger opp til individuelt arbeid i et sosialt fellesskap. Når elevene har muligheten til å diskutere og finne inspirasjon fra andre er dette positivt for læringsutbytte (Lyngsnes & Rismark, 2014, s. 111). En form for organisering er at elevene kan hjelpe andre når de er ferdig med sine oppgaver (Lyngsnes & Rismark, 2014, s. 111). Da

kan eleven få trening i både å forklare hva den sliter med, og den hjelpende eleven kan vise til kunnskap som den må omformulere til forståelig kunnskap. De vil være stillas i hverandre lærings situasjon (Lyngsnes & Rismark, 2014, s. 111). I tillegg er det lett å differensiere da elevene jobber individuelt (Imsen, 2020, s. 194).

Gruppearbeid

En annen måte å legge til rette for praktisk arbeid eller bruk av digitale verktøy i programmeringsundervisning er gruppearbeid. Alt fra å jobbe i mindre grupper til større grupper i hele eller deler av timen. Her må det være et samarbeid mellom elevene og de må kunne skape eller løse ett felles produkt for at det skal kategoriseres som gruppearbeid (Lyngsnes & Rismark, 2014, s. 113). I et sosiokulturelt perspektiv bør ikke elevene ligge på samme nivå i gruppearbeid. Slik kan en benytte styrker og svakheter slik at elevene kan utvikle seg og nå potensialet deres i den proksimale utviklings sone (Lyngsnes & Rismark, 2014, s. 114). En slik gruppe vil bli kalt heterogen ,og en gruppe på likt nivå vil bli kalt homogen. Det er vanskelig å si om den ene er bedre enn den andre, men oppgaven må være meningsfylt og legge opp til samarbeid mellom elevene (Lyngsnes & Rismark, 2014, s. 114).

2.5.6 Vurdering

I skolen blir vurdering ofte brukt som en kunnskapsmåling som en summativ vurdering (vurdering av læring) (Lyngsnes & Rismark, 2014, s. 119). Dette kan bli gjort gjennom prøver eller innlevering der en vil vise hva eleven mestrer. Formativ vurdering (vurdering for læring) legger vekt på læringen i seg selv og veien videre (Wølner, 2013, s. 47). Vurdering for læring bør skje gjennom en faglig progresjon, arbeidet og innsatsen eleven legger i faget (Lyngsnes & Rismark, 2014, s. 119). En gruppe kan få en felles vurdering eller gi individuell vurdering (Lyngsnes & Rismark, 2014, s. 115), men kan være vanskelig å vurdere i forhold til fordeling av arbeid og innsats. Når en bruker summativ vurdering, er det flere måter å vurdere arbeid på. En kan bruke mapper, som Brennan og Resnick (2012) brukte i studiet sitt til hvordan programmerere brukte scratch. Der så de på hvilke blokker i programmering de har brukt (Brennan & Resnick, 2012, s. 12). En kan også bruke logg som en formativ vurdering der eleven skriver selv til læreren om arbeidet de har gjort. Loggen kan bestå av konkrete spørsmål eller erfaringer som eleven har gjort seg gjennom et arbeid (Lyngsnes & Rismark, 2014, s. 125). Vurdering kan gi et inntrykk av verdien til det som har blitt gjort og vurdere læringsprosessen til eleven. Den skal skape et samspill og forståelse mellom lærer og elev slik at vurderingen blir en læringsprosess gjennom periodene med arbeidet (Lyngsnes & Rismark, 2014, s. 127).

2.5.7 Oppsummering av den didaktisk relasjonsmodell

Den didaktiske relasjonsmodellen må brukes i en helhet. Den er ikke laget for å planlegge hver enkelt del av undervisningen. Alle delene som inngår i modellen står i relasjon og koblingen mellom delene er kompleks (Bjørndal & Lieberg, 1978, s. 136–137). De handler om å bruke de erfaringene en som lærer har opparbeidet seg gjennom undervisningssituasjonene. Ut ifra disse kan en planlegge egen undervisning da en kjenner disse erfaringene best selv (Gundem, 1998, s. 47). I

programmeringsundervisning er det viktig at en lærer bruker midlene og rammene den har, slik at læreren selv ikke blir begrensingen i undervisningen (Hattie, 2009, s. 119–120). Videre er det et sentralt punkt å se på læringsmålene for timen. Læringsmålene bør spille på ferdighetene og utviklingene av disse som læringen i seg selv (Bjørndal & Lieberg, 1978, s. 38; Lyngsnes & Rismark, 2014, s. 94). Det er altså prosessen som er målet og ikke selve målet i seg selv. Da dette kan være vanskelig å vurdere trekkes logg fram til å være en god måte å bruke formativ vurdering på som skaper læring hos elevene (Lyngsnes & Rismark, 2014, s. 125). Undervisningen må spille på interessene til elevene slik at de kan føle en nytteverdi mot ferdighetene de lærer (Polya & Conway, 1985, s. 20). Dette kan lærerne oppnå gjennom en differensiert undervisning der elevene får jobbe praktisk, enten individuelt i et fellesskap eller i grupper (Hattie, 2009, s. 34). Ved å spille på disse ulike faktorene i matematikkundervisningen vil lærere i større grad kunne sikte seg inn mot kompetansemålene og algoritmisk tenkning i programmeringsundervisning (Engelsen, 2012, s. 131; NOU 2014: 7, s. 36; Wing, 2006, s. 33).

2.6 Oppsummering av teorigrunnlag

Vi ser det nå som hensiktsmessig å gi en kort oppsummering av teorigrunnlaget presentert i kapittel 2. Definisjoner, momenter og teoretisk grunnlag reist i denne delen av oppgaven er alle relevante for programmering i matematikkundervisningen. Vi har foretatt en nøye gjennomgang av hva programmering er og dens plass i LK20. Vi har presentert teori og forskning om algoritmisk tekning, Super:bit-prosjektet, sosiokulturell læringsteori og den didaktiske relasjonsmodellen.

Blokkprogrammering kan være en effektiv måte å tilegne seg kunnskap og ferdigheter om programmering og algoritmisk tekning. Bakgrunnen for dette munner i at blokkprogrammering har et språk som ligner på vår daglige tale (Lye & Koh, 2014, s. 53). Blokkprogrammering viser seg å være spesielt gunstig for elever med lav måloppnåelse. Der kan utviklingen av ferdigheter og strategier gjennom arbeid med algoritmisk tenkning ha en positiv innvirkning på måloppnåelsen generelt i faget (Lye & Koh, 2014, s. 56). En algoritmisk tenker jobber, i en problemløsnings situasjon, aktivt for samarbeid og deling av innhold for at en skal kunne løse et problem en ikke hadde klart alene (Brennan & Resnick, 2012, s. 10–11; Utdanningsdirektoratet, 2019a). Dette kan foregå gjennom en praktisk tilnærming der en jobber individuelt i fellesskap eller i grupper (Brennan & Resnick, 2012, s. 10–11; Hattie, 2009, s. 210; Lyngsnes & Rismark, 2014, s. 109, 111, 113). Gjennom de sosiale interaksjonene i programmering, som Brennan og Resnick (2012, s. 10–11) beskriver, kan dette knyttes opp mot et sosiokulturelt læringssyn. Et sosiokulturelt læringssyn legger blant annet vekt på at stillasbygging er en fin måte å lære på (Lye & Koh, 2014, s. 59; Skaalvik & Skaalvik, 2021, s. 70). Pólya (1985, s. 5) mener at læreren må ha en forståelse for erfaringene og kunnskapene til eleven for å kunne være en effektiv veileder i en problemløsnings situasjon. Lærere bør i arbeidet med programmering i matematikkundervisningen ha som mål å bruke interessene og forkunnskapene til elevene. På denne måten kan de enklere arbeide for å differensiere i riktig retning for hver enkelt elev. Da blir det også enklere å utnytte elevenes potensiale maksimalt. Teorigrunnlaget trekker også fram at arbeid i par eller mindre grupper er spesielt gunstig for læring av problemløsning (Sevik, 2016, s. 15). Dette står også i sterk tilknytning til læring i sosiale fellesskap i den sosiokulturelle læringsteorien (Imsen, 2020, s. 191). Gjennom systematisk og analytisk arbeid kan løsninger generaliseres slik at de kan kombineres for å løse større problem (Utdanningsdirektoratet, 2019a).

3.0 Metode

I dette kapittelet skal vi gjøre rede for studiens forskningsdesign, utvalg av lærere og hvordan vi har samlet inn studiens datamateriale for å besvare vår problemstilling. Vi ser på grunnlaget for valg av spørsmål i intervjuguiden. Videre presenterer vi en nøye beskrivelse av studiens analyseprosess. Avslutningsvis blir det lagt fram våre refleksjoner knyttet til studiens forskningskvalitet, og pekt på etiske hensyn tatt i betraktning i denne studien. Gjennom hele forskningsprosessen har vi hatt som mål at studien skal være nøytral ut ifra våre egne synspunkter og forkunnskaper. Dette har vi også tatt i betraktning når vi samlet informasjon fra lærerinformantene i denne studien.

3.1 Forskningsdesign og metode

Forskningsdesignet i vårt masterprosjekt beskriver hvordan vi har utført forskningen vår og hvilke metoder som vi har valgt for datainnsamling og analysen. I vårt masterprosjekt har vi samlet inn datamaterialet ved bruk av eksplorerende forskning ettersom at vi er avhengig av å samle inn rik og detaljert informasjon fra lærere om programmering. På denne måten kan vi opparbeide oss klare rammer for å forstå feltet vi undersøker. Eksplorerende forskning er ifølge Høgheim (2020, s. 130) å samle inn rike og detaljerte tekstdata, og har som mål å skape kunnskap om et felt, mennesker uten klare antakelser om hva man kommer til å observere. Som forskere har vi ikke klare rammer for å forstå dette før vi undersøker det (Befring, 2020, s. 48–49). Metoden er fleksibel, og innsamlingsmetoden har tatt sikte på å gå dypt og detaljert (Høgheim, 2020, s. 129).

3.1.1 Kvalitativ tilnærming

Temaet og innfallsvinkelen til forskningen har hatt betydning for valg av metode. Som problemstillingen vår poengterer, er vår interesse å finne ut hvordan matematikklærere legger til rette for læring av matematikk på 6. trinn gjennom bruk av pedagogiske roboter fra Super:bit-satsingen. I tillegg vil fokuset i vårt prosjekt være lærerens erfaringer og meninger knyttet til dette. Som utgangspunkt for vårt masterprosjekt har vi derfor valgt en kvalitativ tilnærming for å svare på problemstillingen vår. Dette er en tilnærming som ofte blir valgt når man ønsker å forstå og beskrive menneskelig erfaring og opplevelser. Når vi i vårt forskningsprosjekt søker innsikt i matematikklæreres indre liv, med tanker, erfaringer og holdninger knyttet til arbeidet med Super:bit i klasserommet (Befring, 2020, s. 92), er det naturlig at metodevalget faller på en kvalitativ tilnærming. Med andre ord kan den kvalitative forskningsmetoden framheve innsikt, og søke etter en dypere forståelse for temaene som blir tatt opp under datainnsamlingen. På denne måten går vi inn i forskningen med en genuin nysgjerrighet på hvordan læreres liv formes, hvordan sosial orden oppstår, framfor å anta at dette kan forklares ut ifra noen spesifikke faktorer (Tjora, 2021, s. 36). Vi tenker at på denne måten har vi best mulighet for å få svar på problemstillingen vår. Ved kvalitativ

datainnsamling blir dataen blant annet hentet ifra intervjuer, observasjoner og tekster (Høgheim, 2020, s. 29), hvor vår datainnsamling bygger på kvalitative intervjuer. Dataen som blir samlet inn ved en kvalitativ metode er ofte i form av skriftlig materiale. Dette betegnes av noen som «myke data» fordi man får et rikt datamaterialet som ofte gir dypgående informasjon om det man forsker på, som må tolkes og analyseres av forskeren (Høgheim, 2020, s. 29).

3.2 Datainnsamling og verktøy

Datainnsamlingen ble utført ved hjelp av intervju, som er den mest utbredte metoden å generere data på innenfor kvalitativ forskning (Tjora, 2021, s. 127). Hensikten med intervjuene var å produsere data om hvordan lærerne la til rette for læring av matematikk når elevene arbeidet med Super:bit. Samtidig var hensikten å produsere data som viste hvilke erfaringer lærerne hadde knyttet til dette. Kvalitative dybdeintervju har ulike strukturer, hvor vi har valgt å forholde oss til et semistrukturert intervju. Dette intervjuet kjennetegnes med at lærerne skal kunne svare åpent og fritt (Befring, 2020, s. 75). På denne måten får lærerne våre mulighet å snakke rundt deres erfaringer og opplevelser knyttet til programmering i skolen. Dette kan være til fordel ettersom at man i en intervjusituasjon tillater digresjoner, og dermed kan komme inn på viktige og relevante temaer, og forhold som vi som forskere nødvendigvis ikke hadde tenkt ut på forhånd. På denne måten kan vi, som forskere, stille oppfølgingsspørsmål for å prøve og få en forståelse for lærernes tanker, erfaringer og holdninger gjennom intervjuet (Tjora, 2021, s. 37–38).

3.2.1 Kvalitativt forskningsintervju

For å lede intervjuet utformet vi en gjennomtenkt intervjuguider, der spørsmålene og rekkefølgen er valgt med tanke på å skaffe oss forståelse for lærerens praksis og erfaringer til bruken av programmering i undervisning (Vedlegg 3). I lys av relevant teori, og på grunnlag av tidligere forskning har vi tenkt gjennom ulike temaer som kan være med å belyse vår problemstilling. Det at vi tenker over hvilke spørsmål vi stiller, og hvorfor, kan være en fordel i analysen. Som Kvale og Brinkmann (2015, s. 141) peker på kan dette gi oss, som forskere, anledning til å avklare lærernes erfaringer, tanker og holdninger.

Intervjuguiden har en tredelt struktur som Tjora (2021, s. 159) presenterer med introduksjon, refleksjon og avrundning. Før intervjuet starter får lærerne praktisk informasjon om blant annet lydopptak og anonymisering av personopplysninger. De blir også minnet om taushetsplikten de har som lærere, slik at de ikke bryter den i intervjusituasjonen ovenfor oss som ekstern intervjuer. I introduksjonsdelen har vi utformet spørsmål om utdanning, ulike begrepsavklaringer og hvor mye læreren har undervist i programmering. Dette er spørsmål som krever lav grad av refleksjon, og er med på å skape en trygghet hos læreren i starten av intervjusituasjonen (Tjora, 2021, s. 160).

Refleksjonsdelen var todelt, en del om lærerens erfaringer knyttet til programmering med Super:bit i undervisning, og en del om hvordan læreren ser på matematikken når elevene arbeider med Super:bit. I refleksjonsdelen har vi skrevet en del hovedspørsmål, med tilhørende stikkord og småsetninger på flere av spørsmålene. En del av spørsmålene til de ulike temaene er introduksjonsspørsmål. Disse spørsmålene har til hensikt å få fram det lærerne selv finner som det

mest sentrale ved det aktuelle temaet (Kvale & Brinkmann, 2015, s. 166). De tilhørende stikkordene og småsetningene er til hjelp for å utforme aktuelle og relevante oppfølgingsspørsmål knyttet til det læreren har pratet om. Dersom læreren kommer innom tema som er utenfor intervjuguiden, eller ikke svarer utfyllende nok på spørsmålene våre, ønsker vi å be læreren om å utdype seg mer eller komme med flere eksempler. Dette kalles i Kvale og Brinkmann (2015, s. 166) for oppfølgingsspørsmål. For å få en rød tråd gjennom intervjuene og for å få tydelige overganger mellom de ulike temaene, har vi lagt opp til å bruke det Kvale og Brinkmann (2015, s. 166) kaller for strukturerende spørsmål.

I den siste delen, avrundingen, legger vi i intervjuguiden opp til at læreren skal få komme med ytterligere refleksjoner eller momenter som læreren selv synes er viktige. Avslutningen har som mål å normalisere situasjonen mellom oss som intervjuer og lærerne etter intervjuet, og ta ned refleksjonsnivået (Tjora, 2021, s. 160). I intervjuets slutfase benytter vi til å takke for lærernes deltakelse og informere om kontaktinformasjon ved eventuelle spørsmål. Samtidig vil vi repetere hvordan vi håndterer lydopptak og personopplysninger og at læreren har rett til å trekke seg til enhver tid.

3.2.2 Utvalg av lærere

For å kunne utforske og besvare problemstillingen vår valgte vi å invitere deltakere til intervju ut ifra spesielle kriterier. Vi valgte å bruke en strategisk utvelgelse av deltakere. Her vurderte vi hvilken målgruppe som kunne være med å delta, ut i ifra hvorvidt vi mente de kunne svare på nødvendige spørsmål knyttet til studien vår (Christoffersen & Johannessen, 2012). Disse kriteriene kalles kriterieutvalg (Tjora, 2021, s. 48). Vi satt to kriterier som skulle være med på å finne lærere som kunne skaffe oss empiri knyttet til problemstillingen vår:

(1) Lærerne skulle være matematikklærere på mellomtrinnet.

(2) Lærerne skulle ha undervisningserfaring med Super:bit.

Viktigheten av at lærerne våre skulle være fra mellomtrinnet, var det flere grunner til. Utdanningene til undertegnede overlapper hverandre på 5.-7. trinn, og vi ser på det som hensiktsmessig å ha ulike erfaringer som kan knyttes til både småskolen og ungdomstrinnet i arbeidet med mellomtrinnet. Med innføringen av LK20 har programmering fått en sentral plass fra og med 5. trinn (Kunnskapsdepartementet, 2019). Videre er Super:bit-prosjektet noe alle 6. klasser i Norge har fått tilbudet om (Super:bit, u.å.a). Det er hovedsakelig utviklet for arbeid på mellomtrinnet og skal være en enorm ressurs som skal være nyttig for matematikkundervisningen på skolene (Vitensentrene, u.å.b). Når vi videre ser på hvordan LK20 er bygd opp, faller programmeringen inn under flere kompetansemål. Noen av kompetansemålene er eksplisitt knyttet opp mot programmering som

dette kompetansemålet fra 6. trinn i matematikk «bruke variabler, løkker, vilkår og funksjoner i programmering til å utforske geometriske figurer og mønstre» (Kunnskapsdepartementet, 2019, s. 10).

I tillegg skulle matematikklærerne ha erfaring med programmering knyttet til den nasjonale Super:bit-satsingen. Med erfaring mener vi lærere som hatt programmeringsundervisning med Super:bit minst en gang. Om lærerne har prøvd å programmere og/eller kodet Super:bit kun en undervisningsøkt, eller om de har prøvd å programmere og/eller kode Super:bit over en lengre periode, er av interesse. Dette fordi vi har et ønske om å nå typiske-tilfeller-sampling, som handler om at en har et utvalg som skal representere en ordinær matematikkundervisning på mellomtrinnet med programmering (Høgheim, 2020, s. 154–155). Vi ser på det som nyttig at vi får et bredt spekter av erfaringer fra matematikklærerne for å danne oss et bilde av hvordan det er å ha en slik form for undervisning knyttet til dagens skole og slik læreplanen er bygd opp nå. Det er heller ikke et mål at vi skal intervju lærerne i størst grad om hvilke tanker de har om programmeringen, men derimot erfaringene de har gjort seg gjennom egen undervisning. Målet med kriterieutvalgene er å optimalisere deltakernes bidrag til å kunne svare på problemstillingen (Tjora, 2021, s. 48).

	Undervisningserfaring i matematikk	Undervisningserfaring i programmering
Lærer 1	Har jobbet i skolen med matematikk i fagkretsen de siste 8 årene.	Har jobbet med programmering i bolker de siste skoleårene (2 år), men beskrives som forholdsvis nybegynner knyttet til programmeringen i seg selv. Har vært på Super:bit-kurs og praktisert dette i programmeringsundervisningen læreren har hatt på mellomtrinnet.
Lærer 2	Har jobbet i skolen med matematikk i fagkretsen de siste 15 årene.	Har brukt programmering i undervisningen sin hele perioden sin som lærer og vi tolker læreren som godt erfaren med bruk av programmering i undervisning. Har ikke vært på Super:bit-kurs, men har mye erfaring med bruken fra undervisning.
Lærer 3	Har jobbet i skolen med matematikk i fagkretsen de siste 4 årene.	Har undervist i programmering i matematikkfaget det siste året med bolker på noen uker, der det er et fokus på programmering i undervisningen. Læreren har ikke vært på Super:bit-kurs, men har praktisert bruken i undervisningen sin.

Lærer 4	Har jobbet i skolen med matematikk i fagkretsen de siste 5 årene.	Programmering har blitt brukt jevnt gjennom året og har undervist i disse de siste årene i skolen (knappe 5 år). Har vært på Super:bit-kurs og praktisert dette i undervisningen.
---------	---	---

Tabell 1: Utvalget av lærerne i intervjuet, her beskrevet med lærernes erfaring og kunnskap.

I tabellen over har vi lagt inn de fire lærerne som har deltatt på prosjektet vårt. Vi har illustrert de ulike lærerne med undervisningserfaring i matematikk og programmering med Super:bit.

3.2.3 Pilot

Vi har i dette prosjektet valgt å gjennomføre et pilotintervju. Målet med dette var å teste ut spørsmålene våre i intervjuguiden, samt bli kjent med hvordan en intervjusituasjon fungerer. For å få et så nært resultat som mulig knyttet opp til vårt prosjekt, oppfylte læreren fra pilotundersøkelsen kriteriene for utvalget i prosjektet. Læreren til pilotintervjuet var derfor matematikklærer på mellomtrinnet som hadde erfaring med bruk av Super:bit i programmeringsundervisningen. På denne måten var pilotintervjuet et viktig hjelpemiddel i utformingen av den endelige intervjuguiden vår (Høgheim, 2020, s. 164). Pilotintervjuet ble gjennomført digitalt over et møte på Zoom. Den viste seg nyttig i utprøvingen av lydopptaker, samt utprøving av reserveopptak til intervjuet. Mens den ene studenten styrte intervjuet og samtalen, noterte den andre ned informasjon og forslag til mulige endringer. Dette viste seg å være en god løsning, og vi ser på det som et positivt tiltak da dette førte til en erfaringsutveksling mellom en aktiv og inaktiv deltaker i intervjusituasjonen. Vi skapte oss en større trygghet i intervjurollen da det var forskjell på tidligere erfaring fra intervjusituasjoner. Etter gjennomføring av pilotintervjuet fikk vi større tiltro til spørsmålene vi hadde utformet i intervjuguiden og gjorde endringer for å heve kvaliteten på intervjuguiden (Høgheim, 2020, s. 165). Det ble gjort endringer som vi gjorde knyttet til rekkefølgen på spørsmålene slik at det var en tydeligere rød tråd gjennom samtalen. I tillegg fikk vi avdekket spørsmål som ikke egnet seg eller svarte på problemstillingen vår.

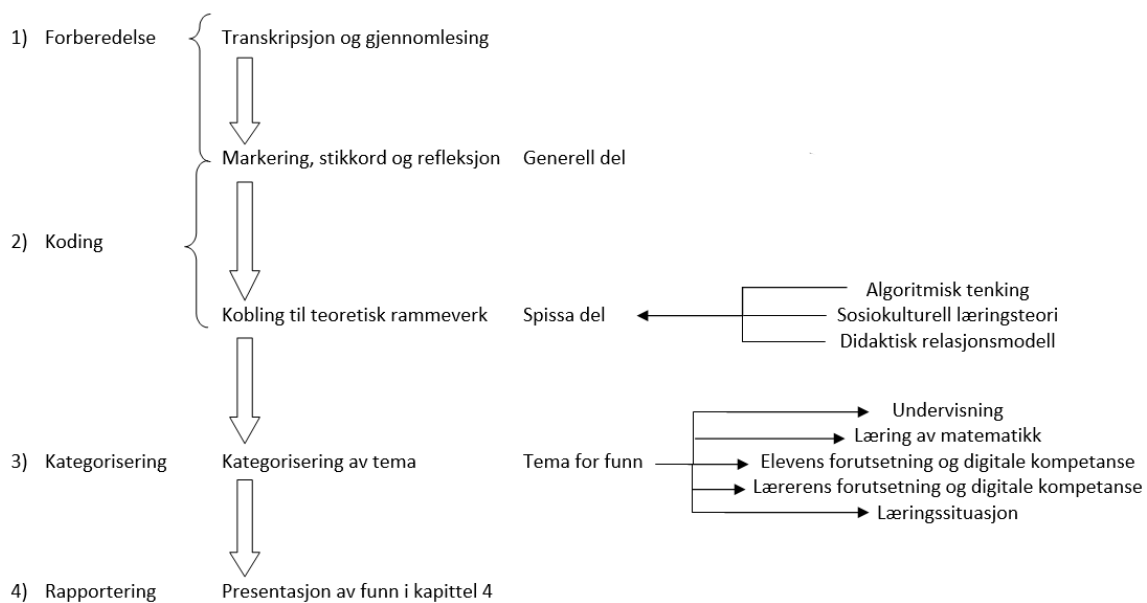
3.2.4 Gjennomføring

Før vi satte i gang med datainnsamlingen, var vi opptatt av et godt forarbeid hvor vi hadde fokus på intervjuets rammer og intervjuguide. Ved å gjøre oss gode erfaringer med Super:bit i forkant av intervjuene, bidro til å skape en større forståelse for hva lærerne prater om i intervjuene. Ettersom at vi har prøvd å forberede oss til intervjuene, kan dette ha ført til at kvaliteten på dataen som er samlet inn ble bedre. I tillegg kan det ha dannet et grunnlag for et etterarbeid som kan styrke dataene som vi har samlet inn gjennom de ulike intervjuene (Brinkmann & Kvale, 2015, s. 125). Da vi kom til gjennomføringen av intervjuene, valgte vi å ta to intervju hver. Vi var interessert i å få

utarbeidet ulike synspunkt gjennom de 4 intervjuene vi hadde. For å videre skape et større eierskap og kontroll på dataene transkriberte vi hverandres intervju slik at vi da har god kontroll på alle 4 intervjuene. Intervjuene ble gjennomført på skolene til lærerne og gjennom intervjuene var intervjuguiden et ledende dokument. Det semistrukturerte intervjuet la derimot til rette for interessante vinklinger gjennom intervjuene som baserer seg på det som fanger interessen til intervjuer og lærer, innenfor rammene som er satt. Rammene som ble satt begrenser seg til hva vi er ute etter i lys av problemstilling og tidsrommet som gjorde at intervjuene og informasjonen følte fullstendig ut gjennom en tidsramme på opp mot 60 minutter. Etter godkjenning av NSD (Vedlegg 1) og skriftlig samtykke fra lærerne ble det brukt lydopptak sammen med et reserveopptak at vi sikra lydopptakene. Disse ble videre lastet opp på HVL sin sikrede database for videre håndtering til transkriberingen. Transkriberingen ble skrevet i den grad ordrett, sett bort fra dialekt. Pauser ble merket med «...» og videre fikk vi anonymisert dokumentene slik at lydfilene deretter kunne slettes for å sikre personvernet til lærerne.

3.3 Analyse av intervju

I dette avsnittet skal vi gå gjennom hvordan vi har arbeidet i analyseprosessen for å få et mest mulig virkelighetsnært bilde av det lærerne har gitt oss gjennom intervjuene. På bakgrunn av at vi bruker tematisk analyse med inspirasjon fra samarbeidet i kollektiv kvalitativ analyse har vi valgt å gå grundig gjennom hva vi har gjort for å gi et tydelig innblikk i framgangsmåten til denne analysen. I denne sammenheng har vi lagd tabell 2, som gir et overblikk over de ulike delene av prosessen som vi videre skal gå inn på. Figuren viser de ulike fasene av analysen (venstre del av figuren). I den høyre delen av figuren kommer det fram hvordan vi har brukt det teoretiske rammeverket i analysen og videre kategorisert tema for funn.



Tabell 2: Den analytiske framgangsmåten og arbeidsprosessen i analysen.

Når vi skulle analysere transkripsjonene fra intervjuene våre var det et arbeid for å avdekke generelle eller typiske mønster i datamaterialet vårt (Grønmo, 2016, s. 266). I dette arbeidet valgte vi å benytte oss av en tematisk analyse. Tematisk analyse er en måte å identifisere, analysere og skildre tema i et datamateriale. Å benytte en analytisk metode gjør at man i en større grad kan sikre oss at analyseprosessen blir grundig og systematisert gjennomført (Johannessen et al., 2018, s. 282). I tillegg hentet vi inspirasjon fra kollektiv kvalitativ analyse der samarbeid er en viktig faktor. Her hentet vi tips til arbeid i fellesskap når vi skulle gjennomfører den tematiske (Eggebø, 2020, s. 109). Denne inspirasjonen var viktig i prosjektet for å kunne styrke analyseprosessen gjennom å være to (Eggebø, 2020, s. 109). Det at vi er to studenter, og derav to stykker som analyserte datamaterialet

for denne masteroppgaven sammen, kan sees på som en strategi for å håndtere forvirring og det kan bidra til å styrke kvaliteten i analysearbeidet vårt.

Tematisk analyse innebærer en analyse hvor vi ser etter temaer i datamaterialet vårt, der temaene er en gruppering av data med viktige fellestrekk. Videre skal temaene grupperes i mer generelle kategoriseringer, og målet er at temaene sammen med kategoriseringen skal gi svar på problemstillingen vår (Johannessen et al., 2018, s. 279). På denne måten kan det hjelpe oss å skape orden i dataene, samt finne nye sammenhenger i datamaterialet (Johannessen et al., 2018, s. 280). Tematisk analyse er delt inn i fire faser for å lete i datamaterialet vårt. De fire fasene er: 1) forberedelse, 2) koding, 3) kategorisering og 4) rapportering. Fase 1, 2 og 3 kommer vi nærmere inn på i avsnittene nedenfor, og fase 4 blir gjort rede for i kapittel 4.

3.3.1 Forberedelse- transkripsjon og koding

I første steg i tematisk analyse, forberedelse, begynte vi med å lytte gjennom alle fire intervjuene hver for oss, uten å ta notater. Videre tok vi fatt på arbeidet med å transkribere lydopptakene vi hadde fra intervjuene. Transkripsjonen ble skrevet i Word ved hjelp av tabeller. Tabellene bestod av to kolonner og x antall rader avhengig av antall utsagn i intervjuet. I første kolonne står det hvem som kom med utsagnet, og i andre kolonne står selve utsagnene. Vi sørget for å transkribere et av våre egne intervju, og et intervju som den andre hadde gjennomført. Dette er ifølge Eggebø (2020, s. 119) en del av en kollektiv kvalitativ analyse, og gir oss som forskere et større innblikk i, og grunnlag for å tolke og analysere intervjuene. Det hjalp oss til å skaffe en god oversikt over hvilke data vi skulle arbeide med før vi startet selve analysearbeidet. Videre i forberedelsesfasen leste vi gjennom transkriberingen i Word og skrev ned stikkord i dokumentet. Stikkordene vi skrev ned, skulle hjelpe oss til å vise vei til deler av intervjuene som vi syntes var spesielt interessante. Disse stikkordene var knyttet til deler av algoritmisk tenkning, sosiokulturell læringsteori og den didaktiske relasjonsmodellen. Dette var begreper vi på forhånd hadde sett på som aktuelle på grunn av arbeidet med tidligere forskning. Her oppdaget vi at de tre delene av teorigrunnlaget vårt ble hyppig nevnt i studiene knyttet til programmering og matematikk. Samtidig som vi skrev ned stikkord, fant vi sitater som vi syntes var spesielt interessante, og markerte dem med farge i Word. Dette gjorde vi individuelt for å ikke bli påvirket av hverandres arbeid. Vi gikk deretter sammen for å gå gjennom arbeidet vi hadde gjort individuelt. Dette resulterte i et Word-dokument med en oversikt over sitater fra intervjuene. Arbeidet beskrevet i dette avsnittet samsvarer med hva Johannessen et al. (2018, s. 284–285) markerer ut som viktige elementer i forberedelsesfasen av tematisk analyse.

3.3.2 Koding og kobling til teoretisk rammeverk

Andre steget i tematisk analyse, er koding. Her startet vi individuelt å kode ett og ett intervju. Vi valgte å gjennomføre alle intervjuene før vi samkjørte oss for å hele veien jobbe i retning av problemstillingen vår. Videre markerte ut sitater som vi så som hensiktsmessig opp mot problemstillingen. Tanker vi gjorde oss knyttet til det teoretiske rammeverket, skrev vi ned som stikkord og refleksjoner i margin (Johannessen et al., 2018, s. 285). De utvalgte delene av det teoretiske rammeverket vårt er algoritmisk tenkning, sosiokulturell læringsteori og den didaktiske relasjonsmodellen. Vi gikk videre sammen for å samkjøre arbeidet vårt. Her åpnet vi opp for litt diskusjon om hva som er viktige temaer og relevante tolkninger av datamaterialet. Dette resulterte i en punktliste som gav oversikt over lærerens uttrykk gjennom intervjuet, og videre brukt til å produsere et sammendrag. I denne prosessen var poenget å framheve og sette ord på viktige poenger i datamaterialet vårt, samt gi datamaterialet vårt en dypere innsikt (Johannessen et al., 2018, s. 284). Når vi valgte å samkjøre arbeidet vårt er dette ifølge Eggebø (2020, s. 107) en effektiv, grundig og pålitelig måte å starte analysearbeidet på. Dette kan bidra til å balansere våre individuelle synspunkter, samt styrke kvaliteten i analysearbeidet vårt (Jupp, 2006, s. 306). Å analysere sammen kan skape rom for en kreativ analytisk prosess hvor vi som forskere kan korrigere hverandre, og videreutvikle tolkninger og lære av hverandre (Eggebø, 2020, s. 106).

Når vi hadde gått gjennom alle fire intervjuene med måten beskrevet ovenfor, skrev vi ut transkripsjonene som inneholdt markeringer av viktige poeng, stikkord og refleksjonsnotater. Vi gikk individuelt gjennom alle utskriftene med intervjuene, og tok for oss ett og ett tema. Temaene som vi koder opp mot tar utgangspunkt i de tre delene fra teorigrunnet, altså algoritmisk tenkning, sosiokulturell læringsteori og den didaktiske relasjonsmodellen. Ved å først arbeide individuelt med utskriftene unngikk vi i stor grad å bli påvirket av hverandre og hverandres notater før vi skulle samkjøre oss igjen for å balansere våre individuelle synspunkter. Vi gikk inn i datamaterialet vårt med bestemte spørsmål i tankene for å unngå at kodingen ble ubrukelig og vilkårlig (Johannessen et al., 2018, s. 285). Spørsmålene var knyttet til temaene i det teoretiske rammeverket. Vi valgte en bred tilnærming når vi gikk inn for å se på disse temaene i transkripsjonene. Dette gjorde det enklere å plukke ut deler fra datamaterialet som passet inn under punktene i de utvalgte temaene våre. Vi skal nå gå inn på punktene vi kodet opp imot på de tre ulike temaene.

Algoritmisk tenkning

Algoritmisk tenkning er en problemløsningsmetode og et kjent tankesett som er nært knyttet til arbeidet med programmering (Angeli et al., 2019, s. 86). Vi valgte videre å basere undertemaene på Brennan og Resnick (2012, s. 2–3) sin definisjon av algoritmisk tenkning. Der er det tre dimensjoner

av algoritmisk tenkning: algoritmiske begrep, algoritmiske øvelser og algoritmiske perspektiv (Brennan & Resnick, 2012, s. 2–3). Vi endte derfor opp med fire punkt til algoritmisk tenkning:

- Algoritmiske begrep
- Algoritmiske øvelser
- Algoritmiske perspektiv

Sosiokulturell læringsteori

For å forstå hvordan elevene lærer var det viktig for oss å se datamaterialet vårt inn mot et læringsperspektiv. Da var det naturlig å velge sosiokulturell læringsteori. Her har vi valgt sentrale elementer fra læringsteori som punkter i tillegg til synspunkter som er kommet fram gjennom intervjuene. Vi har valgt sentrale begreper knyttet til læringsteori som den proksimale utviklingssonen, stillasbygging og læring som fellesskap. Gjennom intervjuene kommer det fram hvordan lærerne hjelper elevene, samt hvordan lærere i dialog med elever legger til rette for læring av matematikk. Vi så derfor på veiledning som et sentralt punkt. I tillegg har vi valgt å ta med medierende artefakter da de pedagogiske robotene kan fungere som et artefakt. Vi endte derfor opp med seks punkter:

- Proksimal utviklingssone
- Stillasbygging
- Veiledning
- Medierende artefakter
- Lærings som fellesskap
- Mediering

Den didaktiske relasjonsmodellen

Den didaktiske relasjonsmodellen utviklet av Bjørndal og Lieberg (1978) skal fungere som en effektiv planlegger av undervisning der den spiller på lærerens erfaringer i undervisningssammenheng (Bjørndal & Lieberg, 1978, s. 136–137; Gudem, 1998, s. 47). Den seksdelte modell danner rammeverket for punktene som vi ser etter i analysen.

- Elevforutsetninger
- Rammefaktorer
- Mål

- Innhold
- Læringsaktiviteter
- Vurdering

Etter arbeidet som er beskrevet i det foregående arbeidet gikk vi sammen for å gå systematisk gjennom hvert intervju for å skrive ned de aktuelle sitatene fra intervjuene inn i en tabell i Word. Vi diskuterte hva som var viktige temaer og hva som var relevante tolkninger knyttet til sitatene. Det var flere sitater som ble brukt på forskjellige steder, og derfor sørget vi for å ha ulike fargekoder slik at vi hadde kontroll på hvilke sitater vi hadde brukt flere ganger. Dette var viktig for prosessen videre inn mot kategorisering av kodene våre.

3.3.3 Kategorisering

Når vi var ferdig med (2) koding, gikk vi tilbake inn i intervjuene for å se på alle sitatene og notatene vi hadde satt inn under flere emner. Vi noterte ned alle sitatene som hadde blitt skrevet ned på ulike punkt knyttet til det teoretiske rammeverket vårt, på arkene fra kodingen. Vi så videre på hvilke temaer som hadde en tilknytning til hverandre gjennom et sitat.

I den neste fasen av kategoriseringen var målet å markere ut sitatene som vi hadde notert oss i forrige del. For å arbeide med dette brukte vi tabellfunksjonen i Word. Vi hadde ett Word-dokument for hvert av temaene vi hadde valgt ut: sosiokulturell læringsteori, algoritmisk tenkning og den didaktiske relasjonsmodellen. I tabellene viser kolonnene til de ulike punktene som var satt av det teoretiske rammeverket. Radene viste til ulike sitater fra de ulike lærerne. I tabell 3 ser vi en oversikt over hva de ulike markeringsfargene betyr i tabellene som er vist til nedenfor. Videre ligger det et utsnitt fra Word-dokumentene for det teoretiske rammeverket vi har basert kodingen på. Arbeidet her gjorde vi sammen, og dobbeltsjekk etter vi var ferdig. Dette var starten på kategoriseringen. Målet var å gjøre kodene om til mer generelle temaer (Johannessen et al., 2018, s. 282).

Fargekoder:

Algoritmisk tenkning

- Definisjon → Grå
- Algoritmiske konsept → Lyseblå
- Algoritmiske øvelser → Blå
- Algoritmiske perspektiv → Mørkeblå

Sosiokulturell læringsteori

- Proksimal utviklingssone → Lysegrønn
- Medierende artefakter → Grønn
- Veiledning → Mørkegrønn
- Stillasbygging → Lysebrun
- Læring som fellesskap → Brun
- Andre → Mørkebrun

Den didaktiske relasjonsmodellen:

- Arbeidsmåter → Lysegul
- Forutsetninger → Gul
- Innhold → Mørkegul
- Mål → Lyserosa
- Rammefaktorer → Rosa
- Vurdering → Mørkerosa

Tabell 3: Fargekoder for oppmerking av sitat på kryss av dokument.

	Den prososiale utviklingssonen	Medierende artefakter	Veiledning	Stillasbygging	Læring som fellesskap	Andre
Lærer 1	«Trenner egentlig litt sån å komme i gang, veiledning og rammer rundt ... de finner jo veldig masse ut av det selv faktisk»	«Sånn microbil, bil, baner og litt sånn forskellig»	«Sånn om jeg ikke er noen ekspert, klarer jeg å lese instruksene ... og hjelpe dem litt der»	«Vi har mange digitale løsninger som vi bruker i skolen min»	«Da siter de gjerne to og to sammen» «Dette er såpass nytt og hvis du ikke har drevet masse med det selv, så må du bare øve deg litt og lære sammen med eleven» «Det er god læring for dem og da, å lære bort det de kan»	«Vi brukte elever fra kodeklubben som hjelpelærere»
Lærer 2	«da jobber de i små grupper i eget tempo med å følge instruksjonsvideo» Differensiering	«fordi de møtt de tingene i hvertfall som vi har hatt opp på hørtfellige møter på skolen» «kommer, liker de at det er noe praktisk og noe bygging og fysisk som skjer istedenfor at det er på skjerm» Bruke programmeringsroboter for å lære matematikk «Det er alt i matematikken som kan knyttes opp. Alt fra grader til omdreining, tid, distanse, sekunder, måling, volum og areal» «Sylinderball i nærmeste skole i kommunen, de bruker til å tegne geometriske figurer» «fordi de med iPad og appene er at ungene er så intuitive som nesten skjøner det ved å trykke, dra og slippe, det er nesten bare å vise hvilken app de skal inn på så skjønner de det nesten bare helt selv»	«Jeg bryr meg ikke så mye med det siden jeg har holdt på så mye med det, fordi jeg vet jeg klarer å finne en løsning eller nødløsning på det, men for mange av lærerne så er det det store løberet»	Differensiering «da jobber de i små grupper i eget tempo med å følge instruksjonsvideo»	«da jobber de i små grupper i eget tempo med å følge instruksjonsvideo» «tørre å ta det til eleven og være med på reisen og finne det ut selv, som jeg synes fungerer vel så bra ofte» «med bilen har de hatt muligheten til å lage dans, der alle programmerer og prøver å synce, da er det seks stykker som prøver å få de til å gjøre alle bevegelsene samtidig og synkront»	«Så en må ha med seg alle disse basistingene, om en ikke kan få programmet til å virke»

Tabell 4: Utklipp fra sitatene knyttet opp mot punktene i sosiokulturell læringsteori.

I tabell 4 er det gitt et utsnitt fra arbeidsdokumentet der vi så på fargekodene fra forrige trinn for å se sammenhengene. Når vi var ferdig med å merke sitat som hadde blitt brukt på ulike temaer, satt vi opp et Excel-dokument der vi skrev ned hvilke temaer som har blitt brukt til samme sitat.

Vi sorterte deretter dokumentet for å få en oversikt over hvilke temaer som kunne høre sammen og skrev beskrivelse til disse. Videre prøvde vi å fargekode de ulike beskrivelsene for å se hvilke rader som kunne ha en sammenheng vist. I neste del lagde vi et nytt ark i Excel-dokumentet. Der sorterte vi temaene ut ifra beskrivelsen vi hadde gjort. I denne fasen diskuterte vi om hvordan vi kunne gruppere de ulike temaene vi hadde kommet fram til (Eggebø, 2020, s. 115–117). Vi endte her opp med seks ulike kategorier.

	Didaktisk relasjonsmodell	Sosiokulturell	Antall	Forlag tema	Beskrivelse	Problemløsning
Algoritmisk tenking	Forutsetning	Artefakt	2	Elevenes forutsetning og digitale kompetanse	Innitliv, digital kompetanse, erfaring, intuitiv og potensielt til prog, kompetanse elev og lærer	Problemløsning
Øvelser	Arbetsmøter	Proksimal utviklingszone	1		lite lærerstyrt og elevene er intuitive	Erfaring
Øvelser	Arbetsmøter	Fellesskap	1		Gruppedevid, differensiering	Erfaring
Øvelser	Forutsetning	Artefakt	1		Ind og intuitiv	Erfaring
	Forutsetning	Veiledning	2	kunnskapsbui og kompetansenutvikling	Digital kompetanse og trygghet (læreren)	Erfaring
	Forutsetning	Andre	1	lærerenrolen, kompetanse, utskilling i skolen	Interesse for å lære elevene	Erfaring
	Forutsetning	Fellesskap	1		Løse sammen med elevene	Erfaring
Øvelser	Rammenaktør	Fellesskap	1		Løse kompetanse i kollegiet. (se intervju+rammer)	Erfaring
Øvelser	Innhold	Fellesskap	4	Undervisning	Oppgavevalg, problemløsning,	Legge til rette
Øvelser	Arbetsmøter	Artefakt	2	ustyrt, form og opplegg	Gruppearbeid og inspirasjon fra andre	Legge til rette
Øvelser	Innhold	Fellesskap	2	praktisk arbeid	pedagogiske robot og tegning	Legge til rette
Konsept	Arbetsmøter	Fellesskap	1		Dans med pedagogisk robot og synt. (UNDETVINGSFORSLAG)	Legge til rette
Konsept	Innhold	Fellesskap	1	Underetema?	Og/eller, variabler og løkker. Valfelleksempe	Legge til rette
Konsept	Arbetsmøter	Fellesskap	2		tegne plier på tavla og programmere elevene (KOBLES TIL VAFTELKSEMPEL)	Legge til rette
Konsept	Mål	Fellesskap	1		forstå for å kunne bruke kompetene (kople til variert, pill på tavle)	Legge til rette
Perspektiv	Arbetsmøter	Fellesskap	5	læringsituasjon	Samarbeid, lære av hverandre, praktisk VS teoretiske	Erfaring
Perspektiv	Arbetsmøter	Veiledning	2	elemenutvikling, elevenes rolle, praktisk vs teoretisk	praktisk arbeid	Erfaring
Perspektiv	Arbetsmøter	Veiledning	1	interesse og motivasjon kanskje?	assisterende oppgaver.	Erfaring
Perspektiv	Innhold	Artefakt	1		Kollega/kommunere bra med praktisk arbeid	Erfaring
Perspektiv	Mål	Artefakt	1		forstå hvordan ting fungerer rundt oss	Erfaring
Perspektiv	Mål	Veiledning	1		Andre byrer elevene skilne, og utfordre sterke elever på en annen måte+ interesse.	Erfaring
Øvelser	Arbetsmøter	Veiledning	5	læring av matematikk	Gruppearbeid, praktiske oppgaver, matematikk som innhold	Legge til rette
Øvelser	Mål	Artefakt	1	innhold, mål, arbeidsform	fart og tid sammenheng (læring av matematikk)	Legge til rette
Øvelser	Mål	Artefakt	1		Lekreget, og vanskelig å sette mål, kompetanse?	Legge til rette
Øvelser	Mål	Andre	1		Grunnleggende matematikk for forståelse for programmering.	Legge til rette

Tabell 5: Utklipp av Excel-ark med inndeling av kategorier.

I det tredje arket (tabell 5) sammenfattet vi enda to tema og ble sittende med fem kategorier. Disse kategoriene danner grunnlaget for de fem hovedfunnene våre. Det var viktig for oss at vi i denne delen av analysen åpnet opp for diskusjoner, uenigheter og forskjellige tolkninger av datamaterialet (Eggebø, 2020, s. 115–117). Dette var nødvendig ettersom vi hadde ulike tanker og tolkninger av hvordan vi skulle dele opp og sortere temaene. Her prøvde vi å se en antydning på hvilke deler som kunne bli knyttet opp til hvilken del av problemstillingen, før vi satt i gang med siste del.

Lærer 1	<p>Eleven fortsetter med 1. DK</p> <p>«Mens ehli, ehli, kanskje flere av guttene gir masse, altså driver masse mye mer med data generelt da, og så de har hatt veldig stor glede av det.»</p> <p>«de er nødt til å være nøkkelige. Fordi... ehli... og det er jo en ting som er litt vanskelig for eleven og. Det å være skikkelig nøkkelig. Det kan bli litt sånn "jeg har jo trykket akkurat riktig kode, og hvorfor går det ikke?" Nei da er jo bare en litt sånn liten detalj da, som gjør til at det ikke fungerer»</p> <p>«det er egentlig lite lærerstyrt... de finner jo veldig masse ut av det selv, faktisk»</p> <p>«noen er jo såne utforskere, som da begynner å "jamen det må være den, og hvis jeg flytter den og slik»</p>	Lærestoffet fortsettes	<p>«53 selv om jeg da ikke er noen ekspert klarer jeg å lese instruksene og hjelpe dem litt der»</p> <p>«Å kanskje vi bare skal hoppe i det»</p> <p>«Gikk inn på alle mulige ting, og prøvde og feila» (LEFFEREN)</p> <p>«Nei, asså. Jeg kunne ingenting om det, og kan ikke kjempemasse om det nå heller»</p> <p>«Vi er ikke noe ekspert, i det hele tatt, vi fanter litt sånn i blinden, men vi tenker at ehlihm. Alternativet er ingenting. Så vi må jo bare prøve»</p> <p>«Jeg tror at mange synes det er veldig skummelt siden det er helt nytt, sånn at de begynner med det? Dere er helt gamle. Ja, men kan dere det da? Nei, har ikke peiling, men vi må bare hoppe i det, samt»</p>	Undervisning	<p>«Vi gjorde egentlig det samme som vi gjorde på kurset. Også har vi brukt det åa etter, og i ettertid. Og funnet informasjon vi der, og funnet oppskriften på forskjellige nyder, og at vi har utstyrt og slik da»</p> <p>«Det var hvertfall forskjellige oppskifter det som vi printet ut»</p> <p>«Da gikk vi gjerne igjennom litt først og så ble det alltid litt sånn praktisk»</p>	Underetema oppskrift	<p>«Vi har brukt det litt sånn i forhold til hva er en algoritme, en oppskrift på matretter og litt såne ting, men det er ikke noe vi har veldig forhold til»</p> <p>«Følge instruksjoner og være nøkkelige»</p> <p>«Nei, det aller fleste var det jo stegvis»</p> <p>«... det med å lese instruksjoner og slik ligger jo der... lese en oppskrift»</p> <p>«Vi fant masse inn sånn oppskifter på forskjellige nivå inne på den super-biten»</p>	Lærings situasjon	<p>«Vi gjorde det egentlig sammen med elevene»</p> <p>«Men det er jo god læring for dem og for å lære bort det de kan»</p> <p>«Også er det jo noen som er veldig flinke som da får en liten sånn assistentrolle. Hoppe som et spesielt ansvar. I tillegg litt «kanne»»</p> <p>«da var det egentlig elevene som komme det, som lærte oss, og så prøvde vi selv da»</p> <p>«Vi brukte elever fra kodeklubben som hjelpelærere»</p>	Læring av matematikk	<p>«da sitter en gjerne to og to sammen sant, en må springe og... måle avstand og regne litt og»</p> <p>«vi sitter litt med å koble det litt mot matematikkaget... at det blir litt mer som lek»</p> <p>«men særlig det da man drev på med bilen, ikke sant, med fart og tid og avstand og... der ble det jo masse matte»</p> <p>«men hva er hensikten, vi mangler liksom formålet.»</p>
---------	---	------------------------	--	--------------	--	----------------------	---	-------------------	--	----------------------	--

Tabell 6: Sitat koblet opp mot kategorier.

I neste del brukte vi temaene som vi hadde satt opp i Excel-arket for å lage et Word-dokument (Tabell 6). Tabellen hadde det samme oppsettet som de foregående tematiske tabellene (Tabell 4). På bakgrunn av størrelsen til undervisning valgte vi å dele den opp i to, dette gikk vi senere tilbake på så disse blir sett som en. Vi satt deretter sitatene som var knyttet til Excel-arket (Tabell 5) og inndelingen av de nye temaene for å sette retningslinjer på hva temaene inneholdt. Etter det så vi over resten av de originale filene fra de overordnede temaene til teorigrunnet og satt inn sitatene til de nye temaene slik at vi har samlet sitatene som vi ser på som nyttige opp mot problemstillingen vår innenfor fem felt. Videre så vi på hvilke av de fem temaene som kunne knyttes opp mot hvilken av de to delene til problemstillingen vi har satt oss i oppgaven. Dette vurderte vi videre for å kunne komme fram til funnene i denne oppgaven.

3.4 Validitet og reliabilitet

I dette kapittelet skal vi presentere ulike deler knyttet til validiteten og reliabiliteten til prosjektet vårt. Dette handler om troverdigheten til prosjektet og hvorvidt de innsamlede dataene kan gi et gyldig og sannferdig uttrykk (Befring, 2020, s. 37; Christoffersen & Johannessen, 2012, s. 24). Lincoln og Guba (1985) beskriver fire troverdighetskriterier som er med på å belyse hvordan en kan styrke validiteten og reliabiliteten til et prosjekt (Schwandt et al., 2007, s. 12). Disse kriteriene har vært en sentral del i konkretiseringen av de metodiske valgene vi har gjort i prosjektet vårt (Lincoln & Guba, 1985, s. 218). De fire kriteriene som blir beskrevet er intern validitet, ekstern validitet, pålitelighet og objektivitet.

3.4.1 Intern validitet

Internvaliditet handler om troverdigheten til prosjektet. Måten vi har prøvd å legge til rette for dette i vårt masterprosjekt er gjennom to ulike former for triangulering (Lincoln & Guba, 1985, s. 291). I vårt prosjekt er det triangulering av forskere og data vi baserer den interne validiteten til prosjektet på (Lincoln & Guba, 1985, s. 291). Forskertriangulering henger sammen med hvordan vi har gjennomført intervjusituasjon og den tematiske analyse med inspirasjons fra samarbeidet i kollektiv kvalitativ analyse (Eggebø, 2020, s. 107). Gjennom å ta to individuelle intervjuer hver, sørger vi for å kunne balansere ut våre individuelle synspunkter (Jupp, 2006, s. 306). Og vi har da videre, gjennom flere samtaler og diskusjoner i analyseprosessen, kommet fram til et produkt av analysen som vi i større grad mener er fri for hverandres subjektive synspunkter. Den kollektive kvalitative analysen mener vi er en styrke i prosjektet da vi i fellesskap jobbet både individuelt og sammen for å styrke kvaliteten på metoden og analysen som er gjort (Eggebø, 2020, s. 118). Dette har blitt en viktig strategi for oss for å hindre forvirring og styrke analysearbeidet sin kvalitet (Tjora, 2017, s. 251–252). I tillegg har vi hentet inn data gjennom intervju av ulike matematikklærere fra forskjellige deler av Sør-Norge for å kunne se disse opp mot hverandre (Jupp, 2006, s. 306). Disse ulike dataene har vi da videre brukt opp mot de ulike delene som danner teorigrunnet for å styrke validiteten på studiet. Vi har valgt å sette oss godt inn i det teoretiske rammeverket som vi baserer vår analyse av intervjuene på: algoritmisk tenkning, sosiokulturell læringsteori og den didaktiske relasjonsmodellen. Vi ser disse i lys av hverandre for å prøve forstå sammenhengen til hvordan analysen kan knyttes opp til teorien (Jupp, 2006, s. 306).

3.4.2 Ekstern validitet

Ekstern validitet handler om overførbarheten til prosjektet og hvorvidt dette kan gjenskapes (Lincoln & Guba, 1985, s. 291). Ekstern validitet er det kriteriet som vi ser på som mest krevende for å styrke oppgaven. Likevel har vi prøvd å styrke den eksterne validiteten gjennom et så tilfeldig som mulig utvalg av lærere. Dette for at det i større grad skal være mulig å illustrere populasjonen. Vi har prøvd

å få til dette gjennom et utvalg og sampling basert på to kriterier: (1) lærerne skulle være matematikklærere på mellomtrinnet som (2) har undervisningserfaring med Super:bit gjennom programmering. Når vi valgte ut lærere til intervjuene våre ble dette gjort ut ifra en typiske-tilfeller-sampling med forsøk på å få matematikklærere som kan dekke tanker og erfaringer til bruk av Super:bit i matematikkundervisning. Vi har et utvalg som kommer fra fire forskjellige skoler i ulike deler av Sør-Norge. Videre har vi gjort et forsøk på å generalisere slutningene i utvalget for å kunne prøve å finne en middelvei gjennom de kriteriene som er satt (Høgheim, 2020, s. 154). Det er viktig å huske at sampling henger tett sammen med generalisering, men vi kan ikke ut ifra samplingen si at dette kan generaliseres (Høgheim, 2020, s. 155–156).

3.4.3 Pålitelighet

Påliteligheten til oppgaven handler om reliabiliteten og er en forutsetning for validiteten i prosjektet (Lincoln & Guba, 1985, s. 292). Påliteligheten omhandler studiet og empirien en samler inn, og i kvalitativ forskning kan det subjektive skjønnet legge til rette for upresise data (Befring, 2020, s. 45). For å hindre dette har vi hatt fokus på å gjøre intervjuene på likest mulig måte, med unntak av noen avsporinger for å reflektere erfaringene lærerne har skapt i praksis. Påliteligheten til studiet styrkes gjennom en godt utarbeidet intervjuguide (Vedlegg 3) og dokument med transkripsjon som brukes som. Det er vanskelig å reprodusere kvalitative intervju da de tar for seg menneskers oppfatninger og meninger (Høgheim, 2020, s. 153). Vårt fokus på å ha en god intervjuguide som blir fulgt gjennom et semistrukturert intervju i samhandling med transkripsjonene er forsøk på å øke gjennomføringsreliabiliteten. Dette omhandler den forskningsmetodiske dokumentasjonen (Befring, 2020, s. 47). Dette gjør det mulig å følge prosedyrene i senere forskningsarbeid. På en annen side er det en liten gruppe lærere som har blitt intervjuet i studiet vårt, så vi kan ikke generalisere slutningene og si at andre vil komme fram til samme resultat. Likevel ser vi på det som rimelig at lærerne som har deltatt i prosjektet har tanker og erfaringer som kan kjennes igjen hos andre lærere som har undervist med Super:bit på mellomtrinnet i matematikkfaget.

3.4.4 Nøytralitet

Objektiviteten til studiet vårt handler i stor grad om samspillet vi to har hatt gjennom forskningsprosjektet. Det er viktig at en er klar over egne erfaringer i arbeidet med lærerne og data, slik at en er klar over hvilke styringer ens synspunkter og holdninger kan ha til videre tolkninger gjennom intervju og analyseprosessen (Lincoln & Guba, 1985, s. 173). Gjennom et forskningsprosjekt kan det å stille seg nøytral være vanskelig med synspunkter og bakgrunn ulike personer har. Det er derfor et viktig element en må ta inn over seg i arbeidet med prosjektet. Vi opplevde derfor at gjennom å være to personer, der vi delte opp intervjuene og valgene som vi gjorde i forhold til både individuell analyse, men også refleksjoner, samtaler og diskusjoner til de ulike valgene vi gjorde når vi

analyserte samme tema, ga oss et større innblikk i våre egne erfaringer. I tillegg var det en viktig faktor for å kunne holde våre egne erfaringer, synspunkter og perspektiver nøytrale gjennom arbeidet med data og lærere (Jupp, 2006, s. 306).

3.5 Etikk

Som vi nylig har gått gjennom har vi valgt en kvalitativ tilnærming med intervju for å prøve å svare på problemstillingen vår etter beste evne. Det var derfor viktig for oss å bruke lydopptak slik at vi i ettertid kunne representere lærernes tanker og erfaringer på en best mulig måte. På bakgrunn av at det var ønskelig å bruke lydopptak, som blir regnet som en personopplysning, var prosjektet vårt meldepliktig til NSD (Norsk senter for forskningsdata). Vi sendte derfor inn søknad og fikk dette godkjent før vi startet datainnsamling (Vedlegg 1). Her sendte vi inn samtykkeskjema og intervjuguide, og beskrev hvordan vi skulle behandle dataen vår.

I forkant av intervjuene har alle lærerne fått tilsendt et informasjonsskriv som beskriver prosjektet og hva det vil ha og si for dem å være med. Det har hele veien vært mulighet for at lærerne trekker seg uten videre spørsmål til avgjørelse, dette for at prosjektet skal være frivillig å delta på. Det skal ikke være noen form for ytre motivasjon for å delta, men en egeninteresse for å dele sine erfaringer fra programmering i matematikkundervisning (Høgheim, 2020, s. 88–89).

I informasjonsskrivet (Vedlegg 2) stod det grundig informasjon om hvordan intervjuet ville gjennomføres, hvordan den ville ha betydning for dem som deltaker, og hvilke rettigheter de hadde både før, under og etter gjennomføring (De nasjonale forskningsetiske komiteene, 2015). Dermed har vi prøvd å gi dem tydelige instruksjoner for hva de skal gjøre dersom de ønsker å trekke seg, har spørsmål til studien eller ønsker innsyn. Alt av datamaterialet ble anonymisert underveis, og vi valgte å bruke benevnelsen han/hun for alle våre lærere uavhengig av kjønn. Alle opptak ble i kort tid etter intervjuene transkribert, og hvert enkelt intervjuobjekt ble anonymisert slik at lærerne ikke kunne identifiseres. Dette er gjort i tråd med NSD sine forskningsetiske retningslinjer.

Vi har gjennom hele forskningsprosjektet prøvd så godt vi kan å holde oss innenfor NESH sine retningslinjer for forskningsetikk (De nasjonale forskningsetiske komiteene, 2016) og behandlet dataene med omhu i forhold til å representere tanker, erfaringer og synspunkter på en anonym og virkelighetsnær måte.

4.0 Presentasjon av funn

I dette kapitlet skal vi presentere funn i studien. Funnene har kommet fram fra analysen av vårt datamateriale. Analysearbeidet synliggjorde flere temaer som vi nå skal presentere. Innunder disse temaene har studien flere funn. På bakgrunn av studiens problemstilling har vi valgt å dele kapitlet i to overordnede deler. Vi gjentar studiens problemstilling:

«Hvordan legger fire lærere til rette for læring av matematikk på mellomtrinnet gjennom bruk av pedagogiske roboter fra Super:bit-satsingen, og hvilke erfaringer gjør de seg knyttet til dette?»

Den todelte inndelingen utgjør derav kapittel 4.1 som omhandler hvordan lærerne legger til rette for læring av matematikk. Videre tar vi i kapittel 4.2 for oss hvilke erfaringer lærerne har gjort seg gjennom arbeidet med programmering i matematikkfaget. Når vi har gått gjennom disse to delene med presentasjon av funn, kommer det en oppsummering av funnene våre i kapittel 4.3.

4.1 Hvordan lærere legger til rette for læring av matematikk ved bruk av Super:bit

For å presentere funnene våre for hvordan lærerne har lagt til rette for læring av matematikk ved bruk av Super:bit, har vi valgt å dele det inn i: «Undervisning» og «Læring av matematikk». Under 4.1.1 blir det vist til funn på hvordan lærerne velger å organisere undervisningen når de bruker Super:bit, hvilket utstyr og læringsaktiviteter de bruker, arbeidsmåter og hvordan de bruker programmering for visualisering og forståelse for andre matematiske temaer. Videre i 4.1.2 blir det beskrevet mer konkret hva læreren gjør målrettet for å legge til rette for læring av matematikk for elevene, hvordan programmering brukes som et verktøy for å lære matematikk og hvordan programmering kan bidra til tilpasset opplæring i matematikkfaget. Det blir i tillegg sett på hva hensikten, målet og vurderingen knyttet til arbeidet med Super:bit er.

4.1.1 Undervisning

I denne delen blir det presentert funn på hva lærerne gjør for å legge til rette for læring av matematikk ved bruk av Super:bit i undervisning. Her blir det trukket fram ulike elementer ved undervisningen som lærerne trekker fram som viktige. Vår studie presenterer funn rundt hvordan lærerne bruker organisering av klassen som et hjelpemiddel for læring av matematikk. Dette handler om hvordan læreren strukturerer undervisningen og hvordan læreren velger at elevene skal arbeide med Super:bit. Videre uttrykker lærerne både hvilket utstyr de bruker knyttet til Super:bit og hva slags læringsaktiviteter de legger til rette for. Lærerne viser til konkrete eksempler på undervisning hvor elevene bruker Super:bit. Her trekker de fram hvordan de velger undervisningsopplegg og hvilke

arbeidsmåter de benytter seg av. Til slutt viser funnene til hva Super:bit kan bidra med inn i matematikken. Her blir det trukket fram både visualisering og forståelse av ulike matematiske begrep.

Organisering av klassen

Lærerne i studien trekker fram at organisering av klassen er betydningsfullt i arbeidet med Super:bit i matematikkundervisningen. Ulike tilnæringsmåter med tanke på organisering av klassen kan bidra til elevenes matematiske læring. Funn i studien viser til at praktisk arbeid, samt både individuelt arbeid i fellesskap og gruppearbeid er viktig å vektlegge. Flere av lærerne velger å organisere undervisningsøktene ved at de først gjennomgår aktiviteten i fellesskap, for så å arbeide individuelt eller i grupper. Etter den felles gjennomgangen vier lærer 1 tid til at elevene får arbeide individuelt på hver sin datamaskin. Her jobber elevene individuelt, og innimellom legger læreren inn avbrekk der elevene skal vise på storskjerm hva de har programmert og hvordan de eventuelt hadde gjort det. Avbrekket trekker lærer 3 og lærer 4 også fram som en viktig del av undervisningen. Lærer 1 sier dette skal hjelpe elevene til å bli inspirert av hverandre. Han/hun sier det slik: «... *Da sitter alle og jobber hver for seg, og så innimellom stopper vi opp og viser på storskjerm det de har lagd for å inspirere hverandre litt.*». Lærer 3 og 4 uttrykte det samme når det gjaldt avbrekk innimellom utforskningen i undervisningen. Her ønsket de at elevene skulle presentere litt hva de hadde gjort og hvordan de hadde kommet fram til de ulike løsningene. Målet med dette var å skape ideer hos de andre elevene, og for å skape rom for å diskutere de ulike løsningene. Det var fort gjort at elevene satt fast og ikke visste hva de skulle gjøre.

Alle fire lærerne trekker fram samarbeid som en nøkkelfaktor i arbeidet med programmering. Tre av dem vektlegger gruppearbeid eller at elevene får arbeide sammen to og to. I disse arbeidsøktene er de opptatt av at elevene skal prøve og feile sammen uten at læreren selv skal gi dem fasitsvaret på oppgaven. Lærer 4 legger opp til at elevene sitter på langbord i programmeringsundervisningen. Han/hun forklarer denne organiseringen med at elevene enklere kan samarbeide på et langbord, samt at det var enkelt å få oppmerksomheten igjen hvis det skulle vises noe på tavla, enten for at elevene skal inspirere hverandre eller for å få klarhet i noe elevene arbeider med. I intervjuet med lærer 1 sier han/hun i sammenheng med hvordan samarbeidet fungerer: «...*og da sitter de gjerne to og to sammen, sant, en må springe og ... måle avstand og regner litt og ...*». Lærer 4 trekker fram at for noen elever kan slikt samarbeid være utfordrende. Han/hun sier: «... *om en da skal jobbe to og to så er det det å vite om en av de gjør jobben eller om noen bare henge på lasset og faktisk lærer*». Dette uttrykker han/hun videre som en utfordring i lærerarbeidet når det kommer til elevenes læringsutbytte av programmeringsaktiviteter i klasserommet.

Studien vår viser videre til at lærer 3 og lærer 4 trekker fram struktur som en viktig faktor i undervisning med programmering. De trekker fram struktur av timen som nødvendig for at læreren selv skal ha en opplevelse av at det er noe utbytte av undervisningen. Videre trekker lærer 4 fram at strukturen er viktig forutsetning for den matematiske læringen til eleven. Han/hun trekker fram at strukturen kan hjelpe elevene til å holde seg fokusert på det området de skal fokusere på. Lærer 3 og lærer 4 uttrykker at undervisningen som inneholder programmering er i liten grad lærerstyrt. For å finne ut av problemene som oppstår underveis i programmeringssituasjon, uttrykker lærer 1 og lærer 4 at de arbeider sammen med elevene for å finne ut av løsningene. Lærer 4 tilføyer: «*De lærer kanskje mer av hverandre enn av meg føler jeg.*».

Videre i faktorer ved organiseringen av klassen, kommer den praktiske undervisningen som et funn i studien. Her blir det trukket fram av flere lærere at den praktiske tilnærmingen til programmering er med på å bidra til læring av matematikk blant elevene. Lærer 1 får fram at han/hun bruker mye praktisk arbeid knyttet til programmering, og han/hun legger vekt på de positive sidene ved en annerledes form for matematikkundervisning. Lærer 1 viser til at det blir brukt mye samarbeid og diskusjoner når de har arbeidet på denne måten, samt at elevene har fått lov til å inspirere hverandre med å dele sine prosjekter mellom hverandre.

Arbeidsmåter

Studien viser til funn på at lærerne opplever oppskrifter på undervisningsopplegg fra internett knyttet til Super:bit som verdifullt. Alle lærerne trekker fram at de for det meste bruker ferdige opplegg fra nettet når de arbeidet med Super:bit i klasserommet. Dette er fordi det framstår som lett tilgjengelig for lærerne, og er med på å få i gang undervisning som er i tilknytning til programmering i matematikk. I tillegg til ferdige undervisningsopplegg som finnes på internett, trekker lærer 1 og lærer 4 fram at de bruker undervisningsopplegg som baserer seg på Super:bit-kurset som de selv hadde vært deltakere i. I undervisningssituasjon sier lærer 2 at det blir i stor grad brukt opplegg hentet fra nettet. Han/hun forklarer dette med at det er en effektiv måte å bruke velutprøvde undervisningsaktiviteter slik at en kan spare tid på planlegging og fokusere på andre oppgaver. Lærer 2 sier det slik:

Super: bit er jo veldig enkelt fordi det ligger ferdig for deg på nettet. Alt som er, ligger ute på nettet. Da trenger man ikke å finne opp kruttet ... så der hentet vi fra NRK på super: bit, disse der ferdige oppleggene, også fikk de en kort introduksjon

på hvordan ting fungerte, så fikk de lenke til instruksjonsvideoen. Da jobba de i små grupper i eget tempo med å følge instruksjonsvideoen.

Funn i studien viser til at tilgangen på de ferdige oppleggene på internett kan gjøre det enklere å differensiere for elevene i arbeidet med Super:bit. Tre av lærerne trekker fram at de ferdige oppleggene på internett har forskjellige nivåer, og ved å følge oppleggene kan det gjøre det enklere å differensiere undervisningen i arbeidet med Super:bit. På denne måten sier lærer 3 at elevene kan oppleve mestring på sitt ferdighetsnivå. Lærer 4 er den eneste av lærerne som trekker fram at han/hun og skolen har brukt arbeidet med Super:bit sammen med elever som sliter med å sitte i ro. Han/hun uttrykker: «...de orker rett og slett ikke 60 min med tradisjonell undervisning.». Alle 4 lærerne trekker også fram at programmering av Super:bit kan foregå enkelt, men kan i tillegg avansere seg ved å programmere Super:bit til å gjøre flere elementer samtidig. Dette var en av grunnene til at det var enkelt å legge til rette for at elevene kunne arbeide på forskjellige nivå, i tillegg til de ferdige undervisningsoppleggene på internett. Samtidig som det er enkelt å differensiere, er det enkelt å holde progresjon i undervisningsøkten når man har en oppskrift å følge, forteller lærerne. Oppleggene på internett er velutprøvde, og de har en god progresjon.

Tre av lærerne trekker fram at arbeid med programmering er en variert tilnærming i undervisningssammenheng. De forteller at denne formen for undervisning er med på å variere matematikkundervisningen. Lærer 4 sier: «... så er det med på å variere undervisningen min, som jeg synes er viktig på en måte». Lærer 2 forteller om hvordan programmeringsundervisningen har vært med på å variere undervisningen sin, og at i arbeidet med ulike matematiske temaer har programmering vært en positiv tilnærming for elevene. Han/hun sier: «...fordi da føler jeg vi favner veldig bredt, med tre tilnæringsmetoder for å lære samme type matematikk hver uke hele tiden.». For lærer 3 har undervisningen inneholdt ulike elementer i innholdet i undervisningen. Han/hun sier: «Det er variert, det var både video og tekst og konkrete ting de skulle gjøre med fysiske ting som bevegede på seg». De fysiske tingene som det blir snakket om er de ulike programmeringsverktøyene.

Utstyr og læringsaktiviteter

I intervju med lærerne kommer det fram at skolene som lærerne jobber på har tilgang på utstyr og undervisningsopplegg til å arbeide med Super:bit. Alle lærerne forteller at de har prøvd seg på forskjellig utstyr og ulike læringsaktiviteter knyttet til arbeidet med programmering i undervisningen sin. Når det gjelder arbeidet med Super:bit har alle lærerne brukt selve hjernen til Super:bit, og de har testet ut ulike opplegg knyttet til den. De klassiske eksemplene som alle lærerne forteller om er at elevene har programmert Super:bit til å vise navnet til elevene og at de har programmert Super:bit til å ha stein, saks og papir mot hverandre. Lærer 3 forteller i tillegg at elevene har programmert

skritteller med Super:bit, samt at elevene har programmert terninger med Super:bit som elevene skulle bruke når de spilte brettspill. Lærer 4 har latt elevene programmere Super:bit til en temperaturmåler. I tillegg til hjernen til Super:bit, trekker alle lærerne fram at de har prøvd ut Bit:bot, som er en del av pakken i Super:bit. Her forteller flere av lærerne om opplegg som er basert på at elevene skal programmere Bit:bot til å danse. Lærer 2 beskriver et arbeid der elevene programmerte Bit:bot:

«Med bilen så har de hatt muligheten til å lage dans, der alle programmerer og prøver å synce. Da er det seks stykker som prøver å få de til å gjøre alle bevegelsene samtidig og synkront ...».

Her beskriver han/hun at elevene skulle programmere Bit:bot til å utføre dansen synkront. Det samme forteller lærer 3 at de har arbeidet med i klassen, men han/hun tilføyer med: *«Men det handlet om at de skulle programmere robotene til å danse. De blinker med lys, så de er jo så kule i mørket».* Det blir trukket fram av lærer 2 og lærer 4 at de har brukt Bit:bot til tegning i sin undervisning. Dette beskriver lærer 4 med: *«Den bilen kan jo da tegne dersom man setter inn en sånn blyant da. Og litt i geometri da de skulle prøve å programmere den til å kjøre en sirkel eller kvadrat eller rektangel.».* Her har læreren brukt Bit:bot til å tegne, og i bruk i geometrien. Det siste som blir nevnt er at lærer 4 har brukt Bit:bot til å kjøre gjennom en by. Byen er en matte og elevene programmerer Bit:bot til å kjøre gjennom byen. Dette beskriver læreren som en aktivitet som krever mye forskjellige arbeidsmåter fra elevene og utforskning for å få til.

Videre i studien viser det til funn at elevene arbeider på en annen måte i arbeidet med programmering og Super:bit. Dette kommer fram blant flere av lærerne, og det blir uttrykt av lærer 4 at:

«Jeg ser for eksempel på Micro: biten at om en skal kjøre i 4000 millisekund så er det litt mer avanserte ting å begynne og jobbe med for dem. Også kjører den med motor på 100% av farten sin, og du vet ikke hvor fort den kjører.»

Her forklarer han/hun at elevene har andre ting å forholde seg til, og at arbeidet med programmering er utforskning for elevene. Lærer 4 trekker fram flere konkrete eksempler på hvilke oppgaver elevene har arbeidet med for å utforske, som også kan knyttes til programmering og koding. En av oppgavene lærer 4 trekker fram er: *«Her trenger en dør, men man har ikke plass til to, men skulle gjerne hatt to. Hvordan kan jeg løse det?».* Denne oppgaven handlet om at elevene skulle planlegge en romløsning som krevde to dører, men hvor det ikke var plass til to dører. En annen oppgave lærer 4 trekker fram er:

«Oppgaven var at de skulle lage ett heispanel. Så da må en trykke på en knapp, og kode det til å kjøre. Hva må en tenke på når en må lage det og hvordan kan det fungere for alle? Kan en snakke til heisen for å få den til å gå om en er blind? Hvor langt ned skal panelet være for at unger eller rullestolbrukere skulle kunne bruke det?»

Læreren forklarer at dette var en fin inngang til programmering for elevene. Han/hun har lagt opp til at koding og programmering skal være digitalt, men viser til at man kan gjøre programmering analogt med disse eksemplene, og at det kan gjøres analogt med ulike løsninger som man finner rundt oss.

I forbindelse med arbeidet med Super:bit er det tre av lærerne som forteller at de arbeidet analogt med programmering i klasserommet. Dette vil si at elevene arbeider uten pedagogiske roboter, men at elevene er «robotene» selv. Elevene slipper den tekniske biten med kodingen, men det blir lagt til rette for å få en forståelse av framgangsmåten. Den analoge programmeringen blir lagt fram på ulikt vis hvordan lærerne arbeidet med det. Lærer 3 valgte å la elevene programmere hverandre til å lage vafler. Læreren forteller at elevene måtte være nøyaktige, og at de måtte gi alle beskjeder. Dette forklarer han/hun med at elevene skulle lære seg at de pedagogiske robotene gjør bare det de får beskjed om fra programmereren, og derfor må elevene være nøyaktige når de skal programmere ulike pedagogiske roboter. Han/hun utfyller: *«... de skulle programmere hverandre til å lage vafler, så da måtte de jo si: "knus eggene i bollen". Da gjør de jo det da, med mindre de skal ta skallet på siden, sant?»*. Lærer 2 valgte å la elevene programmere hverandre to og to ved at en elev tegnet piler på tavla, og den andre skulle bevege seg etter pilene. I mellom pilene kunne det stå forklaringer på hva eleven som var «robot» skulle gjøre, f.eks. hoppe med et ben.

4.1.2 Læring av matematikk

Lærerne vi intervjuet i dette prosjektet, har gjennom intervjuene beskrevet hvordan man kan knytte programmering til matematikken. Vi har valgt å kalle denne kategorien som presenterer funnene vi har gjort for læring av matematikk. Funnene vi har gjort til denne kategorien handler om at lærere i stor grad synes det er vanskelig å knytte programmeringen opp mot matematikken. Likevel beskriver lærerne en undervisning som er preget av mye matematikk. Videre viser lærerne at programmering er et verktøy som det er lett å tilpasse og differensiere undervisningen med. Lærerne beskriver kompetansen sin som for liten slik at de da kommer til kort med hva hensikten med programmeringen i skolen er. Videre har dette påvirkning for valg av målsetning og vurdering i undervisningen deres. Alle disse refleksjonene er med å poengtere og utbrodere funnene vi har gjort til hvordan lærere ser på læring av matematikk gjennom programmering.

Programmering som et verktøy

I intervjuene med lærerne er det tydelig at de synes det er vanskelig å knytte programmeringen opp mot andre delemner i matematikk når de bruker programmering som et verktøy for å lære matematikk. I intervjuet spurte vi lærerne om hvilke tema de så på som sentrale å knytte til arbeidet med programmering. Dette for at en ikke bare skulle få fram hva de tenkte, men også for å åpne opp for hva de selv hadde arbeidet med. Lærerne drar i tillegg inn flere koblinger som vi kan knytte til matematikken som de viser til gjennom undervisning. Lærer 3 beskriver en undervisningsøkt hvor elevene programmerte Super:bit til en skritteller:

"Ehm, ja, også var det med disse batteriene, de var jo veldig kjekke. Siden da koblet de til den dere Super:bit, også lagde de skritt-teller, også har de målt da og gått rundt omkring. De har telt antall skritt bort til et visst punkt også har de telt antall skritt tilbake, også har de regnt ut differansen. Så vi har sneket inn litt matematikk. "

Her bruker læreren Super:bit til skritteller slik at elevene kan gå rundt med denne og måle hvor mange skritt det er til et gitt punkt og tilbake, eller hvor langt eleven kommer på x-antall skritt. Elevene kan sammenligne hvor langt de kom på x- antall skritt, og regne ut differansen på dette. Lærer 3 trekker videre fram arbeidet med læring og forståelse av variabler i arbeidet med programmering i undervisning. Han/hun så for seg å ta opp igjen denne erfaringen de gjorde seg i arbeidet med Super:bit når elevene skulle jobbe med algebra senere. Læreren mente elevene hadde fått sett funksjoner, variabler og algoritmer i en ny sammenheng.

Alle lærerne trekker fram Bit:bot som et nyttig verktøy, og nevner denne når de reflekterer rundt matematiske temaer når elevene arbeider med programmering. Tre av lærerne påpeker at i arbeidet med Bit:bot ser og bruker de litt av potensialet til å arbeide med vei, fart og tid. Lærer 1 trekker fram: «...men særlig det da man drev på med bilen ikke sant, med fart og tid og avstand... der ble det jo masse matte.». Lærer 3 og lærer 4 beskriver at elevene har programmert Bit:bot til å kjøre en viss lengde med ulik fart, og deretter målt hvor lang tid den brukte på strekningen. På den måten har elevene utforsket hvordan vei, fart og tid har en sammenheng, samt arbeidet med måling. Lærer 3 trekker i tillegg fram at elevene skulle programmere Bit:bot til å rotere, noe som gjør at elevene må kjenne til vinkler og grader. Videre viser lærer 4 til at Bit:bot blir brukt innenfor geometri, da elevene har programmert Bit:bot til å kjøre i ulike geometriske figurer.

Det kan tyde på at lærerne er innovent flere matematiske temaer når elevene arbeider med Super:bit og programmering, selv om de synes det kan være vanskelig å se direkte sammenhenger mellom

matematikk og oppgaver knyttet til Super:bit og programmering. Lærer 2 spesifiserer at programmering kan kobles opp mot all type matematikk, men viser til sin egen undervisning der det i større grad blir arbeidet med programmeringen i seg selv. Videre legger lærer 2 til at i arbeidet med programmering er det helt nødvendig å kunne litt matematikk, eller ha en forståelse for hva en skal lage. På denne måten kan man visualisere det en ønsker å uttrykke gjennom et dataprogram. Lærer 2 forteller det slik:

«Vi har alt ifra kalkulator til alt som regner ut for deg, og som gjør alt. Så snart trenger man ikke å kunne noe av det, men så ender en opp om en skal lage noen av disse programmene med å måtte kunne matematikken bak det likevel. Så en må ha med seg alle disse basistingene. Om en ikke kan gangetabellen så sliter en med å få programmering til å virke også.»

Alle lærerne legger fram matematiske tema som man kan knytte programmering opp imot. De viser til færre av disse temaene som de har brukt i egen undervisning. Det viser seg gjennom refleksjon at lærerne i intervjuet kommer på matematikk som de har brukt i programmeringsundervisning uten å være bevisst på det selv.

Differensiering og tilpasset opplæring

Lærerne ser store forskjeller i klassen når det gjelder kunnskaper om, og arbeid med programmering. Det er helt tydelig at differensiering og tilpasset opplæring må fokuseres på i arbeidet med programmering. Lærerne legger til at arbeidet med programmering er veldig enkelt å legge til rette for, og understreker at det finnes ulike felt og vanskelighetsnivå elevene kan jobbe på. Det framkommer av tre av lærerne at differensiering og tilpasset opplæring når elevene arbeider med programmering, kan være enkelt å tilpasse. De forteller at dette kan bidra til at elevene kan utnytte sitt fulle potensial og at de kan arbeide på det nivået de er. Hos lærer 1 og lærer 2 kommer det fram at noen elever jobber kun stegvis med oppskrifter til prosjektet, mens andre jobber mer helhetlig og har en forståelse av bruken til programmeringen. Selv når elevene ikke arbeider individuelt i et fellesskap, kan det legges opp til differensiering gjennom gruppearbeid. Her handler det om arbeidsfordelingen og valg av oppgaver som elevene gjør. Noen kan arbeide med regning, mens andre kan arbeide mer praktisk. Lærer 1 beskriver en undervisningssituasjon slik: *«da sitter en gjerne to og to sammen sant, en må springe og ... måle avstand og regne litt og»*. Her viser lærer 1 til at undervingen inneholder flere arbeidsmåter og at elevene må samarbeide med hverandre.

I arbeidet med programmering sier lærer 3 og lærer 4 at veiledning er sentralt for elevene i læringsprosessen. De trekker fram at veiledningen må foregå på en måte slik at elevene kan finne en

løsning uten at lærerne sier direkte framgangsmåten eller løsningen. I intervjuet sier lærer 4: "*Eller lage en løsning og da må en jo kunne komme til det selv, og ikke bare få det servert.*" Dette kan handle om å stille spørsmål ved framgangsmåten til elevene slik at elevene kan reflektere og finne fram til løsningen av problemet selv.

Ved spørsmålet om hvilke matematiske temaer som lærerne så på som sentrale når de brukte programmering i undervisning, kom visualisering fram. Programmeringen, og spesielt bruk av Bit:bot, ble dratt fram som et viktig hjelpemiddel for elevene for å kunne se sammenhenger av ulike matematiske temaer. Det lærer 4 trekker fram i intervjuet er: *«Få de til å få en forståelse for liksom at fart og tid har en sammenheng.»*. Han/hun prøver å la elevene få en forståelse for at fart og tid har en sammenheng gjennom bruk av Super:bit og Bit:bot. Videre i intervjuet beskriver lærer 4 hvordan elevene utforsker med Bit:bot for å få en forståelse for at fart og tid har en sammenheng. Lærer 4 beskriver at elevene kan justere motorkraften til Bit:bot, og på den måten kan elevene regne ut hvor lang tid Bit:bot bruker på en strekning med 50% motorkraft sammenlignet med 100% motorkraft. Lærer 4 ser på dette som et nyttig verktøy mot å få alle elevene gjennom en praktisk tilnærming, som igjen er med på å fremme tilpasset opplæring og differensiering.

Både lærer 2 og lærer 4 legger fram at programmering har blitt brukt med elever som trenger en annen form enn tradisjonell undervisning. Lærer 2 har brukt dette både i større og mindre deler av en klasse. Læreren beskriver at det har vist seg å være en spesielt engasjerende form for undervisning som fanger mange, spesielt dersom man kan knytte programmeringen opp mot interessene til elevene. Når lærer 4 forteller om hvordan han/hun har brukt programmering med elever som trenger en annen form for undervisning, sier læreren:

«Vi har brukt det litt på elever som har slitt litt med å sitte i ro. Trenger ikke være ADHD elev for det om, men at de trenger, de orker rett og slett ikke 60 min med tradisjonell undervisning.»

Programmeringen blir derfor presentert som en god måte å differensiere og tilpasse undervisningen for å legge til rette for læring av matematikk.

Hensikten, målet og vurdering knyttet til programmering

I intervjuene hadde vi flere spørsmål knyttet til interessen om å finne ut hvordan lærere bruker Super:bit og programmering i matematikken. Det kom i stor grad fram gjennom alle intervjuene tanker knyttet til hvor nyttig Super:bit var, hva programmering tilfører matematikken og programmeringen inn i skolen. Spesielt lærer 1, 3 og 4 så det som vanskelig å knytte programmering opp mot matematikk. Spørsmålet om hva de legger til rette for matematikk har vi gjennom

intervjuene tolket som tradisjonell matematikk gjennom regning og da knyttet opp mot matematiske tema som blant annet geometri og algebra. Lærer 1 og 4 sliter med å se hensikten med programmering. De beskriver undervisningen som lekpreget i den grad at elevene jobber med programmeringen. Vi ser likevel at undervisningsøktene er beskrevet som noe utforskende og med flere elementer som kan knytte opp mot læring av matematikk.

En annen refleksjon som blir dratt fram gjennom intervjuene er hvordan man setter mål for undervisningen, og hva elevene skal lære. Lærer 1 uttrykker det slik: «... *men hva er hensikten? Vi mangler liksom formålet.*». Dette er i sammenheng med hensikten og kompetansen læreren har på feltet munnet ut i at det blir vanskelig å sette tydelige mål for undervisningsøktene med programmering. Dette viser seg i refleksjonene som lærer 4 har gjort seg om vurdering og hvordan en skal måle læringen til elevene i arbeidet med programmering. Lærer 4 trekker fram et eksempel i forhold til hvordan elevene skal bli målt ut ifra ulike prosjekter de velger å jobbe med, og spesielt hvordan elever som jobber i grupper skal vurderes. Denne problemstillingen i forhold til vurdering av gruppearbeid spiller seg ut på hvorvidt elevene skal vurderes felles eller individuelt og hvordan en skal kontrollere dette. Samtidig prøver lærer 4 å lene seg på at man skal gjøre elevene nysgjerrige på programmering. Lærer 4 uttrykker det slik: «*men så prøver jeg å lene meg på den at man skal skape en interesse for faget eller den delen av faget ... en må skape en nysgjerrighet eller en interesse hos elevene ...*».

4.2 Erfaringer

I dette delkapittelet skal vi presentere funnene vi har gjort til den andre delen av problemstillingen vår. Dette handler om hvilke erfaringer lærere har til bruken av programmering i matematikkundervisning. Gjennom intervju og analysearbeidet i etterkant har vi kommet fram til tre kategorier som vi skal presentere funnene våre i. Første kategori vil komme i kapittel 4.2.1 som handler om hvilke forutsetninger elevene har og den digitale kompetansen til elevene. Videre vil vi i 4.2.2 se på læreren sine forutsetninger og digitale kompetanse knyttet til programmering. I det siste delkapittelet 4.2.3 skal vi gå inn på læringssituasjonen som handler om hvordan lærere bruker programmering i undervisning gjennom en praktisk tilnærming. Og hvordan elevene fungerer som en ressurs for læreren.

4.2.1 Elevenes forutsetning og digital kompetanse

Lærerne som vi intervjuet delte flere like tanker om hvordan elevene opptrådte i programmeringsundervisningen. Dette kom fram gjennom at lærerne fortalte hvordan elevene deltok i programmeringsundervisningen, samt hvordan de bidrog. Vår studie viser funn knyttet til at elevene er adaptive og raskt lærende når de arbeider med digitale verktøy i undervisning. I presentasjonen av denne kategorien skal vi komme mer inn på hvordan lærerne opplever elevene sine i undervisning.

Alle lærerne beskriver elevene som adaptive og som raskt lærende når de programmerer. De beskriver på forskjellig grunnlag hvorfor elevene trives med arbeid knyttet til digitale verktøy og programmering. Lærer 1 og lærer 2 beskriver elevene som storforbrukere av teknologi og at de bruker teknologi på ulike måter i hverdagen. Lærer 2 og lærer 4 forteller videre at elevene har helt andre kunnskaper og ferdigheter til det digitale, samt at elevene stort sett har en bedre digital kompetanse knyttet til dette feltet som gjør at de tar programmeringen fort. Dette blir beskrevet som en god ressurs inn mot undervisningen. Lærer 1 beskriver at elevene i stor grad er selvgående i undervisningen i den grad at de kommer fram til flere av løsningene selv. Lærer 2 sier følgende om elevenes bruk av iPad: *"Det er det som er fordelene med en iPad at appene er så intuitive at ungene nesten skjønner det ved å trykke, dra og slippe."* I tillegg viser to av lærerne til at enkelte elever har erfaring med blokkprogrammering og at noen til og med har vært med på kodeklubbene til «Lær Kidsa Koding». Det blir forklart at de som har drevet mye med data, ofte får til mer i programmeringsundervisningen og kan jobbe mer helhetlig med forståelse av hvordan programmering fungerer. De andre elevene jobber ofte stegvis med oppskrifter. Videre viser alle lærerne til at programmering er noe som fanger elevene i stor grad. Lærer 1 forteller at elevene koser seg med programmeringen.

Både lærer 1 og lærer 3 peker på at nøyaktighet er et viktig fokusområde, og de viser til at noen av elevene kan slite med dette. De sier man må være nøyaktig i arbeidet, og ikke slurve med kommunikasjonen med programmeringsverktøyet. Datamaskinen gjør akkurat det man ber den om å gjøre, så dette er sentralt. Lærer 1 trekker fram at programmering er et viktig verktøy for å få en variasjon i undervisning som kan være med å inkludere flere av elevene. Som lærer 3 også nevner er det mange elever som koser seg i programmeringsundervisning, og dette tror både lærer 1 og 3 at er på bakgrunn av en arbeidsform som treffer praktikere og andre som ikke nødvendigvis foretrekker en tradisjonell matematikkundervisning på tavle med regning. Disse elevene blir ofte beskrevet til å skinne i programmeringsundervisning. Dette kommer fram hos lærer 2 og lærer 4 der de viser til programmering som brukt i spesialundervisning og tilpasset opplæring for elever som ikke har mulighet til å jobbe med oppgaver i normale økter med programmering.

Vi kan se at elevene blir beskrevet som flinke innenfor programmering av lærerne. Elevenes kunnskap og adaptive ferdigheter knyttet til digitale verktøy gjør at eleven effektivt lærer prinsippene i programmering gjennom å bare arbeidet med dette. Elevene storkoser seg i arbeidet med programmering og måten en varierer undervisningen slik at andre elever kan skinne i matematikkundervisningen trekker lærerne fram som veldig positivt og inspirerende. Refleksjonene til lærerne baserer dette på flere faktorer som interessefelt innenfor roboter, data, dataspill eller det som læreren måtte klare å knytte programmeringen opp imot.

4.2.2 Læreres forutsetninger og digitale kompetanse

I denne delen legger vi fram funnene i studien vår knyttet til lærernes forutsetninger og digitale kompetanse. Lærerne har både gjennom spørsmål fra oss og på eget initiativ gjennom intervjuet pratet en del om hvordan deres forutsetninger og digitale kompetanse påvirker deres undervisningspraksis. Vi har derfor valgt å dele denne kategorien og presentasjonen av funnene våre i tre deler: lærerens egeninteresse og kunnskap, kurs og kompetanseheving på skolene, samt prøving og feiling.

Egeninteresse og kunnskap hos læreren

Det viser seg at det er store variasjoner mellom de ulike lærerne på hvor mye interesse de har for programmering, kunnskap de har om det og hvilke erfaringer de har gjort seg knyttet til bruken av programmering. Når lærer 1 beskriver sin kunnskap i programmering blir det beskrevet som at han/hun har lav kompetanse på feltet. Lærer 2 forteller derimot at han/hun har arbeidet med programmering i skolen i om lag 10år. Kunnskapen og ferdighetene er derav lik med interessen for programmering og gjenspeiler undervisningsmengden som læreren har hatt i skolen. Lærer 2 liker en praktisk tilnærming på lik linje med lærer 4. De synes ferdighetene med å prøve og feile er spesielt

morsom å jobbe med. Denne interessen for prøving og feiling, sammen med praktisk arbeid, gjenspeiler litt hvordan lærer 2 og 4 bruker programmering. Ikke bare i selve matematikkundervisningen, men også som et verktøy for læring inn mot andre klasser og grupper ved tilpasset opplæring. Lærer 3 forteller at han/hun har liten egeninteresse og lite kunnskap i programmering. Lærer 4 beskriver at det ikke er en stor egeninteresse for å drive på med dette på hobbybasis. De ser derimot nytten i at elevene skal lære seg dette, og at det er god nok grunn for dem til å legge det inn i undervisningen.

Likevel har lærerne interesse for å utvikle sine kunnskaper og ferdigheter knyttet til programmerings bruk i matematikkfaget. Dette kommer spesielt fram hos lærer 1, 3 og 4 da disse er i opplæringsfasen og bruker programmering veldig grunnleggende i timene sine. Dette gjenspeiler seg i at lærerne synes det er spesielt vanskelig å knytte programmering opp mot matematiske tema. Lærer 2 har kunnskaper og ferdigheter som legger til grunn for dette, men har likevel et fokus på å kunne tilrettelegge for at programmeringen blir brukt som et verktøy for å lære matematikk. Lærer 4 drar derimot inn en viktig rammefaktor som begrenser denne utviklingen, og det er tid. Læreren reflekterer rundt interessen sin til å være en faktor for hvorvidt en jobber med dette for å utvikle seg, men kompetanseheving knyttet til programmering blir nedprioritert ovenfor andre oppgaver som læreren har i hverdagen.

Kurs og kompetanseheving

På bakgrunn av at det er fire forskjellige skoler lærerne arbeider på, er det naturlig med en stor variasjon i kursing og praksis av kompetanseheving i skolene. Dette viser seg gjennom studiet vårt å stemme i en viss grad, men vi kan se nyanser som kan knyttes sammen fra de ulike praksisene. Lærer 1 og lærer 4 har deltatt på Super:bit-kurset til Vitensentrene. De beskriver at kurset har fungert som en inspirasjon til hva de selv kan gjøre i undervisningen. Lærer 1, 3 og 4 beskriver skolene deres med at de har lite kursing knyttet til programmering, spesielt ettersom at det er nytt at programmering står eksplisitt nevnt i LK20. I denne sammenhengen uttrykker lærer 3: «I og med at det er i læreplanen så burde vi hatt mer kurs.». Men lærer 4 spesifiserer: «... *det har vært god takhøyde om jeg spør om jeg kan få dra på kurs eller sånt.*». Her uttrykker lærer 4 at det har vært nødvendig å ta eget initiativ for å kunne dra på aktuelle kurs. Lærer 4 har i tillegg til Super:bit-kurset på eget initiativ opparbeidet seg kunnskaper om programmering med prøving og feiling. Læreren har nyttet seg av internett med videoer og forklaring til ulike konsept av programmering, der prøving og feiling har vært en viktig del av prosessen. Dette mener han/hun er et sentralt punkt for elevene i læringen av programmering.

Selv gjennom utforskning, samt prøving og feiling ser lærerne på det som gunstig at skolene kunne spisset seg inn mot et satsingsområde for at en skal få mest utbytte av det. Skolene til lærer 1, 3 og 4 har ingen praksis per intervjuets dato for spesialisering inn mot programmering eller kursing. Disse skolene har likevel bra med utstyr knyttet til programmering, og ledelsen oppfordrer lærere til å prøve det ut i undervisning. Lærer 3 har ikke god nok kunnskap i programmering til at læreren mener undervisningen blir tilstrekkelig, og etterlyser kursing på programmering og en bedre oppfølging. Dette er på bakgrunn av at programmering har kommet inn i læreplanen. På skolen hvor lærer 2 arbeider har de valgt å spesialisere seg inn mot enkelte programmeringsverktøy. Å ha en spissing av kursingen og at kursingen blir satt i et system og oppfulgt, ser både lærer 2 og 4 på som spesielt viktig for å heve kompetansen i kollegiet. Lærer 1 mener videre at kompetansen man skaffer seg gjennom kursing og oppfølging er viktig. Han/hun mener at denne kompetansen kan være viktig for å bruke programmering som et verktøy inn mot matematikkfaget. Det å kunne bruke kursingen til å ha et høyere fokus på verktøyene man kan bruke og hvordan de kan brukes slik at en kan bli tryggere i undervisningen, ser lærer 1 på som viktig. Læreren erfarer at det er vanskelig å opprettholde kunnskapen om programmeringen når det ikke brukes programmering i undervisningen over lengre tid. Læreren mister dermed tryggheten i selve undervisningen om programmering. Lærer 1 savner denne oppfølgingen av skolen eller av Vitensentrene slik at lærerne kan få opprettholde kunnskap og ferdigheter. Det å ha kunnskap om programmering i undervisningen mener lærer 2 kan speile hvor trygg man er i programmeringsundervisningen. Samtidig trekker han/hun fram at det ikke er nødvendig for å få til en god undervisning i programmering.

Kompetanseheving og kursing gjennom et interkommunalt samarbeid er en måte lærer 2 trekker fram. Denne formen for interkommunalt samarbeid gjennom kursing blir ikke vist til hos de andre lærerne. Lærer 3 og lærer 4 legger derimot fram et spesielt godt kollegiesamarbeid og en god delingskultur i skolen, der informasjon er lett tilgjengelig for dem som skulle ønske å utvikle kompetansen sin på feltet. På skolen til lærer 4 er det stor takhøyde for å spørre om å få dra på kurs, men at det er avhengig av at lærerne selv spør. For oss virker det som at skolene til lærer 1, 3 og 4 legger opp i stor grad til utvikling gjennom innkjøp av utstyr og åpenhet for å prøve, samt kurse ansatte. Kursene blir derimot ikke satt i noe system og lærerne som sitter med kunnskap er de som prøver seg fram og er ivrige på å finne ut av og utvikle egen kunnskap. Dette kan være negativt, spesielt for de eldre ifølge lærer 4: «*Men jeg skjønner at Kari og Ola på 60 år synes det er tyngre å kaste seg ut i det*». Læreren forteller at det kan være forståelig at kollegaer med lavere digital kompetanse legger bort programmering fordi de ikke føler at de har nok kunnskap eller trygghet for å få til en god undervisning. Det trengs en felles kompetanseheving for lærerne gjennom strukturert og oppfølgende kursing, samt satsing på feltet fra skolene. På denne måten kan man unngå at det

blir tilfeldig hvilke elever som får lærere som har programmeringskompetanse og som utnytter utstyret som skolene har tilgjengelig. Selv om kursene er gode og bra trengs det en form for oppfølging for å sikre kunnskapen og ferdighetene hos lærerne som skal bruke kompetansen i undervisning.

Prøve og feile

Vi kan i vår studie trekke fram likheter mellom lærerne vi har intervjuet. Alle lærerne ser på det som en effektiv og spennende måte å undervise på ved å prøve og feile, og rett og slett bare hoppe i det. Lærer 4 beskriver det slik: *«det blir litt sånn at du må bare hoppe i det»*, og lærer 1 forteller: *«Vi er ikke noe ekspert i det hele tatt, vi famler litt sånn i blinde, men vi tenker at, ehm ... alternativet er ingenting. Så vi må jo bare prøve.»*. Lærer 1 beskriver det som at læreren ser nytte i undervisningen gjennom å bare hoppe i det og bare prøve ut programmering. Videre forteller lærer 1 om sitt kollegium som er imponert over at han/hun har hoppet i det, selv med lite kunnskap og erfaringer på området. Læreren beskriver resten av kollegiet som skeptisk for å prøve det ut i egen undervisning, på lik linje som de andre lærerne beskriver. Vi får et inntrykk av lærerne som ikke kaster seg ut i undervisningen at de velger å ikke ha programmering på grunn av lav kompetanse i programmering og digitale verktøy, og dermed kan være mindre trygg i lærerrollen ved programmeringsundervisning. Da mener lærer 2 at bedre programmeringskunnskap hos lærerne kan være med å bidra inn mot å trygge dem i undervisningssituasjon. Videre kommer det fram gjennom lærer 4 at undervisning i programmering er en annerledes form for undervisning, som også kan oppleves mer bråkete og urolig. Det blir i intervjuene beskrevet at undervisningen knyttet til programmering skiller seg noe ut ifra tradisjonell tavleundervisning i matematikk, og denne formen for undervisning påpeker lærer 3 og 4 som spennende og noe en bør forvente.

Hvordan lærerne tilegner seg kunnskap om programmering blir i stor grad basert på Super:bit-kurset og egeninteresse for prøving og feiling, både alene og sammen med kollegiet. Lærer 2 forteller at han/hun erfarer at det fungerer bra at lærere prøver seg fram i et kollegium, og samtidig jobber sammen med elevene i programmeringsundervisningen. Lærerne som har prøvd ut denne måten å undervise ser på denne reisen sammen med elevene som lærerik og nyttig læringsprosess for både læreren og elevene.

4.2.3 Erfaring fra lærerens undervisning med Super:bit

I vår studie viser det til funn som handler om hvilke erfaringer lærerne gjør seg knyttet til undervisningen de har gjennomført med bruk av programmering. Gjennom intervju med lærerne kommer det fram at lærernes undervisningserfaring baserer seg i stor grad på å bruke en praktisk tilnærming i undervisning. Elevene blir videre beskrevet som viktige bidragsyttere i undervisningen da

flere av lærerne er ferske på programmering. I siste del tar vi for oss lærernes erfaringer knyttet til nytten av å drive med programmering i undervisningen og hvordan de ser denne kunnskapen nyttig for framtidens arbeidsplasser.

Lærernes erfaring knyttet til praktisk arbeid og motivasjon

Alle lærerne trekker fram erfaringer til arbeidet med blokkprogrammering og pedagogiske roboter hvor denne undervisningen kan bidra inn mot en praktisk tilnærming til matematikkfaget.

Erfaringene deres tilsier at denne praktiske tilnærmingen kan være motiverende for elevene. Lærer 4 viser til erfaringer rundt den praktiske tilnærmingen programmering gir, og spesifiserer at man må godta en annen form for læringssituasjon i klassen når elevene arbeider med programmering i undervisning. Det å kunne jobbe på en annen måte i klasserommet er det derimot flere av lærerne som trekker fram som positive erfaringer. Lærer 4 beskriver det slik: *«Når en kaster noe som dette inn i miksen så samles de rundet det mer.»*. Han/hun erfarer at når elevene arbeider med programmering blir det naturlig å samarbeide og prate sammen. Måten man kan koble teori opp mot noe praktisk med programmering trekker spesielt lærer 2 og lærer 4 fram som veldig nyttig for læringen til elevene. Lærer 2 sine erfaringer trekker fram at elevene synes det er nytt og spennende, og sier det slik: *"Jeg tenker de synes det er nytt og spennende, samtidig som det er avveksling, i tillegg til at det er den praktiske tilnærmingen til faget."*

For at man skal kunne benytte seg av denne formen for undervisning, trekker to av lærerne fram erfaringer knyttet til elevenes interesse som en viktig faktor. Det er viktig at man bygger på interessene til elevene slik at undervisningen og aktiviteten i seg selv har en mening for dem. Lærer 2 og lærer 4 beskriver at man kan koble interessen til elevene opp mot hvilke programmeringsoppgaver man som lærer legger til rette for. For noen elever kan dette for eksempel være bilinteresse, og da kan lærer benytte seg av denne interessen ved å ta i bruk f.eks. Bit:bot. Lærer 4 viser til en erfaring med en undervisning hvor elevene ikke hadde særlig engasjement i arbeidet med Bit:bot. For å engasjere elevene mer forklarer han/hun det slik: *«Prøv å lage en løype i klasserommet som du skal prøve å komme gjennom»*. Da fikk elevene selv bestemme hvor løypen til Bit:bot skulle være. Videre har lærer 4 erfart at dette kan fungere givende for elevene, og dermed at elevene har hatt et større engasjement i disse undervisningsøktene. Lærer 2 og lærer 4 trekker fram at elevene blir engasjerte når de kan programmere en robot via en datamaskin slik at de med en gang kan se resultatet av programmeringen. Erfaringene til lærerne viser til at elevene enkelt ser om de har gjort riktig ved å se om Bit:bot følger ruten den skal kjøre, eller ikke.

Elevers bidrag i programmeringsundervisningen

For oss kommer det tydelig fram at tre av lærerne som er med i prosjektet er i oppstartfasen av sin egen kompetanseutvikling og har en forholdsvis grunnleggende tilnærming og kunnskap til programmering i skolen. De tre lærerne forteller at de arbeidet sammen med elevene i undervisningen, og at de samtidig lærte noe av dem. Lærer 3 sier: «... *de kan jo også lære oss noen ting, «kom å se hva jeg fant ut, så skal jeg vise deg hvordan.»*». Her viser lærer 3 til at elevene prøver seg ut med Super:bit eller Bit:bot, og når de kommer på nye ideer eller løsninger, så er de villig til å dele det med læreren og medelevene. I lærings situasjonen jobber da eleven og læreren side om side for å utvikle kunnskapen sin. Videre forteller lærer 1 at det er en god læring for elevene å få lov til å lære bort det de kan, og at han/hun erfarer at elevene liker å bidra inn i undervisningen.

Elevene er her i stor grad nyttig for utvikling av kompetansen til læreren. Og som lærer 3 trekker fram kan kunnskapshullene til læreren bli dekket av elevenes bidrag med kunnskap. Lærer 1 beskriver elever som har vært med på kodeklubbene som veldig flinke til å programmere. Disse elevene, sammen med andre elever som er god i programmering ble brukt som hjelpelærere eller assistentlærere i programmeringsundervisningen. Dette er noe lærer 4 også trekker fram når han/hun sier: "*De lærer kanskje mer av hverandre enn av meg føler jeg*". Lærer 4 viser til gode erfaringer med at læreren jobber sammen med elevene for å komme fram til ulike løsninger. Lærerne trekker fram at de opplever at elevene får mye ut av å lære bort det de kan, da det kan være en del av læringsprosessen og sette ord på hva de kan.

Lærernes tanker og erfaringer om programmerings plass i skolen

Gjennom intervjuene med lærerne hadde vi som mål å se på hvilke tanker og erfaringer lærerne hadde knyttet til programmerings plass i skolen og i samfunnet generelt. Det blir trukket fram av alle lærerne at programmerings plass i skolen kan ses i lys av den teknologiske utviklingen og et framtidsrettet perspektiv. Lærerne har som nevnt tidligere beskrevet elevene sine som store forbrukere av teknologi. De hverdagslige verktøyene som de bruker som touch-telefon, nettbrett og datamaskin er alle komplekse, digitale produkt som baserer sine funksjoner på programmering. Lærer 2 mener det både er viktig og samtidig interessant for elevene å finne ut hvordan disse digitale midlene henger sammen. Han/hun beskriver videre at elevene på denne måten kan få en tilknytning til verdenen og teknologien den består av, og kanskje få en forståelse for hvordan dette fungerer.

Tre av lærerne trekker fram at programmering i skolen kan være nyttig mot å forberede elevene til framtidens yrker. De forteller videre at morgendagens samfunn kan være preget av nye jobber vi ikke vet om enda, som vil kreve en høyere digital- og programmeringskompetanse. Lærer 3 sier: "*... men at det er vesentlig at de har et forhold til hva programmering er fordi de skal utdannes til så mange*

nye yrker som man ikke har i dag.". Videre sier lærer 2: "...ikke minst tenke på at lærere og ungene skal ha bruk for det de har bruk for om 10-15 år, og det er ingen av oss som vet hvordan det ser ut da". De trekker fram at det mulig vil være behov for allmennkunnskapen til programmering og forståelse for hvordan teknologien som vi omgås og bruker fungerer. Dette er kunnskap for framtiden som videre blir påpekt av lærerne å være i rask utvikling, og det er vanskelig å se at den teknologiske utviklingen vil stoppe. Lærer 2 spesifiserer:

«... jeg tenker at det er mer og mer jobber som setter IT og programmeringskompetanse som høyt da. Om du så skal være elektriker i dag så er det trådløse bryter som skal snakke sammen med det og det og det så det er sånn at selv om man skulle være praktiker eller håndverker er det ting en får bruk for med programmeringen. I tillegg til at det er en måte å tenke på som en kan bruke til fryktelig mye.»

Denne måten å tenke på i disse yrkene som både eksisterer, men også yrker som vi enda ikke vet om, beskriver lærer 2 at kan være en av grunnpilarene for hvorfor programmering har kommet inn i skolen. Lærer 3 og lærer 4 ser på det som at elevene skal kunne få en mulighet til å utvikle sine digitale ferdigheter knyttet til programmering for å gjøre dem klare for morgendagens samfunn. Tre av lærerne spesifiserer at det er viktig å få elevene til å skjønne nytten av programmeringen og få de til å få et tankesett som kan bidra til en effektiv måte å løse digitale og analoge problem på.

4.3 Oppsummering av funn

Funn i studien viser til at lærerne legger opp til arbeid og læring i fellesskap når de arbeider med programmering i matematikkundervisningen. De legger til rette for en undervisning der elevene arbeider individuelt i fellesskap og i grupper. Disse sosiale interaksjonene er med på å skape diskusjoner og refleksjoner rundt eget og andres arbeid.

Geometri og algebra er emner i matematikken som lærerne trekker fram i arbeidet med programmering. Lærerne beskriver at det er vanskelig å koble programmering opp mot matematiske emner i undervisningen. Funn i studien viser at lærerne beskriver en programmeringsundervisning som inneholder mer matematikk enn de selv gir uttrykk for. Selv om lærerne ikke har et stort fokus på programmering for å lære matematikk, så bruker de programmering som et viktig verktøy opp mot differensiering og tilpasset opplæring. Super:bit blir presentert av lærerne som et enkelt verktøy å differensiere og tilpasse matematikkundervisningen med.

Lærerne forklarer at de har lite kompetanse knyttet til arbeidet med programmering inn mot matematikk. I tillegg er det mange av lærerne som har lite erfaring med programmering. De beskriver elevene som flinke og adaptive i arbeidet med programmering og på denne måten kan dermed elevene være viktige bidragsytere. De flinke elevene kan virke støttende i undervisningen for de andre elevene, og kan være en ressurs for læreren. På bakgrunn av lærernes kompetanse synes lærerne det er vanskelig å sette konkrete mål og vurdere programmeringsundervisningen med Super:bit opp mot matematikkfaget. Lærerne savner en systematisk kursing fra skolens side, da de mener at programmering og tankegangen en utvikler i arbeidet med programmering kan bli sentralt for framtidige yrker.

Det er en felles oppfatning at programmering er et visualiserende verktøy som kan brukes i matematikkundervisning gjennom en praktisk tilnærming. Lærerne forteller at denne tilnærmingen kan virke positivt inn mot matematikkfaget, da dette kan bidra til mer variasjon i undervisningen. Gjennom å spille på interessene til elevene i det praktiske og visualiserende arbeidet med programmering, opplever lærerne at dette virker motiverende og engasjerende for elevene. Programmering fanger mange av elevene i stor grad, og funn i studien viser oss at det er andre elever som «skinner» i programmeringsundervisning enn i tradisjonell matematikkundervisning.

5.0 Diskusjon

På bakgrunn av teorigrunnlaget vårt og tidligere forskning har vi sett at programmering på ulike måter kan bidra til elevenes matematiske læring i skolen. I dette kapitlet skal vi drøfte funnene som er presentert i kapittel 4. Funnene skal drøftes opp imot vårt teorigrunnlag og tidligere forskning på feltet. Gjennom drøftingen ønsker vi å besvare studiens problemstilling. Ved å løfte blikket og drøfte hva funnene i studien betyr i en større sammenheng gir det oss mulighet til å se hva de kan tilføre av ny kunnskap. På bakgrunn av problemstillingen vår har vi valgt å dele diskusjonen inn i to deler, henholdsvis 5.1 og 5.2 som også samsvarer med måten analysen er presentert. Til slutt i kapitlet, kapittel 5.3, gir vi en kort oppsummering av diskusjonsdelen.

5.1 Tilrettelegging for læring av matematikk

Den første delen av drøftingskapitlet omhandler hvordan lærerne legger til rette for læring av matematikk i programmeringsundervisningen i lys av teorigrunnlaget vårt. Kapitlet er delt inn i to deler, der den første delen omhandler undervisningen til lærerne. Den andre delen omhandler hvordan lærerne legger til rette for læring av matematikk.

5.1.1 Undervisning

Lærerne i studiet vårt viser til en undervisning med Super:bit hvor de legger opp til at elevene arbeider individuelt i fellesskap og i gruppearbeid. I den sosiokulturelle læringsteorien trekker Vygotsky fram dialogen som en viktig faktor for læring. Gjennom individuelt arbeid i fellesskap og gruppearbeid kan de legge til rette for dialog og samarbeid mellom elevene. Dersom gruppearbeid blir brukt i undervisningen er en avhengig av at læreren legger til rette for oppgaver som er meningsfulle og skaper et godt samarbeid mellom elevene (Lyngsnes & Rismark, 2014, s. 114). For at dette skal bli en effektiv form for læring er det viktig at læreren har kompetanse slik at læreren kan være en viktig veileder for elevene i læringssituasjonen (Hattie, 2009, s. 34). Gruppearbeidet kan da bidra positivt til elevens læring gjennom samarbeid (Brennan & Resnick, 2012, s. 10–11). Likevel kan gruppearbeid være avhengig av lærerens kompetanse til å tilrettelegge for gode samarbeid og meningsfulle oppgaver. Dette peker på hva Lyngsnes og Rismark (2007, s. 62) skriver om at eleven er avhengig av noen som kan peke på kritiske faktorer og lage strukturer som hjelper eleven videre i læringsprosessen. I lys av dette, kan det se ut til at gruppearbeid kan være en krevende form for undervisning da det er flere faktorer enn et godt samarbeid som spiller inn på elevens læringsutbytte i undervisningen. På en annen side har vi individuelt arbeid i fellesskap. Her kan elevene arbeide på sitt nivå, og fokuset vil være på dialogen når elevene arbeider i fellesskap. Gjennom dialog kan elevene inspirere hverandre og bli inspirert til ideer knyttet til sitt eget arbeid med Super:bit. Dette støttes av Säljö (2020, s. 79) som skriver at elever i samarbeid med en mer kapabel partner kan bygge

bro mellom egen kunnskap og kunnskapen til partneren, slik at eleven kan skjønne hvordan en skal løse oppgaven. I tillegg kan de gjennom samarbeid og dialog diskutere og reflektere rundt problemstillinger knyttet til arbeidet med programmering.

Flere av lærerne velger å bruke elever som er flinke med programmering som lærerassistenter. Dette peker på elementer ved Vygotskys syn på læring og sosiokulturell læringsteori. Elevene med større programmeringskunnskaper i klassen kan fungere som et stillas for de andre elevene som synes arbeidet med Super:bit kan være utfordrende. Elevene får gjennom arbeidet med Super:bit delt erfaringer med hverandre som de kan lære og/eller dra nytte av. I lys av Lyngsnes og Rismark (2014, s. 111) kan elevene få trening i å forklare hva de sliter med, og den hjelpende eleven kan vise til kunnskap som må omformuleres til forståelig kunnskap. Det at elever får vise fram og forklare sine egne prosjekter og ideer er i tråd med Brennan og Resnick (2012, s. 10) og hvordan elevene kan utvikle sin algoritmiske tenkning gjennom algoritmiske perspektiv. Vi kan si at elevene støtter hverandre opp som et stillas i lærings- og utviklingsprosessen. Selv om dette er god læring for elevene er elevene avhengig av å få arbeide i sin nærmeste utviklingssone, og de trenger dermed veiledning og støtte fra læreren i lærings situasjonen (Imsen, 2020, s. 196; Skaalvik & Skaalvik, 2021, s. 70). Ettersom at elevene er støttet til hverandre og læreren har lav kunnskap på området, kan man stille spørsmål om hvor god veiledningen er. Dersom læreren ikke er en god veileder i undervisningen, kan det se ut til at elevene ikke får være i sin proksimale utviklingssone, og kan dermed gå glipp av viktig læring.

5.1.2 Læring av matematikk

I dialogen med lærerne blir ikke algoritmisk tenkning dratt fram som en sentral del ved deres egen undervisning. Lærerne legger likevel til flere sentrale matematiske emner som programmering kan passe opp imot gjennom geometri, sannsynlighet og algebra. Sevik et al. (2016) viser til de samme matematiske emnene som sentrale knyttet til programmering i matematikkfaget. Dette kan for eksempel knyttes opp til arbeidet med Super:bit og Bit:bot der en kan arbeide med vei, fart og tid. Lærerne sliter å koble programmering opp mot konkrete emner i undervisningen sin og beskriver det som vanskelig å praktisere koblingen mellom programmering og matematiske emner. På den ene siden uttrykker flere av lærerne at de har lite kunnskap om programmering og har lite erfaring med dette fra undervisningen. Dette kan være en av årsakene til at de synes det er vanskelig å se de matematiske emnene som programmering kan kobles mot. På en annen side viser det seg at lærerne arbeider med matematikk når de programmerer med elevene i klassen. Lærerne selv er ikke bevisst på bruken av matematikk i programmeringsundervisning, men gjennom intervjuet kommer det likevel fram at programmering blir brukt som et verktøy for å lære matematikk. En mulig forklaring kan være at lærerne har nok med programmeringen i seg selv og sliter med å konkretisere koblingen

mellom programmering og matematikk. Dersom dette er tilfellet, vil det kunne være en mulighet for at elevene går glipp av viktig læring. Selv om det er matematikk i programmeringsundervisningen lærerne beskriver, vil det likevel kreve en konkretisering av matematikken for å skape en effektiv læring hos eleven. Læreren vil være sentral i denne rollen som veileder (Lye & Koh, 2014, s. 59).

Flere av lærerne forteller at de bruker oppskrifter hentet fra internett som en ressurs når de har undervisning med Super:bit. Dette har de brukt som et hjelpemiddel mot differensiering og tilpasset opplæring i matematikkundervisningen, noe som også kreves for at elevene skal arbeide i sin proksimale utviklingssone. Dette støttes av Imsen (2014, s. 194) som skriver at det stilles store krav til differensiering av undervisningen for at elevene skal arbeide i sin proksimale utviklingssone. Forskning sier at programmering kan være et viktig hjelpemiddel for lærerne i arbeidet med differensiering av undervisning (Torkildsen & Stedøy, 2018, s. 21). Lærerne kan i arbeidet med programmering legge til rette for en undervisning hvor elevene kan jobbe på sitt eget nivå, og på denne måten kan elevene i større grad arbeide i sin proksimale utviklingssone. Lyngsnes og Rismark (2014, s. 98) beskriver at elever kan få jobbe på sitt eget nivå gjennom differensiering i en utforskende undervisning. Dersom dette ikke gjelder undervisningen til lærerne, vil oppskriftene kunne være begrensende i synet av en utforskende undervisning. På en annen side beskriver lærerne programmeringsundervisningen en ny måte å arbeide på der elevene i større grad er frie til å utforske og løse problemer. Denne tilnærmingen beskriver lærerne som lekpreget. Likevel har denne undervisningsformen trekk med flere elementer som er nært kjent med algoritmisk tenkning (Cervera et al., 2020, s. 2). I arbeidet med en utforskende undervisning er det vist til at en utvikler algoritmisk tenkning gjennom å programmere (Piedade et al., 2020, s. 3). På en annen side er programmering med oppskrifter også regnet som programmering, men en vil ikke kunne utvikle algoritmisk tenkning i like stor grad om en ikke fører en konkretisering av det problemløsende tankesettet (Angeli et al., 2019, s. 86).

5.2 Erfaringer

Den andre delen av drøftingskapittelet tar for seg hvilke erfaringer lærerne har til sin egen og elevens digitale kompetanse. Her går vi nærmere inn på programmering som lærings situasjon. Vi ser på hvordan programmering kan bli brukt som et verktøy for undervisning i matematikk. Til slutt ser vi på hvordan dette kan være med på å skape et engasjement hos elevene.

5.2.1 Lærers og elevens digitale kompetanse

Lærerne har ulik erfaring og kunnskap når det kommer til programmering. 3 av 4 lærere uttrykker at de har kastet seg ut i programmeringsundervisning uten særlig kompetanse og erfaring med programmering fra før. Dette til tross for at forskning sier at kunnskapen til lærerne kan påvirke hvordan integreringen av programmeringen skjer i skolen. Dette påpeker Kaufmann (i Johansen, 2020) ved at lærernes kompetanse er styrende for integreringen av programmering i matematikkfaget. Kunnskapen lærerne sitter med er en faktor for at lærerne skal oppleve en trygghet til å ta dette i bruk i undervisningen (Fjørtoft et al., 2019, s. 60). Lærers digitale kompetanse er en spesielt viktig faktor og en nødvendig forutsetning for god undervisning (Hattie, 2009, s. 36).

Lærers kompetanse og ferdigheter

Den varierende kompetansen til lærerne er i stor grad avhengig av egeninteressen lærerne har for programmering. Resursene som Super:bit har gitt skolene gjennom kurs og pedagogiske roboter til bruk i undervisning, har ifølge lærerne vært viktige hjelpemidler for å gjennomføre programmering i undervisningen. Likevel kan det være et problem at undervisningen per dags dato er preget av oppskrifter, og at undervisningen begrenses noe i forhold til muligheten programmering i seg selv gir. Oppskriftene gjør det lett for lærerne å sette i gang med programmering. Det kan derimot diskuteres om dette kan virke begrensende for potensialet og muligheten programmering gir i forhold til utforskning og problemløsning som Nouri et al. (2020, s. 1) beskriver. Det er realistisk å tenke at undervisningen ikke blir tilstrekkelig og at en ikke får utnyttet potensialet til elevene på den måten at elevene taper viktig læring (Hattie, 2009, s. 34). Måten læreren velger å legge opp undervisningen på er en avgjørende faktor for elevenes læring og prestasjon (Hattie, 2009, s. 36). Det kan da antas at lærernes kompetanse innenfor programmering kan påvirke hvordan de legger opp sin undervisning med Super:bit i klasserommet. Den digitale kompetansen som læreren innehar i bruk med programmering i matematikkfaget er derfor en viktig faktor i utbytte som elevene har gjennom undervisningen (Hattie, 2009, s. 36). Dette har med å utnytte potensialet til programmering som et verktøy i matematikkundervisning.

Elevenes forutsetning og digitale kompetanse

Det blir gjennom lærerne uttrykt at elevene er store forbrukere på teknologi og digitale verktøy i sin hverdag, og dermed beskriver de elevene som adaptive og raskt lærende når de arbeider med programmering i undervisning. Elevenes forutsetninger og digitale kompetanse ligger i stor grad i hvor adaptive elevene er i arbeidet med digitale verktøy, men deres tilknytning til spill og noens erfaring med blokkprogrammering er tenkelig som faktorer for dette. På bakgrunn av lærernes kompetanse bruker de derfor elevene i undervisning og oppskrifter som ressurs for å dekke opp undervisningen deres. Dette kan virke bra for å utvikle elevens algoritmiske tenkning gjennom de algoritmiske perspektivene (Brennan & Resnick, 2012, s. 10). Her har elevene et ønske om å uttrykke seg eller hjelpe andre, enten medelever eller lærer. På en annen side kan det være en utfordring at eleven i seg selv er den styrende faktoren i undervisningen. Læreren vil da i mindre grad kunne legge opp til en undervisning som er med på å differensiere og tilrettelegge for hver enkelt elev utenom de oppskriftene som blir brukt (Torkildsen & Stedøy, 2018, s. 21). I tillegg vil lærerens rolle innenfor stillasbygging svekkes da læreren ikke kan gi tilbakemeldinger og instruksjoner for å hjelpe elevene til å utvikle kognitive strategier (Säljö, 2020, s. 79).

Mål og vurdering

En annen side ved erfaringene til lærerne er hvordan en videre kan legge opp til klare mål og vurdering i et emne en har lite kompetanse i. Mål i undervisning er å sette sentrale mål slik at en kan styre undervisningen etter og mot målene slik at de kan nås (Bjørndal & Lieberg, 1978, s. 39). Når en jobber praktisk og utforskende med programmering bør det på lik linje med formativ vurdering være prosessen som er målet og ikke sluttresultatet (Bjørndal & Lieberg, 1978, s. 38; Wølner, 2013, s. 47). Eleven og læreren skal ha en forståelse for hvordan en kan lære av selve arbeidsprosessen. Videre må det konkretiseres med hva som er matematikken i det en jobber med. Slik kan programmeringen i seg selv brukes som et artefakt gjennom å være et verktøy for å lære matematikk. På en annen side framkommer det fra lærerne at de synes det har vært vanskelig å måle læringen når elevene arbeider med programmering i matematikkundervisningen. De synes det er utfordrende å sette et definert mål for økten. For at en skal få til dette krever det at lærerne har kunnskap om det de skal undervise i og en forståelse for hensikten til programmeringen i matematikkfaget. Den digitale kompetansen og innstillingen til læreren er her sentral og en viktig faktor for at det skal bli en god og tilfredsstillende undervisning (Cuban, 2001, s. 135; Fjørtoft et al., 2019, s. 60; Hattie, 2009, s. 34; Lyngsnes & Rismark, 2014, s. 102). Med lav kompetanse kan det være vanskeligere for læreren å ta gode didaktiske valg til mål og vurdering av undervisningen som en selv utfører. Det vil kunne være realistisk å miste potensialet til programmeringen om lærerne ikke utnytter verktøyet. Som Brennan og Resnick (2012, s. 13) legger til er det stor forskjell på erfarne programmere og nybegynnere i

forhold til hvorvidt de klarer å bruke potensialet programmering har som et verktøy for å lære matematikk. Videre i intervjuene blir det trukket fram en usikkerhet rundt hva elevene egentlig får ut av samarbeidet med sine medelever. Hvordan kan man vite hva elevene faktisk har lært? Lærer 4 trekker fram utfordringen med et samarbeid som kan være det å vite om elevene faktisk lærer noe i samarbeidet, eller om en av elevene bare henger med på lasset uten å delta. Lyngsnes og Rismark (2014, s. 115, 119) viser til at det er nødvendig med en tett oppfølging for at en skal vurdere gruppearbeid og problematikk i forhold til arbeidsfordelingen som lærer 4 trekker fram i intervjuet. Mye kan tyde på at læreren har en viktig rolle i gruppesammensetningen av elevene. For at elevene skal lære må det være et positivt samarbeid, hvor alle parter får delta og være en del av læringsprosessen (Lyngsnes & Rismark, 2014, s. 114).

Kompetanseheving

Lærer 1: «... så er de jo kjempeflinke». Lærerne beskriver elevene som flinke, men hva er egentlig flinke? Det er realistisk å tenke seg at lærere med lav kompetanse innenfor programmering lett kan bli imponert over elevenes kunnskap, selv om kunnskapen elevene besitter kanskje er normalen. Likevel registrerer vi at lærerne har gode erfaringer med måten de kaller å «kaste seg ut i det», som beskriver det å undervise i programmering med lav kompetanse og lite erfaring. Det er helt tydelig at en trenger lærere som prøver ut programmering for at elevene skal få den undervisningen i matematikk som er i tråd med LK20. På en annen side vil en kompetanseheving være nødvendig for å sikre at matematikklærere besitter programmeringskompetanse. Om det ikke skjer en kompetanseheving blant lærerne kan gapet mellom lærere og elever bli større ettersom at det viser seg at elevene er adaptive og raskt lærende. Vi ser på dette som et problem i forhold til hvordan lærere skal føre en tilstrekkelig undervisning som benytter seg av elevenes potensiale gjennom bruk av pedagogiske roboter i programmeringsundervisningen. Dette fordi en vil mangle kompetansen og bredden i programmeringen sin slik at en ikke vil kunne utnytte potensialet til programmeringen i seg selv (Brennan & Resnick, 2012, s. 13). Selv om flere av lærerne har deltatt på Super:bit-kurs og kommet godt i gang med arbeidet, vil en kompetanseheving være med å skape en bredere forståelse for bruken og nytten av programmering i matematikkundervisning. På denne måten kan lærerne og elevene i større grad dra nytte av verktøyet, slik Angeli et al. (2019, s. 90) beskriver at en kan lære fag gjennom bruk av programmering som et verktøy. På denne måten kan en i større grad legge til rette for at elever får kompetente lærere i programmeringsundervisning. Det understrekes sterkt i intervjuene at behovet for flere kurs er nødvendig, men også et behov for oppfølging etter kursene slik at kunnskapen hele veien klarer å holde seg oppdatert. Kursene må da legges i et system slik at lærerens kompetanse kan skape en god og effektiv undervisning (Hattie, 2009, s. 34).

5.2.2 Læringsituasjon

Nå skal vi se på hvordan en praktisk tilnærming gjennom en digital undervisning kan styrke matematikkfaget. Videre diskuterer vi rundt hvorfor programmering kan virke engasjerende for elever i matematikkundervisning.

En annen form for undervisning

Lærerne viser til bruken av programmering som et viktig hjelpemiddel for å variere undervisningen deres med en mer praktisk tilnærming. Dette knyttes til bruken av Super:bit eller andre pedagogiske roboter i undervisningen. Å bruke digitale verktøy i undervisning viser seg å være en fin måte å variere undervisningen på (Hattie, 2009, s. 222). Variasjonen lærerne beskriver er knyttet til programmering som en ny måte å føre matematikkundervisning gjennom praktisk arbeid. Dette praktiske arbeidet er vist til å motivere elever i form av en måte som viker fra den tradisjonelle tavleundervisningen i matematikk (Hattie, 2009, s. 222). Derimot kan dette være problematisk om dette er den eneste formen for praktisk arbeid elevene får i matematikk. I Monitor 2019 kommer det fram at 85% av lærere mener de bruker digitale verktøy for å motivere elever gjennom en variert undervisning som er mer utforskende (Fjørtoft et al., 2019, s. 70). Fokuset i undervisningen må derfor være på selve ferdigheten en tilegner seg gjennom en problemløsende undervisning og ikke sluttproduktet som en produserer (Lyngsnes & Rismark, 2014, s. 109). Dette må være med ellers i matematikkfaget og ikke eksplisitt til programmeringsundervisning. Likevel vil de pedagogiske robotene fungere som en konkretisering gjennom det praktiske arbeidet når en jobber med programmering i matematikkfaget.

Elevers interesse

Med Super:bit-prosjektet er det et mål at elevene skal kunne få skape en interesse til programmering gjennom at prosjektet skal være med å inspirere og bidra til elevens utvikling av digital kompetanse (Super:bit, u.å.a). Prosjektet skal være med på å skape variasjon i undervisningen gjennom at eleven skal kunne få prøve ulike veier å lære matematikk gjennom programmering. Det blir lagt fram at problemløsning gjennom prøving og feiling, og justering av prosjekt er viktige mål som elevene skal lære. Dette knyttes tett opp til arbeidet med utvikling av algoritmisk tenkning i forhold til Brennan og Resnick (2012, s. 6–9) sine algoritmiske øvelser. Lærerne som har deltatt i dette studiet er enstemt om at programmering i matematikkundervisning er med på å skape et spesielt engasjement i undervisningen. Arbeidet med programmering i matematikkfaget fenger i stor grad flere av elevene, også de som kanskje ikke normalt er engasjert i en tradisjonell matematikkundervisning (Grover & Pea, 2013, s. 40). Dette er noe som er forenelig med hva Engelsen (2012, s. 131) trekker fram som en faktor å spille på når det gjelder innhold i undervisning. Dersom det ikke spiller på elevens interesse, kan elevene miste motivasjon til å løse oppgaven og vil senke lærelysten til eleven (Polya & Conway,

1985, s. 5). Hvilke elever er det lærerne egentlig spiller på interessen til? Det er realistisk å se for seg elevene med interesse innenfor data og dataspill som engasjerte i programmeringsundervisning. Disse kan ha et ønske når det gjelder å uttrykke seg selv gjennom å vise eller lære andre noe av det de kan. Dette er i tråd med de algoritmiske perspektivene til Brennan og Resnick (2012, s. 10–11) ved å uttrykke seg selv i lærings situasjon. I tillegg kan den praktiske formen for undervisning være til fordel for elevene som er mer glad i praktisk arbeid. Dette er i kontrast til Monitor 2019 som legger fram at flertallet av elevene i skolen mener bruk av digitale verktøy i undervisning er nyttig og gir dem lærelyst (Fjørtoft et al., 2019, s. 149). På en annen side kan interessen til elevene for programmering forklares med noen av lærernes beskrivelser av programmering som en lekpreget tilnærming. En undervisning vil da være avhengig av en lærer med kompetanse som er en effektiv veileder i programmeringsundervisningen slik at en kan gi oppgaver som virker meningsfulle og interessante for elevene (Lyngsnes & Rismark, 2014, s. 99). Lærer 4 forteller om en elev i klassen som lar seg utfordre i fag som matematikk, som grubleoppgaver eller andre krevende oppgaver. Læreren er usikker på om eleven hadde hatt samme interesse for å bli utfordret i koding i matematikk. Dette kan ha noe med hvorvidt eleven foretrekker praktisk arbeid og en slags problemløsende framgangsmåte.

Vi ser det derfor som realistisk at elever med interesse for data og spill har en økt egeninteresse og forståelse for programmeringen fordi dette er noe de ser nytten av, og er noe som fanger deres interesse. Selv om programmering blir sett på som en motiverende og en fin læringsaktivitet i matematikkundervisningen, er det viktig å knytte oppgavene opp mot interessene til elevene (Polya & Conway, 1985, s. 5). På denne måten kan en utnytte potensialet til programmering som et verktøy for å lære matematikk. Likevel er det komplisert å vurdere hvorvidt dette gjelder fra klasse til klasse gjennom erfaringene til lærerne. Hvorfor programmering spiller på interessen og skaper engasjement kan være ulik, men det tyder på at å spille på interessen kan være en viktig innfallsvinkel for å skape engasjement. På denne måten kan lærerne legge til rette for at elevene skal ha bedre forutsetninger for læring. Ved å skape en forståelse ovenfor elevene i forhold til nytten av programmering kan det være med på å utvikle de algoritmiske perspektivene. I tillegg kan det hjelpe elevene til å bli mer utholdende i arbeidet med programmering i matematikkundervisningen (Brennan & Resnick, 2012, s. 6–11; Utdanningsdirektoratet, 2019a).

5.3 Oppsummering av diskusjon

Undervisningen lærerne beskriver er preget av arbeid i grupper og individuelt arbeid i fellesskap. Gruppearbeid er en fin måte å samarbeide på i programmeringsundervisning (Brennan & Resnick, 2012, s. 10–11). Det er likevel avhengig av lærerens kompetanse for at læreren skal kunne legge opp til gode samarbeid og meningsfulle oppgaver (Hattie, 2009, s. 34). Dette kan gjøre gruppearbeid krevende. Individuelt arbeid i fellesskap kan derfor være en god undervisningsform da elevene får jobbe på sitt nivå, og de sammen med andre kan reflektere og diskutere arbeid knyttet til Super:bit. Gjennom dialog kan de inspirere og bli inspirert av andre elever. I denne dialogen kan elevene uttrykke seg selv og utvikle en algoritmisk tenkning ved å vise fram og forklare egne prosjekt (Brennan & Resnick, 2012, s. 10). Man kan gjennom samarbeid skape seg refleksjoner sammen med andre, vurdere og diskutere problem og utvikle sin kritiske tenkning (Brennan & Resnick, 2012, s. 10–11; Sevik, 2016, s. 15; Skaalvik & Skaalvik, 2021, s. 68). I lys av dette kan det tyde på at matematikklærere har en viktig rolle når det gjelder å legge til rette for læring av matematikk i arbeidet med Super:bit.

Lærerne trekker fram matematiske emner de ser på som sentrale der programmering kan brukes som et verktøy for læring av matematikk. Likevel trekker lærerne fram mangel på kompetanse som en viktig faktor for at de har lite erfaring med bruken av matematiske emner inn i programmeringsundervisning. De er ikke selv bevisst på koblingen mellom programmering og matematikk, men de beskriver likevel en programmeringsundervisning med mye matematikk. Lærerne bruker oppskrifter hentet fra internett i undervisningen, og legger til at dette gjør det lettere å differensiere og tilpasse undervisningen. På en annen side kan det se ut til at dette har sammenheng med deres tolkning av at programmering er mer som lek. Dette kan være en utfordring for læringen til elevene da de ikke får drevet med en konkretisering av algoritmisk tenkning og arbeidet med programmeringsferdigheter (Angeli et al., 2019, s. 86).

Flere av lærerne i studien har lite kompetanse knyttet til programmering. I undervisningen erfarer lærerne at det er enkelt å bruke oppskrifter hentet fra internett. På en annen side kan dette være begrensende for programmeringen i undervisning i forhold til utforskning og utvikling av algoritmisk tenkning. På bakgrunn av lærerens kompetanse blir elever som er flinke i programmering brukt som ressurs i undervisning. Dette er positivt for elevens utvikling av algoritmisk tenkning (Brennan & Resnick, 2012, s. 10), men kan være problematisk om elevene får styre undervisningen uten innvirkning fra lærer. Om læreren ikke er en god veileder, kan eleven gå glipp av viktig læring (Grover & Pea, 2013, s. 39–40; Piedade et al., 2020, s. 2). Samtidig kan lærerens kompetanse sette begrensninger for valg av konkrete mål og vurdering av undervisning. For at det ikke skal være tilfeldig hvilke matematikklærere som besitter programmeringskompetanse vil det være nødvendig med en kompetanseheving. Her krever det en systematisk oppfølging slik at elevene får tilstrekkelig

undervisning knyttet til matematikkfaget i tråd med LK20, samt at de får best mulig utbytte av undervisningen.

Programmering blir brukt som et visualiserende verktøy gjennom en praktisk tilnærming. Dette er med på å engasjere elevene i undervisningen. Programmering blir dratt fram som en annen form for undervisning som er en fin måte å variere på (Hattie, 2009, s. 222). Elevene blir beskrevet som engasjerte i programmeringsundervisning. Det er vanskelig å vurdere hvorfor elevene har et slikt engasjement som blir beskrevet i programmeringsundervisning, men det kan skyldes blant annet en ny tilnærming til matematikkfaget, praktisk arbeid og elevens interesse for data og dataspill. Slike funn i vår studie samsvarer med Monitor 2019 der flere lærere bruker digitale verktøy for å motivere elever gjennom en mer variert og utforskende undervisning (Fjørtoft et al., 2019, s. 70).

6.0 Avslutning

I denne avsluttende delen av oppgaven ønsker vi å besvare studiens problemstilling. Derfor har vi delt kapittelet inn i tre deler. Først tar vi for oss en konklusjon av studien. Deretter ser vi på implikasjoner. Til slutt løfter vi fram veien videre og gir der en pekepinn på hvilke felt det kan tyde på at temaet trenger mer forskning på.

6.1 Konklusjon

Vi benytter nå muligheten til å hente fram igjen problemstillingen vår, for så å komme med en konklusjon av studiet vårt.

Hvordan legger fire lærere til rette for læring av matematikk på mellomtrinnet gjennom bruk av pedagogiske roboter fra Super:bit-satsingen, og hvilke erfaringer gjør de seg knyttet til dette?

Fra studiet vårt kommer det fram at lærere planlegger og gjennomfører undervisning med Super:bit i et sosiokulturelt læringssyn. Undervisningen lærere gjennomfører med programmering i matematikk blir i stor grad lagt opp til at elevene arbeider i fellesskap. Som Brennan og Resnick (2012, s. 10–11) beskriver kan algoritmisk tenkning utvikles gjennom arbeid i sosiale interaksjoner. Dialogen og samarbeidet mellom elevene er viktig slik at de gjennom diskusjon og refleksjon kan utvikle algoritmisk tenkning. Lærerne bruker programmeringsoppskrifter hentet fra internett og elever som er flinke med programmering som viktige ressurser i undervisningen sin. Slike momenter mener lærerne har bidratt til at de har kommet godt i gang med programmeringsundervisning i matematikkfaget. Lærerne erfarer at programmering skaper god variasjon i undervisningen, og det viser seg å være et verktøy som er med å engasjere elevene i stor grad. Det er vanskelig å konkludere hvorfor dette engasjerer elevene, men en mulig forklaring kan være at bruken av Super:bit gir en annen form for undervisning. Bruk av Super:bit i matematikkundervisningen har en praktisk tilnærming og spiller på noen av elevenes interesse for data. Den praktiske tilnærmingen ved bruk av pedagogiske roboter fra Super:bit bidrar til visualisering. Dette er gunstig og gjør at Super:bit kan brukes som et godt verktøy for tilpasset opplæring og differensiering i matematikkundervisningen. Likevel kan dette skyldes lærernes beskrivelse av programmering som en lek-tilnærmet aktivitet. Lærerne erfarer at elevene er en viktige bidragsytere i undervisningen og at de på denne måten er med på å støtte læreren sin kunnskap på feltet. Det blir videre vist til flere matematiske emner i intervjuene knyttet til matematikk. Problemet her er at lærerne i liten grad bruker dette aktivt i egen undervisning. Det blir beskrevet en programmeringsundervisning uten matematikk, men likevel er det flere elementer ved undervisningen vi mener kan knyttes til matematiske emner og algoritmisk

tenkning, men da uten en konkretisering. Denne konkretiseringen er spesielt viktig og nødvendig for en mer effektiv og god undervisning (Angeli et al., 2019, s. 86; Hattie, 2009, s. 34).

Samlet sett vil det kreve en kompetanseheving blant lærerne for at programmering skal kunne bli brukt som et verktøy for å lære matematikk opp mot fagets mål i LK20. Oppskriftene lærerne bruker i arbeidet med Super:bit kan være begrensende for elevenes utforsking, og videre utvikling og læring av algoritmisk tenkning. En kompetanseheving kan hjelpe lærere å sette tydelige mål og vurdere undervisningen med programmering. Videre vil de kunne bruke og konkretisere matematikk og algoritmisk tenkning når elevene programmerer, for å fremme en effektiv form for læring. Fører man ikke en kompetanseheving kan en risikere at kunnskapsgapet mellom lærer og elev blir for stort. Dersom dette er tilfellet vil eleven kunne være den med mest kompetanse, noe som igjen kan føre til at læreren ikke mestrer å veilede eleven tilstrekkelig i programmeringsundervisning. Arbeidet med programmering er ment for å kunne skape et tankesett (algoritmisk tenkning) og en forståelse for hvordan den teknologiske verdenen fungerer. Eleven skal ikke bare være forbrukere, men også ha en forståelse for, og kunne utvikle ny teknologi (Meld. St. 28, 2015-2016, s. 32; NOU 2014: 7, s. 113). En kan nå dette gjennom flere måter, men det handler om å skape en forståelse for tankesettet som en bruker (Wing, 2006, s. 33). Kompetanseheving er trolig bare en av måtene. Gjennom en kompetanseheving vil en konkretisering av programmeringen i skolen være nyttig for å bruke Super:bit som et effektivt verktøy i arbeidet med å lære matematikk. På denne måten kan en unngå at undervisningen blir mangelfull (Fjørtoft et al., 2019, s. 60; Hattie, 2009, s. 36). Vi mener at studien vår kan være et positivt bidrag inn mot andre kvalitative studier med empirisk forskning. Grunnet få informanter kan ikke vår kvalitative studie generaliseres. Det er likevel grunn til å tro at flere av funnene presentert i denne studien vil kunne være sammenlignbare med andre lærere enn kun de fire som har deltatt på våre intervjuer.

6.2 Implikasjoner

Denne studien viser til at programmeringsundervisning spiller på elevens interesser og fenger flere av elevene. Hvorfor er de interessert, hva med de andre elevene? Gjennom intervjuene forteller lærere om elever som «skinner» i programmeringsundervisning og retter en finger mot de elevene som er glad i praktisk arbeid. Hvis matematikklærere med lav kompetanse ikke klarer å føre en konkretisering inn mot matematikkfaget, mener vi at det kan gå på bekostning av elever som er glad i den regnetekniske delen av matematikk.

Det er spesielt positivt at lærere kaster seg ut i programmeringsundervisning selv om lærerne uttrykker at de har lav kompetanse. Super:bit-satsingen og oppskrifter på internett har hjulpet lærerne med å starte med programmering i matematikkundervisningen sin. Likevel får vi et inntrykk

at det er matematikklærere som ikke legger programmering til matematikkundervisningen. Betydningen av en kompetanseheving for å heve undervisningskvaliteten og elevenes læringsutbytte av læringsaktivitetene, tror vi er essensielt for at matematikklæreren skal ha kunnskap til å bruke programmering som et verktøy inn mot matematikkfaget. På denne måten tror vi at lærere i større grad vil ha anledning til å favne alle elevene og ikke bare de som er interessert i programmering. I tillegg vil en høyere programmeringskompetanse kunne føre til at lærere i større grad kan se mulighetene med bruken av programmering, ikke bare i matematikkfaget, men også som en tilnærming til et tverrfaglig samarbeid mellom flere fag.

Lærerne i studien etterspør en systematisering og oppfølging av kursene knyttet til programmering og kompetanseheving. Skolene må kunne ta tak i matematikklærernes kompetansemangel innenfor programmering, slik at elevene får en tilstrekkelig undervisning som er i tråd med LK20. Følgene av å ikke føre en kompetanseheving kan ha stor betydning for hvilke elever som møter kompetente lærere innenfor programmering i matematikkfaget. Med en slik kompetanseheving kan det se ut til at programmering kan være en berikende faktor i matematikkundervisningen.

6.3 Veien videre

Studien vår har benyttet semistrukturerte intervjuer for å innhente data om lærernes tanker og erfaringer knyttet til sin egen praksis i programmeringsundervisning. For videre forskning ser vi på det som nødvendig å benytte ulike metoder for å undersøke flere viktige momenter innenfor programmeringsfeltet i matematikkfaget. Etter vår oppfatning trengs det både mer kvalitativ og kvantitativ forskning på feltet. Ved å benytte seg av kvantitativ forskning kan det bidra til å skape en empiri som komplimenter den kvalitative forskningen, som videre muliggjør for å kunne generalisere noen av slutningene. I tillegg er det tenkelig at observasjon som metode kunne bidratt til å trekke ut andre elementer fra lærerens undervisning enn det et intervju gjør. På denne måten kunne man fått et mer helhetlig bilde av programmering i matematikkfaget. Studien begrenser seg til de lærerne som har hatt programmering i skolen, men ikke de matematikklærerne som ikke har prøvd det ut. Programmering er i full fart inn i skolen gjennom LK20 og Super:bit-satsingen, så her skjer det mye. Likevel ser vi på det som realistisk at det er flere elever i skolen som ikke møter en programmeringsundervisning som er i tråd med LK20. Derfor hadde det vært interessant å inkludere matematikklærere som ikke har begynt å innføre programmering i matematikkundervisningen sin, slik at en kan se på årsakene til hvorfor programmering ikke er tatt i bruk. Videre kan framtidige studier ta for seg elever og deres tanker om programmeringsundervisning i matematikkfaget. En slik forskning kunne gitt et mer helhetlig bilde fra begge sider som innvirker i undervisningen.

7.0 Referanseliste

- Angeli, C., Xerou, E. & Nicolaou, M. (2019). *Investigating K-2 students' computational thinking skills during a problem-solving activity about the water cycle using educational robotics*. 85–91.
- Befring, E. (2020). *Sentrale forskningsmetoder: Med etikk og statistikk* (2. utg.). Cappelen Damm Akademisk.
- Benton, L., Hoyles, C., Kalas, I. & Noss, R. (2017). Bridging Primary Programming and Mathematics: Some Findings of Design Research in England. *Digital Experiences in Mathematics Education*, 3(2), 115–138. <https://doi.org/10.1007/s40751-017-0028-x>
- Bjørndal, B. & Lieberg, S. (1978). *Nye veier i didaktikken?: En innføring i didaktiske emner og begreper*. Aschehoug.
- Brennan, K. & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. *Proceedings of the 2012 Annual Meeting of the American Educational Research Association*, 1, 25.
- Brinkmann, S. & Kvale, S. (2015). *InterViews: Learning the craft of qualitative research interviewing* (3rd ed.). Sage.
- Cervera, N., Diago, P. D., Orcos, L. & Yáñez, D. F. (2020). The Acquisition of Computational Thinking through Mentoring: An Exploratory Study. *Education Sciences*, 10(8), 202. <https://doi.org/10.3390/educsci10080202>
- Christoffersen, L. & Johannessen, A. (2012). *Forskningsmetode for lærerutdanningene*. Abstrakt forl.
- Cuban, L. (2001). *Oversold and underused: Computers in the classroom*. Harvard University Press.
- De nasjonale forskningsetiske komiteene. (2015, 17. juni). *Kvalitativ metode*. Forskningsetikk. <https://www.forskningsetikk.no/ressurser/fbib/metoder/kvalitativ-metode/>
- De nasjonale forskningsetiske komiteene. (2016). *Forskningsetiske retningslinjer for samfunnsvitenskap, humaniora, juss og teologi*. Forskningsetikk. <https://www.forskningsetikk.no/retningslinjer/hum-sam/forskningsetiske-retningslinjer-for-samfunnsvitenskap-humaniora-juss-og-teologi/>
- Eggebø, H. (2020). Kollektiv kvalitativ analyse. *Norsk sosiologisk tidsskrift*, 4(2), 106–122. <https://doi.org/10.18261/issn.2535-2512-2020-02-03>
- Eidslott, H. (2021). Programmering for alle. *Bedre skole*, 33(1), 24–27.

- Engelsen, B. U. (2012). *Kan læring planlegges?: Arbeid med læreplaner—Hva, hvordan, hvorfor* (6. utg.). Gyldendal akademisk.
- Fjørtoft, S. O., Thun, S. & Buvik, M. P. (2019). *Monitor 2019—En deskriptiv kartlegging av digital tilstand i norske skoler og barnehager* (2019:00877; s. 154). SINTEF Digital.
- Frey, C. B. & Osborne, M. A. (2017). The future of employment: How susceptible are jobs to computerisation? *Technological Forecasting & Social Change*, 114(January), 254–280. <https://doi.org/10.1016/j.techfore.2016.08.019>
- Grover, S. & Pea, R. (2013). Computational Thinking in K–12: A Review of the State of the Field. *Educational Researcher*, 42(1), 38–43. <https://doi.org/10.3102/0013189X12463051>
- Grønmo, S. (2016). *Samfunnsvitenskapelige metoder* (2. utg.). Fagbokforl.
- Gundem, B. B. (1998). *Understanding European didactics - an overview: Didactics (didaktik, didaktik(k), didactique): Bd. no. 4 1998*. Universitetet i Oslo, Pedagogisk forskningsinstitutt.
- Haraldsrud, A. D., Sveinsson, H. A. & Løvold, H. H. (2020). *Programmering i skolen*. Universitetsforlaget.
- Hattie, J. (2009). *Visible learning: A synthesis of over 800 meta-analyses relating to achievement*. Routledge.
- Høgheim, S. (2020). *Masteroppgaven i GLU* (1. utgave.). Fagbokforlaget.
- Imsen, G. (2014). *Elevenes verden: Innføring i pedagogisk psykologi* (5. utg.). Universitetsforlaget.
- Imsen, G. (2020). *Elevenes verden: Innføring i pedagogisk psykologi* (6. utg.). Universitetsforlaget.
- Johannessen, L. E. F., Rafoss, T. W. & Rasmussen, E. B. (2018). *Hvordan bruke teori? Nyttige verktøy i kvalitativ analyse*. Universitetsforlaget.
- Johansen, A.-K. (2020, 11. juli). *Programmering vil bli en utfordring for lærere*. forskning. <https://forskning.no/barn-og-ungdom-hogskolen-i-ostfold-matematikk/programmering-vil-bli-en-utfordring-for-laerere/1711838>
- Jupp, V. (2006). *The Sage dictionary of social research methods*. Sage Publications.
- Kunnskapsdepartementet. (2019, 15. november). *Læreplan i matematikk 1.–10. Trinn (MAT01-05)*. Fastsatt som forskrift. Læreplanverket for Kunnskapsløftet 2020. <https://www.udir.no/lk20/mat01-05/om-faget/fagets-relevans-og-verdier?lang=nob>
- Kvale, S. & Brinkmann, S. (2015). *Det kvalitative forskningsintervju* (3. utg.). Gyldendal akademisk.

- Lincoln, Y. S. & Guba, E. G. (1985). *Naturalistic inquiry*. Sage.
- Lindsø, J. F. (2020, 27. februar). *Hva er programmering?* NdlA.
<https://ndla.no/nb/subject:1:1352b19e-e706-4480-a728-c6b0a57ba8ae/topic:1:e08eccc8-5e7e-4b85-9876-dfc5d1f3d920/resource:7e116b54-1c49-4a04-b9c3-250791824228>
- Liseter, I. M. (2020, 29. desember). Robot. I *Store norske leksikon*. <http://snl.no/robot>
- Livingstone, I. & Hope, A. (2011). *Next gen: Transforming the UK into the world's leading talent hub for the video games and visual effects industries : a review*. National Endowment for Science, Technology and the Arts.
- Lye, S. Y. & Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12? *Computers in Human Behavior*, 41, 51–61. <https://doi.org/10.1016/j.chb.2014.09.012>
- Lyngsnes, K. M. & Rismark, M. (2007). *Didaktisk arbeid* (2. utg.). Gyldendal.
- Lyngsnes, K. M. & Rismark, M. (2014). *Didaktisk arbeid* (3. utg.). Gyldendal akademisk.
- Lær Kidsa Koding. (2021, 14. desember). *Koding i skolen*. kidsakoder.
<https://www.kidsakoder.no/skole/>
- Meld. St. 28. (2015-2016). *Fag – Fordypning – Forståelse. En fornyelse av Kunnskapsløftet*. Kunnskapsdepartementet. <https://www.regjeringen.no/no/dokumenter/meld.-st.-28-20152016/id2483955/>
- Micro:bit Educational Foundation. (u.å.a). *Introducing the BBC micro:bit*. Micro:Bit.
<https://microbit.org/get-started/first-steps/introduction/>
- Micro:bit Educational Foundation. (u.å.b). *Introduction*. Micro:bit.
<https://microbit.org/impact/foundation-reports/>
- Micro:bit Educational Foundation. (u.å.c). *Super:bit*. Micro:Bit.
<https://microbit.org/impact/foundation-reports/norway/superbit/>
- Micro:bit Educational Foundation. (u.å.d). *The BBC micro:bit in the UK*. Micro:Bit.
<https://microbit.org/impact/case-studies/milestones-for-the-bbc-microbit/>
- Moldøen, H. (2019, 10. mai). *Super:satsing på programmering med micro:bit for barn i hele Norge*. kidsakoder. <https://www.kidsakoder.no/2019/05/10/supersatsing-pa-programmering-med-microbit-for-barn-i-hele-norge/>
- n00b.no. (u.å.). *Bit:bot* [Fotografi]. n00b.no. <https://oppgaver.n00b.no/micro-bit/bit-bot>
- NOU 2014: 7. (2014). *Elevenes læring i fremtidens skole—Et kunnskapsgrunnlag: Utredning fra ekspertutvalg oppnevnt ved kongelig resolusjon 21. Juni 2013. Avgitt til*

- Kunnskapsdepartementet 3.september 2014*. Kunnskapsdepartementet.
<https://www.regjeringen.no/no/dokumenter/NOU-2014-7/id766593/>
- Nouri, J., Zhang, L., Mannila, L. & Norén, E. (2020). Development of computational thinking, digital competence and 21st century skills when learning programming in K-9. *Education Inquiry*, 11(1), 1–17. <https://doi.org/10.1080/20004508.2019.1627844>
- Nygård, K. (2018). *Programmering i skolen: Hvordan komme i gang?* (M. Saabye, Red.). Pedlex.
- Piedade, J., Dorotea, N., Pedro, A. & Matos, J. F. (2020). On Teaching Programming Fundamentals and Computational Thinking with Educational Robotics: A Didactic Experience with Pre-Service Teachers. *Education Sciences*, 10(9), 214.
<https://doi.org/10.3390/educsci10090214>
- Polya, G. & Conway, J. H. (1985). *How to Solve It: A New Aspect of Mathematical Method* (Expanded Princeton Science Library edition., Bd. 34). Princeton University Press.
- Randen, S. (2008). *Den didaktiske relasjonsmodellen* [Illustrasjon]. Rundsager og kanon.
<http://roundabout-rundsangerogkanon.blogspot.com/2008/11/tennplugger-til-tankevirksomhet.html>
- Rossen, E. (2020, 15. juli). Psprogrammering. I *Store norske leksikon*.
<http://snl.no/programmering>
- Schwandt, T. A., Lincoln, Y. S. & Guba, E. G. (2007). Judging interpretations: But is it rigorous? trustworthiness and authenticity in naturalistic evaluation. *New Directions for Evaluation*, 2007(114), 11–25. <https://doi.org/10.1002/ev.223>
- Sevik, K. (2016). *Programmering i skolen* (s. 32). Utdanningsdirektoratet.
https://www.udir.no/globalassets/filer/programmering_i_skolen.pdf
- Skaalvik, E. M. & Skaalvik, S. (2021). *Skolen som læringsarena: Selvoppfatning, motivasjon og læring* (4. utg.). Universitetsforl.
- Smyrnova-Trybulska, E., Morze, N., Kommers, P., Zuziak, W. & Gladun, M. (2017). Selected aspects and conditions of the use of robots in STEM education for young learners as viewed by teachers and students. *Interactive Technology and Smart Education*, 14(4), 296–312. <https://doi.org/10.1108/ITSE-04-2017-0024>
- Super:bit. (u.å.a). *Hva er Super:bit?* Super:bit. <https://www.superbit.no/hva-er-superbit/>
- Super:bit. (u.å.b). *Lær andre programmering*. Super:bit. <https://www.superbit.no/laer-andre-programmering/>
- Super:bit. (u.å.c). *Utstyr til super:bit*. Super:bit. <https://www.superbit.no/utstyr-til-superbit/>
- Säljö, R. (2001). *Læring i praksis: Et sosiokulturelt perspektiv*. Cappelen akademisk.

- Säljö, R. (2020). Støtte til læring- tradisjoner og perspektiver. I R. J. Krumsvik & R. Säljö (Red.), *Praktisk-pedagogisk utdanning: En antologi* (2. utgave., s. 57–83). Fagbokforlaget.
- Tjora, A. (2017). *Kvalitative forskningsmetoder i praksis* (3. utg.). Gyldendal Akademisk.
- Tjora, A. (2021). *Kvalitative forskningsmetoder i praksis* (4. utg.). Gyldendal.
- Torkildsen, S. H. & Stedøy, I. M. (2018). *Hvorfor problemløsning*. Matematikksenteret.
<https://www.matematikksenteret.no/sites/default/files/attachments/resources/Hvorfor%20probleml%C3%B8sing.pdf>
- Utdanningsdirektoratet. (2019a). *Algoritmisk tenkning*. Udir. <https://www.udir.no/kvalitet-og-kompetanse/profesjonsfaglig-digital-kompetanse/algoritmisk-tenkning/>
- Utdanningsdirektoratet. (2019b). *Den algoritmiske tenkeren* [Illustrasjon]. Udir.
<https://www.udir.no/kvalitet-og-kompetanse/profesjonsfaglig-digital-kompetanse/algoritmisk-tenkning/>
- Utdanningsdirektoratet. (2019c, 3. juli). *Den teknologiske skolesekken*. Udir.
<https://www.udir.no/kvalitet-og-kompetanse/nasjonale-satsinger/den-teknologiske-skolesekken/>
- Vestøl, J. M. (2008). Didaktiske modeller i lærerutdanningen. En analyse av lærerstudenters praksisrefleksjon. *Acta Didactica Norge*, 2(1). <https://doi.org/10.5617/adno.1023>
- Vihovde, E. H. (2020, 27. november). Dataprogram. I *Store norske leksikon*.
<http://snl.no/dataprogram>
- Vitensentrene. (u.å.a). *Micro:bit kodebrikke* [Fotografi]. Vitensentrene.
<https://www.vitensenter.no/superbit/>
- Vitensentrene. (u.å.b). *Super:bit*. Vitensentrene. <https://www.vitensenter.no/superbit/>
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33–35.
<https://doi.org/10.1145/1118178.1118215>
- Wølner, T. A. (2013). *Kriteriebasert vurdering*. Universitetsforl.

Vedlegg

Vedlegg 1: NSD godkjenning av meldeskjema

12.05.2022, 22:55

Meldeskjema for behandling av personopplysninger



Vurdering

Referansenummer

950382

Prosjektittel

Forskningsprosjekt om programmering knyttet til super:bit på mellomtrinnet

Behandlingsansvarlig institusjon

Høgskulen på Vestlandet / Fakultet for lærerutdanning, kultur og idrett / Institutt for språk, litteratur, matematikk og tolkning

Prosjektansvarlig (vitenskapelig ansatt/veileder eller stipendiat)

Type prosjekt

Studentprosjekt, masterstudium

Kontaktinformasjon, student

Prosjektperiode

29.09.2021 - 09.12.2022

Vurdering (1)

08.10.2021 - Vurdert

Det er vår vurdering at behandlingen av personopplysninger i prosjektet vil være i samsvar med personvernlovgivningen så fremt den gjennomføres i tråd med det som er dokumentert i meldeskjemaet med vedlegg 08.10.2021. Behandlingen kan starte.

TYPE OPPLYSNINGER OG VARIGHET

Prosjektet vil behandle alminnelige kategorier av personopplysninger frem til 09.12.2022.

LOVLIG GRUNNLAG

Prosjektet vil innhente samtykke fra de registrerte til behandlingen av personopplysninger. Vår vurdering er at prosjektet legger opp til et samtykke i samsvar med kravene i art. 4 og 7, ved at det er en frivillig, spesifikk, informert og utvetydig bekreftelse som kan dokumenteres, og som den registrerte kan trekke tilbake. Lovlig grunnlag for behandlingen vil dermed være den registrertes samtykke, jf. personvernforordningen art. 6 nr. 1 bokstav a.

PERSONVERNPRINSIPPER

NSD vurderer at den planlagte behandlingen av personopplysninger vil følge prinsippene i

<https://meldeskjema.nsd.no/vurdering/61486483-d0ce-4033-8b93-828811ff7048>

1/2

personvernforordningen om:

- lovlighet, rettferdighet og åpenhet (art. 5.1 a), ved at de registrerte får tilfredsstillende informasjon om og samtykker til behandlingen
- formålsbegrensning (art. 5.1 b), ved at personopplysninger samles inn for spesifikke, uttrykkelig angitte og berettigede formål, og ikke behandles til nye, uforenlige formål
- dataminimering (art. 5.1 c), ved at det kun behandles opplysninger som er adekvate, relevante og nødvendige for formålet med prosjektet
- lagringsbegrensning (art. 5.1 e), ved at personopplysningene ikke lagres lengre enn nødvendig for å oppfylle formålet

DE REGISTRERTES RETTIGHETER

Så lenge de registrerte kan identifiseres i datamaterialet vil de ha følgende rettigheter: åpenhet (art. 12), informasjon (art. 13), innsyn (art. 15), retting (art. 16), sletting (art. 17), begrensning (art. 18), underretning (art. 19), dataportabilitet (art. 20).

NSD vurderer at informasjonen om behandlingen som de registrerte vil motta oppfyller lovens krav til form og innhold, jf. art. 12.1 og art. 13.

Vi minner om at hvis en registrert tar kontakt om sine rettigheter, har behandlingsansvarlig institusjon plikt til å svare innen en måned.

FØLG DIN INSTITUSJONS RETNINGSLINJER

NSD legger til grunn at behandlingen oppfyller kravene i personvernforordningen om riktighet (art. 5.1 d), integritet og konfidensialitet (art. 5.1. f) og sikkerhet (art. 32).

For å forsikre dere om at kravene oppfylles, må dere følge interne retningslinjer og/eller rådføre dere med behandlingsansvarlig institusjon.

MELD VESENTLIGE ENDRINGER

Dersom det skjer vesentlige endringer i behandlingen av personopplysninger, kan det være nødvendig å melde dette til NSD ved å oppdatere meldeskjemaet. Før du melder inn en endring, oppfordrer vi deg til å lese om hvilke type endringer det er nødvendig å melde:

https://nsd.no/personvernombud/meld_prosjekt/meld_endringer.html

Du må vente på svar fra NSD før endringen gjennomføres.

OPPFØLGING AV PROSJEKTET

NSD vil følge opp ved planlagt avslutning for å avklare om behandlingen av personopplysningene er avsluttet.

Lykke til med prosjektet!

Tlf. Personverntjenester: 55 58 21 17 (tast 1)

Vedlegg 2: Informasjonsskriv og samtykkeerklæring

Vil du delta i forskningsprosjektet om programmering knyttet til Super:bit på mellomtrinnet?

Dette er et spørsmål til deg om å delta i et forskningsprosjekt hvor formålet er å se på hvordan lærere bruker programmering i undervisningen – nærmere bestemt materiell fra **Super:bit**. I dette skrivet gir vi deg informasjon om målene for prosjektet og hva deltakelse vil innebære for deg.

Formål

I studiet vårt har vi som formål å se på hvordan matematikklærere på mellomtrinnet bruker materiell fra Super:bit-satsingen i sin programmeringsundervisning. Vi vil høre hvordan de legger til rette for læring av matematikk gjennom arbeidet med Super:bit, og hvilke erfaringer de gjør seg knyttet til dette. Videre vil vi høre lærerens tanker om Super:bit som et hjelpemiddel og tilskudd til skolen sin matematikkundervisning. Forskningsprosjektet vil til slutt ta form av en masteroppgave som vi håper lærere kan se nytte av og finne ideer inn til sin undervisning.

Hvem er ansvarlig for forskningsprosjektet?

Høgskulen på Vestlandet er ansvarlig for prosjektet.

Hvorfor får du spørsmål om å delta?

For å avgrense vårt prosjekt har vi valgt å invitere deltakere ut ifra spesielle kriterier. Informantene våre utgjør et utvalg matematikklærere, hvor kriteriene var at de arbeider på mellomtrinnet og har erfaring med programmering i matematikkundervisningen knyttet til den nasjonale Super:bit-satsingen. Med erfaring mener vi at læreren har undervist minimum et opplegg med Super:bit og gjort seg tanker om undervisningen.

Hva innebærer det for deg å delta?

Om du velger å delta i prosjektet innebærer det et individuelt intervju på ca. 60 minutter. Vi har et ønske om å komme til skolen for å gjennomføre intervjuet, helst i løpet av november 2021. Det vil bli tatt lydopptak og notater fra intervjuet, og intervjuet vil bli transkribert. I intervjuet ønsker vi å få fram hvordan lærere legger til rette for læring av matematikk gjennom bruk av programmering med Super:bit. Samt hvilke erfaringer de gjør seg knyttet til dette.

Det er frivillig å delta

Det er frivillig å delta i prosjektet. Hvis du velger å delta, kan du når som helst trekke samtykket tilbake uten å oppgi noen grunn. Alle dine personopplysninger vil da bli slettet. Det vil ikke ha noen negative konsekvenser for deg hvis du ikke vil delta eller senere velger å trekke deg.

Ditt personvern – hvordan vi oppbevarer og bruker dine opplysninger

Vi vil bare bruke opplysningene om deg til formålene vi har fortalt om i dette skrevet. Vi behandler opplysningene konfidensielt og i samsvar med personvernregelverket. Det vil kun være Benjamin Haastø Sunde og Eivind Øen, samt veilederne Frode Olav Haara og Lene Hayden Taraldsen som vil ha tilgang til informasjonen som kommer fram gjennom intervjuet. Opplysningene som blir gitt her vil ikke gjenkjennes i publikasjonen av masteroppgaven.

Hva skjer med opplysningene dine når vi avslutter forskningsprosjektet?

Opplysningene anonymiseres når prosjektet avsluttes/oppgaven er godkjent, noe som etter planen er desember 2022. Alle navn og lydopptak knyttet til intervjuet vil bli slettet etter prosjektavslutning.

Hva gir oss rett til å behandle personopplysninger om deg?

Vi behandler opplysninger om deg basert på ditt samtykke.

På oppdrag fra Høgskulen på Vestlandet har NSD – Norsk senter for forskningsdata AS vurdert at behandlingen av personopplysninger i dette prosjektet er i samsvar med personvernregelverket.

Dine rettigheter

Så lenge du kan identifiseres i datamaterialet, har du rett til:

- innsyn i hvilke opplysninger vi behandler om deg, og å få utlevert en kopi av opplysningene
- å få rettet opplysninger om deg som er feil eller misvisende
- å få slettet personopplysninger om deg
- å sende klage til Datatilsynet om behandlingen av dine personopplysninger

Hvis du har spørsmål til studien, eller ønsker å vite mer om eller benytte deg av dine rettigheter, ta kontakt med:

- Studenter: Eivind Øen og Benjamin Haastø Sunde

Hvis du har spørsmål knyttet til NSD sin vurdering av prosjektet, kan du ta kontakt med:

- NSD – Norsk senter for forskningsdata AS på epost (personverntjenester@nsd.no) eller på telefon: 55 58 21 17.

Med vennlig hilsen

Benjamin Haastø Sunde og Eivind Øen

Frode Olav Haara og Lene Hayden Taraldsen
(Forsker/veileder)

Samtykkeerklæring

Jeg har mottatt og forstått informasjon om prosjektet programmering knyttet til Super:bit på mellomtrinnet, og har fått anledning til å stille spørsmål.

Jeg samtykker til:

- å delta i intervju med bruk av lydopptak

Jeg samtykker til at mine opplysninger behandles fram til prosjektet er avsluttet.

(Signert av prosjektdeltaker, dato)

Vedlegg 3: Intervjuguide

Innledning

Takk! – Takke respondenten for at de stiller opp

Hvorfor? – Fortelle hva formålet med intervjuet er

Anonymitet? – Fortelle hvordan dataene blir behandlet. Avklare tillatelse til å ta opp intervjuet.

Innhold – Kort gjennomgang av hva intervjuet skal handle om.

Tid – Hvor lang tid kan respondenten regne med at intervjuet tar?

Spørsmål: navn, alder og hvor lenge arbeidet i skolen?

Hvordan legger fire lærere til rette for læring av matematikk på 6.trinn gjennom bruk av pedagogiske roboter fra Super:bit-satsingen, og hvilke erfaringer gjør de seg knyttet til dette?

STRUKTUR:

Del 1:

- Vi har ulike begrep som vi vil du skal beskrive for oss:
 - Programmering
 - Koding
 - Algoritmisk tenkning
- Hvor stor interesse har du for programmering og når kom denne interessen?
- Hvordan har du lært deg å programmere?
- Hvor mye har skolen din lagt til rette for programmering?
 - Utstyr
 - Kurs
 - Annen tilrettelegging?
- Hvilken kjennskap har du til Super:bit eventuelt satsingen?
- Hvilke trinn har du undervist programmering på?

Del 2:

- Hvor mye har du undervist i programmering?
- Hvor mye har du brukt materiell fra Super:bit?
 - Hvilke komponenter fra Super:bit har du brukt og hvorfor?
- Kan du beskrive en undervisningsøkt med Super:bit?
 - Planlegging
 - Gjennomføring
 - Konkrete eksempler på bruk av Super:bit
 - Refleksjon etter undervisning

Del 3:

- Hvilke matematiske begrep ser du som sentrale knyttet til programmering?
 - I læreplanen er algoritmisk tenkning nevnt i den generelle delen. Kommer dette til syne når elevene arbeider med programmering?
- Har du hatt nytte av å jobbe med programmering i matematikk?
 - Har elevene det også?
- Ser du Super:bit som nyttig å ha i skolen, hvorfor?
- Kan du se noe programmering tilfører matematikken i dagens skole/samfunn?
- Hva syns du om programmering i skolen?
- Hvordan tror du programmering vil være en del av morgendagens skole?

Del 4:

- Hvis du skulle trekke ut tre ting som du mener er det viktigste vi har snakket om, hva ville det vært?
- Er det noe mer du vil si eller legge til?
- Kan vi kontakte deg igjen hvis det blir aktuelt?
- Tusen takk for at du stilte opp!