



UiT Norges arktiske universitet

Institutt for pedagogikk og lærerutdanning

Hvilke forestillinger kan elever ha om programmering i matematikk

En kvalitativ studie om elevers forestillinger

Sjur Schjelderup Hermansen

Masteroppgave i matematikdidaktikk 5.-10. trinn – LRU-3903 - mai 2021

Forord

Jeg vil rette en stor takk til min veileder Kjersti Wæge. Uten henne ville nok ikke denne oppgaven vært en realitet. Tross avstand og utfordringer i sammenheng med Covid-19 har Kjersti vært tilgjengelig, blid og ikke minst en utrolig veileder.

En takk går også til mine medstudenter.

Spesielt kaffegjengen på masterkontoret de siste månedene.

Jeg ønsker å takke min familie som alltid stiller opp hvis jeg trenger noe.

Til slutt vil jeg rette en stor takk til Christine for støtte, tålmodighet og kjærlighet, særlig under ferdigstillingen av masteroppgaven.

Tromsø, mai 2021

Sjur Schjelderup Hermansen

Sammendrag

Målet med denne studien har vært å forsøke å få innsikt i hvilke forestillinger elever kan ha til programmering i matematikk. I studien benytter jeg meg av en modell utviklet av McLeod (1992) for å strukturere forestillingene til elevene i fire kategorier. De fire kategoriene er som følger: forestillinger om programmering i matematikk, forestillinger om selvet, forestillinger om programmering i matematikkundervisning og forestilling om den sosiale konteksten.

Studien tar utgangspunkt i en 6. klasse som har deltatt på Nordnorsk vitensenterets super:bit prosjekt. Jeg benytter en kvalitativ metode for å samle inn datamateriale til studien. Jeg benyttet meg av semistrukturerte intervjuer for å snakke med åtte elever fra klassen. Formålet med intervjuene er å forsøke å danne meg et bilde av hvilke forestillinger elevene kan ha om programmering i matematikk. Datamaterialet fra intervjuene transkriberes, for så å struktureres og analyseres med hjelp av modellen til McLeod (1992).

Resultatene fra studien indikerer at elevene har noe forskjellige forestillinger om programmering i matematikk. Jeg tolker resultatet slikt at elevenes forestillinger om programmering i matematikk kan knyttes til undervisningen og arbeidsmetodene de har erfart. Videre at elevens personlige fokus gjenspeiles i forestillinger om selvet. Resultatet fra analysen viser at modellen jeg brukte for å strukturere forestillingene fungerte godt for å organisere og tydeliggjøre forestillingene til elevene.

Innholdsfortegnelse

1	Innledning.....	1
1.1	Forskningsspørsmål.....	2
1.2	Kapitteloppbygging.....	2
2	Forestillinger.....	5
2.1	Organisering av forestillinger.....	6
2.2	Forestillingers natur.....	7
2.3	Strukturering av forestillinger.....	8
3	Programmering.....	11
3.1	Programmering i skolen.....	11
3.2	Programmering i matematikk.....	13
4	Algoritmisk tenkning.....	15
5	Metode.....	17
5.1	Valg av metode.....	17
5.2	Utvalg av informanter.....	17
5.3	Intervju som metode.....	19
5.3.1	Etiske aspekter.....	20
5.3.2	Intervjuguide.....	21
5.3.3	Gjennomføring av intervju.....	23
6	Analyse.....	25
6.1	Intervju – Pia.....	25
6.1.1	Oppsummering intervju med Pia.....	28
6.2	Intervju – Markus.....	28
6.2.1	Oppsummering intervju med Markus.....	32
6.3	Intervju – Tore.....	32
6.3.1	Oppsummering intervju med Tore.....	35

6.4	Intervju – Aslak	36
6.4.1	Oppsummering intervju med Aslak	40
6.5	Intervju – Mari.....	41
6.5.1	Oppsummering intervju med Mari.....	44
6.6	Intervju – Randi	45
6.6.1	Oppsummering intervju med Randi	48
6.7	Intervju – Anne.....	49
6.7.1	Oppsummering intervju med Anne	52
7	Resultat.....	55
7.1	Validitet og reliabilitet.....	59
7.2	Begrensninger og videre forskning.....	61
	Referanseliste	63
	Vedlegg	67
	Vedlegg 1 Meldeskjema for behandling av personopplysninger	67
	Vedlegg 2 Informasjonsskriv og samtykkeskjema.....	72
	Vedlegg 3 intervjuguide	75

Figurliste

Figur 1: McLeods affektive spekter (Wedegge, Skott, Henningsen, & Wæge, 2006).....	8
Figur 2: Kategorisering av forestillinger (McLeod, 1992).....	8
Figur 3: Kategorisering av forestillinger om programmering i matematikk.....	10

1 Innledning

«I dag la regjeringen frem en ny digital strategi for grunnskolen og videregående opplæring. – Vi vil at forståelse for koding og teknologi skal inn i læreplanene allerede fra barneskolen. Med denne strategien peker vi inn i et nytt tiår for norsk skole, sier statsminister Erna Solberg.» (Kunnskapsdepartementet, 2017)

Det har i de siste årene vært kraftige strømninger for å få programmering inn som en del av skolehverdagen. Vi står i dag midt i det Schwab (2017) kaller den fjerde industrielle revolusjon. Før Norge innførte programmering som en del av læreplanen, hadde allerede en rekke europeiske land gjort det samme (Kaufmann & Stenseth, 2020).

Helt siden jeg først hørte at programmering eller koding skulle introduseres i den norske skolen, har det vært noe jeg synes er veldig spennende. Personlig har jeg erfaring med programmering fra tidligere studier, samt en stor interesse for hvordan programmering kan benyttes i skolesammenheng. Det ble derfor naturlig at tema for min masteroppgave skulle omhandle nettopp programmering i skolen.

Når det kommer til undervisning, så har de aller fleste en slags forestilling om hvordan en undervisningssituasjon utspiller seg (Kamiya, 2018). Alt fra hvordan klassen organiseres, til lærerens og elevenes ulike roller og deres samspill. Om vi nå tenker litt mer spesifikt, for eksempel at det er programmering klassen skal arbeide med. Det kan nå hende at vår opprinnelige forestilling om klasserommet og læringsstrukturen ikke passer like godt. For å drive undervisning med programmering må vi kanskje forestille oss at elevene trenger noe annet en blyant og papir framfor seg på pulten?

Det finnes ikke mye forskning om elevers forestillinger om programmering i matematikk. Forsström og Kaufmann (2018) gjennomførte en gjennomgang av tidligere forskning omkring bruken av programmering i matematikkfaget. De gjennomgikk litteratur utelukkende fra land hvor læreplanen hadde underlagt programmering matematikkfaget. Formålet med denne gjennomgangen var å undersøke de forskningsbaserte begrunnelsene for å innføre programmering som en del av pensum i matematikk. Forsström og Kaufmann identifiserte tre hovedmomenter, elevers motivasjon til å lære matematikk, elevers matematiske prestasjoner og samarbeid mellom elever og endringen av lærerens rolle. De fant ut at bruk av programmering i matematikkundervisning i noen tilfeller kunne øke elevens motivasjon for å lære matematikk, og samtidig øke elevprestasjoner i matematikk. I denne sammenheng

diskuteres lærerens og elevenes roller i det de kaller den kollektive læringsprosessen (Forsström & Kaufmann, 2018). Et annet eksempel er Dohn (2020) som har gjennomført et forskningsprosjekt om bruken av programmeringsspråket Scratch i to sjetteklasser. Dohn undersøkte hvordan Scratch-programmering påvirket elevers motivasjon og interesse for programmering i matematikk. Resultatet viste at bruken av Scratch-programmering resulterte i en liten, men ifølge Dohn signifikant negativ effekt på elevers gjennomsnittlige interesse for programmering og matematikk. Kritikken til Scratch-programmering omhandlet blant annet arbeidsflyten, vanskelighetsgraden til oppgavene og begrensninger i Scratch. Den uinspirerende måten oppgavene ble presenterte for elevene på sett i sammenheng med problemene nevnt ovenfor, gjorde det vanskelig for elevene å finne motivasjon for å programmere i Scratch. Dohn (2020) påpeker viktigheten i å utvikle designprinsipper som engasjerer elevene. Videre understreker han behovet for å tilrettelegge slik at man kan utvikle lærerens designkunnskaper.

1.1 Forskningsspørsmål

Formålet med undersøkelsen min er å få innsikt i hva elevene forbinder med programmering i matematikkundervisningen. Jeg stiller meg følgende forskningsspørsmål:

Hvilke forestillinger kan elever ha om programmering i matematikk?

I min studie undersøker jeg 11-12-årige elevers forestillinger om programmering i matematikk. For å svare på forskningsspørsmålet anvender jeg den kvalitative forskningsmetoden intervju. Jeg benytter McLeod (1992) sin modell for å strukturere elevenes forestillinger inn i fire forestillingskategorier for enklere å kunne presentere funnene mine.

1.2 Kapitteloppbygging

I kapittel 2 redegjør jeg for tidligere forskning på elevers forestillinger i sammenheng med matematikk, og redegjør for hvilken definisjon av begrepet forestillinger jeg benytter i studien. Videre forklarer jeg omkring organisering av forestillinger, forestillingers natur og hvordan forestillinger kan struktureres. Kapittel 3 handler om programmering, herunder programmering generelt, programmering i skolen og programmering i matematikk. Temaet

for kapittel 4 er algoritmisk tenkning. Jeg redegjør for begrepets betydning, og begrepets relevans for programmering i skolen. Kapittel 5 omhandler metoden som jeg bruker i studien min. Jeg begrunner hvorfor jeg har valgt denne metoden, og forteller om valg av informanter. Til sist redegjør jeg for intervju som metode. Jeg gjennomgår blant annet jeg de etiske aspektene, utforming av intervjuguide og en beskrivelse av gjennomføringen av intervjuene. I kapittel 6 tar jeg for med analysen. Jeg presenterer intervjuene og deler av transkripsjonene fra intervjuene. Til sist oppsummerer jeg hovedfunnene bak hvert intervju. Kapittel 7 presenterer funnene mine. Jeg drøfter validiteten og reliabiliteten til oppgaven. Til sist gjennomgår jeg eventuelle begrensninger og ideer til videre forskning på temaet.

2 Forestillinger

I min studie ønsker jeg å undersøke hvilke forestillinger (beliefs) elever kan ha om programmering i matematikk. Forskernes interesse for forestillinger i sammenheng med matematikk økte ifølge Kloosterman (2002) betraktelig tidlig på nittitallet. Jeg vil i dette kapitlet presentere teori og forskning om forestillinger i matematikk. Jeg vil benytte den teorien og forskningen som grunnlag for denne studien om forestillinger om programmering i matematikk.

Forskning framhever betydningen av å aktivisere elevene både kognitivt og følelsesmessig når de lærer matematikk (Pehkonen, Törner, & Leder, 2003). Spangler (1992) understreker betydningen av at læreren er bevisst elevenes forestillinger, og at elevene selv er bevisst sine egne forestillinger. Også Markovits og Forgasz (2017) fremhever viktigheten av elevenes forestillinger. De beskriver hvordan elevens forestillinger kan være med på å forme måten eleven forstår matematikk på og hvordan de lærer matematikk best. Videre påpeker de at forestillinger også kan sees i sammenheng med elevens prestasjoner i matematikk. En studie av japanske og amerikanske elevens forestillinger viste at elevene som hadde en forestilling om matematikk som et fag som premierte hardt arbeid, presterte gjennomsnittlig langt bedre enn elever uten disse forestillingene (Markovits & Forgasz, 2017). I forbindelse med forestillinger om programmering i matematikk kan det være hensiktsmessig å skille mellom objektiv og subjektiv kunnskap (Furinghetti & Pehkonen, 2002). Innenfor programmering i matematikk vil objektiv kunnskap være kunnskap som er allment akseptert som gjeldende innenfor fagfeltet. Den subjektive kunnskapen derimot er kun basert på personlige tolkninger og erfaringer. Med andre ord skaper en person sin egen subjektive kunnskap, basert på deres forståelse og tolkning av objektiv kunnskap i sammenheng med personlige livserfaringer. Dermed blir kunnskapen unik fra ett individ til et annet. Kunnskapen blir personlig, og for utenforstående vil ikke den subjektive kunnskapen kunne anses som gjeldende.

Jeg betrakter elevenes forestillinger om programmering i matematikk som en del av deres subjektive kunnskap. Det finnes et bredt spekter av forskning på matematikkrelaterte forestillinger (Op't Eynde, De Corte, & Verschaffel, 2002). I likhet med andre forskere, poengterer Zhang og Morselli (2016, i Hannula et al., 2016) at det ikke finnes noen felles definisjon blant forskere om forestillinger. Jeg velger i denne oppgaven å benytte meg av Op't Eynde et al. (2002) sin definisjon av forestillinger. Definisjonen deres er et forsøk på å

sammenfatte delvis forskjellige definisjoner og karakteristikk om forestillinger fra tidligere forskning.

«Students' mathematics-related beliefs are the implicitly or explicitly held subjective conceptions students hold to be true about mathematics education, about themselves as mathematicians, and about the mathematics class context. These beliefs determine in close interaction with each other and with students' prior knowledge their mathematical learning and problem solving in class (Op't Eynde et al., 2002, p. 27).»

I denne definisjonen omhandler matematikkrelaterte forestillinger ikke bare de implisitte, men også de eksplisitte subjektive oppfatningene til elevene om matematikk som fag, om seg selv som matematikere og om matematikklasserommet som kontekst. Videre sier definisjonen noe om hvordan forestillingene sammen med tidligere kunnskap påvirker hvordan eleven lærer matematikk og problemløsning.

2.1 Organisering av forestillinger

Vi mennesker blir konstant bombardert med inntrykk når vi beveger oss rundt i verden. Vi tolker nye inntrykk på grunnlag av våre tidligere erfaringer og forestillinger. På grunn av dette vil vi hele tiden re-evaluere våre forestillinger. Vi vil hele tiden sammenligne våre egne forestillinger med nye opplevelser, erfaringer og med andre individers reaksjoner.

Et individs samling av forestillinger samles og organiseres i et forestillingssystem. Dette systemet er en samling av alle individets forestillinger både bevisste og ubevisste (Green, 1971). Forestillingssystemet til Green er basert på tre dimensjoner. Dimensjonene er kvasi-logisk (quasi-logicalness), psykologisk sentralitet (psychological centrality) og klynger (cluster structure) (Furinghetti & Pehkonen, 2002).

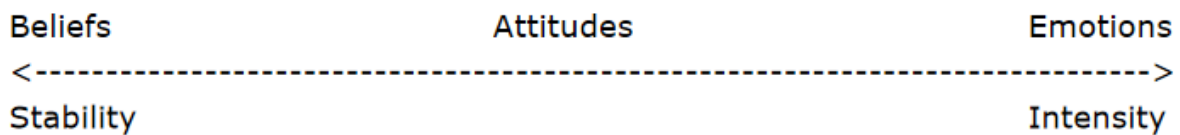
Kunnskapssystemer følger ofte en logisk oppbygning. Green (1971) skriver at et individ organiserer forestillinger på grunnlag av hvordan individet selv tolker koblingen mellom forestillingene. Dermed blir forestillingssystemet *kvasi-logisk*. Det er individuelt og vil ikke gi mening for andre. *Psykologisk sentralitet* betyr at enkelte forestillinger er mer sentrale for et individ enn andre og betyr at graden av psykologisk overbevisning omkring forestillinger varierer. Individets sentrale forestillinger i systemet kan vi kalle primærforestillinger, de

mindre sentrale forestillingene kan vi kalle periferiforestillinger. Forestillinger får dermed sin egen psykologiske styrke basert på graden av overbevisning. Dette resulterer i at primærforestillinger vil være vanskeligere å påvirke eller forandre, sammenlignet med periferiforestillinger (Furinghetti & Pehkonen, 2002). Green (1971) påpeker at ingen forestilling er totalt uavhengig andre forestillinger. Individet organiserer forestillinger i *klynger*. Klyngene består av forestillinger som for individet har sammenheng. På grunn av den kvasi-logiske struktureringen kan forestillinger som er direkte motstridende være i sammen klynge (Furinghetti & Pehkonen, 2002).

2.2 Forestillingers natur

Ikke alle forskere er interessert i å utforske relasjonen mellom forestillinger og kunnskap. Flere forskere mener det er noe unødvendig og «akademisk-filosofisk» over den diskusjonen (Furinghetti & Pehkonen, 2002). Furinghetti og Pehkonen derimot understreker viktigheten med å undersøke sammenhengen mellom forestillinger og kunnskap. De er av den oppfatningen at hvis en ser på relasjonen mellom forestillinger og kunnskap, vil forestillinger kunne regnes som en representasjon av en persons kognitive struktur. Videre argumenterer Furinghetti og Pehkonen for at forestillinger kan sees på som en reaksjon på en bestemt situasjon eller hendelse. Med det utgangspunktet vil forestillinger kunne anses som en type affektiv respons. Noen forskere er av den mening at forestillinger kun er forbundet med kognitive prosesser. Forskere flest erkjenner at forestillinger i stor grad er av kognitiv natur, men at forestillinger innehar noen affektive elementer (Furinghetti & Pehkonen, 2002).

McLeod (1992) hevder også at forestillinger i stor grad er kognitive av natur. Likevel påpeker han at forestillinger spiller en sentral rolle i utviklingen av holdningsmessige og følelsesmessige reaksjoner til matematikk. Derfor skriver McLeod (1992) at man ikke kan separere forestillinger helt fra det affektive domene. Det McLeod (1992) kaller det affektive domene består av tre kategorier, bestående av forestillinger (beliefs), holdninger (attitudes) og følelser (emotions). McLeod argumenterer for at det kan være vanskelig å skille begrepene fra hverandre, men at de varierer i stabilitet og intensitet. Forestillinger og holdninger er forholdsvis stabile, mens følelser fluktuerer som illustrert i figur 1.



Figur 1: McLeods affektive spekter (Wedegge, Skott, Henningsen, & Wæge, 2006)

McLeod skriver videre at forestillinger i større grad enn følelser er kognitive av natur. Forestillinger bygges over en lengre tidsperiode og er relativt stabile. Følelser derimot er intense og i mindre grad begrunnet kognitivt. McLeod (1992) skriver at følelser kan komme og forsvinne raskt. Videre eksemplifiserer han dette gjennom frustrasjonen man føler når man plages med et mattestykke, og gleden man føler når man endelig løser oppgaven. For å tydeliggjøre betydningen av figur 1 beskriver Wedegge et al. (2006) venstresiden som den stabile eller kognitive siden. Høyresiden er den affektive eller intense siden. Altså jo større grad av affekt eller intensitet i responsen, desto mindre grad av stabilitet og kognitiv vurdering.

2.3 Strukturering av forestillinger

Op't Eynde et al. (2002) påpeker at analysen av forestillingers natur får fram strukturelle egenskaper ved forestillinger. Flere forskere har presentert forslag til hvordan forestillinger kan struktureres (Op't Eynde et al., 2002). I denne oppgaven har jeg valgt benytte forestillingsstrukturen til McLeod (1992). McLeod strukturerer elevens affektive respons i sammenheng med forestillinger til matematikk på følgende måte.

Forestillingskategori	Eksempel på forestilling
Forestillinger om matematikk	Matematikk er nyttig
Forestillinger om selvet	Jeg synes matematikk er vanskelig
Forestillinger om matematikkundervisning	Læreren underviser på tavlen
Forestillinger om den sosiale konteksten	Drøfting av oppgaven er nyttig

Figur 2: Kategorisering av forestillinger (McLeod, 1992)

Modellen til McLeod (1992) deler forestillinger i fire kategorier. Det er forestillinger om matematikk, forestillinger om selvet, forestillinger om matematikkundervisning og forestillinger om den sosiale konteksten. Den første kategorien, forestillinger om matematikk omfatter elevens generelle syn på matematikk. Forestillinger om selvet omhandler elevens syn på egne prestasjoner og hva som skal til for å lykkes i matematikk. Forestillinger om matematikkundervisning er knyttet til hvordan matematikkundervisningen skal foregå, lærerens rolle og hvilke oppgaver og arbeidsmetoder som benyttes. Forestillinger om den sosiale konteksten beskriver elevens forestillinger knyttet til de sosiale normene, og de sosiomatematiske normene i klasserommet. Ifølge Cialdini og Goldstein (2004) er det to forskjellige tilnærminger til sosiale normer. Du har beskrivende normer (descriptive norms) og forventingsnormer (injunctive norms). De beskrivende normene beskriver hva som er et vanlig handlingsmønster. I en gruppe er beskrivende normer derfor normgivende for hvordan man skal handle i ulike situasjoner. Forventningsnormene forteller om hva som normalt er godkjent eller ikke godkjent atferd (Cialdini & Goldstein, 2004). De sosiale normene etableres og utvikles gjennom en gjensidig forhandling mellom læreren og elevene (Gravemeijer & Cobb, 2006). Sosiomatematiske normer vil til forskjell fra de mer overordnede sosiale normene i klasserommet bare være gjeldende i en matematisk læringskontekst (Yackel & Cobb, 1996), og eksempler på slike normer er hvilke kriterier, eller forventninger som stilles til et matematisk resonnement i klasserommet. Den sosiomatematiske normen kan være at elevene i tillegg til selve svaret, også må utrede hvordan de kom fram til svaret. Yackel og Cobb (1996) understreker viktigheten av sosiomatematiske normer gjennom hvordan normene påvirker de matematiske samtaler og argumentene, og derfor også læringsmulighetene til både elevene og læreren.

Det finnes lite forskning og modeller knyttet til elevenes forestillinger om programmering i matematikk. I denne studien tilpasser jeg McLeod (1992) sin modell, for å strukturere elevenes forestillinger om programmering i matematikk. Eksemplene i figur 3 er hentet fra intervjuene i studien min.

Forestillingskategori	Eksempler på forestillinger
Forestilling om programmering i matematikk	Du gir instruksjoner til en robot. Programmering handler om å prøve seg fram. Programmering hører til naturfag og matte
Forestillinger om programmering i matematikkundervisningen	Læreren lager en labyrint som elevene skal navigere seg gjennom. Læreren setter mål for timen
Forestillinger om selvet	Jeg får til å programmere når jeg samarbeider med noen. Jeg føler jeg lykkes når jeg forstår hva maskinen vil jeg skal gjøre.
Forestillinger om den sosiale konteksten	Vi jobber som regel i grupper, også diskuterer vi hvordan vi skal programmere for å løse oppgaven. Det kan hjelpe å ha flere ideer til problemløsninger, man får gjort ting raskere når man samarbeider Noen grupper er verre enn andre.

Figur 3: Kategorisering av forestillinger om programmering i matematikk.

Modellen bygger på modellen til McLeod (1992), men jeg har endret noe på rekkefølgen. Jeg har samlet forestillinger om selvet og forestillinger om den sosiale konteksten nederst i modellen. Dette begrunner jeg med at jeg gjennom analysen observerte at mange av forestillingene om programmering i matematikk var basert på forestillinger om programmering i matematikkundervisning. Samtidig var også flere av elevbesvarelsene som fortalte noe om forestillinger om selvet begrunnet ut ifra forestillinger om den sosiale konteksten.

3 Programmering

Programmering er i dag et allsidig verktøy, og derfor et vidt begrep som kan brukes i mange forskjellige sammenhenger (Bueie, 2019). Stenseth, Kaufmann, og Forsström (2019) definerer programmer som arbeid med å utvikle og implementere instruksjoner til dataprogrammer. Bueie (2019) sammenligner et dataprogram med instruksjonene i en kokebok. Begge er en samling detaljerte instruksjoner som skal oppnå et bestemt resultat. Den eneste forskjellen er at det nå må være på et språk som datamaskinen forstår. Disse dataprogrammene gjør datamaskinen i stand til å løse forskjellige oppgaver, løse problemer og støtte menneskelig interaksjoner. I denne oppgaven vil jeg i likhet med Bueie (2019) fokusere på programmering i en matematikkontekst.

3.1 Programmering i skolen

Debatten om teknologi og programmering i skolesammenheng er ikke ny. Drijvers et al. (2009) påpeker at forsøk på programmering i skolen har vært prøvd ut flere ganger siden sekstitallet. Det er trolig flere grunner til at disse forsøkene ikke gav de ønskede resultatene, f.eks. begrensninger knyttet til teknologi, tilgang på teknologi og programvare eller kompetansen til lærerne som skulle undervise. Etter Wing (2006) sin artikkel om viktigheten av det hun kaller «Computational thinking» opplevde programmeringsdebatten igjen en renessanse (Bueie, 2019). I forbindelse med LK20 og særlig den nye læreplanen i matematikk, har programmering vært mye diskutert i Norge. Tema har vært hvorvidt programmering skal få sin egen plass/fag i skolen, eller om det heller burde integreres inn i læreplanen til allerede eksisterende fag (Kaufmann & Stenseth, 2020). Kaufmann og Stenseth (2020) påpeker at hovedfokuset har vært på sammenhengen mellom programmering og matematikk, spesielt på om programmering kan tilføre matematikkundervisning noe unikt.

Bueie (2019) mener at det er naturlig at matematikkfaget får hovedansvaret for programmering, og han viser til at programmering og matematikk på mange måter er svært sammenflettet. Debatten rundt LK20 i matematikk har vært preget av en blanding av teknologioptimister og teknologipessimister (Gjøvik & Torkildsen, 2019). Det er likevel i dagens samfunn stort sett bred enighet om at elever skal undervises i forskjellige digitale ferdigheter. Disse ferdighetene skal kunne gi elevene kompetanse innenfor en rekke kjerneprinsipper innenfor programmering. Hensikten er at elevene skal kunne bli medborgere

i den stadig mer digitale og teknologiske hverdagen. Kaufmann og Stenseth (2020) hevder at flere av dagens elever kan komme til å spille en viktig rolle i å drive den teknologiske utviklingen framover. Derfor vil programmeringsferdigheter være viktige for elevene hvis de skal kunne ta del i samfunnet etter endt skolegang. Bocconi, Chiocciariello, og Earp (2018) påpeker at programmering og computational thinking er både framtidsrettet og nyskapende. Forhåpentligvis vil dette kunne skape økt interesse og motivasjon blant elevene, særlig hos elever som ellers ville hatt liten interesse for matematikk eller andre realfag. Forhåpentligvis kan programmering oppleves som en givende aktivitet, som samtidig skaper et behov og en reell mulighet til å benytte og utvikle forståelse og kompetanse i matematikk.

Heid, Thomas, og Zbiek (2013) poengterer at innføring av programmering ikke automatisk skaper en positiv læringseffekt. De fremhever betydningen av at implementeringen av ny teknologi i skolen må vurderes og sees i sammenheng med læreplanen. Læreren må ha det Heid et al. (2013) kaller for PTK eller pedagogisk teknologi kunnskap. Også Drijvers et al. (2009) konkluderer med at vi kommer til å trenge lærere med både programmeringskompetanse og faglig kompetanse. Det overordnede målet ved å innføre programmering i skolen må da selvsagt være at det skal tilføre elevene en kompetanse som de vanskelig kunne fått på noen annen måte.

*«Barn blir ikke automatisk digitalt kompetente av å få en iPad, på samme måte som voksne ikke blir mekanikere av å kjøre en bil. Det er hva vi gjør med det digitale mediet, som har noe å si.»
(Haraldsrud, Sveinsson, & Løvold, 2020, p. 15).*

Haraldsrud et al. (2020) påpeker at begrepet dybdelæring står sentralt i LK20. Videre at formålet er at dybdelæring skal gi elevene varig forståelse og dybdekunnskap. Dybdelæring forutsetter at opplæringen gir anledning til utforskning og undring, samt rom for å være kreativ og nysgjerrig (Utdanningsdirektoratet, 2018). Programmering vil ifølge Haraldsrud et al. (2020) være et godt verktøy for å drive med dybdelæring. Programmering gir mulighet til å utforske og arbeide med kjente konsepter på nye måter. Dette gjelder ikke bare i matematikk, programmering vil også gi elevene problemløsningsferdigheter og metoder som kan benyttes tverrfaglig (Haraldsrud et al., 2020).

3.2 Programmering i matematikk

I læreplanen i matematikk i LK20 finner vi flere punkter som vi kan koble sammen med programmering. Under fagets relevans, kjerneelementer og grunnleggende ferdigheter finner vi enten indirekte eller direkte henvisninger til bruk av teknologi, programmering eller utvikling av programmeringsferdigheter.

Ser vi på kompetansemålene, finner vi allerede etter andre trinn mål som kan knyttes til algoritmisk tenkning (Utdanningsdirektoratet, 2020). Etter fjerde trinn skal kunne lage algoritmer selv, og uttrykke dem ved bruk av variabler, vilkår og løkker (Utdanningsdirektoratet, 2020). Bueie (2019) beskriver en algoritme som en liste med instruksjoner som er konstruert for å finne et svar på et problem. Senere skal elevene bli fortrolige med å arbeide med variabler, løkker og vilkår. Alt dette før vi kommer til de kompetansemålene som eksplisitt nevner programmering. I denne undersøkelsen har jeg intervjuet elever på sjette trinn. Ifølge læreplanen skal elever i løpet av sjette trinn «bruke variabler, løkker, vilkår og funksjoner i programmering til å utforske geometriske figurer og mønstre».

Et av kjerneelementene i læreplanen for matematikk er utforskning og problemløsning. Her står det at elevene skal lete etter mønster, finne sammenhenger og diskutere seg fram til en felles forståelse. Det legges større vekt på framgangsmåte og strategiene enn selve svaret på oppgaven. Videre står det at problemløsning i matematikk handler om at elevene selv utvikler en metode for å løse et problem. Derfor vil algoritmisk tenkning være et godt verktøy i denne prosessen for å utvikle strategier og framgangsmåter.

4 Algoritmisk tenkning

Bueie (2019) fastslår at artikkelen til Wing (2006) har vært et viktig referansepunkt for diskusjonen rundt programmerings rolle og plass i skolematematikken (Bueie, 2019). Hovedbudskapet i artikkelen til Wing er nytten og viktigheten av evnen til computational thinking. Wings computational thinking blir et paraplybegrep som omfatter problemløsning, abstraksjon, algoritmisk tenkning, modellering og dekomponering (Bueie, 2019). Et av poengene til Bueie er at alle disse egenskapene som Wing framhever, er begreper som man enkelt kan relatere til matematikk (Bueie, 2019).

I denne oppgaven vil jeg basere meg på Utdanningsdirektoratets definisjon, som er at algoritmisk tenkning er den norske oversettelsen av computational thinking. Algoritmisk tenkning blir da på samme måte som computational thinking et paraplybegrep for egenskaper relatert til datamaskiner (Gjøvik & Torkildsen, 2019). Siden algorithmic thinking (algoritmisk tenkning) inkluderes inn under computational thinking, velger Gjøvik og Thorkildsen å oversette algorithmic thinking til algoritmebehandling. Det finnes flere forskjellige definisjoner av begrepet algoritmisk tenkning. Selv om definisjonene varierer noe, er det stort sett enighet blant forskere om de sentrale begrepene. Gjøvik og Torkildsen (2019) trekker fram fem begreper som Bocconi et al. (2018) mener er definerende for algoritmisk tenkning. Begrepene er abstraksjon, algoritmebehandling, generalisering, automatisering og dekomponering. Det er flere likheter mellom paraplybegrepene til Bocconi et al. (2018) og Wing (2006). Abstraksjon, algoritmebehandling/algoritmisk tenkning og dekomponering finner vi hos begge. Under vil jeg beskrive begrepene til Bocconi et al. (2018) litt nærmere.

Abstraksjon er evnen til å trekke ut essensen av flere eksempler eller tilfeller, samtidig som man overser opplysninger som ikke er relevante. *Algoritmebehandling* innebærer det å kunne forstå, følge, forklare og utarbeide trinnvise instruksjoner for problemløsning. *Generalisering* er å se sammenhenger og mønstre i oppgavene. Slik at du kan utarbeide generelle allmenne løsninger og/eller metoder, som kan løse lignende problemer. *Automatisering* betyr å løse problemer gjennom programmering med minimalt behov for menneskelig administrasjon eller helautomatisk. *Dekomponering* er evnen til å bryte en oppgave opp i mindre deloppgaver, slik at man trinnvis kan løse delproblemer som igjen løser hovedproblemet (Gjøvik & Torkildsen, 2019).

Gjøvik og Torkildsen (2019) hevder at programmering på mange måter er det naturlige miljøet for å benytte algoritmisk tenkning. På den annen side skriver Kaufmann og Stenseth (2020) at prøve- og feilemetoden som så ofte knyttes til algoritmisk tenking når elever programmerer, faktisk kan ha et negativ innvirkning på elevens matematiske argument. En moderne datamaskin vil på sekundet kunne simulere resultatet av gjetningen deres. Dermed får elevene øyeblikkelig respons, og dette gir rom for at elevene gang på gang forsøker med forskjellige variabler. Noe som resulterer i at elevene bruker minimalt med tid til å reflektere over resultatet, for så og utarbeid en ny hypotese.

5 Metode

Som nevnt innledningsvis ønsker jeg gjennom forskningsspørsmålet mitt å undersøke elevers forestillinger om programmering i matematikkfaget. Dette innebærer at jeg ønsker å tilegne meg en forståelse for elevers forestillinger. Pehkonen et al. (2003) påpeker at forestillinger ofte er kompliserte og ubevisste, noe som gjør arbeidet med å kartlegge dem krevende. Også Lester (2002) understreker denne utfordringen, kun gjennom innsikt i hvordan mennesker handler og tenker er det mulig å danne seg et bilde av forestillingene deres. Spangler (1992) anbefaler å benytte uformelle samtaler omkring temaet for å få innsikt i forestillinger. Ved bruk av åpne spørsmål vil noen av forestillingene eleven har om programmering i matematikk komme til syne. Nedenfor vil jeg redegjøre for mitt metodevalg og utvalget jeg undersøker.

5.1 Valg av metode

Hvordan samler man på best mulig måte data om elevers forestillinger? Skal jeg benytte en kvantitativ metode, en kvalitativ metode eller en kombinasjon hvis jeg ønsker å lære om elevers forestillinger om programmering i matematikk? Enkelte forskere vil muligens påpeke at en kvantitativ metode vil ha problemer med å gjenspeile virkelighetens kompleksitet på en hensiktsmessig måte (Postholm, Jacobsen, & Søbstad, 2018). I følge Postholm et al. (2018) er kvalitative metoder godt egnet til å innhente data først og fremst i form av ord. Det vil si beskrivende ord, som gir forståelse for menneskers handlinger og meningsskaping. Kvale og Brinkmann (2015) poengterer at kvalitativ forskning først og fremst er ute etter innsikt fra informantens perspektiv. Hvordan oppleves egentlig verden for informanten? En kvalitativ metode går målrettet inn for å beskrive og forstå forskningsobjektets subjektive meninger og handlingsmønstre (Postholm et al., 2018).

På bakgrunn av dette velger jeg å benytte en kvalitativ metode. Formålet er å få et solid datagrunnlag, som jeg kan analysere og tolke for å danne meg en forståelse for elevenes forestillinger om programmering i matematikk.

5.2 Utvalg av informanter

Opprinnelig var planen for min masteroppgave å undersøke hvilken forståelse lærerne har om programmering i matematikk. Gjennom tidligere samtaler med lærere hadde jeg dannet meg

et inntrykk av at ikke alle følte seg kompetente til å undervise om programmering i matematikk. Det virket også som det var mye usikkerhet knyttet til hva programmering i skolen og matematikken egentlig innebar. Disse samtalene syntes jeg var svært interessante, og jeg ønsket derfor først å fremst å finne ut mer om lærernes tanker om temaet. Jeg gikk derfor i gang med å forsøke å finne matematikklærere som ville være informanter til prosjektet mitt. Hverdagen som lærer er hektisk nok alene, samtidig hjalp det nok heller ikke at vi er midt oppi covid-19 pandemien. Jeg startet å lete etter informanter medio desember 2020. Da jeg mot slutten av januar fortsatt var uten noen informanter, tok jeg kontakt med Nordnorsk vitensenter. Gjennom Nordnorsk vitensenter fikk jeg kontaktinformasjon til alle skolene som har deltatt på super:bit prosjektet. Lærerne jeg snakket med, følte seg ikke kompetente nok til å stille til noe intervju. Andre påpekte at de hadde søkt til kommunen om kursing. Resultatet ble til slutt at jeg ikke klarte å skaffe meg tilstrekkelig med informanter i tide for å danne et grunnlag for undersøkelsen min. Jeg valgte derfor å endre fokuset fra lærerne til elevene og deres forestillinger om programmering i matematikk.

Min søken etter elever som informanter var fortsatt basert på samarbeidet med Nordnorsk vitensenterets super:bit prosjekt. Super:bit er et samarbeid mellom tre aktører, Vitensentrene, lære kidsa koding og NRK. Formålet med prosjektet er å skape interesse for teknologi, programmering og algoritmisk tenkning. Alle sjetteklasser kan delta, enten på vitensenteret eller så gjennomføres det på skolen. Prosjektet består av et forarbeid, selve «super:bit oppdraget», etterarbeid og diverse bonusoppgaver. Skolene som har deltatt på prosjektet får også beholde utstyret. Utstyrsesken består blant annet av bilen bit:bot, som kan programmeres gjennom bruk av en mikrokontroller som kalles micro:bit. Dette samarbeidet mellom vitensenteret og skolene vil også garantere at elevene jeg intervjuer faktisk har et forhold til programmering. Jeg ser derfor på elever i disse klassene som interessante og relevante informanter. Derfor er det eneste kriteriet for utvalget mitt at klassen har vært en del av super:bit prosjektet.

Den 19. februar fikk jeg endelig kontakt med en klasse som hadde deltatt på super:bit prosjektet, og som ville la seg intervju. Skolen jeg samarbeidet med bekreftet dette forholdet og satt meg i kontakt med en lærer. Denne læreren hadde gjennomført kurs relatert til super:bit prosjektet. Samtidig har også klassen hatt besøk av vitensenteret på skolen, og læreren har også programmert med klassen sin etter dette. For å få innsikt i hvilke forestillinger elevene i klassen har om programmering i matematikkfaget, fikk jeg læreren til å sette sammen en gruppe med elever. Gruppen skulle gjenspeile mangfoldet i klassen, både

med tanke på kjønn, men også ut ifra lærerens meninger om elevenes kjennskap og kompetanse om programmering. Det ble sendt ut samtykkeskjemaer, og til slutt hadde jeg en gruppe på åtte elever som samtykket til å stille til intervju. Kjønnfordelingen i utvalget var slik, fem jenter og tre gutter. Forskning på kjønn i sammenheng med bruk av programmering i skolen er tvetydig. Dohn (2020) konkluderer med at det ikke er noen forskjell mellom gutter og jenters interesse for koding i Scratch. Kong, Chiu, og Lai (2018) hevder derimot at gutter har større interesse for programmering på grunn av stereotyper fra popkulturen. For eksempel medias portrettering av den mannlige dataeksperten, som de mener blir en rollemodell for guttene, og en «anti-rollemodell» for jentene. Jeg tar denne forskningen til etterretning, og var spent på om jeg ville se tendenser til den ene eller den andre teorien.

5.3 Intervju som metode

Den kvalitative metoden jeg har valgt å benytte for å samle dette datagrunnlaget er intervju. Et kvalitativt forskningsintervju har som hensikt å forstå verden fra intervjuobjektets perspektiv (Kvale & Brinkmann, 2015). Kvale og Brinkmann beskriver intervju som en aktivitet hvor en gjennom prosessen produserer kunnskap. Det er ifølge dem gjennom samspillet mellom intervjuer og intervjuobjekt kunnskapen blir produsert. På samme måte som det finnes mange former for samtaler, finnes det også mange forskjellige former for intervjuer Kvale og Brinkmann (2015). Formålet med et forskningsintervju er at det skal gå dypere i sakens kjerne.

Min kvalitative datainnsamlingsstrategi er semistrukturert intervju. Et semistrukturert intervju er godt egnet når en ønsker å få innsikt i intervjuobjektets tanker om et emnet sett fra intervjuobjektets perspektiv (Kvale & Brinkmann, 2015). Det semistrukturerte intervjuet kan minne mer om en samtale, men det har som andre intervju et tydelig formål. Gjennom intervjuet ønsker man å få mest mulig innsikt omkring informantens meninger og tanker. Strukturen på intervjuet er ikke så rigid som i det strukturerte intervjuet, men heller ikke så fritt som i det ustrukturerte. I det semistrukturerte intervjuet har forskeren tema og forslag til spørsmål gjerne i form av en intervjuguide (Kvale & Brinkmann, 2015; Postholm et al., 2018). Jeg har valgt akkurat denne formen for intervju fordi jeg føler den i størst grad vil kunne hjelpe meg til å samle inn nyttig data fra mine informanter. Dette fordi jeg trenger innsikt i elevenes subjektive forestilling om temaet. Etersom jeg skal gjennomføre intervjuer med elever fra en sjette klasse, ser jeg for meg at det kan være en fordel at jeg er fleksibel.

Med det mener jeg at man i semistrukturerte intervjuer kan omformulere, sufflere eller komme med oppfølgingsspørsmål eller grave dypere hvis det opprinnelige spørsmålet ikke gir meg et tilfredsstillende svar. Spangler (1992)) understreker at en av de beste måtene å få innsikt i forestillinger, er gjennom åpne dialoger omkring emnet. Det er akkurat det som er målet mitt med intervjuene.

Intervjuet tar form som en mer eller mindre målrettet samtale omkring et bestemt tema. Det blir opp til intervjuer å stille spørsmålene der det faller seg naturlig. Rekkefølgen på spørsmålene er dermed ikke av betydning. Du som intervjuer står fritt til å kunne stille nye spørsmål eller be intervjuobjektet utdype besvarelsene sine om det skulle være behov. Intervjuformen åpner samtidig også for at intervjuobjektet kan være med på å forme intervjuet, det kan være temaer som forsker rett og slett ikke har vurdert som relevant eller hatt kunnskap om (Postholm et al., 2018). Dette er kanskje særlig relevant i forhold til mitt forskningsspørsmål, siden jeg ser på forestillinger. Det er for eksempel svært vanskelig for meg som forsker å forutse hvilke forestillinger elevene kan inneha omkring temaet programmering. Derfor mener jeg denne økte fleksibiliteten vil kunne være et særdeles godt hjelpemiddel i selve intervjuprosessen.

5.3.1 Ethiske aspekter

Gjennomføring av forskning på skoleelever innebærer et stort ansvar. Forskning på skoleelever må derfor ifølge NSD (2021b) sees i sammenheng med at barn er en sårbar gruppe. Det kan ikke forventes at barn skal kunne ivareta sine egne interesser. De kan heller ikke samtykke til deltakelse i forskningsprosjekter på vegne av seg selv (NSD, 2021a). Kvale og Brinkmann (2015) vurderer det slikt at etiske problemstillinger preger hele forløpet i en intervjuundersøkelse. Som forsker må man derfor ta hensyn til mulige etiske utfordringer gjennom hele prosessen. Det vil si fra intervjuet, gjennom transkriberingen og til analysen av intervjudataen. Kvale og Brinkmann (2015) beskriver etiske retningslinjer som de mener forskeren må ta stilling til før et forskningsintervju. Jeg vurderer det slikt at i sammenheng med min undersøkelse er det særlig *informert samtykke*, *fortrolighet* og *konsekvens* jeg må vektlegge. I samarbeid med klassens kontaktlærer ble det derfor i forkant av intervjuene sendt

ut informasjonsskriv og samtykkeskjema til elevene og deres foresatte¹. Her informerte jeg om formålet med forskningsprosjektet og hovedtrekkene i forskningsdesignet mitt. Selv om elevene formelt sett ikke kan samtykke på vegne av seg selv, gav jeg også eleven mulighet til å signere på samtykkeskjemaet. Dette for å forsikre meg om at elevene deltar frivillig.

Intervjudataene har gjennom hele prosessen blitt behandlet med fortrolighet. Det vil si jeg har fulgt retningslinjene for databehandling fra NSD – Norsk senter for forskningsdata, med det formål å verne om deltakernes privatliv². Deltakerne var informert om at ingen informasjon som kunne identifisere dem skulle publiseres (Kvale & Brinkmann, 2015). Det var kun jeg som skulle ha tilgang til opptakene fra intervjuene, og lydopptakene ble slettet underveis i transkriberingen. Gjennom oppgaven har jeg vært bevisst på at jeg aldri nevner navnet på skolen og at elevene anonymiseres gjennom oppdiktete navn.

5.3.2 Intervjuguide

Jeg ser for meg at undersøkelser av elevers forestillinger kan by på utfordringer. Eksempelvis er elevene kanskje selv ikke særlige bevisste sine egne forestillinger, og dermed blir det vanskelig for eleven å videreformidle dem. Elevene kan også intensjonelt være tilbakeholden, eller på andre måter unnvike fra å fortelle om sine forestillinger. Det kan være flere grunner til at en elev opptrer slik. Eksempelvis kan eleven ønske å framstille seg selv på en spesiell måte. Det kan være eleven forsøker å blidgjøre forskeren med å svare med det eleven tror forskeren vil høre. Eller så kan eleven være redd for at noe den sier skal bli brukt mot dem eller på andre måter skal få konsekvenser for skolegangen deres. Jeg mener likevel at kvalitative forskningsintervju er den forskningsmetoden som på best mulig måte vil kunne hjelpe meg å samle det nødvendige datagrunnlaget, for å på en tilfredsstillende måte kunne besvare mitt forskningsspørsmål.

Intervjuet mitt har et konkret og spesifikt formål, i denne sammenhengen et forskningsspørsmål som jeg ønsker å få besvart. Intervjuguiden er utarbeidet med ønsket om å få kvalitative beskrivelser av elevens tanker, meninger og opplevelser rundt programmering i matematikk. Forestillinger er ofte ubevisste eller skjulte, derfor stiller jeg ikke direkte

¹ Se vedlegg 2: Informasjonsskriv og samtykkeskjema

² Se vedlegg 1: Meldeskjema for behandling av personopplysninger

spørsmål om elevens forestillinger (Lester, 2002; Pehkonen et al., 2003). Jeg ønsker å stille spørsmål på en måte som kan gi meg et datagrunnlag som indirekte kan fortelle noe om elevens forestillinger. Kvale og Brinkmann (2015) påpeker at intervju spørsmålene bør være korte og enkle. Under vil jeg presentere spørsmålene fra min intervjuguide, sammen med en begrunnelse for noen av spørsmålene³.

1. Hva kjennetegner en skoletime med programmering?
2. Kan du beskrive en time dere har hatt der dere har programmert?
3. Synes du programmering hører til et bestemt fag på skolen?
4. Hva tenker du programmering handler om? Hva ønsker man å oppnå gjennom programmering?
5. Hva vil det si å være god i programmering?
6. Hvilke egenskaper kjennetegner en elev som er flink å programmere?
7. Når synes du at du lykkes i å programmering?
8. Hvilke typer programmeringsoppgaver liker du å arbeide med? Programmering av «bil», lage blokkprogrammer for å løse oppgaver?
9. Når dere programmerer, hvordan organiserer læreren timen og klassen? Individuelt, grupper eller plenum
10. Hvis du kunne bestemme over en time der klassen skulle jobbe med programmering hva skulle dere gjøre?
11. Skulle du ønske dere brukte mer tid på programmering i klassen?
12. Ser du noen unike muligheter ved bruk av programmering i matematikkfaget?
13. Ser du noen likheter mellom å løse oppgaver i programmering og i matematikk?
14. Synes du skolen har tilstrekkelig utstyr for å drive med programmering? Er det noe du savner?

Det første og det andre spørsmålet har jeg i intervjuguiden min kalt overgangsspørsmål, og er det Kvale og Brinkmann (2015) kaller for åpnings spørsmål. De påpeker at spørsmålet som innleder intervjuet gjerne kan omhandle en spesifikk situasjon, i dette tilfellet en undervisningssituasjon i sammenheng med programmering. Håpet er at slike spørsmål skal få intervjuobjektet til å fortelle det de i hovedsak forbinder med programmering i skolen.

I spørsmål tre ønsker jeg å finne ut om elevene ser noen sammenhenger mellom det de gjør i programmeringstimene og andre fag i skolen (gjerne matematikk). Spørsmålet er inspirert av et spørsmål fra Spangler (1992). Tanken bak spørsmålet er at eleven skal resonere rundt

³ Se vedlegg 3: Intervjuguide

programmering og sammenligne prosessen med andre skolefag. Spangler påpeker at forskjellige elever har komplett forskjellige oppfatninger ofte på grunn av deres bakgrunn. Han eksemplifiserer dette med at elever uten kompetanse i musikk, vil si at matematikk og musikk ikke har noen sammenheng. Derimot vil en elev med musikkompetanse fortelle at det faktisk er svært mye matematikk involvert i musikk. Når jeg ber eleven beskrive en annen elev som er flink å programmere, er dette også inspirert av Spangler (1992). Formålet med spørsmålet er å få innsikt i hva eleven tenker kjennetegner en elev som mestrer programmering.

Gjennom spørsmål fem, seks og syv ønsker jeg å finne ut hva eleven legger i å lykkes i programmering. Når og hvorfor føler de at de lykkes, og hva tenker de andre som lykkes i programmering gjør. Svarene til elevene håper jeg vil gi meg et innblikk i elevens forestillinger om selvet.

De siste spørsmålene brukes nesten ikke i analysen. Så på grunnlag av plassmangel, og lite relevans for analysen velger jeg ikke å utdype dem.

5.3.3 Gjennomføring av intervju

Spangler (1992) hevder at den beste måten å få innsikt i elevers forestillinger er gjennom uformelle samtaler om temaet. Med dette i bakhodet ønsket jeg å skape en uformell intervjusituasjon. Jeg hadde som formål å gjøre intervjusituasjonen så avslappet som mulig. Eleven fikk derfor selv velge hvor vi skulle sitte under intervjuet. Jeg valgte bevisst å bruke starten av intervjuene til igjen å informere elevene om rettighetene deres, og hva hensikten med forskningsprosjektet mitt var. Jeg forklarte videre hva som var formålet med intervjuet, og at det var i sammenheng med min masteroppgave. Videre at det kun var jeg som skulle behandle datamaterialet, og at jeg skulle slette lydfilene så snart de var transkribert. Eventuelle personopplysninger skulle jeg anonymisere, og jeg forklarte hva det innebar. Elevene ble igjen informert om at deltakelsen var frivillig. De hadde rett til når som helst å trekke seg fra undersøkelsen, selv etter intervjuet. Jeg understreket også at svarene deres på ingen måter skulle kunne få noen konsekvenser for deres relasjon til læreren eller deres videre skolegang. Jeg poengterte også at jeg ønsket at de skulle svare så ærlig som mulig på spørsmålene. Med dette håpet jeg at de skulle fortelle sine egne tanker, og ikke hva de trodde jeg ville høre.

Kvale og Brinkmann (2015) skriver om bruken av forskjellige typer intervju spørsmål. Jeg bet meg særlig merke i bruken av taushet og oppfølgingsspørsmål under intervjuet. Fra praksis har jeg erfart at det kan være gunstig å la eleven tenke litt på spørsmål du stiller klassen. Det er fristende å velge den første hånden som rekkes opp for å bryte den «ubehagelige stillheten». Formålet med bruk av taushet er å gi intervjuobjektet tid til å reflektere over svaret sitt og spørsmålet, for så selv å bryte tausheten med eventuell tilleggsinformasjon.

Oppfølgingsspørsmål stilles av intervjueren med det formål å få informanten til å utdype eller fortsette tankerekken sin. Kvale og Brinkmann (2015) skriver at en liten bekreftelse, eller bare taushet kan vise at du ønsker at informanten skal fortsette fortellingen. Sentrale temaer kan du understreke ved gjentakelse, og på denne måten også invitere til videre utredelse. Under intervjuene ønsket jeg å forsøke å benytte meg av særlig disse to teknikkene. Jeg håper dette vil oppfordre elevene til å dele sine tanker, meninger og erfaringer med meg, noe som indirekte kunne gi meg innsikt i deres forestillinger.

I ettertid ser jeg at jeg har et forbedringspotensial når det kommer til hele intervju prosessen. Som Kvale og Brinkmann (2015) understreker, så forbedres intervjuferdighetene gjennom intervju praksis. I begynnelsen av intervjuet fortalte jeg elevene at jeg arbeider med en masteroppgave i matematikdidaktikk. Dette bet kanskje enkelte elever seg merke i, noe som kan ha påvirket svarene deres. Jeg blir å drøfte dette i kapittelet om validitet og relabilitet. Jeg har også en uheldig formulering i spørsmål seks, hvor jeg ber eleven beskrive egenskapene til en elev som er *flink*. Spangler (1992) kommenterer sitt spørsmål der han ber eleven beskrive noen som har talent for matematikk. Han forklarer at slike spørsmålet bør stilles med litt finesse for å unngå å såre noens følelser. En bedre alternativ formulering kunne vært: *Hvilke egenskaper kjennetegner en elev som lykkes med å programmere?*

6 Analyse

Jeg gjennomførte totalt åtte intervjuer i klassen og etter intervjuene behandlet jeg datamaterialet. Intervjuene ble transkriberte som en forberedelse til analysearbeidet. Datagrunnlaget for kvalitative studier er ofte omfattende, og forskeren må derfor få oversikt over materialet. Formålet med å analysere den kvalitative dataen er for det første å få organisert datamaterialet som har blitt samlet (Postholm et al., 2018). Gjennom sorteringen av datamaterialet skal dataen gjøres forståelig. Postholm skriver videre at det ofte handler om å lete etter mønstre i dataene slik at disse kan kategoriseres. Gjennom dette arbeidet ble jeg svært godt kjent med elevene og besvarelsene deres. Jeg har gjort den vurderingen at jeg presenterer syv av de åtte intervjuene. I det andre intervjuet jeg gjennomførte fikk jeg ikke en dialog gående med eleven. Intervjuet tilfører derfor ikke undersøkelsen min noe som kan hjelpe meg å forstå elevens forestillinger. Under skal jeg presentere et lite utvalg av utsagn fra intervjuene. Utsagnene jeg har valgt er utsagn som jeg tenker kan gi innsikt i elevens forestillinger om programmering i matematikk.

Jeg har under analysearbeidet forsøkt å få innsikt i elevenes forestillinger knyttet til de fire kategoriene fra McLeod (1992) sin modell om strukturering av forestillinger. Forestillinger om programmering i matematikk, forestillinger om programmering i matematikkundervisningen, forestillinger om selvet og forestillinger om den sosiale konteksten. I tillegg har jeg benyttet meg av Bocconi et al. (2018) og Wing (2006) for å analysere og tolke uttalelsene knyttet til programmering i matematikk. Hva forteller eleven om programmering og hva inngår i begrepene.

6.1 Intervju – Pia

Jeg ber Pia beskrive en typisk skoletime med programmering.

S: Kan du huske en skoletime der dere har programmert? Kan du beskrive timen?

P: Ja, en gang så programmerte vi en sånn liten [...] liten «lego» greie som vi kunne styre på pc'n så kunne den kjøre rundt. Vi satt inn blokker sånn at den enten kunne kjøre bakover ha lys og sånt. Ja så kjørte vi rundt med den.

[...]

S: Hvordan velger du hvordan bilen kjører?

P: Ja, med de blokkene. Kjøre framover, kjøre bakover, kjør til siden, kjør fort for eksempel.. tror jeg..

Pia forteller om en time der de programmerer en bil gjennom å sette inn blokker (med instruksjoner) som forteller bilen hva den skal gjøre.

S: Hvordan organiserer læreren timen og klassen?

P: Vi går i enten toer eller treer grupper, så går vi rundt her for eksempel her eller i første etasje eller så bare går vi rundt, også får vi da en pc og så får vi den «lego» greien også får vi lov å programmere og sette inn blokker som vi vil. Og noen har som mål om å få til det, noen lager hinderløyper til bilen som den må klare å kjøre gjennom.

Pia forteller at de samarbeider i grupper i timene med programmering. Hun forteller at elevene får gå rundt på skolen og velger selv hvor de skal programmere, og hvordan de vil programmere bilen.

Jeg ber Pia fortelle hva hun tenker programmering handler om.

S: Hva tenker du programmering handler om egentlig?

P: Ehh.. jeg er litt usikker kanskje hmm for eksempel programmere noe til å gå framover bakover kanskje kjøre rundt eller Programmere en pc?

[...]

P: Kanskje man da... Kanskje man har programmert en pc til å slå seg av etter en time. Eller noe sånt?

Pia forteller at man kan programmere noe til å gjøre forskjellige ting. Hun beskriver ikke selve programmeringsprosessen, men forteller om ulike metoder hun ser for seg at man kan anvende programmering på, for eksempel hvordan en kan programmere datamaskinen til å slå seg av etter et forhåndsbestemt tidsintervall.

Når jeg spør Pia hva hun forbinder med å lykkes i å programmere svarer hun følgende:

S: Når synes du at du lykkes i å programmere?

P: Hmm.. kanskje når man får til noe som ikke andre får til, og man får det til..

Videre når jeg spør hva hun synes kjennetegner andre elever som lykkes svarer hun.

S: Tenk på noen elever i klassen eller deg selv som er flink til å programmere. Er det noen egenskaper de har tenker du?

P: Ja kanskje litt ja, jeg tror det at en gutt i klassen som har programmert hjemme

S: Okei, hvilke egenskaper vil du si den eleven har?

P: Mm jeg har sett mange som ser på han når vi for eksempel programmerer den lego-greia som kjører rundt. Så har han fått til noe de andre ikke har fått til, men jeg vet ikke hvilke egenskaper.

Pia forteller at å lykkes i programmering eller å være flink til å programmere handler om å få til noe andre ikke får til, eller som andre synes er vanskelig. Hun sier at hun ikke vet hvilke egenskaper som trengs for å lykkes i programmering.

Jeg spør Pia hva klassen skulle gjøre hvis hun fikk styre en skoletime med programmering

S: Hvis du skulle være sjef i klassen en time der dere programmerte hva skulle klassen gjøre da tenker du?

P: Da ville jeg hvis det finnes da hatt en sånn robot ting som ser ut som ett menneske, men det er en robot inni. Som man skulle programmere til å gå opp en trapp, og kanskje gjøre en «backflip» eller noe sånt. Ja, ville folk ha programmert den til å gjøre kule ting.

Pia forteller at hvis hun fikk styre en time med programmering ville de bruke en større robot, som ser ut som et menneske og etterligner menneskelige kvaliteter. Først sier hun at den skal gå opp en trapp, men legger raskt til at den skal ta «backflip» eller noe annet kult.

6.1.1 Oppsummering intervju med Pia

Jeg har presentert utsagn som jeg mener kan være til hjelp for å danne et bilde av hvilke forestillinger Pia har om programmering i matematikk. Det er ingen enkel oppgave, fordi jeg ikke klarer å få henne til å utdype mer av det hun sier. Analysen min viser at Pias beskrivelser av programmeringen er tett knyttet til det arbeidet de har gjennomført på skolen, og min tolkning er at hun forbinder programmeringstimene på skolen med super:bit prosjektet. Pia trekker fram bruken av programmeringsblokker med det formål å få noe til å bevege seg. Når Pia forteller om blokkene hun setter inn, tolker jeg det slik at hun beskriver hvordan hun utarbeider en algoritme for å løse oppgaven (Bocconi et al., 2018). Hennes beskrivelse av hvordan gruppen arbeider kan knyttes til dekomponering og algoritmebehandling (Bocconi et al., 2018). Pia beskriver ikke metoder for å løse problemer eller andre elementer ved algoritmisk tenkning. Når Pia forteller om sin drømmetime, beskriver hun hva de skal arbeide med og resultatet av programmeringen, å programmere en robot som kan utføre mer komplekse bevegelser enn bit:boten.

Min tolkning av intervjudataene er at Pia vektlegger prestasjoner mer enn læring og forståelse i programmering. Hun forteller at hun føler hun lykkes i programmering, når hun klarer noe andre ikke klarer. Hun bruker samme beskrivelse av en flink elev i programmering, at eleven får til ting som andre ikke får til.

Analysen gir oss et lite innblikk i Pias forestillinger om den sosiale konteksten knyttet til programmering i matematikk. Hun forteller at elevene arbeider i grupper på to eller tre elever, og at de ikke behøver å bli værende i klasserommet, men at de kan finne seg en plass på gangen for å programmere. Læreren gir dem stor frihet, både når det gjelder hvor de skal arbeide og hva de vil programmere.

6.2 Intervju – Markus

Markus beskriver en typisk programmeringstime på følgende måte.

S: Okei, når dere programmerer hvordan organiserer læreren timen og klassen?

M: Han (læreren) har en og en halv time eller en time der folk skal gå to og to eller flere og programmere noe sammen den gruppen. Han kan også dele grupper, også må

han lage en sånn vei for eksempel hvis det er en sånn bil eller så sier han bare det man skal gjøre. Så må gruppa gjøre det.

S: Så han starter med å gi dere en oppgave som dere skal løse?

M: Ja, så forklarer han hva vi skal gjøre, og hvordan vi skal gjøre det. Hvor på nettsiden vi skal gå.. [...]

S: Så han forklarer hvor dere finner verktøyene dere trenger for å løse oppgaven da?

M: Ja, akkurat.

Markus forteller at elevene noen ganger selv organiserer seg i grupper, andre ganger fordeler læreren dem. Læreren instruerer elevene i hvor og hvordan de skal navigere datamaskinen, for å finne verktøyene de trenger til arbeidet. Hvis de skal arbeide med bilen forteller Markus at det er læreren som lager en løype som bilen skal manøvrere.

Samtalen med Markus ga meg innsikt i hans tanker om programmering.

S: Hva tenker du programmering handler om?

M: Om du da for eksempel har en robot, så må du da programmere den til å gjøre det og det og det. Så kan du da bruke litt matte, det har litt matte. Du kan også bruke litt matte

S: Så du tenker at du har en oppgave som du skal løse, så må du programmere roboten til å løse den oppgaven.

M: Ja eller så kan man også programmere den til å ja gjøre det og det. Egentlig hva som helst.

Markus forteller at programmering handler om å få noe, gjerne en robot til å gjøre en bestemt handling. Han antyder også at bruk av matematikk kan være viktig når man programmer.

Jeg spør Markus hva det vil si å være god eller flink til å programmere.

S: Hva tenker du det vil si å være god eller flink til å programmere?

M: Ehh, å kunne programmere roboter, og være god i matematikk og ja.

S: Hvorfor tenker du at det å være flink i matematikk vil være til hjelp når man programmerer?

M: Det kan hende at man må ha det pluss det pluss det eller gange eller noe sånt. Man må kunne gange og plusse og sånt.

[...]

M: For eksempel når vi programmerte bilen, så programmerte vi at den skal kjøre framover så må man ha 5000 og 500 og noe sånn, så må man kunne pluss for å programmere. Så man må kunne pluss, minus og gange og sånn for å programmere.

Markus forteller at de som er gode til å programmere kan programmere roboter, og de er også gode i matematikk. Når jeg spør hvorfor han nevner matematikk i denne sammenhengen, svarer Markus noe famlende.

Jeg spør om Markus kan beskrive egenskapene til en sterk elev.

S: Kan du tenke på en elev i klassen, det kan godt være deg selv, som du synes er flink til å programmere?

M: Ja, han X

S: Hvordan egenskaper tenker du den eleven har som gjør den flink til å programmere?

M: har laget flere spill og programmert og sånn

Markus forteller at eleven han tenker på har erfaring med programmering fra tidligere. Eleven har blant annet programmert og laget flere spill.

Jeg spør Markus når han føler han får til å programmere.

S: Når synes du at du får til å programmere?

M: Jeg får til å programmere ... når jeg samarbeider med noen.

S: Når føler du at du får til? Er det når dere får riktig svar på oppgaven, eller at roboten gjør akkurat det du vil den skal gjøre?

M: Nei, jeg bare prøver

S: Du bare prøver? Ok, når du da til slutt får til, er det det som er viktig eller er det bare artig å prøve?

M: Det er bare artig

Markus forteller at han får til programmering når han samarbeider med andre elever. Markus forklarer at han bare prøver forskjellige ting til det blir riktig. Han understreker til tross for dette at det er artig å arbeide med programmering.

Jeg ber Markus forklare litt mer om hvordan samarbeid hjelper han å programmere.

M: Hvis jeg er alene så kan jeg ikke [...] klarer jeg ikke å programmere så godt, jeg trenger litt hjelp av andre.

S: Så det er fint å ha andre en kan diskutere hvordan man skal løse oppgaven med?

M: Ja, og hvis jeg for eksempel skal programmere et spill, så kan den andre i gruppen se at det er feil og si fra om det.

Markus forklarer at han ikke føler han klarer å programmere så godt når han er alene. Han forklarer at han støtter seg til den eller de andre elevene på gruppen for å få bekreftelse for sine tanker og ideer.

Jeg spør Markus om han kan fortelle hva klassen skulle gjort hvis han kunne bestemme i en programmeringstime.

S: Ok, hvis du kunne bestemme over en skoletime der dere skulle programmere hva skulle klassen gjøre da?

M: Programmere bilen til å kjøre langt, eller sånn liten bil.

S: En skikkelig lang oppgave med mange svinger og sånt?

M: Ja...

Markus forklarer at han ønsker å lage en skikkelig lang løype til bilene.

6.2.1 Oppsummering intervju med Markus

Min tolkning er at Markus i stor grad forbinder programmering med arbeidet fra skoletimene. Når Markus får bestemme hva klassen skal arbeide med, foreslår han å forlenge løypen for bilen som læreren har laget. Når de programmerer, forteller Markus at han får hjelp av de andre elevene i gruppen for å finne ut hvor i algoritmen det er feil eller mangler. Dette kan knyttes til algoritmebehandling (Bocconi et al., 2018). Markus forteller at man bruker litt matematikk når man programmerer, for eksempel å bruke ulike regnearter som addisjon og subtraksjon. Min tolkning er at han her henviser til variablene han plasserer i programmeringsblokkene, for eksempel for å styre hvor lenge og hvor fort bilen skal kjøre.

En flink elev er ifølge Markus god til å programmere roboter og til å bruke matematikk. Når Markus beskriver hva en flink elev i klassen gjør, forteller han at eleven har erfaring med programmering og utvikling av spill.

Analysen indikerer at Markus er positivt innstilt til programmering. Min tolkning er at Markus har lav selvtillit til sine egne programmeringsferdigheter. Han forteller at han ikke klarer å programmere så godt når han arbeider alene, men han forteller at når han får hjelp av andre elever så klarer han å programmere.

Markus forteller at læreren forklarer hva klassen skal gjøre i timen, for så å dele klassen i mindre grupper. Det er læreren som lager løypen som bilen skal kjøre gjennom. Markus forteller at klassen som regel går sammen i grupper på to eller tre elever. Han forteller at det er fint å arbeide på denne måten, siden man kan støtte seg på de andre elevene.

6.3 Intervju – Tore

Jeg ber Tore om han kan beskrive en skoletime der klassen jobber med programmering,

S: Kan du beskrive en skoletime der dokker programmere?

T: Lærerikt, folk sitt å jobbe på en data og sånt. Andre ting enn bare bøker. Man kan lage spill, man utvikle hjernen litt

Tore forteller at programmeringstimen er lærerik, og at man utvikler hjernen. Han forteller også at man jobber på datamaskiner, noe som er forskjellige fra tradisjonell undervisning med bøker.

Jeg spør Tore om han kan fortelle meg hvordan læreren organiserer klassen når de skal programmere.

S: Når dere jobber med programmering, hvordan organiserer læreren klassen og timen?

T: Da jobber vi som regel i grupper på to og tre, også diskuterer vi hvordan vi skal programmere for å løse oppgaven

På spørsmålet svarer Tore at klassen deler seg i grupper på to eller tre elever. Gruppen diskuterer programmeringsoppgave sammen for å finne en løsning.

Jeg ber Tore fortelle meg hva han tenker programmering handler om.

S: Hva tenker du programmering egentlig handler om?

T: Det handler om prøve seg fram, på forskjellige ting og [...] og gjøre ting man aldri har gjort før det er litt utfordrende.

S: Hva vil du fram til når du prøver deg fram til i programmering?

T: Ehh da.. da tror jeg kanskje at jeg. Kom frem til? Å lære masse ting å spesielt en ting jeg kan bli god på?

På spørsmålet om hva programmering handler om svarer Tore at det handler om å prøve seg fram på forskjellige ting. Etter en kort pause legger han til at man gjør ting man aldri har gjort før og at det er litt utfordrende. Tore ønsker å lære masse når han programmerer. Videre forteller Tore at han ønsker å bli god på spesielt en ting.

Jeg spør Tore om han forbinder programmering med et spesielt fag på skolen?

S: Synes du noen fag passer bedre til programmering da?

T: Hmm, kanskje noe som naturfag og matte?

S: Hvorfor akkurat de?

T: De har ting man kan bruke data med, og utforske litt

Tore tenker på svaret før han sier naturfag og matte. Når jeg spør hvorfor han forbinder programmering med akkurat de fagene begrunner han dette i at man av og til bruker data for å utforske i fagene.

Jeg spør Tore om han kan fortelle meg om egenskapene til en elev han synes er flink til å programmere. Før jeg også spør om han synes han selv lykkes med programmering.

S: Hvis du kan tenke på en elev du synes er flink å programmere, det kan godt være deg selv. Hvordan egenskaper har den eleven?

T: Jeg tenke den eleven er smart, den tenker klart, og løser da problemene når det er noe som ikke stemmer. Ja..

S: Når synes du at du får til å programmere? Eller lykkes med programmering?

T: Når jeg får til?

S: Kan du beskrive en programmeringsoppgave dere har jobbet med?

T: Vi har jobbet med sanne lys. Laget lysmeldinger

S: Okei, mhm når synes du at du «får til» når dere jobber med det?

T: Mmm, jeg får til ofte. Også har det vært noen utfordrende.

Som svar på spørsmålet ramser Tore opp flere egenskaper. Eleven er smart, den tenker klart og løser problemer hvis noe ikke stemmer. Når det kommer til mitt andre spørsmål svarer Tore at han ofte får til oppgavene, selv om de kan være utfordrende.

Jeg ber Tore fortelle meg om hans syn på å jobbe i grupper.

S: Hvordan synes du det er å jobbe i grupper da? Er det noen fordeler?

T: Det kan hjelpe å ha flere ideer til problemløsninger, man får gjort ting raskere når man samarbeider.

Tore forteller meg at det kan være fordelaktig å ha flere ideer til problemløsninger, samtidig som man får gjort ting raskere når man samarbeider.

Jeg ber Tore fortelle meg hva klassen skulle ha gjort hvis han fikk bestemme over en programmeringstime.

S: Hvis du kunne bestemme over en time der dere skulle programmere, hva skulle klassen da gjøre?

T: Hvis man har utstyr til det, så ville jeg at klassen skulle programmere en robot som kunne gjøre det jeg sa.

[...]

[...] den skulle kunne gå, den skal kunne sitte.

Tore forteller meg at han vil at klassen skal programmere en stemmestyrte robot som skal kunne lytte stemmekommandoer. Så lister han opp noen eksempler på egenskaper som roboten skal kunne utføre.

6.3.1 Oppsummering intervju med Tore

Analysen indikerer at når Tore tenker på programmering, forbinder han dette med å arbeide på en datamaskin. Han mener programmering tilhører matematikk og naturfag. Tore

assosierer disse fagene med bruk av data. Min tolkning er at Tore tenker at disse fagene av og til oppfordrer til bruk av datamaskiner for å utforske ting. Tore sier at programmering handler om å prøve seg fram på forskjellige ting, noe som kan knyttes til algoritmebehandling (Bocconi et al., 2018). Tore forteller at det hjelper med flere ideer når en skal løse programmeringsoppgavene, og at det derfor går raskere når grupper samarbeider. Dette kan knyttes til problemløsning Wing (2006).

Tore forteller at han ønsker å lære nye ting, og bli spesielt god på en ting. Min analyse indikerer at han er positiv til programmering, og han beskriver det som lærerikt. Tore forteller at selv om oppgavene av og til har vært utfordrende, opplever han at han ofte lykkes med å programmere.

En elev som ifølge Tore er flink til å programmere tenker klart og er smart. Eleven løser også problemer i programmet hvis det er noe som ikke stemmer. Dette kan knyttes til algoritmebehandling og problemløsning (Bocconi et al., 2018; Wing, 2006).

Tore forteller at læreren som regel fordeler klassen i grupper på to eller tre elever. Disse gruppene skal så samarbeide for å løse oppgaver.

6.4 Intervju – Aslak

Jeg spør om Aslak kan beskrive en typisk skoletime der klassen programmerer.

S: Kan du for eksempel forsøke å beskrive en skoletime der dere har jobbet med programmering?

A: Ehh altså når vi jobber med programmering har vi som regel jobbet med en mikro:bit som kommer i en eske. Jeg forstod egentlig ikke spørsmålet, kan du spørre en gang til?

S: Gjentar spørsmålet.

A: Nærme starten sier læreren at vi holder på med programmering. Gjør ikke det så ofte, men når vi gjør det har vi ofte sånn mikro:bit får sånn stor eske inn i rommet, eller så går vi opp og henter PCer å gå på programvare som vi da kan programmere.

Aslak forteller at læreren har en felles oppstart, før elevene henter det utstyret de trenger for å begynne arbeidet.

Til spørsmålet om hva programmering handler om svarer Aslak følgende.

S: Hva tenker du programmering egentlig handler om? Eller hva vil man egentlig med programmering?

A: Umm, prøve å få noe eller en bestemt ting til å gjøre det du vil uten at du må styre den med dine egne hender

S: Hva vil det si for deg å være god eller flink til å programmere?

A: Det vil egentlig si at man kan få en robot til å jeg vet ikke vinke eller si hei. Bare vite hva man gjør hvordan man programmere uten å bruke klossa. Sånn bruke programmeringsspråk.

[...]

S: Synes du programmering hører til et bestemt skolefag? Eller hvis du skulle tatt ett skolefag som du synes programmering passer best til?

A: Jeg mener matte, og type språkfag. Fordi programmering er jo en type språk, men så er det jo masse type mattestykka, millisekund fram osv.

S: Hvorfor språkfag?

A: Jeg føle programmering har jo et eget språk. Det finnes flere forskjellige varianter av programmering nesten som språk. Type sånn

S: Ja, kanskje derav navnet programmeringsspråk?

A: Ja, akkurat.

Aslak forteller meg at man gjennom programmering ønsker å kunne styre noe uten å bruke hendene direkte. Han forteller videre at man er god til å programmere når man behersker forskjellige bevegelser. Mest av alt er man god når man programmerer uten bruk av blokkprogrammering. Når jeg spør om han forbinder programmering med et spesielt fag på

skolen svarer Aslak matematikk og språkfag. Han begrunner dette i at programmering er et slags språk, men samtidig bruker man masse matematikk.

Jeg spør om Aslak kan beskrive tankegangen sin når han programmerer.

A: Når jeg går framover i programmering så prøve jeg en ting, så tester jeg. Hvis det ikke går helt greit så justerer jeg på det, hvis det går fint kan det også hende jeg justerer litt på det bare for å få det «fin-tweaket». Det er liksom prøving og feiling som får svar. Det er viktig å kunne teste ting..

Aslak forteller meget detaljert om hvordan han prøver seg fram. Hvordan han justerer eller «tweaker» på ting hvis noe ikke går helt etter planen. Han forklarer at det er prøving og feiling som gir resultater. Han avslutter med å understreke at det er viktig å kunne teste ting.

Jeg spør om hva som skal til for å lykkes med programmering.

S: [...] Enn når synes du at du lykkes i å programmering?

A: Når jeg har fått mål. La oss si at jeg skal programmere en bil. Jeg har lyst til at den skal kjøre i en time eller kunne gjøre «donuts» eller noe sånn ting.. Når jeg da har fått den til å gjøre det har jeg nådd det målet, da må jeg finne meg ett nytt mål for å holde meg opptatt.

Aslak forteller at han føler han lykkes når han når målet han har satt seg. Han legger til at når han har gjort det så finner han seg et nytt mål å arbeide mot.

Når jeg spør Aslak om hva som kjennetegner en elev som er flink å programmere svarer han:

A: Konsentrasjon og ikke gi opp

S: Hva legger du i å ikke gi opp? Er det mye prøv og feiling?

A: Ja for asså du må jo prøve for å vite hva som egentlig går, man prøver og feiler veldig ofte. Det er det sånn egentlig kommer fram til svaret

Aslak forteller meg at det er viktig med konsentrasjon og stå på vilje når man programmerer. Han utdyper det siste med at det kan være mye prøving og feiling før man kommer fram til den riktige løsningen.

Jeg spør Aslak om hvordan klassen arbeider med programmering.

S: Hvordan jobber du individuelt, i grupper, i plenum?

A: Det kommer litt an på, noen ganger så jobber vi individuelt eller bare sånn alene. Eller så jobbe vi i grupper på 2 eller 3

S: Foretrekker du det ene eller det andre?

A: Jeg foretrekker at vi varier litt, noen grupper er verre enn andre.

S: Ser du noen fordeler med å være flere som samarbeider?

A: Ja da har man sånn flere forslag som for eksempel hvis man skal programmer en bil så har man flere forslag til funksjoner og lys og sånn. Hvis noen er tom for ideer kan man høre med andre på gruppa.

Aslak forteller at organiseringen varier fra gang til gang. Noen ganger jobber de alene eller så er de i grupper på to eller tre. Han sier at han liker denne variasjonen, fordi noen grupper er verre enn andre. Aslak uttrykker at det kan være fordelaktig å samarbeide fordi det gir rom for flere forslag. Hvis man ikke selv har noen ideer, kan gruppen hjelpe med forslag til funksjoner og lys.

Jeg ber Aslak fortelle meg hva han tenker om programmering i skolen og utstyret skolen har for å drive med programmering.

S: Skulle du ønske dere brukte mer tid på programmering?

A: Definitivt, det skulle jeg virkelig ønske, det er gøy!

[...]

S: Synes du skolen har nok utstyr til å drive med programmering? Er det noe utstyr du savner?

A: Vi har vel egentlig det utstyret vi trenger, bare alt for lite av det.

S: Okei, så dere må dele?

A: Ja normalt sett er vi gruppe på to. Det er kanskje en som er heldig og får være alene.

S: Har du noe du vil legge til på slutten?

A: Det hadde vært fint om vi fikk litt mer utstyr, for vi mangle dritmye av det utstyret vi egentlig treng.

Aslak forteller at skolen definitivt burde brukt mer tid på å programmere. Når det kommer til utstyret skolen benytter svarer Aslak at skolen har det de trenger, bare at det er for lite av det. Han påpeker at problemet er at de ofte må dele, han beskriver elever som får være alene som heldige.

Jeg ber Aslak fortelle meg hva klassen skulle gjort hvis han fikk styre en time med programmering på skolen.

S: Hvis du skulle få bestemme hva klassen skulle arbeide med i en programmeringstime hva skulle dere da ha gjort?

A: Programmert en playstation-kontroller til å styre en bil.... For det går!

Aslak forteller at han ønsker at klassen skulle omprogrammere en playstation-kontroller slik at den kan brukes til å styre en bil.

6.4.1 Oppsummering intervju med Aslak

Min tolkning er at Aslaks forestillinger om programmering i matematikk er knyttet til det han tenker er formålet med programmering. Han forteller at gjennom programmering kan få en ting til å gjøre det du vil uten å styre den med hendene dine, altså automasjon (Bocconi et al.,

2018). Aslak tenker at programmering tilhører matematikkfaget, samtidig at det også kan sees på som et språkfag. Han forteller at programmering er en type språk, men så bruker man jo masse mattestykker, millisekund fram og så videre. Han forteller at det finnes mange forskjellige programmeringsspråk. Aslak forteller meg detaljert hvordan han tenker når han arbeider med programmering. Aslak beskriver hvordan han dekomponerer oppgaven i mindre deler, og hvordan han tester hver deloppgave. Hvis noe ikke stemmer korrigerer han dette før han går videre til neste deloppgave. Min tolkning er at hans beskrivelse av metoden han bruker kan knyttes til algoritmebehandling, dekomponering og automatisering (Bocconi et al., 2018). Analysen indikerer at Aslak er svært positiv til programmering i skolen, men han ønsker mer undervisning og utstyr.

Ifølge Aslak er du god til å programmere når du klarer å få en robot til å vinke eller si hei. Han forteller videre at du er god hvis du kan programmere uten å bruke blokker, altså bruke et skriftlig programmeringsspråk. Aslak tenker det er viktig å ikke gi opp og beskriver en flink elev som konsentrert. Når Aslak beskriver arbeidsmetoden til eleven forteller han at det er gjennom prøving og feiling man kommer fram til svaret, som kan knyttes til algoritmebehandling (Bocconi et al., 2018).

Analysen indikerer at Aslak har god selvtillit når det kommer til sine programmeringsevner. Aslak føler han lykkes med programmering når han kommer i mål med oppgaven han jobber med. Da forteller han at han straks finner seg en ny oppgave han vil løse.

Aslak forteller at læreren starter timen med å informere elevene om at de skal programmere. Så får de mikro:biten som kommer i en eske, før de henter seg en datamaskin for å starte å programmere. Aslak forteller at de noen gang får jobbe individuelt, men ofte så jobber de i grupper på to eller tre elever. Aslak foretrekker å variere ettersom noen grupper er verre enn andre, samtidig får man flere forslag og ideer når man er flere.

6.5 Intervju – Mari

Jeg spør om Mari kan fortelle meg hva som kjennetegner skoletime med programmering, eller om hun kan beskrive en time der klassen har jobbet med programmering.

S: Hva kjennetegner en skoletime med programmering synes du? Eller kan du beskrive en time der dere har jobbet med programmering?

M: Vi hadde sånne små sånne sånn robot biler. Så hadde vi en sånn chip så putta vi den inni bilen så hadde vi en sånn datamaskin og en ledning som va kobla til bilen også [...] skreiv vi inn på datamaskinen hvilken liksom retning vi ville den skulle gå. Så var det sånne teipbiter på gulvet som vi skulle få den til å gå i en sånn bane. [...]

[...]

S: [...] Når dere programmerer hvordan organiserer læreren timen? Har han en gjennomgang på starten? Du nevnte at dere jobbet i grupper?

M: Mhm, vi jobbe ikke akkurat i gruppe, men vi velger par selv. Det har vi i alle fall gjort tidligere. Eh, også hadde vi en sånn liten introduksjon på starten av timen første gang vi jobbet med det. Også hvis vi slet litt underveis så kom han (læreren) og hjalp oss fordi vi fikk lov til å sitte litt hvor vi ville.

Mari forteller at de arbeidet med en liten robotbil. I bilen puttet de en chip. Videre at de koblet chipen med en ledning til datamaskinen. Så skreiv de på datamaskinen hvordan bilen skulle kjøre. Hun forteller at målet var å kjøre gjennom teipbanen på gulvet. Mari forteller at klassen velger par selv. På starten av timen hadde læreren en liten introduksjon. Hun sier at når de programmerte var læreren tilgjengelig hvis noen skulle trenge hjelp.

Jeg spør Mari hva hun synes det vil si å være god å programmere.

S: Hva synes du det vil si å være god å programmere?

M: At man er liksom veldig smart i matte og sånn. Kan regne store tall og bygge maskiner som gjør forskjellige ting.

Mari forklarer at hvis man er god å programmere, så er man smart i matematikk og sånn. Hun fortsetter med at du kan regne store tall, og bygge maskinene til å gjøre forskjellige ting.

På spørsmålet om Mari forbinder programmering med noen bestemte skolefag svarer hun det følgende.

S: Hvis du tenker på noen fag her på skolen synes du programmering passer bedre til noen fag enn andre?

M: Jaa [...]

S: Hvordan fag ville det vært da?

M: Det passer kanskje til matte fordi at man må tenke veldig mye hvilke kor lang den skal gå, og kor mange grader retning den skal ha.

S: Så da bruke du matematikk for å beregne det?

M: Ja..

Mari tenker at programmering passer til matematikk. Hun begrunner det i at man må tenke veldig mye på hvor langt bilen skal kjøre eller hvor mange grader den skal svinge.

Jeg spør Mari når hun synes hun lykkes i å programmere.

S: Okei, når synes du at du lykkes i å programmere da?

M: Jeg forstår hva det er maskinen vil jeg skal gjøre, selv om jeg kanskje ikke får det til første gangen, så skjønner jeg det etter hvert.

[...]

S: Ja, og synes du det gøy underveis da eller er det bare akkurat når du får akkurat det den skal gjøre da føle du at du lykkes? Eller er det underveis når du skjønner at hvis jeg gjør sånn så løse jeg i hvert fall det problemet

M: Mm, ja det litt deilig å bli bedre å løse problemet. Men det er en veldig frihetsfølelse å bli helt ferdig med oppgaven.

Mari forteller at hun føler hun lykkes når hun forstår hva maskinen vil hun skal gjøre. Hun forklarer at det ikke er så viktig å få riktig løsning på første forsøk, så lenge hun skjønner oppgaven etter hvert. Mari forteller at hun synes det er deilig å føle at hun blir bedre til å løse problemet. Når hun blir helt ferdig føler Mari seg veldig fri.

Jeg spør Mari hva hun tenker om å arbeide i grupper når de programmerer.

S: Hva synes du da om å jobbe i grupper?

M: Umm, det er jo [...] vi kunne enten velge å jobbe for oss sjøl eller i gruppe. Men det er deilig å jobbe i gruppe fordi da blir det like mye ansvar på alle sammen og da får man hjelp hvis man står fast på noe. Også blir man bedre av hverandre...

Ifølge Mari kan de selv velge om de vil jobbe i grupper eller individuelt. Mari forteller at hun foretrekker gruppearbeid fordi ansvaret fordeles på gruppen. Videre forteller hun at elevene hjelper hverandre hvis de står fast, og at de blir bedre av hverandre.

Mari forteller meg hva hun tenker formålet med å programmere er.

M: [...] å lage maskiner smartere så det blir på en måte bedre eller enklere i hverdagen.

S: Så for å løse praktiske problemer?

M: Ja..

Formålet med programmering er ifølge Mari å lage smarte maskiner som gjør hverdagen bedre eller enklere.

6.5.1 Oppsummering intervju med Mari

Analysen indikerer at Mari synes programmering er framtidsrettet, og hun forteller at gjennom programmering kan lage smarte maskiner så det blir bedre eller enklere i hverdagen. Mari forbinder programmering med matematikkfaget. Mari forklarer at hun bruker matematikk når hun beregner hvor langt bit:boten skal kjøre, eller hvor mange grader den skal endre retning. Hun beskriver det som deilig å bli bedre til å løse et problem, og min tolkning er at dette kan knyttes til abstraksjon, som handler om å trekke ut det vesentlige fra en oppgave, og generalisering som handler om å overføre generelle løsninger fra en oppgave til en annen (Bocconi et al., 2018).

Ifølge Mari er man god til å programmere hvis man kan regne med store tall, og bygge maskiner som gjør forskjellige ting.

Mari forteller at det ikke er så viktig å klare oppgaven på første forsøk. Mari føler hun lykkes med programmering når hun forstår hva oppgaven vil hun skal gjøre. Mari foretrekker å arbeide i grupper, fordi ansvaret fordeles på elevene, de kan hjelpe hverandre og de kan lære av hverandre.

Ifølge Mari har læreren en introduksjon på starten av timen, etter introduksjonen er arbeidet stort sett elevstyrt. Hun forteller at elevene selv velger om de skal arbeide alene eller i par/grupper, og at de kan arbeide andre plasser enn på klasserommet. Læreren lar elevene utforsker programmering på egenhånd, og er selv tilgjengelig som støtte. Mari forteller at læreren laget en bane på gulvet av teip som de skulle kjøre gjennom, og at det var veldig gøy.

6.6 Intervju – Randi

Jeg ber Randi beskrive en typisk skoletime der klassen programmerer, og hvordan type oppgaver hun liker.

S: Hvordan jobber dere i klassen med programmering når dere skal jobbe med det?
Hva gjør læreren?

R: Ehh, altså de gangene vi har gjort det så har vi programmert sanne biler, og fått de til å kjøre de veiene vi vil og sånn.

[...]

S: Er det noen programmeringsoppgaver du liker bedre enn andre? Skal det være lett, skal det være veldig frie oppgaver?

R: Ja, jeg synes det burde være ganske fri oppgave, for da blir det mye morsommere.

S: Mhm, på hvordan måte blir det morsommere da?

M: Ehh for da kan man lage masse nye ting å sånn [...] Du får utforsket og teste ut nye ting

Randi forteller at de gangene de har programmert, har de programmert biler til å kjøre slik de vil. Hun foretrekker friere oppgaver, for det blir morsommere da. Hun forteller at man da kan lage nye ting, utforske, og teste ut ting.

Jeg spør Randi hva hun tenker læreren ønsker de skal lære av å programmere.

S: Enn i skolen her da? Hva ønsker læreren at dere skal lære når dere jobber med programmering?

R: Jeg tror egentlig bare at han vil vi skal ha det gøy, og finne ut av nye ting med programmering.

Randi forteller at hun tror læreren bare ønsker at de skal ha det gøy og utforske programmering.

Jeg spør Randi om hun føler hun er god til å programmere.

S: Synes du selv du er god til å programmere?

R: Eh.. nei, hehe.

S: Hvorfor ikke?

R: Jeg skjønner meg ikke helt på det.

[...]

R: Det er kanskje litt sånn suksess når man får den til å kjøre som man vil, og blinke som man vil. Så har man klart det!

Randi forklare at hun ikke skjønner seg helt på programmering, hun synes derfor ikke hun er flink. Hun forteller hun føler hun lykkes når hun får bilen til å gjøre det hun ønsker.

Jeg spør om Randi kan beskrive hva som kjennetegner en elev god til å programmere.

S: Hvis du da tenker på noen elever i klassen du synes er god til å programmere. Hva kjennetegner dem da?

R: Eh, jeg vet ikke helt, bare når vi jobbet i gruppen så var det de som programmerte alt fordi de kunne det fra før av..

Randi forteller at hun ikke helt vet hva som kjennetegner en god elev. Hun forteller at hun har jobbet på gruppe med en elev som endte opp med å programmere alt selv, fordi eleven kunne det fra før.

Jeg spør om Randi ser noen fordeler med å jobbe individuelt.

S: Ser du noen fordeler med å jobbe individuelt?

R: Nei, hehe.

S: Du liker å jobbe i grupper?

R: Ja

S: Hvorfor det?

R: Fordi da synes jeg det kommer flere ideer og det blir mye morsommere å jobbe.

Randi ser ingen fordeler med å arbeide individuelt når de programmerer. Hun forteller at hun synes det kommer flere ideer når man er flere, og at det da blir morsommere å jobbe.

Jeg undersøker hva Randi tenker om å gjøre feil når hun programmerer.

S: Hva synes du om å prøve å feile sånn da? Er det dumt når du ikke får det til med en gang eller er det bare artig?

R: Ja, det er egentlig bare mest artig

S: Er du redd for å gjøre feil når du programmerer?

R: Nei, hehe.

S: Hvorfor ikke?

R: Fordi alle gjør feil

S: Enn i for eksempel matematikk da? Hvis du får et feil svar synes du det er dumt?

R: Ja litt

S: Hvorfor er det annerledes enn i programmering?

R: Jeg vet ikke helt, men det bare.. jeg føle ikke at det er like liksom viktig å vite programmering som fire ganger to liksom

Randi forteller at hun ikke er redd for å gjøre feil når hun programmerer. Hun forklarer at alle elevene gjør feil. Når jeg ber henne sammenligne det å gjøre feil når hun programmerer, og det å gjøre feil når hun regner tradisjonell matematikk, svarer Randi at det ikke er like viktig å programmere som å kunne fire ganger to.

Jeg ber Randi fortelle meg hva klassen skulle gjort hvis hun kunne bestemme over en programmeringstime.

S: Hvis du kunne vært sjef i en time, hva skulle klassen gjort da? Programmering altså.

R: Ehh, fått varmeovnene til å skru seg på når det blir kaldt, for det gjør dem ikke.
Hehe

Randi forteller også at hvis hun fikk bestemme over en programmeringstime, så skulle målet være å få varmeovnene til å slå seg på automatisk når det ble for kaldt i klasserommet.

6.6.1 Oppsummering intervju med Randi

Randi forteller at de gangene de har jobbet med programmering så har de programmert biler med mål om å styre dem dit de ønsker. Analysen indikerer at Randi er læringsfokusert og trives når hun får utforske programmering. Randi ler når jeg spør om hun er redd for å gjøre feil, og hun forteller at alle gjør feil når de programmerer. Hun påpeker at det er litt dumt å gjøre feil i matematikk, men det er ikke like viktig å kunne programmering som å regne fire ganger to. Analysen min indikerer at Randi tenker at målet til læreren er at de skal ha det gøy og utforske programmering på skolen. Når Randi forteller om sin drømmetime, forklarer hun at hun skulle ønske klassen kunne programmere varmeovnene i klasserommet til å slå seg på av seg selv. Dette tolker jeg som at Randi har en forståelse for automasjon, og at programmering også har et praktisk formål (Bocconi et al., 2018).

Når Randi forteller om elever som er god til å programmere, forteller hun at det var de som programmerte alt fordi de kunne det fra før av. Min tolkning er at Randi foretrekker å få samarbeide med elever på samme nivå, slik at hun får anledning til å bidra med arbeidet.

Randi synes ikke selv at hun er flink til å programmere. Ifølge henne selv skjønner hun seg ikke helt på det, men hun forteller at hun føler hun lykkes når bit:boten gjør det hun ber den om.

Min tolkning er at Randi foretrekker mer åpne oppgaver. Hun ønsker at læreren skal tilrettelegge slik at elevene skal få rom til å utforske programmering. Randi forteller at hun ikke kan se noen fordeler med å programmere alene. Arbeid i grupper er både artigere og gir rom for flere ideer.

6.7 Intervju – Anne

Jeg ber Anne om å beskrive hva som kjennetegner en typisk skoletime med programmering.

S: Hva synes du kjennetegner en skoletime med programmering? Eller kan du beskrive en skoletime der dere har jobbet med programmering?

A: Eeeh, vi lagde sånn der bit:bot-greie. Sånn der programmerte biler, det var litt artig fordi vi fikk en sånn der mikro-chip eller nå hva det heter. Også måtte vi programmere den, også fikk vi den til å lyse å kjøre og sånn.

Anne forteller at de fikk en bit:bot bil, som de programmerte til å lyse og kjøre.

Jeg spør Anne om hvordan læreren organiserer timen.

S: Når dere programmerer, hvordan organiserer læreren timen?

A: Han kjøpte eller han ordnet sånne ting vi kan programmere eller noe sånt, så får vi PC og går inn på programmet og sånt, også sier han sånn for eksempel at vi må først få bilen til å lyse en farge, og kjøre en meter fram også snu å gå tilbake. Og sånt..

S: Så dere løser mange oppgaver da?

A: Ja..

S: Hva tenker du om den arbeidsmetoden da? Synes du det er greit å starte med noen sånne enkle ting å bygge det oppover eller vil du ha litt sånn friere oppgaver?

A: Nei.. det er eller sånn det er bedre å starte med noe enkelt, også få det litt ettersom hvor flink til det man er.

Anne forteller at læreren ordner bit:botene, og deler ut datamaskiner. Videre forklarer hun meg at læreren bestemmer hva de skal programmere bit:boten til å gjøre. Anne foretrekker å arbeide med flere enkle oppgaver, for så å øke vanskelighetsgraden etter hvor flink man er.

Jeg stiller Anne spørsmålet om hun synes programmering hører til noen bestemte fag på skolen?

S: Hvis du tenker på fagene du har på skolen synes du programmering passer bedre til noen fag en andre?

A: Jaa.. kanskje til naturfag og sånt

S: Mhm, hvorfor tenker du det?

A: Fordi det liksom.. data eeh eller kanskje samfunnsfag fordi det er liksom utvikling og sånt av data

[...]

S: Hva tenker du programmering handler om? Hva ønsker man å oppnå gjennom å programmere?

A: Å få ting til å skje med bilen eller andre ting, data-ting.

Anne forbinder først programmering med naturfag, så får hun tenkt litt og legger til samfunnsfag. Hun forklarer at det er på grunn av at fagene handler om data og utvikling. Anne forteller senere at programmering handler om å få ting til å skje med bilen eller andre data-ting.

Jeg spør først hva Anne forbinder med å være god i programmering. Så spør jeg henne når hun synes hun lykkes med å programmere.

[...]

A: Ja, men jeg er ikke så flink med data-ting, så jeg får det ikke alltid til, men det er artig hvis man får det til da.

S: Hva synes du det vil si å være god i «data-ting» da eller programmering?

A: Å liksom få ting til med en gang og skjønne hvor man skal plassere inn ting. Og hva man skal skrive og sånt.

S: Hvis du tenker på noen i klassen du synes er god i sanne «data-ting», hvilke egenskaper eller hva gjør de som gjør dem gode?

A: De får ting til med en gang og sånt. De får ting til å skje mye fortere og mye mer riktig og sånt.

[...]

S: Når synes du at du lykkes med å programmere da?

A: På slutten av timen som regel (ler)

S: På slutten av timen ja, så du sier at du ikke får det til med en gang?

A: Nei (bekreftende)

S: Synes du det fortsatt er artig da? Eller synes du det er dumt når du ikke får det til?

A: Nei, det er artig. For jeg får det som regel til på slutten

Anne forteller at du er god hvis du får til oppgaven med en gang. Hun påpeker også at man er god hvis man forstår hvor man skal plassere ting, og hva man skal skrive.

Jeg spør Anne om hvordan de arbeider når de programmerer.

S: Når dere jobber med programmering jobber dere individuelt, i grupper eller i plenum?

A: Vi kan jobbe litt sånn som vi vil, så noen ganger så jobber jeg alene så eh.. jeg har jobba med flere folk før også eller i hvert fall med en annen.

S: Hva synes du om å jobbe alene da? Ser du noen fordeler med det?

A: Ehh, ja, for da kan jeg bestemme litt mer selv, hvis vi får litt friere oppgaver hva jeg kan gjøre ja, men det er lettere når man er to eller flere. Fordi da hvis man ikke får til noe, kan det hende at den andre kan hjelpe til med det og sånt.

Anne forteller at hun har jobbet alene, men også samarbeidet med andre. Hun mener en god ting med å være alene er at du bestemmer selv hva du skal prøve. Samtidig poengterer hun at det kan være en fordel å være flere elever for da kan man hjelpe hverandre.

Jeg ber Anne beskrive hva klassen skulle gjort hvis hun fikk bestemme i en programmeringstime.

A: Ehh, kanskje få den (bit:boten) til å gjøre et slags show eller noe sånt. Få den til å lyse også snurre rundt, å masse slike ting.

Anne forteller at hvis hun fikk bestemme skulle elevene sette opp et show med hjelp av bit:boten.

6.7.1 Oppsummering intervju med Anne

Tolkningen min er at Anne forbinder programmering med super:bit prosjekt, særlig programmeringen av bit:boten. Hvis Anne fikk bestemme over en time skulle klassen kombinere funksjonene til bit:boten til et show. Anne knytter programmering til naturfag og samfunnsfag. Hun begrunner dette i at naturfag og samfunnsfag handler om utvikling, data og sånt. Formålet med å programmerer er ifølge Anne å få ting til å skje med bilen eller andre data-ting, noe som kan knyttes til automasjon (Bocconi et al., 2018).

Anne forteller at en god elev får til ting med engang, de får ting til å skje mye raskere og riktigere enn andre elever. Elementer hun er inne på her er abstraksjon, som handler om å finne det vesentlige i oppgaven, og generalisering som handler om å bruke generelle løsninger i nye oppgaver (Bocconi et al., 2018). Hun forteller at du er god hvis du raskt forstår hvor du skal putte ting, og hva du skal skrive. Min tolkning er at hun henviser til hvordan du plasserer programmeringsblokkene, og hvilke variabler du plasserer i dem. Dette kan knyttes til dekomponering og algoritmebehandling (Bocconi et al., 2018).

Min tolkning er at selv om Anne ikke synes hun er flink til å programmere, synes hun det er gøy. Anne forteller at hun ikke alltid får det til, men at det er artig når man får det til. Hun forteller senere at hun som regel lykkes med å programmere mot slutten av timen.

Ifølge Anne får elevene selv velge om de vil jobbe individuelt eller i grupper. Min tolkning er at Anne ser fordeler og ulemper med begge metodene. Hun sier at når man jobber alene bestemmer man selv hva og hvordan, særlig hvis det er friere oppgaver. Samtidig poengterer hun at det kanskje er lettere å jobbe sammen i en gruppe, fordi man da kan hjelpe hverandre hvis man ikke får til noe.

7 Resultat

Jeg ønsker i dette kapittelet å presentere en oppsummering av resultatene fra undersøkelsen min, for der etter å prøve å besvare forskningsspørsmålet mitt: Hvilke forestillinger kan elever ha om programmering i matematikk? Me bakgrunn i at jeg ikke har funnet annen empirisk forskning på elevers forestillinger om programmering i matematikk, har jeg ikke mulighet til å drøfte funnene mine i sammenheng med annen forskning. Jeg blir derimot å sammenligne funnene mine med forskning på bruk av programmering i matematikkfaget. For å tydeliggjøre funnene mine vil jeg presentere elevenes forestillinger om programmering i matematikk med utgangspunkt i McLeod (1992) sin modell. Det vil si at jeg vil presentere elevenes forestillinger om programmering i matematikk, forestillinger om programmering i matematikkundervisningen, forestillinger om selvet og forestillinger om den sosiale konteksten for å forsøke å si noe helhetlig om elevenes forestillinger om programmering i matematikk.

På grunnlag av transkriberingen av intervjudatamaterialet og det påfølgende arbeidet med å analysere dataene, mener jeg å ha et grunnlag for å si noe om elevenes forestillinger om programmering i matematikk. Min tolkning er at informantene mine har til dels forskjellige forestillinger om programmering i matematikk. For mange er deres forestillinger om programmering i matematikk basert kun på deres erfaringer fra super:bit prosjektet. Slik jeg tolker resultatet er elevenes forestillinger om programmering i matematikk knyttet til bruk av datamaskiner, blokkprogrammering, automasjon, matematikk og begreper fra algoritmisk tenkning. Resultatet indikerer at flere av elevene har forestillinger om at programmering handler om å arbeide på en datamaskin, der de bruker programmeringsblokker for å løse et problem. Slik jeg tolker dataene, er dette særlig i sammenheng med automatiseringen av bit:boten (Bocconi et al., 2018). For eksempel beskriver Pia hvordan hun bruker programmeringsblokker med den hensikt å bevege bit:boten. Resultatet fra studien viser at de aller fleste elevene også har forestillinger som forbinder programmering med matematikkfaget. Funnene mine indikerer at elevene i varierende grad kobler bruken av variabler i blokkene til matematikk. Mari er et godt eksempel på en elev som er bevisst denne forestillingen, og hun forteller at hun bruker matematikk i beregningen av hvor langt bit:boten skal kjøre eller hvor mange grader den skal svinge. Slik jeg tolker intervjudataen fra Tore forbinder også han programmering med matematikk, men han knytter det i større grad til at matematikkfaget i hans øyne legger opp til utforskning og oppgaveløsning på datamaskiner. Resultatet viser også at flertallet av elevene har forestillinger om programmering som kan

sees i sammenheng med algoritmisk tenkning eller computational thinking (Bocconi et al., 2018; Wing, 2006). Elevene nevner dette ikke ordrett, men beskriver heller arbeidsmetoder eller tankemåter som kan knyttes til paraplybegrepene algoritmisk tenkning eller computational thinking (Bocconi et al., 2018; Wing, 2006). I svarene ser jeg særlig tendenser til beskrivelser av hvordan elevene forteller at de prøver seg fram når de programmerer, noe jeg knytter til algoritmebehandling og dekomponering (Bocconi et al., 2018). Aslak er den eleven jeg intervjuet som går mest i detalj når han beskriver hvordan han tenker man skal prøve seg fram i arbeidet med programmering. Analysen av intervjudataen til Aslak, gir meg en forståelse for hvordan han med det formål å automatisere bit:boten dekomponerer oppgaven og bruker algoritmebehandling for å utbedre eventuelle feil.

På starten av intervjuene spurte jeg elevene om hva som kjennetegner en skoletime med programmering, eller om de kunne fortelle om en konkret undervisningssituasjon der de arbeidet med programmering. Mine resultater indikerer at elevene har svært like forestillinger om programmering i matematikkundervisningen. Flesteparten av elevene hadde sterke forestillinger om at programmering i matematikkundervisningen er knyttet til super:bit prosjektet, mer bestemt programmering av bit:boten. Noe som styrker denne påstanden, er at flere av elevene fortalte meg med varierende grader av detaljer hvordan undervisningstimene utspilte seg. Jeg finner ingen av påstandene til elevene å være i direkte motsetning til hverandre, men heller at de utfyller hverandre. Resultatet av en sammenligning av elevenes beskrivelser, gir meg en forståelse for en typisk undervisningssituasjon der klassen programmerer. For å styrke denne påstanden vil jeg presentere en undervisningstime fra start til slutt, basert på funnene mine. Hvert utsagn har som regel to eller flere elever som forteller meg det samme. Gjennom Markus får jeg vite at læreren starter timen. Klassen informeres om at de skal arbeide med programmering, og hva målet med timen er. Samtlige elever jeg intervjuet forteller at læreren legger opp til at de skal arbeide sammen i grupper. Aslak forteller at de henter bit:boten i en stor eske, og at de må opp en etasje for å hente datamaskinene. Pia forteller at elevene har mulighet til arbeide på gangen utenfor klasserommet. Pia forteller også at læreren gir dem stor frihet til å utforske og programmere på egenhånd. Læreren er ifølge Mari tilgjengelig for å hjelpe elevene hvis det er noe de ikke forstår. Markus forteller at læreren lager en bane på gulvet som bit:boten skal manøvrere seg gjennom. Utvalget mitt har et svært begrenset antall undervisningstimer og undervisningsopplegg de baserer forestillingene sine på. Min tolkning er at det derfor kan

være naturlig at resultatet indikerer at elevenes forestillinger til programmering i matematikkundervisningen er svært like.

Allerede under intervjuet følte jeg at jeg fikk en viss innsikt i elevenes forestillinger til selvet. Resultatet av analysen indikerer at elevene har lettere for å fortelle om sine egne prestasjoner, og når de føler at de lykkes når de programmerer. Det er vanskeligere å beskrive hva andre elever som de anser som god i programmering gjør, eller hvilke egenskaper disse elevene innehar. Resultatene viser at elevene har forskjellige forestillinger om egne prestasjoner, og om hva som skal til for å lykkes med programmering i matematikk. Min tolkning er at elevene vektlegger prestasjoner, læring og forståelse ulikt. Med det resultat at elevenes forestillinger om selvet baseres på hva eleven selv velger å fokusere på. Under vil jeg presentere noen av funnene fra intervjuene for å tydeliggjøre denne forskjellen. Slik jeg har tolket intervjudataen fra det første intervjuet, er Pia en elev som er opptatt av å prestere. Pia føler hun lykkes når hun klarer ting som ikke andre klarer. Det samme gjelder andre elever som Pia synes er gode til å programmere. De får til ting som ingen andre klarer. I det tredje intervjuet snakker jeg med Tore. Han har slik jeg vurderer datamaterialet et ønske om å få utforske og lære når han programmerer. Min tolkning er basert på at Tore i løpet av intervjuet forteller at han ønsker å lære nye ting, og bli veldig god på en spesiell ting. Han sier også at det av og til er utfordrende å programmere, men at han som regel føler han lykkes. Det femte intervjuet er med Mari. For Mari er ikke det viktigste å komme fram til en løsning på oppgaven øyeblikkelig. Mari føler hun at hun lykkes når hun forstår hva oppgaven går ut på, slik at hun kan jobbe mot det målet. Jeg tolker intervjudataen til Mari som at hun vektlegger forståelse som det viktigste. Resultatet slik jeg tolker det er at selv om elevene vektlegger forskjellige aspekter når de programmerer så har alle en forestilling om at de lykkes på sin måte når de programmerer.

Resultatene mine indikerer at elevenes forestillinger om den sosiale konteksten var svært lik innad i klassen. Resultatet er kanskje ikke helt uventet, ettersom det er innad i klassen de sosiale normene og de sosiomatematiske normene blir formet (Cialdini & Goldstein, 2004). Det er derfor naturlig at elevene har noenlunde like opplevelser og dermed noenlunde like forestillinger om den sosiale konteksten. Når jeg under intervjuet spurte om elevene kunne fortelle om en skoletime der de hadde arbeidet med programmering, svarte samtlige at klassen hadde arbeidet gruppevis med programmering. Resultatene indikerer at elevene i stor grad lykkes i å samarbeide. Jeg begrunner dette med blant annet at når elevene beskriver arbeidsmetodene sine, så kan de knyttes til algoritmebehandling og dekomponering (Bocconi

et al., 2018). Markus er her et godt eksempel når han forteller at han ikke alltid får til å programmere på egenhånd. Gruppen hjelper Markus til å kartlegge hvor i algoritmen det er feil eller mangler, noe som gjør at Markus føler han lykkes. Et annet poeng som støtter opp under antagelsen min er at av de syv elevene jeg har presentert intervjudata fra var det bare Aslak og Anne som i det hele tatt reflekterte over hvilke fordeler det å arbeide individuelt kunne medføre. Jeg tolker dette som at elevene har en forståelse omkring de sosiale normene og de sosiomatematiske normene, knyttet til hvordan de skal opptre og redegjøre for tankemåtene sine når de programmerer. Slik jeg ser det har elevene forestillinger om at de sammen skal hjelpe hverandre til å forstå oppgaven og drøfte ideer og løsningsforslag. Selv om funnene mine tyder på at elevene har ganske så like forestillinger om den sosiale konteksten, utspiller det seg kanskje ikke alltid slikt i undervisningen. Randi forteller at når hun var på gruppe med en elev som kunne programmere, endte det opp med at den eleven gjorde alle oppgavene fordi eleven kunne det fra før.

Hvis jeg skal forsøke å se resultatene mine i sammenheng med tidligere forskning, blir dette utfordrende. Mye av den tidligere forskningen har som formål å se hva programmering kan tilby matematikkfaget, og innebærer at de har et matematisk perspektiv på programmering (Kaufmann & Stenseth, 2020). Resultatene mine viser at elevene, selv om de er bevisst at de bruker matematikk når de programmerer, ikke har brukt programmering i en for dem tydelig matematisk kontekst. Resultatene mine gir meg derfor lite grunnlag for å trekke slutninger omkring elevenes motivasjon for å lære matematikk, eller hvordan programmering påvirker deres matematiske prestasjoner (Forsström & Kaufmann, 2018). Heller ikke hvordan Scratch-programmering påvirker elevenes motivasjon og interesse for matematikk (Dohn, 2020). Det jeg mener resultatene mine gir meg grunnlag for å sammenligne er hvordan elevene samarbeider og endringen av lærerens rolle (Forsström & Kaufmann, 2018). Det Forsström og Kaufmann (2018) kaller den kollektive læringsprosessen. Slik jeg tolker resultatet mitt har elevene en forestilling om at læreren inntar en annen rolle når de programmerer. Lærerrollen går fra å være mediator mellom kunnskapen og elevene, til å i større grad være en tilrettelegger. Læreren tilrettelegger slik at elevene kan utforske programmering i grupper selvstendig fra læreren. Forsström og Kaufmann (2018) påpeker at man burde se programmering i matematikk som en kollektiv læringsprosess innad i gruppen eller klassen i stedet for å fokusere på individet. Jeg mener at resultatet mitt samsvarer godt med denne påstanden, men med ett unntak. Når Randi fortalte om eleven som kunne oppgavene fra før

av, og som gjorde alle oppgavene før resten av gruppa fikk sett på dem. I resultatet finner jeg beskrivelser jeg synes virker treffende for det vi kan kalle en kollektiv læringsprosess.

7.1 Validitet og reliabilitet

Validitet i sammenheng med kvalitative forskningsprosjekter kan sees i sammenheng med i hvilken grad metoden undersøker det den er ment å undersøke (Kvale & Brinkmann, 2015). Det er vanlig å skille validitet eller gyldighet i indre validitet og ytre validitet (Postholm et al., 2018). Den indre validiteten henviser til hva slags konklusjoner jeg som forsker har grunnlag for å trekke, basert på datagrunnlaget jeg har samlet (Postholm et al., 2018). Ytre validitet kalles også overførbarhet. Det vil si hvor overførbart er resultatet fra min undersøkelse til en undersøkelse i en lignende kontekst. På grunnlag av at jeg har benyttet en kvalitativ metode og forsøkt å være så transparent som mulig med hvordan jeg har kommet fram til funnene mine, vil jeg slik jeg tolker resultatet mitt tenke at det vil være mulig å generalisere noen av funnene mine. En kvalitativ metode legger til grunn en tydelig kontekst og et tidsrom (Postholm et al., 2018). Det som gjør min undersøkelse unik er elevenes erfaringer med programmering, og dermed deres forestillinger knyttet til programmering i matematikk. Det er mulig at lærere som utelukkende har arbeidet med programmering gjennom super:bit prosjektet kan dra kjensel på situasjonen og forestillingene jeg beskriver (Postholm et al., 2018).

Reliabilitet eller pålitelighet henviser til i hvor stor grad vi kan stole på datagrunnlaget jeg legger til grunn for resultatene mine (Postholm et al., 2018). Postholm et al. (2018) knytter reliabilitet til hvordan forskningsprosessen og forskeren kan ha påvirket resultatet av datainnsamlingen. For å styrke reliabiliteten til et studie er det dermed viktig at jeg som forsker drøfter hvordan jeg kan ha påvirket resultatet, og at forskningsprosessen er så transparent som mulig (Postholm et al., 2018).

Gjennom transkriberingen og den påfølgende analysen av intervjuene, er det tydelig for meg at jeg er en nybegynner når det kommer til forskning og bruk av kvalitative intervju som forskningsmetode. Siden vi mennesker samtaler daglig, kan det fort lede oss til å tro at å gjennomføre kvalitative intervjuer er enkelt (Postholm et al., 2018). Det er ikke så enkelt som å ta en diktafon og finne en informant, man må lære intervjuferdighetene gjennom intervju praksis (Kvale & Brinkmann, 2015). I intervju prosessen har jeg også kommet til å

stille lukkede, og/eller ledende spørsmål. Jeg merker at det er særlig i sammenhenger der jeg ønsker at elevene skal knytte programmering til matematikk at jeg stiller denne typen spørsmål. For å styrke reliabiliteten til studien min har jeg vært påpasselig med å unnlate å inkludere disse spørsmålene i analysen. Her er et eksempel fra intervjuet med Randi hvor jeg stiller to svært lukkede spørsmål.

S: Er det litt mere lekent når det er de bilene for eksempel?

R: Ja!

S: Det er ikke meningen at dere skal få rett med engang, at man må prøve seg litt fram?

R: Ja

Jeg ønsket at Randi skulle utdype besvarelsen sin om hvorfor hun føler det er dumt å gjøre feil i matematikk, men ikke når hun programmerer. I stedet for å stille åpne spørsmål stiler jeg to lukkede spørsmål basert på min egen tolkning av situasjonen. Et enklere og bedre spørsmål ville vært å be henne om å utdype svaret. En annen løsning kunne vært å bekrefte svaret for så å være stille, slik at hun selv skulle få mulighet til å reflektere og komme med eventuell tilleggsinformasjon (Kvale & Brinkmann, 2015).

Jeg burde vært flinkere til å stille oppfølgingsspørsmål og bekreftende spørsmål (Kvale & Brinkmann, 2015). Flere plasser i transkriberingen, synes jeg det kan virke som eleven har mer å fortelle om jeg bare hadde gitt eleven litt tid til refleksjon over spørsmålet og svaret. Som uerfaren intervjuer var det nok fristende å stille neste spørsmål for å holde praten gående, og det er nok akkurat det som har skjedd. For eksempel i intervjuet med Tore, får jeg et veldig interessant svar.

S: Kan du beskrive en skoletime der dokker programmere?

T: Lærerikt, folk sitt å jobbe på en data og sånt. Andre ting enn bare bøker. Man kan lage spill, man utvikle hjernen litt

I ettertid ser jeg at jeg kunne stilt flere oppfølgingsspørsmål til det korte svaret til Tore. For eksempel kunne jeg spurt om hva Tore mener er lærerikt? Hva mener du med å jobbe på data? Har du laget noen spill? Hvordan utvikler man hjernen når man programmerer? I stedet for å følge opp det Tore forteller, går jeg videre til spørsmål tre i intervjuguiden.

Som forsker er jeg uerfaren, og jeg har som følge av dette gjort noen feilgrep i sammenheng med forskningen. Jeg mener at jeg gjennom transkriberingen og arbeidet med analysen har rensket vekk disse ledende og/eller lukkende spørsmålene, slik at de funnene jeg presenterer i størst mulig grad representerer elevenes egne ord og tanker.

7.2 Begrensninger og videre forskning

Flere forskere påpekte at det å undersøke forestillinger kunne være en krevende prosess (Lester, 2002; Pehkonen et al., 2003; Spangler, 1992). Gjennom arbeidet med denne undersøkelsen har jeg fått en dypere forståelse for forestillinger. Forståelse for hvordan man kan stille bedre spørsmål som kan gi innsikt i forestillingene til intervjuobjektet. Jeg har også fått drillet på diverse begreper knyttet til programmering, som jeg blir å ta med meg videre inn i arbeidslivet.

Som metode har jeg kun brukt intervju. Jeg tenker det kunne vært svært interessant å kombinere to metoder. For eksempel hadde det vært spennende å observere en skoletime med programmering i matematikk med det formål å sammenligne observasjonsdata med intervjudata. En slik metode gir rom for å bekrefte utsagn fra elevene med observasjonene mine eller visa versa. Ikke minst gir det meg mulighet til å observere de sosiale normene og de sosiomatematiske normene når klassen programmerer. Jeg tenker en slik kombinasjon av metoder ville styrket validiteten og reliabiliteten til resultatene mine.

Resultatet mitt indikerer også at elevene ikke har hatt anledning til å arbeide og reflektere så mye om programmering i matematikk som jeg kanskje skulle ha ønsket. Det er forståelig at elevene forbinder programmering med super:bit prosjektet, og at deres helhetlige forestillinger om programmering i matematikk i stor grad må sees i sammenheng med dette. Likevel synes jeg bruken av modellen til McLeod (1992) hjalp meg til å presentere elevenes forestillinger på en oversiktlig og strukturert måte. Hvis elevene hadde fulgt LK20 fra de startet på skolen, vil jeg våge å påstå at kvaliteten på dataen fra intervjuene mine ville vært bedre. De ville da systematisk vært innom begreper og arbeidsmetoder som direkte og/eller indirekte kan benyttes i programmeringen. Det jeg møter er elever som stort sett har opplevd og arbeidet med programmering i et vakuum i form av super:bit prosjektet. Elevene har derfor ikke grunnlag for å bruke ord eller begreper som man forbinder med algoritmisk tenkning eller matematikk. I tillegg er de sosiale normene og de sosiomatematiske normene for

programmering i matematikk enda ikke fullt utarbeidet. Hvordan man skal opptre eller argumentere og resonere i en programmeringssituasjon kan nok bli enda bedre.

Det virker for meg som om super:bit prosjektet brukes i skolen for først og fremst motivere eller skape interesse for programmering. Jeg må understreke at jeg ikke mener det ikke er et stort læringspotensial i form av super:bit prosjektet. Skolene trenger trolig bare litt tid til å implementere og fordøye de nye kompetansekravene og arbeidsmetodene som følger med LK20. Som flere forskere har understreket gir ikke implementering av programmering og teknologi i skolesammenheng automatisk en positiv læringseffekt (Drijvers et al., 2009; Haraldsrud et al., 2020; Heid et al., 2013). Programmering i skolen må sees i sammenheng med implementeringen, læreplanene og kompetansen til de som skal undervise i programmering (Drijvers et al., 2009; Haraldsrud et al., 2020; Heid et al., 2013). Jeg tenker det derfor ville vært spennende å gjennomføre en lignende studie på et senere tidspunkt. Dette når elevene har hatt anledning til å arbeide med programmering i en mer tradisjonell matematisk kontekst.

Referanseliste

- Bocconi, S., Chiocciariello, A., & Earp, J. (2018). The Nordic approach to introducing Computational Thinking and programming in compulsory education. doi:10.17471/54007
- Bueie, H. (2019). *Programmering for matematikklærere*. Oslo: Universitetsforlaget.
- Cialdini, R., & Goldstein, N. (2004). Social Influence: Compliance and Conformity. *Annual review of psychology*, 55, 591-621. doi:10.1146/annurev.psych.55.090902.142015
- Dohn, N. B. (2020). Students' interest in Scratch coding in lower secondary mathematics. *British Journal of Educational Technology*, 51(1), 71-83. doi:<https://doi.org/10.1111/bjet.12759>
- Drijvers, P., Kieran, C., Mariotti, M.-A., Ainley, J., Andresen, M., Chan, Y. C., . . . Meagher, M. (2009). Integrating Technology into Mathematics Education: Theoretical Perspectives. In C. Hoyles & J.-B. Lagrange (Eds.), *Mathematics Education and Technology-Rethinking the Terrain* (pp. 89-132). Boston, MA: Boston, MA: Springer US.
- Forsström, S. E., & Kaufmann, O. T. (2018). A Literature Review Exploring the use of Programming in Mathematics Education. *International Journal of Learning, Teaching and Educational Research*, 17(12), 18-32. doi:10.26803/ijlter.17.12.2
- Furinghetti, F., & Pehkonen, E. (2002). Rethinking Characterizations of Beliefs. In G. C. Leder, E. Pehkonen, & G. Törner (Eds.), *Beliefs: A Hidden Variable in Mathematics Education?* (pp. 39-57). Dordrecht: Springer Netherlands.
- Gjøvik, Ø., & Torkildsen, H. A. (2019). Algoritmisk tenkning. *Tangenten - Tidsskrift for matematikkundervisning*(3), 31-37. Retrieved from <http://www.caspar.no/tangenten/2019/Tangenten%203%202019%20Gj%C3%B8vik%20Torkildsen.pdf?fbclid=IwAR3lLaDiUI54BzIzEA7nNyqsYGn6xNj1uU31-J6gpc2rJNS6aYDOPUJTxA>
- Gravemeijer, K., & Cobb, P. (2006). Design research from a learning design perspective. In J. V. d. Akker, K. Gravemeijer, S. McKenney, & N. Nieveen (Eds.), *Educational design research* (pp. 17-55): Oxon: Routledge.
- Green, T. F. (1971). *The Activities of Teaching*: Mcgraw-Hill.
- Hannula, M. S., Di Martino, P., Pantziara, M., Zhang, Q., Morselli, F., Heyd-Metzuyanım, E., . . . Goldin, G. A. (2016). Attitudes, Beliefs, Motivation and Identity in Mathematics Education: An Overview of the Field and Future Directions. 1-35. doi:10.1007/978-3-319-32811-9_1

- Haraldsrud, A. D., Sveinsson, H. A., & Løvold, H. H. (2020). *Programmering i skolen*. Oslo: Universitetsforlaget.
- Heid, M. K., Thomas, M. O. J., & Zbiek, R. M. (2013). How Might Computer Algebra Systems Change the Role of Algebra in the School Curriculum? In M. A. Clements, A. J. Bishop, C. Keitel, J. Kilpatrick, & F. K. S. Leung (Eds.), *Third International Handbook of Mathematics Education* (pp. 597-641). New York, NY: Springer New York.
- Kamiya, N. (2018). Teacher and student beliefs. In A. Shehadeh & J. Lontas (Eds.), *TESOL Encyclopedia of English Language Teaching (Vol. Approaches and methods in English for speakers of other languages)*: Wiley.
- Kaufmann, O. T., & Stenseth, B. (2020). Programming in mathematics education. *International Journal of Mathematical Education in Science and Technology*, 1-20. doi:10.1080/0020739X.2020.1736349
- Kloosterman, P. (2002). Beliefs About Mathematics and Mathematics Learning in the Secondary School: Measurement and Implications for Motivation. In G. C. Leder, E. Pehkonen, & G. Törner (Eds.), *Beliefs: A Hidden Variable in Mathematics Education?* (pp. 247-269). Dordrecht: Springer Netherlands.
- Kong, S.-C., Chiu, M. M., & Lai, M. (2018). A study of primary school students' interest, collaboration attitude, and programming empowerment in computational thinking education. *Computers & Education*, 127, 178-189. doi:<https://doi.org/10.1016/j.compedu.2018.08.026>
- Kunnskapsdepartementet. (2017). *Mer koding og teknologi inn i skolen*. (Nr: 84/17). Retrieved from <https://www.regjeringen.no/no/aktuelt/mer-koding-og-teknologi-inn-i-skolen/id2568375/>
- Kvale, S., & Brinkmann, S. (2015). *Det kvalitative forskningsintervju* (T. M. Anderssen & J. Rygge, Trans. 3. utg. ed.). Oslo: Gyldendal akademisk.
- Lester, F. K. (2002). Implications of Research on Students' Beliefs for Classroom Practice. In G. C. Leder, E. Pehkonen, & G. Törner (Eds.), *Beliefs: A Hidden Variable in Mathematics Education?* (pp. 345-353). Dordrecht: Springer Netherlands.
- Markovits, Z., & Forgasz, H. (2017). "Mathematics is like a lion": Elementary students' beliefs about mathematics. *Educational Studies in Mathematics*, 96(1), 49-64. doi:10.1007/s10649-017-9759-2
- McLeod, D. B. (1992). Research on affect in mathematics education: A reconceptualization. *Handbook of research on mathematics teaching and learning*, 1, 575-596.

- NSD, N. s. f. f.-. (2021a). Barnehage- og skoleforskning Retrieved from <https://www.nsd.no/personverntjenester/oppslagsverk-for-personvern-i-forskning/barnehage-og-skoleforskning/>
- NSD, N. s. f. f.-. (2021b). Sårbare grupper. Retrieved from <https://www.nsd.no/personverntjenester/oppslagsverk-for-personvern-i-forskning/sarbare-grupper/>
- Op't Eynde, P., De Corte, E., & Verschaffel, L. (2002). Framing Students' Mathematics-Related Beliefs. In G. C. Leder, E. Pehkonen, & G. Törner (Eds.), *Beliefs: A Hidden Variable in Mathematics Education?* (pp. 13-37). Dordrecht: Springer Netherlands.
- Pehkonen, E., Törner, G., & Leder, G. C. (2003). *Beliefs: a Hidden Variable in Mathematics Education?* Dordrecht, NETHERLANDS, THE: Springer Netherlands.
- Postholm, M. B., Jacobsen, D. I., & Søbstad, R. (2018). *Forskningsmetode for masterstudenter i lærerutdanningen*. Oslo: Cappelen Damm akademisk.
- Schwab, K. (2017). *The Fourth Industrial Revolution*: Crown Publishing Group.
- Spangler, D. A. (1992). ASSESSING STUDENTS' BELIEFS ABOUT MATHEMATICS. *The Arithmetic Teacher*, 40(3), 148-152. Retrieved from <http://www.jstor.org/mime.uit.no/stable/41195291>
- Stenseth, B., Kaufmann, O. T., & Forsström, S. E. (2019). Programmering og matematikk. *Tangenten - Tidsskrift for matematikkundervisning*(2), 7-12. Retrieved from <http://www.caspar.no/tangenten/2019/tangenten%20%202019%20Stenseth%20et%20al.pdf>
- Utdanningsdirektoratet (Producer). (2018, 29. oktober). Film: Dybdeløring. [Videoklipp] Retrieved from <https://www.udir.no/laring-og-trivsel/lareplanverket/stottemateriell-til-overordnet-del/film-dybdelaring/>
- Utdanningsdirektoratet. (2020). *Læreplan i matematikk 1.–10. trinn*. ((MAT01-05)). Retrieved from <https://www.udir.no/lk20/mat01-05>
- Wedega, T., Skott, J., Henningsen, I., & Wæge, K. (2006). *Changing views and practices? A study of the KappAbel mathematics competition*: Norwegian Center for Mathematics Education, NTNU.
- Wing, J. (2006). Computational Thinking. *Communications of the ACM*, 49, 33-35. doi:10.1145/1118178.1118215
- Yackel, E., & Cobb, P. (1996). Sociomathematical Norms, Argumentation, and Autonomy in Mathematics. *Journal for Research in Mathematics Education*, 27(4), 458-477. doi:10.2307/749877

Vedlegg

Vedlegg 1 Meldeskjema for behandling av personopplysninger



NSD sin vurdering

Prosjekttittel

Masteroppgave 2021

Referansenummer

737630

Registrert

15.12.2020 av Sjur Schjelderup Hermansen - she020@post.uit.no

Behandlingsansvarlig institusjon

UiT Norges Arktiske Universitet / Fakultet for humaniora, samfunnsvitenskap og lærerutdanning / Institutt for lærerutdanning og pedagogikk

Prosjektansvarlig (vitenskapelig ansatt/veileder eller stipendiat)

Kjersti Wæge, kjersti.wage@uit.no, tlf: 73551120

Type prosjekt

Studentprosjekt, masterstudium

Kontaktinformasjon, student

Sjur Schjelderup Hermansen, she020@uit.no, tlf: 97732282

Prosjektperiode

01.12.2020 - 15.05.2021

Status

18.03.2021 - Vurdert

Vurdering (2)

18.03.2021 - Vurdert

NSD har vurdert endringen registrert 09.03.2021.

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 den 18.03.2021. Behandlingen kan fortsette.

Utvalg 1 har blitt endret fra matematikklærere til elever som har deltatt på Nordnorsk vitensenters super:bit prosjekt. Vi forutsetter at elevene får tilpasset informasjon om behandlingen som understreker at det er frivillig å delta, og at de selv kan gi sin tilslutning til å delta i prosjektet.

LOVLIG GRUNNLAG

Prosjektet vil innhente samtykke fra foresatte til behandlingen av personopplysninger om elevene. 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 foresatte kan trekke tilbake. Elevene vil også samtykke til deltakelse.

Lovlig grunnlag for behandlingen vil dermed være foresattes samtykke, jf. personvernforordningen art. 6 nr. 1 bokstav a.

OPPFØLGING AV PROSJEKTET

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

Lykke til med prosjektet!

Kontaktperson hos NSD: Tore Andre Kjetland Fjeldsbø

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

12.02.2021 - 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 den 12.02.2021, samt i meldingsdialogen mellom innmelder og NSD. Behandlingen kan starte.

DEL PROSJEKTET MED PROSJEKTANSVARLIG

Det er obligatorisk for studenter å dele meldeskjemaet med prosjektansvarlig (veileder). Det gjøres ved å trykke på “Del prosjekt” i øvre venstre hjørne av meldeskjemaet.

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://www.nsd.no/personverntjenester/fylle-ut-meldeskjema-for-personopplysninger/melde-endringeri-meldeskjema>

Du må vente på svar fra NSD før endringen gjennomføres.

TYPE OPPLYSNINGER OG VARIGHET

Prosjektet vil behandle alminnelige kategorier av personopplysninger frem til 15.05.2021.

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.

TAUSHETSPLIKT

Vi vil minne om at lærerne har taushetsplikt. Den strekker seg lenger enn å unnlate navnene til barna. Lærerne kan ikke fortelle historier på slik måte at andre opplysninger kan identifisere en nåværende eller tidligere elev direkte eller indirekte. Dere er i fellesskap

ansvarlige for at elevene blir omtalt i generelle ordelag under samtalen. Vi anbefaler at dere diskuterer personvern i forkant av intervjuet.

PERSONVERNPRINSIPPER

NSD vurderer at den planlagte behandlingen av personopplysninger vil følge prinsippene i 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), og 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.

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 samtykkeskjema

Til elevens foresatte

19.02.2021

Elevintervju til masteroppgave i uke 9

I forbindelse med en undersøkelse om programmering i matematikkfaget ønskes et intervju av eleven. Intervjuet er en del av min masteroppgave i matematikk ved UiT. Prosjektet handler i hovedsak om elevers forestillinger omkring programmering i matematikk.

Intervjuet kommer til å bli tatt opp med båndopptaker. Eleven kommer til å bli anonymisert i masteroppgaven. Eleven blir tildelt ett tilfeldig navn, det eneste som vil være likt er kjønn og klassetrinn.

Prosjektet skal etter planen avsluttes 15.5.2021. Ved prosjektslutt vil alle personopplysninger og lydopptak bli slettet, og data vil anonymiseres.

Dine(barnets) rettigheter

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

- innsyn i hvilke personopplysninger som er registrert om deg,
- å få rettet personopplysninger om deg,
- få slettet personopplysninger om deg,
- få utlevert en kopi av dine personopplysninger (dataportabilitet), og
- å sende klage til personvernombudet eller Datatilsynet om behandlingen av dine personopplysninger.

Hva gir oss rett til å behandle personopplysninger om deg (barnet)?

Vi behandler opplysninger om deg basert på ditt samtykke.

Deltakelse er frivillig, og alle elever står fritt til å trekke seg når de måtte ønske i løpet av prosjektperioden. Vedlagte svarslipp skal fylles inn og leveres tilbake til skolen innen onsdag 3. mars 2021.

Jeg håper dere vil godkjenne at jeg intervjuer eleven og bruker intervjuet i masteroppgaven.

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

UIT - Institutt for lærerutdanning og pedagogikk ved Sjur Schjelderup Hermansen
she020@uit.no
97732282.

Eller min veileder Kjersti Wæge
kjersti.wage@matematikkcenteret.no.

Vårt personvernombud:
Joakim Bakkevold
personvernombud@uit.no
776 46 322 og 976 915 78

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.

Vennlig hilsen

Sjur Schjelderup Hermansen

Sjur Schjelderup Hermansen

epost: she020@post.uit.no

Samtykkeerklæring, lydopptak av elevintervju

Elevens navn:

Foresatte

- Ja**, jeg/vi samtykker i at lydopptaket av barnet vårt i forbindelse med Master-prosjektet kan brukes slik det er skissert i skrevet.
- Nei**, jeg/vi samtykker IKKE i at lydopptaket av barnet vårt i forbindelse med Master-prosjektet kan brukes slik det er skissert i skrevet.

Dato: _____ Sted: _____

Signatur foresatte : _____

Elev

- Ja**, jeg samtykker i at lydopptaket av meg i forbindelse med Master-prosjektet kan brukes slik det er skissert i skrevet.
- Nei**, jeg samtykker IKKE i at lydopptaket av meg i forbindelse med Master-prosjektet kan brukes slik det er skissert i skrevet.

Dato: _____ Sted: _____

Signatur elev : _____

Vedlegg 3 intervjuguide

Intervjuguide:

Hvilke forestillinger kan elever ha om programmering i matematikk?

Fase 1: Rammesetting

1. Løst prat (+/-2 min)

- Uformell prat

2. Informasjon (3 min)

- Si litt om temaet for samtalen (bakgrunn, formål)
- Forklar hva intervjuet skal brukes til og forklar taushetsplikt og anonymitet
- Spør om noe er uklart og om respondenten har noen spørsmål
- Informer om opptak, sørg for samtykke til opptak
- Start opptak

Fase 2: Erfaringer

3. Overgangsspørsmål: (5 min)

Hva kjennetegner en skoletime med programmering?

Kan du beskrive en time dere har hatt der dere har programmert?

Fase 3: Fokusering

4. Nøkkelspørsmål: (10-15min)

- Synes du programmering hører til et bestemt fag på skolen?
- Hva tenker du programmering handler om? Hva ønsker man å oppnå gjennom programmering?
- Hva vil det si å være god i programmering?
- Hvilke egenskaper kjennetegner en elev som er flink å programmere?
- Når synes du at du lykkes i å programmering?
- Hvilke typer programmeringsoppgaver liker du å arbeide med? Programmering av «bil», lage blokkprogrammer for å løse

oppgaver?

- Når dere programmerer, hvordan organiserer læreren timen og klassen? Individuelt, grupper eller plenum
- Hvis du kunne bestemme over en time der klassen skulle jobbe med programmering hva skulle dere gjøre?
- Skulle du ønske dere brukte mer tid på programmering i klassen?
- Ser du noen unike muligheter ved bruk av programmering i matematikkfaget?
- Ser du noen likheter mellom å løse oppgaver i programmering og i matematikk?
- Synes du skolen har tilstrekkelig utstyr for å drive med programmering? Er det noe du savner?

Fase 4:
Tilbakeblikk

5. Oppsummering (ca. 3min)

- Oppsummere funn
- Har jeg forstått deg riktig?
- Er det noe du vil legge til?

