



Høgskulen på Vestlandet

Masteroppgave

M120UND509-MG-2021-VÅR2-FLOWassign

Predefinert informasjon

Startdato:	04-05-2021 09:00	Termin:	2021 VÅR2
Sluttdato:	18-05-2021 14:00	Vurderingsform:	Norsk 6-trinns skala (A-F)
Eksamensform:	Masteroppgave		
Flowkode:	203 M120UND509 1 MG 2021 VÅR2		
Intern sensor:	(Anonymisert)		

Deltaker

Kandidatnr.:	218
---------------------	-----

Informasjon fra deltaker

Antall ord *:	32879
----------------------	-------

Egenerklæring *: Ja

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

Jeg godkjenner avtalen om publisering av masteroppgaven min *

Ja

Er masteroppgaven skrevet som del av et større forskningsprosjekt ved HVL? *

Ja, LATAcME

Er masteroppgaven skrevet ved bedrift/virksomhet i næringsliv eller offentlig sektor? *

Nei



Høgskulen
på Vestlandet

MASTEROPPGAVE

Algoritmisk tenkning i matematikkopplæringen

Computational Thinking in School Mathematics

Martine Rekstad

Undervisningsvitenskap med fordypning i matematikk

Institutt for språk, litteratur, matematikk og tolkning

Nils Henry Williams Rasmussen

18.05.21

Jeg bekrefter at arbeidet er selvstendig utarbeidet, og at referanser/kildehenvisninger til alle

kilder som er brukt i arbeidet er oppgitt, jf. Forskrift om studium og eksamen ved Høgskulen på Vestlandet, § 12-1.

Forord

Fem år som student ved Høgskulen på Vestlandet avsluttes ved ferdigstillingen av denne oppgaven. Jeg føler meg heldig som har fått mulighet til å ha en så fin studietid hvor jeg har lært mye og møtt mange flotte mennesker.

Selv om dette prosjektet har vært et selvstendig arbeid, er det noen mennesker jeg kan gi noe av æren for at jeg har kommet meg i mål. Takk til de tre lærerne som gjorde undersøkelsen i denne studien mulig. Medstudenter som sendes en særlig stor takk, er gjengen på lesesalen. Den siste tiden av studietilværelsen ble fylt av gode samtaler, støtte, og latter som jeg er sikker på kommer til å forlenge livet mitt. Jeg vil også takke min familie og mine venner for tålmodighet og støtte.

Sist, men ikke minst, ønsker jeg å takke min veileder, Nils Henry Williams Rasmussen, for engasjement, motivasjon og god støtte gjennom hele prosessen fra idé til ferdig produkt. Takk for fine samtaler og god veiledning.

Martine Rekstad

Mai 2021

Sammendrag

Denne studien er et forskningsbasert arbeid som er gjennomført med utgangspunkt i matematikdidaktikk. Masteroppgaven er del av LATACME-prosjektet som er tilknyttet Høgskulen på Vestlandet, og har et overordnet tema om algoritmisk tenkning sin rolle i matematikkopplæringen. Studien har som formål å undersøke hvilke kompetanser algoritmisk tenkning har som intensjon å fremme i matematikkopplæringen.

Studien er av kvalitativ art som tar for seg tre ulike sider ved formålet og svarer på tre forskningsspørsmål. Det første forskningsspørsmålet undersøker hva som er bakgrunnen for implementeringen av algoritmisk tenkning i matematikkopplæringen. Det andre forskningsspørsmålet omhandler hvilke aspekter ved algoritmisk tenkning som kommer frem i tre læreres forståelse og oppfatning av algoritmisk tenkning. Det tredje forskningsspørsmålet ser på hvilke aspekter som kan komme frem i aktiviteter som er nevnt av de tre lærerne.

Aktuelle styringsdokumenter er undersøkt for å belyse intensjonen som ligger til grunn for algoritmisk tenkning i matematikkopplæringen. Deretter har jeg med godkjent søknad fra NSD, gjennomført kvalitativt forskningsintervju med tre matematikklærere om algoritmisk tenkning. For å undersøke hvilke aspekter ved algoritmisk tenkning som kommer frem i nevnte aktiviteter og de tre lærernes forståelse og oppfatning av algoritmisk tenkning, har jeg benyttet et rammeverk for algoritmisk tenkning som tar for seg kjerneferdigheter, tilnærminger og dimensjoner i algoritmisk tenkning.

Funn i studien viser at styringsdokumenter knytter intensjonen bak algoritmisk tenkning i matematikkopplæringen til krav om en økt generell forståelse for teknologiens rolle i samfunnet. Ved å implementere algoritmisk tenkning i matematikkopplæringen, kan man legge til rette for å utvikle nye ferdigheter tilpasset fremtidens behov. Videre viser resultater fra analysen at det er en overensstemmelse mellom aspekter ved algoritmisk tenkning som kommer frem i aktiviteter, og hva lærerne sier om algoritmisk tenkning. Studien avdekker imidlertid at det ikke er et tydelig samsvar mellom intensjonen bak algoritmisk tenkning i matematikkopplæringen, og hva som kommer fram i aktiviteter i matematikkopplæringen og i de tre lærernes forståelse og oppfatning av algoritmisk tenkning.

Abstract

This master's thesis is research-based within mathematics didactics. It is part of the LATAACME project which is affiliated with the University College of Western Norway. The overarching theme is the role of computational thinking within school mathematics.

The aim of the thesis is to investigate which competencies computational thinking has as intention to promote within school mathematics. There are three main questions asked in this master's thesis. The first research question examines the background for the implementation of computational thinking in school mathematics. The second research question assesses which aspects of computational thinking that emerge in three teachers' understandings and perceptions of computational thinking. Finally, the third research question evaluates the aspects that might emerge through activities addressed by the three teachers.

In order to illuminate the research questions, a qualitative approach is utilized. Relevant reports have been examined to investigate the intention of computational thinking in mathematics. Subsequently, with an approved application from NSD, a qualitative research interview was conducted with three mathematics teachers about computational thinking.

In order to investigate which aspects of computational thinking might emerge in the three teachers' individual understandings and perceptions of computational thinking, a framework for computational thinking was used. This framework addresses core skills, approaches, and dimensions, in which computational thinking was used.

Findings in this master's thesis show that the reports link the intention with computational thinking in school mathematics to expectations of a more general understanding of the role of technology in society. Computational thinking in school mathematics can facilitate development of skills and competencies as well as qualifying for the future needs of the society. Computational thinking plays a significant role in this process.

Results from the analysis show that there is a relationship between aspects of computational thinking that emerge in activities, and what the teachers state about computational thinking. However, the study reveals that there is no apparent correspondence between the intention of computational thinking in school mathematics, and what emerges through activities in mathematics, and in the three teachers' understandings and perceptions of computational thinking.

Innholdsfortegnelse

FORORD	I
SAMMENDRAG	II
ABSTRACT	III
1 INNLEDNING	8
1.1 BAKGRUNN FOR VALG AV TEMA	8
1.2 STUDIENS FORMÅL OG FORSKNINGSSPØRSMÅL.....	10
1.3 TIDLIGERE FORSKNING.....	12
1.4 DISPOSISJON.....	14
2 ALGORITMISK TENKNING	15
2.1 DEL 1	15
2.1.1 <i>Utdanningsdirektoratet om algoritmisk tenkning</i>	15
2.1.2 <i>Eksisterende rammeverk for algoritmisk tenkning</i>	19
2.1.3 <i>Rammeverkene sett i et komparativt lys</i>	32
2.2 DEL 2	34
2.2.1 <i>Styringsdokumenter om algoritmisk tenkning</i>	34
2.3 OPPSUMMERING AV ALGORITMISK TENKNING	37
3 TEORETISK RAMMEVERK FOR ALGORITMISK TENKNING	39
3.1 RAMMEVERK FOR ALGORITMISK TENKNING.....	39
3.1.1 <i>Kjerneferdigheter i algoritmisk tenkning</i>	40
3.1.2 <i>Tilnærminger i algoritmisk tenkning</i>	42
3.1.3 <i>Dimensjoner av algoritmisk tenkning</i>	44
4 METODISK TILNÆRMING	45
4.1 KVALITATIV TILNÆRMING	45
4.2 KVALITATIVT FORSKNINGSINTERVJU.....	46
4.3 UTARBEIDELSE AV INTERVJUGUIDE	46
4.3.1 <i>Intervjuspørsmålene</i>	47
4.4 UTVALG	48
4.5 GJENNOMFØRING AV INTERVJU.....	50
4.5.1 <i>Lydopptak</i>	50
4.5.2 <i>Intervjuene</i>	51
4.6 DATAMATERIALET	52
4.6.1 <i>Øvrige dokumenter i datamaterialet</i>	52
4.7 KVALITATIV ANALYSEMETODE.....	53
4.7.1 <i>Transkripsjon</i>	53
4.7.2 <i>Oversiktsanalyse</i>	54

4.7.3	<i>Deduktiv innholdsanalyse</i>	55
4.8	STUDIENS VALIDITET OG RELIABILITET.....	58
4.9	FORSKNINGSETISKE BETRAKTNINGER OG HENSYN.....	61
5	RESULTATER	62
5.1	OVERSIKTSANALYSE.....	62
5.1.1	<i>Algoritmisk tenkning</i>	63
5.1.2	<i>Aktiviteter i matematikkoppleringen</i>	65
5.2	RESULTATER FRA ØVRIGE DOKUMENTER I DATAMATERIALET.....	70
5.2.1	<i>Planleggingsdokument</i>	70
5.2.2	<i>Aktivitet i Scratch – Hvordan lage en løkke</i>	70
5.2.3	<i>Aktivitet med Micro:bit fra Kidsakoder.no</i>	71
5.2.4	<i>Dokument fra faggruppen i matematikk</i>	72
5.3	OPPSUMMERING AV RESULTATER.....	75
5.3.1	<i>Algoritmisk tenkning</i>	75
5.3.2	<i>Aktiviteter i matematikkoppleringen</i>	76
6	ANALYSE	80
6.1	ALGORITMISK TENKNING.....	80
6.1.1	<i>Kjerneferdigheter i algoritmisk tenkning</i>	80
6.1.2	<i>Tilnæringer i algoritmisk tenkning</i>	82
6.1.3	<i>Dimensjoner av algoritmisk tenkning</i>	83
6.1.4	<i>Forekomst av aspekter ved algoritmisk tenkning i de tre lærernes forståelse og oppfatning av algoritmisk tenkning</i>	84
6.2	AKTIVITETER I MATEMATIKKOPPLÆRINGEN.....	85
6.2.1	<i>Aktiviteter med teknologi</i>	87
6.2.2	<i>Aktiviteter uten teknologi</i>	90
6.2.3	<i>Forekomst av aspekter ved algoritmisk tenkning i aktiviteter i matematikkoppleringen</i>	93
7	DISKUSJON	94
7.1	BAKGRUNNEN FOR IMPLEMENTERING AV ALGORITMISK TENKNING I MATEMATIKKOPPLÆRINGEN.....	94
7.2	ASPEKTER VED ALGORITMISK TENKNING I LÆRERNES FORSTÅELSE OG OPPFATNING AV ALGORITMISK TENKNING.....	95
7.3	ASPEKTER VED ALGORITMISK TENKNING I AKTIVITETER I MATEMATIKKOPPLÆRINGEN.....	97
7.4	SAMMENHENG MELLOM AKTIVITETER I MATEMATIKKOPPLÆRINGEN OG LÆRERNES FORSTÅELSE OG OPPFATNING AV ALGORITMISK TENKNING.....	98
8	AVSLUTNING	101
8.1	KONKLUSJON.....	101
8.2	VIDERE FORSKNING.....	103
8.3	AVSLUTTENDE KOMMENTAR.....	104

9	LITTERATURLISTE	105
	VEDLEGG	107
	VEDLEGG 1 – INFORMASJONSSKRIV OG SAMTYKKEERKLÆRING	108
	VEDLEGG 2 – REFERANSE OM GODKJENNING FRA NSD	111
	VEDLEGG 3 – INTERVJUGUIDE.....	112

Figuroversikt

Figur 2-1: Figur hentet med tillatelse fra Utdanningsdirektoratet (2019c)	
https://www.udir.no/kvalitet-og-kompetanse/profesjonsfaglig-digital-kompetanse/algorithmisk-tenkning/	17
Figur 2-2: Figur med OGL license (åpen lisens). Fra Barefoot Computing (u.å. a)	
https://www.barefootcomputing.org/concept-approaches/computational-thinking-concepts-and-approaches?CatItemId=computational-thinking-concepts-and-approaches	17
Figur 2-3:Egenkomponert illustrasjon av algorithmisk tenkning, programmering og koding...	25
Figur 2-4: Skjermdump fra CompuThink-rapport (Bocconi et al., 2016, s. 17)	29
Figur 2-5: Skjermdump som viser tabell fra CompuThink-rapport (Bocconi et al., 2016, s. 19)	
.....	29
Figur 2-6: Oversikt over Brennan og Resnick (2012) sitt rammeverk for algorithmisk tenkning	
.....	33
Figur 2-7: Oversikt over Bocconi et al. (2016;2018) sitt rammeverk for algorithmisk tenkning	
.....	33
Figur 3-1: Oversikt over aspekter ved algorithmisk tenkning.....	40
Figur 4-1: oversikt over analyseprosess	53
Figur 5-1: Oversikt over kontekster knyttet til algorithmisk tenkning i de tre intervjuene.....	62
Figur 5-2: Skjermdump fra video som Lærer 1 har laget om løkke i Scratch.....	71
Figur 5-3: Skjermdump av eksempel på aktivitet på Kidsakoder.no (u.å.)	
https://oppgaver.kidsakoder.no/microbit/pxt_gangespill/gangespill	71
Figur 6-1: oversikt over utvalgte aktiviteter som skal analyseres	86

Tabelloversikt

Tabell 2-1: Oversikt over relevante kompetansemål hentet fra læreplanen i matematikk (Utdanningsdirektoratet, 2019a).....	19
Tabell 2-2: Oversikt over litteratursøk	20
Tabell 2-3: Egenkomponert oversikt over Bocconi et al. (2016,2018) sitt rammeverk for algoritmisk tenkning.....	31
Tabell 5-1: Egenkomponert oversikt over kompetansemål og forslag til aktiviteter som vises i dokumentet til Lærer 3	73
Tabell 5-2: Oversikt over foreslåtte aktiviteter med teknologi på de ulike trinnene på skolen til Lærer 3	77
Tabell 5-3: Oversikt over foreslåtte aktiviteter uten teknologi på de ulike trinnene på skolen til Lærer 3	79
Tabell 6-1: Oversikt over aspekter ved algoritmisk tenkning som kommer frem i intervju med lærerne	85
Tabell 6-2: Aspekter ved algoritmisk tenkning i Scratch.....	88
Tabell 6-3: Aspekter ved algoritmisk tenkning i Micro:bit.....	89
Tabell 6-4: Oversikt over aspekter ved algoritmisk tenkning i Kongen befaler	91
Tabell 6-5: Oversikt over aspekter ved algoritmisk tenkning i Rødt lys.....	92
Tabell 6-6: Oversikt over forekomst av aspekter ved algoritmisk tenkning i aktiviteter i matematikkopplæringen	93
Tabell 7-1: Oversikt over forekomst av aspekter ved algoritmisk tenkning i aktiviteter og lærernes forståelse og oppfatning av algoritmisk tenkning	99

1 Innledning

1.1 Bakgrunn for valg av tema

Samfunnet vårt er i stadig endring. Arbeidsmarkedet, både globalt og lokalt, endrer seg i takt med samfunnet, og har stilt krav til økt behov for spesialisering og høykvalifisert arbeidskraft (OECD, 2019, s. 3). Kompetansepolitikken har hatt en prioritert plass i norsk politikk i de senere år, i håp om å styrke rekruttering av fremtidige samfunnsborgere til en stabil og aktiv tilknytning til arbeidslivet (NOU 2020:2, s. 13). Kompetansebehovutvalget (KBU) ble dannet med formål om å formidle en faglig vurdering av hva det fremtidige Norge sitt kompetansebehov vil være. I Norges offentlige utredninger sin rapport skriver de at Norge er inne i det mange kaller den fjerde industrielle revolusjon (NOU: 2020:2, s. 13). Den fjerde industrielle revolusjonen krever utvikling og nye ferdigheter av dagens samfunnsborgere, især den yngre generasjonen som skal bære samfunnet videre inn i en trygg og bedre fremtid. Den fjerde industrielle revolusjonen kjennetegnes av nye forretningsmodeller, vesentlig grad av automatisering og digitalisering (Schwab, 2017, s. 10). Ved siden av den teknologiske utviklingen, er også andre faktorer som klimaendringer og demokratisk utvikling som er sentrale drivere for disse ferdighetene (NOU: 2020:2, s. 13). I den forbindelse har mange land valgt å lage nye læreplaner og forme utdanningen slik at den samsvarer med de nye ferdighetene og kompetansene som den fjerde industrielle revolusjonen krever. Sverige, Finland, Frankrike, England og Irland er blant flere land som har fornyet læreplaner hvor de har plassert programmering og algoritmisk tenkning som prioriterte temaer i læreplanene (Bocconi et al., 2018, s. 1).

Norge føyer seg også inn i rekken i håp om å forme og utvikle gode fremtidsborgere som passer til ferdighetene og kompetansene som kreves. Algoritmisk tenkning og programmering er nye elementer i matematikkopplæringen som kom med den nye læreplanen som trådte i kraft august 2020. Den nye læreplanen sier at matematikk skal forberede elevene på et samfunn og arbeidsliv i utvikling ved å gi dem kompetanse i utforskning og problemløsning (Utdanningsdirektoratet, 2019a). Å lære elever å programmere vil åpne for muligheter til å kunne forme og utvikle egen teknologi, noe som vil være av stor betydning i en digital fremtid (Forsström & Kaufmann, 2018, s. 19). I forbindelse med innføring av algoritmisk tenkning og programmering i matematikkopplæringen, etterlyses det mer forskning som kan fortelle mer om rammefaktorer, forhold og kvalitet på programmering og algoritmisk

tenkning i matematikkfaget i skolen (Forsström & Kaufmann, 2018, s. 30; Grover & Pea, 2013, s. 42; Weintrop et al., 2016, s. 143).

I en undervisningskontekst vil det ikke være tilstrekkelig å bare bruke teknologi for å forbedre elevenes ytelse (OECD, 2019, s. 179). Man er altså avhengig av hvordan teknologien integreres i undervisningen. Digitale ferdigheter var en av de fem grunnleggende ferdighetene i Kunnskapsløftet som kom i 2006, og ifølge Norges offentlige utredninger (2013:2, s. 99) er det gjort mye bra i innføringen av digitale ferdigheter i skolen. Det vises likevel til noen kritikkverdige forhold som kan være relevant for denne studien. Digitutvalget mener at operasjonaliseringen av digitale ferdigheter har hatt et for lite fokus på forståelsen av selve essensen av hvordan IKT fungerer og av teknologisamfunnet. Utvalget argumenterer for at det har vært et ensidig fokus på et konsumbruk av teknologi, og for lite fokus på den kritiske refleksjonen rundt teknologisamfunnet og innovasjon i teknologi. Digitale ferdigheter innehar en ubalanse hvor digitale ferdigheter er blitt definert med for stor vekt på kommunikasjon, tekst og det humanistiske, og for lite vekt på algoritmer, matematikk og teknologi, mener utvalget (NOU:2013:2, s. 99). Videre hevder forfatterne av NOU-rapporten at vi med dette fokuset kun vil utdanne forbrukere, og ikke skapere (NOU:2013:2, s. 105). Utvalget uttrykker at det er bra at elever lærer seg å uttrykke seg godt i digitale sammensatte tekster, men vi trenger også en befolkning som forstår hva som foregår bak skjermene (2013:2, s. 102).

Basert på den nye læreplanen som trådte i kraft høsten 2020, kan det se ut som nettopp dette fokuset er i ferd med å endre seg, om det ikke allerede har det. Algoritmisk tenkning og programmering er kommet inn i den nye læreplanen, og digitale ferdigheter har omfavnet evnen til å være kreativ og skapende som en del av sin definisjon (Kunnskapsdepartementet, 2017). Man kan stille seg spørrende til om programmering og algoritmisk tenkning i matematikkopplæringen kan være en del av løsningen på kravet om utvikling av nye ferdigheter for fremtiden. Matematikkopplæringens utvikling vil spille en sentral rolle for hvilke ferdigheter fremtidens borgere vil inneha når de skal ut i arbeidslivet i en digital fremtid.

I denne studien er det overordnede temaet algoritmisk tenkning sin rolle i matematikkopplæringen. Bakgrunnen for valget av dette temaet bunner i behov for forskning knyttet til rammefaktorer, forhold og kvalitet på algoritmisk tenkning i didaktiske praksiser. Studien er skrevet som en del av forskningsprosjektet Learning about teaching argumentation for critical mathematics education in multilingual classrooms, - LATAcME. Prosjektet er

knyttet til Høgskulen på Vestlandet (HVL), og har tre underprosjekter hvor min studie er en del av underprosjektet Argumentasjon og IKT. Forskningsprosjektet har som mål å få innsikt i hva som fremmer eller hemmer lærerstudenter i å lære om å undervise i argumentasjon for kritisk matematikk for elever i flerspråklige klasserom på barnetrinnet (Høgskulen på Vestlandet, 2019).

1.2 Studiens formål og forskningsspørsmål

Som nevnt innledningsvis, er det behov for mer forskning som undersøker kvalitet, ramme faktorer og forhold i programmering og algoritmisk tenkning i matematikkopplæringen. I denne studien ønsker jeg å undersøke algoritmisk tenkning sin rolle i matematikkopplæringen i søken etter ramme faktorer og forhold i algoritmisk tenkning i matematikkopplæringen. Studiens overordnede formål er å undersøke hvilke kompetanser algoritmisk tenkning har som intensjon å fremme i matematikkopplæringen. Forskningsspørsmålene som nå vil bli presentert, tar for seg tre ulike sider ved dette formålet.

Forskingsspørsmål 1:

Hva er bakgrunnen for implementering av algoritmisk tenkning i matematikkopplæringen?

Forskingsspørsmål 2:

Hvilke aspekter ved algoritmisk tenkning kommer frem i tre læreres forståelse og oppfatning av algoritmisk tenkning?

Forskingsspørsmål 3:

Hvilke aspekter ved algoritmisk tenkning kan komme frem i nevnte matematiske aktiviteter hos de tre lærerne?

Det første forskningsspørsmålet knyttes til hva som ligger til grunn for implementeringen av algoritmisk tenkning i matematikkopplæringen. Jeg vil da undersøke hvilke kompetanser og hvilken forståelse som er tilsiktet gjennom det Kunnskapsdepartementet ønsker å fremme

gjennom den nye læreplanen i matematikkopplæringen. Å se på hvorfor algoritmisk tenkning er innført i matematikkopplæringen, vil bidra til å få en helhetlig innsikt i intensjonen bak algoritmisk tenkning og dens plass i matematikkfaget i den nye læreplanen.

Det andre forskningsspørsmålet tar utgangspunkt i intensjonen bak algoritmisk tenkning i matematikkopplæringen, og undersøker tre matematikklæreres oppfatning og forståelse av algoritmisk tenkning i matematikkopplæringen. Å få innsikt i kompetanse og forståelse lærere selv mener at algoritmisk tenkning skal føre med seg, er et viktig aspekt i implementeringen av algoritmisk tenkning i matematikkopplæringen. Ved å inkludere hvilken oppfatning og hvilken forståelse lærerne har av algoritmisk tenkning, og ikke bare konkret hva de sier de gjør eller har gjort med algoritmisk tenkning i matematikkundervisningen sin, vil man få et mer sammensatt bilde av implementeringen av algoritmisk tenkning i matematikkopplæringen.

Forskningsspørsmål 3 har som hensikt å undersøke hvilke kompetanser og hvilken forståelse som kan komme frem gjennom spesifiserte aktiviteter i matematikkopplæringen. Når vi skal se på aktiviteter i matematikkopplæringen som blir nevnt i intervju, kan det være utfordrende å klassifisere en intensjon med aktiviteten. Det blir derfor mer nøyaktig å undersøke hvilke aspekter ved algoritmisk tenkning som kommer frem i aktiviteter i matematikkopplæringen.

Implementering av algoritmisk tenkning har foregått i takt med innføring av den nye læreplanen samtidig som denne studien ble utført. Det er derfor naturlig at mengden av aktiviteter som kan knyttes til algoritmisk tenkning som er planlagt og utført, er begrenset. Likevel ønsker jeg å se etter aspekter ved algoritmisk tenkning i aktiviteter i matematikkopplæringen, fordi det vil bidra til et mer sammensatt bilde av implementeringen av algoritmisk tenkning og dens rolle i matematikkopplæringen.

Studiens fokus på algoritmisk tenkning sin rolle i matematikkopplæringen har som hensikt å gi innsikt i rammefaktorer og forhold i algoritmisk tenkning i matematikkopplæringen. Intensjonen bak algoritmisk tenkning i matematikkopplæringen knyttes til rammefaktorer, og forhold beskrives som aspekter ved algoritmisk tenkning som kan komme frem i aktiviteter og lærernes forståelse og oppfatning av algoritmisk tenkning. Synliggjøring av ulike aspekter ved algoritmisk tenkning kan gi et bilde av hvordan algoritmisk tenkning kan spille seg ut i matematikkopplæringen. Jeg skal i neste delkapittel gjøre rede for relevant tidligere forskning.

1.3 Tidligere forskning

Formålet med å gjøre rede for tidligere forskning er å undersøke relevant forskning som er blitt gjort på læreres oppfatning og forståelse av aspekter av algoritmisk tenkning, og plassere min studie i forskningslandskapet til algoritmisk tenkning i matematikkopplæringen.

Synliggjøring av tidligere forskning om algoritmisk tenkning, vil også gjøre det mulig å se hva min studie kan tilføye forskningslandskapet til algoritmisk tenkning.

Store deler av forskning knyttet til algoritmisk tenking omhandler diskusjoner om definisjoner av algoritmisk tenkning. Som nevnt tidligere, påpeker flere forskere at det mangler empirisk forskning som tar for seg programmering og algoritmisk tenkning i didaktiske praksiser (Forsström & Kaufmann, 2018, s. 30; Grover & Pea, 2013, s. 42; Weintrop et al., 2016, s. 143).

Jeg har forsøkt å finne forskning som undersøker hvordan lærere implementerer algoritmisk tenkning i matematikkopplæringen, og hva deres oppfatning av algoritmisk tenkning er. Det nærmeste jeg kommer formålet i min studie i tidligere forskning, er forskning som har sin hovedvekt på programmering.

I forbindelse med programmering blir algoritmisk tenkning ofte nevnt. Både fordeler og ulemper ved implementering av algoritmisk tenking og programmering i matematikkopplæringen har også blitt diskutert (Forsström & Kaufmann, 2018; Weintrop et al., 2016). Forsström og Kaufmann (2018, s. 19) uttrykker at når man integrerer programmering i et eksisterende fag, som matematikk, stiller det helt nye krav til læreren dersom læreren ikke har tidligere erfaringer eller kunnskap med programmering. En av konklusjonene i deres studie, er at fremtiden krever at vi må tilpasse oss for å kunne håndtere endringene som følger med. Algoritmisk tenkning, sammen med kritisk tenkning og tverrfaglighet, er blant annet det som kreves i den tilpasningen.

Det finnes også forskning som gir forslag og eksempler på undervisningsopplegg knyttet til programmering i matematikk, hvor elever har blitt intervjuet (Brennan & Resnick, 2012). I denne forskningen var fokuset å danne et rammeverk bestående av tre dimensjoner av algoritmisk tenkning. Det finnes også en studie som undersøkte forskjeller i utviklingen til algoritmisk tenkning mellom elever som deltok i en serie aktiviteter som ikke inkluderte teknologi, såkalte «unplugged-activities» (Brackmann et al., 2017). Lærere var ikke en del av

hovedfokuset i studien, men i bakgrunnen for studien kommer det frem at lærere finner nytte i slike aktiviteter.

Kong, Lai og Sun (2020), forskere fra The Education University of Hong Kong, ser på forståelse og vurdering av algoritmisk tenkning hos lærere. De peker på manglende empirisk forskning av god kvalitet, som tar for seg lærerutvikling i algoritmisk tenkning sammenlignet med programmering. Dette mener de er bekymringsverdig. Deres studie viser at flere lærere ikke er i stand til å undervise i algoritmisk tenkning av flere ulike årsaker. Eksempelvis mangelfull utdanning, og liten eller ingen erfaring knyttet til algoritmisk tenkning eller programmering. Studien deres konkluderer med at forskning viser at etterutdanning i algoritmisk tenkning og programmering for lærere hovedsakelig består av kurs hvor det er begrenset deltakelse og rom for refleksjon.

I en del forskning har algoritmisk tenkning fungert som et bakteppe i studien samtidig som ulike aktiviteter knyttet til matematikk og programmering er undersøkt (Brennan & Resnick, 2012; Brackmann et al., 2017). Forrström og Kaufmann (2018, s. 26) undersøker hva som er det pedagogiske spørsmålet ved programmering i matematikkopplæringen, og nevner algoritmisk tenkning sammen med kritisk tenkning og tverrfaglighet som en del av en tilpasning etter fremtidens krav.

Sammen med tidligere forskning som er blitt gjort rede for over, vil denne studien forhåpentligvis bidra til økt forståelse og kunnskap om algoritmisk tenkning i matematikkopplæringen. I skrivende stund har den nye læreplanen i matematikk vært i matematikkopplæringen i under ett år. Denne studien gir innsikt i introduksjonsfasen til algoritmisk tenkning i matematikkopplæringen hos tre lærere. Ved å se på lærernes oppfatning og forståelse av algoritmisk tenkning, kan man argumentere for at man får et bilde av potensialet som finnes i implementeringen av algoritmisk tenkning i matematikkopplæringen.

1.4 Disposisjon

Studien er organisert i ni kapitler. [Det første kapittelet](#) inneholder bakgrunn for valg av tema, studiens formål og forskningsspørsmål, tidligere forskning og disposisjon for oppgaven.

[Kapittel 2](#) har som formål å gi innsikt i hva algoritmisk tenkning er, og hva som ligger til grunn for at det er implementert i matematikkopplæringen. Kapittelet er delt i to deler, hvorav [første del](#) består av Utdanningsdirektoratets forklaring av algoritmisk tenkning, samt en oversikt over forskning på algoritmisk tenkning. I [del 2](#) i kapittel 2, tar jeg for meg hva styringsdokumenter sier om bakgrunnen for hvorfor algoritmisk tenkning er implementert i matematikkopplæringen. Hensikten med del 2 er å synliggjøre utgangspunktet til denne studien.

I [kapittel 3](#) vil jeg gå nærmere inn på studiens utgangspunkt som er rammeverket for algoritmisk tenkning. Rammeverket er et konstruert verktøy som er anvendt i analysearbeidet. I arbeidet med analysen har rammeverket for algoritmisk tenkning vært avgjørende for vurderingen av intervjuenes innhold.

I [kapittel 4](#) blir metodiske valg jeg har tatt i underveis i arbeidsprosessen gjennomgått, samtidig som nødvendige hensyn til valg og avgjørelser med potensiell påvirkning på studiets resultater blir belyst.

I [kapittel 5](#) presenteres resultater fra analysen, mens [kapittel 6](#) ser de innsamlede dataene i lys av rammeverket for algoritmisk tenkning som er gjort rede for i kapittel 3.

I [kapittel 7](#) diskuterer jeg funn fra analysen og drøfter dem i lys av studiens rammeverk, teoretisk forankring og relevant litteratur som blir nevnt i denne studien. Jeg har valgt å holde diskusjonen separert fra analysen. Dette er av hensyn til oppgavens struktur, hvor jeg syns det faller naturlig å diskutere resultater fra analysen opp mot teori og tidligere forskning etter at resultater og funn fra analysen er presentert.

Avslutningsvis i [kapittel 8](#), kommer en konklusjon hvor jeg også presenterer tanker rundt videre forskning på algoritmisk tenkning i matematikkopplæringen. Til slutt runder jeg av med en avsluttende kommentar.

2 Algoritmisk tenkning

Temaet i denne studien er algoritmisk tenkning sin rolle i matematikkopplæringen. Det er derfor relevant å utforske bakgrunnen for at det er tatt inn i matematikkopplæringen i skolen. I den sammenheng blir det aktuelt å utdype hva algoritmisk tenkning er, og hvilken plass den har fått i matematikkopplæringen.

Dette kapitlet har som hensikt å belyse hva algoritmisk tenkning er, nærmere sagt vise hva som blir sagt om algoritmisk tenkning, og som kan benyttes som et utgangspunkt i denne studien. Kapitlet er delt opp i to deler, hvorav del 1 omhandler Utdanningsdirektoratets uttalelser om algoritmisk tenkning, og forskningslandskapet uttalelser om ulike definisjoner, klassifiseringer og dimensjoner av algoritmisk tenkning. Del 2 inneholder en redegjørelse av styringsdokumenters uttalelser om bakgrunnen for at algoritmisk tenkning er tatt inn i matematikkopplæringen. Hensikten med å inkludere styringsdokumenter, er å begrunne hvilket utgangspunkt jeg har valgt å benytte i denne studien.

2.1 Del 1

2.1.1 Utdanningsdirektoratet om algoritmisk tenkning

I Utdanningsdirektoratets (2019c) artikkel om algoritmisk tenkning står det at det finnes flere ulike definisjoner av algoritmisk tenkning, men at hovedtrekkene er sammenfallende. I samme artikkel defineres algoritmisk tenkning som følgende:

Algoritmisk tenkning innebærer å bryte ned komplekse problem til mindre, mer håndterlige delproblemer som lar seg løse. Det inkluderer å organisere og analysere informasjon på en logisk måte og å lage fremgangsmåter (algoritmer) for å komme frem til ønsket løsning. Det handler også om å lage abstraksjoner og modeller av den virkelige verden ved å fjerne unødvendige detaljer og fokusere på det som er relevant for den aktuelle problemstilling og løsning. En løsning på et spesifikt problem kan ofte generaliseres, slik at den kan brukes til å løse lignende problemer, og løsninger på flere delproblemer kan kombineres for å løse mer komplekse problem.

(Utdanningsdirektoratet, 2019c)

Utdanningsdirektoratet (2019c) sier at algoritmisk tenkning er viktig i prosessen med å utvikle strategier og fremgangsmåter for å løse problemer. I artikkelen presiseres det videre at begrepet algoritmisk tenkning er den norske oversettelsen av *computational thinking*. Videre

har de presentert en tilpasset figur (se figur 2-1) som opprinnelig er fra Barefoot Computing (UK).

Det kommer ikke tydelig frem nøyaktig fra hvor hos Barefoot Computing Utdanningsdirektoratet har hentet informasjonen som kan knyttes til denne figuren i figur 2-1, men etter undersøkelser har jeg kommet frem til at figuren er lik en forenklet plakat (figur 2-2) som er laget i forbindelse med informasjon knyttet til algoritmisk tenking (på engelsk: «computational thinking»).

På nettsiden hvor vi finner plakaten, vises informasjon om «concepts» og «approaches» knyttet til algoritmisk tenkning. På den nettsiden kan man trykke seg inn på hver og en underkategori knyttet til både «concepts» og «approaches». Når man trykker seg inn på disse underkategoriene, får man informasjon om hva det er, hvorfor det er viktig, og hvordan det ser ut i den engelske læreplanen. I tillegg er det forslag til aktiviteter tilknyttet hver enkeltunderkategori. Nederst på nettsiden finnes lenker til mer informasjon om algoritmisk tenkning (Barefoot Computing, u.å. a). En tilsvarende funksjon som på Barefoot Computings nettside finnes ikke hos Utdanningsdirektoratet. Utdanningsdirektoratet oppgir heller ikke en klar henvisning til akkurat denne nettsiden hos Barefoot Computing, fra Utdanningsdirektoratet, utover at det påpekes at figuren hos Utdanningsdirektoratet er tilpasset fra Barefoot Computing (UK).

Jeg har plassert begge figurene fra Utdanningsdirektoratet (figur 2-1) og Barefoot Computing (figur 2-2) sammen, slik at det blir lettere å sammenligne likhetstrekkene.



FIGUR 2-1: FIGUR HENTET MED TILLATELSE FRA UTDANNINGSDIREKTORATET (2019C) [HTTPS://WWW.UDIR.NO/KVALITET-OG-KOMPETANSE/PROFESJONSFAGLIG-DIGITAL-KOMPETANSE/ALGORITMISK-TENKNING/](https://www.udir.no/kvalitet-og-kompetanse/profesjonsfaglig-digital-kompetanse/algoritmisk-tenkning/)

Concepts and Approaches

Computational thinking involves six different concepts and five approaches to working.

Click on the icons below to find out more about computational thinking concepts.



Click on the icons below to find out more about computational thinking approaches.



FIGUR 2-2: FIGUR MED OGL LICENSE (ÅPEN LISENS). FRA BAREFOOT COMPUTING (U.Å. A) [HTTPS://WWW.BAREFOOTCOMPUTING.ORG/CONCEPT-APPROACHES/COMPUTATIONAL-THINKING-CONCEPTS-AND-APPROACHES?CATITEMURL=COMPUTATIONAL-THINKING-CONCEPTS-AND-APPROACHES](https://www.barefootcomputing.org/concept-approaches/computational-thinking-concepts-and-approaches?CATITEMURL=COMPUTATIONAL-THINKING-CONCEPTS-AND-APPROACHES)

Barefoot Computing er utviklet av lærere, og støttet av forskere, og arbeider for å gi lærere tilgang til ressurser til undervisning, ifølge deres nettside (Barefoot Computing, u.å. b). På nettsiden fremgår det at et av Barefoot Computings formål er å gi barn teknologiske

ferdigheter de trenger for fremtiden innen 2025. Dette gjøres sammen med samarbeidspartnere som BT Group, et stort multinasjonalt telekommunikasjonsselskap, Computing at School, et satsningsprogram i regi av The Chartered Institute of IT. Dette samarbeidet er støttet av flere aktører i Storbritannia, som Department for Education, Openreach, Education Scotland, samt Welsh Government Education (Barefoot Computing, u.å. b).

Frem til nå har jeg vist Utdanningsdirektoratets definisjon av algoritmisk tenkning, og fulgt deres spor tilbake til hva de har valgt å ta utgangspunkt i. Videre skal jeg se nærmere på algoritmisk tenkning i den nye læreplanen i matematikk.

2.1.1.1 Den nye læreplanen i matematikk

Den nye læreplanen startet sin innføring av nytt innhold i skolefagene skoleåret 2020/2021 (Utdanningsdirektoratet, 2020a). For barneskolen, altså fra første til syvende trinn, innebærer dette nytt innhold i alle skolefag. Jeg skal nå se nærmere på algoritmisk tenkning i den nye læreplanen i matematikk.

Som tidligere nevnt, sier den nye læreplanen i matematikk under fagrelevans og verdier at matematikk skal forberede elevene på et samfunn og arbeidsliv i utvikling ved å gi dem kompetanse i utforskning og problemløsning. Dette kan ses i sammenheng med behov for ferdigheter og kompetanse som kreves i det 21. århundre.

Det er også utviklet kjerneelementer, som er det viktigste faglige innholdet elever skal arbeide med i matematikkopplæringen (Utdanningsdirektoratet, 2019b). Ifølge Utdanningsdirektoratet må elevene lære seg kjerneelementene, som består av sentrale begreper, metoder, tenkemåter, kunnskapsområder og uttrykksformer for å mestre og anvende faget. Her vil kjerneelementet Utforskning og problemløsning være sentralt for algoritmisk tenkning og programmering. Utdanningsdirektoratet (2019b) beskriver algoritmisk tenkning i kjerneelementet Utforskning og problemløsning som en viktig del i prosessen med å utvikle strategier og fremgangsmåter for å løse problemer og innebærer å bryte ned et problem i delproblemer som kan løses systematisk. Videre skriver de at det innebærer å vurdere om delproblemene kan løses på best mulig måte med eller uten digitale verktøy.

2.1.1.2 Kompetansemål i den nye læreplanen i matematikk

Under vises tabell 2-1 med relevante kompetansemål fra fjerde til syvende trinn. Det er ikke formulert noe tydelig om programmering eller algoritmisk tenkning i undervisvurdering på fjerde- og femtetrinnet.

TABELL 2-1: OVERSIKT OVER RELEVANTE KOMPETANSEMÅL HENTET FRA LÆREPLANEN I MATEMATIKK (UTDANNINGSDIREKTORATET, 2019A)

Trinn	Kompetansemål	Undervisvurdering
Fjerde trinn	«lage algoritmar og uttrykkje dei ved bruk av variablar, vilkår og lykkjer»	
Femte trinn	«lage algoritmar og uttrykkje dei ved bruk av variablar, vilkår og lykkjer»	
Sjette trinn	«bruke variablar, lykkjer, vilkår og funksjonar i programmering til å utforske geometriske figurar og mønster».	«... læreren skal være i dialog med elevene om utviklingen deres i programmering og geometri».
Syvende trinn	«bruke programmering til å utforske data i tabellar og datasett».	«... læreren skal være i dialog med elevene om deres utvikling i programmering og strategier for å løse problemer».

Frem til nå har vi sett på hva Utdanningsdirektoratet sier om algoritmisk tenkning, og hva som står i den nye læreplanen i matematikk. Jeg skal nå gå nærmere inn på forskningslandskapet til algoritmisk tenkning og se på definisjoner og resultater fra relevant forskning.

2.1.2 Eksisterende rammeverk for algoritmisk tenkning

Med unntak av noen masteroppgaver og artikler i tidsskriftet Tangenten, har det vært utfordrende å finne norsk litteratur og forskning om rammeverktøy eller klassifiseringsverktøy som gjør det mulig å identifisere aspekter ved algoritmisk tenkning i

matematikkopplæringen. Forskning som omhandler programmering i en undervisningskontekst finnes det en del av, og da er gjerne algoritmisk tenkning nevnt som en mindre del av forskningen. Da algoritmisk tenkning er et relativt nytt begrep i en utdanningskontekst, vil det være en fordel å inkludere internasjonal forskning om algoritmisk tenkning. Utdanningsdirektoratet har som nevnt over i [2.1.1 Utdanningsdirektoratet om algoritmisk tenkning](#), presisert i deres artikkel at algoritmisk tenkning er den norske oversettelsen av *computational thinking*. Jeg har derfor sett det nødvendig å inkludere det engelske begrepet i søket etter relevant forskning. Jeg har også sett det nødvendig å ha med begrepet «programmering». Det betyr ikke at jeg ser på algoritmisk tenkning som noe som er avhengig programmering. Min tolkning er heller at mye av forskningen ikke opererer med et tydelig skille mellom algoritmisk tenkning og programmering. Tabell 2-2 under viser hvordan jeg gjennomførte litteratursøk for å finne forskning relatert til algoritmisk tenkning.

TABELL 2-2: OVERSIKT OVER LITTERATURSØK

Tema	Inkludert	Ekskludert
Database	Eric, Google Scholar, Oria, Springerlink	Alle andre databaser
Tid	2000 -d.d.	
Type litteratur	Vitenskapelige artikler, rapporter og bøker	Publikasjoner som ikke er knyttet til utdanningskontekster
Type aktivitet	Publikasjoner som sier noe om algoritmisk tenkning i matematikkopplæringen	Alt som ikke er knyttet til utdanningskontekster
Språk	Norsk, engelsk	Alt annet
Metode	Kvalitative og kvantitative	Ingen
Søkeord	«algoritmisk tenkning», «fagfornyelsen», «matematikk», «digitale ferdigheter», «problemløsning», «programmering» «computational thinking», «mathematics», «problem solving», «programming»	

I litteratursøket kom det frem en god del internasjonal forskning om algoritmisk tenkning. I forskningslandskapet til algoritmisk tenkning ser det ut som det er flere informative diskusjoner rundt det engelske begrepet for algoritmisk tenkning, *computational thinking*. Det er også laget flere rammeverk eller klassifiseringsverktøy for algoritmisk tenkning for å kunne knytte dette til didaktiske praksiser i matematikk- og naturfagsopplæringen (Brennan &

Resnick, 2012; Bocconi et al., 2016). Jeg skal nå gjøre rede for hva forskningslandskapet til algoritmisk tenkning sier om definisjoner, dimensjoner og rammeverk for algoritmisk tenkning. Hensikten er å synliggjøre bakgrunnen for valg av utgangspunkt videre i denne studien.

2.1.2.1 Internasjonal forskning om algoritmisk tenkning – computational thinking

Wing, professor i informatikk ved Columbia University, skrev en artikkel om algoritmisk tenkning i 2006 (s. 33) hvor hun fremmer algoritmisk tenkning som en fundamental ferdighet for alle, og som en uunngåelig skjebne som vil påvirke alt og alle. I 2008 (s. 3717) utdyper hun begrepet algoritmisk tenkning i en ny artikkel, og skriver at essensen med algoritmisk tenkning er abstraksjon. Hun mener at man ved å klare å tenke abstrakt vil ha en bedre evne til å forestille seg hva en datamaskin kan gjøre, og dermed ha mulighet til å tenke forbi fysiske dimensjoner av tid og rom. I samme artikkel skriver Wing videre at man gjennom å lære barn i tidlig skolealder fundamentale aspekter ved algoritmisk tenkning, vil man kunne danne et godt grunnlag for videreutvikling av algoritmisk tenkning senere i livet. I artikkelen stiller hun spørsmål til hvordan man kan lære bort og undervise algoritmisk tenkning på en god og effektiv måte.

I 2010 gjorde hun nok en gang rede for begrepet algoritmisk tenkning og forklarer dette med de tankeprosessene som involveres når et problem formuleres og deres løsninger blir presentert på en måte som gjør at et databehandlingssystem kan utføre en handling på en effektiv måte (Wing, 2010, s. 1).

Aho (2012, s. 832) hevder at begrepet *computation* genererer forvirring da det mangler klare spesifikasjoner av hva kvalifikasjonene til *computation* går ut på. Aho forenkler Wings definisjon til «... computational thinking to be the thought processes involved in formulating problems so their solutions can be represented as computational steps and algorithms».

Grover og Pea (2013, s. 38) stiller lignende spørsmål som Wing, og spør hvordan man på best mulig måte kan fremme algoritmisk tenkning hos barn i skolen. Grover og Pea tar for seg en forvirring rundt definisjonen av algoritmisk tenkning, og gjør rede for ulike aspekter i utviklingen til dette begrepet. De kobler algoritmisk tenkning til programmering og Papert sitt LOGO-programmeringsprosjekt på 1960-tallet. De mener at programmering er relatert til utvikling av algoritmisk tenkning fordi det hevdes at det fremmer ferdigheter knyttet til

problemløsning og logisk tenkning, og motivasjon hos elever. Videre påstår Grover og Pea at dersom elever får mulighet til å utvikle slike ferdigheter med hjelp av programmering, bør programmering kobles til flere matematiske emner enn bare geometri. De stiller også spørsmålet «hvordan kan algoritmisk tenkning bli gjenkjent?».

Grover og Pea (2013, s. 39) peker på et verdifullt aspekt ved begrepet algoritmisk tenkning som har stor betydning i en utdanningskontekst, nemlig det praktiske aspektet som realiserer algoritmisk tenkning i skolen. Dette aspektet kommer fra et CS Principles- (computer science) kurs, ledet av College Board og the National Science Foundation. Basert på syv store ideer ved databehandling (computing), kommer det praktiske aspektet frem.

Weintrop med sine seks kollegaer (2016, s. 128-129) argumenterer for implementering av algoritmisk tenkning i matematikk- og naturfagsopplæringen. De peker på flere fordeler. Den ene fordelen er et pedagogisk aspekt hvor det gjensidige læringsforholdet mellom matematikk og algoritmisk tenkning er fordelaktig. Det er ikke bare fordeler for matematikk å inkludere algoritmisk tenkning, for matematikk gir også algoritmisk tenkning en rikere kontekst. Det er altså en vinn-vinn-situasjon. Weintrop et al. (2016) mener dette gjensidige forholdet er kjernen i deres motivasjon for å forene matematikk, naturfag og algoritmisk tenkning. En annen fordel de peker på, er hvor nært algoritmisk tenkning plasserer matematikken til fremtidens gjeldende yrker.

I tilknytning til prosessen for implementering av algoritmisk tenkning, viser Weintrop et al. (2016, s. 143) til en mulig utfordring. De mener implementeringsprosessen i for liten grad gir veiledning til lærere som skal praktisere og realisere algoritmisk tenkning i klasserommet. De peker på at mye av vanskeligheten stammer fra manglende definisjon av fremgangsmåter, særlig i matematiske eller vitenskapelige sammenhenger.

Enda et argument som Weintrop et al. (2016, s. 143) viser til, er koblingen til utvikling av matematikkunnskaper og algoritmisk tenkning gjennom programmering, på måter som ellers ikke er mulig. Programmer som for eksempel Scratch, er blitt populært for dette, mener de. Samtidig som det arbeides med å lage rammeverk for forståelse og vurdering av algoritmisk tenkning i matematikk, bør det parallelt arbeides med utvikling av læringsaktiviteter, verktøy og læringsmiljø som fremmer algoritmisk tenkning.

I 2016 ble det publisert en rapport fra en studie som heter *Developing computational thinking in compulsory education*, også kalt *CompuThink*, i regi av Joint Research Centre, som har

som mål å gi evidensbasert vitenskapelig støtte til den europeiske kommisjonen. I rapporten tar forfatterne for seg forskning om algoritmisk tenkning og gir en oversikt og analyse av definisjoner og beskrivelser av begrepet (Bocconi et al., 2016). Forfatterne tar for seg forskning fra blant andre forskere som Wing, Grover og Pea, som allerede er nevnt i denne studien.

Basert på oversikten og analysen av forskning om algoritmisk tenkning, beskriver Bocconi et al. (2016) algoritmisk tenkning slik: «computational thinking is regarded as a thought process entailed in designing solutions that can be executed by a computer, a human, or a combination of both». I rapporten påpekes det at fokus for algoritmisk tenkning ikke nødvendigvis er på teknologibruk i seg selv, men snarere mer på å forstå de underliggende forestillingene og mekanismene til teknologi. Det blir presentert to tydelige dimensjoner av algoritmisk tenkning i rapporten. Den første dimensjonen handler om hvordan algoritmisk tenkning er separert fra teknologi, mens den andre dimensjonen går ut på at algoritmisk tenkning er en problemløsningsmetode som innebærer en evne til å finne eller lage løsninger som kan utføres av en datamaskin, et menneske eller begge deler (Bocconi et al., 2016, s. 15). Bocconi et al. presenterer også et rammeverk for algoritmisk tenkning som baserer seg på oversikten og analysen de har laget av forskning gjort på algoritmisk tenkning. Jeg kommer tilbake til dette rammeverket.

2.1.2.2 Algoritmisk tenkning med og uten teknologi

Gjøvik og Torkildsen mener et naturlig miljø for å implementere algoritmisk tenkning er via programmering (2019, s. 34). Som jeg nevnte i avsnittet over, har rapporten til CompuThink-studien presentert to dimensjoner av algoritmisk tenkning (Bocconi et al., 2016, s. 15).

En studie som er nevnt i [1.3 Tidligere forskning](#), oppsummerer et kvasi-eksperiment som har undersøkt ulikheter i utviklingen av algoritmisk tenkning mellom elever som deltok i en serie aktiviteter som ikke inkluderte teknologi, såkalte «unplugged-activities» (Brackmann et al., 2017). Resultatene fra studien viste at elever som deltok på aktivitetene som ikke inkluderte teknologi, forbedret ferdigheter i algoritmisk tenkning betydelig mer enn kontrollgruppen som ikke deltok på de aktivitetene. Forskerne konkluderte med at en slik framkoblede tilnærming til algoritmisk tenkning faktisk var med på å utvikle elevers algoritmiske tenkning. De framkoblede aktivitetene involverte logiske spill, kortspill eller fysiske bevegelser som ble brukt til å representere og forstå datavitenskapelige begreper og algoritmer (Brackmann et al.,

2017, s. 65). Vurderingen av elevenes algoritmiske tenkning ble gjort basert på en egenutviklet «CT-test». Ifølge Brackmann et al. (2017, s. 67) måler testen kandidatens evne til å formulere og løse problemer ved å stole på grunnleggende konsepter ved programmering, som for eksempel sekvenser, løkker, variabler og funksjoner. Testen tar utgangspunkt i fire komponenter i algoritmisk tenkning; dekomposisjon, algoritmebehandling, abstraksjon og generalisering.

Imenes et al. (2021) kobler programmering sammen med matematikk ved å undersøke hvordan lærerstudenter anvender programmering for å løse matematikkoppgaver. Lærerstudentene som var med på forskningsprosjektet brukte Monte Carlo-simuleringer for å lære programmering og simulering. I prosjektet skulle studentene også gjennomføre manuelle simuleringer ved siden av å programmere digitale simuleringer. Studien indikerer blant annet at det kan se ut til at lærerstudenter har nytte av å gjøre matematikkoppgaver ved hjelp av både manuelle og digitale simuleringer.

I artikkelen uttrykker Imenes et al. (2021, s. 8) at det å skrive en kode i utgangspunktet er proseduralt. Å planlegge programmeringsprosessen, hvor man finner den mest effektive og hensiktsmessige løsningen for problemet, krever imidlertid en viss relasjonell forståelse for å planlegge. En fallgrube knyttet til instrumentell forståelse og tilnærming til problemet, er samtidig at man står i fare for å ikke oppdage programmeringsfeil, slik at feilsøkningsprosessen stagnerer. En del av prosessen er også å vurdere svar, noe som krever en relasjonell forståelse, mener Imenes et al. (2021, s. 8). I studien trekkes manuell simulering frem som en fordel fordi det gir studenter mulighet til å se og forstå hva de gjør og hva som skjer i arbeidsprosessen. Funn i studien antyder at når lærerstudenter lykkes med programmeringsaktiviteter knyttet til løsning av statistiske problemer som ikke lar seg løse intuitivt, kan det bidra til at studentene ser matematisk nytte i programmeringsaktiviteter. I artikkelen uttrykker Imenes et al. (2021, s. 21) at det er viktig å hjelpe studenter å koble programmering i statistikk og sannsynlighet til utvikling av forståelse av hvordan en går frem for å løse problemstillinger og vurdere realismen i de aktuelle løsningsforslagene.

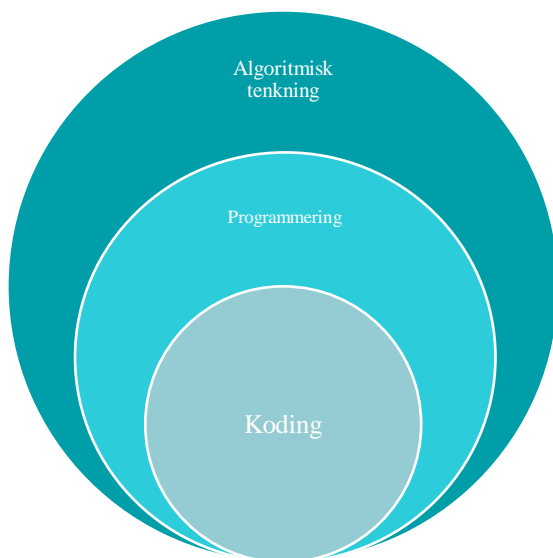
Algoritmisk tenkning, programmering og koding

Ifølge OECD-rapporten (Organisasjon for økonomisk samarbeid og utvikling), Skills of Outlook 2019 (s. 190), brukes programmering og koding trolig om hverandre. Med koding menes imidlertid som skriving på et konkret programmeringsspråk som forteller

datamaskinen instruksjoner den skal følge (Balanskat & Engelhardt, 2015, s. 7). Programmering ses på som en mer generell form for håndtering av analyse, utvikling og implementering av løsninger på problemer ved bruk av en datamaskin (Lye & Koh, 2014, s. 53).

Ifølge Lye og Koh (2014, s. 52) er programmeringsprogrammet Scratch basert på Papert sitt LOGO-programmeringsprosjekt fra 1960-tallet. Lye og Koh mener elever arbeider med algoritmisk tenkning når de programmerer og løser problemer ved hjelp av datamaskiner. Lye og Koh tar utgangspunkt i Brennan og Resnicks rammeverk for algoritmisk tenkning som består av tre dimensjoner, og gjengir det som problemløsende praksiser som oppstår når man programmerer. De understreker likevel at det ikke bør tas for gitt at elever lærer seg problemløsningsferdigheter av seg selv gjennom programmering. De utdyper dette med at dersom kognitive prosesser ikke blir støttet, og det gis mangelfull veiledning, kan dette føre til et begrenset læringsutbytte (Lye & Koh, 2014, s. 58).

Tatt i betraktning at algoritmisk tenkning kan oppnås uten teknologi, kan man se for seg at algoritmisk tenkning omfavner mer enn programmering og koding, og at programmering omfavner mer enn koding. Se illustrasjon i figur 2-3.



FIGUR 2-3: EGENKOMPONERT ILLUSTRASJON AV ALGORITMISK TENKNING, PROGRAMMERING OG KODING

I dette kapittelet har jeg så langt gjort rede for Utdanningsdirektoratets uttalelser om algoritmisk tenkning. Jeg har også greid ut om algoritmisk tenkning sitt forskningslandskap, hvilke undersøkelser som er gjort vedrørende definisjoner, og hvilke spørsmål som er stilt i

undervisningskontekst. Jeg skal nå gå nærmere inn på to ulike eksisterende rammeverk for algoritmisk tenkning som presenteres i forskningslandskapet.

2.1.2.3 *Rammeverk med tre dimensjoner*

Ifølge Brennan og Resnick (2012, s. 1) er det noe usikkerhet knyttet til hva som faller innunder begrepet algoritmisk tenkning (på engelsk: *computational thinking*). Enda større usikkerhet er videre knyttet til hvordan algoritmisk tenkning skal vurderes hos elever. Brennan og Resnick utviklet derfor et rammeverk for å forstå begrepet algoritmisk tenkning basert på deres egen empiriske forskning, og tok utgangspunkt i programmeringsaktiviteter hos barn og unge.

Brennan og Resnicks rammeverk har som hensikt å se på hvordan programmeringsverktøyet Scratch kan bidra til utvikling av algoritmisk tenkning hos elever. Rammeverket består av tre dimensjoner, og er bygget opp på en annen måte enn rammeverket som Bocconi et al. har laget. Brennan og Resnick sine tre dimensjoner er *computational concepts*, *computational practices*, og *computational perspectives*. Lye og Koh (2014, s. 52) gir uttrykk for at disse tre dimensjonene er nyttige for å forstå hvordan elever tilnærmer seg programmering og algoritmisk tenkning i programmet Scratch.

At Brennan og Resnicks rammeverk tar utgangspunkt i programmeringsaktiviteter kan tilsi at aktiviteter som ikke kan knyttes til programmering ikke har en sammenheng med algoritmisk tenkning. Likevel uttrykker Brennan og Resnick at konseptet kan overføres til «ikke-programmeringsaktiviteter». De kommer imidlertid bare med eksempler som kan knyttes til programmeringsaktiviteter med Scratch (2012, s. 3). Videre uttaler de at algoritmisk tenkning kan være til hjelp i å lære ved bruk av Scratch, og at programmering med Scratch er med på å gi en sammenheng og kontekst til samtale om algoritmisk tenkning (2012, s. 2).

Som Aho (2012, s. 832) har uttrykt, er det behov for en ytterligere spesifisering av begrepet *computation*. Begrepet oppfordrer dermed til en avklaring når det skal oversettes til norsk. Dette især fordi det norske begrepet for *computational thinking* er blitt definert som *algoritmisk tenkning*, mens *computation* tradisjonelt sett blir oversatt til norsk som *beregning*. Det er for eksempel en gren innenfor matematikk som heter *applied and computational mathematics*, som er oversatt til *anvendt og beregningsorientert matematikk* (Universitetet i Bergen, 2014). Når man da ser på internasjonal forskning som diskuterer forvirring knyttet til

klare kvalifikasjoner til hva computation går ut på, kan det være utfordrende å få med seg alle dimensjonene når man oversetter begrepet til norsk, fordi begrepet allerede er oversatt til algoritmisk tenking. Dette faller utenfor fokuset i denne studien, men det kan være greit å ha i mente når jeg nå skal ta for meg begreper som muligens ikke er oversatt til norsk. Når jeg skriver algoritmisk tenkning i denne studien, sidestiller jeg det med det engelske begrepet computational thinking, og forutsetter at alle dimensjoner og aspekter følger med i oversettelsen. Jeg skal nå gå nærmere inn på Brennan og Resnicks rammeverk for algoritmisk tenkning.

Computational concepts

Computational concepts er den første av de tre dimensjonene, og består av syv begreper som Brennan og Resnick mener har en sammenheng med både programmeringsaktiviteter og «ikke-programmeringsaktiviteter» i en undervisningskontekst. Dimensjonen handler om begreper som er knyttet til programmering, men Brennan og Resnick understreker at konseptene kan overføres til «ikke-programmeringsaktiviteter» (2012, s. 3). Begreper som løkker, data, vilkår, parallellitet, sekvenser, operatører og data blir gjort rede for, og kobles til eksempler i Scratch, i denne dimensjonen.

Computational practices

Den andre dimensjonen, Computational practices, består av prosesser og strategier for læring og tenkning, hvor fokuset ligger i prosessen fra det tidspunktet man tenker på hva man lærer til hvordan man lærer. Dette er den dimensjonen som i størst grad trolig dekker og gjenspeiler det norske begrepet algoritmisk tenkning. Denne andre dimensjonens egenskaper gjenfinnes i Utdanningsdirektoratet og Barefoot Computings figur om algoritmisk tenkning, under arbeidsmåter og nøkkelbegreper. Brennan og Resnick (2012, s. 6) har gjennom intervjuer og observasjoner sett på kjennetegn som inkrementell, testing, feilsøking, abstrahering og modulering. Lye og Koh (2014, s. 53) gjengir denne dimensjonen som problemløsende praksiser under programmering.

Computational perspectives

Computational perspectives er den siste dimensjonen, som fokuserer på samspill og interaksjoner i programmering. Det legges også vekt på ens egen forståelse av seg selv. Kjennetegn som Brennan og Resnick (2012, s. 10) har vist til i denne forbindelse, er hvordan elever stiller spørsmål og uttrykker seg om teknologien rundt seg. Brennan og Resnick så på kjennetegn på samspill og interaksjoner mellom elever, og hvordan elevene uttrykte for egen rolle. Eksempler på dette kunne være hvordan elever var spørrende, kontaktsøkende og hvordan de uttrykte seg.

2.1.2.4 Rammeverk med kjerneferdigheter, tilnærminger og dimensjoner

I rapporten fra CompuThink-studien har Bocconi et al. (2016) tatt for seg forskning om algoritmisk tenkning, og analysert innhold i forskningslandskapet til algoritmisk tenkning. Dette resulterte i en oversikt over et rammeverk bestående av kjerneferdigheter, (engelsk: «core skills»), tilnærminger og to ulike dimensjoner av algoritmisk tenkning. Som nevnt i delkapittelet [2.1.2](#) om internasjonal forskning knyttet til algoritmisk tenkning, forteller Bocconi et al. (2016, s. 15) også at algoritmisk tenkning består av to ulike dimensjoner. I arbeidet med analysen av litteratur for å finne sammenfallende definisjoner og fenomener ved algoritmisk tenkning har Bocconi et al. laget en tabell som viser ferdigheter og konsepter ved algoritmisk tenkning, som er kommet frem i forskning og litteratur om algoritmisk tenkning. Se figur 2-4 som viser en skjermdump fra rapporten. Basert på oversikten i figur 2-4, har Bocconi et al. kommet frem til at algoritmisk tenkning omfatter seks kjerneferdigheter. De uttrykker videre at det er enkelte forskningsartikler som mener algoritmisk tenkning også består av tilnærminger, og har utarbeidet en oversikt over forskning som mener dette. Oversikten vises i figur 2-5 på neste side, som er en skjermdump av CompuThink-rapporten. Bocconi et al. (2016, s. 19) mener at dersom man kan si at en kompetanse kan ses på som en sum av ferdigheter, kunnskap og holdninger, vil denne oversikten gi en mulighet til å betrakte algoritmisk tenkning som en kompetanse. Bocconi et al. (2016) viser til at det er en sterk enighet i forskningslandskapet til algoritmisk tenkning om at algoritmisk tenkning rommer mer enn programmering.

Barr & Stephenson, 2011	Lee et al., 2011	Grover & Pea, 2013	Selby & Woollard, 2013	Angeli et al., 2016
Abstraction	Abstraction	Abstractions and pattern generalizations	Abstraction	Abstraction
Algorithms & procedures		Algorithmic notions of flow of control	Algorithmic thinking	Algorithms (including Sequencing and Flow of control)
Automation	Automation			
	Analysis			
		Conditional logic		
Problem Decomposition		Structured problem decomposition (modularizing)	Decomposition	Decomposition
		Debugging and systematic error detection		Debugging
		Efficiency and performance constraints	Evaluation	
			Generalizations	Generalization
		Iterative, recursive, and parallel thinking		
Parallelization				
Simulation				
		Symbol systems and representations		
		Systematic processing of information		

FIGUR 2-4: SKJERM Dump fra CompuThink-rapport (Bocconi et al., 2016, s. 17)

Reference	CT dispositions / attitudes / attributes
Barr, Harrison & Conery (2011, p. 51)	Confidence in dealing with complexity Persistence in working with difficult problems The ability to handle ambiguity The ability to deal with open-ended problems The ability to communicate and work with others to achieve a common goal or solution
Woollard (2016, p. 5)	Tinkering Creating Debugging Persevering Collaborating
Weintrop et al. (2015, p. 133)	Confidence in dealing with complexity Persistence in working through challenging problems Ability to deal with open-ended problems

FIGUR 2-5: SKJERM Dump som viser tabell fra CompuThink-rapport (Bocconi et al., 2016, s. 19)

På bakgrunn av resultater og erfaringer knyttet til rapporten fra CompuThink-studien, er det blitt utarbeidet en lignende rapport som tar for seg de nordeuropeiske landene Norge, Sverige, Danmark og Finland (Bocconi et al., 2018). Denne rapporten, *The Nordic approach to introducing computational thinking and programming in compulsory education*, gir en

oversikt over dagens status for algoritmisk tenkning og programmering i nasjonale læreplaner i de nordeuropeiske landene, sammen med en presentasjon av ideer til politiske tiltak. I denne rapporten henviser Bocconi et al. (2018) til den europeiske versjonen av CompuThink-studien, og identifiserer fem grunnleggende kjerneferdigheter innenfor algoritmisk tenkning. Kjerneferdighetene kan relateres til tilnærminger som vises i figur 2-5. Summen av kjerneferdighetene, tilnærmingene og dimensjonen danner en helhetlig algoritmisk tenkning. Siden CompuThink-rapporten fra 2016 har det altså skjedd en liten endring i Bocconi et al. sitt rammeverk for algoritmisk tenkning. Antall kjerneferdigheter er redusert fra seks til fem, og det kommer tydeligere frem i rapporten fra 2018 at tilnærminger er noe som er en del av rammeverket for algoritmisk tenkning. Videre skriver Bocconi et al. at algoritmisk tenkning bærer et potensial som et middel for kreativ problemløsning, som kan skape innovasjon i en rekke andre fagområder. Dermed kan man si at algoritmisk tenkning har en avgjørende rolle i grunnskolen (Bocconi et al., 2018, s. 7). Videre skriver forfatterne at Norge vektlegger algoritmisk tenkning som et middel for å forstå hva som «ligger bak gardinen» i teknologien, samt et fokus på problemløsning og metoder for å skape løsninger. Ifølge Bocconi et al. (2018, s. 12) vil denne konseptualiseringen av algoritmisk tenkning være essensen i det som skiller algoritmisk tenkning fra digitale ferdigheter.

I tidsskriftet for matematikkundervisning, Tangenten, presenterer Gjøvik og Torkildsen (2019) en oversettelse av de fem komponentene som Bocconi et al. (2018) gjør rede for i sin CompuThinkNordic-rapport. Gjøvik og Torkildsen (2019, s. 34) hevder flere av begrepene er til å kjenne igjen i kjerneelementene i den nye læreplanen. I tabell 2-3 har jeg sammenfattet rammeverket for algoritmisk tenkning, som viser de ulike kjerneferdighetene, tilnærmingene, og de to dimensjonene.

TABELL 2-3: EGENKOMPONERT OVERSIKT OVER BOCCONI ET AL. (2016,2018) SITT RAMMEVERK FOR ALGORITMISK TENKNING

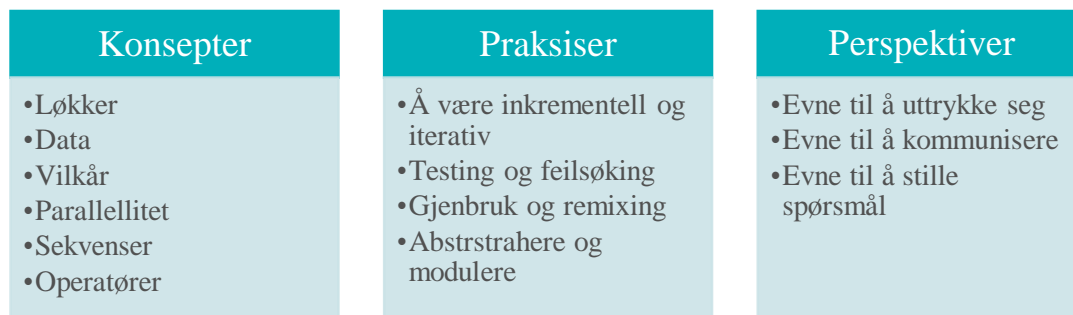
Kjerneferdigheter i algoritmisk tenkning	Definisjon
Abstraksjon	En prosess med å gjøre noe mer forståelig gjennom å ta bort unødvendige og overflødige detaljer. Essensen er å fremheve det viktige og fjerne det mindre viktige for å komme frem til en løsning.
Algoritmebehandling	Komme frem til en løsning ved å følge klare trinn eller steg i en prosess.
Generalisering	Identifisere mønstre, sammenhenger, likheter og utnytte disse funksjonene. Ved å gjøre dette løser man nye problemer basert på tidligere erfaringer om lignende problemer. Kjernen er å se mønstre i både informasjon og prosess. Viktige spørsmål er «ligner dette på tidligere problemer?» og «hvordan er dette annerledes?».
Automatisering	Finne en mest mulig effektiv arbeidsprosess. Dette kan være å gjøre det menneskelige bidraget minimalt i en del av et arbeid, ved å for eksempel la en datamaskin utføre simuleringer.
Dekomponering	Bryte opp et problem eller en prosess på en måte som gjør at man kan forstå, løse og utvikle delene hver for seg for å få en bedre forståelse av helheten. Dette gjør nye problemer mer håndfaste og lettere å forstå og løse.
Tilnærminger	Definisjon
Feilsøking	Innebærer en evne til å vurdere sitt eget arbeid og undersøke feil i prosessen og løse dem.
Kreativitet	En ferdighet som legger til rette for å skape innovative løsninger, og se problemer eller situasjoner fra flere perspektiver.
Samarbeid	En evne til å samarbeide sammen med andre for å løse et problem. Resultatet av samarbeidet er mer enn summen av den enkeltes bidrag, og dermed en verdifull ferdighet.
Utholdenhet	Ha standhaftighet til å holde ut i arbeid med utfordrende problemer.
Dimensjoner av algoritmisk tenkning	
Uten teknologi	Med teknologi
Innebærer hvordan algoritmisk tenkning er separert fra teknologi. Denne dimensjonen legger vekt på hvordan man fremmer algoritmisk tenkning i aktiviteter som ikke involverer teknologi.	Går ut på at algoritmisk tenkning er en problemløsningsmetode som innebærer en evne til å finne eller lage løsninger som kan utføres av en datamaskin, et menneske eller begge deler.

2.1.3 Rammeverkene sett i et komparativt lys

Jeg har nå gjort rede for to forskjellige rammeverk for algoritmisk tenkning. Brennan og Resnicks rammeverk for algoritmisk tenkning er utviklet for programmet Scratch, og har som hensikt å belyse hvordan programmeringsaktiviteter kan bidra til utvikling av algoritmisk tenkning hos barn (2012, s. 1). Bocconi et al. (2016) sin rapport fra studien *CompuThink* har som formål å gi evidensbasert vitenskapelig støtte til den europeiske kommisjonen. Deres rammeverk baserer seg på en oversikt over forskning som viser sammenfallende ferdigheter og kjennetegn ved algoritmisk tenkning. De har altså samlet tidligere forskning om algoritmisk tenkning og sett på hva som kommer frem der. Brennan og Resnick sitt rammeverk bygger i motsetning på egen empirisk forskning. Dette viser blant annet at rammeverkene har ulike grunnlag.

Rammeverkenes oppbygging er også strukturert ulikt. Brennan og Resnicks rammeverk inneholder tre dimensjoner, som derunder har tilhørende konsepter eller begreper. Det virker som dimensjonsinndelingen i Brennan og Resnicks rammeverk er basert på andre premisser enn Bocconi et al. sitt rammeverk.

Brennan og Resnicks rammeverket består av tre dimensjoner som er konsepter, praksiser og perspektiver, med en direkte oversettelse. Konsepter har begreper som er knyttet til konkrete aktiviteter i Scratch. Denne dimensjonen virker til å ha fokus på *hva* man kan tilegne seg i arbeid med Scratch. Dimensjonen praksiser inneholder tilnærminger og ferdigheter som setter søkelys på *hvordan* man tilegner seg kunnskap i Scratch. Med de to dimensjonene dekker man altså både *hva* og *hvordan* man lærer (Brennan & Resnick, 2012, s. 7). Den siste dimensjonen perspektiver, dekker elevenes beskrivelser av deres egen forståelse og utvikling i arbeid med Scratch, samt deres forhold til andre elever og den teknologiske verden rundt dem. I figur 2-6 og 2-7 har jeg illustrert strukturen i de to ulike rammeverkene for algoritmisk tenkning.



FIGUR 2-6: OVERSIKT OVER BRENNAN OG RESNICK (2012) SITT RAMMEVERK FOR ALGORITMISK TENKNING



FIGUR 2-7: OVERSIKT OVER BOCCONI ET AL. (2016;2018) SITT RAMMEVERK FOR ALGORITMISK TENKNING

Bocconi et al. (2016;2018) sitt rammeverk for algoritmisk tenkning er strukturert etter en dimensjon som inkluderer teknologi, og en som ekskluderer teknologi. Rammeverket identifiserer kjerneferdigheter man kan tilegne seg i arbeid med aktiviteter som fremmer algoritmisk tenkning. Det er også identifisert ulike tilnæringsmåter til et problem.

Felles for begge rammeverkene for algoritmisk tenkning er at algoritmisk tenkning kan identifiseres hos elever som arbeider med aktiviteter som kan knyttes til algoritmisk tenkning. Rammeverket til Brennan og Resnick blir imidlertid begrenset til aktiviteter i Scratch i deres artikkel. Man kan dermed si at rammeverket til Brennan og Resnick fungerer i en mer snever grad. Det kan bli utfordrende å bruke dette rammeverket på aktiviteter som ikke involverer teknologi. Samtidig kan det være en fordel at begrepene i de tre dimensjonene er knyttet til konkrete kontekster i Scratch. Rammeverket til Bocconi et al. har ikke slike konkrete eksempler på deres aspekter av algoritmisk tenkning. Rammeverket til Bocconi et al. er en del av et større forskningsprosjekt, og det kan derfor tenkes at det er mindre hensiktsmessig å gjengi helt konkrete eksempler på kontekster på lik linje som det er gjort i Brennan og Resnicks artikkel.

Dimensjonen om kontekster kan indikere at det er vanskelig å overføre denne dimensjonen til andre kontekster uten teknologi, da den inneholder begreper som oftest brukes i programmeringskontekster. Dersom kontekstdimensjonen skulle overføres til andre kontekster uten teknologi, ville man trolig ikke fått med hele dimensjonen i overføringen. At Bocconi m. fl. sitt rammeverk med og uten teknologi ikke ekskluderer noe annet enn teknologi i begge dimensjonene, kan derfor ses på som en styrke ved rammeverket.

I vurderingen av hvilket utbytte elever skal ha av algoritmisk tenkning i matematikkopplæringen, er det nødvendig å tydeliggjøre algoritmisk tenkning sin intensjon i matematikkopplæringen. Først da vil det være klart hvilket rammeverk som er mest hensiktsmessig å benytte som et utgangspunkt i denne studien. Jeg skal nå gå nærmere inn på hva styringsdokumenter sier om algoritmisk tenkning i en utdanningskontekst.

2.2 Del 2

2.2.1 Styringsdokumenter om algoritmisk tenkning

For å undersøke hva som ligger til grunn for implementering av algoritmisk tenkning, og for å klargjøre hva som er intensjonen som ligger til grunn for algoritmisk tenkning i matematikkopplæringen, har det vært nødvendig å undersøke styringsdokumenter. Styringsdokumentene jeg har valgt å ta utgangspunkt i, er rapporter fra Norges offentlige utredninger og OECD (Organisasjon for økonomisk samarbeid og utvikling).

Relevante NOU-rapporter argumenterer med behovet for kompetanse for fremtiden, og henviser ofte til OECD-rapporter. Det har derfor vært gunstig å involvere relevante kildehenvisninger i disse rapportene. Det har ikke vært relevant å involvere samtlige NOU-rapporter som argumenterer for kompetanse for fremtiden. Jeg har derfor valgt å forholde meg til de tre siste aktuelle, som er NOU 2013: 2, NOU 2015: 8 og NOU 2020: 2.

I NOU 2013: 2 Hindre for digital verdiskaping ble utvalget bedt om å kartlegge og identifisere ulike hindre og utfordringer for fremtidig digital verdiskaping i Norge. Rapporten tar for seg flere samfunnsmessige aspekter, men for denne studien er kapittel 6 om digital kompetanse av særlig interesse. Som nevnt innledningsvis i denne studien, har det ifølge Digitutvalget i NOU-rapporten (2013: 2, s. 99) vært for lite fokus på den kritiske refleksjonen rundt teknologisamfunnet og innovasjon i teknologi.

NOU 2015: 8 Fremtidens skole, fungerer som et grunnlag for den nye læreplanen, og Ludvigsen-utvalget fikk i oppdrag å vurdere den daværende grunnopplæringen opp mot krav til kompetanse i et fremtidig samfunns- og arbeidsliv (2015: 8, s. 8). Ludvigsen-utvalget mener at digital kompetanse er en forutsetning for å kunne delta i utdanning og læring, og senere i livet delta aktivt i samfunnslivet (NOU 2015: 8, s. 26).

NOU 2020:2 om fremtidige kompetansebehov, den siste aktuelle NOU-rapporten, er den eneste rapporten som nevner algoritmisk tenkning eksplisitt. Jeg skal nå gå nærmere inn på denne rapporten.

2.2.1.1 NOU 2020:2 – Fremtidige kompetansebehov III – Læring og kompetanse i alle ledd

NOU 2020:2 tar for seg en faglig vurdering av fremtidige kompetansebehov i Norge, og ble avgitt til Kunnskapsdepartementet 10. februar, omtrent et halvt år før Fagfornyelsen trådte i kraft. Først da er algoritmisk tenkning eksplisitt nevnt i norske styringsdokumenter som NOU-er, meldinger til Stortinget, og politiske målsettinger.

I NOU-rapportens underkapittel Kompetanse og ferdigheter for arbeids- og samfunnslivet, om programmering og algoritmisk tenkning, legges det vekt på at teknologi utvikler seg i en rasende fart, og at det derfor er behov for å legge vekt på en mer generell forståelse for digitale teknologier. NOU-rapporten henviser til en OECD-rapport fra 2019 og uttrykker at man trolig styrker evnen til matematisk, kreativt og kritisk tenkning ved å lære algoritmisk tenkning.

NOU 2020:2 (s. 43) beskriver algoritmisk tenkning slik:

Ved å lære såkalt algoritmisk tenkning lærer man tankegangen og konseptene bak mye av informasjonsteknologien. Algoritmisk tenkning skjer ofte innenfor en kontekst av programmering, men trenger ikke å gjøre det. Å lære algoritmisk tenkning ser ut til å kunne styrke evnene til matematisk, kreativ og kritisk tenkning.

Det gjøres altså klart at algoritmisk tenkning ikke er avhengig av en teknologisk kontekst. Det ser også ut til at algoritmisk tenkning befinner seg nærliggende til problemløsning.

Intensjonen bak algoritmisk tenkning ser også ut til å henge sammen med krav til økt generell forståelse og kunnskap om teknologi og dens rolle i vårt samfunn. NOU 2020: 2 henviser til OECD-rapporten fra 2019 når algoritmisk tenkning blir nevnt i rapporten. Jeg har derfor

vurdert det som relevant å undersøke hva som står om algoritmisk tenkning i OECD-rapporten fra 2019.

2.2.1.2 OECD 2019 – The skills outlook – thriving in a digital world

Denne rapporten beskriver hvordan land kan sørge for at dagens teknologiske revolusjon kan forbedre liv, og gjøre det mulig for mennesker å tilegne seg nødvendige ferdigheter for å dra nytte av revolusjonen. Rapporten identifiserer blant annet nasjoners potensielle fallgruver, og viser hvordan man kan legge til rette for å fremme livslang læring.

Rapportens kapittel 5 i rapporten, Learning in a digital environment, er særlig sentral for denne studien. Kapitlet undersøker muligheter teknologi kan gi for kompetanseheving i skoler, høyere utdanning og videre i livet. Den undersøker også læreres bruk av ny teknologi og hvordan politikk kan frigjøre potensiale i teknologi for undervisning og opplæring.

OECD-rapporten påpeker at tilgang til, og bruk av, datamaskiner på skoler ikke er et godt nok tiltak for å forbedre elevers teknologiske ytelse. Effekten av elevers forståelse av teknologi er avhengig av implementeringen av teknologi i undervisningspraksis (OECD, 2019, s. 179). De mener videre at ettersom elevevalueringer stort sett ikke inneholder en vurdering av elevers digitale ferdigheter, er det lite bevis på hvordan man på best mulig måte kan utvikle disse ferdighetene.

Digitalisering av samfunn og økonomier styrker et behov for å utvikle en generell forståelse for teknologi, i motsetning til en spesialisert kompetanse, som raskt kan risikere å bli foreldet (OECD, 2019, s. 179). I et stadig mer digitalisert samfunn skal man forholde seg til en massiv mengde av informasjon og digitale verktøy, samt ha en evne til å beskytte sitt eget personvern og egne data. OECD-rapporten legger vekt på skolens innflytelse på å utvikle de kritiske ferdighetene som kreves for å håndtere et digitalisert samfunn. I den sammenheng menes det at det er nødvendig å bevege seg utover IKT-undervisning slik som vi kjenner den, og over til algoritmisk tenkning (Bocconi et al.2016, i OECD, 2019, s. 189).

Videre i rapporten refereres det til CompuThink-studien og det understrekes at algoritmisk tenkning ikke nødvendigvis innebærer bruk av datamaskiner, men at bruk av datamaskiner likevel kan forekomme i sammenheng med programmering (Bocconi et al., 2016 i OECD, 2019). OECD-rapporten viser til forskning av Lye og Koh og uttrykker at dersom elever får mulighet til å bli eksponert for algoritmisk tenkning gjennom programmering, kan de bedre

ferdigheter innenfor problemløsning, samt få en dypere forståelse for nyere teknologi (Lye & Koh, 2014, i OECD, 2019, s. 190).

2.3 Oppsummering av algoritmisk tenkning

Kapittel 2 har hatt som hensikt å gi en teoretisk oversikt over algoritmisk tenkning i matematikkopplæringen. I lys av studiens formål har det vært hensiktsmessig å undersøke hvordan ulike instanser har tatt stilling til algoritmisk tenkning i den nye læreplanen i matematikk. I dette kapitlet har jeg derfor gjort rede for hva Utdanningsdirektoratet, forskning og styringsdokumenter sier om algoritmisk tenkning, for å kunne undersøke bakgrunnen og algoritmisk tenkning sin intensjon i matematikkopplæringen.

I første del av kapitlet har jeg beskrevet hva Utdanningsdirektoratet sier om algoritmisk tenkning, og hva forskningslandskapet til algoritmisk tenkning sier om ulike definisjoner, klassifiseringer og dimensjoner av algoritmisk tenkning. Kapitlets del 2 er særlig viktig for studiens utgangspunkt, fordi jeg der har gjort rede for hva sentrale styringsdokumenter sier om bakgrunnen for at algoritmisk tenkning er inkludert i matematikkfaget i skolen.

Utdanningsdirektoratet (2019b) uttrykker i kjerneelementet Utforskning og problemløsning at algoritmisk tenkning er viktig i prosessen med å utvikle strategier og fremgangsmåter for å løse problemer. Utdanningsdirektoratet presiserer i deres artikkel om algoritmisk tenkning at begrepet er oversatt fra det engelske begrepet computational thinking (Utdanningsdirektoratet, 2019c). De har også utarbeidet en figur om algoritmisk tenkning som er inspirert av Barefoot Computing, og som illustrerer såkalte nøkkelbegreper og arbeidsmåter i algoritmisk tenkning.

I algoritmisk tenkning sitt forskningslandskap har selve begrepet blitt diskutert, og det er blitt forsøkt å gjøre oppklaringer og presisjoner av begrepet. Det er også blitt stilt sentrale spørsmål til didaktiske praksiser som knytter algoritmisk tenkning fra IKT-verden sammen med matematikk og naturfag i en undervisningskontekst. Weintrop et al. (2014) kaster lys på fordeler ved implementering av algoritmisk tenkning i matematikk- og naturfagopplæringen. En fordel som ble nevnt, er et pedagogisk aspekt hvor det gjensidige læringsforholdet mellom matematikk og algoritmisk tenkning er fordelaktig.

Sammen med oppklaring av begrepet, er det utviklet ulike rammeverk i forsøk på å operasjonalisere algoritmisk tenkning. Bocconi et al. (2016), som blir nevnt i OECD-rapporten fra 2019, har laget et rammeverk bestående av kjerneferdigheter, dimensjoner og

tilnærminger i algoritmisk tenkning. I NOU-rapporten fra 2020 om fremtidige kompetansebehov, refereres det til OECD-rapporten fra 2019 når algoritmisk tenkning nevnes. I OECD-rapporten blir flere forskere som Lye og Koh, Wing, Balanskat og Engelhardt nevnt, sammen med forfatterne fra forskningsrapporten CompuThink-studien. Med deres forskning tatt i betraktning, gjengir OECD-rapporten årsaker til hvorfor algoritmisk tenkning er viktig å implementere i utdanning.

En viktig faktor er også de to dimensjonene i rammeverket for algoritmisk tenkning. Bocconi med hennes kollegaer forteller at algoritmisk tenkning har én dimensjon av algoritmisk tenkning med teknologi, og én uten. Dette sammenfaller med OECD-rapportens konklusjon om at algoritmisk tenkning ikke nødvendigvis innebærer bruk av teknologi.

Det andre rammeverket med tre dimensjoner av algoritmisk tenkning, laget av Brennan og Resnick (2012) tar utgangspunkt i programmeringsaktiviteter i Scratch. Med et slikt utgangspunkt kan det forstås at aktiviteter som ikke knyttes til programmering, er utfordrende å knytte til algoritmisk tenkning. Det sammenfaller ikke med OECD-rapportens konklusjon om at algoritmisk tenkning ikke nødvendigvis innebærer bruk av teknologi.

CompuThink- og CompThinkNordic-studien redegjør for algoritmisk tenkning på en systematisk måte hvor de klassifiserer ulike aspekter ved algoritmisk tenkning. De nevner tidligere forskning og samler fellesnevnerne fra dem, for å tydeliggjøre algoritmisk tenkning (Bocconi et al., 2016;2018). Rapporten definerer algoritmisk tenkning som følgende: «computational thinking is regarded as a thought process entailed in designing solutions that can be executed by a computer, a human, or a combination of both» (Bocconi et al., 2018, s. 7). Jeg oversetter definisjonen til følgende: Algoritmisk tenkning er tankeprosesser som innebærer å finne løsninger som kan utføres av en datamaskin, et menneske, eller begge deler.

På bakgrunn av at rapporten til Bocconi et al. (2016) benyttes som et teoretisk utgangspunkt i OECD-rapporten, samtidig som NOU-rapporten viser til OECD-rapporten når det argumenteres for å implementere algoritmisk tenkning i den nye læreplanen og matematikkopplæringen, velger jeg å bruke Bocconi et al. sitt rammeverk for algoritmisk tenkning som et utgangspunkt i denne studien. Ved å benytte det samme utgangspunktet for teori som relevante styringsdokumenter, øker studiens transparens. Når vi senere i denne studien skal undersøke lærernes oppfatning og forståelse av algoritmisk tenkning i matematikkopplæringen, utgjør rammeverket for algoritmisk tenkning de brillene dette ses gjennom.

3 Teoretisk rammeverk for algoritmisk tenkning

Kapittelet inneholder en presentasjon av rammeverket for algoritmisk tenkning som er benyttet som et utgangspunkt i deler av denne studien. Rammeverktøyet for algoritmisk tenkning fungerer som en bærende struktur i studien, og er med på å sette ulike kontekster i implementeringen av algoritmisk tenkning i perspektiv.

Det første forskningsspørsmålet i studien knyttes til hva som ligger til grunn for implementeringen av algoritmisk tenkning i matematikkopplæringen. Å se på hvorfor algoritmisk tenkning er innført i matematikkopplæringen, gir en helhetlig innsikt i intensjonen som ligger til grunn for algoritmisk tenkning. Algoritmisk tenkning sin intensjon er klar, vil utgangspunktet for undersøkelser av matematikklærernes oppfatning og forståelse av algoritmisk tenkning i matematikkopplæringen være tydelig. Dette utgangspunktet er rammeverket for algoritmisk tenkning, som fungerer som en sammenfatning og operasjonalisering av intensjonen bak algoritmisk tenkning slik at det blir mulig å anvende det i en analyse.

3.1 Rammeverk for algoritmisk tenkning

Rammeverket for algoritmisk tenkning vil i denne studien være utgangspunktet for undersøkelser av hvilke kompetanser algoritmisk tenkning har som intensjon å fremme i matematikkopplæringen. Rammeverket er særlig relevant for forskningsspørsmål 2 og 3, fordi disse legger opp til å undersøke *hvilke aspekter* ved algoritmisk tenkning som kommer frem i lærernes forståelse og oppfatning av algoritmisk tenkning og matematiske aktiviteter. Ved å benytte et rammeverk som har klassifisert ulike aspekter ved algoritmisk tenkning, vil det være mulig å identifisere hvilke sider og aspekter ved algoritmisk tenkning som er fremtredende i lærernes forståelse og oppfatning av algoritmisk tenkning. Det vil også være mulig å se hvilke aspekter ved algoritmisk tenkning som kan komme frem i matematiske aktiviteter lærerne nevner.

Som nevnt i [2.3 Oppsummering av algoritmisk tenkning](#) har jeg valgt å benytte Bocconi et al. sitt rammeverk for algoritmisk tenkning som et utgangspunkt for studiens analyse. Jeg skal nå gå nærmere inn på dette rammeverket.

Bocconi et al. (2016;2018) identifiserer fem grunnleggende kjerneferdigheter i algoritmisk tenkning. Med utgangspunkt i Bocconi et al. (2016, s. 19) sin påstand om at dersom en

kompetanse kan ses på som en sum av kunnskap, ferdigheter og holdninger, vil dette rammeverket for algoritmisk tenkning betrakte algoritmisk tenkning som en kompetanse.

Summen av de ulike aspektene ved algoritmisk tenkning, altså kjerneferdighetene, dimensjonene, holdningene og tilnærmingene danner et helhetlig bilde av algoritmisk tenkning. Ved å se på de ulike aspektene ved algoritmisk tenkning, inkluderer jeg dimensjoner, kjerneferdigheter og tilnærminger i algoritmisk tenkning. Dette rammeverket vil utgjøre studiens utgangspunkt for å svare på forskningsspørsmål 2 og 3. Under i figur 3-1, vises en oversikt over aspektene ved algoritmisk tenkning. Jeg skal nå gjøre rede for de ulike kjerneferdighetene i algoritmisk tenkning.

Algoritmisk tenkning		
Kjerneferdigheter <ul style="list-style-type: none">• Abstraksjon• Algoritmebehandling• Generalisering• Automatisering• Dekomponering	Tilnærminger <ul style="list-style-type: none">• Feilsøking• Kreativitet• Samarbeid• Utholdenhet	Dimensjoner <ul style="list-style-type: none">• Uten teknologi• Med teknologi

FIGUR 3-1: OVERSIKT OVER ASPEKTER VED ALGORITMISK TENKNING

3.1.1 Kjerneferdigheter i algoritmisk tenkning

Kjerneferdigheter utgjør sider ved algoritmisk tenkning som er mulig å betraktes som en del av helheten. De er ikke steg eller trinn man følger, som for eksempel i en oppskrift, men heller flere ulike sider å betrakte algoritmisk tenkning fra. Kjerneferdighetene utelukker heller ikke hverandre. Som jeg nevner i [2.3](#), er algoritmisk tenkning tankeprosesser som innebærer å finne løsninger som kan utføres av en datamaskin, et menneske eller begge deler (Bocconi et al., 2018, s. 7). Utdanningsdirektoratet (2019b) plasserer algoritmisk tenkning i kjerneelementet Utforskning og problemløsning, som blir omtalt som viktig i prosessen for å utvikle strategier og fremgangsmåter for å løse problemer. Algoritmisk tenkning har mye til felles med problemløsning, og dette ser vi også igjen i enkelte av kjerneferdighetene og tilnærmingene.

Ved å se på kjerneferdighetene hver for seg, og deretter koble dem til lærernes forståelse og oppfatning av algoritmisk tenkning, og aktiviteter i matematikkopplæringen, kan man se hvilket potensiale implementeringen av algoritmisk tenkning har i matematikkopplæringen. I denne studien vil formålet med disse kjerneferdighetene i rammeverket for algoritmisk tenkning være å fremheve ulike karaktertrekk og sider ved lærernes forståelse og oppfatning av algoritmisk tenkning.

3.1.1.1 Abstraksjon

Når man arbeider med matematikkoppgaver med mye informasjon, er det viktig å skille nødvendig informasjon fra overflødige detaljer. Essensen i kjerneferdigheten *abstraksjon* er å trekke frem viktige detaljer og se bort fra mindre viktige detaljer for å komme frem til en optimal løsning på problemet. På den måten blir arbeidsprosessen mer forståelig og anvendelig. Dette vil være mer krevende i rike oppgaver av en problemløsende art, og mindre krevende i rutineoppgaver som legger til rette for utvikling av ferdigheter, såkalte «puggeoppgaver». Uansett kan dette være en bevisst og gjennomtenkt del av planleggingen tidlig i prosessen.

3.1.1.2 Algoritmebehandling

Algoritmebehandling går ut på at man kommer frem til en løsning ved å følge steg eller trinn i en prosess for å løse et problem. Man behandler problemet med hjelp av en algoritme og følger en oppskrift for å løse problemet. Med denne kjerneferdigheten vil man også sammenligne algoritmer og se hvilken algoritme som er best for å løse problemet.

3.1.1.3 Generalisering

Å generalisere i algoritmisk tenkning innebærer evnen til å identifisere mønstre, sammenhenger, likheter for å så utnytte disse funksjonene. Dette ligger nært til essensen i problemløsning. Basert på tidligere erfaringer om lignende problemer, kan man løse nye problemer. Ved å gjenkjenne mønstre i problemet vil man kunne ta stilling til hvordan det nye problemer ligner på tidligere problemer og hvordan det nye problemet er annerledes. En generaliserende arbeidsmetode legger til rette for å se problemet i sammenheng med andre problemer, og slik løse problemet ved hjelp av tidligere erfaringer.

3.1.1.4 Automatisering

Ved å automatisere arbeider man for å finne en mest mulig effektiv arbeidsprosess for å løse problemet. *Automatisering* handler om å skape en oversikt for å se hva som trengs for at problemet skal bli løst på en effektiv måte. Et eksempel på dette kan vi finne i Imenes et al. (2021) sin artikkel om lærerstudenter som utfører Monte Carlo-simulering på en datamaskin for å få en bedre forståelse av hva som er rette svar på kompliserte og kontraintuitive sannsynlighetsoppgaver.

3.1.1.5 Dekomponering

Kjerneferdigheten *dekomponering* handler om å dele enten et problem, eller en arbeidsprosess for å løse problemet, opp i mindre bestanddeler. Denne kjerneferdigheten er en sentral del i problemløsning, hvor man forsøker å forstå, løse og utvikle de mindre bestanddelene for å få en bedre forståelse av helheten. Dette kan bidra til å gjøre problemet mer angripelig og lettere å forstå, og deretter lettere å løse.

3.1.2 Tilnærminger i algoritmisk tenkning

I en undervisningskontekst handler tilnærminger om redskaper for utvikling og læring. Ved hjelp av disse redskapene kan man vise sin kompetanse i algoritmisk tenkning. Tilnærminger i algoritmisk tenkning kan forstås som hvilken tilnæringsmåte man har til et problem. Tilnærmingene i algoritmisk tenkning skiller seg fra kjerneferdighetene med at det kan knyttes til tilnærminger til arbeidsprosessen, mens kjerneferdighetene knyttes til selve arbeidsprosessen. Tilnærmingene henger altså likevel tett sammen med kjerneferdighetene i algoritmisk tenkning. Dersom en elev viser en evne til god utholdenhet, men lite kreativitet i arbeid med et problem, kan det gi informasjon om elevens redskap for utvikling og læring i algoritmisk tenkning. Slik informasjon vil være verdifullt for lærere som skal veilede og legge til rette for utvikling og læring i matematikk.

En del av studiens intensjon er å undersøke hvilke aspekter innenfor tilnærminger i algoritmisk tenkning som kommer frem i intervjuene med de tre matematikklærerne. Det er derfor viktig å understreke at hensikten er ikke å måle tilnærminger hos elever. Likevel kan lærernes utsagn i intervjuene knyttes til elever. Tilnærminger kan bidra til å synliggjøre

lærernes forståelse av disse redskapene, som igjen kan illustrere en kompetanse i algoritmisk tenkning. Dette er egnet til å gi innsikt i hvordan lærerne identifisere redskap for utvikling og læring for å vise sin kompetanse i algoritmisk tenkning.

3.1.2.1 Feilsøking

Når man feilsøker vurderer man arbeidet i lys av prosessen og resultatet. Dette forutsetter en evne til å vurdere sitt eget arbeid, avdekke feil i prosessen og løse dem. Feilsøking dreier seg også om å evaluere, teste, verifisere og forutse utfall.

3.1.2.2 Kreativitet

I forbindelse med den fjerde industrielle revolusjonen, kan kreativitet knyttes til skapertrang og innovasjon og kan ses på som attraktivt. Kreativitet handler om å legge til rette for å se på et problem fra flere ulike perspektiver. En kreativ tilnærming til et problem kan være banebrytende, og kan bety at man ser på et problem med nye briller eller på en ny måte. Dette er ofte noe som kreves i rike oppgaver og problemløsning.

3.1.2.3 Samarbeid

Samarbeid blir også en tilnærming som knyttes til ønskede fremtidige ferdigheter for den fjerde industrielle revolusjonen. En evne til å samarbeide med andre for å løse et problem kan være en viktig forutsetning i mange tilfeller. Resultatet av samarbeidet er mer enn summen av den enkeltes bidrag, og *samarbeid* er dermed en verdifull egenskap i utvikling og læring.

3.1.2.4 Utholdenhet

Med en evne til å holde fokus og ha standhaftighet til å holde ut i arbeid med utfordrende problemer, arbeider man på en effektiv måte. Dette er ofte en ferdighet som øves mye på i barneskolen, og gevinsten av det blir stor for barna i deres resterende liv. Med en god utholdenhet vil man kunne klare å løse problemer effektivt uten å miste fokus.

3.1.3 Dimensjoner av algoritmisk tenkning

I dette rammeverket for algoritmisk tenkning er det to dimensjoner av algoritmisk tenkning som er med på å klassifisere hvordan algoritmisk tenkning fremtrer i matematiske aktiviteter eller i lærernes forståelse og oppfatning av algoritmisk tenkning. I dette tilfellet vil jeg si at dimensjonene utelukker hverandre.

3.1.3.1 Med teknologi

Denne dimensjonen inkluderer teknologi som et redskap eller verktøy. Algoritmisk tenkning med teknologi innebærer at man har en evne til å finne eller lage løsninger på problemer som kan utføres av en datamaskin, et menneske eller begge deler. Med algoritmisk tenkning med teknologi fokuserer man på hvordan algoritmisk tenkning blir fremmet i aktiviteter som involverer teknologi, som for eksempel programmering.

3.1.3.2 Uten teknologi

I dimensjonen uten teknologi blir teknologi separert fra algoritmisk tenkning. Teknologi vil altså være fraværende. Denne dimensjonen vektlegger hvordan man fremmer algoritmisk tenkning i aktiviteter som ikke involverer teknologi.

4 Metodisk tilnærming

I dette kapittelet vil jeg gjøre rede for det metodiske designet for arbeidet med datamaterialet i studien og analysearbeidet. Som et ledd i å finne svar på studiens forskningsspørsmål, har jeg valgt å gjennomføre kvalitative semistrukturerte intervjuer med matematikklærere i grunnskolen. Dette er fordi jeg ønsker å undersøke dagens læreres oppfatning og forståelse av algoritmisk tenkning i matematikkopplæringen. Fenomenologien vil dermed fungere som bakteppe i studien. Det blir altså naturlig å legge vekt på hvordan bestemte fenomener, i denne sammenheng algoritmisk tenkning, blir oppfattet, og hvordan man kan få bedre forståelse av algoritmisk tenkning ved å se nærmere på læreres oppfatning og forståelse av algoritmisk tenkning i matematikkopplæringen.

I kapittel 2 har jeg gjort rede for algoritmisk tenkning og sett på hva Utdanningsdirektoratet, forskningslandskapet til algoritmisk tenkning og styringsdokumenter sier om hvorfor algoritmisk tenkning har blitt implementert i matematikkopplæringen. Dette var for å sammenfatte intensjonen bak algoritmisk tenkning slik at det er mulig å anvende det i datamaterialet jeg har samlet inn.

I arbeidet med denne studien har det vært nødvendig å ta hensyn til valg og avgjørelser som kan ha påvirkninger på studiens resultater. Valg av metode er et av dem. I prosessen har det vært avgjørende å stille nødvendige metodologiske spørsmål til studien og min egen forskningspraksis.

I dette kapittelet vil jeg beskrive valg og vansker jeg har møtt på underveis i arbeidsprosessen, hvor avgjørelser og forberedelser er gjort i forbindelse med innsamling av data og utforming av analyseverktøyet. I [delkapittelet 4.8](#) vil studiens validitet og reliabilitet bli diskutert. Til slutt gis det en beskrivelse av etiske vurderinger og hensyn som er tatt i arbeidsprosessen.

4.1 Kvalitativ tilnærming

I boken til Christoffersen og Johannessen (2012, s. 17) om forskningsmetode for lærerutdanning, fremgår det at kvalitativ forskning er ideelt for å undersøke menneskers erfaring og oppfatning. Jeg har derfor benyttet meg av en slik kvalitativ metode, ved å undersøke læreres erfaringer, oppfatning og forståelse av algoritmisk tenkning i matematikkopplæringen.

For å få en dypere forståelse av algoritmisk tenkning sin rolle i matematikkopplæringen, har jeg intervjuet tre matematikklærere på tre ulike skoler i Bergen og omegn. I en slik kvalitativ tilnærming vil det være viktig å fremheve at jeg som forsker analyserer og tolker gitt informasjon med mitt utgangspunkt. Med det tatt i betraktning, vil det også være betydningsfullt at mitt perspektiv og mine valg blir synlige i denne studien. På den måten vil man kunne se hvordan jeg har påvirket studien.

4.2 Kvalitativt forskningsintervju

Gjennom kvalitative forskningsintervju ønsker jeg å få innsikt i læreres oppfatning og forståelse av algoritmisk tenkning. Med en samtale med lærere kan jeg få mulighet til å stille oppfølgingsspørsmål som jeg ellers ikke ville fått mulighet til dersom jeg hadde benyttet en tilnærming som for eksempel spørreundersøkelser.

Studiens formål er å få innsikt i læreres oppfatning og forståelse av algoritmisk tenkning, og hva de sier om hvordan de velger å implementere algoritmisk tenkning i deres matematikkopplæring. Det er derfor hensiktsmessig å benytte en kvalitativ metode i form av et kvalitativt forskningsintervju.

Ved å intervjuere lærere om deres forståelse og oppfatning om algoritmisk tenkning i matematikkopplæringen, gir det meg en mulighet til å forstå ulike kontekster i deres undervisning, sett i lys fra et lærerperspektiv. Jeg antyder altså at kvaliteten på implementeringen av algoritmisk tenkning i matematikkopplæringen, har en sammenheng med didaktiske kontekster. Kontekstene blir i denne sammenhengen aktiviteter i matematikkopplæringen og lærernes og barnas naturlige skolehverdag, hvor jeg har som mål å forstå interaksjoner og didaktiske metoder som kan spille inn på kvaliteten på implementeringen av algoritmisk tenkning i matematikkopplæringen.

4.3 Utarbeidelse av intervjuguide

Intervjuguiden jeg forberedte i forkant av gjennomførelsen av intervju har semistrukturerte egenskaper i den forstand at intervjuet verken er en åpen samtale eller en lukket spørreskjemasamtale. Kvale og Brinkmann (2015, s. 46) mener et semistrukturert intervju ligger nært opp til en samtale i dagliglivet, men har som profesjonelt intervju likevel et

bestemt formål. Dette åpner opp for at jeg som intervjuer har mulighet til å stille oppfølgings spørsmål til lærerne jeg intervjuer.

Det ble utarbeidet en intervjuguide som ble revidert flere ganger, ettersom arbeidet frem mot intervjuene var en ikke-lineær prosess. Jeg fjernet og la til spørsmål etter hvert som jeg arbeidet meg gjennom forskning og rapporter som omhandlet algoritmisk tenkning. På denne måten ble intervjuguiden mer nøyaktig og bedre passet studiens formål.

Alle de tre intervjuobjektene fikk intervjuguiden tilsendt på forhånd, for å ha mulighet til å tenke og reflektere over spørsmålene. Hensikten bak dette grepet, var å gi lærerne en mulighet til å forberede seg på et matematisk tema som er relativt nytt for dem, for å øke sannsynligheten for mer utdypende og reflekterte svar. En intensjon var også å skape mer presise forventinger til intervjuet.

Etter at det første intervjuet var gjennomført, bestemte jeg meg for ikke å gjøre noen endringer i intervjuguiden. Slik ville de gitte prinsippene for intervjuet forbli uendret, og alle de tre lærerne ville få det samme utgangspunktet for intervjuene. Jeg skal nå gå nærmere inn på spørsmålene i intervjuet.

4.3.1 Intervjuspørsmålene

Intervjuspørsmålene ble utformet med fokus på lærernes egne refleksjoner over sine faglige interesser og sitt arbeid tilknyttet implementering av algoritmisk tenkning i matematikkopplæringen. Spørsmålsformuleringer kan være med på å påvirke innholdet i svaret, og ved å være bevisst på dette, ville jeg få mulighet til å følge opp svar på en måte hvor jeg kunne verifisere mine fortolkninger.

I intervjuet ble det stilt ulike typer spørsmål i varierende grad. Oppfølgings spørsmålene var av inngående og fortolkende typer, hvor jeg forsøkte å få lærerne til å gi mer direkte og klargjørende svar. Jeg stilte også indirekte spørsmål som gjerne ikke var knyttet til læreren selv, men heller til skolen. Et eksempel på det er spørsmålet: «Er skolen med i noen prosjekter som kan falle innunder temaene programmering og algoritmisk tenkning?».

Det var viktig å få lærerne til selv å definere begrepet algoritmisk tenkning tidlig i intervjuet, da lærernes refleksjoner rundt deres forståelse og oppfatning av algoritmisk tenkning i matematikkopplæringen er et av hovedfokusene i denne studien. Dette ble gjort for å få frem

lærernes forståelse av begrepet. Å vite hva lærerne tar utgangspunkt i bidrar til å styrke troverdigheten i det påfølgende etterarbeidet med intervjuene.

I forlengelsen av at lærerne redegjorde for sin egen oppfatning av hva algoritmisk tenkning er, ble det stilt spørsmål om hvilke matematiske aktiviteter lærerne mente ville falle innunder temaet algoritmisk tenkning, og hvor nyttig lærerne tror algoritmisk tenkning og programmering vil være for elevene.

4.4 Utvalg

Antallet intervjupersoner avhenger av formålet med undersøkelsen (Kvale & Brinkmann, 2015, s. 148), og det er ønskelig å oppnå et representativt antall informanter i henhold til formålet. Hvis jeg intervjuer for mange lærere vil jeg ha mindre tid til å foreta en dyptgående analyse av intervjuene, og hvis jeg intervjuer for få lærere vil det være vanskelig å generalisere. Gitt rammene masteroppgaver generelt, men særlig rammene og temaene i denne masteroppgaven, har jeg valgt å gjennomføre tre kvalitative forskningsintervju med tre matematikklærere.

Denne studien har et strategisk utvalg, som innebærer at man fastsetter en målgruppe for studien før man velger ut hensiktsmessige informanter som kan bidra til å besvare forskningsspørsmålet (Johannessen et. al., 2016, s. 117). Ved å bestemme målgruppen for studien, velges et fokus som preger hele studien. Dersom jeg hadde valgt å intervju andre informanter enn lærere, som for eksempel skoleledere, ville endret fokuset jeg ønsket å ha i studien min. Jeg kunne også valgt elever som informanter, men da ville jeg antakelig fått svar på resultatet av prosessen for hvordan skolen implementerer algoritmisk tenkning, og ikke svar på hvordan lærere har arbeidet med denne implementeringen.

Mitt ønske var å komme så tett som mulig på hva lærere tenker om algoritmisk tenkning i matematikkopplæringen. Det var derfor naturlig å ha matematikklærere som målgruppen for studien. Videre måtte jeg ta hensyn til at lærerne underviste på de aktuelle trinnene hvor det var kompetansemål som nevner algoritmisk tenkning/programmering. Dette er fortrinnsvis fra fjerde trinn og oppover. Faktorer som alder, yrkeserfaring og kjønn er ikke tatt hensyn til i valg av informanter, da det er blitt lagt større vekt på fag og geografisk tilgjengelighet.

Med hensyn til geografisk tilgjengelighet, var det en fordel dersom informantene var fra Bergen og omegn. Jeg valgte tre skoler med geografisk spredning. Ingen av skolene befant

seg i samme bydel, og en av skolene ligger i en av nabokommunene til Bergen. I lys av at tre lærere utgjør utvalget i denne studien, er det viktig å understreke at det ikke gir en mulighet for å kunne generalisere matematikklæreres synspunkter om algoritmisk tenkning i matematikkopplæringen på et generelt grunnlag. Det betyr at funnene som blir presentert i denne studien er ikke representative for norske lærere generelt, men kun for det utvalget som har deltatt. Resultatene som blir presentert må leses i lys av denne begrensningen.

Basert på medstudenters opplevelser med å få kontakt med informanter til å gjennomføre intervju nå under pandemien, så jeg at det kunne bli vanskelig å få gjennomført forskningsintervju av god kvalitet med lærere jeg ikke hadde kjennskap til fra før av. Jeg besluttet derfor å ta kontakt med lærere på skoler jeg har vært i kontakt med tidligere, enten gjennom praksis i utdanningsløpet eller gjennom tidligere arbeid.

At utvalget kjenner meg fra tidligere arbeidsforhold og praksisforhold, er en faktor jeg må være bevisst på, da det kan ha påvirket studien på flere måter. Min sosiale relasjon til utvalget kan ha påvirket deres vilje til å delta i studien min. Relasjonen kan også ha påvirket selve intervjusituasjonen. Ved å være bevisst på dette, kan jeg dra nytte av det i intervjusituasjonen. Min intensjon har vært å legge så mye som mulig til rette for et rom hvor lærerne føler de kan snakke fritt. Da er det en fordel at utvalget kjenner til meg fra før. Ifølge Kvale og Brinkmann (2015, s. 108), er en av de etiske problemstillingene ved et forskningsintervju den sosiale relasjonen mellom intervjueren og utvalget. Denne relasjonen er nemlig avhengig av min evne til å skape et rom hvor lærerne kan snakke fritt.

Etter at Norsk Samfunnsvitenskapelige Datatjeneste (NSD) godkjente min forskningsstudie (se [vedlegg 2](#)) tok jeg kontakt med informantene pr e-post, og presenterte forskningsstudien og spurte om de kunne tenke seg å delta i denne. Et informasjonsskriv lå vedlagt i e-posten, hvor lærerne fikk opplysninger om forskningsstudiens formål, hvordan jeg hadde tenkt til å samle inn data, hva som var forventet av dem, samt opplysninger om hvordan personvern og anonymitet ble ivaretatt.

I denne studien vil lærerne holdes helt anonyme, og omtales som Lærer 1, Lærer 2 og Lærer 3. Dette er av hensyn til min relasjon til dem, fra enten tidligere arbeidsforhold eller praksisforhold. Skjønn og alder vil heller ikke bli oppgitt, og det vil ikke bli spesifisert hvor i Bergen eller omegn lærernes arbeidsplass befinner seg. Årsaken til det, er at jeg anser slik informasjon som irrelevant for studiens formål og forskningsspørsmål. Alle de tre lærerne jobber som kontaktlærer for hver sin klasse, og alle underviser i matematikk. Lærer 1

underviser på fjerdetrinnet, Lærer 2 underviser på syvende trinn, mens Lærer 3 underviser på sjettettrinnet.

4.5 Gjennomføring av intervju

Intervjuene ble gjennomført i november og desember 2020, og i februar 2021. Alle de tre intervjuene ble foretatt på skolene informantene arbeider på, i arbeidstiden deres. Av hensyn til aktuelle smittevernregler på intervjutidspunktene, ble avholdt ble dato for intervju flyttet på, slik at det var mulig å overholde disse reglene. Jeg kunne ha arrangert intervjuene gjennom et zoom-møte, en videochattetjeneste eller lignende, men vurderte at det var bedre å vente til det var mulig å gjennomføre fysiske intervjuer. Dette bestemte jeg i samråd med informantene. En av faktorene i denne vurderingen, var at det var viktig for meg at informantene følte seg trygge på å kunne gjennomføre et fysisk intervju i henhold til smittevernregler, og av hensyn til smitte av Covid-19. En annen faktor i vurderingen var også at det var viktig å ha tilgang til non-verbal kommunikasjon i intervjuet. En slik tilgang til kroppsspråk sammen med det verbale språket, gir meg mye informasjon om hva som blir sagt, og er et viktig bidrag til mine løpende vurderinger til oppfølgingsspørsmål i intervjuet.

I arbeid med innsamling av datamateriale har jeg tatt avgjørelser som kan ha påvirket resultatene i forskningsstudien. Et eksempel er avgjørelsen på hvordan jeg skulle samle inn data på best mulig måte. Det finnes flere måter å samle data og registrere intervjuer på, herunder ved lydopptak, videoopptak, notatskriving og bruk av hukommelsen, hvorav lydopptak ses på som den vanligste metoden (Kvale & Brinkmann, 2015, s. 205). Jeg valgte å benytte meg av lydopptak. Jeg skal nå gå nærmere inn på dette.

4.5.1 Lydopptak

Det ble gjort lydopptak av alle tre intervjuene. Jeg valgte å ikke skrive notater, for å vie min fulle konsentrasjon og oppmerksomhet til selve samtalen. På den måten kunne jeg komme med oppfølgingsspørsmål når det var behov for det. Ved bruk av lydopptak får jeg informasjon om ordbruk, tonefall og pauser i intervjuet. Denne informasjonen, sammenlagt med de erfaringene jeg sitter igjen med som intervjuer, kan gi meg informasjon om det sosiale samspillet i intervjuet. Dette er en viktig del i tolkningsprosessen hvor man forsøker å sikre at så lite som mulig går tapt i transkripsjonsprosessen.

Lydopptaket ble gjort på datamaskinen min, med videochatteprogrammet Zoom, uten tilgang til internett. Lydfilen ble deretter lagret på en ekstern harddisk som ikke har tilgang til internett. Lydfilene er beholdt under forskningsarbeidet, men vil slettes etter ferdigstilt arbeid.

4.5.2 Intervjuene

Intervjuene varte omtrent i 35- 45 minutter hver, og intervjuguiden ble stort sett benyttet for å kontrollere at jeg hadde vært innom de emnene og spørsmålene jeg ønsket. Dette skapte en fin flyt i samtalen i intervjusamtalen, og fremhevet styrken ved bruk av et semistrukturert intervju, hvor jeg fikk hjelp av intervjuguiden til å holde meg til relevante temaer, og ikke spore av.

Underveis i intervjuet var det nødvendig med fortløpende avgjørelser på oppfølgingsspørsmål, og hvilke svar som skulle følges opp og kommenteres. I alle tre intervjuene forsøkte jeg å få avklart og tydeliggjort sentrale utsagn som jeg der og da vurderte som relevante for studiens formål og forskningsspørsmål. En slik meningsavklaring vil ifølge Kvale og Brinkmann (2015, s. 170) gjøre etterarbeidet lettere. Det samme ser man ved kontrollspørsmål og oppfølgingsspørsmål underveis i intervjuet, hvor troverdigheten styrkes i etterarbeidet. Dette syns jeg var særlig utfordrende, da jeg ikke har noen tidligere erfaring med å intervju andre. Dette er en mulig fallgrube, som Kvale (2007, s. 8) påpeker. Som uerfaren forsker kan man bli lurt av hvor enkelt det tilsynelatende kan være å gjennomføre et kvalitativt intervju, da det i dets natur ligger nær en hverdagslig samtale (Kvale, 2007, s. 8). Først i etterkant, når jeg ser på datamaterialet, vil jeg oppdage mangler eller problemer.

Jeg opplevde det også som krevende å skille mellom begrepene algoritmisk tenkning, programmering og koding for lærerne. Det var også krevende å få lærerne til å koble algoritmisk tenkning til sine matematiske aktiviteter i undervisningen sin.

Jeg ønsker å legge til at å transkribere mine egne intervjuer, erfarte jeg som svært lærerikt. Jeg har lært mye om min egen intervjustil, og jeg har klart å gjøre meg opp noen tanker om de sosiale og emosjonelle aspektene ved intervjusituasjoner samt om den faglige oppmerksomheten som kreves i et godt forskningsintervju.

4.6 Datamaterialet

Transkripsjon av intervjuene, og øvrige dokumenter fra to av lærerne, er studiens datamateriale som har blitt analysert. Resultater sett opp mot presentert teori om algoritmisk tenkning er utgangspunktet for å besvare forskningsspørsmålet.

I intervjuguiden har jeg samlet tanker om hva jeg ønsker å spørre lærerne om, og svarene skal bidra til å belyse studiens forskningsspørsmål. Opplysningene som kommer frem i intervjuene er lærernes egne subjektive synspunkter, erfaringer og oppfatning knyttet til algoritmisk tenkning og programmering. To av lærerne har også delt dokumentasjon på undervisningsopplegg, planleggingsdokumenter, egenkomponerte videoer som er brukt i undervisning, og powerpointpresentasjoner som er brukt i utviklingstid med kollegaer.

I informasjonsskrivet lærerne fikk tilsendt pr e-post, ble det informert om hva jeg ønsket av lærernes deltakelse. I tillegg til å stille opp på et intervju, ble de også spurt om å dele dokumentasjon som viser hva de har arbeidet med i matematikk i forbindelse med algoritmisk tenkning. Det var viktig for meg å få frem at jeg hadde ikke et ønske om at de skulle produsere dokumentasjon i form av planleggingsdokumentasjon eller lignende for denne studien. Det var kun ønskelig å se på dokumenter som allerede var produsert. Dette av hensyn til deres arbeid, samt mitt ønske om at de ikke skulle gjøre noe mer arbeid i studien utover å delta på intervju. Jeg synes dette hensynet var særlig viktig i lys av pandemien.

4.6.1 Øvrige dokumenter i datamaterialet

Øvrige dokumenter i datamaterialet omfatter alle dokumenter utenom transkripsjon av intervjuene.

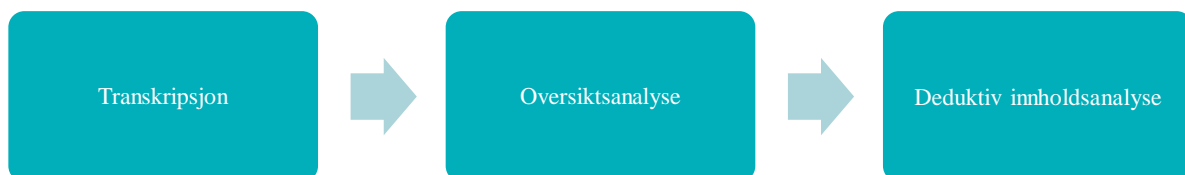
Lærer 1 hadde laget et planleggingsdokument som tok for seg matematiske temaer i matematikkundervisningen, hvor det var strukturert når elevene skulle arbeide med de ulike matematiske temaene i løpet av skoleåret 2020/2021. Videre fikk jeg muligheten til å se egenkomponerte videoer Lærer 1 hadde laget i forbindelse med undervisning knyttet til algoritmisk tenkning og programmering i matematikk.

Lærer 3 har delt dokumenter som tar for seg algoritmisk tenkning, programmering og koding i en undervisningskontekst.

4.7 Kvalitativ analysemetode

I dette delkapittelet presenterer jeg valg og begrunnelser som er gjort i studiens analyseprosess. En kvalitativ analyse inneholder flere steg, men er samtidig ofte en dynamisk prosess hvor man beveger seg frem og tilbake mellom stegene i analyseprosessen. Dette betyr at jeg tidlig i starten på denne studien har gjort meg opp noen tanker om hva jeg ville undersøke og finne. Videre fortsatte mine kontinuerlige fortolkninger og tanker om koblinger mellom datamaterialet og teori gjennom arbeidsprosessen i denne studien. Slike fortolkninger og tanker er muligens ikke særlig strukturerte, men kan likevel fungere godt som et grunnlag for det formelle og langt mer strukturerte analysearbeidet som skjer i etterkant av innsamling av datamateriale. Det er dette strukturerte analysearbeidet jeg nå skal gjøre rede for.

Første steg i analyseprosessen var transkripsjon av intervjuene. Deretter har det vært nødvendig å få en helhetlig oversikt over intervjuenes innhold, før jeg videre har tatt utgangspunkt i aspekter ved algoritmisk tenkning og sett etter innhold fra intervjuene som kan passe med disse. Til slutt har jeg undersøkt hvilke aspekter ved algoritmisk tenkning som kan komme frem i aktivitetene som er nevnt av de tre lærerne. Jeg skal nå ta for meg disse stegene i analyseprosessen. I figur 4-1 illustreres stegene i analyseprosessen.



FIGUR 4-1: OVERSIKT OVER ANALYSEPROSESS

4.7.1 Transkripsjon

I transkripsjonsprosessen blir talespråk gjort om til skriftspråk. Ifølge Kvale og Brinkmann (2015, s. 205) er transkripsjon en fortolkningsprosess hvor man abstraherer og fikserer, og deretter sitter igjen med dekontekstualiserte gjengivelser av intervjusamtaler. Lydopptaket av samtalen er en abstraksjon fra den fysiske tilstedeværelsen, fordi man blant annet mister tilgang til kroppsspråk. Transkripsjonen av lydopptaket gir enda en abstraksjon, der stemmeleie, intonasjoner og åndedrett går tapt. Det er viktig å være oppmerksom på disse konsekvensene, slik at får med viktig informasjon for studien, og tydeliggjøre dette i

etterarbeidet. Dette vil jeg gjøre ved å inkludere deskriptive detaljer fra intervjuet, og slik gi et mer eksakt bilde av intervjusituasjonene.

Lydkvaliteten på de tre forskjellige opptakene varierte, og det ene intervjuet skilte seg litt negativt ut. Dette intervjuet ble gjort i et stort klasserom med mye gjenklang. Det var derfor tidvis mer krevende å transkribere dette intervjuet. I forkant av transkripsjonsprosessen lagde jeg en oversikt over tegnsetning som er blitt brukt i transkripsjonen.

Transkripsjonen er skrevet med linjenummer, for å gjøre det lettere å vise til utsagn. Intervjuene ble i utgangspunktet transkribert ordrett, hvor jeg også inkluderte pauser, trykksterke ord, spørrenotasjoner og bekreftende uttrykk som «mhm». I ettertid ble avskriften redigert med en hensikt om å gjøre den mer leservennlig. Av hensyn til omfanget, unnlot jeg å inkludere småprat og digresjoner i intervjuene. For å sikre lærernes anonymitet, ble også intervjuene transkribert til bokmål, og teksten ble tilpasset skriftspråket med punktum og komma. For å få et rikere inntrykk av konteksten, er kommentarer skrevet i parentes, for eksempel nå lærerne ler eller viser noe. For pauser med en varighet på 0,5 til 1,5 sekunder har jeg valgt å bruke «...», og «...» for pauser lenger enn to sekunder. Jeg har valgt å ha med spørrenotasjon, da det ikke alltid er like lett å vurdere muntlig sjargong som et spørsmål. Trykksterke ord blir beskrevet som «'ord», og uhørbar lyd eller tale blir beskrevet som «X».

Etter alle intervjuene var ferdig transkribert, satt jeg igjen med over 35 sider med transkripsjon. For å få en oversikt over datamaterialet, har det vært nyttig å bearbeide transkripsjonene. Dette har jeg gjort ved å gjennomgå transkripsjonene av intervjuene, og se etter diverse fellestrekk ved alle tre intervjuene. Det er viktig å være bevisst på at hvert ledd i denne prosessen krever avgjørelser som kan få konsekvenser for mitt handlingsrom senere i analyseprosessen. Gjennom arbeidet med teori og datamaterialet, har det vært kontinuerlige vurderinger om tema og kategorier som skal inngå i analyseprosessen. Jeg skal nå gå nærmere inn på oversiktsanalysen som ble gjort i etterkant av transkripsjonen.

4.7.2 Oversiktsanalyse

Ved hjelp av en oversiktsanalyse, har jeg forsøkt å sortere innholdet i intervjuene etter relevante kontekster for implementeringen av algoritmisk tenkning. I bestemmelsen av hva som er relevant innhold i datamaterialet, er det nødvendig ta utgangspunkt i studiens formål og forskningsspørsmål. Det er også behov for å undersøke om intervjuene faktisk klarer å

måle det de har til hensikt å gjøre. Ved å gjennomgå transkripsjonene og se etter fellestrekk i kontekster, vil jeg få en oversikt over kontekstene som spiller inn på implementeringen av algoritmisk tenkning i matematikkopplæringen til de tre lærerne. Det vil også være mulig for meg å se hvordan lærerne betrakter algoritmisk tenkning.

Jeg har med studiens formål og forskningsspørsmål i bakhodet, tatt utgangspunkt i datamaterialet, og laget kategorier basert på hvordan fellestrekk og mønstre har kommet frem i intervjuene. Kvale og Brinkmann (2015, s. 227) kaller dette for en datastyrt koding av datamaterialet. I en datastyrt koding vil kategoriene, eller temaene, åpenbare seg underveis i transkripsjonsprosessen. Dette prinsippet spiller en viktig rolle i grounded theory-tilnærmingen til kvalitativ metode.

I oversiktsanalysen har jeg tatt utgangspunkt i ulike kontekster som de tre lærerne snakker om i intervjuene. Jeg har gjennomgått transkripsjonene og identifisert samtaletema, og laget kategorier basert på dette. Dette gjorde jeg ved å benytte fargekoder i transkripsjonen. Ved å utarbeide en tabell, fikk jeg oversikt over alle lærernes utsagn og kommentarer om de ulike kontekstene, sortert etter type kontekster. Ved hjelp av denne tematiske oversikten over kontekster ble det være lettere å undersøke lærernes utsagn i intervjuene på en oversiktlig måte. Jeg vil gå nærmere inn på innholdet i de ulike kontekstene i [5.1 Oversiktsanalyse](#).

Til slutt satt jeg igjen med et datamateriale som var sortert etter temaer som baserte seg på ulike kontekster som spiller inn på algoritmisk tenkning som rolle i matematikkopplæringen. Gjennom oversikten har jeg tilrettelagt for neste steg i analyseprosessen, nemlig den deduktive innholdsanalysen hvor jeg ser etter hvilke aspekter ved algoritmisk tenkning som kommer frem i intervjuene med de tre lærerne.

4.7.3 Deduktiv innholdsanalyse

I denne studien har jeg benyttet meg av en kvalitativ innholdsanalyse, som defineres som en metode for subjektiv tolkning av tekstdata ved bruk av systematisk klassifisering for å identifisere tema og mønster (Hsieh & Shannon, 2005 referert i Ringdal, 2018 s. 269). Metoden kan deles opp i flere typer, men bare én er relevant for denne studien, nemlig deduktiv innholdsanalyse.

Formålet med en deduktiv innholdsanalyse er å validere eller utvikle en begrepsmodell eller en teori (Ringdal, 2018, s. 269). En slik metode tar utgangspunkt i teori og tidligere forskning.

I denne delen av analysen har jeg tatt utgangspunkt i teori om algoritmisk tenkning og undersøkt hvilke aspekter ved algoritmisk tenkning som kommer frem i intervjuene.

Etter informasjonen i datamaterialet var ferdig sortert i oversiktsanalysen, tok jeg fatt på innholdsanalysen. Innholdsanalysen er strukturert etter kontekstene som kommer frem i oversiktsanalysen, og i hver kontekst er innholdet strukturert etter rammeverket for algoritmisk tenkning. Under hvert aspekt ved algoritmisk tenkning, presenterer jeg hva som kommer frem fra datamaterialet som er sammenfallende med de ulike aspektene ved algoritmisk tenkning.

4.7.3.1 Algoritmisk tenkning i lærernes forståelse og oppfatning

Den første konteksten som kommer frem i oversiktsanalysen omhandler lærernes forståelse og oppfatning av algoritmisk tenkning. Sitater fra intervjuene som kan fortelle noe om lærernes forståelse og oppfatning av algoritmisk tenkning, er plassert i denne konteksten. Sitatene ble vurdert opp mot innholdet i de ulike aspektene ved algoritmisk tenkning, og plassert deretter. I vurderingen om i hvilke aspekter ved algoritmisk tenkning sitatene skulle plasseres, var det avgjørende at lærerne sa noe eksplisitt som kunne være i overensstemmelse med aspektene. På denne måten ble sitater fra intervjuene med de tre lærerne sortert etter de ulike aspektene ved algoritmisk tenkning. Ved å sortere sitatene på denne måten, er det mulig å ta en vurdering av hvilke aspekter som er fremtredende i lærernes forståelse og oppfatning av algoritmisk tenkning.

Denne delen av undersøkelsen knyttes til forskningsspørsmål 2, som tar for seg aspekter ved algoritmisk tenkning i lærernes forståelse og oppfatning. Her er både kjerneferdigheter, tilnærminger og dimensjoner i algoritmisk tenkning inkludert i analysen.

4.7.3.2 Algoritmisk tenkning i aktiviteter i matematikkopplæringen

Den andre konteksten som vises i oversiktsanalysen knyttes til forskningsspørsmål 3, og er delt i to underkategorier som tar for seg aktiviteter som blir nevnt i intervjuene med de tre lærerne. *Aktiviteter med teknologi* inneholder sitater som omhandler beskrivelser av aktiviteter som inkluderer bruk av teknologi. Den andre underkategorien, *Aktiviteter uten teknologi*, tar for seg aktiviteter som ekskluderer bruk av teknologi.

I denne konteksten har jeg samlet sitater som tar for seg aktiviteter i matematikkopplæringen, og knyttet dem til aspekter ved algoritmisk tenkning. I undersøkelsen om hvilke aspekter ved algoritmisk tenkning som kommer frem i aktivitetene som nevnes, er tilnærminger ekskludert. Dette er fordi tilnærminger ikke er relevante når det er selve aktiviteten som skal analyseres. Som beskrevet i [3.1.2 Tilnærminger i algoritmisk tenkning](#), handler tilnærminger om hvilken tilnæringsmåte man har til et problem. Tilnærminger som samarbeid og utholdenhet er noe som ikke kan analyseres ut fra en oppgavetekst eller en aktivitet.

Blant aktivitetene som presenteres i kapittel 5 er det kun et utvalg som inkluderes i denne analysen. Det er mange ressurser og aktiviteter som blir nevnt av de tre matematikklærerne. At lærerne konkret trekker frem og beskriver kun noen av dem, tilsier at det mest sannsynlig er disse aktivitetene som blir brukt i deres matematikkundervisning. Utover det, er det utfordrende å vurdere hvilke aktiviteter som er mer eller mindre sannsynlige. Utvalget av aktivitetene er derfor basert på hvilke aktiviteter som er nevnt i intervjuene med de tre lærerne. Det er også viktig å skille aktiviteter fra for eksempel ressurser. I intervjuene ble flere ressurser nevnt, uten ytterligere spesifisering. De er ikke inkludert i analysen.

Grunnlaget for analysen av aktivitetene med teknologi er beskrivelser av aktivitetene som kommer frem i intervjuene. Aktiviteter uten teknologi har et annet grunnlag da de ikke er beskrevet i intervjuene i like stor grad. For å kunne analysere dem, har jeg hentet informasjon om aktivitetene, slik at jeg får et tilstrekkelig grunnlag for vurdering av aktivitetene. Jeg har hentet beskrivelser av aktivitetene uten teknologi fra Nasjonalt senter for mat, helse og fysisk aktivitet (u.å. b). Lærerne nevner aktivitetene uten teknologi, men det er beskrivelsene til Nasjonalt senter for mat, helse og fysisk aktivitet som gir et tilstrekkelig nok grunnlag for å undersøke hvilke aspekter ved algoritmisk tenkning som er fremtredende i de aktuelle aktivitetene.

Da aktivitetene ble vurdert i analysen, ble beskrivelsen sett opp mot kjerneferdighetene i algoritmisk tenkning og plassert innunder passende kjerneferdighet. På den måten ble det synlig hva i aktiviteten som er sammenfallende med de ulike kjerneferdighetene. Ved å analysere aktivitetene slik, vil aspektene ved algoritmisk tenkning som er til stede i aktivitetene synliggjøres.

4.8 Studiens validitet og reliabilitet

Evnen til å være kritisk til egen forskning er avgjørende i bedømmelsen av forskningens kvalitet. Validitet og reliabilitet, hvor man ser på to sider av samme sak, spiller en sentral rolle for kvaliteten.

Studiens validitet og reliabilitet handler om studiens gyldighet, troverdighet og pålitelighet. Validitet knyttes til hvorvidt datamaterialet er gyldig til å svare på studiens forskningsspørsmål (Holme & Solvang, 1996, s. 153). I denne studien kan spørsmål om validitet kobles til om datamaterialet, altså transkripsjonene fra intervjuene og øvrige dokumenter, kan si noe om hvilke kompetanser algoritmisk tenkning har som intensjon å fremme i matematikkopplæringen. Et annet spørsmål om studiens validitet kan være om det valgte rammeverket for algoritmisk tenkning er de rette brillene å bruke for å undersøke datamaterialet for å få svar på forskningsspørsmålene. For å vurdere studiens validitet, må jeg altså undersøke om den valgte metoden legger til rette for dette. Ved å kontrollere valg og steg som er tatt i arbeidet, vil jeg sikre studiens validitet. Studiens validitet er avhengig av at valgene som er tatt, er begrunnet og fornuftige, og at de understøtter mine konklusjoner i studien.

For å kunne undersøke hvilke kompetanser algoritmisk tenkning har som intensjon å fremme i matematikkopplæringen, må det være tydelig hva som er bakgrunnen for implementeringen av algoritmisk tenkning i matematikkopplæringen. Da vil kompetanser og ferdigheter som er tilsiktet gjennom Kunnskapsdepartementets formål med læreplanen, tydeliggjøres. Det vil også fungere som en operasjonalisering av begrepet algoritmisk tenkning, som er gjort i kapittel 3, hvor et rammeverk for algoritmisk tenkning vil fungere som et teoretisk rammeverk i analysen i denne studien. Rammeverket for algoritmisk tenkning gir mulighet til å undersøke oppfatning og forståelse av algoritmisk tenkning. Dette er det relevant å undersøke, fordi matematikklæreres oppfatning og forståelse kan ha en innvirkning på algoritmisk tenkning sin rolle i matematikkundervisningen deres. Det vil også være betydningsfullt å undersøke hvordan algoritmisk tenkning kan komme frem i aktiviteter i matematikkopplæringen.

Postholm og Jacobsen (2018, s. 223) skriver i sin bok om forskningsmetode for lærerstudenter, at validitet deles inn i to deler: Indre og ytre validitet. Indre validitet deles i to igjen: årsakvaliditet og grunnlaget for å si noe om årsak og virkning.

I årsakvaliditet illustreres i hvor stor grad studiens undersøkelser samsvarer med tidligere forskning. Det er altså en avgjørende faktor i studiens validitet å gjøre seg kjent med tidligere forskning om algoritmisk tenkning og plassere min studie i forskningslandskapet til algoritmisk tenkning i matematikkopplæringen. En del av dette arbeidet er å se hva andre har funnet, og om det kan være meningsbærende for egne funn (Postholm & Jacobsen, 2018, s. 221). Dette er blitt gjort i [1.3 Tidligere forskning](#), hvor jeg har gjort rede for relevant forskning som er nærliggende denne studiens formål, og i [kapittel 2](#), hvor blant annet meningsbærende forskning er presentert.

Den andre delen av indre validitet handler om undersøkelsen av det jeg tror eller sier i denne studien, og forteller om jeg har grunnlag for å si noe om årsak og virkning i studien. Postholm og Jacobsen (2018, s. 233) hevder at man i samfunnsvitenskap ikke kan uttale seg om kausale «lover» på lik linje som det gjøres i naturvitenskapen. Dette betyr i beste fall at jeg kan uttale meg om sannsynligheter, og hvorvidt det er mer eller mindre sannsynlig for noe. Satt i kontekst, betyr dette at jeg i denne studien kan i beste fall komme med forslag basert på funn i denne studien. Det må imidlertid ikke tas for gitt at det vil fungere i praksis.

Ytre validitet dreier seg om hvorvidt det er mulig å overføre resultater og funn fra denne studien til en annen kontekst enn den som er undersøkt (Postholm & Jacobsen, 2018, s. 223). Her spiller forståelsen av intensjonen bak algoritmisk tenkning en stor rolle, da forskningslandskapet diskuterer algoritmisk tenkning i matematikkopplæringen som en ny del av matematikk.

Intensjonen som ligger til grunn for algoritmisk tenkning i matematikkopplæringen, er grunnlaget for rammeverket for algoritmisk tenkning i denne studien. Det har vært en viktig del av arbeidet å få avklart hva som er bakgrunnen for at algoritmisk tenkning har fått en plass i den nye læreplanen og matematikkopplæringen. Man kan altså si at rammeverket for algoritmisk tenkning har ytre validitet fordi rammeverket kan benyttes i andre kontekster hvor man undersøker algoritmisk tenkning i en undervisningskontekst. Videre kan man si at svar på forskningsspørsmål 2 og 3, med resultatene og funnene fra denne studien, vil kunne vise hva som er mulig å avdekke av hvilke aspekter ved algoritmisk tenking som kommer frem i en undervisningskontekst.

Studiens reliabilitet knyttes til hvordan jeg eller analyseprosessen kan ha påvirket funn og resultater i studien. Dette krever at jeg reflekterer over min forskerrolle, og hvordan jeg har påvirket denne studien.

Refleksjoner rundt reliabilitet knyttet til intervjuprosessen, handler om hvordan jeg kan ha påvirket intervjuet ved å stille ledende spørsmål, og om hvilke spørsmål jeg har stilt. I intervjuene valgte jeg å inkludere programmering i spørsmålene om algoritmisk tenkning. Hensikten min der og da, var å gi lærerne flere knagger å henge sine refleksjoner på, da både programmering og algoritmisk tenkning er to nye elementer i den nye læreplanen. Likevel ser jeg nå i ettertid at det er mulig det kan ha ledet lærerne til ikke å ta opp algoritmisk tenkning uten programmering. Lærer 3 tar opp algoritmisk tenkning uten teknologi, og det kan dermed argumenteres med at det ikke har en større betydning. Alle de tre lærerne har blitt stilt de samme spørsmålene, men bare Lærer 3 nevner algoritmisk tenkning uten teknologi.

Som jeg nevner i [4.5.2 Intervjuene](#), syns jeg det var utfordrende å stille med konkrete oppfølgings spørsmål som kunne gi ytterligere avklaringer underveis i intervjuene med lærerne.

Videre i arbeid med intervjuene, spiller transkripsjonsarbeidet en viktig rolle i studiens reliabilitet. I denne arbeidsprosessen blir den fysiske tilstedeværelsen abstrahert til en skriftlig form. Det er viktig å være oppmerksom på dette, og gjøre vurderinger og tiltak som unngår tap av viktig informasjon. Dette er noe som kan gi utslag i analysen, og deretter påvirke funn og resultater i studien. Ulike tiltak for å unngå tap av informasjon er beskrevet i [4.7.1 Transkripsjon](#).

En mulig begrensning i denne studien knyttes til forskningsspørsmål 3 som legger vekt på aktiviteter i matematikkopplæringen. Mange av aktivitetene som nevnes i intervjuene, og som kommer frem i øvrige dokumenter, blir bare nevnt uten videre spesifiseringer. Det kan dermed tenkes at det ikke er tilstrekkelig informasjon for å undersøke hvilke aspekter ved algoritmisk tenkning som kommer frem i alle aktivitetene. Dette er en begrensning i studien, men det danner også et grunnlag for videre forskning. Jeg har likevel valgt å undersøke hvilke aspekter ved algoritmisk tenkning som kommer frem i noen av aktivitetene, da jeg mener det bidrar til å gi et mer sammensatt bilde av implementeringen av algoritmisk tenkning i matematikkopplæringen.

4.9 Forskningsetiske betraktninger og hensyn

I henhold til forskningsetikkloven (2017, § 1) har jeg i denne studien tatt etiske hensyn som samsvarer med anerkjente etiske normer. Den nasjonale forskningsetiske komité for samfunnsvitenskap og humaniora (NESH) har utviklet retningslinjer som fungerer som et hjelpemiddel og veiledning i et forskningsarbeid. Ved å følge deres retningslinjer, sikrer jeg meg kunnskap som sier hva jeg bør ta i betraktning og gjøre for at denne studien skal være ansvarlig.

Den nasjonale forskningsetiske komité for samfunnsvitenskap og humaniora (NESH, 2016, s. 12) uttrykker i deres retningslinjer at det må tas hensyn til informantenes rett til selvbestemmelse. Forskere har også en plikt til å respektere informanters privatliv. I forkant av intervjuene ble det derfor sendt en e-post til lærerne med et informasjonsskriv om studien, samt et samtykkeskjema. Lærerne fikk også tilsendt intervjuguiden på e-post før intervjuet.

Ettersom jeg har intervjuet tre matematikklærere med hjelp av lydopptak, var studien meldepliktig til Norsk senter for forskningsdata (NSD), da blant annet stemme på lydopptak regnes som behandling av personopplysninger. Studien er godkjent av NSD. Lydfilene fra intervjuene har i etterkant av intervjuene blitt oppbevart på eksterne harddisker som ikke har tilgang til internett. Gjengivelse av datamaterialet, som transkripsjonene, er anonymisert for å sikre personvern. Alle opplysninger om lærerne har blitt behandlet konfidensielt og i samsvar med personvernregelverket. Det betyr at jeg har byttet ut lærernes navn med koder i forkant av intervju, og tilgang til datamaterialet er adgangsbegrenset. Videre tiltak er beskrevet i [4.4 Utvalg](#).

Artikkel 9 i Personvernforordningen sier at det er tillatt å arkivere personopplysninger med forbehold om at personens rett til vern av personopplysninger ved anonymisering blir opprettholdt (Personopplysningsloven, 2018, § 1). I intervjuene kan det ha kommet frem navn på skole og andre opplysninger som kan identifisere lærerne. På bakgrunn av det, har det blitt bestemt at opptakene ikke skulle arkiveres etter opptakene var ferdig transkribert.

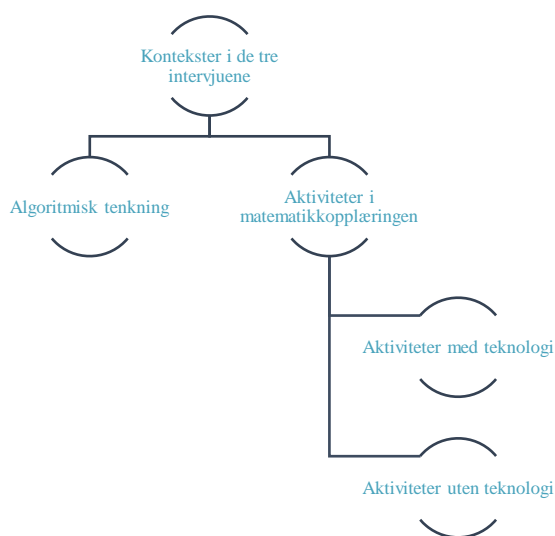
5 Resultater

I dette kapitlet presenteres resultater fra analysen. Datamaterialet består av transkripsjoner av intervju med tre lærere, samt av dokumenter fra to av lærerne. Lærernes beskrivelser og utsagn baserer seg på deres oppfatning, forståelse og erfaringer med algoritmisk tenkning og matematikk. Oversiktsanalysen bygges på nettopp dette, men vil være strukturert etter forskningsspørsmålene, og ikke etter lærerne. Som nevnt i [4.7.2 Oversiktsanalyse](#) i det foregående kapitlet, vil det være nødvendig å ta utgangspunkt i studiens formål og forskningsspørsmål i bestemmelsen av hva som er relevant innhold i datamaterialet. Det blir også et behov for å undersøke om intervjuene faktisk klarer å måle det som de har til hensikt å gjøre.

Jeg har gjennomført en oversiktsanalyse hvor jeg har gått gjennom transkripsjonene, og laget kategorier som baserer seg på ulike kontekster som har kommet frem i intervjuene. Jeg skal nå gå nærmere inn på dette.

5.1 Oversiktsanalyse

Kontekstene som kom frem i intervjuet, er delt opp i to overordnede temaer. De består av *algoritmisk tenkning* og *aktiviteter i matematikkopplæringen*. *Aktiviteter i matematikkopplæringen* er delt opp i underkategoriene *aktiviteter med teknologi* og *aktiviteter uten teknologi*. I figur 5-1, vises de ulike kontekstene.



FIGUR 5-1: OVERSIKT OVER KONTEKSTER KNYTTET TIL ALGORITMISK TENKNING I DE TRE INTERVJUENE

Ved hjelp av disse ulike temaene kan man få muligheten til å få et rikere inntrykk av kontekstene som spiller inn på implementering av algoritmisk tenkning i matematikkopplæringen. Som jeg nevnte i [4.2 Kvalitativt forskningsintervju](#), vil informasjon om kontekster bidra til å forstå interaksjoner og didaktiske metoder som kan spille inn på kvaliteten på implementeringen av algoritmisk tenkning i matematikkopplæringen. Disse kategoriene som nå er kommet frem som temaer fra intervjuene, vil gi meg informasjon om hvilke kontekster som utspiller seg i lærernes matematikkundervisning. Dette er noe som varierer fra lærer til lærer, men det er likevel noen fellestrekk innenfor de nevnte kategoriene.

Av hensyn til studiens omfang, vil jeg ikke inkludere utdrag fra transkripsjonene her, men heller oppsummere, innenfor hvert tema, hva som kommer frem av intervjuene med de tre lærerne.

Jeg skal nå ta for meg intervjutemaene for å se på de ulike kontekstene som spiller inn på implementeringen av algoritmisk tenkning i matematikkopplæringen. Jeg vil legge til relevante uttalelser og kommentarer fra lærerne, og beskrive hva de ulike kontekstene består av.

5.1.1 Algoritmisk tenkning

Denne kategorien inneholder uttalelser og sitater fra lærerne som omhandler deres oppfatning og forståelse av algoritmisk tenkning. Lærernes utgangspunkt er deres egne erfaringer knyttet til deres matematikkundervisning. Det er også relevant hva de sier om hvordan algoritmisk tenkning kan komme frem i deres matematikkundervisning. I intervjuene har lærerne snakket om hva de tenker algoritmisk tenkning kan bety, og hvordan det kan se ut i deres matematikkundervisning. Alle de tre lærerne svarer med sitt utgangspunkt og sine erfaringer, som er med på å forme deres oppfatning av algoritmisk tenkning.

Lærer 1

I intervjuet med Lærer 1 kommer det frem at læreren tror algoritmisk tenkning handler om å bryte opp ting i mindre deler, at det handler om regler og oppskrifter, og at noe dermed gjøres i en bestemt rekkefølge. Læreren sier for eksempel «Det er bare fornemmelse av at det handler om regler og en oppskrift da». Videre sier læreren at algoritmisk tenkning er en del av

matematikkfaget nå, og at elever vil drive med feilsøking i forbindelse med programmeringsaktivitet i matematikk.

Det kommer også frem at Lærer 1 knytter algoritmisk tenkning til programmering. I spørsmål om hva Lærer 1 tenker om kompetansemålene som kan knyttes til algoritmisk tenkning og programmering, sier Lærer 1 at det er fint at målene er konkretisert med begreper som løkker og variabler. Læreren sier videre at i sin undervisning frem til intervjuet fant plass, så har elevene arbeidet litt med variabler, i tråd med kompetansemålene.

Lærer 2

I intervjuet med Lærer 2 kommer det frem at vedkommende forstår algoritmisk tenkning som en problemløsningsmetode hvor man ser på rekkefølger og mønster. Videre i intervjuet sier Lærer 2 at det kreves en viss forståelse for å gjøre ting i en riktig rekkefølge. Lærer 2 nevner ved flere anledninger i intervjuet også en overføringsverdi ved algoritmisk tenkning, hvor det fokuseres på en overføringsverdi til andre problemer og prosesser. Videre sier Lærer 2 at man gjennom algoritmisk tenkning vil kunne se systemer, og tenke gjennom prosessen før man går i gang med problemet.

Lærer 2 ble spurt om algoritmisk tenkning og programmering har vært en del av matematikkundervisningen frem til intervjuet fant plass. Læreren svarte nei, og la til at ønsket for dette var veldig stort, men at det ikke hadde blitt diskutert på lærerteamet. Læreren la også til at det hadde vært en spesiell periode nå under pandemien. I spørsmål til hvilke matematiske områder som er relevant for aktiviteter som Lego League eller aktiviteter med Micro:bit, svarte Lærer 2 algoritmisk tenkning, og utdypet med «at de ser rekkefølger og mønster».

Lærer 3

Lærer 3 påpeker i intervjuet at algoritmisk tenkning blant annet er et verktøy og en problemløsningsmetode, som innebærer at man må arbeide systematisk, tenke strategisk og planlegge. Lærer 3 følger opp med at algoritmisk tenkning blir brukt som et verktøy for finne strategier, og dermed arbeide med ulike matematiske temaer som desimaltall, geometri og statistikk. Læreren eksemplifiserer det og sier «bruke ulike strategier til å regne ut areal og omkrets, der har vi algoritmisk tenkning sant?». Læreren fortsetter, og forteller at

utfordringen med algoritmisk tenkning er å få elevene til å holde ut, eksempelvis «Men min utfordring nå, er at de skal ha tålmodighet i læringen sin, [...] da utvikler du ikke den algoritmiske tenkningen».

Videre i intervjuet trekker Lærer 3 frem at «Evnen til å pugge litt, og tåle å ikke få det til, og prøve igjen og igjen, er også en del av algoritmisk tenkning». I intervjuet blir programmering, algoritmisk tenkning og koding sammenlignet, og det kommer frem at algoritmisk tenkning handler om en tilstedeværelse hvor det å holde ut, og prøve å feile høres med. Å følge en formel, blir også sett på som en del av algoritmisk tenkning, ifølge Lærer 3.

5.1.2 Aktiviteter i matematikkopplæringen

Denne kategorien inneholder sitater og utsagn fra lærerne som handler om aktiviteter i matematikkopplæringen. Kontekster rundt aktivitetene som blir nevnt kan være omstendigheter, elevene og ressurser. Hovedkategorien *Aktiviteter i matematikkopplæringen* er delt opp i to underkategorier: *Aktiviteter med teknologi* og *Aktiviteter uten teknologi*.

5.1.2.1 Aktiviteter med teknologi

Aktiviteter med teknologi innebærer aktiviteter i matematikkopplæringen som inkluderer teknologi. Programmering og koding med digitale verktøy er for eksempel nevnt en god del i intervjuene, og er et eksempel på en aktivitet med teknologi i matematikkopplæringen.

Lærer 1

Lærer 1 sier at programmering er det som får en datamaskin til å utføre noe. På eget initiativ har Lærer 1 prøvd seg på javascript for å få mer kunnskap om råkoding. Lærer 1 forteller at råkoding handler om selve koden som ligger bak blokkene. I intervjuet forteller Lærer 1 om flere programmeringsaktiviteter vedkommende har arbeidet med, og at Lærer 1 har jobbet en god del med programmering i undervisningen. Lærer 1 forteller at skolen har fått Micro:bit av Vilvite-senteret i Bergen. Skolen har egne klassesett med Micro:bit, og en type robotbil som kan programmeres til å kjøre. Lærer 1 sier i intervjuet at klassen derfor har jobbet en god del med programmering.

Videre i intervjuet sier Lærer 1 at programmering kan være litt «gresk», og at det er som å lære seg et nytt språk. Lærer 1 sier også at det beste hadde vært om elevene hadde lært de engelske begrepene for løkke og variabler, rett og slett fordi det ikke finnes et norsk programmeringsspråk. I denne forbindelse uttrykte Lærer 1: «Jeg angrer på at jeg oversatte alt til norsk. Jeg skulle bare latt de lære de engelske ordene med en gang. Det finnes jo egentlig ikke programmering på norsk. Programmering er jo på engelsk».

Det blir også nevnt praktiske aspekter ved programmering, som at det kan være en tidkrevende prosess. Det kommer også frem i intervjuet at det kan være tidkrevende hvis man skriver feil i koden, og må bruke tid på feilsøking. Læreren sier «så lærte jeg jo at det er utrolig tidkrevende fordi hvis du skriver feil i koden din også må du bruke veldig mye tid på å feilsøke, og prøve å rette opp igjen det du har gjort feil».

Lærer 1 påstår at det vil gjøre det lettere for elevene å ha kjennskap til begrepene som er kommet frem i kompetansemålene i den nye læreplanen i matematikk. Læreren sier «[...] og hvis de ikke vet hva en løkke er så må de så de jo skrive en mil med kode». «For at man skal kunne programmere litt ordentlig, så må man kunne ting som løkker og variabler», sier Lærer 1.

Lærer 1 gir også flere eksempler på programmeringsaktiviteter som er benyttet i undervisningen. Det kommer også frem i intervjuet at Lærer 1s skole har flere ansatte som er interessert i programmering, men at man også har flere ansatte som kanskje synes det er krevende og avansert.

I intervjuet forteller Lærer 1 om en teknologimesse som elevene til Lærer 1 deltok på for en stund tilbake. Læreren sier at deres ide var å overføre ren energi fra vindkraft til elbilbatterier. Ifølge læreren programmerte og kodet elevene slik at de overførte signaler fra vindmøllene til bilene, slik at bilene fikk drivkraft. Læreren sier «På standen vår gjorde vi det mulig for publikum å blåse på vindmøllene så bilene kunne kjøre rundt en bane.»

Videre kommer det frem i intervjuet at Lærer 1 har arbeidet en del med programmeringsprogrammet Scratch sammen med elevene sine. I undervisningen har Scratch blant annet blitt brukt til å arbeide med multiplikasjon. Lærer 1 forteller også at Micro:bit er blitt brukt for å arbeide med multiplikasjonsstykker. Læreren sier at aktiviteten er hentet fra kidsakoder.no, og forteller litt om hvordan elevene skal programmere Micro:biten. Det er to knapper på Micro:biten, A og B, hvor A er tiere og B er enere. Når man taster inn riktig svar,

får man et smilefjes, og dersom man taster inn feil svar, får man et surt ansikt. Læreren forteller at noen elever hadde gjort noen feil i programmeringen: «de hadde ikke lagt inn at de skulle legge inn to faktorer for eksempel. Da står det bare faktor og faktor på begge to og da fikk de jo bare null gange null». Lærer 1 sier også at det var noen elever som hadde addert, og ikke multiplisert. På nettsiden Kidsakoder.no finnes det ulike oppgaver knyttet til Scratch, Python, og Micro:bit som Lærer 1 har brukt til å lage introduksjonsvideoer til elevene sine.

Videre forteller Lærer 1 om Mattemestern, et dansk matematikklæreverk som er oversatt til norsk, som de bruker i undervisningen. Lærer 1 trekker frem at i Mattemestern har de et fokus på begreper, i større grad enn det Lærer 1 opplever andre læreverk har.

I intervjuene med lærerne fikk de spørsmål om de inkluderer teknologi i sin undervisning. Lærer 1 har ikke lagt dette inn i sine planleggingsdokumenter, fordi det blir tatt for gitt at det skal benyttes. Lærer 1 sier «Nei, jeg har ikke det inni planene fordi jeg bruker det hver dag». I oppfølgingsspørsmål om hvorfor det er slik, så svarer læreren at det handler om preferanser. Videre sier Lærer 1 at da skolene stengte i mars 2020, hadde elevene flere strenger å spille på under hjemmeundervisningen, fordi de allerede var godt kjente med å bruke teknologi i undervisningen.

Lærer 2

I intervjuet kommer det frem at Lærer 2 har arbeidet litt med programmeringsverktøyet Micro:bit. Det blir midlertidig påpekt at det har stagnert, og at det ikke er blitt jobbet noe særlig med Micro:bit siden i fjor, altså 2020. Lærer 2 sier at vedkommende har gjort aktiviteter med Micro:bit, som å få av og på lys, og at det er arbeidet med løkker. Lærer 2 nevner også Lego League, og sammenligner det med Micro:bit, og uttrykker at Lego League har litt mer variabler. Det kommer også frem at Lærer 2 har fokus på matematiske områder som geometri, rotasjon, hastighet og radius i forbindelse med programmering.

På spørsmål om hvor ofte teknologi blir brukt i undervisningen, svarer Lærer 2 at Chromebooks blir brukt i de alle fleste undervisningstimene. De fikk imidlertid Chromebooks først i forbindelse med hjemmeskolen våren 2020. Kikora, et digitalt læremiddel, blir også brukt en god del på skolen og i matematikkundervisningen. Læreren sier også at skolen har vært med i Lego League tidligere, men at de nå har utdaterte roboter som de håper de kan få

erstattet, slik at de kan få delta igjen. De har også vært med på Mattemaraton, som er i regi av Kikora.

Videre i intervjuet forteller Lærer 2 at elevene fra tid til annen spilte spill i noe som ble kalt for «klassens time». Læreren sier at spillene er kognitive spill som krever at ting skal gjøres i riktig rekkefølge, og at elevene derfor må utarbeide viss form for strategier. I samtaler med elevene har Lærer 2 en intensjon om å gjøre dem bevisste på matematikken i spillene. Lærer 2 sier også at elevene har arbeidet med Excel i forbindelse med arbeide med multiplikasjon.

Lærer 3

Lærer 3 forteller om ulike typer programmeringsaktiviteter som anvendes på de ulike trinnene på skolen. Lærer 3 ser på programmering som veldig teknisk, og bestående av tall og koder. Det blir blant annet sagt at med «programmering så ser jeg for meg bare tall og koder [...] programmering hører til på et høyere nivå, men det gjør kanskje ikke det».

I intervjuet blir programmering, algoritmisk tenkning og koding sammenlignet, og det kommer frem at Lærer 3 ser på programmering som noe som skal være korrekt, og at eventuelle feil blir så synlige, mens koding er noe mer visuelt, og handler om at man skal få noe til å gjøre noe. Lærer 3 sier at koding handler om å planlegge, slik at man får en mindre vei å gå til målet.

I intervjuet snakker Lærer 3 om «unplugged» og «plugged» koding som, ifølge Lærer 3, betyr digital og ikke-digital koding. I intervjuet forteller Lærer 3 at skolen har en faggruppe i matematikk som har laget et dokument med en oversikt over hva man skal forstå med koding, og hvordan det skal implementeres på alle trinn. I fagplanen de har laget, har de tatt hensyn til både digitale og ikke-digitale aktiviteter. Dokumentet inneholder blant annet diverse forslag til aktiviteter. Jeg går nærmere inn på dokumentet i [5.2 Resultater fra øvrige dokumenter i datamaterialet](#).

På trinnet til Lærer 3 har de ikke læreverker i matematikk for tiden. Læreren sier det er skummelt, fordi de ikke har gjort det før. «Ved å jobbe på denne måten, må de fokusere på å verifisere at de lærer det de skal lære, og at de ikke sitter med nesen i boken og blir dødsgod i Multi», sier læreren.

5.1.2.2 *Aktiviteter uten teknologi*

Aktiviteter uten teknologi er aktiviteter i matematikkopplæringen som ekskluderer teknologi. Aktiviteter uten teknologi som lærerne kobler til algoritmisk tenkning, blir presentert under.

Lærer 1

I intervjuet nevner Lærer 1 flere aktiviteter med teknologi som Lærer 1 koblet til algoritmisk tenkning og programmering. Det nevnes derimot ikke noen konkrete aktiviteter uten teknologi, men Lærer 1 sier likevel at det finnes veldig få begrensinger på hva som kan kobles til algoritmisk tenkning. Læreren sier «Du kan gjøre det så matematisk som du vil». Læreren eksemplifiserer det, og sier at man for eksempel kan bruke tall og koordinater.

Lærer 2

I intervjuet kommer det ikke frem noen eksempler på aktiviteter uten teknologi i matematikkopplæringen. Lærer 2 nevner samtidig algoritmisk tenkning som et matematisk tema. Til spørsmål om hvilke matematiske områder Lærer 2 kan knytte til arbeid med Lego League og Micro:bit, svarer Lærer 2 algoritmisk tenkning og utdyper at det er viktig at elevene ser rekkefølger og mønster.

Lærer 3

I intervjuet kommer det frem to konkrete eksempler for hva aktiviteter uten teknologi kan være, nemlig lekene Rødt lys og Kongen befaler. Lærer 3 fortsetter, og sier «det handler litt om at man skal tenke strategisk og at man skal planlegge da».

Som nevnt over, forteller Lærer 3 om en matematikkfaggruppe som har laget en oversikt over hva elever skal forstå med koding, og hvordan det skal implementeres på alle trinn. Her har oversikten og fagplanen tatt hensyn til både «digitale» og «ikke-digitale» aktiviteter. Læreren sier videre: «i tillegg så har vi faktisk hatt en mann inne som har vist oss litt om ikke-digital koding.»

5.2 Resultater fra øvrige dokumenter i datamaterialet

Øvrige dokumenter i datamaterialet omfatter andre dokumenter enn transkripsjonene fra intervjuene med de tre lærerne. Under presenterer jeg relevant informasjon fra disse dokumentene.

5.2.1 Planleggingsdokument

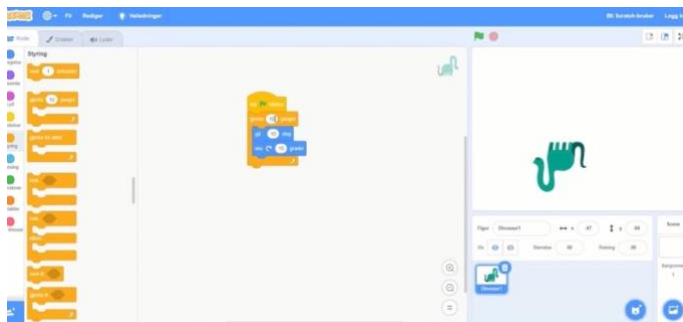
Lærer 1 har gitt meg et planleggingsdokument som tar for seg matematiske temaer i matematikkundervisningen, hvor det er strukturert på hvilke tidspunkter elevene skal arbeide med de ulike matematiske temaene i løpet av skoleåret 2020/2021. Dette dokumentet sier imidlertid ikke noe eksplisitt om algoritmisk tenkning eller programmering. I lys av at Lærer 1 uttrykte at bruk av teknologi i undervisningen er noe som tas for gitt, er ikke teknologi inkludert i planleggingsdokumentet. Lærer 1 sier også i intervjuet at planen er en grovkisse.

5.2.2 Aktivitet i Scratch – Hvordan lage en løkke

Lærer 1 har delt to egenkomponerte videoer. Den ene videoen handler om hvordan man skal lage en løkke i Scratch. Den andre videoen tilhører norskfaget, og blir dermed ikke inkludert i denne studien.

Videoen er laget av Lærer 1, hvor Lærer 1 viser elevene sine hvordan man skal lage en løkke i programmet Scratch. Hensikten med denne aktiviteten er altså at elevene skal kunne øve på å lage en løkke i Scratch.

I figur 5-2 vises en skjermdump av videoen, hvor læreren har lagt inn en løkke i koden. Læreren tar opp begreper som grader og dobling underveis i videoen. Kodesetningen som blir lagt inn legger opp til at figuren skal gå og snu. Det må legges inn antall skritt og antall grader per hendelse, og dette eksperimenterer læreren litt med, for å vise elevene hva som skjer når man endrer antall skritt, men beholder antall grader, eller hvor mange ganger hendelsen skal gjenta seg. Læreren legger inn at løkken skal gå ti ganger, og stiller spørsmål om hvor mange skritt og grader det da blir. Til slutt legger Lærer 1 inn at hendelsen skal gjentas for alltid. Da stopper ikke hendelsen før man fysisk stopper den. Lærer 1 forteller at elevene skal få en figur til å bevege seg. Det betyr altså ikke at elevene må gjøre nøyaktig det samme som læreren har gjort.



FIGUR 5-2: SKJERMDUMP FRA VIDEO SOM LÆRER 1 HAR LAGET OM LØKKE I SCRATCH

5.2.3 Aktivitet med Micro:bit fra Kidsakoder.no

Som nevnt i [5.1.2.1 Aktiviteter med teknologi](#), har Lærer 1 arbeidet med Micro:bit og multiplikasjon, en aktivitet som er hentet fra Kidsakoder.no. Clausen et al. (u.å.) har laget spillet, og skriver i introduksjonen til spillet at det er modifiserbart. Lærer 1 beskriver aktiviteten i intervjuet, og viser til nettsiden med instruksjoner til aktiviteten. Læreren sier det er greie instruksjoner i beskrivelsen og at man kan krysse ut stegene etter hvert som man er ferdig med dem.

I instruksene på nettsiden er det beskrivelser med ord som er supplert med bilder av hvordan det vil se ut når man arbeider med stegene. Se figur 5-3 for skjermdump av nettsiden.



FIGUR 5-3: SKJERMDUMP AV EKSEMPEL PÅ AKTIVITET PÅ KIDSAKODER.NO (U.Å.)
[HTTPS://OPPGAVER.KIDSAKODER.NO/MICROBIT/PXT_GANGESPILL/GANGESPILL](https://oppgaver.kidsakoder.no/microbit/pxt_gangespill/gangespill)

Denne aktiviteten legger opp til at elevene får muligheten til å programmere en Micro:bit, slik at de kan øve på multiplikasjonsstykker. Det siste steget i oppgavene oppfordrer også elevene til å modifisere oppgaven ved å lage addisjonsstykker, eller andre lignende spill.

5.2.4 Dokument fra faggruppen i matematikk

Lærer 3 har gitt meg et dokument som er utarbeidet av faggruppen i matematikk på skolen. I dette dokumentet presenteres programmering, algoritmisk tenkning og koding i matematikk på de ulike trinnene på barneskolen til Lærer 3. Dokumentet er 16 sider langt og inneholder overskriftene *Hva er formålet med koding? Hva må til for at vi kan gjennomføre koding på de ulike trinnene på denne skolen? Hva må til for å drive med programmering? Kompetansemål og hva som skal gjøres hvert år? Idébank, og Erfaringer.*

Under overskriften *Hva er formålet med koding?* presenteres algoritmisk tenkning og programmering. Algoritmisk tenkning blir i dokumentet beskrevet som en tankemåte som er løftet frem i den nye læreplanen. Videre blir det beskrevet som en problemløsningsmetode, og at dette innebærer å tilnærme seg problemer på en systematisk måte. Det står også at «å tenke på denne måten innebærer at det må vurderes hvilke steg som skal til for å løse et problem, og å kunne bruke sin teknologiske kompetanse for å få en datamaskin til å løse (deler av) problemet».

Figuren om algoritmisk tenkning fra Utdanningsdirektoratets artikkel er med i dokumentet. Figuren er inntatt i [2.1.1 Utdanningsdirektoratet om algoritmisk tenkning](#). Under figuren i dokumentet til Lærer 3, kommer faggruppens tolkning av figuren frem, hvor de skriver at komplekse problemer brytes ned til mindre, mer håndterlige, delproblemer som kan løses. De skriver videre at dette inkluderer å organisere og analysere informasjon på en logisk måte, og lage fremgangsmåter for å komme frem til ønsket løsning. De konkluderer med at man må være systematisk, analytisk, skapende, eksperimenterende og åpen for alternative løsninger. For å være dette, må man være nysgjerrig og utforskende i tilnærmingene for å formulere og løse problemer.

Under *Hva må til for å gjennomføre koding på alle trinnene på denne skolen?* nevnes ulike praktiske behov som sikter til teknologisk utstyr, nærmere sagt Chromebooks. I avsnittet *Hva må til for å drive med programmering?* er det foreslått at første til fjerde trinn kan ha fokus på «unplugged» og «plugged», i tillegg til noen enkelte økter som er digitale. For femte til syvende trinn er det foreslått digitale programmeringsaktiviteter i for eksempel Salaby, Scratch og Micro:bit.

Under overskriften *kompetansemål og hva som skal gjøres hvert år* listes trinnene opp med relevante kompetansemål, etterfulgt av forslag til aktiviteter som er knyttet til kompetansemålene. Under har jeg oppsummert dette i tabell 5-1. Tabellen viser at trinnene har ulike kompetansemål, men forslag til aktiviteter endrer seg ikke i like stor grad. På mellomtrinnet (5-7) har de ulike kompetansemål for hvert år, men samme aktiviteter.

TABELL 5-1: EGENKOMPONERT OVERSIKT OVER KOMPETANSEMÅL OG FORSLAG TIL AKTIVITETER SOM VISES I DOKUMENTET TIL LÆRER 3

Trinn	Kompetansemål	Forslag til aktiviteter
1. og 2. trinn	Lage og følge regler og trinnvise instruksjoner i lek og spill	Aktiviteter med teknologi
		<ul style="list-style-type: none"> - Kode-timen (arbeid med blokk-koding) - Scratch Jr. - Arbeid med koderobot på SFO
3. trinn	Lage og følge regler og trinnvise instruksjoner i lek og spill knyttet til koordinatsystemet	Aktiviteter uten teknologi
		<ul style="list-style-type: none"> - Kongen befaler - Rødt lys - Fugl, fisk, midt imellom
4. trinn	Lage algoritmer og uttrykke de ved bruk av variabler, vilkår og løkker	Aktiviteter med teknologi
		<ul style="list-style-type: none"> - Kode-timen (arbeid med blokk-koding) - Scratch Jr. - Salaby.no – kodeskolen
3. trinn	Lage algoritmer og uttrykke de ved bruk av variabler, vilkår og løkker	Aktiviteter uten teknologi
		<ul style="list-style-type: none"> - Skattejakt - Kongen befaler - Battleship - Tampen brenner - Hinderløype med instruksjoner og koder - Rødt lys - Fugl, fisk, midt imellom - Lage oppskrifter for hvordan for eksempel man skal lage og spise en skive - «Blindebukk»-koding
4. trinn	Lage algoritmer og uttrykke de ved bruk av variabler, vilkår og løkker	Aktiviteter uten teknologi
		<ul style="list-style-type: none"> - Skattejakt - Kongen befaler - Battleship - Tampen brenner - Hinderløype med instruksjoner og koder - Rødt lys - Fugl, fisk, midt imellom - Lage oppskrifter for hvordan for eksempel man skal lage og spise en skive

		<ul style="list-style-type: none"> - «Blindebukk»-koding
5. trinn	Lage og programmere algoritmer med bruk av variabler, vilkår og løkker	<p>Aktiviteter med teknologi</p> <ul style="list-style-type: none"> - Kode-timen (arbeid med blokk-koding) - Scratch Jr. - Salaby.no – kodeskolen - Arbeid med Micro:bit - Arbeid med Micro:bot
		<p>Aktiviteter uten teknologi</p> <ul style="list-style-type: none"> - Skattejakt - Kongen befaler - Battleship - Tampen brenner - Hinderløype med instruksjoner og koder - Rødt lys - Fugl, fisk, midt imellom - Lage oppskrifter for hvordan for eksempel man skal lage og spise en skive - «Blindebukk»-koding
6. trinn	Bruke variabler og formler til å uttrykke sammenhenger i praktiske situasjoner	<p>Aktiviteter med teknologi</p> <ul style="list-style-type: none"> - Kode-timen (arbeid med blokk-koding) - Scratch Jr. - Salaby.no – kodeskolen - Arbeid med Micro:bit - Arbeid med Micro:bot
	Bruke variabler, løkker, vilkår og funksjoner i programmering til å utforske geometriske figurer og mønster	<p>Aktiviteter uten teknologi</p> <ul style="list-style-type: none"> - Skattejakt - Kongen befaler - Battleship - Tampen brenner - Hinderløype med instruksjoner og koder - Rødt lys - Fugl, fisk, midt imellom - Lage oppskrifter for hvordan for eksempel man skal lage og spise en skive - «Blindebukk»-koding
7. trinn	Bruke programmering til å utforske data i tabeller og datasett	<p>Aktiviteter med teknologi</p> <ul style="list-style-type: none"> - Kode-timen (arbeid med blokk-koding) - Scratch Jr. - Salaby.no – kodeskolen - Arbeid med Micro:bit - Arbeid med Micro:bot
		<p>Aktiviteter uten teknologi</p> <ul style="list-style-type: none"> - Skattejakt - Kongen befaler - Battleship - Tampen brenner - Hinderløype med instruksjoner og koder - Rødt lys - Fugl, fisk, midt imellom - Lage oppskrifter for hvordan for eksempel man skal lage og spise en skive - «Blindebukk»-koding

Under *Idébank* vises ulike ressurser og forslag til introduksjonsvideoer og aktiviteter. Nettsiden kidsakoder.no nevnes, sammen med en video fra NRK Super. Det er også lenker til forslag til oppgaver i Scratch fra en blogg.

Avsnittet om erfaringer viser informasjon om erfaringer fra ulike aktiviteter som er prøvd ut. På tredje trinn skriver de at de har forsøkt «unplugged-koding», uten å utdype. På fjerdetrinnet har de startet med Kodeskolen til Salaby. Foreldrene på trinnet har fått informasjon om dette, slik at det er noe som kan arbeides med hjemme også. På sjettetrinnet har de startet med «unplugged-koding». De skriver også at de er godt i gang med Kodeskolen på Salaby, og at elevene har begynt å reflektere rundt at dette handler om å planlegge mønster før man trykker.

På syvende trinn har de arbeidet med «unplugged-koding» i flere fag. Videre har de arbeidet med Scratch, Lego Mindstorm og Micro:bit. De har også vært på Vilvite-senteret i Bergen i forbindelse med programmeringsaktiviteter der. Elevene har også forsøkt å lære bort Scratch til elever på tredjetrinnet. Dette har fungert fint med en oppskrift.

5.3 Oppsummering av resultater

Frem til nå har vi sett på de ulike kontekstene som kommer frem i intervjuene. Jeg har nå gjort rede for lærernes sitater, som gir innsikt i deres forståelse og oppfatning av algoritmisk tenkning. Jeg har også presentert hvilke aktiviteter i matematikkopplæringen som er nevnt i intervjuene, og som kommer frem i dokumenter som er gitt av Lærer 1 og Lærer 3.

Intensjonen med å sortere innholdet i de tre intervjuene, har vært å skape et helhetlig inntrykk av intervjuene. Hensikten har også vært å undersøke om intervjuene faktisk klarer å måle det de har til hensikt å måle. Gjennom denne bearbeidelsen har jeg gjort datamaterialet mer tilgjengelig for den kommende analyseprosessen. Under vil jeg presentere en oppsummering av funn som kan bidra til å svare på studiens forskningsspørsmål.

5.3.1 Algoritmisk tenkning

Når jeg har sett på datamaterialet, har det vært nødvendig å ha studiens formål og forskningsspørsmål i bakholdet. De tre lærernes oppfatning og forståelse av algoritmisk tenkning er med på å synliggjøre intensjonen til aktivitetene som kommer frem i lærernes matematikkopplæring. Innsikt i deres forståelse og oppfatning av algoritmisk tenkning vil også

kunne bidra til å se hvilke aspekter ved algoritmisk tenkning som kommer frem hos de tre matematikklærerne.

I denne konteksten *Algoritmisk tenkning*, har jeg samlet uttalelser fra lærerne som omhandler definisjoner, oppfatning og forståelse av algoritmisk tenkning. Felles for alle de tre lærerne, er at de mener algoritmisk tenkning handler om å følge en oppskrift, en algoritme eller rekkefølge på noe. Lærer 1 sier i intervjuet at algoritmisk tenkning er knyttet til programmering. Lærer 3 skiller helt tydelig mellom aktiviteter med teknologi knyttet til algoritmisk tenkning, og aktiviteter uten teknologi knyttet til algoritmisk tenkning. Både Lærer 2 og Lærer 3 sier at algoritmisk tenkning er en problemløsningsmetode, og begge sier at algoritmisk tenkning innebærer å tenke systematisk. Lærer 2 snakker også om en overføringsverdi i algoritmisk tenkning. Videre nevner Lærer 3 tilnærminger som utholdenhet og evne til å samarbeide, når vedkommende snakker om algoritmisk tenkning.

I dokumentet om programmering, koding og algoritmisk tenkning, som Lærer 3 har gitt til denne studien, kommer det frem en forståelse av algoritmisk tenkning. Det står i dokumentet at algoritmisk tenkning er en tankemåte som er løftet frem i den nye læreplanen. Videre blir det beskrevet som en problemløsningsmetode, og at dette innebærer å tilnærme seg problemer på en systematisk måte. Det står også at «å tenke på denne måten innebærer at det må vurderes hvilke steg som skal til for å løse et problem, og å kunne bruke sin teknologiske kompetanse for å få en datamaskin til å løse (deler av) problemet».

Figuren fra Utdanningsdirektoratets artikkel om algoritmisk tenkning er vist i dokumentet. Under figuren konkluderer faggruppen i matematikk på Lærer 3 sin skole at man må være systematisk, analytisk, skapende, eksperimenterende og åpen for alternative løsninger. For å være dette, må man være nysgjerrig og utforskende i tilnærmingene for å formulere og løse problemer.

5.3.2 Aktiviteter i matematikkopplæringen

I dokumentet til Lærer 3 vises det til ulike kompetansemål til de samme aktivitetene på de ulike trinnene. Dette kan tyde på at det er en intensjon om utvikling i de samme aktivitetene, hvor man fokuserer på å arbeide med aktivitetene ut fra ulike kompetansemål etter hvert som elevene blir eldre. Ved å arbeide på den måten, kan man si at det er mer sannsynlig at elevene

arbeider med flere aspekter ved algoritmisk tenkning i de samme aktivitetene, fordi fokus og intensjonen ved aktiviteten forandrer seg.

I intervjuene kommer det frem noen eksempler på aktiviteter i de tre lærernes matematikkundervisning. Under gjengis sitater fra lærerne, som handler om slike aktiviteter. Jeg skal nå gå nærmere inn på aktiviteter med teknologi.

5.3.2.1 Aktiviteter med teknologi

Lærer 1 forteller at programmering har vært en stor del av matematikkundervisningen.

Elevene har arbeidet med Micro:bit og Scratch, og fått muligheten til å være med på større prosjekter. Lærer 2 og Lærer 3 kjenner også til Micro:bit. Lærer 2 sier at de ikke har arbeidet noe særlig med Micro:bit etter at den nye læreplanen trådte i kraft. Lærer 2 forteller videre at de har utdaterte roboter, som gjør det vanskelig å delta i konkurranser som Lego League.

Lærer 3 viser et dokument som faggruppen i matematikk har utarbeidet i arbeidet med innføringen av den nye læreplanen. I dokumentet er det forslag til flere aktiviteter knyttet algoritmisk tenkning. Forslag til aktiviteter med teknologi som kommer frem i dokumentet, er vist under i tabell 5-2.

TABELL 5-2: OVERSIKT OVER FORESLÅTTE AKTIVITETER MED TEKNOLOGI PÅ DE ULIKE TRINNENE PÅ SKOLEN TIL LÆRER 3

1. og 2. trinn	3. trinn	4. trinn	5. trinn	6. trinn	7. trinn
Kode-timen (arbeid med blokk-koding)	Kode-timen (arbeid med blokk-koding)	Kode-timen (arbeid med blokk-koding)	Kode-timen (arbeid med blokk-koding)	Kode-timen (arbeid med blokk-koding)	Kode-timen (arbeid med blokk-koding)
Scratch Jr.	Scratch Jr.	Scratch Jr.	Scratch Jr.	Scratch Jr.	Scratch Jr.
Arbeid med koderobot på SFO	Salaby.no – kodeskolen	Salaby.no – kodeskolen	Salaby.no – kodeskolen	Salaby.no – kodeskolen	Salaby.no – kodeskolen
			Arbeid med Micro:bit	Arbeid med Micro:bit	Arbeid med Micro:bit
			Arbeid med Micro:bot	Arbeid med Micro:bot	Arbeid med Micro:bot

I tabellen vises det et tydelig skille mellom småtrinnet og mellomtrinnet på skolen til Lærer 3. Arbeid med Micro:bit og Micro:bot er noe som gjøres først når elevene går på mellomtrinnet.

Forslagene til aktivitetene knyttet til teknologi etterlater likevel et rom for frihet til å velge og forme aktiviteten til å passe med de gitte rammene for undervisningen. Forslagene er ikke spesifisert ytterligere.

Lærer 1 sier i intervjuet at programmeringsspråk kan være krevende å lære seg. Lærer 1 mener derfor at det er en fordel å lære elevene de engelske begrepene med en gang, da programmeringsspråket er på engelsk. Læreren angret på at elevene har lært de norske begrepene først. Lærer 1 har forsøkt å sette seg inn i javascript på egenhånd, mens Lærer 3 har jobbet litt med å sette seg inn i hva programmering, koding og algoritmisk tenkning er.

Lærer 1 forteller at vedkommende bruker Mattemestern og Kidsakoder.no i undervisningen for å lage oppgaver og introduksjonsvideoer til elevene sine. Lærer 2 nevner Kikora som et digitalt læremiddel som de tar i bruk på skolen. Lærer 3 bruker ikke lærebok i matematikk dette skoleåret, og sier at det er skummelt. Ved å jobbe på denne måten, må de fokusere på å verifisere at elevene lærer det de skal lære, og at de ikke sitter med nesen i boken og blir "dødsgod i Multi", som Lærer 3 uttrykte.

5.3.2.2 Aktiviteter uten teknologi

Lærer 1 nevner ikke noen konkrete aktiviteter uten teknologi, men sier likevel at det finnes veldig få begrensinger på hva som kan kobles til algoritmisk tenkning. Læreren sier «Du kan gjøre det så matematisk som du vil». Lærer 2 nevner ingen aktiviteter uten teknologi, men det kommer frem at Lærer 2 ser på algoritmisk tenkning som et matematisk tema. På spørsmål om hvilke matematiske områder Lærer 2 kan knytte til arbeid med Lego League og Micro:bit, svarer Lærer 2 algoritmisk tenkning, og utdyper at det er viktig at elevene ser rekkefølger og mønstre.

Lærer 3 skiller helt tydelig mellom aktiviteter med og uten teknologi. I dokumentet fra Lærer 3 sin skole beskriver de forslag til aktiviteter uten teknologi som de knytter de ulike kompetansetrinn på de ulike trinnene på skolen. Under har jeg samlet dem i tabell 5-3.

Tabellen tydeliggjør at antall forslag for aktiviteter uten teknologi øker betraktelig fra og med tredjetrinnet. Det er ikke et skille mellom småtrinnet og mellomtrinnet. Fra tredjetrinnet til syvende trinn er aktivitetsforslagene de samme. Det er ikke spesifisert nærmere hva slags hinderløyper det er snakk om, men det eksemplifiseres hvordan elever kan lage en oppskrift på for eksempel hvordan man kan lage og spise en brødskive.

TABELL 5-3: OVERSIKT OVER FORESLÅTTE AKTIVITETER UTEN TEKNOLOGI PÅ DE ULIKE TRINNENE PÅ SKOLEN TIL LÆRER 3

1. og 2. trinn	3. trinn	4. trinn	5. trinn	6. trinn	7. trinn
Kongen befaler	Skattejakt	Skattejakt	Skattejakt	Skattejakt	Skattejakt
Rødt lys	Kongen befaler	Kongen befaler	Kongen befaler	Kongen befaler	Kongen befaler
Tampen brenner	Battleship	Battleship	Battleship	Battleship	Battleship
	Tampen brenner	Tampen brenner	Tampen brenner	Tampen brenner	Tampen brenner
	Hinderløype med instruksjoner og koder	Hinderløype med instruksjoner og koder	Hinderløype med instruksjoner og koder	Hinderløype med instruksjoner og koder	Hinderløype med instruksjoner og koder
	Rødt lys	Rødt lys	Rødt lys	Rødt lys	Rødt lys
	Lage oppskrifter for hvordan for eksempel man skal lage og spise en skive	Lage oppskrifter for hvordan for eksempel man skal lage og spise en skive	Lage oppskrifter for hvordan for eksempel man skal lage og spise en skive	Lage oppskrifter for hvordan for eksempel man skal lage og spise en skive	Lage oppskrifter for hvordan for eksempel man skal lage og spise en skive
	«Blindebukk»-koding	«Blindebukk»-koding	«Blindebukk»-koding	«Blindebukk»-koding	«Blindebukk»-koding

I intervjuet med Lærer 3 presenteres to eksempler på aktiviteter uten teknologi som kan relateres til algoritmisk tenkning. Læreren fortsetter og sier «det handler litt om at man skal tenke strategisk og at man skal planlegge...». Aktivitetene er lekene Kongen befaler og Rødt lys. Begge aktivitetene vises i tabell 5-3, på alle trinn.

Med de ovenfor nevnte kontekstene som utgangspunkt, har jeg bearbeidet datamaterialet, og gjort det mer tilgjengelig for den kommende analysen. I analysen vil jeg forsøke å belyse funn som kan bidra til å svare på forskningsspørsmål 2 og 3, om hvilke aspekter ved algoritmisk tenkning som kommer frem i lærernes forståelse og oppfatning av algoritmisk tenkning, og matematikkaktiviteter som er nevnt av de tre lærerne. Med et utgangspunkt i rammeverket for algoritmisk tenkning, som er gjort rede for i [kapittel 3](#), vil resultatene fra dette kapittelet blir analysert i neste kapittel.

6 Analyse

Analysekapittelet er strukturert etter de ulike kontekstene som er presentert i det foregående kapittelet, og starter med konteksten *Algoritmisk tenkning*. Deretter går jeg videre til *Aktiviteter i matematikkopplæringen*. I hver kontekst vil analysen være strukturert etter de ulike aspektene i rammeverket for algoritmisk tenkning. Med et utgangspunkt i rammeverket for algoritmisk tenkning, søker studien å finne ut hvilke aspekter ved algoritmisk tenkning som kommer frem i lærernes forståelse og oppfatning av algoritmisk tenkning. I tillegg ønsker jeg å undersøke hvilke aspekter ved algoritmisk tenkning som kan komme frem i aktiviteter i matematikkopplæringen.

6.1 Algoritmisk tenkning

Et viktig formål i denne konteksten, er å få innsikt i lærernes oppfatning og forståelse av algoritmisk tenkning på en slik måte at aspekter ved algoritmisk tenkning er representert. Rammeverket for algoritmisk tenkning vil gjøre det mulig å identifisere de ulike aspektene av algoritmisk tenkning i lærernes forståelse og oppfatning av algoritmisk tenkning. Analysen er strukturert etter rammeverket for algoritmisk tenkning som er beskrevet i [kapittel 3](#).

6.1.1 Kjerneferdigheter i algoritmisk tenkning

Hvilke sitater og uttalelser fra intervjuene med de tre lærerne som er sammenfallende med de fem kjerneferdighetene, er undersøkt. Under presenteres sitater og uttalelser i sortert rekkefølge etter kjerneferdighetene i algoritmisk tenkning.

6.1.1.1 Abstraksjon

I intervjuet med Lærer 3, kommer det frem at vedkommende mener algoritmisk tenkning blant annet innebærer at man må tenke systematisk. Lærer 3 sier senere i intervjuet at algoritmisk tenkning kan bli brukt som et verktøy for å finne strategier. Lærer 2 sier at man gjennom algoritmisk tenkning vil kunne se at det er et system, og at man derfor må tenke gjennom arbeidsprosessen før man går i gang med problemet.

6.1.1.2 Algoritmebehandling

Alle de tre lærerne sier i intervjuene at de mener algoritmisk tenkning handler om å følge en oppskrift, en algoritme eller en rekkefølge i en arbeidsprosess for å løse et problem. Lærer 1 sier «Det er bare en fornemmelse av at det handler om regler og oppskrifter». Lærer 2 snakker om at algoritmisk tenkning krever at man gjør ting i en bestemt rekkefølge. Lærer 3 mener at det å følge en formel, er også en del av algoritmisk tenkning.

6.1.1.3 Generalisering

Lærer 2 knytter algoritmisk tenkning til rekkefølger og mønstre. Lærer 2 sier også flere ganger i intervjuet at algoritmisk tenkning innebærer et fokus på en overføringsverdi til andre problemer og arbeidsprosesser. Lærer 3 snakker om at algoritmisk tenkning kan også blir brukt som et verktøy for å finne strategier.

6.1.1.4 Automatisering

Lærer 1 nevner ikke automatisering, men sier blant annet at bruk av teknologi er noe som tas for gitt at skal være en del av undervisningen. Det kan dermed tenkes at aspektet automatisering er et aspekt Lærer 1 tar for gitt, og at denne forståelsen ikke kommer frem i intervjuet.

Lærer 2 og Lærer 3s forklaring av algoritmisk tenkning som en problemløsningsmetode hvor man må tenke systematisk, og at dette kan fungere som et verktøy for å finne strategier, kan forstås som en del av kjerneferdigheten *automatisering*. Det forklares ikke ytterligere hva som menes med systematisk og strategisk. Denne kjerneferdigheten handler om å finne den mest mulige effektive arbeidsprosessen for å løse et problem. Man kan med dette forstå at ved å være systematisk og strategisk, har man som hensikt å finne den mest effektive løsningen på et problem.

6.1.1.5 Dekomponering

Lærer 1 snakker om at læreren tror algoritmisk tenkning handler om å bryte opp ting i mindre deler. Verken Lærer 2 eller Lærer 3 nevner noe som kan falle innunder denne kjerneferdigheten.

6.1.2 Tilnæringer i algoritmisk tenkning

Hensikten er å belyse hvilke tilnæringer i algoritmisk tenkning som kommer frem i intervjuene. Intensjonen er altså ikke å måle hvilke tilnæringer i algoritmisk tenkning som kommer frem hos elever. Ved å inkludere tilnæringer i dette rammeverket, vil jeg få innsikt i hvordan lærerne identifiserer elevers redskap for utvikling og læring for å vise sin kompetanse i algoritmisk tenkning. Med dette tas det for gitt at algoritmisk tenkning kan ses på som en kompetanse.

6.1.2.1 Feilsøking

Lærer 1 nevner feilsøking i forbindelse med programmeringsverktøy. Læreren sier «så lærte jeg jo at det er utrolig tidkrevende fordi hvis du skriver feil i koden din også må du bruke veldig mye tid på å feilsøke, og prøve å rette opp igjen det du har gjort feil».

6.1.2.2 Kreativitet

Det kommer ikke frem noe konkret fra intervjuene med de tre lærerne som kan knyttes til tilnærmingen *kreativitet*.

6.1.2.3 Samarbeid

Lærer 3 nevner samarbeid som en del av algoritmisk tenkning, og sier i den forbindelse at det kan være utfordrende for elevene å «... samarbeide. Hvor de kanskje tør å samarbeide og snakke sammen i den sårbare prosessen hvor man ikke vet alt».

6.1.2.4 Utholdenhet

Lærer 3 sier at det er en utfordring å få elevene til å holde ut i arbeidsprosessen. Da får man ikke utviklet den algoritmiske tenkningen, sier læreren. Lærer 3 mener at algoritmisk tenkning handler om en tilstedeværelse hvor det å holde ut og prøve å feile, hører med.

6.1.3 Dimensjoner av algoritmisk tenkning

Lærer 3 nevner blant annet at algoritmisk tenkning kan benyttes som et verktøy for å finne strategier. Dette er noe som kan forstås som uavhengig av dimensjonene med og uten teknologi.

I intervjuene med de tre lærerne, har mine spørsmål ofte inkludert programmering i forbindelse med algoritmisk tenkning. Som nevnt i [4.8 Studiens validitet og reliabilitet](#), har min hensikt vært å gi lærerne flere kort å spille på, siden både algoritmisk tenkning og programmering er nye elementer i den nye læreplanen. Jeg må imidlertid være oppmerksom på at dette kan være en mulig feilkilde, da det kan ha ledet lærerne til å holde seg i dimensjonen med teknologi i algoritmisk tenkning, og svare på spørsmål som angår algoritmisk tenkning og programmering, og ikke algoritmisk tenkning alene. Det kan likevel argumenteres for at dette er mindre sannsynlig, fordi Lærer 3 nettopp nevner algoritmisk tenkning uten teknologi.

6.1.3.1 Algoritmisk tenkning med teknologi

Lærer 1 sier i intervjuet at algoritmisk tenkning er knyttet til programmering. Læreren beskriver programmering som det som får en datamaskin til å utføre noe. Lærer 1 gir et inntrykk av at vedkommende og flere andre lærere på skolen har en interesse for programmering, og Lærer 1 har også satt seg inn i programmering på eget initiativ.

Lærer 2 hevder algoritmisk tenkning er et matematisk område og en problemløsningsmetode. Læreren mener også det er en overføringsverdi til andre prosesser og problemer. Dette blir nevnt i sammenheng med arbeid med aktiviteter med teknologi. Dermed kan det forstås slik at Lærer 2s forståelse og oppfatning av algoritmisk tenkning kan knyttes til aktiviteter med teknologi.

Lærer 3 skiller helt tydelig mellom aktiviteter med og uten teknologi som kan knyttes til algoritmisk tenkning.

6.1.3.2 *Algoritmisk tenkning uten teknologi*

I dokumentet fra Lærer 3, står det blant annet at algoritmisk tenkning er en tankemåte. Videre blir det sett på som en problemløsningsmetode. Dette er også noe man kan se i sammenheng med Lærer 3 sin beskrivelse av algoritmisk tenkning. Lærer 3 sier også at algoritmisk tenkning er et verktøy og en problemløsningsmetode, men også at algoritmisk tenkning kan innebære at man arbeider systematisk, strategisk og at man følger en formel.

6.1.4 Forekomst av aspekter ved algoritmisk tenkning i de tre lærernes forståelse og oppfatning av algoritmisk tenkning

Kjerneferdighetene *automatisering* og *abstraksjon* kan forstås som å tenke strategisk og systematisk, men begge med ulike formål. Når Lærer 2 og 3 sier at algoritmisk tenkning kan være en problemløsningsmetode hvor man tenker systematisk og strategisk uten å utdype, kan det være utfordrende å forstå formålet og intensjonen. Dette gjør at jeg har plassert disse sitatene og utsagnene fra Lærer 2 og 3 under begge kjerneferdighetene.

I tabell 6-1 vises oversikt over aspekter ved algoritmisk tenkning som kommer frem i intervju med lærerne. Felles for alle tre lærerne, er at kjerneferdigheten *algoritmebehandling* kommer frem i deres forståelse og oppfatning av algoritmisk tenkning. Tilnærmingen *kreativitet* er det eneste aspektet ved algoritmisk tenkning som ikke blir nevnt av noen av lærerne. Lærer 1 er den eneste som eksplisitt uttrykker at *dekomponering* er en del av algoritmisk tenkning. Lærer 3 er alene om å skille mellom algoritmisk tenkning med og uten teknologi.

TABELL 6-1: OVERSIKT OVER ASPEKTER VED ALGORITMISK TENKNING SOM KOMMER FREM I INTERVJU MED LÆRERNE

Kjerneferdigheter i algoritmisk tenkning	Forekomst i intervju		
	Lærer 1	Lærer 2	Lærer 3
Abstraksjon		⊗	⊗
Algoritmebehandling	⊗	⊗	⊗
Generalisering		⊗	⊗
Automatisering		⊗	⊗
Dekomponering	⊗		
Tilnærminger	Forekomst i intervju		
Feilsøking	⊗		
Kreativitet			
Samarbeid			⊗
Utholdenhet			⊗
Dimensjoner av algoritmisk tenkning	Forekomst i intervju		
Uten teknologi			⊗
Med teknologi	⊗	⊗	⊗

6.2 Aktiviteter i matematikkopplæringen

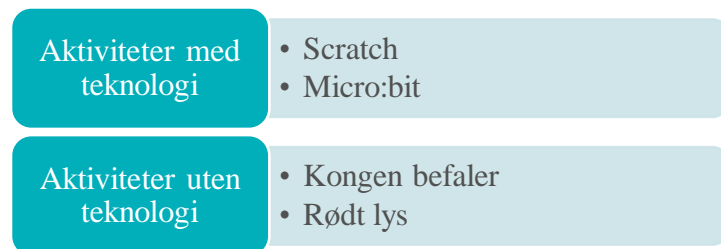
Med utgangspunkt i samme rammeverk for algoritmisk tenkning som er benyttet over, ønsker jeg å undersøke hvilke aspekter ved algoritmisk tenkning som kan komme frem i noen av de matematiske aktiviteter som er blitt presentert i [5.3.2 Aktiviteter i matematikkopplæringen](#).

Hensikten med denne delen av analysen, er å vise hvordan man kan vurdere aktiviteter og se

på hvilke aspekter ved algoritmisk tenkning som kommer frem i aktivitetene. Delen av rammeverket for algoritmisk tenkning som består av tilnærminger, vil dermed ikke være relevant for denne delen av analysen. Som nevnt i [4.7.3.2 Algoritmisk tenkning i aktiviteter i matematikkopplæringen](#), er tilnærminger ekskludert av denne delen av analysen. Aspektene som blir nevnt under vil altså bare være kjerneferdigheter. Jeg har altså valgt å se bort fra tilnærminger når jeg skal analysere aspekter ved algoritmisk tenkning som kommer frem i aktivitetene.

Blant aktivitetene som er presentert i kapittel 5, er det som nevnt i 4.7.3.2, kun et utvalg av aktiviteter som vil analyseres. I utvelgelsen har jeg sett bort fra ressurser som er blitt nevnt av lærerne. Mattemestern.no, Kikora, og Lego league er ressurser som ble nevnt i intervjuene, uten at konkrete aktiviteter ble knyttet til dem. Lærer 1 nevner dog en aktivitet med Micro:bit, som er hentet fra Kidsakoder.no. Lærer 1 snakker litt om denne aktiviteten i intervjuet, og læreren har vist meg instruksjer for aktiviteten som er hentet fra [Kidsakoder.no](#).

De utvalgte aktivitetene er Scratch, Micro:bit, Kongen befaler og Rødt lys. Aktivitetene som skal analyseres vises i figur 6-1.



FIGUR 6-1: OVERSIKT OVER UTVALGTE AKTIVITETER SOM SKAL ANALYSERES

For å kunne undersøke aktivitetene i lys av rammeverket for algoritmisk tenkning, er det behov for informasjon om kontekst og betingelser i aktivitetene som er nevnt. Analysen stiller krav om nok informasjon om aktivitetene for å kunne undersøke hvilke aspekter ved algoritmisk tenkning som kommer frem i aktivitetene. Informasjonen om aktivitetene er hentet fra intervjuene og øvrige nevnte dokumenter i datamaterialet. I aktiviteter uten teknologi, er beskrivelser av aktivitetene hentet fra nasjonalt senter for mat, helse og fysisk aktivitet. Dette blir nærmere presisert i [6.2.2 Aktiviteter uten teknologi](#). Jeg skal nå gå videre til aktiviteter med teknologi.

6.2.1 Aktiviteter med teknologi

Det er to aktiviteter med teknologi som skal analyseres. Begge aktivitetene er hentet fra Lærer 1.

6.2.1.1 Scratch – Løkke

Som nevnt i [5.2.2](#), har Lærer 1 laget en video hvor læreren har som hensikt å vise elevene sine hvordan man kan lage en løkke i Scratch. Det er altså en slags introduksjonsvideo til hvordan man kan lage en løkke i Scratch. Jeg tar utgangspunkt i denne introduksjonen når jeg skal undersøke hvilke aspekter ved algoritmisk tenkning som kommer frem i aktiviteten.

I videoen legger Lærer 1 inn de ulike blokkodene som skal få dinosauren til å gå og snu. Det kan argumenteres for at *dekomponering* er noe som kommer til syne når man arbeider med de ulike kommandoene i blokkodingen, fordi man arbeider med en mindre del av helheten.

Likevel handler dekomponering om å bryte opp et problem for å forstå de ulike delene, og dermed ha en bedre forståelse for helheten. På bakgrunn av dette, kan man si at aktiviteten bør være av en problemløsende art for å kunne sies å inneha aspektet *dekomponering*. I denne aktiviteten ligger arbeidet med blokkodingen likevel nærmere aspektet *algoritmebehandling*, fordi man gjennom å legge inn blokkodene i riktig rekkefølge, og arbeide stegvis, vil man oppnå et resultat basert på rekkefølgen i koden, og aktiviteten er i liten grad problemløsende.

For å vise hvordan man skal legge inn koder, eksperimenterer Lærer 1 med antall skritt og hvor mange grader som skal ligge i koden. Begreper som dobling og grader dukker opp når læreren snakker i videoen, og det blir for eksempel spurt hvor mange skritt og grader det blir, når hendelsen blir satt til å skje ti ganger.

Læreren sier i videoen at elevene skal få figuren deres til å bevege seg. Dette gir elevene valgmuligheter for hvordan figuren deres skal bevege seg. Se tabell 6-2 for hvilke aspekter som kommer frem i denne aktiviteten.

TABELL 6-2: ASPEKTER VED ALGORITMISK TENKNING I SCRATCH

Kjerneferdigheter i algoritmisk tenkning	Forekomst i aktiviteten
Abstraksjon	
Algoritmebehandling	Man legger inn blokkoder for en hendelse ved å ta for seg en og en bevegelse som figuren skal utføre. Da legger man inn de aktuelle blokkodene i riktig rekkefølge. Her kreves det altså at man må gjøre de ulike stegene i riktig rekkefølge for å få figuren til å bevege seg på ønsket måte.
Generalisering	I denne aktiviteten vil det være relevant å se på sammenhengen mellom antall skritt og hvor mange grader figuren skal snu, og utnytte disse funksjonene til det beste formålet. Et annet moment som kan pekes på, er at elevene har arbeidet med Scratch tidligere. Det vil dermed ikke være helt ukjent for dem å arbeide med Scratch. Dersom de har gjort noe lignende tidligere, og de klarer å se sammenhengen med dette, blir dette noe som faller innunder generalisering.
Automatisering	Når elevene skal få figuren til å gå og snu, handler automatisering om å finne den mest effektive koden for dette.
Dekomponering	

6.2.1.2 Micro:bit – multiplikasjon

I intervjuet snakker Lærer 1 om en aktivitet med Micro:bit som læreren hadde funnet på Kidsakoder.no. Læreren forteller at elevene skal trykke inn svar på multiplikasjonsstykker på Micro:bit-brikken, og at de skal følge en liste som gjør feilsøkingen veldig anvendelig. På nettsiden til Kidsakoder vises aktiviteten med introduksjon, sjekklister og flere steg i aktiviteten (Clausen et al. u.å). Dette gjør at aspektet *algoritmebehandling* kommer frem i denne aktiviteten. Aktiviteten presenteres som et gangespill hvor elevene skal legge inn svar på multiplikasjonsstykker ved å trykke på A-knappen for antall tiere, og B-knappen for antall enere. Dersom elevene har svart riktig, vil de få et smilefjes, og dersom de har svart feil, vil de få et surt fjes, sier Lærer 1.

I denne aktiviteten arbeider elevene med å programmere en Micro:bit, slik at de kan arbeide med multiplikasjonsstykker. Selve kodingen på Micro:bit krever en forståelse av tiersystemet, da elevene skal skrive inn enere og tiere på riktig plass. Når elevene først har kodet inn riktig, kan elevene arbeide med multiplikasjonsstykkene. Instruksene i aktiviteten legger også opp til

at elevene må kjenne til begrepene faktor, produkt og variabler. Det er altså et gjensidig læringsforhold mellom matematikk og algoritmisk tenkning i denne aktiviteten.

På lik linje som med aktiviteten Scratch – Løkke, er det også en gråsonerom i Micro:bit, hvor det kan diskuteres om aspektet *dekomponering* er synlig eller ikke. Med samme prinsipp som i aktiviteten med Scratch, er denne aktiviteten i mindre grad av problemløsende art.

Arbeidsprosessen hvor elevene skal programmere Micro:biten, er noe som kan kobles til *algoritmebehandling* av samme årsak som i aktiviteten med Scratch. Elevene legger inn blokkodene i riktig rekkefølge for å få ønsket utfall. Et annet moment som kjennetegner *algoritmebehandling* i Micro:bit, er sjekklister som følger med i beskrivelsen til aktiviteten. I tabell 6-3 blir det presentert hvilke aspekter ved algoritmisk tenkning som kommer frem i denne aktiviteten.

TABELL 6-3: ASPEKTER VED ALGORITMISK TENKNING I MICRO:BIT

Kjerneferdigheter i algoritmisk tenkning	Forekomst i aktiviteten
Abstraksjon	
Algoritmebehandling	<p>På nettsiden til Kidsakoder skal elevene følge en sjekklister og flere steg gjennom arbeidsprosessen. Dette aspektet blir derfor synlig i denne aktiviteten.</p> <p>Elevene vil også programmere og kode Micro:biten for å oppnå et ønsket resultat. Det krever en forståelse av at de må</p>
Generalisering	<p>Det siste steget i aktiviteten steg 4, går ut på å lage lignende spill. Det stilles spørsmål om man klarer å lage et addisjonsspill, subtraksjonsspill, divisjonsspill og så videre. Det siste steget oppfordrer til å lage lignende problemer og løse dem.</p> <p>Når elevene skal kode før de skal løse multiplikasjonsstykkene, må de legge inn variabler i Micro:biten. Elevene må ha kjennskap til tiersystemet for å legge inn enere og tiere som variabler. Deretter skal de også kode produkt og faktorer slik at det kommer opp et multiplikasjonsstykke.</p>
Automatisering	Ved hjelp av å kode Micro:biten, har elevene nå fått Micro:biten til å gi elevene multiplikasjonsstykker som elevene skal svare på. I stedet for å lage stykkene selv, er det nå Micro:biten som lager dem. Derfor blir automatisering et aspekt som blir synlig i denne aktiviteten.
Dekomponering	

6.2.2 Aktiviteter uten teknologi

I dokumentet til Lærer 3, som presenterer ulike forslag til aktiviteter uten teknologi som kan knyttes til algoritmisk tenkning, er det ikke noe ytterligere informasjon om regler eller innhold i de ulike aktivitetene. I intervjuet med Lærer 3, sier vedkommende at aktivitetene Kongen befaler og Rødt lys handler om å tenke strategisk og planlegge. Nasjonalt senter for mat, helse og fysisk aktivitet, har på oppdrag fra Utdanningsdirektoratet ansvar for en idé- og ressursbank som inneholder en rekke aktiviteter som kan benyttes i skolen (Nasjonalt senter for mat, helse og fysisk aktivitet, u.å. b). I deres «aktivitetskasse» er Kongen befaler og Rødt lys beskrevet. Med utgangspunkt i deres beskrivelser av aktivitetene, kan jeg undersøke hvilke aspekter ved algoritmisk tenkning som elever får muligheten til å arbeide med i de nevnte aktivitetene.

6.2.2.1 Kongen befaler

Denne aktiviteten blir nevnt av Lærer 3, som sier det handler om å planlegge og tenke strategisk. Kongen befaler er en tradisjonell aktivitet som jeg kjenner til fra før av. Med utgangspunkt i Nasjonalt senter for mat, helse og fysisk aktivitet (u.å. a) sin «aktivitetskasse» sin beskrivelse av hvordan leken foregår, kan jeg undersøke hvilke aspekter ved algoritmisk tenkning som kan komme frem i Kongen befaler. Det må da understrekes at jeg ikke tar hensyn til eventuelle tilpasninger av aktiviteten. Det betyr ikke nødvendigvis at det ikke er mulig å avansere aktivitetene med et formål om å tilpasse aktivitetene etter behov og ønske.

I Kongen befaler skal en person velges ut til å ha rollen som konge. Kongen skal gi kommandoer og fortelle hva de andre skal gjøre. De andre deltakerne skal gjøre som kongen befaler, men bare når kongen starter kommandoen med «kongen befaler». Den som gjør feil på en kommando, er ute av leken. Sistemann som står igjen, er konge i neste runde. Se tabell 6-4 for en oversikt over hvilke aspekter som kan komme frem i Kongen befaler.

TABELL 6-4: OVERSIKT OVER ASPEKTER VED ALGORITMISK TENKNING I KONGEN BEFALER

Kjerneferdigheter i algoritmisk tenkning	Forekomst i aktiviteten
Abstraksjon	«Kongen» er avhengig av å se hvem av de andre deltakerne som klarer å gjøre som kongen sier. Det kan innebære å klare å se bort fra unødvendige informasjon fra de ulike deltakerne. Da vil «kongen» se på om de har løst oppgaven, og se bort fra mindre viktige detaljer.
Algoritmebehandling	I denne aktiviteten er det også tydelige regler på hvordan man kan vinne, og disse må følges for å kunne vinne. «Kongen» gir en og en kommando, og de andre deltakerne må klare å følge kommandoene for å ikke ryke ut av leken. Den siste som står igjen, har klart oppgaven.
Generalisering	
Automatisering	I vurderingen på hvordan de andre deltakerne har løst oppgaven som «kongen» har gitt dem, vil man ta stilling til om de har løst oppgaven på en tungvint eller effektiv måte.
Dekomponering	

6.2.2.2 Rødt lys

Rødt lys er en tradisjonell lek som er en av aktivitetene som står som forslag til aktivitet i dokumentet fra Lærer 3, og blir nevnt i intervjuet. I dokumentet blir det ikke beskrevet noen kontekster eller betingelser i leken. I intervjuet sier Lærer 3 at det handler om å tenke strategisk og at man må planlegge, og at det er noe som er «unplugged» programmering.

Ifølge «Aktivitetskassen» til Nasjonalt senter for mat, helse og fysisk aktivitet (u.å. c), foregår leken slik at alle bortsett fra én stiller seg opp på en linje. På motsatt side av den linjen, stiller den ene personen seg opp og fungerer som et rødt lys. Det røde lyset skal snu seg med ryggen mot de andre og telle «1-2-3 rødt lys», og da snu seg mot de andre igjen for å se om noen beveger på seg etter beskjeden. De andre skal prøve å komme så nærme eleven som sier rødt lys som mulig, uten at eleven ser noe bevegelse når han eller hun snur seg. Elever som er i bevegelse når rødt lys roper «rødt lys» og snur seg, må returnere til startlinjen. Den eleven som kommer først frem til rødt lys, overtar «rødt lys»-rollen i neste runde. Under i tabell 6-5, vises en oversikt over aspekter ved algoritmisk tenkning i Rødt lys.

TABELL 6-5: OVERSIKT OVER ASPEKTER VED ALGORITMISK TENKNING I RØDT LYS

Kjerneferdigheter i algoritmisk tenkning	Forekomst i aktiviteten
Abstraksjon	<p>Personen som er rødt lys må se etter bevegelse hos de andre deltakerne, og passe på å ikke bli distraheret av eventuelle distraheringer.</p> <p>En prosess med å gjøre noe mer forståelig gjennom å ta bort unødvendige og overflødige detaljer. Essensen er å fremheve det viktige og fjerne det mindre viktige for å komme frem til en løsning.</p>
Algoritmebehandling	<p>Personene som skal nærme seg rødt lys, må følge med på beskjed fra rødt lys om tid, for å kunne bevege seg nærmere og nærmere rødt lys.</p> <p>I denne aktiviteten er det også tydelige regler på hvordan man kan vinne, og disse må følges for å kunne vinne.</p>
Generalisering	
Automatisering	Denne kjerneferdigheten kan handle om å vurdere og finne raskest mulig vei til rødt lys.
Dekomponering	

6.2.3 Forekomst av aspekter ved algoritmisk tenkning i aktiviteter i matematikkopplæringen

Når vi ser på aktivitetene i et komparativt lys, blir det synlig at det er ulike aspekter som kommer frem i de ulike aktivitetene. I aktivitetene med teknologi er det *algoritmebehandling*, *generalisering* og *automatisering* som kommer frem. I aktivitetene uten teknologi blir kjerneferdighetene *abstraksjon*, *algoritmebehandling* og *automatisering* synlig. I tabell 6-6 har jeg plassert aktivitetene ved siden av hverandre, slik at det kommer frem hvilke aspekter ved algoritmisk tenkning som er til stede og fraværende i de ulike aktivitetene. Aspektet *abstraksjon* er fraværende i aktivitetene med teknologi, og *generalisering* viser seg ikke i aktivitetene uten teknologi. Felles for alle aktivitetene er at aspektet *dekomponering* er fraværende.

TABELL 6-6: OVERSIKT OVER FOREKOMST AV ASPEKTER VED ALGORITMISK TENKNING I AKTIVITETER I MATEMATIKKOPPLÆRINGEN

Kjerneferdigheter i algoritmisk tenkning	Aktiviteter i matematikkopplæringen med teknologi		Aktiviteter i matematikkopplæringen uten teknologi	
	Scratch	Micro:bit	Kongen befaler	Rødt lys
Abstraksjon			⊗	⊗
Algoritmebehandling	⊗	⊗	⊗	⊗
Generalisering	⊗	⊗		
Automatisering	⊗	⊗	⊗	⊗
Dekomponering				

7 Diskusjon

Diskusjonskapittelet har som formål å diskutere studiens forskningsspørsmål i lys av teori og funn fra analysen. Her vil jeg svare på studiens forskningsspørsmål ved å presentere sentrale funn fra analysen, og se disse i sammenheng med teori og tidligere forskning.

7.1 Bakgrunnen for implementering av algoritmisk tenkning i matematikkopplæringen

I styringsdokumentene som er nevnt i denne studien, kommer det frem at det er behov for en mer generell forståelse av digitale teknologier, og at det ikke er tilstrekkelig bare å bruke teknologi for å forbedre elevenes ytelse (NOU 2020:2, s. 43; OECD, 2019, s. 179). Dette understøtter en oppfatning om at algoritmisk tenkning ser ut til å ta en rolle i hvordan teknologi skal integreres i undervisningen, med formål om å gi økt forståelse og kunnskap om teknologi. Intensjonen bak algoritmisk tenkning ser altså ut til å henge sammen med krav om utvikling av ferdigheter og kompetanser som står i stil med teknologiens krav og rolle i samfunnet vårt.

Slike ferdigheter og kompetanser omtales av OECD som økt generell forståelse om hva som skjer bak kulissene i teknologien i samfunnet vårt. I NOU 2020:2 (s. 43) hevdes det at man ved algoritmisk tenkning har mulighet til å forstå mer av konseptene bak mye av informasjonsteknologien. Ifølge NOU-rapporten, kan algoritmisk tenkning se ut til å styrke evnen til matematisk, kreativ og kritisk tenkning, og at det er nærliggende problemløsning. Man kan si at ved å implementere algoritmisk tenkning i matematikkopplæringen, vil man kunne legge til rette for å utvikle nye ferdigheter tilpasset fremtidens behov. Dette er selve kjernen i årsaken til hvorfor algoritmisk tenkning er implementert i matematikkopplæringen. Algoritmisk tenkning kan altså være en del av løsningen på kravet om utvikling av nye ferdigheter for fremtiden.

Bakgrunnen for at algoritmisk tenkning er implementert i matematikkopplæringen viser seg å være tosidig. Den ene siden viser til koblingen mellom algoritmisk tenkning og krav om fremtidige ferdigheter, hvor styringsdokumentene vektlegger den generelle forståelsen for teknologiens rolle i samfunnet. Den andre siden kobles til forskere som hevder algoritmisk tenkning bør fremmes som en fundamental ferdighet som bør realiseres i en didaktisk praksis (Wing, 2008, s. 33; Weintrop m. fl., 2016, s. 128). Fra et pedagogisk ståsted kan man peke på

at et gjensidig læringsforhold mellom matematikk og algoritmisk tenkning er en klar fordel for utvikling av ferdigheter som fremtiden krever. Dette er noe jeg ser igjen i funnene i denne studien, hvor aktiviteten med Micro:bit og multiplikasjon legger til rette for et gjensidig læringsforhold mellom algoritmisk tenkning og matematikk.

Til tross for at intensjonen bak algoritmisk tenkning er avklart, er det uansett behov for redskaper til å identifisere algoritmisk tenkning i didaktiske praksiser. Studien til Kong, Lai og Sun (2020) som er nevnt i [1.3 Tidligere forskning](#), viser også at lærere trenger kunnskap om algoritmisk tenkning for å innføre dette i matematikkopplæringen.

Ved hjelp av Bocconi et al. (2016;2018) sitt rammeverk for algoritmisk tenkning som er nevnt i OECD-rapporten fra 2019, er det mulig å identifisere algoritmisk tenkning i en undervisningskontekst. I denne studien har jeg benyttet Bocconi et al. (2016;2018) sitt rammeverk for algoritmisk tenkning, og identifisert hvilke aspekter ved algoritmisk tenkning som kommer frem i tre læreres forståelse og oppfatning av algoritmisk tenkning. Jeg har også undersøkt hvilke aspekter ved algoritmisk tenkning som kommer frem i fire utvalgte aktiviteter som nevnes av de tre lærerne. Bakgrunnen for hvorfor algoritmisk tenkning er implementert i matematikkopplæringen vil, sammen med resultatene fra undersøkelsen i denne studien, gi innsikt i hvilke kompetanser algoritmisk tenkning har som intensjon i å fremme i matematikkopplæringen.

7.2 Aspekter ved algoritmisk tenkning i lærernes forståelse og oppfatning av algoritmisk tenkning

Fra undersøkelsen om aspekter ved algoritmisk tenkning i lærernes forståelse og oppfatning av algoritmisk tenkning, er kjerneferdigheten *algoritmebehandling* det eneste aspektet ved algoritmisk tenkning som kommer frem hos alle de tre lærerne. Kjerneferdigheten *algoritmebehandling* handler om å komme frem til en løsning ved å arbeide steg for steg, eller følge en oppskrift eller en algoritme. Innholdet i begrepet er altså noe som også kommer frem i begrepet algoritmisk tenkning. Dersom man vet hva en algoritme er, kan man anta at en del av algoritmisk tenkning handler om å følge en algoritme.

Med formål om å finne en forklaring på hvorfor *algoritmebehandling* er et fremtredende aspekt ved algoritmisk tenkning hos de tre lærerne, kan vi se på Utdanningsdirektoratets kompetansepakker i forbindelse med innføringen av den nye læreplanen.

Algoritmebehandling kommer tydelig frem i Utdanningsdirektoratets beskrivelse av algoritmisk tenkning. Det er dermed ikke overraskende at dette aspektet ved algoritmisk tenkning er til stede hos alle de tre lærerne. I intervjuet trakk også Lærer 3 frem figuren fra Utdanningsdirektoratets artikkel om algoritmisk tenkning. Det kan dermed tenkes at Lærer 3 sin forståelse og oppfatning av algoritmisk tenkning er preget av artikkelen. Det kan heller ikke utelukkes at også Lærer 1 og Lærer 2 har lest artikkelen.

Tilnærmingen *kreativitet* er den eneste tilnærmingen som er fraværende hos alle de tre lærerne. I figuren om algoritmisk tenking fra Utdanningsdirektoratet, brukes ikke begrepet kreativitet. Det blir heller ikke nevnt i artikkelen. I stedet for brukes begreper som skapende, eksperimenterende og åpen for alternative løsninger. Det kan derfor trolig sies at Utdanningsdirektoratets kompetansepakker er tiltenkt å forberede lærerne på den nye læreplanen, herunder algoritmisk tenkning i matematikkopplæringen. Dette understreker hvor viktig det er å få frem alle aspekter ved algoritmisk tenkning i didaktiske praksiser for å oppfylle intensjonen bak algoritmisk tenkning i matematikkopplæringen.

Lærer 1 er den eneste læreren av de tre som nevner kjerneferdigheten *dekomponering* og tilnærmingen *feilsøking*. Feilsøking og dekomposisjon er begreper som Utdanningsdirektoratet (2019c) nevner i sin artikkel om algoritmisk tenkning. I intervjuet med Lærer 1 kommer det frem at Lærer 1 på eget initiativ har prøvd seg på javascript og lest seg opp på programmering. Det kan altså godt tenkes at Lærer 1 kobler feilsøking til programmering.

Undersøkelsen viser at Lærer 2 og 3 ser på algoritmisk tenkning som en problemløsningsmetode hvor man tenker systematisk og strategisk. Med denne beskrivelsen kommer både *abstraksjon* og *automatisering* frem. Denne gråsonen illustrerer at det er viktig å også forstå intensjonen som ligger til grunn for algoritmisk tenkning i matematikkopplæringen, og ikke bare hvilke aspekter som kommer frem. Lærer 2 og Lærer 3s forståelse av disse aspektene ved algoritmisk tenking kan ses i sammenheng med hva kjerneelementet Utforskning og problemløsning i den nye læreplanen i matematikk (Utdanningsdirektoratet, 2019b) sier om algoritmisk tenkning.

I kjerneelementet Utforskning og problemløsning beskrives algoritmisk tenkning som en viktig del i prosessen med å utvikle strategier og fremgangsmåter for å løse problemer. Videre skriver Utdanningsdirektoratet (2019b) at dette innebærer å bryte ned et problem i delproblemer som kan løses systematisk på best mulig måte med eller uten digitale verktøy.

Dette er imidlertid noe Lærer 2 og 3 ikke viser i intervjuene. De nevner ikke kjerneferdigheten *dekomponering*. Dette kan tyde på at Lærer 2 og Lærer 3 viser til en forståelse og oppfatning av algoritmisk tenkning som i større grad vektlegger utvikling av ferdigheter enn problemløsning. Dette er sammenfallende med at deres forståelse og oppfatning av algoritmisk tenkning ikke innehar tilnærmingen *kreativitet*.

7.3 Aspekter ved algoritmisk tenkning i aktiviteter i matematikkopplæringen

Fra undersøkelsen om aspekter ved algoritmisk tenkning som kan komme frem i aktiviteter i matematikkopplæringen, er det kjerneferdighetene *algoritmebehandling*, *generalisering* og *automatisering* som viser seg i aktiviteter med teknologi, og kjerneferdighetene *abstraksjon*, *algoritmebehandling* og *automatisering* som er fremtredende i aktiviteter uten teknologi. Det kommer også frem langt flere forslag til aktiviteter og bruk av ressurser knyttet til teknologi, sammenlignet med aktiviteter uten teknologi i intervjuene med de tre lærerne. Man kan si at det nevnes forholdsvis få aktiviteter uten teknologi i intervjuene. Lærer 3 er den eneste læreren som eksplisitt nevner aktiviteter uten teknologi som kan knyttes til algoritmisk tenkning. I styringsdokumentene og forskning som er presentert, kobles ofte fokus på algoritmisk tenkning til teknologi som et grunnlag for en forståelse for teknologiens funksjon, og ferdigheter knyttet til den fjerde industrielle revolusjonen. Det kan tyde på at dette fokuset er noe som gjenspeiles i disse funnene hvor vi ser færre eksempler på aktiviteter uten teknologi enn aktiviteter med teknologi.

Det er likevel klart at algoritmisk tenkning ikke er avhengig av en teknologisk kontekst, og aspektene som kommer frem i aktivitetene uten teknologi, indikerer at disse får frem noe aktivitetene med teknologi ikke får frem. I aktivitetene uten teknologi ser vi aspektet *abstraksjon*, noe som er fraværende i aktivitetene med teknologi. I aktivitetene med teknologi ser vi aspektet *generalisering*, som ikke kommer frem i aktivitetene uten teknologi. Dette viser at begge dimensjonene spiller en sentral rolle for hvordan aspektene ved algoritmisk tenkning fremtrer i matematikkopplæringen.

7.4 Sammenheng mellom aktiviteter i matematikkopplæringen og lærernes forståelse og oppfatning av algoritmisk tenking

Når vi ser på aktivitetene som er analysert opp mot lærernes forståelse og oppfatning av algoritmisk tenkning, blir det synlig at *dekomponering* er noe som kommer frem i Lærer 1 sin forståelse og oppfatning, men ikke i noen av aktivitetene. I Lærer 1s aktiviteter med teknologier er *generalisering* til stede, men også fraværende i lærerens forståelse og oppfatning av algoritmisk tenkning.

Dekomponering er ikke til stede i verken Lærer 2 eller Lærer 3 sin forståelse og oppfatning av algoritmisk tenkning, og det kommer heller ikke frem i aktivitetene uten teknologi.

Abstraksjon er noe som kommer til syne i Lærer 2 og Lærer 3 sin forståelse og oppfatning av algoritmisk tenkning. Dette samsvarer med tilstedeværelsen av abstraksjon i aktivitetene uten teknologi.

Det er en tydelig forbindelse mellom lærernes oppfatning og forståelse av algoritmisk tenkning, og aktivitetene som er analysert i kapittel 6. Forbindelsen peker på at det er overensstemmelser i aspekter ved algoritmisk tenkning som kommer frem i aktivitetene, og i hva lærerne sier om algoritmisk tenkning. Dette er ikke overraskende resultater. Den sterke tilstedeværelsen av *algoritmebehandling* passer med lærernes oppfatning av algoritmisk tenkning som feilsøkende og utholdende. På den måten kan man også si at fraværet av *dekomponering* i både aktiviteter og lærernes oppfatning og forståelse av algoritmisk tenkning, kan ses i sammenheng med fraværet av *kreativitet*. Disse resultatene gjør det mulig å se på dette som en indikasjon på at lærerne har et fokus der utvikling av ferdigheter vektlegges i større grad enn problemløsning. Med et utgangspunkt i dette kan vi se to sider av algoritmisk tenkning hvorav en side viser algoritmisk tenkning med problemløsning, og den andre siden legger vekt på utvikling av ferdigheter. Disse dimensjonene utelukker ikke hverandre, men viser heller hva som blir vektlagt.

Se tabell 7-1 for en samlet oversikt over forekomst av aspekter ved algoritmisk tenkning i aktiviteter og lærernes forståelse og oppfatning av algoritmisk tenkning. Tilnærminger ble ikke inkludert i analysen av aktiviteter. Jeg har farget de rutene i tabellen grå for å vise dette.

Algoritmebehandling er til stede, mens *kreativitet* er fraværende i alle aktivitetene og i alle de tre lærernes forståelse og oppfatning av algoritmisk tenkning. Denne forekomsten av aspekter ved algoritmisk tenkning i aktiviteter og lærernes forståelse og oppfatning av algoritmisk

tenkning, viser at lærerne i større grad fokuserer på algoritmisk tenkning med utvikling av ferdigheter.

TABELL 7-1: OVERSIKT OVER FOREKOMST AV ASPEKTER VED ALGORITMISK TENKNING I AKTIVITETER OG LÆRERNES FORSTÅELSE OG OPPFATNING AV ALGORITMISK TENKNING

	Aktiviteter med teknologi	Aktiviteter uten teknologi	Lærer 1	Lærer 2	Lærer 3
Abstraksjon		⊗		⊗	⊗
Algoritmebehandling	⊗	⊗	⊗	⊗	⊗
Generalisering	⊗			⊗	⊗
Automatisering	⊗	⊗		⊗	⊗
Dekomponering			⊗		
Feilsøking			⊗		
Kreativitet					
Samarbeid					⊗
Utholdenhet					⊗

Vi ser altså at det er en sammenheng mellom forekomsten av aspekter ved algoritmisk tenkning i aktiviteter i matematikkopplæringen, og hvilke aspekter som kommer frem i lærernes forståelse og oppfatning av algoritmisk tenkning. Lærer 3 sin oppfatning og forståelse av tilnærmingene *samarbeid* og *utholdenhet* passer med tilstedeværelsen av kjerneferdigheten *algoritmebehandling*. Fraværet av tilnærmingen *kreativitet* passer med fraværet av kjerneferdigheten *dekomponering*.

Intensjonen bak algoritmisk tenkning sammenholdt med sammenhengen mellom forekomst av aspekter ved algoritmisk tenkning i matematikkopplæringen, og hvilke aspekter som kommer frem i lærernes forståelse og oppfatning av algoritmisk tenkning, illustrerer potensialet til algoritmisk tenkning i matematikkopplæringen. Dette danner et bilde av algoritmisk tenkning sin rolle i matematikkopplæringen.

8 Avslutning

8.1 Konklusjon

I denne studien har jeg undersøkt bakgrunnen for implementeringen av algoritmisk tenkning i matematikkopplæringen, og benyttet et rammeverk for algoritmisk tenkning for å belyse hvilke aspekter ved algoritmisk tenkning som kommer frem i tre læreres forståelse og oppfatning av algoritmisk tenkning. Tilsvarende rammeverk er brukt for å undersøke hvilke aspekter som kan komme frem i utvalgte aktiviteter som de tre lærerne nevner. Basert på dette kan det trekkes noen konklusjoner som vil svare på studiens overordnede formål om å undersøke hvilke kompetanser algoritmisk tenkning har som intensjon å fremme i matematikkopplæringen.

I prosessen med å svare på studiens første forskningsspørsmål har nevnte styringsdokumenter spilt en viktig rolle. I styringsdokumentene blir det klart hva som er bakgrunnen for implementeringen av algoritmisk tenkning i matematikkopplæringen. Konklusjonen til NOU- og OECD-rapportene sier det er mye som tyder på at algoritmisk tenkning ser ut til å styrke evnen til matematisk, kreativ og kritisk tenkning, samt at det er nærliggende til problemløsning. Dette er ferdigheter og kompetanser som kan knyttes til den fjerde industrielle revolusjonen, og som er tilsiktet gjennom hva Kunnskapsdepartementet ønsker å få frem i den nye læreplanen. OECD-rapporten viser til forskning som sier at elever vil få bedre ferdigheter innenfor problemløsning dersom de blir eksponert for algoritmisk tenkning gjennom programmering (Lye & Koh, 2014, i OECD, 2019, s. 190). At algoritmisk tenkning ser ut til å styrke matematisk tenkning, er noe som kan ses i sammenheng med det forskere peker på (Weintrop et al., 2016, s.128), nemlig at det er et gjensidig læringsforhold mellom matematikk og algoritmisk tenkning. Dette tyder på at det er en overensstemmelse blant forskere og styringsdokumenter om hvorfor algoritmisk tenkning har fått en plass i matematikkopplæringen. Algoritmisk tenkning er blitt implementert i matematikkopplæringen fordi det styrker nødvendige ferdigheter som møter et krav om en økt generell forståelse av teknologiens rolle og funksjon i samfunnet.

Studiens andre forskningsspørsmål søker etter hvilke aspekter som kommer frem i tre læreres forståelse og oppfatning av algoritmisk tenkning. Funn i denne studien viser at algoritmisk tenkning har enda en dimensjon som deler algoritmisk tenkning inn i algoritmisk tenkning med og uten problemløsning. Studiens undersøkelse, analyse og diskusjon, tilsier at lærerne

har en forståelse og oppfatning av algoritmisk tenkning som ser ut til å fokusere på en dimensjon av algoritmisk tenkning som vektlegger problemløsning i en mindre grad.

Aspektet *algoritmebehandling* er den eneste kjerneferdigheten som kommer frem i alle de tre lærernes forståelse og oppfatning av algoritmisk tenkning, mens aspektet *kreativitet* er den eneste tilnærmingen som er fraværende hos dem. Lærer 3 er alene med å vise en forståelse og oppfatning av at algoritmisk tenkning har dimensjoner med og uten teknologi. De to andre lærerne betrakter kun dimensjonen med teknologi.

Studiens tredje og siste forskningsspørsmålet stiller spørsmål til hvilke kjerneferdigheter i algoritmisk tenkning som kan komme frem i nevnte aktiviteter i matematikkopplæringen. I intervjuene nevner alle de tre lærerne færre eksempler på aktiviteter og ressurser uten teknologi, enn med teknologi. Aktivitetene som er trukket frem i analysen, er aktiviteter som er blitt beskrevet av lærerne. Studiens undersøkelse viser at kjerneferdigheten *dekomponering* er fraværende i alle aktivitetene, samtidig som kjerneferdigheten *algoritmebehandling* er til stede i alle aktivitetene. I aktivitetene uten teknologi kommer kjerneferdigheten *abstraksjon* frem, men ikke i aktiviteter med teknologi. Kjerneferdigheten *generalisering* viser seg i aktiviteter med teknologi, men ikke i aktiviteter uten teknologi.

Vi vet nå at kompetanser som styringsdokumentene sikter til gjennom den nye læreplanen inneholder krav om en økt generell forståelse av teknologi. Videre argumenteres det for at algoritmisk tenkning ser ut til å fremme en matematisk, kreativ og kritisk tenkning, samt ferdigheter innenfor problemløsning. Det kan da tenkes at dette er kompetanser og ferdigheter som algoritmisk tenkning har intensjon om å fremme i matematikkopplæringen.

Sett opp mot resultatene fra forskningsspørsmålene 2 og 3 i denne studien, er likevel dette forholdet i mindre grad samsvarende. Resultatene viser at lærerne har en forståelse og oppfatning av algoritmisk tenkning som i mindre grad vektlegger problemløsning. Dette er også noe som gjenspeiles i hvilke aspekter ved algoritmisk tenkning som viser seg i aktivitetene.

Bakgrunnen for hvorfor algoritmisk tenkning er implementert i matematikkopplæringen sett sammen med resultatene fra disse undersøkelsene, gir innsikt i hvilke kompetanser algoritmisk tenkning har som intensjon å fremme i matematikkopplæringen. Mye tyder på at intensjonen bak algoritmisk tenkning er todelt. Det virker som det er økt fokus på problemløsning samtidig som det er et ønske om å fremme bruk av teknologi som utdanner

skapere fremfor forbrukere. Likevel er det viktig å merke seg at man kan ha en intensjon om å fremme noe, samtidig som det faktiske utfallet kan være noe annet enn intensjonen.

8.2 Videre forskning

I løpet av arbeidsprosessen i denne studien har det dukket opp en rekke nye problemstillinger som er verdt å undersøke nærmere. Som det er nevnt innledningsvis i denne studien, er det behov for ytterligere forskning som undersøker rammefaktorer og forhold knyttet til algoritmisk tenkning i matematikkopplæringen (Forsström & Kaufmann, 2018, s. 30; Grover & Pea, 2013, s. 42; Weintrop et al., 2016, s. 143).

I videre forskning kan det være aktuelt å undersøke lokale læreplaner som er utarbeidet på skolenivå, og se hvordan algoritmisk tenkning kommer frem i de dokumentene. Dokumentet som Lærer 3 har delt med denne studien, har gitt informasjon om hvordan Lærer 3s skole har planlagt hvordan de ønsker å få inn programmering og algoritmisk tenkning i matematikkopplæringen deres. Dersom det er mulig å gjennomføre en større undersøkelse hvor man ser på lokale læreplaner, kan det gi innsikt i hvordan skoler planlegger og legger til rette for algoritmisk tenkning i deres matematikkopplæring.

Parallelt med arbeid knyttet til læreres forståelse og vurdering av algoritmisk tenkning i matematikkopplæringen, bør det også undersøkes elevers tilnærminger til algoritmisk tenkning. I en forlengelse av denne studien kan det da være interessant å undersøke nærmere tilnærminger og dimensjonene i algoritmisk tenkning.

Denne studien har avdekket at det eksisterer mulige dimensjoner av algoritmisk tenkning som vektlegger problemløsning i mindre og større grad. Dette er noe som kan være verdifullt å undersøke videre. Det kommer også frem i denne studien at dimensjonene med og uten teknologi virker til å ha en viktig funksjon for hvordan de ulike aspektene ved algoritmisk tenkning kommer frem i matematikkopplæringen. Å se nærmere på denne funksjonen, vil være relevant for algoritmisk tenkning i matematikkopplæringen.

I denne studien ble ikke tilnærminger i algoritmisk tenkning undersøkt i aktiviteter i matematikkopplæringen. Jeg mener det vil være av stor betydning å undersøke elevers tilnærminger i aktiviteter i matematikkopplæringen i en eventuell videre forskning.

Eksempelvis kan man gjennom en triangulert metode hvor det benyttes både observasjon og intervju, og dermed få et mer sammensatt bilde av hvordan algoritmisk tenkning kommer

frem i aktiviteter i matematikkopplæringen. Det er hevet over enhver tvil at vi går inn en spennende tid. Det er behov for en aktiv diskusjon og empirisk forskning knyttet til hvilken rolle algoritmisk tenkning skal ha i matematikkopplæringen.

8.3 Avsluttende kommentar

Bakgrunnen for denne studien bunner i endringer i samfunnet vårt. Den fjerde industrielle revolusjonen krever nye ferdigheter og kompetanser, og det er behov for mer forskning knyttet til dette. Denne masteroppgaven har undersøkt intensjonen bak algoritmisk tenkning i matematikkopplæringen, oppfatning og forståelse av algoritmisk tenkning hos tre lærere, samt hvordan algoritmisk kommer frem i fire aktiviteter i matematikkopplæringen. For at lærere skal vite hvilke kompetanser og ferdigheter de skal fremme med algoritmisk tenkning i sin matematikkopplæring, er det avgjørende at intensjonen bak algoritmisk tenkning er tydelig og presis.

Styringsdokumenter, internasjonal forskning og den nye læreplanen har spilt en viktig rolle i undersøkelsen av algoritmisk tenkning sin rolle i matematikkopplæringen. Jeg håper denne masteroppgaven er opplysende i den forstand at det som kommer frem om algoritmisk tenkning, er troverdig. Algoritmisk tenkning er bare en liten del av den endringen vi nå befinner oss i, men det er absolutt en viktig del av helheten.

9 Litteraturliste

- Aho, A. (2012). Computation and computational thinking. *The Computer Journal*, 55, 7, 832–835. <https://doi.org/10.1093/comjnl/bxs074>
- Balanskat, A., Engelhardt, K., Ferrari, A. (2017). The integration of Computational Thinking (CT) across school curricula in Europe. *European Schoolnet Perspective*, 2, 65-72. <https://doi.org/10.1145/3137065.3137069>
- Barefoot Computing. (u.å. a) *Computational thinking concepts and approaches*. Barefoot computing. <https://www.barefootcomputing.org/concept-approaches/computational-thinking-concepts-and-approaches?CatItemId=computational-thinking-concepts-and-approaches>
- Barefoot Computing. (u.å. b) *About barefoot computing*. Barefoot computing. <https://www.barefootcomputing.org/about-barefoot>
- Bocconi, S., Chiocciariello, A., Dettori, G., Ferrari, A., Engelhardt, K. (2016). *Developing computational thinking in compulsory education – Implications for policy and practice*. doi:10.2791/792158
- Bocconi, S., Chiocciariello, A., & Earp, J. (2018). *The Nordic approach to introducing computational thinking and programming in compulsory education*. doi:10.17471/54007
- Brackmann, C. P, Román-González, M., Robles, G., Moreno-León, J., Casali, A., Barone, D. (2017). Development of Computational Thinking Skills through Unplugged Activities in Primary School. *Proceedings of the 12th Workshop on Primary and Secondary Computing Education (WiPSCE '17)*. 65–72. <https://doi.org/10.1145/3137065.3137069>
- Christoffersen, L., & Johannessen, A. (2012). *Forskningsmetode for lærerutdanningene* (2. utgave). Oslo: Abstrakt forlag.
- Clausen, E., Silde, T., Revdahl, J. (u.å). *PXT: Gangespill*. Kidsakoder. https://oppgaver.kidsakoder.no/microbit/pxt_gangespill/gangespill
- Forskningsetikkloven. (2017). *Lov om organisering av forskningsetisk arbeid* (LOV-2017-04-28-23). Lovdata. <https://lovdata.no/dokument/NL/lov/2017-04-28-23>
- Forsström, S. E, Kaufmann O. T. (2018). A Literature Review Exploring the use of Programming in Mathematics Education in *International Journal of Learning, Teaching and Educational Research*, 17 (12), 18-32. <https://doi.org/10.26803/ijlter.17.12.2>
- Gjøvik, Ø., Torkildsen, H. A. (2019). Algoritmisk tekning. *Tangenten – tidsskrift for matematikkundervisning*, 30(3). 31–37. <http://www.caspar.no/tangenten/2019/Tangenten%203%202019%20Gj%C3%B8vik%20Torkildsen.pdf>
- Grover, S. & Pea, R. (2013). Computational thinking in K-12: A review of the state of the field. *Educational Researcher*, 42(1), 38–43.
- Holm, I. & Solvang, B. (1996). *Metodevalg og metodebruk* (3. utgave). Oslo: Tano
- Høgskulen på Vestlandet. (2021, 07. januar). LATAcME. Hvl. <https://prosjekt.hvl.no/latacme/>
- Imenes, O. G, Bjarnø, V., Hatlevik, O. E. (2021). *Lærerstudenters bruk av Monte Carlo-simulering for å løse oppgaver om sannsynlighetsregning: En analyse av 16 gruppebesvarelser*. Upublisert.
- Johannessen, A., Tuft, P. A., Christoffersen, L. (2016). *Introduksjon til samfunnsvitenskapelig metode* (5. utgave) Abstrakt forlag

- Kunnskapsdepartementet. (2017). *Rammeverk for grunnleggende ferdigheter*. Udir. <https://www.udir.no/laring-og-trivsel/rammeverk/rammeverk-for-grunnleggende-ferdigheter/>
- Kvale, S. (2007) *Doing interviews* (2. utgave) London: SAGE
- Kvale, S. & Brinkmann, S. (2015). *Det kvalitative forskningsintervju*. (3. utgave) Gyldendal Akademisk
- Lye, S. Y., Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12? *Computers in Human Behavior*, (41) 51-61. <https://doi.org/10.1016/j.chb.2014.09.012>
- Nasjonale forskningsetiske komité for samfunnsvitenskap og humaniora (NESH). (2016). *Forskningsetiske retningslinjer for samfunnsvitenskap, humaniora, juss og teologi*. (4. utgave) Oktan Oslo AS
- Nasjonalt senter for mat, helse og fysisk aktivitet. (u.å. a). *Kongen befaler*. Aktivitetskassen. <https://aktivitetskassen.no/kongen-befaler/>
- Nasjonalt senter for mat, helse og fysisk aktivitet. (u.å. b). *Om aktivitetskassen*. Aktivitetskassen. <https://aktivitetskassen.no/om/>
- Nasjonalt senter for mat, helse og fysisk aktivitet. (u.å. c). *Rødt lys*. Aktivitetskassen. <https://aktivitetskassen.no/rodt-lys/>
- NOU: 2013: 2. (2013). *Hindre for digital verdiskaping*. Fornyings-, administrasjons- og kirke departementet. <https://www.regjeringen.no/contentassets/e2f0d5676e144305967f21011b715c16/no/pdfs/nou201320130002000dddpdfs.pdf>
- NOU: 2015: 8. (2015). *Fremtidens skole – fornyelse av fag og kompetanser*. Kunnskapsdepartementet. <https://www.regjeringen.no/contentassets/da148fec8c4a4ab88daa8b677a700292/no/pdfs/nou201520150008000dddpdfs.pdf>
- NOU: 2020: 2. (2020). *Fremtidige kompetansebehov III*. Kunnskapsdepartementet. <https://www.regjeringen.no/contentassets/053481d65fb845be9a2b1674c35d6d14/no/pdfs/nou202020200002000dddpdfs.pdf>
- OECD. (2019). *OECD Skills Outlook 2019: Thriving in a Digital World*. OECD Publishing, Paris. <https://doi.org/10.1787/df80bc12-en>
- Personopplysningsloven. (2018). *Lov om behandling av personopplysninger* (LOV-2018-06-15- 38). Lovdata. <https://lovdata.no/dokument/NL/lov/2018-06-15-38>
- Postholm, M. & Jacobsen, D. (2018). *Forskningsmetode for masterstudenter i lærerutdanningen*. Oslo: Cappelen Damm AS
- Universitetet i Bergen. (2014). *Anvendt og beregningsorientert matematikk*. Uib. <https://www.uib.no/math/52405/anvendt-og-beregningsorientert-matematikk>
- Utdanningsdirektoratet. (2019a). *Læreplan i matematikk 1.-10. trinn*. (MAT01-05). <https://www.udir.no/lk20/mat01-05>
- Utdanningsdirektoratet. (2019b). *Kjerneelementer*. Udir. <https://www.udir.no/lk20/mat01-05/om-faget/kjerneelementer>
- Utdanningsdirektoratet. (2019c). *Algoritmisk tenkning*. Udir. <https://www.udir.no/kvalitet-og-kompetanse/profesjonsfaglig-digital-kompetanse/algoritmisk-tenkning/>

- Utdanningsdirektoratet. (2020). *Innføring av nye læreplaner*. Udir. <https://www.udir.no/laring-og-trivsel/lareplanverket/fagfornyelsen/innforing-av-nye-lareplaner/>
- Ringdal, K. (2018). *Enhet og mangfold – Samfunnsvitenskap forskning og kvantitativ metode*. (4 utgave) Bergen: Fagbokforlaget
- Schwab, K. (2017). *The fourth industrial revolution*. Crown Business.
- Wing, J. M. (2006). Computational thinking. *Commun. ACM* 49, 3 (March 2006), 33–35. <https://doi.org/10.1145/1118178.1118215>
- Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions Royal Society A.*, 366, 3717–3725. <http://doi.org/10.1098/rsta.2008.0118>
- Wing, J. M. (2010). *Computational thinking: what and why?* <https://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf>
- Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., Wilensky, U. (2016). Defining Computational Thinking for Mathematics and Science Classrooms. *Journal of Science Education and Technology*, 25, 127-147. <https://doi.org/10.1007/s10956-015-9581-5>

Vedlegg

[Vedlegg 1 – Informasjonsskriv og samtykkeerklæring](#)

[Vedlegg 2 – Referanse om godkjenning fra NSD](#)

[Vedlegg 3 – Intervjuguide](#)

Vedlegg 1 – Informasjonsskriv og samtykkeerklæring

Vil du delta i forskningsprosjektet

«Implementering av algoritmisk tenkning i matematikkopplæring»

Dette er et spørsmål til deg om å delta i et forskningsprosjekt hvor formålet er å undersøke hvordan lærere på ulike skoler i Bergen og omegn har forberedt seg for innføringen av deler av den nye læreplanen, Fagfornyelsen. I dette skrivet gir jeg deg informasjon om formålet i prosjektet og hva deltakelse vil innebære for deg.

Formål

Forskningsprosjektet er en masteroppgave hvor oppgavens formål er å få mer innsikt i hvordan den nye læreplanen i matematikk, herunder algoritmisk tenkning implementeres i matematikkopplæringen i skoler i Bergen og omegn. Det vil være nyttig å undersøke begrepet «algoritmisk tenkning», gjennom intervju med lærere, formelle definisjoner og intensjonen bak begrepet. Ønsket mitt er at resultatene fra undersøkelsen kan være veiledende til hvordan man kan utforme undervisningsplaner og realisere praktisk undervisning og opplæring knyttet til algoritmisk tenkning og programmering. Hensikten med oppgaven er å belyse hvordan ulike skoler innfører og implementerer algoritmisk tenkning i matematikkundervisningen.

Hvem er ansvarlig for forskningsprosjektet?

Høgskulen på Vestlandet er ansvarlig for prosjektet.

Hvorfor får du spørsmål om å delta?

I forskningsprosjektets intervju ønskes lærere fra ulike skoler fra ulike steder i Bergen. Det vil være mellom tre og fem deltakere til dette. Skolene som er valgt har kun et kriterium om å ikke være i samme bydel som de andre skolene som deltar.

Hva innebærer det for deg å delta?

Jeg ønsker et enkeltintervju med deg som lærer. Intervjuet vil vare i omtrent 30 minutter. Jeg kommer til å ta lydopptak av intervjuet og skrive notater underveis. Dette er for å sikre at datainnsamlingen til prosjektet blir så riktig som mulig.

Jeg ønsker også å se på halvårsplaner som er lokalt utviklet, ukeplaner, undervisningsplaner og planleggingsskjemaer, planer fra læreverk og andre dokumenter som redegjør for planlegging av undervisning. Det skal ikke lages planleggingsskjemaer for dette prosjektet. Jeg ønsker kun å se på hva som allerede er blitt laget frem til når intervjuet skal holdes.

Det er frivillig å delta

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

Ditt personvern – hvordan vi oppbevarer og bruker dine opplysninger

Vi vil bare bruke opplysningene om deg til formålene vi har fortalt om i dette skrivet. Vi behandler opplysningene konfidensielt og i samsvar med personvernregelverket.

- Opplysningene vil bli bearbeidet slik at det ikke vil være mulig å kjenne igjen deltakerne i prosjektet. Ditt navn og din skole vil bli byttet ut med koder før oppstart av datainnsamling, og en eventuell navneliste vil bli bevart adskilt fra øvrige data.
- Datamaterialet knyttet til forskningsprosjektet, vil ha en adgangsbegrensning, og med det innebærer det også at det føres logg over hvem som har benyttet seg av adgang til datamaterialet, og hvor lenge.
- Opplysningene vil bli oppbevart fortrolig på Høgskulen på Vestlandet, og ikke i private enheter.

I masteroppgaven sin ferdige publisering, vil deltakerne ikke kunne kjennes igjen.

Opplysninger som vil komme frem er at utvalget består av ulike lærere fra ulike skoler i ulike bydeler i Bergen. Oppgaven vil inneholde direktesitat fra intervjuet.

Hva skjer med opplysningene dine når vi avslutter forskningsprosjektet?

Opplysningene anonymiseres når prosjektet avsluttes/oppgaven er godkjent, noe som etter planen er i mai/juni 2021. Personopplysninger som kan knyttes til personidentifikasjon vil da bli slettet eller anonymiseres fullstendig.

Dine rettigheter

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

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

Hva gir oss rett til å behandle personopplysninger om deg?

Vi behandler opplysninger om deg basert på ditt samtykke.

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

Hvor kan jeg finne ut mer?

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

- *Høgskulen på Vestlandet* ved Martine Rekstad, student: 99 35 33 35, 180053@stud.hvl.no, og forsker og veileder Nils Henry Williams Rasmussen, 55 58 55 28, nilsen.henry.williams.rasmussen@hvl.no.
- Vårt personvernombud: Trine Anikken Larsen, Trine.Anikken.Larsen@hvl.no

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

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

Med vennlig hilsen

Prosjektansvarlig
(Forsker/veileder)

Student

Samtykkeerklæring

Jeg har mottatt og forstått informasjon om prosjektet *Implementering av algoritmisk tenkning i matematikkopplæring* og har fått anledning til å stille spørsmål. Jeg samtykker til:

- å delta i intervju
- å delta i spørreskjema før intervju
- dele undervisningsplaner og andre dokumenter som redegjør for planlegging av undervisning

Jeg samtykker til at mine opplysninger behandles frem til prosjektet er avsluttet

(Signert av prosjektdeltaker, dato)

Vedlegg 2 – Referanse om godkjenning fra NSD

Skjermdump hentet fra dialog:

Det innsendte meldeskjemaet med referansekode 343673 er nå vurdert av NSD.

Følgende vurdering er gitt:

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 02.10.2020, samt i meldingsdialogen mellom innmelder og NSD. Behandlingen kan starte.

Vedlegg 3 – Intervjuguide

Intervjuguide

Intervjuene som skal holdes vil ha et fokus på lærernes egen refleksjon over sine faglige interesser og sitt arbeid tilknyttet implementering av Fagfornyelsen i skolen.

Under er det forslag til veiledende spørsmål og forslag til innhold. Intervjuet vil struktureres som et semi-strukturert intervju for å ha mulighet til å komme med oppfølgingsspørsmål dersom jeg ønsker det.

Veiledende spørsmål:

- Hvilket trinn underviser du i matematikk på?
- Har du vært med på en innføring av ny læreplan før?
- Hva vet du om algoritmisk tenkning?
 - Hvor har du lært dette?
- Hva vet du om programmering?
 - Hvor har du lært dette?
- Hvor godt forberedt føler du deg til å undervise i programmering og algoritmisk tenkning?
- Hvor nyttig tror du algoritmisk tenkning og programmering vil være for elevene du har?
- Hvilke matematiske aktiviteter i matematikkopplæringen mener du kan komme innunder temaet algoritmisk tenkning?
- Er skolen med i noen prosjekter som kan falle innunder programmering og algoritmisk tenkning?
- Hvilke læreverker bruker du i din undervisning?
- Mener du skolen din er godt nok utstyrt for å ha gi tilstrekkelig opplæring i programmering og algoritmisk tenkning?
 - Hva har dere av teknologisk utstyr?
 - EKS: Datarom som må bookes, eller har alle elevene chromebooks?
- Hva slags IKT-verktøy bruker du i din matematikkundervisning?
 - Brukes dette regelmessig eller mer som en «problemløsende» metode i undervisningen?
 - Hvor ofte brukes det?
- Hvordan opplever du de konkrete kompetansemålene i matematikk som omhandler algoritmisk tenkning og programmering?
- Hvordan forholder du deg til den nye læreplanen når du planlegger undervisningen? Hvorfor?
- Har du inkludert programmering og algoritmisk tenkning i matematikkundervisningen hittil i år?
 - Hvis nei: Har du tenkt til å ha det senere i dette skoleåret?
 - Hvis ja: har du planlagt dette alene eller sammen med kollegaene på trinnet? Hva er planlagt?
 - Hvis ja på: Har du planlagt alene, eller sammen med kollegaene på trinnet? Hva er planlagt? Og hva er gjennomført?

- Hvilke muligheter ser du ved algoritmisk tenkning i din matematikkundervisning? Hvilke utfordringer?
- Har dere hatt fellestid på skolen hvor dere har diskutert den nye læreplanen i matematikkfaget?
 - Hva har dere gjort?
 - Hva har dere hatt mest fokus på?
- Hvordan mener du covid-19 har påvirket implementeringen av fagfornyelsen?
- Hvordan har covid-19 forandret din undervisning i matematikk?
- Er det noe du ønsker å legge til som kan være nyttig for min oppgave?