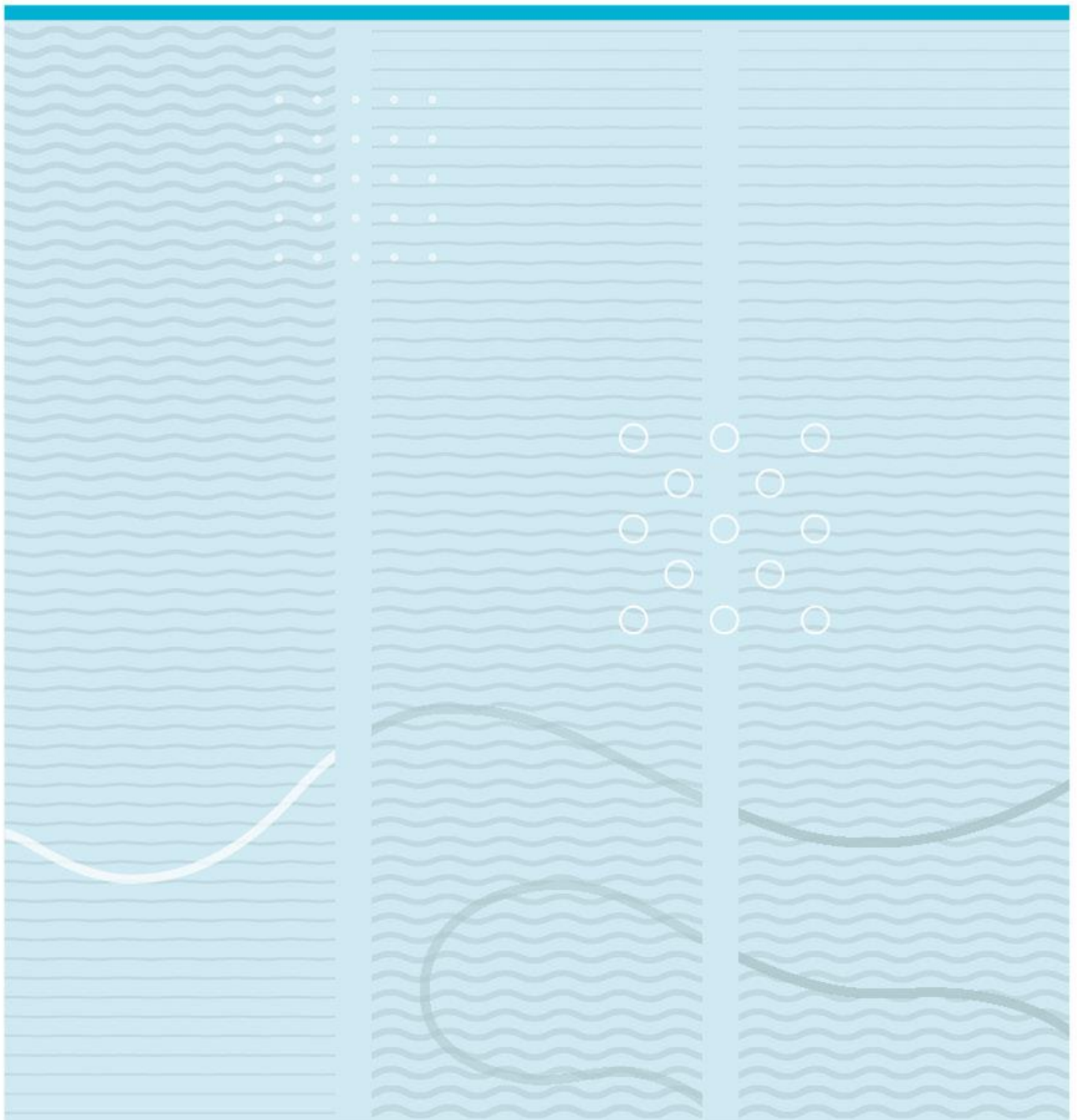


Kay Arne Kongevold Skårdal

Kognitive utfordringer i møte med programmering



Universitetet i Sørøst-Norge
Fakultet for humaniora, idretts- og utdanningsvitenskap
Institutt for matematikk
Postboks 235
3603 Kongsberg

<http://www.usn.no>

© 2022 Kay Arne Kongevold Skårdal

Denne avhandlingen representerer 45 studiepoeng

Sammendrag

Denne studien har som formål, å undersøke hvilke kognitive nivåer som finnes i oppgaver med programmering for to norske læreverk i matematikk. For å dekke det nye kompetansemålet innført ved fagfornyelsen (LK20), vil det undersøkes for i et analogt og digitalt læreverk. Gjennom en innholdsanalyse av de ulike læremidlene, vil jeg besvare følgende spørsmål;

- a) Hvordan er programmeringsoppgaver implementert og vektlagt i læreverkene?
- b) Hvilke nivåer av kognitive krav stiller programmeringsoppgavene til elevene?
- c) Hvilke type svar krever oppgaver med programmering, av elevene?

Dette ble undersøkt for i læreverkene *Maximum* 8. -10. produsert av Gyldendal, samt *Campus Matte* 8. -10. produsert av Campus Inkrement. I analysen ble læreverkene kategorisert for hvilke kognitive krav de stiller til elevene, i tillegg til hvilket type svar oppgavene krever av elevene. Analysens rammeverk er hentet med inspirasjon fra Charalambous et al. (2010), og Stein et al. (2009). Henikten ved å besvare de ulike spørsmålene er å belyse problemstillingen;

I hvilken grad utfordres elevene kognitivt i møte med digitale og analoge programmeringsoppgaver presentert i læreverk for fagfornyelsen?

Resultatene fra studien viser at alle de fire nivåene av kognitive krav er representert i begge læreverkene. Oppgaver som krever høye kognitive krav står for rundt halvparten av oppgavene i det analoge læreverket *Maximum*, mens resterende oppgaver er fordelt på de to lavere nivåene av kognitive krav. For det digitale læreverket *Campus Matte* står oppgaver som krever høye kognitive krav for ca. 65 prosent av den totale oppgave mengden. Totalt sett vil elevene i stor grad bli utfordret kognitivt gjennom arbeid med programmeringsoppgaver produsert for fagfornyelsen.

I læreverket *Maximum* krever den største andelen oppgaver kun et enkelt svar, ved hele ca. 65 prosent. For læreverket *Campus Matte* viser det seg at ca. 78,3 prosent av oppgavene krever et svar med forklaring. For begge læreverkene er det å gi et svar hvor man vurderer gyldigheten til løsningen, sterkt underrepresentert.

Forord

Et femårig studieløp ved Universitetet i Stavanger og Universitetet i Sørøst-Norge, er nå ved veis ende. Gjennom masteroppgaven har jeg fått et innblikk i hva det vil si å drive med forskning, og kunnskaper om kognitive krav i matematikkoppgaver. Etter fullført oppgaven føler jeg meg bedre i stand til å håndtere det nye temaet programmering, og annen tilpasninger av oppgaver for kognitive krav i matematikkfaget. Denne prosessen er noe jeg vil ta med meg videre for å utvikle undervisningspraksisen.

Jeg vil benytte anledningen til å takke venner fra UIS og USN, for en uforglemmelig studietid og mange fine øyeblikker. En ekstra stor takk går til Marthe Bratland, for motiverende ord i en krevende periode. Til slutt vil jeg takke Beate Haugom Bigseth, min veileder, for gode faglige innspill til masteroppgaven.

Drammen, 01.06.2020

Kay Arne Kongevold Skårdal

Innholdsfortegnelse

Sammendrag	3
Forord.....	4
Innholdsfortegnelse	5
1 Innledning.....	7
1.1 Bakgrunn for valg av tema.....	7
1.2 Mål med masteroppgaven	9
1.3 Avgrensning – Godkjennelsesordning for læremidler	10
1.4 Struktur og disposisjon.....	11
1.5 Begrepsavklaring	12
2 Problemstilling og forskningsspørsmål	15
2.1 Problemstilling	15
2.2 Forskningsspørsmål	16
2.3 Digitale verktøy i skolen	17
3 Teori.....	25
3.1 Piaget og Skemp.....	25
3.2 Kognitive krav.....	27
3.3 Dybdeløring.....	31
3.4 Problemløsning for programmering.....	32
3.4.1 Algoritmisk tenkning	33
3.5 Forskning – kognitive krav i læremidler.....	34
4 Metode	35
4.1 Utvalg.....	35
4.2 Innholdsanalyse.....	37
4.2.1 Analysens rammeverk.....	37
4.2.2 Vertikal analyse.....	39
4.2.3 Oversikt av analysens rammeverk	43
4.2.4 Koding av oppgaver	44
4.3 Reliabilitet og validitet.....	45
4.4 Ethiske refleksjoner.....	46
5 Resultater.....	47
5.1 Horisontal analyse.....	47
5.1.1 Bakgrunnsinformasjon - Maximum	48
5.1.2 Helhetlig struktur - Maximum	49

5.1.3	Læringsmålet programmering.....	53
5.1.4	Bakgrunnsinformasjon – Campus Matte	58
5.1.5	Helhetlig struktur	59
5.2	Læringsmålet programmering - Campus Matte.....	63
5.3	Vertikal analyse	69
5.3.1	Kognitive krav – Maximum 8-10	69
5.3.2	Type svar – Maximum 8. -10. trinn	72
5.3.3	Kognitive krav – Campus Matte 8-10.....	73
5.3.4	Type svar – Campus Matte 8-10.....	75
5.4	Kognitive krav og Type svar.....	76
6	Diskusjon	79
6.1	Drøfting av horisontal analyse	79
6.1.1	Bakgrunnsinformasjon	79
6.1.2	Helhetlig struktur	80
6.2	Drøfting av vertikal analyse.....	82
6.2.1	Diskusjon av kognitive krav	82
6.2.2	Diskusjon av type svar	83
7	Konklusjon	85
	Litteraturliste.....	87
	Oversikt over tabeller og figurer	93

1 Innledning

I masteroppgaven vil to læreverker i matematikk undersøkes for kognitive krav, knyttet til matematikkoppgaver i emnet programmering. Dette gjøres ved bruk av en todelt analyse, bestående av en horisontal og vertikal del. Målet med den todelt analysen er å danne en utdannings kontekst rundt programmering, for så vurdere emnet etter hvordan det fremtrer i konteksten det er en del av. Problemstillingen lyder slikt;

I hvilken grad utfordres elevene kognitivt i møte med digitale og analoge programmeringsoppgaver i læreverker produsert for fagfornyelsen?

Videre i kapittel 1. vil jeg gjenta problemstillingen, og utdype dens bakgrunn, mål og mening.

1.1 Bakgrunn for valg av tema

Programmering er kompetansen for det 21. århundre, ifølge et dokument fra Senter for IKT i utdanningen (2016). Senteret for IKT henviser til prosjektet *Assessment & Teaching of 21st Century Skills*, som har definert det de mener er de essensielle ferdighetene for det 21. århundret. En av de brede kategoriene som ferdighetene kategoriseres i, er betegnet «Ways of thinking». I beskrivelsen av kategorien ilegges ord som kreativitet, innovasjon, kritisk tenkning, problemløsning og det å lære å lære (Sevik et al., 2016, s. 10). 15. juni 2015 leverte Ludvigsen-utvalget, ledet av Sten Ludvigsen, en rapport om fremtidens skole. Oppdraget var å vurdere grunnopplæringens fag, opp mot krav til kompetanse i et fremtidig samfunn. Ludvigsen-utvalget poengterer fire kompetanseområder der det er behov for utvikling. Kompetanseområdene samsvarer med *Assessment & Teaching of 21st Century Skills* fokus på ferdigheter. Blant annet ved «kompetanse i å lære» og «kompetanse i å utforske og skape» (NOU 2015: 8, s. 8). Ludvigsen utvalgets trekker linjer mellom det å ha kompetanse i «å lære», og metakognisjon. Det vil si at elevene evner å se sin egen læring, hvilken hensikt læringen har og fremgangsmåtene som blir brukt for å oppnå læring. Kompetansen «utforske og skape» inneholder blant annet kritisk tenking og problemløsning. Her knytter utvalget kompetansen til aspekter ved innovasjon og kreativitet (NOU 2015:8, s. 10).

Kompetansene nevnt ovenfor av Ludvigsen-utvalget er en stor del av programmeringsfaget. Det å drive problemløsning, være kreativ og forstå læringsprosessen er grunnleggende ferdigheter for å kunne drive programmering (Kaufmann, 2018, s. 75-80). For kunnskapsløftet 2020 (LK20) innførte myndighetene programmering, som en del av matematikkfaget. Programmering finnes nå i kompetansemålene for matematikkfaget, og under den grunnleggende digitale ferdigheten (Kunnskapsdepartementet, 2019). Læremidler produsert etter fastsettelsen av fagfornyelsen, har implementert programmering som en del i sine læremidler. Ulike forlag tilbyr oppdaterte analoge og digitale læreverker, med innhold av ulike former for programmering. Problemstillingen som undersøkes i masteroppgaven, omhandler hvilke kvaliteter programmering bringer med seg inn i fagfornyelsen. Er det sånn at elevene blir utfordret av oppgaver med programmering, slik at det stimuleres til metakognisjon og Ludvigsen-utvalget sine andre kompetanser for fremtiden? Eller stimulerer oppgaver med programmering til kompetanser, forbundet til den tradisjonelle pugge skolen?

Personlig grunn for valg av tema

Norsk forskning omkring programmering er minimal, og i tillegg til forskning på læremidler som tilbyr programmering, er enda mindre. Årsaken til dette er fordi programmering ikke har vært en del av tidligere læreplaner. For institusjoner som jobber med å forberede studenter til læreryrket har ikke programmering vært en del av matematikkfaget. Selv arbeid ved digitale verktøy, som middel for å oppnå læring, har vært fraværende i den nye femårige utdanningen ved Universitet i Stavanger. Ensrettet fokus på matematiske objekter og teori, uten særlig innblikk i matematikdidaktikk. Kjentegnet det generelle undervisningsforløpet. Fraværende fokus på programmering, digitale verktøy og matematikdidaktikk, har ført til en usikkerhet rundt håndteringen av programmering i matematikkfaget. Forutsetningene for å oppfylle LK20, og samtidig gi elevene et godt læringsutbytte ved bruk av programmering i undervisningen, er lave. Usikkerheten rundt programmering i matematikkfaget gjenspeiles også i den generelle oppfatningen til ansatte i skolen. Dette er ikke den generelle oppfatningen alle ansatte på skolen har, men er basert på egne erfaringer, basert på samtaler med ansatte i skolen, studenter og studenter på kryss av universiteter.

Dette medfølger at jeg i starten av yrkeskarrieren, trolig vil hente inspirasjon fra læreverker, for å kunne se deres tolkning av kompetansemålene i LK20 og integreringen

programmering som et digitalt verktøy. Masteroppgaven vil derfor tjenes som en kvalitetssjekk av læremidler på ungdomstrinnet, og som en innføring i det matematiske temaet for videre arbeid i yrkeskarrieren. Ettersom ulike forlag tilbyr varierte læreverker, både digitalt og analogt, vil jeg i denne masteroppgaven analysere for læreverker i begge formatene. Målet med undersøkelsen av programmering som tema, er å tydeliggjøre hvilke krav temaet stiller elevene. Ved å være bevisst på hvilke krav programmering stiller elevene, tror jeg at det er lettere å tilpasse oppgavene til det elevmangfoldet som finnes i skolen.

1.2 Mål med masteroppgaven

I masteroppgaven vil jeg ta for meg et spesifikt analogt læreverker; *Maximum* og det digitale læreverket; *Campus Matte*. *Maximum* er produsert av forlaget Gyldendal og er oppdatert på lik linje med *Campus Inkremt*'s læreverker i matematikk for fagfornyelsen (LK20). Læremidlene; *Maximum 8-10. 2. utgave* og *Campus Matte 8-10.*, vil utgjøre oppgavens datamateriale. Gjennom masteroppgaven vil jeg foreta meg en horisontal og vertikal analyse av læreverkene. Hensikten med den horisontale analysen er å få et oversiktsbilde på hva som tilbys og hvordan forfatterne har valgt å implementere programmering i læreverkene. Et tydelig bilde av læreverkenes struktur og vektlegging av programmering vil sette kompetansemålet inn i en kontekst (Charalambous et al., 2010, s. 123). Etterfulgt av den horisontale analysen vil jeg foreta en vertikal analyse av programmeringsoppgavene i læreverkene. I den vertikale analysen vil jeg analysere programmeringsoppgavene for hvilke kognitive krav elevene stilles (Charalambous et al., 2010, s. 123). For å definere og sette kriterier for de ulike nivåene av kognitive krav vil jeg hente inspirasjon fra Stein et al. (2009) og deres *Task analysis guide* (TAG). Den horisontale og vertikale analysen vil bli utført for begge læreverkene med et mål om å svare på problemstillingen:

I hvilken grad utfordres elevene kognitivt i møte med digitale og analoge programmeringsoppgaver i læreverker produsert for fagfornyelsen?

For å belyse problemstillingen vil jeg utlede to sentrale forskningsspørsmål. Det første spørsmålet omhandler programmeringens implementering og vektlegging i læreverkene, mens det andre spørsmålet omhandler kognitive krav i oppgaver med programmering. Spørsmålene lyder slik:

1. Hvordan er programmeringsoppgaver implementert og vektlagt i læreverkene?
2. Hvilke nivåer av kognitive krav stiller programmeringsoppgavene til elevene?

Det første forskningsspørsmålet vil hovedsakelig besvares gjennom den horisontale analysen, mens det andre forskningsspørsmål vil besvares gjennom den vertikale analysen.

1.3 Avgrensning – Godkjennelsesordning for læremidler

Forskning på læremidler er et komplekst forskningsfelt med varierende datamateriell, og de er tett knyttet opp mot politiske makter og samfunnets stadige endrede kompetansebehov. Hvilke kompetanser og ferdigheter et samfunn ser på som essensielle for fremtiden, er unikt og springer ut fra samfunnet selv. Hvilke type forskningsmaterieell og mengde avhenger i stor grad av hvilke samfunn som studeres. For å gjøre en avgrensning av forskningsfeltet vil jeg derfor kort nevne godkjenningsordningen (1889-2000) for læremidler i Norge, som vil avgrense forskningsfeltet.

Godkjenningsordningen 1889 - 2000

Tradisjonen med en godkjenningsordning av læremidler ble avvirket juni 2000. Ordningen sørget for en regulering av innhold og påvirket lærerens praksis gjennom læreboken ettersom lærebokens plass i den norske skole (Bratholm, 2001). Godkjenningsordning for lærebøker 1889-2000 var en statelig ordning som satte føringene for innholdet i den tradisjonelle læreboken. Innholdet i lærebøkene la grunnlaget for undervisningen i klasserommet. Hva som ble godkjent innhold og hva som ikke var akseptabelt i samfunnet og politikken.

Ordningen var diskutert innad i politikken, og kritikken lot seg ikke vente på. Blant annet fikk ordningen kritikk fra Ytringsfridoms-kommisjonen som påpekte at godkjenningsordningen fremstod som et kontrollorgan, og kunne være til hindring mot mangfold av perspektiver på utdanningsfeltet. Smith-utvalget påpekte at ordningen begrenset friheten til lærerne i valg av riktig læremiddel tilpasset den lokale konteksten (Bratholm, 2001). På den andre siden var flere instanser på banen og uttrykte sin bekymring for standardisering og kvaliteten på bøkene, som kunne komme uten en slik

ordning. Norske lærarlag, Lærarforbundet, LO og Nasjonalt læremiddelsenter var alle negative til opphevelsen godkjenningsordningen (Prop. 44 (1999-2000), s. 28).

Argumentene for å beholde ordningen var at den sikret kvalitet i språk og innhold, samt trygghet til foreldre om at lærebøkene var i tråd med læreplanen (Prop. 44 (1999-2000)).

Tilgjengelig forskningsmateriale etter opphevelsen av godkjennelsesordningen sammenlignet med slik det var, har derfor økt. Ikke bare mengden forskningsmaterieell, men også konkurransen rundt å være den fremste aktøren på læremidler. Gjennom konkurranse og opplæringlovens formelle krav er læremidlene i stadig endring. Et av de klareste eksemplene på den hurtige utviklingen av læremidlene, er digitale mediers inntog i samfunnet og skolen. I matematikkfaget ser vi dette med innførelsen av digitale verktøy som en del av den grunnleggende digitale ferdigheten. Digitale ferdigheters inntog i LK06, opphevelsen av godkjenningsordningen i år 2000 og samfunnets digitale utvikling etter tusenårsskiftet, fører utdanningen i en digitaliserende retning. Dette fant sted senest ved innføringen av programmering i matematikkfaget ved LK20, og er også grunnen til at jeg vil avgrense forskningsfeltet til forskning utført etter den digitale vendingen i skolen og samfunnet.

Rapporten *Internasjonal forskning på læremidler – en kunnskapsstatus* (2011)

konkluderer med at det finnes lite forskning rundt grunnleggende ferdigheter. Fra kunnskapsstatusen trekker rapporten frem tre typiske trekk ved læremiddelforskning;

- Innhold, form og bruk.

En viktig avgrensning i masteroppgaven er at den kun tar for seg innholde-form som blir presentert i læreverkene. Dermed er det ikke mulig å si noe om hvordan programmeringsoppgaver fremtrer i de ulike klasserommene. Det er bare mulig å si noe om innholdet i læreverkene og hvilke utgangspunkt læremidlene tilbyr læreren.

1.4 Struktur og disposisjon

Etterfulgt av en enkel avgrensning for forskningsfelt i oppgaven, ble det innledningsvis gjort en kort presentasjon av problemstilling med tilhørende forskningsspørsmål.

Problemstillingen er kontinuerlig nevnt og gjentatt gjennom masteroppgaven, men behandles først i kapittel 2., sammen med forskningsspørsmålene. I kapittel 2., utdypes problemstillingen sammen med forskningsspørsmålene før de blir satt inn i en historisk kontekst om digitale verktøy i skolen.

Med en forståelse av hva som legges i masteroppgavens mål fra kapittel 2., vil kapittel 3., danne teorigrunnet for masteroppgaven. Teorien er valgt ut med bakgrunn i problemstillingen slik at den gjenspeiler målet med oppgaven, og har til hensikt å belyse empirien vitenskapelig. Gjennom teorien vil det utføres analyser og tolkninger som til sammen utgjør innholdsanalysen.

For å redegjøre for oppgavens metode vil kapittel 4., legge vekt på hvilke valg som er tatt i arbeidet med masteroppgavens problemstilling. Dette gjøres for å øke oppgavens validitet, samt reliabilitet. Et viktig punkt her er hvordan datainnsamlingen har foregått og hvordan tolkningene av datamaterialet er utført. Gjennom et kritisk blikk vurderes veivalgene og den generelle utformingen av oppgaven med hensyn til det som er ønsket undersøkt i problemstillingen.

På bakgrunn av de to foregående kapitlene blir det i kapittel 5., presentert resultater av innholdsanalysen. Her presenteres og kategoriseres resultater fra den horisontale og vertikale analysen. Fra teorikapittelet (3.) vil forhåndsbestemte kategorier bli brukt for å kategorisere datamaterialet. Interessante funn som avviker fra normalen vil bli trukket frem for videre behandling i kapittel 6. For å øke oppgavens reliabilitet vil det her også bli presentert eksempeloppgaver som kjennetegner de ulike nivåene i den vertikale analysen. Avslutningsvis vil resultatene drøftes i kapittel 6., der målet er å knytte teori og empiri sammen for å besvare problemstillingen. Her vil det løftes frem spenninger problemstillingen byr på. For å oppsummere vil kapittel 7., gjenta de viktigste momentene fra masteroppgaven der de blir satt inn i et større perspektiv før det konkluderes rundt problemstillingen.

1.5 Begrepsavklaring

Før selve presentasjonen vil det bli presentert en begrepsavklaring av sentrale begreper som er viktige for å kunne orientere seg i læremiddelforskningen. Begrepene er hentet fra Utdanningsdirektoratets (Udir) nettside. Begrepene under er hentet fra Udir sin artikkel om; *Læremidler og læringsteknologi i skole og opplæring* (Utdanningsdirektoratet, 2021). Her legger Udir føringer for hva som kan kategoriseres som læremidler og hva de ulike begrepene skal inneholde. Når et forlag ønsker å gi ut et

læremiddel ser de til *Lov om grunnskolen og den vidaregåande opplæringa* (Opplæringsloven) definisjoner av et læremiddel og Kunnskapsløftet.

Læremiddel kjennetegnes ifølge Udir ved at det «er utviklet til bruk i opplæringen», «er i regelmessig bruk» og «dekker elementer i læreplanverket» (Utdanningsdirektoratet, 2021). Det utdypes videre at dette gjelder alle trykte kilder, men også digitale midler.

Læreverk og **læringsunivers** kan forstås som to sider av samme sak. Der læreverk hovedsakelig blir brukt til et forlags samlede trykte læringsmidler, er læringsunivers tilsvarende digitalt. I masteroppgaven vil det bli mest korrekt å bruke læreverk om det analoge læreverket Maximum, og læringsunivers om det digitale læreverket Campus Matte.

Læringsressurs forstås ifølge Udir som «materiell med fagrelevant informasjon som blir integrert i læringsarbeidet på en didaktisk måte» (Utdanningsdirektoratet, 2021). Materiell som ligger utenfor forlagets læreverk og læringsunivers kan fremkomme i læringsmidlene. Spesielt programmering og programmer som støtter undervisning i det nye kompetansemålet for LK20. I studien vil det være viktig å vite hva som er en del av det digitale læremidlet og hva som er en digitale ressurs.

Digitale verktøy er relevant for å sette problemstillingen inn i en kontekst. Jeg velger derfor å sitere hele utsnittet fra LK06.

«Digitale verktøy har funksjonalitet for å skape, bearbeide, lagre og/eller dele fagrelevant informasjon. Innholdet hentes fra ulike kilder, for eksempel fra læremidler eller læringsressurser. Digitale verktøy er programvare eller plattformer som ikke primært er utviklet med tanke på eller for bruk i læringsarbeidet. Typiske eksempler er spillbaserte verktøy, redigeringsprogram for medieinnhold (tekst, tall, lyd, film, bilde), samhandlingsverktøy, presentasjonsverktøy og sosiale medier.» (Utdanningsdirektoratet, 2021).

Den grunnleggende digitale ferdigheten i LK20 er også relevant for oppgavens mål om å utforske programmering og vil derfor også presenteres i sin helhet her.

«Digitale ferdigheter Digitale ferdigheter i matematikk innebærer å kunne bruke graftegner, regneark, CAS, dynamisk geometriprogram og programmering til å utforske og løse matematiske problemer. Videre innebærer det å finne, analysere, behandle og presentere informasjon ved hjelp av digitale verktøy. Utviklingen av digitale ferdigheter innebærer i økende grad å bruke og velge hensiktsmessige digitale verktøy som hjelpemiddel for å utforske, løse og presentere matematiske problemer.» (Kunnskapsdepartementet, 2019, s. 5).

2 Problemstilling og forskningsspørsmål

I kapittel 2. vil jeg gå nærmere inn på hva forskningsspørsmål og problemstilling inneholder. Jeg vil aktualisere problemstillingen ved å sette den inn i en utdannings kontekst. Konteksten er utformet med bakgrunn i den grunnleggende digitale ferdigheten, slik den er presentert i LK06, og LK20.

2.1 Problemstilling

Implementering av ny læreplan er ikke alltid en selvfølge og møter ofte kritikk. Ser vi til Goodlad (1979) fem nivåer for læreplaner, kan det tenkes at den ideelle og formelle læreplan, ikke ser lik ut den operasjonaliserte læreplan. Som styringsdokument for den norske skolen er LK20 åpen for tolkning, særlig i spørsmål om metode. Hvordan læreplanens mål skal oppnås, ligger på den profesjonelle pedagog. Valgene en lærer tar er avgjørende for elevens læring. Dette gjelder også for læremidler, og hvilken type kunnskap og metoder som tilbys her. Ved problemstillingen skal det undersøkes for hvilke kognitive krav oppgaver med programmering stiller elevene. Oppgaver med programmering defineres i denne oppgaven, som oppgaver der det skal lages et program eller en annen form for bruk, eller analysering av koder.

Ved problemstillingen er målet å få frem hvilke kognitive krav, og type svar, oppgaver med programmering stiller til elevene gjennom to ulike læreverk. Det kognitive aspektet ses i sammenheng med «21st. century skills», presentert innledningsvis gjennom blant annet Ludvigsen-utvalget. Videre lyder problemstillingen slik:

I hvilken grad utfordres elevene kognitivt i møte med digitale og analoge programmeringsoppgaver presentert i læreverk for fagfornyelsen?

Til å svare på problemstillingen er forlaget Gyldendal ved læreverket *Maximum* og Campus Inkrement ved *Campus Matte*, valgt ut og vil være forskningsobjekt i denne innholdsanalysen. Gyldendal vil representere det analoge læreverket, og vil heretter omtales bare Gyldendal eller *Maximum*. Campus Inkrement vil derimot representere det digitale læreverket i masteroppgaven, og vil bli omtalt som *Campus Matte*. Begge læreverkene har produsert eller oppdatert sine verk for fagfornyelsen, og inneholder oppgaver om programmering.

2.2 Forsknings spørsmål

Udir omtaler på sine nettsider at; «Skolen skal planlegge, gjennomføre, tilpasse og vurdere opplæringen slik at elevene oppnår den kompetansen læreplanen beskriver» (Utdanningsdirektoratet, 2021, s. 8). Her er det tydelig at det er skolens ansvar for tilpasning av opplæringens innhold slik at elever oppnår den kompetansen som er beskrevet. Kompetansen skal oppnås gjennom dybdelæring der sentrale elementer og sammenhenger i fag trekkes frem. Forståelsen elevene utvikler gjennom dybdelæring skal gjøre elevene i stand til å anvende kunnskapen på ulike nivåer (Utdanningsdirektoratet, 2021, s. 11). Tilpasningene og metoden for å oppfylle læreplanens hensikt ligger på den enkelte pedagog. Ifølge en kartleggingsrapport om læremidler og læremiddelpraksis, Skjelbred (2003), har læreboken fremdeles en sentral plass i det norske klasserommet. Dersom lærere henter inspirasjon fra ulike læremidler er det essensielt at kvaliteten holder stand. For at elevene skal bedrive dybdelæring må de utfordres kognitivt. Oppgaver der fokuset er på prosessen og hvordan de sentrale elementene i faget kan sees i sammenhenger på ulike plan. For å besvare problemstillingen på en systematisk og oversiktlig måte deles den inn i de to tidligere nevnte forsknings spørsmålene:

- d) Hvordan er programmeringsoppgaver implementert og vektlagt i læreverkene?
- e) Hvilke nivåer av kognitive krav stiller programmeringsoppgavene til elevene?

Forsknings spørsmål a

Hvordan er programmeringsoppgaver implementert og vektlagt i læreverkene, er stilt med bakgrunn i programmering som en del av kompetansemålene i matematikkfaget, i tillegg til programmering som en del av den grunnleggende digitale ferdigheten for matematikkfaget. I den grunnleggende digitale ferdigheten skal elevene kunne «bruke graftegner, regneark, CAS, dynamisk geometriprogram og programmering til å utforske og løse matematiske problemer» (Kunnskapsdepartementet, 2019, s. 15). Etersom at det står at elevene skal utforske og løse matematiske problemer, antyder læreplanen at fokuset skal være på både prosessen som på løsningen. Ved å se på hvordan programmeringsoppgavene er implementert og vektlagt vil det belyse problemstillingen på flere områder; dybdelæring, kognitive krav og hvordan programmering brukes som et enkelt verktøy for å løse matematiske problem eller som et middel for utforskning.

Forsknings spørsmål b

Hvilke nivåer av kognitive krav stiller programmeringsoppgavene til elevene stilles for å belyse hvordan elever utfordres av oppgaver med programmering.

Programmeringsoppgavene som vil bli vurdert og analysert, kjenntegnes ved at de

- Inneholder en form for bruk av et programmeringsverktøy
- Arbeid med koder og programmeringsspråk

Som nevnt tidligere i læreplanens generelle del, er et sentralt aspekt med LK20 at elevene skal drive dybdeløring i fagene. Dybdeløring i et kognitivt perspektiv forutsetter at en har fokus på hva som skal læres og hvordan denne læringen skjer. Ved å se på det kognitive nivåer i programmeringsoppgaver er det mulig å finne ut om oppgavene oppfordrer til en dypere forståelse (Gilje et al., 2018: 24).

2.3 Digitale verktøy i skolen

Det vi kjenner som digitale ferdigheter i dag ble først innført med kunnskapsløftet 2006 (LK06), sammen med 4 andre grunnleggende ferdigheter;

- å kunne lese
- å kunne regne
- å kunne uttrykke seg muntlig
- å kunne uttrykke seg skriftlig

Sammen med de andre grunnleggende ferdighetene, ble de innført etter en større debatt om hvilken kompetanse som skulle være definert som essensiell i det fremtidige samfunnet. Utvalg og definisjon av kompetanse stod Organisasjonen for økonomisk samarbeid og utvikling (OECD) for. Prosjektet ble kalt *Definition and selection of competence* (DeSeCo), og skulle undersøke hvilke kompetanser som ble vurdert som viktige. Blant kompetansene som ble fremhevet, var blant annet det å kunne orientere seg digitalt sentralt. I etterkant av det internasjonale prosjektet kommer det norske svaret på *DeSeCo* funn, nemlig digitale ferdigheter slik det opptrådte i LK06. De fem grunnleggende ferdighetene er også er videreført til LK20. Av stortingsmelding nr. 28 kommer det frem at departementet vil videreføre de grunnleggende ferdighetene og innarbeidet, dem i læreplanen der det vurderes som faglig relevant (Meld. St. 28 (2015–2016), s. 32).

Digitale verktøy

Innføringen av LK06 og digitale ferdigheter som en av de fem grunnleggende ferdighetene understreker ferdigheten sin plass, ikke bare internasjonal, men også i

Norge som en fremtidig kompetanse. I stortingsmelding nr. 30 (Meld. St. 30 (2003-2004), s.48) definerer stortinget hva digital kompetanse innebærer. Her beskriver Stortinget en sammensatt kompetanse av enkle informasjons- og kommunikasjonsteknologi (IKT) ferdigheter. Kreativitet ved bruk av programvarer, søk og håndtering av data. I tillegg til en kritisk evne til å tolke og evaluere informasjon, beskrives kompetansens formål som det «...å kunne bruke og hente frem, lagre, skape, presentere, vurdere og utveksle informasjon» (Meld. St. 30 (2003-2004), s.34). En lignende definisjon er også brukt i gjeldende læreplan for matematikkfaget (MAT01-05). Definisjonen inneholder blant annet at «Digitale ferdigheter i matematikk inneber å kunne bruke grafteiknar, rekneark, CAS, dynamisk geometriprogram og programmering til å utforske og løyse matematiske problem», og «...bruke og velje formålstenlege digitale verktøy som hjelpemiddel for å utforske, løyse og presentere matematiske problem» (Kunnskapsdepartementet, 2019, s. 5). Ulikhetene av de to tilnærmingene til den digitale ferdighetene er derimot ganske tydelige. Utviklingen har gått fra en digital ferdighet der fokuset er på informasjon og presentasjon, til utforskning og problemløsning. Norges offentlige utredninger (NOU, 2013: 2) påpeker at den grunnleggende ferdigheten, slik den var presentert i LK06, ikke var et mål i seg selv. Ferdigheten ble nærmere et middel for å oppnå andre kompetansemål (NOU 2013: 2, s. 99). NOU (2013) påpeker at også at den reviderte digitale ferdigheten fra 2012 er «...nesten ensidig på kommunikasjon og presentasjon, og ikke på utvikling, programmering, beregning, analysering eller teknisk forståelse» (NOU 2013: 2, s. 101). I den nye definisjonen av digitale ferdigheter ser vi at ferdigheten har i større grad har bitt et mål i seg selv. Gjennom utforskning og problemløsning av matematiske oppgaver er ikke digitale verktøy kun ment for å presentere og kommunisere andre kompetansemål.

Arbeidet med digitale ferdigheter - tilbakeblikk

For å forstå hvor vi befinner oss i arbeidet med digitale ferdigheter og det digitale verktøy som programmering er, kan et historisk tilbakeblikk rundt temaet sette problemstillingen inn i kontekst. Etter implementeringen av grunnleggende ferdigheter i LK06 ble det igangsatt flere prosjekter med hensikt i å følge utviklingen av læreplanen og virkningen av den. Blant prosjektene som startet som en del av følgeforskningen til LK06 finnes;

- Sammenheng mellom undervisning og læring (SMUL)

- Forvaltningsnivåenes og institusjonenes rolle i implementeringen av reformen (FIRE)

SMUL prosjektet som hovedsakelig var en evaluering hovedsakelig basert på en analyse av ulike undervisningsformer, tok også for seg undervisning i grunnleggende ferdigheter. I det femte kapittelet i SMUL-prosjektet, konkluderes det med at målene for timen i større grad var knyttet til fagkunnskap i de forskjellige fagene fremfor den grunnleggende ferdigheten. Dette samsvarer med NOU (2013: 2), nevnt tidligere i kapittelet om hvordan grunnleggende ferdigheter må vike for fagkunnskapen. Etter en utvidet undersøkelse av planer, praksis og lærernes egne tanker omkring grunnleggende ferdigheter i LK06, konkluderer rapporten at ferdighetene ikke behandles som et mål i seg selv (Hodgson et al., 2012, s. 186). Tilrådingen av rapporten kom med, etter undersøkelsene omkring grunnleggende ferdigheter, var at temaet burde klargjøres. Grunnleggende ferdigheter må kommuniseres på en måte slik at lærerne forstår hva det vil si å arbeide med temaet i klasserommet (Hodgson et al., 2012, s. 190).

FIRE-rapporten var også en del av evalueringen til LK06. I evalueringens intervju-analyse ble det stilt spørsmål omkring grunnleggende ferdigheter. Rapporten finner tilnærmet lik informasjon som SMUL, og konkluderer med; «...at både rektorer og lærere er svært oppmerksomme på de grunnleggende ferdighetene, men uten at det ser ut til å ha fått særlige store konsekvenser for undervisningen» (Aasen et al., 2012, s. 243). Det utdypes videre at grunnleggende ferdigheter «Oppfattes som et anliggende for den enkelte lærer mer enn for skolen som fellesskap.» (Aasen et al., 2012, s. 243). En slik oppfatning bekrefter i større grad at det er mangel på et felles arbeid og en felles forståelse rundt grunnleggende ferdigheter. Til tross for dette mener et betydelig flertall i tabell 7.16 (Aasen et al., 2012, s. 249) at læreplanverket har lyktes i å få digitale ferdigheter inn i de ulike fagene.

Læreplanverket har lyktes med å få lærerne til å fokusere på undervisning i digitale ferdigheter i de ulike fagene	Vet ikke	Helt uenig	Nokså uenig	Verken enig eller uenig	Nokså enig	Helt enig	Total (N)
Kontaktlærere 4. og 7. trinn	3	1	4	14	54	23	160
Lærere på 10. trinn	4	4	11	15	56	9	140
Lærere i videregående skole	6	3	7	19	51	13	612

Tabell 1: Undervisningen av digitale ferdigheter i alle fag. Lærere 2011. Prosent. (Aasen et al., 2012, s. 249).

At digitale ferdigheter fokuseres på i fagene uten at det uttrykkes en felles forståelse av hvordan arbeidet med temaet ser, ut er oppsiktsvekkende. Det kan tenkes at undervisningen i forhold til temaet var særdeles varierende og ulik. Usikkerheten gjenspeiler seg i tabell 7.8 (Aasen et al. 2012) der en god andel rektorer på grunnskolenivå (ca. 46%) svarer at de er avhengig av at læreboka ivaretar temaet for å sikre et godt fokus og arbeide med det. Selv om den kvantitative spørreundersøkelsen ikke tar opp hva lærerne legger i begrepet, er det avslørende hvor varierende dette arbeidet kan se ut. Et eksempel er påstanden stilt i tabell 7.21 om hvor ofte lærere gjør noe for å utvikle elevene rundt digitale verktøy;

Hvor ofte gjør du noe for at elevene skal utvikle seg til å bli gode til å kunne bruke digitale verktøy?	Vet ikke	Sjeldnere	Ukentlig	Omtrent hver dag	Omtrent hver eneste time	Total (N)
Kontaktlærere 4. og 7. trinn	1	13	50	33	2	161
Lærere på 10. trinn	0	26	47	23	4	137
Lærere i videregående skole	2	18	35	29	16	616

Tabell 2: *Hypptigheten av arbeid med digitale verktøy. Lærer 2011. Prosent.* (Aasen et al., 2012, s. 250).

Ut ifra tallene ser vi at det er en betydelig del som svarer ukentlig, eller sjeldnere enn ukentlig. Dersom læreplanens intensjon om at grunnleggende ferdigheter skal inngås i hvert fag, er det interessant å se en så stor andel på ukentlig eller sjeldnere. Funnene i FIRE viser at det står i kontrast til lærernes mening om at LK06 har lyktes med digitale ferdigheter, mens det stemmer overens med oppfatningen om at digitale ferdigheter er koblet til IKT-utstyr og lærerens kompetanse fremfor arbeidet i klasserommet (Aasen et al. 2012, s. 251).

Fra politisk styringsdokument til klasserommet er det ikke bare SMUL som har påpekt arbeidet med grunnleggende ferdigheter generelt. Artikkelen *Ti år med grunnleggende ferdigheter – hva vet vi, og hvor går vi* (2018) av Gilje og Røyne henviser til nettopp flere av evalueringene når de skriver om en felles tendens i evalueringsprosjektene etter LK06. Tendensen Gilje og Røyne beskriver er oppfattelsen av at temaet grunnleggende ferdigheter ikke ble ansett som viktig, eller ikke var spesifisert nok. Forståelsen av hva grunnleggende ferdigheter egentlig inneholdt, var varierende og for personavhengig. FIRE-rapporten beskriver to måter å forstå arbeidet med grunnleggende ferdigheter på. Den ene er at ferdighetene arbeides eksplisitt med. I dette tilfelle vil det være en form for undervisning i hvordan de utøves, mens den andre ser på grunnleggende ferdigheter

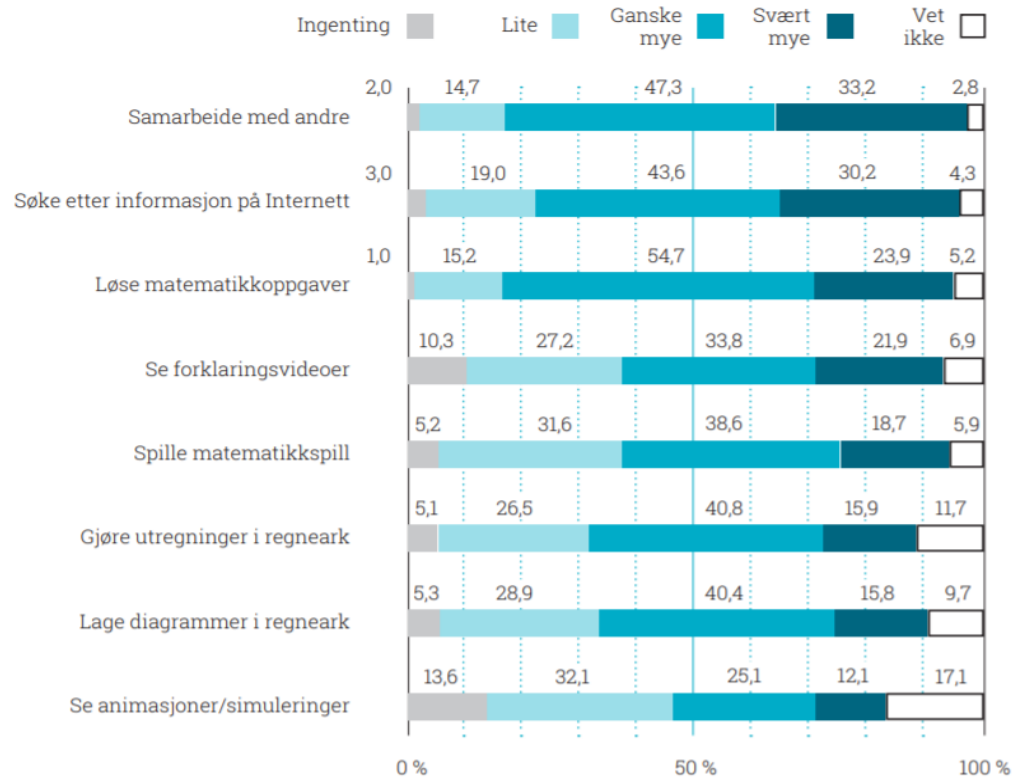
som en implisitt del av undervisningen. Det vil si at det ikke er egen undervisning i grunnleggende ferdigheter, men at de er en del av den vanlige undervisningen der det eksempelvis brukes digitale verktøy som løsningsmetode (Aasen et al., 2012, s. 253).

Forskning på digitale verktøy i LK06

I tillegg til følgeforskningen vil jeg trekke frem to undersøkelser som har sett nærmere på digitale verktøy i skolen. Den første studien er ARK&PAPP, som baserer seg på 12 casestudier blant fire forskjellige fag, der matematikk er ett av dem. Rapporten fra Gilje og Rødnes (2016) ser på innhold og form, og tar opp hvilken funksjon læremidlene har i læringsarbeidet. Forskningen ble gjort for å kunne avdekke statusen til læremidler i den norske skole, samt grunnleggende ferdigheters plass. Studien ble oppsummert i tre rapporter hvor den ene rapporten omhandler forholdet mellom læremiddel og grunnleggende ferdigheter. Rapporten ARK&PAPP, har ikke fokusert primært på grunnleggende ferdigheter, men danner til sammen et bredt bilde av hvordan praksisen foregår. Det legges to forhold til grunn for en forståelse av grunnleggende ferdigheter i undersøkelsene. Det første forholdet er at grunnleggende ferdigheter «må forstås som redskaper for å *tilegne seg* faglig kunnskap, og som redskaper for å *presentere* fagstoff man tilegner seg.» (Rødnes & Gilje, 2016, s. 9). Rapporten henviser også til følgeforskningen av LK06, der grunnleggende ferdigheter var uklart definert i plandokumenter. Det kom frem at det ikke bare var blant lærerne usikkerheten lå. Også skoleledere uttrykte en usikkerhet rundt temaet. Funnene i ARK&PAPP, nyanserer funnene i følgeforskningen til LK06 ved at det i case-studiene er grunnlag for å si at det jobbes med ferdighetene på ulike måter (Rødnes & Gilje, 2016, s. 41). Det påpekes at undersøkelsene i følgeforskningen hovedsakelig var basert på intervju, spørreundersøkelser og dokumentanalyser, med noen klasseromsobservasjoner. Rapporten understreker at elevens digitale ferdigheter er hovedsakelig blitt forsket på gjennom spørreundersøkelser og ikke empirisk, altså hvordan digitale ferdigheter inngår i arbeidsformene i fagene (Rødnes & Gilje, 2016, s. 8-9). Lærerne ble i en undersøkelse utført av Waagene & Gjerustad (Rødnes & Gilje, 2016, s. 10), stilt en påstand om lærebokas betydning for implementeringen av grunnleggende ferdigheter. I de fire fagene som ble undersøkt, deriblant matematikk, besvarte mellom 50 og 60 prosent spørsmålet ved at de er enige i bokens viktige posisjon.

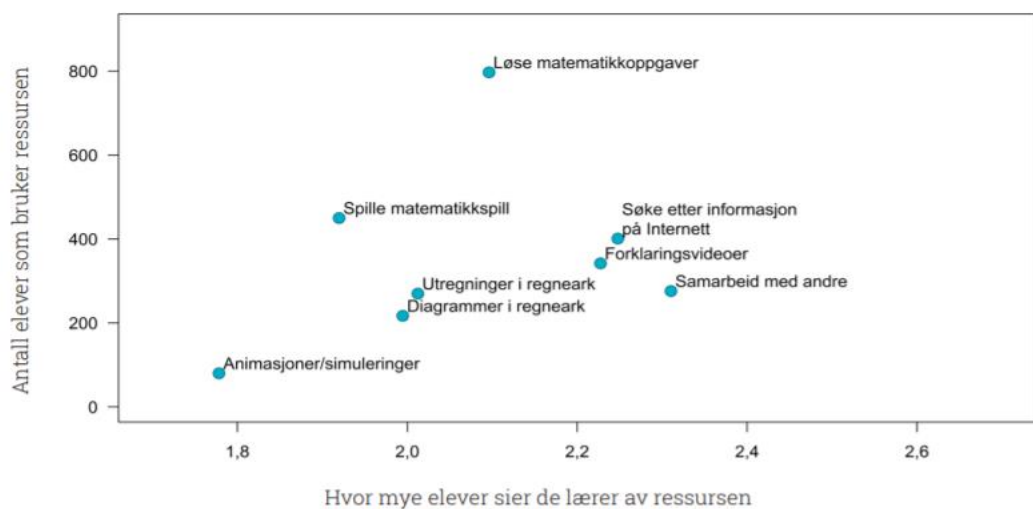
I casene som omhandlet matematikk og den digitale ferdigheten gjennomgår elevene arbeid på digitale plattformer. ARK&PAPP rapporterer om at spillene elevene gjennomfører i svært liten grad fører til bedre forståelse. Spillene ser ikke ut til å åpne opp for at elevene kan tenke selv og reflektere over hva de gjør. Utformingen av spillene, og dets mulighet for muntlig interaksjon finner rapporten seg som avgjørende for spillets relevans. Dette poengterer rapporten ved å fremheve muntlighet som en viktig del i helklasseundervisningen for å skape refleksjon (Rødnes & Gilje, 2016, s. 37).

Monitor er en kartlegging av den digitale tilstanden i skolen og er i større grad enn de andre basert på forskning rundt digitale verktøy. Rapporten omfatter ikke bare bruken av digitale verktøy i skolen, men også lærerne og elevenes digitale kompetanse, samt skolelederens digitale prioritering. Monitor (2016) legger vekt på digital teknologi i matematikkfaget og belyses gjennom en kvantitativ studie blant lærere, skoleledere og elever på syvende trinn. Rapporten fra 2016 rapporterer at lærere flest bruker nettressurser som er tilknyttet læreverkene som finnes på skolene. Spesielt for Monitor 2016 var dens fokus på IKT i matematikkfaget. I undersøkelsen av digitale ferdigheter til lærere og elever, var 8 av de 15 spørsmålene knyttet til matematikkfaget. Det gjennomsnittlige resultatet på 39,2% for elevene på matematikk spørsmålene, uttrykker en kompetanse i digitale ferdigheter på under middels nivå (Egeberg et al. 2017, s. 94). Rapporten uttrykker at bruken av digital teknologi forekommer sjeldnere i matematikkfaget enn andre fag. Av IKT-aktivitetene som forekommer i undervisningsforløpet har elevene svart på opplevd læringsutbytte i figur 7.7, hentet fra Monitor 2016.



Figur 1: *Opplevd læringsutbytte av aktiviteter med IKT. Tall i prosent.* (Egeberg et al. 2017, s. 98).

Av figur 7.7 ser vi at aktiviteter med andre og utforsking, scorer høyt på opplevd læringsutbytte av aktiviteter med IKT. I tillegg kommer det å løse matematikkoppgaver greit ut i undersøkelsene, noe som er uttrykt i figur 7.8 hentet fra Monitor 2016.



Figur 2: *Aktiviteter med IKT og opplevd læringsutbytte.* (Egeberg et al. 2017, s. 99).

Figuren viser sammenheng mellom antall elever som utfører aktiviteten og opplevd læringsutbytte der verdien 3 er «svært mye», 2 er «ganske mye», mens 1 er «lite». Figur 7.8 kan være en indikasjon på hvilke aspekter ved IKT som må forbedres, eller tilpasses slik at elevene opplever større læringsutbytte. Samarbeid med andre elever i IKT-aktiviteter i matematikkfaget viser å gi elevene størst oppfatning av oppnådd læringsutbytte (Egeberg et al. 2017, s. 99).

Oppsummert for kapittel 2, kan vi tenke oss at problemstillingens utdannings kontekst preges av elementer av usikkerhet og svak implementering av læreplanens formål (LK06). Nasjonale tolkninger og implementering av internasjonal forskning (DeSeCo) med mangel på kommunikasjon og formidling, resulterte i ulik praksis rundt grunnleggende ferdigheter. Dette påpekes blant annet av NOU (2013: 2), SMUL og FIRE. Følgeforskningen var klar over at det var bevissthet rundt grunnleggende ferdigheter, særlig digitale ferdigheter, uten at det hadde endrende virkning på undervisningen. Monitor (2016) har derimot undersøkt tilstanden i skolen etter at kommunikasjonen er forbedret, og digitale verktøy har fått en større forståelse i grunnskolen. Funnene i Monitor (2017, s. 99) viser at elevene opplever mest læringsutbytte av å samarbeide med andre. Dette trekker også ARK&PAPP frem når de skriver at det er avgjørende at de digitale aktivitetene oppfordrer til muntlig interaksjon (Rødnes & Gilje, 2016, s. 37). Ettersom programmering er blitt innført under den digitale ferdigheten og som kompetansemål vil det være interessant å se hvordan det uttrykkes i skolen gjennom læremidler.

3 Teori

I kapittel 3, vil jeg redegjøre for masteroppgavens teoretiske grunnlag. Først vil jeg kort introdusere konstruktivismens syn på kunnskap ved Piaget. For å supplere konstruktivismens syn på kunnskap vil jeg nevne Skemp (1976) sin fremhevelse av instrumentell og relasjonell forståelse. Dette gjøres for å sette kognitive krav inn i en kontekst og det overordnede læringsperspektivet problemstillingen vil drøftes ut i fra. Videre vil jeg redegjøre for kognitive krav slik Charalambous et al. (2010) har brukt begrepet i sin undersøkelse av «Addition and subtraction of fractions in three countries» (Charalambous et al. 2010). For å øke oppgavens reliabilitet vil jeg eksemplifisere med oppgaver som uttrykker ulike nivåer av kognitive krav presentert ved Matematikksenteret (Valenta, 2016). Dette gjøres for å få en forståelse av hva som kjennetegner de ulike kategoriene slik at det dannes et grunnlag for forståelse av kognitive krav før analysen. For å avslutte teorikapittelet vil jeg redegjøre for, og se temaene algoritmisk tenkning og problemløsning, i lys av dybdelæring slik det presenteres i forskningen på feltet. Teorien som er presentert under skal belyse den innledningsvise problemstillingen:

Hvilke kognitive krav stilles elevene ovenfor i møte med programmering digitalt og i analogt læremiddel produsert etter kunnskapsløftet 2020?

3.1 Piaget og Skemp

Piagets syn på kunnskap

Kunnskap kan forstås og kategoriseres på ulikt vis, alt etter hvilken læringsteori man ser kunnskap ut ifra. Ifølge konstruktivismen ved Piaget, skilles det mellom to typer kunnskap. Den første typen kunnskap kalles *figurativ kunnskap*. Dette er kunnskap basert på fysisk læring og relateres ofte til den tradisjonelle pugge-skolen. Slik kunnskap kommer av repeterende øvelser og lagres i hukommelsessenteret. Figurativ kunnskap verken har eller trenger en logisk forankring for å kunne anvendes (Imsen, 2015, s. 154). Den andre formen for kunnskap er ifølge Piaget og konstruktivismen *operativ kunnskap*. Slik type kunnskap kommer av det Piaget kaller *logisk-matematisk læring* (LM-læring). Imsen skriver: «LM-læring framkommer som resultat av assimilasjon og akkomodasjon...» (Imsen, 2015, s. 154). Begrepene *assimilasjon* og *akkomodasjon* er innenfor konstruktivismen knyttet til *Kognitive skjema*. Kognitive

skjema er indre rammer for tenkning. Dette innebærer at mennesker gjennom tankeprosesser utvikler ulike typer av indre referanserammer om hvordan et fenomen fungerer. Samspeilet mellom disse skjemaene og strukturen av dem er avgjørende for hvilket nivå av tenkning som kan bedrives (Imsen, 2015, s. 150-151). Prosesen ved å utvikle de kognitive skjemaene kjennetegnes av de to begrepene, *assimilasjon* og *akkomodasjon*. *Assimilasjon* vil si å tilpasse ny informasjon med allerede eksisterende skjemaer. Dersom den nye informasjonen passer inn i eksisterende skjema, vil skjemaet utvides ytterligere. Er det derimot strid mellom ny informasjon og allerede eksisterende skjemaer vil det skje en omstrukturering. Når individet erstatter etablert kunnskap i skjemaer til fordel for ny kunnskap, har det skjedd en *akkomodasjon* (Imsen, 2015, s. 153-154). Gjennom slike tolkningsprosesser utvikles det kunnskaper basert på objektets handling fremfor dens representasjon og fysiske egenskaper. Hovedfordelen ved operativ kunnskap, slik Imsen (2015) presenterer det, er at den ikke begrenses til enkelte temaer og skiller dem ved å bruke ordene «pugging» og forståelse. Her menes det at operative kunnskap går utover den fysiske representasjonen, mens figurativ kunnskap er låst til objektets fremstilling og menneskets sanseevner.

Skemps instrumentelle og relasjonelle forståelse i matematikk

Når kunnskap skal formidles til elever er det ulike fremgangsmåter som kan tas i bruk. Fremgangsmåten som tas i bruk er avgjørende for hvilke forståelse elevene får for temaet. I Skemp (1976) sitt verk; *Relational understanding and instrumental understanding*, blir det omtalt to typer av forståelse elever kan utvikle i matematikkundervisning. De to typene av forståelse om et matematisk emne er; «Relational understanding» (oversatt: relasjonell forståelse) og «Instrumental understanding» (oversatt: instrumentell forståelse). Skemp eksemplifiserer forskjellen mellom de to begrepene i artikkelen ved å antyde at den ene i større grad fokuserer på forståelse, og den andre på pugging av symbol uten dypere tenkning av symbolets mening. Oppgaven ser slik ut:

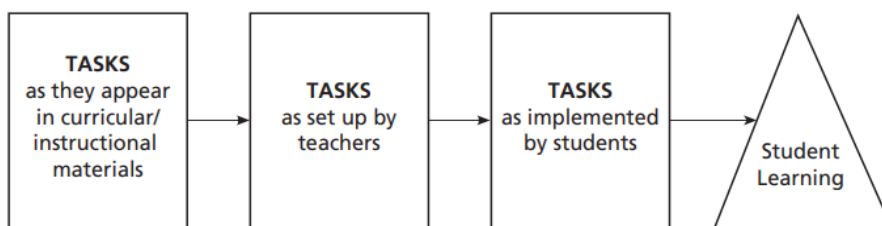
«Suppose that a teacher reminds a class that the area of a rectangle is given by $A=L*B$. A pupil who has been away says he does not understand, so the teacher gives him an explanation along these lines. “The formula tells you that to get the area of a rectangle, you multiply the length by the breadth.” “Oh, I see,” says the child, and gets on with the exercise.» (Skemp, 1976, s. 20).

Eksemplet viser hvordan eleven tilnærmer seg kunnskap om geometri i planet ved areal av et rektangel. På veien mot kunnskap oppfordrer lærer til det Skemp vil beskrive som en instrumentell forståelse av det matematiske temaet. Ved å fortelle elevene at formelen for areal til et rektangel er $A = \text{Lengde} * \text{Bredde}$, vil elevene kunne få en type forståelse av areal til det gitte rektangelet. Den instrumentelle forståelsen vil hjelpe eleven i møte med rektangler videre i undervisningen, men skaper ikke en dypere forståelse av formelen i seg selv (Skemp, 1976). Motsetningen vil være en forståelse som promoterer prosess fremfor resultat. Wæge & Nosrati (2015) refererer til Skemp i sin artikkel og skriver at; «Relasjonell forståelse innebærer å bygge opp begrepsmessige strukturer og se sammenhenger mellom begrepene.» (Wæge & Nosrati, 2015, s. 4). Ser vi Skemps tidligere eksempel om areal til et rektangel i lys av en relasjonell forståelse vil denne kjennetegnes av å forstå *hvordan* og *hvorfor* det er slikt. Ved å se hvordan fremgangsmåten ($A=L*b$) står i forhold til teori om areal, vil eleven kunne utvikle en forståelse for hvorfor det er slikt. Wæge & Nosrati (2015) trekker linjer fra Skemp forståelse av begrepene til Hiebert og Lefevre (1986) *prosedyrekunnskap* og *begrepsmessig kunnskap*. Det trekkes også linjer mellom instrumentell forståelse og tradisjonell undervisningsformer, samt relasjonell forståelse og utforskende undervisning (Wæge & Nosrati, 2015, s. 4).

3.2 Kognitive krav

Tar vi utgangspunkt i Imsen (2015) og Piagets teori om assimilasjon og akkomodasjon, foregår det tenking hos individet som oppfatter verden. Måten et individ tenker er avgjørende for typen kunnskap og skjemaer som konstrueres. Skemp (1976) prater om to typer forståelser med ulik grad av tenkning, henholdsvis instrumentell og relasjonell forståelse. Hvilken type forståelse individet utvikler avhenger av hvilke problemer og utfordringer individet blir utsatt for. Her kommer Stein et al. (1998) og hans inndeling i ulike nivåer av tenkning.

Ifølge Stein et al. (1998) vil ulike utfordringer og problemer kreve ulike nivåer av tenkning for å kunne bli løst. Hvilket nivå av tenkning som kreves avhenger av hvilken fase oppgaven presenteres i. «The mathematical tasks framework», hentet fra Stein et al. (1998, s. 11), inneholder tre faser en oppgave befinner seg i:



Figur 3: *The mathematical tasks framework* (Stein et al., 1998, s. 11).

Rammeverket fra Stein et al. (1998) inneholder tre faser for hvordan oppgaver presenteres for elevene. Den første fasen omfatter hvordan oppgaver fremstår i læreverk. Den andre fasen handler om hvordan oppgavene fremstilles av læreren. Den siste og tredje fasen går gjennom hvordan elever implementerer og gjennomfører oppgavene. De tre fasene vil være avgjørende for elevens læring og i hvilken grad av tenkning oppgavene stiller. Stein et al. (1998) poengterer at oppgaver som er ment å bli stilt høye krav til tenkning kunne bli redusert fra en fase til en annen (Stein et al., 1998, s. 11). Etersom at masteroppgaven er en innholdsanalyse av læreverk, er det kun den første fasen det er mulig å si noe om.

Stein et al. (2000) har delt inn kognitive krav i fire forskjellige nivåer, ettersom hvordan oppgaver kan fremstå i de ulike fasene av *The mathematical tasks framework* (1998), nevnt ovenfor i figur 3. Nivåene er delt inn med stigende krav til kognisjon der nivå fire er høyest (med egen oversettelse);

1. Memorization Tasks (Memorisering)
2. Procedures Without Connections Tasks (Prosedyrer uten sammenheng)
3. Procedures With Connections Tasks (Prosedyrer med sammenheng)
4. Doing Mathematics Tasks (Å gjøre matematikk)

(Stein et al., 2000, s. 16).

For å beskrive de ulike nivåene av oppgaver som stiller kognitive krav, velger jeg å utdype med eksempeloppgaver fra Valenta (2016). *Kognitive krav i matematikkundervisningen* (Valenta, 2016) tar utgangspunkt i Stein et al. (2000) sitt nivå av differensiering innenfor kognitive krav i matematikkoppgaver. Oppgavene er hentet fra artikkelen til Valenta (2016);

Memorering

Hvor mange cm er det i en meter? Dette er et spørsmål som handler om fakta, og hensikten med spørsmålet er gjerne reproduksjon og memorering.

Figur 4: *Oppgaver med lave kognitive krav.* (Valenta, 2016, s. 3).

Av figur 4 er det tydelig at spørsmålet kun ønsker et svar, eller en faktaopplysning. Sammenhengen og prosessen bak overgangen mellom måleenheter er ikke påkrevd, og de blir heller ikke utfordret i oppgaven. Dersom en elev får presentert at man multipliserer med 100 for hvert steg man tar mellom måleenhetene: $1 \text{ km} = 100 \text{ m} = 10\,000 \text{ cm}$, vil eleven memorisere denne huskeregelen. Hvorfor huskeregelen er som den er, vil da være et vanskeligere spørsmål for eleven å svare på. Oppgaver som oppfordrer til reproduksjon av slike regler med hensikt å memorere, befinner seg på det laveste kognitive nivået (Valenta, 2016, s. 3).

Prosedyrer uten sammenheng

Ved å ta et steg videre fra memorering, finner vi *prosedyrer uten sammenheng*. Slike prosedyrer medfølger ofte algoritmer eller andre ferdige fremgangsmåter, uten at prosedyren er knyttet til en større sammenheng. Under blir det vist til et klassisk eksempel på addisjonsalgoritme, som er bygget på posisjonssystemet;

Regn ut.

		5	7
+		3	8
=			

	1	5	3
+	2	6	2
=			

		3	2	5
+		1	3	5
=				

	1	4	6
+	2	2	6
=			

	2	7	1
+	2	2	5
=			

		2	6	3
+		1	4	5
=				

Figur 5: *Kognitive krav i matematikkoppgaver* (Valenta, 2016, s. 4).

En standard addisjonsalgoritme som blir presentert i figur 5, viser hvordan prosedyrer kan tas i bruk uten at de er koblet til en sammenheng. Tallene er ikke koblet mot en større forståelse av posisjonssystemet. Målet med oppgavene er å øve på en bestemt algoritme, uten at det kreves forklaring eller begrunnelse for dens gyldighet. Alt elevene

trenger for å løse oppgaven er å kunne bruke prosedyren i praksis. Det andre nivået kjennetegnes derfor også for å stille lave kognitive krav til eleven (Valenta, 2016, s. 4).

Prosedyrer med sammenheng

Går man over til oppgaver med høye kognitive krav, har vi kategorien *prosedyrer med sammenheng*. I motsetning til prosedyrer uten sammenheng har prosedyrer med sammenheng et fokus på å utvikle en dypere forståelse av prosedyrer og begreper. Gjennom ulik presentasjon av begreper og prosedyrer er søkelyset tatt bort fra den spesifikke algoritmen og flyttet til selve prosedyren. Det er her viktigere å forstå hvordan og hvorfor, enn å finne et riktig svar. En begrepsmessig forståelse er målet ved oppgaver i denne kategorien av kognitive krav (Valenta, 2016, s. 5).

Hvis vi tenker på $3 \cdot 17$ som 3 bunker med 17 klosser i hver bunke, hvordan kan vi da forklare at $3 \cdot 17 = 3 \cdot 10 + 3 \cdot 7$?

Figur 6: *Kognitive krav i matematikkoppgaver* (Valenta, 2016, s. 5).

Over ser vi eksempel på en oppgave om multiplikasjon. Gjennom regnefortelling og symbol setter oppgaven ord på matematikken. Regnefortellingen forklarer med ord hvordan vi kan tenke i møte med multiplikasjon og hva symbolene faktisk uttrykker. Ut i fra prosedyren som er beskrevet i regnefortellingen kan elevene resonnerer seg frem til svaret og dermed utvikle en egnet strategi. Oppgaven spør ikke etter et enkelt svar, men en forklaring bygget på kunnskaper om multiplikasjon og addisjon.

Å gjøre matematikk

Til slutt har vi kategorien *å gjøre matematikk* presentert av Stein et al. (2000). Valenta (2016) beskriver denne kategorien av høyt kognitivt krav slik: «Matematisk tenking innebærer utforskning, systematisering, utvikling av strategier og resonnering» (Valenta, 2016, s. 6). Valenta (2016) utdypere videre med oppgaven under:

Hvor mange personer er det plass til i gymsalen på vår skole?
Elevene skal selv sette opp noen rammer som de finner rimelige. Hva skal personene gjøre i gymsalen og hvor mye plass må hver person ha? Hvilken informasjon trenger man da og hvordan kan man få fatt i den? Hvordan gå frem videre?

Figur 7: *Kognitive krav i matematikkoppgaver* (Valenta, 2016, s. 7).

Oppgaven i figur 7 er fokusert rundt prosessen uten å antyde en riktig løsning på problemet. Ved å sette rammene for oppgaven, blir det elevenes jobb å analysere og identifisere matematikken i oppgaven. Ved å bruke tidligere kunnskaper om geometri og andre matematiske operasjoner kan oppgaven løses på ulike måter, med ulike begrunnelser. Refleksjon rundt elevenes valg og matematiske forståelse er i sentrum ettersom oppgaven spør om begrunnelse for dette (Valenta, 2016, s. 7).

3.3 Dybdelæring

Begreper som *utforskning*, *systematisk*, *strategier* og *resonnering* slik Valenta (2016) uttrykker om oppgaver med høye kognitive krav, finner vi i matematikkfagets kjerneelementer. I matematikkfagets kjerneelement; *Utforskning og problemløsning*, finner ser vi at: «Utforskning i matematikk handler om at elevene leter etter mønstre, finner sammenhenger og diskuterer seg fram til en felles forståelse,» og at «Elevene skal legge mer vekt på strategiene og framgangsmåtene enn på løsningene» (Kunnskapsdepartementet, 2019, s. 2). Gjennom diskusjon skal elevene resonnerer seg fram til en felles forståelse av det matematiske objektet. Elevene skal ifølge kjerneelementet *Resonerer og argumentasjon*; «...forstå at matematiske regler og resultater ikke er tilfeldige, men har klare begrunnelser» (Kunnskapsdepartementet, 2019, s. 3). Gjennom utforskning og eksponering av ulike fremgangsmåter med diskusjon som verktøy, vil elevene kunne se sammenhenger i faget.

Dette samsvarer med beskrivelsen Ludvigsen-utvalget (2015) presenterte om hva dybdelæring handler om. Utvalget skriver blant annet at: «Å lære noe grundig og ikke overfladisk krever en aktiv involvering fra elevenes side...» og «lærerens arbeid med å fremme dybdelæring forutsetter varierte arbeidsformer» (NOU, 2015: 8, s. 11). Dette omfatter åpne oppgaver der elevene får arbeidet variert, reflektert over læringen, samt diskutert og konkludert seg fram mot en felles forståelse vil fremme en dypere læring (NOU, 2015: 8, s. 14). Udir definerer dybdelæring som: «...å gradvis utvikle kunnskap og varig forståelse av begreper, metoder og sammenhenger i fag og mellom fagområder» (Utdanningsdirektoratet, 2019, s. 1).

3.4 Problemløsning for programmering

Programmering er innført som en del av grunnleggende ferdigheter i LK20, som videre er integrert i kompetansemålene for matematikkfaget. I de grunnleggende ferdighetene står programmering oppført som et av mange digitale verktøy, der formålet for utforskning og å løse matematiske problemer står høyt.

«Digitale ferdigheter i matematikk innebærer å kunne bruke graftegner, regneark, CAS, dynamisk geometriprogram og programmering til å utforske og løse matematiske problemer.» (Udir – Digitale ferdigheter).

Å kunne bruke et digitalt verktøy/programmering til å utforske matematiske problemer er en form for problemløsningsmetode. Sevik mfl. sier at «Å lære koding innebærer å lære en systematisk tilnærming til problemløsning. Koding innebærer mye prøving og feiling og systematisk feilsøking (de-bugging).» (Sevik mfl., 2016, s.11 – programmering i matematikkundervisningen). Begrepet «koding» blir ifølge Kaufmann mfl. forbundet i skoleverket med å lage mindre enkle programmer. (Kaufmann mfl., 2021, s. 75 – programmering i matematikkundervisningen). En kode kan anses å være instruksjonene som gis for å kunne utføre diverse oppgaver på en datamaskin. Prosessen ved å identifisere problemer og finne løsninger samt feilsøke løsningene er det tidligere Senter for IKT i Utdanningen forstår begrepet «Programmering» ved (Senter for IKT i utdanningen, Programmering i skolen, s. 9). I kompetansemålene for 8., og 9. trinn i matematikk er programmering beskrevet slik;

Elevene skal kunne:

- utforske hvordan algoritmer kan skapes, testes og forbedres ved hjelp av programmering
- simulere utfall i tilfeldige forsøk og beregne sannsynligheten for at noe skal inntreffe, ved å bruke programmering

(Udir, MAT01-05, Kompetansemål – matematikk)

Kompetansemålene gjenspeiler den grunnleggende digitale ferdigheten ved å bruke programmering på en utforskende måte i matematikkfaget, og til å løse spesifikke problemer ved hjelp av programmering.

Strategier og prosesser

Et sentralt spørsmål i masteroppgaven vil således være å definere hvordan dybdelæring og problemløsning ser ut i programmeringsoppgaver. I Udir sin definisjon av problemløsning begrepet finner vi ut at; «Problemløsning i matematikk handler om at

elevene utvikler en metode for å løse et problem de ikke kjenner fra før» (Kunnskapsdepartementet, 2019, s. 2). I møte med matematiske problemer vil dette si at elevene tar i bruk hensiktsmessige strategier og prosesser for å løse problemer. Dette poengteres av blant annet Sevik et al. (2016) som sier at programmering også «...handler om å abstrahere løsningen slik at den kan brukes til å løse lignende problemer i fremtiden...» (Sevik mfl., 2016, s.13). Problemløsning for programmering vil si å lage et sett med strategier og prosesser som kan anvendes som metode for å løse lignende problemer. Slik tenkning rundt problemløsning i programmering kalles ofte *Computational thinking*, oversatt *Algoritmisk tenkning*.

3.4.1 Algoritmisk tenkning

«Algoritmisk tenkning» er en problemløsningsmetode. Algoritmisk tenkning innebærer å tilnærme seg problemer på en systematisk måte, både når vi formulerer hva det er vi ønsker å løse, og når vi foreslår mulige løsninger.» (Utdanningsdirektoratet, 2019, s. 1). Slik begrepet *algoritmisk tenking* er uttrykt av Udir, ser vi at det legges vekt på systematikk i tilnærmingen til, og formuleringen av mulige løsninger. Kaufmann et al. (2018) deler begrepet algoritmisk tenking inn i 4 deler:

1. Dekomposisjon

- Handler om å ta problemet og dele det inn i flere biter slik at problemet lar seg enklere løse ved å løse de forskjellige bitene. Strategien ved å dekomponere problemet og løse mindre biter av det kan være med på å sette fokuset på prosessen fremfor løsningen.

2. Pattern recognition

- «Pattern recognition» eller «mønstergjenkjenning» (egen oversettelse), handler om å kjenne igjen matematiske mønstre som er innlært fra før. Elevene går gjennom grunnleggende faglig kompetanse som dem skal kunne, og kunne bruke i matematikkfaget. For å løse komplekse problemer er det nødvendig at elevene gjenkjenner mønstre slik at dem kan anvende den matematikken de har lært.

3. Å skape algoritmer

- Handler om å lage fremgangsmåter til problemet. Gjennom dekomposisjon av problemet i biter, bruke tidligere kunnskaper og gjenkjenne mønstre kan fremgangsmåter lages.

4. Automatisering og generalisering

- Å generalisere løsninger slik at de kan brukes til å løse andre matematiske problem. Her kreves det abstrakt resonnering og at man kan se struktur og regelmessigheter.

(Kaufmann et al., 2018, s. 80).

3.5 Forskning – kognitive krav i læremidler

Temaet det forskes på i denne masteroppgaven er kognitive krav i forhold til det matematiske emnet programmering. Denne sammenhengen er forsket på før av blant annet Tangen (2017), som undersøker hvordan kognitive krav i arbeid med digitale matematikkoppgaver utvikler seg i løpet av tre undervisningsøkter. Her henvises det til tidligere forskning, som omtaler at de kognitive kravene ofte blir redusert i betydelig grad, gjennom undervisningsforløpet. Selv konkluderer Tangen (2017, s. 65) med at digitale matematikkoppgaver som ikke brukes på riktig måte, er med på å redusere tilgangen til de kognitive kravene, som ligger på et høyere kognitivt nivå.

En annen studie av Steen (2019) tar for seg kognitive krav og forståelse av funksjoner. I denne oppgaven ses det på et annet matematisk emnet, der fokuset er å finne ut hvilke kognitive nivå, oppgavene ligger på. Her trekker Steen (2019) frem de to typiske nivåene, *prosedyrer uten sammenheng* og *prosedyrer med sammenheng*, som var betydelig representert i datamaterialet.

I studien av Hofstad & Liland (2021) viser det seg, at det er det laveste nivået av kognitive krav som hovedsakelig stilles elevene i den ene av de analyserte matematikkbøkene. Algebra oppgavene i denne studien viser seg å være representert i de tre første nivåene av kognitive krav. Det eneste som utmerker seg er overvekten av memorering for det ene læreverket. Dette medfølger en betydelig større andel av oppgaver med lave kognitive krav, enn det er oppgaver med høye kognitive krav i det ene læreverket. Gjenstående læreverk viser seg å ha en 45/55 fordeling mellom høy-lav kognitive krav.

4 Metode

For å kunne belyse problemstillingen har jeg valgt å bruke metoden kvalitativ innholdsanalyse. Thagaard (2019) presenterer to sentrale elementer i kvalitativ forskning; *innlevelse* i temaet som undersøkes, og *systematikk* (Thagaards, 2019, s. 14). I kapittel 4 vil jeg systematisk presentere ulike valg som er tatt i forskningsprosessen og dermed begrunne disse. Dette er valg om masteroppgavens utvalg av datamateriell, valg av metode, valg av emne, valg knyttet til oppgavens gyldighet og pålitelighet. Valgene som er tatt skal på best mulig måte belyse problemstillingen presentert i kapittel 2, gjentatt under:

I hvilken grad utfordres elevene kognitivt i møte med digitale og analoge programmeringsoppgaver i læreverker produsert for fagfornyelsen?

4.1 Utvalg

Oppgavens problemstilling krever at det analyseres et digitalt og analogt læreverker i matematikk oppdatert for LK20. Den første vurderingen av læreverker gikk ut på hvilke læreverker som er blitt brukt i løpet av lærerutdanningens praksis (2017-2021). Basert på egne erfaringer fra praksis, er det hovedsakelig Gyldendals *Maximum*, Cappelen Damms *Faktor* og Undervisningsforlaget *Grunntall*, som har blitt brukt. I en tidligere masteroppgave av Heimstad & Strand (2018) ble det undersøkt for salgstall til henholdsvis *Faktor* og *Maximum*. I undersøkelsen sier Cappelen Damm at deres læreverker *Faktor* og Gyldendals *Maximum* er de største forlagene (Heimstad & Strand, 2018, s. 33). Dette utelukker Undervisningsforlagets *Grunntall* fra utvalget av læremidler.

Videre er det et krav i problemstillingen at det undersøkes for læreverker oppdatert for fagfornyelsen. Begge Cappelen Damm og Gyldendal har produsert ferdige læreverker for LK20. Til det analoge aspektet ved problemstillingen kvalifiserer begge bøkene seg, men for den digitale siden er det annerledes. Cappelen Damm (2022) skriver på sine sider om sin digitale ressurs; «Her kommer det også et eget hefte om programmering» (Cappelen Damm, 2022). Den digitale lærerressursen fremstår mer som en ressurs, enn et digitalt læreverker i matematikk. Av denne grunn vil ikke Cappelen Damm Faktor være kvalifisert for å svare på oppgavens digitale aspekt. Sammenlignet med Gyldendals

Maximum, tilbyr Gyldendal *Maximum Smart Øving*. I Maximum Smart Øving tilbyr Gyldendal et læremiddel for 8-10. som: «gir hver elev et personlig læringsforløp og gir deg som lærer en kontinuerlig oversikt over elevens mestring.» (Gyldendal, 2022). I forhold til Faktor tilbyr Gyldendal et digitalt læremiddel som i større grad fremstår som et læreverk i matematikk. Videre ble forlagene undersøkt for om de hadde tilgjengelig vurderingseksemplarer på deres nettsider. Cappelen Damms vurderingseksemplar lå bak kommunal feide bruker, men Gyldendals vurderingseksemplar var tilgjengelig på deres nettsider. Av denne grunn, og større praksiserfaring med læreverket, falt valget på Gyldendals Maximum 8. (Tofteberg, Tangen, Bråthe, Stedøy & Alseth, 2020), Maximum 9. (Tofteberg, Tangen, Bråthe, Stedøy & Alseth, 2021) og Maximum 10. (Tofteberg, Tangen, Bråthe & Stedøy, 2021).

Da det analoge læreverket var i boks ble det sendt forespørsel på mail om tilgang til det digitale læreverket Maximum Smart Øving og eventuelt Skolestudio, som er deres digitale plattform. Skolestudio inneholder Maximum fagrom der det blant annet finnes integrert digitalt verktøy om programmering, som er relevant for oppgavens problemstilling. Forespørsel om tilgang ble avslått og begrunnet med at det ikke var mulig for at privatpersoner å få tilgang til de digitale læremidlene. Ettersom at det heller ikke var mulig å bestille den analoge versjonen (bokform) av det digitale læremiddelet, ble Maximum Smart Øving kuttet fra masteroppgaven. Erstatningen for Gyldendals digitale læreverk ble Campus Inkrementt digitale læreverk i matematikk *Campus Matte*. Erfaringsmessig har jeg hørt mye om omvendt undervisning, og har i tillegg bruker av læremiddelet i egen skolegang. På forespørsel om tilgang fikk jeg innvilget tilgang for masteroppgavens periode for *Campus Matte 8*. (Thue, Moldeklev & Røyland, 2020), *Campus Matte 9*. (Thue, Moldeklev & Røyland, 2020) og *Campus Matte 10*. (Thue, Moldeklev & Røyland, 2021). I læreverkene tilbys det programmeringsoppgaver der programmeringsoppgavene er integrert på ulikt vis. Oppgavene om programmering kan finnes i egne kapitler, samtidig som det også er en del av andre matematiske emner. Programmering fremstår som et digitalt verktøy for å oppnå andre mål, og som også er et mål i seg selv. Campus Inkrement har adskilt programmeringsoppgaver i et kapittel for seg selv. I analysen vil hele læreverket analyseres for programmeringsoppgaver, også der programmering fremstår som et digitalt verktøy. Dette kommer av masteroppgavens mål, som er å finne ut hvilke kognitive krav programmeringsoppgavene stiller til elevene gjennom læreverkene.

Årsaken til at masteroppgaven kun foretar en analyse av to læreverker i matematikk er på grunn av oppgavens lengde og progresjon i det matematiske emnet. Det vil si at et ønske i masteroppgaven er å se utviklingen i de kognitive kravene oppgavene stiller elevene. I løpet av ungdomstrinnet lærer elevene utallige begreper og sammenhenger. Når elevene kommer på 10. trinn fremstår kompetansemålene i matematikkfaget mer utforskende, begrunnede og argumenterende. Her vil det være naturlig å tenke at programmering også har en utvikling, og stiller større kognitive krav til elevene. Med tanke på masteroppgavens omfang og lengde, ville flere læreverker og færre årstrinn gått ut over dette aspektet.

4.2 Innholdsanalyse

Til å analysere utvalget og data presentert i det analoge og digitale læreverket, er det i denne masteroppgaven tatt i bruk *innholdsanalyse*. Innholdsanalyse befinner seg under det større begrepet *tekstanalyse*. I masteroppgaven vil det hovedsakelig foretas en kvalitativ tilnærming til innholdsanalyse, med kvantitative preg. En kvalitativ tilnærming betegnes ofte som: «...fortolkende tekstanalyse» (Fauskanger & Mosvold, 2014, s. 127). Årsaken til at innholdsanalyse er valgt som metode, er dens fleksibilitet i form av kvalitativ og kvantitativ data. «Researchers regard content analysis as a flexible method for analyzing text data» (Hsieh & Shannon, 2005, s. 1277). Fleksibiliteten i innholdsanalyse er essensielt for å kunne analysere og tolke den komplekse forskningsdataen. Quality research lister opp tre forskjellige innganger til kvalitativ innholdsanalyse. Av de tre har jeg brukt *Directed content analysis* (Hsieh & Shannon, 2005, s. 1281). Gjennom en deduktiv fremgangsmåte startet jeg analysen ved å forhåndsbestemme koder hentet fra tidligere forskning på læremidler. Ved å kvantifisere oppgavene inn i forhåndsbestemte kategorier og foreta en opptelling av disse, er oppgaven kvantitativ. Likevel vil den største delen av oppgaven være kvalitativ, ettersom det kreves en tolkningsprosess for å plassere oppgavene i riktig kategori.

4.2.1 Analysens rammeverk

Analysens rammeverk er todelt mellom en *horisontal* og *vertikal* analyse. Todelingen av analysen er hentet fra Charalambous et al. (2010) og deres komparative undersøkelse av læremidler fra tre forskjellige land. Den horisontale analysen vil ta for seg struktur og vektlegging av programmeringens plass i læremidlene, mens den vertikale analysen

går i dybden på programmerings emnet ved å se på kognitive krav. Den horisontale og vertikale analysen vil utføres for begge læreverkene; *Campus Matte* og *Maximum*. Under vil jeg redegjøre for hva de ulike delene av innholdsanalysen omhandler. Dette gjøres for å begrunne valg tatt i henhold til selektering av forhåndsbestemte kategorier, hentet fra annen forskning.

4.2.1.1 Horisontal analyse

Målet med den horisontale analysen er å få en oversikt over læremidlenes bakgrunnsinformasjon, samt deres vektlegging av programmering i læreverkene for LK20. Dette gjøres for å beskrive læreverkene, samt og få en god kontekst for hvordan programmering er behandlet og vektlagt i forhold til andre temaer i matematikkfaget. En gjennomgang av læreverkenes generelle oppbygning og struktur, med fokus på programmering, vil tydeliggjøre temaets plass og konteksten problemstillingen befinner seg i. For den horisontale analysen vil Charalambous et al. (2010) og deres analyseverktøy være retningsgivende med enkelte justeringer.

I artikkelen *A Comparative Analysis of the Addition and Subtraction of Fractions in Textbooks from Three Countries* (Charalambous , Delaney , Hsu & Mesa 2010) brukes blant annet en horisontal analyse til å undersøke lærebøker fra tre forskjellige land. Et av hovedmålene med undersøkelsen av lærebøkene var å finne forskjeller og likheter ved presentasjonen av addisjon og subtraksjon i lærebøkene. Her vil jeg bruke deres utviklede tabell presentert under med egen oversettelse:

Horisontal analyse av lærebok

BAKGRUNNSINFORMASJON	HELHETLIG STRUKTUR
- Tittel	Antall leksjoner og antall sider per leksjon
- Antall bøker	Struktur av leksjoner
- Antall sider (tetthet)	Læringsmål/tema
- Profil til forfattere og rådgivende utvalg	Rekkefølge på temaer
- Utgiver og utgivelsesdato	-
- Ekstramateriale	-

Tabell 3: *Addition and subtraction of fractions in three countries* (Charalambous et al. 2010, s. 123).

Selv om den horisontale analysen ikke er på et dypere nivå kan bakgrunnsinformasjon og struktur ha noe og si om konteksten til det matematiske kompetansemålet programmering. Informasjon om forfattere, antall bøker og ekstra materiale vil være gunstig for forsker og leser. Ved å analysere læreverkene for tilleggsmateriell vil vi være sikre på at det ikke er utelatt for eksempel et eget hefte i programmering. Strukturen kan fortelle oss om programmering er plassert i et kapittel for seg selv, eller om det er innbakt i andre kompetansemål og temaer. Oppbygningen av temaer fungerer som et utgangspunkt for lærere i planleggingsfasen. Gjennom den horisontale analysen kan vi si noe om denne oppbygningen og hvordan læreverket oppfordrer til å implementere programmering i matematikkundervisningen.

Det er viktig å her presisere her at strukturen i læreverkene er ment for å beskrive det utgangspunktet lærere møter i sin planleggingsfase. Dersom læreren velger en tradisjonell matematikkundervisning der elevene «følger boka», vil strukturen kunne fortelle oss hvordan elevene møter temaet programmering i skolen. Gitt at en læreren bruker boken som utgangspunkt for undervisning, men tilpasser fremgangen til lokalt læreplanarbeid, vil ikke strukturen kunne si oss hvordan elevene møter programmering i læreverkene.

De to kategoriene *bakgrunnsinformasjon* og *helhetlig struktur* med utgangspunkt fra tabellen til Charalambous et. al. (2010), vil danne oversiktsbilde over programmeringens plass i læremidlene. Bakgrunnsinformasjon vil legge vekt på læremidlenes bakgrunn og utforming, mens kategoriens helhetlig struktur vil ta for seg integreringen av programmering. Utgangspunktet analysen av *helhetlig struktur* er LK20 sin presisering av 5. matematiske kunnskapsområder: 1. Tall og tallforståelse, 2. Algebra, 3. Funksjoner, 4. Geometri, 5. Statistikk og 6. Sannsynlighet (Kunnskapsdepartementet, 2019, s. 3). Læreverkene vil sammenlignes med utgangspunktet fra LK20, i tillegg til hverandre.

4.2.2 Vertikal analyse

Den vertikale analysen vil i motsetning til den horisontale analysen gå i dybden på det matematiske temaet programmering. Her vil jeg også hente inspirasjon fra Charalambous et al., men vil utføre analysen med Stein et al. definisjon av kognitive

krav. Charalambous et al. (2010, s. 122) deler den vertikale analysen inn i tre forskjellige deler. Den første delen *Communicated to students* (oversatt; «Formidlet til elevene»), tar for seg hvordan det matematiske innholdet blir formidlet til elevene. Den andre delen *Required of students* (oversatt: «Krevd av elever») omhandler hvilke krav læreboken stiller elevene direkte, uten tilpasninger av innholdet fra pedagog. Her er det ikke snakk om matematikkens vanskelighetsgrad, men hvilke kognitive krav læremidlet stiller elevene. Det siste området i den vertikale analysen til Charalambous et al. er *Connections* (oversatt: «Koblinger»). Koblinger handler om forbindelse mellom ulike temaer i matematikkfaget, i klasserommet og til situasjoner utenfor skolen (Charalambous et al., 2010, s. 122). Hovedkategoriene til den vertikale analysen fra Charalambous et al. er oversatt og oppsummert i tabellen under:

Vertikal analyse av lærebok

Formidlet til elevene	Krevd av elevene	Koblinger
Matematisk innhold <ul style="list-style-type: none"> ➤ Emne spesifikk konstruksjon og struktur. ➤ Definisjoner, regler, konvensjoner. ➤ Illustrasjoner-representasjoner (irrelevante, relevante for konteksten, men ikke for matematikken, støtter matematikken). 	<ul style="list-style-type: none"> ➤ Potensielle kognitive krav (memorering, prosedyrer med koblinger, prosedyrer uten koblinger, å gjøre matematikk). ➤ Type svar (bare svar, svar og en matematisk setning, forklaring, begrunnelse). 	<ul style="list-style-type: none"> ➤ Koblinger innenfor og mellom tråder. ➤ Klasserom instruksjoner → Koblinger til lærebok. ➤ Koblinger til situasjoner på utsiden av skolen.

Tabell 4: *Addition and subtraction of fractions in three countries* (Charalambous et al., 2010, s. 123).

Ettersom masteroppgaven baserer seg på en innholdsanalyse og har et formål om å kunne si noe om kognitive krav i programmeringsoppgaver, vil den andre av de tre delene i Charalambous et al. være hovedtyngden i oppgaven. Kolonnen *Krevd av elever* i tabell 4, inneholder *kognitive krav* og *type svar*. Til å undersøke kognitive krav i oppgavene vil jeg bruke Stein et al. (1998) kategorisering av de ulike kognitive nivåene. Dette ble generelt eksemplifisert i kapittel 3, men vil gjennomgå igjen for temaet programmering. Årsaken til dette er for å få en forståelse av den kvalitative

tolkningsprosessen som ligger bak kategoriseringen av oppgaver inn i de ulike kognitive nivåene.

Kognitive krav

Memorering (lave kognitive krav)

Programmeringsoppgaver med lave kognitive krav kjennetegnes slik Valenta (2016) beskrev i kapittel 3. Målet med slike oppgaver er å trene en gitt algoritme, regel eller en bestemt fremgangsmåte. I slike oppgaver utfører elevene den eksakt samme oppgaven flere ganger, uten krav til begrunnelse eller refleksjon. Oppgaver om programmering, som vil bli kategorisert her, er oppgaver som følger en tidligere gitt fremgangsmåte. Det vil si at oppgavene krever at elevene for eksempel reproduserer en fremgangsmåte fra tidligere oppgaver eller presentert i læremidlenes eksempler (Valenta, 2016, s. 3).

Oppgavene under kategorien *Memorering* kjennetegnes slik:

- Reproduserer tidligere fremgangsmåter, eller fremgangsmåter gitt i eksempler.
- Hensikten med oppgaven er memorering av en gitt algoritme, regel eller fremgangsmåte.
- Det trengs ingen strategi for å løse oppgaven.
- Oppgaven er ikke knyttet til underliggende temaer.

Prosedyrer uten sammenheng (lave kognitive krav)

Videre til det andre nivået som kjennetegnes for å stille lave kognitive krav er *prosedyrer uten sammenheng*. Slik Valenta (2016) beskriver nivået med utgangspunkt i Stein et al. nivå-differensiering, tas det her i bruk prosedyrer. Selv om fokuset ligger på prosedyren, er ikke prosedyren koblet til underliggende begreper og sammenhenger (Valenta, 2016, s. 4). Programmeringsoppgaver i denne kategorien oppfordrer til bruk av en tidligere lært prosedyre, med et mindre fokus på memorering. Oppgaver på det andre nivået av kognitive krav kjennetegnes ved at:

- Fokuset er på prosedyren – ikke på memorering.
- Prosedyren er ikke koblet til underliggende begreper og sammenhenger.
- Prosedyren er en god nok forklaring i seg selv.

Prosedyrer med sammenheng (høye kognitive krav)

Programmeringsoppgaver av det nest høyeste kognitive nivået, ifølge Stein et al. (2000) tar i bruk prosedyrer og kobler disse mot underliggende matematiske sammenhenger. På

det tredje nivået av kognitive krav oppfordrer oppgavene ved begrepsforståelse og å se sammenhenger. Prosedyren blir brukt som en inngang til en større forståelse for matematikkfaget (Valenta, 2016, s. 5). Programmeringsoppgaver på det tredje nivået av kognitive krav kjennetegnes ved at:

- Prosedyrene oppfordrer til en større matematisk forståelse.
- Prosedyren oppfordrer til arbeid med begreper og sammenhenger.
- Har ikke et ensidig fokus på prosedyren som algoritme, men forståelsen av den.

Å gjøre matematikk (høyre kognitive krav)

Det siste nivå av kognitive krav krever at elevene driver med matematisk tenkning. For å gjenta Valenta (2016); «Matematisk tenkning innebærer utforskning, systematisering, utvikling av strategier og resonnering» (Valenta, 2016, s. 6). Programmeringsoppgaver på dette nivået setter rammene for oppgaven, men det er elevenes jobb å utvikle matematikken. Her refererer jeg til underkapittel 3.4.1 – Algoritmisk tenkning.

Kaufmann et al. (2018) beskriver algoritmisk tenkning i fire punkter. De fire punktene samsvarer med Valenta sitt utsagn (2016, s. 6) og vil derfor kjennetegne oppgaver på det høyeste kognitive kravet. Høyeste nivået kjennetegnes ved at:

- Dekomposisjon – deler problemet i flere biter med fokus på ulike prosesser.
- Gjenkjenne mønstre – undersøke de ulike bitene for gjenkjennbare mønstre.
- Skape algoritme – lage fremgangsmåter som kan svarer på problemet.
- Automatisering og generalisering – videreføre fremgangsmåten til andre oppgaver. Refleksjon rundt fremgangsmåten.

(Kaufmann et al., 2018, s. 80).

Type svar

For å supplere den kognitive analysen vil jeg undersøke oppgavene for hvilke type svar oppgavene krever av elevene. Årsaken til dette er sammenhengen mellom kognitive krav og type svar oppgavene krever. Inspirasjonen er hentet fra Charalambous et al. (2010) og består av de tre kategoriene: «(1) *Provide only answer...*, (2) *explain their answer or the process they followed to get that answer, and* (3) *justify the reasonableness of the approach they pursued in solving the task or the rationality of their answer.*» (Charalambous et al., 2010, s. 129). Under forklares hva som ligger i de forskjellige svar-typene Charalambous et al. (2010) opererer med, som også vil bli brukt i denne masteroppgaven:

- (1) – Her kreves det kun et svar. For å løse oppgaven må eleven kun formet et svar, uten at det stilles krav til forklaring eller begrunnelse.
- (2) – Her må elevene forklare prosessen eller det svaret dem fikk. Oppgaven stiller spørsmål til forklaring.
- (3) – Elevene må vurdere gyldigheten av fremgangsmåten som er brukt, eller begrunne rasjonaliteten i svaret. Det vil si at svaret må begrunnes i en fornuftig fremgangsmåte som ikke er tilfeldig.

4.2.3 Oversikt av analysens rammeverk

Oppsummert under er en oversikt over elementene fra tidligere forskning omkring temaet kognitive krav, gjort av Charalambous et al. (2010) og Stein et al. (2000). Målet med elementene er å kunne belyse masteroppgavens forskningsspørsmål:

- a) Hvordan er programmeringsoppgaver implementert og vektlagt i læreverkene?
- b) Hvilke nivåer av kognitive krav stiller programmeringsoppgavene til elevene?

og problemstilling:

I hvilken grad utfordres elevene kognitivt i møte med digitale og analoge programmeringsoppgaver i læreverker produsert for fagfornyelsen.

Forskningsspørsmål (a) vil som nevnt innledningsvis undersøkes i den horisontale analysen, mens forskningsspørsmål (b) vil undersøkes i den vertikale analysen. Sammen skal forskningsspørsmålene belyse den overordnede problemstillingen i oppgaven.

Innholdsanalysens rammeverk

Analogt læreverker (Gyldendal)	Digitalt læreverker (Campus Inkrement)
<ul style="list-style-type: none"> • Maximum 8, 2. utgave • Maximum 9, 2. utgave • Maximum 10, 2. utgave 	<ul style="list-style-type: none"> • Campus Matte 8 • Campus Matte 9 • Campus Matte 10

Horisontal analyse

Bakgrunnsinformasjon	Helhetlig struktur
<ul style="list-style-type: none"> - Tittel - Antall bøker - Antall sider - Profil til forfattere og rådgivende utvalg 	<ul style="list-style-type: none"> - Antall leksjoner/antall sider for hver leksjon - Struktur av leksjoner - Læringsmål/tema - Rekkefølge på tema

- Utgiver og utgivelsesdato	
- Ekstra materiell	
Vertikal analyse	
Kognitive krav	Type svar
1. Memorering	1. Kun svar
2. Prosedyrer uten sammenheng	2. Svar med forklaring til svar eller prosess
3. Prosedyre med sammenheng	3. Begrunne fremgangens/svarets gyldighet
4. Å gjøre matematikk	

Tabell 5: *Innholdsanalysens rammeverk.*

4.2.4 Koding av oppgaver

Innledningsvis i kapittel 4 ble det nevnt at masteroppgaven baserer seg på en «Directed content analysis (Hsieh & Shannon, 2005, s. 1281). I en slik tilnærming til innholdsanalyse er kategorier definert ut ifra eksisterende teori og forskning. Det vil si at kategoriene som brukes i analysen er definert før analysen finner sted. Kategoriene beskrives og defineres før analysen, slik jeg har gjort i kapittel 3.2 og 4.2.2 for *kognitive krav* og *type svar*. For oppgaver som ikke kan kategoriseres innenfor de forhåndsbestemte kategoriene, må det analyseres og vurderes om det trengs en ny kategori (Hsieh & Shannon, 2005, s. 1281-1283). I den vertikale analysen vil dataen kodes inn i disse forhåndsbestemte teoriene for *kognitive krav* og *type svar*:

Kognitive krav	Koder
Memorering (lave kognitive krav)	M-L
Prosedyrer uten sammenheng (lave kognitive krav)	PUS-L
Prosedyrer med sammenheng (høye kognitive krav)	PMS-H
Å gjøre matematikk (høye kognitive krav)	ÅGM-H
Type svar	Koder
Kun svar	KS
Svar med forklaring	SMF
Vurdere gyldigheten av løsning (fremgangsmåte/svar)	VGL

Tabell 6: *Koder for vertikal analyse.*

4.3 Reliabilitet og validitet

Spørsmål knyttet til oppgavens reliabilitet handler om forskningen er gjennomført på en pålitelig måte. Ved pålitelig måte menes at vi forventer at andre kan trekke sammen konklusjoner dersom de bruker den samme metoden og data (Thaagard, 2018, s.187). Thaagard (2018) trekker også frem begrepet *credibility*, altså oppgavens troverdighet som en essensiell rolle i oppgavens reliabilitet. Valgene rundt oppgavens metode må presenteres slik at det ikke finnes noen tvil om hvordan forskere har samlet inn, analysert og tolket data. På denne måten kan leseren koble empiri mot forskningsfunn, samt vurdere den metodiske tilnærmingen og vurdere metodens gyldighet (Thaagard, 2018, s. 181). Et eksempel her kan være å redegjøre for kategoriseringen som er brukt i denne masteroppgaven. Ved å bruke innholdsanalyse med en deduktiv tilnærming, blir datamaterialet kodet og kategorisert ut fra tidligere teorier. En redegjørelse og beskrivelse av kategoriene, og deres overførbarhet til ny kontekst er sentralt for å poengtere deres gyldighet. I tillegg er det viktig å reflektere over data som ikke faller inn under de forhåndsbestemte kategoriene. Ved å være bevisst på dette aspektet, kan oppgaven reflektere perspektiver utledet fra kategoriene, samt data utenfor kategoriene. Thaagard (2018) skriver at slik; «...kan vi nyansere den videre analysen og ikke bli fanget av det perspektivet kategoriene fremhever» (Thaagard, 2018, s. 155).

Begrepet validitet i forhold til forskning handler om forskningens resultat og tolkning av data (Thaagard, 2018, s. 189). Silverman (2014) trekker frem begrepet «*transparens*» (Silverman, 2014, s. 84). Begrepet bunner i at man beskriver den teoretiske konteksten tolkningen gjøres ut ifra, og vurderer tolkningenes gyldighet i forhold til virkeligheten de er utledet fra. I kvalitative metoder der det inngår tolkningsprosesser, står forskerens tilknytning til miljøet sentralt. Før jeg avslutter med noen etiske refleksjoner vil jeg presentere dette forholdet mellom forsker og miljøet det skal forskes i.

Læremidler har vært sporadisk brukt gjennom praksis og vikariat i skolen. Selv har jeg brukt læremidlene i undervisningssammenheng, og samtidig blitt undervist i egen skolegang, samt vært en del av fellesskapet i skolen der læremiddel debatten foregår. Fra egen skolegang er oppfattelsen at læremidler er repetisjon og reproduksjon av kunnskap. Fra utdanningen og praksisen er den generelle oppfattelsen delt. På den ene siden er de som betrakter læremidler som god hjelp i undervisningen, mens den andre siden kobler læremidler mot den tradisjonelle skolen (pugge-skolen). Ved læremidler

menes den klassiske «læreboka». Uansett er denne tilknytningen til miljøet viktig å løfte frem og anerkjenne. Når miljøet er kjent fra før, kan det utvikles; «...en forståelse innenfra» (Thaagard, 2018, s. 190). Det vil si at erfaringer jeg selv har med skolemiljøet og læremidler danner grunnlaget for tolkningsprosessen. Thaagard (2018) skriver at; «Vi utvikler tolkningen av de fenomenene vi studerer, både i relasjon til nye kunnskaper og til egne erfaringer» (Thaagard, 2018, s. 190). Ved å reflektere over eget ståsted og se det i lys av tolkningsprosessen har leseren mulighet for å vurdere tolkningens validitet.

4.4 Ethiske refleksjoner

Ettersom masteroppgaven baserer seg på en innholdsanalyse av ulike læreverker der personopplysninger ikke inngår, vil det ikke være nødvendig å melde forskningsprosjektet til Norsk Senter for forskningsdata (NSD), eller innhente forskningsetisk samtykke (NESH, 2021).

5 Resultater

I kapittel 5 vil jeg gjøre en presentasjon av funn ved den horisontale og vertikale analysen. Drøftinger og koblinger opp mot teori og tidligere forskning, vil behandles i et eget kapittel. Ettersom valget falt på en deduktiv fremgangsmåte vil jeg eksemplifisere enkelte oppgaver innenfor de forhåndsbestemte kategoriene. Dette gjøres for å øke oppgavens reliabilitet og skape forståelse for tolkningsprosessen av ulike oppgaver inn i de forskjellige kategoriene av kognitive nivåer. Funn som ikke passet inn, eller som var vanskelig å kategorisere vil også trekkes frem for videre belysning i masteroppgavens drøftingskapittel. Først vil det være en presentasjon av funn fra den horisontale analysen, etterfulgt av funnene i den vertikale analysen. Læreverkene *Maximum* og *Campus Matte* vil behandles adskilt for å skape systematikk i teksten. Forskningsspørsmål a), vil belyses i den horisontale analysen, mens forskningsspørsmål b), belyses av den vertikale analysen. Sammen danner forskningsspørsmålene grunnlaget for å svare på den overordnede problemstillingen;

I hvilken grad utfordres elevene kognitivt i møte med digitale og analoge programmeringsoppgaver presentert i læreverker for fagfornyelsen?

Forskningsspørsmål;

- a) Hvordan er programmeringsoppgaver implementert og vektlagt i læreverkene?
- b) Hvilke nivåer av kognitive krav stiller programmeringsoppgavene til elevene?

5.1 Horisontal analyse

Den horisontale analysen fremkommer av en systematisk gjennomgang av de tre lærebøkene *Maximum 8.*, *Maximum 9.* og *Maximum 10.*, samt de tre digitale læremidlene; *Campus Matte 8.*, *Campus Matte 9.* og *Campus Matte 10.*

Gjennomgangen av de seks læremidlene er inspirert av Charalambous et. al. (2010), og de vil undersøke læreverkene for det Charalambous et al. (2010) kaller *bakgrunnsinformasjon* og *helhetlig struktur*. Nevnt tidligere ble det gjort en avgrensning i bakgrunnsinformasjon, hvor punktet om forfatterens profil og rådgivende utvalg er utelukket. Oppgaven vil ikke gå nærmere inn på forfatterens profil da dette ikke virker relevant for problemstillingen.

5.1.1 Bakgrunnsinformasjon - Maximum

Fra kapittel 4, tabell 5: *Innholdsanalysens rammeverk* er målene for bakgrunnsinformasjon listet opp. Av de opplistede punktene er tittel, antall bøker, antall sider, forfattere, ekstra materiell, utgiver og utgivelsesdato presentert under:

Tittel	Forfattere	Utgiver	Utgivelsesår	Sidetall
Maximum 8, Grunnbok	Grete Normann Tofteberg Linda Tangen Bråthe Ingvill Stedøy Bjørnar Alseth Janneke Tangen	Gyldendal	2020	293
Maximum 9, Grunnbok	Grete Normann Tofteberg Linda Tangen Bråthe Ingvill Stedøy Bjørnar Alseth Janneke Tangen	Gyldendal	2021	302
Maximum 10, Grunnbok	Grete Normann Tofteberg	Gyldendal	2021	307

Linda Tangen Bråthe Ingvill Stedøy Janneke Tangen

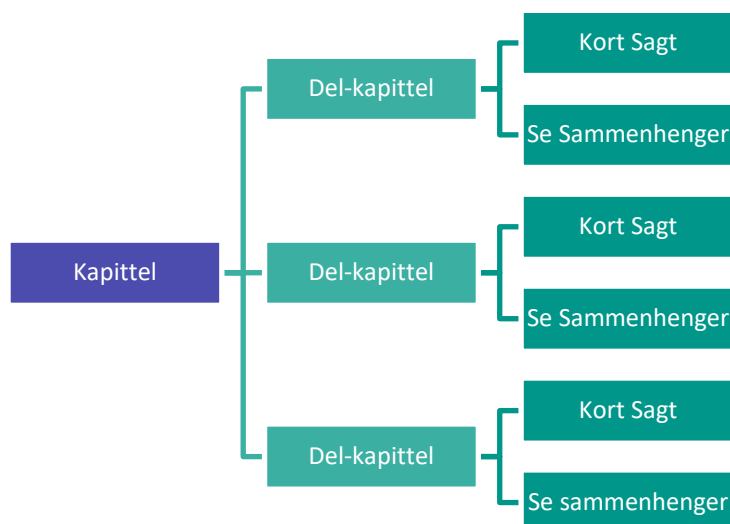
Tabell 7: *Læreverk Maximum med forfattere.*

Tabellen vist ovenfor presenterer tittel på læremidler, forfattere, utgivelsesdato, utgiver og antall sider for de tre læremidlene i læreverket *Maximum*. I tillegg til *Maximum grunnbok*, tilbyr Gyldendal også *Lærerens bok*, *oppgavebok*, *regelsamling* for ungdomstrinnet og et digitalt læremiddel, som *Maximum Smart Øving*. Dette tilleggsmaterialet vil ikke være i masteroppgavens søkelys, ettersom fokuset ligger på grunnboken de fleste elever møter i skolen. For *Maximum Smart Øving* ville det vært berikende for oppgaven om programmering også kunne bli belyst fra Gyldendal sitt digitale læremiddel i matematikk. Uheldigvis er det ingen tilgang for privatpersoner på dette læremidlet ennå, og heller ikke mulig å bestille den analoge versjonen av *Maximum Smart Øving*.

Det man derimot kan legge merke til ved bakgrunnsinformasjonen, er bruken av forfattere. Læremidlenes utforming og oppgavetyperne i bøkene er svært avhengig av forfatterens forståelse av læring. Ved å bruke de samme forfatterne kan det tenkes at det læringsperspektivet som finnes i *Maximum 8.*, også er kontinuerlig brukt gjennom *Maximum 9.*, og til *Maximum 10.*

5.1.2 Helhetlig struktur - Maximum

Ser man på den helhetlige strukturen, kan man finne en liste med punkter som blir presentert i tabell 5: nemlig antall leksjoner, antall sider for hver leksjon, struktur av leksjoner, læringsmål og rekkefølge på tema (Charalambous et al. 2010). Under er den generelle oppbygningen av lærebøkene som finnes i læremidlene *Maximum 8. -10. trinn*;



Figur 8: *Generell oppbygning av læremidlene Maximum 8, -10. Grunnbok.*

Går vi i dybden på strukturen i et «typisk» delkapittel i *Maximum*, er den generelle og gjennomgående tendensen at delkapittelet starter med å presentere ulike læringsmål. Videre går bøkene systematisk gjennom de læremålene som blir nevnt i starten av delkapittelet. Til læringsmålene er det en forklarende tekst som forklarer sentrale begreper. Dette blir gjort for å øke forståelsen rundt målene. Forklaringene blir satt til liv og eksemplifisert gjennom boksene «Slik kan du tenke» i lærebøkene. Her er det som regel flere eksempler på hvordan elevene kan tenke i arbeidet med læringsmålene. Mellom forklaringer og eksemplifisering er det nivådelte oppgaver som skal utfordre elevene på deres nivåer. Til slutt består delkapitlene av enkelte «oppdrag» og «aktivitets» sider. Figur 2, som blir vist nedenfor, viser den generelle vektleggingen og oppbyggingen av delkapitler slik den fremstår i *Maximum*.



Figur 9: *Maximum generelle oppbygning av delkapitler.*

Maximum er for grunnbøkene delt inn i ulike kapitler etter temaer som dekkes i bøkene. I *Maximum 8*. finner vi kapitlene; «Tall og tallregning, algebra, funksjoner og likninger og formler» (Tofteberg et al., 2020). Videre i *Maximum 9*. er kapitlene delt inn i;

«statistikk, sannsynlighet, plangeometri og romgeometri» (Tofteberg et al., 2021). Til slutt i *Maximum 10*. omhandler kapitlene; «likninger og algebra, funksjoner, økonomi og se flere sammenhenger» (Tofteberg et al., 2021). Slik jeg har tolket Charalambous et al. (2010), betyr «leksjoner», mindre delkapitler. For eksempel er kapittel 2: algebra, i *Maximum 8*. delt inn i mindre deler (Tofteberg et al., 2020). De mindre delene i et kapittel betegnes som delkapittel i denne undersøkelsen, mens oppgavene innenfor delkapitlene omtales som leksjoner.

Hvert av de fire kapitlene i *Maximum 8*. er delt inn i tre delkapitler, utenom kapittel fire, som består av fire delkapitler. I tillegg til hvert av kapittel inneholder delkapitlene en oppsummerende del, som blir kalt for «kort sagt», og en del for å koble sammenhenger kalt «se sammenhenger» (Tofteberg et al., 2020). Kort sagt inneholder en oppsummering læringsmålene for kapitlene, og skal ifølge Tofteberg et al. (2020) være til hjelp ved elevenes egenvurderingen. «Se sammenhenger» består ifølge Tofteberg et al. (2020) av oppgaver som går i dybden i matematikkfaget, og samtidig skal promotere sammenhenger i og utenfor matematikkfaget. I Tabell 7, 8 og 9, presenteres en oversikt over læremidlenes struktur av leksjoner, samt vektleggingen av hvert delkapittel. Det blir også presentert kapittel i form av sidetall. Kryss i tabellene 7-9, betyr at læremiddelet ikke inneholder disse delkapitlene. Tallet i tabellene oppgir antall sidetall, mens tallet i parentes angir hvor mange leksjoner delkapittelet består av.

Maximum 8 - 2.utgave

	Tall og tallregning	Algebra	Funksjoner	Likninger og formler
Delkapittel 1	18 (4)	14 (3)	8 (3)	6 (3)
Delkapittel 2	22 (6)	22 (4)	18 (4)	18 (4)
Delkapittel 3	27 (6)	11 (3)	13 (3)	12 (3)
Delkapittel 4	X	X	X	10 (3)
Delkapittel 5	X	X	X	X
Kort Sagt	5	3	7	6
Se sammenheng	10	12	12	14
Sum	82	62	58	66

Tabell 7: Struktur for *Maximum 8*. (2. utgave).

Tabell 7 viser at *Maximum 8. - 2. utgave* deler de overordnede kapitlene 3-4 opp i delkapitler. Hvert av delkapitlene er fulgt opp med en oppsummering av læringsmålene i læremidlet «kort sagt», og supplert med «se sammenhenger». *Maximum 8.* har totalt 268 sider, 4 kapitler, 13 delkapitler og 49 leksjoner. Dette utgjør i snitt 67 sider for hvert delkapittel, 3,25 delkapitler for hvert kapittel, samt 3,77 leksjoner for hvert delkapittel. Totalt tildeles hver leksjon ca. 5,46 sider. En lignende struktur finner vi også i *Maximum 9.*

Maximum 9. - 2.utgave

	Statistikk	Sannsynlighet	Plangeometri	Romgeometri
Delkapittel 1	26 (9)	12 (4)	34 (8)	14 (3)
Delkapittel 2	16 (7)	16 (5)	10 (3)	14 (2)
Delkapittel 3	9 (5)	17 (5)	16 (5)	5 (2)
Delkapittel 4	X	X	X	X
Delkapittel 5	X	X	X	X
Kort Sagt	5	5	6	5
Se sammenheng	16	14	18	14
Sum	72	64	84	52

Tabell 8: *Struktur for Maximum 9. (2. utgave).*

Maximum 9. har i likhet med *Maximum 8.*, også «kort sagt» og «se sammenhenger» i seg. Totalt inneholder *Maximum 9.* 272 sider, 4 kapitler, 12 delkapitler og 58 leksjoner. Dette utgjør i snitt 68 sider for hvert delkapittel, 3 delkapitler for hvert kapittel, samt ca. 4,83 leksjoner for hvert delkapittel. Totalt tildeles hver leksjon ca. 4,68 sider. Her er det tatt med, som for *Maximum 8.* og *Maximum 10.*, «se sammenhenger» og «kort sagt». Dette forekommer av at delene oppsummerer og supplerer leksjonene de er en del av i samme kapittel. Strukturen vi ser i *Maximum 8* og *9*, fortsetter også i *Maximum 10*, ved unntak av kapittel 4. Dette kapitlet inneholder ikke «kort sagt» eller «se sammenhenger», da hele kapitlet fokuserer på sammenhenger i matematikkfaget.

Maximum 10. - 2.utgave

	Likninger og algebra	Funksjoner	Økonomi	Se flere sammenhenger

Delkapittel 1	14 (4)	12 (4)	20 (5)	8 (1)
Delkapittel 2	14 (4)	14 (4)	12 (4)	10 (1)
Delkapittel 3	19 (6)	24 (5)	13 (3)	8 (1)
Delkapittel 4	X	9 (3)	X	24 (4)
Delkapittel 5	X	X	X	12 (1)
Kort Sagt	5	9	7	X
Se sammenheng	14	12	12	X
Sum	66	80	64	62

Tabell 9: *Struktur for Maximum 10. (2. utgave).*

Totalt inneholder *Maximum 10.* 272 sider, 4 kapitler, 15 delkapitler og 50 leksjoner.

Dette utgjør i snitt 68 sider for hvert delkapittel, 3,75 delkapitler for hvert kapittel, samt ca. 3,33 leksjoner for hvert delkapittel. Totalt tildeles hver leksjon ca. 5,44 sider.

5.1.3 Læringsmålet programmering

Interessen for masteroppgaven er blant annet å belyse programmeringen sin plass blant de andre kompetansemålene. Programmering er i LK20, uttrykt som et digitalt verktøy i den grunnleggende digitale ferdigheten (Kunnskapsdepartementet, 2019, s. 5). Dette uttrykkes også i kompetansemålene for matematikkfagets 8., 9., og 10. trinn. Ved å se på læreverkets læringsmål kan vi få en formening om temaets vektlegging. I analysen av læringsmål vil jeg foreta meg en opptelling av mål med programmering, samt mål som ikke inneholder programmering. Hvordan programmering defineres som et digitalt verktøy finnes i kapittel 1.5 – *Begrepsavklaring*. Som kompetansemål i matematikkfaget for 8. trinn er programmering fastsatt slik; «utforske hvordan algoritmer kan skapes, testes og forbedres ved hjelp av programmering» (Kunnskapsdepartementet, 2019, s. 12). Tabell 10, 11 og 12, inneholder en oversikt over totalt antall læringsmål for kapitlene i læreverket *Maximum*, samt læringsmål som inneholder programmering.

Læringsmål Maximum 8. – 2. utgave

	Antall læringsmål	Antall læringsmål med programmering
Tall og tallregning	13	0
Algebra	8	1

Funksjoner	9	1
Likninger og formler	10	1
Sum	40	3

Tabell 10: *Læringsmål Maximum 8. – 2.utgave.*

For *Maximum 8* ble det analysert totalt 40 læringsmål for programmering, derav 3 læringsmål som oppfylte kravet. For å oppfylle kravet må læringsmålet være i henhold til definisjonen om programmeringsoppgaver, som også ble nevnt tidligere. For å gjenta det, vil dette si oppgaver som inneholder det å bruke et digitalt programmerings program, forme eller tolke koder. For den totale mengden læringsmål presentert i læremiddelet, inneholder 7,5% av læringsmålene programmering i henhold til gitt definisjon. Videre i *Maximum 9*, kan vi se en lignende tendens og samme fokus på programmering i læringsmålene.

Læringsmål Maximum 9. – 2. utgave

	Antall læringsmål	Antall læringsmål med programmering
Statistikk	9	0
Sannsynlighet	9	1
Plangeometri	9	0
Romgeometri	8	0
Sum	35	1

Tabell 11: *Læringsmål Maximum 9. – 2. utgave.*

I matematikkfagets kompetansemål for 9. trinn, kan vi se at programmering innebærer det å kunne; «simulere utfall i tilfeldige forsøk og beregne sannsynligheten for at noe skal inntreffe, ved å bruke programmering» (Kunnskapsdepartementet, 2019, s. 13). Ut ifra kompetansemålet ville det være naturlig å finne læringsmål med dette innholdet i kapittel 2, sannsynligvis i *Maximum*. Her er hele 1. læringsmål tildelt programmering. Av den totale mengden på 35 læringsmål, utgjør programmering ca. 2,86%. Dette er en betydelig nedgang fra *Maximum 8*. Nedgangen fortsetter videre i *Maximum 10*.

Maximum 10. – 2. utgave

	Antall læringsmål	Antall læringsmål med programmering
Likninger og	9	0

algebra		
Funksjoner	12	0
Økonomi	9	0
Se flere sammenhenger	16	0
Sum	46	0

Tabell 12: *Læringsmål Maximum 10. – 2. utgave.*

Tar vi for oss matematikkfaget sine kompetansemål, kan vi se at programmering blir uttrykt som det å kunne; «utforske matematiske egenskaper og sammenhenger ved å bruke programmering» (Kunnskapsdepartementet, 2019, s. 14). Ut ifra kompetansemålet ville det vært naturlig at programmering ble uttrykt i kapittel 4; «Se flere sammenhenger» (Tofteberg et al., 2021). Læringsmålene i dette kapitlet oppfattes som åpne og generelle, men nevner ikke programmering. Hverken her eller i noen andre læringsmål er programmering ikke til å se. Av den totale mengden læringsmål utgjør programmering 0%.

En kort oppsummering av læringsmålene over, viser at programmering ikke uttrykkes i en betydelig grad. Selv om det i liten grad uttrykkes i læringsmålene i *Maximum*, viser en nærmere undersøkelse av det analoge læreverket at programmering brukes innad i flere forskjellige kapitler, i større grad enn det læringsmålene skulle tilsi. For å supplere undersøkelsen av læringsmål, vil jeg derfor presentere en oversikt over kapitlene, samt deres innhold av oppgaver med programmering. Oversikten under er utarbeidet ved å bruke sentrale begreper som knyttes til programmering, eller oppgaver der programmering inngår. Begrepene læreverket *Maximum* ble undersøkt for er; programmering, kod, koder, digitalt verktøy og program. Tabell 13, 14 og 15 gjenspeiler kun de grunnleggende oppgavene i boken. Her vil det ikke tas med såkalte «oppdrag» og «aktivitet» sider (Tofteberg et al., 2020). Årsaken til dette er på grunn av oppgavens størrelse. Slike oppgaver krever flere elever eller organisering fra lærer for å kunne bli gjennomført, samt fungere på en gunstig og optimal måte.

Antall oppgaver vs. antall oppgaver med programmering – Maximum 8.

	Antall oppgaver	Antall oppgaver med programmering

Kapittel 1 – Tall og tallregning	156	0
Kapittel 2 – Algebra	85	11
Kapittel 3 – Funksjoner	80	4
Kapittel 4 – Likninger og formler	99	9
Sum	420	24

Tabell 13: *Antall oppgaver vs. antall oppgaver med programmering – Maximum 8.*

Selv om programmering ikke blir uttrykt i en betydelig grad i læremiddelets læringsmål, ser vi at programmering likevel finnes under flere kapitler. Av den totale mengden på 420 oppgaver i *Maximum 8.*, omfatter 24 av dem, en form for programmering. Dette utgjør ca. 5,71% av den totale oppgave mengden. Dette finner vi også igjen i *Maximum 9*, hvor programmering er å finne i alle kapitlene.

Antall oppgaver vs. antall oppgaver med programmering – Maximum 9.

	Antall oppgaver	Antall oppgaver med programmering
Kapittel 1 – Statistikk	102	7
Kapittel 2 – Sannsynlighet	79	9
Kapittel 3 – Plangeometri	129	7
Kapittel 4 – Romgeometri	86	2
Sum	396	25

Tabell 14: *Antall oppgaver vs. antall oppgaver med programmering – Maximum 9.*

Av den totale mengden oppgaver i læremiddelet, omfatter oppgaver med programmering ca. 6,31% av disse. Dette er en minimal økning fra *Maximum 8*, som hovedsakelig skyldes færre oppgaver totalt. For *Maximum 10*, ser vi derimot en betydelig nedgang i antall oppgaver med programmering.

Antall oppgaver vs. antall oppgaver med programmering - Maximum 10.

	Antall oppgaver	Antall oppgaver med programmering
Kapittel 1 – Likninger og algebra	101	2
Kapittel 2 – Funksjoner	95	8

Kapittel 3 – Økonomi	115	1
Kapittel 4 – Se flere sammenhenger	80	4
Sum	391	15

Tabell 15: *Antall oppgaver vs. antall oppgaver med programmering – Maximum 10.*

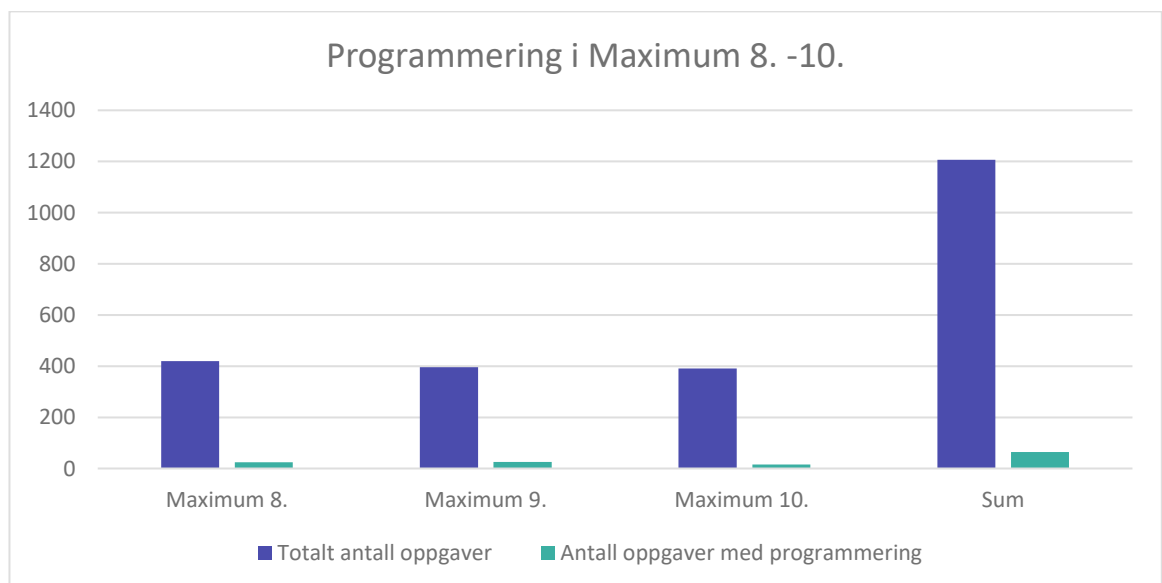
Tabell 15, sammenlignet med tabell 14 og 13, viser at det er færre oppgaver totalt sett. Dette ser man både gjelder oppgaver med programmering og oppgaver uten. Av de totalt 391 oppgavene, utgjør programmering ca. 3,83% av disse. Oppsummeringen under viser totalt antall oppgaver i læreverkene, samt totalt antall oppgaver med programmering.

Antall oppgaver med programmering – Maximum 8. -10.

	Totalt antall oppgaver	Oppgaver med programmering
Maximum 8.	420	24
Maximum 9.	396	25
Maximum 10.	391	15
Sum	1207	64

Tabell 16: *Antall oppgaver med programmering – Maximum 8. -10.*

Tabell 16, presentert i et diagram ved figur 10, viser fordelingen av antall oppgaver i forhold til oppgaver med, og oppgaver uten programmering;



Figur 10: *Programmering i Maximum 8. -10.*

Gjennomgangen av *Maximum 8.* viser til at ca. 5,71% av oppgavene i *Maximum 8.*, omhandler programmering. Det vil si at disse oppgavene spesifikt oppfordrer til bruk av programmering som løsningsmetode, eller at en blir bedt om å skrive en kode for et programmeringsverktøy. I analysen av *Maximum 9.*, finner vi en tilsvarende vektlegging av programmering med ca. 6,31%, av kapitlets oppgaver. Etter en minimal økning fra *Maximum 8.* til *Maximum 9.*, finner vi i *Maximum 10.*, en betydelig reduksjon. Her er det ca. 3,83% av alle oppgavene som berører programmering. Ut ifra analysen av læreverkets oppgaver, ser vi programmering som et tema som blir integrert i andre kapitler. Rekkefølgen på temaer vil derfor ikke bli analysert, ettersom det ikke har en hensikt for masteroppgaven.

5.1.4 Bakgrunnsinformasjon – Campus Matte

Campus Inkrement er en digital plattform som tilbyr læreverk i realfag. Campus Inkrement beskriver sine læreverk med fokus på dybdelæring og tilpasset opplæring. Læreverket *Campus Matte* for grunnskolen består av tre digitale læremidler; *Campus Matte 8*, *Campus Matte 9*, *Campus Matte 10*, samt et tilhørende kompendium til hvert av ungdomstrinnene. Kompendiet fungerer som et slags oppslagsverk og repetisjon av oppgavene som finnes digitalt. Til de digitale oppgavene er det også forklarende videoer med eksempler på oppgaver og ulike fremgangsmåter. Ettersom denne masteroppgaven kun ser på matematikkoppgavene vil tilleggsmateriellet og eksempeloppgaver ikke være i fokus. Under er en presentasjon av de digitale læremidlene med forfatter, utgiver og årstall. Antall sider under, stemmer overens med E-boken, som tilhører læreverket. Her er oppgavene ekskludert, ettersom de bare finnes på den digitale plattformen *Campus Matte*.

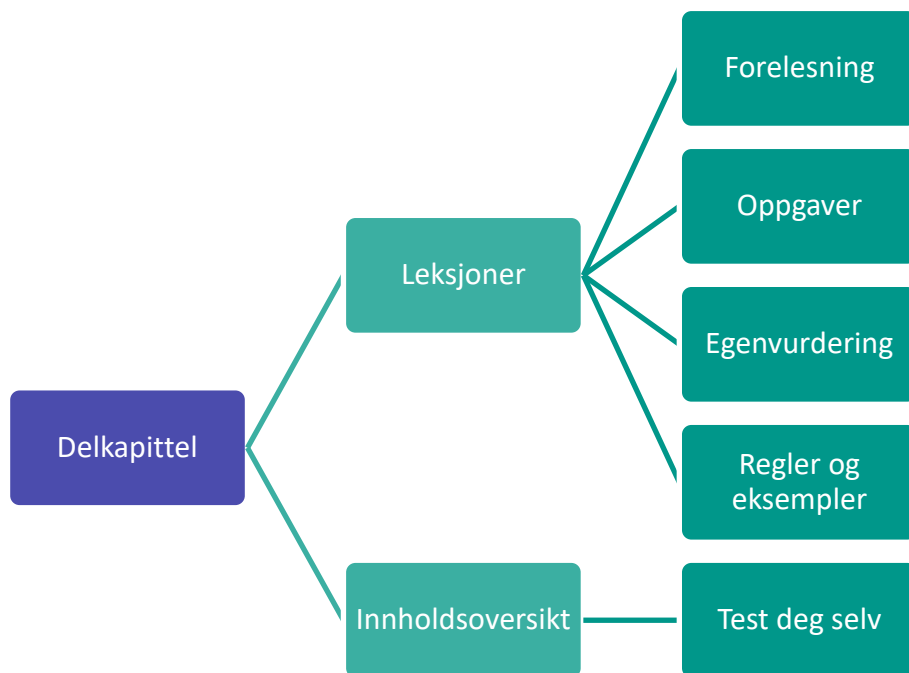
Tittel	Forfattere	Utgiver	Utgivelsesår	Sidetall
Campus Matte 8	Bjørn Ove Thue Rolf-Anders Moldeklev	Inkrement AS	August 2020	90

	Ole Thomas Røyland			
Campus Matte 9	Bjørn Ove Thue Rolf-Anders Moldeklev Ole Thomas Røyland	Inkrement AS	Mai 2020	83
Campus Matte 10	Bjørn Ove Thue Rolf-Anders Moldeklev Ole Thomas Røyland	Inkrement AS	April 2021	101

Tabell 17: Læreverket *Campus Matte* med forfattere.

5.1.5 Helhetlig struktur

Likt for læreverket *Campus Matte*, som Gyldendal sitt *Maximum*, er læreverkets forfattere gjennomgående i de tre årstrinnene. Dette kan være en indikator på at innholdet er ensidig formet etter den gjeldende forståelsen av læring. Oppbygningen til læremidlene i læreverket *Campus Matte*, kan grovt skisseres inn slik figur 11. viser til nedenfor;



Figur 11: Grov skisse av oppbygningen til Campus Matte.

Grovskissen av oppbygningen til læreverket *Campus Matte*, viser til at læremidlene er inndelt i delkapitler. Under et delkapittel kommer man først til en innholdsoversikt over kapitlet. Innholdsoversikten viser til antall oppgaver som er blitt løst, forelesninger som er gjennomført og egenvurdering. Innholdsoversikten har også en «Test deg selv»-kartleggingsprøve. Dersom man ikke velger å gå inn på innholdsoversikten i delkapitlene, kommer man direkte til leksjonene. Leksjoner er igjen delt inn i forelesninger, oppgaver, egenvurdering, regler og eksempler. Vektleggingen av de forskjellige elementene i oppbygningen av et generelt kapittel i læreverket *Campus Matte*, fremstår slik;



Figur 12: Vektlegging av forskjellige element i oppbygningen.

Oppgavene tar en betydelig del av delkapitlene, etterfulgt av korte forelesninger, som er med tidkrevende enn enkle gjennomganger av regler, eksempler og egenvurdering. Sammen utgjør figurene over den grove skissen, oppbygningen til læreverket. Ettersom *Campus Matte* er et digitalt læreverk er det ikke oppgitt antall sider. I delkapitlet 5.1.4, har jeg derfor nevnt kompendiet til læreverket *Campus Matte*. Kompendiet inkluderer eksempler og regler fra delkapitlene, men ekskluderer oppgaver, ettersom disse kun finnes digitalt. Tabell 18, 19 og 20 viser en oversikt over antall kapitler, leksjoner i kapitler, og sider i kompendiet.

Antall; kapitler, leksjoner og sider i kompendium – Campus Matte 8.

	Antall leksjoner	Antall sider i kompendium
1. Tall	1.1 – 1.8	9
2. Regning	2.1 – 2.10	11
3. Brøk	3.1 – 3.9	9
4. Potenser	4.1 – 4.9	8
5. Likninger og algebra	5.1 – 5.9	6
6. Funksjoner	6.1 – 6.5	6
7. Mål og enheter	7.1 – 7.8	9
8. Programmering	8.1 – 8.10	15
Sum	68	73

Tabell 18: *Antall; kapitler, leksjoner og sider i kompendium – Campus Matte 8.*

Campus Matte 8. trinn deler det digitale læremidlet inn i åtte forskjellige kapitler. Innenfor hvert kapittel finnes det leksjoner. Leksjonene er mindre matematiske objekter som skal hjelpe elevene til å få en bedre forståelse rundt det overordnede temaet for kapitlet. Oppsummert for *Campus Matte 8. trinn*, finnes det 8. kapitler, 68 leksjoner og 73 sider i kompendiet. I snitt utgjør dette 8,5 leksjoner for hvert kapittel og ca. 9,13 sider til hvert av de 8 kapitlene. En slik inndeling finner vi også i *Campus Matte 9. trinn*, men med færre temaer, leksjoner og sider i kompendiet.

Antall; kapitler, leksjoner og sider i kompendium – Campus Matte 9.

	Antall del-leksjoner	Antall sider i kompendium

1. Omkrets og areal	1.1 – 1.6	7
2. Overflate og volum	2.1 – 2.5	5
3. Formlikhet og Pytagoras	3.1 – 3.9	9
4. Statistikk	4.1 – 4.15	15
5. Sannsynlighetsregning	5.1 – 5.10	10
6. Programmering	6.1 – 6.14	18
Sum	59	64

Tabell 19: *Antall; kapitler, leksjoner og sider i kompendium – Campus Matte 9.*

Det digitale læremiddelet for 9. trinn består av totalt seks overordnede kapitler. Innenfor de ni kapitlene finner vi totalt 59 leksjoner, og 64 tilhørende sider i kompendiet med regler og eksempler. I snitt inneholder kapitlene ca. 9,83 leksjoner, og ca. 10,67 sider i kompendiet. Før oversikten over den helhetlige strukturen legges sammen og presenteres, må også læreverket for 10. trinn tas med.

Antall; kapitler, leksjoner og sider i kompendium – Campus Matte 10.

	Antall del-leksjoner	Antall sider i kompendium
1. Prosent	1.1 – 1.7	7
2. Potenser	2.1 – 2.3	4
3. Økonomi	3.1 – 3.10	15
4. Algebra	4.1 – 4.7	8
5. Likninger	5.1 – 5.14	19
6. Funksjoner	6.1 – 6.11	14
7. Programmering	7.1 – 7.16	21
Sum	68	88

Tabell 20: *Antall; kapitler, leksjoner og sider i kompendium – Campus Matte 10.*

For *Campus Matte 10.* trinn er det totalt sju kapitler, med 68 tilhørende leksjoner, og totalt 88 tildelte sider i kompendiet for 10. trinn. I snitt utgjør dette ca. 9,71 leksjoner i hvert av de syv kapitlene. I kompendiet til *Campus Matte 10.* er det totalt 88 sider, fordelt på syv temaer. For hvert av kapitlene utgjør dette i snitt ca. 12,57 sider med eksempler og regelsamling.

Tabellen vist nedenfor oppsummerer funnene i den strukturelle analysen av *Campus Matte 8.-10. trinn* for antall leksjoner, sider i kompendiet, leksjoner til kapitlene og antall sider i kompendiet, som hører til de ulike kapitlene. Sammenligner vi de ulike trinnene ser vi at 8. trinn har noe lavere antall leksjoner til kapitlene enn gjennomsnittet. Antall sider i kompendiet øker med årstrinn og ligger på 1,78 sider fler for hvert kapittel enn gjennomsnittet i læreverket. Ellers er det ingen åpenbare funn.

	Antall: leksjoner	Antall: sider kompendium	Antall: leksjoner til kapittel	Antall: sider kompendium til kapittel
Campus Matte 8.	68	73	8,5	9,13
Campus Matte 9.	59	64	9,83	10,67
Campus Matte 10.	68	88	9,71	12,57
Sum	195	225		
Gjennomsnitt for læremidlene	65	75	9,35	10,79

Tabell 21: Oversikt over helhetlig struktur for læreverket *Campus Matte*.

5.2 Læringsmålet programmering - Campus Matte

Campus Matte har ikke valgt å integrere programmering inn under andre temaer i matematikken. Programmering sett som en del av den digitale ferdigheten ville nok hatt en slik utforming, men *Campus Matte* velger å sette programmering inn i et eget kapittel. For hvert av ungdomstrinnene er det tildelt et visst antall sider og leksjoner til programmering. Leksjonene gjentar seg for hvert trinn, og man vil derfor kunne finne oppgaver i programmering på 8. trinn og på 9. trinn sine digitale læremidler i matematikkfaget. Ettersom dette er en repetisjon, vil jeg ikke gjennomgå de samme leksjonene to ganger. *Campus 8.-10.* inneholder også kompetanser fra 5. -7.trinn. Dette vil heller ikke bli gjennomgått da læreverket fra Gyldendal ikke er undersøkt for annet enn ungdomstrinnet. *Campus Matte* uttrykker ikke tydelige læringsmål i sitt digitale læreverk. Det nærmeste som kan tolkes til å være læringsmål, finnes i egenvurderingen. I egenvurderingen skal elevene selv vurdere sentrale mål for leksjonen, samt egen måloppnåelse. Egenvurderingen vil derfor tolkes som læringsmål i masteroppgaven og vil bli presentert nedenfor i tabell 22, 23 og 24. Tabellene uttrykker i likhet med den

horisontale analysen for læreverket *Maximum*, læringsmål med, og læringsmål uten programmering.

Læringsmål Campus Matte 8. Trinn

	Antall læringsmål	Antall læringsmål med programmering
1. Tall	30	-
2. Regning	34	-
3. Brøk	27	-
4. Potenser	31	-
5. Likninger og algebra	24	-
6. Funksjoner	15	-
7. Mål og enheter	27	-
8. Programmering	6	6
Sum	194	6

Tabell 22: Læringsmål Campus Matte 8. trinn.

Læringsmål Campus Matte 9. Trinn

	Antall læringsmål	Antall læringsmål med programmering
1. Omkrets og areal	19	-
2. Overflate og volum	19	-
3. Formlikhet og Pytagoras	30	-
4. Statistikk	46	-
5. Sannsynlighetsregning	31	-
6. Programmering	9	9
Sum	156	9

Tabell 23: Læringsmål Campus Matte 9. trinn.

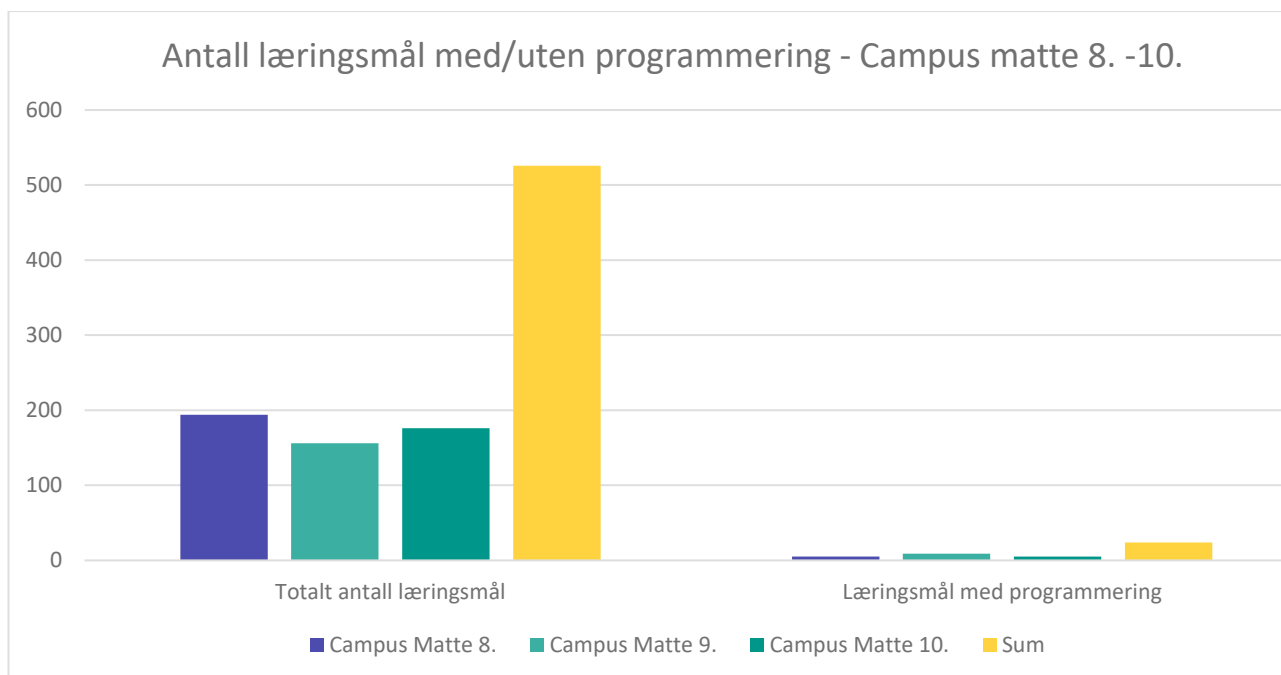
Læringsmål Campus Matte 10. Trinn

	Antall læringsmål	Antall læringsmål med programmering
1. Prosent	21	-
2. Potenser	11	-

3. Økonomi	33	-
4. Algebra	22	-
5. Likninger	44	-
6. Funksjoner	39	-
7. Programmering	5	5
Sum	176	5

Tabell 24: *Læringsmål Campus Matte 10. trinn.*

Figuren vist nedenfor oppsummerer oversikten over læringsmål i *Campus Matte* for 8., 9., og 10. trinn. Ut i fra tabellen ser vi at det er *Campus Matte 9. trinn* som har størst fokus på programmering i læringsmålene sine, da vektleggingen av programmering har en andel på ca. (5,77%) av alle læringsmålene på det digitale læremiddelet. *Campus Matte 10. trinn* fokuserer minst på programmering blant de tre læremidlene i læreverket ved ca. (2,84%) av læringsmålene i læremiddelet. På 8. trinn omhandler ca. (3,09%) av læringsmålene emnet. Totalt sett utgjør programmering ca. (3,80%) av læringsmålene i læreverket *Campus Matte 8-10. trinn.*



Figur 13: *Antall læringsmål med/uten programmering – Campus Matte 8. -10.*

For å supplere den horisontale analysen vil jeg også her avslutningsvis undersøke læreverket for oppgaver med programmering. Dette gjøres som i *Maximum*, der jeg ser etter de samme kodeordene. Programmeringsoppgaver slik jeg definerer det, inneholder

en form for bruk av digitalt verktøy, eller at det skal utarbeides en kode. *Campus Matte* har et eget kapittel for programmering med en progresjons oversikt fra 5. trinn til 10. trinn. Her vil kun programmeringsoppgavene behandles for ungdomstrinnet, slik jeg også vil gjøre i den vertikale analysen. Tabellene 25, 26 og 27 uttrykker dette forholdet mellom oppgaver med programmering, samt oppgaver uten programmering.

Antall oppgaver vs. antall oppgaver med programmering – Campus Matte 8.

1. Tall	280	0
2. Regning	348	0
3. Brøk	516	0
4. Potenser	404	0
5. Likninger og algebra	337	0
6. Funksjoner	206	0
7. Mål og enheter	364	0
8. Programmering	44	44
Sum	2499	44

Tabell 25: *Antall oppgaver vs. antall oppgaver med programmering – Campus Matte 8.*

Antall oppgaver vs. antall oppgaver med programmering – Campus Matte 9.

1. Omkrets og areal	167	0
2. Overflate og volum	101	0
3. Formlikhet og Pytagoras	167	0
4. Statistikk	580	0
5. Sannsynlighetsregning	256	0
6. Programmering	110	110
Sum	1381	110

Tabell 26: *Antall oppgaver vs. antall oppgaver med programmering – Campus Matte 9.*

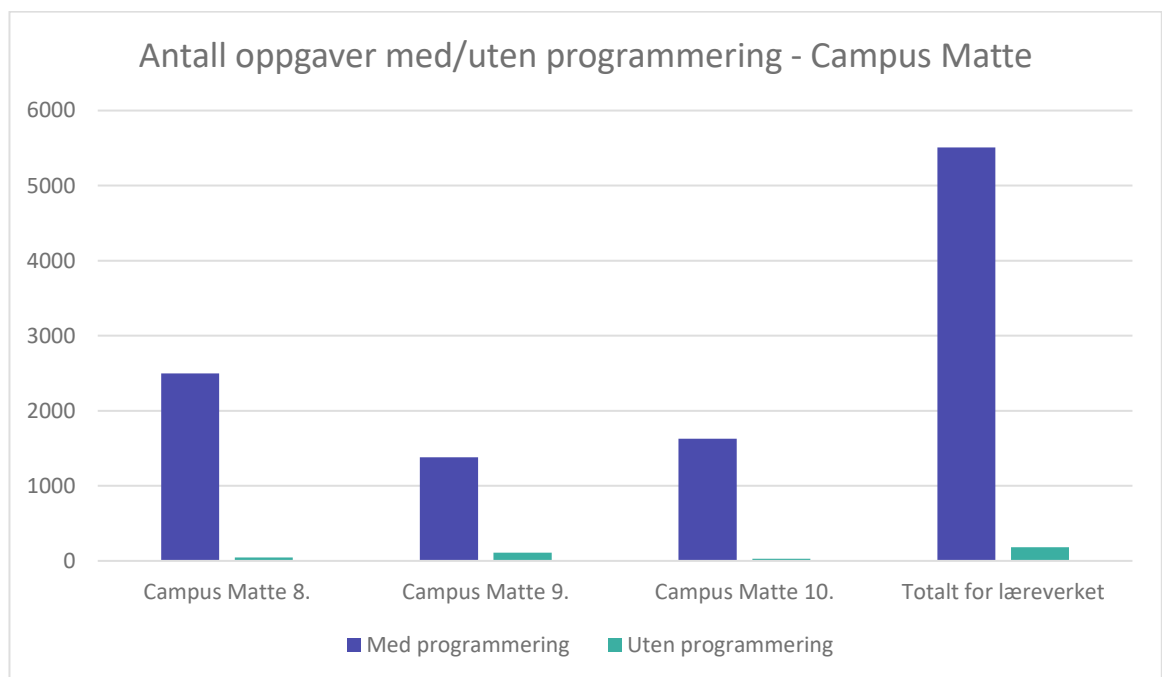
Antall oppgaver vs. antall oppgaver med programmering – Campus Matte 10.

1. Prosent	214	0
2. Potenser	120	0
3. Økonomi	365	0
4. Algebra	311	0

5. Likninger	324	0
6. Funksjoner	266	0
7. Programmering	29	29
Sum	1629	29

Tabell 27: Antall oppgaver vs. antall oppgaver med programmering – Campus Matte 10.

I oversikten over er totalt 5509 oppgaver gjennomgått for læreverket. Av de 5509 oppgavene inneholdt 183 oppgaver en form for programmering, slik det ble definert innledningsvis. Dette utgjør ca. (3,32%) av den totale mengden oppgaver i læreverket. Diagrammet under viser til forholdet mellom oppgaver med programmering, og oppgaver uten programmering i *Campus 8-10. trinn*;



Figur 14: Antall oppgaver med/uten programmering – Campus Matte.

Av oversikten over antall oppgaver i *Maximum* på ungdomstrinnet, inneholder 64 av totalt 1207 oppgaver programmering. Dette er ekskludert «aktiviteter» og «oppdrag». Tilsvarende i *Campus Matte* for ungdomstrinnet finner vi 183 programmeringsoppgaver av totale 5509 oppgaver. Andelen programmeringsoppgaver utgjør ca. (5,30%) av den totale mengden oppgaver i *Maximum*, mens jeg i *Campus Matte* finner at ca. (3,32%) inneholder programmering. Vektleggingen viser at *Campus Matte*, til tross for antall

oppgaver, ikke har valgt å legge mer vekt på programmering sammenlignet med *Maximum*, i forhold til de andre matematiske temaene.

Med tanke på antall sider kan det være vanskelig å sammenligne strukturen ettersom oppgavene er adskilt fra forklaringer i det digitale læreverket *Campus Matte*. Kompendiet, som er et vedlegg, inneholder eksempler, og ikke oppgavene som er produsert for læreverket. Jeg vil derfor ikke gå videre med dette punktet i denne analysen.

Fra den horisontale analysen av læreverket *Maximum* og *Campus Matte* for ungdomstrinnet, er det noen sentrale funn jeg ønsker å trekke frem. Det første punktet er læringsmål i de forskjellige verkene. Begge læreverkene skriver at de er oppdatert for LK20. Da vil det være naturlig at kompetansemålene er integrert i læreverket. Læreverkene har integrert kompetansemålene om programmering på ulikt vis. For å gjenta dette, uttrykkes programmering i LK20 slik;

8. trinn -Utforske hvordan algoritmer kan skapes, testes og forbedres ved hjelp av programmering.
9. trinn -simulere utfall i tilfeldige forsøk og beregne sannsynligheten for at noe skal inntreffe, ved å bruke programmering.
10. trinn -utforske matematiske egenskaper og sammenhenger ved å bruke programmering.

(Kunnskapsdepartementet, 2019).

Kompetansemålene i matematikk fra læreplanen uttrykkes i læreverkene slik tabell 28 viser. Ettersom *Campus Matte* ikke uttrykte konkrete læringsmål, vil egen vurderingen og de målene som finnes her, kun være betraktet som læringsmål. Hovedforskjellen blant læremidlene om læringsmål i programmering, er at *Campus Matte* sitt læremiddel i en større grad inneholder detaljerte læringsmål. *Maximum* inneholder fire romslige læringsmål elevene skal jobbe mot. Innenfor disse læringsmålene er det ulike elementer elevene må mestre for å kunne nå disse læringsmålene. *Campus Matte* derimot, inneholder et detaljert sett med 17 læringsmål omkring temaet programmering. Tettheten læringsmålene medfølger til et læremiddel, er i større grad preget av læringsmål, mer enn det *Maximum* er. Et sentralt spørsmål her vil være hvilken påvirkning denne utformingen har på temaet programmering, og oppgavetyper som

stilles elevene. Er for eksempel en detaljert målstyring knyttet til enklere oppgaver av lavere kognitive krav, der det kun kreves et svar? Eller medfølger romslige læringsmål i programmering oppgaver med rom for utforskning og problemløsning, som stiller høyere kognitive krav?

5.3 Vertikal analyse

Grunnlaget for funn i den vertikale analysen baserer jeg på Stein et al. (2009) «Task analysis guide» og Charalambous et al. (2010) «Type svar. «Task analysis guide» vil i den vertikale analysen bidra til å analysere læreverkets oppgaver for kognitive nivåer, mens «type svar» vil fokusere på hvilke type svar oppgavene krever av elevene. I kapittel 5.3 vil resultatene av den vertikale analysen presenteres, og interessante funn vil trekkes frem. Avslutningsvis vil det være en oppsummering av funn som vil bli videreført til det neste kapittelet.

5.3.1 Kognitive krav – Maximum 8-10

Maximum 8. 2. utgave – Grunnbok

	Maximum 8.
Memorering	13,04%
Prosedyrer uten sammenheng	43,48%
Prosedyrer med sammenheng	43,48%
Å gjøre matematikk	0%
Sum	100%

Tabell 29: *Fordeling av kognitive krav Maximum 8. 2. utgave.*

I tabell 29 vises resultatene av undersøkelsen av *Maximum 8.* for kognitive krav. Som nevnt tidligere i teorikapittelet, anses «memorering» og «prosedyrer uten forbindelse» som å stille lave kognitive krav. Oppgaver av denne typen ble identifisert gjennom punktene som ble beskrevet i analysens rammeverk; «Task analysis guide» (Charalambou et al. 2010). For å nevne korte kjennetegn på slike oppgaver er at de:

1. Reproduserer fakta og formler uten noen kobling mot matematiske objekter. Fra programmeringsoppgavene viser det seg at noen oppgaver er veldig lik oppgaveteksten. Oppgavene har ikke fokus på det matematiske objektet eller prosedyren.

2. Fra memorering og til prosedyrer uten forbindelser oppfordrer oppgavene til å ta i bruk en prosedyre. Prosedyren derimot fremstår ikke som knyttet opp til et spesifikt matematisk objekt. Slike prosedyrer har mer fokus på å finne riktig svar enn å skape matematisk forståelse.

I analysen kan vi ser at den største delen av oppgaver har blitt tolket og kategorisert til kategoriene, PUF (prosedyrer uten sammenheng) og PMS (prosedyrer med sammenheng). Resterende oppgaver er kategorisert som M (memorering), ettersom at det ikke var noen oppgaver som falt under kategorien ÅGM (å gjøre matematikk). En lignende fordeling ser vi også i *Maximum 9*, der majoriteten av oppgaver, henholdsvis er kategorisert som PUS og PMS. Ulikt fra *Maximum 8*, kan vi her finne at 8,70% av oppgaver om programmering er kategorisert ÅGM.

Maximum 9. 2. utgave – Grunnbok

	Maximum 9.
Memorering	8,70%
Prosedyrer uten sammenheng	47,83%
Prosedyrer med sammenheng	34,78%
Å gjøre matematikk	8,70%
Sum	100%

Tabell 30: *Fordeling av kognitive krav Maximum 9. 2. utgave.*

I maximum 9, blir også programmeringsoppgavene fordelt på de tre første nivåene av kognitive krav. Her er PUS representert i et større omfang enn ved *Maximum 8*. PMS ser derimot ut til å ha avtatt i forhold til 8. trinn. Nivået som omhandler prosedyrer med forbindelse ble identifisert ved at oppgavene hadde fokus på:

1. Prosedyren og forståelsen denne medbringer. Altså ikke nødvendigvis på at det skulle produseres ett riktig svar.
2. Ulike fremgangsmåter var representert og oppgavene oppfordret til å ta i bruk forskjellige fremgangsmåter.

Maximum 10. 2. utgave – Grunnbok

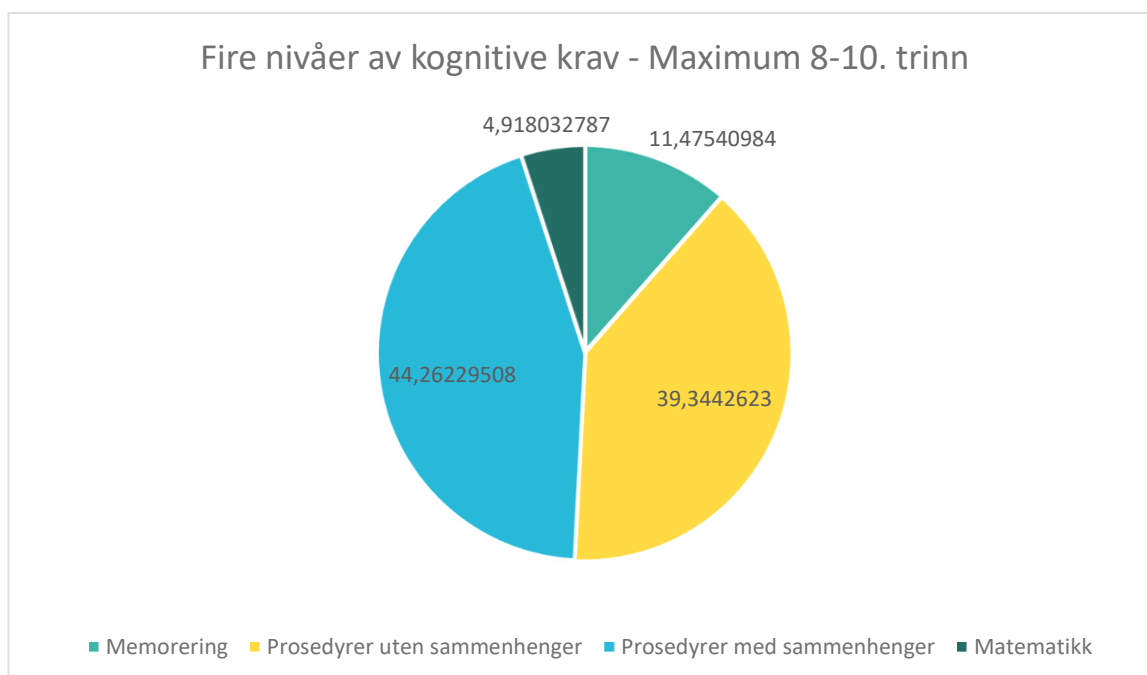
	Maximum 10.
Memorering	13,3%
Prosedyrer uten sammenheng	20%

Prosedyrer med sammenheng	60%
Matematikk	48%
Sum	6,67%

Tabell 31: *Fordeling av kognitive krav Maximum 10. 2. utgave*

Til slutt i den vertikale analysen finner vi en betydelig forflytning av oppgaver mot høyere kognitive krav. Kategorien PMS er representert i 60% av tilfellene i oppgaver om programmering for matematikk 10. trinn. Forflytningen til et høyere krav har skjedd ved en reduksjon av PUS, forflyttet over til PMS. Oppgaven på dette nivået ble identifisert gjennom de samme kriteriene som ble brukt for de to andre lærebøkene. Det som kjennetegner oppgaven på det høyeste nivået er at de;

1. Krever at elevene forstår ulike matematiske konsepter.
2. Krever at elevene analyserer ulike deler for så å løse oppgaven.
3. Ikke en entydig fremgangsmåte mot målet.
4. Fokuset er på prosedyren og matematikken som oppstår gjennom utforskning og analyse.



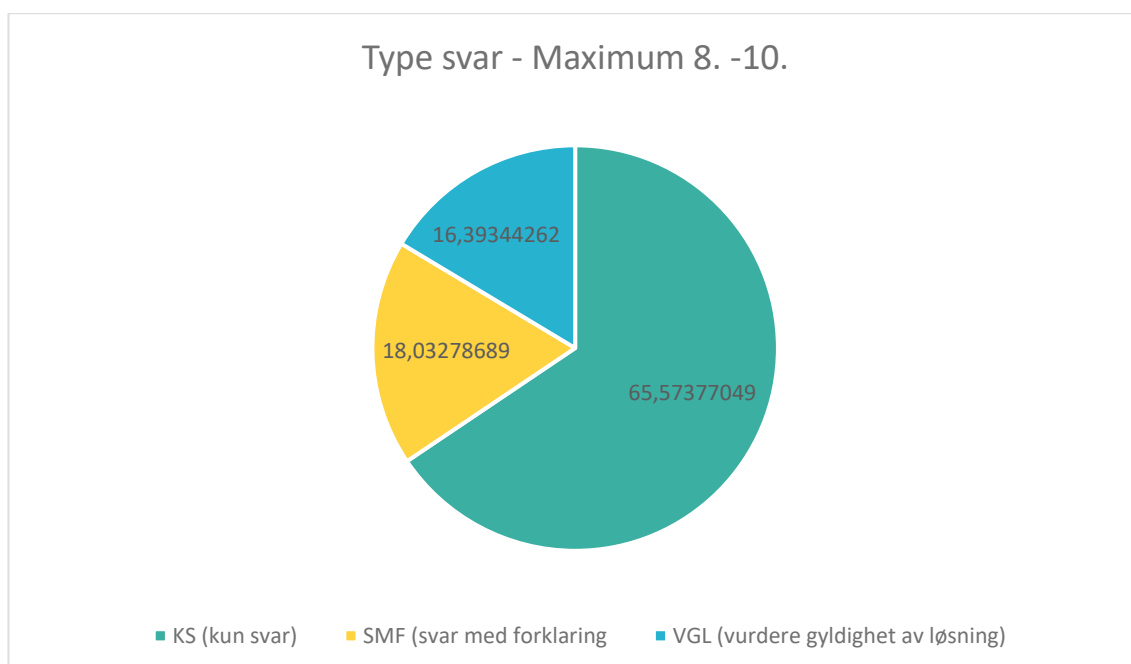
Figur 15: *Fire nivåer av kognitive krav – Maximum 8. -10. trinn.*

Sektordiagrammet ovenfor viser fordelingen av oppgaver blant de fire forskjellige kognitive nivåene i *Maximum 8-10. Grunnbok*. Legger vi sammen oppgavene, ser vi at den totale summen av programmeringsoppgaver i læreverket *Maximum 8-10. grunnbok 2. utgave*, utgjør 61 oppgaver. Her er det ikke tatt med sider med såkalt «aktivitet» eller

«oppdrag». Av den totale mengden oppgaver ser vi at ca. 44,26% av oppgavene faller inn under kategorien *Prosedyrer med forbindelser*. Det vil si at det finnes gjennomsnittlig 6,74 oppgaver av denne typen i hver bok. Ettersom ca. halvparten av oppgavene er på et høyere kognitivt nivå, er det interessant å se at det bare er ca. 4,92% av disse oppgavene i de tre lærebøkene som har blitt kategorisert til det høyeste nivået. Av den totale mengden tilsvarer kategorien ÅGM 4,92% av den totale mengden. Dette utgjør i snitt ca. tre oppgaver i programmering på det høyeste nivået, fordelt i de tre læremidlene. Kategoriene som anses for å stille lave kognitive krav til elevene, utgjør til sammen ca. 50,82% av den totale mengden programmeringsoppgaver. Av denne mengden er andelen ca. 39,34% av oppgaver som faller innenfor *Prosedyrer uten forbindelse*, mens ca. 11,48%, er av kategorien *Memorering*. Gjennomsnittlig for de tre læreverkene, vil hvert læreverk inneholde ca. 7,75 oppgaver, på de laveste kognitive nivåene. Ser vi programmeringsoppgavene i sammenheng med resten av oppgavene i læreverket *Maximum 8-10.*, utgjør oppgavene, som nevnt i den horisontale delen, ca. 5,3% prosent av den totale mengden oppgaver.

5.3.2 Type svar – Maximum 8. -10. trinn

I undersøkelsen av hvilke type svar oppgaver om programmering krever av elevene, er det blitt operert med kategoriene KS (kun svar), SMF (svar med forklaring) og VGL (vurdere gyldighet av løsningen). Hvilken kategori oppgavene i programmering kategoriseres i, avhenger av hvilken type svar oppgavene krever av elevene. Dersom en oppgave er delt i flere deloppgaver, der noen oppgaver kun krever svar, mens andre deloppgaver i krever en forklaring, vil oppgaven kategoriseres som SMF. Figur 15. viser fordelingen av type svar, for læreverket *Maximum 8. -10.*;



Figur 16: *Type svar – Maximum 8. -10.*

Sektordiagrammet i figur 15 viser at ca. 65,57% av oppgavene kategoriseres til KS, ca. 18,03% av oppgavene krever SMF, mens ca. 16,39% oppfordrer leseren til å vurdere gyldigheten av løsningen.

5.3.3 Kognitive krav – Campus Matte 8-10

Videre vil jeg gjøre en lignende vertikal analyse for det digitale læreverket *Campus Matte 8-10*. trinn. Etersom kjennetegn på de ulike kategoriene har blitt beskrevet ovenfor, vil jeg ikke utdype de noe ytterligere i dette delkapittelet. I *Campus matte for 8. trinn* finner vi 20, av de totalt 44 oppgavene er kategorisert i kategoriene for lave kognitive krav. Disse kategoriene utgjør ca. 45,44% av alle oppgaver med programmering i læremiddelet. En overvekt er å finne for kategoriene som stiller høye kognitive krav, med til sammen 24 oppgaver, av de totalt 44 oppgavene. Dette utgjør ca. 54,54% av alle oppgavene i læremiddelet.

Fordeling av kognitive krav - *Campus matte 8.*

	Campus Matte 8.
Memorering	13,63%
Prosedyre uten forbindelse	31,81%
Prosedyrer med forbindelse	34,09%
Matematikk	20,45%

Tabell 32: *fordeling av kognitive krav – Campus Matte 8.*

For *Campus Matte 9. trinn*, har ingen matematikkoppgaver blitt vurdert til det laveste nivået M. Av den totale mengden på 88 oppgaver om programmering, er 53,4% av oppgavene kategorisert PUS. Selv om det ikke er noen oppgaver kategorisert på det laveste nivået M, er over halvparten av oppgavene å finne i den andre kategorien PUS, som stiller lave kognitive krav. Oppgaver som krever høye kognitive krav, utgjør de resterende, ca. 46,59%. Sammenlignet med *Campus Matte 8.*, ser vi en lignende fordeling av antall oppgaver mellom høye og lave kognitive krav. Forskjellen derimot er en overvekt av oppgaver på et lavere kognitivt nivå, enn det er oppgaver på høyt kognitivt nivå.

Fordeling av kognitive krav - *Campus matte 9.*

	Campus Matte 9.
Memorering	0%
Prosedyre uten forbindelse	53,4%
Prosedyrer med forbindelse	27,7%
Matematikk	19,31%

Tabell 33: *fordeling av kognitive krav – Campus Matte 9.*

Videre for *Campus Matte 10. trinn* ser vi en lik tendens som i *Maximum 8*, der overvekten er på oppgaver med høye kognitive krav. I læremiddelet finner vi totalt 25 oppgaver med programmering, hvorav 22 av dem befinner seg i kategoriene PMS og ÅGM. Kun tre oppgaver befinner seg på PUS, mens ingen oppgaver på M. Oppgaver som er kategorisert for å kreve lave kognitive krav, utgjør 12% av hele oppgaven. De resterende 88% av oppgavene er kategorisert for å stille høye kognitive krav.

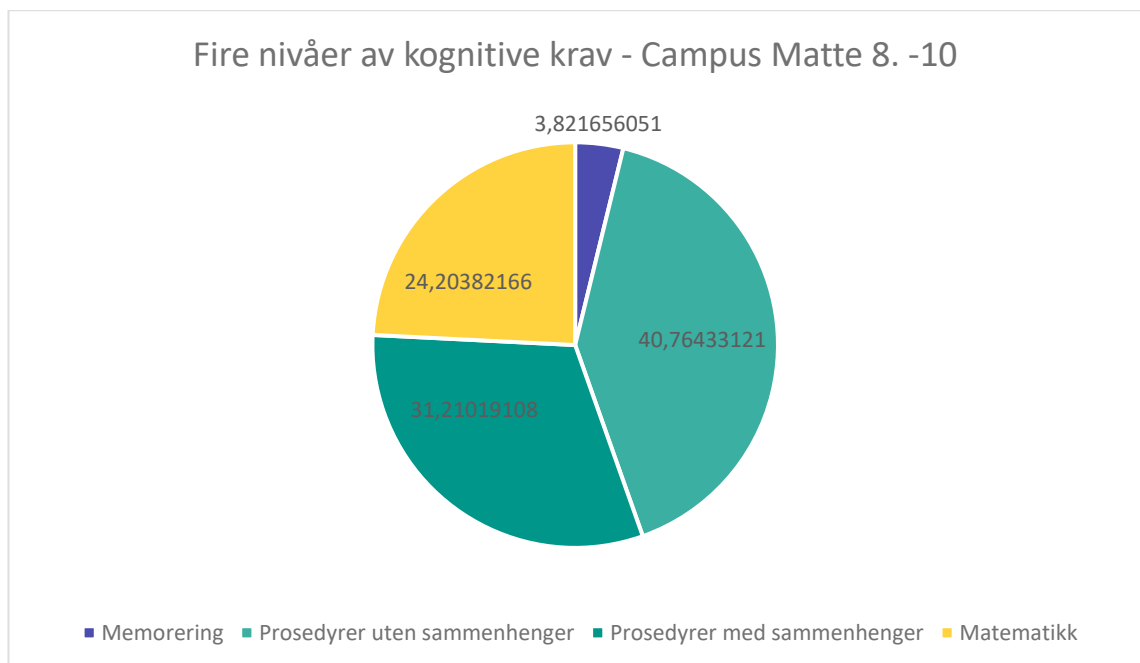
Overvekten her er betydelig på oppgaver med høye kognitive krav, i forhold til lave.

Fordeling av kognitive krav - *Campus matte 10.*

	Campus Matte 10.
Memorering	0%
Prosedyre uten forbindelse	12%
Prosedyrer med forbindelse	40%
Matematikk	48%

Tabell 34: *fordeling av kognitive krav – Campus Matte 10.*

Under presenteres en vektlegging av de ulike kognitive nivåene for læreverket *Campus Matte*;



Figur 17: *Fire nivåer av kognitive krav – Campus Matte 8. -10.*

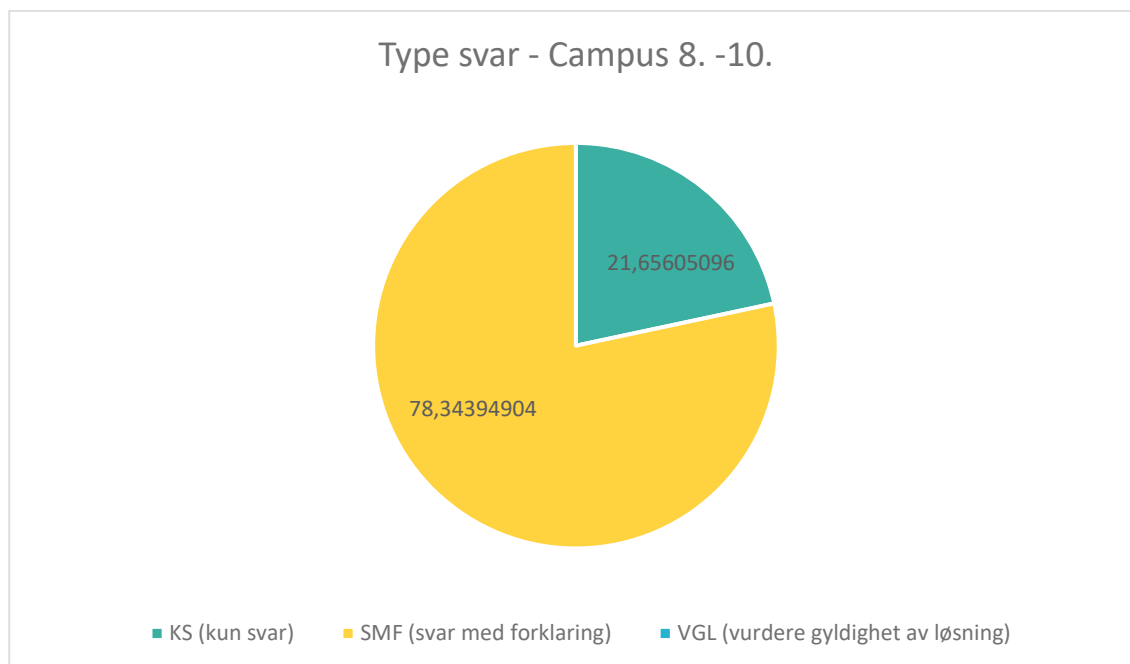
På figur 16 ser vi at totalt for læreverket er ca. 3,82% av oppgavene kategorisert til det laveste nivået av kognitive krav. Det neste nivået av kognitive krav, utgjør ca. 40,76% av oppgavene, mens det nest høyeste nivået, utgjør ca. 31,21% av alle oppgavene. Til slutt står det høyeste nivået av kognitive krav, for ca. 24,2% av oppgavene. Totalt sett, utgjør kategoriene som betegnes ved lave kognitive nivåer, ca. 44,58% av den totale mengden oppgaver i læreverket. Overvekten her er å finne på oppgaver som ligger på et høyere kognitivt nivå. Disse utgjør de resterende ca. 55,42% av den totale mengden oppgaver med programmering i læreverket *Campus Matte 8. -10.*

5.3.4 Type svar – Campus Matte 8-10

I undersøkelsen av hvilke type svar oppgaver om programmering krever av elevene, er det også her, for det digitale læreverket, operert med kategoriene KS (kun svar), SMF (svar med forklaring) og VGL (vurdere gyldighet av løsningen). En utfordring i kategoriseringen av oppgavene var at oppgavene ofte kun forventet et svar.

Oppgavetyper forventet svar, men at svaret kom i form av et program. Det vil si at elever må formulere en fremgangsmåte (program), som gir det riktige svaret på oppgaven, for å kunne bestå oppgaven. Her var det ikke mulig å kun taste inn ett enkelt svar. Oppgaver som krever slike typer svar vil derfor kategoriseres i kategorien SMF. Årsaken til at slike oppgaver ikke kategoriseres i VGL, er fordi de ikke krever at eleven

vurderer gyldigheten av løsningen. Fremgangsmåten som blir presentert gjennom et program, forklarer hvilke valg som er tatt, ikke en vurdering av dem.



Figur 18: *Type svar – Campus Matte 8. -10.*

Sektordiagrammet i figur 16, viser at ca. 21,65% av oppgavene kun krever et svar for å kunne løses, mens de resterende oppgavene på ca. 78,34%, krever en form for forklaring. Fraværende i læreverket *Campus Matte 8. -10.* er den siste kategorien hvor oppgavene oppfordrer brukeren til å vurdere gyldigheten av løsningen som er brukt. Dette kan ha sammenheng med oppgaver som er tilpasset for å gi elevene en tilbakemelding. For at elevene skal kunne se progresjonen sin oppsummert, må oppgavene ha et riktig svar for å kunne gi en tilbakemelding. Oppgaver der elevene må vurdere gyldigheten av sine egne løsninger vil derfor være vanskelige å vurdere for det digitale læreverket. Til tross for dette har læreverket sine oppgaver en tydelig forventning om at de krever et svar med en forklaring.

5.4 Kognitive krav og Type svar

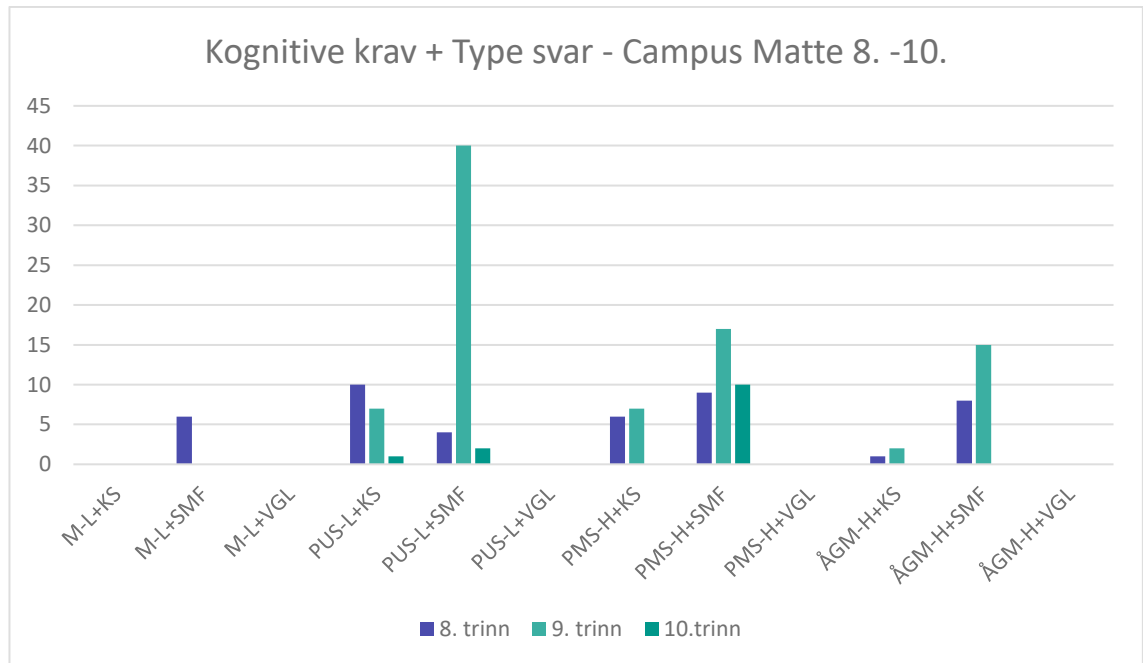
Dersom vi setter de to delene fra «Task analysis guide» (Charalambous et al., 2010), «Kognitive krav» og «Type svar» sammen. Kan vi belyse hvilke type svar av oppgaver av ulike kognitive krav krever. Kategoriene vi får er;

| KS | SMF | VGL

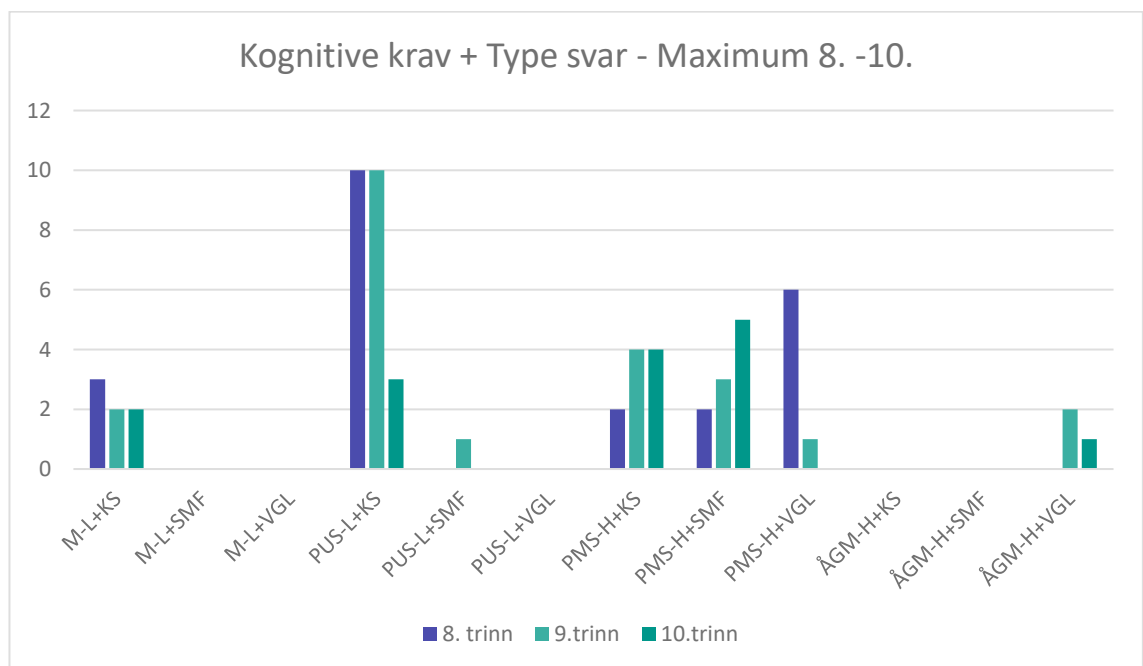
M-L	M-L+KS	M-L+SMF	M-L+VGL
PUS-L	PUS-L+KS	PUS-L+SMF	PUS-L+VGL
PMS-H	PMS-H+KS	PMS-H+SMF	PMS-H+VGL
ÅGM-H	ÅGM-H+KS	ÅGM-H+SMF	ÅGM-H+VGL

Tabell 35: Mulige kombinasjoner av kognitive krav og type svar.

Oppsummert for det digitale læreverket *Campus Matte*, utgjør de forskjellige kombinasjonene av kategorier;



Figur 19: Kognitive krav + type svar – Campus Matte 8. -10.



Figur 20: Kognitive krav + type svar – Maximum 8. -10.

6 Diskusjon

I kapitlet vil jeg ta for meg ulike funn fra kapittel 5, og diskutere funnene opp mot teori. For at en innholdsanalyse skal være kvalitativ, er det ikke nok å fremheve ulike mønster fra datamaterialet. Mønstrene må også fortolkes, og kobles opp mot teori. Til å diskutere funn fra innholdsanalysen, vil jeg bruke tidligere forskning presentert i kapittel 3. teori. Først vil jeg se på den horisontale analysen for begge læreverkene *Maximum* og *Gyldendal*, etterfulgt av den vertikale for læreverkene. En enkel sammenligning mellom de to læreverkene vil være naturlig for å løfte frem sider ved forskningsspørsmålene og problemstillingen. Under gjentas forskningsspørsmålene og overordnet problemstilling for masteroppgaven;

I hvilken grad utfordres elevene kognitivt i møte med digitale og analoge programmeringsoppgaver presentert i læreverk for fagfornyelsen?

- a) Hvordan er programmeringsoppgaver implementert og vektlagt i læreverkene?
- b) Hvilke nivåer av kognitive krav stiller programmeringsoppgavene til elevene?

6.1 Drøfting av horisontal analyse

Forskningsspørsmål (a), nevnt ovenfor skal prøve å besvares gjennom den horisontale analysen. Bakgrunnen for forskningsspørsmålet er å få en oversiktlig forståelse av hvordan programmering er integrert i de to forskjellige læreverkene *Maximum* og *Campus Matte*. Ved å analysere læreverkene for; antall leksjoner, sider for hver leksjon, struktur og læringsmål, med fokus på programmering, kan vi si noe om temaets plass i de nye læremidlene som er oppdatert for LK20. Den helhetlige strukturen fungerer også som en presentasjon av oppgavens omfang og gyldighet. Ettersom oppgaven ikke foretar seg en større analyse av et betydelig antall læreverk, er det begrensninger for hvilke konklusjoner som kan tas.

6.1.1 Bakgrunnsinformasjon

Ut ifra bakgrunnsinformasjonen er det ikke enkelt å trekke noen særlig relevante koblinger opp mot problemstillingen. Det som er verdt å merke, er at de to forskjellige forlagene *Gyldendal* og *Campus Inkrement*, har brukt samme forfattere gjennom

bøkene. I læreverket *Maximum* har de samme forfatterne formet de to første bøkene, mens den siste boken, var det kun 4 av forfatterne. For læreverket *Campus Matte* har de samme forfatterne formet kompendiet, som hører til det digitale læremidlet. Å bruke samme forfattere gjennom hele læreverket for ungdomstrinnene kan både være positivt og negativt.

Utdanningsdirektoratet har utviklet et kvalitetskriterium for læremidler (2018). I kunnskapsgrunnlaget for kvalitetskriteriet, gjengir Udir paragraf 17.1 i opplæringsloven; «Med læremiddel meiner ein alle trykte, ikkje-trykte og digitale element som er utvikla til bruk i opplæringa. Dei kan vere enkeltstående eller gå inn i ein heilskap, og dekkjer aleine eller til saman kompetansemål i Læreplanverket for Kunnskapsløftet» (Utdanningsdirektoratet, 2018, s. 5). Selv om læreverkene dekker kompetansemålene i LK20, vil den generelle utformingen av boken, i stor grad, være påvirket av forfatternes forståelse av læring. Er forfatterne tilhengere av, eller bevisst på teoriene til Piaget, eller forståelsen til Skemp? Oppfordrer læreverkene til en dypere forståelse, slik det er uttrykt i LK20 sitt krav om dybdelæring?

Fra bakgrunnsinformasjonen vil det være naturlig å tenke at de forskjellige læremidlene i læreverkene inneholder like elementer, ettersom de er utformet av samme personer. Dette er med tanke på strukturen til boken, som omtales i delkapittel 6.1.2, og utformingen av oppgavetyper.

6.1.2 Helhetlig struktur

Forskningsspørsmål a)

Hvordan er programmeringsoppgaver implementert og vektlagt i læreverkene?

I den helhetlige strukturen er det analysert for læreverkenes; antall leksjoner, sider for hver leksjon, struktur av leksjoner og læringsmål (Charalambous et al. 2010). Fra den helhetlige strukturen ønsker jeg å trekke frem bruken og vektlegging av læringsmål samt oppgaver med programmering. For å få en forståelse av den overordnede problemstillingen, er det viktig å vite hvilken kontekst programmering befinner seg i skolen. Ved å se på læringsmål og antall oppgaver der programmering inngår, kan vi få en formening om denne konteksten.

I læreverket *Maximum* finner vi totalt 91 læringsmål. Disse 91 læringsmålene er fordelt på 12 kapitler i læreverket. Av de 91 læringsmålene læreverket fremmer for leseren, inneholder 4 av disse en form for programmering. Det vil si at programmering endten refereres spesifikt til, eller at det skal lages/tolkes koder til et program. Tilsvarende i det digitale læreverket *Campus Matte*, finner vi 526 læringsmål. Målene for undervisningen er uttrykt i elevenes egenvurdering, og er i forhold til *Maximum*, beskrevet veldig detaljert. Av de 526 læringsmålene omhandler 20 av disse programmering. Dette utgjør totalt ca. 3,8% av den totale mengden læringsmål i *Campus Matte* 8. -10. trinn. Sammenlignet med *Maximum* hvorav ca. 4,40% av oppgavene inneholder en form for programmering. Dersom læringsmålene skulle vært fordelt ut over de 31 kompetansemålene i matematikkfaget, ville hvert kompetansemål tildeles ca. 2,94 læringsmål i *Maximum*, og ca. 16,97 læringsmål i *Campus Matte*. Dette utgjør henholdsvis 3,23% av den fordelte mengden i *Maximum*, og ca. 3,23% av den fordelte mengden for *Campus Matte*. Når vi sammenligner antall læringsmål som er tildelt programmering, ser vi at begge læreverkene har flere læringsmål for kompetansemålet programmering enn hva snitte ville være dersom læringsmålene var delt likt mellom kompetansemålene.

Ser vi på vektlegging av oppgaver med programmering i læreverkene, viser det seg at *Maximum* inneholder 1207 oppgaver, derav 64 inneholder en form for programmering. Oppgaver med programmering utgjør ca. 5,30% av den totale mengden oppgaver i læreverket *Maximum*. For *Campus Matte* viser analysen at læreverket inneholder totalt 5509 oppgaver, der 183 av disse, er oppgaver med programmering. Dette utgjør ca. 3,32% av den totale mengden oppgaver i læreverket. Fordeler vi antall oppgaver på antall delkapitler i boken, får vi for *Maximum* 1207 oppgaver fordelt på 40 delkapitlet, ca. 30,18 oppgaver til hvert delkapittel. For læreverket *Campus Matte*, får vi ca. 262,3 oppgaver for hvert delkapittel dersom vi deler den totale mengden 5509, på 21 forskjellige delkapitler. Sammenligner vi med hvor mange oppgaver som faktisk er tildelt programmering, ser vi at læreverket *Maximum* i stor grad vektlegger kompetansemålet i programmering, mens *Campus Matte* tildeler andre temaer oppgaver, fremfor programmering.

For å oppsummere den horisontale analysen, og svare på forskningsspørsmål (a), vil jeg fremheve at programmering er vektlagt i begge læreverkene sine læringsmål. I antall oppgaver, er programmering underrepresentert i læreverket *Campus Matte*, sammenlignet med andre matematiske emner. For læreverket *Maximum* derimot, er oppgaver med programmering godt representert, sammenlignet med andre matematiske emner. Den horisontale analysen tyder på at oppgaver med formes og vektlegges på ulikt vis. Det er vanskelig å trekke noen klare tråder mellom teori og empiri i den horisontale analysen. Bakgrunnsinformasjon og den helhetlige strukturen skal være med på å sette programmering inn i en kontekst, den vertikale analysen gjøres ut ifra.

6.2 Drøfting av vertikal analyse

I henhold til Charalambous et al. (2010) ble det gjennomført en vertikal analyse for kognitive krav, og hvilken type svar, oppgaver om programmering krever av elevene. Målet med den vertikale analyse er å belyse forskningsspørsmål (b), og den overordnede problemstillingen nevnt innledningsvis i kapittelet. Til å definere kognitive krav, og hva som kjennetegner dem, ble det hentet inspirasjon fra Valenta (2016) og Stein et al. (2000) kjennetegn på de ulike nivåene.

6.2.1 Diskusjon av kognitive krav

Funn fra den vertikale analysen indikerer at, «prosedyrer med sammenhenge», er det mest utbredte kognitive nivået i læreverket *Maximum*. Med en andel på ca. 44,26% utgjør det kognitive nivået, nesten halvparten av oppgavene. Dersom vi legger sammen dette nivået med det siste nivået «matematikk», som også stiller høye kognitive krav, utgjør de to nivåene tilnærmet halvparten av oppgavene i læreverket. Det vil si at den resterende halvparten stiller lave kognitive krav til elevene. Denne 45/55 fordelingen finner vi ikke igjen for læreverket *Campus Matte*. Her er overvekten på 65% av oppgavene, på de to høye nivåene av kognitive krav, mens ca. 35% av oppgavene har lavere kognitive krav. Ser vi til Hofstad & Liland (2021) finner vi en lignende fordeling av oppgaver på de forskjellige kognitive nivåene for læreverket *Maximum*. Dette kan indikere at tolkningsprosessen, som er tatt i bruk, samsvarer med andres tolkningsprosess til kognitive krav. Derimot finner vi ikke et lignende resultat for fordelingen av oppgaver for det digitale læreverket *Campus Matte*. Fordelingen av oppgaver på de kognitive nivåene, strider med forskningen på feltet, ved at største delen av oppgaver befinner seg på kategoriene som stiller høye kognitive krav.

Årsaken til at *Campus Matte* har en betydelig større andel av oppgaver med høye kognitive krav, er fordi det er mange av oppgavene i læreverket som er blitt kategorisert innenfor det høyeste nivået. Dette kan tyde på at læreverkets tydelige undervisningsform «omvendt undervisning» kommer til uttrykk i datamaterialet. Gjennom analysen fikk man som forsker en oppfattelse av at *Campus Matte* bestod av åpne oppgaver, der elevene i større grad må utforske for å komme frem til en løsning. Selv om oppgavene fremstår mer utforskende, er det ikke nødvendigvis slik. I *Maximum* virket også oppgavene utforskende, men da disse stod tett på eksempler og ferdige fremgangsmåter, ble det utforskende aspekter betydelig redusert. I læreverket *Campus Matte* står oppgavene adskilt fra eksempler og fremgangsmåter. Oppgavenes karakter fremstår derfor mer åpne.

Når man kategoriserer oppgaver for kognitive krav, er det med hensikt å finne ut hvilke muligheter som finnes i oppgavene. For oppgaver i programmering fra de to læreverkene, er muligheten for at elevene blir utfordret kognitivt, helt tydelig der.

6.2.2 Diskusjon av type svar

Når oppgavene i læreverkene skulle kategoriseres for hvilke type svar de krevde av elevene, utviklet det seg et klar tendens. For læreverket *Maximum* finner vi den største delen av oppgaver innenfor kategorien KS (kun svar). Hele 65. prosent av oppgavene i læreverket falt under denne kategorien. De resterende oppgavene ble fordelt omtrent likt i de resterende kategoriene SMF (svar med forklaring) og VGL (vurdere gyldighet av løsning). For læreverket *Campus Matte* er den store andelen fordelt på SMF, ved unntak av ca. 21. prosent, kategorisert i KS. I analysen av *Campus Matte* tolker jeg ingen av oppgavene inn i den siste kategorien VGL.

Tar vi et tilbakeblikk til kapittel 3. hvor Valenta (2016) presenterer kjennetegn ved de ulike kognitive nivåene. Ser vi at oppgaver på et høyt kognitivt nivå beskrives med begreper som *utforskning*, *systematisk*, *strategier* og *resonnering*. Når elevene går frem i oppgaver med programmering, er det essensielt at dette siste begrepet om *resonnering* er ivaretatt. Her kan vi trekke tråder til Piagets begreper om *assimiliasjon* og *akkomodasjon*. For at en elev skal kunne utvikle en bedre, eller ny forståelse, krever det en form for resonnering. Passer den nye informasjonen inn i de allerede eksisterende

skjemar, eller trengs det en ny organisering av kunnskapen. Når elevene ikke behøver å svare utdypende, altså mer enn en forklaring der de aktivt resonneres over valge i løsningen, kan det tenkes at det ikke vil utvikles en dypere forståelse. Ser vi på analysen av kognitive krav, er det helt tydelig at de fleste oppgavene stiller høye kognitive krav. Muligheten for å drive dybdelæring og er helt klart der, men når vi snakker om forståelse og resonnering over de læringsprosessene som utføres, er det ikke samsvar. Problemløsningsoppgaver der elevene i stor grad har fokuset på å finne riktig svar, samsvarer ikke med Kunnskapsdepartementet (2019) uttalselse; «Elevene skal legge mer vekt på strategiene og framgangsmåtene enn på løsningene» (Kunnskapsdepartementet, 2019, s. 2).

6.2.3 Kognitive krav + Type svar

Drøftingen over belyses gjennom kombinasjonen av *kognitive krav* og *type svar*. I figur 19., og 20. ser vi at kategorien PUS-L+KS, og PUS-L+SMF har den største andelen av lave kognitive oppgaver. Det betyr at det er få oppgaver i enden av spekteret ved kategorien memorering. Oppgaver som har lave kognitive krav, krever bare KS (kun svar) og SMF (svar med forklaring). Ingen av oppgavene i kategoriene M og PUS, krever at elevene skal vurdere gyldigheten av løsningen. Et lignende mønster finner vi også for de høye kognitive nivåene, hvor majoriteten er samlet rundt KS og SMF. Til tross for dette ser vi at det finnes også oppgaver kategorisert som ÅGM-H+VGL. Dette mindretallet av oppgaver kan tenkes å få sitt fulle potensial utfyllt, ved at det krever en aktiv resonnering hos eleven.

7 Konklusjon

I denne studien har jeg gjennomført en horisontal og vertikal analyse av to læreverker i matematikk, for å belyse hvilke kognitive krav oppgaver med programmering stiller. Dette er gjort gjennom den todelte analysen hvor det ble sett på læreverkenes struktur, i tillegg til at det ble analysert for hvilke kognitive krav og svar oppgavene stiller. Avslutningsvis er funnene koblet opp mot teori og tidligere forskning på kognitive krav.

Fra analysen av de to ulike læreverkene kom det frem at oppgavene som krever lave kognitive krav utgjør halvparten, men den andre halvparten er oppgaver som krever høye kognitive krav. Av kategoriene var kategoriene PUS (prosedyre uten sammenheng) og PMS (prosedyre med sammenheng), særlig fremtredende i begge læreverkene. For det digitale læreverket var det en overvekt på PMS, og en stor andel av oppgaver falt også inn under den høyeste kategorien ÅGM (å gjøre matematikk). Utgangspunktet for oppgaver med høye kognitive krav ser derfor ut til å ha gode muligheter for læring i møte med eleven. Denne muligheten derimot blir kraftig insnevret av hvilke type svar oppgaver med programmering stiller elevene. Kun ca. 16 prosent av oppgavene i læreverket *Maximum* krevde at elevene skulle vurdere gyldigheten av løsningen sin (VGL).

For problemstillingen:

I hvilken grad utfordres elevene kognitivt i møte med digitale og analoge programmeringsoppgaver i læreverker produsert for fagfornyelsen?

Vil svare være at elevene i stor grad utfordres kognitivt i møte med programmering i de undersøkte læreverkene for matematikk. En fordeling av oppgaver mellom høyt og lavt kognitivt nivå, med overvekt på de høye nivåene gir grunnlag for en slik konklusjon.

Derimot vil jeg trekke frem hvilke type svar oppgavene krever av elevene og begrepet dybdelæring. I læreplanens overordnet del for LK20, er det spesifisert at elevene skal drive med dybdelæring. Det vil si at elevene blant annet skal utsettes for oppgaver som fremmer utforskning, resonnering og generalisering. Gjennom oppgavene utfordres elevene kognitivt, men når det kommer til svaret, krever et fåtall av oppgavene at elevene skal vurdere gyldigheten av den læringsprosessen som har funnet sted. For å

utvikle det Skemp kaller relasjonell forståelse, krever det en refleksjon omkring de læringsprosessene finner sted. Av denne grunn kan det tenkes at elevene i stor grad utfordres kognitivt, men at de ikke får full uttelling av oppgavene i forhold til de mulighetene som ligger i dem.

Det som ville vært interessant for videre forskning rundt kognitive krav og programmering, er å se hvordan temaet implementeres i klasserommet, og hvilke konsekvenser dette har for de kognitive nivåene oppgavene befinner seg på.

Litteraturliste

Bratholm, B. (2001). *Godkjenningsordningen for lærebøker 1889- 2001, en historisk gjennomgang*. Høgskolen i Vestfold - bibliotek. <http://www-bib.hive.no/tekster/hveskrift/notat/2001-05/not5-2001-02.html>

Cappelen Damm (u.å). *Matematikk 8-10 fra Cappelen damm*. Hentet 15. mai 2022 fra <https://www.cappelendammundervisning.no/verk/Matematikk%208-10%20fra%20Cappelen%20Damm-153429>

Charalambous, Delaney, S., Hsu, H.-Y., & Mesa, V. (2010). A Comparative Analysis of the Addition and Subtraction of Fractions in Textbooks from Three Countries. *Mathematical Thinking and Learning*, 12(2), 117–151. <https://doi.org/10.1080/10986060903460070>

Den nasjonale forskningsetiske komité samfunnsvitenskap og humaniora. (2021, 16. desember). *Forskningsetiske retningslinjer for samfunnsvitenskap og humaniora*. <https://www.forskningsetikk.no/retningslinjer/hum-sam/forskningsetiske-retningslinjer-for-samfunnsvitenskap-og-humaniora/>

Egeberg, G. Hultin, H. & Berge O. (2017). *Monitor skole 2016: Skolens digitale tilstand*. Senter for IKT i utdanningen. https://www.udir.no/globalassets/filer/tall-og-forskning/rapporter/2016/monitor_2016_bm_-_2._utgave.pdf

Gilje, Ø. Landfald, F. Ø. Ludvigsen, S. (2018). Dybdeløring – historisk bakgrunn og teoretisk tilnærming. *Bedre skole*, 2018(4.), 22-27. https://www.uv.uio.no/forskning/satsinger/fiks/kunnskapsbase/dybdelering/gilje_landfald_ludvigsen_bedre_skole_2018.pdf

Goodlad, J. I. et al. (1979). *Curriculum Inquiry: The Study of Curriculum Practice*. New York: McGraw-Hill Book Company

Gyldendal (u.å). *Maximum Smart Øving*. Hentet 15. mai 2022 fra <https://www.gyldendal.no/grs/maximum/10/maximum-8-10-smart-oving/p-10023667-no/>

Heimstad, C. & Strand, K. (2018). *Kognitive utfordringer i to norske lærebokserier fra ungdomsskolen – en mixed methods studie* [Masteroppgave, Universitetet i Tromsø]. UIT Munin. <https://munin.uit.no/handle/10037/13791>

Hodgson, J. Rønning, W., & Tomlinson, P. (2012). *Sammenheng mellom undervisning og læring: en studie av læreres praksis og deres tenkning under kunnskapsløftet: sluttrapport: Vol. nr. 4/2012* (s. 230). Nordlandsforskning. <https://www.udir.no/globalassets/filer/tall-og-forskning/rapporter/2012/smul.pdf>

Hofstad, B. & Liland, C. (2021). *En studie av algebraoppgaver i to lærebøker for 8. trinn basert på den nye læreplanen*. [Masteroppgave, Nord Universitet]. Nord Open. <https://hdl.handle.net/11250/2773989>

Hsieh, Hsiu-Fang & Shannon, Sarah. (2005). Three Approaches to Qualitative Content Analysis. *Qualitative health research*. 15. 1277-88. 10.1177/1049732305276687. https://www.researchgate.net/publication/7561647_Three_Approaches_to_Qualitative_Content_Analysis

Imsen. (2015). *Elevens verden: innføring i pedagogisk psykologi* (5. utg., s. 154). Universitetsforlaget.

Kari Anne Rødnes, & Øystein Gilje. (2018). Ti år med grunnleggende ferdigheter - hva vet vi, og hvor går vi? *Norsk pedagogisk tidsskrift*, 102(3), 201–213. <https://doi.org/10.18261/issn.1504-2987-2018-03-02>

Kaufmann, O. T. (2018). Stenseth, B. & Holone, H. (2018). Programmering i matematikkundervisning. I A. Norstein & F. Haara (Red.), *Matematikkundervisning i en digital verden* (s. 75-80). Cappelen Damm Akademisk.

Knudsen, V. S. (Red.), (2011). *Internasjonal forskning på læremidler: en kunnskapsstatus*. (Rapport 12/2003). Høgskolen i Vestfold. https://www.udir.no/globalassets/filer/tall-og-forskning/rapporter/2012/laremidler_hive.pdf

Kunnskapsdepartementet. (2019). *Læreplan i matematikk 1.-10. trinn (MAT01-05)*. Fastsett som forskrift. Læreplanverket for kunnskapsløftet 2020.
<https://data.udir.no/kl06/v201906/laereplaner-lk20/MAT01-05.pdf?lang=nob>

Meld. St. 28 (2015-2016). *Fag – Fordypning – Forståelse – En fornyelse av Kunnskapsløftet*. Kunnskapsdepartementet.
<https://www.regjeringen.no/no/dokumenter/meld.-st.-28-20152016/id2483955/>

Meld. St. 30 (2003-2004). *Kultur for læring*. Kunnskapsdepartementet.
<https://www.regjeringen.no/no/dokumenter/meld.-st.-28-20152016/id2483955/>

Mosvold, Reidar & Fauskanger, Janne. (2014). Innholdsanalysens muligheter i utdanningsforskning. *Norsk Pedagogisk Tidsskrift*. 98. 127–139. 10.18261/ISSN1504-2987-2014-02-07.
https://www.researchgate.net/publication/264553777_Innholdsanalysens_muligheter_i_utdanningsforskning

NOU 2013: 2. (2013). *Hindre for digital verdiskapning*. Fornyings-, administrasjons- og kirkedepartementet.
<https://www.regjeringen.no/contentassets/e2f0d5676e144305967f21011b715c16/no/pdf/s/nou201320130002000dddpdfs.pdf>

NOU 2015: 8. (2015). *Fremtidens skole – Fornyelse av fag og kompetanser*. Kunnskapsdepartementet. [regjeringen.no/no/dokumenter/nou-2015-8/id2417001/](https://www.regjeringen.no/no/dokumenter/nou-2015-8/id2417001/)

Prop. 44 (1999-2000). *Om lov om endringer i Lov av 17.juli 1998 nr.6*. Tilråding frå Kyrkje-, utdannings- og forskingsdepartementet.
<https://www.regjeringen.no/no/dokumenter/otprp-nr-44-1999-2000-/id586147/sec4>

Rødnes, K. & Gilje, Ø. (2016). *Grunnleggende ferdigheter: På tvers eller i fag?* Universitetet i Oslo. https://www.uv.uio.no/iped/forskning/prosjekter/ark-app/rodnes_gilje_ark_app_grf_2016.pdf

Sevik, K. et al. (2016). Notat nr.2: Programmering i skolen. Senter for IKT i utdanningen. Hentet fra <https://www.udir.no/kvalitet-og-kompetanse/profesjonsfaglig-digital-kompetanse/notat-om-programmering-i-skolen/>

Silverman, D. (2014). *Interpreting qualitative data* (5 utg.). SAGE.

Skemp, R. R. (1976). Relational understanding and instrumental understanding. *Mathematics teaching*, 77(1), 20-26.

<http://www.davidtall.com/skemp/pdfs/instrumental-relational.pdf>

Skjelbred, D. (2003). *Valg, vurdering og kvalitetsutvikling av lærebøker og andre læremidler*. (Rapport 12/2003). Høgskolen i Vestfold. <http://www-bib.hive.no/tekster/hveskrift/rapport/2003-12/rapport12.pdf>

Smith, M. S., & Stein, M. K. (1998). Selecting and Creating Mathematical Tasks: From Research to Practice. *Mathematics Teaching in the Middle School*, 3(5), 344–350.

<http://mathedseminar.pbworks.com/w/file/fetch/92864991/Smith%20and%20Stein%20-%201998%20-%20Selecting%20and%20Creating%20Mathematical%20Tasks%20From%20Re.pdf>

Steen, J. (2019). *Kognitive krav og forståelse av funksjoner*. [Masteroppgave,

Universitetet i Agder]. UIA. AURA. <https://uia.brage.unit.no/uia-xmlui/bitstream/handle/11250/2623625/Steen%2C%20J%20%20Lin%20%20%20B8kken.pdf?sequence=1>

Stein M. K., Smith, M. S., Henningsen, M. A., & Silver, E. A. (2000). *Implementing standards-based mathematics instruction: A casebook for professional development*, s. 16. New York: Teachers College Press.

<https://www.semanticscholar.org/paper/Implementing-Standards-Based-Mathematics-A-Casebook-Stein-Smith/deac6a43706ea11bdd9fb07a73b54a08f0c114fb>

Stein, M. K., & Smith, M. S. (1998). Mathematical Tasks as a Framework for Reflection: From Research to Practice, *Mathematics Teaching in the Middle*

School, 3(4), 268-275. Retrieved May 26, 2022,
from <https://pubs.nctm.org/view/journals/mtms/3/4/article-p268.xml>

Stein, M.K., Smith, M., Henningsen, M. & Silver, E. (2009). *Implementing Standards-Based Mathematics Instruction. A Casebook for Professional Development*. (2.utg.) New York: Teachers College, Columbia University.

Tangen, P. (2017). *Kognitive krav i arbeidet med digitale matematikkoppgaver*. [Masteroppgave, Norges teknisk-naturvitenskapelige universitet]. NTNU. NTNU Open. <https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/2454612>

Utdanningsdirektoratet. (2017). *Overordnet del – kompetanse i fagene*. Utdanningsdirektoratet. <https://www.udir.no/lk20/overordnet-del/prinsipper-for-laring-utvikling-og-danning/kompetanse-i-fagene/?lang=nob&curriculum-resources=true>

Utdanningsdirektoratet. (2019). *Dybdeløring*. Utdanningsdirektoratet. <http://www.udir.no/laring-og-trivsel/dybdelaring/>

Utdanningsdirektoratet. (2021, 12. mars). *Læremidler og læringsteknologi i skole og opplæring*. <https://www.udir.no/om-udir/tilskudd-og-prosjektmidler/tilskudd-til-laremidler/begrepsavklaring-skole/>

Utdanningsdirektoratet. (2021, 31. august). *Hvordan ta i bruk nye læreplan*. <https://www.udir.no/laring-og-trivsel/lareplanverket/stotte/hvordan-ta-i-bruk-lareplanen/>

Valenta, A. (2016). Kognitive krav i matematikkoppgaver. *Matematikksenteret: Nasjonalt senter for matematikk i opplæringen*. https://www.matematikksenteret.no/sites/default/files/media/filer/MAM/Valenta%20Kognitive%20krav%20i%20matematikkoppgaver_0.pdf

Wæge, K., & Nosrati, M. (2015). Sentrale kjennetegn på god læring og undervisning i matematikk. *Matematikksenteret: Nasjonalt senter for matematikk i opplæringen*.

<https://utdanningsforskning.no/artikler/2015/sentrale-kjennetegn-pa-god-laring-og-undervisning-i-matematikk/>

Aasen, Møller, Rye, Ottesen, Prøitz, & Hertzberg. (2012). *Kunnskapsløftet som styringsreform - et løft eller et løfte? Forvaltningsnivåenes og institusjonenes rolle i implementeringen av reformen*. NIFU. <http://hdl.handle.net/11250/280885>

Oversikt over tabeller og figurer

Læringsmål – Maximum vs. Campus Matte

<ol style="list-style-type: none">1. «å utforske algoritmer med og uten programmering»2. «å bruke ulike digitale verktøy til å utforske, analysere og behandle funksjoner»3. «å forklare og bruke formler i utregninger og programmering»4. «du skal kunne utforske sannsynlighet ved å simulere hendelser med og uten digitale verktøy»	<ol style="list-style-type: none">1. «Jeg kan lese og forstå programmer som utfører matematiske beregninger»2. «Jeg kan skrive programmer som utfører matematiske beregninger»3. «Jeg kan vurdere om programmer som utfører matematiske beregninger fungerer feilfritt»4. «Jeg kan lese og forstå program som benytter «gjett-og-sjekk»-metoden»5. «Jeg kan lage dataprogrammer som løser problemer ved hjelp av «gjett-og-sjekk» metoden»6. «Jeg forbedre mine egne algoritmer slik at programmene mine finner løsningen raskere»7. «Jeg kan lese og forstå programmer som regner ut sannsynligheter»8. «Jeg kan skrive programmer som regner ut sannsynligheter»9. «Jeg kan finne det største eller elngste elementet i en liste ved hjelp av programmering»10. «Jeg kan finne det minste eller korteste elementet i en liste ved hjelp av programmering»11. «Jeg kan finne antall elementer som tilfredsstill et bestemt krav ved hjelp av programmering»
---	--

	<p>12. «Jeg kan lese og forstå programmer som anslår sannsynligheter fra forsøk eller simuleringer»</p> <p>13. «Jeg kan skrive programmer som anslår sannsynligheter fra forsøk og simuleringer»</p> <p>14. «Jeg kan bruke programmering til å bestemme antall elementer i figur nr 200 i et geometrisk mønster»</p> <p>15. «Jeg kan bruke programmering til å bestemme summen av de første 200 tallene i et tallmønster».</p> <p>16. «Jeg kan finne funksjonene i et program»</p> <p>17. «Jeg kan forklare hva funksjonene i et program gjør»</p>
--	--

Tabell 28: Læringsmål – Maximum vs. Campus Matte.

Maximum 8. trinn	
Kombinasjoner	Antall oppgaver
M-L+KS	3
M-L+SMF	0
M-L+VGL	0
PUS-L+KS	10
PUS-L+SMF	0
PUS-L+VGL	0
PMS-H+KS	2
PMS-H+SMF	2
PMS-H+VGL	6
ÅGM-H+KS	0
ÅGM-H+SMF	0

ÅGM-H+VGL	0
Sum	

Tabell 36: Kognitive krav + Type svar for Maximum: 8.trinn.

Maximum 9. trinn	
Kombinasjoner	Antall oppgaver
M-L+KS	2
M-L+SMF	0
M-L+VGL	0
PUS-L+KS	10
PUS-L+SMF	1
PUS-L+VGL	0
PMS-H+KS	4
PMS-H+SMF	3
PMS-H+VGL	1
ÅGM-H+KS	0
ÅGM-H+SMF	0
ÅGM-H+VGL	2
Sum	

Tabell 37: Kognitive krav + Type svar for Maximum: 9.trinn.

Maximum 10. trinn	
Kombinasjoner	Antall oppgaver
M-L+KS	2
M-L+SMF	0
M-L+VGL	0
PUS-L+KS	3
PUS-L+SMF	0
PUS-L+VGL	0
PMS-H+KS	4
PMS-H+SMF	5
PMS-H+VGL	0

ÅGM-H+KS	0
ÅGM-H+SMF	0
ÅGM-H+VGL	1
Sum	

Tabell 38: Kognitive krav + Type svar for Maximum: 10.trinn.

Campus Matte 8. trinn	
Kombinasjoner	Antall oppgaver
M-L+KS	0
M-L+SMF	6
M-L+VGL	0
PUS-L+KS	10
PUS-L+SMF	4
PUS-L+VGL	0
PMS-H+KS	6
PMS-H+SMF	9
PMS-H+VGL	0
ÅGM-H+KS	1
ÅGM-H+SMF	8
ÅGM-H+VGL	0
Sum	

Tabell 39: Kognitive krav + Type svar for Campus Matte: 8.trinn.

Campus Matte 9. trinn	
Kombinasjoner	Antall oppgaver
M-L+KS	0
M-L+SMF	0
M-L+VGL	0
PUS-L+KS	7
PUS-L+SMF	40
PUS-L+VGL	0
PMS-H+KS	7

PMS-H+SMF	17
PMS-H+VGL	0
ÅGM-H+KS	2
ÅGM-H+SMF	15
ÅGM-H+VGL	0
Sum	

Tabell 40: Kognitive krav + Type svar for Campus Matte: 9.trinn.

Campus Matte 10. trinn	
Kombinasjoner	Antall oppgaver
M-L+KS	0
M-L+SMF	0
M-L+VGL	0
PUS-L+KS	1
PUS-L+SMF	2
PUS-L+VGL	0
PMS-H+KS	0
PMS-H+SMF	10
PMS-H+VGL	0
ÅGM-H+KS	0
ÅGM-H+SMF	0
ÅGM-H+VGL	0
Sum	0

Tabell 41: Kognitive krav + Type svar for Campus Matte: 10.trinn.

Maximum 8. – 2. utgave

Maximum 8. 2. utgave							
Oppgaver	Kognitive krav				Type svar		
	M-L	PUS-L	PMS-H	ÅGM - H	KS	SMF	VGL
Kapittel 1 – Tall og tallregning							
Ingen oppgaver funnet.							
Kapittel 2 - Algebra							

2.56			X				X
2.57		X			X		
2.58			X			X	
2.60		X			X		
2.61			X				X
2.62			X			X	
2.63	X				X		
2.64			X				X
2.65		X			X		
2.72	X				X		
2.78			X				X
Kapittel 3 - Funksjoner							
3.27		X			X		
3.65		X			X		
3.75			X				X
3.80			X				X
Kapittel 4 – Likninger og formler							
4.53	X				X		
4.54		X			X		
4.69			X		X		
4.77		X			X		
4.79		X			X		
4.93			X		X		
4.97		X			X		
4.99		X			X		
sum	3	10	10	0	15	2	6

Maximum 9. – 2. utgave

Maximum 9. 2. utgave							
Oppgaver	Kognitive krav				Type svar		
	M-L	PUS-L	PMS-H	ÅGM - H	KS	SMF	VGL
Kapittel 1 – Statistikk							
1.54		X				X	
1.55			X		X		

1.56			X			X	
1.57	X				X		
1.58	X				X		
1.59		X			X		
1.95		X			X		
Kapittel 2 - Sannsynlighet							
2.29			X		X		
2.34		X			X		
2.35			X			X	
2.36		X			X		
2.48			X				X
2.52		X			X		
2.72			X		X		
2.74		X			X		
Kapittel 3 - Plangeometri							
3.10		X			X		
3.15		X			X		
3.18		X			X		
3.19				X			X
3.41		X			X		
3.105			X		X		
Kapittel 4 – Romgeometri							
4.7			X			X	
4.72				X			X
sum	2	11	8	2	16	4	3

Maximum 10. – 2. utgave

Maximum 10. 2. utgave							
Oppgaver	Kognitive krav				Type svar		
	M-L	PUS-L	PMS-H	ÅGM - H	KS	SMF	VGL
Kapittel 1 – Likninger og algebra							
1.15	X				X		
1.84			X		X		
Kapittel 2 - Funksjoner							

2.51			X		X		
2.52		X			X		
2.53		X			X		
2.54			X		X		
2.55			X			X	
2.56			X			X	
2.83	X				X		
2.93			X			X	
Kapittel 3 - Økonomi							
3.114		X			X		
Kapittel 4 – Se flere sammenhenger							
4.13			X		X		
4.24			X			X	
4.29			X			X	
4.30				X			X
sum	2	3	9	1	9	5	1

Campus Matte – 8. trinn

Campus Matte 8. trinn							
Oppgaver	Kognitive krav				Type svar		
	M-L	PUS-L	PMS-H	ÅGM - H	KS	SMF	VGL
Kalkulatorer (8. trinn)							
1a		X			X		
1b				X		X	
1c	X					X	
1d	X					X	
1e	X					X	
2a			X			X	
2b				X		X	
2c	X					X	
2d	X					X	
2e				X		X	
3a				X		X	

3b			X			X	
3c			X			X	
3d			X			X	
4a				X		X	
4b			X			X	
4c			X			X	
5a		X				X	
5b		X				X	
5c		X				X	
6a			X			X	
6b			X			X	
6c	X					X	
Problemløsning (8. trinn)							
1a			X			X	
1b				X		X	
1c		X				X	
1d			X		X		
1e			X		X		
1f			X		X		
2a				X	X		
2b		X			X		
2c		X			X		
2d		X			X		
3a		X			X		
3b			X		X		
3c		X			X		
4a		X			X		
4b		X			X		
4c		X			X		
4d		X			X		
5a				X		X	
5b				X		X	
6a			X		X		

6b			X		X		
Sum	6	14	15	9	17	27	0

Campus Matte – 9. trinn

Campus Matte 9. trinn							
Oppgaver	Kognitive krav				Type svar		
	M-L	PUS-L	PMS-H	ÅGM - H	KS	SMF	VGL
Sannsynlighetsregning (9. trinn)							
1a			X			X	
1b				X		X	
1c				X		X	
1d				X		X	
2a			X		X		
2b				X	X		
2c		X			X		
2d		X			X		
3b			X			X	
3c		X				X	
3e			X			X	
3f		X				X	
4a				X		X	
4b				X	X		
4c		X			X		
5a			X		X		
5b		X			X		
5c			X		X		
5d			X		X		
6a			X		X		
6b			X		X		
6c			X		X		
6d		X					
7b				X		X	

7c			X			X	
Antall, minst og størst (9. trinn)							
1a			X			X	
1b				X		X	
1c		X				X	
1d		X				X	
1e		X				X	
2a		X				X	
2b		X				X	
2c		X				X	
3a				X		X	
3b				X		X	
4a		X			X		
4b		X			X		
5a		X				X	
5b		X				X	
5c		X				X	
5d		X				X	
6a		X				X	
6b		X				X	
6c		X				X	
7a		X				X	
7b		X				X	
7c		X				X	
7d		X				X	
7e		X				X	
8a		X				X	
8b		X				X	
9a		X				X	
9b		X				X	
10a		X				X	
10b		X				X	
10c			X			X	

11		X			X	
12a		X			X	
12b		X			X	
Forsøk og simulering (9. trinn)						
1b			X		X	
1c			X		X	
2b			X		X	
2c		X			X	
2d		X			X	
2e			X		X	
3b			X		X	
3c		X			X	
4b				X	X	
4c			X		X	
4d			X		X	
5a		X			X	
5c			X		X	
6a		X			X	
6b		X			X	
6c		X			X	
6d		X			X	
6e		X			X	
6f			X		X	
Geometriske mønstre og tallmønstre (9. trinn)						
1c				X	X	
1e				X	X	
2c			X		X	
2e		X			X	
3c				X	X	
3d				X	X	
4b			X		X	
5c				X	X	
6c		X			X	

7				X		X	
Sum	0	47	24	17	16	72	0

Campus Matte – 10. trinn

Campus Matte 10. trinn							
Oppgaver	Kognitive krav				Type svar		
	M-L	PUS-L	PMS-H	ÅGM - H	KS	SMF	VGL
Funksjoner (10. trinn)							
1			X			X	
2			X			X	
3				X		X	
4				X		X	
5				X		X	
6				X		X	
7			X			X	
8			X			X	
9				X		X	
10				X		X	
Matematiske problemer (10. trinn)							
1a			X			X	
1b			X			X	
1c		X				X	
2a			X			X	
3a			X			X	
3d			X			X	
4a			X			X	
4b				X		X	
4c				X		X	
5a				X		X	
5b		X				X	
6		X			X		
7a				X		X	

7b				X		X	
8				X		X	
sum	0	3	10	12	1	24	0