

MASTEROPPGAVE

«Ok, da prøver vi ...»

Elevers matematiske kompetanse gjennom
programmering

Students' mathematical competence through
programming

Tonje Lindberg

Master i undervisningsvitenskap med fordypning i matematikk
Fakultet for lærerutdanning, kultur og idrett (FLKI)

15. mai 2019

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

kilder som er brukt i arbeidet er oppgitt, jf. *Forskrift om studium og eksamen ved Høgskulen på Vestlandet, § 10.*

FORORD

Etter fem år i lærerutdanningen ved Høgskolen på Vestlandet og en fullført masteroppgave, er jeg klar for å sette ny lærdom ut i livet som lektor i grunnskolen. Det har vært fem lærerike år med både opp- og nedturer. Arbeidet med masteroppgaven har vært den mest utfordrende oppgaven, men arbeidet med denne har gitt meg kunnskap og erfaringer jeg ikke ville vært foruten. Det har vært en utrolig rik læringsprosess hvor jeg har fått utviklet meg både faglig og personlig.

Jeg er veldig takknemlig for all oppmuntring og støtte jeg har fått fra medstudenter, venner, kollegaer, familie, lærere og veiledere. Først vil jeg takke min hovedveileder Ragnhild Hansen og med-veileder Inge Olav Hauge for gode råd, konstruktiv kritikk og støttende tilbakemeldinger i arbeidet med masteroppgaven. Takk til Oda Åbø Pettersen som har vært en trygg og god studievenninne gjennom hele lærerutdanningen, og en dyktig samarbeidspartner i forbindelse med dette prosjektet.

Uten velvillighet fra lærere, elever, foresatte og frivillige i Kodeklubben hadde dette prosjektet ikke vært gjennomførbart. Jeg er derfor takknemlig for den gjestfrihet og velvillighet jeg møtte fra dem, og at de har bidratt til å gjøre denne studien både spennende, lærerik og interessant.

Det er mange jeg kan takke for hjelp og støtte, men jeg vil særlig takke gjengen på lesesalen for blant annet oppmuntrende lunsj- og kaffepauser og interessante diskusjoner. Uten mamma og pappa, mine trofaste mentorer og støttespillere, hadde jeg heller aldri kommet meg igjennom de to siste årene. Jeg er veldig takknemlig for den støtte, oppmuntring og tålmodighet dere har vist meg gjennom denne tiden. Også takk til min lillesøster Tabita, for å alltid være der for meg. Til slutt vil jeg takke min kusine Anja for oppmuntrende samtaler og for at du tok deg tid til å korrekturlese oppgaven min. Jeg ser nå tilbake på studietiden som en lærerik og fin tid, og gleder meg nå til å møte den nye hverdagen som venter.

Tonje Lindberg

15. mai 2019

SAMMENDRAG

Samfunnet er i stadig endring, og utdanningen må følge med i samfunnets utvikling. I 2020 vil nye læreplaner bli tatt i bruk i norske skoler, med fornyelse av fagene som skal ruste elevene for fremtiden. En av fornyelsene er at programmering blir en del av matematikkpensumet. I denne studien er det blitt undersøkt hvordan programmering kan fremme matematisk kompetanse. Studien er avgrenset ved følgende to forskningsspørsmål: 1) *Hvordan kommer matematisk kompetanse til uttrykk i elevers samtaler når de programmerer?* 2) *Hvordan konstruerer elever matematisk kunnskap når de programmerer?*

Som bakteppe for studien er det benyttet teori jeg anser som relevant til å undersøke oppgavens forskningsspørsmål. Basert på Kilpatrick, Swafford & Findells (2001) rammeverk om matematiske komponenter og Niss & Højgaard Jensens (2002) inndeling av matematisk kompetanse, har jeg utformet et rammeverk som er benyttet til å belyse det første forskningsspørsmålet. Seymour Papert (1991, 1993) og konstruksjonismens læringssyn er bruk som teoretisk grunnlag for det andre forskningsspørsmålet, om hvordan elever konstruerer matematisk kunnskap når de programmerer. Matematisk kunnskap er i denne sammenheng sett på som en del av en helhetlig matematisk kompetanse.

Studien har tatt utgangspunkt i fire elevpar fra 7. trinn sitt arbeid med å programmere en femkant i Scratch. Metoden for analysen av elevsamtalene har kvalitativ hermeneutisk tilnærming. Analysen har vist at programmering kan bidra positivt til å fremme matematisk kompetanse. Elevene har sammen kommet frem til en løsning på oppgaven gjennom en lærerik prosess, der matematisk kompetanse er fremmet på ulike områder. Samarbeid om slike matematiske oppgaver har gitt innblikk i hvordan elever konstruerer matematisk kunnskap. Samarbeid, kreativitet, resonnering, engasjement, prøving og feiling er noen av kvalitetene som har preget læringsprosessen.

Formålet med studien er å bidra til økt kunnskap om hvordan programmering kan fremme elevers matematiske kompetanse. Ønsket er at studien vil være nyttig kunnskap for lærere og andre roller i utdanningssektoren. Jeg håper også det vil bidra til større interesse for hvordan programmering kan styrke matematikkundervisningen. Arbeidet med studien har gitt meg større innblikk i hvordan jeg som fremtidig mattelærer kan kombinere matematikk og programmering på en lærerik måte. Jeg har også fått en dypere kjennskap til forskning på feltet.

ABSTRACT

Society is constantly changing, and education must keep up with its development. By 2020, new curricula will be introduced in Norwegian schools, with development and revisions of the subjects that will prepare students for the future. One of the developments is that programming becomes a part of the mathematics curriculum. In this study, it has been investigated how programming can promote mathematical competence. The study is limited by the following two research questions: 1) *How does mathematical competence express in pupils' conversations when they program?* 2) *How do students construct mathematical knowledge when programming?*

As a backdrop for the study, theories I consider relevant to investigating the research questions have been used. Based on Kilpatrick, Swafford & Findell's (2001) framework on mathematical components and Niss & Højgaard Jensen's (2002) division of mathematical competence, I have designed a framework that has been used to elucidate the first research question. Seymour Papert's (1991, 1993) learning perspective of constructionism is used as a theoretical basis for the second research question, about how students construct mathematical knowledge when programming. In this context, mathematical knowledge is regarded as part of a comprehensive mathematical competence.

The study has been based on four pairs of students at grade 7, and their work on programming a pentagon in Scratch. The method for analyzing the student conversation has a qualitative hermeneutic approach. The analysis has proven that programming can contribute positively to promoting mathematical competence. Together, the pupils have come up with a solution to the task through an educational process, in which mathematical competence has been promoted in different areas. Collaboration on such mathematical tasks has provided insight on how students construct mathematical knowledge. Collaboration, creativity, reasoning, commitment, trial and error are some of the qualities that have characterized the learning process.

The purpose of this study is to contribute to increased knowledge about how programming can promote pupils' mathematical competence. The aim is that the study will be of useful knowledge for teachers and other people in the education sector. I also hope that it will contribute to greater interest in how programming can strengthen mathematics teaching. The work on the study has given me a greater insight into how I, as a future math teacher, can combine mathematics and

M120UND509, kandidatnummer: 107, 15. mai 2019

programming in an educational way. I have also gained a deeper knowledge of research in this field.

INNHold

| | |
|---|-----|
| Forord | II |
| Sammendrag | III |
| Abstract | IV |
| 1. Innledning | 1 |
| 1.2. Bakgrunn for valg av tema | 2 |
| 1.2.1. Hvorfor programmering i skolen? | 3 |
| 1.2.2. Nou 2015:8 – fremtidens skole..... | 3 |
| 1.2.3. Endringer for fremtiden - Nye læreplaner | 4 |
| 1.3. Avgrensning av tema og forskningsspørsmål..... | 5 |
| 1.3.1. Del av et større prosjekt..... | 7 |
| 1.4. Begrepsavklaring..... | 7 |
| 1.4.1. Matematisk kompetanse | 7 |
| 1.4.2. Programmering | 8 |
| 1.4.3. Scratch – blokkbasert programmering..... | 9 |
| 1.5. Tidligere forskning | 10 |
| 1.5.1. Tidligere forskning på programmering og matematikk..... | 10 |
| 1.5.2. Tidligere forskning på Scratch | 12 |
| 1.6. Oppgavens struktur..... | 13 |
| 2. Teori | 15 |
| 2.2. Programmering og matematikk | 15 |
| 2.2.1. Matematikk..... | 15 |
| 2.2.2. Hvorfor programmering i matematikk?..... | 16 |
| 2.2.3. Algoritmisk tenkning..... | 17 |
| 2.3. Et nytt sammensatt rammeverk | 18 |
| 2.3.1. Kilpatrick, Swafford & Findell (2001) – fem tråder av matematisk kyndighet | 18 |
| 2.3.2. Niss & Højgaard Jensen (2002) – åtte matematiske kompetanser | 22 |
| 2.3.3. Sammenheng mellom Kilpatrick, Swafford & Findell (2001) og Niss & Højgaard Jensen (2002)..... | 26 |
| 2.4. Papert og konstruksjonismen..... | 28 |
| 3. Metode | 31 |
| 3.2. Metodisk tilnærming | 31 |
| 3.3. Datainnsamling..... | 33 |
| 3.3.1. Utvalg og kontekst..... | 34 |

| | | |
|-----------|--|-----------|
| 3.3.2. | Utforming av oppgaver i Scratch..... | 35 |
| 3.3.3. | Dokumentasjon av datainnsamlingen..... | 35 |
| 3.3.4. | Gjennomføring av datainnsamling | 36 |
| 3.4. | Analyseverktøy..... | 38 |
| 3.4.1. | Kilpatrick, Swafford & Findell (2001) og Niss & Højgaard Jensen (2002) som analyseverktøy..... | 39 |
| 3.4.2. | Papert og konstruksjonismen som analyseverktøy | 41 |
| 3.4.3. | Analyseprosessen | 41 |
| 3.5. | Etiske overveielser | 46 |
| 3.5.1. | Forskning på barn | 47 |
| 3.5.2. | Samtykke | 47 |
| 3.6. | Reliabilitet og validitet | 47 |
| 4. | Resultat og analyse | 50 |
| 4.2. | resultater fra koding av elevenes samtaler..... | 51 |
| 4.2.1. | Resultater presentert i prosent | 51 |
| 4.2.2. | Resultater presentert i antall koder | 53 |
| 4.2.3. | Felleskoden [K B S R]..... | 54 |
| 4.2.4. | De fem matematiske komponentene..... | 55 |
| 4.3. | Matematisk kompetanse uttrykt i elevsamtaler | 55 |
| 4.3.1. | Gruppe 4..... | 56 |
| 4.3.2. | Gruppe 2..... | 61 |
| 4.4. | Konstruksjon av matematisk kunnskap | 69 |
| 4.4.1. | Gruppe 3..... | 69 |
| 5. | Diskusjon..... | 76 |
| 5.2. | Hva kan analysen fortelle om elevenes matematiske kompetanse? | 76 |
| 5.1.1. | Er ikke elevene engasjert i arbeidet? | 79 |
| 5.2. | Learning by making..... | 80 |
| 5.3. | Kompetanse og ferdigheter for fremtiden | 82 |
| 5.4. | Fagfornyelsen – matematikkfaget i endring | 84 |
| 5.4.1. | Algoritmisk tenkning..... | 84 |
| 5.5. | Muligheter og begrensninger ved bruk av Scratch..... | 85 |
| 6. | Avslutning | 87 |
| 6.2. | Avsluttende refleksjoner..... | 87 |
| 6.3. | Pågående forskning og veien videre..... | 88 |
| | Litteraturliste..... | 90 |

VEDLEGG

| | |
|--|-----|
| Vedlegg I – Oppgaver til pilot..... | 96 |
| Vedlegg II – Oppgaver til Opplegget | 97 |
| Vedlegg III – Samtykkeskjema barn | 98 |
| Vedlegg IV – Samtykkeskjema foresatte | 99 |
| Vedlegg V – Tegn og koder i transkripsjoner | 102 |
| Vedlegg VI – Transkripsjon – femkant, gruppe 1 | 103 |
| Vedlegg VII – Transkripsjon – femkant, gruppe 2 | 106 |
| Vedlegg VIII – Transkripsjon – femkant, gruppe 3 | 109 |
| Vedlegg IX – transkripsjon – femkant, gruppe 4 | 112 |

FIGURER

| | |
|---|----|
| Figur 1: Skjerm bilde av Scratch med egne beskrivelser av programmet..... | 9 |
| Figur 2: Modell hentet fra s. 117 i Kilpatrick, Swafford & Findell (2001) | 19 |
| Figur 3: Modell av matematiske kompetanser (Niss & Højgaard Jensen, 2002, s. 45) | 23 |
| Figur 4: En interaktiv modell for analyse av kvalitative data. Kilde: Miles & Huberman (1994) i Ringdal (2018, s. 252)..... | 42 |
| Figur 5: Utklipp av bilde-i-bilde-video fra gjennomføring av prosjektet | 43 |
| Figur 6: Utdrag fra egen transkripsjon av datamaterialet..... | 44 |
| Figur 7: Utklipp fra transkripsjon med koder..... | 45 |
| Figur 8: Tabell over koder identifisert i elevsamtaler | 51 |
| Figur 9: Søylediagrammer som viser koding av matematiske komponenter i prosent | 52 |
| Figur 10: Søylediagram som viser koding av matematiske komponenter i antall | 53 |
| Figur 11: Resultater av koden [K B S R] i elevsamtalene | 54 |
| Figur 12: Utklipp fra tegneark - gruppe 4 | 56 |
| Figur 13: Utklipp fra Scratch - gruppe 4 | 57 |
| Figur 14: Utklipp fra tegneark - gruppe 2 | 62 |
| Figur 15: Utklipp fra tegneark - gruppe 2 | 62 |
| Figur 16: Utklipp fra Scratch - gruppe 2 | 66 |
| Figur 17: Utklipp fra Scratch - gruppe 2 | 67 |
| Figur 18: Utklipp fra Scratch - gruppe 3 | 71 |
| Figur 19: Utklipp fra Scratch - gruppe 3 | 71 |
| Figur 20: Flere utklipp fra Scratch - gruppe 3 (se også transkripsjoner Vedlegg VIII)..... | 73 |
| Figur 21: Utklipp fra Scratch - gruppe 3 | 74 |
| Figur 22: Hanne og Vera sin endelige femkant som er programmert i Scratch..... | 75 |

1. INNLEDNING

Elever som begynner på skolen i 2019 vil sannsynligvis være ferdigutdannet og ute i arbeidslivet i løpet av 2030-årene, og slik være aktører i samfunn- og arbeidsliv til vi nærmer oss det neste århundre. Tidligere kommunal- og moderniseringsminister Jan Tore Sanner nevnte i innlegget *Vekst og arbeidsplasser i teknologiens tidsalder* fra 2017 hvilke endringer vi som samfunn står overfor (Kommunal- og moderniseringsdepartementet, 2017, s. 31). Her trekker han frem en industrimelding fra 1980 med formuleringen:

«Den teknologiske utviklingen har ytterligere skutt fart. Automatisering og anvendelse av avansert datateknikk på stadig flere områder har skapt nye muligheter og utfordringer».

(Kommunal- og moderniseringsdepartementet, 2017)

Han påpeker videre at de trolig hadde rett den gangen, men om den teknologiske utviklingen hadde «skutt fart» i 1980, går den enda raskere i dag. Vi kan flere steder både lese og høre om at dagens elever trolig vil få jobb på arbeidsplasser som enda ikke finnes. Denne utviklingen gjør at det må fokuseres på ferdigheter, kunnskap og kompetanse som vil være relevant i dag, og ikke minst i fremtiden.

Samfunnet er i stadig utvikling, noe som krever at skolen følger med i denne forandringen. For å forberede dagens unge til en fremtid vi enda ikke vet hva vil innebære, er ulike tiltak satt i gang. Blant disse finner vi flere rammeverk som tar for seg ferdigheter for det 21. århundre, såkalte *Frameworks for 21st Century Skills* (Dede, 2010). Ett av disse er *Framework for 21st Century Learning*, et amerikansk rammeverk som er utviklet av The Partnership for 21st Century Skills med innspill fra andre utdanningsekspertene og forretningsledere (Dede, 2010; Partnership for 21st Century Skills, 2019, u.å.). Her presenteres en visjon om hvilke ferdigheter, kunnskap, ekspertise og støttesystemer elever vil kunne trenge for å lykkes i arbeid og hverdagsliv i det 21. århundre. Dette rammeverket utpekes fordi det har hatt innvirkning på flere lignende rammeverk som har kommet i senere tid. Innholdet gjenspeiles derfor flere steder hvor *21st Century Skills* eller *ferdigheter for det 21. århundre* omtales. Kjerneferdigheter som går igjen i slike rammeverk er anvendelse av teknikk og informasjon, kommunikasjon, samarbeid, kreativitet, kritisk tenking og problemløsning (Van Laar, Van Deursen, Van Dijk &

De Haan, 2017). De ferdighetene som fremheves i disse rammeverkene er selvsagt ferdigheter som også var relevante i det 20. århundre. Forskjellen er at de i dag og i fremtiden vil ha et fornyet fokus med et innhold og formål som påvirkes av nye behov og forhold ettersom samfunnet er i endring (Dede, 2010). Her har den teknologiske utviklingen stor innvirkning på hva som vil være relevant og nødvendig læring i både dagens og fremtidens samfunn. Det er likevel viktig å påpeke at kjerneferdighetene for det 21. århundre strekker seg utover det som har med informasjons- og kommunikasjonsteknologi (IKT) og annen teknologi å gjøre.

Regjeringen la i 2017 frem en ny digitaliseringsstrategi for grunnsopplæringen, som et tiltak for å gi elever opplæring i teknologi (Kunnskapsdepartementet, 2017). Som ett av flere tiltak er algoritmisk tenkning og programmering lagt frem som et forslag. Utdanningsdirektoratet la i 2016 frem rapporten *Teknologi og programmering for alle* (Utdanningsdirektoratet, 2016). Her anbefaler den utnevnte ekspertgruppen til Utdanningsdirektoratet at teknologi og programmering bør bli et praktisk fag i grunnsopplæringen. Programmering blir sett på som sentral kunnskap for å tilegne seg kunnskap om hvordan digital teknologi fungerer. En slik kunnskap vil ha betydning for utviklingen i et digitalt samfunn. Det er imidlertid bestemt at digital teknologi ikke skal utgjøre et eget fag i skolen, på tross av anbefalingene som ble fremmet i Utdanningsdirektoratet sin rapport (Utdanningsdirektoratet, 2016, s. 50). Programmering er nå fastsatt som ett av de nye elementene i matematikk gjennom Fagfornyelsen (Utdanningsdirektoratet, 2018a). Programmering er nå fastsatt som ett av de nye elementene i matematikk gjennom Fagfornyelsen og ikke i et nytt teknologifag i skolen, fordi Utdanningsdirektoratet ikke ønsker å etablere et nytt obligatorisk skolefag (Kjerneelementgruppen for matemtaikk, 2017).

1.2. BAKGRUNN FOR VALG AV TEMA

Endringer for fremtiden og nye elementer i matematikkfaget i skolen har det siste året fanget min interesse. Videre i oppgaven vil jeg rette fokuset mot programmering i skolen og matematikkfaget, og hvordan dette vil kunne virke inn på fremtidens skole. Som bakgrunn for valg av tema har jeg sett på hvorfor programmering skal inn i skolen og har fått plass i matematikkfaget. Relevant for denne fornyelsen er NOU-utredningen NOU 2015: 8 (2015) og de nye læreplanene med endringer for fremtiden (Utdanningsdirektoratet, 2017). Jeg har derfor

kort gjort rede for bakgrunn for valg av tema, før jeg avgrensner av tema og presenterer oppgavens problemstilling og forskningsspørsmål.

1.2.1. HVORFOR PROGRAMMERING I SKOLEN?

Vi beveger oss nå inn i den fjerde industrielle revolusjon. Denne revolusjonen er preget av en sammensmeltning av teknologier der linjene mellom det fysiske, digitale og biologiske blir mer uklare (Schwab, 2016, s. 12). For å gripe mulighetene og å kunne håndtere nye utfordringer kreves det et samfunn som er engasjert i både utvikling, bruk og styring av teknologi ("Civil Society in the Fourth Industrial Revolution: Preparation and Response," 2019).

Ideen om å integrere programmering i skolen er imidlertid ikke ny. Allerede på 1960-tallet ble programmeringsspråket LOGO utviklet med tanke på utdanning. Tanken om å lære barn programmering var i stor grad basert på ideer av Seymour Papert. Barn skulle bruke datamaskiner som et verktøy for å utvikle kreativitet, innovasjon og konkretisering av algoritmisk tenkning. Dette skulle også gjøre elevene til bedre problemløsere (Papert, 1993; Senter for IKT i utdanningen, 2016). Papert sine ideer medførte imidlertid ikke til store endringer i utdanningen. Lenge forble dette noe som bare spesielt interesserte hadde kunnskap om (Senter for IKT i utdanningen, 2016).

De siste tiårene har det på tross av dette vokst frem en internasjonal bevegelse for å fremme programmering i skolen (Senter for IKT i utdanningen, 2016). Norge har, som flere andre land, sett at det er et behov for å integrere programmering i skolepensumet. Et av hovedargumentene er at dette skal bidra til å styrke sentrale ferdigheter, som logisk tenkning og problemløsning, i et stadig mer digitalisert samfunn. Å lære elever å programmere vil også åpne for muligheter til å kunne forme og utvikle egen teknologi, noe som vil være av stor betydning i en digital fremtid (Forsström & Kaufmann, 2018, s. 18).

1.2.2. NOU 2015:8 – FREMTIDENS SKOLE

NOU-utredningen *Fremtidens skole* (NOU 2015: 8, 2015) ble i juni 2015 utredet av Ludvigsenutvalget, et offentlig utvalg satt av Kunnskapsdepartementet. Utredningen skulle

legge et kunnskapsgrunnlag ved å foreslå valg om kompetanser for fremtiden og fornyelser av fagene. Her ble det vurdert hva elever i grunnskolen vil ha behov for å lære i et 20-30 års perspektiv. Med et fremtidsrettet perspektiv ble det foreslått fornyelser av fagene i skolen for å møte fremtidige kompetansebehov i arbeids- og samfunnslivet (NOU 2015: 8, 2015, s. 8, 47). Som beskrevet innledningsvis ble NOU-utredningen om fremtidens skole (NOU 2015: 8, 2015) tatt hensyn til i utformingen av de nye læreplanene LK20 og LK20S i forbindelse med Fagfornyelsen (Utdanningsdirektoratet, 2018a). Utredningen har dermed også virket inn på denne studiens tema og relevans i fremtidens skole.

1.2.2.1. MATEMATIKK OG TEKNOLOGI

Under fagspesifikk kompetanse i NOU-utredningen er *matematikk, naturfag og teknologi* trukket frem som ett av de sentrale fagområdene i norsk skole. Matematikk blir særskilt omtalt i NOU-utredningen fordi matematikk også inngår i de andre fagområdene – *språk, samfunnsfag og etikkfag og praktiske og estetiske fag*. Matematisk kompetanse er gitt en spesiell plass i fremtidens skole, og det vises til bred enighet om behov for matematisk kompetanse fremover (NOU 2015: 8, 2015, s. 24). I et arbeidsperspektiv vil matematisk kompetanse bidra til å kvalifisere elevene til fremtidig deltakelse i arbeidslivet, og styrke Norges konkurransekraft. I et samfunnsperspektiv vil kompetanse i matematikk styrke den enkelte borgers mulighet til å delta aktivt i et demokrati (NOU 2015: 8, 2015, s. 57). I NOU-utredningen er matematisk kompetanse beskrevet ved hjelp av de fem komponentene *forståelse, beregning, anvendelse (strategisk tankegang), resonnering og engasjement*. Komponentene støttes av hverandre, og er tett sammenflettet. Utviklingen av disse må skje parallelt, da en sterk forbindelse mellom dem vil øke elevenes mulighet for å utvikle en varig, fleksibel, nyttig og relevant matematisk kompetanse (NOU 2015: 8, 2015, s. 57).

1.2.3. ENDRINGER FOR FREMTIDEN - NYE LÆREPLANER

En introduksjonsvideo av Utdanningsdirektoratet for Fagfornyelsen avslutter med sitatet:

Det elevene lærer skal være relevant for fremtiden.

(Utdanningsdirektoratet, 2017)

I 2020 vil nye læreplaner være klare for å bli tatt i bruk i norske skoler. Mer relevant innhold, tydeligere prioriteringer og bedre sammenheng mellom fagene skal gjøre de nye læreplanene mer relevant for fremtiden (Utdanningsdirektoratet, 2018b). Ny kunnskap, ny teknologi og nye utfordringer er bakgrunnen for at læreplanene for opplæringen i skolen i ferd med å fornyes. Hensikten er at elevene skal rustes til å kunne løse fremtidens utfordringer (Utdanningsdirektoratet, 2017).

Realfag er ett av satsningsområdene som er fremtredende i fornyelsen av skolefagene. Digitaliseringen av samfunnet medfører betydelige endringer i matematikkfaget og andre realfags innhold (NOU 2015: 8, 2015, s. 44). NOU-utredningen NOU 2015: 8 (2015, s. 53) trekker frem at fremtidig satsning på realfag bør ha et overordnet mål om å forbedre alle barn og unges kompetanse i matematikk, naturfag og teknologi. Dette frembringer et behov om endring av kompetansemål, for å koble matematikk tettere til blant annet teknologiske fag som krever mye matematikk (NOU 2015: 8, 2015, s. 54). Fagfornyelsen av matematikkfaget retter fokus mot at elever skal arbeide mer med metoder og tenkemåter, slik at de får en større forståelse for faget. En av endringene i matematikkfaget vil bli at programmering og algoritmisk tankegang skal inn i faget, og vil kunne dekkes på ulike måter under de fastsatte kjerneelementene (Kunnskapsdepartementet, 2018).

Ett av kompetansemålene fra siste høring av de nye læreplanene (våren 2019), utpeker seg som særlig relevant til denne oppgavens fokus. Etter 6. trinn er ett av målene for opplæringen i matematikk at eleven skal kunne «*bruke variabler, lykkjer, vilkår og funksjonar i programmering til å utforske geometriske figurar og mønster*» (Utdanningsdirektoratet, 2019). Dette har ikke vært et kompetansemål det bevisst har vært arbeidet mot i studien, fordi de nye læreplanene er blitt utformet parallelt med dette masterprosjektet. Det forteller likevel noe om den direkte relevansen denne forskningen vil kunne ha for det «nye» matematikkfaget i grunnskolen.

1.3. AVGRENSNING AV TEMA OG FORSKNINGSSPØRSMÅL

Min interesse for programmering ble vekket da jeg ble gjort kjent med at det ville få en plass i pensumet for matematikkfaget i grunnskolen. Programmering var et emne jeg kunne lite om fra før. Jeg ble derfor nysgjerrig på sammenhengen mellom matematikk og programmering, og

hvilke muligheter dette kunne tilføre matematikkundervisningen. Med bakgrunn i dette er følgende problemstilling blitt utformet:

Hvordan kan programmering fremme matematisk kompetanse hos elever?

Til å undersøke denne problemstillingen er det tatt utgangspunkt i elevenes samtaler hvor de i par programmerer geometriske figurer i programmet Scratch. I denne studien er programmering i hovedsak betraktet som et verktøy til å arbeide med matematikk. Informantene i studien er elever fra 7. trinn, ettersom det var dette trinnet det åpnet seg mulighet for å samarbeide med. For å avgrense og mer konkret gå inn i oppgavens problemstilling med bakgrunn i studiens mål og forutsetninger, er den blitt undersøkt i lys av de to følgende forskningsspørsmålene:

- 1) *Hvordan kommer matematisk kompetanse til uttrykk i elevers samtaler når de programmerer?*
- 2) *Hvordan konstruerer elever matematisk kunnskap når de programmerer?*

Til å undersøke det første forskningsspørsmålet er det nødvendig å beskrive hva matematisk kompetanse er, for å kunne identifisere den i elevenes samtaler. Til å gjøre dette er det utformet et eget teoretisk rammeverk for matematisk kompetanse, som støtter seg til en sammensetning av to rammeverk. Det første rammeverket er utviklet av Kilpatrick, Swafford & Findell (2001), og tar for seg fem komponenter av matematisk kyndighet. Dette er beriket med et rammeverk fra Niss & Højgaard Jensen (2002), som omhandler åtte matematiske delkompetanser av en helhetlig matematisk kompetanse. Senere i kapittel 2.3 er det utdypet hvordan dette nye teoretiske rammeverket er utformet til å styrke analysen av elevenes samtaler.

På bakgrunn av datamaterialets egenart og metodevalget i oppgaven, er det tatt utgangspunkt i ideer om læring som Seymour Papert introduserte (1991, 1993). Ideene dannet senere grunnlaget for et nytt læringssyn, kalt for *konstruksjonismen*. Beskrivelser av dette synet på

læring er blitt brukt som bakgrunn til å gjøre rede for det andre forskningsspørsmålet og er beskrevet i kapittel 2.4. Målet er å gi innsikt i hvordan elever konstruerer matematisk kunnskap når de programmerer, hvilket vil ha betydning for hvordan matematisk kompetanse kan fremmes ved å kombinere matematikk og programmering i undervisningen.

1.3.1. DEL AV ET STØRRE PROSJEKT

Studien er en del av LATAcME (<https://prosjekt.hvl.no/latacme/>), et større forskingsprosjekt ved Høgskolen på Vestlandet (HVL). LATAcME står for *Learning About Teaching Argumentation for Critical Mathematics Education in multilingual classrooms*. Dette prosjektet er rettet mot to obligatoriske matematikkfag i grunnskolelærerutdanningen for 1.-7. trinn. Det overordnede faglige fokuset er på argumentasjon og kritisk matematikdidaktikk i et flerspråklig klasserom. Forskningsprosjektet er delt i tre underprosjekt, hvor min studie er en del av underprosjektet *argumentasjon og IKT* (<https://prosjekt.hvl.no/latacme/delprosjekt/ikt/>).

1.4. BEGREPSAVKLARING

For å utdype forskningsspørsmålene vil jeg kort gjøre rede for to sentrale begrep før jeg fortsetter. Disse er *matematisk kompetanse* og *programmering*. I tillegg vil jeg gi en kort innføring i *Scratch* – programmeringsverktøyet og -programmet som er benyttet i studien.

1.4.1. MATEMATISK KOMPETANSE

I oppgavens problemstilling og det første forskningsspørsmål som tidligere beskrevet blir det referert til *matematisk kompetanse*. I rammeverket av Kilpatrick et al. (2001) skildres det de har kalt for *matematisk kyndighet*, mens Niss & Højgaard Jensen (2002) diskuterer *matematisk kompetanse*. Jeg har valgt å bruke begrepet *matematisk kompetanse* som en paraplybetegnelse for disse to begrepene som vil utdypes senere i oppgaven.

Kompetansebegrepet er i *Overordnet del av læreplanverket* definert som følger:

Kompetanse er å kunne tilegne seg og anvende kunnskaper og ferdigheter til å mestre utfordringer og løse oppgaver i kjente og ukjente sammenhenger og situasjoner. Kompetanse innebærer forståelse og evne til refleksjon og kritisk tenkning.

(Utdanningsdirektoratet, 2018c)

Denne definisjonen av kompetansebegrepet vil være gjeldene i forbindelse med fornyelsen av fagene i LK20 og LK20S. Nortvedt & Pettersen (2016) har tidligere knyttet sammen matematisk forståelse, kunnskap og ferdigheter ved å bruke betegnelsen *matematisk kompetanse*. Jeg har derfor vurdert det slik at jeg på samme måte kan knytte sammen Kilpatrick et al. (2001) og Niss & Højgaard Jensen (2002) ved å bruke begrepet *matematisk kompetanse* som en dekkende betegnelse for det disse to diskuterer. I det andre forskningsspørsmålet brukes begrepet *matematisk kunnskap*. Dette vil i denne sammenheng bli sett på som en del av en helhetlig matematisk kompetanse.

1.4.2. PROGRAMMERING

Programmeringsspråk er et konstruert språk som kun brukes til å skrive programmer til datamaskiner for å kontrollere og styre dem (Rossen, 2017). Det finnes flere programmeringsspråk, som for eksempel Java Script, LOGO og Scratch. Disse brukes for å bestemme hva datamaskinen skal gjøre og få dem til å «kommunisere» med hverandre. Programmeringsprosessen kjennetegnes ved utforming og implementering av instruksjoner for dataprogram, også kjent som koding. Denne prosessen gjør det mulig for datamaskiner å utføre spesifikke oppgaver, løse problemer og støtte menneskelig interaksjon (Forsström & Kaufmann, 2018, s. 19). Programmering er likevel mer enn et konstruert språk. Man må også finne ut hva man skal gjøre og hvordan man kan gjøre det på en presis måte, noe som krever algoritmisk tenkning (Lynnebakken, 2018). Forenklet kan man si at ved programmering settes koder sammen til en oppskrift man gir til datamaskinen. Denne oppskriften utfører så datamaskinen 100% etter punkt og prikke, i nøyaktig den rekkefølgen kodene er skrevet. Oppskriften må derfor være presis ut fra det man vil at datamaskinen skal gjøre.

1.4.3. SCRATCH – BLOKKBASERT PROGRAMMERING

For å undersøke matematisk kompetanse i elevers arbeid med programmering, er det blitt valgt et programmeringsverktøy som er tilpasset barn. Det har vært viktig å velge et program som ikke krever forkunnskaper til programmering, og kan kombineres i arbeid med matematikk på barnetrinnet. Et program som innfrir disse kravene er Scratch. Dette er derfor blitt valgt som programmeringsverktøy for denne studien.

Scratch er et gratis blokkbasert program på nett (<http://scratch.mit.edu>) utviklet av Lifelong Kindergarten Group ved MIT Media Lab. Programmet er utviklet med mål om å skape en tilnærming til programmering som kan appellere til alle. Det krever heller ingen tidligere erfaringer med programmering. Scratch er utformet slik at det er enkelt å bruke, samtidig som det med sitt fleksible oppsett åpner for mange ulike bruksmuligheter (Resnick et al., 2009). Dette er kvaliteter ved programmet som gjør at det er egnet for bruk på barnetrinnet.

Programmeringsspråket i Scratch består av flere klosser med ulike kommandoer. Klossene er funksjonene i skriptet. Disse er organisert i en skriptmeny i kategorier som *bevegelse*, *utseende*, *styring* osv. for lettere å kunne finne fram til ønsket kloss/funksjon (se Figur 1).



FIGUR 1: SKJERMBILDE AV SCRATCH MED EGNE BESKRIVELSER AV PROGRAMMET

Programmeringen foregår ved at klossene fra skriptmenyen dras bort i et handlingsvindu. Her settes klossene sammen til blokker, som representerer skriptet i programmet. Blokkene settes

til de ulike visuelle delene i programmeringsvinduet, som for eksempel en figur eller bakgrunn, for å bestemme dens funksjon(er) og/eller handling(er). Rekkefølgen på klossene i blokken utføres kronologisk og fører til handling i programmeringsvinduet (se Figur 1). I dette prosjektet har programmeringen gått ut på at elevpar har fått i oppgave å samarbeide om å programmere en pennfigur til å tegne geometriske figurer (se oppgaven i Vedlegg II).

1.5. TIDLIGERE FORSKNING

Kombinasjon av programmering og matematikk har en forholdsvis kort historie i grunnutdanningen. Dette er derfor et stort område som fortsatt vil ha behov for mer forskning, noe flere av studiene jeg her trekker frem påpeker. Selv om programmering i matematikkfaget er relativt nytt, er det likevel gjort relevant og nyttig forskning som ser på denne sammenhengen. Jeg vil her først ta for meg tidligere forskning gjort på programmering generelt i sammenheng med matematikk, før jeg spisser meg inn på forskning gjort på Scratch.

1.5.1. TIDLIGERE FORSKNING PÅ PROGRAMMERING OG MATEMATIKK

I en litteraturgjennomgang av Forsström og Kaufmann (2018) er til sammen 15 artikler som henviser til empiriske studier blitt analysert. Her vurderes utdanningens potensiale ved å integrere programmering i matematikkfaget. Studiene Forsström og Kaufmann (2018) omtaler diskuterer ideen om programmering satt i sammenheng med matematikk i grunnskoleutdanningen. I deres litteraturgjennomgangen er det identifisert et behov for mer forskning på hvordan programmering kan integreres i skolen, hvilken plass den skal ha og hvorfor den skal inn i skolen og på barnetrinnet. Det fremmes også et behov for å gi lærere nødvendig kunnskap om hvordan programmering kan innlemmes i skolefagene på en fornuftig og hensiktsmessig måte.

Karakteristiske tema i gjennomgangen av de 15 studiene er økning av elevers motivasjon for å lære matematikk, forbedring av matematiske prestasjoner og økt samarbeid ved ulik utøvelse av lærerrollen (Forsström & Kaufmann, 2018, s. 29). Det kommer frem at programmering tilbyr

en virkelighetsnær kobling i matematikklæringen, som er en viktig faktor for å motivere elever (Ke, 2014; Leonard et al., 2016 referert i Forsström & Kaufmann, 2018). Dette fremmes som et viktig aspekt i matematikkfaget, da det i matematikk vanligvis kan se ut til å være vanskelig for elever å finne formålet med det de lærer (Lambic, 2011 referert i Forsström & Kaufmann, 2018).

Store deler av forskningen som ser på sammenhengen mellom programmering og matematikk fokuserer på geometri. Ifølge analysen av studiene er geometri et emne i matematikk som har en naturlig kobling til programmeringsaktiviteter. I forskning gjort av Grover og Pea (2013) relateres programmering til utvikling av algoritmisk tenkning (Grover & Pea, 2013 referert i Forsström & Kaufmann, 2018). Begrunnelsen er at programmering fremmer problemløsningsferdigheter, logisk tenkning og motiverer elever til å lære matematikk. Om programmering fremmer dette vil det også trolig kunne benyttes som en ressurs i andre emner enn geometri. Selv om programmering kan knyttes til emner som geometri i matematikk, vil koblingen til matematikkfaget avhenge av de samlede avgjørelsene elevene gjør gjennom problemløsningsaktiviteten (Grover & Pea, 2013 referert i Forsström & Kaufmann, 2018).

Den politiske avgjørelsen om å integrere programmering i matematikkfaget kan ifølge Forsström og Kaufmann (2018) forsvares, på bakgrunn av potensialet som gjør programmering velegnet for arbeid med matematikk. En slik avgjørelse om endring vil kreve en fornyelse i undervisningspraksisen i matematikkfaget. Det vil ikke være formålstjenlig å ha «tradisjonell» undervisning, hvor elevene arbeider alene og læreren er den styrende i læringsaktiviteten. Dette er på grunn av de karakteristiske trekkene for elevs læringsprosess i arbeid med programmering. Disse læringssituasjonene er ofte forbundet med samarbeid, og lærerrollen utarter seg annerledes enn i mange andre mer tradisjonelle undervisningsøker i matematikkfaget (Forsström & Kaufmann, 2018, s. 30). Integrering av programmering i matematikkundervisning medfører dermed også et behov for å endre vurderingspraksis og å skape et godt støttesystem rund dette i skolen (Husain, Kamal, Ibrahim, Huddin & Alim, 2017; Taylor, Harlow & Forret, 2010).

1.5.2. TIDLIGERE FORSKNING PÅ SCRATCH

Flere positive funn er gjort i studier hvor Scratch er brukt i arbeid med programmering i skolen og i matematikkfaget. Studier gjort på Scratch i undervisningssammenheng fremhever blant annet positiv effekt på de affektive sidene av elevenes motivasjon og interesse for programmering (Malan & Leitner, 2007; Meerbaum-Salant, Armoni & Ben-Ari, 2013; Sáez-López, Román-González & Vázquez-Cano, 2016; Wilson & Moffat, 2010 referert i Lang-Ree, 2016, s. 60). Med affektive sider nevnes motivasjon, engasjement, entusiasme, innsats og interesse som karakteristisk i arbeidsøktene med programmering i Scratch. Andre positive funn er aktiv læring i form av deltakelse og kommunikasjon, bruk av fagbegreper og grunnleggende konsepter, og opplevelsen av at læringsarbeidet har vært nyttig og lærerikt (Sáez-López, Román-González & Vázquez-Cano, 2016 referert i Lang-Ree, 2016, s. 20).

Studien gjort av Moreno-León, Robles, og Román-González (2016) har tatt for seg elever fra 2. og 6. trinn. Resultatene viser en akselererende læringskurve i matematikk hos elevene på 6. trinn, med særlig god effekt fra de sosiale studiene. Den store læringseffekten på det sosiale arbeidet mellom elevene uttrykkes å kunne komme av en bedre justering av de kognitive og motiverende faktorene. Begrunnelsen er at elevene i de sosiale studiene viste et høyere nivå av glede, engasjement og selvtillit i eget læringsarbeid. Elevene på 2. trinn viste verken stigende eller nedadgående læringskurve, noe som konkluderes med å komme av forskjellen på nivå av kognitiv modning i forhold til elevene på 6. trinn.

Taylor, Harlow, og Forret (2010) har i likhet med Moreno-León et al. (2016) sett fordeler ved samarbeid i aktiviteter med Scratch. Undersøkelsen er gjort med elever i 9-10 års alderen der Scratch er blitt introdusert og brukt i samarbeidsaktiviteter på PC og interaktiv tavle. Studien beskriver hvordan elevens engasjement med Scratch skaper gode muligheter for samarbeid. Funnene viser også en nødvendighet for å bruke problemløsningsprosessen, som å sette seg mål og generere og teste ideer. Gjennomføringen av Scratch-prosjektet viser en sømløs forbindelse mellom matematikklæringen og de sosiale kravene. I denne artikkelen av studien er forskerne fremdeles i starten av å bearbeide data, og det gjenstår for denne å undersøke innflytelsen Scratch har hatt på den matematiske tenkingen. Likevel viser funn at arbeid med Scratch har hjulpet barn til å forbedre problemløsningsstrategier i matematikk, kreativ tenkning, systematisk resonnering og samarbeid.

Husain, Kamal, Ibrahim, Huddin, og Alim (2017) sin studie understreker at implementering av programmering i utdanning ikke bare vil kunne tilby programmeringsferdigheter, men også ferdigheter innen språk, matematikk og vitenskap. Både matematisk tenkning, problemløsning, sekvensering og logisk tenkning blir involvert i arbeidet med å utforme et programmeringsprosjekt. I studien er det utforsket metoder for å lære bort grunnleggende matematiske konsepter og effekten det har på elevers problemløsningsferdigheter ved å lære å programmere i Scratch. Resultatene viser blant annet at elevenes arbeid i Scratch førte til at de bedre kunne forstå grunnleggende matematiske konsepter og ulike aspekter av matematikk.

1.6. OPPGAVENS STRUKTUR

I dette første kapitlet er det gjort rede for valg av tema, noen sentrale begrep, beskrivelse av programmeringsverktøyet som er benyttet og tidligere forskning. I det andre kapitlet presenteres det teoretiske bakteppet for studien. Her er det gått nærmere inn på hva matematikk er og hvordan matematikk og programmering har en forbindelse. Oppgavens teoretiske rammeverk, altså Kilpatrick, Swafford & Findell (2001) sine fem tråder av matematisk kyndighet og Niss & Højgaard Jensen (2002) sin beskrivelse av åtte matematiske kompetanser, vil bli presentert her. Disse er begge presentert hver for seg, før jeg har satt dem sammen slik at de utgjør et nytt teoretisk rammeverk til å undersøke det første forskningsspørsmålet. Konstruksjonismen ved Seymour Papert (1991, 1993) er også en del av oppgavens teoretiske bakteppe, og er presentert her for å senere kunne undersøke det andre forskningsspørsmålet.

I tredje kapittel er det tatt for seg oppgavens forskningsmetodiske tilnærming. Her presenteres metodiske valg og redegjørelser tilknyttet forskningsprosessen, innsamling av data, utvalg fra datamaterialet og analyse av disse. I dette kapitlet er også analyseverktøyet operasjonalisert. Analyseverktøyet for første forskningsspørsmål er operasjonalisert ved å utforme koder for ulike komponenter av matematisk kompetanse. Det er også gitt eksempler på bruk av det teoretiske rammeverket for identifisering og analyse av matematisk kompetanse. Til andre forskningsspørsmål er analyseverktøyet operasjonalisert ved å beskrive hva jeg har lagt vekt på fra Seymour Papert (1991) sin beskrivelse av læring og konstruering av kunnskap.

I det fjerde kapitlet fremstilles resultat og analyse av transkripsjoner og samtaleutdrag fra datainnsamlingen av en arbeidsøkt hvor elever i par arbeid med programmering i Scratch. Dette

kapitlet er delt i tre. Første del presenterer resultater fra koding av transkripsjoner som er gjort. Disse henger sammen med andre del som er en analyse av enkeltutdrag fra transkripsjonene i lys av oppgavens teoretiske rammeverk for det første forskningsspørsmålet. Tredje del av analysen fremstiller en analyse av enkeltutdrag fra transkripsjonene i lys av det teoretiske grunnlaget for det andre forskningsspørsmålet.

I kapittel fem diskuteres ulike funn og emner knyttet til den forskningen som er blitt gjort. I diskusjonen er analyse og teori knyttet sammen for å drøfte oppgavens problemstilling med forskningsspørsmål.

Avslutningsvis i sjette kapittel oppsummeres studiens funn opp mot problemstillingen og forskningsspørsmålene. Her kommer jeg med noen avsluttende refleksjoner knyttet til dette forskningsprosjektet, før jeg nevner pågående forskning og tanker om veien videre.

2. TEORI

I dette kapitlet er det gjort rede for relevant teori som skal styrke studien. Oppgavens problemstilling går ut på å undersøke hvordan programmering kan fremme matematisk kompetanse hos elever. Det er derfor relevant å utdype forholdet mellom matematikk og programmering. I den sammenheng er det også gjort rede for hva matematikk og algoritmisk tenkning er. Sist i kapitlet er oppgavens teoretiske bakteppe presentert. Dette er todelt, da forskningsspørsmålene er belyst gjennom to ulike teoretiske angrepsvinkler. Første del består av et nytt sammensatt teoretisk rammeverk som beskriver matematisk kompetanse. Dette bygger på rammeverket av Kilpatrick, Swafford & Findell (2001) som beskriver matematiske komponenter, og rammeverket av Niss & Højgaard Jensen (2002) som diskuterer ulike matematiske kompetanser. Det nye rammeverket er senere anvendt for å undersøke hvordan matematisk kompetanse kommer til uttrykk i elevenes samtaler når de programmerer. Den andre delen presenterer et læringssyn av Seymour Papert (1991, 1993) og konstruksjonismen som er spesielt rettet mot matematikk. Dette utgjør det teoretiske grunnlaget for å senere kunne si noe om hvordan elever konstruerer matematisk kunnskap når de programmerer.

2.2. PROGRAMMERING OG MATEMATIKK

Innledningsvis ble det gjort rede for hva programmeringsbegrepet innebærer. Her vil jeg gi innsikt i hvorfor programmering og matematikk har en forbindelse, utdype hva matematikk er og hva det vil si å gjøre matematikk. Videre vil jeg ta for meg det som ofte nevnes i sammenheng med programmering og matematikk, nemlig algoritmisk tenkning. Det engelske ordet for dette er *computational thinking*.

2.2.1. MATEMATIKK

I følge Schoenfeld (1992, s. 3) er matematikk en iboende sosial aktivitet, hvor et fellesskap av trente utøvere (matematikkforskere) engasjerer seg i vitenskap om mønstre. Disse mønstrene kan være basert på å systematisk beskrive naturen eller prinsipp for regulariteter i systemer. De kan være definert aksiomatisk eller teoretisk og kalles da for «ren matematikk», eller modeller

av systemer som er abstrahert fra virkelige verdensobjekter og kalles «anvendt matematikk» (Schoenfeld, 1992, s. 3).

[...] mathematical notation no more is mathematics than musical notation is music. A page of sheet music represents a piece of music, but the notation and the music are not the same; the music itself happens when the notes on the page are sung or performed on a musical instrument. It is in its performance that the music comes alive; it exists not on the page but in our minds. The same is true for mathematics.

(Devlin, 2000, s. 9)

På samme måte som Devlin (2000) sammenligner matematikk og musikk, beskriver Boaler (2009, s. 30) at matematikk er som en forestilling, en levende handling og en måte å entre verden på. Når de matematiske notasjonene leses av en kompetent utøver (som en matematiker) blir matematikken levendegjort, som en symfoni i leserens sinn (Devlin, 2000, s. 9). Med et slikt syn på matematikk vil det derfor ikke være tilstrekkelig å lære elever å memorere og pugge ulike metoder og regler for å lære dem matematikk. Å lære elever å anvende matematikk slik som matematikere gjør handler om å fremstille problem (problem posing), komme med antakelser og gjetninger, utforske og avgrense ideer og kommunisere og diskutere ideer med andre. Gjennom en slik anvendelse får elevene kjenne på hva sant matematisk arbeid er. De får mulighet til å oppleve gleden som ligger i dette arbeidet, samtidig som matematikk læres og utvikles på den mest produktive måten (Boaler, 2009, s. 31).

2.2.2. HVORFOR PROGRAMMERING I MATEMATIKK?

At matematikk og programmering ses som en naturlig sammenkobling, kommer blant annet av at programmeringsprosessen kan settes i sammenheng med matematisk tenkning. Grover & Pea (2013) referert i Forsström & Kaufmann (2018, s. 19) hevder at dette særlig er på grunn av relasjonen programmeringsprosessen har til utvikling av algoritmisk tenkning. Denne måten å tenke på blir sett på som sentral for det digitale samfunnet og utvikling av ferdigheter som problemløsning, kreativitet og logisk tenking i det 21. århundre (Forsström & Kaufmann, 2018, s. 19). Finland og Sverige har allerede integrert programmering i matematikkfaget, med bakgrunn i å fremme problemløsning og logisk tenkning, og å øke elevens motivasjon for å lære

matematikk (Bocconi, Chiocciariello, & Earp, 2018; Opetushallitus, 2014; Skolverket, 2018 referert i Forsström & Kaufmann, 2018, s. 19). Som sine to naboland, vil også Norge nå integrere programmering i matematikkfaget gjennom Fagfornyelsen i 2020. Selv om det kan finnes gode argumenter for å legge programmering til et eget fag, er det utvikling av algoritmisk tenkning som er brukt som hovedargument til å legge det matematikkfaget (Kjerneelementgruppen for matemtaikk, 2017). I neste avsnitt er det derfor gitt en beskrivelse av hva algoritmisk tenkning går ut på.

2.2.3. ALGORITMISK TENKNING

En algoritme kan forklares som en type oppskrift, og kan sammenlignes med en strikkeoppskrift, matoppskrift eller regneoppskrift i matematikk. Gjennom å sette sammen komponenter i en bestemt rekkefølge fører dette til at du får et produkt som følge av denne oppskriften. Brukes det andre komponenter eller rekkefølgen byttes om, vil du kunne få et annet produkt. Algoritmisk tenkning, også omtalt som algoritmisk problemløsning, blir av Utdanningsdirektoratet (2016, s. 18-19) beskrevet som det å løse et problem gjennom å nøye angi en presis rekkefølge av komponenter. Wing (2006) beskriver algoritmisk tenkning som en grunnleggende ferdighet i likhet med lesing, skriving og aritmetikk. Som Wing (2006) skriver består algoritmisk tenkning av både styrkene og begrensningene til databehandlingsprosesser, uansett om det er mennesker eller maskiner som utfører dem. Algoritmisk tenkning går ut på å bryte et problem ned i mindre deler slik at det blir håndterbart. Det er en måte som mennesker tenker på for å løse problemer, og ikke noe datamaskinen gjør (Wing, 2006, s. 35). Programmering kan knyttes tett til algoritmisk tenkning fordi det handler om å utforme et programmeringsspråk gjennom algoritmer som datamaskinen kan tolke.

Algoritmisk tenkning kan forbindes med blant annet matematisk tenkning fordi datavitenskap (computer science) tar utgangspunkt i og er formelt grunnet på matematikk (Wing, 2006). I matematikk benyttes også algoritmisk tankegang på den måten at tall og andre komponenter må settes sammen på bestemte måter for at det skal fremkomme et gyldig resultat. Likevel vil det kunne være forskjell i å tenke algoritmisk i matematikk og programmering, ved at det i programmering brukes et annet *språk* eller sett med symboler. Programmeringsspråk består ofte av bokstaver, ord eller andre koder, mens det i matematikk brukes tall og andre symboler. En annen forskjell er også den logiske oppbyggingen av *oppskriften*. Å regne ut et

matematikkstykke kan for eksempel gjøres i flere steg, før det settes sammen til en helhet og man kan komme frem til en løsning. I programmering vil denne fleksibiliteten ikke være den samme, ettersom kodenens rekkefølge alltid vil følges 100% i kronologisk rekkefølge. Det vil likevel tenkes at algoritmisk tenkning kan fremmes gjennom både matematisk arbeid og programmering, og vil kunne styrke kunnskapen på begge områder.

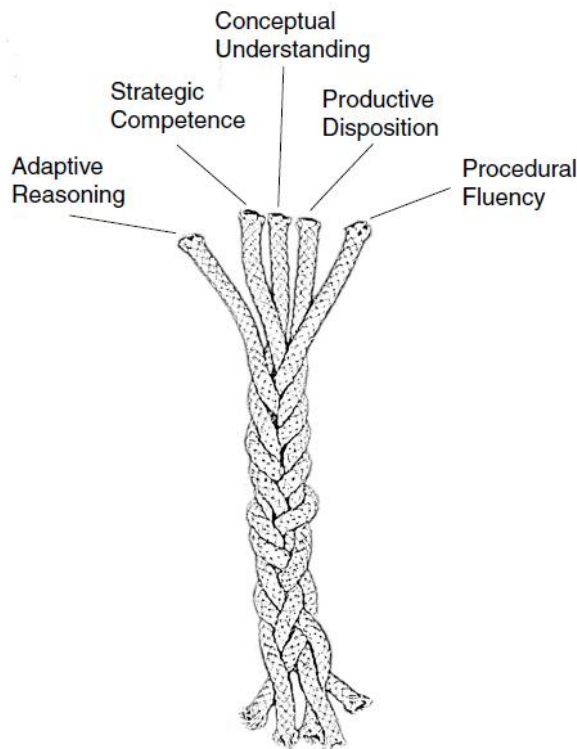
2.3. ET NYTT SAMMENSATT RAMMEVERK

Til det første forskningsspørsmålet i denne oppgaven har jeg utformet et eget teoretisk rammeverk. Dette er sammensatt av de to rammeverkene som vil bli presentert her. Etter disse er presentert, er sammenhengen mellom de to rammeverkene diskutert. Sammen utgjør disse et nytt teoretisk rammeverk til å undersøke elevers matematiske kompetanse.

2.3.1. KILPATRICK, SWAFFORD & FINDELL (2001) – FEM TRÅDER AV MATEMATISK KYNDIGHET

Kilpatrick, Swafford og Findell (2001) beskriver de kognitive forandringene som de mener er ønskelig å fremme hos barn slik at de kan oppnå en vellykket matematikkopplæring. Begrepet de har valgt å bruke til å inneslutte alle aspekter av ekspertise, kompetanse, kunnskap og fasiliteter i matematikk, er *matematisk kyndighet* (*Mathematical proficiency*). Matematisk kyndighet blir beskrevet gjennom fem komponenter, og illustreres som *tråder*. Disse trådene er ikke uavhengige, men *sammenflettet* og representerer ulike aspekter av en kompleks helhet (Kilpatrick et al., 2001, s. 116).

Intertwined Strands of Proficiency



FIGUR 2: MODELL HENTET FRA S. 117 I KILPATRICK, SWAFFORD & FINDELL (2001)

*«The five strands are interwoven and interdependent in the development of proficiency in mathematics»
(Kilpatrick et al., 2001, s. 116)*

For å kunne utvikle og oppnå matematisk kyndighet må det fokuseres på alle de fem trådene, da de er gjensidig avhengige av hverandre. Disse komponentene fremmer et rammeverk for å diskutere kunnskap, ferdigheter, evner og oppfatninger som til sammen utgjør matematisk kyndighet. Denne kyndigheten skal gjøre elevene i stand til å beherske matematiske utfordringer i dagliglivet, og i senere skolegang og arbeidsliv kunne fortsette å studere matematikk (Kilpatrick et al., 2001, s. 116).

De fem trådene det er snakk om er *conceptual understanding*, *procedural fluency*, *strategic competence*, *adaptive reasoning* og *productive disposition* (Kilpatrick et al., 2001, s. 117). Til å finne norske begrep for disse komponentene er det tatt utgangspunkt i NOU-utredningen NOU 2015: 8 (2015, s. 57) og masteroppgaven til Johnsen & Olsen (2015) sine oversettelser.

Begrepene som videre vil bli brukt er *konseptuell forståelse* (conceptual understanding), *beregning* (procedural fluency), *strategisk kompetanse* (strategic competence), *resonnering* (adaptive reasoning) og *engasjement* (productive disposition).

Konseptuell forståelse (conceptual understanding) blir referert til som en integrert og funksjonell forståelse av matematiske ideer. *Konseptuell forståelse* ligger nært til *begrepsforståelse*, et kanskje mer vanlig norsk ord. Innholdet i disse har mye til felles, men *konseptuell forståelse* består av mer enn å ha forståelse for begrep. Det handler også om forståelse for ideer og å kunne drive disse videre. For å bruke et begrep som dekker Kilpatrick et al. (2001) sitt innhold av denne komponenten er det valgt å bruke betegnelsen *konseptuell forståelse*. Har man en *konseptuell forståelse* blir det klart hvorfor en matematisk idé er viktig, og i hvilke kontekster denne kan være nyttig. Kunnskapen man allerede har er organisert til en sammenhengende helhet. Dette gjør det mulig å lære nye ideer ved å sette de allerede lærte ideene i sammenheng med de nye. En slik forståelse støtter også opp under å bevare kunnskap. Om man lærer gjennom å skape en forståelse for det som læres, blir det lettere å huske, bruke og senere kunne rekonstruere kunnskapen om den blir glemmt. Et annet viktig poeng som nevnes av Kilpatrick et al. (2001) er at man kan forstå noe før det kan uttrykkes verbalt. Dette gjør at det ikke alltid vil være lett å identifisere en *konseptuell forståelse*. Likevel sies det at en tydelig indikator for en *konseptuell forståelse* er å være i stand til å kunne representere matematiske situasjoner på ulike måter, og å vite hvordan forskjellige representasjoner kan være hensiktsmessig til ulike formål. Om en elev besitter en *konseptuell forståelse* innen et matematisk område, kan han/hun se sammenhenger rundt konsepter og prosedyrer. Dermed kan han/hun kunne komme med argumenter for å forklare hvorfor noen faktorer er konsekvenser av andre. Dette kan medføre at elevens selvtillit vil øke, noe som igjen vil skape grunnlag for å kunne bevege seg videre til et nytt nivå av *konseptuell forståelse*. På denne måten vil konsepter og prosedyrer kunne utvides til nye områder (Kilpatrick et al., 2001, s. 118-119).

Beregning (procedural fluency) refererer her til kunnskap om prosedyrer eller matematiske operasjoner, kunnskap om når og hvor de er hensiktsmessig, og ferdighet til å uttrykke dem fleksibelt, nøyaktig og effektivt. *Beregning* kan for eksempel være kunnskap om måter å estimere resultat av en prosedyre. Denne komponenten henger tett sammen med og er spesielt nødvendig for å oppnå *konseptuell forståelse*, som å forstå plassverdisystemet eller å se likheter

og ulikheter mellom ulike regnemetoder. Å kunne anvende ulike metoder for *beregning* innebærer å kunne bruke kognitive metoder, som hoderegning, for å finne ulike summer, ulikheter, produkt eller mengder. *Beregning* går også ut på å kunne bruke metoder hvor en benytter seg av for eksempel kalkulatorer, datamaskiner eller ulike manipulative materialer (Kilpatrick et al., 2001, s. 121).

Komponenten *strategisk kompetanse (strategic competence)* tar utgangspunkt i å representere, formulere og løse matematiske problem. Denne komponenten ligner det som tradisjonelt kalles for problemløsning og problemformulering i matematikken. Elevene må kunne formulere problemet, slik at matematikk kan brukes til å løse det. *Strategisk kompetanse* innebærer å kjenne til en variasjon av ulike løsningsmetoder, i tillegg til å ha kunnskap om hvilke strategier som vil kunne være nyttig for å løse et spesifikt problem (Kilpatrick et al., 2001, s. 124). Det er særlig et gjensidig støttende forhold mellom komponentene *strategisk kompetanse, konseptuell forståelse og beregning*. Etter hvert som elevene er i prosessen med å utvikle *beregning*, utspilles den *strategiske kompetansen*. Gradvis som forståelsen for operasjonen styrkes, utvikles den *konseptuelle forståelsen* parallelt med *beregning*. Dette kan føre til videre utvikling av den *strategiske kompetansen*, fordi man etter hvert lærer å erstatte anstrengende og kompliserte prosedyrer med noen mer presise og effektive (Kilpatrick et al., 2001, s. 127).

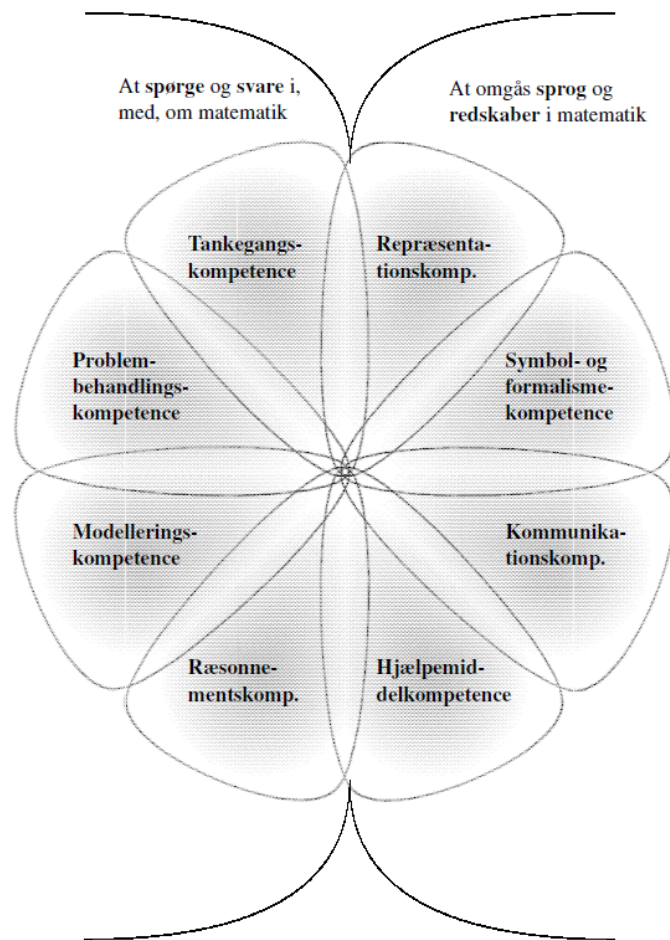
Kilpatrick, Swafford & Findell legger til grunn en vid forståelse av begrepet om *resonnering (adaptive reasoning)* i matematikk – «*Adaptive reasoning refers to the capacity to think logically about the relationships among concepts and situations*» (Kilpatrick et al., 2001, s. 129). *Resonnering* betegner altså den kapasiteten man har til å tenke logisk om forhold rundt konsepter og situasjoner. Begrepet om *resonnering* i matematisk sammenheng går med dette utover det som omhandler formell og gyldig bevisføring. Her inkluderes andre typer forklaringer og begrunnelser som kan være mer uformelle, og intuitive og induktive resonnement som er basert på mønster, analogi og metaforer. Analoge resonnement, metaforer og imaginære og fysiske representasjoner beskrives som «tankeverktøy» og knyttes opp mot denne type resonnement. *Resonnering* blir i denne sammenhengen referert til som limet som holder alt i matematikk sammen, og er «ledestjernen» som viser vei for læring (Kilpatrick et al., 2001, s. 129).

Komponenten *engasjement (productive disposition)* handler om å kunne se matematikk som fornuftig, nyttig og verdifullt, samtidig som man har en positiv innstilling til faget og erkjenner

verdien av å beherske matematikk. Utvikler en de resterende fire komponentene (*konseptuelle forståelse, beregning, strategisk kompetanse og resonnering*) vil en kunne få troen på at matematikk kan læres og brukes, og at en selv vil kunne være i stand til finne løsninger (Kilpatrick et al., 2001, s. 131). *Engasjement* vil med andre ord utvikles om de andre «trådene» eller komponentene styrkes og har et tett sammenflettet forhold til hverandre.

2.3.2. NISS & HØJGAARD JENSEN (2002) – ÅTTE MATEMATISKE KOMPETANSER

I samarbeid med sin forskningsgruppe beskriver Niss & Højgaard Jensen (2002, s. 43) matematisk kompetanse. Denne kompetansen beskrives som å ha kunnskap om og å forstå, utøve, anvende og å kunne ta stilling til matematikk og matematisk virksomhet i flere ulike sammenhenger, hvor det inngår eller vil komme til å inngå matematikk. For å skape en oversikt over det mangfoldet som inngår i begrepet matematisk kompetanse, er det blitt presentert to hovedkategorier – *å kunne spørre og svare i og med matematikk* og *å kunne håndtere matematikkens språk og redskaper* (Niss & Højgaard Jensen, 2002, s. 44). Disse to hovedkategoriene består hver av fire sentrale kompetanser, fordelt etter innholdet i de til sammen åtte matematiske kompetansene, illustrert i figuren under.



FIGUR 3: MODELL AV MATEMATISKE KOMPETANSER (NISS & HØJGAARD JENSEN, 2002, s. 45)

Den første kategorien, *å spørre og svare i, med, om matematikk*, tar for seg kompetansene *tankegangskompetanse*, *problembehandlingskompetanse*, *modelleringskompetanse* og *resonnementekompetanse*. Hovedsakelig går denne kategorien ut på å kunne stille og svare på spørsmål i og med matematikk, og å kunne forstå, bedømme og frembringe argument for matematiske spørsmål.

For å kunne spørre og svare i og med matematikk, bidrar *tankegangskompetanse* med å kunne utøve den matematiske tankegangen. Dette vil være å kunne utforme og stille spørsmål og svar, som krever å kunne kjenne til, forstå og håndtere de gitte matematiske begrepene rekkevidde. Etter hvert som begrep forankres på ulike områder og forståelsen utvides, øker dette også mulighetene for abstraksjon og generalisering (Niss & Højgaard Jensen, 2002, s. 47). For å kunne svare på spørsmål i og med matematikk vil det være behov for *problembehandlingskompetanse* og *modelleringskompetanse*.

Problembehandlingskompetansen benyttes for å komme frem til og formulere til et svar på matematiske problem. Et matematisk problem betegner et spørsmål der det er nødvendig med en matematisk undersøkelse for å komme til svaret. Hva som regnes som et matematisk problem er relativt i forhold til personen som står overfor det (Niss & Højgaard Jensen, 2002, s. 48). Fremgangsmåten for å løse et problem avhenger av å kunne stille opp, oppdage, formulere, avgrense og presisere forskjellige matematiske problem.

Modelleringskompetansen går ut på evnen til å sette forhold utenfor matematikken i sammenheng med matematiske behandlinger gjennom å analysere og bygge matematiske modeller som belyser disse sammenhengene (Niss & Højgaard Jensen, 2002, s. 48, 52). Ettersom elevene i dette prosjektet ikke arbeidet med forhold utenfor matematikken, er denne kompetansen ikke blitt vektlagt noe særlig videre.

Korrektheten av svaret står sentralt i *resonnementskompetansen*, hvor et matematisk resonnement skal kunne følges og bedømmes (Niss & Højgaard Jensen, 2002, s. 47, 54). Dette kan være både muntlig og skriftlig, og inkluderer også å vite hva som skiller et bevis fra andre resonnement. I tillegg går denne kompetansen ut på å utrede og gjennomføre formelle og uformelle resonnement, som å omforme heuristiske resonnement til matematisk gyldige bevis.

Den andre kategorien som går ut på å kunne håndtere språk og redskaper i matematikk tar for seg kompetansene *representasjonskompetanse*, *symbol- og formalismekompetanse*, *kommunikasjonskompetanse* og *hjelpemiddelkompetanse*. Disse kompetansene innebærer å være i stand til å omgås og ta i bruk ulike representasjoner som utgjør matematikkens symbolspråk og formalisme. De innebærer også å kunne bruke ulike representasjoner og notasjoner for å kommunisere i, med og om matematikk, samt å kunne bruke, forholde seg til og beherske diverse tekniske hjelpemidler for matematiske virksomheter (Niss & Højgaard Jensen, 2002, s. 45-46).

Representasjonskompetansen omfatter å kunne håndtere forskjellige representasjoner av matematiske omstendigheter (Niss & Højgaard Jensen, 2002, s. 56). Da er det nødvendig å kunne forstå de ulike representasjonene i form av å omforme, tolke og skille mellom de matematiske objektene, fenomenene, problemene eller situasjonene. Denne kompetansen vil kanskje særlig komme til uttrykk i sammenheng med problembehandlingskompetansen og modelleringskompetansen. Den kan for eksempel synliggjøres i form av geometriske figurer eller verbale representasjoner.

Symbol- og formalismekompetansen har også en sterk tilknytning til representasjonskompetansen. Det er fordi denne kompetansen går ut på å kunne håndteresymboler og formaliteter som benyttes i matematiske representasjoner. Dette innebærer å omforme symbol- og formelspråk, oversette mellom matematisk symbolspråk og naturlig språk, og å kunne behandle og gjøre seg bruk av symbolholdige utsagn og uttrykk. Videre innebærer kompetansen å ha innsikt i de formelle matematiske system sin karakter og «spilleregler» (Niss & Højgaard Jensen, 2002, s. 58).

For å kunne kommunisere i, med og om matematikk må man benytte seg av *kommunikasjonskompetansen*. Karakteristisk for denne kompetansen er evnen til å kunne sette seg inn i og tolke andres matematiske skriftlige, muntlige eller visuelle «tekster» og utsagn, samt å kunne variere måter å uttrykke seg på overfor mottakeren. Denne kompetansen knyttes særlig sterkt opp mot representasjonskompetansen og symbol- og formalismekompetansen (Niss & Højgaard Jensen, 2002, s. 60).

Den siste av de åtte sentrale kompetansene er *hjelpemiddelkompetansen*. Det ligger gjerne i ordet at denne kompetansen handler om å kunne ta i bruk og forholde seg til diverse tekniske hjelpemidler for matematiske virksomheter. Dette inneslutter også det å ha innblikk de ulike hjelpemidlenes muligheter og begrensninger i forskjellige situasjoner (Niss & Højgaard Jensen, 2002, s. 62). Et programmeringsverktøy som Scratch vil kunne være et slikt teknisk hjelpemiddel med muligheter og begrensninger.

Disse kompetansene er tett sammenføydd og generell i natur, noe som gjør at de inngår i ethvert matematisk område og emne, og er uavhengig av hvilket nivå det arbeides på. En matematisk kompetanse er i seg selv selvstendig og velavgrenset, men dette betyr ikke nødvendigvis at det er skarpe grenser som skiller kompetansene fra hverandre. Modellen i Figur 3 gir en visuell fremstilling av hvordan hver av kompetansene forbindes og overlapper hverandre. Her kan man se at kompetansene utgjør hver sin del av den helhetlige matematiske kompetansen. Niss & Højgaard Jensen (2002, s. 43) bruker om dette uttrykket *knutepunkt* som befinner seg i en *klynge* av ting. Dette menes med at kompetansene ikke må isoleres, men settes i sammenheng med hverandre.

2.3.3. SAMMENHENG MELLOM KILPATRICK, SWAFFORD & FINDELL (2001) OG NISS & HØJGAARD JENSEN (2002)

Kilpatrick, Swafford & Findell (2001) beskriver her «The Strands of Mathematical Proficiency» (s. 115), hvor fem tråder skal illustrere de fem komponentene av matematisk kyndighet. Disse utgjør til sammen det som de kaller for *matematisk kyndighet* (*mathematical proficiency*). Som de selv fremstiller det er fem matematiske komponenter illustrert som et sammenflettet tau. Hver av disse komponentene blir sett på som en del av en helhet hvor de har et tett gjensidig forhold til hverandre. Niss & Højgaard Jensen (2002) har delt det de beskriver som *matematisk kompetanse* inn i to kategorier, som til sammen består av åtte delkompetanser av en helhetlig matematisk kompetanse. De to kategoriene delkompetansene er delt inn i er som nevnt *å kunne spørre og svare i og med matematikk* og *å kunne håndtere matematikkens språk og redskaper* (Niss & Højgaard Jensen, 2002, s. 44). I likhet med Kilpatrick et al. (2001) beskriver også Niss & Højgaard Jensen (2002) den sammenhengen og det gjensidige forholdet mellom delene – de åtte kompetansene, og hvordan disse er knyttet sammen til en helhet.

De har begge hver sin «paraply» som dekker en helhetlig oppfatning av det å besitte og utøve matematisk kunnskap, forståelse og ferdigheter. Det er likevel noen forskjeller som skiller de to helhetene. Blant annet inkluderer Kilpatrick et al. (2001) betegnelsene forståelse, ferdighet, kompetanse og engasjement, som til sammen utgjør en persons matematiske *kyndighet*. Niss & Højgaard Jensen (2002) har i motsetning beskrevet sin helhet med å omtale ulike kompetanser som er en del av en helhetlig matematisk *kompetanse*. Hver av de to «paraplyene» dekker tilsynelatende stort sett det samme området, men med ulike beskrivelser og betegnelser. De store likhetene mellom de to rammeverkene kan beskrives som en overlapp mellom de to komplekse helhetene de her fremstiller. Med denne beskrivelsen av rammeverkenes sammenheng menes det at flere kjennetegn av de komponentene Kilpatrick et al. (2001) beskriver vil være de samme hos flere av de delkompetansene Niss & Højgaard Jensen (2002) presenterer.

Likevel er det noen forskjeller mellom området Kilpatrick et al. (2001) dekker sammenlignet med Niss & Højgaard Jensen (2002). En av forskjellene er at Kilpatrick et al. (2001) har med en komponent av matematisk kyndighet de har kalt for *engasjement*. Beskrivelsen gitt av denne komponenten skiller seg ut fra det Niss & Højgaard Jensen (2002) omtaler som del av matematisk kompetanse. Kilpatrick et al. (2001) beskriver likevel dette som en komponent som

utvikles parallelt med at andre områder styrkes og får en sterkere forbindelse. Det kan derfor tenkes at en elev utvikler *engasjement* ved å øke sin matematiske kompetanse slik Niss & Højgaard Jensen (2002) beskriver den, men at det ifølge Niss & Højgaard Jensen (2002) ikke påpekes som en forutsetning for å utvikle matematisk kompetanse. Enda en forskjell er at Niss & Højgaard Jensen (2002) inkluderer *modelleringskompetanse* som en del av en helhetlig matematisk kompetanse. Dette er ikke eksplisitt uttrykt av Kilpatrick et al. (2001) som en forutsetning for å utvikle matematisk kyndighet. *Modelleringskompetansen* kan likevel falle inn under for eksempel komponenten *konseptuell forståelse*, ved å se hvorfor en modell representerer forhold fra virkeligheten på en bedre måte enn andre.

Man kan se at det er store likheter som gjør at en kan trekke flere paralleller mellom Kilpatrick et al. (2001) og Niss & Højgaard Jensen (2002). Samtidig uttrykker de begge at alle komponentene eller delkompetansene ikke er sterkt adskilte, men har et gjensidig tett forhold til hverandre. Det vil ut fra dette være naturlig å kunne se på disse to rammeverkene som to ulike tilnærminger til en helhetlig matematisk kompetanse. De belyser begge store deler av det samme området, men beskriver det på ulike måter – en styrke jeg har valgt å benytte meg av i studien.

2.3.3.1. ET SAMMENSATT TEORETISK RAMMEVERK FOR MATEMATISK KOMPETANSE

I denne oppgaven er det som nevnt valgt å bruke fellesbetegnelsen *matematisk kompetanse*. I et eget sammensatt rammeverk for denne kompetansen er det hovedsakelig lagt vekt på rammeverket av Kilpatrick et al. (2001), men med berikelse fra rammeverket av Niss & Højgaard Jensen (2002). Den helhetlige matematiske kompetansen er delt inn ved bruk av de fem matematiske komponentene – *konseptuell forståelse*, *beregning*, *strategisk kompetanse*, *resonnering* og *engasjement*.

Kilpatrick et al. (2001) sine fem matematiske komponenter presenterer færre kategorier som på en mer generell måte beskriver hva som kan falle inn under kompetansebegrepet, sammenlignet med Niss & Højgaard Jensen (2002). De fem komponentene ser også ut til å fungere som paraplybegrep hvor kjernen i den matematiske kompetansen Niss & Højgaard Jensen (2002) beskriver er innesluttet. Niss & Højgaard Jensen (2002) har mer detaljerte og konkrete

skildringer av hvordan matematisk kompetanse utarter seg. Disse fungerer derfor godt som ledetråder og «knagger» for å identifisere den matematiske kompetansen uttrykt i elevenes samtale. Dette vil berike utdypelsen av de matematiske komponentene. På denne måten utfyller de to rammeverkene hverandre slik at de sammen kan utgjøre et nytt teoretisk rammeverk for å undersøke elevenes matematiske kompetanse.

2.4. PAPERT OG KONSTRUKSJONISMEN

Seymour Papert var en mann som etter allmenn oppfatning kan sies å ha vært forut for sin tid. I boken hans «Mindstorms», som først ble utgitt i 1980, forteller han om hvordan datamaskiner mer og mer vil bli en del av folks hverdag. Ønsket hans med denne boken var derfor å snakke om hvordan datamaskiner påvirker ens tenkning og læring (Papert, 1993, s. 3). Her diskuterer han hvordan datamaskinens tilstedeværelse kan bidra til mentale prosesser. De påvirker hvordan mennesker tenker, ikke bare instrumentelt, men på mer essensielle, konseptuelle måter, etter datamaskinen ikke lenger er fysisk tilstede (Papert, 1993, s. 4).

Papert har et særskilt fokus på barn og læring. I boken «Mindstorms» deler han allerede i 1993 sin visjon om hvordan han ser for seg at *space-age objects* i form av små datamaskiner, vil krysse de kulturelle barrierer og nå den private verdens barn over alt (Papert, 1993, s. 4). Som nevnt tidligere var dette imidlertid ukjente og fjerne tanker for mange den gangen, og fikk ikke skikkelig feste i syn på læring og utdanning. Likevel, bare noen få titalls år etter denne boken, har Papert sin visjon vist seg å stemme. Flere og flere klasser i skolen blir nå *digitalisert* – på den måten at hver elev har en personlig PC eller nettbrett for læring i skolen.

Matematikk er av Papert brukt som eksempel til å beskrive interaksjon mellom teknologi og sosiale prosesser for å konstruere ideer om menneskelig kapasitet. I tillegg er den brukt til å underbygge en teori som beskriver hvordan læring fungerer (Papert, 1993, s. 7). I boken *Mindstorms* støtter han seg på en modell av Jean Piaget, om hvordan barn selv er «byggmestere» for sine egne intellektuelle strukturer. Han kaller dette for «læring uten å bli lært» («learning without being taught») (Papert, 1993, s. 7). For å bygge trengs det materialer, og på dette punktet sier Papert selv at han skiller seg fra Piaget sine tanker. Papert har andre tanker om den rollen han tilskriver de omkringliggende kulturer som kilde til disse materialene. Han skriver at Piaget ville forklart en langsom læringsprosess av visse konsept gjennom deres

omfattende kompleksitet eller formalitet. Papert skriver at han selv forklarer den kritiske faktoren som en relativ fattigdom i kulturen i de materialene som ellers kunne gjort konseptene mer konkret og enkle. Andre ganger kan kulturen tilby materialene, men stå i veien for bruken (Papert, 1993, s. 7). For å legge til rette for læring holder det derfor ikke å kun tilby «materialene». Det må skapes en kultur rundt som legger til rette for læring, og ikke blokkerer for at «byggeprosessen» kan utføres av «byggmesteren».

Første gang begrepet *constructionism* ble introdusert, som på norsk er oversatt til *konstruksjonisme*, var i 1991 i artikkelen «Situating Constructionism» av Papert (1991). Her blir ideen innledningsvis forklart gjennom å tenke på konstruksjonen av kunnskap som «*learning-by-making*». I følge konstruksjonismen oppstår læring som en prosess der de lærende konstruerer eller skaper kunnskap i sinnet etter hvert som de samhandler med ting og mennesker i verden rundt dem (Logo Foundation, 2015a). For å bedre forstå hva denne teorien går ut på vil jeg derfor bevege fokuset over på teoriens ringe begynnelse. Inspirasjonen til tankene bak konstruksjonismen er en anekdote fra Papert selv. Anekdoten er fra en hendelse cirka 20 år før han introduserer konstruksjonismen som et syn på læring;

Han var hentet inn til en skole som hadde satt i gang et prosjekt i matematikk. Elevene skulle arbeide i LOGO. Dette er et programmeringsverktøy han selv på 1960-tallet, i samarbeid med et arbeidsteam, utformet med tanke på å bruke det i undervisning (Logo Foundation, 2015b). Men det var ikke dette som fanget hans oppmerksomhet. Det var klasserommet han passerte på veien. Her arbeidet elevene med såpeskulpturer, og han la merke til hvordan elevene arbeidet med dette. Såpeskulpturprosjektet pågikk i flere uker. Det tillot elevene å tenke, drømme, betrakte, få en ny idé, prøve ideen og enten droppe den eller gå videre på den. Elevene hadde tid til å snakke, se andre elevers arbeid og deres reaksjon til eget arbeid. Ikke så ulikt matematikk for en matematiker, men veldig annerledes enn den tradisjonelle matematikkundervisningen. Seymour Papert så det som en drømmesituasjon hvis matematikken kunne bli som dette. Denne måten å arbeide på ble lenge værende i hans tanker som «*soap-sculpture math*» (såpeskulpturmatematikk), før ideen videre førte til det som ble et vendepunkt og opphavet for konstruksjonismen (Papert, 1991).

Med Papert og konstruksjonismens perspektiv på læring legges det vekt på læring gjennom å være en aktiv deltaker i læringsarbeidet. Læring skjer mens man er i en prosess – altså mens

man konstruerer noe. Å komme frem til en løsning er ikke det viktigste for å lære, men prosessen mot målet bidrar til læring og utvikling. Det sosiale er også en viktig faktor som fremheves i konstruksjonismen. Enten det er å snakke sammen, dele ideer eller å ta andres reaksjoner på eget arbeid i betraktning. Både å samarbeide og å arbeide alene for så å kunne få tilbakemelding av andre, vil ut fra dette læringsperspektivet bidra til økt læring (Papert, 1991, 1993).

3.METODE

I dette kapitlet vil jeg gjøre rede for oppgavens metodiske tilnærming (3.2). Flere valg og avgjørelser er blitt gjort i løpet av gjennomføringen av prosjektet, noe som har fått betydning for studien. Jeg vil ta for meg de valg og forberedelsene som er blitt gjort i forbindelse med innsamling av data (3.3) og utforming av analyseverktøy (3.4). Jeg vil også påpeke etiske avveielser og hensyn som er tatt (3.5), og valg som er gjort med tanker på reliabilitet og validitet i forhold til datamaterialet og metoden for innsamling (3.6). Datainnsamlingen til masterprosjektet er gjort i samarbeid med min medstudent Oda Åbø Pettersen. Samarbeid om datainnsamlingen er en anerkjent metode for innsamling av. Å være to i arbeidet med datainnsamlingen har lettet arbeidsmengden og åpnet for muligheten til å håndtere et større utvalg informanter. Oda har også vært en god samtalepartner, slik at vi sammen har fått et solid datagrunnlag for studiene våre. Avgjørelser som er tatt i forbindelse med datainnsamlingen og forberedelser til denne er derfor gjort i samarbeid med henne. Det kjennes derfor naturlig å referere til *vi* i noen sammenhenger i denne delen, med *vi* menes meg og min medstudent Oda som har samarbeidet om datainnsamlingen til hvert vårt studie.

3.2. METODISK TILNÆRMING

Samfunnsvitenskapen baserer seg på systematisk forskning som bygger på ulike forhold i samfunnet, og handler om menneskers liv og virksomheter i samfunnet (Grønmo, 2016). Studien baserer seg på virksomheter i skolen, som inngår i samfunnsmessige forhold, og det er derfor naturlig å anvende en samfunnsvitenskapelig metode.

Av de to hovedformene kvalitativ og kvantitativ forskningsmetode har jeg valgt en kvalitativ tilnærming til innsamling og analyse av data, som har færre og mer detaljerte datapunkter enn kvantitativ. I analysen er det tatt en kvalitativ fenomenologisk tilnærming, i lys av hermeneutisk vektlegging av fortolkning og forståelse. Dette er for å kunne få en dypere forståelse for de spørsmål oppgaven reiser gjennom problemstillingen og forskningsspørsmålene. Jeg vil videre referere til oppgavens metode som en kvalitativ hermeneutisk tilnærming, da en fenomenologisk tilnærming som denne kan falle inn under det hermeneutiske begrepet.

I kvalitativ forskning er en fenomenologisk tilnærming, i en ikke-filosofisk forstand, utbredt. Fenomenologi er i denne sammenheng et begrep for interessen i å forstå sosiale fenomen ut fra aktørens egne perspektiv, og beskrive verden slik informanten opplever den. Dette gjøres ut fra den forståelsen av at den virkelige virkeligheten er den mennesker oppfatter (Kvale & Brinkmann, 2009, s. 45). Fenomen som sies å uttrykke mening eller ha en betydning, kalles for meningsfulle fenomen. Blant meningsfulle fenomen finner vi tekster, språklige uttrykk og menneskelige handlinger. At fenomenene må fortolkes og kunne forstås er karakteristisk for slike fenomen, og går med dette under *hermeneutikken* (Gilje & Grimen, 1993, s. 142-143).

Hermeneutikk er læren om tolkninger (Mellin-Olsen & Lindén, 1996, s. 5). I samfunnsvitenskapen er hermeneutikken relevant, fordi datamateriale ofte består av meningsfulle fenomener, som handlinger, muntlige ytringer og tekster (Gilje & Grimen, 1993, s. 144). Fortolkning og forståelse utgjør en fundamental og helt vesentlig del av samfunnsvitenskapelige fag. Det er fordi store deler av forskningsprosessen i samfunnsvitenskapen kan oppfattes som en fortolkningsprosess (Gilje & Grimen, 1993, s. 144).

I denne studien er det gjennomført en kvalitativ forskning med en hermeneutisk tilnærming, i det jeg prøver å forstå elevenes handlinger i Scratch ut fra elevenes egne perspektiver og slik de opplever dem. For å gjøre dette er det valgt å ta utgangspunkt i fire elevpar sitt arbeid med programmering i Scratch. I en studie som denne har jeg som forsker betydning for forskningsprosessen og resultatene. Både prosessen og resultatene vil derfor være personavhengig (Mellin-Olsen & Lindén, 1996, s. 21). Objektiviteten i forskningen vil med bakgrunn i dette knyttes til i hvor stor grad andre vil godta det jeg som forsker har tolket. Hvilken mening som tillegges de fenomen som tolkes og er forsøkt å forstå vil kunne variere fra aktør til aktør (Gilje & Grimen, 1993, s. 143).

En grunntanke i hermeneutikken er at man alltid vil forstå noe ut fra grunnlaget av de forutsetninger man har. Hans-Georg Gadamer (1900-2002) står bak de mest kjente begrepene for slike forutsetninger, og kalte disse *forforståelse* eller *for-dommer* (Gadamer, 2004; Gilje & Grimen, 1993). For å i det hele tatt kunne forstå noe, er forforståelse et nødvendig vilkår (Gilje & Grimen, 1993, s. 148). I denne studien vil den forforståelse og bakgrunnskunnskap jeg har som forsker være en del av forståelsesprosessen i arbeidet med å forstå og tolke det teoretiske grunnlaget og datamaterialet i analysen.

Grensen for de forståelsesmuligheter enhver har, forklares med begrepet *horisont* (Kjørup, 2008, s. 75). Vår forståelseshorisont grunnes på vår forforståelse, og kan utvides ved åpenhet. Gadamer (2004, s. 291) beskriver forståelse som en prosess der horisonter, som hver for seg eksisterer, smelter sammen. Dette er målet med forståelse, og kalles en *horisontsammensmeltning*. Dette leder videre til det kanskje mest sentrale begrepet i hermeneutikken – *den hermeneutiske sirkel*. Denne betegnelsen brukes om den kontinuerlige prosessen hvor en gjennom all fortolkning beveger seg mellom del og helhet. Det kan være mellom det som skal fortolkes og konteksten det fortolkes i, eller mellom det som fortolkes og ens egen forforståelse (Gilje & Grimen, 1993, s. 153). I denne spiralen åpnes det for stadig dypere forståelse av meningen (Kvale & Brinkmann, 2009, s. 217).

I denne studien vil for eksempel slik jeg har tolket og forstått de ulike delene av det teoretiske bakteppet, påvirke den forståelsen jeg vil formidle som forsker gjennom analysen. All annen teori og bakgrunn som er benyttet vil også være ledende gjennom den fortolkningen som er gjort i analysen. Ut fra et hermeneutisk perspektiv kan man si at oppgaven presenterer ulike deler, som sammen virker inn på en større helhet. Denne helhetens mål er å undersøke og belyse oppgavens forskningsspørsmål og problemstilling.

3.3. DATAINNSAMLING

Bjørndal (2017, s. 12-13) peker på at vurderinger gjøres på grunnlag av rimelige konstateringer. Er konstateringsprosessen grundig, vil man være sikrere på at vurderingene fører til gode avgjørelser, og ikke er preget av tilfeldigheter. I denne delen vil valg som er tatt med tanke på utvalg og kontekst bli gjort rede for. Utvalg og kontekst for datainnsamlingen er avgjort etter kriterier for hva jeg og Oda har vurdert som mest hensiktsmessig i henhold til det oppgavene våre skal undersøke. I delkapitlet 3.3.1 er utvalg og kontekst utdypet.

For å få skape gode forutsetninger til datainnsamlingen og utforming av oppgaver til denne, er det gjort ulike forberedelser. Sammen med Oda har jeg satt meg inn i programmeringsverktøyet Scratch. Vi har også vært i kontakt med Kodeklubben - en frivillig organisasjon for barn og unge som bidrar til lek og læring med programmering (<http://www.kodeklubben.no/>). I tillegg har vi vært delvis deltakende observatører på et Scratch-kurs i regi av Kodeklubben. En observasjon vi gjorde her var at oppgavene elevene utførte var sterkt preget av oppskrifter som

skulle følges. Med utgangspunkt i denne observasjonen ble tanken om å prøve ut noen mer utforskende og åpne oppgaver styrket.

Tanker og erfaringer som ble gjort under observasjonene fra Scratch-kurset ble med på å forme den videre prosessen i å utforme oppgaver til pilotgjennomføringen og gjennomføringen av hovedopplegget – videre kun kalt for *Opplegget*. Utforming av oppgaver i Scratch er kort beskrevet i delkapittelet 3.3.2. Dokumentasjon av datainnsamlingen ble gjort med video-, lyd- og skjermopptak, og er omtalt i delkapittel 3.3.3. Videre er det i delkapitlet 3.3.4 gjort rede for gjennomføringen av datainnsamlingen ved å nærmere beskrive de ulike fasene, som er *Pilot (3.3.4.1) tilpassing av oppgaver (3.3.4.2) og Opplegget (3.3.4.3)*.

3.3.1. UTVALG OG KONTEKST

Etter å ha vært i kontakt med Kodeklubben for å få tak i elever som kunne ha kjennskap til programmering i Scratch, ble det knyttet kontakt med en klasse på 7. trinn. Et utvalg på åtte elever ble tilfeldig valgt ut til å delta i datainnsamlingen. Disse elevene var allerede noe kjent i Scratch fra tidligere gjennomførte kurs i regi av Kodeklubben. Dette var ikke et direkte krav som ble stil, men en fordel vi valgte å dra nytte av.

At det er sett som hensiktsmessig med en kvalitativ tilnærming i denne studien har medført et mindre utvalg av informanter. Ved dette har ressursene under gjennomføringen av prosjektet blitt brukt mer konsentrert, i stedet for å bli fordelt på et større utvalg. Avgjørelser er tatt med større tilpasning for elevgruppen, og forberedelsene har blitt mer presise i henhold til utvalget. Dette har ført til at analysen har kunne gitt en dypere innsikt i datamaterialet i henhold til forskningens formål.

De deltakende elevene ble satt i par under gjennomføring av datainnsamlingen. Bakgrunnen er at elevenes samarbeid og kommunikasjon er av betydning for å belyse oppgavens spørsmål ut fra de gitte teoretiske rammene. Utvalget på åtte elever ble satt i henholdsvis fire par for å sikre et rikt nok utvalg uten et for omfattende etterarbeid. Det ble ikke gitt noen føringer til sammensetningen av elevparene.

3.3.2. UTFORMING AV OPPGAVER I SCRATCH

Oppgavene som er laget til dette prosjektet (Vedlegg I) er preget av de observasjoner gjort i Scratch-kurset. Elevene som har deltatt i dette masterprosjektet hadde som nevnt tidligere vært med på et lignende Scratch-kurs. På kurset våre informanter hadde deltatt på ble det arbeidet med samme oppskriftsbaserte oppgaver fra Kodeklubben sin nettside (kodeklubben.no) som elevene på kurset vi observerte, beskrevet ovenfor. Derimot har den siste oppgaven i begge kursene utpekt seg fra de tidligere oppgavene i samme kurs. I den siste oppgaven var oppdraget å lage et spill uten å følge en bestemt fremgangsmåte. På bakgrunn av dette bestemte vi oss for å utforme mer åpne oppgaver som la opp til samarbeid, kreativitet og problemløsning. Først ble oppgaver til piloten utformet (Vedlegg I). Under gjennomføringen av disse erfarte vi at det ble mye fokus på det tekniske som tok fokus fra det matematiske. Det ble derfor gjort tilpassinger av oppgavene før gjennomføring av Opplegget (Vedlegg II). Dette er gjort rede for i egne avsnitt (3.3.4).

3.3.3. DOKUMENTASJON AV DATAINNSAMLINGEN

Video- og lydopptak oppfattes gjerne som det sterkeste beviset vi har på hva og hvordan noe har skjedd. Det er derfor viktig å påpeke at dette ikke gir en kopi av virkeligheten, men kun representerer den (Bjørndal, 2017, s. 82). Fordelen ved å bruke video- og lydopptak er at det kan gi en detaljert ikke-tolket gjengivelse av det som skjer i situasjonen (Tjora, 2017, s. 103).

For å samle data som kunne være mulig å analysere til prosjektet, ble det valgt å bruke video- og lydopptak i tillegg til skjermopptak av de utvalgte elevparenes arbeidsøkt med Scratch. Det ble valgt å kombinere de tre ulike opptakene for å styrke reliabiliteten og validiteten til datamaterialet. For å sikre god lyd kvalitet ble det koblet til ekstra lydutstyr til videokameraet. Å sikre god lyd kvalitet av elevsamtalene styrket muligheten for en mest mulig korrekt gjengivelse av det som ble sagt i gjennomføringen av datainnsamlingen. Dokumentasjonen som er gjort har gitt et omfattende og massivt datamateriale. Det ble derfor valgt å bruke videobildene kun som supplering, mens lyd- og skjermopptak vil bli lagt størst vekt på. Videoopptak ble gjort for å lettere kunne se helheten av situasjonen, sammenhengen mellom samtalen og arbeidet på datamaskinen, og å kunne fange opp eventuelle relevante gestikuleringer.

Bruk av video- og lydopptak tilfører imidlertid datainnsamlingen en ekstra dimensjon av etiske hensyn i henhold til behandling av personidentifiserbare opplysninger. Dette er omtalt senere i eget avsnitt (kapittel 3.5). Skjermopptak gir et klart bilde av hva som foregår på skjermen hvor selve oppgaven i Scratch er utført.

Datamaterialet fra piloten og Opplegget er i etterkant slått sammen til én film for hver av arbeidsøktene. Dette ble gjort med en bilde-i-bilde-funksjon. På denne måten ble video- lyd- og skjermopptak fra de ulike arbeidsøktene synkronisert. Dette ble gjort for å bedre kunne se sammenhengen mellom de ulike mediene.

3.3.4. GJENNOMFØRING AV DATAINNSAMLING

Datainnsamlingen er bygd opp av tre deler. Først ble det gjennomført en pilot. Erfaringer herfra ble så tatt i betraktning for å få best mulig tilpassede oppgaver til Opplegget. De tre delene, *pilot*, *tilpassing av oppgaver* og *Opplegget* er beskrevet nedenfor i de følgende tre delkapitler.

3.3.4.1. PILOT

I pilotgjennomføringen ble de åtte utvalgte elevene satt i par for å samarbeide med oppgavene vi hadde forberedt (Vedlegg I). Gjennomføringen ble delt i to deler der to og to elevpar gjennomførte oppgavene samtidig. Tiden som ble satt til gjennomføringen var cirka 45 minutter (en vanlig skoletime). På forhånd ble det bestemt at meg og min medstudent skulle fungere som veiledere. Vi skulle være så passiv som mulig for å få dem til å tenke selv, men ikke i en slik grad at de stoppet opp i arbeidet. I starten ble oppgaven gått igjennom med elevene for å oppklare eventuelle uklarheter. Underveis ble det gitt hjelp til grunnleggende tekniske funksjoner i Scratch og stilt hjelpende spørsmål om elevene trengte veiledning for å komme seg videre. Slike hjelpende spørsmål kunne være «Hva har dere tenkt her?» eller «Finnes det andre måter å løse det på?».

Hensikten med piloten var at vi ønsket å få innsikt i hva elevene egentlig kunne i Scratch fra før, hvordan de arbeidet sammen i par og hvilken hjelp vi så var nødvendig. Vi ønsket også å forberede elevene til Opplegget i form av å gjøre dem trygge på hverandre, programmet Scratch,

opptaksutstyret og oss som nye voksne. I korte trekk var hensikten med pilotgjennomføringen å sikre et best mulig utgangspunkt for andre arbeidsøkt i Scratch (Opplegget).

3.3.4.2. TILPASSING AV OPPGAVER

Oppgavene som ble laget til pilotgjennomføringen (Vedlegg I) var i første omgang ment til å gjøre elevene kjent med nødvendige funksjoner og kommandoer. Etter piloten kom vi frem til at det måtte gjøres noen større endringer på oppgavene til Opplegget. Blant annet fikk vi inntrykk av at elevene var ganske usikker på programmering i Scratch. Vi vurderte derfor at det kom til å bli for vanskelig og tidkrevende med det vi først hadde tenkt til Opplegget. Konsekvensen av piloten var at vi gikk bort fra ideen om å la elevparene lage regnefortelling ved hjelp av det de gjorde i pilotgjennomføringen. Med utgangspunkt i de erfaringene vi gjorde og refleksjoner vi delte i etterkant, ble det bestemt å endre oppgavene til å programmere geometriske figurer. Bakgrunnen for dette valget var at vi observerte under piloten at elevene brukte mest tid på første oppgave. Denne gikk ut på at de skulle få Scratch-katten til å gå i firkant. Vi tenkte derfor at videre arbeid med geometriske figurer kom til å gi en større mulighet til å få frem matematisk kompetanse hos elevene.

3.3.4.3. OPPLAGGET

Med erfaring fra piloten ble Opplegget gjennomført med de samme elevgruppene, men med nye oppgaver. Oppgavene gikk ut på at elevene skulle samarbeide for å programmere geometriske figurer i Scratch (Vedlegg II). Vår rolle var her den samme som ved piloten. Vi ga dem heller ikke her direkte hjelp til hvordan oppgaven skulle løses. Før elevene satte i gang ble oppgavearket gjennomgått. Enkelte tekniske operasjoner ble også vist, som «penn»-funksjonen, «når ___ trykkes»-funksjonen (kommandotasten til skriptet) og hvordan de kunne slette og angre ulike komponenter i Scratch (se Figur 1: Skjerm bilde av Scratch med egne beskrivelser av programmet). I oppgaven ble det forklart at penn-figuren i programmet skulle tegne de geometriske figurene. Dette skulle gjøres ved å programmere penn-figuren slik at man til slutt kunne trykke på én gitt kommandotast for å få hele den geometriske figuren til å vise i Scratch. Det var ikke et alternativ å tegne inn figuren på andre måter enn ved å programmere

den ved hjelp av funksjons-klosser. Oppgaven påpekte også at de skulle forklare hvordan de tenkte overfor den andre og lytte til hverandres forslag (Vedlegg II).

3.4. ANALYSEVERKTØY

Bjørndal (2017, s. 12-13) skriver at vurdering bygger på et utvalg av kriterier. Skal man vurdere kvaliteten på en virksomhet vil det være nødvendig med noen kriterier å vurdere ut fra. Til å belyse de to forskningsspørsmålene i denne oppgaven er det benyttet to ulike teoretiske angrepsvinkler som utgjør kriterier for analysen. Første forskningsspørsmål er belyst i analysen gjennom det nye rammeverket, satt sammen av rammeverket av Kilpatrick, Swafford & Findell (2001) og rammeverket av Niss & Højgaard Jensen (2002). Gjennom fem matematiske komponenter er det undersøkt hvordan matematisk kompetanse kommer til uttrykk i elevers samtaler når de programmerer. Hvordan det nye sammensatte rammeverket er brukt som analyseverktøy er omtalt i delkapittel 3.4.1. Det andre forskningsspørsmålet går ut på å undersøke hvordan elever konstruerer matematisk kunnskap når de programmerer. Dette forskningsspørsmålet er undersøkt ved å analysere samtaleutdrag i lys av Seymour Papert (1991, 1993) og konstruksjonismens syn på læring. Hvordan dette læringssynet er brukt som analyseverktøy i studien er utdypet i delkapittel 3.4.2. Gjennom å undersøke de to forskningsspørsmålene er oppgavens problemstilling belyst – som går ut på hvordan programmering kan fremme matematisk kompetanse hos elever.

På bakgrunn av at oppgaven er basert på en kvalitativ hermeneutisk tilnærming, befinner jeg meg i et fortolkningslandskap. Analyseverktøyet består derfor av relevant teori som ved fortolkning skal kunne gi svar på problemstillingen med forskningsspørsmål gjennom analysen av det utvalgte datamaterialet. Bearbeidelsen og analysen av datamaterialet har gått igjennom en analyseprosess bestående av flere deler. I delkapittel 3.4.3 er analyseprosessen beskrevet, hvor det er utdypet valg av situasjoner for videre analyse (3.4.3.1), transkribering (3.4.3.2) og koding av de utvalgte situasjonene fra elevsamtalene (3.4.3.3).

3.4.1. KILPATRICK, SWAFFORD & FINDELL (2001) OG NISS & HØJGAARD JENSEN (2002) SOM ANALYSEVERKTØY

Til å identifisere matematisk kompetanse i elevenes samtaler er teori av Kilpatrick et al. (2001) satt i sammenheng med Niss & Højgaard Jensen (2002) til et nytt og sammensatt teoretisk rammeverk. I teorikapitlet (kapittel 2.3.3 og 2.3.3.1) er det gjort rede for hvordan disse to teoriene styrker hverandre ved at de på hver sin måte kan sies å dekke en helhetlig matematisk kompetanse. Som nevnt her er det valgt å bruke begrepene og inndelingen fra Kilpatrick et al. (2001) og dele inn matematisk kompetanse i de fem komponentene *konseptuell forståelse, beregning, strategisk kompetanse, resonnering og engasjement*.

Niss & Højgaard Jensen (2002) sin teori om de åtte matematiske delkompetansene er i denne sammenheng valgt som en supplerende og berikende teori til Kilpatrick et al. (2001), ettersom komponentene fra Kilpatrick et al. (2001) er brukt som utgangspunkt for analysen. Beskrivelsene Niss & Højgaard Jensen (2002) bruker om de matematiske delkompetansene er gjort med andre ord og uttrykk enn slik Kilpatrick et al. (2001) beskriver de matematiske komponentene. Da de begge omtaler hva matematisk kompetanse er, kan dette dermed berike min fortolkning og forståelse av de matematiske komponentene. På denne måten er teorien anvendt for å identifisere og videre analysere den matematiske kompetanse elevene uttrykker gjennom samtale i arbeid med matematikk ved programmering i Scratch.

For å bedre forklare for hvordan det nye sammensatte rammeverket er anvendt som analyseverktøy, vil jeg komme med noen beskrivende eksempler:

Vi kan se på komponenten *beregning*. Beskrivelsen av denne går ut på at en har kunnskap om prosedyrer eller matematiske operasjoner. Kilpatrick et al. (2001) beskriver denne komponenten nærmere ved at en har kunnskap om når og hvor de er hensiktsmessig, og at en viser ferdighet til å uttrykke dem fleksibelt, nøyaktig og effektivt. Niss & Højgaard Jensen (2002) beskriver blant annet *representasjonskompetansen* ved at en kan håndtere ulike representasjoner av matematiske omstendigheter. Videre går *symbol- og formalismekompetansen* ut på å kunne håndtere symbol og formaliteter i matematiske representasjoner. Dette innebærer å kunne omforme symbol- og formelspråk og oversette mellom matematiske språk og naturlig språk. For å kunne uttrykke kunnskap om prosedyrer eller matematiske operasjoner vil det være nødvendig å gjøre seg bruk av blant annet

representasjonskompetansen og symbol- og formalismekompetansen. Niss & Højgaard Jensen (2002) har dermed en beskrivelse av delkompetanser som kan passe godt til å kunne identifisere komponenten *beregning*. Disse tilfører på denne måten en mer konkret beskrivelse av hva som kan inngå i det å uttrykke *beregning* fleksibelt, nøyaktig og effektivt og å vite når og hvor matematiske prosedyrer og operasjoner er hensiktsmessig.

Representasjonskompetansen går også ut på å forstå de ulike representasjonene i form av å omforme, tolke og skille mellom matematiske objekt, fenomen, problem eller situasjoner. Forståelse for de ulike representasjonene er en vesentlig del av komponenten *konseptuell forståelse* for å kunne forstå hvorfor en matematisk idé er viktig, og i hvilken kontekst denne kan være nyttig. Kilpatrick et al. (2001) utdyper videre komponenten *konseptuell forståelse* ved at en kan være i stand til å kunne representere matematiske situasjoner på ulike måter, og vite hvordan de kan være hensiktsmessige til ulike formål. Om elevene omformer, tolker og skiller mellom matematiske objekt, fenomen, problem eller situasjoner, kan dette brukes som en utvidet beskrivelse å gå etter for å identifisere *konseptuell forståelse*.

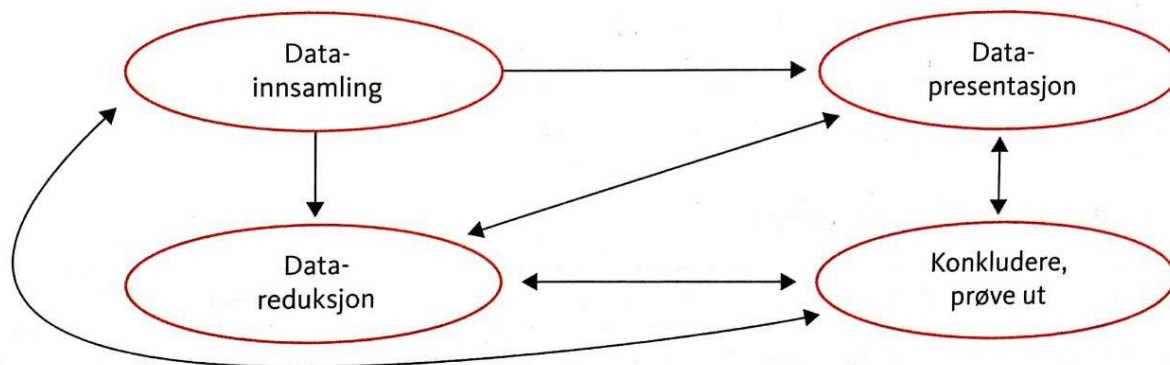
Flere av Niss & Højgaard Jensen (2002) sine beskrivelser av de ulike matematiske delkompetansene kan passe inn under Kilpatrick et al. (2001) sine matematiske komponenter. Dette gjør beskrivelser av komponentene mer konkrete og enklere å identifisere. De matematiske komponentene kan på denne måten beskrives mer konkret ved å se disse to teoriene i sammenheng. Begrepene for Kilpatrick et al. (2001) sine matematiske komponenter er også beskrevet på en måte som gjør at de kan inneslutte flere av de matematiske delkompetansene til Niss & Højgaard Jensen (2002). Det er derfor valgt å legge størst vekt på teori av Kilpatrick et al. (2001) i analysen av datamaterialet. Niss & Højgaard Jensen (2002) sine beskrivelser av de ulike matematiske delkompetansene er brukt til å bedre få en forståelse av hva som kan legges til grunn for de fem matematiske komponentene gjennom mer konkrete beskrivelser, slik som eksemplene over viser. Det er derfor ikke lagt særlig vekt på hva Niss & Højgaard Jensen (2002) har kalt de ulike delkompetansene, men i stedet den beskrivelsen av hva som inngår i en helhetlig matematisk kompetanse for å skape et fyldig teoretisk rammeverk for forståelsen av dette begrepet.

3.4.2. PAPERT OG KONSTRUKSJONISMEN SOM ANALYSEVERKTØY

Studiens andre forskningsspørsmål tar sikte på å skape en dypere forståelse av hvordan elever konstruerer matematisk kunnskap når de programmerer. Det er derfor tatt utgangspunkt i slik Seymour Papert (1991) har beskrevet sine ideer om læring og å konstruere kunnskap. Papert er mannen med ideene som har lagt grunnlaget for konstruksjonismen. Ideen om «*learning-by-making*» er sentral i dette læringsperspektivet, hvor læring ifølge konstruksjonismen oppstår i en prosess. I denne prosessen konstruerer eller skaper de lærende kunnskap i sinnet etter hvert som de er i samhandling med mennesker og verden rundt dem (Logo Foundation, 2015a). For å undersøke hvordan elever konstruerer matematisk kunnskap er det tatt særlig utgangspunkt i anekdoten av Papert om noen elevers arbeid med et såpeskulpturprosjekt, som er beskrevet tidligere i kapittel 2.4. I beskrivelsen av denne hendelsen trekker han frem hvordan denne måten å arbeide på tillot elevene å tenke, drømme, betrakte, få en ny idé, prøve den for så å gå videre eller droppe den, tid til å snakke, se andre elevers arbeid og deres reaksjon til eget arbeid (Papert, 1991). Disse beskrivelsene har gitt meg noen konkrete stikkord å gå etter for å undersøke hvordan elevene i denne studien konstruerer matematisk kunnskap når de programmerer i Scratch.

3.4.3. ANALYSEPROSESSEN

Det finnes ingen standardiserte teknikker for hvordan kvalitativ data skal analyseres, noe som kan gjøre analyseprosessen kompleks. Hensikten med analyse av data er å bevege seg fra observasjoner til vitenskap (Ringdal, 2018, s. 252). Analyseprosessen for dette prosjektet kan illustreres gjennom en modell av Miles & Huberman (1994) referert i Ringdal (2018, s. 252). Modellen representerer en generell framstilling av en interaktiv analyseprosess for kvalitativ data (Figur 4).



FIGUR 4: EN INTERAKTIV MODELL FOR ANALYSE AV KVALITATIVE DATA. KILDE: MILES & HUBERMAN (1994) I RINGDAL (2018, s. 252)

Analysen av datamaterialet består av de tre elementene *datareduksjon*, *datapresentasjon* og å *konkludere/verifisere* (Ringdal, 2018, s. 252). *Datareduksjon* er en viktig del av analysen, og er blitt gjort her for å gjøre datamaterialet mer håndterbart og overkommelig i omfang. Datamaterialet i denne studien består av fire videoer på cirka 45 minutter hver. Ut fra dette er det gjort en *datareduksjon* der fire situasjoner, en fra hver gruppe på om lag 10-15 minutter hver, er valgt ut til analysen. *Datapresentasjon* har også en viktig rolle i analysen (Ringdal, 2018, s. 252). *Datapresentasjon* av de fire utvalgte situasjonene er gjort ved at de er blitt transkribert. Datamaterialet er videre redusert ved at transkripsjonene er kodet ut fra det nye sammensatte rammeverket. Resultatet fra kodingen er så presentert ved at de er satt inn i tabeller og diagram (kapittel 4.2). Kodingen er gjort for å kartlegge og skape oversikt over datamaterialet opp mot teorien for det første forskningsspørsmålet. Denne er blitt til hjelp i den videre analysen av enkeltutdrag. Datamaterialet er så ytterligere redusert i analysen. Her er enkeltutdrag fra transkripsjonene presentert og analysert i henhold til forskningsspørsmålene og det todelte teoretiske bakteppet. Spiren til konklusjonen har vokst frem under analyseprosessen (Ringdal, 2018, s. 253). Som forsker er jeg innom elementer for konklusjonen flere ganger ved det som fremkommer gjennom ulike prosesser i analysen, men den endelige konklusjonen kommer ikke før i oppgavens avslutning.

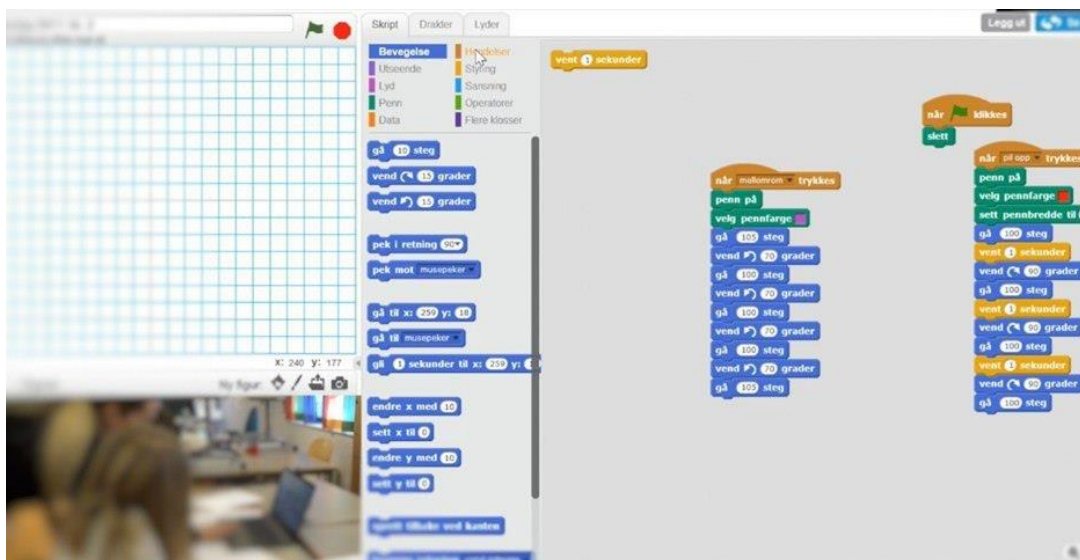
3.4.3.1. UTVALGTE SITUASJONER

Situasjonene som er valgt ut for videre analyse er hentet fra Opplegget (se kapittel 3.3.4.3). I de utvalgte situasjonene arbeider hver av de fire gruppene med å programmere en femkant i

Scratch. Det er valgt å ta en situasjon fra hver av de deltakende gruppene for å få med mangfoldet av elever som deltok. Grunnen til at femkant-situasjonen er valgt fra alle gruppene, er for å kunne skape et så likt sammenligningsgrunnlag som mulig. Elevene hadde da vært igjennom de samme oppgavene før femkanten og dermed fått gjort seg erfaringer i Scratch. Av de ulike geometriske figurene som skulle lages var femkanten den figuren alle gruppene fikk gjennomført. Det var også her de i størst grad brukte ulike fremgangsmåter, og elevenes samtaler så ut til å være mest fruktbare i arbeidet med denne figuren. I denne studien er det derfor valgt å fokusere på alle de fire gruppens arbeid med å programmere en femkant.

3.4.3.2. TRANSKRIBERING

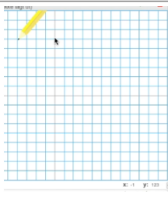

Datamaterialet som er benyttet i denne oppgaven består av transkripsjoner av fire situasjoner. Transkripsjonene er gjort fra video med lyd og skjermopptak av elevenes arbeid, satt sammen med bilde-i-bilde videofunksjon til én video (Figur 5). På denne måten er de ulike komponentene synkronisert for å bedre få oversikt over samspillet mellom elevenes samtale og arbeid i Scratch.



FIGUR 5: UTKLIPP AV BILDE-I-BILDE-VIDEO FRA GJENNOMFØRING AV PROSJEKTET

I transkripsjonene er de ulike partenes ytringer gitt ved H for eleven til høyre, V for eleven til venstre og Stud. for student (meg og min medstudent). Dette er gjort for å anonymisere elevene

og samtidig kunne gå tilbake i datamaterialet og vite hvem som sier hva etter hvor de sitter. Elevene er gitt fiktive navn i samtaleutdragene som er valgt ut i oppgavens analyse.

| Tid | Samtalen | Bemerkninger | I Scratch |
|-------|---|--|---|
| 34:10 | V: ok, la oss bare ta en og en ... H: Jeg tipper dette kommer til å ta like lang tid som sist gang. V: Vi begynner her ... [V flytter på pennfigur i PV] | Begynner på femkant. V styrer PC V ser på skjerm mens han skriver på ark. H følger med. |  |
| 34:20 | V: Der. Da skriver vi koordinatene ... [finder frem blyant og ark] V: her er minus en ... minus en ... en, to, tre. | |  |

FIGUR 6: UTDRAK FRA EGEN TRANSKRIPSJON AV DATAMATERIALET

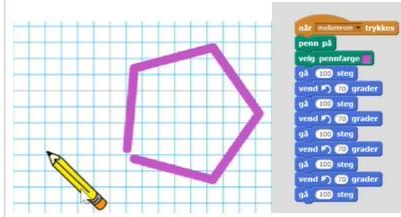
Transkripsjonene består av fire kolonner – «tid», «samtalen», «bemerkninger» og «i Scratch» (Figur 6). Tiden refererer til spilletiden i det totale opptaket av elevene – som f.eks. over er 34 min og 10 sekund inn i opptaket (Figur 6). Andre kolonne inneholder transkripsjon av samtalen. Dette er det som blir sagt og gjort, men er redusert til det som er forstått å være av betydning i utførelsen av oppgaven. Enkelte gester eller bevegelser elevene gjør som er antatt å ha betydning i analysen, er tatt med i klammer i transkripsjonene, som for eksempel i Figur 6 – [finder frem blyant og ark]. Noen ganger snakker elevene om ting som jeg har vurdert til å ikke være direkte relevant til oppgaveløsningen, som å lete etter en teknisk funksjon i Scratch, eller det er en stund hvor elevene er stille. Disse gangene er det lagt inn en parentes med en kort beskrivelse, og det som blir sagt er ikke transkribert ordrett. Hver gang elevene tester skriptet er dette tydelig markert i transkripsjonen av samtalen. Kolonnen i transkripsjonen med bemerkninger inneholder informasjon som kan være relevant for å få med mer av samtalen helhet. Funksjonsklossene elevene legger inn i skriptet er lagt til denne kolonnen og er satt i klammer. Siste kolonne er utklipp fra skjerm bilde av elevenes programmering i Scratch. Transkripsjonene er lagt ved i sin helhet i slutten av oppgaven (Vedlegg V – Vedlegg IX).

3.4.3.3. KODING

Koding av transkripsjonene fra de utvalgte situasjonene av elevenes samtaler er gjort ut fra det nye rammeverket jeg har utformet (se kapittel 2.3). Kodingen er gjort for å kartlegge og få en oversikt over hvordan de ulike komponentene av matematisk kompetanse kommer til uttrykk i elevenes samtale når de programmerer. I denne forbindelse vil jeg påpeke at kodingen bygger

på min fortolkning av datamaterialet opp mot rammeverket for kodene. Jeg som forsker spiller derfor en avgjørende rolle i kodingen og de resultatene som kommer av denne. Det kan derfor diskuteres hvor vidt andre ville kodet datamateriale med samme resultat, ettersom forståelse og tolkning er knyttet til personen som gjør det. For å styrke oppgavens reliabilitet og validitet er alle transkripsjonene lagt ved med kodene jeg har satt (Vedlegg V – Vedlegg IX). Dette vil bidra til å styrke studiens transparens av den forskingen som er gjort.

Til koding av datamaterialet er det laget en egen kolonne til venstre i transkripsjonene. Hver av de matematiske komponentene som er tatt utgangspunkt i her har fått kode etter første bokstav. Kodene benyttet for de ulike komponentene er [K] *Konseptuell forståelse*, [B] *beregning*, [S] *strategisk kompetanse*, [R] *resonnering* og [E] *engasjement*. Et utklipp fra koding av en transkribert situasjon er vist under for å illustrere:

| | | |
|---|---|--|
| <p>K S E</p> <p>K B S R E</p> <p>K S R</p> <p>K S R E</p> <p>K B S R</p> <p>K S R</p> | <p>19:58 --- (tester skript igjen) ---</p> <p>H: Yeey! Nei, var det? [ser på figur i PV]</p> <p>--- (student kommer og justerer utstyr) ---</p> <p>V: En, to, tre, fire, fem ... Jah! Det var det</p> <p>H: Ja, men den er jo ikke helt inntil? Er den?</p> <p>V: jo, jo, jo, jo ... du ser ikke det</p> <p>H: [henvender seg til stud.] Er den godkjent?</p> <p>V: [ser bort på student] Den er godkjent?!</p> <p>H: siden, se då ... den er ikke helt inntil [viser på figur]</p> |  <p>(19:58)</p> |
|---|---|--|

FIGUR 7: UTKLIPP FRA TRANSKRIPSJON MED KODER

Hver ytring fra elevene som er transkribert i kolonnen «samtaalen» er analysert for hver kode. Disse er gitt koder etter det jeg har tolket at ytringen gir uttrykk for. Om en ytring er blitt tolket til å passe med det teoretiske rammeverkets beskrivelse av den enkelte komponenten, er bokstaven for den aktuelle koden satt i kolonnen til venstre (se Figur 7). Antall koder som er gitt de enkelte ytringene varierer. Kodene er satt ut fra hva jeg har tolket å ligge til grunn for en ytring ut fra slik jeg har forstått og tolket det teoretiske rammeverket. Under vil jeg vise et eksempel på en enkel analyse av hvordan én ytring kan gi uttrykk for flere koder samtidig:

I andre ytring i Figur 7 sier H «En, to, tre, fire, fem ... Jah! Det var det». Denne er blitt gitt alle kodene [K B S R E] fordi det er tolket og forstått som at alle de matematiske komponentene kommer til uttrykk når eleven sier dette. [K] for *konseptuell forståelse* er identifisert fordi eleven teller «En, to, tre, fire, fem ...» og kommer med et bekreftende utsagn etterpå «Jah! Det var det» (Figur 7). Eleven kan sies å med dette vise en forståelse for ideen femkant og hvordan den kan se ut. [B] for *beregning* kommer til uttrykk ved at eleven teller, trolig kantene til figuren

i Scratch, og med dette gir uttrykk for summen av antall kanter i figuren. [S] for *strategisk kompetanse* vises ved at eleven uttrykker «*En, to, tre, fire, fem [...]*». Dette kan forstås som at eleven arbeider med å løse problemet og gjør dette ved å telle høyt. [R] for *resonnering* blir uttrykt ved at eleven teller høyt og sier «*Jah! Det var det*» (Figur 7). Ytringen kan fortelle at eleven tenker logisk ved å telle at en femkant må ha fem kanter, og figuren eleven ser har også dette. At eleven uttrykker dette høyt overfor den andre eleven kan være et uttrykk for forklaring og begrunnelse overfor medeleven om hvorfor dette er en femkant. [E] for *engasjement* kommer til uttrykk i det eleven sier «*Jah!*» (Figur 7). Denne ytringen kan tolkes som en positiv opplevelse av mestring, og kan ha innvirkning på den positive innstillingen til faget.

På en slik måte er hver ytring gått igjennom og satt opp mot det nye rammeverket. Den enkelte komponent kodet i alle transkripsjonene før det er gått videre til å kode neste komponent. Tiltaket er gjort for å få en mest mulig lik tolkning av kodene gjennom forståelsen av rammeverket. Dette er for å tolke elevenes ytringer ut fra en mest mulig lik tolkning og forståelse av komponentene, ettersom både tolkning og forståelse har dynamiske menneskelige faktorer som påvirker kodingen av transkripsjonene.

3.5. ETISKE OVERVEIELSER

Etikk knyttes til de moralske spørsmål ved undersøkelsen og dens mål (Kvale & Brinkmann, 2009). Flere etiske hensyn er tatt i arbeid med studien. Etiske overveielser er gjort for å overholde informantenes anonymitet, i avgjørelser for valg av metode, forberedelser og gjennomføring av datainnsamling, utforming av oppgaver og behandling av data. Dette medfører også at reliabiliteten til oppgaven styrkes og blir ivaretatt. På bakgrunn av de medier som er brukt til datainnsamling, er prosjektet gjennom LATACME (forskningsprosjektet denne studien er en del av) meldt til Norsk Senter for Forskningsdata (NSD). Data som er brukt i oppgaven er anonymisert og vil ikke være personidentifiserbar. Det er også blitt behandlet og oppbevart slik at det kun har vært og vil være tilgjengelig for de som er gitt tillatelse til det.

3.5.1. FORSKING PÅ BARN

Forskningsetikk handler blant annet om beskyttelse (Tangen, 2010). Forskning som involverer barn har mye til felles med andre studier, men har også særlige hensyn som må tas stilling til for å ivareta barnet. Backe-Hansen & Frønes (2012, s. 17) påpeker at barn har rett på beskyttelse, men også utvikling. Dette fører igjen til kravet om barnets mulighet for å delta i forskning som vil kunne bidra til å støtte den faglige utviklingen. Forarbeidet som er gjort før datainnsamlingen har vært til hjelp for å bli bedre kjent med elevene og deres forutsetninger. Oppgavene som er brukt i Pilot og Opplegget er tilpasset etter beste evne til det ferdighetsnivået som er tenkt å kunne passe til elevenes forutsetninger, med utgangspunkt i alder og utvikling. Tilpassingen av oppgavene mellom Pilot og Opplegget og definering av lærerrollen under gjennomføringene er også gjort med tanke på elevenes forutsetninger, slik at elevene skulle oppleve mestring.

3.5.2. SAMTYKKE

For å opprettholde barnets beskyttelse er det en forutsetning i forskning at det blir gitt samtykke og tillatelse til deltakelse (Ruyter, 2003, s. 28). Et samtykkeskjema til informantenes foresatte ble utformet, ettersom dette er et krav når det skal forskes på barn (Backe-Hansen & Frønes, 2012, s. 17). Samtykkeskjemaets innhold ble i forkant godkjent av NSD og undertegnet før datainnsamlingen. I utgangspunktet er dette eneste samtykkeskjema som kreves fra informantene for å kunne gjennomføre datainnsamlingen. Likevel valgte vi å skrive et forenklet samtykkeskjema for elevene i en språkform som de kunne forstå. Dette var en etisk avveielse, og ble gjort for å vise elevene respekt og gi dem et «eierskap» til sin deltakelse i prosjektet. Samtykkeskjemaene er lagt inn som vedlegg (Vedlegg III og Vedlegg IV).

3.6. RELIABILITET OG VALIDITET

Forskningens *reliabilitet* har å gjøre med forskningsresultatens konsistens og troverdighet. Spørsmål knyttet til reliabilitet kan handle om hvorvidt studiens resultat på et annet tidspunkt kan reproduseres av andre (Kvale & Brinkmann, 2009, s. 250). Metoden for denne studien bygger på en kvalitativ hermeneutisk tilnærming. Forskningsprosessen og resultatene vil med

dette være personavhengig (Mellin-Olsen & Lindén, 1996, s. 21). Dette medfører at forskningens objektivitet og troverdighet vil avhenge av min forståelse og mine tolkninger som forsker i prosjektet, og hvorvidt andre vil godta disse.

En svakhet ved oppgaven er ulike begrensninger som påvirker hva som kommer frem av elevenes matematiske kompetanse. Først og fremst er det kun noen minutter av elevenes arbeid og forholdsvis korte dialogutsnitt som er brukt som datamateriale i denne oppgaven. I tillegg er det kun mulig å si noe om den kompetansen som blir uttrykt. At elevparene er gode nok samarbeidspartnere vil derfor også ha betydning for å fremme den andres kompetanse i det som uttrykkes. Hvordan dialogen mellom elevene fungerer og hvordan de utfyller hverandre, vil være av betydning. At de matematiske komponentene er så sammenflettet, kan også gjøre det utydelig å identifisere rett komponent, og jeg kan gjøre en feilaktig tolkning.

Transkribering av elevenes arbeid er et viktig element i denne studien i, og vil ha konsekvenser for analyse og resultat. Å transkribere betyr å transformere, å skifte fra den ene til den andre formen, noe som bringer med seg spørsmål tilknyttet reliabilitet. I transkripsjonene blir den fysiske samtalen mellom elevene abstrahert og fiksert til en skriftlig form. Situasjonen oversettes med dette mellom to narrative former – fra muntlig diskurs til skriftlig diskurs. I denne transformasjonen kreves det flere vurderinger og beslutninger underveis i konstruksjonene fra talespråk til skriftlig språk (Kvale & Brinkmann, 2009, s. 186-187). Utfoldelsestempo, stemmeleie, kroppsspråk og gester som umiddelbart fremtrer i samtalen mellom elevene, vil i stor grad gå tapt i oversettelsen til skriftlig form av situasjonen. I tillegg vil sammenhengen mellom elevenes samtale og det de gjør i Scratch kunne svekkes av å bli transkribert. Dette er noe som vil kunne gi utslag i analysen av datamaterialet, og vil få betydning for resultatet som fremkommer. For å sikre at færrest mulig detaljer går tapt i transformasjonen fra video til transkripsjon er det gjort ulike tiltak. Disse er blitt forklart i eget avsnitt i kapittel 3.4.3.2.

Et annet spørsmål som knyttes til studiens reliabilitet er om man ville fått et lignende resultat om undersøkelsen hadde vært gjort av en annen eller på en annen gruppe informanter. Siden studien er kvalitativ og består av et utvalg på åtte elever, vil forskningen ikke kunne generaliseres eller gis allmenn gyldighet. På den andre siden vil det at forskningen er kvalitativ medføre at det er gjennomført en omfattende og grundig analyse og tolkning av innhentet data.

I kvalitativ metode er det heller ikke hensikten at forskeren skal være utskiftbar, men må derimot være saklig og pålitelig (Widerberg, 2001, s. 18).

Validitet i forskning handler om sannhet, riktighet og styrke i det som uttales. I samfunnsvitenskap dreier validitet seg om hvor vidt oppgaven undersøker det den er ment å undersøke. En valid begrunnelse vil være et fornuftig, velfundert, berettiget, sterkt og overbevisende argument (Kvale & Brinkmann, 2009, s. 250-251). Å avgrense problemstillingen ved å undersøke den ved to mer konkrete forskningsspørsmål, vil styrke forskningens validitet. Til å underbygge og støtte validiteten i forskningens argument, er analysen forankret i relevant teori ut fra det forskningsspørsmålene spør om. Sammensettingen av to rammeverk for å utforme et nytt har vært et metodisk valg, og er blitt gjort for å styrke oppgavens validitet. Hvor vidt dette er oppnådd vil imidlertid avhenge av hvordan jeg har kombinert de to rammeverkene, og i hvilken grad jeg har formidlet min fortolkning av teorien på en hensiktsmessig og klar måte. Dette gjelder også validiteten knyttet til det teoretiske grunnlaget for å undersøke det andre forskningsspørsmålet.

4. RESULTAT OG ANALYSE

Problemstillingen i denne oppgaven går ut på å undersøke hvordan programmering kan fremme matematisk kompetanse hos elever. Til å undersøke problemstillingen er det valgt å analysere datamaterialet gjennom de to gitte forskningsspørsmålene;

1) *Hvordan kommer matematisk kompetanse til uttrykk i elevers samtaler når de programmerer?*

2) *Hvordan konstruerer elever matematisk kunnskap når de programmerer?*

Opptakene som er gjort av elevenes arbeid i dette prosjektet viser hvordan åtte elever på 7. trinn arbeider, når de i par skal programmere geometriske figurer i Scratch.

Analysen er delt inn i tre deler. Første del (4.2) presenterer samlede resultater av koder som er identifisert i de fire utvalgte situasjonene. Her arbeider hver av de fire gruppene som var med i prosjektet med å programmere en femkant (Vedlegg I). Fremstilling av resultatene fra kodingen skaper grunnlag for å kunne si noe om hvordan fem matematiske komponenter er uttrykt i de ulike gruppenes samtaler. Komponentene er utformet fra det nye sammensatte rammeverket, basert på rammeverket av Kilpatrick, Swafford og Findell (2001) med berikelse fra rammeverket av Niss og Højgaard Jensen (2002). Disse komponentene er *konseptuell forståelse [K]*, *beregning [B]*, *strategisk kompetanse [S]* og *resonnering [R]* og *engasjement [E]*. I andre del (4.3) er enkeltutdrag fra transkripsjonene presentert og analysert, med utgangspunkt i de matematiske komponentene. Analyseverktøyet som er benyttet her er utformet på betingelsen å kunne identifisere matematisk kompetanse uttrykt i elevenes samtaler. Gjennom de to første delene er det første forskningsspørsmålet blitt undersøkt. Siste og tredje del av analysen (4.4) har en hermeneutisk tilnærming til samtaleutdrag fra en av gruppene. Analysen her er gjort med utgangspunkt i Seymour Papert (1991) og konstruksjonismens syn på læring, for å forstå elevenes handlinger i Scratch ut fra sine egne perspektiv og slik jeg opplever dem. Gjennom analyse av elevsamtaler med bakgrunn i et slikt læringsperspektiv, er det undersøkt hvordan elever konstruerer matematisk kunnskap når de programmerer, som er det andre forskningsspørsmålet.

4.2. RESULTATER FRA KODING AV ELEVENES SAMTALER

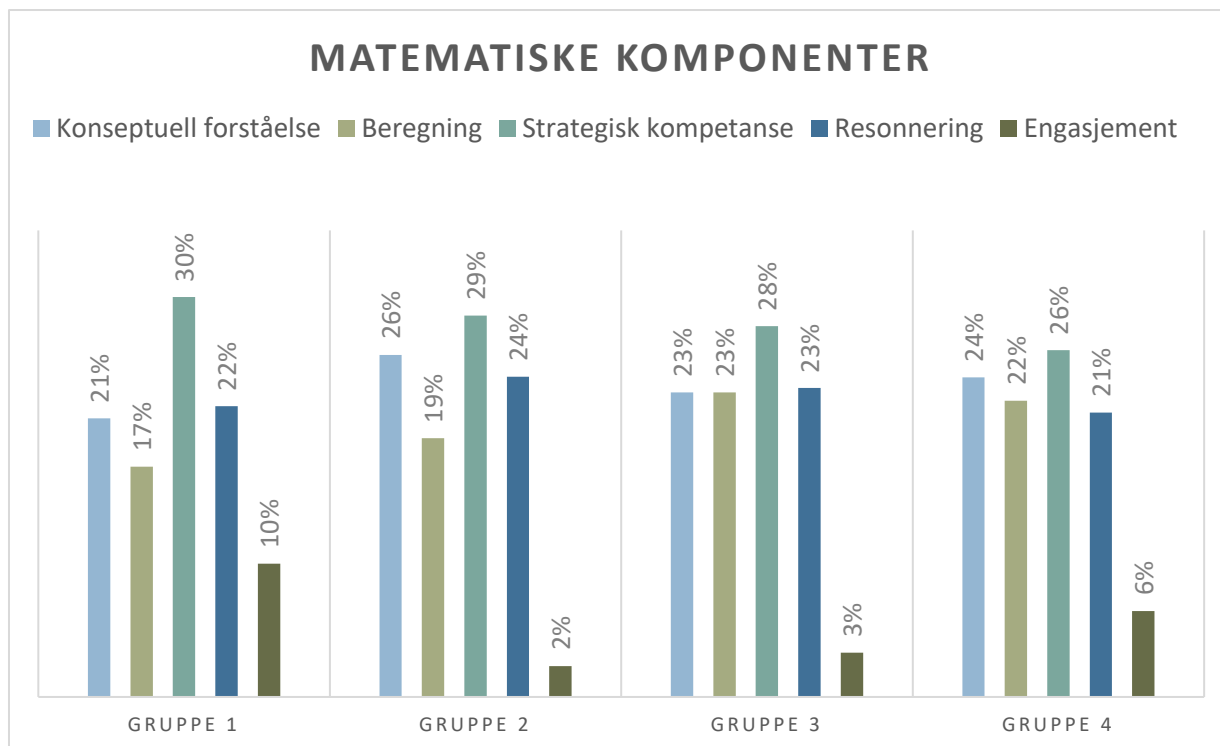
Resultatene fra analysen av de fire gruppernes arbeid med programmering av femkant i Scratch er presentert i tabellen under. Her vises frekvensen av kodene som er identifisert i de utvalgte situasjonene fra datamaterialet. I tillegg er det laget en kolonne som viser prosentvis forekomst av hver kode, for enklere å kunne få en oversikt over hvor mye hver av komponentene forekommer.

| Matematisk komponent | Gr. 1 | Prosent Gr.1 | Gr. 2 | Prosent Gr.2 | Gr. 3 | Prosent Gr.3 | Gr. 4 | Prosent Gr.4 |
|------------------------|-------|--------------|-------|--------------|-------|--------------|-------|--------------|
| Konseptuell forståelse | 23 | 21 % | 78 | 26 % | 69 | 23 % | 82 | 24 % |
| Beregning | 19 | 17 % | 59 | 19 % | 69 | 23 % | 76 | 22 % |
| Strategisk kompetanse | 33 | 30 % | 87 | 29 % | 84 | 28 % | 89 | 26 % |
| Resonnering | 24 | 22 % | 72 | 24 % | 70 | 23 % | 73 | 21 % |
| Engasjement | 11 | 10 % | 7 | 2 % | 10 | 3 % | 22 | 6 % |

FIGUR 8: TABELL OVER KODER IDENTIFISERT I ELEVSAMTALER

4.2.1. RESULTATER PRESENTERT I PROSENT

For å kunne sammenligne gruppene visuelt er resultatene i Figur 8 også representert i Figur 9 som søylediagram. Denne fremstillingen visualiserer forholdet mellom de ulike matematiske komponentene for hver av gruppene. Her kan man se at de fire komponentene *konseptuell forståelse [K]*, *beregning [B]*, *strategisk kompetanse [S]* og *resonnering [R]* er mest fremtredende, men særlig *strategisk kompetanse*. Komponentene *engasjement [E]* er i alle grupper minst representert blant kodene som er satt i analyse av datamaterialet.



FIGUR 9: SØYLEDIAGRAMMER SOM VISER KODING AV MATEMATISKE KOMPONENTER I PROSENT

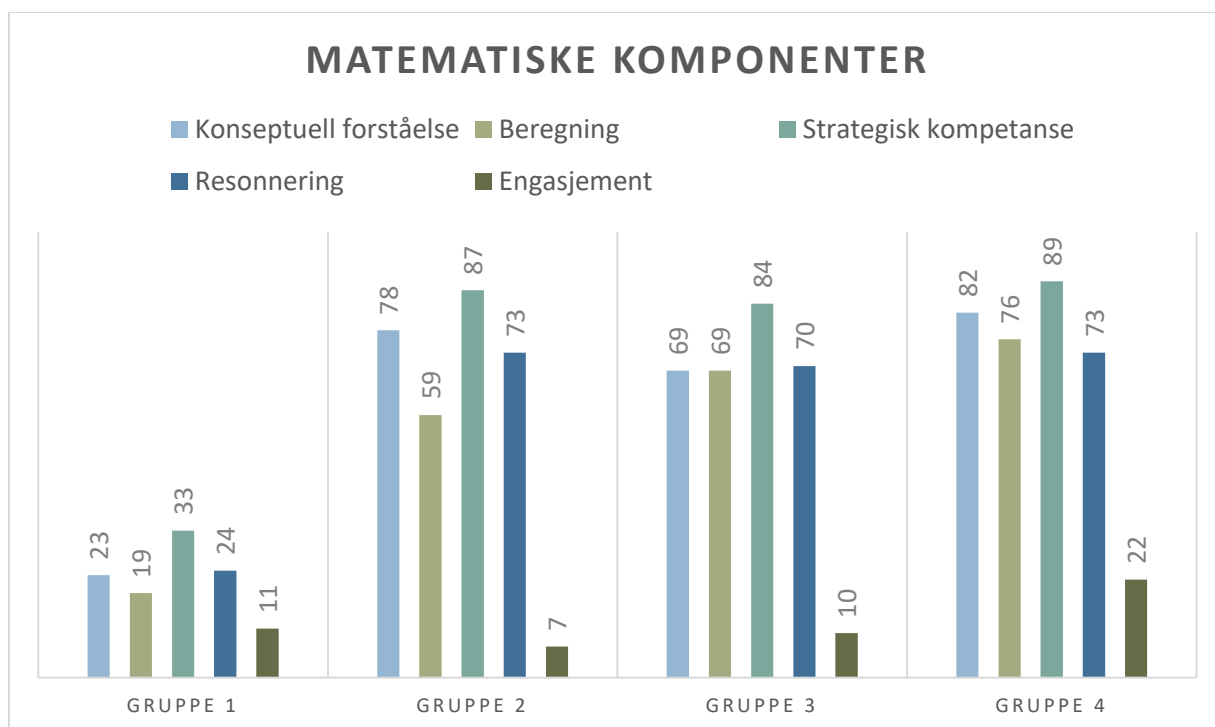
Alle gruppene har størst andel *strategisk kompetanse* og minst andel *engasjement*. Av de resterende tre komponentene ligger resultatene på rundt 20% for *konseptuell forståelse* og *resonnering*, men litt mer varierende prosentandel for komponenten *beregning*. Komponentene *engasjement* skiller seg også ut når gruppene sammenlignes. Det kan være flere faktorer som gjør at denne modellen viser dette, noe jeg vil komme tilbake til litt senere. Ser vi på hver av gruppene individuelt, er det mer variasjon mellom hver av komponentene. Særlig gruppe 1, men også gruppe 2 har større variasjon mellom hver av komponentene, mens gruppe 3 og 4 har jevnere prosentfordeling. Om resultatene fra hver av gruppe sammenlignes kan man se at forekomsten av komponentene i hver gruppe har flere likheter.

Tiden gruppene brukte på å gjennomføre oppgaven med å programmere en femkant i Scratch var gruppe 1: 7:14 minutter, gruppe 2: 11:07 minutter, gruppe 3: 10:15 minutter og gruppe 4: 13:37 minutter. Dette er også en faktor å ta hensyn til for å kunne sammenligne de ulike gruppene med hverandre. Gruppe 1, som har brukt kortest tid på å løse oppgaven, har gitt meg som forsker en kortere dialog å ta utgangspunkt i for å identifisere matematiske komponenter. I motsetning har gruppe 4, som brukte rett i underkant av dobbelt så lang tid som gruppe 1, en lengre dialog med flere ytringer å identifisere matematiske komponenter. Dette er en vesentlig

faktor med hensyn til analysering av prosentvis representasjon av de samlede resultatene. Figur 9 kan derfor brukes til å gi en foreløpig indikasjon på hvilke komponenter som er mest og minst fremtredende i elevenes samtaler.

4.2.2. RESULTATER PRESENTERT I ANTALL KODER

For å bedre få innsikt i hva som ligger i disse resultatene har jeg tatt med en tabell som viser antall koder i tillegg til modellen som viser prosentfordeling (Figur 9). Dette er fordi modellen med prosentfremstillingen av resultatene gjør at sammenligning mellom gruppene kan feiltolkes ut fra hvordan denne modellen fremstiller resultatene fra kodingen. Modellen under viser derfor antall koder for hver av de fem matematiske komponentene som er identifisert og registrert i de fire utvalgte situasjonene.



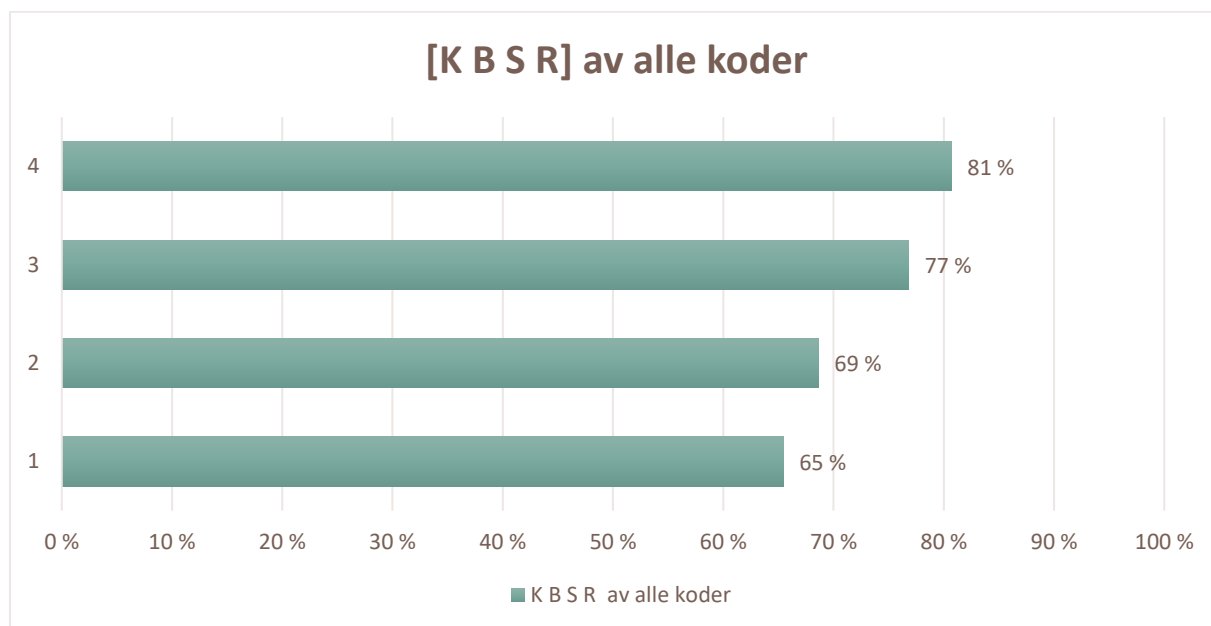
FIGUR 10: SØYLEDIAGRAM SOM VISER KODING AV MATEMATISKE KOMPONENTER I ANTALL

Ser man resultatene representert i antall koder som er registrert, viser resultatene andre sider ved kodingen av elevenes samtaler. Særlig blir det tydelig at komponenten *engasjement* gir et annet bilde i Figur 10 sammenlignet med Figur 9. Her kommer det frem at det er gruppe 4 som flest ganger har blitt registrert med koden [E] (for *engasjement*), mens gruppe 1, 2 og 3 har

rundt halvparten. Vi kan også se at gruppe 1 som har den korteste dialogen skiller seg ut med vesentlig færre antall koder av hver av komponentene. Totalt har gruppe 4 flest koder registrert, med gruppe 2 og 3 like bak.

4.2.3. FELLESKODEN [K B S R]

I Figur 9 og Figur 10 kan man se at de fire første komponentene gjentas mange ganger, flere til sammen enn det er ytringer i samtalen. Dette er fordi disse fire komponentene [K], [B], [S] og [R] er registrert samtidig i flere ytringer. Det er derfor også gjort en opptelling av hvor mange ganger de fire kodene samlet er blitt registrert i en ytring for hver av gruppene. Felleskoden som er brukt for å omtale dette er [K B S R] for *konseptuell forståelse, beregning, strategisk kompetanse og resonnering*. Komponenten [E] for *engasjement* er ikke tatt med i denne felleskoden fordi denne skiller seg ut fra de andre fire komponentene. Ytringer der koden [E] er sammen med kodene [K B S R] er telt med, fordi en ytring som har registrert kodene [K B S R E] inneholder denne felleskoden [K B S R].



FIGUR 11: RESULTATER AV KODEN [K B S R] I ELEVSAMTALENE

I Figur 11 viser y-aksen gruppenummer, mens x-aksen viser prosentvis frekvens for hvor ofte kodene K (*konseptuell forståelse*), B (*beregning*), S (*strategisk kompetanse*) og R (*resonnering*) var blitt kodet i samme ytring. Resultatene i Figur 11 viser at elevsamtalene fra alle gruppene

har mellom 65% og 81% av kodene hvor [K B S R] er registrert i samme ytring, sammenlignet med summen av alle kodene som er registrert. De resterende prosentene er dermed ytringer som består av enten en, to eller tre av kodene for de fem komponentene. Av denne modellen kan man se at de matematiske komponentene [K], [B], [S] og [R] sjeldent opptrer alene i en ytring. Hadde modellen vist hvor ofte en kode av en komponent hadde opptrådt alene, ville den resterende andelen vært enda mindre.

4.2.4. DE FEM MATEMATISKE KOMPONENTENE

Som søylediagrammene i Figur 9 og Figur 10 over viser, opptrer de fem matematiske komponentene i alle de fire gruppens arbeid med femkanten, men i ulik grad. Komponenten *strategisk kompetanse* utpeker seg som den mest fremtredende, tett etterfulgt av *konseptuell forståelse*, *resonnering* og *beregning*. Diagrammene viser også at det er liten kontrast mellom synligheten av disse fire komponentene i elevenes samtaler. Som teorien i rammeverket understreker, er det ikke tydelige skillelinjer mellom disse komponentene (Kilpatrick et al., 2001; Niss & Højgaard Jensen, 2002).

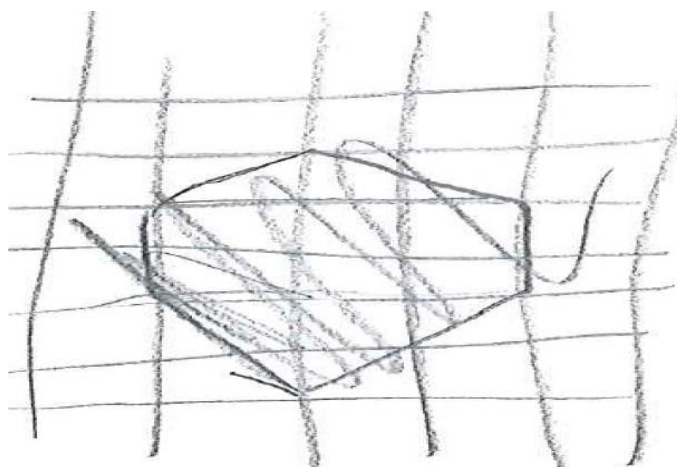
4.3. MATEMATISK KOMPETANSE UTTRYKT I ELEVSAMTALER

For å se mer konkret på hvordan de ulike matematiske komponentene er identifisert og kommer til uttrykk i elevenes samtaler, har jeg tatt for meg noen utdrag fra transkripsjonene. Et kriterium for utvelgelse av enkeltutdragene er at flere komponenter er i spill samtidig. På denne måten vil det være enklere å se samspillet og forholdet mellom komponentene. Det er også valgt situasjoner med utgangspunkt i at det kan skape et rikt grunnlag for diskusjon i henhold til det første forskningsspørsmålet – *Hvordan kommer matematisk kompetanse til uttrykk i elevers samtaler når de programmerer?*

4.3.1. GRUPPE 4

«Aa? ... Ja, men det blir jo ikke en femkant?»

Første utdrag er hentet fra gruppe 4 (Vedlegg IX) som består av to jenter, her kalt Hedda og Vilde. Elevene er ved utdragets begynnelse 16 minutter inne i arbeidsøkten, og har til nå arbeidet i litt over 7 minutter med å programmere en femkant. I forkant har de diskutert hvordan de kan lage en femkant og forsøkt å tegne den på et kladdeark de har fått tilgjengelig (Figur 12).



FIGUR 12: UTKLIPP FRA TEGNEARK - GRUPPE 4

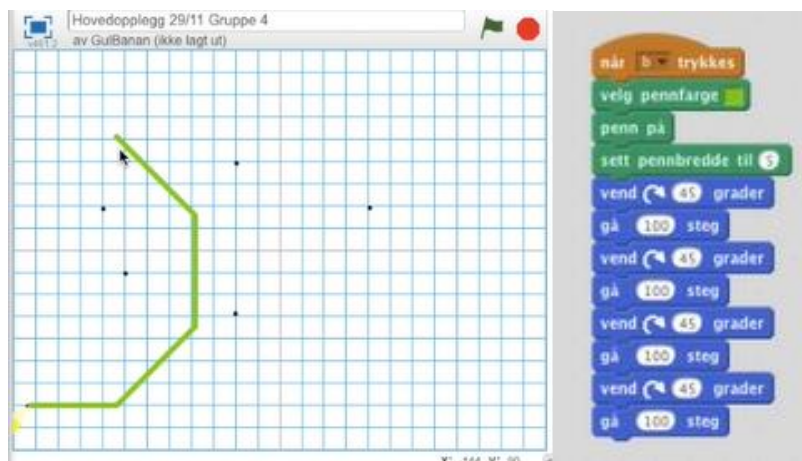
Figur 12 viser et utklipp fra elevenes forsøk på å tegne en femkant, uten helt å lykkes. Videre har de fått hjelp til å sette inn prikker i bakgrunnen til programmeringsvinduet. Dette er for å kunne bruke prikkene som hjelp til å se hvor pennfiguren skal flytte seg i Scratch for å få en femkant. De har snakket om koordinater i programmeringsvinduet for å få pennen til å flytte seg mellom punktene de har satt. Etter å ha forsøkt å løse oppgaven på denne måten endrer de strategi da Vilde uttalt at *«Men det blir jo veldig stress å finne sånne koordinater. Vi bare tegner den»* (Vedlegg IX-a). Hedda og Vilde virker fortsatt usikker på hvordan femkanten skal se ut, da de spør en av studentene om hjelp til dette. De får da fortalt at så lenge den har fem kanter og fem hjørner, så er det bra nok (Vedlegg IX-a). De begynner etter dette å snakke om hvor mange grader den (pennfiguren) skal snu seg. Dette er de ikke sikker på, og de blir enig om å bare prøve seg frem og se hva som skjer. De går med dette bort fra punktene og koordinatene de først snakket om. De bruker klossene [vend # grader] og [gå # steg] i tillegg til kommandotasten og pennfunksjonsklossene (de fire øverste klossene i skriptet). Elevene har

testet skriptet flere ganger underveis for å se hva som skjer, og har gjort justeringer etter hvert. Hedda og Vilde fortsetter:

(tid 16:00)

- 1) *Vilde: Okei, hvis vi tar slett*
- 2) *Hedda: og så hvis vi tar den rett opp nå, og så til siden. Da har vi laget et hus. Da kan vi krysse av huset*
- 3) *[begge ler]*
- 4) *Vilde: nei ... [litt lavt]*
- 5) *[Vilde klør seg i hodet]*
- 6) *Vilde: Okei, vent litt. Hvis vi tar pekeren, og så «b»*

---(tester skript)---



FIGUR 13: UTKLIPP FRA SCRATCH - GRUPPE 4

- 7) *Vilde: aa..? Ja, men det blir jo ikke en femkant?*
- 8) *Hedda: En, to, tre, fire, fem [peker på skjermen mens hun teller]*
- 9) *Vilde: Da må den rett opp der da [peker på skjermen]*
- 10) *Hedda: Det blir en sekskant ...*
- 11) *Vilde: En, to, tre, fire ... hvis vi får den der til å gå opp der. Hvordan gjør vi det? [peker med musepeker på skjerm]*

I dette utdraget fra elevene i gruppe 4 er alle de fem komponentene identifisert gjentatte ganger ved kodingen. Den **konseptuelle forståelsen** kommer frem i elevenes samtale der de snakker sammen om hvordan femkanten skal se ut. Hedda foreslår å gjøre noen endringer så de også får

laget et hus (2), en figur oppgaven sier de skal programmere senere (Vedlegg II – Oppgaver til Opplegget). Hedda viser med dette at hun ser sammenheng rundt konsepter ved å forklare at en femkant også kan være et hus. Vilde viser også *konseptuell forståelse* ved å si «nei» (4), fordi hun tar i betraktning og tolker det Hedda sier og argumenterer kort imot med et nei. Dette viser at Vilde trolig har en annen oppfatning enn Hedda av sammenhengen mellom det matematiske konseptet femkant og hus. Det blir likevel ikke noen videre argument rundt denne lille konflikten i *konseptuell forståelse*. Jentene fortsetter så med å teste skriptet og får fire sammenhengende kanter med 45 graders vinkel mellom hver (Figur 13). De fortsetter å uttrykke sin *konseptuelle forståelse* ved å se etter sammenhenger, stille spørsmål og svar om operasjoner og prosedyrer og å oppdage og formulere forklaringer rundt representasjonen til den matematiske situasjonen. Hedda og Vilde ser i denne situasjonen ut til å bruke både klosser i skript, og hvordan disse fungerer sammen, til å se hvilken representasjon denne kombinasjonen gir. Hedda teller i ytring 8 helt opp til fem, selv om det kun er fire kanter i figuren som vises i Scratch. Vilde følger opp denne forklaringen ved å svare at «*da må den rett opp der da*» (9). Denne ytringen kan tyde på at hun forstår deler av den *konseptuelle forståelsen* Hedda har, og kommer derfor med et løsningsforslag. I denne situasjonen ser Hedda og Vilde ut til å være enig om flere deler av femkanten som konsept. De er begge innforstått med at vinklene og kantene i femkanten kan være ulike, og at disse må henge sammen og utgjøre fem hjørner.

Komponenten *beregning* er også tilstede i Hedda og Vildes samtale. For at Hedda skal kunne forklare hvordan figuren kan bli til et hus, har hun sannsynligvis gjort en hoderegning på at dette er mulig. Hun uttrykker kunnskap om en matematisk operasjon, noe som krever å tolke og sette seg inn i den matematiske «teksten» som vises. Teksten her er det som vises i Scratch av skript og visuelle fremstillinger. Noe av det samme gjør også Vilde i det hun sier at det ikke blir en femkant (7). Pennfiguren i Scratch fortsetter ikke å tegne, men likevel snakker hun om at det *blir* (7). Dette tyder på at hun har kunnskap om prosedyren og den matematiske operasjonen som vil skje om skriptet hadde fortsatt det repeterende mønsteret, nemlig at det ikke hadde blitt en femkant. Hedda snakker også om at det *blir* da hun sier «*det blir en sekskant*» (10). Dette kan vise at hun har tenkt som Vilde, bortsett fra at denne beregningen ville ikke blitt rett om det repeterende mønsteret hadde fortsatt.

Kjennetegn ved komponenten *strategisk kompetanse* er blant annet å representere, formulere og løse matematiske problem. Videre kjennetegnes denne komponenten ved å kunne kjenne til en variasjon av ulike løsningsmetoder, og å ha kunnskap om hvilke strategier som kan være nyttig for å løse et spesifikt problem. Jentene viser *strategisk kompetanse* i det de setter sammen skriptet og arbeider med å representere og løse det matematiske problemet – å programmere en femkant. Gjennom samtalen kommer begge jentene med konkrete forslag til hvordan de kan løse dette spesifikke problemet. Hedda viser også at hun kjenner til en variasjon av løsningsmetoder i det hun trekker inn sammenhengen mellom hus og femkant. Hvert utsagn i samtaleutdraget kan sees på som et ledd i å løse det gitte matematiske problemet, og kan dermed sies å på ulike måter være deler av å uttrykke den *strategiske kompetansen*.

Komponenten *resonnering* kommer til uttrykk gjennom ytringene i samtaleutdraget. Hedda og Vilde ser begge ut til å vise evnen til å tenke logisk rundt konseptet femkant, og hvordan denne skal programmeres. Dette gjør de ved å stille spørsmål knyttet til den matematiske situasjonen og gi forklaringer og begrunnelser på sine resonnement. Elevene knytter resonnementene sine til det som skjer i Scratch sammen med det oppgaven spør om. De snakker konkret om «pekeren» (6) (pennfiguren i Scratch), peker på skjermen, teller og kommuniserer med hverandre om hvordan de skal løse oppgaven. At elevene skal løse oppgaven sammen utløser et behov for å forklare hva de tenker overfor den andre, med andre ord å uttrykke *resonnering*. Dette kan man se gjennom samtalen elevene fører.

Den siste komponenten *engasjement* kan identifiseres her om man tar med det som er satt i klammer i transkripsjonen (gester og annet uttrykt). Elevenes latter er her tolket som en indisia på at elevene har en positiv opplevelse av arbeidet. Å identifisere *engasjement* i den enkelte ytring kan være komplekst. Det er fordi denne komponenten beskriver en indre drivkraft, en innstilling og en tro på egne evner som elevene drives av. For eksempel er kjennetegn på at elevene er interessert i å finne en løsning ikke eksplisitt uttrykt i rammeverket som en beskrivelse av komponenten *engasjement*. Rammeverket påpeker likevel at *engasjement* utvikles ved at de andre komponentene styrkes. Dette gjør det nødvendigvis ikke enklere å identifisere denne komponenten. Et spørsmål som «*hvordan gjør vi det?*» (11) kan også tolkes som at eleven er motivert for å løse oppgaven, men er likevel ikke kodet som *engasjement* i

denne oppgaven. *Engasjement* vil derfor kunne tolkes å være mer til stede enn det er kodet, på bakgrunn av at de andre komponentene stiller sterkt.

I utdraget under fortsetter jentene fra samtalen over. De arbeider nå videre med hvordan de kan endre skriptet til å få programmert en femkant:

12) Hedda: *Hvordan fikk vi den til å gå skrå igjen?*

13) Vilde: *diagonalt ...? [snakker lavt]*

(V flytter blyantfigur og trykker [slett] i Scratch)

14) Vilde: *Eee ... nå må den snu seg ... [viser med armene] mer enn 45 grader ...*

15) Hedda: *da må den ... 90?*

16) Vilde: *hvordan ...*

---(Vilde trykker «b» - tester skript)---

(får samme figur som sist - Figur 13, fordi ingen endringer er gjort i skriptet)

17) Vilde: *Hvordan går den opp der? [peker på skjermen] Den må snu seg fff... [sier lavt] det går ikke an å finne det ut ...*

---(liten pause med stillhet)---

18) Vilde: *Mmm ... hvis vi prøver en til da. Vend ... (setter inn kloss)*

19) Vilde: *Litt ... det er litt mindre enn 90 ...*

---(liten pause med stillhet)---

20) Vilde: *hvis vi ... [lavt]*

21) Hedda: *85? 80? 85?*

22) Vilde: *70 ... (skriver inn i blokk) Nå bare tipper jeg*

I dette samtaleutdraget kan man se nærmere på hvordan de fire komponentene **konseptuell forståelse**, **beregning**, **strategisk kompetanse** og **resonnering** bygger på hverandre. Felleskoden [K B S R] kan sies å være fremtredende i dette utdraget. Jentene snakker her om

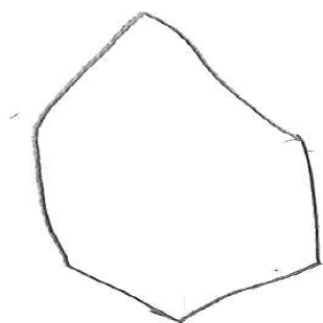
grader, snu seg, vende, diagonalt, gå skrå, mer enn og litt mindre. Dette er forklarende og beskrivende betegnelser for hvilken operasjon som skal gjøres, og hvordan den kan utføres. Begge viser at de ser sammenhenger rundt konseptet femkant, hva som skjer i programmeringsvinduet og det som står i skriptet. Jentene har allerede prøvd med 45 grader gjentagende i skriptet, og har trolig sett at det må gjøres en endring i hvor mange grader pennfiguren skal endre retning. De foreslår begge tall som er større enn 45, noe som kan vise at de har sett hvordan tallene og rekkefølgen i skriptet henger sammen. Beregningsferdigheter er nødvendig for å se en slik sammenheng, og å finne ut av hvilke tall som kan prøves ut. Verken Hedda eller Vilde skriver eller tegner her på ark ved siden av. De ser dermed ut til å gjøre disse beregningene i hodet og gjennom å uttrykke sine resonnement overfor hverandre. At jentene resonnerer høyt gjør at den *strategiske kompetansen* kommer til uttrykk. Dette gjør at de kan ta i bruk hverandres resonnement til å finne en videre strategi for å løse oppgaven. En strategi som elevene ser ut til å benytte seg av er å prøve ut og som Vilde sier «*nå bare tipper jeg*» (22). De tester også ut skriptet flere ganger underveis ettersom det gjøres endringer. Denne strategien kan se ut til å hjelpe dem til å tenke logisk, se sammenhenger mellom konsept og prosedyrer, og hjelpe dem på veien videre mot en mulig løsning.

4.3.2. GRUPPE 2

«Ok, da prøver vi ...»

Elevene i gruppe 2 består av en jente og en gutt, her gitt de fiktive navnene Hedvig og Viktor (Vedlegg VII). På tidspunktet samtaleutdraget under begynner er elevene omtrent 12 minutter inne i den totale arbeidsøkten. Til nå har de arbeidet et drøyt halvminutt med å programmere en femkant.

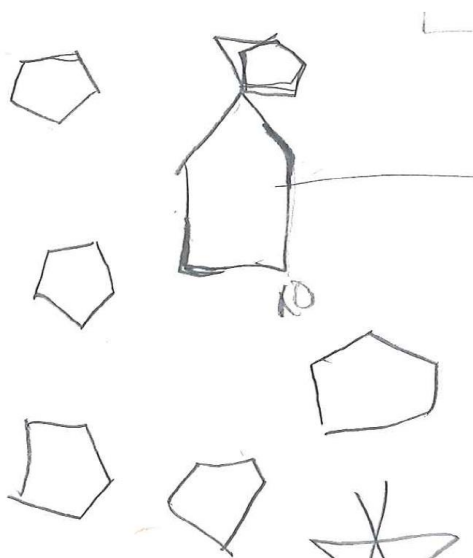
---(Viktor har fra starten begynt å tegne på et ark)---



FIGUR 14: UTKLIPP FRA TEGNEARK - GRUPPE 2

- 1) Viktor: Er det femkant?
- 2) Hedvig: ja
- 3) Viktor: den
- 4) Hedvig: Nei, det er ... vent litt ... få se
- 5) Viktor: Jo, det er en, to, tre, fire, fem
- 6) Hedvig: En, to, tre, fire, fem, seks. Det er en sekskant
- 7) Hedvig: Nei, var det en femkant vi skulle lage? ... Men kanskje det er en annen måte å tegne det på

---(Viktor fortsetter å tegne på arket)---



FIGUR 15: UTKLIPP FRA TEGNEARK - GRUPPE 2

- 8) Viktor: sånn!
- 9) Hedvig: Men liksom ...
- 10) Viktor: det går jo an

- 11) Hedvig: *[puster ut] Det går jo ikke*
- 12) Viktor: *Jo, se ... den*
- 13) Hedvig: *Ja, men da blir det jo et hus ... [lavi] en femkant*
- 14) Viktor: *Nei ... Ja, men sånn [Peker på arket]*
- 15) Hedvig: *Ja, men det er jo et hus*
- 16) Viktor: *Nei, det er ikke et hus. Ser det ut som et hus?*
- 17) Hedvig: *Liksom ... Femkant og hus er jo det samme. [henvender seg til Viktor]*
- 18) Viktor: *Ja, men da lager vi bare et hus ... og en femkant*
- 19) Hedvig: *to like*
- 20) Viktor: *så gjør vi sånn, sånn, sånn, sånn, sånn*
- 21) Hedvig: *Men det blir jo ikke rett. Da blir det jo ... Alle vinklene må jo være like. Må ikke de?*
- 22) Viktor: *Nei ... De må jo ikke det ... Må de?*
- 23) Hedvig: *Må ikke de?*
- 24) Viktor: *Må alle vinkler være like?*
- 25) Student: *Hva er en femkant?*
- 26) Viktor: *En ting med fem kanter.*
- 27) Student: *Er det noe mer som er viktig? Eller er det bare det?*
- 28) Viktor: *Kanskje at alle vinklene er like? ... Eller?*
- 29) Hedvig: *Jeg vet ikke*

Samtaleutdraget over fra Hedvig og Viktor varer i omtrent 1 ½ minutt. På den tiden er kodene for *konseptuelle forståelse*, *beregning*, *strategisk kompetanse* og *resonnering* gjentakende i analysen. Koden for *engasjement* er ikke registrert i dette utdraget, som har med min forståelse av det teoretiske rammeverkets krav for identifisering av denne komponenten. Dette kan selvsagt diskuteres, noe jeg vil komme tilbake til om denne komponenten senere i diskusjonskapittelet (se kapittel 5.1.1). Det er derfor her rettet fokus mot de fire andre komponentene som er identifisert.

Den *konseptuelle forståelsen* kommer tydelig til uttrykk ettersom samtalen dreier seg rundt hvordan en femkant kan se ut. Elevene viser at de ser sammenhenger mellom geometriske figurer som sekskant, femkant og hus. Dette kan man se ved at kantene til Viktor sin første tegning av en figur telles, og de finner ut at denne har seks kanter og dermed er en sekskant.

Det gjøres så justeringer ved å ta bort en kant slik at det kun er fem kanter på de neste figurene Viktor tegner. Hedvig oppdager og uttrykker da at dette kanskje også er et hus. Setningen Hedvig kommer med at «*femkant og hus er jo det samme*» (17) kan tolkes som et uttrykk for hennes *konseptuelle forståelse*. Figuren hun sier ligner et hus har også fem kanter, en sammenheng som her blir oppdaget og uttrykt. Hun går imidlertid ikke videre med denne ideen i samtalen med Viktor.

Hvilke krav som må oppfylles for at en figur kan sies å være et hus er ikke fortalt på forhånd til disse elevene, samme gjelder femkanten. Det er derfor opp til Hedvig og Viktor seg imellom å bli enige og finne ut av hva som kreves for å tegne figuren. Dette gjør at det oppstår behov for å kommunisere og uttrykke sin *konseptuelle forståelse* overfor hverandre.

I samtalen viser elevene kunnskap om noen kriterier til femkanten. I denne sammenheng trekker de også inn de to geometriske figurene sekskant og hus. Elevenes kompetanse i *beregning* uttrykkes både muntlig og skriftlig i form av tegning. Denne komponenten av matematisk kompetanse går ut på mer enn å behandle og anvende symbol- og formelspråk. Det handler også om å ha kunnskap om når og hvor de er hensiktsmessig, og ferdigheter til å uttrykke dem fleksibelt, nøyaktig og effektivt. Viktor viser ved sine tegninger en slik kunnskap og skaper med dette også et utgangspunkt for å kommunisere sine tanker om *beregning* til Hedvig. Hun tolker så disse tegningene ut fra sin forståelse og uttrykker med dette sin kompetanse til *beregning* muntlig. Ved å kunne delta i samtalen med Viktor kan det vise at Hedvig kan sette seg inn i og tolke Viktor sine matematiske visuelle illustrasjoner av en femkant. Hun gjør seg bruk av Viktor sine resonnement og tegninger for å kunne uttrykke sine ferdigheter effektivt. Viktor viser også at han kan sette seg inn i og tolke Hedvig sine resonnement, i det han argumenterer for sin forståelse og kunnskap av *beregning* av en femkant.

I det både Viktor og Hedvig uttrykker *konseptuelle forståelse* og *beregning*, representerer, formulerer og løser de matematiske problem. Det matematiske problemet her er først og fremst å finne ut av hvordan en femkant kan se ut før de kan gå videre til å programmere den. Dette kjennetegner den *strategiske kompetansen* og gjør seg med dette gjeldende i elevenes samtale. Viktor sine tegninger av femkanter er med på å formulere og representere det matematiske problemet, og er et virkemiddel på veien mot å sammen komme til en løsning. Dette er ett av flere ledd i fremgangsmåten, ettersom elevene videre arbeider med å overføre tankene de kommer med her til å programmere femkanten i Scratch. Problemet formuleres blant annet

gjennom samtalen ved de uttrykte resonnementene som «*Er det femkant?*» (1), «*Nei, var det en femkant vi skulle lage? ... Men kanskje det er en annen måte å tegne det på*» (7), «*Det går jo ikke*» (11) og «*Ja, men det blir jo et hus ... [lavt] en femkant*» (13). Samtalen mellom Hedvig og Viktor kan sies å være med å uttrykke kunnskapen de har om hvilke strategier som vil kunne være nyttig for å løse dette problemet. Videre programmeringsarbeid med femkanten er avgjørende for hva de blir enig om. Enten det er at hus og femkant er det samme, eller om de bestemmer seg for å sette alle vinkler like.

Resonnering inkluderer både forklaringer og begrunnelser, både uformelle, intuitive og induktive. Den viser kapasiteten man har til å tenke logisk om forhold rundt konsepter og situasjoner. En slik form for resonnering kan sies å være ett av hovedredskapene elevene her tar seg bruk av i arbeidet med femkanten. *Resonnering* kommer til uttrykk gjennom samarbeidet til Hedvig og Viktor når det skjer en form for konflikt i møte med representasjoner, den andres resonnement, oppgaven og de tilgjengelige hjelpemidlene. Med konflikt menes det at elevene ser ut til å ikke vite akkurat hvordan de skal gå frem, og tenker ulikt om forskjellige forhold. Her oppstår det et behov for resonnere seg frem til neste steg i løsningsprosessen i samhandling med den andre, enten det er å utforme en forklaring eller begrunnelse til medeleven, tolke hva oppgaven spør etter, ta i bruk representasjoner eller utforme en representasjon. Dette styres av elevenes kapasitet til å tenke logisk om forhold rundt konsepter og den situasjonen de befinner seg i gjennom samarbeidet.

Allerede fra starten av samtalen i arbeid med femkanten snakker Hedvig om at de må ha «*gå et eller annet steg*» og «*så må vi vende et eller annet*» (Vedlegg VII-a). I utdraget over diskuteres det hva som er kriteriene til en femkant. Spørsmål som kommer frem er «*Må alle vinkler være like?*» (24) og «*Hva er en femkant?*» (25). Hedvig og Viktor er enige om figuren må ha fem kanter, men virker ikke til å få oppklart om det er et krav til figuren at alle vinkler må være like.

I den fortsettende samtalen etter utdraget over ender de opp med å gjøre alle vinklene like «*fordi da kan det se ut som det samme fra alle sider*» (Vedlegg VII-b). Fra dette går de videre til å snakke om «*hvor mange grader er en femkant?*» (Vedlegg VII-b). De begynner å snakke om 90 grader, og kommer frem til at det må bli mer enn det. De prøver med 120 grader og får en vinkel i Scratch med denne vinkelen. Så blir 160 foreslått av Hedvig, men rett etterpå sier hun selv at «*nei, da går den jo bare innover. Da må vi ta mindre da ...*» (Vedlegg VII-b). Et nytt

forslag fra kommer fra Viktor om å prøve 60 grader. Dette er i korte trekk beskrivelsen av det som har skjedd før samtaleutdraget under fortsetter:

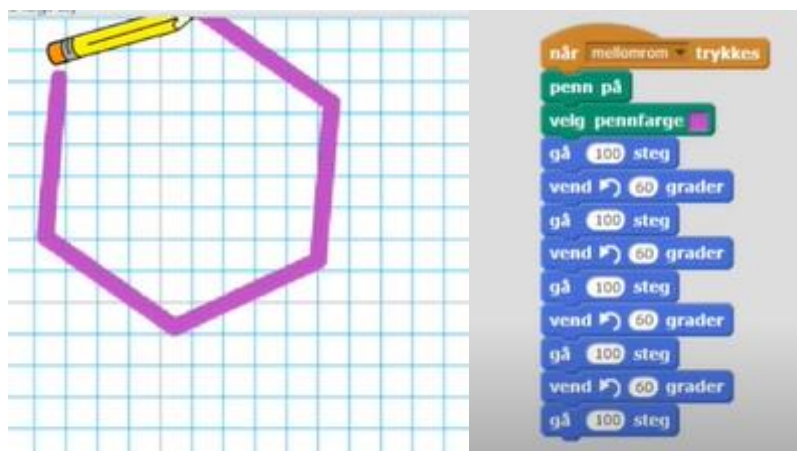
30) Viktor: Var det 60 grader?

31) Hedvig: Vi prøver ...

--- (skriver inn i klosser i blokk) ---

32) Hedvig: Ok, da prøver vi ...

-- (tester skript) --



FIGUR 16: UTKLIPP FRA SCRATCH - GRUPPE 2

33) Hedvig: Ja?

34) Viktor: Vi mangler én ...se?

35) Hedvig: Ja, jeg skjønnte ikke helt ...

36) Viktor: Nei ...?

37) Hedvig: Men trykk på grønt flagg [peker på skjermen]

--(tester igjen - får samme figur som over)---

38) Hedvig: Ta Gå 100 steg

39) Viktor: Nei, da blir det en sekskant

40) Hedvig: Åja ... nei, men da må vi ... da kan vi ikke ha 60 grader. Da må vi ha mindre.
Nei ... mer

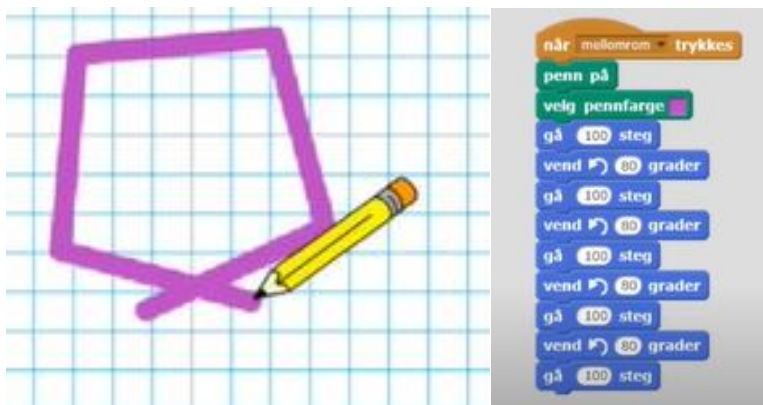
41) Viktor: Mhm

42) Hedvig: Det var jo derfor han ... 70 grader kanskje? Eller 80? Skal vi prøve 80?

43) Viktor: Mhm?

44) Hedvig: 80 grader?

--- (endrer i skript - prøver igjen) ---



FIGUR 17: UTKLIPP FRA SCRATCH - GRUPPE 2

Hedvig og Viktor er nå enig om å programmere en figur der alle de fem vinklene er like og sidene like lange. De får da med sikkerhet en femkant, og en løsning på det matematiske problemet oppgaven spør om (Vedlegg II – Oppgaver til Opplegget). I samtalen kan man se at det fortsatt ikke er sikkerhet om hvilke tall som skal stå som grader i skriptet, da det fortsatt prøves ut nye forslag. I skriptet er funksjonen [gå # steg] gjentatt 5 ganger med [vend # grader] fire ganger, plassert annenhver mellom klossene [gå # steg] (se skript i Figur 17). Hedvig og Viktor har nå endret fokus fra å snakke om form og kanter til å snakke om hvor mange grader vinklene skal være mellom hver kant. Den *konseptuelle forståelsen* de viser går ut på hva gradene gjør med den fysiske figuren. Med dette menes at de gjennom samtalen blant annet viser forståelse for sammenhengen mellom hva lavere og høyere tall som grader i skriptet fører til. Dette kan man blant annet se i Hedvigs ytring «Åja ... nei, men da må vi ... da kan vi ikke ha 60 grader. Da må vi ha mindre. Nei ... mer» (40). Viktor samtykker til å forsøke 60 grader i det han svarer «mhm» (41).

I dette utdraget er komponenten *beregning* fremtredende i form av å uttrykke de matematiske operasjonene fleksibelt, nøyaktig og effektivt. Dette uttrykkes vet at de blant annet bruker matematiske begrep som 60 grader, 100 steg, sekskant, mindre og mer. For å kunne komme med forslag og justeringer underveis i samtalen er det nødvendig å ta i bruk tidligere beregningsferdigheter. Ettersom endringene de gjør i skriptet gir umiddelbare svar i det de tester skriptet, kan de eventuelle antakelsene de hadde fra før justeres etter formålet – å programmere en femkant. Dette vil kunne være med på å utvide deres kunnskap om når og hvor en prosedyre eller operasjon er hensiktsmessig. Det innebærer her hvilke tall som skal settes inn i de ulike

klossene, hvilken rekkefølge de skal stå i og hvor mange som skal være av hver kloss. I dette utdraget kommer det også frem hvordan den algoritmiske tankegangen utspiller seg. Det er her avgjørende hvilket antall klosser de har, og hvilken rekkefølge klossene står i, da dette gir utslag i hvordan figuren blir seende ut.

Algoritmisk tenkning kan også tenkes å passe inn under *strategisk kompetanse*. Begrunnelsen er at algoritmisk tenkning handler om å ha kunnskap om hvordan rekkefølgen til de ulike faktorene som spiller inn påvirker hverandre. For å kunne representere, formulere og løse problemet elevene arbeider med, kreves det algoritmisk tenkning på grunn av skriptet som skal settes sammen. Elevene er avhengige av å sette sammen et skript som medfører at pennfiguren i Scratch tegner en femkant. Stegene i denne problemløsningsprosessen påvirkes av elevenes *strategiske kompetanse*, i det de benytter den kompetansen de har til å kunne formulere problemet overfor seg selv og medeleven. Dette kan man se i utdraget over hvor elevene sammen finner frem til løsningsmetoder som passer for å løse det matematiske problemet. For eksempel i ytring 30 og 31: Viktor: «*var det 60 grader?*», Hedvig: «*vi prøver*». Her blir 60 grader foreslått, og de velger å prøve det ut i skriptet for å se hva som skjer. I ytring 34-37 ser Viktor en sammenheng som ikke Hedvig ser, noe hun uttrykker slik at Viktor får mulighet til å forklare. Likevel løser forsøker hun å løse dette ved å prøve seg frem i Scratch: Viktor: «*Vi mangler bare én ... se?*», Hedvig: «*Ja, jeg skjønnte ikke helt ...*», Viktor: «*Nei ...?*», Hedvig: «*Men trykk på grønt flagg [peker på skjermen]*».

Den *strategiske kompetansen* går også ut på elevenes kompetanse til å forholde seg til de tekniske hjelpemidlene og ta i bruk disse til å gjøre endringer i Scratch. For å representere løsningen på det matematiske problemet krever det at elevene kan håndtere de forskjellige representasjonene av de matematiske omstendighetene. Dette gjør Hedvig og Viktor i det de sammen kommer frem til endringer som kan gjøres i skriptet for å komme nærmere målet. Ytring 39-42 fra samtaleutdraget over er et eksempel som viser hvordan Viktor og Hedvig sammen diskuterer hvor mange grader pennfiguren skal vende for å få en femkant i Scratch: Viktor: «*Nei, da blir det en sekskant*», Hedvig: «*Åja ... nei, men da må vi ... da kan vi ikke ha 60 grader. Da må vi ha mindre. Nei ... mer*», Viktor: «*Mhm*», Hedvig: «*Det var jo derfor han ... 70 grader kanskje? Eller 80? Skal vi prøve 80?*». De viser med dette forståelse for de ulike representasjonene ved å tolke det som skjer i Scratch, og formulerer mulige løsningsforslag – som hvor mange grader pennfiguren skal vende.

Som nevnt i analysen av tidligere utdrag er *resonnering* et redskap som også her blir brukt for å forklare og begrunne sine tanker og sin forståelse overfor mottakeren. I dette utdraget kan vi se at både Hedvig og Viktor uttrykker *konseptuell forståelse*, *beregning* og *strategisk kompetanse* gjennom *resonnering*. I deres arbeid har *resonnering* vist seg å være et nyttig redskap for å utvide sin forståelse i samhandling med hverandre. Gjennom *resonnering* har tankene deres kommet til uttrykk overfor samarbeidspartneren, som i eksempelet fra ytring 39-42 viser over. Her har Viktor gjort en beregning om at «*Nei, da blir det en sekskant*» (39). Da kommer Hedvig med flere ulike forslag til hvor mange grader pennfiguren skal vende for at de skal få en femkant. Hun snakker om mer og mindre (ytring 40), og nevner ulike konkrete forslag på grader til Viktor (ytring 42). Gjennom å resonnerer høyt overfor hverandre har de kunnet tolke hverandres ytringer og sammen klart å programmere en femkant i Scratch.

4.4. KONSTRUKSJON AV MATEMATISK KUNNSKAP

Denne delen av analysen fokuserer på det andre forskningsspørsmålet, som går ut på å undersøke hvordan elever konstruerer matematisk kunnskap når de programmerer. Det er derfor gjort en kvalitativ hermeneutisk tilnærming i analysen av utdrag fra transkripsjoner av gruppe 3. Analysen tar utgangspunkt i min forståelse av Seymour Papert (1991, 1993) og konstruksjonismens perspektiv på læring til å belyse forskningsspørsmålet.

4.4.1. GRUPPE 3

«Må vi lage én femkant og ett hus? Det er jo det samme»

Under vises et samtaleutdrag fra jentene i gruppe 3 sitt arbeid med femkanten (Vedlegg VIII). Jentene er gitt navnene Hanne og Vera. På tidspunktet utdraget under begynner har Hanne og Vera arbeidet cirka 1 ½ minutt med femkanten, og de er cirka 9 minutter inne i arbeidsøkten med oppgavene i Scratch. Denne situasjonen begynner med at Hanne foreslår å «*gå til punkt, gå til punkt*» (1). Metoden går sannsynligvis ut på å bestemme noen koordinater pennen skal bevege seg mellom, og angi disse i rett rekkefølge i skriptet. Denne ideen blir foreslått overfor Vera som betrakter den og gir sin reaksjon:

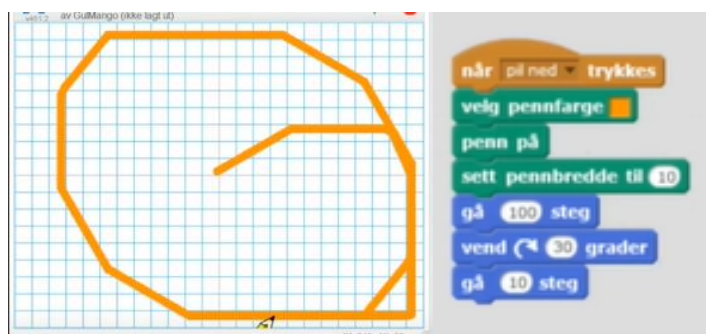
- 1) *Hanne: Ok ... Vi kan ha gå til punkt, gå til punkt ... For da er det litt lettere. Da kan vi gjøre sånn ... [peker på rutenett i Scratch]*
- 2) *Vera: Ja, men da må vi trykke for hver gang ... Da må vi endre på den.*
- 3) *Hanne: ... sånn ... Da lager vi hus. Femkant er hus.*
- 4) *Vera: [Henvender seg til Stud.] Må vi lage én femkant og ett hus? Det er jo det samme*
- 5) *Hanne: Man kan jo gjøre det til det samme ... [ler litt]*

I utdraget fra Hanne og Veras samtale kan man se at ideer og forslag til fremgangsmåter for å programmere en femkant blir presentert overfor samarbeidspartneren. Etter Hanne har foreslått å «gå til punkt, gå til punkt» (1) uttrykker Vera «Ja, men da må vi trykke for hver gang ... Da må vi endre på den» (2). Oppgaveteksten forteller blant annet at figuren skal kunne tegnes i én omgang (Vedlegg II), noe som Vera her ser ut til å mene Hannes metode kommer i konflikt med. Oppgaveteksten ber også elevene om å programmere andre figurer enn femkanten, deriblant et hus (Vedlegg II). Her har oppgavens retningslinjer skapt begrensninger for hvordan den kan løses, men gjør også elevene oppmerksomme på de geometriske kravene til figurene femkant og hus. «Hus» er ikke en geometrisk form med fastsatte krav. Elevene kan derfor bestemme mer fritt hvordan denne skal se ut. Fordi både hus og femkant kan bestå av femkant, kan samme figur representere både hus og femkant. Hanne og Vera velger å gå videre med denne ideen da de uttrykker i ytring 4 og 5 at hus og femkant er det samme.

Cirka et minutt videre i samtalen er imidlertid ideen om å benytte metoden «gå til punkt, gå til punkt» (2) gått bort ifra. Hanne og Vera har nå begynt å prøve ut skript med funksjonsklossene [gå # steg] og [vend # grader] (se blå klosser i skript i Figur 18). Til nå er skriptet allerede blitt testet flere ganger.

- 6) *Hanne: Jeg skal bare fikse den der ned, og så skal vi bare teste ...*
- 7) *Vera: Vi må bare trykke på pil ned.*
- 8) *Hanne: åja*
- 9) *Vera: hold inne*

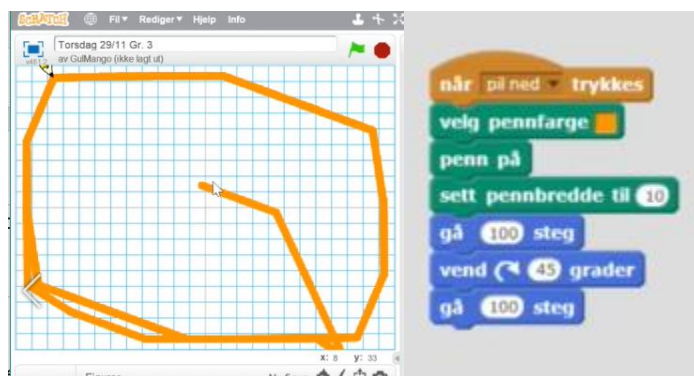
---(tester skript – holder inne kommandotasten)---



FIGUR 18: UTKLIPP FRA SCRATCH - GRUPPE 3

- 10) Vera: Ok, det var litt mer. Emmm ...
- 11) Hanne: Ja, men det var.. Hvis vi tar 20. 20.
- 12) Stud.: Husk at vi vil at hele figuren skal bli tegnet med å bare trykke én gang.
- 13) Hanne: jaja, men vi bare sjekker.. Eh ...
- 14) Vera: Vi bare tester. Em, ta 45 [peker på skript]. Jeg fikk veldig lyst til å prøve 45.
- 15) Hanne: 45? Men da snur du jo helt ...
- 16) Vera: Prøv.
- 17) Hanne: Åja, gå 100 steg.
- 18) Vera: og så tar du 45.
- 19) Hanne & Vera: 45 ...
- 20) Hanne: Og så må vi gjøre sånn ... ja, du kan

---(Vera tester skript – holder inne kommandotasten)---

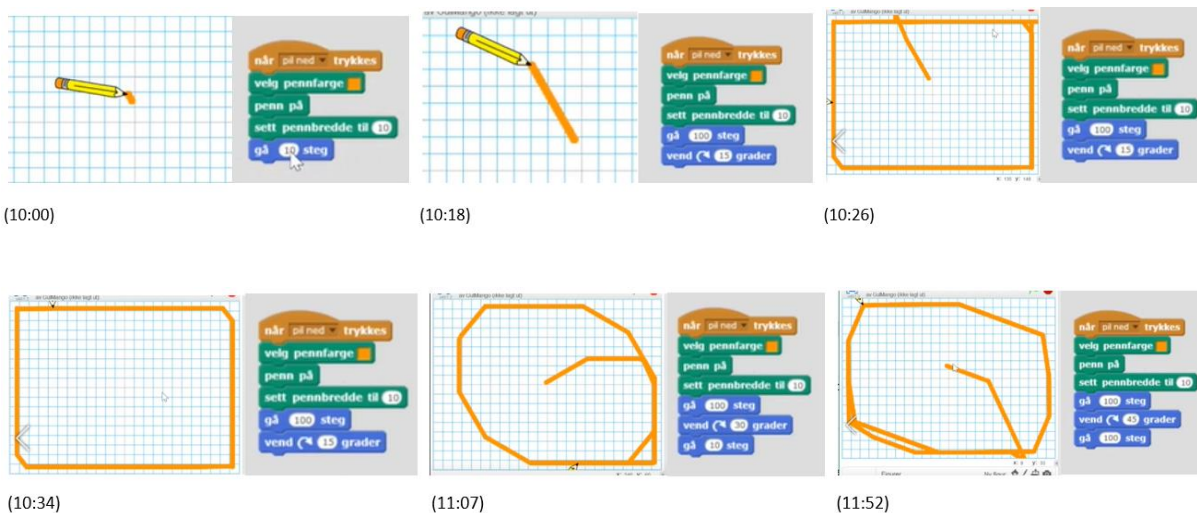


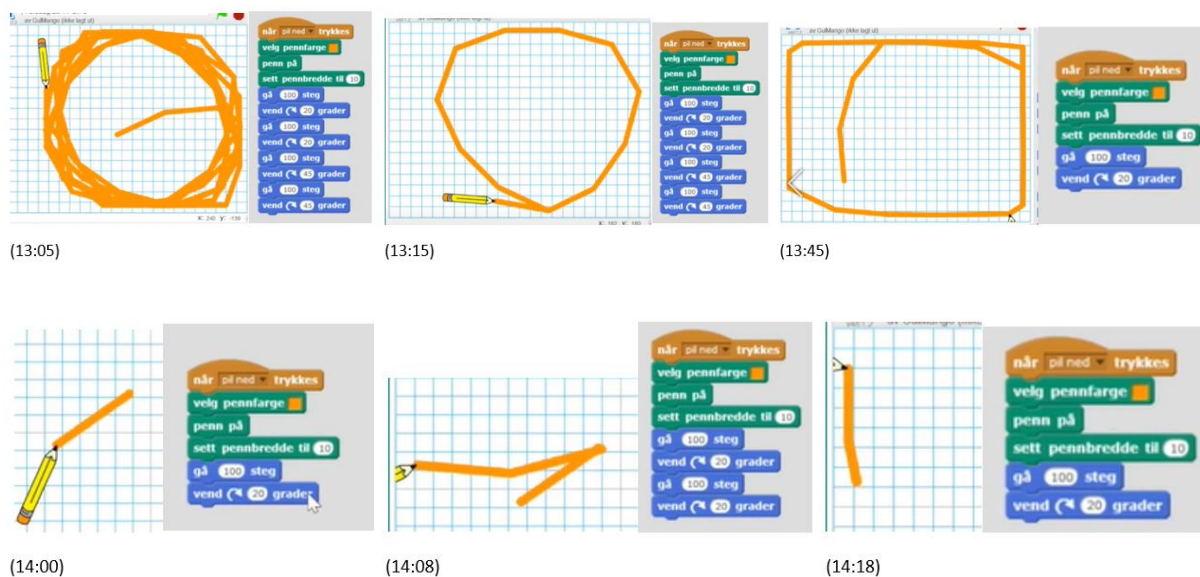
FIGUR 19: UTKLIPP FRA SCRATCH - GRUPPE 3

- 21) Vera: eee ...
- 22) [Begge ler]

I utdraget over kan man se at jentene fortsetter med å teste skriptet flere ganger ettersom nye forslag kommer frem og endringer gjøres. Man kan blant annet se de sier «Jeg skal bare fikse den der ned, og så skal vi bare teste ...» (6), «jaja, men vi bare sjekker [...]» (13), «Vi bare tester [...] Jeg fikk veldig lyst til å prøve [...]» (14) og «Prøv.» (16). I tillegg brukes moderatorer som *men, vi bare, kan* og andre uttrykk som «*eee ...*», «*og så [...]*», «*Åja*» og [*begge ler*] (22). Dette kan tolkes som at elevene er i tenkende resonnement. En prosess hvor tanker og ideer konstrueres og deles, for å kommunisere og samhandle med medeleven om dette kan være noe å gå videre på.

Jentene fortsetter med ideen om å benytte funksjonene [gå # steg] og [vend # grader], og prøver den ut gjentatte ganger ved å endre i tallene (#) i funksjonsklossene. De tester skriptene flere ganger, og holder ofte inne kommandotasten «*pil ned*». Dette gjør at skriptet blir kjørt flere ganger sammenhengende etter hverandre (se utklipp fra Scratch under i Figur 20). Utklippene under viser aktiviteten i Scratch etter hver gang skriptet er testet, før siste test av skript blir gjort mot slutten:





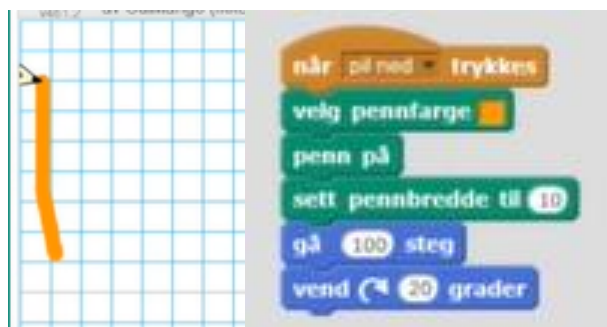
FIGUR 20: FLERE UTKLIPP FRA SCRATCH - GRUPPE 3 (SE OGSÅ TRANSKRIPSJONER VEDLEGG VIII)

Her kan vi se at jentene gjentatte ganger har forsøkt å finne rette kombinasjon av grader pennen skal vende, steg den skal gå, og repetisjoner og sammensetning av de to funksjonsklossene [gå # steg] og [vend # grader]. Hanne og Vera er begge samstemt med å bruke [gå 100 steg] for hver kant pennen tegner, men varierer antall grader pennen skal snu seg for hver gang. De begynner først med [vend 15 grader] før de etter 4 tester av skriptet øker til [vend 30 grader]. Etter dette øker de til [vend 45 grader], men legger så inn to funksjonsklosser med [vend 20 grader] etterfulgt av to med [vend 45 grader], og med [gå 100 steg] mellom hver av disse. På tiden 13:45 tar de så bort igjen dette lengre skriptet, og beholder [gå 100 steg] og [vend 20 grader]. For å få pennen til å tegne en sammenhengende figur har de derfor trykket på skriptet så det er blitt kjørt gjentatte ganger. På denne måten får de sett hva som skjer om skriptet hadde blitt repetert nedover for å tegne hele figuren ved ett klikk på kommandotasten. På tiden 14:08 har jentene flyttet pennen etter å ha fått den til å tegne en strek, og har derfor ikke en direkte sammenheng med skriptet som vises. Ved å se på hvordan jentene arbeider her, kan vi se at de er klar over at det er gradene som er den avgjørende faktoren for å programmere en femkant ut fra fremgangsmåten de benytter. Gjentatte ganger forsøker de derfor å endre tallene i funksjonsklossene ut fra hva som skjedde forrige gang de testet skriptet.

Etter å ha forsøkt denne metoden en del ganger går jentene imidlertid bort fra å prøve videre med å endre grader. De går i stedet tilbake til ideen Hanne presenterte i første utdrag om å «gå

til punkt, gå til punkt» (1). Hanne og Vera har til nå gjort de ulike testene som utklippene over viser, før fremgangsmåten igjen tar en ny vending:

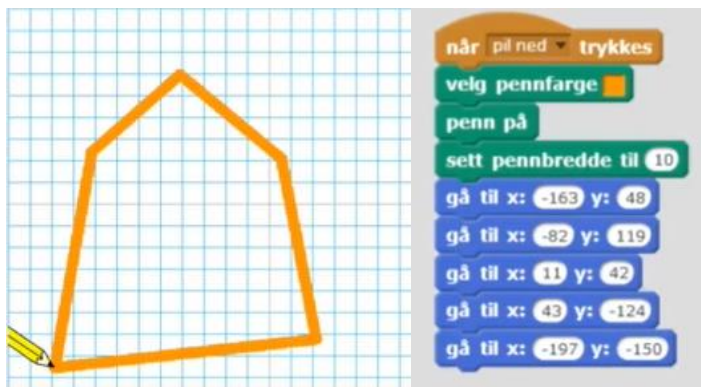
---(tester skript (14:18))---



FIGUR 21: UTKLIPP FRA SCRATCH - GRUPPE 3

- 23) Hanne: *mmm.. Kom igjen!*
- 24) Vera: *Eller så kan vi gå til punkt, gå til punkt, gå til punkt ... [fortsetter å gjenta mens Vera snakker]*
- 25) Hanne: *jaha, det var jo det jeg mente i sted.*
- 26) Vera: *[fortsetter å gjenta og peker i Scratch]*
- 27) Hanne: *Ja, det gjør vi. Ja, shhh!*
- 28) *[Vera ler]*
- 29) Hanne: *Jeg bare fjerner dette. Så kan du ...*
- 30) Vera: *[hvisker] Gå til punkt..*
- 31) Hanne: *okei, greit ... ee..*
- 32) Vera: *Nå skal jeg prøve. Bevegelse..*

Ideen som tidligere ble presentert av Hanne, men som det ble gått bort fra, blir nå igjen foreslått av Vera. Jentene har til nå forsøkt å få pennfiguren til å vende i ulike grader gjentatte ganger, men har kommet til et punkt hvor denne blir forkastet til fordel for en annen. Figuren og det tilhørende skriptet de til slutt ender opp med etter å ha fulgt metoden de nå begge er enige om å gå videre på, blir sende slik ut:



FIGUR 22: HANNE OG VERA SIN ENDELIGE FEMKANT SOM ER PROGRAMMERT I SCRATCH

Hanne og Vera har til da arbeidet i litt over 10 minutter med å programmere femkanten. De har på den tiden vært igjennom en arbeidsprosess hvor de sammen har kommet frem til en mulig løsning på oppgaven. I utdragene og utklippene over fra jentene i gruppe 3 kan man se hvordan kunnskap konstrueres etter hvert som de samhandler med hverandre. I dette samtaleutdraget kan det se ut til at det er ulike kunnskaper elevene konstruerer i den prosessen de går igjennom. For eksempel konstrueres trolig kunnskap om figurens egenskaper, ideer om hvordan Scratch kan brukes til å forme en femkant, og kunnskap om forskjellige sammenhenger mellom de ulike handlingene som utføres i Scratch. Å prøve ut ulike ideer, teste dem og se om de kan beholdes eller forkastes ser ut til å være sentral i Hanne og Veras arbeidsprosess.

5. DISKUSJON

I starten av oppgaven ble følgende problemstilling presentert: «*Hvordan kan programmering fremme matematisk kompetanse hos elever?*». Problemstillingen er blitt avgrenset ved å bli undersøkt i lys av de to følgende forskningsspørsmålene:

- 1) *Hvordan kommer matematisk kompetanse til uttrykk i elevers samtaler når de programmerer?*
- 2) *Hvordan konstruerer elever matematisk kunnskap når de programmerer?*

Med utgangspunkt i elevers samtaler i arbeid med programmering og matematikk er forskningsspørsmålene undersøkt for å belyse oppgavens problemstilling.

I dette kapitlet vil jeg trekke frem noen av de sentrale funnene i analysen, og diskutere hvor vidt funnene jeg har gjort besvarer forskningsspørsmålene. Jeg vil begynne med å ta for meg hva analysen kan fortelle om elevenes matematiske kompetanse. Videre vil jeg si noe om hvordan Seymour Papert sitt læringssyn kan knyttes til å si noe om hvordan elevene konstruerer matematisk kunnskap når de programmerer. Jeg har også sett det som relevant å diskutere om studiens analyse og resultat kan si noe om det fremmes relevante ferdigheter for det 21. århundre. Før jeg konkluderer vil jeg trekke studiens relevans til endringer av nye læreplaner i matematikk (Utdanningsdirektoratet, 2019). Dette er for å knytte studien opp mot matematikkfaget i skolen i dag, et fag jeg selv vil komme til å undervise i for dagens og fremtidens samfunnsborgere.

5.2. HVA KAN ANALYSEN FORTELLE OM ELEVENES MATEMATISKE KOMPETANSE?

Elevenes matematiske kompetanse er blitt belyst ved bruke et eget utformet rammeverk, sammensatt av to teoretiske rammeverk (se kapittel 2.3). Dette rammeverket har gjort det mulig å se etter de ulike komponentene av den matematiske kompetansen elevene uttrykker i arbeid med Scratch.

De fire komponentene *konseptuell forståelse*, *beregning*, *strategisk kompetanse* og *resonnering* er fremtredende i analysen. Dette gjenspeiler det både Kilpatrick et al. (2001) og Niss & Højgaard Jensen (2002) trekker frem, nemlig at matematisk kompetanse er en helhet bestående av flere deler som har et tett sammenknyttet gjensidig forhold. Elevene tar i bruk sin *konseptuelle forståelse*, *strategiske kompetanse*, *gjør beregninger* og bruker *resonnering* for å programmere en femkant. Dette har ført til at *engasjement* også er kommet til uttrykk i arbeidet. Den åpne oppgaven forteller ikke noe om fremgangsmåte og hva som er «rett svar». Elevene får derfor behov for å ta i bruk tidligere kunnskap for å generere ny kunnskap. Sammen utvikler de en felles forståelse slik at de kan arbeide videre med oppgaven.

Både samarbeid, den åpne oppgaven og Scratch som verktøy så ut til å ha vært medvirkende til at elevene sin matematiske kompetanse kom til uttrykk. Scratch er et verktøy som krever at elevene må vise fremgangsmåten gjennom sammensetning av et skript. Flere ganger i analysen kunne vi se at elevene var uenige om fremgangsmåten eller ikke forstod hva den andre mente. Vi kan si at det oppstod små «konflikter» mellom elevenes matematiske kompetanse etter hvert som den kom til uttrykk. Dette trenger ikke bety at den ene hadde en høyere «grad» av matematisk kompetanse enn den andre. I stedet kan dette vise at elevene tenkte ulikt om matematikk. Gjennom å samarbeide om å programmere en femkant, kom det gjennom kommunikasjonen frem hvordan elevene tenkte. Dermed kom også den matematiske kompetansen til uttrykk. I analysen av elevsamtalene kunne man se at ulike deler av den matematiske kompetansen ble utfordret etter hvert som elevene beveget seg videre i oppgaven. I starten av samtalene var den *konseptuelle forståelsen* fremtredende, og kom ofte til uttrykk gjennom *resonnering*. Dette kan ha kommet av at elevene skulle samarbeide, og det ble behov for å gjøre seg forstått overfor hverandre. Med for eksempel en ulik *konseptuell forståelse* av hva en femkant er, ble det nødvendig å komme til en felles forståelse. Man kan se at elevene arbeidet med dette ved å stille spørsmål til hverandre, tegne, forklare og argumentere. Dette ble også en del av den *strategiske kompetansen*, hvor de brukte hverandres ulikheter til å skape forståelse og tenke nytt.

Trekker man paralleller til hermeneutikken her kan man si at det skjedde flere horisontsammensmeltinger i elevenes forståelse underveis i arbeidet. Hver for seg hadde de en ulik forforståelse av hvordan noe kunne gjøres. Gjennom kommunikasjon og samarbeid fikk de delt hverandres tanker og kompetanse – hvor forståelse inngår. Etter hvert kom de til enighet om eller inngikk kompromisser for å forsøke den enes fremgangsmåte. Her ble Scratch et nyttig

verktøy, som raskt viser hvordan ulike sammensetninger av skriptet førte til forskjellige handlinger i programmeringsvinduet. Dermed ble det også enklere å vise overfor den andre hvordan man tenkte.

Senere i samtalen til både gruppe 4 og 2 ble flere begreper tatt i bruk og beregninger gjort. Skriptet ble testet gjentatte ganger hos begge gruppene, og endringer ble gjort ofte. Komponentene *beregning* og *strategisk kompetanse* ble mer fremtredende her enn tidligere i samtalen. Dette kan komme av at den *konseptuelle forståelsen* til elevene var blitt utvidet og de var kommet til en mer lik forståelse av ulike konsept, som hvordan femkanten kunne se ut. Dermed kunne de konsentrere seg mer om hvordan de skulle programmere femkanten, og komponentene *strategisk kompetanse* og *beregning* ble mer fremtredende. Elevenes samtale dreide seg etter hvert om å finne ut hvor mange grader pennfiguren skulle vende, eller hvor langt det skulle være mellom de forskjellige kantene i femkanten. Ulike strategier ble forsøkt, og flere beregninger ble gjort. Komponentene *beregning* ble derfor styrket ettersom den *strategiske kompetansen* ble tatt i bruk. Dette styrket også den *konseptuelle forståelsen* fordi det ble mer klart hvordan en femkant var satt sammen. Resonneringen gjorde at beregninger kunne komme til uttrykk, slik at de sammen kunne arbeide videre i Scratch.

Av ytringer som ble analysert, var det størst andel av ytringer som inneholdt alle de fire kodene [KBSR]. Årsaken til dette kan komme av det gjensidige forholdet mellom komponentene, noe som har gjort det vanskelig å skille dem fra hverandre i en analyse. Dette kan forklare den store andelen av disse kodene, og det store omfanget av alle de fire kodene registret i samme ytring. I tillegg kan det styrke påstanden om at den matematiske kompetansen ble fremmet da de programmerte i Scratch. At komponentene ble identifisert i flere ulike ytringer og situasjoner, kan tyde på at de forskjellige delene av elevenes matematiske kompetanse ble utfordret, utvidet og fremmet gjennom arbeidet. Den nye kompetansen elevene tilegnet seg her vil være med på å skape grunnlag for videre kompetanseutvikling.

Denne studien har vist at de matematiske komponentene jeg har tatt utgangspunkt i er blitt utviklet parallelt i elevenes samtaler. Komponentene har vist seg å bli styrket av hverandre, med et sterkt gjensidig forhold. I NOU-utredningen NOU 2015: 8 (2015, s. 57) er det tatt utgangspunkt i de fem komponentene fra Kilpatrick et al. (2001) for å beskrive matematisk kompetanse. Her er det påpekt at utviklingen av de matematiske komponentene må skje parallelt. Dette er fordi en sterk forbindelse mellom komponentene vil øke elevenes mulighet

for å utvikle varig, nyttig og relevant matematisk kompetanse. Ut fra det NOU-utredningen (NOU 2015: 8, 2015) trekker frem her, vil altså programmering i Scratch kunne brukes til å utvikle varig, nyttig og relevant matematisk kompetanse.

5.1.1. ER IKKE ELEVENE ENGASJERT I ARBEIDET?

Å dele opp matematisk kompetanse i helt adskilte kategorier har vist seg å være vanskelig. Dette kan være på grunn av den matematiske kompetansen består av en kompleks helhet, der de ulike delene styrkes av hverandre og har et tett gjensidig forhold (Kilpatrick et al., 2001; Niss & Højgaard Jensen, 2002). I denne forbindelsen er det påfallende at komponenten *engasjement* skiller seg så markant ut i resultatene fra kodingen, sammenlignet med de andre komponentene identifisert i datamaterialet. Tidligere forskning viser tydelig at programmering har potensiale for å fremme engasjement og motivasjon hos elever i lignende arbeidsøker. Samarbeid mellom medelever har også vist seg å ha en positiv effekt på å fremme engasjement i arbeidssituasjoner med matematikk (Forsström & Kaufmann, 2018). Resultatet som denne analysen viser, kan komme av kriteriet som rammeverket stiller til å identifisere denne komponenten. Datamaterialet dekker et relativt kort tidsrom til å se etter komponenten *engasjement*, med kun noen minutters samtale per gruppe. Det vil derfor være begrenset muligheter for å se om elevene opplever matematikk som fornuftig, nyttig og verdifullt, eller om de erkjenner verdien av å beherske matematikk.

Å undersøke om elevene har en positiv innstilling til faget kan også være utfordrende å identifisere, ut fra det begrensede datamaterialet og mitt sammensatte rammeverk. Likevel har dette sett ut til å være den mest opplagte indikatoren å gå etter for å kunne identifisere komponenten *engasjement* i elevsamtalene. Det kan da diskuteres hva som ligger i å *ha en positiv innstilling til faget*. Hvordan kan en innstilling komme til uttrykk i samtalen? At elevene ler eller viser tegn til glede er likevel brukt her som en indikator på at de har en positiv innstilling til faget.

Det kan tenkes at det blir for snevert å se etter uttrykk for glede for å se etter *engasjement*. En faktor som kunne vært forbundet med denne komponenten er elevenes interesse i oppgaven og i å finne en løsning. I analysen kan man se at elevene stadig kommer med nye forslag til endringer i fremgangsmåten, stiller spørsmål, tar tenkepauser, tegner, prøver og feiler. Dette

kan være tegn på engasjement i arbeidet, om komponenten *engasjement* hadde blitt tillagt en slik utvidet beskrivelse. Det omtalte tette og gjensidige forholdet mellom de fem matematiske komponentene, gjør at resultatene for komponenten *engasjement* kan gi et noe fortegnert bilde. Fenomenet blir også påpekt av Kilpatrick et al. (2001), hvor det uttales at *engasjement* utvikles om de andre trådene styrkes og har et gjensidig tett forhold. Siden analysen viser dette for de fire andre komponentene, vil en kunne anta at *engasjement* har vært mer fremtredende enn det kan se ut til i resultatene som er fremstilt her.

5.2. LEARNING BY MAKING

Papert (1991) beskriver ideen om «*Learning-by-making*» som en sentral idé bak konstruksjonismen og denne teoriens måte å se på læring. Den lærende går igjennom en prosess hvor læring oppstår ettersom han/hun konstruerer eller skaper kunnskap i sinnet. I denne læringsprosessen er det samhandling med andre mennesker og ting i miljøet rundt som framprovoserer kunnskap. Dette tillater de lærende aktørene å tenke, drømme, betrakte, få nye ideer som kan brukes videre eller forkastes, tid til å snakke, se andres arbeid og deres reaksjon på eget arbeid. Denne måten å lære på blir sammenlignet med hvordan det er å arbeide med matematikk for en matematiker (Papert, 1991). Flere kvaliteter ved denne måten å lære på gjør konstruksjonismen høyst aktuell i arbeid med matematikklæring. Dette illustreres om man setter konstruksjonismens læringsperspektiv opp mot beskrivelsen av hva matematikk er.

Ut fra slik Schoenfeld (1992) beskriver matematikk, er det en iboende sosial aktivitet som først kommer til uttrykk når et fellesskap engasjerer seg i å studere mønster. Boaler (2009) indikerer at matematikk er en levende handling, som en forestilling og en måte å entre verden på. Matematiske notasjoner er en måte å representere matematikk på, og å kommunisere den matematikken som er i ens sinn (Boaler, 2009; Devlin, 2000). Basert på dette vil kommunikasjon og samhandling med andre føre til at matematikk kommer til uttrykk. Elevenes matematiske kompetanse vil komme til uttrykk der matematikk blir anvendt på denne måten. I denne sammenhengen har Scratch fungert som et redskap, som tilbyr matematiske notasjoner for å uttrykke matematikk i samhandling med andre.

Gjennom å programmere geometriske figurer som femkanten i Scratch, har elevene fått mulighet til å anvende matematikk på flere måter. Forhold mellom mønstre er blitt belyst og

studert, og ulike uttrykksformer er benyttet for å konstruere og kommunisere matematisk kunnskap. Elevene har kommet med gjetninger og antakelser flere ganger, utforsket og avgrenset ideer, og kommunisert og diskutert disse med sin samarbeidspartner. I denne arbeidsprosessen er også elevenes glede og engasjement uttrykt i arbeidet (mer en komponenten *engasjement* viser). I følge Boaler (2009) kan man si at elevene har fått anvendt matematikk på en lignende måte som en matematiker ville gjort. De har fått mulighet til å kjenne på gleden som ligger i matematikk, samtidig som matematikk er lært på den mest produktive måten, ut fra slik Boaler (2009) uttrykker det.

I analysen har Papert (1991) sin idé om «*learning-by-making*» gjort seg gjeldende i det elevene har opptrådt som «byggmestre» for egen læring og konstruerer matematisk kunnskap. Kunnskapen blir konstruert individuelt, men skjer etter hvert som elevene har samhandler med hverandre, programmet Scratch, oppgaven (Vedlegg II) og med meg og min medstudent. Den matematiske kunnskapen kommer til uttrykk i samarbeidet med oppgaven. Å kunne anvende matematikk, slik som Devlin (2000) og Boaler (2009) beskriver det i kapittel .2.2.1 (om matematikk), har store likheter knyttet til Papert (1991, 1993) sin beskrivelse av en god læringssituasjon i matematikk – slik matematikere gjør det. De har til felles at anvendelse av matematikk bør handle om problemløsning, komme med antakelser og gjetninger, utforske og avgrense ideer, og å kommunisere og samhandle med andre i prosessen. Dette er noe som har vært gjenkjennelig i arbeids- og læringsprosessen til elevene i dette prosjektet. Da elevene arbeidet i Scratch inntok meg og min medstudent Oda en relativt passiv rolle. På bakgrunn av dette kan man si det skjedde læring uten å bli lært – slik Papert beskriver i boken *Mindstorms* (1993, s. 7). I stedet for å få formidlet kunnskap gjennom en lærer eller lærebok som forklarer, ble kunnskapen her i stedet konstruert mens elevene sammen lagde noe i Scratch. For å fortsette i samme metafor som Papert (1993) har brukt, kan man si at det ble tilrettelagt en form for kultur. Denne tilbydde «materialene» og et grunnlag for «byggmesterne» til å konstruere kunnskap.

Arbeidsprosessen ble en viktig del i læringssituasjonen til elevene i dette prosjektet, og tilbød flere mestringsituasjoner underveis. Det kan derfor tenkes at elevene opplevde prosessen som en viktig del av læringsarbeidet. Da femkanten skulle programmeres, hadde alle gruppene ulike fremgangsmåte og programmerte ulike varianter av femkanten. Hadde dette blitt tatt med videre i undervisningen, kunne det skapt et godt grunnlag for å se andre måter å løse oppgaven på. Elevene kunne fått tilbakemeldinger av hverandre, og fått forståelse for at det finnes flere måter

å løse en oppgave på. En slik undervisning kan bidra til å styrke en kultur der matematikk handler om å utforske mønster, som er mer enn å kunne beherske og anvende ulike algoritmer og formler. Dette vil imidlertid avhenge av hvordan man som lærer legger til rette for en slik matematikkundervisning, og tar i bruk mulighetene som finnes i de ulike situasjonene.

5.3. KOMPETANSE OG FERDIGHETER FOR FREMTIDEN

Ulike rammeverk for det 21. århundre trekker frem kjerneferdigheter som anvendelse av teknikk og informasjon, kommunikasjon, samarbeid, kreativitet, kritisk tenking og problemløsning (Van Laar et al., 2017). I elevenes arbeidsøkt med geometri i Scratch har elevene måtte anvende teknikk og informasjon, slik at pennfiguren ble programmert rett til å tegne den ønskede figuren. Ettersom oppgaven var en samarbeidsoppgave ble kommunikasjonsferdigheter og samarbeid benyttet og testet. Dette viste seg å få elevene til å tenke kritisk om egne ideer ettersom det oppstod uoverensstemmelser i elevenes forståelse. Kreative forslag ble kommunisert og utarbeidet, og ble en særlig nødvendig ferdighet å benytte da oppgaven var åpen og oppmuntret til kreative løsninger. Alle disse faktorene var både fremtredende og viktige i elevenes problemløsningsprosess med å utføre oppgaven i Scratch. I en fremtidig arbeidshverdag vil en slik måte å arbeide på kunne være naturlig. Samarbeid vil fortsatt være viktig på arbeidsplassen, der kommunikasjon er en naturlig del av dette. Karakteristisk for innovativ tenkning er at man ikke er redd for å komme med kreative nye løsninger og å prøve og feile. En slik måte å arbeide på i grunnopplæringen vil kunne utvikle elevens kunnskap og kompetanse. Dette vil være nyttig for å skape aktive deltakere i dagens og morgendagens samfunn (NOU 2015: 8, 2015, s. 7).

Situasjonene som er analysert viser kun indikasjoner på at det fremmes kompetanse og ferdigheter som vil være viktig i det 21. århundre. Det vil derfor ikke være nok med kun enkelte lignende arbeidsøkter i opplæringen med et slikt mål. Å utvikle kompetanse krever tid og stadig nye utfordringer. Etter hvert som elevene blir kjent med Scratch vil de også her kunne «låses» til kjente løsningsmetoder, noe som kan hindre dem til å tenke nytt. Skal elevene arbeide med å programmere i Scratch vil det kreves oppgaver som tar utgangspunkt i elevenes forutsetninger. Da vil de både utfordres og oppleve mestring. Flere faktorer vil være avgjørende for at ønskede kvaliteter kan fremmes og utvikles. Disse faktorene kan verne sammensetningen av de ulike elevenes kompetanse, lærerens tilrettelegging, den gitte oppgaven og de

tilgjengelige ressursene. I dette arbeidet vil læreren være sentral. Dette er fordi det er lærerens oppgave å sette sammen disse avgjørende faktorene på en formålstjenlig måte ut fra forutsetningene.

For at det skal være relevant å integrere programmering i matematikkundervisningen, ser jeg det som naturlig at programmering og matematiske tema kan kombineres. Her har arbeidet med geometri i Scratch åpnet for nye muligheter. Som denne studien viser kan programmering i Scratch brukes til å fremme kompetanse og ferdigheter for det 21. århundre, samtidig som det fremmes matematisk kompetanse. Scratch krever at elevene forstår prosessen og oppbyggingen for å få det ønskede produktet eller resultatet. Ved å sette sammen et skript, blir elevene mer bevisst på de matematiske prosessene bak figuren som programmeres. Måten elevene har arbeidet på i dette prosjektet har vist seg å kreve en variert bruk av kompetanse og ferdigheter, som også fremheves som viktig i dagens og fremtidens samfunn.

Gjennom programmering av geometriske figurer i Scratch, har elevene arbeidet med forstå prosessen og oppbyggingen av de matematiske operasjoner og prosedyrer. I følge Røttingen (2018) vil en slik kompetanse være viktig i en digital fremtid der mye gjøres av roboter, maskiner og programmer. Dette kan være med på å øke bevisstheten rundt produkter vi benytter oss av, eller kanskje vil være med på å videreutvikle. Evnen til å sette seg inn i et matematisk problem og å prøve og feile til man finner en løsning, vil være en del av den kompetansen som fortsatt vil være viktig for fremtiden.

For å lykkes med fremtidens arbeidsliv er vi avhengige av gode utdanninger som evner å tilpasse seg endringene i samfunnet. Et endret arbeidsliv er nødt til å få inn relevant kompetanse for å tilpasse seg og tilegne seg det nye.

(Røttingen, 2018)

Utdraget fra nettmagasinet Dagens Perspektiv trekker frem det samfunnsansvaret grunnopplæringen er en stor del av. Vi må tørre å tenke nytt og tilpasse oss de endringene samfunnet er i. Utdanningen må være bevegelig og dynamisk slik at vi beveges i takt med samfunnets kompetansebehov. For å blant annet sikre innovasjon, et konkurransedyktig næringsliv og kompetanseutvikling i andre vitenskapsfag er vi avhengig av matematikk. At en god opplæring av matematisk kompetanse vil være viktig fremover er det derfor bred enighet om (NOU 2015: 8, 2015, s. 24). Å utvikle matematisk kompetanse i en retning som vil tjene samfunnets kompetansebehov er viktig. Gjennom å utvikle matematisk kompetanse i

forbindelse med andre kompetansebehov i samfunnet, vil undervisningen i grunnskolen kunne legge et godt grunnlag for den videre utviklingen.

5.4. FAGFORNYELSEN – MATEMATIKKFAGET I ENDRING

Denne våren (2019) er fagfornyelsen for siste gang ute for høring før den skal ut i skolen (Utdanningsdirektoratet, 2019). Formålet med endringene er å knytte læreplanene tettere opp mot elevenes hverdag, og forberede dem på et samfunn og arbeidsliv i endring. I læreplanene for matematikkfaget vil det legges vekt på at elevene skal bli gode problemløsere. Det skal i større grad legges til rette for utforsking, og kommunikasjon skal ta mer plass i matematikkfaget. I tillegg trekkes det frem at elevene skal kunne oppdage sammenhenger i og mellom kunnskapsområder i matematikkfaget og andre fag. Med dette skal det gis mer dybdelæring og forståelse i faget.

Settes denne oppgavens perspektiv på matematikk og matematikklæring opp mot fornyelsen av læreplanene i matematikk, kan det trekkes flere paralleller. Papert (1991) ønsket å se matematikkundervisning mer slik en matematiker ville arbeidet med matematikk. Da vil både utforsking og kommunikasjon, problemløsning og å kunne gjøre oppdagelser alene og sammen med andre være en naturlig del av det å arbeide med matematikk. Ut fra det som er kommet frem i studien, kan de endringen som gjøres i matematikkfaget nå gjenspeiles i arbeidsøkten til elevene i dette prosjektet.

5.4.1. ALGORITMISK TENKNING

Algoritmisk tenkning trekkes frem flere steder som den sentrale linken som kobler sammen programmering og matematikk (Forsström & Kaufmann, 2018; Kjerneelementgruppen for matemaikk, 2017; Utdanningsdirektoratet, 2016; Wing, 2006). Algoritmisk tenkning og programmering er derfor gitt plass i matematikkfaget, ut fra slik det står i siste høring av de nye læreplanene. Bakgrunnen for at algoritmisk tenkning skal vektlegges, er at det legges frem som en viktig problemløsningsstrategi. I denne forbindelse kan programmering fungere som et verktøy til å utforske og løse problemer, og å utvikle matematisk forståelse

(Utdanningsdirektoratet, 2019). Denne studien har vist at programmering har potensiale til å innfri disse forventningene.

Elevenes algoritmiske tenkning er blitt utfordret og har utgjort en sentral del i elevenes programmeringsarbeid i Scratch. Sammensettingen av skriptet i Scratch er kanskje den delen av arbeidet som tydeligst kan vise elevenes algoritmiske tankegang, ved at det krever en slik måte å tenke på. Komponentene *strategisk tankegang* og *konseptuell forståelse* er også de to komponentene som er mest fremtredende i analysen. Flere ganger hvor disse komponentene er identifisert, kan det knyttes til algoritmisk tenkning og problemløsning. For eksempel dreier store deler av elevenes samtaler seg rundt hvordan skriptet skal settes sammen for at det skal bli en femkant. Etter hvert i arbeidsprosessen har denne måten å arbeide på fått elevene til å anvende matematikk på ulike måter. Programmering kan ut fra dette sies å fremme matematisk kompetanse blant annet gjennom algoritmisk tenkning.

5.5. MULIGHETER OG BEGRENSNINGER VED BRUK AV SCRATCH

Denne studien har vist at programmering i Scratch kan fungere godt for å fremme elevers matematiske kompetanse, her med særlig vekt på geometri. Et verktøy som Scratch kan åpne for muligheter til å arbeide med matematikk på flere måter. Det kan fremme kvaliteter som støttes av både teori, læringsperspektiv, rammeverk og læreplaner som beskriver hvordan en god matematikkundervisning kan være. Programmering i Scratch kan også skape et godt grunnlag for samarbeid, kommunikasjon, diskusjon, problemløsning, algoritmisk tenkning, kreativitet, mestring og engasjement. Elevenes matematiske kompetanse kan gjennom dette fremmes på en formålstjenlig måte.

Selv om det finnes gode muligheter til å fremme matematisk kompetanse gjennom programmering, kreves det bevisst arbeid fra lærerens side for å dra nytte av dette potensialet. Det må tas hensyn til elevenes forutsetninger, og tilpasses etter hva man ønsker å oppnå med å integrere et slikt verktøy i undervisningen. Å programmere i Scratch byr på flere nye utfordringer elevene kanskje ikke er vant med. Blant annet er det et nytt språk – programmeringsspråket – som er satt sammen på en annen måte enn andre språk elevene kjenner fra før. Begrepsavklaringer vil være nødvendig for å ta hensyn til elevgruppen, men kan også by på gode samtaler rundt bevisstgjøring av disse.

I pilotgjennomføringen ble det lagt merke til hvordan tekniske utfordringer kunne begrense elevene i arbeid med å utføre de oppgavene som ble gitt. Det viste seg å være en fordel at elevene fikk gjort seg kjent med programmet på forhånd. Da fikk de prøve seg frem og teste ut ulike muligheter i Scratch slik at de ble bedre kjent med verktøyet. I gjennomføringen av Opplegget ble det likevel mye fokus på det tekniske, som å finne frem til ulike funksjoner i Scratch. Gjennom masterprosjektet opplevde jeg at tekniske utfordringer noen ganger kunne overskygge eller begrense det matematiske. En del tid gikk derfor med til utfordringer som ikke var direkte relevant til den matematiske oppgaven. Ut fra disse erfaringene, ser jeg det derfor som nødvendig at det brukes tid til å sette seg inn i programmet. Da får både lærer og elever tid til å sette seg inn i programmeringsverktøyet. Som lærer kan det også være lurt å på forhånd sette seg inn noen av de utfordringer elevene kan møte på. Da kan disse utfordringene bli forklart i forkant, og det kan brukes mer tid på andre faglige eller uforutsette utfordringer. På denne måten kan man også bli mer fleksibel og kreativ til å se de forskjellige mulighetene som programmet har.

6. AVSLUTNING

I denne oppgaven er det undersøkt hvordan programmering kan fremme matematisk kompetanse hos elever, som er oppgavens problemstilling. Ved bruk relevant teori og innsamlet data er problemstillingen undersøkt gjennom de to forskningsspørsmålene:

- 1) *Hvordan kommer matematisk kompetanse til uttrykk i elevers samtaler når de programmerer?*
- 2) *Hvordan konstruerer elever matematisk kunnskap når de programmerer?*

I dette kapitlet presenteres noen avsluttende refleksjoner. Til slutt vil jeg rette blikket mot videre forskning på feltet jeg har undersøkt.

6.2. AVSLUTTENDE REFLEKSJONER

Formålet med studien har vært å få en bedre innsikt i hvordan programmering kan brukes som et verktøy for å fremme elevenes matematiske kompetanse i matematikkundervisning. Bakgrunnen for denne interessen har vært implementeringen av programmering i matematikkfaget i grunnskolen som en del av fagfornyelsen (Utdanningsdirektoratet, 2018a). I tillegg har jeg i arbeid med dette prosjektet fått et større perspektiv på hvilken betydning denne endringen kan ha for fremtidens samfunn. Som nyutdannet lærer i grunnskolen vil jeg bære ansvar for å integrere programmering i matematikkfaget på en hensiktsmessig og god måte for elevene. Denne studien har vist at programmering i Scratch kan bidra positivt til å fremme ulike komponenter av en helhetlig matematisk kompetanse hos elever. Særlig er det kommet frem hvordan ulike komponenter av elevenes matematiske kompetanse gjensidig avhenger og styrkes av hverandre. Måten elevene arbeidet på viser at kvaliteter som samarbeid, kreativitet, resonnering, engasjement og å prøve og feile har preget arbeidet og problemløsningsprosessen. Dette har ført til at elevene sammen har kommet frem til en løsning på oppgaven gjennom en lærerik prosess. Det er blitt lagt til rette for å konstruere matematisk kunnskap og komponenter av matematisk kompetanse er blitt fremmet på forskjellige områder.

Studien som er gjennomført bygger på et lite utvalg informanter. Det kunne derfor vært interessant å se om det samme resultatet hadde forekommet med en annen elevgruppe, et større

utvalg informanter eller et lavere eller høyere alderstrinn. Det ville også vært interessant å se utbytte av å arbeide alene med programmeringsverktøyet for å fremme matematisk kompetanse. Hva elevene får ut av å anvende programmering og matematikk sammen, vil imidlertid avhenge av hvordan jeg som lærer setter meg inn i og tar i bruk verktøyet. Her vil det være avgjørende å ta hensyn til elevenes forutsetninger og undervisningens formål.

I denne studien er det rettet fokus på hvordan programmering kan fremme kompetanse knyttet til matematikkfaget. Scratch er brukt som et verktøy til hovedsakelig å lære matematikk, med programmering som et hjelpemiddel. I utgangspunktet er Scratch utviklet for å lære barn å programmere. Derfor har programmet et mye større bruksområde og potensial om formålet er å lære elevene hva programmering er og hvordan det fungerer. Å lære elever å programmere er i seg selv nyttig læring ifølge rammeverk for ferdigheter i det 21. århundre (Dede, 2010). Om målet er å lære elevene å programmere, vil det ikke være tilstrekkelig å kun arbeide med programmering i matematikkfaget. Det kunne derfor videre vært interessant å se på hvordan programmering kunne vært kombinert med andre fag, eller eventuelt vært en del av et eget IKT-fag i grunnskolen. I tillegg til å kunne fremme matematisk kompetanse, har studien vist andre kvaliteter gjennom å arbeide i par med programmering. Både kreativitet, samarbeid, problemløsning og engasjement er noe av det som har preget elevenes arbeid med oppgavene i Scratch. Det kan derfor tenkes at disse kvalitetene ville kunne gjort seg gjeldende og virket positivt på læring ved å bruke Scratch til å lære flere emner. En videre forskning på dette kunne vært interessant å sett resultatet av.

Med denne studien har jeg fått større innblikk i hvordan jeg som lærer kan bruke programmering i undervisning for å fremme matematisk kompetanse. Jeg har også fått dypere kjennskap til forskning innen dette feltet, og utvidet min forståelse for matematikklæring. Dette er noe jeg også ønsker for andre lærere og studenter. Jeg håper derfor denne studien kan bidra til en økt interesse for hvordan programmering kan brukes på en formålstjenlig måte i matematikkundervisning.

6.3. PÅGÅENDE FORSKNING OG VEIEN VIDERE

Denne studien er en del av LATICME (Learning About Teaching Argumentation for Critical Mathematical Education in multilingual classrooms), et større forskningsprosjekt som fortsatt

pågår og vil være avsluttet ved utgangen av 2021. Dette forskningsprosjektet kan følges på nettsiden <https://prosjekt.hvl.no/latacme/>. Prosjektets overordnede fokus er på argumentasjon og kritisk matematikdidaktikk i flerspråklige klasserom. *Argumentasjon og IKT* er ett av tre delprosjekt. Denne studien utgjør kun en liten brikke i LATACME sitt forskningsfelt. Det vil derfor være interessant å følge med på den videre forskning og de resultatene som vil komme frem i forbindelse med LATACME. Flere masterstudenter vil trolig tilknyttes dette forskningsprosjektet, og jeg håper disse vil la seg inspirere av meg og mine medstudenters studier i LATACME til videre forskning.

Bakgrunnen for studien har lagt mye vekt på rammeverk, utredninger og læreplaner som sier noe om hva elever bør lære som vil være relevant for fremtiden. Den fjerde industrielle revolusjonen og den teknologiske utviklingen har vi kun sett starten av. Utviklingen vil fortsette å skyte fart, noe som påvirker samfunnet, arbeidslivet og dermed også utdanningen. Dette vil kreve mer forskning for å kunne si noe om hva slike endringer kan føre til. En endring vi befinner oss i nå er blant annet å bruke programmering i matematikkundervisning, noe denne studien har forsøkt å rette søkelys mot. Det vil imidlertid fortsatt være et stort behov for mer forskning knyttet til dette feltet. I nærmeste fremtid vil det kanskje særlig være interessant å forske videre på dette etter de nye læreplanene er blitt en del av skolehverdagen til flere elever og lærere. Hvordan dette vil påvirke elevenes læring vil være spennende å følge med på og se utbytte av. Samtidig vil det være spennende å følge med på utviklingen av den kompetansen samfunnet og arbeidslivet etterspør.

LITTERATURLISTE

- Backe-Hansen, E. & Frønes, I. (2012). *Metoder og perspektiver i barne- og ungdomsforskning*. Oslo: Gyldendal akademisk.
- Bjørndal, C. R. P. (2017). *Det vurderende øyet : observasjon, vurdering og utvikling i undervisning og veiledning* (3. utg. utg.). Oslo: Gyldendal akademisk.
- Boaler, J. (2009). *The elephant in the classroom : helping children learn and love maths* (Rev. ed. utg.). London: Souvenir.
- Civil Society in the Fourth Industrial Revolution: Preparation and Response. (2019). Cologny/Geneva: World Economic Forum. Hentet fra http://www3.weforum.org/docs/WEF_Civil_Society_in_the_Fourth_Industrial_Revolution_Response_and_Innovation.pdf
- Dede, C. (2010). Comparing Frameworks for 21st Century Skills. I J. Bellanca & R. Brandt (Red.), *21st Century Skills: Rethinking how Students Learn*. Hentet fra https://books.google.no/books?hl=no&lr=&id=aGYXBwAAQBAJ&oi=fnd&pg=PT79&dq=framework+for+21st+century+skills&ots=wbf5xZ5U0J&sig=NPkjih1N6cA6kBTZlxcIFqRwWdU&redir_esc=y#v=onepage&q&f=true
- Devlin, K. (2000). *The math gene : how mathematical thinking evolved and why numbers are like gossip*. New York: Basic Books.
- Forsström, S. E. & Kaufmann, O. T. (2018). A systematic literature review exploring the use of programming in mathematics education. Østfold University Collage, Faculty of Education, Norway, Østfold.
- Gadamer, H.-G. (2004). *Sandhed og metode : grundtræk af en filosofisk hermeneutik*. Århus: Systime.
- Gilje, N. & Grimen, H. (1993). *Samfunnsvitenskapenes forutsetninger : innføring i samfunnsvitenskapenes vitenskapsfilosofi* ([3. prøveutg.]. utg.). Oslo: Universitetsforlaget.

Grønmo, S. (2016). *Samfunnsvitenskapelige metoder* (2. utg. utg.). Bergen: Fagbokforl.

Husain, H., Kamal, N., Ibrahim, M. F., Huddin, A. B. & Alim, A. A. (2017). Engendering problem solving skills and mathematical knowledge via programming. *Journal of Engineering Science and Technology*, 12(Special Issue 12), s. 1-11. Hentet fra <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85044470431&partnerID=40&md5=e09134067e58a830c0e73a6bae82be6f>

Johnsen, E. & Olsen, E. (2015). Elevers matematiske forklaringer - Praktisk matematisk baserte likheter og ulikheter Hentet fra <https://hdl.handle.net/10037/8109>

Kilpatrick, J., Swafford, J. & Findell, B. (2001). *Adding it up: helping children learn mathematics*.

Kjerneelementgruppen for matemtaikk. (2017, 21. september). Kjerneelementer i matematikk, men hvorfor programmering? Hentet fra <http://udirbloggen.no/kjerneelementer-i-matematikk-men-hvorfor-programmering/>

Kjørup, S. (2008). *Menneskevidenskabene : 2 : Humanistiske forskningstraditioner* (2. udg. utg.). Frederiksberg: Roskilde Universitetsforlag.

Kommunal- og moderniseringsdepartementet. (2017, 25. januar). *Vekst og arbeidsplasser i teknologiens tidsalder*. Hentet fra <https://www.regjeringen.no/no/aktuelt/vekst-og-arbeidsplasser-i-teknologiens-tidsalder/id2536213/>

Kunnskapsdepartementet. (2017). Mer koding og teknologi inn i skolen: Kunnskapsdepartementet. Hentet fra <https://www.regjeringen.no/no/aktuelt/mer-koding-og-teknologi-inn-i-skolen/id2568375/>

Kunnskapsdepartementet. (2018). Fornyer innholdet i skolen. Hentet fra <https://www.regjeringen.no/no/aktuelt/forny-er-innholdet-i-skolen/id2606028/>

Kvale, S. & Brinkmann, S. (2009). *Det kvalitative forskningsintervju* (2. utg. utg.; T. M. Anderssen & J. Rygge, Overs.). Oslo: Gyldendal akademisk.

Lang-Ree, H. L. (2016). "Vi må tenke og ikke bare tegne" - En kvalitativ studie om bruk av programmering som verktøy i arbeid med matematikk Hentet fra

https://brage.bibsys.no/xmlui/bitstream/handle/11250/2396148/Lang-Ree_2016.pdf?sequence=

Logo Foundation. (2015a). Logo and Learning. Hentet fra http://el.media.mit.edu/logo-foundation/what_is_logo/logo_and_learning.html

Logo Foundation. (2015b). Logo History. Hentet fra http://el.media.mit.edu/logo-foundation/what_is_logo/history.html

Lynnebakken, H. (2018, 11. oktober). Programmering er ikke det samme som koding: - Den største jobben er å finne ut hva du ønsker å gjøre. Hentet fra <https://www.digi.no/artikler/programmering-er-ikke-det-samme-som-koding-den-storste-jobben-er-a-finne-ut-hva-du-onsker-a-gjore-og-formulere-det-pa-en-presis-mate/448366>

Mellin-Olsen, S. & Lindén, N. (1996). *Samtalen som forskningsmetode : tekster om kvalitiv [i.e. kvalitativ] forskningsmetode som del av pedagogisk virksomhet*. Landås: Caspar forlag.

Moreno-León, J., Robles, G. & Román-González, M. (2016). Code to learn: Where does it belong in the K-12 curriculum? *Journal of Information Technology Education: Research*, 15, s. 283-303. doi:10.28945/3521 Hentet fra <http://www.informingscience.org/Publications/3521>

Niss, M. & Højgaard Jensen, T. r. (2002). *Kompetencer og matematiklæring – Ideer og inspiration til udvikling af matematikundervisning i Danmark*. Uddannelsesstyrelsens temahæfteserie nr. 18 -2002. Hentet fra <http://www.gymnasieforskning.dk/wp-content/uploads/2013/10/Kompetencer-og-matematiklaering1.pdf>

Nortvedt, G. A. & Pettersen, A. (2016). 6 Matematikk. I *Stø kurs* (s. 107-135)

NOU 2015:8. (2015). *Fremtidens skole. Fornyelser av fag og kompetanser*. Oslo: Hentet fra <https://www.regjeringen.no/no/dokumenter/nou-2015-8/id2417001/?ch=1&q=>

NOU 2015: 8. (2015). *Fremtidens skole. Fornyelser av fag og kompetanser*. Oslo: Hentet fra <https://www.regjeringen.no/no/dokumenter/nou-2015-8/id2417001/?ch=1&q=>

M120UND509, kandidatnummer: 107, 15. mai 2019

Papert, S. (1991). Situating Constructionism. I S. H. Papert, Idit (Red.), *Constrictionism*. Hentet fra <http://www.papert.org/articles/SituatingConstructionism.html>

Papert, S. (1993). *Mindstorms : children, computers, and powerful ideas* (2nd ed. utg.). New York: Basic Books.

Partnership for 21st Century Skills. (2019). Framework for 21st Century Learning Hentet fra <http://www.battelleforkids.org/networks/p21/frameworks-resources>

Partnership for 21st Century Skills. (u.å.). Framework for 21st Century Learning Hentet fra <https://wvde.state.wv.us/counselors/links/initiatives/21st-century-framework.pdf>

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., ... Kafai, Y. (2009). Scratch: Programming For All. *Communications of the ACM*, 52(11), s. 60-67. doi:10.1145/1592761.1592779 Hentet fra <https://m-cacm.acm.org/magazines/2009/11/48421-scratch-programming-for-all/fulltext>

Ringdal, K. (2018). *Enhet og mangfold : samfunnsvitenskapelig forskning og kvantitativ metode* (4. utg. utg.). Bergen: Fagbokforl.

Rossen, E. (2017, 13. desember). Programmeringsspråk. Hentet fra <https://snl.no/programmeringsspr%C3%A5k>

Ruyter, K. W. (2003). *Forskningsetikk : beskyttelse av enkeltpersoner og samfunn*. Oslo: Gyldendal akademisk.

Røttingen, J.-A. (2018, 14. januar). Forskning om framtidens arbeidsliv. Hentet fra <https://www.dagensperspektiv.no/2018/forskning-om-framtidens-arbeidsliv-0>

Schoenfeld, A. H. (1992). Learning to think mathematically: Problem solving, metacognition, and sense-making in mathematics. I D. Grouws (Red.), *Handbook for Research on Mathematics Teaching and Learning* (s. 334-370).

Schwab, K. (2016). *The Fourth Industrial Revolution*. Hentet fra <https://luminariaz.files.wordpress.com/2017/11/the-fourth-industrial-revolution-2016-21.pdf>

Senter for IKT i utdanningen. (2016). *Programmering i skolen*. Senter for IKT i utdanningen.

Hentet fra https://www.udir.no/globalassets/filer/programmering_i_skolen.pdf

Tangen, R. (2010). «Beretninger om beskyttelse» – Ethiske dilemmaer i forskning med sårbare grupper – barn og ungdom. *Norsk pedagogisk tidsskrift*, 94(04), s. 318-332. Hentet fra

<http://www.idunn.no/npt/2010/04/art09>

Taylor, M., Harlow, A. & Forret, M. (2010). Using a Computer Programming Environment and an Interactive Whiteboard to Investigate Some Mathematical Thinking. *Procedia - Social and Behavioral Sciences*, 8, s. 561-570. doi:10.1016/j.sbspro.2010.12.078

Hentet fra <https://dx.doi.org/10.1016/j.sbspro.2010.12.078>

Tjora, A. H. (2017). *Kvalitative forskningsmetoder i praksis* (3. utg. utg.). Oslo: Gyldendal akademisk.

Utdanningsdirektoratet. (2016). *Teknologi og programmering for alle - En faggjennomgang med forslag til endringer i grunnlæringen - august 2016*. Oslo: Hentet fra

<https://www.udir.no/globalassets/filer/tall-og-forskning/forskningsrapporter/teknologi-og-programmering-for-alle.pdf>

Utdanningsdirektoratet. (2017). Hvorfor fornyer vi fagene? [Video]. Hentet fra

<https://vimeo.com/250453658>

Utdanningsdirektoratet. (2018a). Fagfornyelsen. Hentet 4. november 2018 fra

<https://www.udir.no/laring-og-trivsel/lareplanverket/fagfornyelsen/>

Utdanningsdirektoratet. (2018b, 5. mars). Fagfornyelsen - siste innspillsrunde kjerneelementer.

Hentet fra <https://hoering.udir.no/Hoering/v2/197?notatId=358>

Utdanningsdirektoratet. (2018c, 22. oktober). Overordnet del av læreplanverket. Hentet 11. april 2019 fra

<https://www.udir.no/laring-og-trivsel/lareplanverket/overordnet-del/prinsipper-for-laring-utvikling-og-danning/kompetanse-i-fagene/>

Utdanningsdirektoratet. (2019, 18. mars). Høring - læreplaner i matematikk. Hentet fra

<https://hoering.udir.no/Hoering/v2/343>

M120UND509, kandidatnummer: 107, 15. mai 2019

Van Laar, E., Van Deursen, A. J. A. M., Van Dijk, J. A. G. M. & De Haan, J. (2017). The relation between 21st-century skills and digital skills: A systematic literature review. *Computers in Human Behavior*, 72, s. 577-588. doi:10.1016/j.chb.2017.03.010 Hentet fra <https://dx.doi.org/10.1016/j.chb.2017.03.010>

Widerberg, K. (2001). *Historien om et kvalitativt forskningsprosjekt : en alternativ lærebok* (K. Bolstad, Overs.). Oslo: Universitetsforl.

Wing, J. M. (2006). Computational Thinking. *Communications of the ACM*, 49(3), s. 33-35.

VEDLEGG I – OPPGAVER TIL PILOT

Scratch!



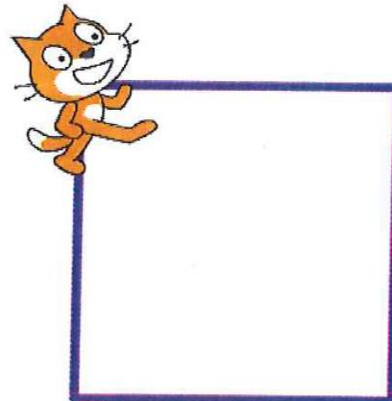
Oppgave 1: Utforsk Scratch

Bruk 5 minutt på å utforske ulike funksjoner i Scratch. Gi figuren et navn og legg til en ekstra bakgrunn.

Oppgave 2: Forflytning!

Kan dere få figuren til å gå:

- I en firkant
- Skrått (som et kryss)
- Til kanten av bildet



Oppgave 3: Bytt bakgrunn

Kan dere få bakgrunnen til å skifte til en ny bakgrunn når figuren kommer til kanten?

VEDLEGG II – OPPGAVER TIL OPPLEGGET



Figurer i Scratch!

I denne oppgaven skal dere lage ulike figurer ved hjelp av programmering i Scratch. Bruk funksjonen «Penn» når dere programmerer figurene slik at de blir synlige.



Dette er en samarbeidsoppgave der det er viktig å snakke sammen under arbeidet. Lytt til den andre sine forslag, forklar hva du har tenkt, og kom med forslag til ulike måter å løse oppgaven på. Kanskje det finnes mer enn 1 måte å programmere pennen på?

Pennen skal programmeres slik at de geometriske figurene tegnes i 1 omgang. Bruk kommandoen «når ___ trykkes» for å starte hver figur.

Programmer de geometriske figurene:

- Firkant
- Femkant
- En sirkel
- Stjerne
- Hus



VEDLEGG III – SAMTYKKESKJEMA BARN

Vil du delta i forskningsprosjekt om programmering i Scratch?

Hei! Vi heter Tonje Lindberg og Oda Åbø Pettersen og kommer fra Høgskolen på Vestlandet. Vi ønsker å undersøke hvordan dere arbeider med programmering i Scratch. Da trenger vi å filme noen av dere og ta skjermopptak av datamaskinen. Dette gjør vi fordi vi skal skrive en masteroppgave om programmering og matematikk.

Du kan velge om vi skal få lov til å filme deg. Dersom du ønsker å være med vil det kun være vi som jobber med prosjektet som får se og høre opptakene av deg. Dette er helt frivillig, og du kan ombestemme deg når som helst. Ingen vil kunne gjenkjenne deg når de leser masteroppgavene våre.

Hvis du lurere på noe, ta gjerne kontakt:

- Oda Åbø Pettersen på tlf.: 46657720 eller på epost: 145522@stud.hvl.no
- Tonje Lindberg på tlf.: 93417083 eller på epost: 145057@stud.hvl.no



Samtykkeerklæring forskningsprosjektet

Jeg har mottatt og forstått informasjon om prosjektet og fått anledning til å stille spørsmål. Jeg synes det er greit at jeg:

- deltar i videoopptak

Jeg synes det er greit at mine opplysninger behandles frem til prosjektet er avsluttet, 31.12.2021.

(Min signatur (elev), dato)

Samtykkeerklæring for bruk av videoer:

- Jeg synes det er greit at videosnutter der jeg er med kan vises i presentasjoner og undervisning.

(Min signatur (elev), dato)

VEDLEGG IV – SAMTYKKESKJEMA FORESATTE

Vil du delta i forskningsprosjektet «Argumentasjon og kritisk matematikkundervisning i flerspråklige klasserom»?

Dette er et spørsmål om å delta i et forskningsprosjekt om argumentasjon og kritisk tenkning i matematikkundervisning i flerspråklige klasserom. I dette skrivet informerer vi kort om innholdet i prosjektet og hva deltakelse innebærer.

Bakgrunn og formål

Prosjektet handler om å fremme lærerstudenters kompetanse i å legge til rette for argumentasjon og kritisk matematikkundervisning for elever i flerspråklige klasserom på barnetrinnet. Dette kan være å kritisk kunne vurdere matematikkforklaringer og å se matematikkens rolle i argumentasjon om aktuelle samfunnsproblemer. Skolene som er med i prosjektet er partnerskoler eller praksisskoler som allerede er en del av et samarbeid mellom Bergen kommune og Høgskulen på Vestlandet (HVL). Prosjektet varer i fire år og er et forskningssamarbeid mellom lærere og elever ved partnerskoler og tilsatte og studenter ved matematikklærerutdanningen ved HVL.

Som en del av dette forskningsprosjektet ønsker to masterstudenter (Oda Åbø Pettersen og Tonje Lindberg) å samle inn datamateriale der elever arbeider med programmering i Scratch. Målet er å få innsikt i hvordan matematisk argumentasjon og programmering kan påvirke hverandre.

Hvem er ansvarlig for forskningsprosjektet?

Høgskolen på Vestlandet er ansvarlig for prosjektet, og det er ledet av Professor Tamsin Meaney. Prosjektet gjennomføres i samarbeid med Bergen Kommune, og det er støttet av Norges forskningsråd.

Hvorfor får du spørsmål om å delta?

Vi spør om å la ditt/deres barn få delta i prosjektet fordi klassen barnet går i har drevet med Scratch tidligere og er dermed kjent med programmering fra før av.

Hva innebærer det for deg å delta?

Det vil bli gjennomført undervisning der elevene arbeider i par/grupper med programmet Scratch. Det er særlig fokus på matematisk argumentasjon der elevene oppfordres til å reflektere, diskutere og dele kunnskap. Deltagelse innebærer at noe av undervisningen barnet deltar i blir observert og filmet. Alle elever får tilbud om samme undervisningen, men kun de som samtykker, blir observert og filmet.

Det er frivillig å delta

Det er frivillig å delta i studien, og du/dere kan uten grunngeving når som helst trekke ditt/deres samtykke. Hvis du/dere trekker barnet fra prosjektet vil alle opplysninger om barnet bli anonymisert. Det vil ikke få negative konsekvenser hvis du/dere ikke ønsker at barnet skal delta, eller senere velger å trekke ditt/deres barn fra prosjektet.

Ditt/deres barns personvern – hvordan vi oppbevarer og bruker barnets opplysninger

Alle personopplysninger blir behandlet konfidensielt og personidentifiserbart materiale lagres på HVL sin forskningsserver, sikret med brukernavn og passord. Kun deltakere i prosjektgruppen og eventuelt transkriberingsfirma har tilgang til materialet. Deltakere vil ikke kunne bli identifisert i publikasjoner.

Prosjektet skal avsluttes 31.12.2023. Etter denne dato vil alle personidentifiserende data slettes og materialet vil ikke lengre være lagret på HVL sin forskningsserver. Videre bruk av dataene blir i presentasjoner, undervisning, eventuelle oppfølgingsstudier og senere forskning basert på transkribert og anonymisert materiale.

Dine/deres rettigheter

Så lenge ditt barn kan identifiseres i datamaterialet, har du/dere rett til:

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

Hva gir oss rett til å behandle personopplysninger om ditt/deres barn?

Vi behandler opplysninger om ditt/deres barn basert på ditt/deres samtykke. På oppdrag fra HVL har Norsk senter for forskningsdata (NSD) vurdert at behandlingen av personopplysninger i dette prosjektet er i samsvar med personvernregelverket.

Hvor kan jeg finne ut mer?

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

- Prosjektleder Tamsin Meaney på tlf.: 55 58 55 69 eller epost: Tamsin.Jillian.Meaney@hvl.no
- Masterstudentene:
 - Oda Åbø Pettersen på tlf.: 46657720 eller epost: 145522@stud.hvl.no
 - Tonje Lindberg på tlf.: 93417083 eller epost: 145057@stud.hvl.no
- HVL sitt personvernombud: Advokat Halfdan Mellbye, personvernombud@hvl.no
- NSD – Norsk senter for forskningsdata AS, på epost: personverntjenester@nsd.no eller telefon: 55 58 21 17.



Samtykkeerklæring forskningsprosjektet

Jeg/vi har mottatt og forstått informasjon om prosjektet «Argumentasjon og kritisk matematikkundervisning i flerspråklige klasserom» og fått anledning til å stille spørsmål. Jeg/vi samtykker til at barnet mitt/vårt kan:

- delta i videopptak
- delta i lydopptak

Jeg/vi samtykker til at mitt/vårt barns opplysninger behandles frem til prosjektet er avsluttet, 31.12.2023.

(Signert av prosjektdeltakers foresatte, dato)

Samtykkeerklæring for bruk av videoer:

- Jeg samtykker i at videosnutter der mitt/vårt barn er med kan vises i presentasjoner og undervisning.

(Signert av prosjektdeltakers foresatte, dato)

VEDLEGG V – TEGN OG KODER I TRANSKRIPSJONER

TEGN I TRANSKRIPSJONENE

V - Elev til venstre

H - Elev til høyre

S: Student

[...] - en kloss fra Scratch

i [...] - For et tall i en kloss som ikke er endret enda.

---(...)--- - Det skjer ting jeg har valgt å ikke direkte transkribere, men heller kort beskriver.

PV – Programmeringsvindu

HV – handlingsvindu

SM - Skriptmeny

Tall som står i klosser fra før: [gå # steg] # = 10, [vend # grader] # = 15

KODER OG TID I VEDLEGG AV TRANSKRIPSJONER:

Vedlegg VI – Gruppe 1:

Analyse:

K – Konseptuell forståelse → Antall: 23

B – Beregning → Antall: 19

S – Strategisk kompetanse → Antall: 33

R – Resonnering → Antall: 24

E – Engasjement → Antall: 11

[K B S R] – 18

Tid: 7:14 min

Vedlegg VII – Gruppe 2:

Analyse:

K – Konseptuell forståelse → Antall: 78

B – Beregning → Antall: 59

S – Strategisk kompetanse → Antall: 87

R – Resonnering → Antall: 72

E – Engasjement → Antall: 7

[K B S R] – 52

Tid: 11:07 min

Vedlegg VIII – Gruppe 3:

Analyse:

K – Konseptuell forståelse → Antall: 69

B – Beregning → Antall: 69

S – Strategisk kompetanse → Antall: 84

R – Resonnering → Antall: 70

E – Engasjement → Antall: 10

[K B S R] – 58

Tid: 10:15 min

Vedlegg IX – Gruppe 4:

Analyse:

K – Konseptuell forståelse → Antall: 82

B – Beregning → Antall: 76

S – Strategisk kompetanse → Antall: 89

R – Resonnering → Antall: 73

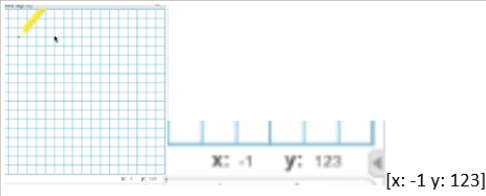
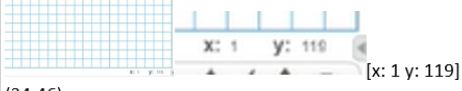

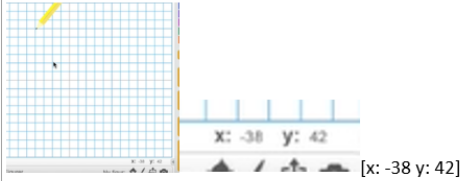


E – Engasjement → Antall: 22

[K B S R] – 69

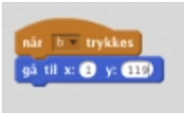
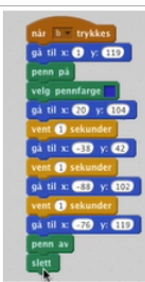
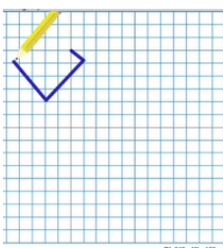
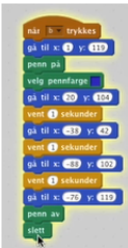
Tid: 13:37 min

VEDLEGG VI – TRANSKRIPSJON – FEMKANT, GRUPPE 1


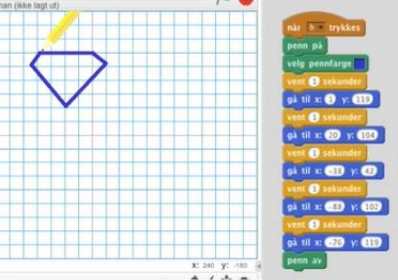
Vedlegg VI-a

| Koder | Tid | Samtalen | Bemerkninger | I Scratch |
|---|---|--|--|--|
| S K B S R K B S R K B S R | 34:10 34:20 | V: ok, la oss bare ta en og en ... H: Jeg tipper dette kommer til å ta like lang tid som sist gang. V: Vi begynner her ... (V flytter på pennfigur i PV) V: Der. Da skriver vi koordinatene ... (finder frem blyant og ark.) V: her er minus en ... minus en ... en, to, tre. Det er koordinatene. (V flytter musepeker i PV) H: [gjesper, trekker genseren opp til ansiktet, hvisker noe for seg selv] V: [mumler noe lavt – klarer ikke høre] (V flytter musepeker i PV) | Begynner på femkant. V styrer PC V ser på skjerm mens han skriver på ark. H følger med. |  (34:20)  (34:46)  |
| K B S R R S K B S R K B S R K B S R K B S R | 34:40 34:52 35:05 35:17 35:39 | V: okei, der. En ... H: og en, nitten V: ... en, en, nitten [skriver samtidig på ark] V: Da går vi ned lenger her ... (V flytter på musepeker i PV) V: og det er ... [skriver på ark] to ... tjue ... en, null, fire ... Og så går vi ned sånn ... (flytter på musepeker i PV) V: Så var det sånn ... [tegner på ark] minus tre, åtte, førtito ... (V flytter på musepeker i PV) V: Og så opp her ... opp her er [tegner på ark samtidig] .. minus åtte ... der ... og en, null, to (V går tilbake til PC. Flytter musepeker i PV. Skriver videre på ark) V: og opp til blyanten igjen er det minus syv, seks, en, en, ni | |  (35:05)  (35:17)  |
| K B S R K B S R | 35:59 | H: åh ... [puster ut, tar seg til ansiktet] Da har vi litt igjen ... V: okei ... der har vi koordinatene ... [streker fort på ark]. Der ... en, to, tre, fire, fem. Okei ... [peker samtidig med blyant på ark] (V tilbake på PC) (transkripsjon fortsetter i neste rad) | | |
| K B S R | 36:00 | V: Vi begynner med ... la oss se [ser ned på arket han har skrevet på] med den første ... som er en, nitten ... H: [gjesper. Ser bort på arket til V. trykker på piltaster] V: Hva gjorde du? H: Bare trykker på pilene ... V: hm hm.. [«humrer»] | | |

Vedlegg VI-b

| | | | |
|---|--|---|--|
| <p>E</p> <p>K S R</p> <p>S</p> <p>S</p> <p>K B S R</p> <p>S</p> <p>K B S R</p> <p>K B S R</p> <p>K B S R</p> <p>S</p> | <p>H: Du vet at det filmes, det vi sier? V: Jeg vet det ... Det er derfor jeg snakker litt høyt M** (navn til H) og ikke tenker inni meg. Åpenbart partnership hæ? Hehe ... partner .. (ser bort på H og smiler. H smiler tilbake) H: [mumler noe jeg ikke klarer å høre] V: Okei ... okei ... da går vi (setter inn kloss) okei, da begynner vi ... H: [tegner] V: [mumler noe. Puster tungt ut] okei ... (ser på arket) en ...en ... (skriver inn i kloss) H: Det går jo an å bare tegne et hus ...? (V fortsetter å skrive inn i kloss) V: aa...[stille, og drar på ordet]... en ... En, en, ni ... og så tar vi penn. Penn ... og så ... skal vi bare ta random farge? Og så ... penn på ---[teknisk – legger inn pennfunksjon]--- V: så må vi ta koordinatene igjen ... (legger inn kloss) ... en til en ... V: Okei, da har vi den her borte ... (peker med blyant på ark. H følger med). Og neste som var to ... to, null ... tjue ... null fire .. okei. (V skriver inn i kloss) V: (tegner på ark samtidig) okei, okei ... den, den ... den neste er ... minus trettiåtte. Vi må faktisk først ... styring (trykker på kategorien «styring» i skriptmeny) (legger inn kloss) V: Vent et sekund ... koordinater ... (legger inn klosser) (V skriver inn tall og snakker samtidig)</p> | <p>V setter inn kloss [gå til x:# y:#]. H sitter og tegner/skriver på ark ved siden av. V skriver inn i kloss [gå til x: 1 y: 119] V styrer pc. H følger med på skjerm. Ser noen ganger bort på andre ting i rommet. V legger inn [gå til x:# y:#] Skriver inn x:#=20 y:#=104 V legger inn [vent 1 sekunder] V legger inn x3 [gå til x:# y:#] i HV, men ikke på blokk Skriver inn koordinater i klossene og setter de inn.</p> |  <p>(36:55)</p> |
| <p>S R</p> <p>S</p> <p>E</p> <p>S</p> <p>K S R E</p> <p>E</p> <p>E</p> <p>E</p> <p>K B S R</p> <p>S</p> | <p>V: Okei, første er ... minus tre, åtte ... til førtito ... (V ser på arket) V: Minus åtte, åtte til ... to ... hundrede og to ... minus 76, en, en, ni... vent ett sekund ... (V legger inn klosser og skriver inn tall) H: Nå får vi bare håpe denne her fungerer ... og så må vi ha penn av ... V: penn... Penn av ... og slett H: og så må du trykke på b-en V: oog [drar på ordet] b. ---(tester skript)--- 40:02 H&V: hh [begge puster dypt inn. V tar hendene til ansiktet. Begge følger med på skjermen] ---(skriptet kjører, men blir slettet rett etterpå)--- H: Nå har du slett ... (peker på skjermen) Du må jo ikke ha sletten innpå... V: okei, okei, okei ... H: det var nære ... V: b, b, b ... ---(tester skript igjen)--- H: bam, dam, dum ... [sier det samtidig som blyanten tegner i PV] [begge puster dypt inn mens figuren tegnes i PV] H: ånei! [løfter armene over hodet] (V smiler, puster dypt inn og ut) Stud: Oi! H: én vegg ... V: okei, la meg bare få trykke igjen ... b ---(tester samme skript igjen)---</p> | <p>Legger også inn [vent 1 sekunder] mellom hver koordinat (se bilde for tiden 39:59). Legger også inn [penn av] og [slett]</p> <p>får samme figur som i 40:02, bare at den ikke forsvinner</p> |  <p>(39:59)</p>   <p>(gul kant rundt betyr at skriptet kjører) (40:02)</p> |

Vedlegg VI-c


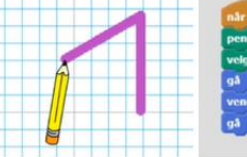
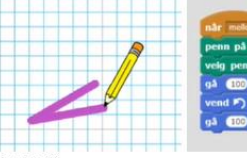
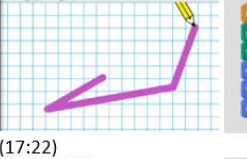
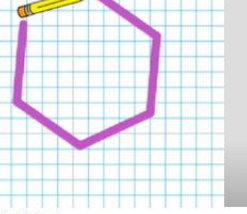
| | | | |
|---|--|--|---|
| <p>K B S R E</p> <p>S</p> <p>K B S R E</p> <p>E</p> <p>K B S E</p> <p>E</p> <p>K R</p> | <p>V: åå ... Vi er bare en linje vekke fra ... H: Det er akkurat ... V: (snakker samtidig som stud.) Hvorfor har vi på denne her koordinaten her? (V flytter på deler av blokk) Stud: det klarer dere jo. En linje til? V: ja, men det er på grunn av denne koordinaten ... f**, jeg må få denne ene koordinaten vekk... 40:46 ---(teknisk – flytter og legger til klosser)--- 41:05 ---(ferdig med endringer i blokk)--- H: ok, plis, fungerer! B. V: b! 41:09 ---(tester skript)--- V: Ja! Wohoo! [reiser hendene i været og smiler] H: [smiler og slår i bordet] Stud.: Den var skikkelig fin! H: Bra diamant!</p> | <p>Får samme figur som de to forrige gangene. Pennen tegner oppå strekene som allerede er i PV fra forrige skript-test.</p> <p>V flytter øverste kloss med koordinat lengre ned – under de grønne penn-klossene, og letter inn en ekstra [vent 1 sekunder] i mellom (se bilde for tiden ...)</p> |  <p>(41:05)</p>  <p>(41:09)</p> |
|---|--|--|---|

VEDLEGG VII – TRANSKRIPSJON – FEMKANT, GRUPPE 2

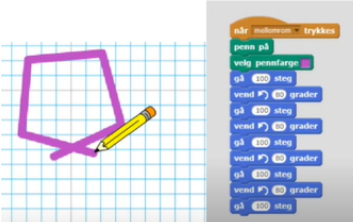
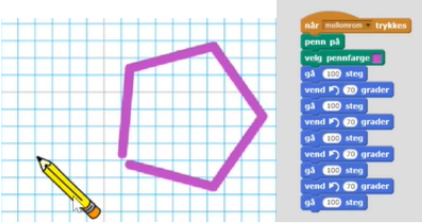
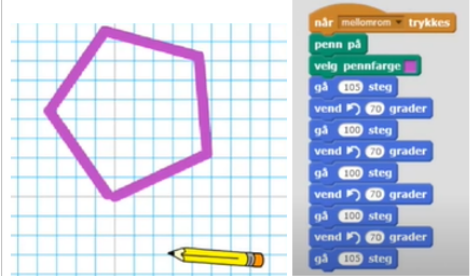
Vedlegg VII-a

| Koder | Tid | Samtalen | Bemerkninger | I Scratch |
|---|--|---|---|----------------------------|
| K B S R K B S R | 11:15 | H: Og så må vi ha gå et eller annet steg. V: Det er jo femkant da ... Se her. Vent da ... [snakker lavt] en ... | V: Begynner å tegne på ark | I Scratch |
| K B S R | 11:19 | H: Vi kan ta 100 steg nå også V: [hvisker noe for seg selv] H: Vi kan ha mellomrom ... det kan vi ha. | V tegner på ark og H jobber i Scratch. Begge snakker lavt for seg selv. | H: Setter inn [gå # steg] |
| K B S R B S R | 11:34 11:36 | V: en, to, tre, fire, fem, seks ... (tegner) sekskant H: Gå 10 steg ... Gå 100 steg | H: Tegner fortsatt på ark. Snakker lavt for seg selv mens han tegner. | Setter inn [gå 100 steg] |
| B S K B S R | 11:40 | V: [lavt for seg selv] en, to, tre, fire, fem, seks ... nei. H: [snakker høyt samtidig som V snakker] og så må vi vende et eller annet | V og H snakker samtidig | Setter inn [vend # grader] |
| R K S R K B S R K B S R K B S R | 11:48 | V: Er det femkant? H: ja V: den H: Nei, det er ... vent litt ... få se V: Jo, det er en, to, tre, fire, fem H: En, to, tre, fire, fem, seks. Det er en sekskant H: Nei, var det en femkant vi skulle lage? ... Men kanskje det er en annen måte å tegne det på | V henvender seg til H Snakker sammen tegner på arket til H | |
| K K S R K R K B S R K S R K B S R | 12:03 | V: sånn! H: Men liksom ... V: det går jo an H: [puster ut] Det går jo ikke V: Jo, se ... den | tegner på arket. H følger med V peker på arket. H tegner videre H tegner | |
| K B S R K S R K S R | 12:20 | H: Ja, men da blir det jo et hus ... [lavt] en femkant V: Nei ... Ja, men sånn [Peker på arket] H: Ja, men det er jo et hus V: Nei, det er ikke et hus. Ser det ut som et hus? | V tar over tegningen V peker på arket | |
| K B S R K B S R K B S R K S R K S R K B S R K B S R S R S R S K S R K S R K B S R K S K S | 12:38 12:48 13:10 13:34 | H: Liksom ... Femkant og hus er jo det samme. [henvender seg til V] V: Ja, men da lager vi bare et hus ... og en femkant H: to like V: så gjør vi sånn, sånn, sånn, sånn, sånn H: Men det blir jo ikke rett. Da blir det jo ... Alle vinklene må jo være like. Må ikke de? V: Nei ... De må jo ikke det ... Må de? H: Må ikke de? V: Må alle vinkler være like? S: Hva er en femkant? V: En ting med fem kanter. S: Er det noe mer som er viktig? Eller er det bare det? V: Kanskje at alle vinklene er like? ... Eller? H: Jeg vet ikke S: Dere får ta det dere tenker er riktig da V: Ja, men ... kom igjen da ... Ja, men jeg glemte hva det var S: Dere får diskutere det da H: Det går jo ikke an! S: Hva mener du? H: Det går jo ikke an å få alle like? V: Like? H: Her tror jeg ... Nå er den bra. Nå er alle like. S: Hva er det som er viktig med en femkant? H: At den har fem kanter [ler] Sånn! H: Sånn! Nå er alle like ... S: Jeg vil ikke si hva som er rett og galt. Dere får gjøre det dere tenker er riktig. V: Åja ... H: Sånn. Det er riktig. | H henvender seg til V V tenger V henvender seg til student Både H og V tegner på ark Stud. Henvender seg til H V tegner fortsatt. H viser tegning til V. | |

Vedlegg VII-b

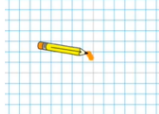



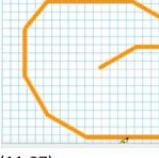
| | | | | |
|--|--|---|--|---|
| <p>K S R</p> <p>S</p> | | <p>S: Hvis du mener det du tenker er riktig, så kan du jo forklare ... H: Det er fordi da kan det se ut som det samme fra alle sidene S: Ok, hva tenker du da? V: Samme ...? S: Ok, du tenker det samme? Ok V: Kan vi ikke bare ...? H: Ja, men det blir ... V: Hvordan skal vi gjøre det?</p> | <p>V Fortsetter å tegne</p> <p>Henvender seg til H</p> <p>Svarer til S</p> <p>S henvender seg til V</p> | |
| <p>K B S</p> <p>K B S R</p> <p>S</p> <p>K B S R</p> <p>K B S R E</p> <p>K B S R</p> <p>K B S R</p> <p>S R</p> <p>S</p> <p>B S</p> | <p>14:12</p> <p>14:13</p> <p>14:38</p> | <p>V: Vent da ... hvor mange grader er en femkant? ... (trykker i scratch) Skal vi se ... ---(tester skript)--- H: Jeg aner ikke. En firkant det er jo 90 grader ... En femkant ... det aner jeg ikke ... V: mellomrom ... H: Då e det i hvert fall over 90 grader V: Går det ikke an å trykke mellomrom? H: Bare trykk på grønt flagg --- (jobber med det tekniske) --- H: Sånn. Jah ... Da må vi ha over 90 grader tror jeg H: Vi må ha over 90 grader tror jeg, siden alle liksom har stumpe vinkler V: Ja, ok ... Da blir det ... 120? H: Ja, kanskje det? Jeg vet ikke ... V: Vi må bare prøve da H: Gå 100 steg --- (prøver å lage kopi av skript, får ikke til) --- V: Vi gidder ikke å kopiere, vi bare gjør sånn som vi har gjort.</p> | <p>V begynner i Scratch</p> <p>V trykker i Scratch</p> <p>H trykker på grønt flagg</p> | <p>Trykker på blokken</p>  <p>(14:13)</p> <p>Skriver inn [vend 120 grader]</p> <p>Setter inn [gå 100 steg]</p> <p>Trykker på blokken og peker med pil på figur</p> |
| <p>K B S R</p> <p>K</p> <p>K B S R</p> <p>K B S R</p> <p>S</p> <p>K B S R</p> <p>K B S R</p> <p>K B S R</p> <p>K B S R</p> <p>K B S R</p> <p>K B S R</p> <p>S</p> <p>B S</p> <p>S</p> <p>K B S R</p> | <p>16:49</p> <p>16:50</p> <p>17:09</p> <p>17:20</p> <p>17:22</p> <p>17:49</p> <p>18:37</p> | <p>---(tester skript)--- H: Jeg tror det er litt mye ... Nei, eh ... litt lite. Altså, jeg tror vi må ha mer enn 120 grader V: Mhm H: Kanskje 160 ... (endrer skript) ... skal vi se ---(tester skript)--- H: Nei, da går den jo bare innover. Da må vi jo ta mindre da ... V: hm? Hvordan? H: Skal vi se ... (endrer til 60 grader) Må bare prøve noe ... ---(tester skript)--- V: Jeg tror det er 60 grader vi skal ja H: Ja, vi prøver da ... Gå 100 steg, vend ... (setter inn klosser i blokken) H: Vi må ha fem sånne gå ... en, to, tre, fiire ... V: og så ... H: fem ... V: og så en til ... H: Nei, ikke en til V: nei, ikke fem H: Jo! (H fortsetter å sette inn klosser) V: Så må vi fikse alle de. H: Ja ... Det kan du gjøre V: Var det 60 grader? H: Vi prøver ... --- (skriver inn i klosser i blokk) --- H: Ok, da prøver vi ... --- (tester skript) --- H: Ja? V: Vi mangler én ...se? H: Ja, jeg skjønnte ikke helt ...</p> | <p>Tester skript. Får en enda spissere vinkel.</p> <p>H trykker på blokk. Får en mer stump vinkel.</p> <p>Utvider skript</p> <p>V skriver inn i HV på PC</p> <p>Tester skript</p> <p>V peker på figur Setter på flere klosser --> [gå 100 steg] +[vend 60 grader] x 5 (annenhver)</p> |  <p>(16:49)</p>  <p>(17:09)</p>  <p>(17:22)</p>  <p>(18:37)</p> |

Vedlegg VII-c

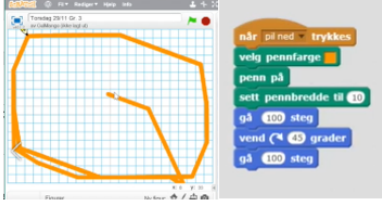
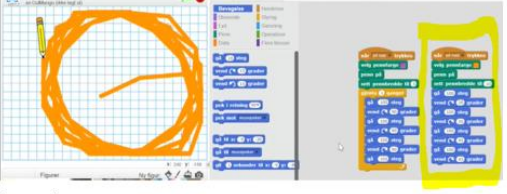
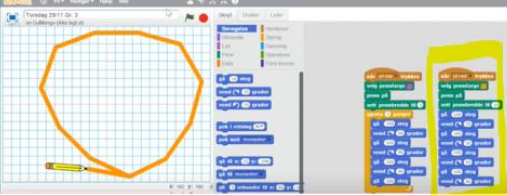
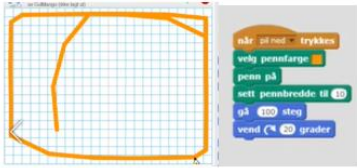
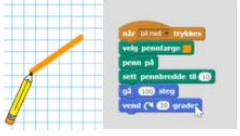
| | | | |
|---|--|---|---|
| <p>S</p> <p>K B S R</p> <p>K B S R</p> <p>K B S R</p> <p>K</p> <p>K B S R</p> <p>K</p> <p>K B S R</p> <p>K S</p> <p>K B S R</p> <p>K B S R</p> <p>K B S E</p> <p>E</p> <p>K S E</p> <p>K B S R E</p> <p>K S R</p> <p>K S R E</p> <p>K B S R</p> | <p>18:50 V: Nei ...?</p> <p>H: Men trykk på grønt flagg (peker på skjermen) --(tester igjen - får samme figur som på 18:37)--</p> <p>H: Ta Gå 100 steg</p> <p>18:52 V: Nei, da blir det en sekskant</p> <p>H: Åja ... nei, men da må vi ... da kan vi ikke ha 60 grader. Da må vi ha mindre. Nei ... mer</p> <p>V: Mhm</p> <p>H: Der var jo derfor han ... 70 grader kanskje?</p> <p>Eller 80? Skal vi prøve 80?</p> <p>V: Mhm?</p> <p>H: 80 grader?</p> <p>--- (endrer i skript - prøver igjen) ---</p> <p>19:29 [ler og klør seg i hodet]</p> <p>H: Ja, da må vi jo ta ...</p> <p>V: vent da ... Er ikke det sekskant? Nei, femkant?</p> <p>H: Jo, det er jo det vi skal ha? ... 70 grader da? 70 er sikker rett. Siden 60 ikke er ... på en måte ikke er ... ja, da prøver vi 70.</p> <p>V: Ja, det kan hende ... hvis det er 70 så blir jeg glad!</p> <p>H: ja, jeg også!</p> <p>--- (tester skript igjen) ---</p> <p>19:58 H: Yeeey! Nei, var det? [ser på figur i PV]</p> <p>--- (student kommer og justerer utstyr) ---</p> <p>V: En, to, tre, fire, fem ... Jah! Det var det</p> <p>H: Ja, men den er jo ikke helt inntil? Er den?</p> <p>V: jo, jo, jo, jo ... du ser ikke det</p> <p>H: [henvender seg til stud.] Er den godkjent?</p> <p>V: [ser bort på student] Den er godkjent?!</p> <p>H: siden, se då ... den er ikke helt inntil [viser på figur]</p> | <p>Trykker på grønt flagg og tester skript igjen. Får samme figur.</p> <p>H endrer i skript</p> <p>H endrer skript - tester</p> <p>Får femkant med "hull i"</p> <p>Ser og peker på skjerm</p> | <p>Endrer til [vend 80 grader] i alle fem</p> <p>Får figuren:</p>  <p>(19:29)</p> <p>Setter til [vend 70 grader] x 5</p>  <p>(19:58)</p> |
| <p>K S R</p> <p>K B S R</p> <p>K B S R</p> <p>K B S R</p> <p>K B S R</p> <p>K S</p> <p>K E</p> <p>K B S R</p> <p>K</p> <p>K B S R</p> <p>K B S R</p> | <p>20:44 S: ååå ... dere er jo veldig nærme da?</p> <p>V: Ja, men hvordan skal vi gjøre det da?</p> <p>S: Hvorfor er det er lite hull der da? Kan vi tenke oss det?</p> <p>V: vet ikke ...</p> <p>H: Eh ... jeg vet ikke ... Kanskje vi skal gå 101 steg?</p> <p>S: Prøv og slett den og tegn den en gang til da, og se om det bare ...</p> <p>H: ok</p> <p>--- (tester skript - får fortsatt samme figur) ---</p> <p>20:49 S: Hm ... pitte litt som mangler?</p> <p>V: Men hvordan skal vi gjøre det? Det går jo ikke</p> <p>S: Kan det være noe med tallene?</p> <p>V: ahr ... kan prøve 102. nei, 105. på den første og den andre. Nei, første og siste</p> <p>H: nei, kanskje det er noe i gradene eller?</p> <p>V: Nei, det er ikke noe i gradene. [hvisker] vi har prøvd de for mye</p> <p>H: Men kanskje noen som er lengre enn andre?</p> <p>--- (Teknisk – få frem igjen pennfigur i PV) ---</p> <p>22:13 V: Se ... Hvis ... (trykker på blokk)</p> <p>H: ok, greit</p> <p>22:15 --- (tester skript)---</p> <p>V: Jah!</p> <p>H: jeg får se. Ja, men nå er jo ... Er noen sider lengre enn andre nå?</p> <p>V: Neeei ...</p> <p>H: 105 [ler litt]</p> <p>22:22 V: Ja, det er nok. Nå skal vi lage sirkel.</p> | <p>Endrer i skript</p> <p>Tester skript igjen.</p> <p>Får en ny femkant uten "hull"</p> | <p>Endrer til [gå 105 steg]</p> <p>Har endret første og siste [gå # steg] til [gå 105 steg]</p>  <p>(22:15)</p> |

VEDLEGG VIII – TRANSKRIPSJON – FEMKANT, GRUPPE 3

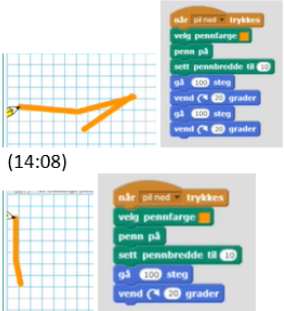

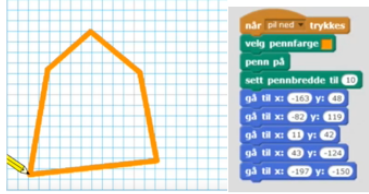
Vedlegg VIII-a

| Koder | Tid | Samtalen | Bemerkninger | I Scratch |
|---|---------------------------------|---|--|---|
| K B S R K B S R S S | 07:30 07:45 | H: Ok, femkant. Da må vi gjøre det fem ganger. V: nei ... Vi kan ikke ha 90. Vi må ha 45, må vi ikke? H: Vi begynner med "når mellomrom trykkes" (en kloss) V: Da må vi ha pil ned ---(Snakker om og fikser tekniske ting i Scratch)--- | Begynner på femkant V peker på blokk snakker om hvilken tast de skal gi blokken som utløsende kommando, og setter inn pennfunksjon i blokken | I Scratch |
| K B S R K B S R K B S R K B S R K B S R | 09:12 09:44 | H: Ok ... Vi kan ha gå til punkt, gå til punkt ... For da er det litt lettere. Da kan vi gjøre sånn ... [peker på rutenett i Scratch] V: Ja, men da må vi trykke for hver gang ... Da må vi endre på den. H: ... sånn ... Da lager vi hus. Femkant er hus. V: [Henvender seg til Stud.] Må vi lage én femkant og ett hus? Det er jo det samme H: Man kan jo gjøre det til det samme ... [ler litt] S: Løs det sånn som dere tenker er riktig. V: Okei. Eee.. Skript, bevegelse ... ---(Finner "bevegelse" i skriptmeny)--- | Flytter litt frem og tilbake på penn i programmeringsvinduet. V ser bort på H. H henvender seg til Stud. Finner «Bevegelse i skriptmenyen | |
| K B S R K B S R K B S R K B S R | 09:52 10:00 10:11 | H: Vi kan begynne med gå 10. Gå 50 steg? V: Nei, 100 H: nei, det blir ... [setter inn kloss - gå] ---(tester skript)--- H: Ok, greit, 100 V: [Ler] Se hvor lite den ... | V Setter inn funksjonen [gå 10 steg]. Endrer til 100 |  (10:00) |
| K B S R K B S R K B S R K B S R K S R | 10:18 10:26 10:34 | [setter inn kloss - vend] H: Vend. Nå skal jeg bare sjekke. Sånn ... V: Nei, du må trykke på pil ned H: å ja ... ---(tester skript)--- H: trykk en gang til da. Hold inne ... ---(tester skript med å trykke på det gjentatte ganger)--- H: Oii ... Var det en firkant? [ler litt] V: Nei, det var ikke en firkant. Vent, vi trenger jo ikke lage hus, på grunn av ... ---(V Holder inne kommandotast – tester skript)--- V: se da, se da ... En, to, tre, fire ... åå, det er seks. Aaa ... H: [ler litt] ja, men se da ... emm ... | V setter inn [vend # grader]. Trykker på pil ned (startkommandoen til blokk) gjentatte ganger. |  (10:18)  (10:26)  (10:34) |
| K B S R K B S R K S R K B S R K B S R | 10:46 | V: 100 steg. Gå ... H: Gå ... Vend. V: [hvisker] Husk at jeg skal trykke neste gang [fniser] H: Ja. Vend ... V: 305 [fniser] H: Nei ... H: Jeg skal bare fikse den der ned, og så skal vi bare teste ... V: Vi må bare trykke på pil ned. H: å ja | H styrer PC. Setter inn [gå # steg]. Ender på [vend # grader] til #=30 H henter frem penn i PV. Tester skript (trykker flere ganger på kommandotast-pil ned) Endrer til [vend 20 grader]. Trykker pil ned. |  (11:07) |

Vedlegg VIII-b

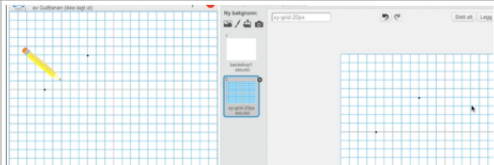
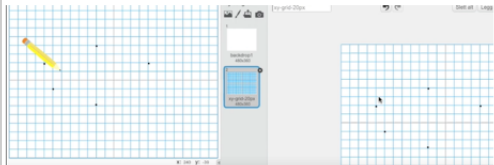
| | | | |
|--|---|---|---|
| <p>S R</p> <p>K B S R</p> <p>K B S R</p> <p>S</p> <p>K B S R E</p> <p>K B S R</p> <p>S</p> <p>K B S R</p> <p>K B S R</p> <p>K B S</p> <p>K B S R</p> | <p>11:07</p> <p>V: hold inne ---(tester skript – holder inne kommandotasten)--- V: Ok, det var litt mer. Emmm ... H: Ja, men det var.. Hvis vi tar 20. 20. S: Husk at vi vil at hele figuren skal bli tegnet med å bare trykke én gang. H: jaja, men vi bare sjekker.. Eh ... V: Vi bare tester. Em, ta 45 [peker på skript]. Jeg fikk veldig lyst til å prøve 45. H: 45? Men da snur du jo helt ... V: Prøv. H: Åja, gå 100 steg. V: og så tar du 45. H & V: 45 ... H: Og så må vi gjøre sånn ... ja, du kan ---(V tester skript – holder inne kommandotasten)--- V: eee ... [Begge ler]</p> <p>11:52</p> | <p>H endrer i skript. Ender til [gå 100 steg] (på andre slik kloss) og [vend 45 grader]</p> <p>H holder inne pil ned</p> <p>H trykker taster inn i Scratch. V tar over</p> |  <p>(11:52)</p> |
| <p>K B S R</p> <p>K B S</p> <p>K B S R</p> <p>K B S R</p> <p>K B S R</p> <p>B S R</p> <p>B S R</p> | <p>11:56</p> <p>H: ok, greit. Og så tar vi ... Ee.. Hvis vi ... Vent. Bare vent ... Ikke endre enda da. Men hvis vi bare går vend 15, gå ... V: Vend. H: Vend. Og så bare gjentar vi det 5... Det kan ikke være 45 grader. Det er jo forskjellig vinkel på alle. V: Nei.. H: Ja, men se. Hvis vi har vend 20 grader på denne. 20..og em..20, 45 ... Og så går vi 100 steg der.. [snakker litt lavt] V: 100.. H: ...100, 100...</p> | <p>H peker på skript. V trykker på [vend 45 grader]. H tar over PC.</p> <p>H setter inn [vend # grader] (#=15) + [gå # steg] + [vend # grader]</p> <p>V endrer [vend # grader] til #=20, neste [vend] til #=20, to neste #=45. Ender også så alle [gå # steg] har #=100</p> | |
| <p>S</p> <p>E</p> <p>K S R</p> <p>B S</p> <p>B S R</p> <p>K B S R</p> <p>K B S R</p> <p>K B S R</p> <p>R</p> <p>B S R</p> <p>S</p> <p>K B S R</p> <p>K B S E</p> | <p>13:00</p> <p>V: skal vi bare sjekke? H: Åja..</p> <p>13:05</p> <p>---(tester skript – holder inne kommandotast)--- V: oiisann ... H: eee ... Skal vi lage sirkel? V: Hvor mange blyanter er det? Vent ... Vi trykker stopp.</p> <p>13:15</p> <p>---(tester skript – trykker kommandotast flere ganger)--- H: Ok, greit. Det holder. V: den bare fortsetter. H: ok, da må vi endre på.. Men hvis vi først tar ... Jeg skal bare sjekke noe. [trykker i Scratch] Åja. Ee.. Okei? V: nei, nå gikk den bare rett opp. H: åja.. V: du må bare trykke en gang til. Trykk ned. ---(tester skript)--- H: åja, nei... se... V: Hvor mange runder har du nå? H: se. Sånn.. [trykker i Scratch]. Eller så gjør vi sånn... V: Okei, jeg begynner å hate den oransje fargen nå. H: da må vi bytte... Jeg liker den. [ler litt]. Se, sånn... ---(tester skript)--- H: Det er var jo bra ... Så hvis den hadde gått opp der, og så sånn.. 14:00 ---(Endrer skriptet, flytter penn til toppen av forrige strek og trykker på skriptet igjen)--- V: åja, wow.</p> | <p>Trykker og holder nede pil ned.</p> <p>Tester skript noen få ganger.</p> <p>V styrer PC</p> <p>Flytter frem og tilbake på noen klosser i skript</p> |  <p>(13:05)</p>  <p>(13:15)</p>  <p>(13:45)</p>  <p>(14:00)</p> |

Vedlegg VIII-c



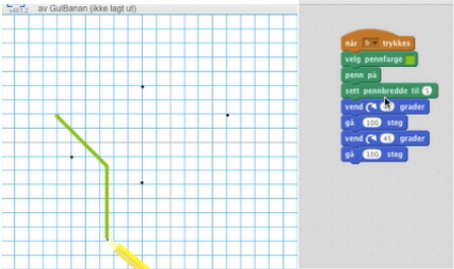
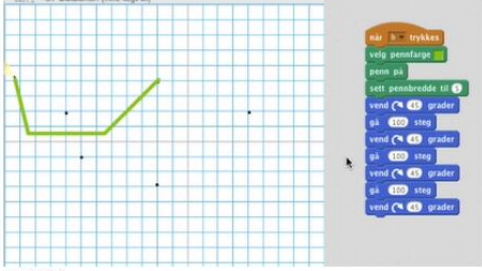
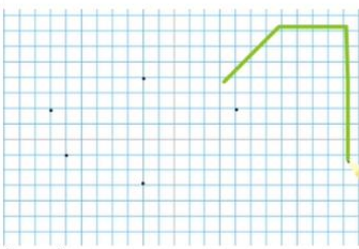
| | | | |
|--|---|---|---|
| <p>K B S R E K B S R K B S R S E B S R K S R K B S R K B S R K S S K B S R K B S R K B S R K B S R K B S R K B S R K B S</p> | <p>14:18 H: sånn.. Sånn ... [trykker på kommandotast] ---(tester skript)--- H: mmm.. Kom igjen! V: Eller så kan vi gå til punkt, gå til punkt, gå til punkt ... [fortsetter å gjenta mens V snakker] 14:25 H: jaha, det var jo det jeg mente i sted. V: [fortsetter å gjenta og peker i Scratch] H: Ja, det gjør vi. Ja, shhh! [V ler] H: Jeg bare fjerner dette. Så kan du ... V: [hvisker] Gå til punkt.. H: okei, greit ... ee.. V: Nå skal jeg prøve. Bevegelse.. H: gå ... V: Gå til ... H: mhm.. V: Da må vi bare.. Jeg må bare.. H: Okei, ja.. Og så må vi [peker i PV] ... Den. V: Den begynner der.. H: Ja, og så skal den opp ... [peker i PV] V: Og så skal den.. Nei, vi begynner her. Så.. Oi (penn spretter tilbake. Prøver igjen) H: Så skal den opp til dit. [peker i PV] V: Ja, okei, vi tar ... H: nei, vent ... 15:11 ---(prøver å flytte pennfiguren frem og tilbake og finne ut av hvor den skal gå for at den skal gå i femkant)---</p> | <p>H tar vekk skript de har laget til nå. V tar over PC V setter inn [gå til x:# y:#]. Flytter litt på blokk. H peker i PV, mens V styrer PC. Flytter frem og tilbake på pennfiguren i PV.</p> |  <p>(14:08)</p>  <p>(14:18)</p> |
| <p>K B S R K B S R</p> | <p>16:12 V: og så drar vi den opp der. H: Hva skal du nå? V: 163, 48.. Husk minus 163.. Åja, det står der.</p> | <p>V flytter pennfigur i PV (refererer til kloss i skriptmeny som har endret</p> | |
| <p>K B S R K B S R K B S R K B S R K K B S R K B S R</p> | <p>16:54 S: hvis dere trenger å tegne på ark, så kan dere det. H&V: ja.. H: og så tar vi dit.. V: Nå går den liksom sånn ... [illustrerer med bevegelser og lyder] H: og så går den til dit ... ish.. [flytter på pennfigur]. Du kan ta.. V: Nå har den gått sånn, sånn, sånn ... (peker i Scratch med musen. Beveger den i retning så det ligner starten på formen til et "hus") H: mhm.. V: Og så bort igjen dit. H: Nei, vent (flytter pennfigur til ca der den startet) ---(flytter litt frem og tilbake på klosser i Scartch)---</p> | <p>tall til akkurat den posisjonen pennfiguren står i koordinatsystemet i PV) H styrer nå PC Flytter pennen rundt i PV V overtar PC Flytter pennfiguren rundt i PV og drar bort funksjonene [gå til x:# y:#] etter som denne klossen endres etter koordinatene til pennfiguren. Der så bort i skriptet til figuren.</p> | |
| <p>K E K B S R E S R B S E K B S R E K B S R</p> | <p>17:45 ---(tester skript)--- H&V: Oii.. V: En, to, tre, fire.. [teller på figuren i PV]. Ja! Vi klarte det! V: [vender seg til Stud.] Er det en femkant? S: Ja, er det fem kanter da? H&V: Ja...[ler] H: det er et hus [ler]... nei...[ler] V: Ja, det er et hus. Noen hus er lit skjevere enn andre.. Så.. H: okei, hva skal vi lage nå?</p> | |  <p>(17:45)</p> |

VEDLEGG IX – TRANSKRIPSJON – FEMKANT, GRUPPE 4

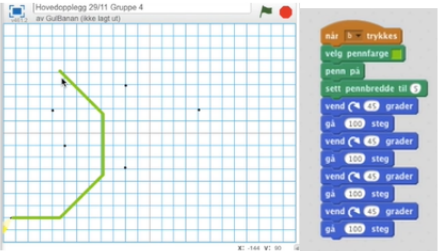
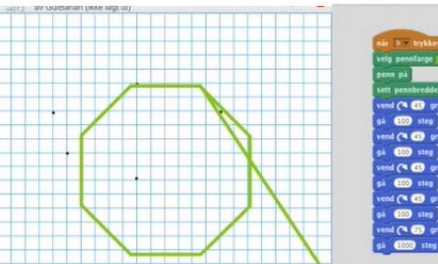
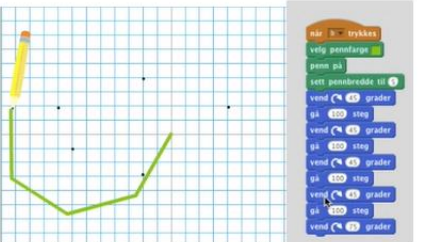
Vedlegg IX-a

| Koder | Tid | Samtalen | Bemerkninger | I Scratch |
|---|-------|--|---|--|
| K B S R | 9:40 | V: Femkant.. Ok, da må jeg bare se først. Se.. Dette blir ikke fint, men jaja. Jeg bare tegner mange streker ... Hvis vi sier at dette er rutenettet. H: mhm ...? | V tegner på arket mens H følger med | |
| K B S R | | V: Ok, så må femkanten ... hvis den står der, så må den ... [klør seg i hodet og ser på arket] øø, nei.. Sånn, sånn, sånn.. Den.. [tegner].. Nei.. [rabler over noe på arket]. Ok.. Hvordan lager vi en femkant? [begge ler litt] | | |
| K B S R | 10:26 | V: Hvis vi ... Ok, jeg har ingen idé.. (H spør stud. Om hjelp) V: em, på pennen.. Går det an liksom å tegne sånn at man vet hvor femkanten liksom skal være? Eller gå på en måte? Sånn at vi liksom kan sette sånne punkter? Går det an? S: Mmm, ja.. At den skal flytte seg sånn? V: Ee.. Ja, eller sånn at du bare kan vite hvordan den femkanten skal se ut (viser med musepeker i PV) S: Da kan dere kanskje gå inn i bakgrunnen og så sette noen sånne punkter her... ---(stud. Forklarer hvor punkter kan settes i bakgrunn)--- | H begynner å bevege på pennfigur i Scratch. H snur seg bak mot stud. Viser/forklarer med musepeker i Scratch | |
| K B S R | | V: Ee.. Ja, eller sånn at du bare kan vite hvordan den femkanten skal se ut (viser med musepeker i PV) S: Da kan dere kanskje gå inn i bakgrunnen og så sette noen sånne punkter her... ---(stud. Forklarer hvor punkter kan settes i bakgrunn)--- | | |
| K B S R K S K B S R K B S R K B S R | 11:18 | V: Ok, øøø ... H: Hvis vi.. Du vet den dere med koordinater? V: Mhm.. H: Hvis du tar koordinatene og så gir du den koordinater sånn at det blir til en femkant.. V: [hvisker] Nei, det blir jo ikke en femkant.. H: Jo, hvis vi gir [peker i PV i Scratch] sånn, og så gir vi den koordinater og så sånn, sånn, sånn, sånn og så tilbake dit. Hvis vi gir den akkurat det det skal være.. | V styrer PC og musen. H forklarer og peker på skjermen. |  (11:54) |
| K B S R | | V: Men det blir jo veldig stress å finne sånne koordinater.. Vi bare tegner den. ---(tegner prikker i bakgrunnen i Scratch)--- H: Hvordan ser en femkant ut? H: (Henvender seg til Stud.) Kan vi søke på hvordan en femkant ser ut? S: Tenker bare at så lenge den har fem kanter og fem hjørner så... | V setter prikker i bakgrunnen til PV |  (12:17) |
| S | | V: Åja, så går det fint? S: ... så er det en femkant. | | |
| K B S R E | 12:08 | V: Åja, okei. Da så... En, to, tre, fire, fem [begge ler forsiktig]. Der har vi en femkant. H: Ja! [ler litt og smiler] | Tegner videre i PV i Scratch | |
| K S E K B S R | | V: Ok, en, to, tre, fire, fem. Ok, da skal femkanten vår se sånn ut da. H: Ja.. | | |
| K S K B S R | | V: Men hvordan.. Vi må jo finne ut hvor mange grader den skal snu seg. Hvordan finner vi ut det? Vi kan jo ikke bare gjette. H: Jo, for hvis du tar på ... [peker i PV] | V styrer PC | |
| K B S R B S R | | V: Nei, vent.. Nå bare prøver jeg noe. Emm ... Først så skal den... H: Rotere? | V trykker frem og tilbake i Scratch V setter inn [vent # grader] | |
| K B S R | | | | |

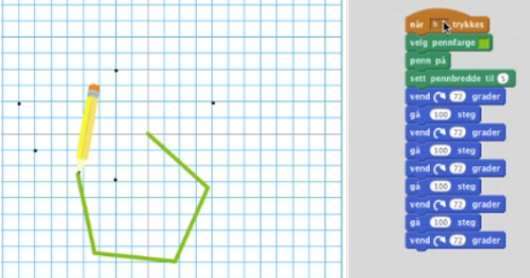
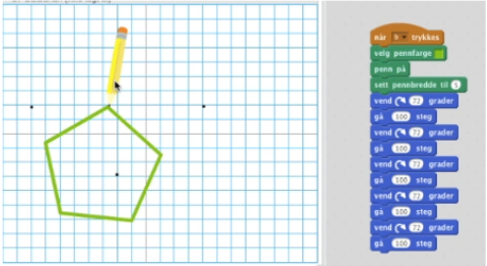
Vedlegg IX-b

| | | | |
|--|---|--|---|
| <p>K B S R R K B S R R K B S R R K B S R E K B S R E K B S R R E K B S R R K B S R R K B S R R S E K B S R R K B E S K B S R R K B S R R K B S R R K B S R R</p> | <p>13:33 V: Vent litt. Hvis vi bare glemmer de punktene nå. ---(setter inn blokk, legger inn pennfunksjon)--- V: Emm ... jeg må bare se hva som skjer der ---(tester skript)--- V: ok, nå må den gå.. H: Jah, nå rettet den seg opp. V: åh, ja! Og så må den gå ... vi sier 100 steg. Ok, hvis vi tar på slett nå, og så velge retning ... nei, årh ... den skal der ja.. en, to, tre b (trykker «b») ---(tester skript)--- [begge ler litt] V: okei, da begynner den nedover der tydeligvis ... H: Og hvis den går rett nå, må den vende igjen V: og så ta på slett [nesten hvisker] V: og så.. H: Såå .. må den gå (H setter inn klosser) V: Jeg er litt usikker på hva som skjer nå, fordi vi har ikke ... 14:42 ---(tester skript)--- [begge ler] V: Jah, mhm .. Ja, men da går den sånn da [viser med musepeker på PC] H: Jah! V: Okei, så da må vi .. H: Det er i hvert fall en begynnelse. V: Da begynner vi igjen her oppe (flytter pennfigur) V: Okei, ee.. så må den vende igjen ... 15:00 (setter inn blokk)</p> | <p>V skriver inn [vend 45 grader]. Setter på pennfunksjon. Pennen vender 45 grader i PV H legger inn [gå 100 steg] Trykker på re-start symbolet (tar bort alle streker), og tester skript. Får en strek V legger inn [vend 45 grader]. Flytter blyanten tilbake på prikken helt til venstre i PV. V legger inn [gå 100 steg] V re-starter (altså tar bort alle streker igjen) V flytter blyant til øverste prikk i PV</p> |   <p>(14:02)</p>  <p>(14:42)</p> |
| <p>K B S R R K B S R R S K K B S R R K B S R R S K B S</p> | <p>15:30 V: Vi må ha fire, nei fem sånne der greier. H: Og så 100 skritt ... V: Hva skjer nå da? Ee ... vi må bare teste H: b ... ---(tester skript)--- [V ler forsiktig og klør seg i hodet] V: Ja? H: Det er bedre enn ingenting. Men det ligner litt på den der du tegnet der [Peker i PV på skjermen] [ler litt] V: Ok, så lenge det er femkanter så går det fint. (H flytter på blyanten) V: Da fortsetter vi bare ... (H setter inn kloss) V: vent ... hvis vi da tar ... [snakker lavt] H: ee.. «b» 15:52 ---(tester skript)--- Stud.: Den går sånn beint ned når den treffer kanten. Så det er derfor den ser sånn ut.</p> | <p>V setter inn [vend 45 grader] V legger inn [gå 100 steg], [vend 45 grader]. Trykker re-start og tester skript V flytter blyanten til prikken helt til høyre V setter inn [gå 100 steg] V trykker [slett] og «b»</p> |  <p>(15:30)</p>  <p>(15:52)</p> |
| <p>S R K B S R R E K S R</p> | <p>16:00 V: Okei, hvis vi tar slett H: og så hvis vi tar den rett opp nå, og så til siden. Da har vi laget et hus. Da kan vi krysse av huset [begge ler] V: nei ... [litt lavt]</p> | <p>V trykker [slett]</p> | |

Vedlegg IX-c

| | | | |
|--|--|---|--|
| <p>K B S R R</p> <p>K B S R E</p> <p>K B S R R</p> <p>K B S R R</p> <p>K B S R R</p> <p>K B S R</p> <p>K B S R</p> <p>K B S R R</p> <p>K B S R R</p> <p>K B S R R</p> <p>K B S R R</p> <p>K B S R R</p> <p>K B S R R</p> <p>K B S R R</p> <p>K B S R R</p> <p>K B S R R</p> <p>K B S R R</p> <p>K B S R R</p> <p>K B S R R</p> | <p>16:14 [v klør seg i hodet] V: Okei, vent litt. Hvis vi tar pegeren, og så «b» ---(tester skript)--- V: aa..? Ja, men det blir jo ikke en femkant? H: En, to, tre, fire, fem [peker på skjermen mens hun teller] V: Da må den rett opp der da [peker på skjermen] H: Det blir en sekskant ... V: En, to, tre, fire ... hvis vi får den der til å gå opp der. Hvordan gjør vi det? [peker med musepeker på skjerm] H: Hvordan fikk vi den til å gå skrå igjen? V: diagonalt ...? [snakker lavt] (V flytter og trykker i Scratch) V: Eee ... nå må den snu seg ... [viser med armene] mer enn 45 grader ... H: da må den ... 90? V: hvordan ... 16:57 ---[v trykker «b» - tester skript]--- V: Hvordan går den opp der? [peker på skjermen] Den må snu seg fff... [sier lavt] det går ikke an å finne det ut ... 17:11 ---[liten pause med stillhet]--- V: Mmm... hvis vi prøver en til da. Vend ... (setter inn kloss) V: Litt ... det er litt mindre enn 90 ... ---[liten pause med stillhet]--- V: hvis vi ... [lavt] H: 85? 80? 85? V: 70 ... (skriver inn i blokk) Nå bare tipper jeg ... H: Og så trykker vi på (skal til å trykke på tast)</p> | <p>V trykker «b»</p> <p>V peker på skjerm</p> <p>V peker med musepeker på skjerm på begge endene av den grønne streken i PV. Trykker på [slett]. Flytter på blyantfigur i PV V gjør bevegelser med armene for å illustrere.</p> <p>V tester skript. Fortsatt samme skript og samme som skjer i PV</p> <p>V legger inn [vend # grader]</p> <p>V skriver inn [vend #-75 grader]</p> |  <p>(16:14)</p> |
| <p>K B S R R</p> <p>K B S R R</p> <p>K B S R R</p> <p>K B S R R</p> <p>B S</p> <p>K B S R R</p> <p>E</p> <p>E S</p> <p>K B S R R</p> | <p>17:50 V: Nei, vent. Du kan ikke trykke før vi har tatt på ... (tar musepeker bort på [gå # steg]) H: Åja ... ja. 200 (V legger inn blokk) H: [peker på skjerm] De må jo være lengre enn ... V: Åja ... ja, vi bare ... (H skriver nytt tall #-1000 i blokk) V: 1000? H: åja ... [begge ler] H: Okei, skal du få æren? [ler] 18:00 ---(tester skript)--- [begge ler] V: Hva skjedde nå? Hvorfor snudde den seg så mange ganger? H: Eee... skal vi se (tar over PC). Hvis vi tar vekk den. Slett, og så finner vi pennen. Hvor er pennen? ---(teknisk - finner penn)---</p> | <p>V legger inn [gå 100 steg]</p> <p>H endrer til [gå 1000 steg]</p> <p>H styrer PC. Drar bort siste kloss [gå 1000 steg] fra blokk. Trykker på [slett]. stud. Hjelper de med å få frem pennfigur.</p> |  <p>(18:00)</p> |
| <p>S R</p> <p>K B S R R</p> <p>K B S R R</p> | <p>18:43 V: Åå.. jeg skjønner ikke. [ser bort på H] ---(tester skript)--- H: Åå.. det var et eller annet der. Hvis vi tar det ned med 500 skritt. V: Nei, men den snudde seg så mange ganger. [peker med musepeker i PV]. Han gikk sånn, sånn og sånn. Han skulle liksom gå rett fram. ... Hh...[puster ut] dette her går ikke ... [puster ut]. Ok, jeg tror vi har gjort feil...</p> | <p>V ser ned på arket sitt hvor hun tidligere har tegnet. Tester skript, får samme som før de la til [gå 1000 steg]. V styrer PC.</p> |  <p>(18:43)</p> |

Vedlegg IX-d

| | | | |
|---|---|--|---|
| <p>K B S R R</p> <p>K B S R R</p> <p>K B S R R</p> <p>K B S R E E</p> <p>E</p> <p>E</p> <p>K B S R E E</p> <p>K B S R R</p> <p>K B S R R</p> <p>K B S R R</p> <p>K B S R R</p> <p>E</p> <p>K B S</p> <p>K B S R R</p> | <p>H: Jaaa ... [drar på ordet] (peker på skjerm). Men hvis vi prøver å hakke den ned litt, så ser vi hvor den kommer.</p> <p>V: det er sikkert ikke riktig det heller. For det er jo ... [tegner på ark] ---(stille – tegner, trykker)---</p> <p>H: [hvisker stille] 72 ... 72 ...</p> <p>V: og nå ... [ler litt] hvis vi ... (taster inn tall i blokk)</p> <p>H: Skal vi hente frem pennen igjen eller?</p> <p>V: Hvorfor er pennen vekke?</p> <p>H: Jeg trykket på den der ...</p> <p>---(teknisk – finner frem penn)---</p> <p>V: ok, plis! Jeg vil bare se hva som skjer nå ... (trykker «b») Nå!</p> <p>---(tester skript)---</p> <p>H: hh... [puster ut med lyd]</p> <p>V: hhh ... [puster inn – tar seg i ansiktet] Det var nesten!</p> <p>H: [peker på skjerm] Og så tar du den bare, og så setter den til 100 (V tegner noe på arket)</p> <p>V: [hvisker noe]</p> <p>H: [sier lavt] Kanskje ta en til... (trykker på tallene i [vend ...])</p> <p>H: Nei, det skal være 72</p> <p>Stud.: Oi ...</p> <p>V: åå ... [tar seg i ansiktet]</p> <p>H: [sier lavt] to, tre, fire, fem ...</p> <p>V: Nei, men hvorfor? [klør seg i hodet]</p> <p>H: [peker i PV på skjerm] Vi trenger noen skritt der, for da bare vender den. Men ikke akkurat 500 da ...</p> | <p>V tegner på ark. H trykker litt i Scratch.</p> <p>V skriver inn # = 72 i alle [vend # grader]</p> <p>V tester skript</p> <p>V beveger musepeker over figuren i PV mens hun hvisker noe for seg selv.</p> <p>Trykker på tallene i [vend 72 grader], men ender ikke tallene. Stud. Kommer bort og står bak elevene.</p> <p>V flytter kloss som de tidligere har tatt bort [gå 500 steg] og legger den til</p> |  <p>(20:53)</p> |
| <p>K B S R R</p> <p>E</p> <p>K B S R E E</p> <p>K B S R E E</p> <p>K B S E E</p> <p>E</p> <p>E</p> <p>K B S R R</p> <p>K B S R R</p> <p>K B S R R</p> <p>K</p> <p>K</p> | <p>V: Ja, men da tror jeg det blir en sekskant ... (legger til kloss)</p> <p>H: Og så sletter vi strekene</p> <p>V: Ja ... (trykker [slett])</p> <p>---(tester skript)---</p> <p>[begge to trekker pusten. V tar hendene til ansiktet. Begge smiler, lener seg bakover og tar hendene halvveis opp i været]</p> <p>H: jj... .. er ikke det et hus og?</p> <p>V: hh...nei ... en, to, tre, fire, fem ... det blir fem! (samtidig som H snakker)</p> <p>Stud.: [nikker]</p> <p>V: yes! [snur seg til stud. Og gir high five til stud og H] Endelig!</p> <p>[begge smiler og ler]</p> <p>Stud.: Knallbra! Supert!</p> <p>V: Det hjalp å ...</p> <p>H: e det et hus og? (samtidig som V snakker)</p> <p>V: ... ta 365 delt på 5</p> <p>Stud: mhm ...</p> <p>H: Er det et hus også? (ser opp på stud.)</p> <p>V: nei, det er ikke et hus.</p> <p>Stud.: det kan være et hus, men om dere har mer tid, kan dere kan gjerne lage et litt mer detaljert hus også.</p> <p>V: mhm ... ja ...</p> <p>H: jah ...</p> | <p>blokken igjen. Endrer kloss til #=100</p> |  <p>(21:42)</p> |